



โครงการวิศวกรรมคอมพิวเตอร์

ฐานที่พ้อจฉริยะ

Intelligent Military Base

นายสหัสส์ พวงเรืองศรี

นายปุณณพัฒน์ มนต์มี

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ศรีราชา

มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา

ปีการศึกษา 2563



ใบรับรองโครงการวิศวกรรมคอมพิวเตอร์

ปริญญาวิศวกรรมศาสตรบัณฑิต (วิศวกรรมคอมพิวเตอร์และสารสนเทศศาสตร์)

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ศรีราชา

เรื่อง ฐานทัพอัจฉริยะ

Intelligent Military Base

นามผู้จัดทำ นายสหัสส พวงเรืองศรี

นายปัญญาพัฒน์ มนต์มี

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาโครงการ

(อาจารย์ กรวิทย์ ออกผล)

กรรมการโครงการ

(อาจารย์ ไพรัช สร้อยทอง)

กรรมการโครงการ

(อาจารย์ เพ็ญพรรณ ใจสวัสดิ์)

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ศรีราชา

มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา รับรองแล้ว

(รศ.ดร.อนันต์บรรหารสกุล)

หัวหน้าภาควิชาวิศวกรรมคอมพิวเตอร์

โครงการวิศวกรรมคอมพิวเตอร์
เรื่อง

ฐานที่พอจฉริยะ

Intelligent Military Base

โดย

นาย สหัสส พวงเรืองศรี 6030301055

นาย บุญญพัฒน์ มนต์มี 6030300644

เสนอ

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ศรีราชา

มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา

ปีการศึกษา 2563

นายสหัสส พวงเรืองศรี \ นายปุณณพัฒน์ มันต์มี \ 2563 \ ฐานทัพอจฉริยะ \
ปริญญาวิศวกรรมศาสตรบัณฑิต (วิศวกรรมคอมพิวเตอร์และสารสนเทศศาสตร์) \
ภาควิชาวิศวกรรมคอมพิวเตอร์ \ คณะวิศวกรรมศาสตร์ศรีราชา

บทคัดย่อ

โครงการนี้มีวัตถุประสงค์เพื่อเพิ่มประสิทธิภาพสำหรับเจ้าหน้าที่ทหารเพื่อการรับรู้เหตุการณ์หรือการเคลื่อนไหวที่เกิดขึ้นบริเวณรอบๆ ของฐานทัพในระยะ 10 Km โดยการตรวจจับในรูปแบบต่างๆ เช่น การตรวจจับความร้อนจาก การเคลื่อนไหวโดยใช้ PIR Sensor, Raspberry pi Raspberry Pi 8MP Camera Module V2, การตรวจจับสัญญาณ Wi-Fi (Wi-Fi sniffer), การตรวจจับหมายเลขเครื่องของโทรศัพท์มือถือและหมายเลขซิมของโทรศัพท์มือถือ (Mobile Detection สำหรับการพัฒนาต่อในอนาคต) ซึ่งจะมีการแจ้งเตือนผ่านทาง Web application และ mobile application ในรูปแบบต่างๆ เช่น เสียง โดยจากการทดลองสามารถรับรู้ได้ทันเวลา โดยการแจ้งเตือนจากเสียง และจะมีการส่งไฟล์ภาพและไฟล์เสียงมาเก็บไว้ที่ Server

Sahawat Poungruangsri \ Punyaphat Monmee \ 2020 \ Intelligent Military Base \ Bachelor of Engineering (Computer Engineering and Informatics) \ Department of Computer Engineering \ Faculty of Engineering at Siracha

Abstract

The objective of this project is to increase efficiency for military officer to recognize events or movements occurring in the area surrounding the base at 10 Km. By detecting with various methods such as motion heat detection using PIR Sensor, Raspberry pi Raspberry Pi 8 MP Camera Module V2, Wi-Fi signal detection (Wi-Fi sniffer), mobile phone serial number detection and mobile phone sim number (mobile detection for future development) by transmission notifications through web application and mobile application in various forms such as sound, images Which can be recognize timely from the experiment by the notification of the sound and will transmission image and audio files to the server

