

ระบบติดตามและรายงานสภาพแวดล้อมแบบบูรณาการด้วย LORA และ GPS  
สำหรับการขนส่งทางรถไฟ  
LORA AND GPS INTEGRATED ENVIRONMENTAL MONITORING AND REPORTING  
SYSTEM FOR RAIL TRANSPORTATION



โดย

นายรัตน์พล

ประศรีพัฒน์

นายรุ่งเรือง

จันทร์เรือง

นายวงศธร

มาตย์หงษา

ปฏิญานិพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมโทรคมนาคม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2566

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ระบบติดตามและรายงานสภาพแวดล้อมแบบบูรณาการด้วย LoRa และ GPS  
สำหรับการขนส่งทางรถไฟ

LORA AND GPS INTEGRATED ENVIRONMENTAL MONITORING AND  
REPORTING SYSTEM FOR RAIL TRANSPORTATION

โดย

นายรัตน์พล	ประศรีพัฒน์	63010818
นายรุ่งเรือง	จันทร์เรือง	63010826
นายวงศธร	มาตย์หงษา	63010835

อาจารย์ที่ปรึกษา

ผศ.ดร.กฤษณ์ วรจุจระ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2566

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2566

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบติดตามและรายงานสภาพแวดล้อมแบบบูรณาการด้วย LORA และ GPS สำหรับการขนส่งทางรถไฟ

LORA AND GPS INTEGRATED ENVIRONMENTAL MONITORING AND REPORTING  
SYSTEM FOR RAIL TRANSPORTATION

ผู้จัดทำ

1. นายรัตน์พล ประศรีพัฒน์ 63010818
2. นายรุ่งเรือง จันทร์เรือง 63010826
3. นายวงศธร มาตย์หงษา 63010835



..... อาจารย์ที่ปรึกษา

(ผศ.ดร.กฤษณ์ วงจรูจีระ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## กิตติกรรมประกาศ

โครงการเรื่อง "ระบบติดตามและรายงานสภาพแวดล้อมแบบบูรณาการด้วย LORA และ GPS สำหรับการขนส่งทางรถไฟ" สำเร็จลุล่วงไปได้ด้วยดี ด้วยความกรุณาและความช่วยเหลือจากอาจารย์ที่ปรึกษาและทุกท่านที่เกี่ยวข้อง คณะผู้จัดทำขอขอบพระคุณ ผศ.ดร.กฤษณ์ วงรุจิระ ที่ได้ให้คำแนะนำ ชี้แนะแนวคิด และแนวทางในการแก้ปัญหาในการทำงาน รวมถึงสอดแทรกทักษะและประสบการณ์อันมีค่าให้แก่คณะผู้จัดทำ

คณะผู้จัดทำขอขอบพระคุณคณาจารย์และเจ้าหน้าที่ประจำภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้อบรมสั่งสอน ประสิทธิ์ประสาทวิชา ความรู้ และประสบการณ์อันล้ำค่าแก่คณะผู้จัดทำ

คณะผู้จัดทำขอกราบขอบพระคุณบิดามารดาและครอบครัว ที่ให้ความรัก ความห่วงใย และเป็นกำลังใจที่สำคัญเสมอมา รวมถึงสนับสนุนให้ออกาสทางด้านการศึกษามีค่าอย่างยิ่งแก่คณะผู้จัดทำ

นายรัตน์พล ประศรีพัฒน์  
นายรุ่งเรือง จันทรเรือง  
นายวงศธร มาตย์หงษา

ผู้จัดทำ

ระบบติดตามและรายงานสภาพแวดล้อมแบบบูรณาการด้วย  
LORA และ GPS สำหรับการขนส่งทางรถไฟ  
LORA AND GPS INTEGRATED ENVIRONMENTAL  
MONITORING AND REPORTING SYSTEM FOR  
RAIL TRANSPORTATION

โดย นายรัตนพล ประศรีพัฒน์ 63010818  
นายรุ่งเรือง จันทร์เรือง 63010826  
นายวงศธร มาตย์หงษา 63010835

อาจารย์ที่ปรึกษา ผศ.ดร.กฤษณ์ วงรุจิระ

### บทคัดย่อ

ในปัจจุบันอุตสาหกรรมขนส่งพบกับหลายปัญหาที่ต้องการการแก้ไขและควบคุมอย่างมีประสิทธิภาพ หนึ่งในปัญหาหลักคือการติดตามและควบคุมคุณภาพการขนส่งสินค้าและสภาพแวดล้อมในระหว่างการขนส่งที่ยากเนื่องจากข้อมูลที่สำคัญสำหรับการติดตามนี้บ่อยครั้งหายไปหรือไม่สมบูรณ์ ซึ่งส่งผลให้การควบคุมคุณภาพและความปลอดภัยของสินค้าที่ขนส่งกับความเป็นไปได้ของความเสียหายต่าง ๆ เช่น การสูญเสียสินค้าหรือการส่งมอบที่ไม่ตรงตามระยะเวลาที่กำหนด

โครงการนี้ถูกพัฒนาขึ้นเพื่อแก้ไขปัญหาดังกล่าวในอุตสาหกรรมขนส่ง โดยโครงการนี้ใช้เทคโนโลยีไร้สาย LoRa เพื่อรวบรวมข้อมูลสภาพแวดล้อม เช่น อุณหภูมิและความชื้น จากอุปกรณ์ End Devices และ GPS เพื่อตำแหน่งปัจจุบันของสินค้าและอุปกรณ์ ข้อมูลนี้จากนั้นถูกส่งไปยังเซิร์ฟเวอร์ฐานข้อมูลผ่านเครือข่าย LTE เพื่อการประมวลผลและรายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงการนี้มีการพัฒนาเว็บไซต์ที่ช่วยให้ผู้ใช้สามารถติดตามสถานะและสภาพแวดล้อมของการขนส่งได้แบบเรียลไทม์ โดยใช้กราฟและแผนที่แบบเรียลไทม์ เพื่อแสดงข้อมูลอย่างชัดเจนและอัปเดตข้อมูลอัตโนมัติ ผู้ใช้สามารถรับข้อมูลในขณะเดียวกันขณะที่การขนส่งกำลังเกิดขึ้น ทำให้มีการควบคุมและปรับปรุงการขนส่งได้อย่างมีประสิทธิภาพและแม่นยำมากขึ้น

## ABSTRACT

At present, the transportation industry is facing several issues that require efficient solutions and control. One of the primary problems is tracking and controlling the quality of goods transportation and environmental conditions during transit, which is often challenging due to missing or incomplete essential data. This results in the quality and safety control of transported goods being susceptible to various risks, such as loss of goods or delivery delays.

This project has been developed to address these challenges in the transportation industry. The project utilizes LoRa wireless technology to collect environmental data, such as temperature and humidity, from End Devices, and GPS to determine the current location of goods and devices. This data is then transmitted to a database server through the LTE network for processing and reporting.

The project includes the development of a web interface that allows users to track the status and environmental conditions of transportation in real-time. It uses graphs and real-time maps to display data clearly and updates the information automatically. Users can receive information while the transportation is ongoing, enabling more efficient and precise control and optimization of transportation.

## สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	III
สารบัญรูป	V
สารบัญตาราง	VI
<b>บทที่ 1</b>	
<b>บทนำ</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงการ	2
<b>บทที่ 2</b>	
<b>ทฤษฎีและหลักการที่เกี่ยวข้อง</b>	<b>3</b>
2.1 ภาษา C++ สำหรับ ARDUINO	3
2.2 การสื่อสารแบบ I2C	4
2.3 หลักการทำงานของเซนเซอร์วัดอุณหภูมิและความชื้น	6
2.3.1 เซ็นเซอร์วัดอุณหภูมิ (TEMPERATURE SENSOR)	6
2.3.2 เซ็นเซอร์วัดความชื้น (HUMIDITY SENSOR)	7
2.4 LORA	8
2.4.1 เทคโนโลยี LORA ทำงานอย่างไร	9
2.4.2 คุณสมบัติหลักและประโยชน์ของเทคโนโลยี LORA	9
2.4.3 การมอดูเลต LORA	10
2.5 GPS	11
2.6 LTE CAT-1	12
2.7 REST API	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.8 FLASK (WEB FRAMEWORK)	14
2.9 DATABASE	14
2.9.1 MARIADB	14
2.10 WEB COOKIES	15
2.11 โปรแกรมที่ใช้ในโครงการงาน	17
2.11.1 VISUAL STUDIO CODE	17
2.11.2 PLATFORMIO บน VSCODE	18
2.11.3 TERMIUS	19
2.11.4 SDR# (SDRSHARP)	20
2.11.5 POSTMAN	21
2.11.6 PHPMYADMIN	22
2.12 ภาษาคอมพิวเตอร์ที่ใช้ในโครงการงาน	23
2.12.1 PYTHON	23
2.12.2 SQL	24
2.12.3 HTML	25
2.12.4 CSS	26
2.12.5 JS	27
2.12.6 C++	28
2.12.7 JSON	29
2.13 เครื่องมือที่ใช้ในการทดลอง	30
2.13.1 บอร์ด ESP32 NODEMCU ESP-WROOM-32	30
2.13.2 บอร์ด ESP32 LILYGO T-SIM7670E	32
2.13.3 โมดูล LORA SX1276 ย่านความถี่ 900-931 เมกะเฮิรตซ์	33
2.13.4 เซนเซอร์วัดอุณหภูมิและความชื้น AHT10 I2C	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.13.5 GPS- MODULE GY-NEO-6MV2 UBLOX พร้อมสายอากาศอุปกรณ์เพิ่มเติม	36
2.13.6 RASPBERRY PI 3 MODEL B	37
2.13.7 THE SPARKFUN RTK SURVEYOR	39
2.12.8 RTL-SDR DONGLE	40
<b>บทที่ 3 การออกแบบและการจัดทำปริญญานิพนธ์</b>	<b>41</b>
3.1 การออกแบบภาพรวมโครงการ	41
3.2 การออกแบบในส่วนของอุปกรณ์ทางฝั่ง END DEVICES	42
3.2.1 การออกแบบและสร้างแบบวงจรในส่วนของอุปกรณ์ทางฝั่ง END DEVICES	42
3.2.2 การออกแบบโปรแกรมสำหรับส่งค่า และเก็บค่าจากเซนเซอร์ด้วยโมดูล LORA ในย่านความถี่ 923 เมกะเฮิรตซ์	46
3.3 การออกแบบในส่วนของอุปกรณ์ทางฝั่ง GATEWAY	48
3.3.1 การออกแบบและสร้างแบบวงจรในส่วนของอุปกรณ์ทางฝั่ง GATEWAY	48
3.3.2 การออกแบบโปรแกรมสำหรับรับข้อมูลจาก END DEVICES เก็บข้อมูลจาก GPS MODULE และส่งข้อมูลขึ้น DATABASE	51
3.4 การออกแบบการทำงานของฝั่ง SERVER	57
3.4.1 การออกแบบและสร้างเส้นทางการเชื่อมต่อระหว่างโลกภายนอกกับ SERVER	57
3.4.2 การออกแบบและสร้าง DATABASE สำหรับการเก็บข้อมูลต่างๆ	57
3.4.3 การออกแบบและสร้าง API เพื่อไว้ใช้จัดการกับ DATABASE และ GATEWAY	59
3.5 การออกแบบในส่วนของหน้าเว็บไซต์	77

## สารบัญ (ต่อ)

	หน้า
3.5.1	77
3.5.2	77
3.5.3	78
3.6	96
3.6.1	96
3.6.2	97
3.6.3	97
3.6.4	97
3.6.5	97
3.6.6	97
3.6.7	97
3.7	98
3.7.1	98
3.7.2	104
3.7.3	108
3.7.4	108
<b>บทที่ 4</b>	<b>109</b>
4.1	109
4.1.1	109
4.1.2	114
4.1.3	118
4.1.4	118
4.1.5	119

## สารบัญ (ต่อ)

	หน้า
4.2 การทดสอบอุปกรณ์ทางฝั่ง GATEWAY	120
4.2.1 การทดสอบรับคำสั่งสัญญาณเพื่อดูความแม่นยำของ GPS ที่เชื่อมต่อกับ GATEWAY	120
4.2.2 การทดสอบการรับข้อมูลที่ส่งมาจากฝั่ง END DEVICE	124
4.2.3 การทดสอบการส่งข้อมูลทั้งหมดที่รวบรวมได้ขึ้น DATABASE	125
4.3 การทดสอบการใช้งานเว็บไซต์	125
4.4 การทดสอบการใช้งานจริง	133
4.4.1 การทดสอบในใช้งานจริงในการเดินทางโดยสารสาธารณะ ในเส้นทางทางที่ได้กำหนดไว้	134
4.4.2 การทดสอบในใช้งานจริงในการเดินทางโดยรถไฟในเส้นทางทางที่ได้กำหนดไว้	136
<b>บทที่ 5</b>	
<b>ผลการทดลอง</b>	138
5.1 สรุปผล	138
5.1.1 สรุปการทดลองในส่วนของอุปกรณ์ทางฝั่ง END DEVICES	138
5.1.2 สรุปการทดลองในส่วนของอุปกรณ์ทางฝั่ง GATEWAY	140
5.1.3 สรุปการทดลองในการทดสอบการใช้งานเว็บไซต์	141
5.1.4 สรุปการทดลองในการทดสอบการใช้งานจริง	141
5.2 ข้อเสนอแนะ	141
<b>บรรณานุกรม</b>	142
<b>ภาคผนวก ก</b>	
คำสั่งสำหรับการทำงานของฝั่ง END DEVICES	146
<b>ภาคผนวก ข</b>	
คำสั่งสำหรับการทำงานของฝั่ง GATEWAY	151
<b>ภาคผนวก ค</b>	
คำสั่งสำหรับการทำงานของฝั่ง API	169

## สารบัญ (ต่อ)

	หน้า	
ภาคผนวก ง	คำสั่งสำหรับการทำงานของหน้าเว็บไซต์ LOGIN	181
ภาคผนวก จ	คำสั่ง CSS สำหรับการทำงานบนหน้าเว็บไซต์ทั้งหมด	184
ภาคผนวก ฉ	คำสั่งการทำงานสำหรับหน้าเว็บไซต์ DASHBOARD	189
ภาคผนวก ช	คำสั่งการทำงานสำหรับหน้าเว็บไซต์ SETTINGS	212
ภาคผนวก ซ	คำสั่งการทำงานสำหรับหน้าเว็บไซต์ PAST RECORDS	219
ภาคผนวก ฌ	คำสั่งการทำงานสำหรับหน้าเว็บไซต์ ALERT	241
ภาคผนวก ญ	คำสั่งการทำงานสำหรับหน้าเว็บไซต์ ADMIN	253

## สารบัญรูป

รูปที่	หน้า
2.1 การส่งสัญญาณเริ่มต้นส่งข้อมูล (START)	5
2.2 การรับ-ส่งข้อมูล	5
2.3 การสิ้นสุดการส่งข้อมูล (STOP)	6
2.4 LORA	8
2.5 การทำงานของ REST API	13
2.6 MARIADB	14
2.7 WEB COKIES	15
2.8 VISUAL STUDIO CODE	17
2.9 PLATFORMIO บน VSCODE (1)	18
2.10 PLATFORMIO บน VSCODE (2)	18
2.11 TERMIUS	19
2.12 ภาพหน้าจอแสดงโปรแกรม SDR#	20
2.13 POSTMAN	21
2.14 PHPMYADMIN	22
2.15 C++	28
2.16 ตัวอย่าง JSON ในรูปแบบของ OBJECT	29
2.17 ตัวอย่าง JSON ในรูปแบบของ ARRAY	29
2.18 บอร์ด ESP 32	30
2.19 บอร์ด ESP32 LILYGO T-SIM7670E	32
2.20 โมดูล LORA SX1276 ย่านความถี่ 900-931 เมกะเฮิรตซ์	34
2.21 เซนเซอร์วัดอุณหภูมิและความชื้น AHT10 I2C	36
2.22 GPS- MODULE GY-NEO-6MV2 UBLOX	37
2.23 RASPBERRY PI 3 MODEL B	38
2.24 THE SPARKFUN RTK SURVEYOR	39

## สารบัญรูป

รูปที่	หน้า
2.25 RTL-SDR DONGLE	40
3.1 บล็อกไดอะแกรมของระบบติดตามและรายงานสภาพแวดล้อมแบบบูรณาการด้วย LORA และ GPS สำหรับการขนส่ง	41
3.2 แบบจำลองวงจร END DEVICES	42
3.3 บอร์ด ESP32 ร่วมกับ LORA MODULE ในย่านความถี่ 923 เมกะเฮิร์ตซ์	44
3.4 ขบวนการเชื่อมต่อเซนเซอร์ AHT10 I2C และ DIODE LED ทำการรับข้อมูลจากแบบจำลอง	45
3.5 กล่องอุปกรณ์ END DEVICE หลังประกอบอุปกรณ์ทั้งหมดลงกล่อง	45
3.6 แผนผังการทำงานของโปรแกรมสำหรับรับ และเก็บค่าจากเซนเซอร์ด้วยโมดูล LORA ย่านความถี่ 923 เมกะเฮิร์ตซ์ กับบอร์ด ESP32 พร้อมทั้งเก็บข้อมูลในรูปแบบ JSON	47
3.7 แบบจำลองวงจร GATEWAY	48
3.8 บอร์ด ESP32 LILYGO ร่วมกับ LORA MODULE และ GPS MODULE	50
3.9 กล่องอุปกรณ์ GATEWAY หลังประกอบอุปกรณ์ทั้งหมดลงกล่อง	50
3.10 แผนผังแสดงขั้นตอนการทำงานของโปรแกรมสำหรับรับค่าและเก็บค่าจากเซนเซอร์ด้วยโมดูล LORA ในส่วนของ SETUP	53
3.11 แผนผังแสดงขั้นตอนการทำงานของโปรแกรมสำหรับรับค่า และเก็บค่าจากเซนเซอร์ด้วยโมดูล LORA ในส่วนของ LOOP	56
3.12 เส้นทางเชื่อมต่อระหว่างโลกภายนอกกับ SERVER และ ตาราง PORT FORWARDING บน ROUTER	57
3.13 การใช้ CLI (COMMAND LINE INTERFACE) เพื่อเข้าถึงข้อมูลบน DATABASE	58
3.14 การใช้ PHPMYADMIN เพื่อเข้าถึงข้อมูลบน DATABASE	58
3.15 การบันทึกข้อมูลของแต่ละ NODE	60

## สารบัญรูป

รูปที่	หน้า
3.16 การดึงข้อมูลล่าสุด	61
3.17 การแสดงค่าการกำหนดค่าปัจจุบัน	62
3.18 การส่งค่าการกำหนดค่าใหม่	63
3.19 การดึงข้อมูลที่บันทึกในวันนี้	64
3.20 การดึงข้อมูลทั้งหมดที่บันทึกไว้	65
3.21 การเปิด/ปิดใช้งาน NODE	66
3.22 QR CODE ของแต่ละ NODE	66
3.23 การแสดงรายชื่อ NODE ที่ใช้งานอยู่	67
3.24 การแสดงข้อมูลตามช่วงวันที่ที่กำหนดเอง	68
3.25 การแก้ไขเงื่อนไขการแจ้งเตือน	69
3.26 การแสดงเงื่อนไขการแจ้งเตือนปัจจุบัน	70
3.27 การเพิ่มอีเมลสำหรับรับการแจ้งเตือน	71
3.28 การลบอีเมลสำหรับรับการแจ้งเตือน	72
3.29 การแสดงรายชื่ออีเมลและความถี่ในการรับการแจ้งเตือนที่กำหนดไว้	73
3.30 การเพิ่มผู้ใช้งานเข้าไปในระบบ	74
3.31 การยืนยันตัวตนผู้ใช้งาน	75
3.32 การดึงผู้ใช้งานทั้งหมด	76
3.33 การใช้งานโปรแกรม FIGMA สำหรับการออกแบบเว็บไซต์	78
3.34 รูปแสดงหน้าต่างสำหรับการเข้าสู่เว็บไซต์	79
3.35 รูปแสดงหน้าต่างสำหรับการจัดการผู้ใช้ของแอดมิน	79
3.36 รูปแสดงการทำงานของเมนู HOME	80
3.37 รูปแสดงการทำงานของเมนู PAST RECORDS	80
3.38 รูปแสดงการทำงานของเมนู SETTING	81
3.39 รูปแสดงหน้าต่างสำหรับการปรับเงื่อนไขการแจ้งเตือน	81

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่	หน้า	
3.40	รูปแสดงหน้าต่างสำหรับการเพิ่มอีเมลสำหรับการรับการแจ้งเตือน	82
3.41	รูปแสดงหน้าต่างสำหรับการเลือกตำแหน่งปลายทางที่ต้องการ	82
3.42	รูปแสดงปุ่ม LOG OUT	83
3.43	รูปแสดงหน้า DASHBOARD	83
3.44	แผนผังแสดงขั้นตอนการทำงานของการอัปเดตค่าในส่วนของ DASHBOARD	84
3.45	คำสั่งการทำงานสำหรับการปรับเปลี่ยน API ENDPOINT	85
3.46	ข้อมูลที่ได้รับมาจาก RESPONSE ของ API	85
3.47	คำสั่งการทำงานสำหรับกรองข้อมูลตามชื่อ NODE หาค่าเฉลี่ย และ อัปเดต DASHBOARD ตาม ELEMENT ID	86
3.48	ข้อมูลค่าเฉลี่ยอุณหภูมิและความชื้นของแต่ละ NODE พร้อมชื่อ ELEMENT ID	86
3.49	แผนผังแสดงขั้นตอนการทำงานของการอัปเดตค่าสถานะของปุ่มเปิดปิด	87
3.50	คำสั่งการทำงานสำหรับการอัปเดตสถานะของปุ่ม ON/OFF	88
3.51	รูปแสดงเว็บไซต์ในส่วนของ GATEWAY REAL-TIME LOG	89
3.52	แผนผังแสดงขั้นตอนการทำงานของการอัปเดตค่าในกล่องข้อความ	90
3.53	คำสั่งการทำงานสำหรับการนำเวลามาเปรียบเทียบกับ UPDATE ค่าลงในกล่องข้อความ	91
3.54	คำสั่งการทำงานสำหรับการนำค่า BATTERY จาก ARRAY ตัวสุดท้ายมา UPDATE ค่าลงในกล่องข้อความ	91
3.55	คำสั่งการทำงานสำหรับการรอกการกดปุ่ม CLEAR LOG เพื่อลบข้อความทั้งหมด	91
3.56	รูปแสดงกล่องข้อมูลกราฟอุณหภูมิของแต่ละ NODE	92

## สารบัญรูป

รูปที่	หน้า	
3.57	รูปแสดงกล่องข้อมูลกราฟความขึ้นของแต่ละ NODE	92
3.58	แผนผังแสดงขั้นตอนการทำงานของกราฟ	93
3.59	คำสั่งสำหรับการดึงค่าจาก SLIDER และ สร้างกราฟ	94
3.60	รูปแสดงกล่องข้อมูลตำแหน่งของ GATEWAY	94
3.61	แผนผังแสดงขั้นตอนการทำงานของโปรแกรมปิดกั้นแผนที่	95
3.62	รูปแสดงกล่องข้อมูลตำแหน่งของ GATEWAY	96
3.63	แผนผังแสดงขั้นตอนการทดสอบของโปรแกรม LORA	99
3.64	แผนผังการออกแบบทดสอบความเร็วในการส่งข้อมูลที่แต่ละ SF	101
3.65	แผนผังแสดงขั้นตอนการทำงานของโปรแกรมสำหรับการทดสอบหาความแม่นยำของเซ็นเซอร์	103
3.66	แผนผังแสดงขั้นตอนการทำงานของกราฟหาตำแหน่ง GPS	105
3.67	แผนผังการออกแบบการทดสอบการรับข้อมูลที่ส่งมาจากฝั่ง END DEVICE	107
3.68	แผนผังการออกแบบการทดสอบการส่งข้อมูลทั้งหมดที่รวบรวมได้ขึ้น DATABASE	108
4.1	กล่อง END DEVICE กับ SDR	109
4.2	BANDWIDTH ที่ได้จากการทดสอบ	110
4.3	FRAME ที่ได้จากการทดสอบ	110
4.4	SPREADING FACTOR (SF) = 7	111
4.5	SPREADING FACTOR (SF) = 8	111
4.6	SPREADING FACTOR (SF) = 9	112
4.7	SPREADING FACTOR (SF) = 10	112
4.8	SPREADING FACTOR (SF) = 11	113
4.9	SPREADING FACTOR (SF) = 11	113
4.10	การทดสอบรับและส่งข้อมูล LOG PACKET	114

## สารบัญรูป

รูปที่	หน้า
4.11 LOG ของแต่ละ PACKET ที่ได้รับเมื่อมีค่า SPREADING FACTOR = 7 (มี CRC)	114
4.12 LOG ของแต่ละ PACKET ที่ได้รับเมื่อมีค่า SPREADING FACTOR = 9 (มี CRC)	115
4.13 LOG ของแต่ละ PACKET ที่ได้รับเมื่อมีค่า SPREADING FACTOR = 12 (มี CRC)	115
4.14 LOG ของแต่ละ PACKET ที่ได้รับเมื่อมีค่า SPREADING FACTOR = 7 (ไม่มี CRC)	116
4.15 LOG ของแต่ละ PACKET ที่ได้รับเมื่อมีค่า SPREADING FACTOR = 9 (ไม่มี CRC)	116
4.16 LOG ของแต่ละ PACKET ที่ได้รับเมื่อมีค่า SPREADING FACTOR = 12 (ไม่มี CRC)	117
4.17 การทดสอบวัดระยะทางที่ไกลที่สุดที่รับและส่งข้อมูลได้	118
4.18 ตำแหน่งพิกัดที่ตึกภาคโทรคมนาคม คณะวิศวกรรมศาสตร์	120
4.19 ตำแหน่งที่ตึกภาคโทรคมนาคม คณะวิศวกรรมศาสตร์	120
4.20 พิกัดตำแหน่งที่หอประชุม คณะวิศวกรรมศาสตร์	121
4.21 ตำแหน่งที่หอประชุม คณะวิศวกรรมศาสตร์	121
4.22 พิกัดตำแหน่งที่ตึก 12 ชั้น คณะวิศวกรรมศาสตร์	122
4.23 ตำแหน่งที่ตึก 12 ชั้น คณะวิศวกรรมศาสตร์	122
4.24 การทดสอบรับค่าสัญญาณ GPS เมื่อทำการเคลื่อนที่	123
4.25 การทดสอบการรับข้อมูลที่ส่งมาจากฝั่ง END DEVICE	124
4.26 หน้าการทดสอบการส่งข้อมูลทั้งหมดที่รวบรวมได้ขึ้น DATABASE	125
4.27 รูปแสดงการ LOGIN ด้วย USERNAME “USER1”	126
4.28 รูปแสดงการ LOGIN ด้วย USERNAME “USER2”	126

## สารบัญรูป

รูปที่	หน้า	
4.29	รูปแสดงหน้าต่าง DASHBOARD ของ USER1	127
4.30	รูปแสดงแสดงหน้าต่าง PAST RECORDS ของ USER1	128
4.31	รูปแสดงหน้าต่างการเปลี่ยนแปลงการตั้งค่าของ END DEVICE สำหรับ USER1	129
4.32	รูปแสดงหน้าต่างการเปลี่ยนการตั้งค่าเงื่อนไขการแจ้งเตือนของ USER1	129
4.33	รูปแสดงหน้าต่าง DASHBOARD ของ USER2	130
4.34	รูปแสดงแสดงหน้าต่าง PAST RECORDS ของ USER2	131
4.35	รูปแสดงหน้าต่างการเปลี่ยนแปลงการตั้งค่าของ END DEVICE สำหรับ USER2	132
4.36	รูปแสดงหน้าต่างการเปลี่ยนการตั้งค่าเงื่อนไขการแจ้งเตือนของ USER2	132
4.37	รูปแสดงการ LOGIN ด้วย USERNAME“ADMIN”	132
4.38	รูปแสดงหน้าต่างการจัดการผู้ใช้ของผู้ดูแล	133
4.39	ข้อมูลใน DATABASE (1)	134
4.40	ข้อมูลอุณหภูมิที่ได้รับ (1)	134
4.41	ข้อมูลความชื้นที่ได้รับ (1)	135
4.42	ข้อมูลพิกัดตำแหน่งที่ได้รับ (1)	135
4.43	ข้อมูลใน DATABASE (2)	136
4.44	ข้อมูลอุณหภูมิที่ได้รับ (2)	136
4.45	ข้อมูลความชื้นที่ได้รับ (2)	137
4.46	ข้อมูลพิกัดตำแหน่งที่ได้รับ (2)	137

## สารบัญตาราง

ตารางที่	หน้า	
3.1	การเชื่อมต่อบอร์ด ESP32 ร่วมกับ เซนเซอร์ AHT10	42
3.2	การเชื่อมต่อบอร์ด ESP32 ร่วมกับ LORA MODULE SX1276	43
3.3	การเชื่อมต่อบอร์ด ESP32 ร่วมกับ DIODE LED	43
3.4	การเชื่อมต่อบอร์ด ESP32 ร่วมกับ BATTERY	43
3.5	การเชื่อมต่อบอร์ด ESP32 LILYGO T-SIM7670E ร่วมกับโมดูล GPS-NEO-6M	49
3.6	การเชื่อมต่อบอร์ด ESP32 LILYGO T-SIM7670E ร่วมกับ LORA MODULE SX1276	49
4.1	SPREADING FACTOR (SF) 7, 9 และ 12 ทั้งสองกรณีที่มีการใช้ CRC และไม่ใช่ CRC	117
4.2	การทดสอบความเร็วในการส่งข้อมูลที่แต่ละ SF	118
4.3	ตารางแสดงผลการทดสอบความแม่นยำของเซ็นเซอร์ในการวัดอุณหภูมิและความชื้นของ END DEVICE	119

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันอุตสาหกรรมขนส่งพบกับหลายปัญหาที่ต้องการการแก้ไขและควบคุมอย่างมีประสิทธิภาพหนึ่งในปัญหาหลักคือการติดตามและควบคุมคุณภาพการขนส่งสินค้าและสภาพแวดล้อมในระหว่างการขนส่งที่ยากเนื่องจากข้อมูลที่เป็นจำเป็นสำหรับการติดตามนี้บ่อยครั้งหายไปหรือไม่สมบูรณ์ซึ่งส่งผลให้การควบคุมคุณภาพและความปลอดภัยของสินค้าที่ขนส่งกับความเป็นไปได้ของความเสียหายต่าง ๆ เช่น การสูญเสียน้ำหนักหรือการส่งมอบที่ไม่ตรงตามระยะเวลาที่กำหนด

โครงการนี้ถูกพัฒนาขึ้นเพื่อแก้ไขปัญหาดังกล่าวในอุตสาหกรรมขนส่งทางรถไฟ โดยโครงการนี้ใช้เทคโนโลยีไร้สาย LoRa เพื่อรวบรวมข้อมูลสภาพแวดล้อม เช่น อุณหภูมิและความชื้นจากอุปกรณ์ End Devices และ GPS เพื่อตำแหน่งปัจจุบันของสินค้าและอุปกรณ์ ข้อมูลนี้จากนั้นถูกส่งไปยังเซิร์ฟเวอร์ฐานข้อมูลผ่านเครือข่าย LTE เพื่อการประมวลผลและรายงาน

โครงการนี้มีการพัฒนาเว็บไซต์ที่ช่วยให้ผู้ใช้สามารถติดตามสถานะและสภาพแวดล้อมของการขนส่งได้แบบเรียลไทม์ โดยใช้กราฟและแผนที่แบบเรียลไทม์ เพื่อแสดงข้อมูลอย่างชัดเจนและอัปเดตข้อมูลอัตโนมัติ ผู้ใช้สามารถรับข้อมูลในขณะเดียวกันขณะที่การขนส่งกำลังเกิดขึ้น ทำให้มีการควบคุมและปรับปรุงการขนส่งได้อย่างมีประสิทธิภาพและแม่นยำมากขึ้น

### 1.2 วัตถุประสงค์

1. เพื่อศึกษาหลักการทำงานและการส่งข้อมูลผ่านระบบ LoRa
2. เพื่อศึกษาวิธีการติดตามตำแหน่งผ่านการใช้งานระบบ GPS
3. เพื่อศึกษาการใช้งาน Database ในการเก็บข้อมูลต่างๆที่ได้จาก Gateway
4. เพื่อศึกษาการทำงานของระบบ Cloud และการทำงานของระบบ Server
5. เพื่อศึกษาโครงสร้าง API เพื่อใช้ในการรับและส่งข้อมูลระหว่าง Web Server บน Cloud กับ Database

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 1.3 ขอบเขตของปริญญาโท

1. สร้างอุปกรณ์ End Devices ที่ใช้วัดอุณหภูมิและความชื้น ที่สามารถส่งข้อมูลผ่านเครือข่าย LoRa
2. สร้างอุปกรณ์ Gateway ที่สามารถหาข้อมูลตำแหน่งด้วย GPS และติดต่อสื่อสารกับ Server ผ่านเครือข่าย LTE ได้
3. สามารถทำให้อุปกรณ์ End Devices สามารถสื่อสารและรับส่งข้อมูลกับทาง Gateway ได้
4. สร้างโครงสร้าง API ที่ไว้จัดการกับข้อมูลที่ถูเก็บไว้บน Database
5. สร้างหน้า Website ที่แสดงผลข้อมูลอุณหภูมิ ความชื้น ตำแหน่งเพื่อใช้วิเคราะห์คุณภาพในการขนส่ง

## บทที่ 2

### ทฤษฎีและหลักการที่เกี่ยวข้อง

#### 2.1 ภาษา C++ สำหรับ ARDUINO

ภาษาซีพลัสพลัส (อังกฤษ: C++) เป็นภาษาโปรแกรมคอมพิวเตอร์เนกประสงค์ มีโครงสร้างภาษาที่มีการจัดชนิดข้อมูลแบบสแตติก (statically typed) และสนับสนุนรูปแบบการเขียนโปรแกรมที่หลากหลาย (multi-paradigm language) ได้แก่ การโปรแกรมเชิงกระบวนการคำสั่ง, การนิยามข้อมูล, การโปรแกรมเชิงวัตถุ, และการโปรแกรมแบบเจเนริก (generic programming) ภาษาซีพลัสพลัสเป็นภาษาโปรแกรมเชิงพาณิชย์ที่นิยมมากภาษาหนึ่งนับตั้งแต่ช่วงทศวรรษ 1990

โครงสร้างของโปรแกรมภาษา C++ ของ Arduino จะประกอบไปด้วย 5 ส่วนคือ

##### 1. Preprocessor directives

Preprocessor directives เป็นส่วนที่เขียนไว้บนสุดของโปรแกรม และจะขึ้นต้นด้วย # ซึ่ง Code ในส่วนตรงนี้ จะทำงานก่อนที่จะ Compile Code ส่วนใหญ่จะไว้ใช้กำหนดค่าคงที่ หรือ Import library

##### 2 ส่วนของการกำหนดค่า (Global declarations)

เป็นส่วนของการประกาศตัวแปรภายนอก Function หรือประกาศ Function ต่างๆ ซึ่งจะเป็นการประกาศแบบ Global หมายความว่าทุกๆ Function จะสามารถเรียกใช้ตัวแปร หรือ Function ที่ประกาศแบบนี้ได้

##### 3 ฟังก์ชัน setup() และ ฟังก์ชัน loop()

ฟังก์ชัน setup() และฟังก์ชัน loop() เป็นคำสั่งที่ Arduino บังคับต้องให้มีในทุกโปรแกรม โดยทั้งสอง Function นี้ Arduino กำหนดให้มีหน้าที่ดังนี้

- Function setup() จะถูกเรียกใช้ทีครั้งแรกที่โปรแกรมเริ่มทำงาน ส่วนใหญ่จะเอาไว้ใช้กำหนดค่าตัวแปรเริ่มต้น หรือเริ่มต้นใช้งาน library ต่างๆ
- Function loop() Function นี้จะถูกเรียกใช้หลังจาก Function setup() และจะทำงานแบบวนลูปไม่รู้จบ หมายความว่าเมื่อ loop() ทำงานเสร็จ ก็จะวนกลับมาทำงาน loop() อีกครั้ง วนแบบนี้ไปเรื่อยๆ

#### 4 การสร้างฟังก์ชันและการใช้งานฟังก์ชัน (Users-defined function)

นอกจาก Function setup() และ loop() แล้วเรายังสามารถสร้าง Function ขึ้นมาใช้งานเองได้ [1]

## 2.2 การสื่อสารแบบ I2C

I<sup>2</sup>C นั้นย่อมาจาก Inter-Integrated Circuit - IIC หรือในรูป I<sup>2</sup>C (อ่านว่า "ไอ-สแคว-ซี" หรือจะอ่านว่า "ไอ-ทู-ซี" หรือ "ไอ-ไอ-ซี") เป็นการสื่อสารแบบ Serial รูปแบบหนึ่ง ถูกคิดค้นขึ้นใน ค.ศ.1982 โดย Philip semiconductor (ปัจจุบันคือ NXP Semiconductors) โดยมีจุดประสงค์เพื่อใช้รับส่งข้อมูลความเร็วต่ำระหว่างอุปกรณ์ มีจุดเด่นคือเชื่อมต่อแบบบัสสามารถเชื่อมต่ออุปกรณ์ได้เป็นจำนวนมาก โดยใช้สายสัญญาณเพียง 2 เส้นเท่านั้น จึงช่วยลดปริมาณของสายสัญญาณ และอุปกรณ์มีขนาดเล็กลง โดยสายสัญญาณ 2 เส้นนั้น คือ

SCL (Serial Clock Line) มีหน้าที่ส่งสัญญาณนาฬิกาเพื่อใช้สำหรับควบคุมการรับส่งข้อมูล ซึ่งความเร็วนี้ตามมาตรฐานคือ 100kHz และมีโหมดอื่นคือ Fast Mode มีความเร็วสูงสุด 400kHz, Hi-Speed Mode มีความเร็วสูงสุด 3.4MHz และ Ultra Fast Mode มีความเร็วสูงสุดที่ 5MHz โดยไมโครคอนโทรลเลอร์แต่ละตัวก็จะมีความเร็วของสัญญาณนาฬิกาต่างกัน (บอร์ด Arduino หรือ STM ที่เราใช้กันทั่ว ๆ ไปนี้ส่วนมากจะรองรับความเร็ว Standard และ Fast Mode เท่านั้น)

SDA (Serial Data Line) เป็นสายที่ใช้รับส่งข้อมูลที่ต้องการจะสื่อสารและเนื่องจากการรับส่งข้อมูลของสายสัญญาณทั้งสองเป็นแบบ Open-Drain จึงจำเป็นต้องต่อ Pull-Up Resistor ที่สายสัญญาณทั้งสอง และต้องอาศัยไฟเลี้ยงด้วย

#### 1 การเริ่มต้นส่งข้อมูล (Start)

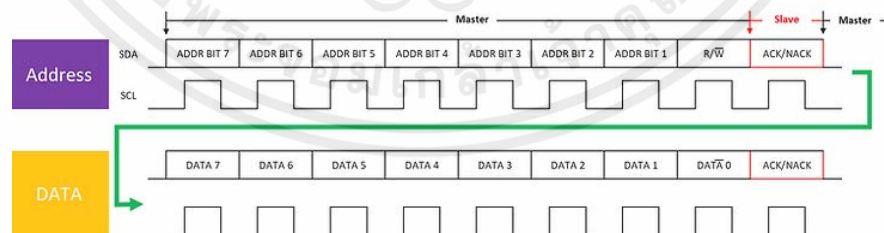
ในสถานะว่าง (Idle) ไม่มีการรับ-ส่งข้อมูล ทั้ง SDA และ SCL มีสถานะเป็น 1 ค้างทั้งคู่ เมื่อเริ่มมีการสื่อสารเกิดขึ้น ขา SDA จะเปลี่ยนลอจิกจาก 1 เป็น 0 ในขณะที่ SCL ยังมีสถานะเป็น 1 ค้างอยู่



รูปที่ 2.1 การส่งสัญญาณเริ่มต้นส่งข้อมูล (Start)

## 2 การรับ-ส่งข้อมูล

ใน 1 เฟรม จะประกอบด้วยข้อมูล 8 บิต และ ACK/NACK อีก 1 บิต รวมการรับ-ส่งข้อมูล 1 เฟรม มี 9 บิต หลักจาก Master ส่งสัญญาณเริ่มต้น (Start) ออกไป เฟรมแรกสุดจะต้องเป็นหมายเลขของ Slave ที่จะสื่อสารในบิตที่ 7 ถึงบิตที่ 1 และทิศทางการรับส่งข้อมูลในบิตที่ 0 (R/^W bit : Read / ^Write bit) หากในรอบนั้น Master ต้องการส่งข้อมูลไป Slave ... ในบิตที่ 0 จะกำหนดเป็น 0 (^W) ... หากในรอบนั้น Master ต้องการรับข้อมูลจาก Slave ... ในบิตที่ 0 จะกำหนดเป็น 1 (R) จากนั้น Master จึงปล่อยลอยสาย SDA .... หาก Slave หมายเลขที่กำหนดมีอยู่จริง ตัว Slave จะเข้าควบคุมสาย SDA แทน โดยดึงสัญญาณสาย SDA ให้เป็น 0 เพื่อบอก Master ว่าพร้อมรับ-ส่งข้อมูลแล้ว เมื่อ Master ส่งสัญญาณที่ขา SCL จาก 0 -> 1 -> 0 ทางฝั่ง Slave จะปล่อยลอยสาย SDA แล้ว Master จะกลับมาควบคุมสาย SDA อีกครั้ง หลังจากนั้นหากมีการรับ-ส่งข้อมูล ก็จะรับ-ส่งกันต่อไป หากไม่มีการรับ-ส่งข้อมูล ก็จะส่งสัญญาณสิ้นสุด (Stop) ออกไป



รูปที่ 2.2 การรับ-ส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3 การสิ้นสุดการส่งข้อมูล (Stop)

หากรับ-ส่งข้อมูลเรียบร้อยแล้ว ฝั่ง Master จะปล่อยสาย SCL ให้ลอยก่อน จากนั้นจึงปล่อยสาย SDA ให้ลอยตาม เพื่อเป็นการส่งสัญญาณ Stop บอกฝั่ง Slave ทุกตัวว่าจบการรับ-ส่งข้อมูลในรอบนั้นแล้ว [2]



รูปที่ 2.3 การสิ้นสุดการส่งข้อมูล (Stop)

## 2.3 หลักการทำงานของเซนเซอร์วัดอุณหภูมิและความชื้น

เซนเซอร์วัดอุณหภูมิและความชื้นเป็นอุปกรณ์ที่ใช้ในการวัดค่าอุณหภูมิและความชื้นในอากาศหรือสภาพแวดล้อม โดยทำงานโดยอาศัยหลักการทางกายภาพที่แตกต่างกัน ดังนี้

### 2.3.1 เซนเซอร์วัดอุณหภูมิ (Temperature sensor)

เซนเซอร์วัดอุณหภูมิมียหลายประเภท แต่ละประเภทมีหลักการทำงานที่แตกต่างกัน แต่หลักการพื้นฐานคือใช้คุณสมบัติทางกายภาพของวัสดุที่ไวต่ออุณหภูมิ เช่น การเปลี่ยนแปลงความยาว การเปลี่ยนแปลงความต้านทานไฟฟ้า การเปลี่ยนแปลงความจุไฟฟ้า เป็นต้น

1.เทอร์มิสเตอร์ (Thermistor) เป็นเซนเซอร์วัดอุณหภูมิชนิดหนึ่งที่มีการเปลี่ยนแปลงความต้านทานไฟฟ้าตามอุณหภูมิเทอร์มิสเตอร์ชนิด NTC (Negative Temperature Coefficient) มีค่าความต้านทานลดลงเมื่ออุณหภูมิเพิ่มขึ้นและชนิด PTC (Positive Temperature Coefficient) มีค่าความต้านทานเพิ่มขึ้นเมื่ออุณหภูมิเพิ่มขึ้น

2.เทอร์มิคอปเปิล (Thermocouple) เป็นเซนเซอร์วัดอุณหภูมิชนิดหนึ่งที่ใช้หลักการของการสร้างกระแสไฟฟ้าขึ้นระหว่างโลหะสองชนิดที่แตกต่างกัน เมื่อโลหะทั้งสองถูกสัมผัสกับความร้อน กระแสไฟฟ้าที่สร้างขึ้นจะแปรผันตามอุณหภูมิ

3. เซนเซอร์วัดอุณหภูมิแบบอินฟราเรด (Infrared Thermometer) เป็นเซนเซอร์วัดอุณหภูมิชนิดหนึ่งที่ใช้หลักการของการแผ่รังสีอินฟราเรดจากวัตถุ เมื่อวัตถุมีอุณหภูมิสูงขึ้น ปริมาณรังสีอินฟราเรดที่แผ่ออกมาก็จะมากขึ้น เซนเซอร์วัดอุณหภูมิแบบอินฟราเรดจะตรวจจับปริมาณรังสีอินฟราเรดที่แผ่ออกมาจากวัตถุ แล้วแปลงเป็นค่าอุณหภูมิ [3]

### 2.3.2 เซนเซอร์วัดความชื้น (Humidity Sensor)

เซนเซอร์วัดความชื้นมีหลายประเภท แต่ละประเภทมีหลักการทำงานที่แตกต่างกัน แต่หลักการพื้นฐานคือใช้คุณสมบัติทางกายภาพของวัสดุที่ไวต่อความชื้น เช่น การเปลี่ยนแปลงความต้านทานไฟฟ้า การเปลี่ยนแปลงความจุไฟฟ้า การเปลี่ยนแปลงการแพร่กระจายของไอน้ำ เป็นต้น

1. เซนเซอร์วัดความชื้นแบบต้านทาน (Resistive Humidity Sensor) เป็นเซนเซอร์วัดความชื้นชนิดหนึ่งที่ใช้หลักการของการเปลี่ยนแปลงความต้านทานไฟฟ้าของวัสดุเมื่อสัมผัสกับความชื้น วัสดุที่ใช้ในการทำเซนเซอร์วัดความชื้นแบบต้านทาน เช่น โพลีเมอร์ คาร์บอน อิเล็กโทรไลต์

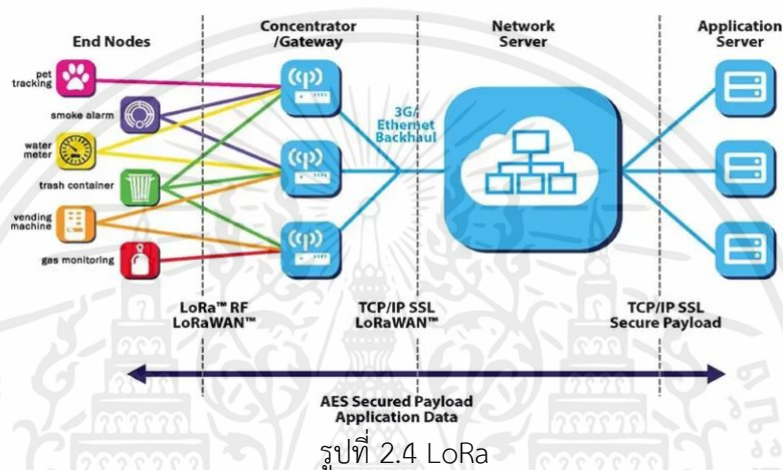
2. เซนเซอร์วัดความชื้นแบบคาปาซิทีฟ (Capacitive Humidity Sensor) เป็นเซนเซอร์วัดความชื้นชนิดหนึ่งที่ใช้หลักการของการเปลี่ยนแปลงความจุไฟฟ้าของวัสดุเมื่อสัมผัสกับความชื้น วัสดุที่ใช้ในการทำเซนเซอร์วัดความชื้นแบบคาปาซิทีฟ เช่น โพลีเมอร์ อิเล็กโทรไลต์

3. เซนเซอร์วัดความชื้นแบบออสซิลเลชัน (Oscillation Humidity Sensor) เป็นเซนเซอร์วัดความชื้นชนิดหนึ่งที่ใช้หลักการของการเปลี่ยนแปลงความถี่การสั่นของวัสดุเมื่อสัมผัสกับความชื้น วัสดุที่ใช้ในการทำเซนเซอร์วัดความชื้นแบบออสซิลเลชัน เช่น โพลีเมอร์

เซนเซอร์วัดอุณหภูมิและความชื้นมักถูกนำมาใช้ร่วมกันในอุปกรณ์เดียวกัน เพื่อวัดค่าอุณหภูมิและความชื้นพร้อมกัน เช่น เครื่องวัดอุณหภูมิและความชื้นในอากาศ เครื่องวัดความชื้นในดิน เครื่องวัดความชื้นในอาหาร เป็นต้น [4]

## 2.4 LoRa

LoRa ย่อมาจาก Long Range Radio และมีเป้าหมายหลักสำหรับ Internet of Things (IoT) และเครือข่าย M2M. เป็นวิธีการมอดูเลตแบบไร้สายแบบใหม่ที่ออกแบบมาอย่างแม่นยำสำหรับการเชื่อมต่อระยะไกลและการสื่อสารที่ใช้พลังงานต่ำ. เทคโนโลยีนี้จะช่วยให้เครือข่ายหลายผู้เช่าหรือเครือข่ายสาธารณะเชื่อมต่อแอปพลิเคชันจำนวนมากที่ทำงานบนเครือข่ายเดียวกันได้



รูปที่ 2.4 LoRa

โพรโตคอล LoRa ประกอบด้วยเลเยอร์ต่างๆ รวมถึงแอปพลิเคชันและระดับอุปกรณ์ สำหรับการสื่อสารที่ปลอดภัย, การเข้ารหัสที่เครือข่าย. แต่ละเกตเวย์ LoRa มีความสามารถในการจัดการโหนดได้มากถึงล้านโหนด. สัญญาณสามารถขยายระยะทางได้อย่างมีนัยสำคัญ, ซึ่งหมายความว่าต้องใช้โครงสร้างน้อยกว่า, ทำให้การสร้างเครือข่ายเร็วขึ้นและถูกกว่ามากในการนำไปใช้

ได้รับการพัฒนาครั้งแรกโดย Cycleo of Grenoble, ฝรั่งเศสซึ่ง Semtech เข้าซื้อกิจการใน 2012 ภายหลัง, พันธมิตรที่ไม่แสวงหาผลกำไรสร้างมาตรฐาน LoRaWAN – โพรโตคอลการสื่อสารที่สร้างขึ้นบนการมอดูเลต LoRa. LoRaWAN กำหนดสถาปัตยกรรมของระบบในขณะที่ LoRa ให้การเชื่อมต่อทางกายภาพระยะไกล

### 2.4.1 เทคโนโลยี LoRa ทำงานอย่างไร

LoRa wireless ใช้การมอดูเลตสเปกตรัมที่กระจายสัญญาณแถบความถี่แคบผ่านแบนด์วิธช่องสัญญาณที่กว้างขึ้น ความมหัสจรรย์ไร้สายระยะไกลมาจากการปรับแต่งที่ชาญฉลาด

Chirp Spread Spectrum (CSS) นี้คือเทคนิคการมอดูเลตเลเยอร์ทางกายภาพที่ใช้โดยวิทย์ LoRa. CSS ทำงานโดยรับสัญญาณไซน์และเปลี่ยนความถี่เป็นเส้นตรงเมื่อเวลาผ่านไป สร้างสัญญาณเจ็บบ สัญญาณนี้จะถูกมอดูเลตเป็นความถี่พาหะ. โดยการส่งสัญญาณผ่านแบนด์วิธที่กว้างขึ้น LoRa กระจายพลังงานในช่วงความถี่ที่กว้างขึ้น สิ่งนี้จะเพิ่มความต้านทานต่อการรบกวนและเสียงรบกวน พร้อมทั้งปรับปรุงการรับสัญญาณที่เกิดเวทย์

อัตราข้อมูลที่ปรับเปลี่ยนได้ (ADR) นี้คือวิธีการของ LoRa ในการเพิ่มประสิทธิภาพอัตราข้อมูลแบบไดนามิกระหว่างอุปกรณ์ปลายทางและเกตเวทย์ การใช้ ADR โหนดสามารถเปลี่ยนเป็นอัตราข้อมูลที่เร็วขึ้นเมื่อสัญญาณรบกวนต่ำ และเปลี่ยนเป็นอัตราข้อมูลที่มีประสิทธิภาพมากขึ้นเมื่อสัญญาณรบกวนเพิ่มขึ้น การเพิ่มประสิทธิภาพนี้ช่วยให้มั่นใจได้ถึงอัตราข้อมูลที่ดีที่สุดภายใต้เงื่อนไขของช่องสัญญาณที่เปลี่ยนแปลง เพิ่มอายุการใช้งานแบตเตอรี่ของโหนดและความจุเครือข่ายโดยรวมให้สูงสุด CSS และ ADR ร่วมกันทำให้ LoRa มีช่วงและความยืดหยุ่นที่น่าประทับใจในขณะที่ใช้พลังงานน้อยที่สุด สัญญาณจะดูเหมือนสัญญาณรบกวนในระบบอื่นๆ เพิ่มความปลอดภัย

### 2.4.2 คุณสมบัติหลักและประโยชน์ของเทคโนโลยี LoRa

1. การเชื่อมต่อระยะไกล เปิดใช้งานช่วงของ 15+ กิโลเมตร ในการตั้งค่าชนบทและ 5+ กิโลเมตรในเขตเมือง อนุญาตให้ครอบคลุมในระดับเมือง
2. พลังงานต่ำ ชิพ LoRa ดูดพลังงาน ปล่อยให้เซนเซอร์ทำงาน 5-10+ ปีกับแบตเตอรี่. ลดต้นทุนการบำรุงรักษาได้อย่างมาก
3. การส่งสัญญาณที่ปลอดภัย การเข้ารหัส AES-128 บิตช่วยรักษาความปลอดภัยข้อมูลและป้องกันการดัดแปลง
4. แถบสเปกตรัมทั่วโลก LoRa ดำเนินการในย่านความถี่ ISM ที่ไม่มีใบอนุญาต เช่น 433, 868, และ 915MHz เปิดใช้งานความเข้ากันได้ทั่วโลก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.ความจุสูง เกดเวย์ LoRa เดียวสามารถจัดการข้อความนับล้านต่อวันจากอุปกรณ์หลายพันเครื่อง

6.มาตรฐานเปิด LoRaWAN เป็นข้อมูลจำเพาะแบบเปิดที่ดูแลโดย LoRa Alliance ที่ไม่แสวงหาผลกำไร. ขับเคลื่อนการยอมรับและการทำงานร่วมกัน

#### 2.4.3 การมอดูเลต LoRa

เทคโนโลยี LoRa เป็นการมอดูเลตไร้สายหรือทางกายภาพ (PHY) ชั้นซิกนอล ใช้เพื่อสร้างการเชื่อมโยงการสื่อสารระยะไกล ชั้นกายภาพ LoRa ใช้รูปแบบของการปรับสเปกตรัมการแพร่กระจาย ระบบมอดูเลต LoRa ใช้พัลส์ควบคุมความถี่เชิงเส้นแบบวงกว้าง ระดับของความถี่ที่เพิ่มขึ้นหรือลดลงเมื่อเวลาผ่านไปจะใช้ในการเข้ารหัสข้อมูลที่จะส่ง เช่น รูปแบบของการปรับเสียงเจ็บบ การมอดูเลตประเภทนี้ช่วยให้ระบบไร้สาย LoRa ทำการมอดูเลตสัญญาณที่ต่ำกว่าระดับเสียงรบกวน 20dB เมื่อมอดูเลตขึ้นรวมกับการแก้ไขข้อผิดพลาดในการส่งต่อ FEC เมื่อเปรียบเทียบกับระบบ FSK แบบดั้งเดิม ลิงก์งบประมาณสำหรับระบบ LoRa สามารถให้การปรับปรุงได้มากกว่า 25dB อันเป็นผลมาจากจุดที่การส่งสัญญาณแพร่กระจายในลักษณะสุ่มหลอก อาจเป็นเรื่องยากสำหรับผู้ที่ไม่ใช่ LoRa ที่จะตรวจจับและดูเหมือนเสียงรบกวน ซึ่งสามารถรองรับการรักษาความปลอดภัยของระบบ ข้อได้เปรียบเพิ่มเติมของระบบคือการปรับเสียงรบกวนและระบบ โดยทั่วไป ทนต่อการขัดเซยความถี่และผลที่ตามมา สามารถใช้คริสตัลอสซิลเลเตอร์พื้นฐานกับ a 20-30 การยอมรับ ppm มากกว่าออสซิลเลเตอร์ที่จ่ายตามอนุกรม TCXO สิ่งนี้สามารถประหยัดต้นทุนได้ดีภายในวงจรอิเล็กทรอนิกส์ของโหนด [5]

## 2.5 GPS

Global Positioning System (GPS) เป็นระบบนำทางดาวเทียมที่ใช้สัญญาณจากดาวเทียมอยู่ในอวกาศเพื่อระบุตำแหน่งและนำทางในพื้นที่บนโลก ระบบ GPS ถูกพัฒนาขึ้นโดยทหารสหรัฐในช่วงทศวรรษ 1970s และเริ่มใช้งานในปี 1978 โดยเป็นระบบที่ใช้มากที่สุดในโลกสำหรับการระบุตำแหน่งและนำทางในปัจจุบัน

ระบบ GPS ประกอบด้วยองค์ประกอบหลายอย่างดังนี้:

1. ดาวเทียม GPS: ระบบ GPS มีดาวเทียม GPS ที่วงโคจรอยู่ในอวกาศ โดยมีจำนวนมากกว่า 30 ดวงในปัจจุบัน (โดยจะอัปเดตและเพิ่มจำนวนดาวเทียมตลอดเวลา) ดาวเทียมเหล่านี้ส่งสัญญาณกลับสู่โลกที่อุปกรณ์ GPS บนพื้นผิวโลกรับได้.
2. อุปกรณ์ GPS: อุปกรณ์ GPS คือเครื่องมือหรืออุปกรณ์ที่ใช้รับสัญญาณจากดาวเทียม GPS เพื่อคำนวณตำแหน่งและเวลาปัจจุบัน อุปกรณ์ GPS บวกกับข้อมูลจากอย่างอื่น เช่น แผนที่และฐานข้อมูล จะช่วยให้ผู้ใช้สามารถระบุตำแหน่งที่ตัวเองบนแผนที่หรือพื้นที่ที่ต้องการ
3. สถานีควบคุม: มีสถานีควบคุม GPS ที่ตั้งอยู่บนโลก ซึ่งใช้ในการควบคุมและปรับปรุงการทำงานของดาวเทียม GPS เหล่านี้ สถานีควบคุมเป็นส่วนสำคัญในการให้ความแม่นยำแก่ระบบ GPS

ระบบ GPS ทำงานโดยการรับสัญญาณจากอย่างน้อยสี่ดวงดาวเทียม GPS และใช้เทคโนโลยีทางสายลัด (trilateration) เพื่อคำนวณตำแหน่งของอุปกรณ์ GPS บนพื้นผิวโลก ระบบ GPS สามารถให้ความแม่นยำในการระบุตำแหน่งที่ตัวเองในระดับเมตรต่อกัน และมีการใช้งานในหลายสาขาอุตสาหกรรม เช่น การนำทางทางบก การบิน การทำแผนที่ การติดตามพาหนะ การทำงานในสนามบิน การนำทางทางเรือ และในอุปกรณ์บุคคลเช่นสมาร์ทโฟน นาฬิกาอัจฉริยะ และอุปกรณ์ IoT (Internet of Things) ที่ต้องการระบุตำแหน่งแม่นยำ [6]-[7]

## 2.6 LTE Cat-1

LTE Category 1 (CAT-1) เป็นมาตรฐานการสื่อสารไร้สายความเร็วปานกลางที่ ออกแบบมาโดยเฉพาะสำหรับกรณีการใช้งาน Internet of Things (IoT) และ Machine to Machine (M2M) Long Term Evolution (LTE) เป็นมาตรฐานการเชื่อมต่อรุ่นที่สี่ (4G) เมื่อเวลาผ่านไป ได้มีการก่อให้เกิดรูปแบบ low power wide area network (LPWAN) ที่ใช้พลังงานต่ำ จำนวนหนึ่ง ซึ่งได้รับการออกแบบมาเพื่อสนับสนุนโครงการ IoT ในภาคอุตสาหกรรมเชิงพาณิชย์ และผู้บริโภคมมาตรฐาน LTE Category 1 เปิดตัวครั้งแรกในปี 2551 ทำให้เป็นมาตรฐาน LTE เฉพาะ IoT รุ่นแรก รุ่นเสาอากาศเดี่ยว CAT-1 BIS เปิดตัวในปี 2559

Data rates : CAT-1 ให้ความเร็วสูงสุด 10 เมกะบิตต่อวินาที (Mb/s) สำหรับ ดาวน์ลิงก์ และ 5Mb/s บนอัปลิงก์ ทำให้เหมาะสำหรับแอปพลิเคชัน M2M และ IoT ที่มีพีเจอร์ หลากหลายมากมาย รวมถึงแอปพลิเคชันที่ต้องการการสตรีมวิดีโอ CAT-1 ยังรองรับ Voice over LTE (VoLTE) ทำให้เหมาะสำหรับการใช้งาน เช่น การควบคุมการเข้าออกระยะไกล และการช่วยเหลือในการใช้ชีวิต ซึ่งจำเป็นต้องมีการรองรับด้วยเสียง

Latency : มาตรฐาน CAT-1 มีเวลาแฝงระหว่าง 50 ถึง 100 มิลลิวินาที ซึ่ง เปรียบเทียบได้ดีกับมาตรฐาน LTE IoT ที่ใช้พลังงานต่ำกว่า เช่น Cat M1 และ NB-IoT ซึ่งเวลาแฝง อาจสูงถึงไม่กี่วินาที สิ่งนี้ช่วยปรับปรุงประสบการณ์ผู้ใช้สำหรับฟังก์ชันต่างๆ เช่น การสตรีมข้อมูล การตรวจสอบระยะไกล และการเฝ้าระวัง

Power usage : CAT-1 รองรับโหมด idle และโหมด sleep ช่วยเพิ่ม ประสิทธิภาพการใช้พลังงานสูงสุด ซึ่งช่วยยืดอายุแบตเตอรี่และอายุการใช้งานของอุปกรณ์ มี ประโยชน์อย่างยิ่งสำหรับอุปกรณ์อุตสาหกรรมที่ติดตั้งในสถานที่เข้าถึงยาก ส่งผลให้มีการ บำรุงรักษาน้อยลงและต้นทุนรวมในการเป็นเจ้าของลดลง.

Coverage : CAT-1 ได้รับประโยชน์จากข้อตกลงการโรมมิ่งที่ครอบคลุมที่กำหนดระหว่าง ผู้ให้บริการเครือข่ายโทรศัพท์เคลื่อนที่ ซึ่งช่วยให้มีตัวเลือกหลายเครือข่ายที่ยืดหยุ่นและการโรมมิ่ง ทั่วโลก ซึ่งทำให้เป็นตัวเลือกที่ดีสำหรับอุปกรณ์ IoT ที่ต้องการการเชื่อมต่ออย่างต่อเนื่องในขณะ เดินทาง (เช่น อุปกรณ์สวมใส่ของผู้บริโภค เป็นต้น) นอกจากนี้ยังมีการส่งมอบในรถ ซึ่งหมายความว่า การเชื่อมต่อจะยังคงอยู่ในรถที่กำลังเคลื่อนที่ [8]

## 2.7 REST API

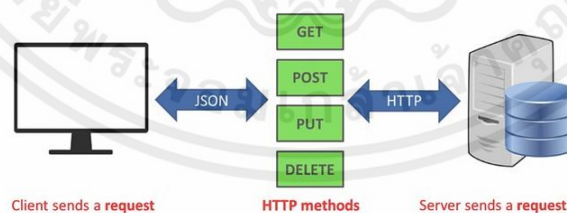
Rest API คือรูปแบบในการสื่อสารของคอมพิวเตอร์ผ่านทางอินเทอร์เน็ตรูปแบบหนึ่ง โดยการสื่อสารนี้จะเกิดขึ้นจากที่ไหนก็ได้ไม่ว่าจะเป็น คอมพิวเตอร์ตัวกับเซิร์ฟเวอร์ที่ไหนซักที่ หน้าเว็บ frontend สื่อสารกับส่วน backend App มีถือสื่อสารกับ backend หรืออาจจะนำไปใช้กับบอร์ด IoT อย่างเช่น ESP32 ก็สามารถใช้งาน Rest API ได้เช่นกัน

Rest API จะทำงานอยู่บนโปรโตคอล HTTP/HTTPS (ที่เราใช้กันปกติในการสื่อสารบน web browser) โดยที่การทำงานของมันก็จะมีการใช้งานผ่าน method ต่างๆ และการสื่อสารก็มักจะทำโดยผ่านชุดข้อมูลที่มี format เป็น JSON

อีกทั้งในการสื่อสารแต่ละครั้งโดยส่วนใหญ่ก็จะมีการพ่วง status code กลับไปด้วย เพื่อบอกว่าการทำงานของมันเป็นไปได้สมบูรณ์หรือไม่

methods ต่างๆของ Rest API ที่ใช้งานบ่อยก็จะมีดังนี้ (ไม่ได้ครบทุกตัว)

1. GET method : หน้าที่ของ GET Method คือใช้สำหรับดึงค่าข้อมูลอะไรซักอย่างจากเซิร์ฟเวอร์
2. POST method : หน้าที่ของ POST Method คือใช้สำหรับสร้างชุดข้อมูลอะไรซักอย่างบนเซิร์ฟเวอร์
3. PUT method : หน้าที่ของ PUT Method คือใช้แก้ไขข้อมูลที่มีอยู่บนเซิร์ฟเวอร์อยู่
4. DELETE method : หน้าที่ของ DELETE Method ก็จะใช้เกี่ยวข้องกับการลบข้อมูลออกจากเซิร์ฟเวอร์ [9]



รูปที่ 2.5 การทำงานของ Rest API

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.8 FLASK (WEB FRAMEWORK)

Flask คือ micro web framework ที่เขียนด้วย python Flask ถูกจัดให้อยู่ใน microframework เพราะที่ไม่ต้องการใช้เครื่องมือหรือไลบรารีเฉพาะ ดังนั้นจึงมีแค่ที่จำเป็นเลยเล็ก แต่เขียนไม่ก็บรรทัดกันได้เลย Flask ไม่มีทั้ง database หรือการตรวจการตรวจแบบฟอร์ม (form validation) แต่ยังรองรับ extensions ที่ช่วยเพิ่มความสามารถอื่นได้ ถ้ารองรับการใช้กับ Flask ตัวอย่างเว็บ ที่นำ Flask ไปใช้ คือ Pinterest LinkedIn เป็นต้น [10]

## 2.9 DATABASE

Database หรือ ฐานข้อมูล คือ กลุ่มของข้อมูลที่ถูกเก็บรวบรวมไว้ โดยมีความสัมพันธ์ซึ่งกันและกัน โดยไม่ได้บังคับว่าข้อมูลทั้งหมดนี้จะต้องเก็บไว้ในแฟ้มข้อมูลเดียวกันหรือแยกเก็บหลายๆ แฟ้มข้อมูล ซึ่งถูกจัดเก็บอย่างเป็นระบบ โดยมีซอฟต์แวร์เข้ามาควบคุมกระบวนการใช้งาน การทำงาน หรือการประมวลผล ทำให้ผู้ใช้สามารถใช้ข้อมูลได้อย่างมีประสิทธิภาพ นอกจากนี้ยังมีหลายภาษาที่สามารถทำงานร่วมกับฐานข้อมูล MySQL ได้ อาทิ C,C++ , Python, Java และอื่นๆอีกมากมาย [11]

### 2.9.1 MariaDB



MariaDB คือ ซอฟต์แวร์สำหรับจัดการกับฐานข้อมูล MariaDB เป็นหนึ่งในฐานข้อมูล ที่ได้รับความนิยมมากที่สุดในโลก MariaDB ถูกพัฒนาขึ้นโดยนักพัฒนาเดิมของ MySQL อีกทั้ง MariaDB ยังเป็นระบบจัดการฐานข้อมูลที่เป็น Open Source (ฟรี) มักใช้เป็นอีกทางเลือกทดแทน ในการจัดการ MySQL ของแพลตฟอร์มยอดนิยมอย่าง LAMP stack หรือ Linux, Apache, MySQL, PHP/Python/Perl นอกจากนี้ MariaDB ถูกพัฒนาออกมาเพื่อต้องการมาแทนที่ MySQL [12]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.10 Web cookies



รูปที่ 2.7 Web cookies

Web cookies เป็นไฟล์ข้อความขนาดเล็กที่เว็บไซต์สร้างขึ้นบนคอมพิวเตอร์หรืออุปกรณ์ของคุณเมื่อคุณเข้าชมเว็บไซต์ ไฟล์เหล่านี้ช่วยให้เว็บไซต์จดจำข้อมูลเกี่ยวกับคุณและการตั้งค่าของคุณ เช่น ภาษาที่คุณต้องการใช้ รายการสินค้าในตะกร้าสินค้าของคุณ หรือว่าคุณเคยลงชื่อเข้าใช้หรือไม่

ประเภทของ Cookies บนเว็บไซต์

- Session Cookies: คุณก็ตัวนี้เปรียบเสมือนหน่วยความจำชั่วคราวของเว็บไซต์ที่คอยบันทึกข้อมูลความเคลื่อนไหวที่เกิดขึ้นบนหน้าเว็บไซต์ โดยจะไม่เก็บข้อมูลที่เกี่ยวข้องกับคอมพิวเตอร์และข้อมูลส่วนตัว เมื่อปิดเบราว์เซอร์ ไฟล์คุณก็ตัวนี้จะถูกลบทิ้งไปทันที

- Secure Cookies: เป็นคุณก็ตัวที่สามารถส่งผ่านการเชื่อมต่อที่ถูกเข้ารหัสเอาไว้แล้วเท่านั้น ซึ่งคุณก็ตัวนี้จะช่วยปกป้องข้อมูลสำคัญ ไม่ให้มีผู้ดักกระหว่างทางนั่นเอง แต่แอกเกอร์ยังสามารถเข้าโจมตีและควบคุมคุณก็ตัวนี้ผ่านช่องทางที่ไม่ปลอดภัยได้

- First-Party Cookies: คุณก็ตัวนี้ถือเป็นหน่วยความจำถาวรของเว็บไซต์ที่เมื่อคุณเข้าใช้งานเว็บนั้นในครั้งแรก จะมีการตั้งค่าทุกอย่างเอาไว้ เพื่อในอนาคตหากคุณเข้ามาเยี่ยมชมเว็บไซต์อีก จะสามารถใช้งานในรูปแบบเดิมได้ แต่ข้อเสียของคุณก็ตัวประเภทนี้ก็คือ เจ้าของเว็บไซต์สามารถติดตามพฤติกรรมการใช้งานของเราได้นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Third-Party Cookies: คุณก็ตัวนี้เป็นคุกกี้ที่อนุญาตให้ Third-Party สามารถเข้าถึง และแสดงโฆษณาต่อผู้เข้าชมเว็บไซต์ได้ โดยที่ Third-Party Cookie จะแอบติดตามความเคลื่อนไหวของเรา ทั้งประวัติการเล่นเว็บไซต์ พฤติกรรมออนไลน์ ไปจนถึงลักษณะการใช้จ่ายต่าง ๆ เป็นต้น

#### วัตถุประสงค์ของ Web Cookies

- เพื่อปรับปรุงประสบการณ์การใช้งานของคุณ: cookies ช่วยให้เว็บไซต์จดจำการตั้งค่าของคุณและนำเสนอเนื้อหาที่เกี่ยวข้องกับคุณ ตัวอย่างเช่น cookies ช่วยให้เว็บไซต์จดจำภาษาที่คุณต้องการใช้ หรือสินค้าที่คุณเคยดูไว้

- เพื่อติดตามการวิเคราะห์: cookies ช่วยให้เว็บไซต์ติดตามจำนวนผู้เข้าชม วิธีที่ผู้ใช้โต้ตอบกับเว็บไซต์ และข้อมูลอื่นๆ ที่เป็นประโยชน์สำหรับการวิเคราะห์และพัฒนาเว็บไซต์

- เพื่อการโฆษณา: cookies ช่วยให้เว็บไซต์แสดงโฆษณาที่เกี่ยวข้องกับคุณ โดยติดตามพฤติกรรมการท่องเว็บของคุณ

#### การจัดการ Web Cookies

สามารถจัดการ web cookies ผ่านการตั้งค่าเบราว์เซอร์ของและสามารถเลือกที่จะบล็อก cookies ทั้งหมด อนุญาต cookies ทั้งหมดหรืออนุญาต cookies เฉพาะบางประเภท

#### ความเป็นส่วนตัวและ Web Cookies

Web cookies บางประเภทอาจเก็บข้อมูลส่วนบุคคลของ เช่น ชื่อ ที่อยู่อีเมลหรือหมายเลขโทรศัพท์ เว็บไซต์ควรแจ้งให้ทราบถึง cookies ที่ใช้วัตถุประสงค์และวิธีการจัดการ cookies เรามีสิทธิ์ที่จะปฏิเสธการใช้ cookies หรือลบ cookies ที่เก็บไว้บนคอมพิวเตอร์ของ [35]

## 2.11 โปรแกรมที่ใช้ในโครงการ

### 2.11.1 Visual Studio Code



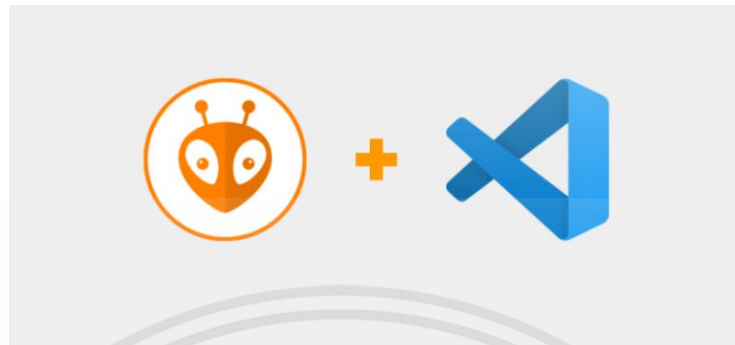
รูปที่ 2.8 Visual Studio Code

วิซวลสตูดิโอโค้ด (อังกฤษ: Visual Studio Code) เป็นโปรแกรมแก้ไขซอร์สโค้ดที่พัฒนาโดยไมโครซอฟท์สำหรับ Windows, Linux และ macOS มีการสนับสนุนสำหรับการดีบั๊ก การควบคุม Git ในตัวและ GitHub การเน้นไวยากรณ์ การเติมโค้ดอัจฉริยะ ตัวอย่าง และ code refactoring มันสามารถปรับแต่งได้หลายอย่าง ให้ผู้ใช้สามารถเปลี่ยนธีม แป้นพิมพ์ลัด การตั้งค่า และติดตั้งส่วนขยายที่เพิ่มฟังก์ชันการทำงานเพิ่มเติม ซอร์สโค้ดนั้นฟรีและโอเพนซอร์สและเผยแพร่ภายใต้สิทธิ์การใช้งาน MIT ไบนารีที่คอมไพล์แล้วเป็นฟรีแวร์และฟรีสำหรับการใช้ส่วนตัวหรือเพื่อการค้า