กิตติกรรมประกาศ

การดำเนินโครงการ “ฐานทัพอัจฉริยะ” จะไม่สามารถสำเร็จลุล่วงได้ด้วยดี หากขาดการสนับสนุน และกำลังจากหลาย ๆ ฝ่าย อาทิ

ผศ.ดร. ประวิทย์ ชุมชู และ Prof. Kailas Patil อาจารย์ที่ปรึกษาโครงการที่ได้ให้คำปรึกษา แนะนำ แนวคิด และแนวทางในการแก้ไขปัญหาเมื่อเกิดข้อผิดพลาดขึ้นในการพัฒนาโครงการรวมถึงทฤษฎีต่างๆ ที่เกี่ยวข้อง กับโครงการนอกจากนี้ยังคงอยู่ติดตามความคืบหน้าโครงการอย่างสม่ำเสมอ รวมถึงการให้ข้อมูลความรู้เพื่อนำไปใช้ในการพัฒนาโครงการ

ผศ. เพ็ญพรรณ ใช้ชุดเจริญ สำหรับ คำแนะนำต่างๆ เกี่ยวกับปัญหาที่เจอ กำลังใจในการทำงาน

ขอขอบคุณครอบครัว ที่คอยเป็นกำลังใจ และแรงผลักดัน รวมถึงการสนับสนุนการทำสหกิจครั้งนี้ ผู้จัดทำขอพระขอบพระคุณทุกท่านเป็นอย่างสูง ณ ที่นี่ที่ได้ช่วยให้การดำเนินโครงการในครั้งนี้สำเร็จลุล่วงไปได้ด้วยดีและเปิดโอกาสให้ทีมของผู้พัฒนาได้มีโอกาสเรียนรู้ในการทำโครงการนี้

นายสหัสส์ พากเรืองศรี

นายปุณณพัฒน์ มนตรี

พฤษภาคม 2563

สารบัญ

ใบรับรองโครงการวิศวกรรมคอมพิวเตอร์	ก
โครงการวิศวกรรมคอมพิวเตอร์.....	ข
บทคัดย่อ.....	ค
Abstract	ง
กิตติกรรมประกาศ	จ
บทที่ 1	1
บทนำ.....	1
1. คำสำคัญ (Key Words).....	1
2. หลักการและเหตุผล.....	1
3. วัตถุประสงค์.....	1
4. ปัญหาหรือประโยชน์เป็นเหตุผลให้พัฒนาระบบ	2
5. เป้าหมายและขอบเขตของโครงการ.....	2
6. กลุ่มผู้ใช้งาน	2
7. ประโยชน์ที่ได้รับ	2
8. แผนการดำเนินงาน.....	3
บทที่ 2	4
ความรู้พื้นฐานและงานวิจัยที่เกี่ยวข้อง	4
1. งานที่เกี่ยวข้อง.....	4
2. ความรู้พื้นฐาน เทคนิค เทคโนโลยีและเครื่องมือที่ใช้.....	7
บทที่ 3	16
อุปกรณ์และวิธีการ.....	16
1. เนื้อเรื่องย่อ.....	16
2. รายละเอียดวิธีการ.....	17

บทที่ 4	24
ผลและวิจารณ์	24
1. ผลการสอบระบบโดยรวม	24
2. ผลการทดสอบระบบในส่วนย่อย	24
3. ผลการทดสอบจากผู้ใช้งานจริง	28
บทที่ 5	29
สรุปและข้อเสนอแนะ	29
ภาคผนวก	30
ก1	31
คู่มือการติดตั้งระบบ	31
ก2	64
คู่มือการใช้งาน	64
เอกสารอ้างอิง	90