วิซวลสตูดิโอโค้ดใช้โอเอทีอีเป็นเฟรมเวิร์กที่ใช้ในการปรับใช้แอปพลิเคชัน Node.js สำหรับเดสก์ท็อปที่รันบนเอ็นจิน Blink แม้ว่าจะใช้เฟรมเวิร์กโอเอทีอี ซอฟต์แวร์นี้ไม่ได้ใช้อะตอม และใช้คอมโพเนนต์ตัวแก้ไขเดียวกัน (ชื่อรหัส "Monaco") กับที่ใช้ใน Azure DevOps (เดิมชื่อ Visual Studio Online และ Visual Studio Team Services)

ในการสำรวจนักพัฒนา Stack Overflow 2019 วิซวลสตูดิโอโค้ดได้รับการจัดอันดับให้เป็นเครื่องมือสำหรับนักพัฒนาด้านสิ่งแวดล้อมที่ได้รับความนิยมมากที่สุดโดย 50.7% ของผู้ตอบแบบสอบถาม 87,317 รายอ้างว่าใช้งาน [13]

### 2.11.2 PlatformIO บน VSCode



รูปที่ 2.9 PlatformIO บน VSCode

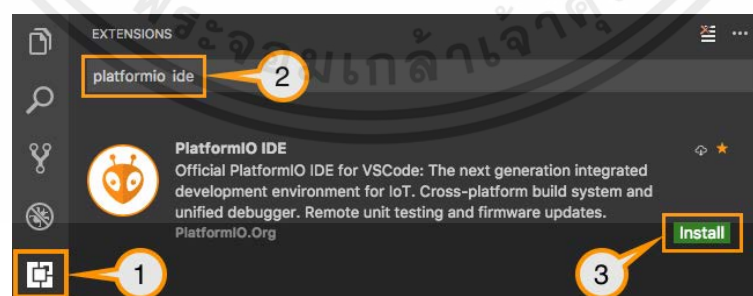
PlatformIO เป็น Open-source software สำหรับพัฒนา Embedded หรือพัฒนา IoT ได้ในแบบ Cross-platform หมายความว่า PlatformIO จะช่วยให้เราสามารถพัฒนาโปรแกรมสำหรับ Embedded ตระกูล ใดๆก็ได้ เช่น Arduino, ESP8266, EMBED ฯลฯ

สำหรับ PlatformIO มีสองแบบให้เลือกใช้งานตามความถนัดคือ หรือจะใช้งานร่วมกันก็ได้

- PlatformIO CLI: การใช้งานแบบ Command Line
- PlatformIO IDE: การใช้งานผ่าน IDE ต่างๆ เช่น VSCode, Atom TextEdito

วิธีติดตั้ง PlatformIO เพื่อใช้งานบน VSCode

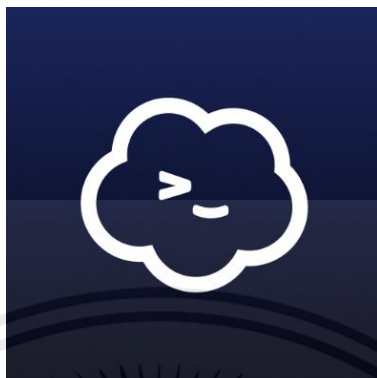
1. เปิด VSCode Extension Manager ขึ้นมา
2. ค้นหา Official PlatformIO IDE extension
3. กด Install PlatformIO IDE [14]



รูปที่ 2.10 PlatformIO บน VSCode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.11.3 Termius



รูปที่ 2.11 Termius

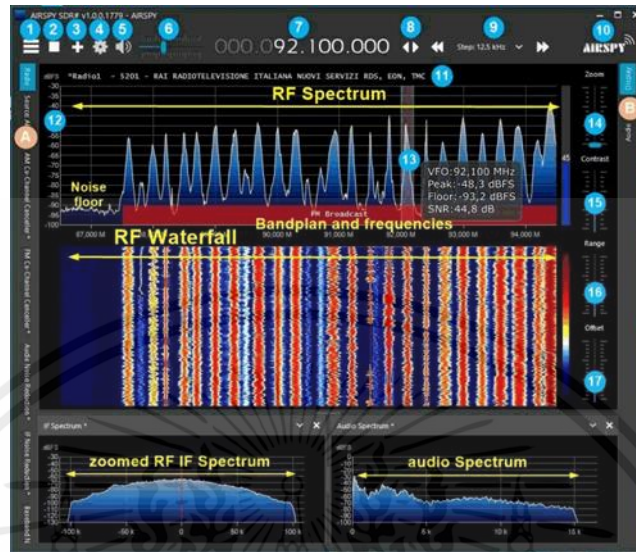
Termius เป็นโปรแกรมจำลองเทอร์มินัลแบบข้ามแพลตฟอร์มที่ช่วยให้ผู้ใช้สามารถเชื่อมต่อกับเซิร์ฟเวอร์ระยะไกลและอุปกรณ์ IoT ได้อย่างปลอดภัยและมีประสิทธิภาพ Termius รองรับโปรโตคอล SSH, Telnet, SFTP, FTP และอื่น ๆ และมีคุณสมบัติมากมายที่ช่วยให้ผู้ใช้สามารถทำงานกับเซิร์ฟเวอร์ระยะไกลได้อย่างสะดวกสบายยิ่งขึ้น

คุณสมบัติที่สำคัญของ Termius:

1. การเชื่อมต่อที่ปลอดภัยและเชื่อถือได้: Termius ใช้การเข้ารหัส AES-256 เพื่อป้องกันการเชื่อมต่อระยะไกล
2. ความสามารถในการปรับแต่งได้สูง: Termius ช่วยให้ผู้ใช้ปรับแต่งเทอร์มินัลได้ตามต้องการ รวมถึงการตั้งค่าแบบอักษร สี และปุ่มลัด
3. คุณสมบัติที่มีประโยชน์มากมาย: Termius มีคุณสมบัติมากมายที่ช่วยให้ผู้ใช้ทำงานกับเซิร์ฟเวอร์ระยะไกลได้อย่างสะดวกสบายยิ่งขึ้น เช่น ความสามารถในการบันทึกเซสชัน ความสามารถในการแชทกับผู้ใช้ระยะไกล และความสามารถในการใช้ส่วนขยาย

Termius สามารถใช้ได้ทั้งบนเดสก์ท็อปและอุปกรณ์มือถือ และให้บริการฟรีสำหรับการใช้งานส่วนบุคคล [15]

### 2.11.4 SDR# (SDRSharp)



รูปที่ 2.12 ภาพหน้าจอแสดงโปรแกรม SDR#

SDR Sharp เป็นเครื่องมือวิเคราะห์สัญญาณวิทยุแบบโอเพนซอร์สที่พัฒนาโดย Eric Bloom เครื่องมือนี้ช่วยให้ผู้ใช้สามารถรับสัญญาณวิทยุจากหลากหลายแหล่งและวิเคราะห์สัญญาณเหล่านั้นโดยใช้ชุดเครื่องมือขั้นสูง SDR Sharp สามารถใช้สำหรับการฟังวิทยุ การตรวจสอบเครือข่ายไร้สาย การติดตามดาวเทียม และอื่นๆ อีกมากมาย

#### 1. คุณสมบัติหลักของ SDR Sharp

- รองรับสเปกตรัมวิทยุตั้งแต่ 100 Hz ถึง 6 GHz
- รองรับสัญญาณ FM, AM, SSB, CW, และอื่นๆ
- รองรับการกรองสัญญาณและการปรับแต่ง
- รองรับการบันทึกและเล่นสัญญาณ
- รองรับการถอดรหัสสัญญาณวิทยุ

#### 2. การใช้งาน SDR Sharp

SDR Sharp สามารถใช้งานได้ง่าย เพียงดาวน์โหลดและติดตั้งซอฟต์แวร์ จากนั้นเชื่อมต่อเครื่องรับสัญญาณวิทยุกับคอมพิวเตอร์ SDR Sharp จะตรวจจับเครื่องรับสัญญาณและแสดงสัญญาณวิทยุที่รับได้และสามารถใช้สำหรับการฟังวิทยุ การตรวจสอบเครือข่ายไร้สาย การติดตามดาวเทียม และอื่นๆ อีกมากมาย ตัวอย่างเช่น ผู้ใช้สามารถใช้ SDR Sharp เพื่อฟังสถานีวิทยุท้องถิ่น ตรวจสอบสัญญาณ Wi-Fi ของเพื่อนบ้าน หรือติดตามดาวเทียมเทียม [16]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.11.5 Postman



รูปที่ 2.13 POSTMAN

Postman เป็นแพลตฟอร์ม API สำหรับนักพัฒนา บริษัทมีสำนักงานใหญ่ในซานฟรานซิสโกและมีสำนักงานในบังกาลอร์ซึ่งเป็นสถานที่ก่อตั้ง ณ เดือนกุมภาพันธ์ 2023 บรูซไปรษณีย์ รายงานว่ามีผู้ใช้ที่ลงทะเบียนมากกว่า 25 ล้านคนและ API แบบเปิด 75,000 รายการ ซึ่งระบุว่า เป็นศูนย์กลาง API สาธารณะที่ใหญ่ที่สุดในโลก

Postman เริ่มต้นในปี 2014 โดยเป็นโปรเจกต์เสริมของวิศวกรซอฟต์แวร์ Abhinav Asthana อดีตนักศึกษาฝึกงานที่ Yahoo Bangalore ซึ่งพบว่าการพัฒนาและทดสอบ API เป็นเรื่องยาก เขาเปิดตัวบรูซไปรษณีย์เป็นแอปฟรีใน Chrome เว็บสโตร์ Abhinav คัดเลือกอดีตเพื่อนร่วมงาน Ankit Sobti และ Abhijit Kane เพื่อก่อตั้ง Postman Inc ในปี 2014 ผู้ร่วมก่อตั้งทั้งสามคนเป็นผู้นำบริษัทในปัจจุบัน โดยมี Abhinav ดำรงตำแหน่ง CEO และ Sobti ดำรงตำแหน่ง CTO

ในเดือนพฤษภาคม 2558 บรูซไปรษณีย์ระดมทุนได้ 1 ล้านดอลลาร์ ในเดือนสิงหาคม 2021 บริษัทระดมทุน Series D ได้ 225 ล้านดอลลาร์ที่มูลค่า 5.6 พันล้านดอลลาร์

ในปี 2023 Postman ได้รับการเสนอชื่อให้อยู่ในรายชื่อ Forbes Cloud 100 เป็นเวลาสามปีติดต่อกัน โดยอยู่ในอันดับที่ 28 [17]

## 2.11.6 phpMyAdmin



รูปที่ 2.14 phpMyAdmin

phpMyAdmin เป็นเครื่องมือจัดการฐานข้อมูล MySQL แบบเว็บ ช่วยให้ผู้ใช้สามารถจัดการฐานข้อมูลได้โดยไม่ต้องเขียนภาษา SQL โดยตรง

คุณสมบัติหลักของ phpMyAdmin:

- สร้าง, ปรับแต่ง และลบ ฐานข้อมูล ตาราง และฟิลด์
- Import และ Export ข้อมูล ในรูปแบบต่างๆ เช่น CSV, SQL, XML
- ดำเนินการกับเพียร์ (Query) ด้วย SQL
- จัดการผู้ใช้ กำหนดสิทธิ์การเข้าถึงฐานข้อมูล

การใช้งาน phpMyAdmin:

1. เข้าสู่ phpMyAdmin: โดยทั่วไปสามารถเข้าได้ผ่านทาง <http://localhost/phpmyadmin/> หรือ URL ที่ทางผู้ดูแลระบบกำหนด
2. เข้าสู่ระบบ: กรอกชื่อผู้ใช้และรหัสผ่านฐานข้อมูล
3. จัดการฐานข้อมูล:
  - เลือกฐานข้อมูลที่ต้องการจากเมนูทางซ้าย
  - ทำงานกับตาราง, เพิ่ม/ลบ/แก้ไขข้อมูล,
  - Import/Export ข้อมูล
  - เขียน Query [36]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.12 ภาษาคอมพิวเตอร์ที่ใช้ในโครงการงาน

### 2.12.1 Python

Python คือภาษาโปรแกรมระดับสูง ออกแบบโดย Guido van Rossum ชาวนเนเธอร์แลนด์ เริ่มการพัฒนาในปี พ.ศ. 2533 และเผยแพร่สู่สาธารณะครั้งแรกในปี พ.ศ. 2535 Python ได้รับความนิยมเพิ่มขึ้นอย่างรวดเร็วในช่วงไม่กี่ปีที่ผ่านมา และกลายเป็นหนึ่งในภาษาโปรแกรมที่ได้รับความนิยมมากที่สุดในโลก

Python มีคุณสมบัติหลายประการที่ทำให้เป็นภาษาโปรแกรมที่ได้รับความนิยม

1. อ่านง่ายและเขียนง่าย ภาษา Python ใช้ไวยากรณ์ที่เรียบง่ายและตรงไปตรงมา ทำให้ง่ายต่อการเข้าใจและเรียนรู้ของเนกประสงค์: Python สามารถใช้ในการพัฒนาซอฟต์แวร์ประเภทต่างๆ รวมถึงเว็บแอปพลิเคชัน วิทยาศาสตร์ข้อมูล แมชชีนเลิร์นนิง และอื่นๆ

2. มีประสิทธิภาพ: Python มีประสิทธิภาพเพียงพอสำหรับการใช้งานจริง

3. ฟรีและโอเพนซอร์ส: Python เป็นซอฟต์แวร์ฟรีและโอเพนซอร์ส ทุกคนสามารถดาวน์โหลดและใช้งานได้โดยไม่เสียค่าใช้จ่าย

4. มีเครื่องมือและไลบรารีมากมาย ภาษา Python มีเครื่องมือและไลบรารีมากมายที่ช่วยให้นักพัฒนาสามารถพัฒนาโปรแกรมได้อย่างรวดเร็วและมีประสิทธิภาพ

ภาษา Python มีโครงสร้างพื้นฐานดังนี้

1. ตัวแปร ตัวแปรในภาษา Python ใช้เพื่อเก็บข้อมูล ค่าของตัวแปรสามารถเปลี่ยนแปลงได้ตลอดเวลา

2. คำสั่ง คำสั่งในภาษา Python ใช้เพื่อบอกคอมพิวเตอร์ให้ทำงานบางอย่าง

3. เงื่อนไข เงื่อนไขในภาษา Python ใช้เพื่อกำหนดว่าคำสั่งใดจะทำงานเมื่อใด

4. ลูป ลูปในภาษา Python ใช้เพื่อทำงานซ้ำๆ

5. ฟังก์ชัน ฟังก์ชันในภาษา Python ใช้เพื่อรวบรวมชุดคำสั่งไว้ทำงาน

ร่วมกัน [18]

### 2.12.2 SQL

SQL หรือ Structured Query Language คือภาษาที่ใช้ในการเข้าถึงฐานข้อมูล ตั้งแต่ดึงข้อมูลไปจนถึงจัดการข้อมูล ซึ่ง SQL เป็นหนึ่งในภาษา programming ที่ได้รับการยอมรับมากที่สุด เรามักจะใช้งานภาษา SQL ผ่านโปรแกรมต่างๆ ที่ต้องทำงานกับระบบฐานข้อมูล

#### 1. หลักการเบื้องต้นของภาษา SQL

SQL เป็นภาษาที่ไม่ซับซ้อน สามารถเข้าใจได้ง่าย เพราะมีหลักการคล้ายกับไวยากรณ์ภาษาอังกฤษ เช่น เวลาเราต้องการแสดงข้อมูลชื่อลูกค้าจากฐานข้อมูลเราสามารถพิมพ์คำสั่งว่า

```
SELECT name
```

```
FROM customers
```

```
WHERE city = 'Bangkok' OR 'Chiang Mai';
```

คำสั่งนี้มีความหมายว่า “เลือกคอลัมน์ชื่อจากราย customers โดยเลือกเฉพาะลูกค้าที่อยู่ในกรุงเทพหรือเชียงใหม่” จะเห็นได้ว่า เราสามารถเข้าใจภาษา SQL ได้ง่ายคล้ายการอ่านประโยค ทำให้คนที่ไม่ใช่สายเขียนโปรแกรมก็สามารถเข้าใจภาษานี้ได้ง่าย [19]

### 2.12.3 HTML

เอชทีเอ็มแอล (อังกฤษ: HTML: Hypertext Markup Language ภาษามาร์กอัปข้อความหลายมิติ) เป็นภาษามาร์กอัปหลักในปัจจุบันที่ใช้ในการสร้างเว็บเพจ หรือข้อมูลอื่นที่เรียกดูผ่านทางเว็บเบราว์เซอร์ ซึ่งตัวโค้ดจะแสดงโครงสร้างของข้อมูล ในการแสดง หัวข้อ ลิงก์ ย่อหน้า รายการ รวมถึงการสร้างแบบฟอร์ม เชื่อมโยงภาพหรือวิดีโอด้วย โครงสร้างของโค้ดเอชทีเอ็มแอลจะอยู่ในลักษณะภายในวงเล็บสามเหลี่ยม

ในปัจจุบัน HTML เป็นมาตรฐานหนึ่งของ ISO ซึ่งจัดการโดย World Wide Web Consortium (W3C) ในปัจจุบัน ทาง W3C ผลักดัน รูปแบบของ HTML แบบใหม่ ที่เรียกว่า XHTML ซึ่งเป็นลักษณะของโครงสร้าง XML แบบหนึ่งที่มีหลักเกณฑ์ในการกำหนดโครงสร้างของโปรแกรมที่มีรูปแบบที่มาตรฐานกว่า มาทดแทนใช้ HTML รุ่น 4.01 ที่ใช้กันอยู่ในปัจจุบัน

#### 2.12.3.1 Markup

ภาษามาร์กอัป (อังกฤษ: markup language) คือประเภทภาษาคอมพิวเตอร์ที่แสดงทั้งข้อมูล และข้อมูลรูปแบบเข้าด้วยกัน โดยข้อมูลรูปแบบอธิบายถึงโครงสร้างหรือการแสดงผลซึ่งส่วนนี้เรียกว่า มาร์กอัป โดยจะอยู่ร่วมกับข้อมูลปกติ ภาษามาร์กอัปที่รู้จักกันดีที่สุดคือ HTML ตามความเป็นมาแล้ว ภาษารูปแบบนี้ได้มีการใช้ในอุตสาหกรรมการพิมพ์ในการติดต่อสื่อสารงานพิมพ์ระหว่างผู้เขียน บรรณาธิการ และเครื่องพิมพ์

- Markup มีการอธิบายจุดประสงค์ของข้อมูล ตัวอย่างเช่น

```
<h2>ฟุตบอล</h2>
```

กำหนดให้เบราว์เซอร์คำนวณ "ฟุตบอล" เป็นลักษณะของหัวข้ออันดับที่ 2 มาร์กอัปโครงหลัก โดยปกติไม่ได้กำหนดลักษณะการแสดงผล แต่อย่างไรก็ตาม ทางเบราว์เซอร์กำหนดการแสดงผลมาตรฐานของมาร์กอัป โดยปกติจะแสดงผลในลักษณะที่ตัวอักษรขนาดใหญ่ และมีความหนา

- มาร์กอัปสำหรับ การแสดงผล อธิบายการแสดงผลของ ข้อความโดยไม่ได้มีความหมายอื่นในทางโครงสร้าง ตัวอย่างเช่น

```
<b>ตัวหนา</b> <i>ตัวเอียง</i> <u>ขีดเส้นใต้</u>
```

กำหนดให้คำว่า "ตัวหนา" แสดงผลในลักษณะตัวหนา เช่นเดียวกับการแสดงผลใน ตัวเอียง หรือ ขีดเส้นใต้ [20]

### 2.12.4 CSS

CSS คือ ภาษาที่ใช้สำหรับตกแต่งเอกสาร HTML/XHTML ให้มีหน้าตา สี สัน ละเอียด พื่นหลัง เส้นขอบและอื่นๆ ตามที่ต้องการ CSS ย่อมาจาก Cascading Style Sheets มีลักษณะเป็นภาษาที่มีรูปแบบในการเขียน Syntax แบบเฉพาะและได้ถูกกำหนดมาตรฐานโดย W3C เป็นภาษาหนึ่งในการตกแต่งเว็บไซต์ ได้รับความนิยมนอย่างแพร่หลาย

ประโยชน์ของ CSS

CSS มีประโยชน์อย่างหลากหลาย ซึ่งได้แก่

1. ช่วยให้เนื้อหาภายในเอกสาร HTML มีความเข้าใจได้ง่ายขึ้น และในการแก้ไขเอกสารก็สามารถทำได้ง่ายกว่าเดิม เพราะการใช้ CSS จะช่วยลดการใช้ภาษา HTML ลงได้ในระดับหนึ่ง และแยกระหว่างเนื้อหากับรูปแบบในการแสดงผลได้อย่างชัดเจน
2. ทำให้สามารถดาวน์โหลดไฟล์ได้เร็ว เนื่องจาก code ในเอกสาร HTML ลดลง จึงทำให้ไฟล์มีขนาดเล็กลง
3. สามารถกำหนดรูปแบบการแสดงผลจากคำสั่ง style sheet ชุดเดียวกัน ให้มีการแสดงผลในเอกสารแบบเดียวกันทั้งหน้าหรือในทุกๆ หน้าได้ ช่วยลดเวลาในการปรับปรุงและทำให้การสร้างเอกสารบนเว็บมีความรวดเร็วยิ่งขึ้น นอกจากนี้ยังสามารถควบคุมการแสดงผล ให้คล้ายหรือเหมือนกันได้ในหลาย Web Browser
4. ช่วยในการกำหนดการแสดงผลในรูปแบบที่มีความเหมาะสมกับสื่อต่างๆ ได้เป็นอย่างดี
5. ทำให้เว็บไซต์มีความเป็นมาตรฐานมากขึ้นและมีความทันสมัย สามารถรองรับการใช้งานในอนาคตได้ดี [21]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.12.5 JS

JavaScript เป็นภาษาโปรแกรมแบบไดนามิกที่ใช้สร้างเว็บไซต์แบบโต้ตอบ ภาษาโปรแกรมนี้ทำงานฝั่งไคลเอนต์ หมายความว่าทำงานบนเบราว์เซอร์ของผู้ใช้โดยไม่ต้องส่งข้อมูลไปยังเซิร์ฟเวอร์

JavaScript ทำได้

- เพิ่มปฏิสัมพันธ์ให้กับเว็บไซต์ เช่น ปุ่มคลิก เมนูแบบเลื่อนลง แอนิเมชัน
- ตรวจสอบข้อมูลป้อนเข้าจากผู้ใช้
- จัดการกับ DOM (Document Object Model) ซึ่งเป็นโครงสร้างของเว็บไซต์
- ทำงานร่วมกับ Ajax เพื่อดึงข้อมูลจากเซิร์ฟเวอร์โดยไม่ต้องโหลดหน้าใหม่
- สร้างเกมและแอปพลิเคชันเว็บ

ไวยากรณ์พื้นฐานของ JavaScript:

- ตัวแปร: ใช้คำหลัก var let หรือ const ประกาศตัวแปร
- ประเภทข้อมูล: JavaScript มีประเภทข้อมูลหลักๆ เช่น string number boolean object array
- โครงสร้างควบคุม: ใช้ if else switch for while ควบคุมการไหลของโปรแกรม
- ฟังก์ชัน: ใช้ function ประกาศฟังก์ชันเพื่อนำโค้ดมาใช้ซ้ำ
- วัตถุ: ใช้ {} ประกาศวัตถุเพื่อเก็บข้อมูล [37]

## 2.12.6 C++



รูปที่ 2.15 C++

ภาษาซีพลัสพลัส (อังกฤษ: C++) เป็นภาษาโปรแกรมคอมพิวเตอร์อเนกประสงค์ มีโครงสร้างภาษาที่มีการจัดชนิดข้อมูลแบบสแตติก (statically typed) และสนับสนุนรูปแบบการเขียนโปรแกรมที่หลากหลาย (multi-paradigm language) ได้แก่ การโปรแกรมเชิงกระบวนคำสั่ง การนิยามข้อมูล การโปรแกรมเชิงวัตถุ และการโปรแกรมแบบเจเนริก (generic programming)

### 2.12.6.1 รูปแบบของการออกแบบภาษาซีพลัสพลัส

- ในทางทฤษฎี ภาษาซีพลัสพลัสควรจะมีความเร็วเทียบเท่าภาษาซี แต่ในการเขียนโปรแกรมจริงนั้น ภาษาซีพลัสพลัสเป็นภาษาที่มีการเปิดกว้างให้โปรแกรมเมอร์เลือกรูปแบบการเขียนโปรแกรม ซึ่งทำให้มีแนวโน้มที่โปรแกรมเมอร์อาจใช้รูปแบบที่ไม่เหมาะสม ทำให้โปรแกรมที่เขียนมีประสิทธิภาพต่ำกว่าที่ควรจะเป็น และภาษาซีพลัสพลัสนั้นเป็นภาษาที่มีความซับซ้อนมากกว่าภาษาซี จึงทำให้มีโอกาสเกิดบั๊กขณะคอมไพล์มากกว่า

- มาตรฐานของภาษาซีพลัสพลัส ถูกออกแบบมาเพื่อไม่ให้เกิดการเจาะจงแพลตฟอร์มคอมพิวเตอร์ [22]

ตัวอย่างโค้ด

```
#include <iostream>
int main ()
{
    std :: cout << "Hello World"<< endl ;
    return 0 ;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.12.7 JSON

JavaScript Object Notation (JSON) คือรูปแบบการแลกเปลี่ยนข้อมูลคอมพิวเตอร์ที่อยู่ในรูปข้อความธรรมดา (plain text) ที่ทั้งมนุษย์และโปรแกรมคอมพิวเตอร์สามารถอ่านเข้าใจได้ มาตรฐานของฟอร์แมต JSON คือ RFC 4627 มี Internet media type เป็น application/json และมีนามสกุลของไฟล์เป็น .json

JSON ประกอบด้วยข้อมูลในรูปแบบของวัตถุ (object) หรืออาร์เรย์ (array) โดยวัตถุประกอบด้วยคู่คีย์-ค่า (key-value pair) โดยใช้เครื่องหมายจุลภาค (,) เป็นตัวคั่นคู่คีย์-ค่า และโคลอน (:) เป็นตัวคั่นระหว่างคีย์กับค่า อาร์เรย์ประกอบด้วยรายการข้อมูล โดยรายการข้อมูลแต่ละรายการคั่นด้วยเครื่องหมายจุลภาค (,)

ตัวอย่าง JSON รูปแบบ object :

```
{
  "name": "John Doe",
  "age": 30,
  "address": {
    "street": "123 Main Street",
    "city": "San Francisco",
    "state": "CA",
    "zipcode": "94105"
  }
}
```

รูปที่ 2.16 ตัวอย่าง JSON ในรูปแบบของ Object

ตัวอย่าง JSON รูปแบบ array :

```
[
  "apple",
  "banana",
  "orange"
]
```

รูปที่ 2.17 ตัวอย่าง JSON ในรูปแบบของ array

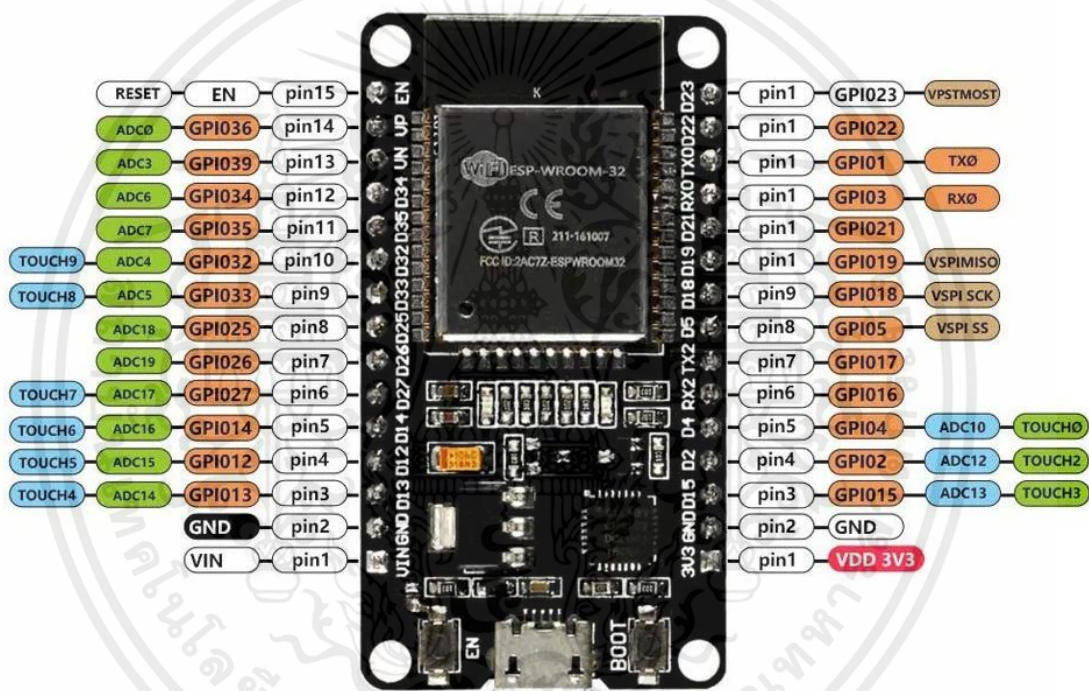
JSON นิยมใช้ในเว็บแอปพลิเคชัน โดยเฉพาะ AJAX โดย JSON เป็นฟอร์แมตทางเลือกในการส่งข้อมูลนอกเหนือไปจาก XML ซึ่งนิยมใช้กันอยู่แต่เดิม สาเหตุที่ JSON เริ่มได้รับความนิยมเป็นเพราะกระชับและเข้าใจง่ายกว่า XML [23]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.13 เครื่องมือที่ใช้ในการทดลอง

### 2.13.1 บอร์ด ESP32 NodeMCU ESP-WROOM-3

MCU – ESP32 เป็น Micro Controller ที่รองรับการเชื่อมต่อ WiFi , Bluetooth – BLE ในตัว ภาษาที่ใช้ในการพัฒนาโปรแกรมคือ ภาษา C หรือ Python ภาษา Python ต้องทำการอัปเดตเฟิร์มแวร์ให้รองรับ Python การพัฒนาโปรแกรมขึ้นอยู่กับผู้ที่พัฒนา โปรแกรม IDE ที่ใช้พัฒนาคือ Arduino IDE หรือ Visual Studio สำหรับ Visual Studio จำเป็นต้องติดตั้ง Plugin Espressif IDF หรือ PlatformIO IDE และต้อง Enable (Arduino)



รูปที่ 2.18 บอร์ด ESP 32

ESP32 รองรับการเชื่อมต่อ WiFi และ BLE หรือ Bluetooth ได้โดยไม่ต้องซื้อโมดูลเพิ่มเติม บอร์ด ESP32 เองยังมีการทำงานที่แบ่งเป็น 2 Core และ Pin I/O เลือกฟังก์ชันการทำงานได้ใน Pin เดียวกัน เช่น การแปลง Analog to Digital หรือ Digital to Analog การเชื่อมต่อ SD Card Camera PWD RTC และ Touch เป็นต้น [24]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ข้อมูลเฉพาะของ บอร์ด ESP32

#### ระบบประมวลผล:

- ซีพียู : Xtensa dual-core (or single-core) 32-bit LX6 microprocessor, ทำงานที่ความถี่ 160 หรือ 240 เมกะเฮิร์ตซ์ และมีประสิทธิภาพสูงถึง 600 DMIPS
- ตัวประมวลผลร่วมที่ใช้พลังงานต่ำเป็นพิเศษ (ULP)
- หน่วยความจำ: แรม 520 กิโลไบต์ , รม 448 กิโลไบต์

#### การเชื่อมต่อไร้สาย:

- Wi-Fi มาตรฐาน 802.11 b/g/n
- Bluetooth รองรับ v4.2 BR/EDR and BLE (แบ่งปันคลื่นวิทยุกับ Wi-Fi)

#### อินเตอร์เฟซอุปกรณ์ต่อพ่วง:

- GPIOs ที่สามารถโปรแกรมได้ จำนวน 34 ช่อง
- 12-บิต SAR ADC มากถึง 18 ช่อง
- DACs แบบ 8-bit จำนวน 2 ช่อง
- ตัวตรวจจับสัมผัส (GPIOs การตรวจจับแบบ capacitive) จำนวน 10 ช่อง
- SPI จำนวน 4 ช่อง
- I<sup>2</sup>S จำนวน 2 ช่อง
- I<sup>2</sup>C จำนวน 2 ช่อง
- UART จำนวน 2 ช่อง
- SD/SDIO/CE-ATA/MMC/eMMC host controller
- SDIO/SPI slave controller
- อินเทอร์เน็ต Ethernet MAC พร้อม DMA แยกต่างหากและรองรับ IEEE 1588

#### Precision Time Protocol เวอร์ชันที่วางแผนไว้สนับสนุน CAN bus 2.0

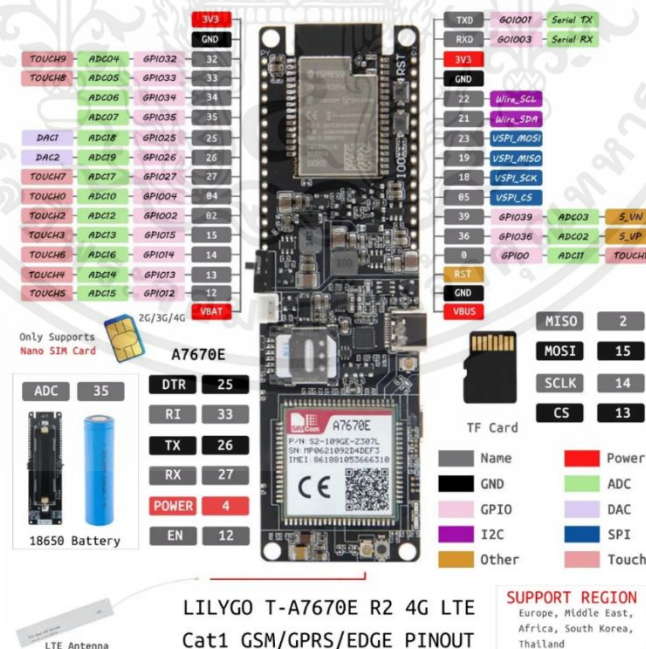
- รีโมตคอนโทรลเลอร์อินฟราเรด (TX/RX, สูงสุด 8 ช่อง)
- การนับพัลส์ (สามารถถอดรหัสแบบ full quadrature)
- PWM สำหรับมอเตอร์ Motor PWM
- LED PWM (สูงสุด 16 ช่อง)

### 2.13.2 บอร์ด ESP32 Lilygo T-SIM7670E

LILYGO TTGO T-SIM-A7670E R2 เป็นไมโครคอนโทรลเลอร์ล้ำสมัยที่ใช้ชิป ESP32 ซึ่งผสานรวมกับโมดูล SIM-A7670E โมดูลอินเทอร์เน็ตนี้มีความสามารถด้านการสื่อสารไร้สายที่หลากหลาย รวมถึง GNSS (GPS), Wi-Fi, Bluetooth และ LTE การรวมกันนี้จะช่วยปลดล็อกโลกแห่งความเป็นไปได้ในการสื่อสารไร้สาย ทำให้เป็นเครื่องมือที่มีความหลากหลายพอๆ กับที่เป็นนวัตกรรมใหม่ ออกแบบมาเพื่อความทนทานและประสิทธิภาพ เป็นตัวเลือกที่เหมาะสมสำหรับนักพัฒนา

ด้วยการเชื่อมต่อ USB-C สำหรับการเขียนโปรแกรมและที่ใส่แบตเตอรี่ 18650 ในตัวก็พร้อมที่จะเป็นส่วนหนึ่งของโครงการต่างๆ อินเทอร์เน็ตเพื่อการสื่อสารที่กว้างขวางทำให้สามารถเชื่อมต่อและควบคุมอุปกรณ์ต่อพ่วงได้หลากหลาย ทำให้สามารถปรับให้เข้ากับการใช้งานที่แตกต่างกันได้มากมายใน wireless domain เป็นอุปกรณ์ที่มีประสิทธิภาพโดดเด่น โดยรองรับระบบ GNSS, Wi-Fi, Bluetooth ที่แตกต่างกัน และแม้แต่เครือข่ายโทรคมนาคม 2G และ 4G LTE CAT 1 ทำให้เป็นตัวเลือกที่สำคัญสำหรับแอปพลิเคชัน IoT และการส่งข้อมูลระยะไกล

วงจรแบตเตอรี่ได้รับการออกแบบมาเพื่อรองรับแบตเตอรี่ Li-ion และ Li-Po พร้อมด้วยการป้องกันในตัวที่ช่วยให้มั่นใจได้ถึงการจัดการพลังงานที่ปลอดภัยและมีประสิทธิภาพ [25]



รูปที่ 2.19 บอร์ด ESP32 Lilygo T-SIM7670E

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ลักษณะเฉพาะกำลังและไฟฟ้า

- แรงดันไฟฟ้า: 4.4 โวลต์ถึง 6 โวลต์ กระแสตรง
- แรงดันขา IO: 3.3 โวลต์
- IO Pin Source Current สูงสุด : 20 มิลลิแอมป์
- IO Pin Sink Current สูงสุด : 28 มิลลิแอมป์
- ความถี่สัญญาณนาฬิกา: 240 เมกะเฮิร์ตซ์

### หน่วยความจำภายใน

- SRAM: 520 กิโลไบต์
- PSRAM: 8 เมกะไบต์
- Flash Memory: 4 เมกะไบต์

### อินเทอร์เฟซการสื่อสาร

- การเชื่อมต่อฮาร์ดแวร์ : I2C, I2S, SPI, UART, PWM, ADC, DAC, GPIO, USB

CDC

- พิน IO: 24
- อินเทอร์เฟซฮาร์ดแวร์ที่มีอยู่ : 8 PWM channels, 16 ADC pins, 2 DAC pins,

3 UART interfaces, 2 I2C interfaces, 2 SPI interfaces

- USB to Serial Converter: CH9102F
- รูปแบบการ์ด SD ที่รองรับ : Micro SD
- รูปแบบซิมการ์ดที่รองรับ: Nano-SIM

### การสื่อสารไร้สาย

- อินเทอร์เน็ต: GNSS, Wi-Fi, Bluetooth, Telecom
- ชิพ Wi-Fi และบลูทูธ: ESP32
- ความถี่ Wi-Fi : 2.4 กิกะเฮิร์ตซ์
- เวอร์ชันบลูทูธ: 4.2 (LE, Classic)
- ชิพ GNSS: A7670E (GPS, Beidou, GLONASS)
- ความแม่นยำ GNSS สูงสุด: 2.5 เมตร
- อัตราการอัปเดต GNSS สูงสุด: 10 เฮิร์ตซ์
- ชิพโทรคมนาคม: A7670E (2G, 4G LTE CAT 1)
- แบนด์ที่รองรับ: 900 เมกะเฮิร์ตซ์, 1800 เมกะเฮิร์ตซ์, B1, B3, B5, B7, B8, B20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

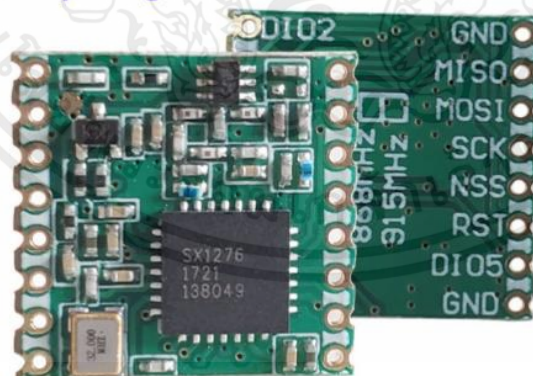
วงจรแบตเตอรี่

- ชิพแบตเตอรี่: CN3065
- ประเภทแบตเตอรี่ที่รองรับ: Li-ion, Li-Po
- แรงดันการชาร์จสูงสุด: 4.2 โวลต์
- กระแสไฟชาร์จสูงสุด: 600 มิลลิแอมป์
- คุณสมบัติการป้องกัน: แรงดันไฟฟ้าเกิน, กระแสเกิน, แรงดันตก
- ช่องใส่แบตเตอรี่: 18650 จำนวน 1 ก้อน

### 2.13.3 โมดูล LoRa SX1276 ย่านความถี่ 900-931 เมกะเฮิร์ตซ์

โมดูล Lora 915 เมกะเฮิร์ตซ์ SX1276 เป็น โมดูลลอว์รา 915 เมกะเฮิร์ตซ์เป็นโมดูลส่งสัญญาณระยะไกล ใช้ไฟเลี้ยง 1.8-3.7V LoRa1276 ใช้ชิปรับส่งสัญญาณ Semtech RF SX1276 ซึ่งใช้ LoRa TM Spread Spectrum modulation frequency hopping technique คุณสมบัติของระยะไกลและความไวสูง (-139 dBm) ทำให้โมดูลนี้ทำงานได้ดีกว่าโมดูล FSK และ GFSK หลายสัญญาณจะไม่ส่งผลกระทบต่อกันแม้ในสภาพแวดล้อมที่มีความถี่ของฝูงชน มันมาพร้อมกับประสิทธิภาพการป้องกันการรบกวนที่แข็งแกร่ง [26]

**915MHz** RF LoRa module SX1276  
chip Long-Distance communication



รูปที่ 2.20 โมดูล LoRa SX1276 ย่านความถี่ 900-931 เมกะเฮิร์ตซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลจำเพาะ:

- กำลังส่ง: 0.1 วัตต์
- แบนด์: 915 - 930 เมกะเฮิรตซ์

คุณสมบัติ:

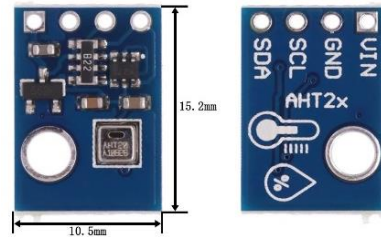
- การใช้เทคโนโลยี FSK / GFSK และเทคโนโลยีสเปกตรัม LoRa (ระยะไกล)
- การสื่อสารแบบฮาล์ฟดูเพล็กซ์
- ป้องกันการรบกวนชั้นสูง (อัตราการปฏิเสธช่องสัญญาณ: 56db)
- ความไวในการรับสัญญาณสูง: 139dbm (32M พาสซีฟ 10ppm คริสตัล)
- ISM multi-band ไม่จำเป็นต้องใช้ความถี่ฟรี
- หลายความถี่และอัตราการส่งข้อมูลที่หลากหลายสำหรับการเลือก สามารถใช้ใน

เทคโนโลยี FDMA และ FM

- ใช้พลังงานต่ำในการรับกระแสไฟฟ้า: 10-12mA
- การส่งสัญญาณทางไกลพิเศษสูงสุด 5 กิโลเมตร [27]

#### 2.13.4 เซนเซอร์วัดอุณหภูมิและความชื้น AHT10 I2C

เซนเซอร์ AHT10 เป็นเซนเซอร์วัดอุณหภูมิและความชื้น ออกแบบมาให้มีขนาดเล็ก สื่อสารกับ MCU แบบ I2C ใช้สายสัญญาณเพียงแค่ 2 เส้น ใช้ไฟเลี้ยง 1.8-6V AHT10 ซึ่งเป็น เซนเซอร์อุณหภูมิและความชื้น กำหนดมาตรฐานใหม่ในด้านขนาดและความชาญฉลาด: มันถูกฝังอยู่ในแพ็คเกจ SMD ไร้สารตะกั่วแบนสองแถว ซึ่งเหมาะสำหรับการบัดกรีแบบรีโฟลว์ โดยมีด้านล่าง ขนาด 4 x 5 มิลลิเมตร. และความสูง 1.6 มิลลิเมตร. เซนเซอร์จะส่งสัญญาณดิจิทัลที่ปรับเทียบแล้ว ในรูปแบบมาตรฐาน I2C AHT10 มาพร้อมกับชิปเฉพาะ ASIC ที่ออกแบบใหม่ องค์กรประกอบการ ตรวจสอบความชื้นแบบคาปาซิทีฟเซมิคอนดักเตอร์ MEMS ที่ได้รับการปรับปรุง และองค์กรประกอบการ ตรวจสอบอุณหภูมิบนชิปมาตรฐาน ประสิทธิภาพได้รับการปรับปรุงอย่างมากเกินกว่าระดับความ น่าเชื่อถือของเซนเซอร์รุ่นก่อนหน้า เซนเซอร์อุณหภูมิและความชื้นรุ่นแรกได้รับการปรับปรุงเพื่อให้ มีเสถียรภาพมากขึ้นในสภาพแวดล้อมที่รุนแรง [28]



รูปที่ 2.21 เซนเซอร์วัดอุณหภูมิและความชื้น AHT10 I2C

ลักษณะเฉพาะ:

- ขนาดโมดูล 16\*11 มิลลิเมตร
- ประเภทอินเทอร์เฟซ : I2C
- ช่วงแรงดันที่ทำงาน : 1.8 - 6.0 โวลต์
- ขนาดอินเทอร์เฟซ : 4\*2.54 มิลลิเมตร
- ความแม่นยำของความชื้น :  $\pm 2\%$
- ความละเอียดความชื้น : 0.024%
- ความแม่นยำของอุณหภูมิ :  $\pm 0.3$  เซลเซียส
- ความละเอียดอุณหภูมิ : 0.01 เซลเซียส
- อุณหภูมิในการทำงาน : -40 เซลเซียสถึง 85 เซลเซียส

### 2.13.5 GPS- Module GY-NEO-6MV2 Ublox พร้อมสายอากาศอุปกรณ์

เพิ่มเติม

โมดูล GY-NEO6MV2 Ublox GPS สำหรับระบุตำแหน่งต่างๆบนโลกเป็นค่า ละติจูด ลองจิจูด มี Library พร้อมใช้งาน รองรับไฟ 3-5 โวลต์การใช้งานโมดูล GPS ครั้งแรก (Cold Start) จะต้องรอล็อกสัญญาณประมาณ 15 นาที ระหว่างล็อกสัญญาณครั้งแรกต้องเอาโมดูลภายนอกอาคาร หากล็อกสัญญาณได้แล้วไฟ Led จะกะพริบ U-BLOX NEO-6M โมดูลมาพร้อมกับเสาอากาศเซรามิกประสิทธิภาพสูงซึ่งเทียบเท่ากับเสาอากาศที่ใช้งานในตัว โมดูลมาพร้อมกับ EEPROM ข้อมูลการกำหนดค่าทั้งหมดสามารถเก็บไว้ใน EEPROM การกำหนดค่าที่หลากหลายเพื่อตอบสนองความต้องการของคุณ โมดูลยังมาพร้อมกับแบตเตอรี่สำรองแบบชาร์จไฟได้ ( เพื่อรองรับการ warm start หลังจากปิดแหล่งจ่ายไฟหลักแล้ว พลังงานแบตเตอรี่สำรองสามารถรักษาข้อมูลตัวรับสัญญาณ GPS ไว้ได้ครึ่งชั่วโมง )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

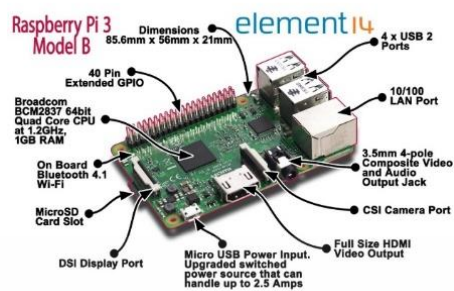


รูปที่ 2.22 GPS- Module GY-NEO-6MV2 Ublox

โมดูล GY-NEO6MV2 Ublox GPS สำหรับระบุตำแหน่งต่างๆ บนโลกเป็นค่า ละติจูด ลองจิจูด มี Library พร้อมใช้งาน รองรับไฟ 3-5 โวลต์การใช้งานโมดูล GPS ครั้งแรก (Cold Start) จะต้องรอถือสัญญาณประมาณ 15 นาที ระหว่างถือสัญญาณครั้งแรกต้องเอาโมดูลภายนอกอาคาร หากถือสัญญาณได้แล้วไฟ Led จะกะพริบ U-BLOX NEO-6M โมดูลมาพร้อมกับเสาอากาศเซรามิกประสิทธิภาพสูงซึ่งเทียบเท่ากับเสาอากาศที่ใช้ภายในตัว โมดูลมาพร้อมกับ EEPROM ข้อมูลการกำหนดค่าทั้งหมดสามารถเก็บไว้ใน EEPROM การกำหนดค่าที่หลากหลายเพื่อตอบสนองความต้องการของคุณ โมดูลยังมาพร้อมกับแบตเตอรี่สำรองแบบชาร์จไฟได้ ( เพื่อรองรับการ warm start หลังจากปิดแหล่งจ่ายไฟหลักแล้ว พลังงานแบตเตอรี่สำรองสามารถรักษาข้อมูลตัวรับสัญญาณ GPS ไว้ได้ครึ่งชั่วโมง ) [29]-[31]

#### 2.13.6 Raspberry Pi 3 Model B

บอร์ด Raspberry Pi 3 Model B จาก Raspberry Pi Foundation ใช้ซีพียู Broadcom BCM2837 64-bit Quad-Core ARM Cortex-A53 ARMv8 ความเร็ว 1.2 GHz มีหน่วยความจำ LPDDR 2 SDRAM ขนาด 1 GB ชิพ Broadcom BCM43438 เป็น Wi-Fi 802.11 b/g/n และ Bluetooth 4.1 (Classic and Low-Energy) พร้อมสายอากาศแบบ Chip Antenna บนบอร์ด [32]



รูปที่ 2.23 Raspberry Pi 3 Model B

คุณสมบัติของ Pi 3:

- ชิพเซ็ต Broadcom BCM2837 ทำงานที่ความเร็ว 1.2 กิกะเฮิร์ตซ์
- ARM Cortex-A53 แบบควอดคอร์ 64 บิต
- แลนไร้สาย 802.11b/g/n
- บลูทูธ 4.1 (คลาสสิกและพลังงานต่ำ)
- หน่วยความจำ 1GB LPDDR2
- รองรับการกระจาย ARM GNU/Linux และ Windows 10 IoT ล่าสุดทั้งหมด
- ขั้วต่อ microUSB สำหรับแหล่งจ่ายไฟ 2.5 แอมป์
- พอร์ตอีเธอร์เน็ต 10/100 จำนวน 1 พอร์ต
- ขั้วต่อวิดีโอ/เสียง HDMI จำนวน 1 พอร์ต
- ขั้วต่อวิดีโอ/เสียง RCA จำนวน 1 ช่อง
- พอร์ต USB 2.0 จำนวน 2 ช่อง
- พิน GPIO จำนวน 40 พิน
- ชิพเสาอากาศ
- ขั้วต่อ DSI
- ช่องเสียบการ์ด microSD
- ขนาด: 85 x 56 x 17 มิลลิเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.13.7 The SparkFun RTK Surveyor

SparkFun RTK Surveyor เป็นตัวรับสัญญาณ GNSS ที่ใช้งานง่ายสำหรับการวางตำแหน่งระดับเซนติเมตร เหมาะสำหรับการสำรวจ อุปกรณ์ที่ตั้งโปรแกรมไว้ล่วงหน้ายังสามารถใช้สำหรับการขับเคลื่อนอัตโนมัติ การนำทาง การติดตามทรัพย์สิน และแอปพลิเคชันอื่นๆ ที่มองเห็นท้องฟ้าได้ชัดเจน RTK Surveyor ยังสามารถใช้เป็นสถานีฐานได้อีกด้วย เพียงกดสวิตช์ เครื่องสำรวจ RTK สองตัวก็สามารถใช้สร้างระบบ RTK ที่มีความแม่นยำในการวางตำแหน่งแนวอนที่ 14 มิลลิเมตร. การเชื่อมต่อ Bluetooth® ในตัวผ่าน ESP32 WROOM ทำให้ผู้ใช้สามารถใช้ RTK Surveyor กับแอปพลิเคชัน GIS ที่เลือกได้บนโทรศัพท์หรือแท็บเล็ต แบตเตอรี่ในตัวช่วยให้ใช้งานภาคสนามได้นานถึงสี่ชั่วโมง และเข้ากันได้กับแบตเตอรี่แบบ USB ทั่วไป

RTK Surveyor ทำงานร่วมกับซอฟต์แวร์ GIS ทั่วไปสำหรับ Android และ iOS รวมถึง SW Maps Android / iOS, Field Genius, SurvPC, Survey Master, Vespucci, QGIS, QField และซอฟต์แวร์ GIS ใด ๆ ที่รองรับ NMEA ผ่าน Bluetooth



รูปที่ 2.24 The SparkFun RTK Surveyor

อุปกรณ์นี้สามารถใช้ได้ในสี่โหมด:

1. การวางตำแหน่ง GNSS (ความแม่นยำ ~ 30 ซม.) : ในโหมดตำแหน่ง อุปกรณ์จะรับสัญญาณ L1/L2 จากเสาอากาศที่ผู้ใช้จัดเตรียมให้ และตัวรับสัญญาณ GNSS คุณภาพสูงจะให้ค่าละติจูด/ลองจิจูดและความสูงด้วยความแม่นยำประมาณ 300 มิลลิเมตร.
2. การวางตำแหน่ง GNSS ด้วย RTK (ความแม่นยำ 1.4 ซม.) : ในโหมดการวางตำแหน่งด้วย RTK อุปกรณ์จะรับสัญญาณ L1/L2 จากเสาอากาศและข้อมูลการแก้ไขจากสถานีฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลการแก้ไขสามารถรับได้จากลิงก์มือถือไปยังแหล่งแก้ไขออนไลน์ หรือผ่านลิงก์วิทยุไปยังการตั้งค่า RTK Surveyor ครั้งที่ 2 เป็นสถานีฐาน

3. สถานีฐาน GNSS : ในโหมดสถานีฐาน อุปกรณ์จะติดตั้งอยู่ในตำแหน่งชั่วคราว (เช่น ขาดังกล้าง) และเริ่มส่งข้อมูลการแก้ไขผ่านการเชื่อมต่อวิทยุหรืออินเทอร์เน็ต ฐานมักจะใช้ร่วมกับหน่วยที่สองที่ตั้งค่าเป็น 'การวางตำแหน่งด้วย RTK' เพื่อให้ได้ความแม่นยำสัมพัทธ์ 14 มิลลิเมตร

4. เซิร์ฟเวอร์ NTRIP สถานีฐาน GNSS : ในโหมดเซิร์ฟเวอร์สถานีฐาน NTRIP เสภาอากาศจะถูกติดตั้งในตำแหน่งกิ่งหรือถาวร (เช่น หลังคา) สายเคเบิลจะเดินในอาคาร และอุปกรณ์จะเชื่อมต่อผ่าน WiFi เพื่อส่งข้อมูลการแก้ไขไปยังล้อ NTRIP เพื่อให้ผู้ใช้สามารถเข้าถึง ข้อมูลการแก้ไขผ่านการเชื่อมต่อมือถือหรืออินเทอร์เน็ต ฐานประเภทนี้เป็นวิธีที่ง่ายมากในการตั้งค่าแหล่งที่มาของการแก้ไขสัมบูรณ์ที่แม่นยำมาก

RTK Surveyor มีสายเคเบิลให้มาสองเส้น ช่วยให้ผู้ใช้สามารถเสียบวิทยุ Serial Telemetry ที่ใช้งานง่ายของเราหรือลิงค์วิทยุของตนเองได้ หากแหล่งกำเนิดการแก้ไขในพื้นที่อยู่ภายในรัศมี 10 กม. ผู้ใช้ยังสามารถใช้โทรศัพท์เพื่อให้ข้อมูลการแก้ไขผ่านลิงก์ Bluetooth® ได้ (ไม่จำเป็นต้องใช้คลื่นวิทยุภายนอก) [33]

### 2.13.8 RTL-SDR DONGLE

RTL-SDR Dongle คือ อุปกรณ์ SDR ที่มีราคาถูก นำมาใช้ในการทำอุปกรณ์รับสัญญาณทีวีระบบ DVB-T โดยอุปกรณ์แบบ Dongle ใช้ Chipset RTL2832U ด้วยหลักการที่ว่าใช้ซอฟต์แวร์ติดต่อสัญญาณข้อมูล I/O โดยตรงทำให้อุปกรณ์รับสัญญาณทีวีระบบ DVB-T กลายเป็น wideband software defined radio โดย RTL-SDR Dongle [34]



รูปที่ 2.25 RTL-SDR Dongle

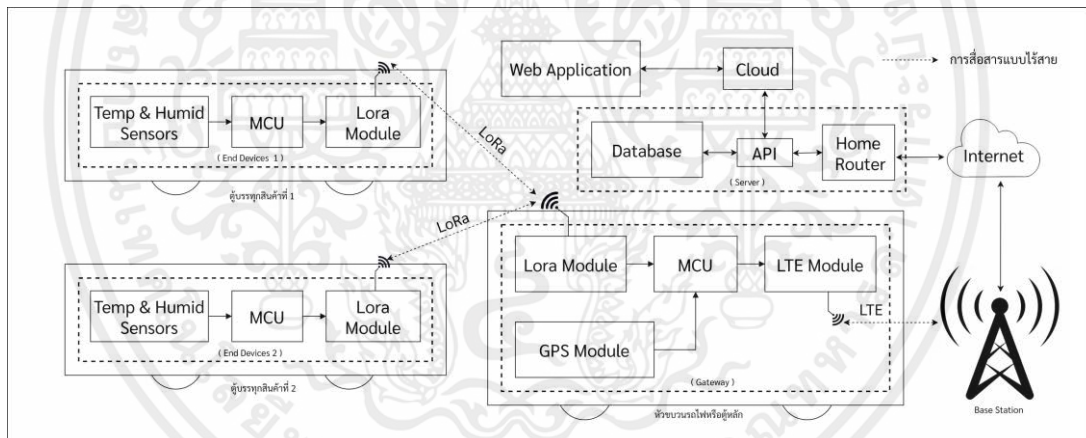
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การออกแบบและการจัดทำปริญญานิพนธ์

#### 3.1 การออกแบบภาพรวมโครงการ

โครงการนี้จัดทำขึ้นโดยมีวัตถุประสงค์ เพื่อสร้างระบบติดตามและรายงานสภาพอากาศแบบบูรณาการด้วย LoRa และ GPS ซึ่งการทำงานของระบบติดตามและรายงานสภาพอากาศนี้ประกอบไปด้วย 3 ส่วนประกอบหลัก ได้แก่ ส่วนของ End Devices ส่วนของ Gateway ส่วนของ Monitoring System โดยของ End Devices จะมีเซ็นเซอร์วัดอุณหภูมิและเซ็นเซอร์วัดความชื้นทำการส่งข้อมูลโดยใช้โมดูล LoRa ย่านความถี่ 900-931 เมกะเฮิรตซ์ ไปที่ Gateway โดย Gateway จะทำหน้าที่หาตำแหน่งปัจจุบันและส่งข้อมูลที่รับมาจาก End Devices ไปยังเซิร์ฟเวอร์ฐานข้อมูลผ่านเครือข่าย LTE จากนั้นจะมีการประมวลผลข้อมูลและแสดงผลบนหน้าเว็บไซต์ข้อมูลนี้สามารถนำไปใช้เพื่อติดตามและควบคุมคุณภาพของการขนส่งได้



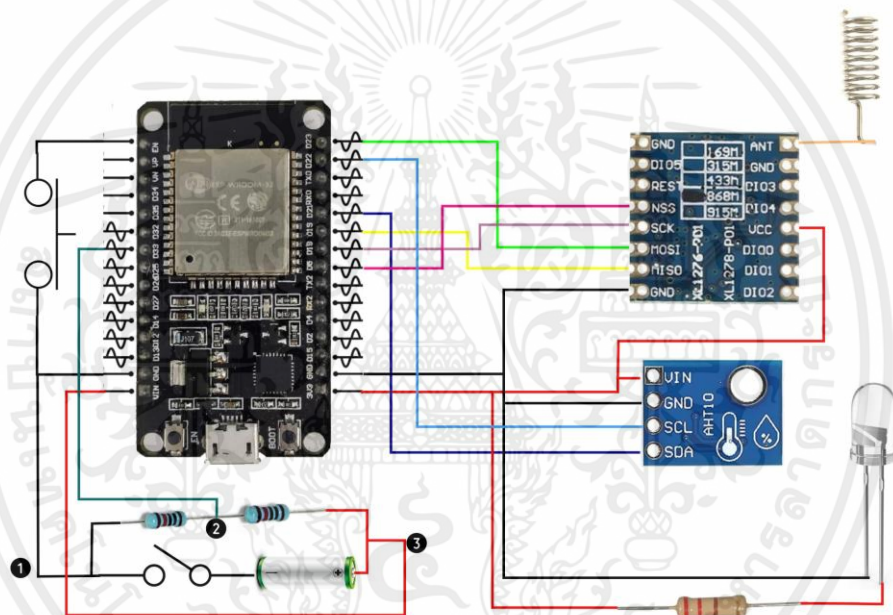
รูปที่ 3.1 บล็อกไดอะแกรมของระบบติดตามและรายงานสภาพแวดล้อมแบบบูรณาการด้วย LoRa และ GPS สำหรับการขนส่งทางรถไฟ

ในภาคการศึกษา 2/2566 ส่วนของ End Devices ที่ทำหน้าที่เก็บข้อมูลที่ได้จากออกแบบอัลกอริทึม และได้เขียนชุดคำสั่งในการส่งข้อมูลจากเซ็นเซอร์ จากนั้นส่งข้อมูลไปยัง Gateway เพื่อรวบรวมข้อมูลที่ได้จากแบบจำลอง เพื่อใช้ในการวิเคราะห์และคาดการณ์เพื่อติดตามและควบคุมคุณภาพของการขนส่งได้ เช่น ตรวจสอบอุณหภูมิและความชื้นของสินค้าในตู้คอนเทนเนอร์บนรถไฟหรือรถบรรทุกตู้สินค้าได้อย่างมีประสิทธิภาพ

## 3.2 การออกแบบในส่วนของอุปกรณ์ทางฝั่ง End Devices

### 3.2.1 การออกแบบ และสร้างแบบวงจรในส่วนของอุปกรณ์ทางฝั่ง END DEVICES

การศึกษา ออกแบบ และสร้างอุปกรณ์ทางฝั่ง End Devices ซึ่งประกอบไปด้วยบอร์ด ESP32 ร่วมกับ LoRa Module ย่านความถี่ 923 เมกะเฮิรตซ์ พร้อมทั้งเซนเซอร์ AHT10 I2C ที่ทำหน้าที่วัดอุณหภูมิและความชื้น ของภาคส่ง และมี Diode LED ที่แสดงค่าสถานะตอนส่ง และปุ่มกดที่ทำหน้าที่ Reset การทำงาน แสดงดังรูปที่ 3.2



รูปที่ 3.2 แบบจำลองวงจร End Devices

ตารางที่ 3.1 การเชื่อมต่อบอร์ด ESP32 ร่วมกับ เซนเซอร์ AHT10

ESP32	AHT10 I2C
3.3V	Vin
GND	GND
D21	SDA
D22	SCL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางที่ 3.1 แสดงการเชื่อมต่อของโมดูล LoRa ในย่านความถี่ 923 เมกะเฮิรตซ์ ร่วมกับบอร์ด ESP32 และเซนเซอร์ AHT10 ซึ่งเซนเซอร์ AHT10 เป็นเซนเซอร์วัดอุณหภูมิและความชื้นรุ่นใหม่ ออกแบบมาให้มีขนาดเล็ก สื่อสารกับ MCU แบบ I2C ได้ง่ายใช้ไฟเลี้ยง 1.8-6V

ตารางที่ 3.2 การเชื่อมต่อบอร์ด ESP32 ร่วมกับ LoRa Module SX1276

ESP32	SX1276 Pins
GND	GND
3.3V	VCC
D5	NSS
D18	SCK
D19	MISO
D23	MOSI

ตารางที่ 3.3 การเชื่อมต่อบอร์ด ESP32 ร่วมกับ Diode LED

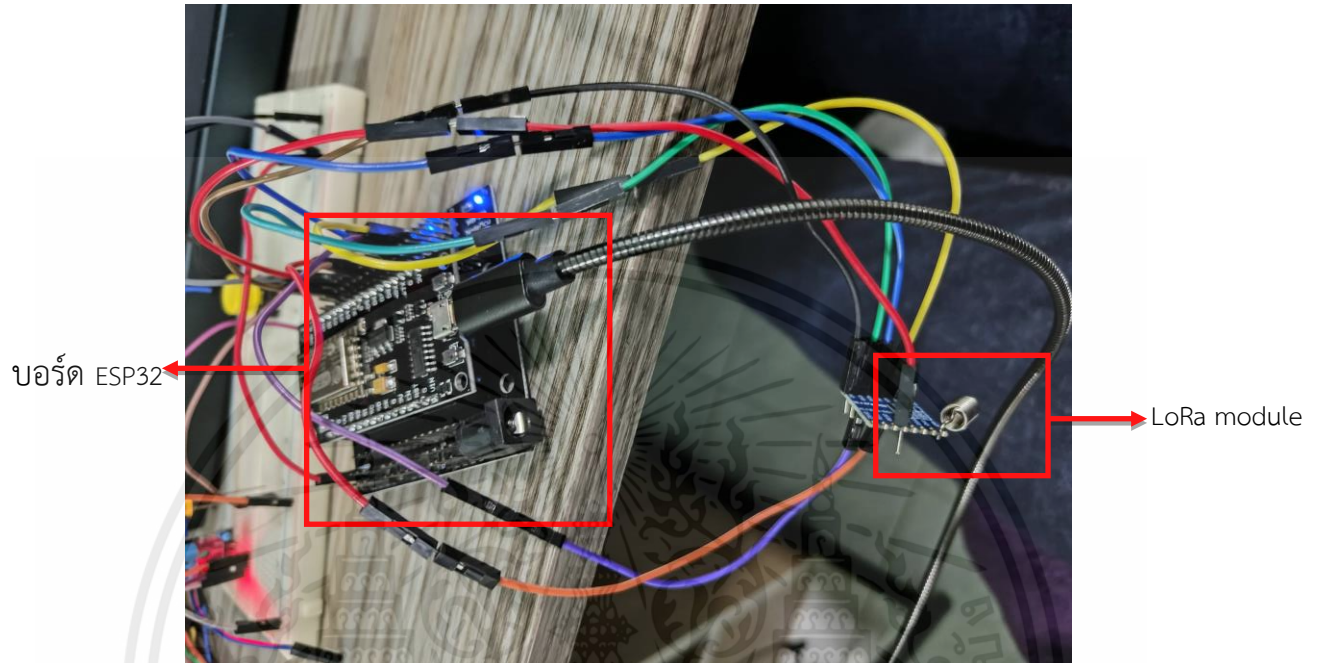
ESP32	Diode LED
D26	ขา +
GND	ขา -

ตารางที่ 3.4 การเชื่อมต่อบอร์ด ESP32 ร่วมกับ Battery

ESP32	Diode LED
Vin	หมายเลข 3
GND	หมายเลข 1
D33	หมายเลข 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

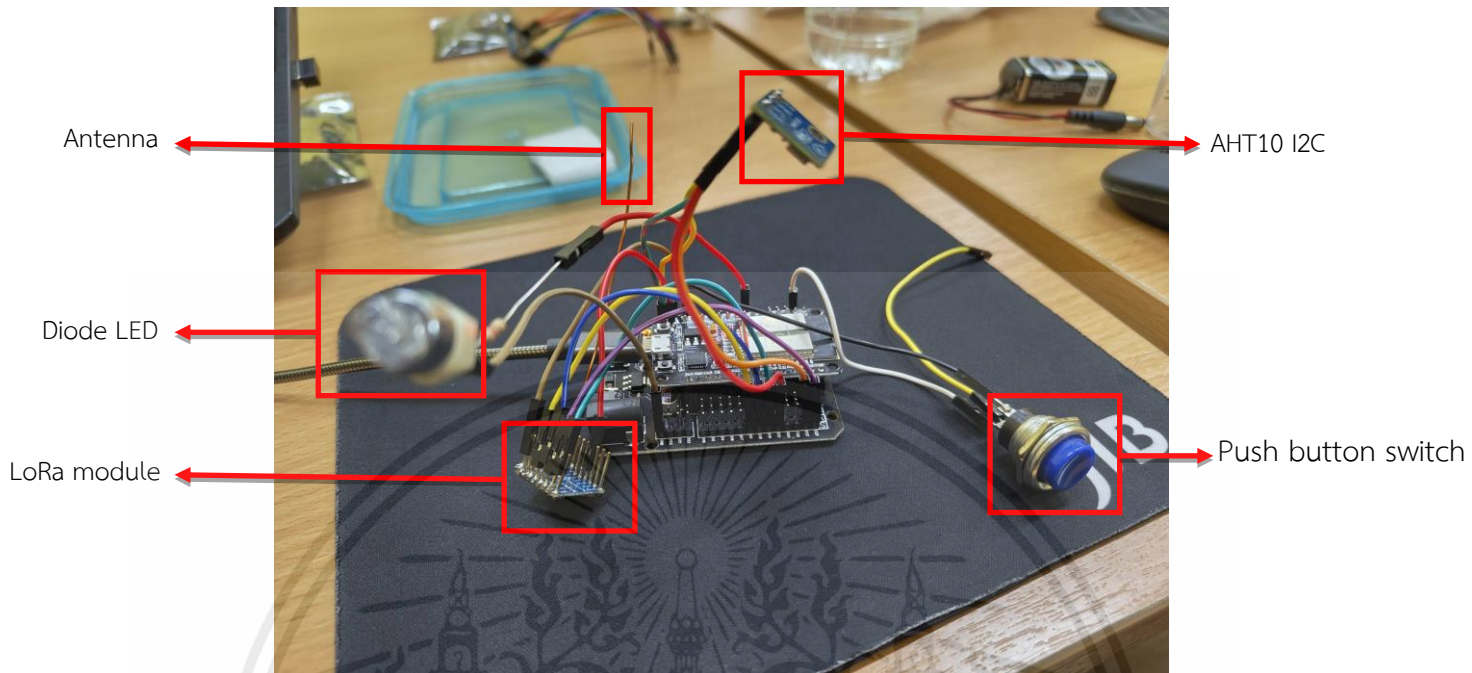
1. เชื่อมต่อบอร์ด ESP32 ร่วมกับ LoRa Module ในย่านความถี่ 923 เมกะเฮิร์ตซ์ แสดงดังรูปที่ 3.3



รูปที่ 3.3 บอร์ด ESP32 ร่วมกับ LoRa Module ในย่านความถี่ 923 เมกะเฮิร์ตซ์

2. เมื่อทำการกำหนดขาการเชื่อมต่อเซนเซอร์ AHT10 I2C และ Diode LED และการเชื่อมต่อโมดูล LoRa กับบอร์ด ESP32 แล้วเซนเซอร์จะทำการรับข้อมูลจากแบบจำลองและส่งต่อข้อมูลผ่าน LoRa ในย่านความถี่ 923 เมกะเฮิร์ตซ์ โดยการรับค่าจะแสดงดังรูปที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 ขาการเชื่อมต่อเซนเซอร์ AHT10 I2C และ Diode LED ทำการรับข้อมูลจากแบบจำลอง และส่งต่อข้อมูลผ่าน LoRa ในย่านความถี่ 923 เมกะเฮิร์ตซ์

3. นำอุปกรณ์ทั้งหมดดังรูปที่ 6 มาประกอบลงกล่อง



รูปที่ 3.5 กล่องอุปกรณ์ End Device หลังประกอบอุปกรณ์ทั้งหมดลงกล่อง

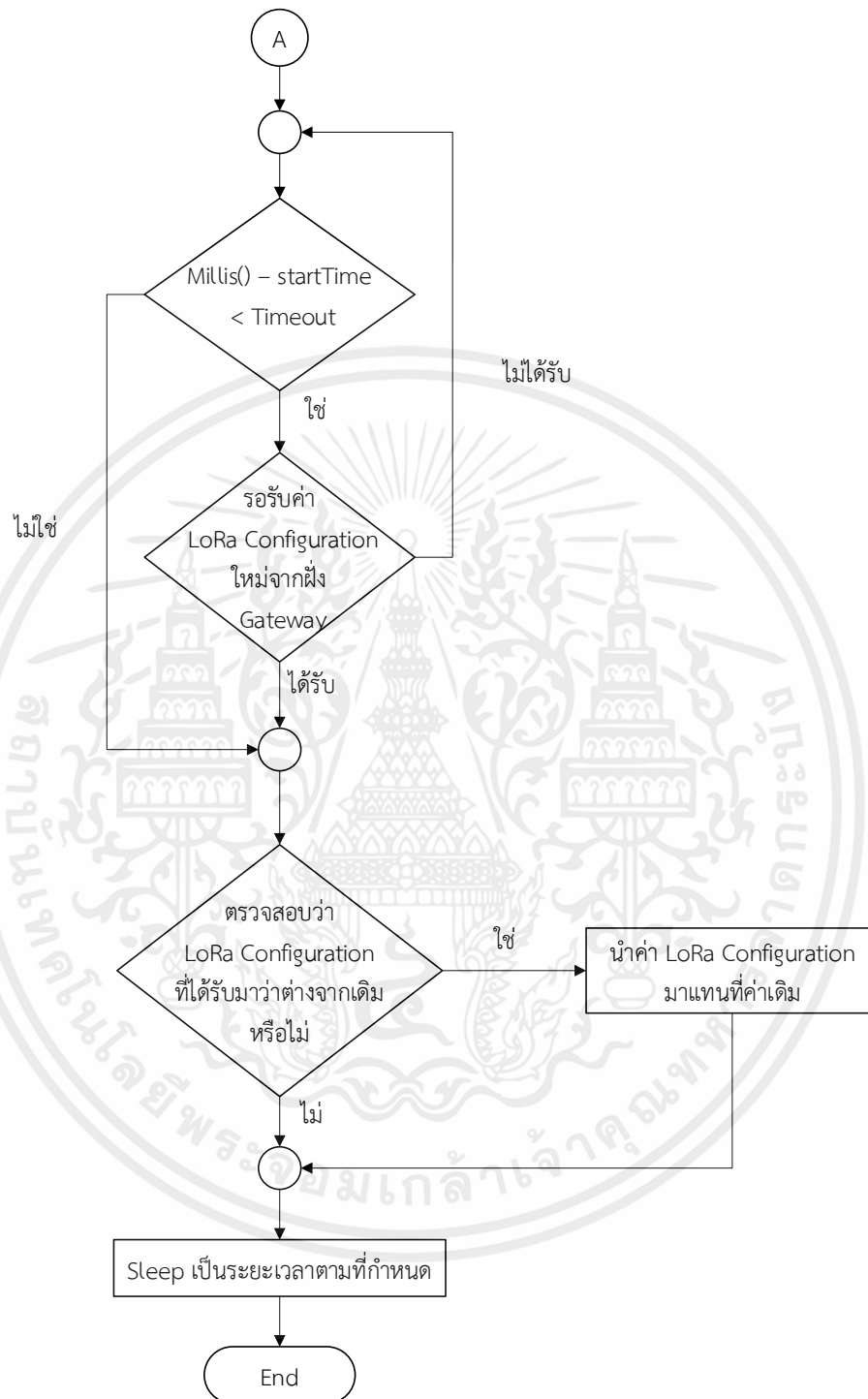
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.2 การออกแบบโปรแกรมสำหรับส่งค่า และเก็บค่าจากเซนเซอร์ด้วยโมดูล LoRa ในย่านความถี่ 923 เมกะเฮิรตซ์

ในส่วนนี้ทำการเก็บค่าจากเซนเซอร์และส่งไปยังโมดูล LoRa ในย่านความถี่ 923 เมกะเฮิรตซ์ โดยการทำงานของโมดูล LoRa สามารถส่งข้อมูลจากเซนเซอร์ในระยะทางไกลแบบไร้สาย ซึ่งโมดูล LoRa มีความสามารถเป็นได้ทั้งตัวรับและตัวส่งที่ผ่านความถี่ 923 เมกะเฮิรตซ์ ซึ่งเป็นความถี่ที่ได้รับการยกเว้นให้สามารถใช้งานได้โดยไม่ต้องขออนุญาต โดยโปรแกรมสำหรับเก็บค่าข้อมูลจากเซนเซอร์ AHT10 สำหรับวัดอุณหภูมิและความชื้น แสดงดังรูปที่ 3.6



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



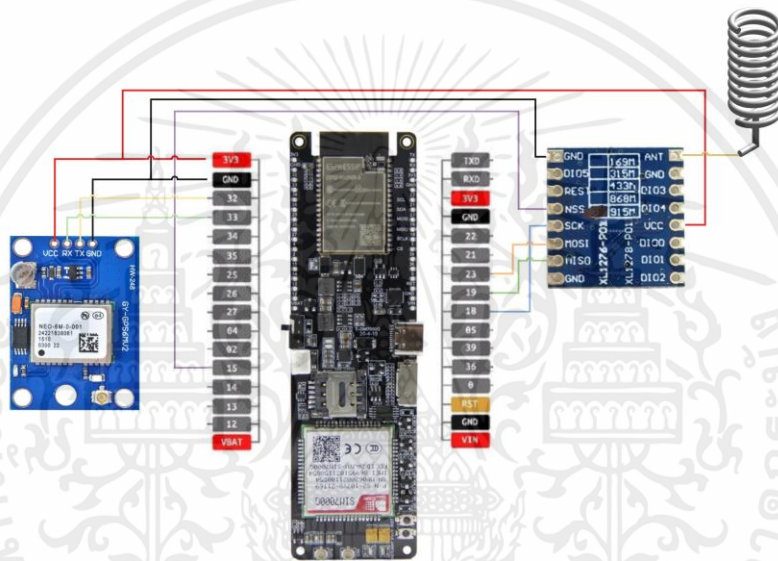
รูปที่ 3.6 แผนผังการทำงานของโปรแกรมสำหรับรับ และเก็บค่าจากเซนเซอร์ด้วยโมดูล LoRa ย่านความถี่ 923 เมกะเฮิรตซ์ กับบอร์ด ESP32 พร้อมทั้งเก็บข้อมูลในรูปแบบ JSON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 การออกแบบในส่วนของอุปกรณ์ทางฝั่ง GATEWAY

#### 3.3.1 การออกแบบ และสร้างแบบวงจรในส่วนของอุปกรณ์ทางฝั่ง GATEWAY

การศึกษาและสร้างโปรแกรมสำหรับเก็บค่าข้อมูลจาก End Devices ซึ่งมีการติดตั้ง เซนเซอร์วัดอุณหภูมิและความชื้น และทำการส่งข้อมูลที่รวบรวมมาไปที่อุปกรณ์ Gateway ผ่านระบบ LoRa ผ่านย่านความถี่ 923 เมกะเฮิรตซ์ โดยตัวอุปกรณ์ Gateway จะทำการเก็บค่าอุณหภูมิ ความชื้น และค่าพิกัด GPS เพื่อส่งข้อมูลทั้งหมดขึ้น Database ผ่านทาง Internet



รูปที่ 3.7 แบบจำลองวงจร Gateway

ในระยะเวลาที่ผ่านมา ทำการทดลองนำโมดูล GPS NEO 6M และโมดูล LoRa ย่านความถี่ 923 เมกะเฮิรตซ์ เข้าร่วมกับบอร์ด ESP32 Lilygo T-SIM7670E เพื่อเป็นการรับค่าที่ได้มาจากฝั่งส่ง End Device และทำการรับค่าสัญญาณ GPS จากดาวเทียม

ตารางที่ 3.5 การเชื่อมต่อบอร์ด ESP32 Lilygo T-SIM7670E ร่วมกับโมดูล GPS-NEO-6M

ESP32 Lilygo T-SIM7670E	GPS-NEO-6M
3.3V	VCC
GND	GND
33	Rx
32	Tx

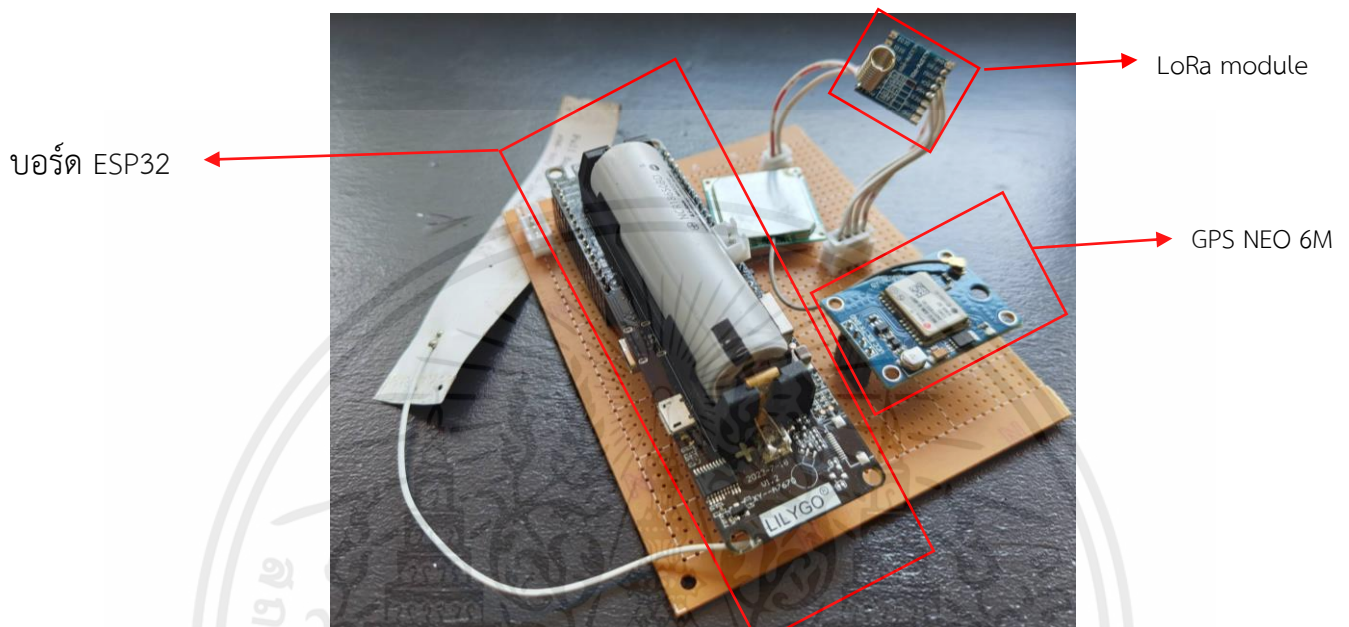
จากตารางที่ 3.5 แสดงการเชื่อมต่อของโมดูล GPS-NEO-6M ร่วมกับบอร์ด ESP32 Lilygo T-SIM7670E และเสาอากาศขนาด 25 \* 25 มิลลิเมตร เพื่อใช้ในการรับค่าสัญญาณ GPS จากดาวเทียม

ตารางที่ 3.6 การเชื่อมต่อบอร์ด ESP32 Lilygo T-SIM7670E ร่วมกับ LoRa Module SX1276

ESP32 Lilygo T-SIM7670E	SX1276 Pins
GND	GND
3.3V	VCC
D15	NSS
D18	SCK
D19	MISO
D23	MOSI

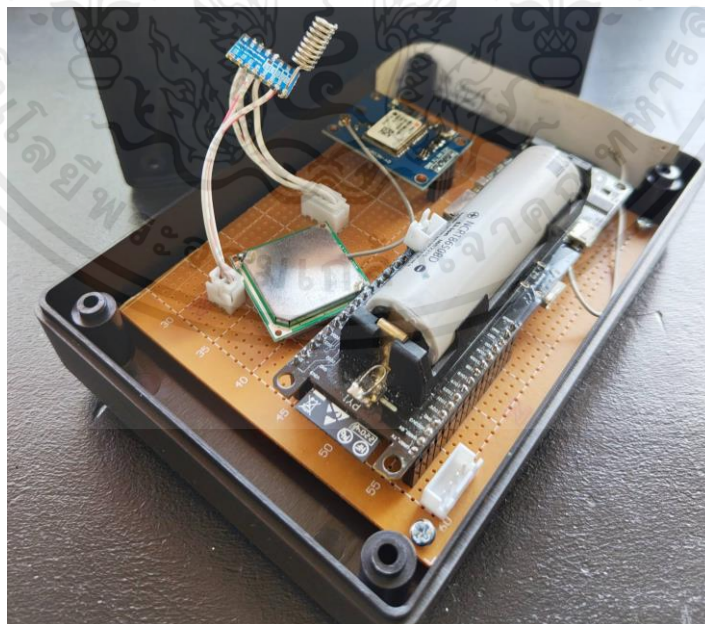
จากตารางที่ 3.6 แสดงการเชื่อมต่อของบอร์ด ESP32 Lilygo ร่วมกับโมดูล LoRa ย่านความถี่ 923 เมกะเฮิรตซ์

1. เชื่อมต่อบอร์ด ESP32 Lilygo ร่วมกับ LoRa Module ในย่านความถี่ 923 เมกะเฮิรตซ์ และโมดูล GPS NEO 6M แสดงดังรูปที่ 3.8



รูปที่ 3.8 บอร์ด ESP32 Lilygo ร่วมกับ LoRa Module และ GPS Module

2. นำอุปกรณ์ทั้งหมดดังรูปที่ 6 มาประกอบลงกล่อง แสดงดังรูปที่ 3.9



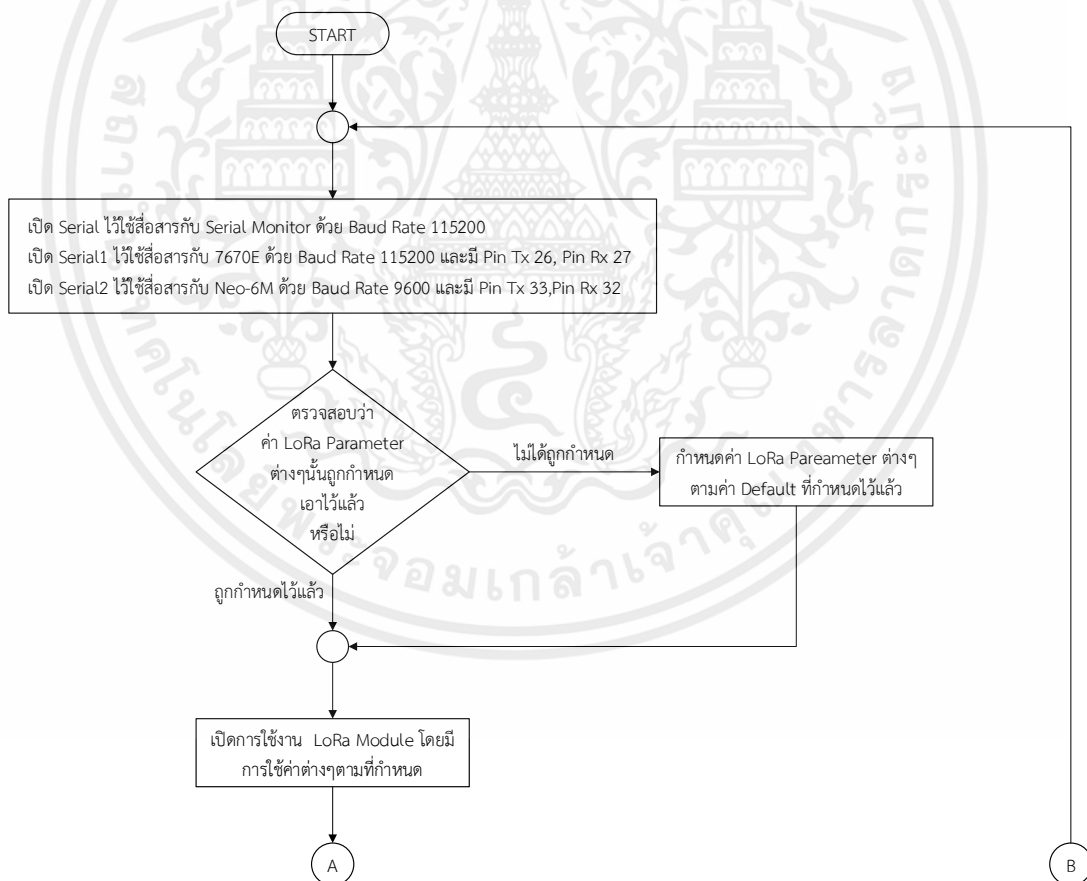
รูปที่ 3.9 กล่องอุปกรณ์ Gateway หลังประกอบอุปกรณ์ทั้งหมดลงกล่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

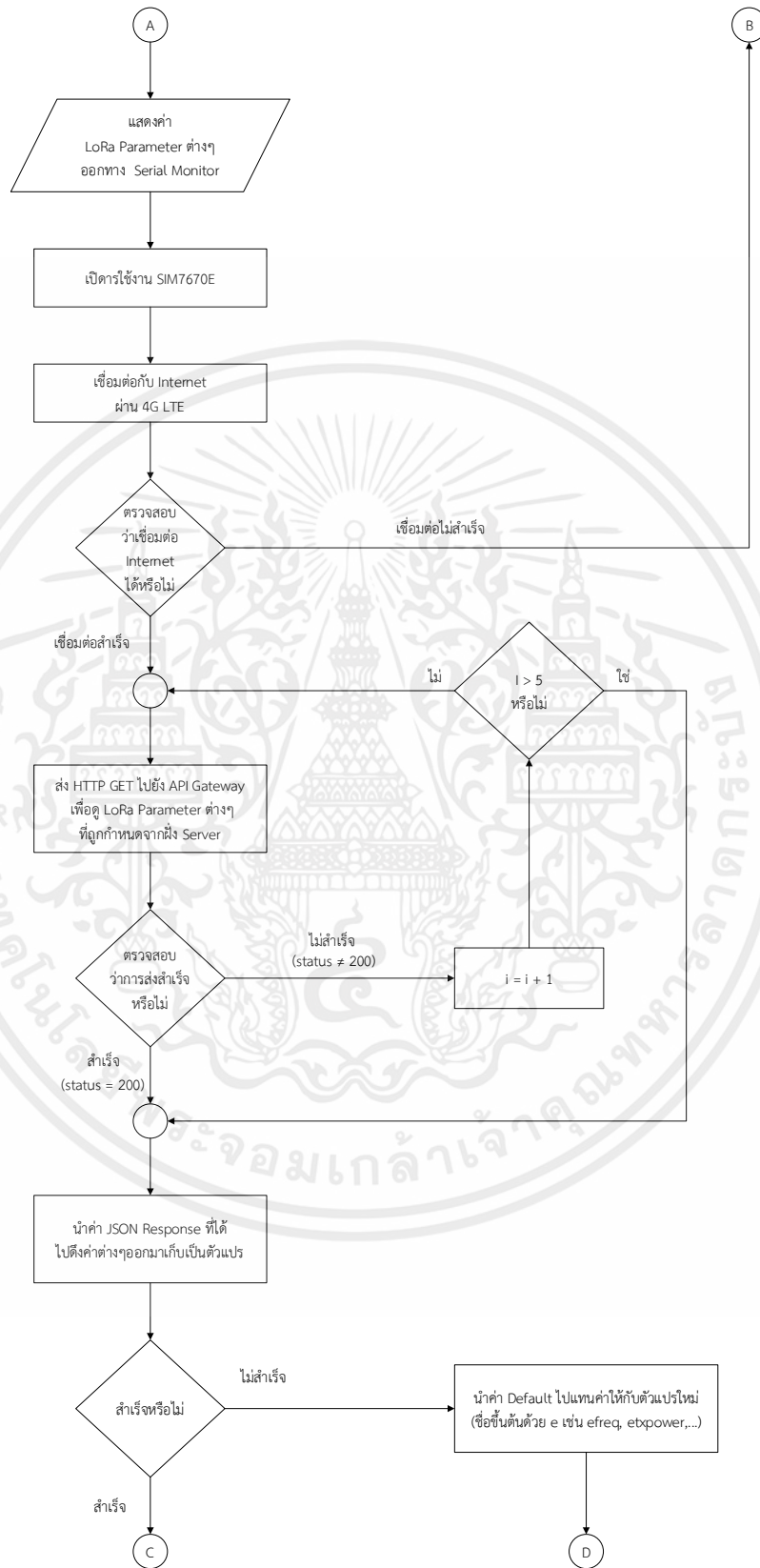
### 3.3.2 การออกแบบโปรแกรมสำหรับรับข้อมูลจาก End Devices เก็บข้อมูลจาก GPS MODULE และส่งข้อมูลขึ้น DATABASE

ในการออกแบบ Gateway ได้ทำการเขียนโปรแกรมโดยมีคำสั่งการทำงานดังนี้ เริ่มต้นด้วยการเปิด Serial เพื่อเชื่อมต่อและสื่อสารกับ module ต่างๆ และทำการเชื่อมต่อกับอินเทอร์เน็ต ต่อมาก็ทำการรับค่า LoRa Parameter ใหม่ที่ถูกส่งมา จากนั้นระบบก็จะรอรับข้อมูลที่ส่งมาจากฝั่ง End Device เมื่อได้รับข้อมูลระบบจะทำการส่งค่า LoRa configuration ใหม่กลับไปยัง End Device และทำการหาตำแหน่งปัจจุบัน ต่อมาระบบจะนำเอาข้อมูลจากทุกขั้นตอนมาจัดเรียงและส่งไปยัง Database เพื่อเก็บข้อมูลต่อไป ขั้นตอนการทำงานทั้งหมดนี้ถูกแสดงในรูปแบบของ flowchart ตามภาพด้านล่าง โดยจะแบ่งเป็นส่วนคือ setup และ loop ดังนี้

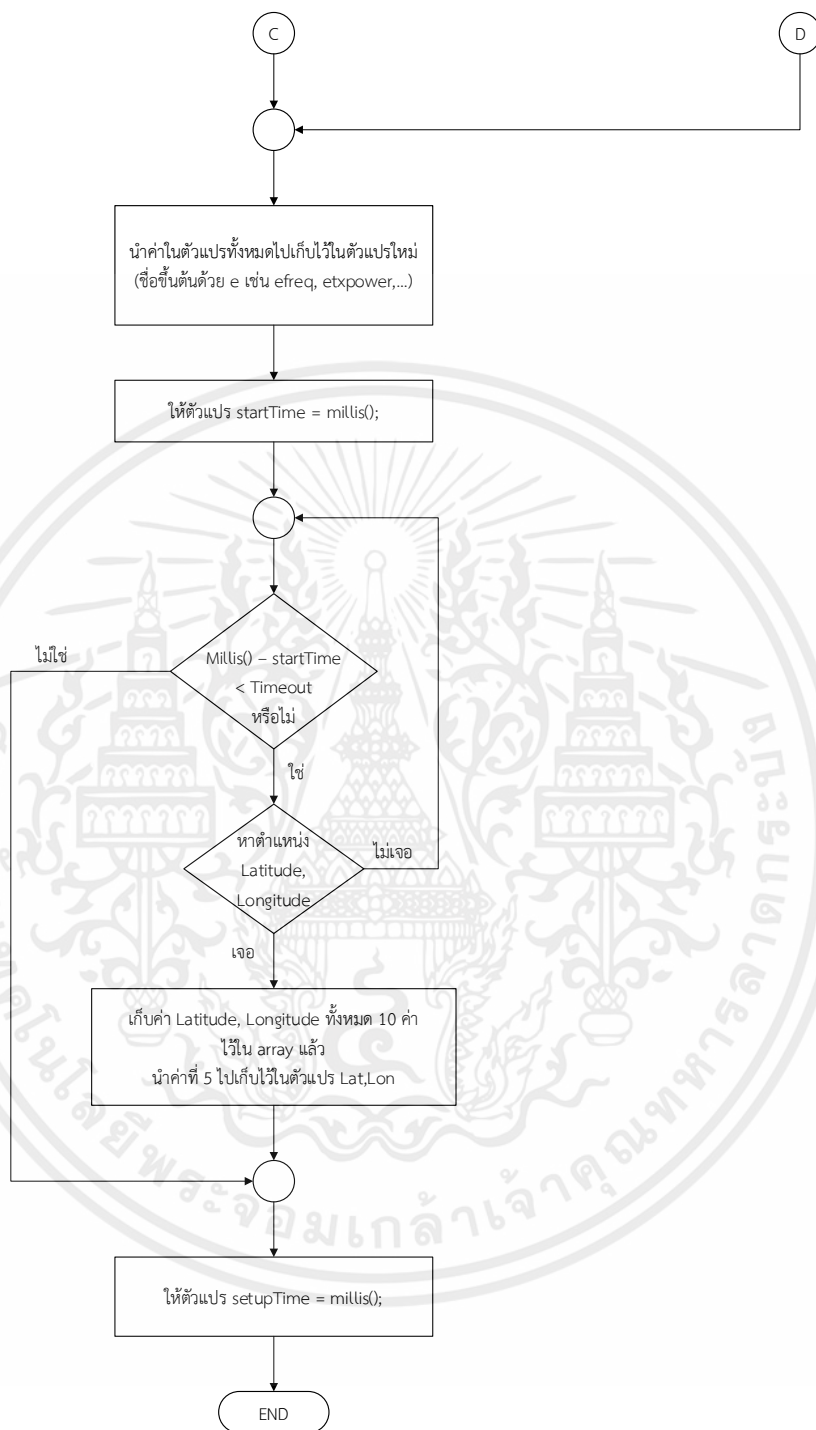
#### 3.3.2.1 แผนผังแสดงขั้นตอนการทำงานของโปรแกรมสำหรับรับค่าและเก็บค่าจากเซนเซอร์ด้วยโมดูล LoRa ในส่วนของ Setup



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



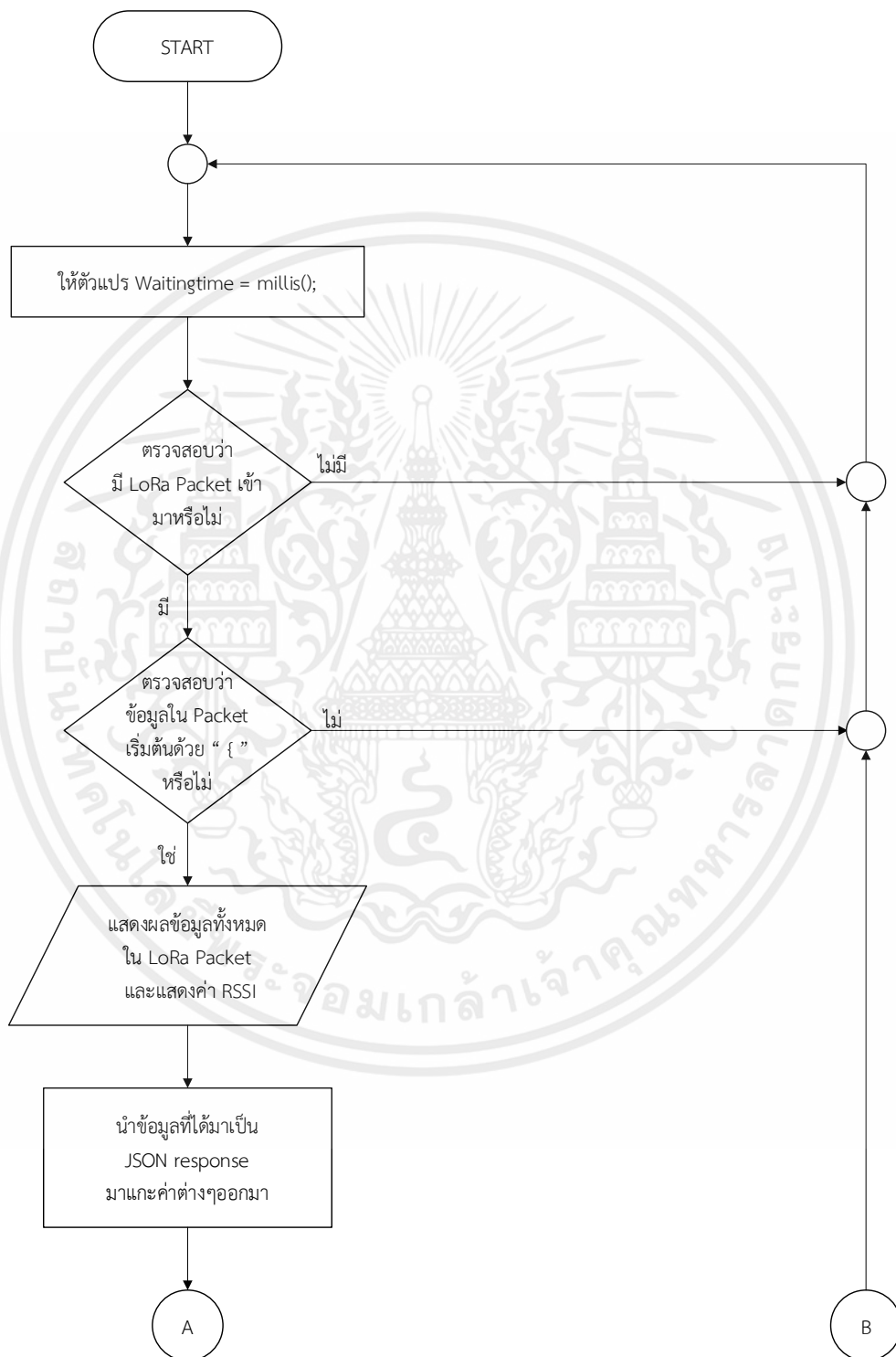
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



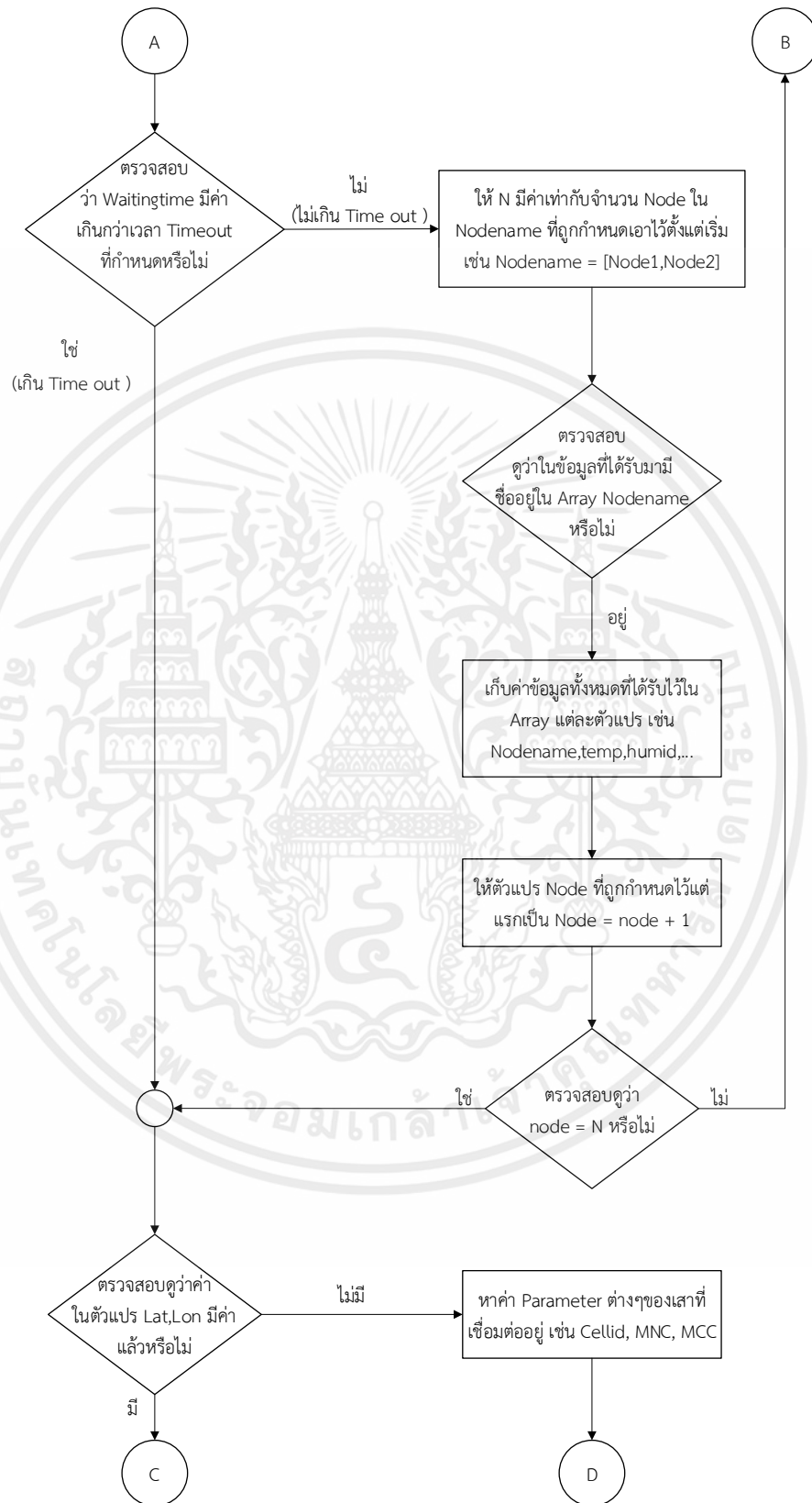
รูปที่ 3.10 แผนผังแสดงขั้นตอนการทำงานของโปรแกรมสำหรับรับค่าและเก็บค่าจากเซนเซอร์ด้วย  
โมดูล LoRa ในส่วนของ Setup

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

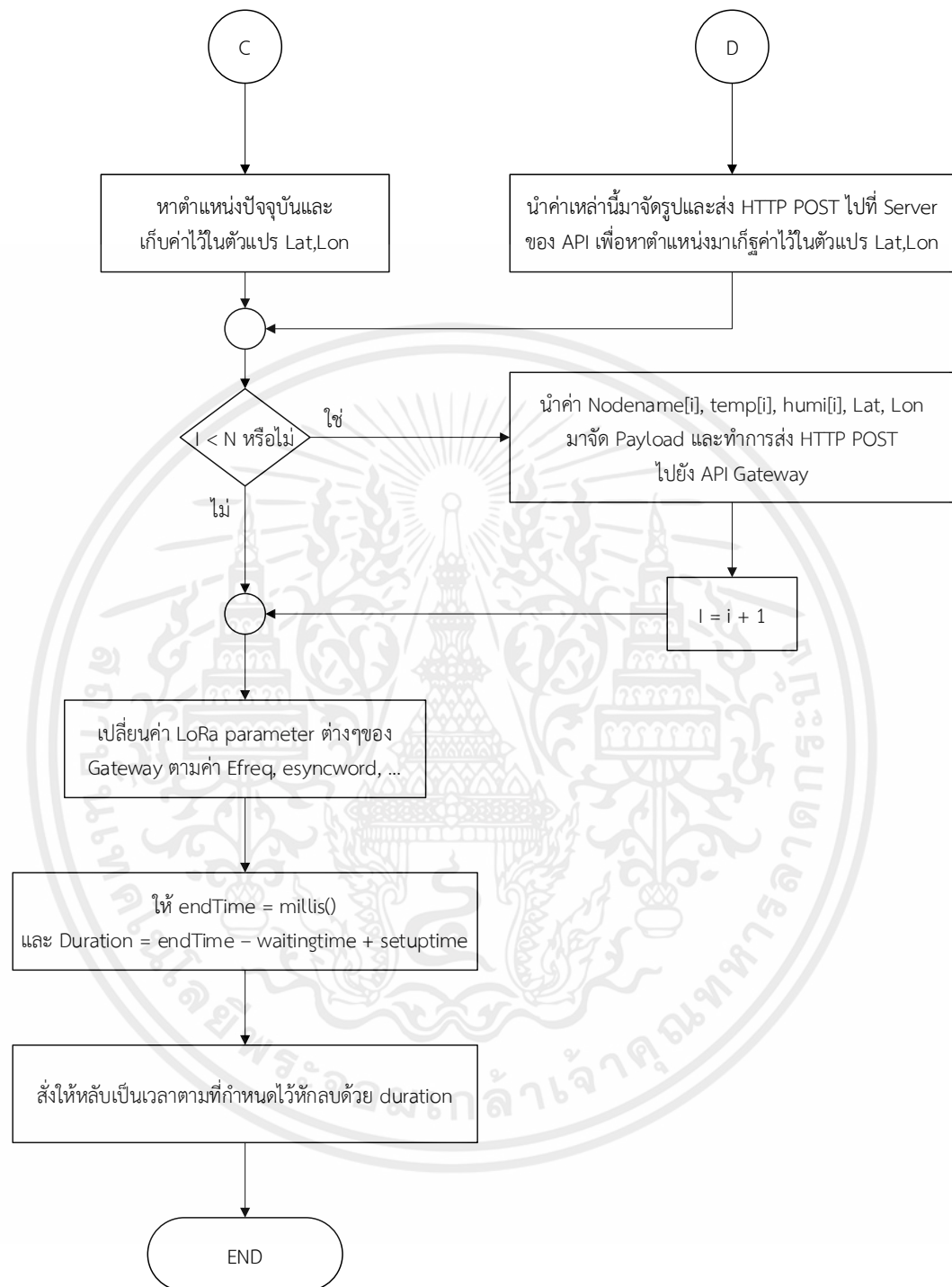
### 3.3.2.2 แผนผังแสดงขั้นตอนการทำงานของโปรแกรมสำหรับรับค่า และ เก็บค่าจากเซนเซอร์ด้วยโมดูล LoRa ในส่วนของ loop



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



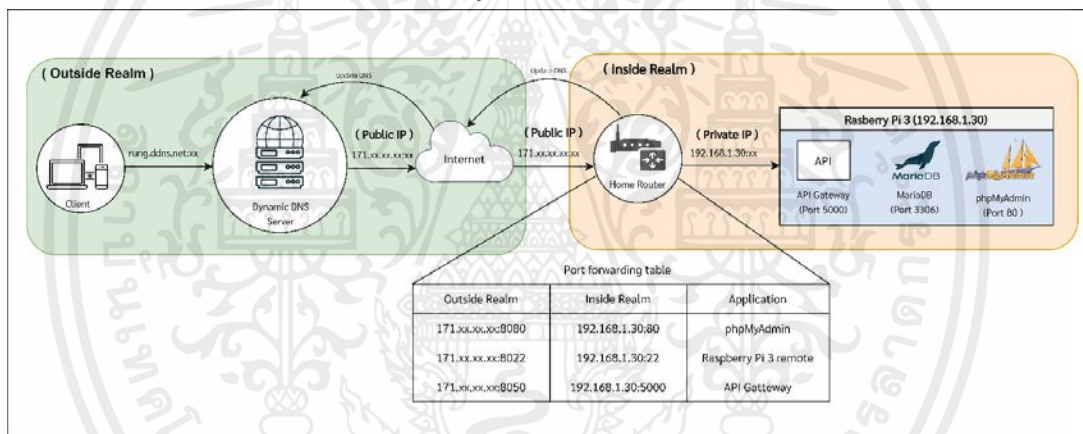
รูปที่ 3.11 แผนผังแสดงขั้นตอนการทำงานของโปรแกรมสำหรับรับค่า และเก็บค่าจากเซนเซอร์ด้วย โมดูล LoRa ในส่วนของ loop

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4 การออกแบบการทำงานของฝั่ง Server

#### 3.4.1 การออกแบบและสร้างเส้นทางการเชื่อมต่อระหว่างโลกภายนอกกับ Server

เมื่อผู้ใช้ (client) ที่อยู่เครือข่ายภายนอก (outside realm) ต้องการเชื่อมต่อกับเครือข่ายภายใน (inside realm) จะต้องเริ่มต้นจากผู้ใช้งานส่งข้อมูลไปยัง DDNS (Dynamic Domain Name System) Server เมื่อได้รับข้อมูล DDNS Server จะทำการแปลงชื่อโดเมนที่ผู้ใช้ให้มาเป็นที่อยู่ IP ของเราเตอร์ จากนั้นจะส่งข้อมูลต่อไปให้เราเตอร์ เมื่อข้อมูลมาถึงเราเตอร์ เราเตอร์จะทำการแปลงที่อยู่ IP และพอร์ตให้เป็นที่อยู่ IP และพอร์ตภายในตามตารางของ Port Forwarding Table ที่ได้กำหนดไว้ล่วงหน้า การกระทำนี้ทำให้อุปกรณ์ที่อยู่ภายในเครือข่ายสามารถติดต่อกับอุปกรณ์ภายนอกเครือข่ายได้โดยใช้ที่อยู่ IP และพอร์ตที่เราเตอร์กำหนดไว้ โดยภาพรวมของระบบและตาราง Port forwarding แสดงไว้ตามรูปที่ 3.12



รูปที่ 3.12 เส้นทางการเชื่อมต่อระหว่างโลกภายนอกกับ Server และ ตาราง Port forwarding บน Router

#### 3.4.2 การออกแบบและสร้าง Database สำหรับการเก็บข้อมูลต่างๆ

ในการเก็บข้อมูลต่าง ๆ เช่น อุณหภูมิ, ความชื้น, และตำแหน่งทางละติจูด (latitude) และลองจิจูด (longitude) เราจำเป็นต้องใช้ระบบฐานข้อมูลหรือ Database โดยในที่นี้เราจะใช้ MariaDB ซึ่งได้มีการติดตั้งอยู่บน Raspberry Pi 3 เราสามารถใช้คำสั่ง CLI (command line interface) ในการจัดการต่างๆ ได้แต่การใช้คำสั่ง CLI อาจมีความซับซ้อนและยุ่งยาก เพื่อแก้ไขปัญหานี้ เราจึงใช้ phpMyAdmin ซึ่งเป็นโปรแกรมจัดการฐานข้อมูลแบบ GUI (graphical user interface) แทนการใช้คำสั่ง CLI โดย phpMyAdmin จะช่วยให้เราสามารถจัดการฐานข้อมูล MariaDB ได้อย่างง่ายดายและสะดวกยิ่งขึ้น