สารบัญตาราง

ตารางที่ 1 แผ่นการดำเนินงาน.....	3
----------------------------------	---

สารบัญภาพ

รูปภาพ 1 ภาพประกอบ INTRUDER ALERT AND SECURITY SYSTEM	4
รูปภาพ 2 ภาพประกอบ Home Security System with Text Alert and Motion Detective.....	5
รูปภาพ 3 VS Code	7
รูปภาพ 4 Python.....	7
รูปภาพ 5 C Language.....	8
รูปภาพ 6 Node JS.....	8
รูปภาพ 7 JavaScript.....	9
รูปภาพ 8 Express Framework.....	9
รูปภาพ 9 PostgreSQL	10
รูปภาพ 10 Flutter	10
รูปภาพ 11 Dart	11
รูปภาพ 12 STM32 B-L072Z-LRWAN1.....	11
รูปภาพ 13 Raspberry pi4.....	12
รูปภาพ 14 PIR sensor.....	12
รูปภาพ 15 Raspberry Pi 8MP Camera Module V2.....	13
รูปภาพ 16 Wireless USB adapter.....	13
รูปภาพ 17 Chirpstack.....	14
รูปภาพ 18 MQTT.....	14
รูปภาพ 19 LoRaWAN	15
รูปภาพ 20 flowchart การทำงานของ Web application Mobile application	17
รูปภาพ 21 flowchart การส่งข้อมูล	19
รูปภาพ 22 Timing Diagram การส่งข้อมูล.....	21
รูปภาพ 23 Inside	22
รูปภาพ 24 Out side	23
รูปภาพ 25 ส่ง 613 เฟรม ได้รับ 612 เฟรม.....	25

รูปภาพ 26 ส่ง 613 เฟรม ได้รับ 608 เฟรม	25
รูปภาพ 27 ส่ง 613 เฟรม ได้รับ 608 เฟรม.....	26
รูปภาพ 28 ส่ง 613 เฟรม ได้รับ 611 เฟรม.....	26
รูปภาพ 29 ส่ง 613 เฟรม ได้รับ 612 เฟรม.....	26
รูปภาพ 30 ส่ง 613 เฟรม ได้รับ 612 เฟรม.....	27
รูปภาพ 31 ส่ง 613 เฟรม ได้รับ 612 เฟรม.....	27
รูปภาพ 32 ส่ง 613 เฟรม ได้รับ 611 เฟรม.....	27
รูปภาพ 33 ส่ง 613 เฟรม ได้รับ 612 เฟรม.....	28
รูปภาพ 34 ส่ง 613 เฟรม ได้รับ 611 เฟรม.....	28
รูปภาพ 35 Download Visual Studio	31
รูปภาพ 36 เปิดโปรแกรม	31
รูปภาพ 37 Run ไฟล์ .exe	32
รูปภาพ 38 กด Next	32
รูปภาพ 39 เลือกพื้นที่จัดเก็บ.....	33
รูปภาพ 40 กด Next	33
รูปภาพ 41 Create a desktop icon และ Add to PATH.....	34
รูปภาพ 42 ติดตั้งโปรแกรม.....	34
รูปภาพ 43 เสร็จสิ้นการติดตั้งโปรแกรม Visual Studio	35
รูปภาพ 44 ติดตั้ง Git โดยดาวโหลดจาก https://git-scm.com/downloads	35
รูปภาพ 45 ใช้ Git clone โปรเจค flutter	35
รูปภาพ 46 ตั้งค่า Path ให้ไฟล์เดอร์ Flutter	36
4. อัปเกรด Flutter อัพเดทเวอชั่นของ Flutter โดยใช้ command บน CMD รูปภาพ 47 อัปเกรด Flutter command flutter doctor	36
รูปภาพ 48 ตั้งค่า Emulator ใน Android Studio	37
รูปภาพ 49 ตั้งค่า Emulator ใน Android Studio	37
รูปภาพ 50 create อุปกรณ์	38
รูปภาพ 51 ทำการเลือกชนิดของอุปกรณ์	38
รูปภาพ 52 เลือก System image.....	39
รูปภาพ 53 ติดตั้ง OS Linux version 18.04.5.....	42