```

rung@raspberrypi:~ $ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 19
Server version: 10.5.21-MariaDB-0+deb11u1 Raspbian 11

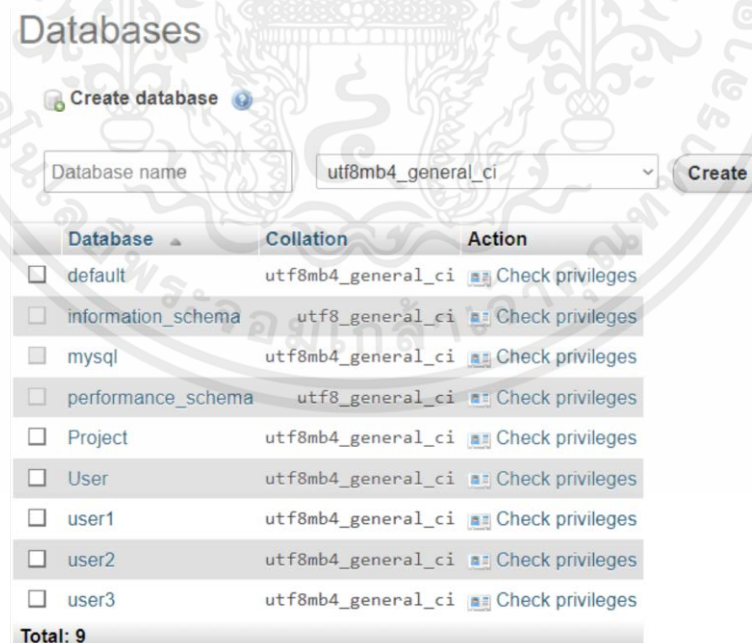
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| Project  |
| User     |
| default  |
| information_schema |
| mysql    |
| performance_schema |
| user1    |
| user2    |
| user3    |
+-----+
9 rows in set (0.002 sec)

```

รูปที่ 3.13 การใช้ CLI (command line interface) เพื่อเข้าถึงข้อมูลบน Database



รูปที่ 3.14 การใช้ phpMyAdmin เพื่อเข้าถึงข้อมูลบน Databas

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3 การออกแบบและสร้าง API เพื่อไว้ใช้จัดการกับ Database และ Gateway

ในการติดต่อและจัดการข้อมูลต่างๆ บน Database หรือการจัดการกับค่าต่างๆบน Gateway นั้น บางครั้งอาจทำตรงๆไม่ได้ เนื่องจาก Database หรือ Gateway อาจมีข้อจำกัดบางอย่าง เช่น การเข้าใช้งานอาจต้องผ่านการตรวจสอบสิทธิ์ก่อน หรือมีเงื่อนไขในการใช้งานบางอย่าง เราจึงจำเป็นต้องสร้าง API ขึ้นมาเพื่อไว้ใช้จัดการกับข้อจำกัดเหล่านั้น ในที่นี่เราจะใช้ Python และ Flask ซึ่งเป็น framework web development โดย API ที่เราสร้างจะช่วยให้เราสามารถจัดการกับข้อจำกัดต่าง ๆ ดังกล่าวได้อย่างสะดวกและง่ายมากยิ่งขึ้น เราสามารถใช้ URI (Uniform Resource Identifier) ร่วมกับแต่ละ endpoint ของ API เพื่อดำเนินการต่าง ๆ กับข้อมูลหรือค่าที่ต้องการได้ ซึ่งแต่ละ endpoint จะมีวิธีการใช้งานดังนี้

### 3.4.3.1 การบันทึกข้อมูลของแต่ละ Node

ใช้ API Endpoint: /api/data และ HTTP Method: POST เพื่อส่งข้อมูลจาก Node เข้าสู่ระบบโดยมีข้อมูลที่จำเป็นดังนี้

nodename: ชื่อของ Node ที่ส่งข้อมูล

temperature: อุณหภูมิ (องศาเซลเซียส)

humidity: ความชื้นสัมพัทธ์ (%)

latitude: ละติจูด (องศา)

longitude: ลองจิจูด (องศา)

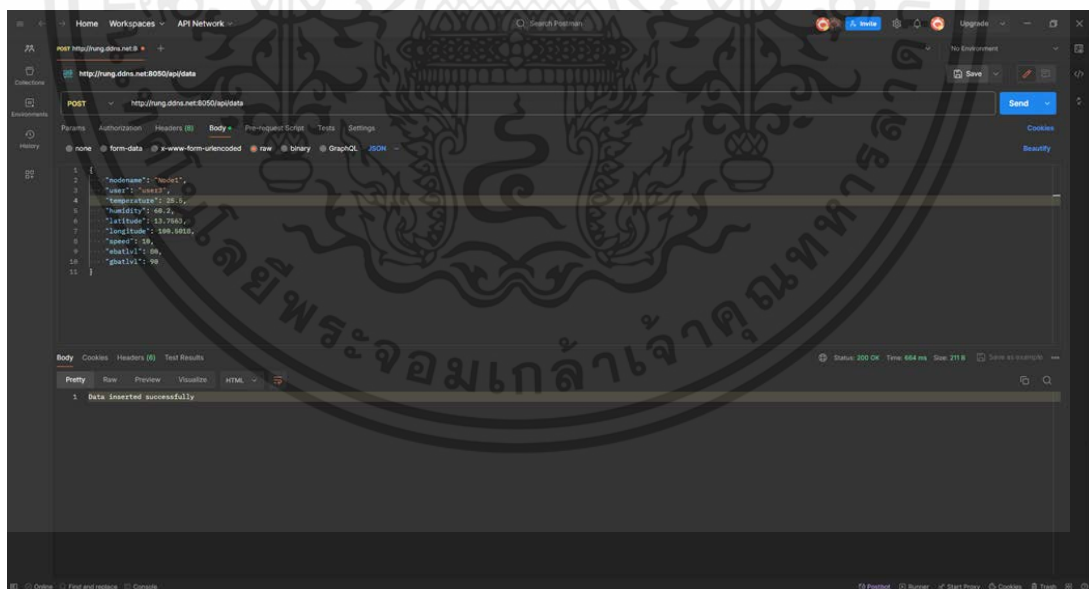
speed: ความเร็ว (กิโลเมตรต่อชั่วโมง)

ebatlvl: ระดับแบตเตอรี่ของ Node (%)

gbatlvl: ระดับแบตเตอรี่ของ Gateway (%)

และมีตัวอย่างคำสั่งการร้องขอ POST `http://runc.ddns.net:8050/api/data` Content-Type: `application/json` `{"nodename":"Node1","temperature":25.5,"humidity":60,"latitude":13.75,"longitude":100.48,"speed":0,"ebatlvl":75,"gbatlvl":90}`

และถ้าหากสำเร็จจะมีข้อความตอบกลับนี้ "Data inserted successfully" พร้อมสถานะ 200 OK

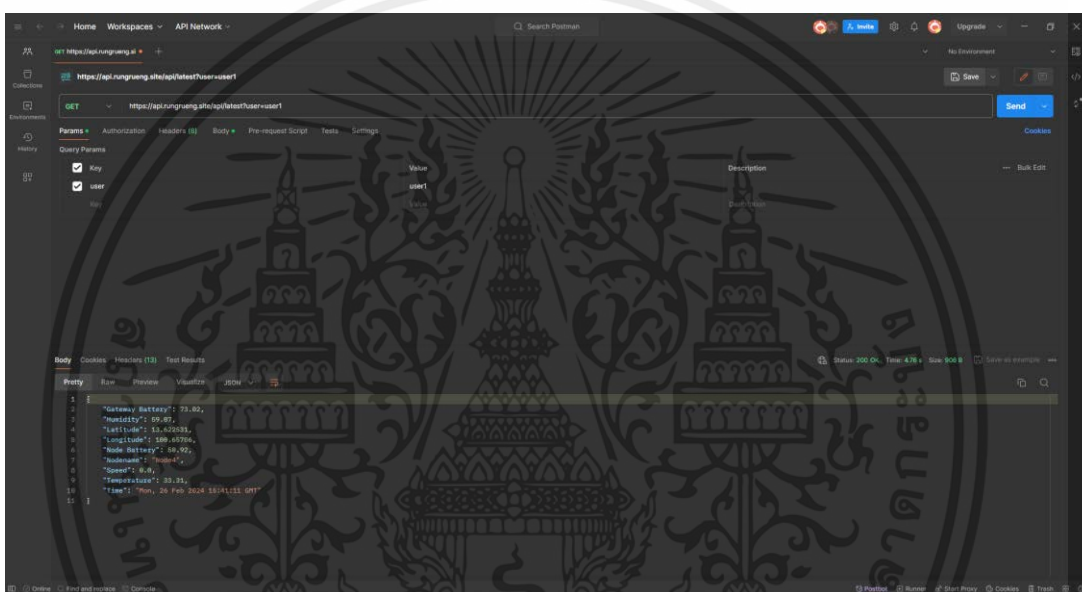


รูปที่ 3.15 การบันทึกข้อมูลของแต่ละ Node

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.2 การดึงข้อมูลล่าสุด

ใช้ API Endpoint: /api/latest และ HTTP Method: GET เพื่อดึงข้อมูลล่าสุดที่บันทึกไว้ การตอบกลับ: ระบบจะคืนข้อมูลล่าสุด 1 รายการในรูปแบบ JSON ประกอบด้วยข้อมูลทั้งหมดของ Node ที่ส่งข้อมูลมาล่าสุด โดยมีตัวอย่างการตอบกลับดังนี้ {"Time":"2024-0-16 17:38:50","Nodename":"Node1","Temperature":25.5,"Humidity":60,"Latitude":13.75,"Longitude":100.48,"Speed":0,"Node Battery":75,"Gateway Battery":90} พร้อมสถานะ 200 OK

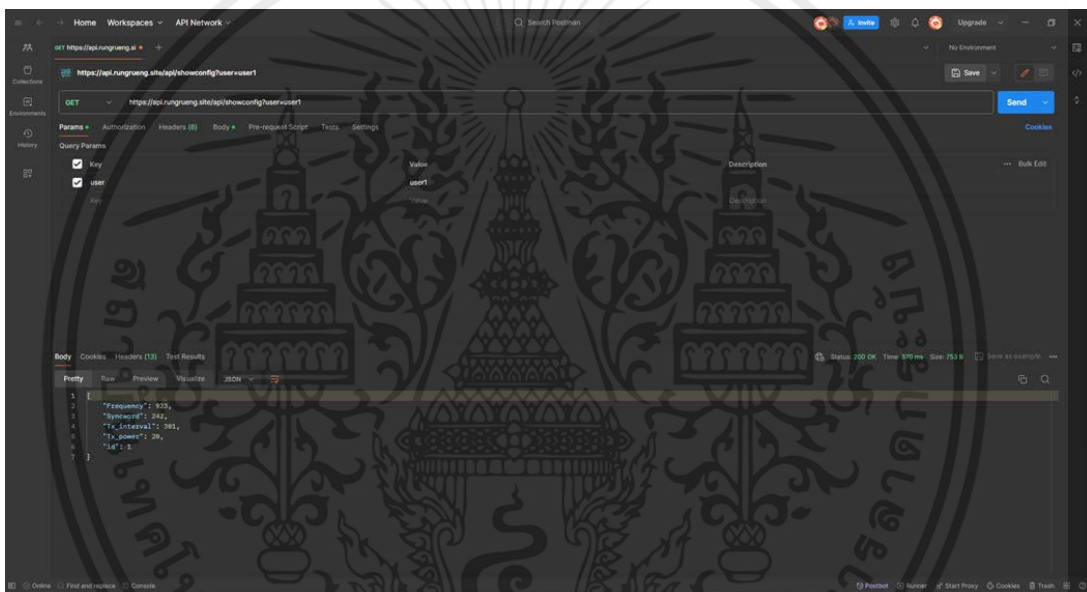


รูปที่ 3.16 การดึงข้อมูลล่าสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.3 การแสดงค่าการกำหนดค่าปัจจุบัน

ใช้ API Endpoint: /api/showconfig และ HTTP Method: GET เพื่อดึงค่าการกำหนดค่าปัจจุบันของระบบ การตอบกลับ: ระบบจะคืนข้อมูลการกำหนดค่าปัจจุบันในรูปแบบ JSON ประกอบด้วยพารามิเตอร์ต่างๆ ดังนี้ Frequency: ความถี่ในการส่งข้อมูล (หน่วย MHZ), Syncword: ค่า Syncword ที่ใช้ในการส่งข้อมูล, Tx\_interval: ระยะห่างของการส่งข้อมูลแต่ละครั้ง (หน่วยนาทีก), Tx\_power: กำลังในการส่งข้อมูล (หน่วย dBm) โดยมีตัวอย่างการตอบกลับดังนี้ {"Frequency":923,"Syncword":242,"Tx\_interval":3,"Tx\_power":20,"id":1} พร้อมสถานะ 200 OK

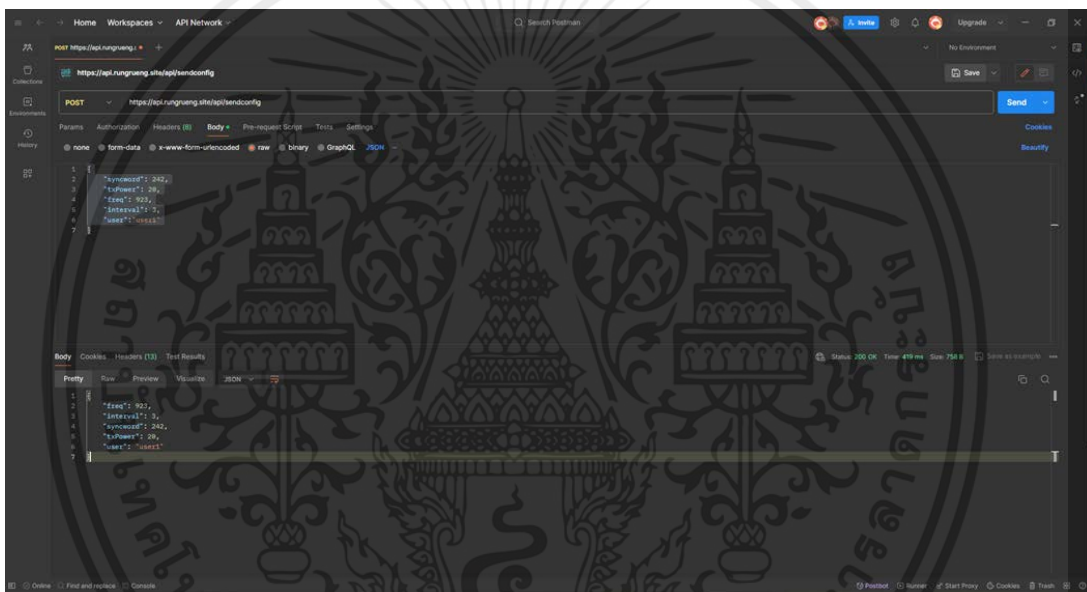


รูปที่ 3.17 การแสดงค่าการกำหนดค่าปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.4 การส่งค่าการกำหนดค่าใหม่

ใช้ API Endpoint: /api/sendconfig และ HTTP Method: POST เพื่อส่งค่าการกำหนดค่าใหม่เข้าสู่ระบบ โดยมีข้อมูลที่จำเป็นดังนี้ syncword: ค่า Syncword ใหม่, txPower: กำลังส่งข้อมูลใหม่ (หน่วย dBm), freq: ความถี่ในการส่งข้อมูลใหม่ (หน่วย MHz), interval: ระยะห่างของการส่งข้อมูลแต่ละครั้ง (หน่วยนาทีก) และมีตัวอย่างคำสั่งการร้องขอดังนี้ "POST http://rung.ddns.net:8050/api/sendconfig Content-Type: application/json {"syncword":245,"txPower":23,"freq":930,"interval":4}" และถ้าหากสำเร็จจะมีข้อความตอบกลับนี้ "Data inserted successfully" พร้อมสถานะ 200 OK

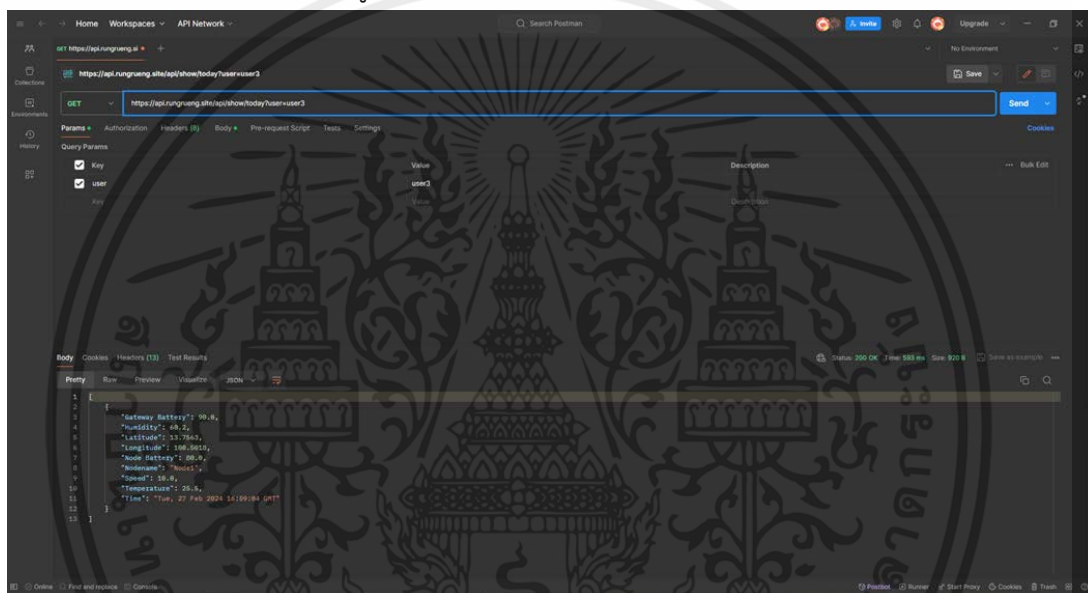


รูปที่ 3.18 การส่งค่าการกำหนดค่าใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.5 การดึงข้อมูลที่บันทึกวันนี้

ใช้ API Endpoint: /api/show/today และ HTTP Method: GET เพื่อดึงข้อมูลทั้งหมดที่บันทึกไว้ในวันนี้ การตอบกลับ: ระบบจะคืนข้อมูลที่บันทึกไว้ในวันนี้ทั้งหมดในรูปแบบ JSON เป็นรายการ โดยแต่ละรายการประกอบด้วยข้อมูลของ Gateway ทั้งหมด โดยมีตัวอย่างการตอบกลับดังนี้ [{"Time":"2024-02-16 17:38:50","Nodename":"Node1","Temperature":25.5,"Humidity":60,"Latitude":13.75,"Longitude":100.48,"Speed":0,"NodeBattery":75,"Gateway Battery":90}, ... (รายการข้อมูลอื่นๆ สำหรับวันนี้)] พร้อมสถานะ 200 OK

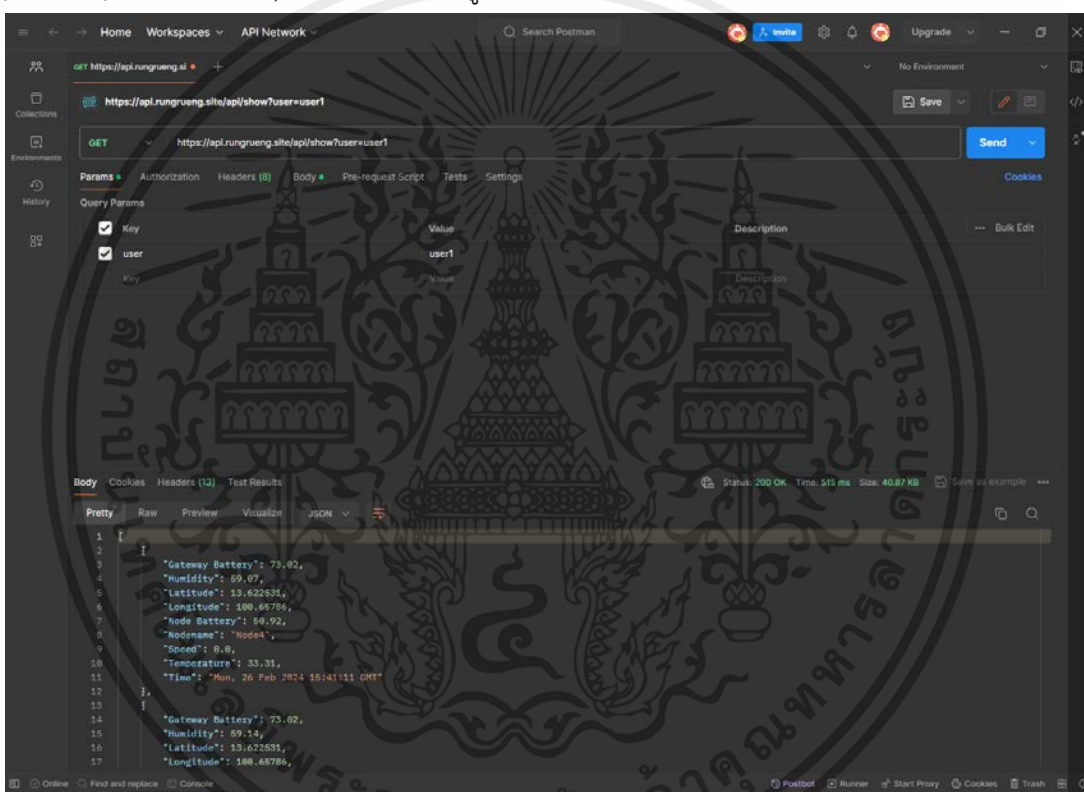


รูปที่ 3.19 การดึงข้อมูลที่บันทึกวันนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.6 การดึงข้อมูลทั้งหมดที่บันทึกไว้

ใช้ API Endpoint: /api/show และ HTTP Method: GET เพื่อดึงข้อมูลทั้งหมดที่บันทึกไว้ในระบบ การตอบกลับ: ระบบจะคืนข้อมูลทั้งหมดที่บันทึกไว้ในรูปแบบ JSON จำนวน 160 รายการล่าสุด โดยแต่ละรายการประกอบด้วยข้อมูลของ Gateway ทั้งหมด โดยมีตัวอย่างการตอบกลับดังนี้ [{"Time":"2024-02-16 17:38:50","Nodename":"Node1","Temperature":25.5,"Humidity":60,"Latitude":13.75,"Longitude":100.48,"Speed":0,"NodeBattery":75,"Gateway Battery":90}, ... (อื่นๆ 159 รายการข้อมูล)] พร้อมสถานะ 200 OK

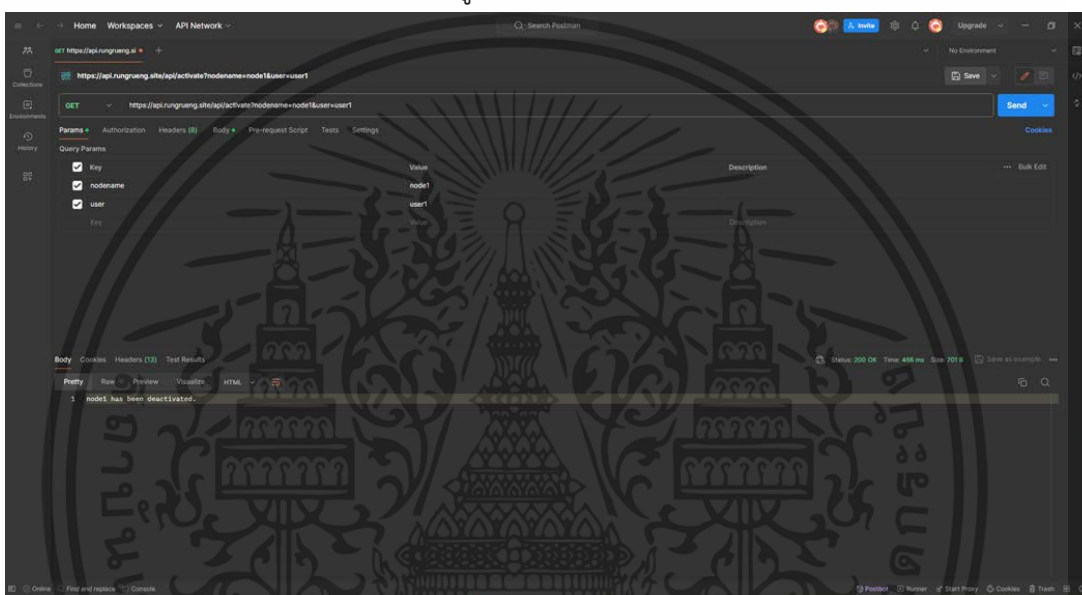


รูปที่ 3.20 การดึงข้อมูลทั้งหมดที่บันทึกไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.7 การเปิด/ปิดใช้งาน Node

ใช้ API Endpoint: /api/activate และ HTTP Method: GET เพื่อเปิด/ปิดใช้งาน Node โดยมี พารามิเตอร์ nodename เป็นชื่อของ Node ที่ต้องการเปิด/ปิดการใช้งาน และมีตัวอย่างคำสั่งการร้องขอ "GET http://rung.ddns.net:8050/api/activate?nodename=node1" และถ้าหากสำเร็จจะมีข้อความตอบกลับนี้ "node1 has been activated." พร้อมสถานะ 200 OK ซึ่ง API นี้จะใช้ร่วมกับ QR code เพื่อให้ผู้ใช้งานมีความสะดวกในการเปิด/ปิดแต่ละ Node



รูปที่ 3.21 การเปิด/ปิดใช้งาน Node

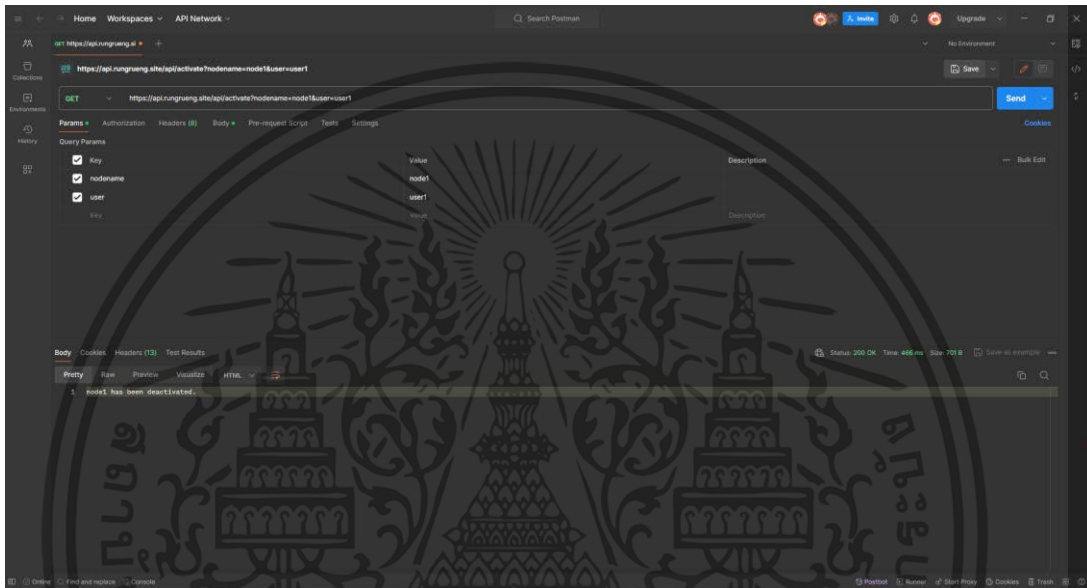


รูปที่ 3.22 QR code ของแต่ละ Node

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.8 การแสดงรายชื่อ Node ที่ใช้งานอยู่

ใช้ API Endpoint: /api/node และ HTTP Method: GET เพื่อดึงรายชื่อ Node ที่ใช้งานอยู่ทั้งหมดในระบบ การตอบกลับ: ระบบจะคืนรายชื่อ Node ที่ใช้งานอยู่ในรูปแบบ JSON โดยมีตัวอย่างการตอบกลับดังนี้ {"nodenames": ["Node1", "Node2", "Node3"]} พร้อมสถานะ 200 OK

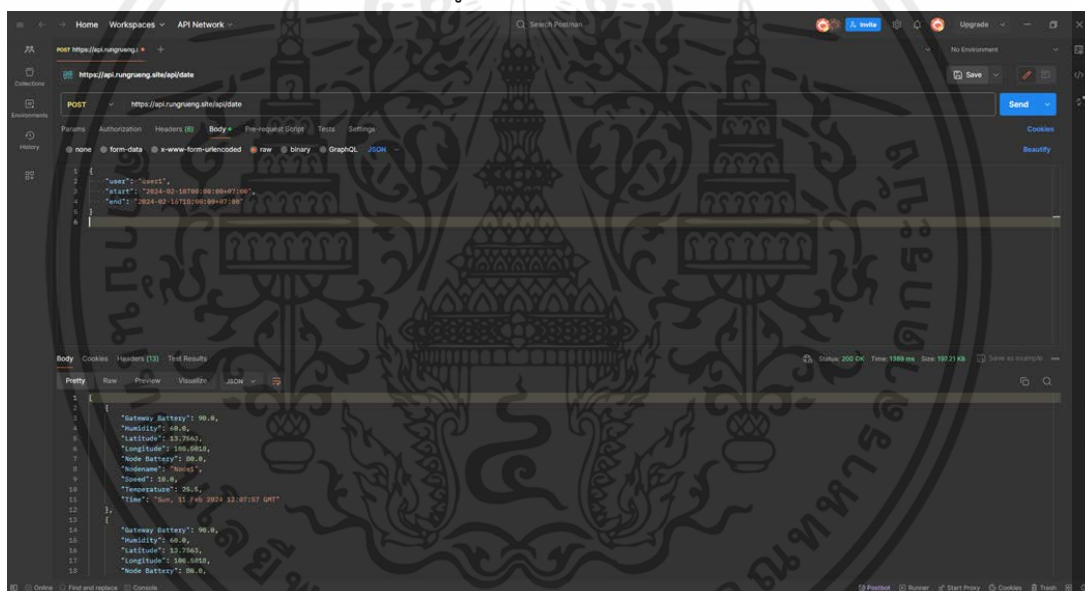


รูปที่ 3.23 การแสดงรายชื่อ Node ที่ใช้งานอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.9 การแสดงข้อมูลตามช่วงวันที่ที่กำหนดเอง

ใช้ API Endpoint: /api/date และ HTTP Method: POST เพื่อดึงข้อมูลที่บันทึกไว้ภายในช่วงวันที่ที่กำหนด มีข้อมูลที่จำเป็นดังนี้ start: วันที่และเวลาเริ่มต้นของช่วงข้อมูล, end: วันที่และเวลาสิ้นสุดของช่วงข้อมูลโดยมีตัวอย่างคำสั่งการร้องขอ "POST http://rung.ddns.net:8050/api/date Content-Type: application/json {"start":"2024-02-10T00:00:00+07:00", "end":"2024-02-16T18:00:00+07:00"}" การตอบกลับ: ระบบจะคืนข้อมูลที่บันทึกไว้ภายในช่วงวันที่ที่กำหนดในรูปแบบ JSON โดยแต่ละรายการประกอบด้วยข้อมูลทั้งหมด และจะมีตัวอย่างการตอบกลับดังนี้ [{"Time":"2024-02-16 17:38:50","Nodename":"Node1","Temperature":25.5,"Humidity":60,"Latitude":13.75,"Longitude":100.48,"Speed":0,"NodeBattery":75,"Gateway Battery":90}, ... (รายการข้อมูลอื่นๆ ภายในช่วงวันที่ที่กำหนด)] พร้อมสถานะ 200 OK

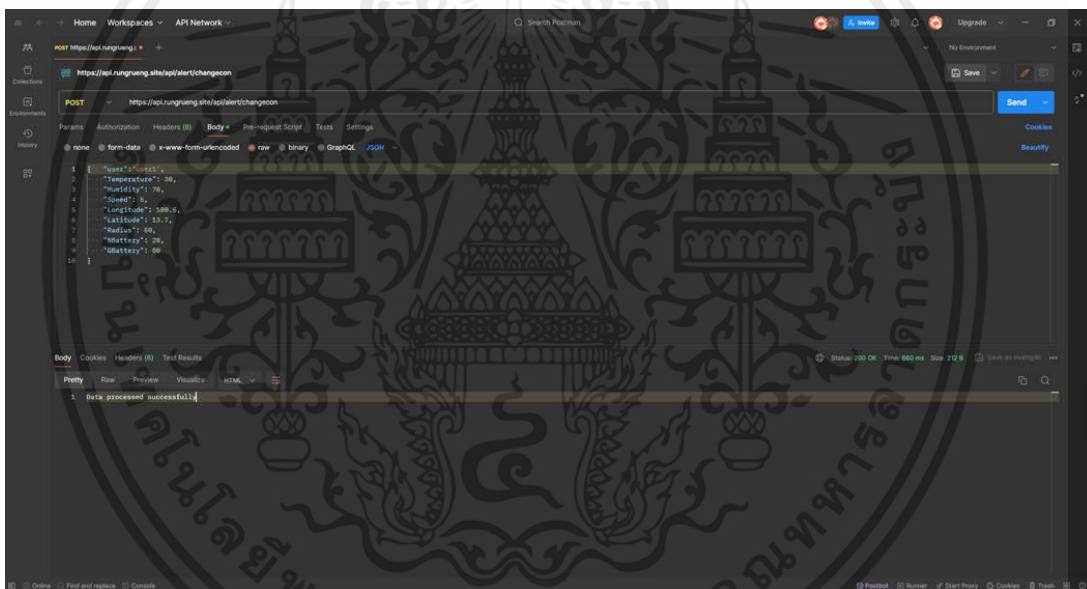


รูปที่ 3.24 การแสดงข้อมูลตามช่วงวันที่ที่กำหนดเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.10 การแก้ไขเงื่อนไขการแจ้งเตือน

ใช้ API Endpoint: /api/alert/changecon และ HTTP Method: POST เพื่อแก้ไขเกณฑ์การแจ้งเตือนทั้งหมดในระบบ มีข้อมูลที่จำเป็นดังนี้ Temperature: อุณหภูมิ (หน่วย °C), Humidity: ความชื้น (หน่วย %), Speed: ความเร็ว (หน่วย กม./ชม.), Longitude: ลองจิจูด (หน่วย °), Latitude: ละติจูด (หน่วย °), Radius: รัศมี (หน่วย กม.), NBattery: ระดับแบตเตอรี่ Node (หน่วย %), GBattery: ระดับแบตเตอรี่ Gateway (หน่วย %) โดยมีตัวอย่างคำสั่งการร้องขอ "POST http://rung.ddns.net:8050/api/alert/changecon Content-Type: application/json {"Temperature": 30, "Humidity": 70, "Speed": 5, "Longitude": 100.5, "Latitude": 13.7, "Radius": 50, "NBattery": 20, "GBattery": 80}" และถ้าหากสำเร็จจะมีข้อความตอบกลับนี้ "Data processed successfully" พร้อมสถานะ 200 OK

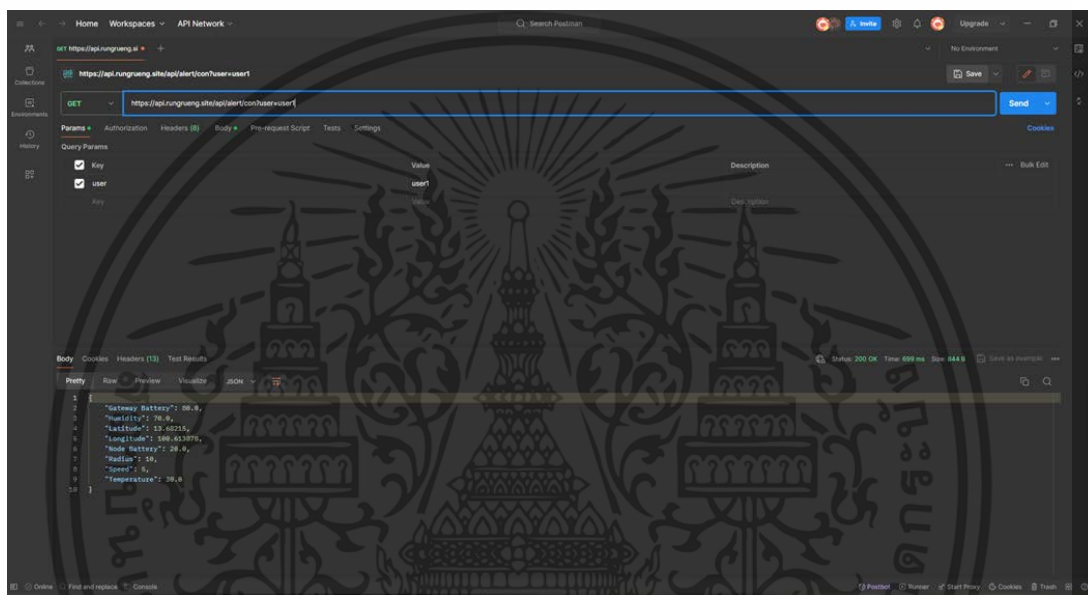


รูปที่ 3.25 การแก้ไขเงื่อนไขการแจ้งเตือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.11 การแสดงเงื่อนไขการแจ้งเตือนปัจจุบัน

ใช้ API Endpoint: /api/alert/con และ HTTP Method: GET เพื่อดึงเกณฑ์การแจ้งเตือนทั้งหมดที่ตั้งค่าไว้ในระบบ การตอบกลับ: ระบบจะคืนเกณฑ์การแจ้งเตือนปัจจุบันทั้งหมดในรูปแบบ JSON โดยมีตัวอย่างการตอบกลับดังนี้ {"Temperature":30,"Humidity":70,"Speed":5,"Longitude":100.5,"Latitude":13.7,"Radius":50,"NodeBattery":20,"GatewayBattery":80} พร้อมสถานะ 200 OK

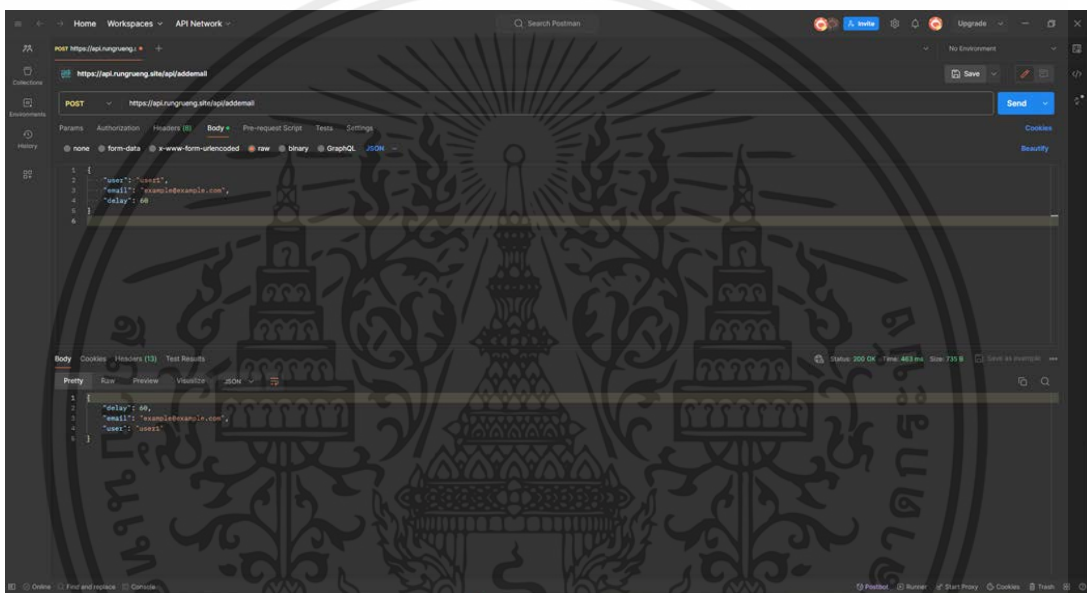


รูปที่ 3.26 การแสดงเงื่อนไขการแจ้งเตือนปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.12 การเพิ่มอีเมลสำหรับรับการแจ้งเตือน

ใช้ API Endpoint: /api/addemail และ HTTP Method: POST เพื่อเพิ่มอีเมลสำหรับรับการแจ้งเตือน และกำหนดความถี่ในการรับแจ้งเตือน (ถ้าต้องการ) มีข้อมูลที่จำเป็นดังนี้ email (ไม่บังคับ): ที่อยู่อีเมลที่ต้องการเพิ่ม, delay: ระยะเวลาหน่วงระหว่างการแจ้งเตือน (หน่วยวินาที) โดยมีตัวอย่างคำสั่งการร้องขอ "POST http://rungg.ddns.net:8050/api/addemail Content-Type: application/json {"email": "example@example.com", "delay": 60 }" และถ้าหากสำเร็จจะมีข้อความตอบกลับนี้ "Email inserted successfully." พร้อมสถานะ 200 OK

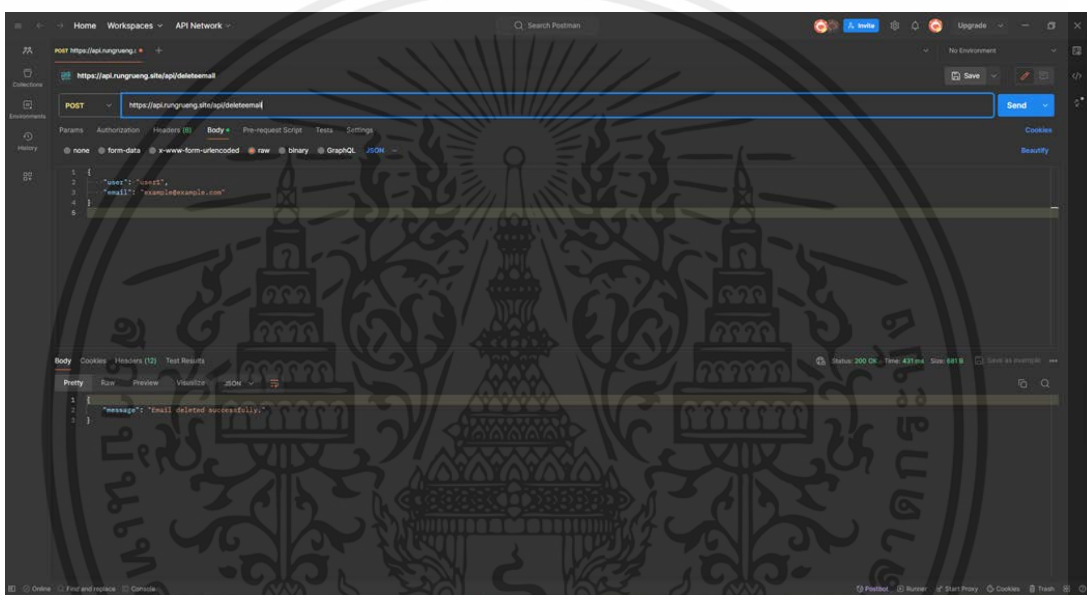


รูปที่ 3.27 การเพิ่มอีเมลสำหรับรับการแจ้งเตือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.13 การลบอีเมลสำหรับรับการแจ้งเตือน

ใช้ API Endpoint: /api/deleteemail และ HTTP Method: POST เพื่อลบอีเมลสำหรับรับการแจ้งเตือน มีข้อมูลที่จำเป็นดังนี้ email: อีเมลที่ต้องการลบ โดยมีตัวอย่างคำสั่งการร้องขอ "POST http://rung.ddns.net:8050/api/deleteemail Content-Type: application/json {"email": "example@example.com"}" และถ้าหากสำเร็จจะมีข้อความตอบกลับนี้ "Email deleted successfully." พร้อมสถานะ 200 OK



รูปที่ 3.28 การลบอีเมลสำหรับรับการแจ้งเตือน

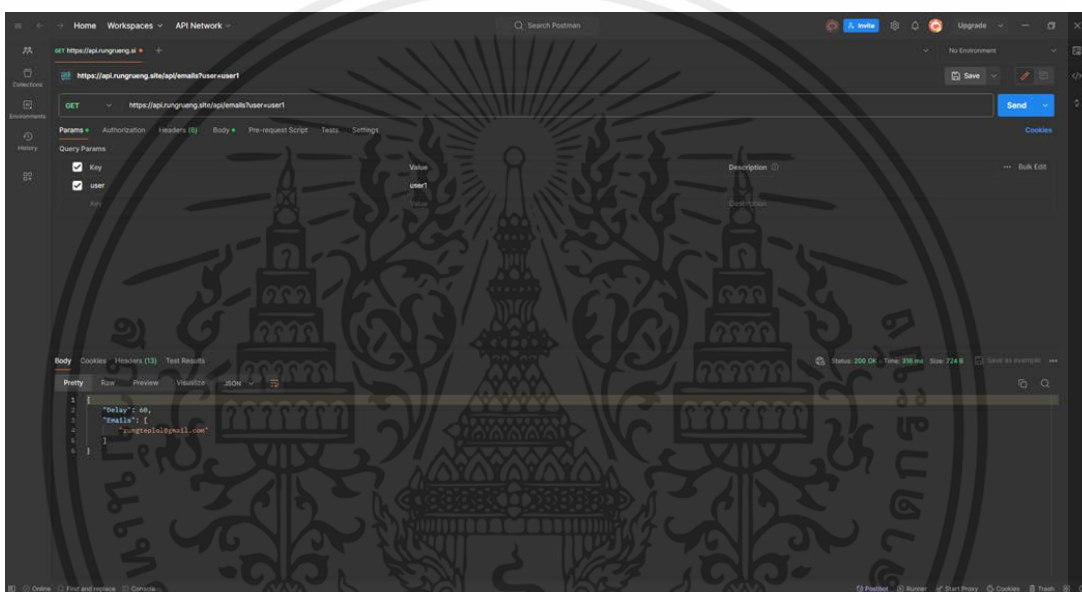
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.14 การแสดงรายชื่ออีเมลและความถี่ในการรับการแจ้งเตือนที่กำหนดไว้

ใช้ API Endpoint: /api/emails และ HTTP Method: GET เพื่อดึงรายชื่ออีเมลทั้งหมดที่ตั้งค่าไว้สำหรับการแจ้งเตือน และความถี่ในการรับแจ้งเตือน การตอบกลับ: ระบบจะคืนข้อมูลอีเมลและความถี่ในการรับแจ้งเตือนในรูปแบบ JSON โดยมีตัวอย่างการตอบกลับดังนี้

```
{"Delay":60,"Emails":["example1@example.com","example2@example.com"]}
```

พร้อมสถานะ 200 OK

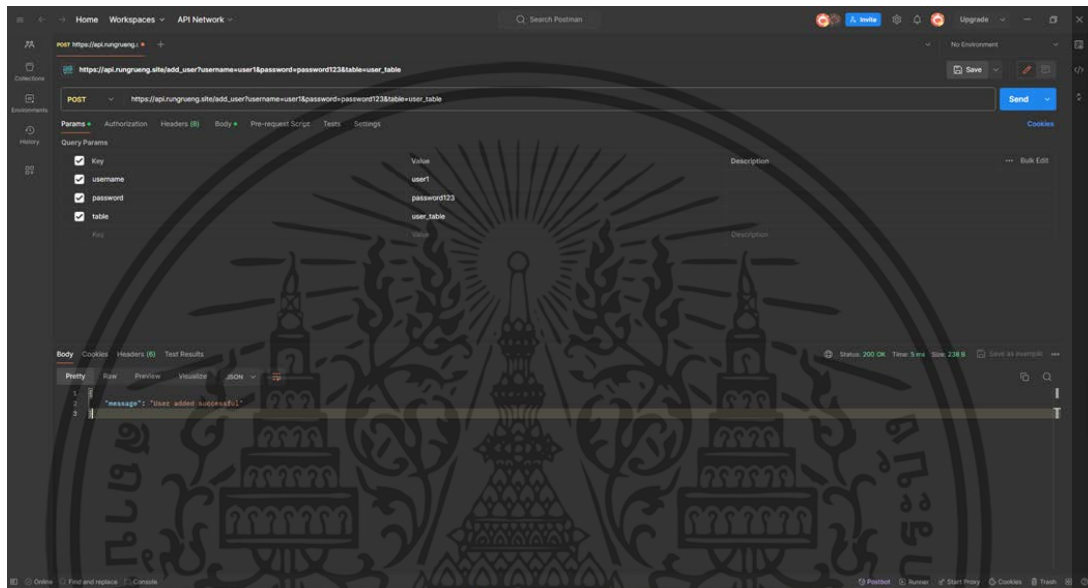


รูปที่ 3.29 การแสดงรายชื่ออีเมลและความถี่ในการรับการแจ้งเตือนที่กำหนดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.15 การเพิ่มผู้ใช้งานเข้าไปในระบบ

ใช้ API Endpoint: /add\_user และ HTTP Method: GET เพื่อสร้างผู้ใช้งานใหม่ขึ้นมาในระบบ โดยมีข้อมูลที่จำเป็นดังนี้ user: ชื่อของผู้ใช้งาน และมีตัวอย่างคำสั่งการร้องขอ “GET https://api.rungrueng.site/add\_user?username=user1&password=password123&table=user\_table” การตอบกลับ: “User added successful” พร้อมสถานะ 200 OK

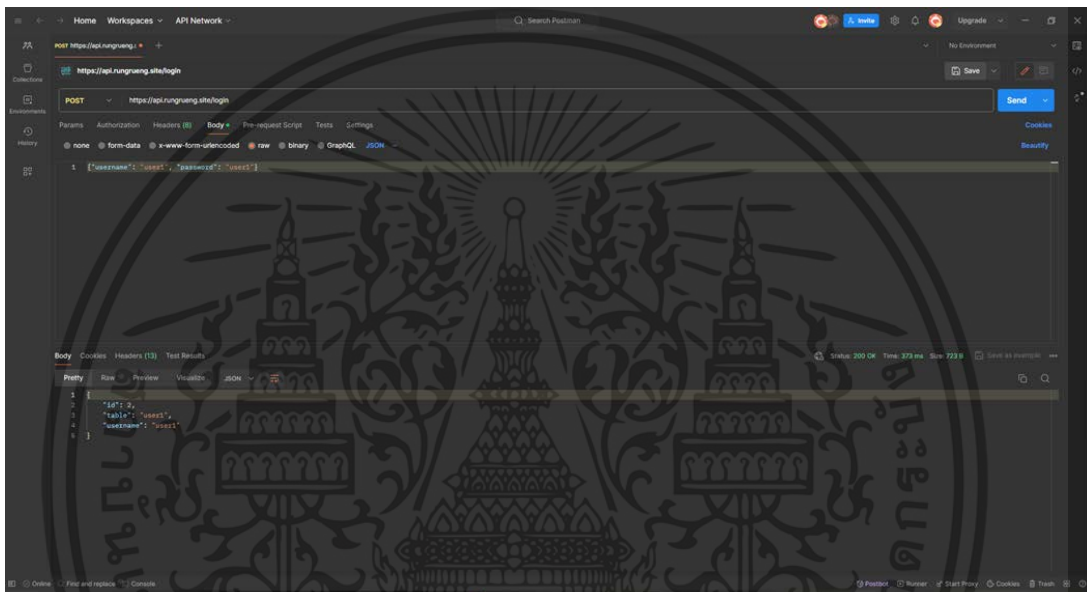


รูปที่ 3.30 การเพิ่มผู้ใช้งานเข้าไปในระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.16 การยืนยันตัวตนผู้ใช้งาน

ใช้ API Endpoint: /login และ HTTP Method: POST เพื่อตรวจสอบข้อมูลการเข้าสู่ระบบที่ถูกส่งมา โดยมีข้อมูลที่จำเป็นดังนี้ username: ชื่อของผู้ใช้งาน, password: รหัสผ่านของผู้ใช้นั้น และมีตัวอย่างคำสั่งการร้องขอ POST `https://api.rungrueng.site/login` Content-Type: `application/json` `{"username": "john_doe", "password": "secret123"}` การตอบกลับ: `{"id":2,"table":"user1","username":"user1"}` สถานะ 200 OK

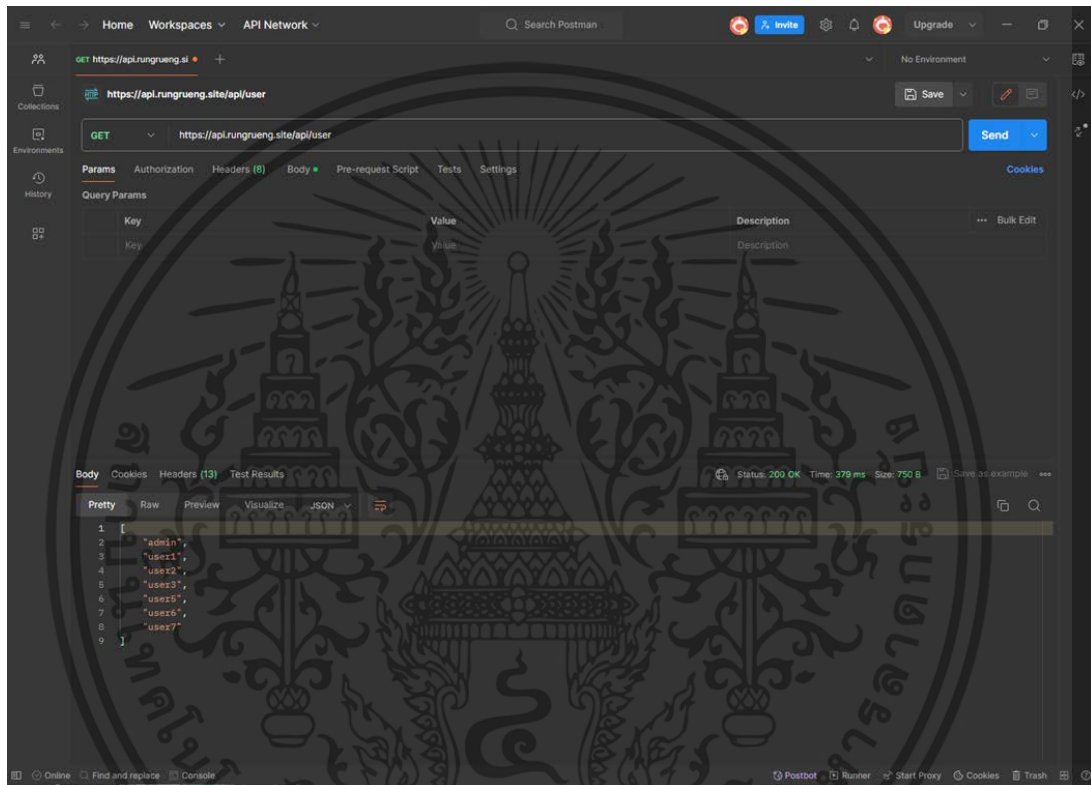


รูปที่ 3.31 การยืนยันตัวตนผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.3.17 การดึงผู้ใช้งานทั้งหมด

ใช้ API Endpoint: /api/user และ HTTP Method: GET เพื่อดึงข้อมูล username ของผู้ใช้งานทั้งหมดจากฐานข้อมูลและมีตัวอย่างคำสั่งการร้องขอ“GET https://api.rungrueng.site/api/user” การตอบกลับ: ระบบจะคืนค่า username ทั้งหมดในระบบในรูปแบบ JSON โดยมีตัวอย่างการตอบกลับดังนี้ {“user1”, “user2”, “user3”} พร้อมสถานะ 200 OK



รูปที่ 3.32 การดึงผู้ใช้งานทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5 การออกแบบในส่วนของหน้าเว็บไซต์

ในส่วนนี้เราจะทำการออกแบบหน้าเว็บไซต์เพื่อการใช้งานที่มีประสิทธิภาพและครบถ้วนในการแสดงผลข้อมูลต่าง ๆ โดยการแยกแยะผู้ใช้อย่างชัดเจน เพื่อให้แต่ละผู้ใช้สามารถปรับแต่งการตั้งค่าต่าง ๆ ได้อย่างสะดวก นอกจากนี้ยังมีหน้าแอดมินที่เป็นที่ช่วยให้ผู้ดูแลระบบสามารถจัดการข้อมูลต่าง ๆ ได้อย่างมีประสิทธิภาพ ซึ่งในการออกแบบส่วนหน้าเว็บไซต์นั้นจะถูกแบ่งออกเป็นขั้นตอนตามลำดับดังนี้

#### 3.5.1 วัตถุประสงค์และเป้าหมายของเว็บไซต์

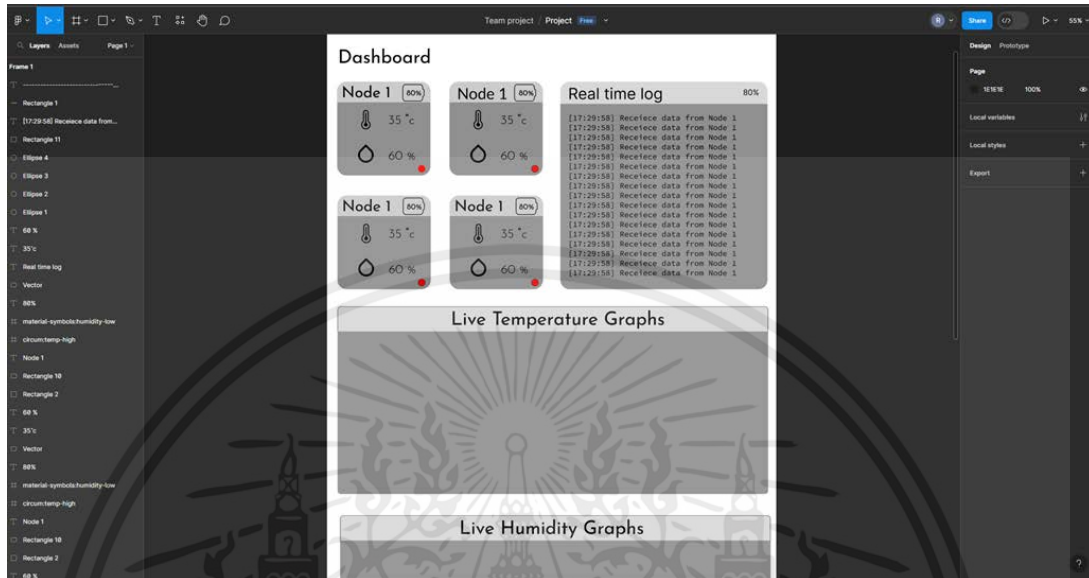
ในขั้นตอนนี้เราได้ทำการกำหนดวัตถุประสงค์และเป้าหมายของเว็บไซต์ โดยมีวัตถุประสงค์ดังนี้

- แสดงค่าเฉลี่ยอุณหภูมิและความชื้นของแต่ละ Node
- แสดงกราฟอุณหภูมิและความชื้น
- แสดงสถานะแบตเตอรี่และสถานะการทำงานของแต่ละ Node
- แสดงตำแหน่งของแต่ละ Node ณ แต่ละช่วงเวลา
- แสดงค่าที่ Gateway ได้รับมาแบบ Realtime
- สามารถดูข้อมูลย้อนหลังได้
- สามารถแก้ไขการตั้งค่าต่างของแต่ละ Node ได้
- สามารถแก้ไขเงื่อนไขการแจ้งเตือนและช่องทางได้
- แบ่งหน้าการใช้ของแต่ละผู้ใช้ออกจากกัน
- มีหน้าแอดมินสำหรับการจัดระบบ

#### 3.5.2 การออกแบบโครงสร้างและกราฟิกของเว็บไซต์

ในขั้นตอนการออกแบบโครงสร้างของเว็บไซต์ เว็บไซต์จะแบ่งออกเป็นส่วนสำหรับแสดงค่าเฉลี่ยอุณหภูมิและความชื้นของแต่ละ และเว็บไซต์จะแสดงผลค่าที่ Gateway ได้รับมาแบบเรียลไทม์ลงในกล่องข้อความ นอกจากนี้ยังจะนำเสนอข้อมูลอุณหภูมิและความชื้นในรูปแบบกราฟ เพื่อให้เห็นแนวโน้มการเปลี่ยนแปลงได้ง่ายขึ้น และสุดท้าย เว็บไซต์จะระบุตำแหน่งของ Node บนแผนที่ เพื่อให้ผู้ใช้งานสามารถเข้าใจตำแหน่งของ Node ได้ง่ายขึ้น โดยในสุดท้ายได้นำทั้งหมดนี้มา

ออกแบบโดยใช้ Figma ซึ่งเป็นโปรแกรมที่เอาไว้ใช้ออกแบบเว็บไซต์ และแสดงเว็บไซต์ที่ออกแบบมาได้ดังรูป



รูปที่ 3.33 การใช้งานโปรแกรม Figma สำหรับการออกแบบเว็บไซต์

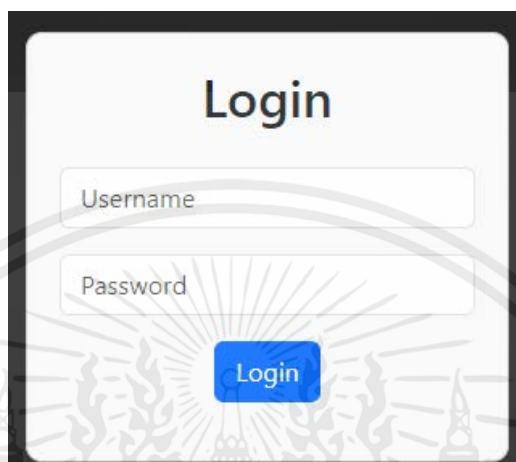
### 3.5.3 การพัฒนาเว็บไซต์

ในขั้นตอนการพัฒนาเว็บไซต์ โดยใช้ภาษา HTML, CSS และ JS หลักๆ จะใช้ API ที่สร้างขึ้นมาก่อนหน้านี้แล้ว โดย API นี้จะทำหน้าที่ส่งข้อมูลต่างๆ ที่จำเป็นสำหรับการแสดงผลเว็บไซต์ การพัฒนาเว็บไซต์ด้วย API มีข้อดีคือช่วยลดระยะเวลาในการพัฒนาเว็บไซต์ เนื่องจากไม่จำเป็นต้องเขียนโค้ดเพื่อดึงข้อมูลขึ้นมาเอง เพียงทำการส่ง request ไปยัง API ก็สามารถดึงข้อมูลที่ต้องการมาใช้งานได้ทันที โดยเราอาจเพิ่มการกำหนดเวลาเอาไว้เช่น การ request ไปยัง API ทุก 10 วินาที เพื่อให้ข้อมูลที่ได้มีความเป็นปัจจุบันที่สุด โดยการทำงานของเว็บไซต์จะแบ่งออกเป็น 7 ส่วนหลักๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.3.1 Login

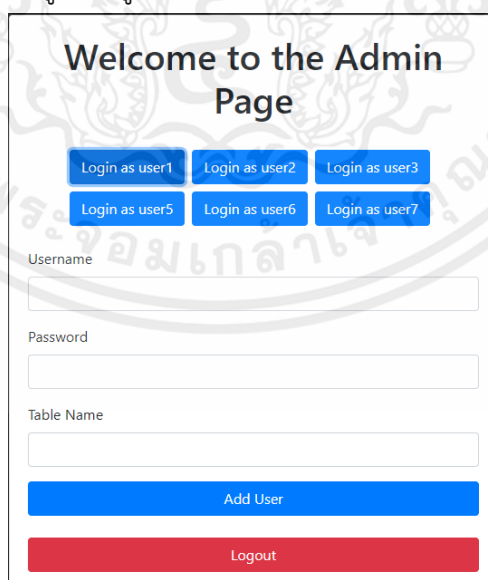
ส่วนนี้มีหน้าที่รับค่า Username และ Password เพื่อใช้ในการยืนยันตัวตนก่อนเข้าสู่หน้าหลักของระบบ



รูปที่ 3.34 รูปแสดงหน้าต่างสำหรับการเข้าสู่เว็บไซต์

### 3.5.3.2 Admin

ส่วนนี้มีหน้าที่ในการจัดการกับแต่ละผู้ใช้งานรวมถึงการสร้างผู้ใช้งานเพิ่มเติมเข้าสู่ระบบ นอกจากนี้ยังสามารถเข้าถึงข้อมูลของผู้ใช้แต่ละคนได้



รูปที่ 3.35 รูปแสดงหน้าต่างสำหรับการจัดการผู้ใช้ของแอดมิน

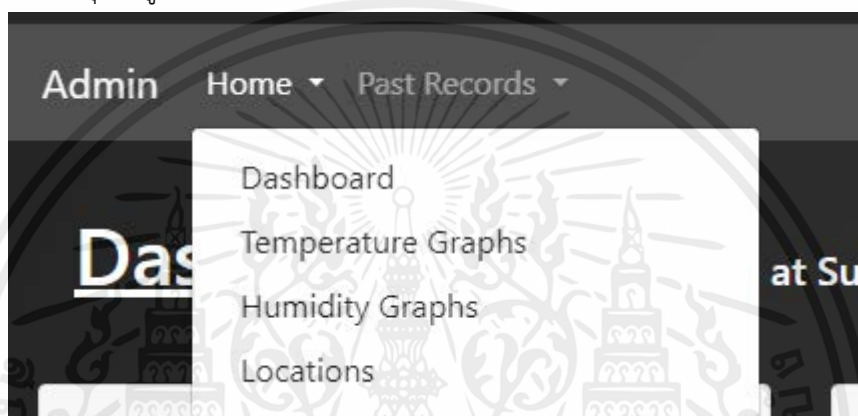
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.3.3 Website Navigation Bar

ส่วนนี้จะทำหน้าที่แสดงเมนูต่างๆ ที่ผู้ใช้สามารถเรียกใช้ได้ โดยเมนูหลักมีอยู่ 5 เมนู ได้แก่

#### 1) Home

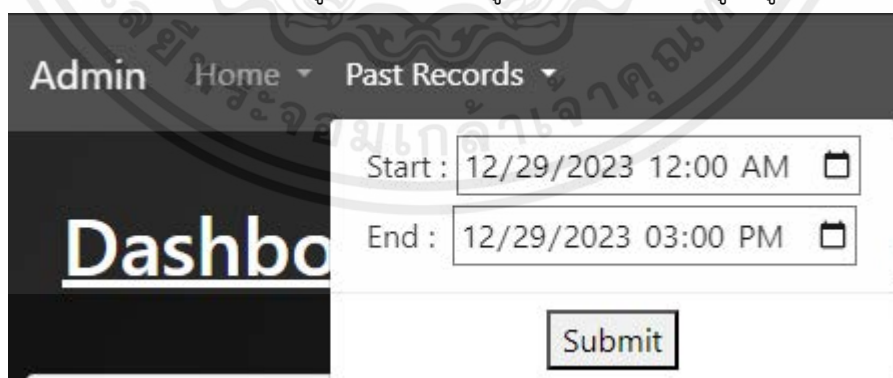
เมื่อกดเมนูนี้ จะแสดงหน้าต่างๆ ที่พาผู้ใช้ไปยังส่วนต่างๆ ของหน้าเว็บไซต์ เช่น หน้าข้อมูลหลัก กราฟอุณหภูมิ กราฟความชื้น ตำแหน่ง เป็นต้น



รูปที่ 3.36 รูปแสดงการทำงานของเมนู Home

#### 2) Past Records

เมื่อกดเมนูนี้ จะแสดงหน้าต่างๆ ให้ผู้ใช้เลือกเวลาเริ่มต้นและสิ้นสุด จากนั้นเว็บไซต์จะสร้างหน้าต่างๆ ใหม่ขึ้นมาเพื่อแสดงข้อมูล ณ วันเวลาที่ผู้ใช้กรอกมา ถ้ามีข้อมูลอยู่



รูปที่ 3.37 รูปแสดงการทำงานของเมนู Past records

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3) Setting

เมื่อกดเมนูนี้ จะแสดงหน้าต่างให้ผู้ใช้สามารถปรับปรุงแก้ไขการตั้งค่าต่างของแต่ละ Node ได้

รูปที่ 3.38 รูปแสดงการทำงานของเมนู Setting

### 4) Alert

เมื่อกดเมนูนี้ จะทำการนำผู้ใช้งานไปที่หน้า Alert ที่จะมีหน้าต่างให้ผู้ใช้สามารถตั้งค่าการแจ้งเตือนต่างๆได้

รูปที่ 3.39 รูปแสดงหน้าต่างสำหรับการปรับเงื่อนไขการแจ้งเตือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Email Settings**

Recipient Email:

Delay between Alerts :

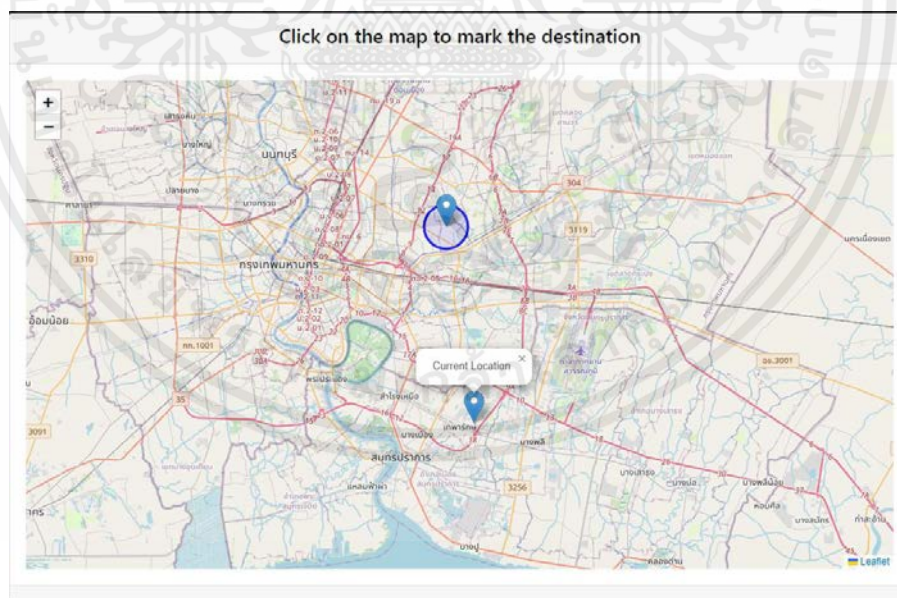
 Sec

---

**List of recipient email**

rungteplol@gmail.com X

รูปที่ 3.40 รูปแสดงหน้าต่างสำหรับการเพิ่มอีเมลสำหรับการรับการแจ้งเตือน



รูปที่ 3.41 รูปแสดงหน้าต่างสำหรับการเลือกตำแหน่งปลายทางที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5) Logout

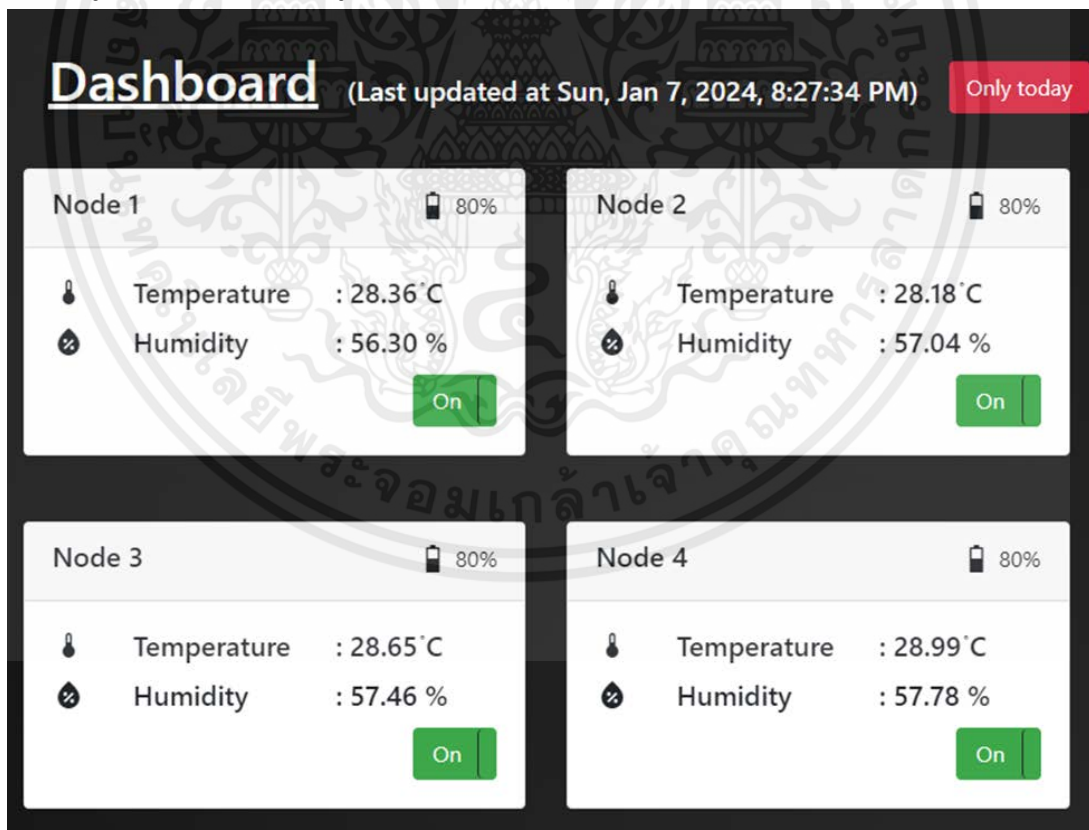
เมื่อกดเมนูนี้ จะทำการนำผู้ใช้งานออกจากระบบที่ใช้อยู่ปัจจุบัน



รูปที่ 3.42 รูปแสดงปุ่ม Log out

## 3.5.3.4. Dashboard

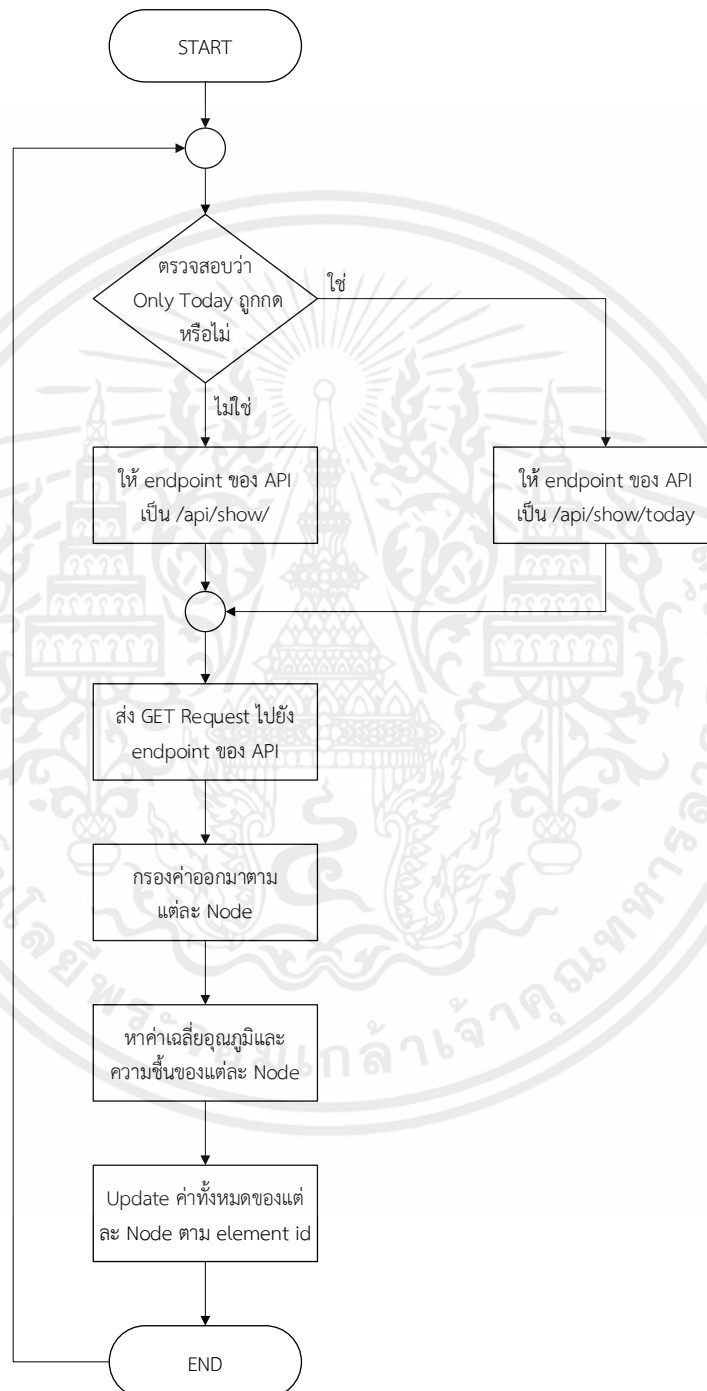
ในส่วนนี้จะทำการนำค่าอุณหภูมิและความชื้นของแต่ละ Node มาหาค่าเฉลี่ย และนำค่าแบตเตอรี่มาเพื่อแสดงผลที่แต่ละกล่องข้อความ นอกจากนี้ แต่ละกล่องยังสามารถเปิด/ปิดการทำงาน Node นั้นๆ ได้ ซึ่งเมื่อ Node ใดถูกปิดการใช้งาน Node นั้นๆ จะไม่สามารถส่งข้อมูลไปยังฐานข้อมูลได้ และข้อความที่อยู่



รูปที่ 3.43 รูปแสดงหน้า Dashboard

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้างคำว่า Dashboard จะแสดงเวลาล่าสุดที่มีการอัปเดตข้อมูลขึ้นมา และปุ่ม Only today จะทำให้เว็บไซต์แสดงเฉพาะค่าที่ได้รับภายในวันปัจจุบันเท่านั้น โดยส่วนของ Dashboard และ Flowchart การทำงานของส่วนนี้จะถูกแสดงไว้ตามรูปที่ 3.44-3.48



รูปที่ 3.44 แผนผังแสดงขั้นตอนการทำงานของ การอัปเดตค่าในส่วน ของ Dashboard

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

async function fetchData() {
  let response;
  if (today) {
    response = await fetch(url + "/api/show/today");
  } else {
    response = await fetch(url + "/api/show");
  }

  const data = await response.json();

  if (Array.isArray(data) && data.length > 0) {
    console.log("fetchData successful: Received valid data.");
  } else {
    console.error("Invalid or empty data received:", data);
  }
  return data;
}

```

รูปที่ 3.45 คำสั่งการทำงานสำหรับการปรับเปลี่ยน API Endpoint

(index)	Gateway	Battery	Humidity	Latitude	Longitude	Nodename	Speed	Temperature	Time
0	92.82	49.24	49.24	13.622655	100.657806	'Node1'	0	31.15	'Fri, 29 Dec 2023 14:07:24 GMT'
1	92.82	49.02	49.02	13.622655	100.657806	'Node2'	0	31.15	'Fri, 29 Dec 2023 14:07:25 GMT'
2	92.82	48.97	48.97	13.622655	100.657806	'Node3'	0	31.12	'Fri, 29 Dec 2023 14:07:27 GMT'
3	92.82	48.95	48.95	13.622655	100.657806	'Node4'	0	31.14	'Fri, 29 Dec 2023 14:07:27 GMT'
4	92.82	48.94	48.94	13.622655	100.657806	'Node5'	0	31.13	'Fri, 29 Dec 2023 14:07:29 GMT'
5	93.06	49.4	49.4	13.622638	100.657898	'Node1'	0	31.11	'Fri, 29 Dec 2023 14:12:40 GMT'
6	93.06	49.4	49.4	13.622638	100.657898	'Node2'	0	31.12	'Fri, 29 Dec 2023 14:12:42 GMT'
7	93.06	49.4	49.4	13.622638	100.657898	'Node3'	0	31.11	'Fri, 29 Dec 2023 14:12:43 GMT'
8	93.06	49.42	49.42	13.622638	100.657898	'Node4'	0	31.11	'Fri, 29 Dec 2023 14:12:45 GMT'
9	93.06	49.43	49.43	13.622638	100.657898	'Node5'	0	31.11	'Fri, 29 Dec 2023 14:12:47 GMT'

รูปที่ 3.46 ข้อมูลที่ได้รับมาจาก Response ของ API

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function filterDataByNode(data, node) {
    return data.filter((item) => item.NodeName === node);
    // Example filterDataByNode(data, "Node1")
}

function updateAvg(data, idtemp, idhumi) {
    const temperatures = data.map((item) => item.Temperature);
    const humidity = data.map((item) => item.Humidity);
    let sumTem = 0;
    let sumHum = 0;

    for (let i = 0; i < temperatures.length; i++) {
        sumTem += temperatures[i];
    }
    const averageTemp = sumTem / temperatures.length;

    for (let i = 0; i < humidity.length; i++) {
        sumHum += humidity[i];
    }
    const averageHumidity = sumHum / humidity.length;

    console.log(`Average Temperature [${idtemp}]: ${averageTemp.toFixed(2)} °C`);
    console.log(`Average Humidity [${idhumi}]: ${averageHumidity.toFixed(2)} %`);

    const avgTempElement = document.getElementById(idtemp);
    avgTempElement.innerHTML = `: ${averageTemp.toFixed(2)} °C`;

    const avgHumiElement = document.getElementById(idhumi);
    avgHumiElement.innerHTML = `: ${averageHumidity.toFixed(2)} %`;
}

```

รูปที่ 3.47 คำสั่งการทำงานสำหรับกรองข้อมูลตามชื่อ Node หาค่าเฉลี่ย และ อัปเดต Dashboard ตาม Element ID

```

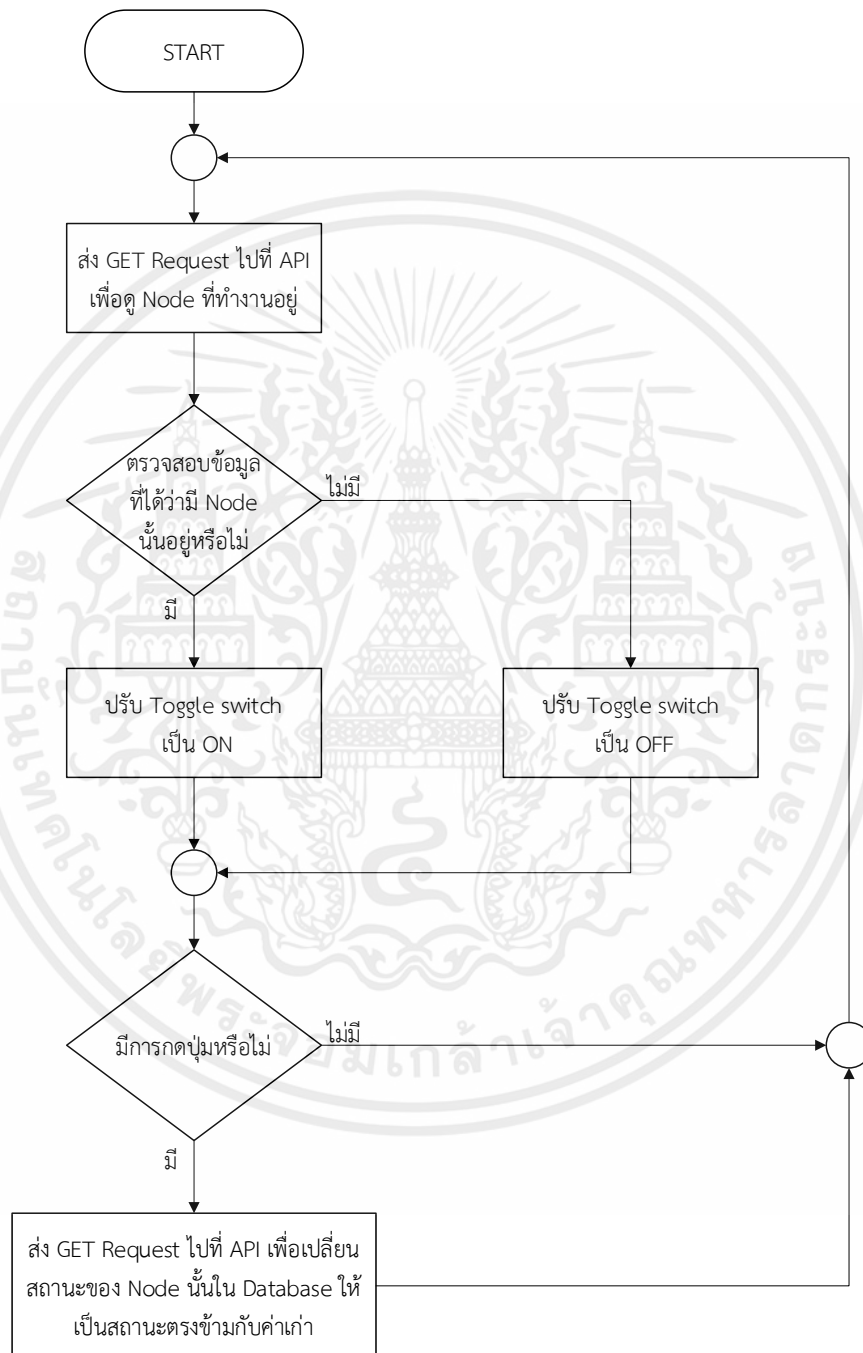
Average Temperature [node1avgtemp]: 28.18 °C
Average Humidity [node1avghumi]: 56.30 %
Average Temperature [node2avgtemp]: 28.36 °C
Average Humidity [node2avghumi]: 57.04 %
Average Temperature [node3avgtemp]: 28.65 °C
Average Humidity [node3avghumi]: 57.46 %
Average Temperature [node4avgtemp]: 28.99 °C
Average Humidity [node4avghumi]: 57.78 %

```

รูปที่ 3.48 ข้อมูลค่าเฉลี่ยอุณหภูมิและความชื้นของแต่ละ Node พร้อมชื่อ Element ID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปในส่วนของ Dashboard จะเห็นได้ว่ามีปุ่มที่ไว้ใช้เปิดปิดการทำงานของแต่ละ Node อยู่ โดยสถานะของแต่ละปุ่มจะถูกอัปเดตทุก 5 วินาที เพื่อให้ค่าที่แสดงเป็นเรียลไทม์ โดยจะมีการทำงานตาม Flowchart รูปที่ 3.49



รูปที่ 3.49 แผนผังแสดงขั้นตอนการทำงานของ การอัปเดตค่าสถานะของปุ่มเปิดปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

async function fetchDataAndToggle() {
    const response = await fetch(url + "/api/node");
    const data = await response.json();
    updateCheckbox(data, "Node1", "node1toggle");
    updateCheckbox(data, "Node2", "node2toggle");
    updateCheckbox(data, "Node3", "node3toggle");
    updateCheckbox(data, "Node4", "node4toggle");
}

function updateCheckbox(data, nodeName, checkboxId) {
    const nodeExists = data.nodenames.includes(nodeName);
    document.getElementById(checkboxId).checked = nodeExists;
    if (nodeExists) {
        $('#${checkboxId}`).bootstrapToggle("on");
    } else {
        $('#${checkboxId}`).bootstrapToggle("off");
    }
}

function toggleChanged(nodeName, checkboxId) {
    console.log(nodeName + " is toggled");
    $('#${checkboxId}').bootstrapToggle("toggle");
    fetch(url + "/api/activate?nodename=" + nodeName);
}

fetchDataAndToggle();
setInterval(fetchDataAndToggle, 5000);

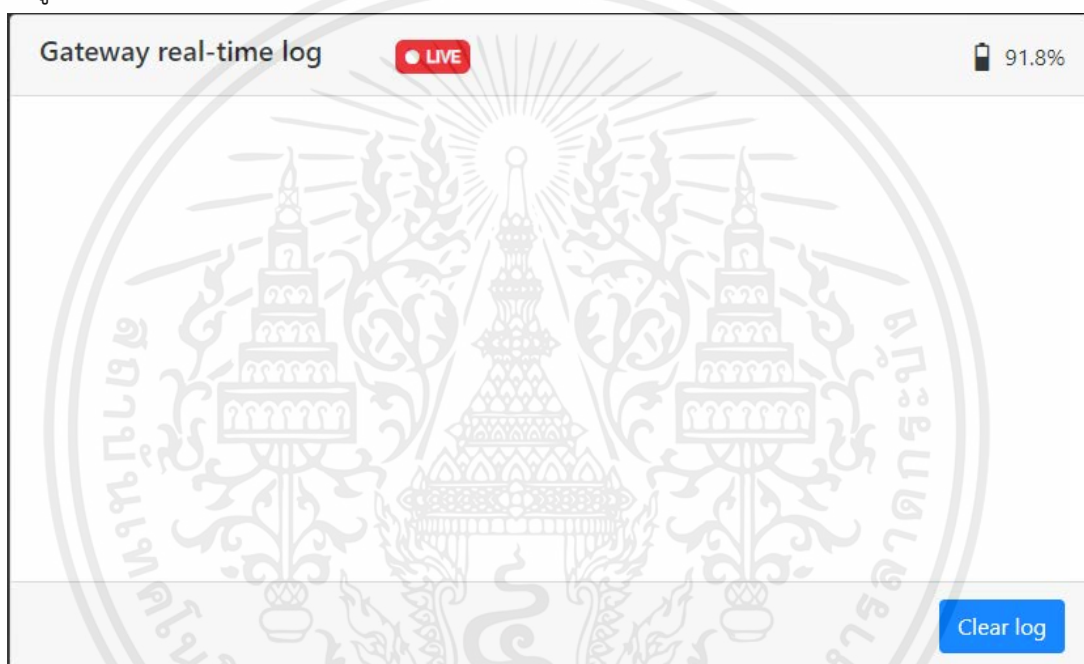
```

รูปที่ 3.50 คำสั่งการทำงานสำหรับการอัปเดตสถานะของปุ่ม on/off

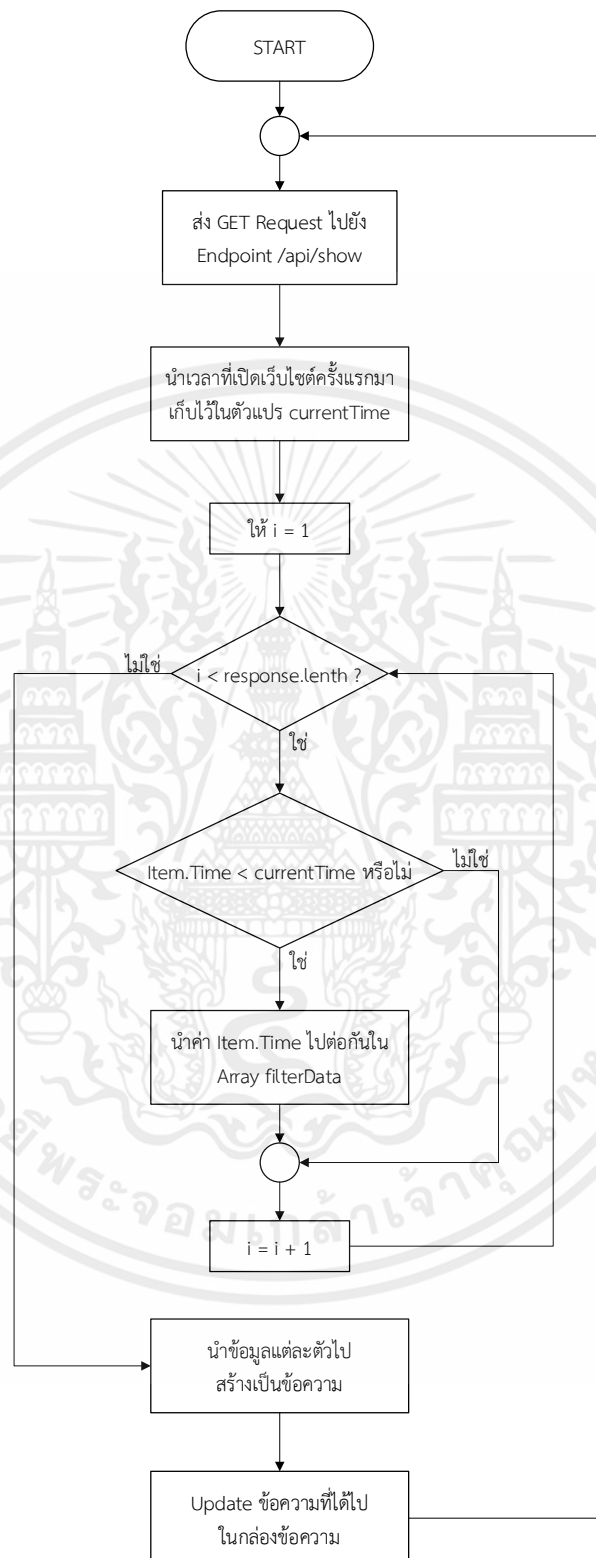
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.3.5 Gateway real-time log

ในส่วนนี้จะป็นกล่องข้อความที่จะทำหน้าที่แสดงข้อมูลทั้งหมดที่ฝั่ง Gateway ได้รับมาแบบ Realtime โดยข้อความที่แสดงจะประกอบด้วยเวลาที่ได้รับชื่อ Node ค่าอุณหภูมิ ความชื้น และปริมาณแบตเตอรี่ของ Gateway ที่มุมขวาบนของกล่องข้อความ โดยข้อความจะแสดงออกมาทีละบรรทัด และมีปุ่ม Clear log ทางด้านขวาล่างเพื่อลบข้อความทั้งหมดที่ถูกแสดงออกมาในกล่องข้อความ ซึ่ง Flowchart แสดงขั้นตอนการทำงานและรูปภาพในส่วนนี้จะถูกแสดงไว้ ดังรูปที่ 3.51-3.55



รูปที่ 3.51 รูปแสดงเว็บไซต์ในส่วนของ Gateway real-time log



รูปที่ 3.52 แผนผังแสดงขั้นตอนการทำงานของการทำงานของการอัปเดตค่าในกล่องข้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
function updateLog(responseData) {
  const options = {
    hour: "numeric",minute: "numeric",second: "numeric",timeZone: "UTC",
  };
  responseData.forEach((element) => {
    const elementTimestamp = new Date(element.Time).getTime();
    const currentTimestamp = currentTime.getTime() + 7 * 60 * 60 * 1000;
    if (elementTimestamp > currentTimestamp) {
      filteredData.push(element);
      filteredData.reverse();
    }
  });
  console.log(filteredData);
  const messages = filteredData.map((item) => {
    const formattedTime = new Date(item.Time).toLocaleString("en-US", options);
    return `[${formattedTime}] Received data from ${item.Nodename} temperature
    is ${item.Temperature} °C and humidity is ${item.Humidity}% <br>`;
  });
  const cardTextElement = document.getElementById("realtimeLog");
  cardTextElement.innerHTML = messages.join("");
}
}
```

รูปที่ 3.53 คำสั่งการทำงานสำหรับการนำเวลามาเปรียบเทียบและ Update ค่าลงในกล่องข้อความ

```
async function updategatewaybat(data) {
  const latestdata = data[data.length - 1];
  const gatewaybatelement = document.getElementById("gatewaybatt");
  var gatewaybat = latestdata["Gateway Battery"];
  if (gatewaybat < 0) {
    gatewaybatelement.innerHTML = "Charging";
  } else {
    gatewaybatelement.innerHTML = gatewaybat + "%";
  }
}
}
```

รูปที่ 3.54 คำสั่งการทำงานสำหรับการนำค่า battery จาก array ตัวสุดท้ายมา Update ค่าลงในกล่องข้อความ

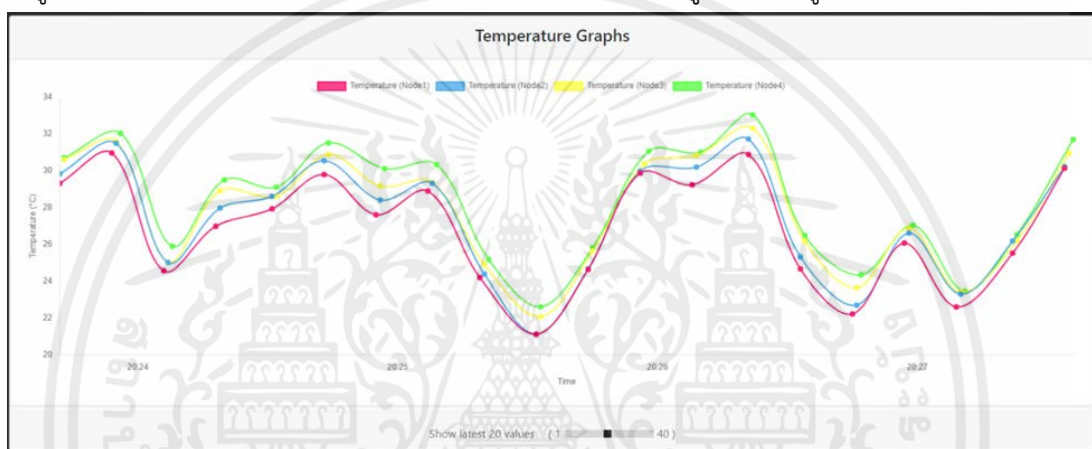
```
document.getElementById("clearLogBtn").addEventListener("click", function() {
  filteredData = [];
  currentTime = new Date();
  const cardTextElement = document.getElementById("realtimeLog");
  cardTextElement.innerHTML = "";
});
});
```

รูปที่ 3.55 คำสั่งการทำงานสำหรับการรอกการกดปุ่ม Clear log เพื่อลบข้อความทั้งหมด

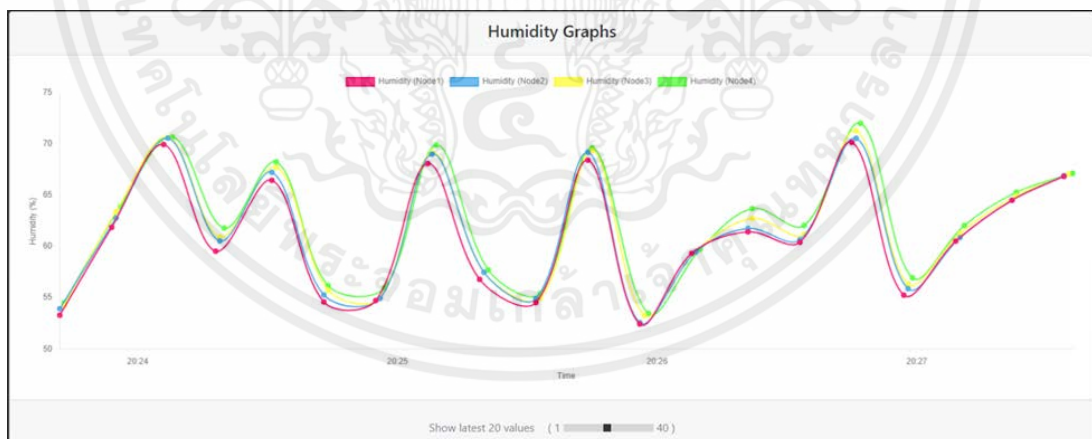
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.3.6 Temperature and Humidity Graphs

ในส่วนนี้จะมียกกล่องที่นำค่าอุณหภูมิและความชื้นของแต่ละ Node มาสร้างเป็นกราฟ เพื่อให้การวิเคราะห์ข้อมูลเป็นไปอย่างง่าย ในแต่ละกล่องของกราฟนั้น เราสามารถเลือกที่จะแสดงหรือไม่แสดง Node ที่เราสนใจ เพื่อให้ผู้ใช้สามารถเลือกดูข้อมูลของ Node ที่ต้องการได้ตามต้องการ นอกจากนี้ ที่ด้านล่างของแต่ละกล่องกราฟ จะมี Slider ให้ผู้ใช้สามารถเลื่อนเพื่อปรับจำนวนข้อมูลที่จะแสดงในกราฟได้ โดยมีค่าเริ่มต้นที่ 20 ค่า และสามารถปรับได้ตั้งแต่ 1 ถึง 40 ค่า ซึ่งรูปภาพและ Flowchart การทำงานของเว็บไซต์ในส่วนนี้จะถูกแสดงในรูปที่ 3.56-3.59

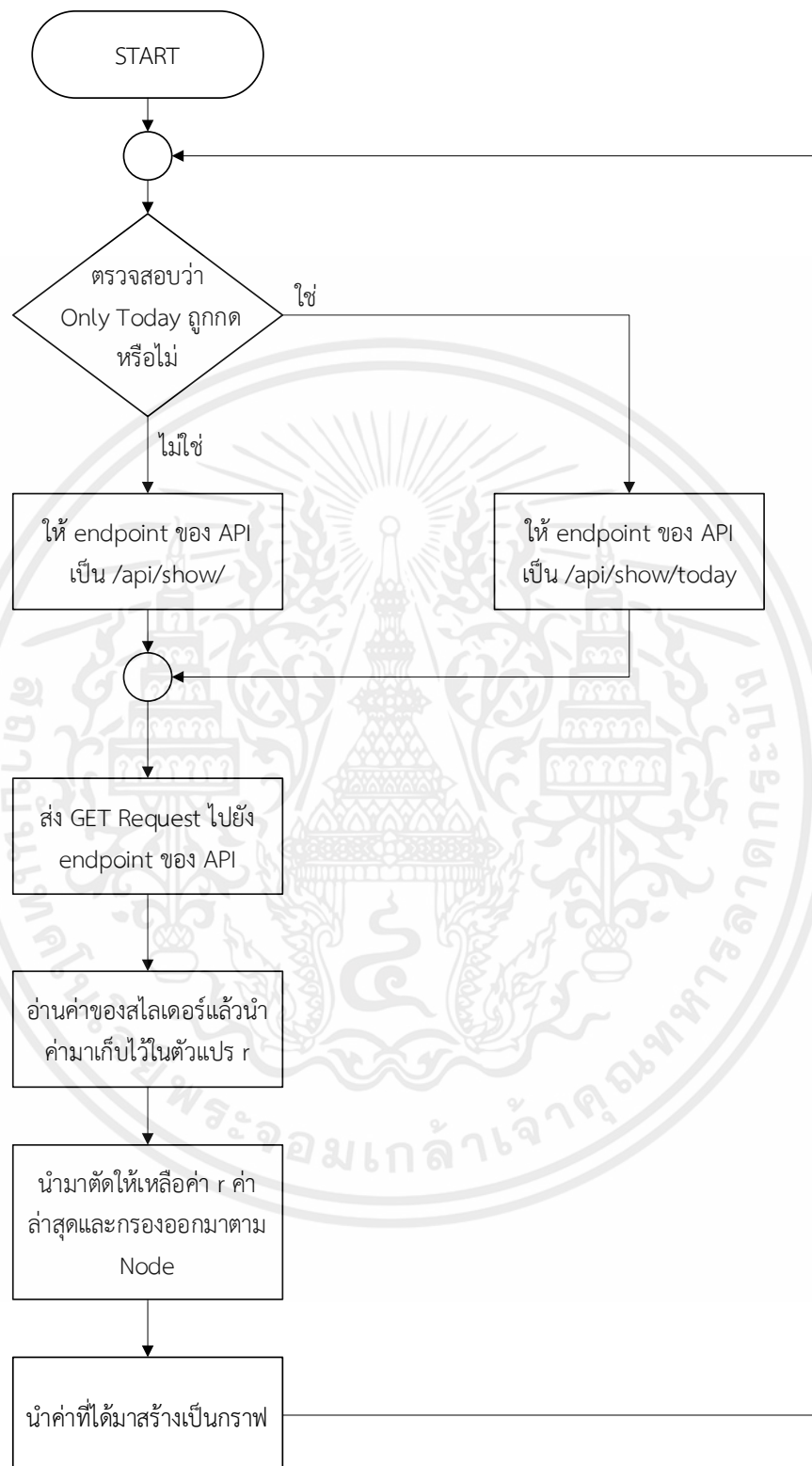


รูปที่ 3.56 รูปแสดงกล่องข้อมูลกราฟอุณหภูมิของแต่ละ Node



รูปที่ 3.57 รูปแสดงกล่องข้อมูลกราฟความชื้นของแต่ละ Node

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.58 แผนผังแสดงขั้นตอนการทำงานของกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

let slider = document.getElementById("myRange");
let sliderHumi = document.getElementById("myRangehumi");
r = slider.value;
rH = sliderHumi.value;
slider.oninput = function () {
  r = this.value;
  var output = document.getElementById("sliderValue");
  console.log(output.innerHTML);
  output.innerHTML = r;
};

sliderHumi.oninput = function () {
  rH = this.value;
  var output = document.getElementById("sliderValuehumi");
  console.log(output.innerHTML);
  output.innerHTML = rH;
};

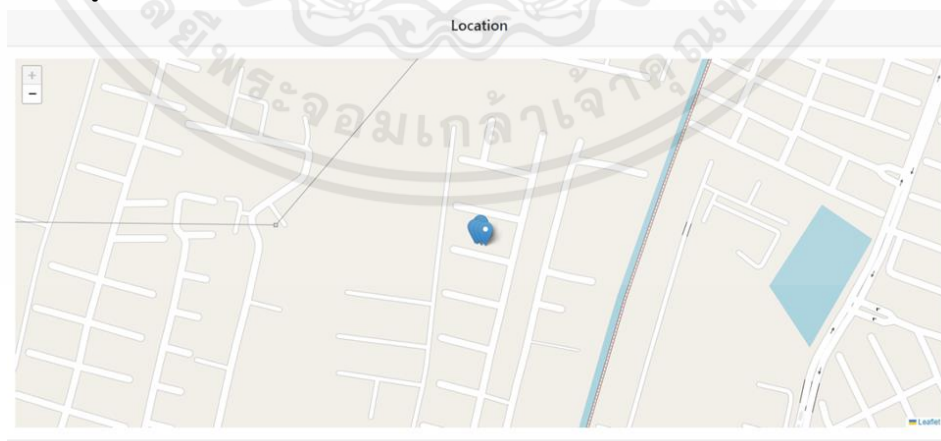
const temperatureChart = new Chart(document.getElementById(canvasId), config);
const humidityChart = new Chart(document.getElementById(canvasId), config);

```

รูปที่ 3.59 คำสั่งสำหรับการดึงค่าจาก Slider และ สร้างกราฟ

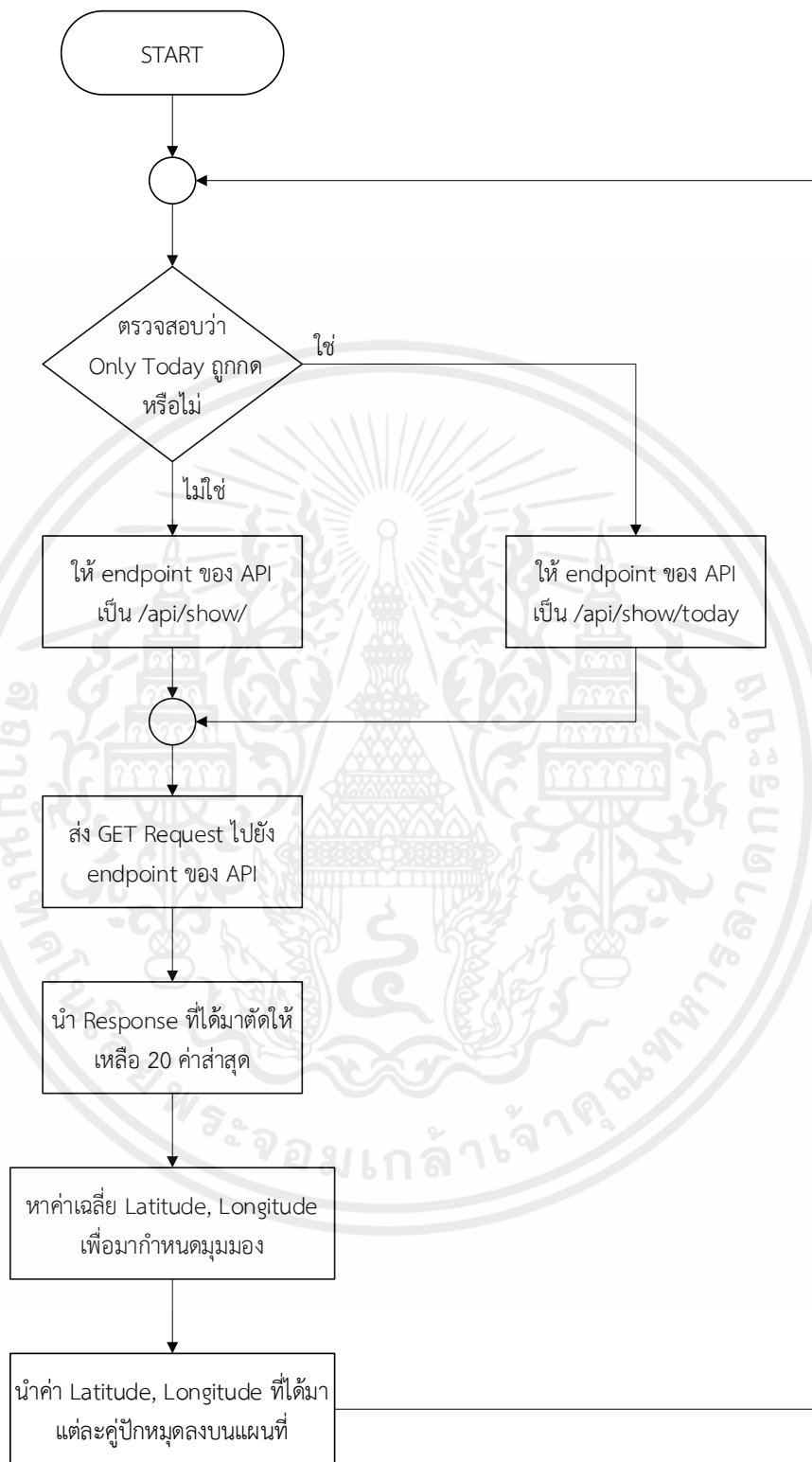
### 3.5.3.7 Location

ในส่วนนี้ของเว็บไซต์จะมีกล่องที่นำค่า Latitude และ Longitude มาแสดงออกในรูปแบบของแผนที่ โดยจะทำการปักหมุดลงไปที่ตำแหน่งที่ระบุนั้น ผู้ใช้สามารถนำเมาส์ไปวางหรือกดดูที่หมุดนั้น เพื่อแสดงข้อมูลที่เกี่ยวข้อง เช่น เวลา, อุณหภูมิ, และความชื้นที่อยู่ในตำแหน่งนั้น โดยหมุดที่ถูกปักลงบนแผนที่จะแสดงข้อมูลจาก 20 ค่าล่าสุดที่ได้รับมาโดยการทำงานของเว็บไซต์ในส่วนนี้ จะถูกแสดงในรูปแบบด้านล่าง