รูปภาพ 54 ติดตั้ง Chirpstack.....	42
รูปภาพ 55 ใช้คำสั่ง git clone https://github.com/brocaar/chirpstack-docker.git	43
รูปภาพ 56 ทำการติดตั้ง Compose	43
รูปภาพ 57 ติดตั้ง Docker Engine.....	44
รูปภาพ 58 ใช้คำสั่ง sudo docker-compose up	44
รูปภาพ 59 ตั้งค่าโดยใช้คำสั่ง vi chirpstack-docker/configuration/chirpstack-network-server/chirpstack-network-server.toml.....	45
รูปภาพ 60 ตั้งค่าความถี่	45
รูปภาพ 61 login เพื่อเข้าไปตั้งค่าต่างๆ	46
รูปภาพ 62 ตั้งค่าในส่วนต่างๆ ภายใน Web application ของ Chirp stack.....	46
รูปภาพ 63 กด ADD เพื่อทำการเพิ่ม Network-server	47
รูปภาพ 64 กด UPDATE NETWORK-SERVER หลังจากใส่ข้อมูลครบ	47
รูปภาพ 65 สร้าง Gateway-profiles.....	48
รูปภาพ 66 กด UPDATE GATEWAY-PROFILE หลังจากใส่ข้อมูลครบ	48
รูปภาพ 67 สร้าง Service-profiles.....	49
รูปภาพ 68 กด CREATE SERVICE- PROFILE เมื่อใส่ข้อมูลครบ	49
รูปภาพ 69 กดสร้าง Device-profiles.....	50
รูปภาพ 70 กด UPDATE DEVICE-PROFILE เมื่อใส่ข้อมูลครบ.....	50
รูปภาพ 71 กด UPDATE DEVICE-PROFILE เมื่อใส่ข้อมูลครบ.....	51
รูปภาพ 72 กด UPDATE DEVICE-PROFILE เมื่อใส่ข้อมูลครบ.....	51
รูปภาพ 73 กด UPDATE DEVICE-PROFILE เมื่อใส่ข้อมูลครบ.....	52
รูปภาพ 74 กดสร้าง Gateways	52
รูปภาพ 75 กดอนุญาตให้เข้าถึงตำแหน่ง	53
รูปภาพ 76 ใส่ข้อมูลให้ครบ	53
รูปภาพ 77 กด CREATE GATEWAY เมื่อใส่ข้อมูลครบ	54
รูปภาพ 78 กดสร้าง Application	54
รูปภาพ 79 กด CREATE APPLICATION เมื่อใส่ข้อมูลครบ	55
รูปภาพ 80 กดสร้าง Device	55
รูปภาพ 81 กด CREATE DEVICE เมื่อใส่ข้อมูลครบ	56