รูปที่ 3.60 รูปแสดงกล่องข้อมูลตำแหน่งของ Gateway

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.61 แผนผังแสดงขั้นตอนการทำงานของการทำงานของการปักหมุดลงแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function drawmap(data) {
  var latitudes = mapDataValues(data, "Latitude").slice(-20);
  var longitudes = mapDataValues(data, "Longitude").slice(-20);
  let sum = 0;

  for (let i = 0; i < latitudes.length; i++) {
    sum += latitudes[i];
  }

  const averagelat = (sum / latitudes.length).toFixed(6);
  sum = 0;

  for (let i = 0; i < longitudes.length; i++) {
    sum += longitudes[i];
  }

  const averagelon = (sum / longitudes.length).toFixed(6);

  map = L.map("map").setView([averagelat, averagelon], 20);
  L.tileLayer("https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png").addTo(map);

  for (var i = 0; i < latitudes.length; i++) {
    if (latitudes[i] !== null && longitudes[i] !== null) {
      var marker = L.marker([latitudes[i], longitudes[i]]).addTo(map);
      marker.bindPopup(`Location ${i + 1}`).openPopup();
    }
  }
}

```

รูปที่ 3.62 รูปแสดงกล่องข้อมูลตำแหน่งของ Gateway

### 3.6 เครื่องมือที่ใช้ในการทดลอง

โครงการนี้มีอุปกรณ์และเครื่องมือที่ใช้ในการทดลองดังนี้

#### 3.6.1 ไมโครคอนโทรลเลอร์

ในปริญญาานิพนธ์นี้ ผู้จัดทำเลือกใช้ ESP32 Lilygo T-SIM7670E เป็นบอร์ดไมโครคอนโทรลเลอร์ที่ใช้ทำ Gateway โดยมีการใช้โมดูลสำหรับใช้ส่งและรับข้อมูลผ่าน LoRa กับโมดูลระบุตำแหน่งพิกัด GPS ทำงานร่วมกับบอร์ดไมโครคอนโทรลเลอร์และบอร์ดไมโครคอนโทรลเลอร์ ESP 32 เป็นบอร์ดไมโครคอนโทรลเลอร์ที่ใช้ทำ End Device โดยมีการใช้โมดูลสำหรับใช้ส่งและรับข้อมูลผ่าน LoRa โมดูลสำหรับวัดอุณหภูมิและความชื้น และหลอด LED ทำงานร่วมกับบอร์ดไมโครคอนโทรลเลอร์

### 3.6.2 คอมพิวเตอร์บอร์ดเดี่ยว (Single-board computer)

ในปฏิญานพนธ์นี้ ผู้จัดทำเลือกใช้ raspberry pi3 เป็นคอมพิวเตอร์บอร์ดเดี่ยว โดยใช้เป็นตัวเปิด Server สำหรับการใช้ API และเก็บข้อมูล Database สำหรับการใช้งาน

### 3.6.3 โมดูลระบุตำแหน่งพิกัด GPS และอุปกรณ์วัดระยะทาง

ในปฏิญานพนธ์นี้ ผู้จัดทำเลือกใช้โมดูลระบุตำแหน่งพิกัด GPS รุ่น GPS-NEO-6M-001 เชื่อมต่อกับบอร์ดไมโครคอนโทรลเลอร์ ESP32 Lilygo T-SIM7670E เพื่อระบุตำแหน่งของตัว Gateway พร้อมกันนั้นยังใช้ SparkFun RTK Surveyor ในการวัดระยะทางในการทดสอบ

### 3.6.4 โมดูลสำหรับใช้ส่งและรับข้อมูลผ่าน LoRa

ในปฏิญานพนธ์นี้ ผู้จัดทำเลือกใช้โมดูลสำหรับใช้ส่งและรับข้อมูลผ่าน LoRa รุ่น SX1276 ย่านความถี่ 900-931 เมกะเฮิร์ตซ์เชื่อมต่อกับบอร์ดไมโครคอนโทรลเลอร์ ESP32 Lilygo T-SIM7670E และบอร์ดไมโครคอนโทรลเลอร์ ESP 32 เพื่อส่งข้อมูลหากันระหว่างบอร์ดไมโครคอนโทรลเลอร์

### 3.6.5 โมดูลสำหรับวัดอุณหภูมิและความชื้น

ในปฏิญานพนธ์นี้ ผู้จัดทำเลือกใช้ AHT10 Digital Temperature and Humidity Sensor ในการวัดอุณหภูมิ โดยเชื่อมต่อกับบอร์ดไมโครคอนโทรลเลอร์ ESP 32 และมีการรับส่งข้อมูลแบบ I2C ระหว่างบอร์ดไมโครคอนโทรลเลอร์กับโมดูลสำหรับวัดอุณหภูมิและความชื้น เพื่อให้ได้ค่าอุณหภูมิ

### 3.6.6 เครื่องมือสำหรับวัดสัญญาณ LoRa

ในปฏิญานพนธ์นี้ ผู้จัดทำเลือกใช้ RTL-SDR โดยการต่อเสาอากาศในย่านที่ต้องการรับบนเครื่องมือแล้วทำการเชื่อมต่อกับคอมพิวเตอร์แล้วทำการวัดสัญญาณ LoRa เพื่อนำรูปที่ทดสอบได้มาแสดงบนรายงาน

### 3.6.7 แบตเตอรี่

ในปฏิญานพนธ์นี้ ผู้จัดทำเลือกใช้แบตเตอรี่ 18650 Lithium Battery เพื่อเป็นแหล่งจ่ายไฟให้กับบอร์ดไมโครคอนโทรลเลอร์ ESP32 Lilygo T-SIM7670E และบอร์ดไมโครคอนโทรลเลอร์ ESP 32 โดยแต่ละบอร์ดไมโครคอนโทรลเลอร์จะจ่ายไฟให้กับโมดูลต่างๆ ตามลำดับดังนี้

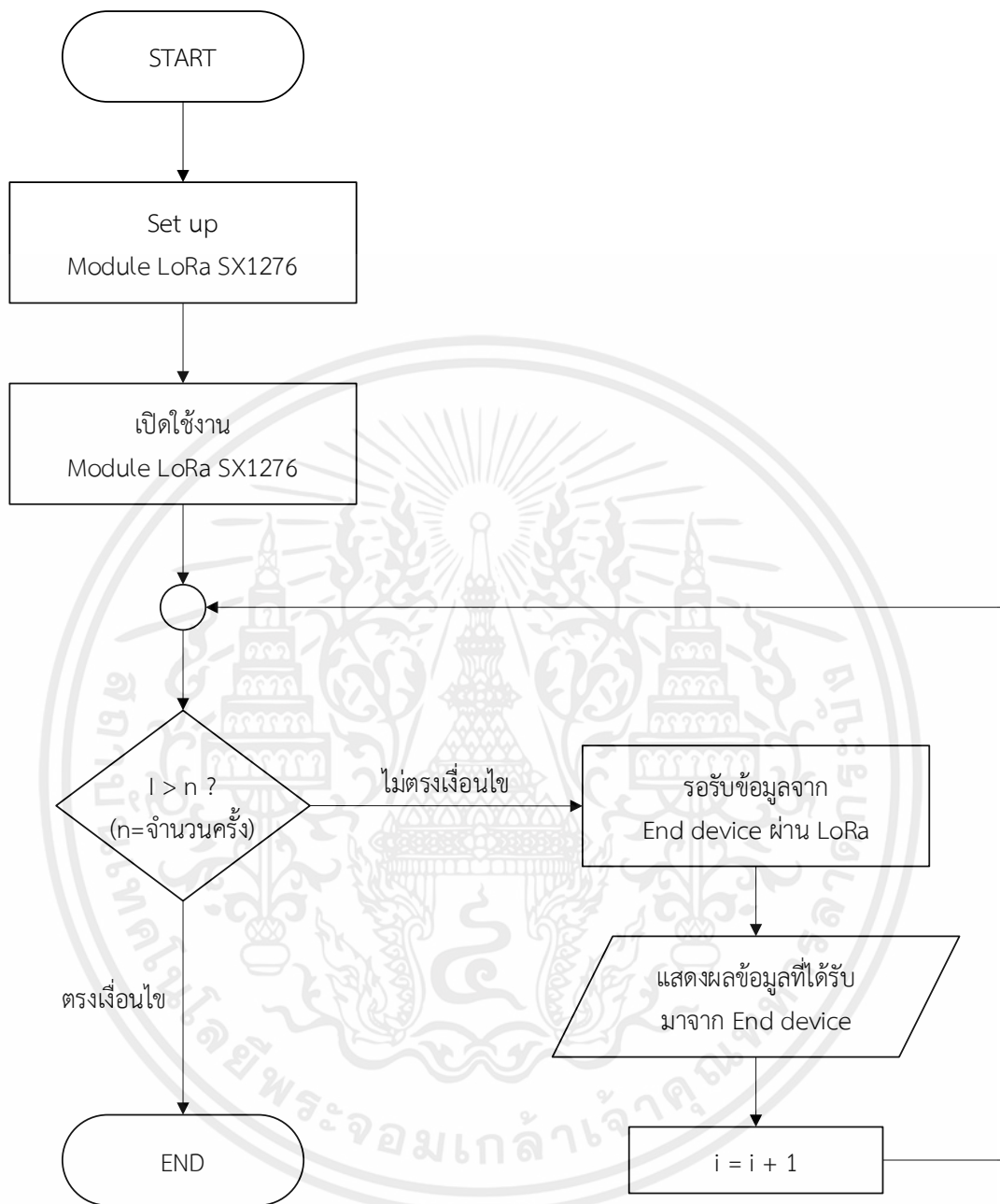
- 1) บอร์ดไมโครคอนโทรลเลอร์ ESP32 Lilygo T-SIM7670E จะจ่ายไฟต่อไปยังโมดูล สำหรับใช้ส่งและรับข้อมูลผ่าน LoRa และโมดูลระบุตำแหน่งพิกัด GPS
- 2) บอร์ดไมโครคอนโทรลเลอร์ ESP 32 จะจ่ายไฟต่อไปยังหลอด LED ,โมดูล AHT10 และโมดูลสำหรับใช้ส่งและรับข้อมูลผ่าน LoRa

### 3.7 การจัดเก็บผลการทดลอง

#### 3.7.1 การออกแบบการทดสอบอุปกรณ์ทางฝั่ง END DEVICES

##### 3.7.1.1 การออกแบบทดสอบการทำงานของระบบ LORA

แผนผังแสดงขั้นตอนการทดสอบของโปรแกรม — LoRa แผนผังเริ่มต้นด้วยการ กำหนดค่าโมดูล LoRa SX1276 จากนั้นเปิดใช้งานโมดูล LoRa SX1276 เมื่อโมดูล LoRa SX1276 เปิดใช้งานแล้ว โปรแกรมจะเข้าสู่ลูป ลูปนี้จะวนซ้ำไปเรื่อย ๆ จนกว่าเงื่อนไขจะตรงกัน ในลูป โปรแกรมจะรอรับข้อมูลจาก End Device หากได้รับข้อมูล โปรแกรมจะแสดงผลข้อมูลนั้น หากจำนวนครั้งที่โปรแกรมทำงานไปแล้วมากกว่าหรือเท่ากับจำนวนครั้งที่กำหนดไว้ โปรแกรมจะสิ้นสุดการทำงานดังแสดงในรูปที่ 3.63



รูปที่ 3.63 แผนผังแสดงขั้นตอนการทดสอบของโปรแกรม LoRa

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.7.1.2 การออกแบบทดสอบดูการส่งข้อมูลของแต่ละ PACKET ที่ได้รับ เมื่อมีสิ่งกีดขวาง

ในส่วนนี้เราทำการส่งข้อมูลจากอุปกรณ์ End devices ไปยัง Gateway โดยข้อมูลประกอบด้วยจำนวนแพ็กเก็ตที่ถูกส่งมาจากฝั่งส่ง ตามด้วยข้อมูลเกี่ยวกับอุณหภูมิ, ความชื้น, และค่าความแรงสัญญาณ (RSSI) ตามลำดับ กระบวนการส่งข้อมูลนี้จะดำเนินต่อไปเรื่อย ๆ จนกว่าฝั่งรับจะได้รับข้อมูลทั้งหมด 20 ครั้ง โดยที่ข้อมูลทั้งหมดจะถูกบันทึกลงในไฟล์ txt เพื่อให้ง่ายต่อการวิเคราะห์เพิ่มเติม หลังจากการบันทึกข้อมูลเสร็จสิ้น จะทำการเปลี่ยนค่า SF (Spreading Factor) และทำซ้ำขั้นตอนทั้งหมดอีกครั้ง

การทดลองนี้ถูกแบ่งเป็น 2 ส่วน คือการทดลองเมื่อเปิดใช้งาน CRC และการทดลองเมื่อไม่เปิดใช้งาน CRC เพื่อทดสอบความมีประสิทธิภาพในทั้งสองกรณี กระบวนการนี้ช่วยให้เราสามารถปรับปรุงและปรับแต่งการสื่อสารในระบบ LoRa ให้มีประสิทธิภาพและมีประสิทธิภาพมากยิ่งขึ้นในอนาคต

- 1) การทดสอบแสดงผล LOG PACKET ของแต่ละ SF (มี CRC)
- 2) การทดสอบแสดงผล LOG PACKET ของแต่ละ SF (ไม่มีCRC)

### 3.7.1.3 การออกแบบทดสอบความเร็วในการส่งข้อมูลที่แต่ละ SF และ Bandwidth

ในส่วนของการทดลองนี้ได้วางแผนการทำงานของโปรแกรมตามขั้นตอนดังนี้คือ แรกเริ่มจากการเปิดใช้งาน LoRa module และสร้างตัวแปร String ขนาด 255 ตัวอักษร หลังจากนั้น เราทำการส่งข้อมูลที่เตรียมไว้ผ่าน LoRa โดยนับเวลาที่ใช้ในการส่งข้อมูล เพื่อนำเอาเวลานี้และจำนวนตัวอักษรที่ถูกแปลงเป็นบิตมาคำนวณหาความเร็วในการส่งข้อมูลต่อไป หลังจากนั้นทำการปรับแบนด์วิดท์และ Spreading Factor (SF) และทำการทดลองซ้ำขั้นตอนนี้ เพื่อวัดผล แสดงตามรูปที่ 3.64



รูปที่ 3.64 แผนผังการออกแบบทดสอบความเร็วในการส่งข้อมูลที่แต่ละ SF

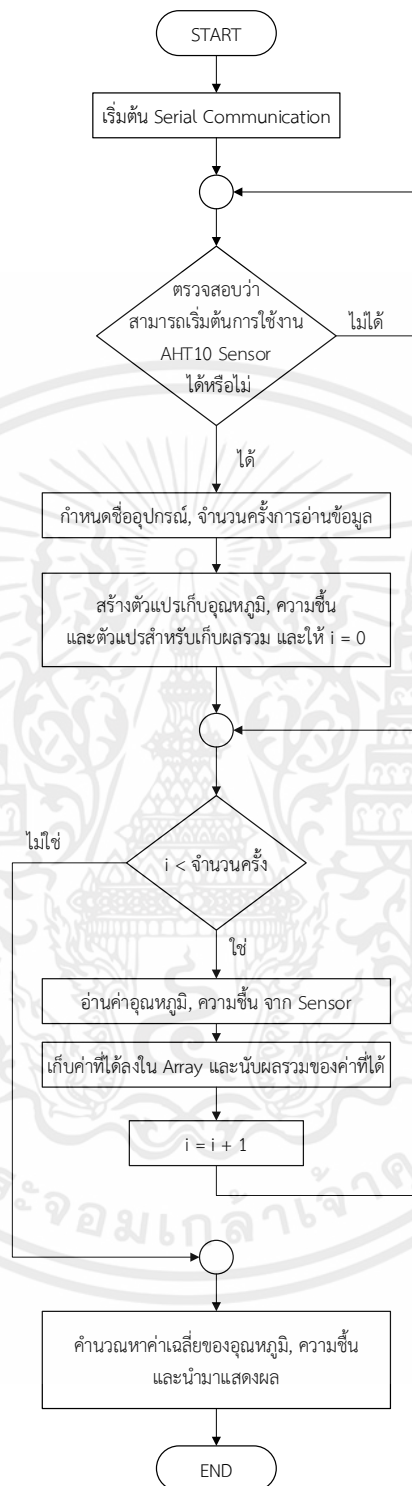
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.7.1.4 การออกแบบทดสอบวัดระยะทางที่ไกลที่สุดที่รับและส่งข้อมูลได้

ในส่วนของการทดลองนี้ได้ดำเนินการเปิด End Device เพื่อให้ส่งข้อมูลตลอดเวลา และทำการขยับ Gateway พร้อมกับ RTK Surveyor ให้ห่างออกไปเรื่อย ๆ โดยในขณะที่ขยับ ได้ตรวจสอบว่าข้อมูลถูกรับครบถ้วนหรือไม่ จนกระทั่งไม่สามารถรับข้อมูลได้อีก จากนั้นทำการหาตำแหน่งปัจจุบันจากเครื่อง RTK Surveyor ที่มีความแม่นยำระดับเซนติเมตร และนำตำแหน่งนั้นมาเทียบกับตำแหน่งที่ End Device ถูกวางไว้ เพื่อนำไปคำนวณหาระยะทางต่อไป

### 3.7.1.5 การออกแบบการทดสอบความแม่นยำของเซ็นเซอร์

ในการทดลองนี้ได้ทำการวัดความแม่นยำของเซ็นเซอร์ โดยการให้ End device แต่ละตัวทำการวัดอุณหภูมิและความชื้นจากเซ็นเซอร์จำนวน 100 ครั้ง แล้วจึงนำค่าทั้งหมดเหล่านี้มาหาค่าเฉลี่ย และเทียบกับค่าที่ได้จากอุปกรณ์วัดจริง เพื่อหาค่าความผิดพลาดของแต่ละ End device และเพื่อตรวจสอบความแม่นยำของเซ็นเซอร์ที่ติดอยู่กับ End device แต่ละตัวว่ามีค่าความคลาดเคลื่อนอย่างไร การทดสอบนี้ช่วยให้เข้าใจได้ว่าเซ็นเซอร์ใน End device แต่ละตัวมีความแม่นยำอย่างไร เมื่อเปรียบเทียบกับเครื่องวัดจริง ๆ ซึ่งจะช่วยในการปรับปรุงและปรับแก้ปัญหาที่เกี่ยวข้องกับความแม่นยำของเซ็นเซอร์ได้อย่างมีประสิทธิภาพ โดยในการทดสอบได้มีการออกแบบโปรแกรมให้ทำงานตาม Flowchart ที่แสดงดังรูปที่ 3.65



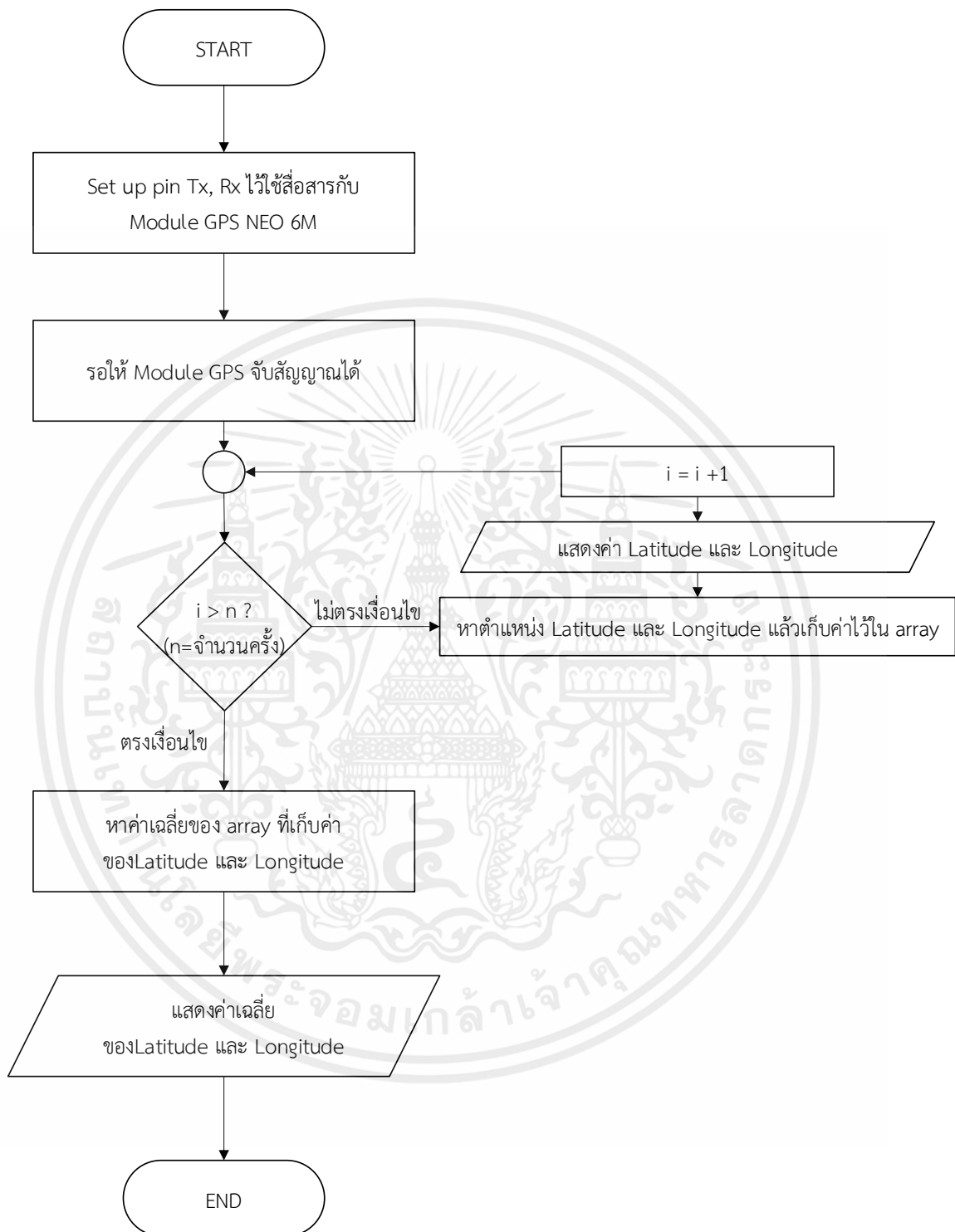
รูปที่ 3.65 แผนผังแสดงขั้นตอนการทำงานของโปรแกรมสำหรับการทดสอบหาความแม่นยำของเซ็นเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.7.2 การออกแบบการทดสอบอุปกรณ์ทางฝั่ง LORA GATEWAY

3.7.2.1 การออกแบบทดสอบปรับค่าสัญญาณเพื่อดูความแม่นยำของ GPS ที่เชื่อมต่อกับ Gateway

แผนผังแสดงขั้นตอนการทำงานของการทำงานของการหาตำแหน่ง GPS โดยใช้โมดูล GPS NEO 6M แผนผังเริ่มต้นด้วยการตั้งค่าพิน Tx และ Rx บนโมดูล GPS NEO 6M จากนั้นโมดูล GPS NEO 6M จะเริ่มหาสัญญาณดาวเทียมเมื่อโมดูล GPS NEO 6M จับสัญญาณดาวเทียมได้สำเร็จ โปรแกรมจะหาตำแหน่ง Latitude และ Longitude ของอุปกรณ์ จากนั้นจะเก็บค่า Latitude และ Longitude ไว้ใน array และจะวนซ้ำไปเรื่อยๆ จนกว่าจำนวนครั้งจะเท่ากับจำนวนที่กำหนดไว้ ในแต่ละรอบ และจะแสดงค่าตำแหน่ง Latitude และ Longitude เมื่อจำนวนครั้งครบตามที่กำหนดไว้ จะหาค่าเฉลี่ยของค่า Latitude และ Longitude ใน array แสดงค่าเฉลี่ยของค่า Latitude และ Longitude บนหน้าจอแสดงในรูปที่ 3.66



รูปที่ 3.66 แผนผังแสดงขั้นตอนการทำงานของการทำงานของการหาค่าตำแหน่ง GPS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนนี้ทำการนำข้อมูลจาก Gateway โดยข้อมูลประกอบด้วยตำแหน่ง GPS เป็นค่าละติจูด (latitude) และลองจิจูด (longitude) จะทำการเก็บข้อมูลทั้งหมด 20 ครั้ง โดยที่ข้อมูลทั้งหมดจะถูกบันทึกลงในไฟล์ txt เพื่อให้ง่ายต่อการวิเคราะห์เพิ่มเติม หลังจากการบันทึกข้อมูลเสร็จสิ้น เราจะทำการเปลี่ยนค่า SF (Spreading Factor) และทำซ้ำขั้นตอนทั้งหมดอีกครั้ง

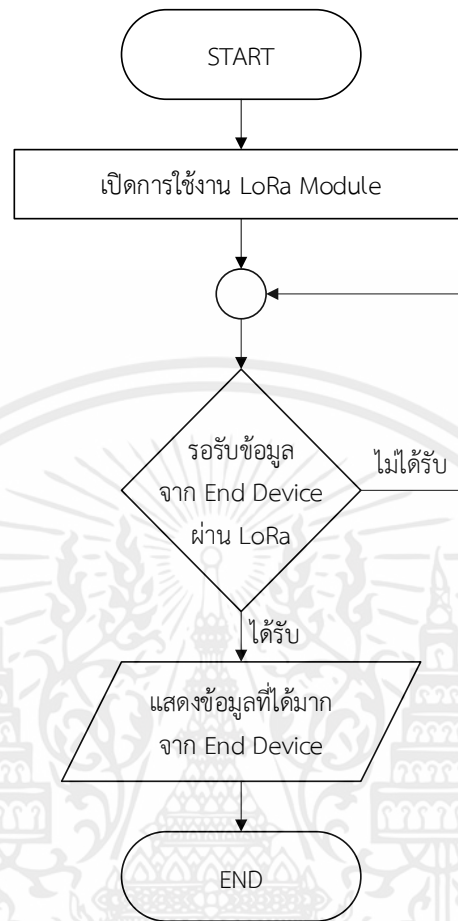
การทดลองนี้ถูกแบ่งเป็น 2 ส่วน คือการทดลองรับค่าสัญญาณ GPS เมื่ออยู่กับที่และการทดลองรับค่าสัญญาณ GPS เมื่อทำการเคลื่อนที่ เพื่อทดสอบความแม่นยำของทั้งสองกรณี กระบวนการนี้ช่วยให้เราสามารถปรับปรุงและปรับแต่งการสื่อสารในระบบ LoRa ให้มีประสิทธิภาพและมีประสิทธิภาพมากยิ่งขึ้นในอนาคต

1) การทดสอบรับค่าสัญญาณ GPS เมื่ออยู่กับที่

2) การทดสอบรับค่าสัญญาณ GPS เมื่อทำการเคลื่อนที่

### 3.7.2.2 การออกแบบการทดสอบการรับข้อมูลที่ส่งมาจากฝั่ง End Device

ในส่วนของการทดลองนี้เราได้เขียนคำสั่งการทำงานของโปรแกรมโดยมีลำดับการทำงานดังนี้คือ เปิดการใช้งาน LoRa module และทำการรอรับข้อมูลที่จะถูกส่งมาจากทางฝั่งของ End Device แล้วจึงนำข้อมูลเหล่านี้มาแสดงผลเพื่อให้สามารถตรวจสอบความถูกต้องและความสมบูรณ์ของข้อมูลที่ได้รับ โดยลำดับการทำงานทั้งหมดจะถูกแสดงไว้ใน flowchart แสดงดังรูปที่ 3.67

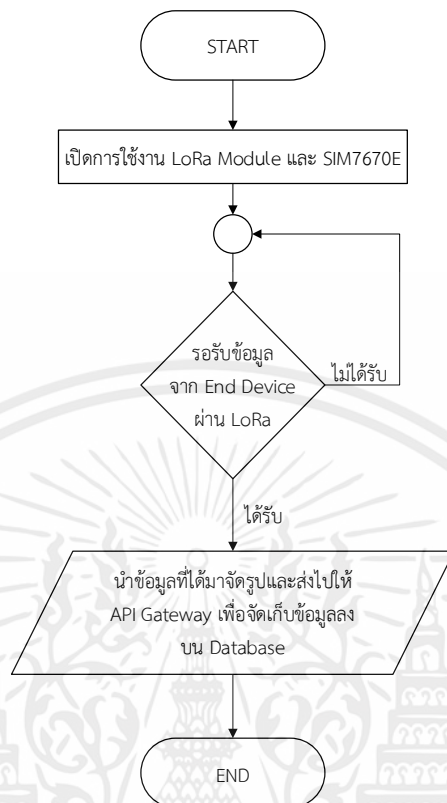


รูปที่ 3.67 แผนผังการออกแบบการทดสอบการรับข้อมูลที่ส่งมาจากฝั่ง End Device

### 3.7.2.3 การออกแบบการทดสอบการส่งข้อมูลทั้งหมดที่รวบรวมได้ขึ้น

Database

ในส่วนของการทดลองนี้เราได้เขียนคำสั่งการทำงานของโปรแกรมโดยมีลำดับการทำงานดังนี้คือ เปิดการใช้งาน LoRa module และ SIM7670E จากนั้นจะทำการรอรับข้อมูลที่จะถูกส่งมาจากทางฝั่งของ End Device เมื่อได้รับข้อมูลแล้วจะทำการจัดรูปข้อมูลแล้วส่งต่อไปที่ API Gateway โดยที่ฝั่งนั้นจะทำการนำข้อมูลที่ได้ไปจัดเก็บเอาไว้บน Database ซึ่งการทดสอบนี้ทำขึ้นเพื่อทดสอบดูการทำงานว่า Gateway สามารถส่งข้อมูลไปจัดเก็บที่ Database ได้อย่างเรียบร้อยดีหรือไม่ โดยลำดับการทำงานทั้งหมดจะถูกแสดงไว้ใน flowchart แสดงดังรูปที่ 3.68



รูปที่ 3.68 แผนผังการออกแบบการทดสอบการส่งข้อมูลทั้งหมดที่รวบรวมได้ขึ้น Database

### 3.7.3 การออกแบบการทดสอบการใช้งานเว็บไซต์

ในกระบวนการทดสอบนี้ เราได้ทำการเปิดใช้งาน Node และ Gateway เพื่อเปิดทางให้แต่ละ Node สามารถส่งข้อมูลมายัง Gateway และทำให้ Gateway สามารถส่งข้อมูลต่อไปยังเซิร์ฟเวอร์ได้โดยตรง หลังจากที่ข้อมูลถูกส่งไปยังเซิร์ฟเวอร์ จะมีการจัดเก็บข้อมูลเพื่อนำมาแสดงผลบนเว็บไซต์ ทำให้ผู้ใช้สามารถดู ข้อมูลต่าง ๆ ที่ได้รับจากเซิร์ฟเวอร์ได้ตามข้อมูลของผู้ใช้งานแต่ละคน รวมถึงการทดลองการใช้งานหน้า admin เพื่อจัดการกับข้อมูลผู้ใช้ทั้งหมด

### 3.7.4 การออกแบบการทดสอบการใช้งานจริง

ในส่วนนี้ได้ทำการนำอุปกรณ์ที่ออกแบบมาไปทดสอบในสถานที่จริงโดยการให้ End Devices ตรวจวัดค่าสภาพแวดล้อมตามสถานที่ต่างๆ และส่งข้อมูลไปที่ Gateway เพื่อหาตำแหน่ง จัดระเบียบข้อมูล และส่งข้อมูลเหล่านั้นขึ้นสู่ฐานข้อมูล หลังจากนั้นเราจะนำข้อมูลเหล่านั้นมาวิเคราะห์เพื่อประเมินประสิทธิภาพในการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการทดลอง

ผู้จัดทำได้ทำการเก็บผลการทำงานของระบบ โดยแบ่งการทดลองและจัดเก็บผลการทดลองเป็นส่วน ๆ ดังต่อไปนี้

#### 4.1 การทดสอบอุปกรณ์ทางฝั่ง End Devices

##### 4.1.1 การทดสอบการทำงานของระบบ LoRa

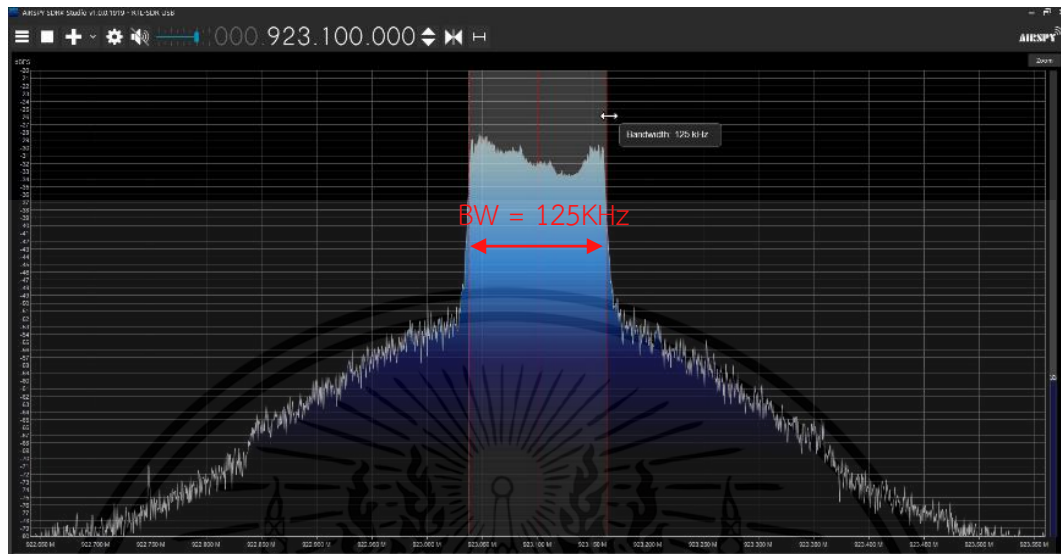
ในการทดสอบนี้เราใช้ข้อมูลจาก RTL-SDR Dongle เพื่อดูว่า Spreading Factor (SF) ในระบบ LoRa ซึ่งมีค่าตั้งแต่ 7 ถึง 12 จะมีผลต่อสัญญาณอย่างไร โดยทั่วไป ค่า SF ที่สูงขึ้น จะทำให้การกระจายข้อมูลเกิดขึ้นมากขึ้น และจะมีผลต่อระยะทางการส่งข้อมูล ความเร็วในการสื่อสาร และการใช้พลังงานของระบบ ซึ่งในการทดสอบนี้เราจะดูความสัมพันธ์เหล่านี้ได้จากกราฟ Waterfall ซึ่งจะมีแกนตั้งเป็นเวลาและแกนนอนเป็นความถี่ ข้อมูลนี้มีประโยชน์สำหรับการตรวจสอบสัญญาณในระบบ LoRaจากการทดสอบ Bandwidth และ Frame ที่ได้



รูปที่ 4.1 กล้อง End Device กับ SDR

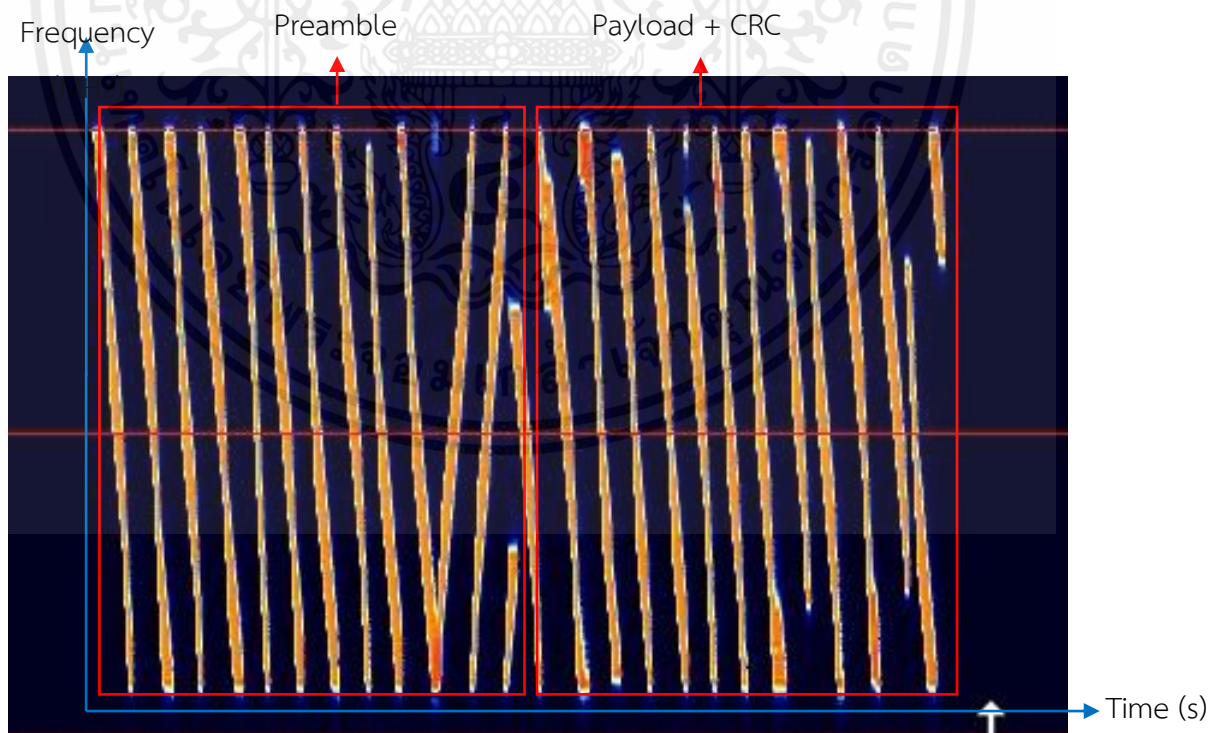
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.1.1 จากการทดสอบ Bandwidth ที่ได้



รูปที่ 4.2 Bandwidth ที่ได้จากการทดสอบ

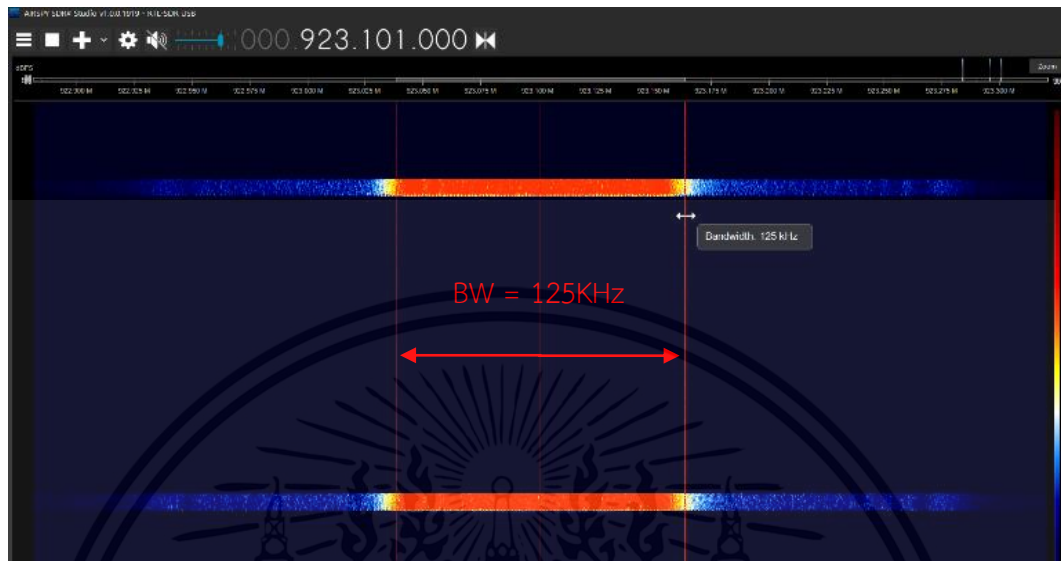
#### 4.1.1.2 จากการทดสอบ Frame ที่ได้



รูปที่ 4.3 Frame ที่ได้จากการทดสอบ

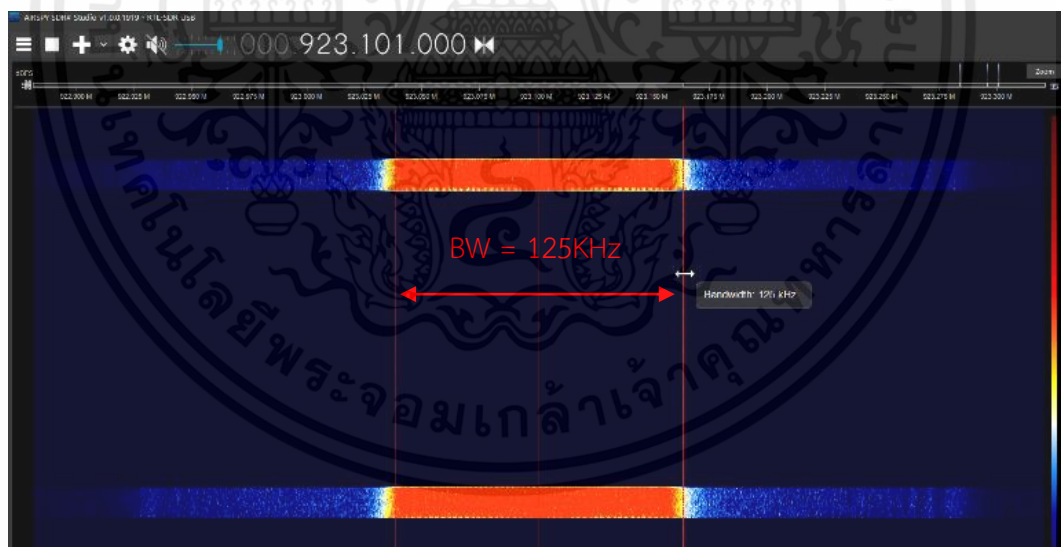
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.1.3 ที่ Spreading Factor (SF) = 7



รูปที่ 4.4 Spreading Factor (SF) = 7

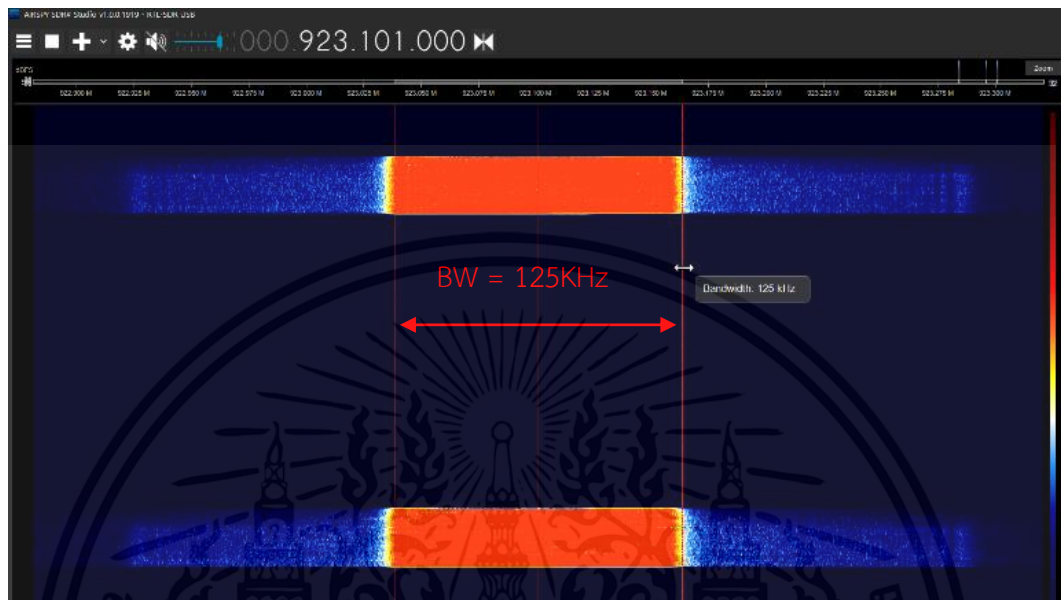
#### 4.1.1.4 ที่ Spreading Factor (SF) = 8



รูปที่ 4.5 Spreading Factor (SF) = 8

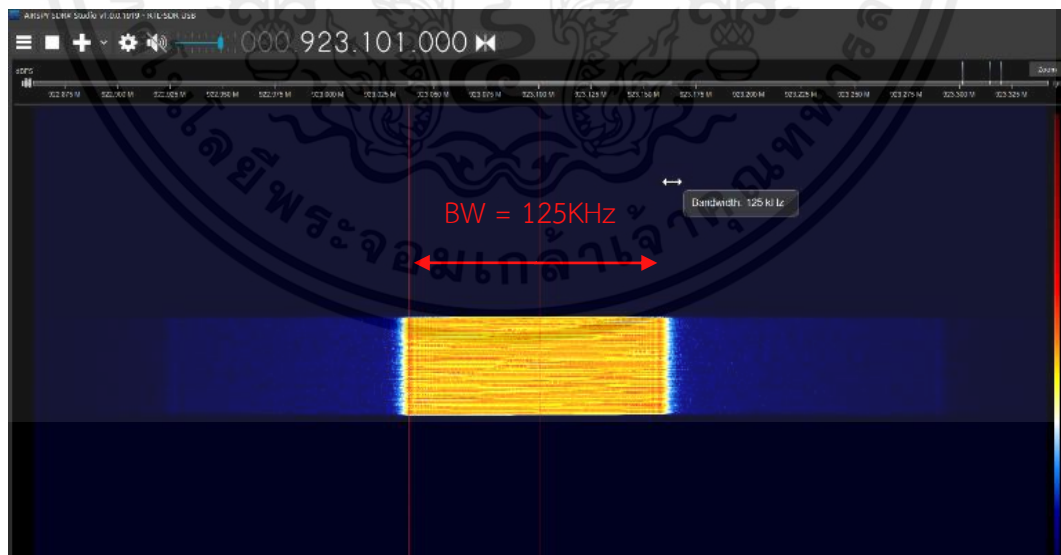
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.1.5 ที่ Spreading Factor (SF) = 9



รูปที่ 4.6 Spreading Factor (SF) = 9

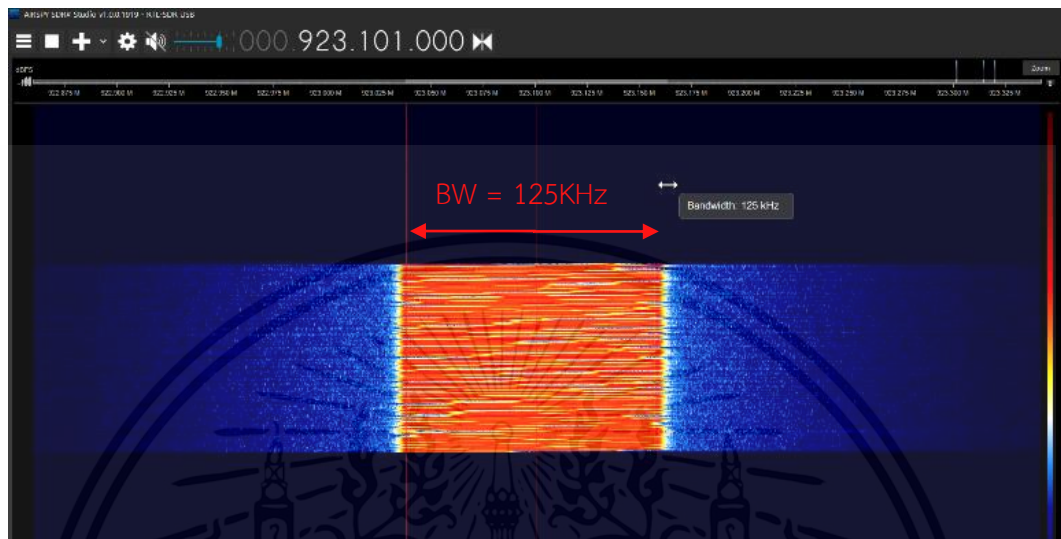
#### 4.1.1.6 Spreading Factor (SF) = 10



รูปที่ 4.7 Spreading Factor (SF) = 10

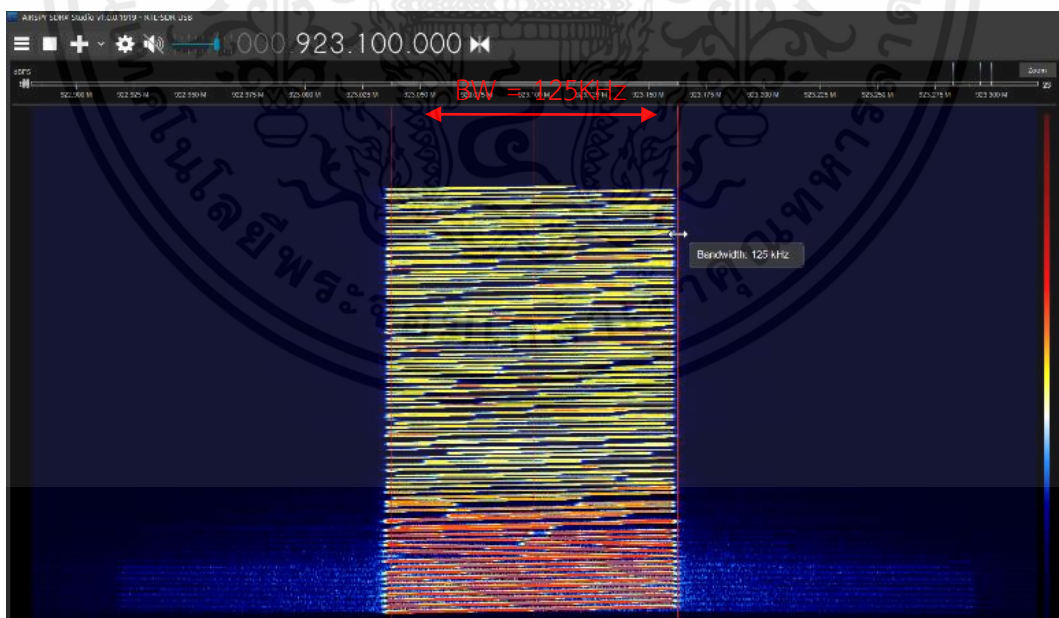
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.1.7 ที่ Spreading Factor (SF) = 11



รูปที่ 4.8 Spreading Factor (SF) = 11

#### 4.1.1.8 ที่ Spreading Factor (SF) = 12

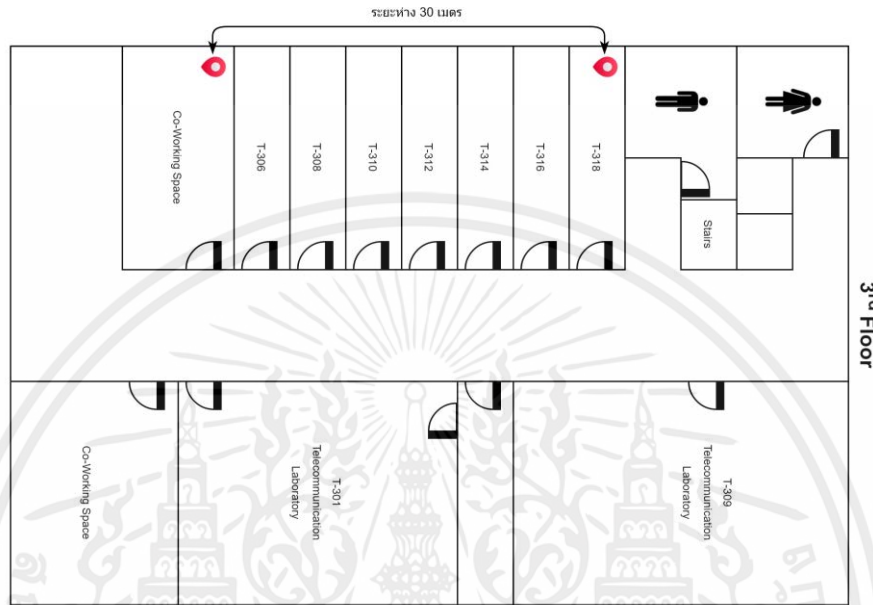


รูปที่ 4.9 Spreading Factor (SF) = 12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.1.2 การทดสอบการส่งข้อมูลแต่ละ Packet ที่ได้รับเมื่อมีสิ่งกีดขวาง

#### 4.1.2.1 การทดสอบแสดงผล Log Packet ของแต่ละ SF (มี CRC)



รูปที่ 4.10 การทดสอบรับและส่งข้อมูล Log Packet

ในการทดสอบมีระยะทางอยู่ที่ 30 เมตร และมีกำแพงกั้นอยู่ 7 ชั้นจากการทดสอบ  
หา Log Packet

1) ที่ค่า Spreading Factor = 7 (มี CRC)

```

SyncWord: F1
Frequency: 923000000
SpreadingFactor: 7 (CRC)
SignalBandwidth: 125000
LoRa Initializing OK!

Packet Test from 20 values
Packet #1 Received packet '{"nodeName":"Node1","PacketID":0,"Temperature":29.11,"Humidity":65.86}' with RSSI -98
Packet #2 Received packet '{"nodeName":"Node1","PacketID":1,"Temperature":29.15,"Humidity":66.37}' with RSSI -97
Packet #3 Received packet '{"nodeName":"Node1","PacketID":2,"Temperature":29.18,"Humidity":66.77}' with RSSI -97
Packet #4 Received packet '{"nodeName":"Node1","PacketID":4,"Temperature":29.22,"Humidity":67.12}' with RSSI -97
Packet #5 Received packet '{"nodeName":"Node1","PacketID":6,"Temperature":29.25,"Humidity":67.18}' with RSSI -97
Packet #6 Received packet '{"nodeName":"Node1","PacketID":7,"Temperature":29.26,"Humidity":67.21}' with RSSI -97
Packet #7 Received packet '{"nodeName":"Node1","PacketID":8,"Temperature":29.28,"Humidity":67.18}' with RSSI -97
Packet #8 Received packet '{"nodeName":"Node1","PacketID":10,"Temperature":29.3,"Humidity":67.15}' with RSSI -97
Packet #9 Received packet '{"nodeName":"Node1","PacketID":11,"Temperature":29.32,"Humidity":67.17}' with RSSI -97
Packet #10 Received packet '{"nodeName":"Node1","PacketID":12,"Temperature":29.33,"Humidity":67.19}' with RSSI -98
Packet #11 Received packet '{"nodeName":"Node1","PacketID":13,"Temperature":29.35,"Humidity":67.21}' with RSSI -97
Packet #12 Received packet '{"nodeName":"Node1","PacketID":14,"Temperature":29.36,"Humidity":67.24}' with RSSI -97
Packet #13 Received packet '{"nodeName":"Node1","PacketID":17,"Temperature":29.4,"Humidity":67.28}' with RSSI -98
Packet #14 Received packet '{"nodeName":"Node1","PacketID":27,"Temperature":29.53,"Humidity":67.29}' with RSSI -97
Packet #15 Received packet '{"nodeName":"Node1","PacketID":34,"Temperature":29.6,"Humidity":67.22}' with RSSI -98
Packet #16 Received packet '{"nodeName":"Node1","PacketID":35,"Temperature":29.61,"Humidity":67.22}' with RSSI -98
Packet #17 Received packet '{"nodeName":"Node1","PacketID":37,"Temperature":29.63,"Humidity":67.23}' with RSSI -97
Packet #18 Received packet '{"nodeName":"Node1","PacketID":38,"Temperature":29.63,"Humidity":67.23}' with RSSI -97
Packet #19 Received packet '{"nodeName":"Node1","PacketID":41,"Temperature":29.66,"Humidity":67.24}' with RSSI -98
Packet #20 Received packet '{"nodeName":"Node1","PacketID":42,"Temperature":29.67,"Humidity":67.21}' with RSSI -97
    
```

รูปที่ 4.11 Log ของแต่ละ Packet ที่ได้รับเมื่อมีค่า Spreading Factor = 7 (มี CRC)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) จากการทดสอบหา Log Packet ที่ค่า Spreading Factor =

9 (มี CRC)

```

SyncWord: F1
Frequency: 923000000
SpreadingFactor: 9 (CRC)
SignalBandwidth: 125000
LoRa Initializing OK!

Packet Test from 20 values

Packet #1 Received packet '{"nodeName":"Node1","PacketID":0,"Temperature":28.94,"Humidity":68.25}' with RSSI -96
Packet #2 Received packet '{"nodeName":"Node1","PacketID":1,"Temperature":28.98,"Humidity":68.54}' with RSSI -97
Packet #3 Received packet '{"nodeName":"Node1","PacketID":2,"Temperature":29.01,"Humidity":68.69}' with RSSI -97
Packet #4 Received packet '{"nodeName":"Node1","PacketID":3,"Temperature":29.03,"Humidity":68.75}' with RSSI -97
Packet #5 Received packet '{"nodeName":"Node1","PacketID":4,"Temperature":29.05,"Humidity":68.81}' with RSSI -98
Packet #6 Received packet '{"nodeName":"Node1","PacketID":5,"Temperature":29.07,"Humidity":68.84}' with RSSI -98
Packet #7 Received packet '{"nodeName":"Node1","PacketID":6,"Temperature":29.09,"Humidity":68.85}' with RSSI -96
Packet #8 Received packet '{"nodeName":"Node1","PacketID":7,"Temperature":29.11,"Humidity":68.86}' with RSSI -96
Packet #9 Received packet '{"nodeName":"Node1","PacketID":8,"Temperature":29.12,"Humidity":68.86}' with RSSI -96
Packet #10 Received packet '{"nodeName":"Node1","PacketID":9,"Temperature":29.14,"Humidity":68.89}' with RSSI -97
Packet #11 Received packet '{"nodeName":"Node1","PacketID":10,"Temperature":29.15,"Humidity":68.9}' with RSSI -97
Packet #12 Received packet '{"nodeName":"Node1","PacketID":11,"Temperature":29.17,"Humidity":68.93}' with RSSI -96
Packet #13 Received packet '{"nodeName":"Node1","PacketID":12,"Temperature":29.19,"Humidity":68.94}' with RSSI -97
Packet #14 Received packet '{"nodeName":"Node1","PacketID":13,"Temperature":29.2,"Humidity":68.99}' with RSSI -96
Packet #15 Received packet '{"nodeName":"Node1","PacketID":14,"Temperature":29.22,"Humidity":68.96}' with RSSI -98
Packet #16 Received packet '{"nodeName":"Node1","PacketID":15,"Temperature":29.24,"Humidity":68.97}' with RSSI -99
Packet #17 Received packet '{"nodeName":"Node1","PacketID":16,"Temperature":29.25,"Humidity":68.95}' with RSSI -97
Packet #18 Received packet '{"nodeName":"Node1","PacketID":17,"Temperature":29.26,"Humidity":68.99}' with RSSI -98
Packet #19 Received packet '{"nodeName":"Node1","PacketID":19,"Temperature":29.29,"Humidity":68.96}' with RSSI -99
Packet #20 Received packet '{"nodeName":"Node1","PacketID":20,"Temperature":29.3,"Humidity":68.93}' with RSSI -96

```

รูปที่ 4.12 Log ของแต่ละ Packet ที่ได้รับเมื่อมีค่า Spreading Factor = 9 (มี CRC)

3) จากการทดสอบหา Log Packet ที่ค่า Spreading Factor =

12 (มี CRC)

```

SyncWord: F1
Frequency: 923000000
SpreadingFactor: 12 (CRC)
SignalBandwidth: 125000
LoRa Initializing OK!

Packet Test from 20 values

Packet #1 Received packet '{"nodeName":"Node1","PacketID":21,"Temperature":29.47,"Humidity":69.2}' with RSSI -97
Packet #2 Received packet '{"nodeName":"Node1","PacketID":22,"Temperature":29.49,"Humidity":69.13}' with RSSI -98
Packet #3 Received packet '{"nodeName":"Node1","PacketID":23,"Temperature":29.51,"Humidity":69.08}' with RSSI -97
Packet #4 Received packet '{"nodeName":"Node1","PacketID":24,"Temperature":29.53,"Humidity":68.99}' with RSSI -97
Packet #5 Received packet '{"nodeName":"Node1","PacketID":25,"Temperature":29.55,"Humidity":68.94}' with RSSI -96
Packet #6 Received packet '{"nodeName":"Node1","PacketID":26,"Temperature":29.57,"Humidity":68.85}' with RSSI -97
Packet #7 Received packet '{"nodeName":"Node1","PacketID":27,"Temperature":29.6,"Humidity":68.83}' with RSSI -98
Packet #8 Received packet '{"nodeName":"Node1","PacketID":28,"Temperature":29.61,"Humidity":68.76}' with RSSI -97
Packet #9 Received packet '{"nodeName":"Node1","PacketID":29,"Temperature":29.63,"Humidity":68.73}' with RSSI -96
Packet #10 Received packet '{"nodeName":"Node1","PacketID":30,"Temperature":29.66,"Humidity":68.73}' with RSSI -98
Packet #11 Received packet '{"nodeName":"Node1","PacketID":31,"Temperature":29.68,"Humidity":68.73}' with RSSI -96
Packet #12 Received packet '{"nodeName":"Node1","PacketID":32,"Temperature":29.7,"Humidity":68.68}' with RSSI -97
Packet #13 Received packet '{"nodeName":"Node1","PacketID":33,"Temperature":29.73,"Humidity":68.72}' with RSSI -97
Packet #14 Received packet '{"nodeName":"Node1","PacketID":34,"Temperature":29.75,"Humidity":68.75}' with RSSI -96
Packet #15 Received packet '{"nodeName":"Node1","PacketID":35,"Temperature":29.77,"Humidity":68.76}' with RSSI -98
Packet #16 Received packet '{"nodeName":"Node1","PacketID":36,"Temperature":29.79,"Humidity":68.71}' with RSSI -97
Packet #17 Received packet '{"nodeName":"Node1","PacketID":37,"Temperature":29.81,"Humidity":68.68}' with RSSI -97
Packet #18 Received packet '{"nodeName":"Node1","PacketID":38,"Temperature":29.83,"Humidity":68.64}' with RSSI -98
Packet #19 Received packet '{"nodeName":"Node1","PacketID":39,"Temperature":29.85,"Humidity":68.63}' with RSSI -97
Packet #20 Received packet '{"nodeName":"Node1","PacketID":40,"Temperature":29.87,"Humidity":68.59}' with RSSI -98

```

รูปที่ 4.13 Log ของแต่ละ Packet ที่ได้รับเมื่อมีค่า Spreading Factor = 12 (มี CRC)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.1.2.2 การทดสอบแสดงผล Log Packet ของแต่ละ SF (ไม่มีCRC)

- 1) จากการทดสอบหา Log Packet ที่ค่า Spreading Factor = 7 (ไม่มี CRC)

```

SyncWord: F1
Frequency: 923000000
SpreadingFactor: 7
SignalBandwidth: 125000
LoRa Initializing OK!

Packet Test from 20 values
Packet #1 Received packet '{"nodeName": "Node1", "PacketID": 64, "Temperature": 29.52, "Humidity": 67.63}' with RSSI -97
Packet #2 Received packet '{"nodeName": "Node1", "PacketID": 70, "Temperature": 29.6, "Humidity": 67.44}' with RSSI -97
Packet #3 Received packet '{"nodeName": "Node1", "PacketID": 71, "Temperature": 29.6, "Humidity": 67.4}' with RSSI -96
Packet #4 Received packet '{"nodeName": "Node1", "PacketID": 76, "Temperature": 29.66, "Humidity": 67.22}' with RSSI -96
Packet #5 Received packet '{"nodeName": "Node1", "PacketID": 77, "Temperature": 29.67, "Humidity": 67.2}' with RSSI -97
Packet #6 Received packet '{"nodeName": "Node1", "PacketID": 78, "Temperature": 29.68, "Humidity": 67.17}' with RSSI -97
Packet #7 Received packet '{"nodeName": "Node1", "PacketID": 80, "Temperature": 29.7, "Humidity": 67.12}' with RSSI -97
Packet #8 Received packet '{"nodeName": "Node1", "PacketID": 81, "Temperature": 29.71, "Humidity": 67.08}' with RSSI -97
Packet #9 Received packet '{"nodeName": "Node1", "PacketID": 82, "Temperature": 29.72, "Humidity": 67.07}' with RSSI -97
Packet #10 Received packet '{"nodeName": "Node1", "PacketID": 83, "Temperature": 29.73, "Humidity": 67.03}' with RSSI -96
Packet #11 Received packet '{"nodeName": "Node1", "PacketID": 84, "Temperature": 29.74, "Humidity": 67.04}' with RSSI -97
Packet #12 Received packet '{"nodeName": "Node1", "PacketID": 85, "Temperature": 29.74, "Humidity": 67.01}' with RSSI -98
Packet #13 Received packet '{"nodeName": "Node1", "PacketID": 86, "Temperature": 29.76, "Humidity": 67.01}' with RSSI -97
Packet #14 Received packet '{"nodeName": "Node1", "PacketID": 87, "Temperature": 29.76, "Humidity": 66.99}' with RSSI -97
Packet #15 Received packet '{"nodeName": "Node1", "PacketID": 87, "Temperature": 29.76, "Humidity": 66.99}' with RSSI -97
Packet #16 Received packet '{"nodeName": "Node1", "PacketID": 87, "Temperature": 29.76, "Humidity": 66.99}' with RSSI -97
Packet #17 Received packet '{"nodeName": "Node1", "PacketID": 99, "Temperature": 29.87, "Humidity": 66.72}' with RSSI -97
Packet #18 Received packet '{"nodeName": "Node1", "PacketID": 100, "Temperature": 29.88, "Humidity": 66.68}' with RSSI -97
Packet #19 Received packet '{"nodeName": "Node1", "PacketID": 103, "Temperature": 29.91, "Humidity": 66.64}' with RSSI -97
Packet #20 Received packet '{"nodeName": "Node1", "PacketID": 112, "Temperature": 30.0, "Humidity": 66.21}' with RSSI -97

```

รูปที่ 4.14 Log ของแต่ละ Packet ที่ได้รับเมื่อมีค่า Spreading Factor = 7 (ไม่มี CRC) จากการ

- 2) ทดสอบหา Log Packet ที่ค่า Spreading Factor = 9 (ไม่มี CRC)

```

SyncWord: F1
Frequency: 923000000
SpreadingFactor: 9
SignalBandwidth: 125000
LoRa Initializing OK!

Packet Test from 20 values
Packet #1 Received packet '{"nodeName": "Node1", "PacketID": 86, "Temperature": 30.97, "Humidity": 64.13}' with RSSI -97
Packet #2 Received packet '{"nodeName": "Node1", "PacketID": 84, "Temperature": 30.98, "Humidity": 64.1}' with RSSI -98
Packet #3 Received packet '{"nodeName": "Node1", "PacketID": 90, "Temperature": 31.0, "Humidity": 64.07}' with RSSI -98
Packet #4 Received packet '{"nodeName": "Node1", "PacketID": 91, "Temperature": 31.01, "Humidity": 64.05}' with RSSI -97
Packet #5 Received packet '{"nodeName": "Node1", "PacketID": 92, "Temperature": 31.01, "Humidity": 64.03}' with RSSI -98
Packet #6 Received packet '{"nodeName": "Node1", "PacketID": 94, "Temperature": 31.02, "Humidity": 63.95}' with RSSI -97
Packet #7 Received packet '{"nodeName": "Node1", "PacketID": 96, "Temperature": 31.03, "Humidity": 63.76}' with RSSI -98
Packet #8 Received packet '{"nodeName": "Node1", "PacketID": 97, "Temperature": 31.03, "Humidity": 63.67}' with RSSI -97
Packet #9 Received packet '{"nodeName": "Node1", "PacketID": 98, "Temperature": 31.04, "Humidity": 63.58}' with RSSI -97
Packet #10 Received packet '{"nodeName": "Node1", "PacketID": 99, "Temperature": 31.03, "Humidity": 63.55}' with RSSI -97
Packet #11 Received packet '{"nodeName": "Node1", "PacketID": 100, "Temperature": 31.04, "Humidity": 63.53}' with RSSI -97
Packet #12 Received packet '{"nodeName": "Node1", "PacketID": 101, "Temperature": 31.05, "Humidity": 63.51}' with RSSI -97
Packet #13 Received packet '{"nodeName": "Node1", "PacketID": 102, "Temperature": 31.06, "Humidity": 63.49}' with RSSI -98
Packet #14 Received packet '{"nodeName": "Node1", "PacketID": 103, "Temperature": 31.07, "Humidity": 63.5}' with RSSI -98
Packet #15 Received packet '{"nodeName": "Node1", "PacketID": 104, "Temperature": 31.07, "Humidity": 63.51}' with RSSI -98
Packet #16 Received packet '{"nodeName": "Node1", "PacketID": 105, "Temperature": 31.07, "Humidity": 63.46}' with RSSI -97
Packet #17 Received packet '{"nodeName": "Node1", "PacketID": 107, "Temperature": 31.08, "Humidity": 63.41}' with RSSI -97
Packet #18 Received packet '{"nodeName": "Node1", "PacketID": 108, "Temperature": 31.09, "Humidity": 63.35}' with RSSI -98
Packet #19 Received packet '{"nodeName": "Node1", "PacketID": 109, "Temperature": 31.09, "Humidity": 63.29}' with RSSI -97
Packet #20 Received packet '{"nodeName": "Node1", "PacketID": 110, "Temperature": 31.09, "Humidity": 63.24}' with RSSI -96

```

รูปที่ 4.15 Log ของแต่ละ Packet ที่ได้รับเมื่อมีค่า Spreading Factor = 9 (ไม่มี CRC)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) จากการทดสอบหา Log Packet ที่ค่า Spreading Factor = 12 (ไม่มี CRC)

```

SyncWord: F1
Frequency: 923000000
SpreadingFactor: 12
SignalBandwidth: 125000
LoRa Initializing OK!
RSSI Test from 20 values

Packet #1 Received packet '{"nodeName": "Node1", "PacketID": 0, "Temperature": 30.13, "Humidity": 64.18}' with RSSI -97
Packet #2 Received packet '{"nodeName": "Node1", "PacketID": 1, "Temperature": 30.15, "Humidity": 64.24}' with RSSI -96
Packet #3 Received packet '{"nodeName": "Node1", "PacketID": 2, "Temperature": 30.16, "Humidity": 64.31}' with RSSI -95
Packet #4 Received packet '{"nodeName": "Node1", "PacketID": 3, "Temperature": 30.21, "Humidity": 64.36}' with RSSI -97
Packet #5 Received packet '{"nodeName": "Node1", "PacketID": 4, "Temperature": 30.23, "Humidity": 64.38}' with RSSI -98
Packet #6 Received packet '{"nodeName": "Node1", "PacketID": 5, "Temperature": 30.24, "Humidity": 64.39}' with RSSI -97
Packet #7 Received packet '{"nodeName": "Node1", "PacketID": 6, "Temperature": 30.26, "Humidity": 64.4}' with RSSI -96
Packet #8 Received packet '{"nodeName": "Node1", "PacketID": 7, "Temperature": 30.27, "Humidity": 64.45}' with RSSI -96
Packet #9 Received packet '{"nodeName": "Node1", "PacketID": 8, "Temperature": 30.29, "Humidity": 64.45}' with RSSI -97
Packet #10 Received packet '{"nodeName": "Node1", "PacketID": 9, "Temperature": 30.3, "Humidity": 64.45}' with RSSI -95
Packet #11 Received packet '{"nodeName": "Node1", "PacketID": 10, "Temperature": 30.31, "Humidity": 64.44}' with RSSI -97
Packet #12 Received packet '{"nodeName": "Node1", "PacketID": 11, "Temperature": 30.33, "Humidity": 64.45}' with RSSI -96
Packet #13 Received packet '{"nodeName": "Node1", "PacketID": 12, "Temperature": 30.34, "Humidity": 64.44}' with RSSI -97
Packet #14 Received packet '{"nodeName": "Node1", "PacketID": 13, "Temperature": 30.36, "Humidity": 64.44}' with RSSI -97
Packet #15 Received packet '{"nodeName": "Node1", "PacketID": 14, "Temperature": 30.37, "Humidity": 64.43}' with RSSI -97
Packet #16 Received packet '{"nodeName": "Node1", "PacketID": 15, "Temperature": 30.38, "Humidity": 64.44}' with RSSI -97
Packet #17 Received packet '{"nodeName": "Node1", "PacketID": 16, "Temperature": 30.39, "Humidity": 64.41}' with RSSI -97
Packet #18 Received packet '{"nodeName": "Node1", "PacketID": 17, "Temperature": 30.4, "Humidity": 64.42}' with RSSI -97
Packet #19 Received packet '{"nodeName": "Node1", "PacketID": 18, "Temperature": 30.42, "Humidity": 64.43}' with RSSI -97
Packet #20 Received packet '{"nodeName": "Node1", "PacketID": 19, "Temperature": 30.43, "Humidity": 64.44}' with RSSI -97

```

รูปที่ 4.16 Log ของแต่ละ Packet ที่ได้รับเมื่อมีค่า Spreading Factor = 12 (ไม่มี CRC)

ผลการทดสอบและการวิเคราะห์ข้อมูลสำหรับการส่งแพ็กเก็ตผ่าน LoRa โดยใช้ Spreading Factor (SF) 7, 9 และ 12 ทั้งสองกรณีที่มีการใช้ CRC และไม่ใช่ CRC ดังนี้ ตารางที่ 4.1 Spreading Factor (SF) 7, 9 และ 12 ทั้งสองกรณีที่มีการใช้ CRC และไม่ใช่ CRC

SF	% Packet loss		% Packet false	
	มี CRC	ไม่มี CRC	มี CRC	ไม่มี CRC
7	51.16%	57.14%	0%	60%
9	0%	16.00%	0%	30%
12	0%	0%	0%	0%

SF 7 และ SF 9 โดยไม่ใช่ CRC มีจำนวนแพ็กเก็ตที่เสียหายสูงมาก (60% และ 30% ตามลำดับ) แสดงว่าการส่งด้วย SF 7 และ SF 9 ในโหมดนี้ไม่เหมาะสมสำหรับการส่งข้อมูลโดยไม่มี การตรวจสอบ CRC เนื่องจากมีข้อมูลที่สูญหายมากมาย การใช้ CRC ช่วยลดจำนวนแพ็กเก็ตที่เสียหายลงมาก (0% หรือเพียง 6%) ในทั้ง SF 7 และ SF 9 โดยมีจำนวนแพ็กเก็ตที่สูญหายมีค่าเท่ากันหรือน้อยกว่า SF 7 โดยไม่ใช่ CRC ในกรณีของ SF 12 ทั้งใช้และไม่ใช่ CRC มีประสิทธิภาพที่ดีที่สุดโดยไม่มีแพ็กเก็ตที่เสียหายหรือสูญหาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากข้อมูลการทดสอบทั้งหมดจึงทำให้เลือกใช้ SF9 โดยใช้ CRC เพราะเป็นการตั้งค่าที่ดีที่สุดสำหรับการส่งข้อมูล LoRa ในกรณีนี้ เนื่องจากมีประสิทธิภาพสูงและไม่มีแพ็กเก็ตที่เสียหรือสูญหายเลย และนอกจากนี้ SF9 ยังใช้เวลาในการส่งน้อยกว่า SF12 ด้วย เพราะมี Spreading Factor ต่ำกว่า SF12 ซึ่งจะทำให้การส่งข้อมูลเสร็จสิ้นเร็วขึ้น ดังนั้น SF9 น่าจะเป็นตัวเลือกที่เหมาะสมสำหรับการส่งข้อมูลในกรณีนี้โดยพิจารณาทั้งความเสถียรและเวลาในการส่งข้อมูล

#### 4.1.3 การทดสอบความเร็วในการส่งข้อมูลที่แต่ละ SF และ Bandwidth

ตารางที่ 4.2 การทดสอบความเร็วในการส่งข้อมูลที่แต่ละ SF

SF	6	7	8
Bandwidth 125 KHz	233 ms	404 ms	702 ms
	1.09 Kbps	631 Kbps	362 Kbps
Bandwidth 250 KHz	118 ms	202 ms	353 ms
	216 Kbps	1.262 Kbps	722 Kbps

#### 4.1.4 การทดสอบวัดระยะทางที่ไกลที่สุดที่รับและส่งข้อมูลได้



รูปที่ 4.17 การทดสอบวัดระยะทางที่ไกลที่สุดที่รับและส่งข้อมูลได้

ระยะในการส่งของ LoRa ของ End Device กับ Gateway ที่ทดสอบได้พบว่ามีระยะทางที่ทำได้สูงสุดอยู่ที่ระยะ 400 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.5 การทดสอบความแม่นยำของเซ็นเซอร์

จากการทดสอบเพื่อการวัดค่าความแม่นยำของอุณหภูมิและความชื้นจากเซ็นเซอร์บน End device แต่ละตัว แล้วนำผลลัพธ์ที่ได้จากการทดสอบมาเปรียบเทียบกับค่าอุณหภูมิและความชื้นจริงที่ได้จากเครื่องวัดจริงๆ ณ เวลานั้นเพื่อหาค่าความคลาดเคลื่อน ซึ่งผลลัพธ์ที่ได้จะถูกแสดงตามตารางที่ 4.3

ตารางที่ 4.3 ตารางแสดงผลการทดสอบความแม่นยำของเซ็นเซอร์ในการวัดอุณหภูมิและความชื้นของ End Device

ชื่อของ End device	ค่าเฉลี่ย อุณหภูมิ จาก อุปกรณ์ (°C)	ค่าเฉลี่ย ความชื้น จาก อุปกรณ์ (%)	อุณหภูมิ จริง ณ เวลานั้น (°C)	ความชื้น จริง ณ เวลา นั้น (%)	ค่าความคลาดเคลื่อน ของอุณหภูมิ (%)	ค่าความคลาดเคลื่อน ของความชื้น (%)
Node1	33.66	67.12	32.90	71.00	-2.31	5.46
Node2	34.05	64.98	34.80	70.00	2.16	7.17
Node3	33.50	64.39	32.90	68.00	-1.82	5.31
Node4	35.42	72.93	34.00	70.00	-4.18	-4.19

## 4.2 การทดสอบอุปกรณ์ทางฝั่ง Gateway

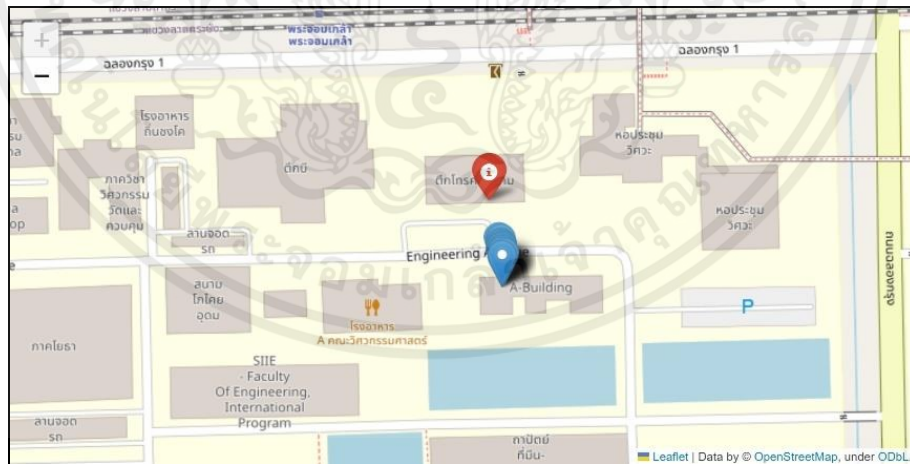
### 4.2.1 การทดสอบรับค่าสัญญาณเพื่อดูความแม่นยำของ GPS ที่เชื่อมต่อกับ Gateway

#### 4.2.1.1 การทดสอบรับค่าสัญญาณ GPS เมื่ออยู่กับที่

1) ตำแหน่งที่ตีภาคโทรคมนาคม คณะวิศวกรรมศาสตร์

Latitude 0:	13.727083	Longitude 0:	100.776276
Latitude 1:	13.727074	Longitude 1:	100.776283
Latitude 2:	13.727056	Longitude 2:	100.776283
Latitude 3:	13.727054	Longitude 3:	100.776283
Latitude 4:	13.727040	Longitude 4:	100.776291
Latitude 5:	13.727030	Longitude 5:	100.776291
Latitude 6:	13.727027	Longitude 6:	100.776291
Latitude 7:	13.727023	Longitude 7:	100.776291
Latitude 8:	13.727003	Longitude 8:	100.776291
Latitude 9:	13.727020	Longitude 9:	100.776291
Latitude 10:	13.727027	Longitude 10:	100.776291
Latitude 11:	13.727047	Longitude 11:	100.776291
Latitude 12:	13.727042	Longitude 12:	100.776291
Latitude 13:	13.727042	Longitude 13:	100.776291
Latitude 14:	13.727042	Longitude 14:	100.776291
Latitude 15:	13.727051	Longitude 15:	100.776291
Latitude 16:	13.727042	Longitude 16:	100.776291
Latitude 17:	13.727033	Longitude 17:	100.776283
Latitude 18:	13.727015	Longitude 18:	100.776299
Latitude 19:	13.727004	Longitude 19:	100.776291
Average of Lat and Lon : 13.727036,100.776268			

รูปที่ 4.18 ตำแหน่งพิกัดที่ตีภาคโทรคมนาคม คณะวิศวกรรมศาสตร์



รูปที่ 4.19 ตำแหน่งที่ตีภาคโทรคมนาคม คณะวิศวกรรมศาสตร์

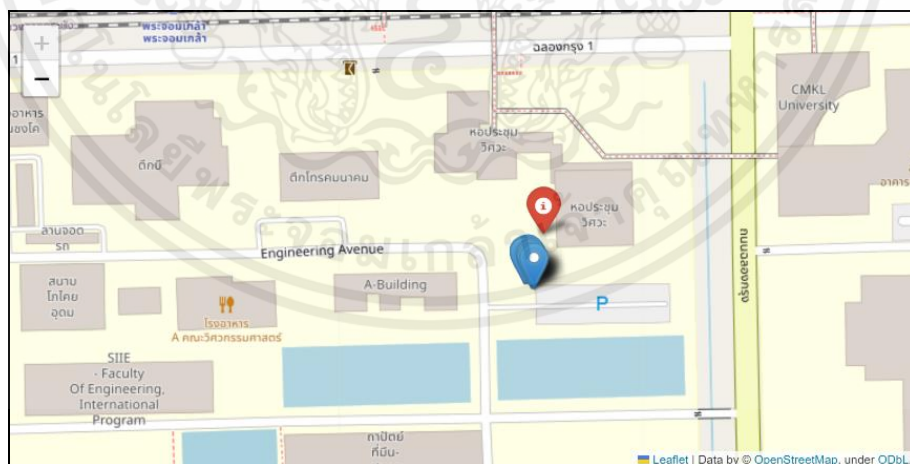
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าพิกัดที่ได้จาก GPS มีความแม่นยำ โดยค่าเฉลี่ยของค่าละติจูด (Latitude) คือ 13.727036 และค่าลองจิจูด (Longitude) คือ 100.776268 ห่างจากตำแหน่งจริงที่ค่าละติจูด (Latitude) คือ 13.727404 และค่าลองจิจูด (Longitude) คือ 100.776231 มีระยะทางที่วัดได้เท่ากับ 30.89 เมตร ซึ่งมีความแม่นยำใกล้เคียงกับตำแหน่งจริงของสถานที่ที่ GPS อ้างอิง

## 2) ตำแหน่งที่หอประชุม คณะวิศวกรรมศาสตร์

All values	
Latitude 0:	13.726975
Latitude 1:	13.726983
Latitude 2:	13.726982
Latitude 3:	13.726991
Latitude 4:	13.726997
Latitude 5:	13.727000
Latitude 6:	13.727006
Latitude 7:	13.727004
Latitude 8:	13.727007
Latitude 9:	13.727010
Latitude 10:	13.727014
Latitude 11:	13.727015
Latitude 12:	13.727015
Latitude 13:	13.727003
Latitude 14:	13.727003
Latitude 15:	13.727005
Latitude 16:	13.727007
Latitude 17:	13.727003
Latitude 18:	13.727004
Latitude 19:	13.727007
Longitude 0:	100.777092
Longitude 1:	100.777084
Longitude 2:	100.777084
Longitude 3:	100.777077
Longitude 4:	100.777077
Longitude 5:	100.777077
Longitude 6:	100.777077
Longitude 7:	100.777069
Longitude 8:	100.777069
Longitude 9:	100.777069
Longitude 10:	100.777069
Longitude 11:	100.777069
Longitude 12:	100.777069
Longitude 13:	100.777061
Longitude 14:	100.777054
Longitude 15:	100.777054
Longitude 16:	100.777054
Longitude 17:	100.777054
Longitude 18:	100.777054
Longitude 19:	100.777054
Average of Lat and lon : 13.727002,100.777092	

รูปที่ 4.20 พิกัดตำแหน่งที่หอประชุม คณะวิศวกรรมศาสตร์



รูปที่ 4.21 ตำแหน่งที่หอประชุม คณะวิศวกรรมศาสตร์

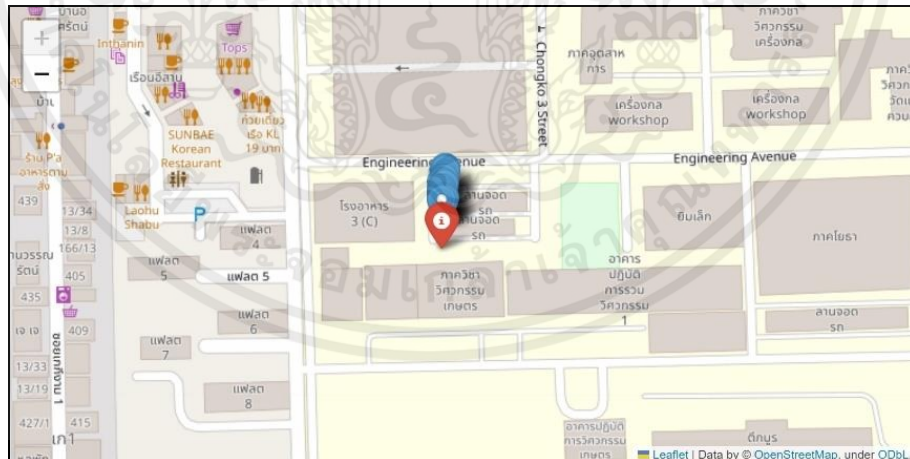
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าพิกัดที่ได้จาก GPS มีความแม่นยำ โดยค่าเฉลี่ยของค่าละติจูด (Latitude) คือ 13.727002 และค่าลองจิจูด (Longitude) คือ 100.77092 ห่างจากตำแหน่งจริงที่ค่าละติจูด (Latitude) คือ 13.727228 และค่าลองจิจูด (Longitude) คือ 100.777136 มีระยะทางที่วัดได้เท่ากับ 25.39 เมตร ซึ่งมีความแม่นยำใกล้เคียงกับตำแหน่งจริงของสถานที่ที่ GPS อ้างอิง

### 3) ตำแหน่งที่ตึก 12 ชั้น คณะวิศวกรรมศาสตร์

----- All values -----	
Latitude 0: 13.726901	Longitude 0: 100.772491
Latitude 1: 13.726903	Longitude 1: 100.772491
Latitude 2: 13.726898	Longitude 2: 100.772491
Latitude 3: 13.726899	Longitude 3: 100.772491
Latitude 4: 13.726895	Longitude 4: 100.772484
Latitude 5: 13.726894	Longitude 5: 100.772491
Latitude 6: 13.726859	Longitude 6: 100.772484
Latitude 7: 13.726843	Longitude 7: 100.772476
Latitude 8: 13.726836	Longitude 8: 100.772484
Latitude 9: 13.726832	Longitude 9: 100.772484
Latitude 10: 13.726812	Longitude 10: 100.772476
Latitude 11: 13.726802	Longitude 11: 100.772476
Latitude 12: 13.726788	Longitude 12: 100.772476
Latitude 13: 13.726782	Longitude 13: 100.772476
Latitude 14: 13.726773	Longitude 14: 100.772484
Latitude 15: 13.726770	Longitude 15: 100.772476
Latitude 16: 13.726766	Longitude 16: 100.772476
Latitude 17: 13.726767	Longitude 17: 100.772476
Latitude 18: 13.726768	Longitude 18: 100.772484
Latitude 19: 13.726769	Longitude 19: 100.772484
-----	
Average of Lat and Lon : 13.726828,100.772476	

รูปที่ 4.22 พิกัดตำแหน่งที่ตึก 12 ชั้น คณะวิศวกรรมศาสตร์

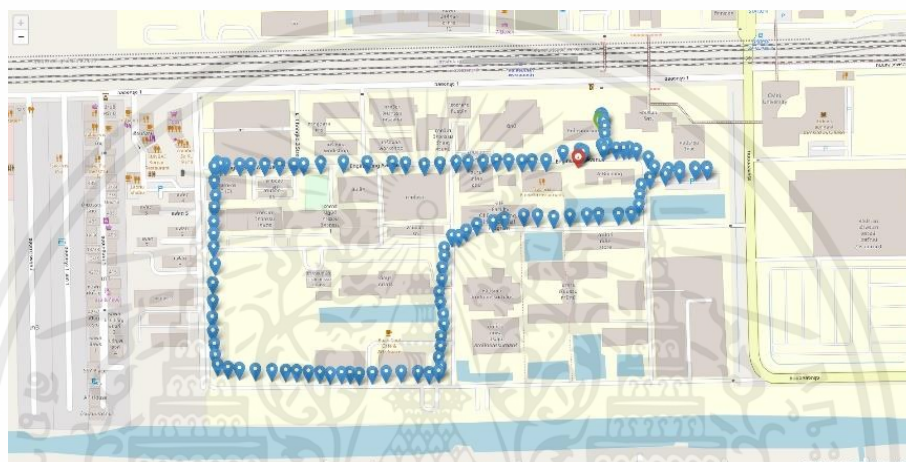


รูปที่ 4.23 ตำแหน่งที่ตึก 12 ชั้น คณะวิศวกรรมศาสตร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าพิกัดที่ได้จาก GPS มีความแม่นยำ โดยค่าเฉลี่ยของค่าละติจูด (Latitude) คือ 13.726828 และค่าลองจิจูด (Longitude) คือ 100.772476 ห่างจากตำแหน่งจริงที่ค่าละติจูด (Latitude) คือ 13.722295 และค่าลองจิจูด (Longitude) คือ 100.772479 มีระยะทางที่วัดได้เท่ากับ 18.85 เมตร ซึ่งมีความแม่นยำใกล้เคียงกับตำแหน่งจริงของสถานที่ที่ GPS อ้างอิง

#### 4.2.1.2 การทดสอบรับค่าสัญญาณ GPS เมื่อทำการเคลื่อนที่



รูปที่ 4.24 การทดสอบรับค่าสัญญาณ GPS เมื่อทำการเคลื่อนที่

ค่าสัญญาณ GPS ที่ได้รับค่ามาสามารถใช้งานได้จริง แม้ว่าอาจมีความผิดพลาดเล็กน้อยบางครั้ง โดยรวมแล้ว GPS สามารถใช้ในการระบุตำแหน่งและนำทางได้ ความผิดพลาดเล็กน้อยอาจส่งผลให้ความแม่นยำของตำแหน่งลดลงบ้าง แต่ระดับความคลาดเคลื่อนนี้พบน้อยมากและไม่ส่งผลกระทบต่อการทำงาน



### 4.2.3 การทดสอบการส่งข้อมูลทั้งหมดที่รวบรวมได้ขึ้น Database

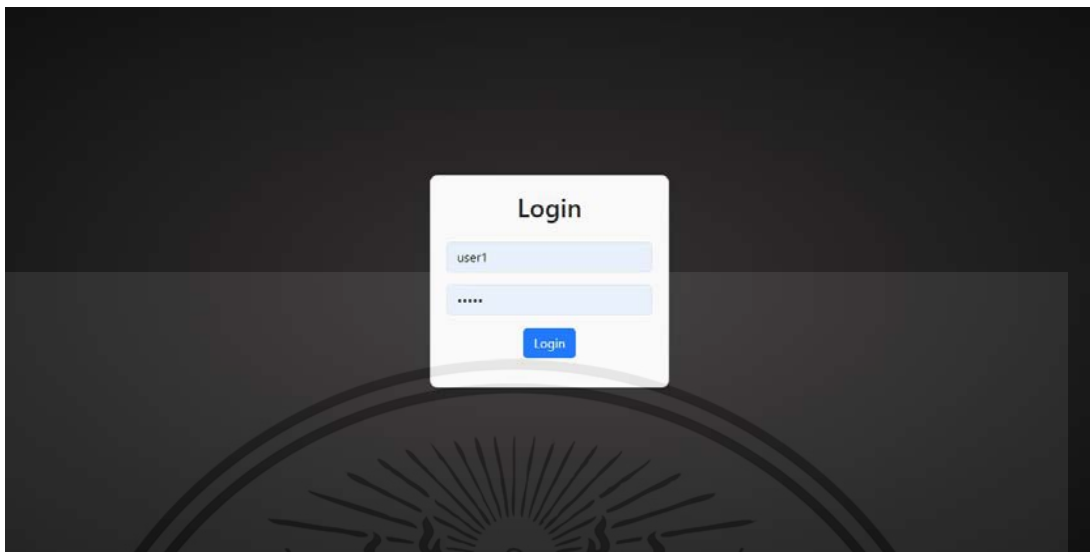
Time	Nodename	Temperature	Humidity	Speed	Latitude	Longitude
2023-11-12 16:56:28	Node1	31.05	65.65	0	13.622633	100.657745
2023-11-12 16:56:30	Node2	31.05	65.71	0	13.622633	100.657745
2023-11-12 16:59:37	Node1	31.06	65.26	0.59	13.622540	100.657860
2023-11-12 16:59:39	Node2	31.07	65.25	0.59	13.622540	100.657860
2023-11-12 17:02:45	Node1	31.11	65.56	0.65	13.622562	100.657745
2023-11-12 17:02:45	Node2	31.11	65.58	0.65	13.622562	100.657745
2023-11-12 17:12:09	Node1	31.21	65.65	1.93	13.622516	100.657761
2023-11-12 17:12:11	Node2	31.21	65.67	1.93	13.622516	100.657761
2023-11-12 17:15:19	Node1	31.24	65.78	1.19	13.622634	100.657707
2023-11-12 17:15:19	Node2	31.24	65.75	1.19	13.622634	100.657707
2023-11-12 17:18:26	Node1	31.22	65.61	1.06	13.622462	100.657829
2023-11-12 17:18:28	Node2	31.22	65.60	1.06	13.622462	100.657829
2023-11-12 17:21:34	Node1	31.21	65.83	0.85	13.622819	100.657768
2023-11-12 17:21:35	Node2	31.21	65.83	0.85	13.622819	100.657768
2023-11-12 17:24:43	Node1	31.19	66.12	0.91	13.622550	100.657829
2023-11-12 17:24:45	Node2	31.19	66.12	0.91	13.622550	100.657829
2023-11-12 17:27:55	Node1	31.18	66.29	0	13.622526	100.657761
2023-11-12 17:27:57	Node2	31.18	66.28	0	13.622526	100.657761

รูปที่ 4.26 หน้าจอการทดสอบการส่งข้อมูลทั้งหมดที่รวบรวมได้ขึ้น Database

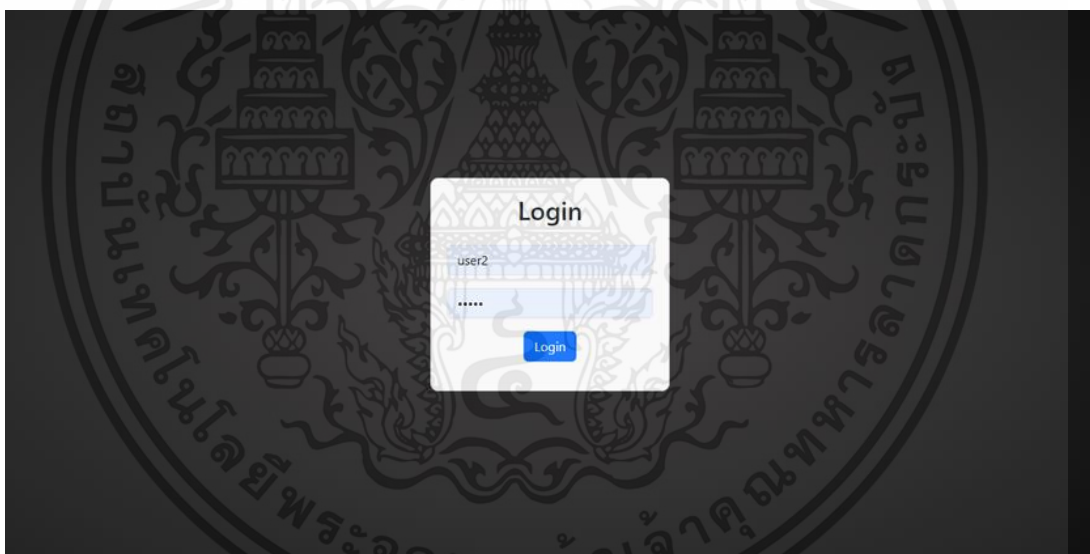
การทดสอบการส่งข้อมูลทั้งหมดที่รวบรวมได้ขึ้น Database การทดสอบระบบ LoRa ได้ทำการทดสอบการส่งข้อมูลจาก End Devices ไปยัง Gateway และจัดเก็บข้อมูลที่ได้รับในฐานข้อมูล (Database) การทดสอบนี้ช่วยยืนยันว่าข้อมูลสามารถส่งถึงและถูกบันทึกในฐานข้อมูลอย่างถูกต้องและสามารถระบุได้ว่าข้อมูลที่ถูกรับและส่งและบันทึกไว้ในฐานข้อมูลเป็นอย่างไร รวมถึงการจัดเก็บข้อมูลเพื่อการประมวลผลหรือการใช้งานอื่นๆได้

### 4.3 การทดสอบการใช้งานเว็บไซต์

ในส่วนนี้เราได้ทำการทดสอบการใช้งานของ Website เพื่อทดสอบความสามารถและประสิทธิภาพของฟังก์ชันต่างๆ ที่เราได้พัฒนาไว้สำหรับผู้ใช้งาน ซึ่งประกอบไปด้วยฟังก์ชันหลายอย่าง เช่น การแสดงผลข้อมูลผ่าน Dashboard, การตั้งค่าเพื่อปรับปรุงการทำงานของอุปกรณ์ที่เชื่อมต่อ (End device), และการตั้งค่าการแจ้งเตือนผ่าน Alert ที่สามารถปรับแก้ไขได้ตามต้องการของผู้ใช้งานทั่วไป ทั้งนี้ผู้ใช้สามารถเข้าถึงฟังก์ชันเหล่านี้ได้โดยตรง นอกเหนือจากนี้ยังมีส่วนที่เป็นของผู้ดูแลระบบ (admin) ที่มีสิทธิ์ในการเข้าถึงและจัดการผู้ใช้อื่นๆ ทั้งหมดในระบบ โดยผลที่ได้ทดสอบจะแสดงออกมาดังนี้

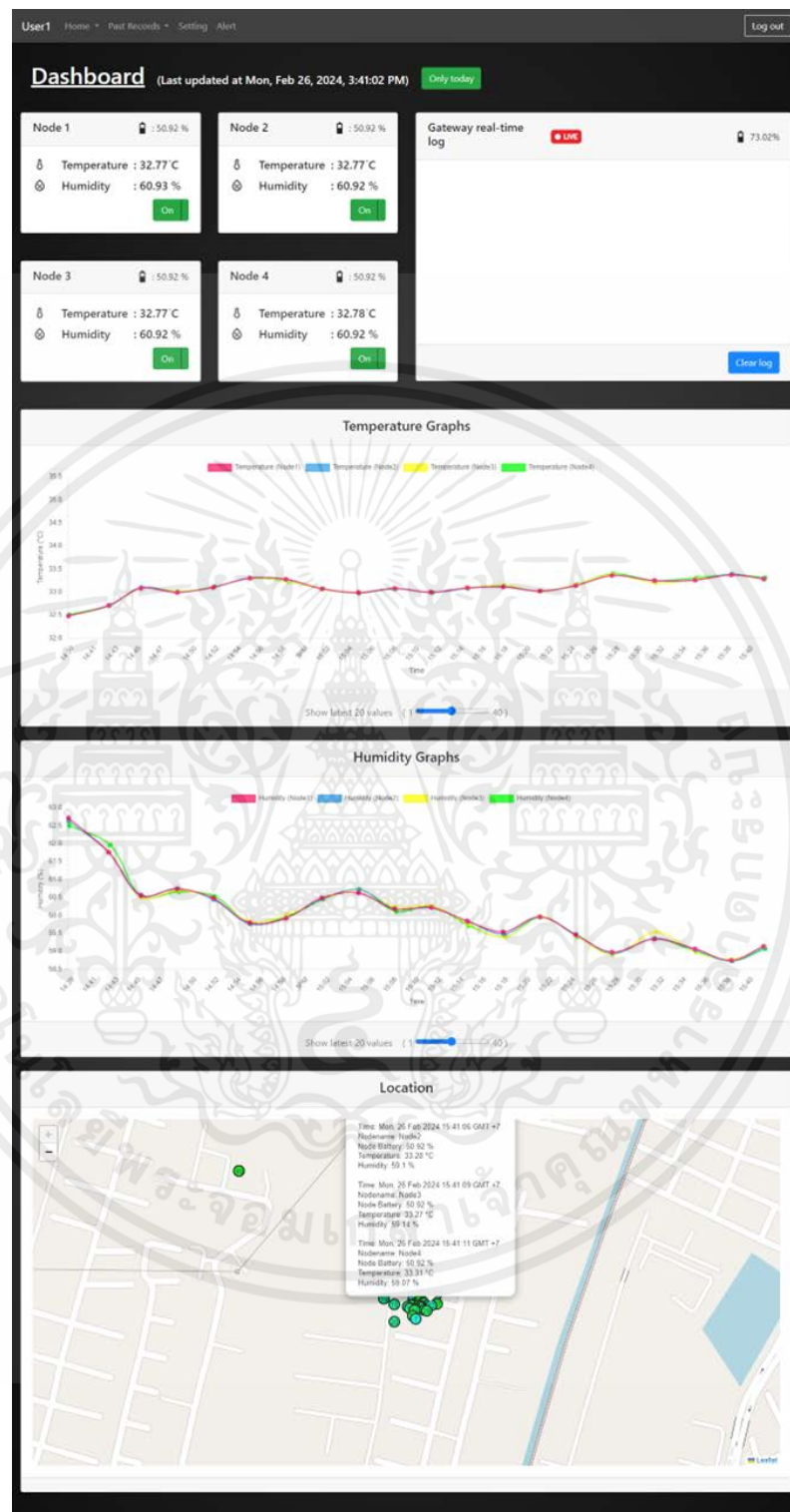


รูปที่ 4.27 รูปแสดงการ Login ด้วย Username “user1”



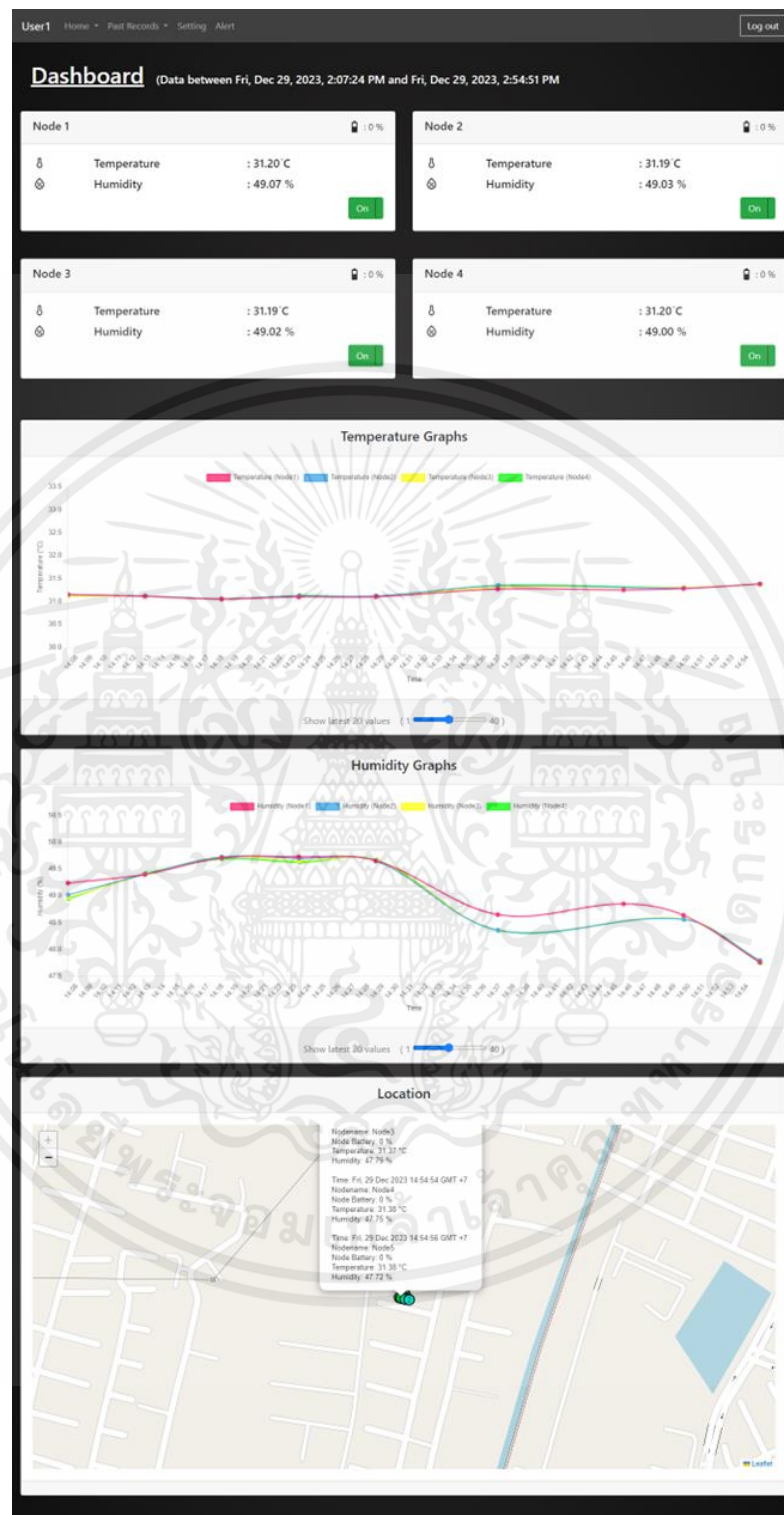
รูปที่ 4.28 รูปแสดงการ Login ด้วย Username “user2”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.29 รูปแสดงหน้าต่าง Dashboard ของ user1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.30 รูปแสดงแสดงหน้าต่าง Past records ของ user1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LoRa Settings

Syncword (0-255):  
242

Tx Power (0-20):  
20 dBm

Frequency (900-931):  
923 MHz

Transmission Interval:  
30 minutes

Submit Reset

rungrueng.site says  
Submitting Form with the following values:  
Syncword: 242 (F2)  
Tx Power: 20 dBm  
Frequency: 923 MHz  
Transmission Interval: 30 minutes

OK Cancel

รูปที่ 4.31 รูปแสดงหน้าต่างการเปลี่ยนแปลงการตั้งค่าของ End device สำหรับ user1

Alert condition

Temperature (Maximum temperature):  
40

Humidity (Maximum humidity):  
80

Speed (Maximum Speed to trigger alert):  
100

Latitude (Latitude of the destination):  
13.783653

Longitude (Longitude of the destination):  
100.634766

Distance (Distance from destination to trigger alert):  
2 km

Node Battery (Minimum node battery level to trigger alert):  
50 %

Gateway Battery (Minimum gateway battery level to trigger alert):  
50 %

Submit Reset

rungrueng.site says  
Submitting Form with the following values:  
Temperature: 40 °C  
Humidity: 80 %  
Speed: 100 Km/h  
Latitude: 13.783653  
Longitude: 100.634766  
Radius: 2 Km  
Node Battery: 50 %  
Gateway Battery: 50 %

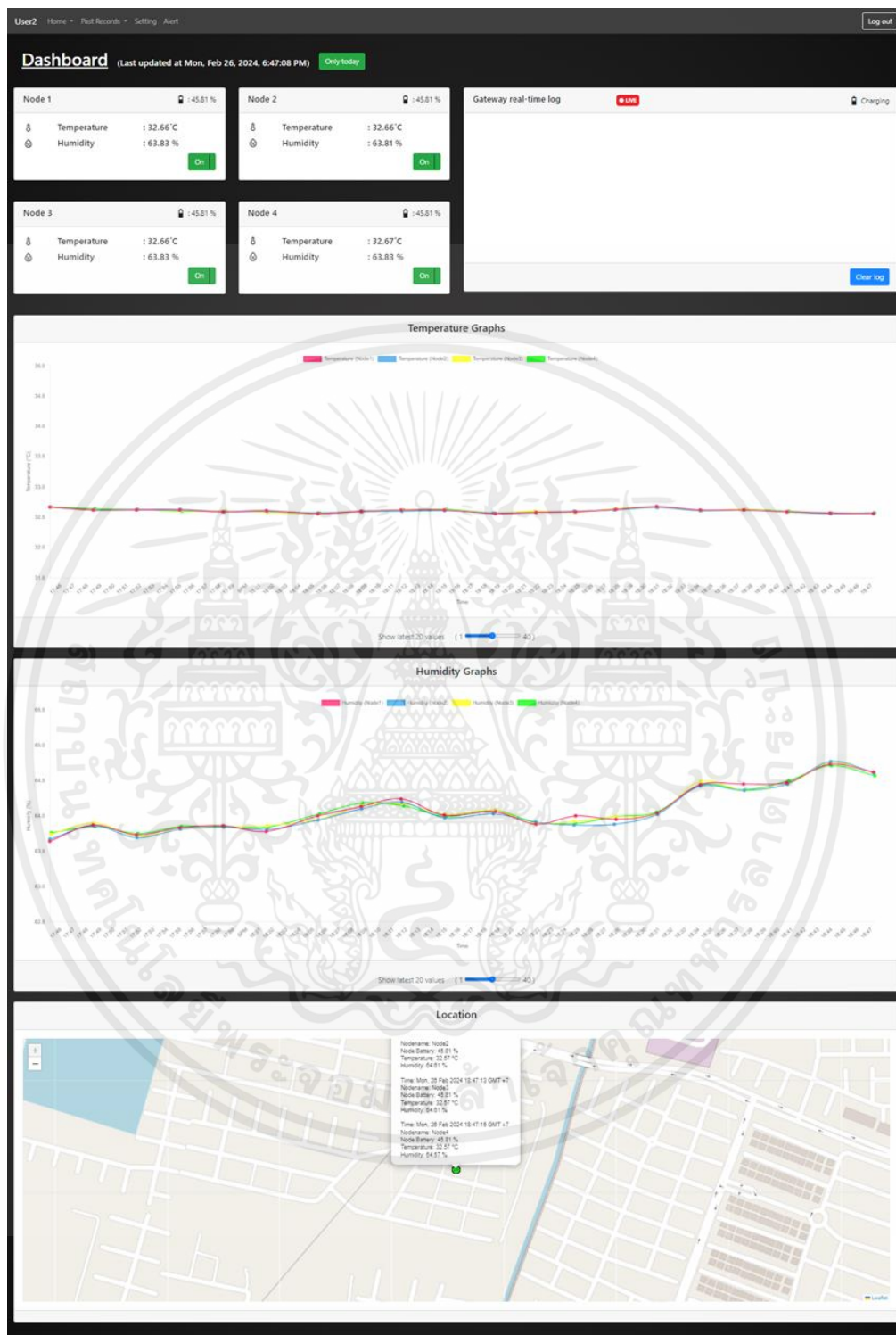
OK Cancel

List of recipient email

rungrueng@gmail.com  
63010415@vsnr.ac.th  
63010826@vsnr.ac.th

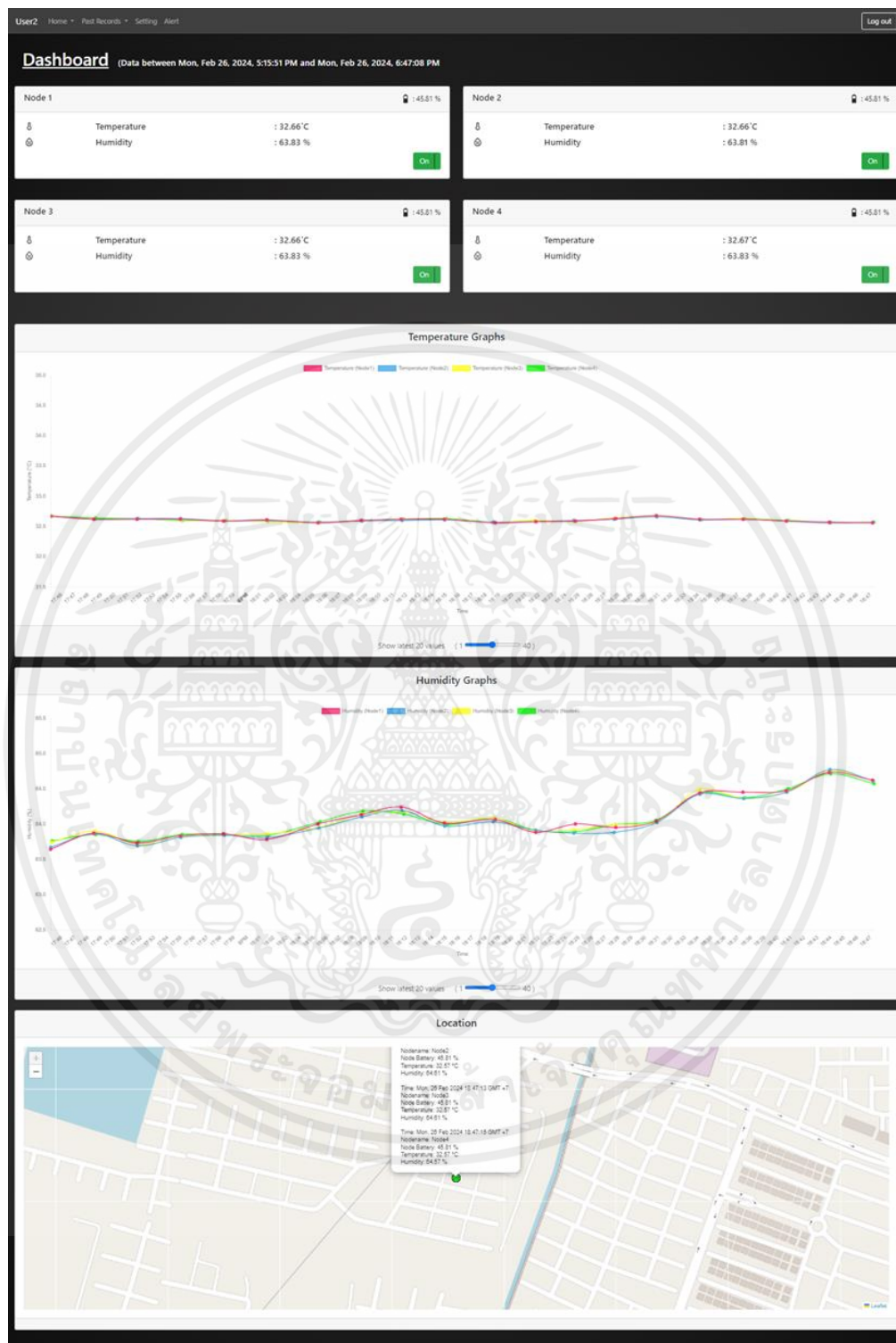
รูปที่ 4.32 รูปแสดงหน้าต่างการเปลี่ยนการตั้งค่าเงื่อนไขการแจ้งเตือนของ user1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



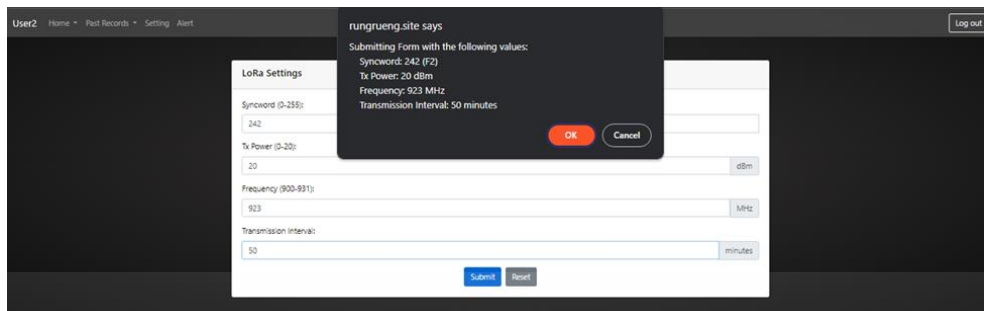
รูปที่ 4.33 รูปแสดงหน้าต่าง Dashboard ของ user2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

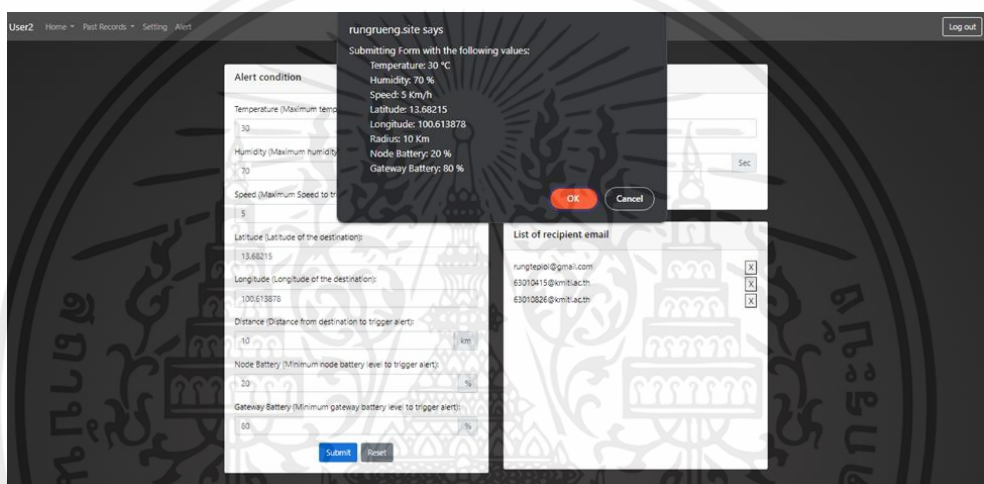


รูปที่ 4.34 รูปแสดงแสดงหน้าต่าง Past records ของ user2

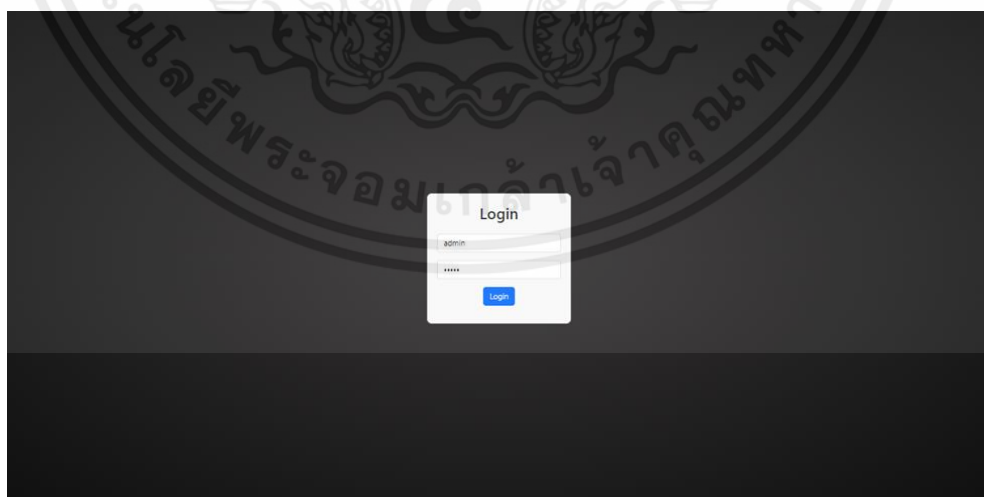
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.35 รูปแสดงหน้าต่างการเปลี่ยนแปลงการตั้งค่าของ End device สำหรับ user2

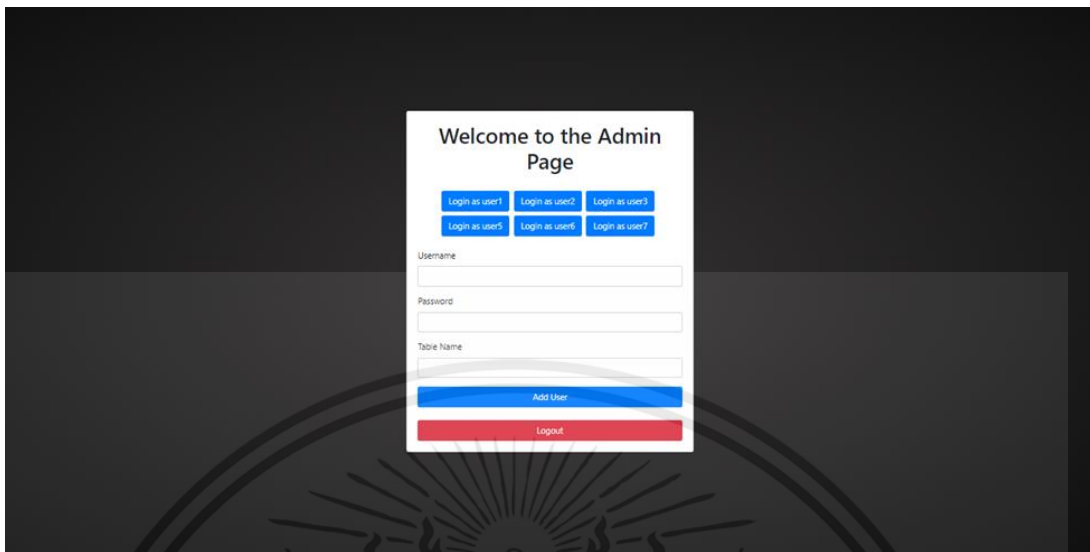


รูปที่ 4.36 รูปแสดงหน้าต่างการเปลี่ยนการตั้งค่าเงื่อนไขการแจ้งเตือนของ user2



รูปที่ 4.37 รูปแสดงการ Login ด้วย Username “admin”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.38 รูปแสดงหน้าตาการจัดการผู้ใช้ของผู้ดูแล

#### 4.4 การทดสอบในการใช้งานจริง

การทดลองนี้ทำการเก็บข้อมูลต่างๆ จากอุปกรณ์ End device ซึ่งได้แก่ อุณหภูมิ ความชื้น โดยข้อมูลเหล่านี้ถูกส่งไปยัง LoRa Gateway และจัดเก็บในฐานข้อมูล (Database) จากนั้นนำข้อมูลเหล่านี้มาแสดงผลเป็นกราฟและ mark ค่าตำแหน่งต่างๆลงไปในแผนที่ ซึ่งการทดสอบนี้แบ่งออกเป็น 2 สถานการณ์โดยผลการจากทดลองทั้งหมดไม่ว่าจะเป็น ข้อมูลที่ถูกจัดเก็บไว้บนฐานข้อมูล กราฟแสดงข้อมูลอุณหภูมิความชื้น และกราฟแสดงข้อมูลความเร็ว จะถูกแสดงไว้ในรูปด้านล่าง

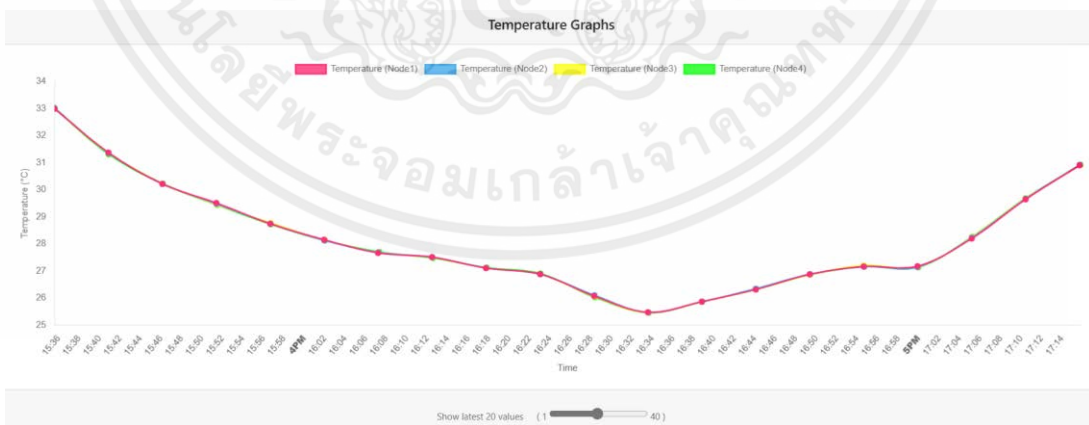
### 4.4.1 การทดสอบในใช้งานจริงในการเดินทางโดยรถสาธารณะ ในเส้นทาง ทางที่ได้กำหนดไว้

#### 4.4.1.1 ข้อมูลใน Database

id	Time	NodeName	Temperature	Humidity	Speed	Latitude	Longitude	Node Battery	Gateway Battery
419	2024-02-29 17:39:11	Node1	31.09	63.02	0	13.727643	100.748497	79.12	90.85
420	2024-02-29 17:39:23	Node3	31.91	62.85	0	13.727643	100.748497	79.12	90.85
421	2024-02-29 17:39:23	Node4	31.91	62.68	0	13.727643	100.748497	79.12	90.85
422	2024-02-29 17:42:44	Node1	32.06	63.82	NULL	13.728956	100.746086	79.42	87.68
423	2024-02-29 17:42:50	Node2	32.04	59.98	NULL	13.728956	100.746086	79.42	87.68
424	2024-02-29 17:42:56	Node3	32.00	54.50	NULL	13.728956	100.746086	79.42	87.68
425	2024-02-29 17:43:01	Node4	31.94	50.99	NULL	13.728956	100.746086	79.42	87.68
426	2024-02-29 17:46:03	Node1	30.33	42.97	NULL	13.729743	100.696535	78.02	90.48
427	2024-02-29 17:46:07	Node2	30.30	43.13	NULL	13.729743	100.696535	78.02	90.48
428	2024-02-29 17:46:11	Node3	30.24	42.72	NULL	13.729743	100.696535	78.02	90.48
429	2024-02-29 17:46:15	Node4	30.23	42.80	NULL	13.729743	100.696535	78.02	90.48
430	2024-02-29 17:48:18	Node1	29.36	43.15	NULL	13.725564	100.704320	77.49	88.83
431	2024-02-29 17:48:22	Node2	29.29	43.54	NULL	13.725564	100.704320	77.49	88.83
432	2024-02-29 17:48:24	Node3	29.26	43.38	NULL	13.725564	100.704320	77.49	88.83
433	2024-02-29 17:48:28	Node4	29.29	42.90	NULL	13.725564	100.704320	77.49	88.83
434	2024-02-29 17:52:56	Node3	28.01	47.05	NULL	13.741923	100.635612	76.71	89.11
435	2024-02-29 17:53:00	Node4	27.98	47.44	NULL	13.741923	100.635612	76.71	89.11
436	2024-02-29 17:53:02	Node1	28.06	54.32	NULL	13.741923	100.635612	76.78	89.11
437	2024-02-29 17:53:04	Node2	28.06	50.63	NULL	13.741923	100.635612	76.78	89.11
438	2024-02-29 17:56:09	Node1	28.42	51.71	NULL	13.744170	100.600480	75.28	89.76
439	2024-02-29 17:56:13	Node2	28.42	51.15	NULL	13.744170	100.600480	75.28	89.76
440	2024-02-29 17:56:15	Node3	28.41	51.04	NULL	13.744170	100.600480	75.28	89.76
441	2024-02-29 17:56:16	Node4	28.41	50.92	NULL	13.744170	100.600480	75.28	89.76
442	2024-02-29 17:59:23	Node1	28.44	46.19	NULL	13.749857	100.562504	76.38	88.38
443	2024-02-29 17:59:24	Node2	28.47	46.58	NULL	13.749857	100.562504	76.38	88.38
444	2024-02-29 17:59:26	Node3	28.47	46.57	NULL	13.749857	100.562504	76.38	88.38
445	2024-02-29 17:59:29	Node4	28.46	46.43	NULL	13.749857	100.562504	76.38	88.38
446	2024-02-29 18:18:06	Node1	30.65	60.91	0	13.751005	100.562055	72.57	147.06
447	2024-02-29 18:18:08	Node2	30.63	61.48	0	13.751005	100.562055	72.57	147.06
448	2024-02-29 18:18:11	Node3	30.63	60.59	0	13.751005	100.562055	72.57	147.06
449	2024-02-29 18:18:13	Node4	30.62	60.33	0	13.751005	100.562055	72.57	147.06
450	2024-02-29 18:18:14	Node1	30.64	60.60	0	13.751005	100.562055	72.57	147.06

รูปที่ 4.39 ข้อมูลใน Database (1)

#### 4.4.1.2 ข้อมูลอุณหภูมิที่ได้รับ



รูปที่ 4.40 ข้อมูลอุณหภูมิที่ได้รับ (1)

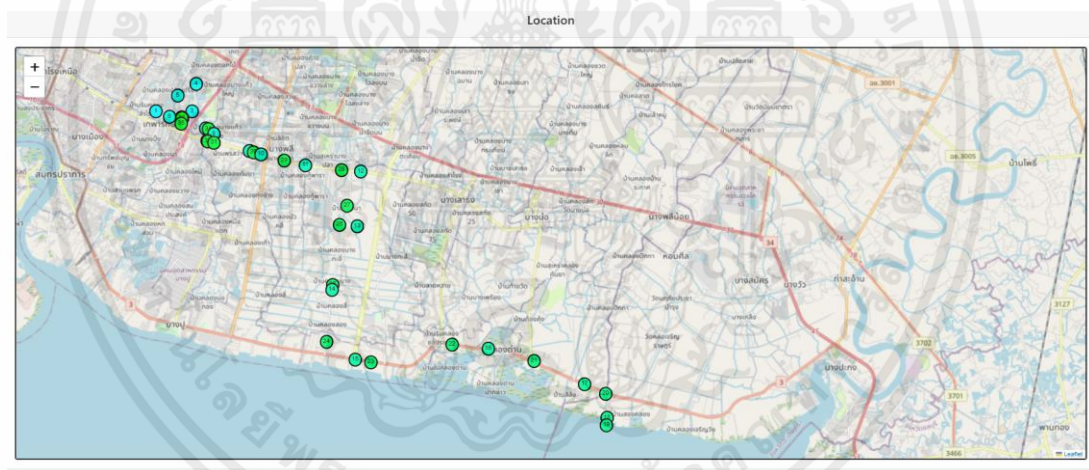
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.1.3 ข้อมูลอุณหภูมิที่ได้รับ



รูปที่ 4.41 ข้อมูลความชื้นที่ได้รับ (1)

#### 4.4.1.4 ข้อมูลพิกัดตำแหน่งที่ได้รับ



รูปที่ 4.42 ข้อมูลพิกัดตำแหน่งที่ได้รับ (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

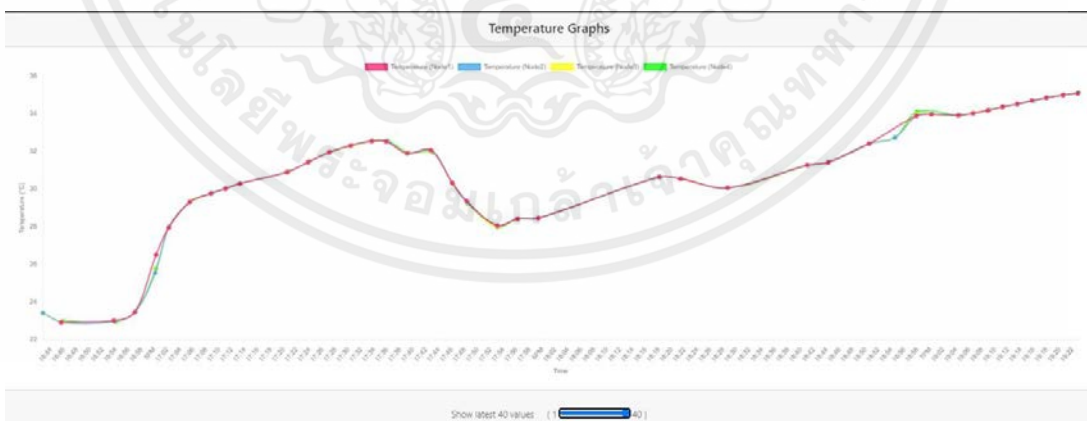
## 4.4.2 การทดสอบในใช้งานจริงในการเดินทางโดยรถไฟในเส้นทางทางที่ได้กำหนดไว้

### 4.4.2.1 ข้อมูลใน Database

Time	Nodename	Temperature	Humidity	Speed	Latitude	Longitude
2023-11-01 12:27:34	Node1	33.27	53.14	0.67	13.750085	100.564201
2023-11-01 12:27:34	Node2	33.28	53.10	0.67	13.750085	100.564201
2023-11-01 12:29:34	Node1	33.37	53.18	0.94	13.749553	100.562683
2023-11-01 12:29:34	Node2	33.37	53.18	0.94	13.749553	100.562683
2023-11-01 12:32:10	Node1	33.48	53.34	37.29	13.748912	100.568886
2023-11-01 12:32:14	Node2	33.48	53.29	37.28	13.748912	100.568886
2023-11-01 12:33:36	Node1	33.57	53.01	61.93	13.746344	100.579956
2023-11-01 12:33:36	Node2	33.58	52.99	61.93	13.746344	100.579956
2023-11-01 12:37:37	Node1	33.78	53.19	39.23	13.742845	100.599289
2023-11-01 12:37:39	Node2	33.79	53.20	39.23	13.742845	100.599289
2023-11-01 12:39:34	Node1	33.89	53.25	55.23	13.741152	100.614471
2023-11-01 12:39:36	Node2	33.89	53.24	55.23	13.741152	100.614471
2023-11-01 12:43:35	Node1	34.12	53.33	29.32	13.738444	100.638489
2023-11-01 12:43:37	Node2	34.12	53.31	29.32	13.738444	100.638489
2023-11-01 12:45:35	Node1	34.24	53.29	21.84	13.737984	100.641312
2023-11-01 12:45:37	Node2	34.25	53.27	21.84	13.737984	100.641312
2023-11-01 12:48:36	Node1	34.41	53.14	90.04	13.734649	100.672363
2023-11-01 12:48:38	Node2	34.42	53.13	90.04	13.734649	100.672363
2023-11-01 12:51:35	Node1	34.61	53.06	39.87	13.732450	100.691238
2023-11-01 12:51:37	Node2	34.61	53.05	39.87	13.732450	100.691238
2023-11-01 12:54:34	Node1	34.78	52.83	79.89	13.728952	100.723045

รูปที่ 4.43 ข้อมูลใน Database (2)

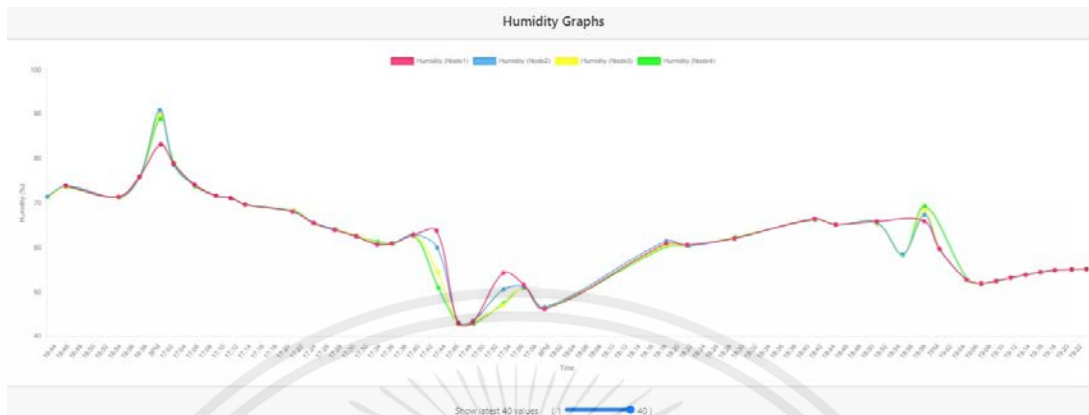
### 4.4.2.2 ข้อมูลอุณหภูมิที่ได้รับ



รูปที่ 4.44 ข้อมูลอุณหภูมิที่ได้รับ (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.2.3 ข้อมูลความชื้นที่ได้รับ



รูปที่ 4.45 ข้อมูลความชื้นที่ได้รับ (2)

#### 4.4.2.4 ข้อมูลพิกัดตำแหน่งที่ได้รับ



รูปที่ 4.46 ข้อมูลพิกัดตำแหน่งที่ได้รับ (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปผลและข้อเสนอแนะ

#### 5.1 สรุปผล

##### 5.1.1 สรุปการทดลองในส่วนของอุปกรณ์ทางฝั่ง END DEVICES

###### 5.1.1.1 สรุปการทดสอบดูการทำงานของระบบ LORA

จากผลการทดลอง พบว่าค่า SF (Spreading Factor) มีผลต่อเวลาที่ใช้ในการส่งสัญญาณอย่างมีนัยสำคัญ โดยเมื่อค่า SF เพิ่มขึ้น (สูงขึ้น) แล้วจะทำให้เวลาในการส่งสัญญาณเพิ่มขึ้นเช่นกัน ตัวอย่างเช่น เมื่อค่า SF เท่ากับ 7 ใช้เวลาในการส่งสัญญาณ 10 มิลลิวินาที แต่เมื่อค่า SF เท่ากับ 12 ใช้เวลาในการส่งสัญญาณ 180 มิลลิวินาที ดังนั้น การเลือกค่า SF ให้เหมาะสมจะช่วยให้ระบบ LoRa ทำงานได้อย่างมีประสิทธิภาพในสถานการณ์ที่ต้องการ เพราะฉะนั้นจึงสรุปได้ว่าค่า SF เป็นปัจจัยที่สำคัญในการกำหนดประสิทธิภาพของระบบ LoRa โดยการเลือกค่า SF ให้เหมาะสมจะช่วยให้ระบบ LoRa ทำงานได้อย่างมีประสิทธิภาพในสถานการณ์ที่ต้องการ

5.1.1.2 สรุปการทดสอบดูการส่งข้อมูลของแต่ละ Packet ที่ได้รับเมื่อมีสิ่งกีดขวาง

จากผลการทดลองการส่งแพ็กเก็ตผ่าน LoRa โดยใช้ Spreading Factor (SF) 7, 9 และ 12 ทั้งสองกรณีที่มีการใช้ CRC และไม่มี CRC พบว่า SF 7 และ SF 9 โดยไม่ใช้ CRC มีจำนวนแพ็กเก็ตที่เสียสูงมาก (60% และ 30% ตามลำดับ) แสดงว่าการส่งด้วย SF 7 และ SF 9 ในโหมดนี้ไม่เหมาะสมสำหรับการส่งข้อมูลโดยไม่มีตรวจสอบ CRC เนื่องจากมีข้อมูลที่สูญหายมาก การใช้ CRC จะช่วยลดจำนวนแพ็กเก็ตที่เสียลงมาก (0% หรือเพียง 6%) ในทั้ง SF 7 และ SF 9 โดยมีจำนวนแพ็กเก็ตที่สูญหายมีค่าเท่ากันหรือน้อยกว่า SF 7 โดยไม่ใช้ CRC ในกรณีของ SF 12 ทั้งใช้และไม่ใช้ CRC มีประสิทธิภาพดีที่สุดโดยไม่มีแพ็กเก็ตที่เสียหรือสูญหาย จากข้อมูลการทดสอบทั้งหมดจึงทำให้เลือกใช้ SF9 โดยใช้ CRC เพราะเป็นการตั้งค่าที่ดีที่สุดสำหรับการส่งข้อมูล LoRa ในกรณีนี้ เนื่องจากมีประสิทธิภาพสูงและไม่มีการแพ็กเก็ตที่เสียหรือสูญหายเลย และนอกจากนี้ SF9 ยังใช้เวลาในการส่งน้อยกว่า SF12 ด้วย เพราะมี Spreading Factor ต่ำกว่า SF12 ซึ่งจะทำให้การส่งข้อมูลเสร็จสิ้นเร็วขึ้น ดังนั้น SF9 จึงเป็นตัวเลือกที่เหมาะสมสำหรับการส่งข้อมูลในกรณีนี้โดยพิจารณาทั้งความเร็วและเวลาในการส่งข้อมูล

### 5.1.1.3 สรุปการทดสอบความเร็วในการส่งข้อมูลที่แต่ละ SF และ Bandwidth

จากผลการทดลองในครั้งนี้ เป็นการทดลองเพื่อเปรียบเทียบความเร็วในการส่งข้อมูลภายใต้เงื่อนไขที่แตกต่างกัน ประกอบด้วยตัวแปร 2 ประการ คือ ความกว้างของแบนด์วิดท์ (Bandwidth) และ Spreading Factor (SF) ผลการทดลองพบว่า ความเร็วในการส่งข้อมูลจะเพิ่มขึ้นเมื่อความกว้างของแบนด์วิดท์เพิ่มขึ้น แต่ความเร็วในการส่งข้อมูลจะลดลงเมื่อ Spreading Factor เพิ่มขึ้น โดยสรุปแล้ว ความเร็วในการส่งข้อมูลสูงสุดอยู่ที่ 2.16 Kbps ภายใต้เงื่อนไขความกว้างของแบนด์วิดท์ 250 kHz และ Spreading Factor 6 อย่างไรก็ตาม เงื่อนไขนี้ต้องใช้ Mode Explicit header ซึ่งเป็นโหมดที่ต้องบอกความยาวของ Payload ไปด้วย ซึ่งอาจไม่รองรับในบางอุปกรณ์ ดังนั้น จากการทดลองครั้งนี้ จึงเลือกใช้ Spreading Factor 8 ซึ่งเป็นค่าที่เหมาะสมที่สุดสำหรับการใช้งานทั่วไป เนื่องจากความเร็วในการส่งข้อมูลยังอยู่ในระดับที่ยอมรับได้ และไม่ต้องกังวลเรื่องความเข้ากันได้กับอุปกรณ์อื่น

### 5.1.1.4 สรุปการทดสอบวัดระยะทางที่ไกลที่สุดที่รับและส่งข้อมูลได้

จากผลการทดลองพบว่า ระยะการส่งข้อมูลผ่าน LoRa โดยใช้ Spreading Factor (SF) 8 และ ความกว้างของแบนด์วิดท์ 125 kHz อยู่ที่ประมาณ 400.09 เมตร ซึ่งถือว่าอยู่ในเกณฑ์ที่น่าพอใจ เนื่องจาก SF 8 เป็นค่าที่เหมาะสมสำหรับการส่งข้อมูลในระยะทางที่ใช้ในโครงการนี้ และยังมีความเร็วในการส่งข้อมูลที่เหมาะสมไม่น้อยหรือมากเกินไป

### 5.1.1.5 สรุปการทดสอบความแม่นยำของเซ็นเซอร์

จากข้อมูลในตารางที่ 4.3 พบว่า End device แต่ละตัวมีค่าความคลาดเคลื่อนของอุณหภูมิและความชื้นที่แตกต่างกันอย่างมีนัยสำคัญ โดย Node2 และ Node3 มีค่าความคลาดเคลื่อนที่ต่ำกว่า Node1 และ Node4 โดย Node4 มีค่าความคลาดเคลื่อนของทั้งอุณหภูมิและความชื้นเป็นลบ ซึ่งสามารถตีความได้ว่าการวัดที่ไม่แม่นยำ เพียงแต่มีทิศทางการควบคุมค่าผิดพลาดในทิศทางเดียวกันนั้น อาจแสดงให้เห็นถึงปัญหาของเซ็นเซอร์ที่ใช้บน Node4 หรือปัญหาในกระบวนการวัดเอง ดังนั้น การปรับปรุงหรือแก้ไขปัญหานั้นจำเป็นต้องพิจารณาข้อมูลเพิ่มเติมและการวิเคราะห์เพิ่มเติมเพื่อหาวิธีการแก้ไขที่เหมาะสมให้กับแต่ละ Node อย่างมีประสิทธิภาพและความแม่นยำในการใช้งานที่สูงขึ้น

## 5.1.2 สรุปการทดลองในส่วนของอุปกรณ์ทางฝั่ง GATEWAY

5.1.2.1 สรุปการทดลองรับค่าสัญญาณเพื่อดูความแม่นยำของ GPS ที่เชื่อมต่อกับ GATEWAY

จากผลการทดลองพบว่าสามารถระบุตำแหน่งได้อย่างแม่นยำ โดยเฉลี่ยมีความคลาดเคลื่อนไม่เกิน 30 เมตร ในทุกสถานที่ที่ทำการทดลอง ได้แก่ ดิเกภาคโทรคมนาคม หอประชุม และตึก 12 ชั้น คณะวิศวกรรมศาสตร์ ปัจจัยที่ส่งผลต่อความแม่นยำของ GPS ได้แก่ ความเข้มของสัญญาณ GPS และความเป็นเส้นตรงของเส้นทางที่ได้รับข้อมูล GPS อย่างไรก็ตาม ค่าเฉลี่ยที่ได้มีความแม่นยำใกล้เคียงและเป็นค่าที่น่าเชื่อถือสำหรับการใช้งานทั่วไป นอกจากนี้ ผลการทดลองยังพบว่าค่าสัญญาณ GPS ที่ได้รับค่ามาสามารถใช้งานได้จริง แม้ว่าอาจมีความผิดพลาดเล็กน้อยบางครั้ง โดยรวมแล้ว GPS สามารถใช้ในการระบุตำแหน่งและนำทางได้ ความผิดพลาดเล็กน้อยอาจส่งผลให้ความแม่นยำของตำแหน่งลดลงบ้าง แต่ระดับความคลาดเคลื่อนนี้พบน้อยมากและไม่ส่งผลกระทบต่อการทำงานนี้

### 5.1.2.2 สรุปการทดสอบการรับข้อมูลที่ส่งมาจากฝั่ง End Device

จากการทดสอบการรับข้อมูลที่ส่งมาจากฝั่ง End Device ในการทดสอบระบบ LoRa ได้ทำการทดสอบการรับข้อมูลที่ส่งมาจาก End Device โดยใช้ Gateway ผลการทดสอบพบว่าข้อมูลที่รับผ่าน Gateway มีความถูกต้องและความครบถ้วน ซึ่งบ่งชี้ว่าระบบ LoRa ทำงานอย่างถูกต้องและสามารถรับข้อมูลจาก End Device ได้

### 5.1.2.3 สรุปการทดสอบการส่งข้อมูลทั้งหมดที่รวบรวมได้ขึ้น Database

จากผลการทดลองพบว่าข้อมูลสามารถส่งถึงและถูกบันทึกไว้ในฐานข้อมูลอย่างถูกต้อง โดยข้อมูลที่ส่งและบันทึกไว้ในฐานข้อมูลประกอบด้วยข้อมูลต่างๆ เช่น ชื่ออุปกรณ์ อุณหภูมิ ความชื้น และเวลา เป็นต้น ข้อมูลเหล่านี้จะสามารถนำไปใช้ในการประมวลผลหรือการใช้งานอื่นๆ ได้

### 5.1.3 สรุปการทดลองในการทดสอบการใช้งานเว็บไซต์

จากผลการทดสอบการใช้งานเว็บไซต์เพื่อทดสอบความสามารถและประสิทธิภาพของฟังก์ชันต่างๆที่ได้พัฒนาไว้สำหรับผู้ใช้งาน สรุปผลได้ดังนี้คือ สามารถการเข้าสู่ระบบ และแสดงข้อมูลผ่าน Dashboard และ Past records ของผู้ใช้งานทั้งหมดได้ และสามารถทำการตั้งค่าพารามิเตอร์ต่างๆของ End device และ Gateway ได้ รวมไปถึงการตั้งค่าการแจ้งเตือนได้ตามต้องการ นอกจากนี้ผู้ดูแลระบบยังสามารถจัดการผู้ใช้งานทั้งหมดในระบบได้ ผลการทดสอบทั้งหมดเป็นที่เรียบร้อยและมีประสิทธิภาพที่ดี

### 5.1.4 สรุปการทดลองในการทดสอบการใช้งานจริง

ผลการทดลองพบว่าระบบสามารถเก็บค่าต่างๆได้อย่างถูกต้อง โดยข้อมูลทั้งหมดที่ได้มา ไม่ว่าจะเป็นอุณหภูมิ ความชื้น เวลา ความเร็ว และตำแหน่ง เมื่อนำมาแสดงผลเป็นกราฟและ mark ค่าตำแหน่งต่างๆลงไปในพื้นที่แล้ว แสดงให้เห็นว่ามีค่าความถูกต้องที่ค่อนข้างรับได้ หมายความว่าข้อมูลต่างๆเหล่านี้สามารถนำมาใช้ในการประมวลผลหรือการใช้งานอื่นๆ ได้อย่างมีประสิทธิภาพ

## 5.2 ข้อเสนอแนะ

1. ข้อมูล Speed หรือข้อมูลตำแหน่ง Latitude และ Longitude ที่ได้รับมาจาก GPS module อาจไม่มีความแม่นยำมากนัก ซึ่งอาจเป็นมาจากสภาพแวดล้อม ณ เวลานั้นๆ เช่น สภาพอากาศ ความหนาแน่นของอาคาร หรือสิ่งกีดขวางอื่นๆ นอกจากนี้ ผลการทดลองยังพบว่าการหาสัญญาณดาวเทียมโดย GPS module อาจมีเวลาค่อนข้างนาน ซึ่งอาจเป็นมาจากสภาพแวดล้อม ณ เวลานั้นๆ ควรทำการปรับปรุงความแม่นยำของข้อมูลตำแหน่ง เช่น การใช้เทคโนโลยีเสริมอื่นๆ
2. อุปกรณ์ทั้งในฝั่งของ Gateway และในฝั่งของ End Device อาจไม่ได้มีการจัดเรียงที่ดีหรือสวยงามมากนัก ซึ่งอาจส่งผลต่อประสิทธิภาพในการทำงานของระบบ โดยอาจทำให้อุปกรณ์เกิดความร้อนหรืออุปกรณ์การทำงานที่ผิดพลาดไม่เป็นไปตามสมควรได้ ควรทำการปรับปรุงการจัดเรียงอุปกรณ์ เช่น การใช้โครงสร้างที่เป็นระเบียบหรือการใช้วัสดุที่ทนทาน
3. ข้อมูลที่ได้รับมาจากอุปกรณ์ต่างๆนั้นอาจมีน้อยเกินไป ซึ่งอาจทำให้การนำไปวิเคราะห์หรือนำไปใช้นั้นอาจไม่ได้ประสิทธิภาพเท่าที่ควร ควรทำการปรับปรุงปริมาณข้อมูล เช่น การเพิ่มจำนวนอุปกรณ์หรือการเพิ่มระยะเวลาในการเก็บข้อมูล เป็นต้น

## บรรณานุกรม

- [1] ดร. ธีรพล รอดทอง. *พื้นฐานภาษา C สำหรับ Arduino (C Programming for Arduino)*.  
กรุงเทพฯ : สำนักพิมพ์เอสไอที, 2564.
- [2] สนธยา นงนุช. “การสื่อสารผ่าน I2C.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.glab.co.th/post/%E0%B>
- [3] Neonics Electronics. “รู๊จักเซนเซอร์อุณหภูมิ (Temperature Sensor).” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.neonics.co.th/thermometers/temperature-sensor.html>
- [4] Factomart. “Principle of Humidity Sensor.” [ออนไลน์].  
เข้าถึงได้จาก : <https://mall.factomart.com/principle-of-humidity-sensor/>
- [5] Moko Smart. “เทคโนโลยี LORA.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.mokosmart.com/th/lora-technology/>
- [6] Wikipedia contributors. “Global Positioning System.” [ออนไลน์].  
เข้าถึงได้จาก : [https://en.wikipedia.org/wiki/Global\\_Positioning\\_System](https://en.wikipedia.org/wiki/Global_Positioning_System)
- [7] Wikipedia contributors. “GNSS applications.” [ออนไลน์].  
เข้าถึงได้จาก : [https://en.wikipedia.org/wiki/GNSS\\_applications](https://en.wikipedia.org/wiki/GNSS_applications)
- [8] Wireless Logic. “What is CAT-1?.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.wirelesslogic.com/iot-glossary/what-is-cat-1/>
- [9] StackPython. “REST API ด้วย Flask 101.” [ออนไลน์].  
เข้าถึงได้จาก : <https://stackpython.co/tutorial/rest-api-flask-101>
- [10] Piravit Chenpittaya. “Flask (Web Framework).” [ออนไลน์].  
เข้าถึงได้จาก : <https://piravit-chenpittaya.medium.com/flask>.
- [11] Mindphp. “Databases.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.mindphp.com/>
- [12] คณะวิทยบริการ มหาวิทยาลัยศรีนครินทรวิโรฒ. “MariaDB: ทางเลือกของ MySQL ในอนาคต.” [ออนไลน์].  
เข้าถึงได้จาก : <https://km.cc.swu.ac.th/archives/4177>

- [13] วิกีพีเดีย สารานุกรมเสรี. “Visual Studio Code.” [ออนไลน์].  
เข้าถึงได้จาก : <https://th.wikipedia.org/wiki/>
- [14] ธิติ. “PlatformIO : แพลตฟอร์มพัฒนา Embedded สุดคุ้มค่า.” [ออนไลน์].  
เข้าถึงได้จาก : <https://thiti.dev/blog/28/>
- [15] Termius. “แพลตฟอร์ม SSH แบบข้ามแพลตฟอร์มสำหรับเดสก์ท็อปและอุปกรณ์มือถือ.” [ออนไลน์].  
เข้าถึงได้จาก : <https://termius.com/>
- [16] SDR Sharp. “เครื่องมือวิเคราะห์สัญญาณวิทยุแบบโอเพนซอร์ส.” [ออนไลน์].  
เข้าถึงได้จาก : <https://sdrsharp.com/>
- [17] Postman. “แพลตฟอร์ม API สำหรับนักพัฒนา.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.postman.com/>
- [18] Python. “ภาษาโปรแกรมระดับสูง.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.python.org/>
- [19] Digi. “รู้จักกับฐานข้อมูลและภาษา SQL.” [ออนไลน์].  
เข้าถึงได้จาก : <https://digi.data.go.th/รู้จักกับฐานข้อมูลและภาษา>
- [20] W3Schools. “HTML Tutorial.” [ออนไลน์].  
เข้าถึงได้จาก : [https://www.w3schools.com/html/html\\_intro.asp](https://www.w3schools.com/html/html_intro.asp)
- [21] W3Schools. “CSS Tutorial.” [ออนไลน์].  
เข้าถึงได้จาก : [https://www.w3schools.com/css/css\\_intro.asp](https://www.w3schools.com/css/css_intro.asp)
- [22] Wikipedia. “C++.” [ออนไลน์].  
เข้าถึงได้จาก : <https://en.wikipedia.org/wiki/C%2B%2B>
- [23] Wikipedia. “JSON.” [ออนไลน์].  
เข้าถึงได้จาก : <https://en.wikipedia.org/wiki/JSON>
- [24] Espressif. “ESP32.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.espressif.com/en/products/socs/esp32>
- [25] Opencircuit. “LILYGO TTGO T-SIM-A7670E R2.” [ออนไลน์].  
เข้าถึงได้จาก : <https://opencircuit.shop/product/lilygo-ttgo-t-sim-a7670e-r2-esp32-esp32-wrover>

- [26] Analog Read. “LoRa SX1276 Module 915 MHz.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.analogread.com/product/1476/lora-915mhz-module-sx1276>
- [27] MCU City. “915MHz Super Low Power RF LoRa Module SX1276 Chip Long-Distance Communication.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.mcucity.com/product/2498/915mhz-super-low-power-rf-lora-module-sx1276-chip-long-distance-communication>
- [28] Cybertice. “AHT10 High-Precision Digital Temperature and Humidity Sensor I2C.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.cybertice.com/product/2978>
- [29] Arduitrionics. “NEO-6M Ublox GPS Module NEO6MV2.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.arduitronics.com/product/1540/neo-6m-ublox-gps-module-neo6mv2->
- [30] Arduino4. “GY-NEO6MV2 Ublox GPS Module.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.arduino4.com/product/231/gy-neo6mv2-ublox-gps-module>
- [31] CyberTice. “GPS Module GY-NEO-6MV2 Ublox.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.cybertice.com/product/674/gps-module-gy-neo-6mv2-ublox>
- [32] CyberTice. “Raspberry Pi 3 Model B+.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.cybertice.com/product/851/raspberry-pi-3-model>
- [33] SparkFun Electronics. “RTK Surveyor.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.sparkfun.com/products/18443>
- [34] CyberTice. “RTL-SDR Dongle.” [ออนไลน์].  
เข้าถึงได้จาก : <https://www.cybertice.com/product/674/gps-module-gy-neo-6mv2-ublox>

- [35] Cookie Wow “Cookie Consent คืออะไร และจำเป็นต้องเว็บไซต์อย่างไร ” [ออนไลน์].  
เข้าถึงได้จาก : <https://cookiewow.com/th/blogs/what-is-cookies-consent-and-how-it-is-important-for-your-website>
- [36] Docker Hub “phpMyAdmin” [ออนไลน์].  
เข้าถึงได้จาก : <https://hub.docker.com/r/phpmyadmin/phpmyadmin>
- [37] Mozilla Developer Network “JavaScript” [ออนไลน์].  
เข้าถึงได้จาก : <https://developer.mozilla.org/en-US/docs/Web/JavaScript>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include <ArduinoJson.h>
#include <LoRa.h>
#include <Adafruit_AHTX0.h>
Adafruit_AHTX0 aht;
Adafruit_Sensor *aht_humidity, *aht_temp;
#define LED 26
#define ss 5
#define rst 4
#define dio0 34
#define VBAT_PIN 33
#define BATTV_MAX 4.2
#define BATTV_MIN 3.2
#define BATTV_LOW 3.4
float battpc;
String enddeviceslist[] = {"Node1"};
int waittime = 1000;
const int enddevices_num = sizeof(enddeviceslist) / sizeof(enddeviceslist[0]);
int node = 0;
RTC_DATA_ATTR int SyncWord;
RTC_DATA_ATTR int TxPower;
RTC_DATA_ATTR long freq;
RTC_DATA_ATTR double interval;
RTC_DATA_ATTR int spreadingFactor;
RTC_DATA_ATTR long signalBandwidth;
int defaultSyncWord = 0xF1;
int defaultTxPower = 20;
long defaultfreq = 923E6;
double defaultinterval = 0.1;
int defaultSpreadingFactor = 9;
long defaultSignalBandwidth = 125E3;
int indexs = 0;
#define user "user2"
#define timeout 50000
float readBatteryVoltage()
{
  float totalVoltage = 0.0;
  for (int i = 0; i < 10000; i++)
  {
    totalVoltage += ((float)analogRead(VBAT_PIN) / 4095) * 3.3 * 2 * 1.055;
  }
  float averageVoltage = totalVoltage / 10000.0;
  return averageVoltage;
}
String createJsonString(String Nodename, float tempfl, float humifl, float battpc)
{
  StaticJsonDocument<512> doc;
  int randomPacketID = random(99999, 1000000);
  doc["NodeName"] = Nodename;
  doc["User"] = user;
  doc["PacketID"] = randomPacketID;
  doc["Temperature"] = round(tempfl * 100.00) / 100.00;
  doc["Humidity"] = round(humifl * 100.00) / 100.00;
  doc["BatLvl"] = round(battpc * 100.00) / 100.00;
  String jsonString;
  serializeJson(doc, jsonString);
  return jsonString;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
void sleep(float sec)
{
    double min_d = sec / 60;
    esp_sleep_enable_timer_wakeup((interval - min_d) * 60 * 1000000);
    esp_deep_sleep_start();
}
void blinkLED(int numBlinks, int blinkDuration = 500)
{
    for (int i = 0; i < numBlinks; i++)
    {
        digitalWrite(LED, HIGH);
        delay(blinkDuration);
        digitalWrite(LED, LOW);
        delay(blinkDuration);
    }
}
void processJsonInput(const char *jsonInput)
{
    StaticJsonDocument<512> doc;
    DeserializationError error = deserializeJson(doc, jsonInput);
    if (error)
    {
        return;
    }
    if (doc["SyncWord"] != SyncWord)
    {
        SyncWord = doc["SyncWord"];
    }
    if (doc["TxPower"] != TxPower)
    {
        TxPower = doc["TxPower"];
    }
    if (doc["freq"] != freq)
    {
        freq = doc["freq"];
    }
    if (doc["interval"] != interval)
    {
        interval = doc["interval"];
    }
}
void blinkLEDInfinitely()
{
    while (true)
    {
        digitalWrite(LED, HIGH);
        delay(200);
        digitalWrite(LED, LOW);
        delay(200);
    }
}
void setup()
{
    Serial.begin(115200);
    pinMode(VBAT_PIN, INPUT);
    setCpuFrequencyMhz(80);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

pinMode(LED, OUTPUT);
float battVoltage = readBatteryVoltage();
if (battVoltage < BATTV_LOW)
{
}
battpc = ((battVoltage - BATTV_MIN) / (BATTV_MAX - BATTV_MIN)) * 100;
unsigned long startTime = millis();
if (!aht.begin())
{
  blinkLEDIninitely();
}
aht_temp = aht.getTemperatureSensor();
aht_humidity = aht.getHumiditySensor();
if (SyncWord == 0 || TxPower == 0 || freq == 0 || interval == 0)
{
  SyncWord = defaultSyncWord;
  TxPower = defaultTxPower;
  freq = defaultfreq;
  interval = defaultinterval;
  spreadingFactor = defaultSpreadingFactor;
  signalBandwidth = defaultSignalBandwidth;
}
LoRa.setPins(ss, rst, dio0);
while (!LoRa.begin(freq))
{
  blinkLEDIninitely();
  delay(500);
}
delay(waittime);
LoRa.setTxPower(TxPower);
LoRa.setSyncWord(SyncWord);
LoRa.setSpreadingFactor(spreadingFactor);
LoRa.setSignalBandwidth(signalBandwidth);
LoRa.enableCrc();
for (int i = 0; i < enddevices_num; i++)
{
  delay(1000);
  sensors_event_t humidity, temp;
  aht_humidity->getEvent(&humidity);
  aht_temp->getEvent(&temp);
  delay(100);
  String jsonOutput = createJsonString(enddeviceslist[i], temp.temperature,
  humidity.relative_humidity, battpc);
  String data = jsonOutput;
  for (int i = 0; i < 2; i++)
  {
    blinkLED(3, 300);
    LoRa.beginPacket();
    LoRa.print(data);
    LoRa.endPacket();
    delay(500);
  }
}
}
void loop()
{
  LoRa.setSyncWord(0XF2);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

unsigned long recvstartTime = millis();
bool packetReceived = false;
while (millis() - recvstartTime < timeout)
{
  int packetSize = LoRa.parsePacket();
  if (packetSize)
  {
    packetReceived = true;
    char LoRaData[255];
    int dataIndex = 0;
    while (LoRa.available())
    {
      char receivedChar = LoRa.read();
      LoRaData[dataIndex] = receivedChar;
      if (++dataIndex >= sizeof(LoRaData) - 1)
        break;
      if (dataIndex == 1 && receivedChar != '{')
      {
        dataIndex = 0;
        break;
      }
    }
    if (packetSize < 65)
    {
      if (LoRaData[0] == '{')
      {
        LoRaData[dataIndex] = '\0';
        processJsonInput(LoRaData);
        blinkLED(5, 100);
        break;
      }
    }
  }
  if (!packetReceived)
  {
    ESP.restart();
  }
  float durationSeconds = (millis() - recvstartTime) / 1000.0;
  sleep(durationSeconds);
  indexs++;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include <Arduino.h>
#include <HardwareSerial.h>
#include <ArduinoJson.h>
#include <TinyGsmClientSIM7600.h>
#include <LoRa.h>
#include <WiFi.h>
#include <TinyGPS++.h>
#define SerialAT Serial1
#define SerialMon Serial
TinyGsmSim7600 modem(SerialAT);
TinyGsmSim7600::GsmClientSim7600 client(modem);
TinyGPSPlus gps;
String URI = "http://rung.ddns.net:8050/";
int PORT = 8050;
String enddeviceslist[] = {"Node1", "Node2", "Node3"};
String nodenametemp[100];
int nodenametemp_num = sizeof(enddeviceslist) / sizeof(enddeviceslist[0]);
int enddevices_num = sizeof(enddeviceslist) / sizeof(enddeviceslist[0]);
int node = 0;
bool recvall = false;
#define SerialAT Serial1
#define SerialMon Serial
#define MIN_SPACE 500000
#define UART_BAUD 115200
#define PIN_DTR 25
#define PIN_TX 26
#define PIN_RX 27
#define PWR_PIN 4
#define PIN_RST 5
#define PIN_SUP 12
#define BAT_ADC 35
#define ss 15
#define rst 14
#define dio0 13
#define apiKey (String) "pk.71031a62fba9814c0898ae766b971df1"
bool DEBUG = true;
String response, LoRaData, latText, lonText, gpsinfo;
String serialres;
String latitude, longitude;
String extractedString = "";
String guser = "user2";
String nodeName[1000], user[300];
float degrees = 0.0;
float lat, lon;
float temp[200], humi[200], ebattlvl[200];
unsigned long setupstartTime, waitingtime, setupendtime, setupTime;
double speed = -1;
bool speedcheck;
float battPercentage;
float battVoltage;
RTC_DATA_ATTR int gSyncWord;
RTC_DATA_ATTR int gTxPower;
RTC_DATA_ATTR long gfreq;
RTC_DATA_ATTR double ginterval;
RTC_DATA_ATTR int gspreadingFactor;
RTC_DATA_ATTR long gsignalBandwidth;
int dgSyncWord = 0xF1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int dgTxPower = 20;
int dgspreadingFactor = 9;
long dgfreq = 923E6;
long dgsignalBandwidth = 125E3;
double dginterval = 3;
int eSyncWord;
int eTxPower;
int espreadingFactor;
long efreq;
long esignalBandwidth;
double einterval;
double interval;
int sendSyncWord = 5;
struct NETWORK_INFO
{
    String date;
    String time;
    String lat;
    String lon;
    char type[10];
    char mode[10];
    String mcc;
    String mnc;
    int lac = 0;
    String cid;
    char freq_b[15];
    double rsrq = 0;
    double rsrp = 0;
    double rssi = 0;
    int rssnr;
};
NETWORK_INFO networkinfo;
bool wifiConnected = false;
const char *ssid = "o";
const char *password = "00000000";
const char *server = "rung.ddns.net";
const int port = 8050;
String host = String(server);
WiFiClient wifiClient;
void modemPowerOn()
{
    const int SUPPLY_PIN = PIN_SUP;
    const int RESET_PIN = PIN_RST;
    const int POWER_PIN = PWR_PIN;
    pinMode(SUPPLY_PIN, OUTPUT);
    digitalWrite(SUPPLY_PIN, HIGH);
    pinMode(RESET_PIN, OUTPUT);
    digitalWrite(RESET_PIN, LOW);
    delay(100);
    digitalWrite(RESET_PIN, HIGH);
    delay(3000);
    digitalWrite(RESET_PIN, LOW);
    pinMode(POWER_PIN, OUTPUT);
    digitalWrite(POWER_PIN, LOW);
    delay(100);
    digitalWrite(POWER_PIN, HIGH);
    delay(1000);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    digitalWrite(POWER_PIN, LOW);
}
void connectToWiFi()
{
    SerialMon.print("Connecting to WiFi");
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED)
    {
        delay(500);
        SerialMon.print(".");
    }
    wifiConnected = true;
}
String sendAT(String command, int interval, boolean debug)
{
    String response = "";
    SerialAT.println(command);
    long int startTime = millis();
    while (((millis() - startTime)) < interval)
    {
        while (SerialAT.available() > 0)
        {
            int readData = SerialAT.read();
            response += char(readData);
        }
        SerialAT.flush();
        if (debug)
        {
            SerialMon.print(response);
        }
        return response;
    }
}
void changeGWconfig()
{
    if (eSyncWord != gSyncWord)
    {
        gSyncWord = eSyncWord;
    }
    if (eTxPower != gTxPower)
    {
        gTxPower = eTxPower;
    }
    if (efreq != gfreq)
    {
        gfreq = efreq * 1000000;
    }
    if (einterval != ginterval)
    {
        ginterval = einterval;
    }
}
void waitForGPSFix(float timeoutMinutes)
{
    unsigned long startTime = millis();
    unsigned long timeoutMillis = timeoutMinutes * 60000;
    while (millis() - startTime < timeoutMillis)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

networkinfo.cid = values[4];
delay(1000);
}
void sendLocationRequestwifi()
{
if (!WiFi.isConnected())
{
return;
}
String payload = "{\"token\":\"" + apiKey + "\",\"radio\":\"lte\",\"mcc\":\"" +
networkinfo.mcc + "\",\"mnc\":\"" + networkinfo.mnc + "\",\"cells\":{\"lac\":\"" +
networkinfo.lac + "\",\"cid\":\"" + networkinfo.cid + "\",\"psc\":0}},\"address\":1}";
String response;
WiFiClient wifiClient;
if (!wifiClient.connect("ap1.unwiredlabs.com", 80))
{
return;
}
String request = "POST /v2/process.php HTTP/1.1\r\n";
request += "Host: ap1.unwiredlabs.com\r\n";
request += "Content-Type: application/x-www-form-urlencoded\r\n";
request += "Content-Length: ";
request += String(payload.length());
request += "\r\n\r\n";
request += payload;
wifiClient.print(request);
const unsigned long TIMEOUT_DURATION = 10000;
unsigned long startTime = millis();
while (wifiClient.connected() && (millis() - startTime) < TIMEOUT_DURATION)
{
if (wifiClient.available())
{
String responseLine = wifiClient.readStringUntil('\n');
response += responseLine;
}
}
wifiClient.stop();
delay(1500);
int startIndex = response.indexOf("\"lat\":");
int endIndex = response.indexOf("\"lon\":");
latText = response.substring(startIndex + 6, endIndex);
startIndex = endIndex + 7;
endIndex = response.indexOf("\"accuracy\":");
lonText = response.substring(startIndex, endIndex);
}
void sendLocationRequest()
{
String payload = "{\"token\":\"" + apiKey + "\",\"radio\":\"lte\",\"mcc\":\"" +
networkinfo.mcc + "\",\"mnc\":\"" + networkinfo.mnc + "\",\"cells\":{\"lac\":\"" +
networkinfo.lac + "\",\"cid\":\"" + networkinfo.cid + "\",\"psc\":0}},\"address\":1}";
String response;
client.connect("ap1.unwiredlabs.com", 80);
String request = "POST /v2/process.php HTTP/1.1\r\n";
request += "Host: ap1.unwiredlabs.com\r\n";
request += "Content-Type: application/x-www-form-urlencoded\r\n";
request += "Content-Length: ";
request += String(payload.length());

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

request += "\r\n\r\n";
request += payload;
client.print(request);
while (client.connected())
{
    while (client.available())
    {
        char c = client.read();
        response += c;
        client.write(c);
    }
}
client.stop();
int startIndex = response.indexOf("\"lat\":");
int endIndex = response.indexOf(",\"lon\":");
latText = response.substring(startIndex + 6, endIndex);
startIndex = endIndex + 7;
endIndex = response.indexOf(",\"accuracy\":");
lonText = response.substring(startIndex, endIndex);
}
void sendHttpRequest()
{
    bool DEBUG = true;
    for (int i = 0; i < nodenametemp_num; i++)
    {
        unsigned long startTime = millis();
        unsigned long timeout = 10000;
        String payload = "{";
        payload += "\"nodename\": \"" + nodenametemp[i] + "\", ";
        payload += "\"user\": \"" + user[i] + "\", ";
        payload += "\"temperature\": " + String(temp[i]) + ", ";
        payload += "\"humidity\": " + String(humi[i]) + ", ";
        payload += "\"latitude\": " + String(latText) + ", ";
        payload += "\"longitude\": " + String(lonText) + ", ";
        payload += "\"ebatlvl\": " + String(ebatlvl[i]) + ", ";
        payload += "\"gbatlvl\": " + String(battPercentage, 2) + ", ";
        if (speedcheck == true)
        {
            payload += "\"speed\": " + String(speed);
        }
        else
        {
            payload += "\"speed\": null";
        }
        payload += "}";
        String response;
        client.connect("rung.ddns.net", PORT);
        String request = "POST /api/data HTTP/1.1\r\n";
        request += "Host: rung.ddns.net\r\n";
        request += "Content-Type: application/json\r\n";
        request += "Content-Length: ";
        request += String(payload.length());
        request += "\r\n\r\n";
        request += payload;
        client.print(request);
        while (client.connected() && (millis() - startTime) < timeout)
        {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

while (client.available())
{
    char c = client.read();
    response += c;
    client.write(c);
}
}
String successMessage = "Data inserted successfully";
if (response.indexOf(successMessage) != -1)
{
}
else
{
}
client.stop();
delay(1500);
}
}
void sendHttpRequestwifi()
{
    if (!wifiConnected)
    {
        connectToWiFi();
    }
    WiFiClient wifiClient;
    if (!wifiClient.connect(host.c_str(), 8050))
    {
        wifiClient.connect(host.c_str(), 8050);
        return;
    }
    for (int i = 0; i < nodenametemp_num; i++)
    {
        if (!wifiClient.connect(host.c_str(), 8050))
        {
            wifiClient.connect(host.c_str(), 8050);
            return;
        }
        String payload = "{";
        payload += "\"nodename\": \"" + nodenametemp[i] + "\", ";
        payload += "\"user\": \"" + user[i] + "\", ";
        payload += "\"temperature\": " + String(temp[i]) + ", ";
        payload += "\"humidity\": " + String(humi[i]) + ", ";
        payload += "\"latitude\": " + String(latText) + ", ";
        payload += "\"longitude\": " + String(lonText) + ", ";
        payload += "\"ebatlvl\": " + String(ebattlvl[i]) + ", ";
        payload += "\"gbatlvl\": " + String(battPercentage, 2) + ", ";
        if (speedcheck == true)
        {
            payload += "\"speed\": " + String(speed);
        }
        else
        {
            payload += "\"speed\": null";
        }
        payload += "}";
        String request = "POST /api/data HTTP/1.1\r\n";
        request += "Host: ";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

request += host;
request += "\r\n";
request += "Content-Type: application/json\r\n";
request += "Content-Length: ";
request += String(payload.length());
request += "\r\n\r\n";
request += payload;
wifiClient.print(request);
while (wifiClient.connected())
{
  if (wifiClient.available())
  {
    String response = wifiClient.readStringUntil('\n');
  }
}
wifiClient.stop();
delay(5000);
}
}
String fetchJsonConfig()
{
  String response;
  int maxRetries = 10;
  for (int retry = 1; retry <= maxRetries; retry++)
  {
    if (client.connect("rung.ddns.net", PORT))
    {
      String request = "GET /api/showconfig?user=" + guser + " HTTP/1.1\r\n";
      request += "Host: rung.ddns.net\r\n";
      request += "Connection: close\r\n";
      request += "\r\n";
      client.print(request);
      unsigned long startTime = millis();
      while (client.connected() && (millis() - startTime) < 10000)
      {
        while (client.available())
        {
          char c = client.read();
          response += c;
          Serial.write(c);
        }
      }
      client.stop();
      delay(1000);
      int start = response.indexOf('{');
      int end = response.lastIndexOf('}');
      if (start != -1 && end != -1 && end > start)
      {
        String jsonPart = response.substring(start, end + 1);
        return jsonPart;
      }
      else
      {
        response = "";
      }
    }
  }
  else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
    }
}
return "";
}
String fetchJsonConfigWIFI()
{
String response;
int maxRetries = 10;
for (int retry = 1; retry <= maxRetries; retry++)
{
if (wifiClient.connect(server, port))
{
String request = "GET /api/showconfig?user=" + guser + " HTTP/1.1\r\n";
request += "Host: " + String(server) + "\r\n";
request += "Connection: close\r\n";
request += "\r\n";
wifiClient.print(request);
unsigned long startTime = millis();
while (wifiClient.connected() && (millis() - startTime) < 10000)
{
while (wifiClient.available())
{
char c = wifiClient.read();
response += c;
Serial.write(c);
}
}
wifiClient.stop();
delay(1000);
int start = response.indexOf('{');
int end = response.lastIndexOf('}');
if (start != -1 && end != -1 && end > start)
{
String jsonPart = response.substring(start, end + 1);
return jsonPart;
}
else
{
response = "";
}
}
else
{
}
}
return "";
}
String fetchActiveNode()
{
String response;
int maxRetries = 10;
for (int retry = 1; retry <= maxRetries; retry++)
{
if (client.connect("rung.ddns.net", PORT))
{
String request = "GET /api/node?user=" + guser + " HTTP/1.1\r\n";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

request += "Host: rung.ddns.net\r\n";
request += "\r\n";
client.print(request);
unsigned long startTime = millis();
while (client.connected() && (millis() - startTime) < 10000)
{
  while (client.available())
  {
    char c = client.read();
    response += c;
    Serial.write(c);
  }
}
client.stop();
delay(2000);
int start = response.indexOf('{');
int end = response.lastIndexOf('}');
if (start != -1 && end != -1 && end > start)
{
  String jsonPart = response.substring(start, end + 1);
  return jsonPart;
}
else
{
  response = "";
}
}
else
{
}
}
return "";
}
String fetchActiveNodeWIFI()
{
  String response;
  int maxRetries = 10;
  for (int retry = 1; retry <= maxRetries; retry++)
  {
    if (wifiClient.connect(server, port))
    {
      String request = "GET /api/node?user=" + guser + " HTTP/1.1\r\n";
      request += "Host: " + String(server) + "\r\n";
      request += "\r\n";
      wifiClient.print(request);
      unsigned long startTime = millis();
      while (wifiClient.connected() && (millis() - startTime) < 10000)
      {
        while (wifiClient.available())
        {
          char c = wifiClient.read();
          response += c;
          Serial.write(c);
        }
      }
      wifiClient.stop();
      delay(2000);
    }
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int start = response.indexOf('{');
int end = response.lastIndexOf('}');
if (start != -1 && end != -1 && end > start)
{
    String jsonPart = response.substring(start, end + 1);
    return jsonPart;
}
else
{
    response = "";
}
}
else
{
}
}
return "";
}
void parseActiveNode(String jsonInput)
{
    const char *jsonString = jsonInput.c_str();
    StaticJsonDocument<500> doc;
    DeserializationError error = deserializeJson(doc, jsonString);
    if (error)
    {
        return;
    }
    JSONArray nodenames = doc["nodenames"];
    const size_t arraySize = nodenames.size();
    if (arraySize > 1)
    {
        String nodenamesArray[arraySize];
        for (size_t i = 0; i < arraySize; i++)
        {
            nodenamesArray[i] = nodenames[i].as<String>();
        }
        for (size_t i = 0; i < arraySize; i++)
        {
        }
        if (enddevices_num == arraySize)
        {
            for (size_t i = 0; i < arraySize; i++)
            {
                for (size_t j = 0; j < enddevices_num; j++)
                {
                    if (enddeviceslist[i] == nodenamesArray[j])
                    {
                        enddeviceslist[i] = nodenamesArray[j];
                    }
                }
            }
        }
        else
        {
            enddevices_num = arraySize;
            for (size_t j = 0; j < enddevices_num; j++)
            {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        enddeviceslist[j] = nodenamesArray[j];
    }
}
for (size_t i = 0; i < arraySize; i++)
{
}
for (size_t i = 0; i < enddevices_num; i++)
{
}
}
else
{
}
}
void parseJsonConfig(String jsonInput)
{
    StaticJsonDocument<512> doc;
    DeserializationError error = deserializeJson(doc, jsonInput);
    if (error)
    {
        eSyncWord = dgSyncWord;
        eTxPower = dgTxPower;
        efreq = dgfreq;
        interval = dginterval;
        return;
    }
    eSyncWord = doc["Syncword"];
    eTxPower = doc["Tx_power"];
    efreq = doc["Frequency"];
    if (efreq < 1000000)
    {
        efreq = efreq * 1000000;
    }
    einterval = doc["Tx_interval"];
    changeGWconfig();
}
String createJsonString(int SyncWord, int TxPower, long freq, double interval)
{
    StaticJsonDocument<512> doc;
    if (gSyncWord == 0 || gTxPower == 0 || gfreq == 0 || ginterval == 0)
    {
        gSyncWord = dgSyncWord;
        gTxPower = dgTxPower;
        gfreq = dgfreq;
        gspreadingFactor = dgspreadingFactor;
        gsignalBandwidth = dgsignalBandwidth;
        ginterval = dginterval;
    }
    if (SyncWord == 0)
    {
        SyncWord = dgSyncWord;
    }
    if (TxPower == 0)
    {
        TxPower = dgTxPower;
    }
    if (freq == 0)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    {
        freq = dgfreq;
    }
    if (interval == 0)
    {
        interval = dginterval;
    }
    doc["SyncWord"] = SyncWord;
    doc["TxPower"] = TxPower;
    if (freq < 1000)
    {
        doc["freq"] = freq * 1000000;
    }
    else
    {
        doc["freq"] = freq;
    }
    doc["interval"] = interval;
    String jsonString;
    serializeJson(doc, jsonString);
    return jsonString;
}
bool connect2LTE()
{
    boolean DEBUG = 1;
    delay(1000);
    sendAT("AT+NETCLOSE", 1000, DEBUG);
    delay(1000);
    sendAT("AT+CPIN?", 2000, DEBUG);
    delay(1000);
    sendAT("AT+CSOCKSETPN=1", 3000, DEBUG);
    String res = sendAT("AT+NETOPEN", 3000, DEBUG);
    if (res.indexOf("OK") == -1 || res.indexOf("not") != -1)
    {
        connectToWiFi();
    }
    else
    {
        String response = sendAT("AT+IPADDR", 5000, DEBUG);
        if (response.indexOf("ERROR") != -1)
        {
            if (!wifiConnected)
            {
                connectToWiFi();
            }
        }
    }
}
return true;
}
void sleep(float sec)
{
    double min_d = sec / 60;
    esp_sleep_enable_timer_wakeup((ginterval - min_d) * 60 * 0.5 * 1000000);
    esp_deep_sleep_start();
}
void readBattLevel()
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

const int numOfReadings = 10000;
const float batteryFullVoltage = 4.2;
const float batteryOffVoltage = 2.5;
int battReadingsSum = 0;
for (int i = 0; i < numOfReadings; i++)
{
  battReadingsSum += analogRead(BAT_ADC);
}
float avgReading = static_cast<float>(battReadingsSum) /
static_cast<float>(numOfReadings);
battVoltage = avgReading * (3.3 / 4096.0) * 2.0 * 1.079691;
battPercentage = 100.0 * (1.0 - ((batteryFullVoltage - battVoltage) / (batteryFullVoltage
batteryOffVoltage)));
if (battPercentage > 100.00)
{
  battPercentage = 100.00;
}
}
void setup()
{
  setupstartTime = millis();
  Serial.begin(UART_BAUD);
  Serial1.begin(UART_BAUD, SERIAL_8N1, PIN_RX, PIN_TX);
  Serial2.begin(9600, SERIAL_8N1, 33, 32);
  LoRa.setPins(ss, rst, dio0);
  if (gSyncWord == 0 || gTxPower == 0 || gfreq == 0 || ginterval == 0)
  {
    gSyncWord = dgSyncWord;
    gTxPower = dgTxPower;
    gfreq = dgfreq;
    gspreadingFactor = dgspreadingFactor;
    gsignalBandwidth = dgsignalBandwidth;
    ginterval = dginterval;
  }
  while (!LoRa.begin(gfreq))
  {
    delay(500);
  }
  LoRa.setTxPower(gTxPower);
  LoRa.setSyncWord(gSyncWord);
  LoRa.setSpreadingFactor(gspreadingFactor);
  LoRa.setSignalBandwidth(gsignalBandwidth);
  LoRa.enableCrc();
  modemPowerOn();
  delay(500);
  connect2LTE();
  if (wifiConnected)
  {
    parseJsonConfig(fetchJsonConfigWIFI());
    parseActiveNode(fetchActiveNodeWIFI());
  }
  else
  {
    parseJsonConfig(fetchJsonConfig());
    parseActiveNode(fetchActiveNode());
  }
  setupendtime = millis();
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

setuptime = setupendtime - setupstartTime;
for (int i = 0; i < enddevices_num; i++)
{
    nodenametemp[i] = enddeviceslist[i];
}
nodenametemp_num = enddevices_num;
recvall = false;
}
void loop()
{
    waitingtime = millis();
    int packetSize = LoRa.parsePacket();
    if (packetSize)
    {
        char LoRaData[255];
        int dataIndex = 0;
        while (LoRa.available())
        {
            char receivedChar = LoRa.read();
            LoRaData[dataIndex] = receivedChar;
            dataIndex++;
            if (dataIndex >= sizeof(LoRaData) - 1)
            {
                LoRaData[dataIndex] = '\0';
                break;
            }
            if (dataIndex == 1 && receivedChar != '{')
            {
                dataIndex = 0;
                break;
            }
        }
        if (dataIndex > 0)
        {
            LoRaData[dataIndex] = '\0';
            DynamicJsonDocument doc(256);
            deserializeJson(doc, LoRaData);
            String input_nodename = doc["NodeName"].as<String>();
            String input_user = doc["User"].as<String>();
            float input_temp = doc["Temperature"];
            float input_humi = doc["Humidity"];
            float input_batt = doc["BatLvl"];
            for (int i = 0; i < enddevices_num; i++)
            {
                if (enddeviceslist[i] == input_nodename)
                {
                    user[node] = input_user;
                    temp[node] = input_temp;
                    humi[node] = input_humi;
                    ebattlvl[node] = input_batt;
                    nodenametemp[node] = input_nodename;
                    for (int j = i; j < enddevices_num - 1; j++)
                    {
                        enddeviceslist[j] = enddeviceslist[j + 1];
                    }
                    enddevices_num--;
                    node = node + 1;
                }
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        break;
    }
}
}
if (enddevices_num == 0)
{
    recvall = true;
    delay(5000);
    String jsonOutput = createJsonString(eSyncWord, eTxPower, efreq, einterval);
    for (int i = 0; i < 4; i++)
    {
        LoRa.setTxPower(20);
        LoRa.setSyncWord(242);
        LoRa.beginPacket();
        LoRa.print(jsonOutput);
        LoRa.endPacket();
        delay(1000);
    }
}
if (waitingtime > (ginterval * 60 * 1000) + setuptime)
{
    recvall = true;
}
if (recvall)
{
    waitForGPSFix(0.017);
    float latValue = lat;
    float lonValue = lon;
    if ((latValue > -90 && latValue < 0) || (latValue > 0 && latValue <= 90) &&
        (lonValue > -180 && lonValue < 0) || (lonValue > 0 && lonValue <= 180))
    {
        GPSavg(5);
        speed = gps.speed.kmph();
        speedcheck = true;
        readBattLevel();
        if (wifiConnected)
        {
            sendHttpRequestwifi();
        }
        else
        {
            sendHttpRequest();
        }
    }
    else
    {
        speedcheck = false;
        readcellinfo();
        readBattLevel();
        if (wifiConnected)
        {
            sendLocationRequestwifi();
            sendHttpRequestwifi();
        }
        else
        {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        sendLocationRequest();
        sendHttpRequest();
    }
}
unsigned long endTime = millis();
unsigned long duration = endTime - waitingtime + setupTime;
float durationSeconds = duration / 1000.0;
changeGWconfig();
sleep(durationSeconds);
}
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

from flask import Flask, request, jsonify
from haversine import haversine
from datetime import datetime, date
import mysql.connector, pytz, requests, json, time, re
from flask_cors import CORS
app = Flask(__name__)
CORS(app)
values_list = []
db_config = {
    "user": "root",
    "password": "project112233",
    "host": "192.168.1.30",
    "database": "Project",
}
login_config = {
    "user": "root",
    "password": "project112233",
    "host": "192.168.1.30",
    "database": "User",
}
def get_db_config(user):
    return {
        "user": "root",
        "password": "project112233",
        "host": "192.168.1.30",
        "database": user,
    }
def parse_api_time(api_time_str):
    return datetime.strptime(api_time_str, "%a, %d %b %Y %H:%M:%S GMT")
def increment_node(previous_node):
    match = re.match(r"([a-zA-Z]+)(\d+)", previous_node)
    if match:
        non_numeric_part, numeric_part = match.groups()
        incremented_numeric_part = str(int(numeric_part) + 1)
        new_node = non_numeric_part + incremented_numeric_part
        return new_node
    return previous_node
def caldis(coord1, coord2):
    distance_km = haversine(coord1, coord2)
    return distance_km
def check_alert(nodename, temperature, humidity, speed, latitude, longitude,
node_battery, gateway_battery):
    try:
        connection = mysql.connector.connect(**db_config)
        cursor = connection.cursor()
        query = "SELECT * FROM Alert"
        cursor.execute(query)
        alert_data = cursor.fetchall()
        if not alert_data:
            return "No alert zone data found."
        alert_message = ""
        subject = f"Alert from {nodename}!! ("
        if len(alert_data) >= 8:
            if temperature > alert_data[0]:
                subject += " Temperature"
                alert_message += f"Temperature is {temperature}, which is above the alert
zone of {alert_data[0]}.n"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if humidity > alert_data[1]:
    subject += ",Humidity"
    alert_message += f"Humidity is {humidity}, which is above the alert zone of
    {alert_data[1]}.\n"
if speed > alert_data[2]:
    subject += ",Speed"
    alert_message += (
        f"Speed is {speed}, which is above the alert zone of {alert_data[2]}.\n"
    )
input_coord = (latitude, longitude)
alert_coord = (
    alert_data[4],
    alert_data[3],
)
distance_km = caldis(input_coord, alert_coord)
if distance_km < alert_data[5]:
    subject += ",Distance"
    alert_message += f"The distance between the points ({distance_km:.2f} kilometers)
    exceeds the alert radius ({alert_data[5]} kilometers). It is near the destination.\n"
    if len(alert_data) >= 10:
        if node_battery < alert_data[6]:
            subject += ",Node Battery"
            alert_message += f"Node Battery is {node_battery}, which is below the alert zone of
            {alert_data[6]}.\n"
            if gateway_battery < alert_data[7]:
                subject += ",Gateway Battery"
                alert_message += f"Gateway Battery is {gateway_battery}, which is below the alert
                zone of {alert_data[7]}.\n"
            if alert_message:
                subject += ")"
                index = subject.index("(")
                subject = subject[: index + 1] + subject[index + 2:]
                send_email(subject, alert_message)
                return alert_message
            else:
                return "All parameters are within the alert zone."
except Exception as e:
    return "Error checking alert."
finally:
    if connection.is_connected():
        cursor.close()
        connection.close()
def send_email(subject, content):
    api_url = "http://127.0.0.1:5000/api/emails"
    response = requests.get(api_url)
    emails = response.json().get("Emails", []) if response.status_code == 200 else []
    for email in emails:
        payload = {
            "lib_version": "2.6.4",
            "user_id": "UeWeiQW2WB-E8JmBj",
            "service_id": "service_t3z6fhx",
            "template_id": "template_be137zh",
            "template_params": {
                "to_email": email,
                "Subject": subject,
                "Content": content,
            },
        },

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
headers = {
    "Accept": "*/*",
    "Accept-Encoding": "gzip, deflate, br",
    "Accept-Language": "en-US,en",
    "Content-Type": "application/json",
    "Origin": "http://127.0.0.1:5500",
}
url = "https://api.emailjs.com/api/v1.0/email/send"
send_response = requests.post(
    url, data=json.dumps(payload), headers=headers)
if send_response.status_code == 200:
    print(f"Email sent successfully to {email}.")
else:
    print(f"Failed to send email to {email}. Status code:", send_response.status_code)
    time.sleep(1)
# ! API Endpoint 1: Insert Data
@app.route("/api/data", methods=["POST"])
def insert_data():
    user = request.json.get("user")
    if not user:
        return jsonify({"error": "User not provided in query parameters"}), 400
    db_config = get_db_config(user)
    current_time = datetime.now(pytz.timezone("Asia/Bangkok"))
    current_timestamp = datetime.now(pytz.timezone("Asia/Bangkok")).timestamp()
    api_url_latest = "http://127.0.0.1:5000/api/latest?user=" + user
    api_url_delay = "http://127.0.0.1:5000/api/emails?user=" + user
    delay_response = requests.get(api_url_delay)
    if delay_response.status_code != 200:
        return "Failed to fetch delay from API", 500
    datetime_str = "Mon, 21 Feb 2022 15:30:00 UTC"
    delay_data = delay_response.json()
    delay = delay_data.get("Delay", 10)
    response = requests.get(api_url_latest)
    if response.status_code != 200:
        return "Failed to fetch latest data from API", 500
    latest_time_str = ""
    latest_data = response.json()
    if latest_data is not None:
        latest_time_str = latest_data.get("Time")
        latest_time_obj = datetime.strptime(latest_time_str, "%a, %d %b %Y
%H:%M:%S %Z")
    else:
        latest_time_obj = datetime.strptime(datetime_str, "%a, %d %b %Y %H:%M:%S
%Z")
    latest_timestamp = int(latest_time_obj.timestamp()) - 25200
    time_difference = current_timestamp - latest_timestamp
    if request.method == "POST":
        data = request.get_json()
        if all(key in data for key in ("nodename", "temperature", "humidity")):
            nodename = data.get("nodename", "")
            if nodename == "":
                latest_data = fetch_latest_data()
                previousnode = latest_data[1] if latest_data else ""
                nodename = increment_node(previousnode)
            temperature = data.get("temperature", None)
            humidity = data.get("humidity", None)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

latitude = data.get("latitude", None)
longitude = data.get("longitude", None)
speed = data.get("speed", None)
ebatlvl = data.get("ebatlvl", None)
gbatlvl = data.get("gbatlvl", None)
if speed is not None and speed < 5:
    speed = 0
try:
    connection = mysql.connector.connect(**db_config)
    if connection.is_connected():
        data_to_insert = {
            "Time": current_time.strftime("%Y-%m-%d %H:%M:%S"),
            "Nodename": nodename,
            "Temperature": temperature,
            "Humidity": humidity,
            "Latitude": latitude,
            "Longitude": longitude,
            "Speed": speed,
            "ebatlvl": ebatlvl,
            "gbatlvl": gbatlvl,
        }
        cursor = connection.cursor()
        query = """
        INSERT INTO Data (Time, Nodename, Temperature, Humidity, Latitude,
        Longitude, Speed, `Node Battery`, `Gateway Battery`)
        VALUES (%(Time)s, %(Nodename)s, %(Temperature)s, %(Humidity)s,
        %(Latitude)s, %(Longitude)s, %(Speed)s, %(ebatlvl)s, %(gbatlvl)s)
        """
        cursor.execute(query, data_to_insert)
        connection.commit()
        if speed is None:
            speed = 0
        if time_difference < delay:
            check_alert(nodename, temperature, humidity, speed, latitude, longitude,
            ebatlvl, gbatlvl)
            return "Data inserted successfully"
    except Exception as e:
        return "Error inserting data into database", 500
    finally:
        if connection.is_connected():
            connection.close()
else:
    return "Invalid data", 400
else:
    return "Only POST requests are supported", 405
# TODO API Endpoint 2: Fetch Latest Data
@app.route("/api/latest")
def fetch_latest_data():
    try:
        user = request.args.get('user')
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        connection = mysql.connector.connect(**db_config)
        if connection.is_connected():
            cursor = connection.cursor(dictionary=True)
            query = """

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SELECT * FROM Data
ORDER BY Time DESC
LIMIT 1
"""
cursor.execute(query)
latest_data = cursor.fetchone()
json_data = jsonify(latest_data)
return json_data, 200
except Exception as e:
    error_message = {"error": str(e)}
    return jsonify(error_message), 500
finally:
    if "connection" in locals() and connection.is_connected():
        connection.close()
# TODO API Endpoint 3: Fetches and displays the current configuration
@app.route("/api/showconfig")
def show_config():
    try:
        user = request.args.get('user')
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        connection = mysql.connector.connect(**db_config)
        if connection.is_connected():
            cursor = connection.cursor(dictionary=True)
            query = """
SELECT * FROM Configuration
"""
            cursor.execute(query)
            rows = cursor.fetchall()
            return jsonify(rows[0])
        except Exception as e:
            return jsonify({"error": "An error occurred"}), 500
        finally:
            if "connection" in locals() and connection.is_connected():
                connection.close()
# TODO API Endpoint 4: Sends new configuration settings.
@app.route("/api/sendconfig", methods=["POST"])
def send_config():
    try:
        user = request.json.get("user")
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        connection = mysql.connector.connect(**db_config)
        if connection.is_connected():
            syncword = request.json.get("syncword")
            tx_power = request.json.get("txPower")
            freq = request.json.get("freq")
            interval = request.json.get("interval")
            cursor = connection.cursor()
            query = """
UPDATE Configuration
SET
    Syncword = %s,
    Tx_power = %s,
    Frequency = %s,

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        Tx_interval = %s
        WHERE id = 1;
        """
        cursor.execute(query, (syncword, tx_power, freq, interval))
        connection.commit()
        return jsonify({"message": "Data inserted successfully"})
    finally:
        if "connection" in locals() and connection.is_connected():
            connection.close()
# TODO API Endpoint 5: Fetches data recorded today.
@app.route("/api/show/today")
def show_values_today():
    try:
        user = request.args.get('user')
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        connection = mysql.connector.connect(**db_config)
        if connection.is_connected():
            cursor = connection.cursor(dictionary=True)
            today = date.today()
            query = """
            SELECT * FROM Data
            WHERE DATE_FORMAT(Time, '%a, %d %b %Y') = %s
            """
            cursor.execute(query, (today.strftime("%a, %d %b %Y"),))
            rows = cursor.fetchall()
            return jsonify(rows)
        except Exception as e:
            return jsonify({"error": "An error occurred"}), 500
    finally:
        if "connection" in locals() and connection.is_connected():
            connection.close()
# TODO API Endpoint 6: Fetches all recorded data.
@app.route("/api/show")
def show_values():
    try:
        user = request.args.get('user')
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        connection = mysql.connector.connect(**db_config)
        if connection.is_connected():
            cursor = connection.cursor(dictionary=True)
            query = """
            SELECT * FROM Data ORDER BY Time DESC LIMIT 160
            """
            cursor.execute(query)
            rows = cursor.fetchall()
            return jsonify(rows)
        except Exception as e:
            return jsonify({"error": "An error occurred"}), 500
    finally:
        if "connection" in locals() and connection.is_connected():
            connection.close()
# TODO API Endpoint 7: Activates or deactivates a specific node.
@app.route("/api/activate", methods=["GET"])

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

def activate():
    try:
        nodename = request.args.get("nodename")
        user = request.args.get('user')
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        connection = mysql.connector.connect(**db_config)
        cursor = connection.cursor()
        if nodename:
            cursor.execute(
                f"SELECT * FROM Node WHERE nodename = %s", (nodename,))
            node = cursor.fetchone()
            if node:
                new_status = not node[2]
                cursor.execute(
                    f"UPDATE Node SET status = %s WHERE nodename = %s",
                    (new_status, nodename),
                )
                connection.commit()
                return f"{nodename} has been {'activated' if new_status else 'deactivated'}."
            else:
                return f"{nodename} not found."
        else:
            return "Nodename not provided."
    except Exception as e:
        return False
    finally:
        if "connection" in locals() and connection.is_connected():
            connection.close()
# TODO API Endpoint 8: Fetches active node names.
@app.route("/api/node", methods=["GET"])
def node():
    try:
        user = request.args.get('user')
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        connection = mysql.connector.connect(**db_config)
        cursor = connection.cursor()
        query = "SELECT nodename FROM Node WHERE status = 1"
        cursor.execute(query)
        results = cursor.fetchall()
        nodenames = [result[0] for result in results]
        return jsonify({"nodenames": nodenames})
    except Exception as e:
        return "Error fetching true statuses."
    finally:
        if connection.is_connected():
            cursor.close()
            connection.close()
# TODO API Endpoint 9: Fetches data within a custom date range.
@app.route("/api/date", methods=["POST"])
def fetch_custom_data():
    try:
        user = request.json.get("user")
        if not user:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    return jsonify({"error": "User not provided in query parameters"}), 400
db_config = get_db_config(user)
start = request.json.get("start")
end = request.json.get("end")
print(f"Received POST data - Start Datetime: {start}, End Datetime: {end}")
connection = mysql.connector.connect(**db_config)
if connection.is_connected():
    cursor = connection.cursor(dictionary=True)
    query = f"""
    SELECT * FROM Data
    WHERE Time BETWEEN '{start}' AND '{end}'
    """
    cursor.execute(query)
    data = cursor.fetchall()
    return jsonify(data)
finally:
    if "connection" in locals() and connection.is_connected():
        connection.close()
# TODO API Endpoint 10: Changes alert conditions.
@app.route("/api/alert/changecon", methods=["POST"])
def change_con():
    try:
        user = request.json.get("user")
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        data = request.json
        temperature = data.get("Temperature")
        humidity = data.get("Humidity")
        speed = data.get("Speed")
        longitude = data.get("Longitude")
        latitude = data.get("Latitude")
        radius = data.get("Radius")
        nbat = data.get("NBattery")
        gbat = data.get("GBattery")
        connection = mysql.connector.connect(**db_config)
        if connection.is_connected():
            cursor = connection.cursor(dictionary=True)
            cursor.execute("DELETE FROM Alert")
            insert_query = "INSERT INTO Alert (Temperature, Humidity, Speed, Longitude,
            Latitude, Radius, `Node Battery`, `Gateway Battery`) VALUES (%s, %s, %s, %s,
            %s, %s, %s, %s)"
            cursor.execute(insert_query, (temperature, humidity, speed, longitude, latitude,
            radius, nbat, gbat))
            connection.commit()
            return "Data processed successfully"
    except Exception as e:
        return f"Error processing data: {str(e)}"
    finally:
        if connection and connection.is_connected():
            connection.close()
# TODO API Endpoint 11: Fetches current alert conditions.
@app.route("/api/alert/con", methods=["GET"])
def con():
    try:
        user = request.args.get('user')
        if not user:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    return jsonify({"error": "User not provided in query parameters"}), 400
db_config = get_db_config(user)
connection = mysql.connector.connect(**db_config)
cursor = connection.cursor()
query = "SELECT * FROM Alert"
cursor.execute(query)
results = cursor.fetchall()
for row in results:
    data = {
        "Temperature": row[0],
        "Humidity": row[1],
        "Speed": row[2],
        "Longitude": row[3],
        "Latitude": row[4],
        "Radius": row[5],
        "Node Battery": row[6],
        "Gateway Battery": row[7],
    }
    return jsonify(data)
except Exception as e:
    return "Error fetching data."
finally:
    if connection.is_connected():
        cursor.close()
        connection.close()
# TODO API Endpoint 12: Adds an email address for alerts.
@app.route("/api/addemail", methods=["POST"])
def insert_email():
    email = request.json.get("email", None)
    delay = request.json.get("delay", None)
    try:
        user = request.json.get("user")
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        connection = mysql.connector.connect(**db_config)
        cursor = connection.cursor()
        query = "SELECT * FROM Email WHERE email = %s"
        cursor.execute(query, (email,))
        existing_email = cursor.fetchone()
        if email == "" and delay is not None:
            update_query = """
                UPDATE Email
                SET Delay = %s
                WHERE id = 1
                """
            cursor.execute(update_query, (delay,))
            connection.commit()
            return jsonify({"message": "Changed delay."})
        if existing_email:
            return jsonify({"message": "Email already exists."}), 409
        insert_query = "INSERT INTO Email (email, Delay) VALUES (%s, %s)"
        cursor.execute(insert_query, (email, delay))
        connection.commit()
        update_query = """
            UPDATE Email
            SET Delay = %s
        """

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

WHERE id = 1
"""

cursor.execute(update_query, (delay,))
connection.commit()
return jsonify({"message": "Email inserted successfully."}), 201
except Exception as e:
    return jsonify({"message": "Error inserting email."}), 500
finally:
    if connection.is_connected():
        cursor.close()
        connection.close()
# TODO API Endpoint 13: Deletes an email address from alerts.
@app.route("/api/deleteemail", methods=["POST"])
def delete_email():
    email = request.json.get("email")
    if not email:
        return jsonify({"message": "Email is required."}), 400
    try:
        user = request.json.get("user")
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        connection = mysql.connector.connect(**db_config)
        cursor = connection.cursor()
        query = "SELECT * FROM Email WHERE email = %s"
        cursor.execute(query, (email,))
        existing_email = cursor.fetchone()
        if not existing_email:
            return jsonify({"message": "Email does not exist."}), 404
        delete_query = "DELETE FROM Email WHERE email = %s"
        cursor.execute(delete_query, (email,))
        connection.commit()
        cursor.execute("SELECT COUNT(*) FROM Email")
        count = cursor.fetchone()[0]
        if count == 1:
            reset_auto_increment_query = "ALTER TABLE Email AUTO_INCREMENT =
            2"
            cursor.execute(reset_auto_increment_query)
            connection.commit()
        return jsonify({"message": "Email deleted successfully."}), 200
    except Exception as e:
        return jsonify({"message": "Error deleting email."}), 500
    finally:
        if connection.is_connected():
            cursor.close()
            connection.close()
# TODO API Endpoint 14: Fetches configured email addresses and delay.
@app.route("/api/emails", methods=["GET"])
def get_emails():
    try:
        user = request.args.get('user')
        if not user:
            return jsonify({"error": "User not provided in query parameters"}), 400
        db_config = get_db_config(user)
        connection = mysql.connector.connect(**db_config)
        cursor = connection.cursor()
        query = "SELECT email, delay FROM Email"