รูปภาพ 82 กด Device ที่สร้างไว้	56
รูปภาพ 83 กด (RE)ACTIVATE DEVICE เมื่อใส่ข้อมูลครบ	57
รูปภาพ 84 Setup Kerlink Wirnet Station Gateway.....	57
รูปภาพ 85 ตั้งค่า IP Address ให้เป็นแบบ Static โดยเข้าผ่าน SSH.....	58
รูปภาพ 86 ตั้งค่า Gateway ID ให้ตรงกับบน Web application ของ Chirpstack	58
รูปภาพ 87 ตั้งค่าความถี่ให้ตรงกับ Web application ของ Chirpstack	59
รูปภาพ 88 คำสั่งดู Log เมื่อมีค่าส่งผ่าน Kerlink Wirnet Station Gateway.....	59
รูปภาพ 89 Setup B-L072Z-LRWAN1 Device เพื่อที่จะให้อุปกรณ์สื่อสารกับ Chirpstack application server	60
รูปภาพ 90 เพิ่ม Device	61
รูปภาพ 91 เพิ่มค่า Network Session and Application Session key	62
รูปภาพ 92 ตั้งค่าความถี่ใหม่เมื่อกัน.....	63
รูปภาพ 93 หน้า login	64
รูปภาพ 94 หน้าแดรกของ Admin และ Supervisor	64
รูปภาพ 95 เพิ่ม user	65
รูปภาพ 96 เพิ่ม device	65
รูปภาพ 97 เพิ่มกล้อง CCTV	66
รูปภาพ 98 หน้า home ของ Admin และ Supervisor	66
รูปภาพ 99 แก้ไข กล้อง CCTV	67
รูปภาพ 100 หน้าโชว์ log ของทุกคน	67
รูปภาพ 101 หน้า แก้ไข log ของ Admin และ Supervisor.....	68
รูปภาพ 102 หน้าโชว์ Base ของ Admin และ Supervisor.....	68
รูปภาพ 103 เพิ่มสถานที่	69
รูปภาพ 104 แก้ไขสถานที่	69
รูปภาพ 105 หน้าโชว์ Device ทุกดัว	70
รูปภาพ 106 แก้ไข Device.....	70
รูปภาพ 107 log in email ระดับ Leader.....	71
รูปภาพ 108 Add user ของ Leader.....	71
รูปภาพ 109 เพิ่ม Device ของ leader.....	72

รูปภาพ 110 เพิ่มกล้องของ Leader	72
รูปภาพ 111 หน้า Home ของ Leader	73
รูปภาพ 112 เพิ่ม กล้อง Cctv ของ Leader	73
รูปภาพ 113 หน้า ดู log ของ ระดับ Leader.....	74
รูปภาพ 114 หน้าโผร์ email ของ Leader.....	74
รูปภาพ 115 หน้าจัดการ Device ของ Leader.....	75
รูปภาพ 116 หน้าแก้ไข Device ของ Leader	75
รูปภาพ 117 log in ระดับ Soldier	76
รูปภาพ 118 หน้าเพิ่ม Device ของ Soldier	76
รูปภาพ 119 หน้าเพิ่มกล้อง Cctv ของ Soldier	77
รูปภาพ 120 หน้า home ของ Soldier.....	77
รูปภาพ 121 สามารถดูกล้องแบบขยายให้ได้	78
รูปภาพ 122 ดูข้อมูลของอุปกรณ์.....	78
รูปภาพ 123 โชว์การแจ้งเตือน เมื่อมีคนผ่านที่ PIR.....	79
รูปภาพ 124 หน้า Login สำหรับ mobile application.....	80
รูปภาพ 125 การแจ้งเตือนเมื่อ Login สำเร็จ	80
รูปภาพ 126 การแจ้งเตือนเมื่อ Login ไม่สำเร็จ	81
รูปภาพ 127 หน้าแรกสำหรับแสดงกล้อง CCTV	81
รูปภาพ 128 ในส่วนของการ เพิ่ม, ลบ, แสดง log ของ User จะสามารถแสดงเฉพาะตำแหน่งที่เท่ากันหรือต่ำกว่า	82
รูปภาพ 129 หน้าแสดงหน้า Profile.....	82
รูปภาพ 130 กรอกลายละเอียดของ CCTV เพื่อเพิ่ม CCTV	83
รูปภาพ 131 กรอกลายละเอียดของ Device เพื่อเพิ่ม Device	83
รูปภาพ 132 แสดง User ทั้งหมดที่อยู่ในระบบ เงื่อนไขในการแสดงขึ้นอยู่กับ ตำแหน่ง	84
รูปภาพ 133 สำหรับ Delete หรือ Edit User.....	84
รูปภาพ 134 แสดงรายละเอียดของ Device	85
รูปภาพ 135 เมื่อเลือกแล้วจะแสดงให้ว่า เปิดหรือปิด	85
รูปภาพ 136 แก้ไขข้อมูล Device	86
รูปภาพ 137 แสดงข้อมูลของฐานแต่ละฐาน.....	86

รูปภาพ 138 ในส่วนของการ เพิ่ม, ลบ, แสดง โลจ ของ User จะสามารถแสดงเฉพาะตำแหน่งที่เท่ากันหรือต่ำกว่า	87
รูปภาพ 139 หน้าแสดงหน้า Profile.....	87
รูปภาพ 140 กรอกลายละเอียดของ CCTV เพื่อเพิ่ม CCTV.....	88
รูปภาพ 141 กรอกลายละเอียดของ Device เพื่อเพิ่ม Device	88
รูปภาพ 142 แสดง Profile และสามารถ เพิ่มอุปกรณ์ และเพิ่ม CCTV ได้.....	89
รูปภาพ 143 หน้าแสดงหน้า Profile.....	89

บทที่ 1

บทนำ

1. คำสำคัญ (Key Words)

- LoRaWAN
- IOT
- MQTT
- Protocol
- STM32

2. หลักการและเหตุผล

เนื่องจากเหตุการณ์ความไม่สงบที่ผ่านมาในอดีตในบริเวณภาคใต้ของประเทศไทยนั้นทำให้เราตระหนักรถึงความปลอดภัยของเจ้าหน้าที่ทางการทหารมากขึ้นโดยส่วนมากสาเหตุที่เกิดเหตุการณ์ความไม่สงบอาจจะเกิดจากความขัดแย้งต่างๆอย่างไรก็ตามการคำนึงถึงความปลอดภัยของเจ้าหน้าที่ทหารนั้นถือเป็นสิ่งความสำคัญมากที่สุด เราจึงมีความคิดที่จะพัฒนาโครงการนี้ไม่เพียงแค่รักษาความปลอดภัยของเจ้าหน้าที่ทางการทหารแต่รวมถึงเพื่อลดการความเสี่ยงและการสูญเสีย เช่น ทรัพยกรรมทางทหารบุคลากรทางทหาร

จากปัญหาที่เกิดขึ้นเจ้าหน้าที่ทางการทหารต้องการความปลอดภัยที่มากยิ่งขึ้น หลายคนอาจจะมองถึงการเพิ่มจำนวนเจ้าหน้าที่ทางการทหารให้มากขึ้นเพื่อเข้ามาเพื่อความปลอดภัยที่มากขึ้น แต่หากมีเทคโนโลยีที่สามารถตรวจจับในรูปแบบต่างๆได้ในระยะใกล้จะสามารถช่วยให้ไม่ต้องเพิ่มจำนวนอีกทั้งยังลดการสูญเสีย เพราะสามารถรับรู้และมีเวลาเตรียมตัวรับมือมากยิ่งขึ้น

3. วัตถุประสงค์

- 3.1 เพื่อผู้ใช้สามารถส่งรูปภาพได้ในระยะใกล้ (10 Km)
- 3.2 เพื่อผู้ใช้สามารถส่งเสียงได้ในระยะใกล้ (10 Km)
- 3.3 เพื่อผู้ใช้สามารถรับรู้การแจ้งเตือนที่ระยะใกล้ (10km)
- 3.4 เพื่อผู้ใช้สามารถรับรู้ Mac WIFI ที่ระยะใกล้ (10km)

4. ปัญหาหรือประโยชน์เป็นเหตุผลให้พัฒนาระบบ

ปัญหาเกิดจากระยะที่ไกลเกินการรับรู้ของทหาร แต่ถ้ามีอุปกรณ์สำหรับการตรวจจับโดยการติดตั้งไว้ในพื้นที่ที่ไกลๆ (10Km) ก็จะช่วยในการรับรู้และทหารสามารถเตรียมตัวในการยับยั้งเหตุการณ์ความไม่สงบ และเพิ่มประสิทธิภาพระบบรักษาความปลอดภัย