```

```
cursor.execute(query)
emails = [row[0] for row in cursor.fetchall()]
query = "SELECT Delay FROM Email WHERE email = %s"
cursor.execute(query, ("delay",))
delay = cursor.fetchone()
response = {"Delay": delay[0], "Emails": emails[1:]}
return jsonify(response), 200
except Exception as e:
    return jsonify({"message": "Error fetching emails."}), 500
finally:
    if connection.is_connected():
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ง

คำสั่งสำหรับการทำงานของหน้าเว็บไซต์ Login

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha.1/dist/css/bootstrap.min.css" rel="stylesheet">
    <style>
      body,
      html {
        height: 100%;
      }
      .login-container {
        height: 100%;
        display: flex;
        justify-content: center;
        align-items: center;
      }
      .login-form {
        width: 300px;
        padding: 20px;
        border: 1px solid #ccc;
        border-radius: 10px;
        background-color: #f9f9f9;
      }
      body {
        background: radial-gradient(#343333, #101010) !important;
      }
      .btn-center {
        display: flex;
        justify-content: center;
      }
    </style>
  </head>
  <body>
    <div class="login-container">
      <div class="login-form">
        <h2 class="text-center mb-4">Login</h2>
        <form id="login-form">
          <div class="mb-3">
            <input type="text" class="form-control" id="username"
              placeholder="Username">
          </div>
          <div class="mb-3">
            <input type="password" class="form-control" id="password"
              placeholder="Password">
          </div>
          <div class="mb-3 btn-center">
            <button type="submit" class="btn btn-primary">Login</button>
          </div>
        </form>
      </div>
    </div>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha.1/dist/js/bootstrap.bundle.min.js"></script>
  </body>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

document.addEventListener("DOMContentLoaded", function () {
  const loginForm = document.getElementById("login-form");
  loginForm.addEventListener("submit", async function (e) {
    e.preventDefault();
    const username = document.getElementById("username").value;
    const password = document.getElementById("password").value;
    try {
      const response = await fetch("https://api.rungrueng.site/login", {
        method: "POST",
        headers: {
          "Content-Type": "application/json",
        },
        body: JSON.stringify({
          username,
          password,
        }),
      });
      const data = await response.json();
      if (response.ok) {
        const expirationDate = new Date(Date.now() + 1000 * 60 * 1000);
        document.cookie = "username=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
        document.cookie = "table=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
        document.cookie = "id=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
        if (data.message === "True") {
          window.location.replace("admin.html");
          document.cookie = `username=${username}; expires=${expirationDate.toUTCString()}; path=/`;
          document.cookie = `table=admin; expires=${expirationDate.toUTCString()}; path=/`;
          document.cookie = `id=1; expires=${expirationDate.toUTCString()}; path=/`;
        } else {
          window.location.replace("index.html");
          document.cookie = `username=${data.username}; expires=${expirationDate.toUTCString()}; path=/`;
          document.cookie = `table=${data.table}; expires=${expirationDate.toUTCString()}; path=/`;
          document.cookie = `id=${data.id}; expires=${expirationDate.toUTCString()}; path=/`;
        }
      } else {
        alert(data.message);
      }
    } catch (error) {
      console.error("Error:", error);
    }
  });
});
</script>
</body>
</html>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#Dashboard {
  text-align: left;
  text-decoration: underline;
  padding: 10px;
  border-radius: 5px;
  background-color: #f0f0f000;
  color: rgb(255, 255, 255);
}
html {
  scroll-behavior: smooth;
}
#temperatureChart,
#humidityChart {
  border: 2px solid #ff030300;
}
#lastUpdateTime {
  font-size: 50%;
  text-decoration: none;
  color: white;
}
body {
  background: radial-gradient(#343333, #101010) !important;
}
.material-symbols-outlined {
  font-size: 2em;
  vertical-align: middle;
  display: inline-block;
}
.batteryvalue {
  vertical-align: middle;
  display: inline-block;
}
.card-title {
  display: flex;
  align-items: center;
}
.icon-style {
  font-size: 2em;
  vertical-align: middle;
  display: inline-block;
  margin-right: 0.5em;
}
#map {
  height: 600px;
}
.position-relative {
  position: relative;
}
.invisible-button {
  position: absolute;
  top: 0;
  left: 0;
  width: 85%;
  height: 100%;
  cursor: pointer;
  z-index: 1;
  opacity: 10;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
/* Credit https://codepen.io/thedarkwolf/pen/oVZYrJ */
.live-indicator-block {
  .live-indicator {
    font-family: "Roboto", sans-serif;
    background: #ea2429;
    color: #fff;
    padding: 6px 7px;
    line-height: 1;
    border-radius: 6px;
    text-transform: uppercase;
    display: inline-block;
    vertical-align: middle;
    font-size: 12px;
    font-weight: bold;
    width: auto;
  }
  .blink {
    animation: blinker 1s cubic-bezier(0.5, 0, 1, 1) infinite alternate;
    font-size: 10px;
    margin-right: 5px;
    vertical-align: baseline;
  }
}
}
}
@keyframes blinker {
  from {
    opacity: 1;
  }
  to {
    opacity: 0;
  }
}
/* Credit https://codepen.io/jackrugile/pen/JddmaX*/
.loader {
  background: #000;
  background: radial-gradient(#222, #000);
  bottom: 0;
  left: 0;
  overflow: hidden;
  position: fixed;
  right: 0;
  top: 0;
  z-index: 99999;
}
.loader-inner {
  bottom: 0;
  height: 60px;
  left: 0;
  margin: auto;
  position: absolute;
  right: 0;
  top: 0;
  width: 100px;
}
}
.loader-line-wrap {
  animation: spin 2000ms cubic-bezier(0.175, 0.885, 0.32, 1.275) infinite;
  box-sizing: border-box;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

height: 50px;
left: 0;
overflow: hidden;
position: absolute;
top: 0;
transform-origin: 50% 100%;
width: 100px;
}
.loader-line {
border: 4px solid transparent;
border-radius: 100%;
box-sizing: border-box;
height: 100px;
left: 0;
margin: 0 auto;
position: absolute;
right: 0;
top: 0;
width: 100px;
}
.loader-line-wrap:nth-child(1) {
animation-delay: -50ms;
}
.loader-line-wrap:nth-child(2) {
animation-delay: -100ms;
}
.loader-line-wrap:nth-child(3) {
animation-delay: -150ms;
}
.loader-line-wrap:nth-child(4) {
animation-delay: -200ms;
}
.loader-line-wrap:nth-child(5) {
animation-delay: -250ms;
}
.loader-line-wrap:nth-child(1) .loader-line {
border-color: hsl(0, 80%, 60%);
height: 90px;
width: 90px;
top: 7px;
}
.loader-line-wrap:nth-child(2) .loader-line {
border-color: hsl(60, 80%, 60%);
height: 76px;
width: 76px;
top: 14px;
}
.loader-line-wrap:nth-child(3) .loader-line {
border-color: hsl(120, 80%, 60%);
height: 62px;
width: 62px;
top: 21px;
}
.loader-line-wrap:nth-child(4) .loader-line {
border-color: hsl(180, 80%, 60%);
height: 48px;
width: 48px;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    top: 28px;
  }
.loader-line-wrap:nth-child(5) .loader-line {
  border-color: hsl(240, 80%, 60%);
  height: 34px;
  width: 34px;
  top: 35px;
}
@keyframes spin {
  0%,
  15% {
    transform: rotate(0);
  }
  100% {
    transform: rotate(360deg);
  }
}
.navbar {
  z-index: 1000;
  min-height: 50px;
  border: 1px solid transparent;
}
.dropdown-menu {
  z-index: 1000;
  width: 285px;
  min-height: 50px;
  display: flex;
  justify-content: center;
  align-items: center;
}
.nav-items {
  width: 100%;
  height: 100vh;
  position: fixed;
  overflow: hidden;
  display: flex;
  left: -100%;
  z-index: 10001;
  background-color: #fff;
  flex-direction: column;
  transition: all 0.5s;
}
#check:checked ~ ul {
  left: 10000;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ฉ

คำสั่งการทำงานสำหรับหน้าเว็บไซต์ Dashboard

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Project</title>
    <meta charset="utf-8" />
    <link rel="stylesheet" href="style.css" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <script type="text/javascript"
      src="https://cdn.jsdelivr.net/npm/chart.js/dist/chart.umd.min.js"></script>
    <script
      src="https://cdn.jsdelivr.net/npm/chartjs-adapter-date-fns/dist/chartjs-adapter-
      date-fns.bundle.min.js"></script>
    <link rel="stylesheet"
      href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
    <script
      src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"></
      script>
    <script
      src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <link rel="stylesheet"
      href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,w
      ght,FILL,GRAD@20..48,100..700,0..1,-50..200" />
    <link href="https://cdn.jsdelivr.net/npm/bootstrap5-toggle@5.0.4/css/bootstrap5-
      toggle.min.css" rel="stylesheet" />
    <script src="https://cdn.jsdelivr.net/npm/bootstrap5-toggle@5.0.4/js/bootstrap5-
      toggle.jquery.min.js"></script>
    <link href="https://gitcdn.github.io/bootstrap-toggle/2.2.2/css/bootstrap-
      toggle.min.css" rel="stylesheet" />
    <script src="https://gitcdn.github.io/bootstrap-toggle/2.2.2/js/bootstrap-
      toggle.min.js"></script>
    <link rel="stylesheet"
      href="https://cdnjs.cloudflare.com/ajax/libs/animate.css/4.1.1/animate.min.css" />
    <link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet" />
    <link rel="stylesheet" href="https://unpkg.com/leaflet@1.9.4/dist/leaflet.css"
      integrity="sha256-
      p4NxAoJBhIIN+hmNHzRCf9tD/miZyoHS5obTRR9BMY=" crossorigin="" />
    <script src="https://unpkg.com/leaflet@1.9.4/dist/leaflet.js"
      integrity="sha256-20nQCchB9co0qIjJZRGuk2/Z9VM+kNiyxNV1lvTIZBo="
      crossorigin=""></script>
    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.0.6/css/all.css">
    <link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">
  </head>
  <div class="loader" id="loadingMessage">
    <div class=" loader-inner">
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
    </div>
  </div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

</li>
</li>
<li class="nav-item">
  <a class="nav-link" href="setting.html" target="_blank">Setting</a>
</li>
<li class="nav-item">
  <a class="nav-link" href="alert.html" target="_blank">Alert</a>
</li>
</ul>
<ul class="navbar-nav ml-auto">
  <!-- Log out button -->
  <li class="nav-item">
    <button class="btn btn-outline-light" onclick="logout()">Log out</button>
  </li>
</ul>
</div>
</nav>
<div class="container-fluid" style="margin-top: 0%;">
  <div class="row mt-4">
    <div class="col-12">
      <h1 class="p-2">
        <span id="Dashboard">Dashboard</span>&nbsp;<span
          id="lastUpdateTime"></span>&nbsp;<span
            <button id="toggleUrlButton" type="button" class="btn btn-
              success">Only today</button>
          </h1>
        </div>
      </div>
      <div class="row">
        <div class="col-12 col-md-12 col-lg-6 animate__animated animate__animated
          animate__pulse">
          <div class="row mt-4 flex-row">
            <div class="col-md-6">
              <div class="card mb-4">
                <div class="card-header">
                  <div class="row">
                    <div class="col">
                      <h5>Node 1</h5>
                    </div>
                    <div class="col-auto">
                      <span class="material-symbols-outlined">battery_4_bar</span>
                      <span class="batteryvalue" id="node1batt">80%</span>
                    </div>
                  </div>
                </div>
              </div>
            </div>
            <div class="card-body">
              <div class="row">
                <div class="col-12 align-content-center">
                  <div class="row">
                    <div class="col-2 col-lg-1 col-xl-2">
                      <h5 class="card-title">
                        <span class="material-symbols-outlined icon-style">
                          thermometer
                        </span>
                      </h5>
                    </div>
                    <div class="col-5 col-lg-3 col-xl-5">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <h5 class="card-title temp">Temperature</h5>
    </div>
    <div class="col-5 col-lg-7 col-xl-5">
        <h5 class="card-title" id="node2avgtemp"></h5>
    </div>
</div>
</div>
</div>
<div class="row">
    <div class="col-12">
        <div class="row align-items-center">
            <div class="col-2 col-lg-1 col-xl-2">
                <h5 class="card-title">
                    <span class="material-symbols-outlined icon-style">
                        humidity_percentage
                    </span>
                </h5>
            </div>
            <div class="col-5 col-lg-3 col-xl-5">
                <h5 class="card-title humi">Humidity</h5>
            </div>
            <div class="col-5 col-lg-7 col-xl-5">
                <h5 class="card-title" id="node1avghumi"></h5>
            </div>
        </div>
        </div>
        <div class="row justify-content-end">
            <div class="col-auto">
                <label class="invisible-button"
                    onclick="toggleChanged('Node1', 'node1 toggle')"></label>
                <input id="node1 toggle" type="checkbox"
                    checked data-toggle="toggle"
                    data-onstyle="success" data-offstyle="danger" />
            </div>
        </div>
    </div>
</div>
</div>
</div>
<div class="col-md-6 mb-4">
    <div class="card mb-4">
        <div class="card-header">
            <div class="row">
                <div class="col">
                    <h5>Node 2</h5>
                </div>
                <div class="col-auto">
                    <span class="material-symbols-outlined">battery_4_bar</span>
                    <span class="batteryvalue" id="node2batt">80%</span>
                </div>
            </div>
        </div>
    </div>
</div>
<div class="card-body">
    <div class="row">
        <div class="col-12">
            <div class="row align-items-center">
                <div class="col-2 col-lg-1 col-xl-2">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<h5 class="card-title">
  <span class="material-symbols-outlined icon-style">
    thermometer
  </span>
</h5>
</div>
<div class="col-5 col-lg-3 col-xl-5">
  <h5 class="card-title temp">Temperature</h5>
</div>
<div class="col-5 col-lg-7 col-xl-5">
  <h5 class="card-title" id="node1avgtemp"></h5>
</div>
</div>
</div>
</div>
<div class="row">
  <div class="col-12">
    <div class="row align-items-center">
      <div class="col-2 col-lg-1 col-xl-2">
        <h5 class="card-title">
          <span class="material-symbols-outlined icon-style">
            humidity_percentage
          </span>
        </h5>
      </div>
      <div class="col-5 col-lg-3 col-xl-5">
        <h5 class="card-title humi">Humidity</h5>
      </div>
      <div class="col-5 col-lg-7 col-xl-5">
        <h5 class="card-title" id="node2avghumi"></h5>
      </div>
    </div>
  </div>
  <div class="row justify-content-end">
    <div class="col-auto">
      <label class="invisible-button"
        onclick="toggleChanged('Node2', 'node2toggle')"></label>
      <input id="node2toggle" type="checkbox" checked data-
        toggle="toggle"
        data-onstyle="success" data-offstyle="danger" />
    </div>
  </div>
</div>
</div>
</div>
</div>
</div>
<div class="row">
  <div class="col-md-6 mb-4">
    <div class="card mb-4">
      <div class="card-header">
        <div class="row">
          <div class="col">
            <h5>Node 3</h5>
          </div>
          <div class="col-auto">
            <span class="material-symbols-outlined">battery_4_bar</span>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <span class="batteryvalue" id="node3batt">80%</span>
    </div>
</div>
</div>
</div>
<div class="card-body">
    <div class="row">
        <div class="col-12">
            <div class="row align-items-center">
                <div class="col-2 col-lg-1 col-xl-2">
                    <h5 class="card-title">
                        <span class="material-symbols-outlined icon-style">
                            thermometer
                        </span>
                    </h5>
                </div>
                <div class="col-5 col-lg-3 col-xl-5">
                    <h5 class="card-title temp">Temperature</h5>
                </div>
                <div class="col-5 col-lg-7 col-xl-5">
                    <h5 class="card-title" id="node3avgtemp"></h5>
                </div>
            </div>
        </div>
        </div>
        <div class="row">
            <div class="col-12">
                <div class="row align-items-center">
                    <div class="col-2 col-lg-1 col-xl-2">
                        <h5 class="card-title">
                            <span class="material-symbols-outlined icon-style">
                                humidity_percentage
                            </span>
                        </h5>
                    </div>
                    <div class="col-5 col-lg-3 col-xl-5">
                        <h5 class="card-title humi">Humidity</h5>
                    </div>
                    <div class="col-5 col-lg-7 col-xl-5">
                        <h5 class="card-title" id="node3avghumi"></h5>
                    </div>
                </div>
            </div>
        </div>
        </div>
        <div class="row justify-content-end">
            <div class="col-auto">
                <label class="invisible-button"
                    onclick="toggleChanged('Node3', 'node3toggle')"></label>
                <input id="node3toggle" type="checkbox" checked data-
                    toggle="toggle"
                    data-onstyle="success" data-offstyle="danger" />
            </div>
        </div>
    </div>
</div>
</div>
</div>
</div>
</div>
<div class="col-md-6">
    <div class="card mb-4">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<div class="card-header">
  <div class="row">
    <div class="col">
      <h5>Node 4</h5>
    </div>
    <div class="col-auto">
      <span class="material-symbols-outlined">battery_4_bar</span>
      <span class="batteryvalue" id="node4batt">80%</span>
    </div>
  </div>
</div>
<div class="card-body">
  <div class="row">
    <div class="col-12">
      <div class="row align-items-center">
        <div class="col-2 col-lg-1 col-xl-2">
          <h5 class="card-title">
            <span class="material-symbols-outlined icon-style">
              thermometer
            </span>
          </h5>
        </div>
        <div class="col-5 col-lg-3 col-xl-5">
          <h5 class="card-title temp">Temperature</h5>
        </div>
        <div class="col-5 col-lg-7 col-xl-5">
          <h5 class="card-title" id="node4avgtemp"></h5>
        </div>
      </div>
    </div>
  </div>
  <div class="row">
    <div class="col-12">
      <div class="row align-items-center">
        <div class="col-2 col-lg-1 col-xl-2">
          <h5 class="card-title">
            <span class="material-symbols-outlined icon-style">
              humidity_percentage
            </span>
          </h5>
        </div>
        <div class="col-5 col-lg-3 col-xl-5">
          <h5 class="card-title humi">Humidity</h5>
        </div>
        <div class="col-5 col-lg-7 col-xl-5">
          <h5 class="card-title" id="node4avghumi"></h5>
        </div>
      </div>
    </div>
  </div>
</div>
<div class="row justify-content-end">
  <div class="col-auto">
    <label class="invisible-button"
      onclick="toggleChanged('Node4', 'node4toggle')"></label>
    <input id="node4toggle" type="checkbox" checked data-
      toggle="toggle"
      data-onstyle="success" data-offstyle="danger" />
  </div>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

    )
  </div>
</div>
</div>
</div>
</div>
<div class="row mt-4 mt-sm-4 mt-lg-4">
  <div class="col-12">
    <div class="card text-center" id="humiditygraph">
      <div class="card-header">
        <h4>Humidity Graphs</h4>
      </div>
      <div class="card-body">
        <canvas id="humidityChart" width="100%" height="30"></canvas>
      </div>
      <div class="card-footer text-muted">
        <div class="slidecontainer" style="text-align: center; font-size: medium; padding-top: 1%">
          Show latest <span id="sliderValuehumi">20</span>
          <span style="margin-right: 1%">values</span> (
          <span id="sliderMinValuehumi">1</span>
          <input type="range" min="1" max="40" value="20" class="slider"
            id="myRangehumi" />
          <span id="sliderMaxValuehumi ">40</span>
        )
      </div>
    </div>
  </div>
</div>
</div>
</div>
</div>
<div class="row mt-4 mt-sm-4 mt-lg-4 mb-5">
  <div class="col-12">
    <div class="card text-center" id="Locations">
      <div class="card-header">
        <h4>Location</h4>
      </div>
      <div class="card-body">
        <div id="map"></div>
      </div>
      <div class="card-footer text-muted"></div>
    </div>
  </div>
</div>
</div>
</div>
</div>
<script> var container = document.querySelector(".container-fluid");
var dropdown = document.getElementById("navbarDropdownMenuLink");
var dropdownItems = document.querySelectorAll(".dropdown-item");
let dataLengthOld = 0;
let dataLengthNew = 0;
let url = "https://api.rungrueng.site";
let today = true;
let temperatureChart;
let humidityChart;
let r = 20;
let rH = 20;
let currentTime = new Date();
let map;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

let filteredData = [];
let node1Data, node2Data, node3Data, node4Data;
function getCookie(name) {
  const cookies = document.cookie.split(";");
  for (let cookie of cookies) {
    const [cookieName, cookieValue] = cookie.split("=");
    if (cookieName.trim() === name) {
      return cookieValue;
    }
  }
  return null;
}
let username = getCookie("username");
function logout() {
  document.cookie = "username=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  document.cookie = "table=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  document.cookie = "id=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  window.location.replace("login.html");
}
document.addEventListener("DOMContentLoaded", function () {
  function capitalizeFirstLetter(string) {
    return string.charAt(0).toUpperCase() + string.slice(1);
  }
  const navbarBrand = document.getElementById("name");
  if (username) {
    navbarBrand.textContent = capitalizeFirstLetter(username);
  }
  if (!username) {
    window.location.replace("login.html");
    return;
  }
});
async function submitForm() {
  var start = document.getElementById("start").value;
  var end = document.getElementById("end").value;
  window.location.href = `results.html?start=${start}&end=${end}` + "&user=" +
  username;
}
async function fetchData() {
  let response;
  if (today) {
    response = await fetch(url + "/api/show/today?user=" + username);
  } else {
    response = await fetch(url + "/api/show?user=" + username);
  }
  const data = await response.json();
  if (Array.isArray(data) && data.length > 0) {
  } else {
  }
  return data;
}
document.addEventListener("DOMContentLoaded", function () {
  var randomDelay = Math.floor(Math.random() * 1001);
  var styleElement = document.createElement("style");
  styleElement.textContent = `
    :root {
      --animate-duration: ${randomDelay + 500}ms !important;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        --animate-delay: 1s !important;
    }
};
document.head.appendChild(styleElement);
setTimeout(function () {
    $("#loadingMessage").hide();
}, randomDelay);
});
function mapDataValues(data, key) {
    return data.map((item) => (item[key] === 0 ? null : parseFloat(item[key])));
}
function createChart(canvasId, data1, data2, data3, data4, label1, label2, label3, label4,
yAxisLabel, type) {
    function mapTimestamps(data) {
        return data.map((item) => new Date(item.Time).toISOString().replace(".000Z", ""));
    }
    const latestData1 = data1;
    const latestData2 = data2;
    const latestData3 = data3;
    const latestData4 = data4;
    let timestamp1 = mapTimestamps(latestData1);
    let timestamp2 = mapTimestamps(latestData2);
    let timestamp3 = mapTimestamps(latestData3);
    let timestamp4 = mapTimestamps(latestData4);
    if (type === "temp") {
        let timestamp1temp = timestamp1.slice(-1 * r);
        let timestamp2temp = timestamp2.slice(-1 * r);
        let timestamp3temp = timestamp3.slice(-1 * r);
        let timestamp4temp = timestamp4.slice(-1 * r);
        let temperatures1 = mapDataValues(latestData1, "Temperature");
        let temperatures2 = mapDataValues(latestData2, "Temperature");
        let temperatures3 = mapDataValues(latestData3, "Temperature");
        let temperatures4 = mapDataValues(latestData4, "Temperature");
        temperatures1 = temperatures1.slice(-r);
        temperatures2 = temperatures2.slice(-r);
        temperatures3 = temperatures3.slice(-r);
        temperatures4 = temperatures4.slice(-r);
        let sum = 0;
        for (let i = 0; i < temperatures1.length; i++) {
            sum += temperatures1[i];
        }
        const averagetem = (sum / temperatures1.length).toFixed(2);
        const convertedData1 = timestamp1temp.map((timestamp, index) => ({
            x: timestamp,
            y: temperatures1[index],
        }));
        const convertedData2 = timestamp2temp.map((timestamp, index) => ({
            x: timestamp,
            y: temperatures2[index],
        }));
        const convertedData3 = timestamp3temp.map((timestamp, index) => ({
            x: timestamp,
            y: temperatures3[index],
        }));
        const convertedData4 = timestamp4temp.map((timestamp, index) => ({
            x: timestamp,
            y: temperatures4[index],
        }));
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    ));
    const data = {
      datasets: [
        {
          label: label1,
          data: convertedData1.map((item) => ({
            x: item.x,
            y: item.y,
          })),
          yAxisID: "y1",
          backgroundColor: "rgba(0, 255, 0, 0.8)",
          borderColor: "rgba(0, 255, 0, 1)",
          borderWidth: 2,
        },
        {
          label: label2,
          data: convertedData2.map((item) => ({
            x: item.x,
            y: item.y,
          })),
          yAxisID: "y1",
          backgroundColor: "rgba(255, 255, 0, 0.8)",
          borderColor: "rgba(255, 255, 0, 1)",
          borderWidth: 2,
        },
        {
          label: label3,
          data: convertedData3.map((item) => ({
            x: item.x,
            y: item.y,
          })),
          yAxisID: "y1",
          backgroundColor: "rgba(54, 162, 235, 0.8)",
          borderColor: "rgba(54, 162, 235, 1)",
          borderWidth: 2,
        },
        {
          label: label4,
          data: convertedData4.map((item) => ({
            x: item.x,
            y: item.y,
          })),
          yAxisID: "y1",
          backgroundColor: "rgba(255, 26, 104, 0.8)",
          borderColor: "rgba(255, 26, 104, 1)",
          borderWidth: 2,
        },
      ],
    };
    const config = {
      type: "line",
      data,
      options: {
        animation: {
          duration: false,
        },
      },
      elements: {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    line: {
      tension: 0.35,
    },
  },
  scales: {
    x: {
      type: "time",
      ticks: {
        major: {
          enabled: true,
        },
      },
    },
    time: {
      unit: "minute",
      displayFormats: {
        minute: "HH:mm",
      },
    },
    title: {
      display: true,
      text: "Time",
    },
    grid: {
      color: "rgba(255, 255, 255, 0.2)",
    },
  },
  y1: {
    grid: {
      color: "rgba(255, 255, 255, 0.2)",
    },
    title: {
      display: true,
      text: yAxisLabel,
    },
    suggestedMin: parseFloat(averagetem) - 1,
    suggestedMax: parseFloat(averagetem) + 2,
  },
},
},
plugins: {
  legend: {
    display: true,
    position: "top",
  },
},
},
};
const temperatureChart = new Chart(document.getElementById(canvasId), config);
temperatureCharts = temperatureChart;
} else if (type === "humid") {
  let timestamp1humi = timestamp1.slice(-1 * rH);
  let timestamp2humi = timestamp2.slice(-1 * rH);
  let timestamp3humi = timestamp3.slice(-1 * rH);
  let timestamp4humi = timestamp4.slice(-1 * rH);
  let humidities1 = mapDataValues(latestData1, "Humidity");
  let humidities2 = mapDataValues(latestData2, "Humidity");
  let humidities3 = mapDataValues(latestData3, "Humidity");
  let humidities4 = mapDataValues(latestData4, "Humidity");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

humiditys1 = humiditys1.slice(-rH);
humiditys2 = humiditys2.slice(-rH);
humiditys3 = humiditys3.slice(-rH);
humiditys4 = humiditys4.slice(-rH);
sum = 0;
for (let i = 0; i < humiditys1.length; i++) {
  sum += humiditys1[i];
}
const averagehumidity = (sum / humiditys1.length).toFixed(2);
const convertedData1 = timestamp1humi.map((timestamp, index) => ({
  x: timestamp,
  y: humiditys1[index],
}));
const convertedData2 = timestamp2humi.map((timestamp, index) => ({
  x: timestamp,
  y: humiditys2[index],
}));
const convertedData3 = timestamp3humi.map((timestamp, index) => ({
  x: timestamp,
  y: humiditys3[index],
}));
const convertedData4 = timestamp4humi.map((timestamp, index) => ({
  x: timestamp,
  y: humiditys4[index],
}));
const data = {
  datasets: [
    {
      label: label1,
      data: convertedData1.map((item) => ({
        x: item.x,
        y: item.y,
      })),
      yAxisID: "y1",
      backgroundColor: "rgba(0, 255, 0, 0.8)",
      borderColor: "rgba(0, 255, 0, 1)",
      borderWidth: 2,
    },
    {
      label: label2,
      data: convertedData2.map((item) => ({
        x: item.x,
        y: item.y,
      })),
      yAxisID: "y1",
      backgroundColor: "rgba(255, 255, 0, 0.8)",
      borderColor: "rgba(255, 255, 0, 1)",
      borderWidth: 2,
    },
    {
      label: label3,
      data: convertedData3.map((item) => ({
        x: item.x,
        y: item.y,
      })),
      yAxisID: "y1",
      backgroundColor: "rgba(54, 162, 235, 0.8)",

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

borderColor: "rgba(54, 162, 235, 1)",
borderWidth: 2,
},
{
label: label4,
data: convertedData4.map((item) => ({
x: item.x,
y: item.y,
})),
yAxisID: "y1",
backgroundColor: "rgba(255, 26, 104, 0.8)",
borderColor: "rgba(255, 26, 104, 1)",
borderWidth: 2,
},
],
];
const config = {
type: "line",
data,
options: {
animation: {
duration: false,
},
elements: {
line: {
tension: 0.35,
},
},
scales: {
x: {
type: "time",
ticks: {
major: {
enabled: true,
},
},
},
time: {
unit: "minute",
displayFormats: {
minute: "HH:mm",
},
},
},
title: {
display: true,
text: "Time",
},
},
grid: {
color: "rgba(255, 255, 255, 0.2)",
},
},
y1: {
grid: {
color: "rgba(255, 255, 255, 0.2)",
},
},
title: {
display: true,
text: yAxisLabel,
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    },
    suggestedMin: parseFloat(averagehumidity) - 1.5,
    suggestedMax: parseFloat(averagehumidity) + 1,
  },
},
plugins: {
  legend: {
    display: true,
    position: "top",
  },
},
},
};
const humidityChart = new Chart(document.getElementById(canvasId), config);
humidityCharts = humidityChart;
} else {
}
}
function updateCardLayout() {
  var screenWidth = window.innerWidth;
  var screenHeight = window.innerHeight;
  if (screenWidth <= 500) {
    var humidityChartCanvas = document.getElementById("humidityChart");
    var temperatureChartCanvas = document.getElementById("temperatureChart");
    humidityChartCanvas.height = 120;
    temperatureChartCanvas.height = 120;
  }
  if (screenWidth <= 1300 && screenWidth > 992) {
    var tempElements = document.querySelectorAll(".temp");
    tempElements.forEach(function (element) {
      element.textContent = "Temp";
    });
    var humiElements = document.querySelectorAll(".humi");
    humiElements.forEach(function (element) {
      element.textContent = "Humid";
    });
  } else {
    var tempElements = document.querySelectorAll(".temp");
    tempElements.forEach(function (element) {
      element.textContent = "Temperature";
    });
    var humiElements = document.querySelectorAll(".humi");
    humiElements.forEach(function (element) {
      element.textContent = "Humidity";
    });
  }
}
updateCardLayout();
window.addEventListener("resize", function () {
  updateCardLayout();
});
async function fetchDataAndToggle() {
  try {
    const response = await fetch(url + "/api/node?user=" + username);
    const data = await response.json();
    updateCheckbox(data, "Node1", "node1toggle");
    updateCheckbox(data, "Node2", "node2toggle");
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        updateCheckbox(data, "Node3", "node3toggle");
        updateCheckbox(data, "Node4", "node4toggle");
    } catch (error) {}
}
function updateCheckbox(data, nodeName, checkboxId) {
    const nodeExists = data.nodenames.includes(nodeName);
    document.getElementById(checkboxId).checked = nodeExists;
    if (nodeExists) {
        `${checkboxId}`.bootstrapToggle("on");
    } else {
        `${checkboxId}`.bootstrapToggle("off");
    }
}
function toggleChanged(nodeName, checkboxId) {
    `${checkboxId}`.bootstrapToggle("toggle");
    fetch(url + "/api/activate?nodename=" + nodeName + "&user=" + username);
}
function filterDataByNode(data, node) {
    return data.filter((item) => item.Nodename === node);
}
function updateAvg(data, idtemp, idhumi) {
    const temperatures = data.map((item) => parseFloat(item.Temperature));
    const humidity = data.map((item) => parseFloat(item.Humidity));
    let sumTem = 0;
    let sumHum = 0;
    for (let i = 0; i < temperatures.length; i++) {
        sumTem += temperatures[i];
    }
    const averageTemp = sumTem / temperatures.length;
    for (let i = 0; i < humidity.length; i++) {
        sumHum += humidity[i];
    }
    const averageHumidity = sumHum / humidity.length;
    const avgTempElement = document.getElementById(idtemp);
    avgTempElement.innerHTML = `: ${averageTemp.toFixed(2)} °C`;
    const avgHumiElement = document.getElementById(idhumi);
    avgHumiElement.innerHTML = `: ${averageHumidity.toFixed(2)} %`;
}
function updatenodebatt(data, idbatt) {
    if (data.length > 0) {
        const nodeData = data[data.length - 1];
        let batteryLevels = nodeData["Node Battery"];
        const avgBattElement = document.getElementById(idbatt);
        if (batteryLevels < 0) {
            avgBattElement.innerHTML = `: Charging`;
        } else {
            avgBattElement.innerHTML = `: ${batteryLevels} %`;
        }
    } else {
    }
}
function updateLastUpdateTime(data) {
    let updateLastUpdateTime = data;
    const lastUpdateTimeElement = document.getElementById("lastUpdateTime");
    if (lastUpdateTimeElement) {
        const latestDataPoint = updateLastUpdateTime[updateLastUpdateTime.length - 1];
        if (latestDataPoint && latestDataPoint.Time) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

let slider = document.getElementById("myRange");
let sliderHumi = document.getElementById("myRangehumi");
r = slider.value;
rH = sliderHumi.value;
slider.oninput = function () {
  r = this.value;
  var output = document.getElementById("sliderValue");
  output.innerHTML = r;
};
sliderHumi.oninput = function () {
  rH = this.value;
  var output = document.getElementById("sliderValuehumi");
  output.innerHTML = rH;
};
slider.onmouseup = async function () {
  const data = await fetchData();
  const node1Data = filterDataByNode(data, "Node1");
  const node2Data = filterDataByNode(data, "Node2");
  const node3Data = filterDataByNode(data, "Node3");
  const node4Data = filterDataByNode(data, "Node4");
  temperatureCharts.destroy();
  createChart("temperatureChart", node1Data, node2Data, node3Data, node4Data,
  "Temperature (Node1)", "Temperature (Node2)", "Temperature (Node3)", "Temperature
(Node4)", "Temperature (°C)", "temp");
};
sliderHumi.onmouseup = async function () {
  const data = await fetchData();
  const node1Data = filterDataByNode(data, "Node1");
  const node2Data = filterDataByNode(data, "Node2");
  const node3Data = filterDataByNode(data, "Node3");
  const node4Data = filterDataByNode(data, "Node4");
  humidityCharts.destroy();
  createChart("humidityChart", node1Data, node2Data, node3Data, node4Data,
  "Humidity (Node1)", "Humidity (Node2)", "Humidity (Node3)", "Humidity (Node4)",
  "Humidity (%)", "humid");
};
toggleUrlButton.addEventListener("click", () => {
  today = !today;
  let dataLengthOld = 0;
  if (temperatureCharts && temperatureCharts.destroy && humidityCharts &&
humidityCharts.destroy) {
    temperatureCharts.destroy();
    humidityCharts.destroy();
  }
  if (today) {
    toggleUrlButton.classList.add("btn-success");
    toggleUrlButton.classList.remove("btn-danger");
  } else {
    toggleUrlButton.classList.add("btn-danger");
    toggleUrlButton.classList.remove("btn-success");
  }
  main();
});
function drawmap(data) {
  if (map && map.remove) {
    map.eachLayer(function (layer) {
      layer.remove();
    });
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    });
    map.off();
    map.remove();
  }
  let number;
  for (let i = 400; i < data.length; i++) {
    let d = data.slice(-i);
    if (d[0].Nodename === "Node1") {
      number = i;
      break;
    }
  }
  var latestData = data;
  var latitudes = mapDataValues(latestData, "Latitude");
  var longitudes = mapDataValues(latestData, "Longitude");
  if (latitudes.length > 0 && longitudes.length > 0) {
    let sum = 0;
    for (let i = 0; i < latitudes.length; i++) {
      sum += latitudes[i];
    }
    const averagelat = (sum / latitudes.length).toFixed(6);
    sum = 0;
    for (let i = 0; i < longitudes.length; i++) {
      sum += longitudes[i];
    }
    const averagelon = (sum / longitudes.length).toFixed(6);
    map = L.map("map").setView([averagelat, averagelon], 20);
    L.tileLayer("https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png").addTo(map);
    var counter = 0;
    for (var i = 0; i < latestData.length; i++) {
      if (latestData[i].Latitude !== null && latestData[i].Longitude !== null) {
        var colorIntensity = 255 * (i / latestData.length);
        var markerColor = `rgb(0, 250, ${255 - colorIntensity})`;
        var nodeBattery = latestData[i]["Node Battery"] < 0 ? 0 :
latestData[i]["Node Battery"];
        var popupContent = `Time: ${latestData[i].Time} +7<br>Nodename:
${latestData[i].Nodename}<br>Node Battery: ${nodeBattery} %<br>Temperature:
${latestData[i].Temperature} °C<br>Humidity: ${latestData[i].Humidity} %`;
        for (var j = i + 1; j < latestData.length; j++) {
          if (latestData[i].Latitude === latestData[j].Latitude &&
latestData[i].Longitude === latestData[j].Longitude && Math.abs(new
Date(latestData[i].Time) - new Date(latestData[j].Time)) <= 10000) {
            nodeBattery = latestData[j]["Node Battery"] < 0 ? 0 :
latestData[j]["Node Battery"];
            popupContent += `<br><br>Time: ${latestData[j].Time}
+7<br>Nodename: ${latestData[j].Nodename}<br>Node Battery: ${nodeBattery}
%<br>Temperature: ${latestData[j].Temperature} °C<br>Humidity:
${latestData[j].Humidity} %`;
            i = j;
          }
        }
      }
      counter++;
      var marker = L.marker([latestData[i].Latitude, latestData[i].Longitude], {
        icon: L.divIcon({
          className: "custom-marker",
          iconSize: [20, 20],
          iconAnchor: [10, 10],

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        html: `<div style="background-color: ${markerColor};
        border: 2px solid black; width: 20px; height: 20px; border-radius:
        50%;">${counter}</div>`,
    }),
    }).addTo(map);
    marker.bindPopup(popupContent).openPopup();
    }
    } else {
    }
}
function createGraph(node1Data, node2Data, node3Data, node4Data) {
    temperatureCharts.destroy();
    createChart("temperatureChart", node1Data, node2Data, node3Data, node4Data,
    "Temperature (Node1)", "Temperature (Node2)", "Temperature (Node3)", "Temperature
    (Node4)", "Temperature (°C)", "temp");
    humidityCharts.destroy();
    createChart("humidityChart", node1Data, node2Data, node3Data, node4Data,
    "Humidity (Node1)", "Humidity (Node2)", "Humidity (Node3)", "Humidity (Node4)",
    "Humidity (%)", "humid");
}
async function main() {
    const data = await fetchData();
    dataLengthNew = data.length;
    if (dataLengthOld == 0) {
        node1Data = filterDataByNode(data, "Node1");
        node2Data = filterDataByNode(data, "Node2");
        node3Data = filterDataByNode(data, "Node3");
        node4Data = filterDataByNode(data, "Node4");
        updateAvg(node1Data, "node1avgtemp", "node1avghumi");
        updateAvg(node2Data, "node2avgtemp", "node2avghumi");
        updateAvg(node3Data, "node3avgtemp", "node3avghumi");
        updateAvg(node4Data, "node4avgtemp", "node4avghumi");
        updatenodebatt(node1Data.reverse(), "node1batt");
        updatenodebatt(node2Data.reverse(), "node2batt");
        updatenodebatt(node3Data.reverse(), "node3batt");
        updatenodebatt(node4Data.reverse(), "node4batt");
        createChart("temperatureChart", node1Data, node2Data, node3Data, node4Data,
        "Temperature (Node1)", "Temperature (Node2)", "Temperature (Node3)",
        "Temperature (Node4)", "Temperature (°C)", "temp");
        createChart("humidityChart", node1Data, node2Data, node3Data, node4Data,
        "Humidity (Node1)", "Humidity (Node2)", "Humidity (Node3)", "Humidity
        (Node4)", "Humidity (%)", "humid");
        updateLastUpdateTime(node1Data.reverse());
        updategatewaybat(node1Data);
        drawmap(data.reverse());
        dataLengthOld = dataLengthNew;
    } else if (dataLengthOld !== dataLengthNew) {
        node1Data = filterDataByNode(data, "Node1");
        node2Data = filterDataByNode(data, "Node2");
        node3Data = filterDataByNode(data, "Node3");
        node4Data = filterDataByNode(data, "Node4");
        updateAvg(node1Data, "node1avgtemp", "node1avghumi");
        updateAvg(node2Data, "node2avgtemp", "node2avghumi");
        updateAvg(node3Data, "node3avgtemp", "node3avghumi");
        updateAvg(node4Data, "node4avgtemp", "node4avghumi");
        updatenodebatt(node1Data.reverse(), "node1batt");
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    updatenodebatt(node2Data.reverse(), "node2batt");
    updatenodebatt(node3Data.reverse(), "node3batt");
    updatenodebatt(node4Data.reverse(), "node4batt");
    createGraph(node1Data, node2Data, node3Data, node4Data);
    updatelog(data);
    updategatewaybat(node1Data.reverse());
    updateLastUpdateTime(node1Data);
    drawmap(data.reverse());
    dataLengthOld = dataLengthNew;
  }
}
main();
fetchDataAndToggle();
setInterval(fetchDataAndToggle, 1000);
setInterval(main, 10000); </script>
</body>
</html>

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





```

<body>
  <div class="container">
    <div class="row justify-content-center mt-4">
    </div>
    <div class="row mt-4">
      <div class="col-md-12">
        <div class="card mb-4">
          <div class="card-header">
            <div class="row">
              <div class="col">
                <h5>LoRa Settings</h5>
              </div>
            </div>
          </div>
          <div class="card-body">
            <form id="loraSettingsForm">
              <div class="form-group">
                <label for="syncword">Syncword (0-255):</label>
                <div class="input-group">
                  <input type="number" class="form-control" id="syncword"
                    name="syncword" value="242">
                  <div class="input-group-append">
                  </div>
                </div>
              </div>
              <div class="form-group">
                <label for="txPower">Tx Power (0-20):</label>
                <div class="input-group">
                  <input type="number" class="form-control" id="txPower"
                    name="txPower" value="20">
                  <div class="input-group-append">
                    <span class="input-group-text">dBm</span>
                  </div>
                </div>
              </div>
              <div class="form-group">
                <label for="frequency">Frequency (900-931):</label>
                <div class="input-group">
                  <input type="number" class="form-control" id="frequency"
                    name="frequency"
                    value="923">
                  <div class="input-group-append">
                    <span class="input-group-text">MHz</span>
                  </div>
                </div>
              </div>
              <div class="form-group">
                <label for="transmissionInterval">Transmission Interval:</label>
                <div class="input-group">
                  <input type="text" class="form-control" id="transmissionInterval"
                    value="3"
                    name="transmissionInterval">
                  <div class="input-group-append">
                    <span class="input-group-text">minutes</span>
                  </div>
                </div>
              </div>
            </form>
          </div>
        </div>
      </div>
    </div>
  </div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <div class="row justify-content-center">
            <button type="submit" class="btn btn-primary"
                onclick="submitForm()">Submit</button>
            <button type="button" class="btn btn-secondary ml-2"
                onclick="resetForm()">Reset</button>
        </div>
    </form>
</div>
</div>
</div>
</div>
</div>
</div>
<script> let url = "https://api.rungrueng.site";
function getCookie(name) {
    const cookies = document.cookie.split(";");
    for (let cookie of cookies) {
        const [cookieName, cookieValue] = cookie.split("=");
        if (cookieName.trim() === name) {
            return cookieValue;
        }
    }
    return null;
}
let username = getCookie("username");
async function submitFormss() {
    var start = document.getElementById("start").value;
    var end = document.getElementById("end").value;
    window.location.href = `results.html?start=${start}&end=${end}`;
}
function logout() {
    document.cookie = "username=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
    document.cookie = "table=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
    document.cookie = "id=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
    window.location.replace("login.html");
}
document.addEventListener("DOMContentLoaded", function () {
    function capitalizeFirstLetter(string) {
        return string.charAt(0).toUpperCase() + string.slice(1);
    }
    const navbarBrand = document.getElementById("name");
    if (username) {
        navbarBrand.textContent = capitalizeFirstLetter(username);
    }
    if (!username) {
        window.location.replace("login.html");
        return;
    }
});
document.addEventListener("DOMContentLoaded", function () {
    var randomDelay = Math.floor(Math.random() * 1001);
    var styleElement = document.createElement("style");
    styleElement.textContent = `
        :root {
            --animate-duration: ${randomDelay + 500}ms !important;
            --animate-delay: 1s !important;
        }
    `;
});

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

document.head.appendChild(styleElement);
setTimeout(function () {
    $("#loadingMessage").hide();
}, randomDelay);
});
async function submitForm() {
    const url = "https://api.rungrueng.site";
    const syncword = document.getElementById("syncword").value;
    const txPower = document.getElementById("txPower").value;
    const frequency = document.getElementById("frequency").value;
    const transmissionInterval = document.getElementById("transmissionInterval").value;
    const body = {
        syncword: syncword,
        txPower: txPower,
        freq: frequency,
        interval: transmissionInterval,
        user: username,
    };
    const syncwordHex = parseInt(syncword).toString(16).toUpperCase();
    const message = `Submitting Form with the following values:
        Syncword: ${syncword} (${syncwordHex})
        Tx Power: ${txPower} dBm
        Frequency: ${frequency} MHz
        Transmission Interval: ${transmissionInterval} minutes`;
    if (!confirm(message)) {
        return;
    }
    const options = {
        method: "POST",
        headers: {
            "Content-Type": "application/json",
        },
        body: JSON.stringify(body),
    };
    try {
        const response = await fetch(url + "/api/sendconfig", options);
        if (response.ok) {
            alert("Form submitted! LoRa Settings have been changed.");
        } else {
            alert("Error submitting form. Please try again later.");
        }
    } catch (error) {
        alert("Error submitting form. Please try again later.");
    }
    location.reload();
}
async function fetchDataAndUpdateForm() {
    try {
        const response = await fetch(url + "/api/showconfig?user=" + username);
        const data = await response.json();
        document.getElementById("syncword").value = data.Syncword;
        document.getElementById("txPower").value = data.Tx_power;
        document.getElementById("frequency").value = data.Frequency;
        document.getElementById("transmissionInterval").value = data.Tx_interval;
    } catch (error) {}
}
function resetForm() {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
document.getElementById("syncword").value = "242";
document.getElementById("txPower").value = "20";
document.getElementById("frequency").value = "923";
document.getElementById("transmissionInterval").value = "3";
submitForm();
}
fetchDataAndUpdateForm();
</script>
</body>
</html>
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Project</title>
    <meta charset="utf-8" />
    <link rel="stylesheet" href="style.css" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <script type="text/javascript"
      src="https://cdn.jsdelivr.net/npm/chart.js/dist/chart.umd.min.js"></script>
    <script
      src="https://cdn.jsdelivr.net/npm/chartjs-adapter-date-fns/dist/chartjs-adapter-date-
      fns.bundle.min.js"></script>
    <link rel="stylesheet"
      href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
    <script
      src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"></
      script>
    <script
      src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <link rel="stylesheet"
      href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,w
      ght,FILL,GRAD@20..48,100..700,0..1,-50..200" />
    <link href="https://cdn.jsdelivr.net/npm/bootstrap5-toggle@5.0.4/css/bootstrap5-
      toggle.min.css" rel="stylesheet" />
    <script src="https://cdn.jsdelivr.net/npm/bootstrap5-toggle@5.0.4/js/bootstrap5-
      toggle.jquery.min.js"></script>
    <link href="https://gitcdn.github.io/bootstrap-toggle/2.2.2/css/bootstrap-
      toggle.min.css" rel="stylesheet" />
    <script src="https://gitcdn.github.io/bootstrap-toggle/2.2.2/js/bootstrap-
      toggle.min.js"></script>
    <link rel="stylesheet"
      href="https://cdnjs.cloudflare.com/ajax/libs/animate.css/4.1.1/animate.min.css" />
    <link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet" />
    <link rel="stylesheet" href="https://unpkg.com/leaflet@1.9.4/dist/leaflet.css"
      integrity="sha256-p4NxAoJBhIIN+hmNHzRCf9tD/miZyoHS5obTRR9BMY="
      crossorigin="" />
    <script src="https://unpkg.com/leaflet@1.9.4/dist/leaflet.js"
      integrity="sha256-20nQCchB9co0qIjJZRGuk2/Z9VM+kNiyxNV1lvTIZBo="
      crossorigin=""></script>
    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.0.6/css/all.css">
    <link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">
  </head>
  <div class="loader" id="loadingMessage">
    <div class="loader-inner">
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
    </div>
  </div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

</li>
</li>
<li class="nav-item">
  <a class="nav-link" href="setting.html" target="_blank">Setting</a>
</li>
<li class="nav-item">
  <a class="nav-link" href="alert.html" target="_blank">Alert</a>
</li>
</ul>
<ul class="navbar-nav ml-auto">
<li class="nav-item">
  <button class="btn btn-outline-light" onclick="logout()">Log out</button>
</li>
</ul>
</div>
</nav>
<div class="container-fluid" style="margin-top: 0%;">
<div class="row mt-4">
<div class="col-12">
<h1 class="p-2">
  <span id="Dashboard">Dashboard</span>&nbsp;<span
  id="lastUpdateTime"></span>
</h1>
</div>
</div>
<div class="row">
<div class="col-12 col-md-12 col-lg-12 animate__animated
animate__animated animate__pulse">
<div class="row mt-4 flex-row">
<div class="col-md-6">
<div class="card mb-4">
<div class="card-header">
<div class="row">
<div class="col">
<h5>Node 1</h5>
</div>
<div class="col-auto">
<span class="material-symbols-outlined">battery_4_bar</span>
<span class="batteryvalue" id="node1batt">80%</span>
</div>
</div>
</div>
<div class="card-body">
<div class="row">
<div class="col-12 align-content-center">
<div class="row">
<div class="col-2 col-lg-1 col-xl-2">
<h5 class="card-title">
  <span class="material-symbols-outlined icon-style">
  thermometer
  </span>
</h5>
</div>
<div class="col-5 col-lg-3 col-xl-5">
<h5 class="card-title temp">Temperature</h5>
</div>
<div class="col-5 col-lg-7 col-xl-5">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <h5 class="card-title" id="node2avgtemp"></h5>
    </div>
</div>
</div>
</div>
</div>
<div class="row">
    <div class="col-12">
        <div class="row align-items-center">
            <div class="col-2 col-lg-1 col-xl-2">
                <h5 class="card-title">
                    <span class="material-symbols-outlined icon-style">
                        humidity_percentage
                    </span>
                </h5>
            </div>
            <div class="col-5 col-lg-3 col-xl-5">
                <h5 class="card-title humi">Humidity</h5>
            </div>
            <div class="col-5 col-lg-7 col-xl-5">
                <h5 class="card-title" id="node1avghumi"></h5>
            </div>
        </div>
        </div>
        </div>
        <div class="row justify-content-end">
            <div class="col-auto">
                <label class="invisible-button"
                    onclick="toggleChanged('Node1', 'node1toggle')"></label>
                <input id="node1toggle" type="checkbox" checked data-
                    toggle="toggle"
                    data-onstyle="success" data-offstyle="danger" />
            </div>
        </div>
    </div>
</div>
</div>
</div>
<div class="col-md-6 mb-4">
    <div class="card mb-4">
        <div class="card-header">
            <div class="row">
                <div class="col">
                    <h5>Node 2</h5>
                </div>
                <div class="col-auto">
                    <span class="material-symbols-outlined">battery_4_bar</span>
                    <span class="batteryvalue" id="node2batt">80%</span>
                </div>
            </div>
        </div>
    </div>
</div>
<div class="card-body">
    <div class="row">
        <div class="col-12">
            <div class="row align-items-center">
                <div class="col-2 col-lg-1 col-xl-2">
                    <h5 class="card-title">
                        <span class="material-symbols-outlined icon-style">
                            thermometer
                    </span>
                </h5>
            </div>
        </div>
    </div>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        </span>
      </h5>
    </div>
  </div>
  <div class="col-5 col-lg-3 col-xl-5">
    <h5 class="card-title temp">Temperature</h5>
  </div>
  <div class="col-5 col-lg-7 col-xl-5">
    <h5 class="card-title" id="node1avgtemp"></h5>
  </div>
</div>
</div>
<div class="row">
  <div class="col-12">
    <div class="row align-items-center">
      <div class="col-2 col-lg-1 col-xl-2">
        <h5 class="card-title">
          <span class="material-symbols-outlined icon-style">
            humidity_percentage
          </span>
        </h5>
      </div>
      <div class="col-5 col-lg-3 col-xl-5">
        <h5 class="card-title humi">Humidity</h5>
      </div>
      <div class="col-5 col-lg-7 col-xl-5">
        <h5 class="card-title" id="node2avghumi"></h5>
      </div>
    </div>
  </div>
  <div class="row justify-content-end">
    <div class="col-auto">
      <label class="invisible-button"
        onclick="toggleChanged('Node2', 'node2toggle')"></label>
      <input id="node2toggle" type="checkbox" checked data-
        toggle="toggle"
        data-onstyle="success" data-offstyle="danger" />
    </div>
  </div>
</div>
</div>
</div>
</div>
</div>
<div class="row">
  <div class="col-md-6 mb-4">
    <div class="card mb-4">
      <div class="card-header">
        <div class="row">
          <div class="col">
            <h5>Node 3</h5>
          </div>
          <div class="col-auto">
            <span class="material-symbols-outlined">battery_4_bar</span>
            <span class="batteryvalue" id="node3batt">80%</span>
          </div>
        </div>
      </div>
    </div>
  </div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</div>
<div class="card-body">
  <div class="row">
    <div class="col-12">
      <div class="row align-items-center">
        <div class="col-2 col-lg-1 col-xl-2">
          <h5 class="card-title">
            <span class="material-symbols-outlined icon-style">
              thermometer
            </span>
          </h5>
        </div>
        <div class="col-5 col-lg-3 col-xl-5">
          <h5 class="card-title temp">Temperature</h5>
        </div>
        <div class="col-5 col-lg-7 col-xl-5">
          <h5 class="card-title" id="node3avgtemp"></h5>
        </div>
      </div>
    </div>
  </div>
  <div class="row">
    <div class="col-12">
      <div class="row align-items-center">
        <div class="col-2 col-lg-1 col-xl-2">
          <h5 class="card-title">
            <span class="material-symbols-outlined icon-style">
              humidity_percentage
            </span>
          </h5>
        </div>
        <div class="col-5 col-lg-3 col-xl-5">
          <h5 class="card-title humi">Humidity</h5>
        </div>
        <div class="col-5 col-lg-7 col-xl-5">
          <h5 class="card-title" id="node3avghumi"></h5>
        </div>
      </div>
    </div>
  </div>
  <div class="row justify-content-end">
    <div class="col-auto">
      <label class="invisible-button"
        onclick="toggleChanged('Node3', 'node3toggle')></label>
      <input id="node3toggle" type="checkbox" checked data-
        toggle="toggle"
        data-onstyle="success" data-offstyle="danger" />
    </div>
  </div>
</div>
</div>
</div>
</div>
<div class="col-md-6">
  <div class="card mb-4">
    <div class="card-header">
      <div class="row">
        <div class="col">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    <h5>Node 4</h5>
  </div>
  <div class="col-auto">
    <span class="material-symbols-outlined">battery_4_bar</span>
    <span class="batteryvalue" id="node4batt">80%</span>
  </div>
</div>
</div>
<div class="card-body">
  <div class="row">
    <div class="col-12">
      <div class="row align-items-center">
        <div class="col-2 col-lg-1 col-xl-2">
          <h5 class="card-title">
            <span class="material-symbols-outlined icon-style">
              thermometer
            </span>
          </h5>
        </div>
        <div class="col-5 col-lg-3 col-xl-5">
          <h5 class="card-title temp">Temperature</h5>
        </div>
        <div class="col-5 col-lg-7 col-xl-5">
          <h5 class="card-title" id="node4avgtemp"></h5>
        </div>
      </div>
    </div>
    <div class="row">
      <div class="col-12">
        <div class="row align-items-center">
          <div class="col-2 col-lg-1 col-xl-2">
            <h5 class="card-title">
              <span class="material-symbols-outlined icon-style">
                humidity_percentage
              </span>
            </h5>
          </div>
          <div class="col-5 col-lg-3 col-xl-5">
            <h5 class="card-title humi">Humidity</h5>
          </div>
          <div class="col-5 col-lg-7 col-xl-5">
            <h5 class="card-title" id="node4avghumi"></h5>
          </div>
        </div>
      </div>
    </div>
  </div>
  <div class="row justify-content-end">
    <div class="col-auto">
      <label class="invisible-button"
        onclick="toggleChanged('Node4', 'node4toggle')></label>
      <input id="node4toggle" type="checkbox" checked data-
        toggle="toggle"
        data-onstyle="success" data-offstyle="danger" />
    </div>
  </div>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        </div>
    </div>
</div>
</div>
</div>
</div>
<div class="col-sm-12 col-md-12 col-lg-6 mt-sm-0 mt-lg-4 ">
</div>
</div>
<div class="row mt-4 mt-sm-4 mt-lg-0">
<div class="col-12">
<div class="card text-center" id="temperaturegraph">
<div class="card-header">
<h4>Temperature Graphs</h4>
</div>
<div class="card-body">
<canvas id="temperatureChart" width="100%" height="30"></canvas>
</div>
<div class="card-footer text-muted">
<div class="slidecontainer" style="text-align: center; font-size: medium;
padding-top: 1%">
Show latest <span id="sliderValue">20</span>
<span style="margin-right: 1%">values</span> (
<span id="sliderMinValue">1</span>
<input type="range" min="1" max="40" value="20" class="slider"
id="myRange" />
<span id="sliderMaxValue ">40</span>
)
</div>
</div>
</div>
</div>
</div>
</div>
<div class="row mt-4 mt-sm-4 mt-lg-4">
<div class="col-12">
<div class="card text-center" id="humiditygraph">
<div class="card-header">
<h4>Humidity Graphs</h4>
</div>
<div class="card-body">
<canvas id="humidityChart" width="100%" height="30"></canvas>
</div>
<div class="card-footer text-muted">
<div class="slidecontainer" style="text-align: center; font-size: medium;
padding-top: 1%">
Show latest <span id="sliderValuehumi">20</span>
<span style="margin-right: 1%">values</span> (
<span id="sliderMinValuehumi">1</span>
<input type="range" min="1" max="40" value="20" class="slider"
id="myRangehumi" />
<span id="sliderMaxValuehumi ">40</span>
)
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<div class="row mt-4 mt-sm-4 mt-lg-4 mb-5">
<div class="col-12">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<div class="card text-center" id="Locations">
  <div class="card-header">
    <h4>Location</h4>
  </div>
  <div class="card-body">
    <div id="map"></div>
  </div>
  <div class="card-footer text-muted"></div>
</div>
</div>
</div>
</div>
<script> var container = document.querySelector(".container-fluid");
var dropdown = document.getElementById("navbarDropdownMenuLink");
var dropdownItems = document.querySelectorAll(".dropdown-item");
function getCookie(name) {
  const cookies = document.cookie.split(";");
  for (let cookie of cookies) {
    const [cookieName, cookieValue] = cookie.split("=");
    if (cookieName.trim() === name) {
      return cookieValue;
    }
  }
  return null;
}
let username = getCookie("username");
let dataLengthOld = 0,
dataLengthNew = 0,
url = "https://api.rungrueng.site",
temperatureChart,
humidityChart,
r = 20,
rH = 20,
currentTime = new Date(),
map,
filteredData = [],
Databetween;
async function submitForm() {
  var start = document.getElementById("start").value;
  var end = document.getElementById("end").value;
  window.location.href = `results.html?start=${start}&end=${end}` + `&user=` +
username;
}
function logout() {
  document.cookie = "username=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  document.cookie = "table=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  document.cookie = "id=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  window.location.replace("login.html");
}
document.addEventListener("DOMContentLoaded", function () {
  function capitalizeFirstLetter(string) {
    return string.charAt(0).toUpperCase() + string.slice(1);
  }
  const navbarBrand = document.getElementById("name");
  if (username) {
    navbarBrand.textContent = capitalizeFirstLetter(username);
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    if (!username) {
        window.location.replace("login.html");
        return;
    }
});
async function fetchData() {
    const urlParams = new URLSearchParams(window.location.search);
    const start = urlParams.get("start");
    const end = urlParams.get("end");
    document.getElementById("start").value = start || "2023-12-29T00:00";
    document.getElementById("end").value = end || "2023-12-29T15:00";
    if (!start || !end) {
        return null;
    }
    const response = await fetch("https://api.rungrueng.site/api/date", {
        method: "POST",
        headers: {
            "Content-Type": "application/json",
        },
        body: JSON.stringify({
            start: start,
            end: end,
            user: username,
        }),
    });
    const data = await response.json();
    if (Array.isArray(data) && data.length > 0) {
    } else {
    }
    return data;
}
document.addEventListener("DOMContentLoaded", function () {
    var randomDelay = Math.floor(Math.random() * 1001);
    var styleElement = document.createElement("style");
    styleElement.textContent = `
        :root {
            --animate-duration: ${randomDelay + 500}ms !important;
            --animate-delay: 1s !important;
        }
    `;
    document.head.appendChild(styleElement);
    setTimeout(function () {
        $("#loadingMessage").hide();
    }, randomDelay);
});
function mapDataValues(data, key) {
    return data.map((item) => (item[key] === 0 ? null : item[key]));
}
function createChart(canvasId, data1, data2, data3, data4, label1, label2, label3, label4,
yAxisLabel, type) {
    function mapTimestamps(data) {
        return data.map((item) => new Date(item.Time).toISOString().replace(".000Z", ""));
    }
    const latestData1 = data1;
    const latestData2 = data2;
    const latestData3 = data3;
    const latestData4 = data4;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

let timestamp1 = mapTimestamps(latestData1);
let timestamp2 = mapTimestamps(latestData2);
let timestamp3 = mapTimestamps(latestData3);
let timestamp4 = mapTimestamps(latestData4);
if (type === "temp") {
  let timestamp1temp = timestamp1.slice(-1 * r);
  let timestamp2temp = timestamp2.slice(-1 * r);
  let timestamp3temp = timestamp3.slice(-1 * r);
  let timestamp4temp = timestamp4.slice(-1 * r);
  let temperatures1 = mapDataValues(latestData1, "Temperature");
  let temperatures2 = mapDataValues(latestData2, "Temperature");
  let temperatures3 = mapDataValues(latestData3, "Temperature");
  let temperatures4 = mapDataValues(latestData4, "Temperature");
  temperatures1 = temperatures1.slice(-r);
  temperatures2 = temperatures2.slice(-r);
  temperatures3 = temperatures3.slice(-r);
  temperatures4 = temperatures4.slice(-r);
  let sum = 0;
  for (let i = 0; i < temperatures1.length; i++) {
    sum += temperatures1[i];
  }
  const averagetem = (sum / temperatures1.length).toFixed(2);
  const convertedData1 = timestamp1temp.map((timestamp, index) => ({
    x: timestamp,
    y: temperatures1[index],
  }));
  const convertedData2 = timestamp2temp.map((timestamp, index) => ({
    x: timestamp,
    y: temperatures2[index],
  }));
  const convertedData3 = timestamp3temp.map((timestamp, index) => ({
    x: timestamp,
    y: temperatures3[index],
  }));
  const convertedData4 = timestamp4temp.map((timestamp, index) => ({
    x: timestamp,
    y: temperatures4[index],
  }));
  const data = {
    datasets: [
      {
        label: label1,
        data: convertedData1.map((item) => ({
          x: item.x,
          y: item.y,
        })),
        yAxisID: "y1",
        backgroundColor: "rgba(0, 255, 0, 0.8)",
        borderColor: "rgba(0, 255, 0, 1)",
        borderWidth: 2,
      },
      {
        label: label2,
        data: convertedData2.map((item) => ({
          x: item.x,
          y: item.y,
        })),
      },
    ],
  };
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

yAxisID: "y1",
backgroundColor: "rgba(255, 255, 0, 0.8)",
borderColor: "rgba(255, 255, 0, 1)",
borderWidth: 2,
},
{
label: label3,
data: convertedData3.map((item) => ({
x: item.x,
y: item.y,
})),
yAxisID: "y1",
backgroundColor: "rgba(54, 162, 235, 0.8)",
borderColor: "rgba(54, 162, 235, 1)",
borderWidth: 2,
},
{
label: label4,
data: convertedData4.map((item) => ({
x: item.x,
y: item.y,
})),
yAxisID: "y1",
backgroundColor: "rgba(255, 26, 104, 0.8)",
borderColor: "rgba(255, 26, 104, 1)",
borderWidth: 2,
},
],
};
const config = {
type: "line",
data,
options: {
animation: {
duration: false,
},
elements: {
line: {
tension: 0.35,
},
},
scales: {
x: {
type: "time",
ticks: {
major: {
enabled: true,
},
font: (context) => {
const boldedTicks = context.tick && context.tick.major ? "900" : "";
return {
weight: boldedTicks,
};
},
},
time: {
unit: "minute",

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

    ));
    const convertedData2 = timestamp2humi.map((timestamp, index) => ({
      x: timestamp,
      y: humiditys2[index],
    }));
    const convertedData3 = timestamp3humi.map((timestamp, index) => ({
      x: timestamp,
      y: humiditys3[index],
    }));
    const convertedData4 = timestamp4humi.map((timestamp, index) => ({
      x: timestamp,
      y: humiditys4[index],
    }));
    const data = {
      datasets: [
        {
          label: label1,
          data: convertedData1.map((item) => ({
            x: item.x,
            y: item.y,
          })),
          yAxisID: "y1",
          backgroundColor: "rgba(0, 255, 0, 0.8)",
          borderColor: "rgba(0, 255, 0, 1)",
          borderWidth: 2,
        },
        {
          label: label2,
          data: convertedData2.map((item) => ({
            x: item.x,
            y: item.y,
          })),
          yAxisID: "y1",
          backgroundColor: "rgba(255, 255, 0, 0.8)",
          borderColor: "rgba(255, 255, 0, 1)",
          borderWidth: 2,
        },
        {
          label: label3,
          data: convertedData3.map((item) => ({
            x: item.x,
            y: item.y,
          })),
          yAxisID: "y1",
          backgroundColor: "rgba(54, 162, 235, 0.8)",
          borderColor: "rgba(54, 162, 235, 1)",
          borderWidth: 2,
        },
        {
          label: label4,
          data: convertedData4.map((item) => ({
            x: item.x,
            y: item.y,
          })),
          yAxisID: "y1",
          backgroundColor: "rgba(255, 26, 104, 0.8)",
          borderColor: "rgba(255, 26, 104, 1)",
        },
      ],
    };

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        borderWidth: 2,
    },
],
};
const config = {
  type: "line",
  data,
  options: {
    animation: {
      duration: false,
    },
    elements: {
      line: {
        tension: 0.35,
      },
    },
    scales: {
      x: {
        type: "time",
        ticks: {
          major: {
            enabled: true,
          },
        },
        time: {
          unit: "minute",
          displayFormats: {
            minute: "HH:mm",
          },
        },
        title: {
          display: true,
          text: "Time",
        },
        grid: {
          color: "rgba(255, 255, 255, 0.2)",
        },
      },
      y1: {
        grid: {
          color: "rgba(255, 255, 255, 0.2)",
        },
        title: {
          display: true,
          text: yAxisLabel,
        },
        suggestedMin: parseFloat(averagehumidity) - 1.5,
        suggestedMax: parseFloat(averagehumidity) + 1,
      },
    },
  },
  plugins: {
    legend: {
      display: true,
      position: "top",
    },
  },
},
},

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    };
    const humidityChart = new Chart(document.getElementById(canvasId), config);
    humidityCharts = humidityChart;
  } else {
  }
}
function updateCardLayout() {
  var screenWidth = window.innerWidth;
  var screenHeight = window.innerHeight;
  if (screenWidth <= 500) {
    var humidityChartCanvas = document.getElementById("humidityChart");
    var temperatureChartCanvas = document.getElementById("temperatureChart");
    humidityChartCanvas.height = 120;
    temperatureChartCanvas.height = 120;
    main();
  }
  if (screenWidth <= 1300 && screenWidth > 992) {
    var tempElements = document.querySelectorAll(".temp");
    tempElements.forEach(function (element) {
      element.textContent = "Temp";
    });
    var humiElements = document.querySelectorAll(".humi");
    humiElements.forEach(function (element) {
      element.textContent = "Humid";
    });
  } else {
    var tempElements = document.querySelectorAll(".temp");
    tempElements.forEach(function (element) {
      element.textContent = "Temperature";
    });
    var humiElements = document.querySelectorAll(".humi");
    humiElements.forEach(function (element) {
      element.textContent = "Humidity";
    });
  }
}
updateCardLayout();
window.addEventListener("resize", function () {
  updateCardLayout();
});
async function fetchDataAndToggle() {
  try {
    const response = await fetch(url + "/api/node?user=" + username);
    const data = await response.json();
    updateCheckbox(data, "Node1", "node1toggle");
    updateCheckbox(data, "Node2", "node2toggle");
    updateCheckbox(data, "Node3", "node3toggle");
    updateCheckbox(data, "Node4", "node4toggle");
  } catch (error) {}
}
function updateCheckbox(data, nodeName, checkboxId) {
  const nodeExists = data.nodenames.includes(nodeName);
  document.getElementById(checkboxId).checked = nodeExists;
  if (nodeExists) {
    $('#${checkboxId}`).bootstrapToggle("on");
  } else {
    $('#${checkboxId}`).bootstrapToggle("off");
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
  }
  function toggleChanged(nodeName, checkboxId) {
    $('#${checkboxId}`).bootstrapToggle("toggle");
    fetch(url + "/api/activate?nodename=" + nodeName + "&user=" + username);
  }
  function filterDataByNode(data, node) {
    return data.filter((item) => item.Nodename === node);
  }
  function updateAvg(data, idtemp, idhumi) {
    const temperatures = data.map((item) => item.Temperature);
    const humidity = data.map((item) => item.Humidity);
    let sumTem = 0;
    let sumHum = 0;
    for (let i = 0; i < temperatures.length; i++) {
      sumTem += temperatures[i];
    }
    const averageTemp = sumTem / temperatures.length;
    for (let i = 0; i < humidity.length; i++) {
      sumHum += humidity[i];
    }
    const averageHumidity = sumHum / humidity.length;
    const avgTempElement = document.getElementById(idtemp);
    if (avgTempElement) {
      avgTempElement.innerHTML = ` ${averageTemp.toFixed(2)} °C`;
    }
    const avgHumiElement = document.getElementById(idhumi);
    if (avgHumiElement) {
      avgHumiElement.innerHTML = ` ${averageHumidity.toFixed(2)} %`;
    }
  }
  function updateLastUpdateTime(data) {
    const lastUpdateTimeElement = document.getElementById("lastUpdateTime");
    if (lastUpdateTimeElement) {
      let startDataPoint = data[0];
      let endDataPoint = data[data.length - 1];
      let startdate = new Date(startDataPoint.Time);
      let enddate = new Date(endDataPoint.Time);
      const options = {
        weekday: "short",
        year: "numeric",
        month: "short",
        day: "numeric",
        hour: "numeric",
        minute: "numeric",
        second: "numeric",
        timeZone: "UTC",
      };
      startdate = startdate.toLocaleString("en-US", options);
      enddate = enddate.toLocaleString("en-US", options);
      Databetween = `(Data between ${startdate} and ${enddate})`;
      lastUpdateTimeElement.textContent = Databetween;
    }
  }
}
function updatelog(responseData) {
  const options = {
    hour: "numeric",

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        minute: "numeric",
        second: "numeric",
        timeZone: "UTC",
    });
    responseData.forEach((element) => {
        const elementTimestamp = new Date(element.Time).getTime();
        const currentTimestamp = currentTime.getTime() + 7 * 60 * 60 * 1000;
        if (elementTimestamp > currentTimestamp) {
            filteredData.push(element);
            filteredData.reverse();
        }
    });
    const messages = filteredData.map((item) => {
        const formattedTime = new Date(item.Time).toLocaleString("en-US", options);
        return `[${formattedTime}] Received data from ${item.Nodename} temperature is
    ${item.Temperature} °C and humidity is ${item.Humidity}% <br>`;
    });
    const cardTextElement = document.getElementById("realtimelog");
    cardTextElement.innerHTML = messages.join("");
    }
    let slider = document.getElementById("myRange");
    let sliderHumi = document.getElementById("myRangehumi");
    r = slider.value;
    rH = sliderHumi.value;
    slider.oninput = function () {
        r = this.value;
        var output = document.getElementById("sliderValue");
        output.innerHTML = r;
    };
    sliderHumi.oninput = function () {
        rH = this.value;
        var output = document.getElementById("sliderValuehumi");
        output.innerHTML = rH;
    };
    slider.onmouseup = async function () {
        const data = await fetchData();
        const node1Data = filterDataByNode(data, "Node1");
        const node2Data = filterDataByNode(data, "Node2");
        const node3Data = filterDataByNode(data, "Node3");
        const node4Data = filterDataByNode(data, "Node4");
        temperatureCharts.destroy();
        createChart("temperatureChart", node1Data, node2Data, node3Data, node4Data,
    "Temperature (Node1)", "Temperature (Node2)", "Temperature (Node3)", "Temperature
    (Node4)", "Temperature (°C)", "temp");
    };
    sliderHumi.onmouseup = async function () {
        const data = await fetchData();
        const node1Data = filterDataByNode(data, "Node1");
        const node2Data = filterDataByNode(data, "Node2");
        const node3Data = filterDataByNode(data, "Node3");
        const node4Data = filterDataByNode(data, "Node4");
        humidityCharts.destroy();
        createChart("humidityChart", node1Data, node2Data, node3Data, node4Data,
    "Humidity (Node1)", "Humidity (Node2)", "Humidity (Node3)", "Humidity (Node4)",
    "Humidity (%)", "humid");
    };
    function drawmap(data) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (map && map.remove) {
  map.eachLayer(function (layer) {
    layer.remove();
  });
  map.off();
  map.remove();
}
let number;
for (let i = 400; i < data.length; i++) {
  let d = data.slice(-i);
  if (d[0].Nodename === "Node1") {
    number = i;
    break;
  }
}
var latestData = data;
var latitudes = mapDataValues(latestData, "Latitude");
var longitudes = mapDataValues(latestData, "Longitude");
if (latitudes.length > 0 && longitudes.length > 0) {
  let sum = 0;
  for (let i = 0; i < latitudes.length; i++) {
    sum += latitudes[i];
  }
  const averagelat = (sum / latitudes.length).toFixed(6);
  sum = 0;
  for (let i = 0; i < longitudes.length; i++) {
    sum += longitudes[i];
  }
  const averagelon = (sum / longitudes.length).toFixed(6);
  map = L.map("map").setView([averagelat, averagelon], 20);
  L.tileLayer("https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png").addTo(map);
  var counter = 0;
  for (var i = 0; i < latestData.length; i++) {
    if (latestData[i].Latitude !== null && latestData[i].Longitude !== null) {
      var colorIntensity = 255 * (i / latestData.length);
      var markerColor = `rgb(0, 250, ${255 - colorIntensity})`;
      var nodeBattery = latestData[i]["Node Battery"] < 0 ? 0 : latestData[i]["Node Battery"];
      var popupContent = `Time: ${latestData[i].Time} +7<br>Nodename:
      ${latestData[i].Nodename}<br>Node Battery: ${nodeBattery} %<br>Temperature:
      ${latestData[i].Temperature} °C<br>Humidity: ${latestData[i].Humidity} %`;
      for (var j = i + 1; j < latestData.length; j++) {
        if (latestData[i].Latitude === latestData[j].Latitude &&
        latestData[i].Longitude === latestData[j].Longitude &&
        Math.abs(new Date(latestData[i].Time) - new Date(latestData[j].Time)) <= 10000) {
          nodeBattery = latestData[j]["Node Battery"] < 0 ? 0 : latestData[j]["Node Battery"];
          popupContent += `<br><br>Time: ${latestData[j].Time}
          +7<br>Nodename: ${latestData[j].Nodename}<br>Node Battery: ${nodeBattery}
          %<br>Temperature: ${latestData[j].Temperature} °C<br>Humidity:
          ${latestData[j].Humidity} %`;
          i = j;
        }
      }
      counter++;
      var marker = L.marker([latestData[i].Latitude, latestData[i].Longitude], {
        icon: L.divIcon({

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        className: "custom-marker",
        iconSize: [20, 20],
        iconAnchor: [10, 10],
        html: `

${counter}</div>`,
    }),
    }).addTo(map);
    marker.bindPopup(popupContent).openPopup();
    }
    } else {
    }
}
function createGraph(node1Data, node2Data, node3Data, node4Data) {
    temperatureCharts.destroy();
    createChart("temperatureChart", node1Data, node2Data, node3Data, node4Data,
"Temperature (Node1)", "Temperature (Node2)", "Temperature (Node3)", "Temperature
(Node4)", "Temperature (°C)", "temp");
    humidityCharts.destroy();
    createChart("humidityChart", node1Data, node2Data, node3Data, node4Data,
"Humidity (Node1)", "Humidity (Node2)", "Humidity (Node3)", "Humidity (Node4)",
"Humidity (%)", "humid");
}
function updatenodebatt(data, idbatt) {
    if (data.length > 0) {
        const nodeData = data[data.length - 1];
        const batteryLevels = nodeData["Node Battery"];
        const avgBattElement = document.getElementById(idbatt);
        avgBattElement.innerHTML = `: ${batteryLevels} %`;
    } else {
    }
}
}
async function main() {
    const data = await fetchData();
    dataLengthNew = data.length;
    if (dataLengthOld == 0) {
        const node1Data = filterDataByNode(data, "Node1");
        const node2Data = filterDataByNode(data, "Node2");
        const node3Data = filterDataByNode(data, "Node3");
        const node4Data = filterDataByNode(data, "Node4");
        updateLastUpdateTime(node1Data);
        console.table(data);
        updateAvg(node1Data, "node1avgtemp", "node1avghumi");
        updateAvg(node2Data, "node2avgtemp", "node2avghumi");
        updateAvg(node3Data, "node3avgtemp", "node3avghumi");
        updateAvg(node4Data, "node4avgtemp", "node4avghumi");
        updatenodebatt(node1Data, "node1batt");
        updatenodebatt(node1Data, "node2batt");
        updatenodebatt(node1Data, "node3batt");
        updatenodebatt(node1Data, "node4batt");
        createChart("temperatureChart", node1Data, node2Data, node3Data, node4Data,
"Temperature (Node1)", "Temperature (Node2)", "Temperature (Node3)",
"Temperature (Node4)", "Temperature (°C)", "temp");
        createChart("humidityChart", node1Data, node2Data, node3Data, node4Data,
"Humidity (Node1)", "Humidity (Node2)", "Humidity (Node3)", "Humidity
(Node4)", "Humidity (%)", "humid");
        drawmap(data);
    }
}


```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        dataLengthOld = dataLengthNew;
    }
}
main();
fetchDataAndToggle();
</script>
</body>
</html>
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Alert</title>
    <meta charset="utf-8" />
    <link rel="stylesheet" href="style.css" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
    <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"></script>
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <link rel="stylesheet" href="https://unpkg.com/leaflet@1.9.4/dist/leaflet.css" integrity="sha256-p4NxAoJBhIIN+hmNHzRCf9tD/miZyoHS5obTRR9BMY=" crossorigin="" />
    <script src="https://unpkg.com/leaflet@1.9.4/dist/leaflet.js" integrity="sha256-20nQCchB9co0qIjJZRGuk2/Z9VM+kNiyxNV1lvTIZBo=" crossorigin=""></script>
  </head>
  <div class="loader" id="loadingMessage">
    <div class="loader-inner">
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
      <div class="loader-line-wrap">
        <div class="loader-line"></div>
      </div>
    </div>
  </div>
  <nav class="navbar navbar-expand-lg navbar-dark navbar-light animate__animated animate__fadeInDownBig" style="background-color: #484748d7">
    <a id="name" class="navbar-brand" href="/">Admin</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNav">
      <ul class="navbar-nav">
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" href="#" id="navbarDropdownMenuLink" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">Home
        </a>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

}
.email-container {
display: flex;
justify-content: space-between;
margin-bottom: 5px;
}
</style>
<body>
<div class="container">
<div class="row justify-content-center mt-4">
</div>
<div class="row mt-4">
<div class="col-md-6">
<div class="card mb-4">
<div class="card-header">
<div class="row">
<div class="col">
<h5>Alert condition</h5>
</div>
</div>
</div>
<div class="card-body">
<form id="condition">
<div class="form-group">
<label for="temperature">Temperature (Maximum temperature to
trigger alert):</label>
<div class="input-group">
<input type="number" class="form-control" id="temperature"
name="Temperature"
value="">
<div class="input-group-append">
<span class="input-group-text">°C</span>
</div>
</div>
</div>
<div class="form-group">
<label for="humidity">Humidity (Maximum humidity to trigger
alert):</label>
<div class="input-group">
<input type="number" class="form-control" id="humidity"
name="Humidity" value="">
<div class="input-group-append">
<span class="input-group-text">%</span>
</div>
</div>
</div>
<div class="form-group">
<label for="speed">Speed (Maximum Speed to trigger alert):</label>
<div class="input-group">
<input type="number" class="form-control" id="speed" name="Speed"
value="">
<div class="input-group-append">
<span class="input-group-text">km</span>
</div>
</div>
</div>
<div class="form-group">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<label for="latitude">Latitude (Latitude of the destination):</label>
<input type="text" class="form-control" id="latitude" value=""
name="Latitude">
</div>
<div class="form-group">
<label for="longitude">Longitude (Longitude of the
destination):</label>
<input type="text" class="form-control" id="longitude" value=""
name="Longitude">
</div>
<div class="form-group">
<label for="distance">Distance (Distance from destination to trigger
alert):</label>
<div class="input-group">
<input type="number" class="form-control" id="radius"
name="Radius" value="">
<div class="input-group-append">
<span class="input-group-text">km</span>
</div>
</div>
</div>
<div class="form-group">
<label for="nodeBattery">Node Battery (Minimum node battery level to
trigger
alert):</label>
<div class="input-group">
<input type="number" class="form-control" id="nodeBattery"
name="NBattery" value="">
<div class="input-group-append">
<span class="input-group-text">%</span>
</div>
</div>
</div>
<div class="form-group">
<label for="gatewayBattery">Gateway Battery (Minimum gateway
battery level to
trigger
alert):</label>
<div class="input-group">
<input type="number" class="form-control" id="gatewayBattery"
name="GBattery"
value="">
<div class="input-group-append">
<span class="input-group-text">%</span>
</div>
</div>
</div>
<div class="row justify-content-center">
<button type="submit" class="btn btn-primary"
onclick="submitForm()">Submit</button>
<button type="button" class="btn btn-secondary ml-2"
onclick="resetForm()">Reset</button>
</div>
</form>
</div>
</div>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<div class="col-md-6">
  <div class="card mb-4">
    <div class="card-header">
      <div class="row">
        <div class="col">
          <h5>Email Settings</h5>
        </div>
      </div>
    </div>
    <div class="card-body">
      <form id="email">
        <div class="form-group">
          <label for="temperature">Recipient Email:</label>
          <input type="email" class="form-control" id="Email" name="emails"
            value="">
          <div class="input-group">
            <label for="nodeBattery">Delay between Alerts :</label>
            <div class="input-group">
              <input type="number" class="form-control" id="delay"
                name="Delay" value="">
              <div class="input-group-append">
                <span class="input-group-text">Sec</span>
              </div>
            </div>
          </div>
          <div class="row justify-content-center">
            <button type="submit" class="btn btn-primary"
              onclick="submitFormEmail()">Submit</button>
          </div>
        </form>
      </div>
    </div>
    <div class="card mb-4" style="height: 506px; width: 100%">
      <div class="card-header">
        <div class="row">
          <div class="col">
            <h5>List of recipient email</h5>
          </div>
        </div>
      </div>
      <div class="card-body" id="allemail">
      </div>
    </div>
  </div>
</div>
</div>
<div class="row mt-4 mt-sm-4 mt-lg-4 ">
  <div class="col-12">
    <div class="card text-center" id="Locations">
      <div class="card-header">
        <h4>Click on the map to mark the destination</h4>
      </div>
      <div class="card-body">
        <div id="map"></div>
      </div>
      <div class="card-footer text-muted"></div>
    </div>
  </div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    </div>
  </div>
</div>
<script> let defaultLatitude,
  defaultLongitude,
  url = "https://api.rungrueng.site",
  map,
  marker,
  markers,
  circle;
function getCookie(name) {
  const cookies = document.cookie.split(";");
  for (let cookie of cookies) {
    const [cookieName, cookieValue] = cookie.split("=");
    if (cookieName.trim() === name) {
      return cookieValue;
    }
  }
  return null;
}
let username = getCookie("username");
function logout() {
  document.cookie = "username=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  document.cookie = "table=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  document.cookie = "id=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  window.location.replace("login.html");
}
document.addEventListener("DOMContentLoaded", function () {
  function capitalizeFirstLetter(string) {
    return string.charAt(0).toUpperCase() + string.slice(1);
  }
  const navbarBrand = document.getElementById("name");
  if (username) {
    navbarBrand.textContent = capitalizeFirstLetter(username);
  }
  if (!username) {
    window.location.replace("login.html");
    return;
  }
});
async function submitForms() {
  var start = document.getElementById("start").value;
  var end = document.getElementById("end").value;
  window.location.href = `results.html?start=${start}&end=${end}`;
}
document.addEventListener("DOMContentLoaded", function () {
  var randomDelay = Math.floor(Math.random() * 1001);
  var styleElement = document.createElement("style");
  styleElement.textContent = `
:root {
--animate-duration: ${randomDelay + 500}ms !important;
--animate-delay: 1s !important;
}
`;
  document.head.appendChild(styleElement);
  setTimeout(function () {
    $("#loadingMessage").hide();
  },

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }, randomDelay);
  });
  async function fetchDataAndUpdateForm() {
  try {
    const response = await fetch(url + "/api/alert/con?user=" + username);
    const data = await response.json();
    const resemail = await fetch(url + "/api/emails?user=" + username);
    const dataemail = await resemail.json();
    const emails = dataemail.Emails;
    const emailContainer = document.getElementById("allemail");
    emails.forEach((email) => {
      const containerDiv = document.createElement("div");
      containerDiv.classList.add("email-container");
      const emailDiv = document.createElement("div");
      emailDiv.textContent = email;
      const deleteButtonDiv = document.createElement("div");
      const deleteButton = document.createElement("button");
      deleteButton.textContent = "X";
      deleteButton.onclick = () => deleteEmail(email);
      deleteButtonDiv.appendChild(deleteButton);
      containerDiv.appendChild(emailDiv);
      containerDiv.appendChild(deleteButtonDiv);
      emailContainer.appendChild(containerDiv);
    });
    document.getElementById("temperature").value = data.Temperature;
    document.getElementById("humidity").value = data.Humidity;
    document.getElementById("speed").value = data.Speed;
    document.getElementById("latitude").value = data.Latitude;
    document.getElementById("longitude").value = data.Longitude;
    defaultLatitude = data.Latitude;
    defaultLongitude = data.Longitude;
    document.getElementById("radius").value = data.Radius;
    document.getElementById("delay").value = dataemail.Delay;
    document.getElementById("nodeBattery").value = data["Node Battery"];
    document.getElementById("gatewayBattery").value = data["Gateway Battery"];
    initializeMap();
  } catch (error) {
    console.error("Error fetching data:", error);
  }
}

  async function submitFormEmail() {
    const email = document.getElementById("Email").value;
    const delay = document.getElementById("delay").value;
    const body = {
      email: email,
      delay: delay,
      user: username,
    };
    const message = `Submitting Form with the following values:
    Recipient Email: ${email}
    Delay between Alerts :${delay}`;
    if (confirm(message)) {
      alert("Form submitted! Email Settings have been changed.");
    } else {
      return;
    }
  }
  const options = {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

method: "POST",
headers: {
  "Content-Type": "application/json",
},
body: JSON.stringify(body),
};
try {
  const response = await fetch(url + "/api/addemail", options);
  if (response.ok) {
    console.log("Form submitted successfully!");
  } else {
    console.log("Error submitting form.");
  }
} catch (error) {
  console.error("Error submitting form:", error);
}
}
async function initializeMap() {
  var defaultLatLng = L.latLng(defaultLatitude, defaultLongitude);
  map = L.map("map").setView([defaultLatitude, defaultLongitude], 15);
  L.tileLayer("https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png", {
    maxZoom: 19,
  }).addTo(map);
  marker = new L.marker(defaultLatLng).addTo(map);
  document.getElementById("latitude").value = defaultLatitude;
  document.getElementById("longitude").value = defaultLongitude;
  var radius = document.getElementById("radius").value;
  circle = L.circle(defaultLatLng, {
    color: "blue",
    fillColor: "blue",
    fillOpacity: 0.1,
    radius: radius * 1000,
  }).addTo(map);
  marker.bindPopup("Destination").openPopup();
  var currentPositionData = await latestpos();
  currentPositionData = JSON.parse(currentPositionData);
  var currentLat = parseFloat(currentPositionData.Latitude);
  var currentLng = parseFloat(currentPositionData.Longitude);
  if (!isNaN(currentLat) && !isNaN(currentLng)) {
    var currentLatLng = L.latLng(currentLat, currentLng);
    var currentMarker = new L.marker(currentLatLng).addTo(map);
    currentMarker.bindPopup("Current Location").openPopup();
    var midpoint = L.latLng([(parseFloat(defaultLatitude) + currentLat) / 2,
      (parseFloat(defaultLongitude) + currentLng) / 2]);
    map.setView(midpoint, 11);
  }
}
map.on("click", function (e) {
  if (marker) {
    map.removeLayer(marker);
  }
  if (circle) {
    map.removeLayer(circle);
  }
  marker = new L.marker(e.latLng).addTo(map);
  document.getElementById("latitude").value = e.latLng.lat;
  document.getElementById("longitude").value = e.latLng.lng;
  var radius = document.getElementById("radius").value;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

circle = L.circle(e.latlng, {
  color: "blue",
  fillColor: "blue",
  fillOpacity: 0.1,
  radius: radius * 1000,
}).addTo(map);
});
}
async function submitForm() {
const temperature = document.getElementById("temperature").value;
const humidity = document.getElementById("humidity").value;
const speed = document.getElementById("speed").value;
const latitude = document.getElementById("latitude").value;
const longitude = document.getElementById("longitude").value;
const radius = document.getElementById("radius").value;
const nodeBattery = document.getElementById("nodeBattery").value;
const gatewayBattery = document.getElementById("gatewayBattery").value;
const body = {
  Temperature: temperature,
  Humidity: humidity,
  Speed: speed,
  Longitude: longitude,
  Latitude: latitude,
  Radius: radius,
  NBattery: nodeBattery,
  GBattery: gatewayBattery,
  user: username,
};
const message = `Submitting Form with the following values:
Temperature: ${temperature} °C
Humidity: ${humidity} %
Speed: ${speed} Km/h
Latitude: ${latitude}
Longitude: ${longitude}
Radius: ${radius} Km
Node Battery: ${nodeBattery} %
Gateway Battery: ${gatewayBattery} %`;
if (confirm(message)) {
  alert("Form submitted! Alert condition have been changed.");
} else {
  return;
}
const options = {
  method: "POST",
  body: JSON.stringify(body),
  headers: {
    "Content-Type": "application/json",
  },
};
};
try {
const response = await fetch(url + "/api/alert/changecon", options);
if (response.ok) {
  console.log("Form submitted successfully!");
} else {
  console.log("Error submitting form.");
}
} catch (error) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    console.error("Error submitting form:", error);
  }
  location.reload();
}
async function deleteEmail(email) {
  try {
    const response = await fetch(url + "/api/deleteemail", {
      method: "POST",
      headers: {
        "Content-Type": "application/json",
      },
      body: JSON.stringify({
        email: email,
        user: username,
      }),
    });
    const result = await response.json();
    const resemail = await fetch(url + "/api/emails?user=" + username);
    const dataemail = await resemail.json();
    const emails = dataemail.Emails;
    const emailContainer = document.getElementById("allemail");
    while (emailContainer.firstChild) {
      emailContainer.removeChild(emailContainer.firstChild);
    }
    emails.forEach((email) => {
      const containerDiv = document.createElement("div");
      containerDiv.classList.add("email-container");
      const emailDiv = document.createElement("div");
      emailDiv.textContent = email;
      const deleteButtonDiv = document.createElement("div");
      const deleteButton = document.createElement("button");
      deleteButton.textContent = "X";
      deleteButton.onclick = () => deleteEmail(email);
      deleteButtonDiv.appendChild(deleteButton);
      containerDiv.appendChild(emailDiv);
      containerDiv.appendChild(deleteButtonDiv);
      emailContainer.appendChild(containerDiv);
    });
  } catch (error) {
    console.error("Error deleting email:", error);
  }
}
function resetForm() {
  document.getElementById("temperature").value = 40;
  document.getElementById("humidity").value = 80;
  document.getElementById("speed").value = 100;
  document.getElementById("latitude").value = 13.783653;
  document.getElementById("longitude").value = 100.634766;
  document.getElementById("radius").value = 2;
  document.getElementById("nodeBattery").value = 50;
  document.getElementById("gatewayBattery").value = 50;
  submitForm();
}
async function latestpos() {
  try {
    const response = await fetch(url + "/api/latest?user=" + username, {
      method: "GET",

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
headers: {  
  "Content-Type": "application/json",  
},  
});  
const result = await response.json();  
return JSON.stringify(result);  
} catch (error) {  
  console.error("Error deleting email:", error);  
}  
}  
fetchDataAndUpdateForm();  
</script>  
</body>  
</html>
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Admin Page</title>
  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <style>
    .center-content {
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
    }
    body {
      background: radial-gradient(#343333, #101010) !important;
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="row">
      <div class="col-md-6 offset-md-3 center-content">
        <div class="card">
          <div class="card-body">
            <h1 class="card-title text-center">Welcome to the Admin Page</h1>
            <div id="userList" class="mt-4 text-center"></div>
            <form id="addUserForm" class="mt-4">
              <div class="form-group">
                <label for="username">Username</label>
                <input type="text" class="form-control" id="username" required>
              </div>
              <div class="form-group">
                <label for="password">Password</label>
                <input type="password" class="form-control" id="password"
required>
              </div>
              <div class="form-group">
                <label for="table">Table Name</label>
                <input type="text" class="form-control" id="table" required>
              </div>
              <button type="submit" class="btn btn-primary btn-block">Add
User</button>
              <button class="btn btn-danger btn-block mt-4"
onclick="logout()">Logout</button>
            </form>
          </div>
        </div>
      </div>
    </div>
  </div>
</body>
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.4/dist/umd/popper.min.js"></scrip

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

t>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
<script>
function loginAsUser(username) {
  const expirationDate = new Date(Date.now() + 10 * 60 * 1000);
  document.cookie = `username=${username}`;
  expires=${expirationDate.toUTCString()}; path=/;`;
  window.open("index.html", "_blank");
}
async function fetchUserData() {
  try {
    const response = await fetch("http://rung.ddns.net:8050/api/user");
    if (!response.ok) {
      throw new Error("Failed to fetch user data.");
    }
    const userList = await response.json();
    const userListDiv = document.getElementById("userList");
    userListDiv.innerHTML = "";
    userList.forEach(user => {
      if (user !== "admin") {
        const button = document.createElement("button");
        button.textContent = `Login as ${user}`;
        button.className = "btn btn-primary mr-2 mt-2";
        button.onclick = () => loginAsUser(user);
        userListDiv.appendChild(button);
      }
    });
  } catch (error) {
    console.error(error);
  }
}
async function addUser(username, password, table) {
  try {
    const response = await
    fetch(`http://rung.ddns.net:8050/add_user?username=${username}&password=${password}&table=${table}`, {
      method: "GET"
    });
    if (!response.ok) {
      throw new Error("Failed to add user.");
    }
    alert("User added successfully.");
    fetchUserData();
  } catch (error) {
    console.error(error);
    alert("Failed to add user.");
  }
}
document.getElementById("addUserForm").addEventListener("submit", function
(event) {
  event.preventDefault();
  const username = document.getElementById("username").value;
  const password = document.getElementById("password").value;
  const table = document.getElementById("table").value;
  addUser(username, password, table);
});

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

window.onload = function () {
  const cookies = document.cookie.split(";").map(cookie =>
    cookie.trim().split("="));
  const idCookie = cookies.find(cookie => cookie[0] === "id");
  const tableCookie = cookies.find(cookie => cookie[0] === "table");
  const usernameCookie = cookies.find(cookie => cookie[0] === "username");
  if (idCookie && idCookie[1] === "1" && tableCookie && tableCookie[1] ===
    "admin" && usernameCookie && usernameCookie[1] === "admin") {
    fetchUserData();
  } else {
    alert("Unauthorized access. Redirecting to index page.");
    window.location.href = "login.html";
  }
}
function logout() {
  document.cookie = "username=; expires=Thu, 01 Jan 1970 00:00:00 UTC;
  path=/;";
  document.cookie = "table=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  document.cookie = "id=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
  window.location.replace("login.html");
}
</script>
</body>
</html>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้