5. เป้าหมายและขอบเขตของโครงการ

- รับรู้การเคลื่อนไหวได้จากระยะไกล
- ส่งไฟล์ภาพและเสียงจากระยะไกล
- สามารถรับค่าจากอุปกรณ์ด้วย ChirpStack
- จับเลข Mac WIFI จากระยะไกล

6. กลุ่มผู้ใช้งาน

เจ้าหน้าที่ทางการทหาร

7. ประโยชน์ที่ได้รับ

สามารถรับรู้การเคลื่อนไหวได้ก่อนและสามารถเตรียมตัวรับมือกับเหตุการณ์ความไม่สงบ

8. แผนการดำเนินงาน

ตารางที่ 1 แผนการดำเนินงาน

ลำดับ	กิจกรรม	ปี พ.ศ. 2563	ปี พ.ศ. 2564					
			ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.
1	เลือกหัวข้อและศึกษาข้อมูลของโครงการ							
2	เรียนรู้การใช้ The Thing network							
3	ศึกษาและติดตั้ง ChirpstackOS และลง Raspberry pi เพื่อทดลองระบบการทำงานและติดตั้งบน Server Ubuntu version 18.04 สำหรับการใช้งานจริง							
4	ศึกษาการใช้งาน LoRaWAN และ Gateway kerlink station ให้สามารถติดต่อกับ Chirpstack							
5	ศึกษาการใช้งาน B-L072Z เพื่อรับส่งข้อมูลจาก Chirpstack server							
6	เรียนรู้การเขียน Mobile applications และ Web applications							
7	ออกแบบ database สำหรับ Mobile applications และ Web applications							
8	เริ่มเขียน Mobile applications และ Web applications							
9	สร้าง API สำหรับ Mobile applications และ Web applications บางส่วน							
10	ซั่งซื้อและทดลอง อุปกรณ์ต่างๆ							
11	เรียนรู้การสร้างรูปภาพและเสียงระหว่าง Raspberry pi กับ STM32 B-L072Z							
12	รวมระบบระหว่าง Web application และ Mobile applications							
13	ออกแบบกล่องอุปกรณ์							
14	ทดสอบการส่งเสียงและรูป							
15	ทดลองระบบที่ติดตั้ง							

บทที่ 2

ความรู้พื้นฐานและงานวิจัยที่เกี่ยวข้อง

1. งานที่เกี่ยวข้อง

1.1. งานที่เกี่ยวข้อง

1.1.1. INTRUDER ALERT AND SECURITY SYSTEM

ระบบรักษาความปลอดภัยมีความสำคัญมากในสังคมปัจจุบัน ด้วยความก้าวหน้าทางเทคโนโลยีทำให้แต่ละคนไม่ต้องกังวลกับการรักษาความปลอดภัยให้กับบ้านหรือทรัพย์สินของตน เป้าหมายของโครงการคือการติดตั้งอุปกรณ์รักษาความปลอดภัยที่ตรวจจับการเคลื่อนไหวของผู้บุกรุกส่งข้อความไปยังเจ้าของและจับภาพของผู้บุกรุก เช่นเซอร์ PIR จะตรวจจับการเคลื่อนไหวโดยการตรวจจับความแตกต่างระหว่างระดับความร้อนอินฟราเรดหรือรังสีที่ตรวจจับโดยวัตถุรอบๆ [1] (ดูรูปภาพที่ 1)

© 2019 IJRAR March 2019, Volume 6, Issue 1 www.ijrar.org (E-ISSN 2248-1269, P-ISSN 2349-5139)
INTRUDER ALERT AND SECURITY SYSTEM
Dr. Y. Ramkrishna¹, D. Leelavathi², V.S. Sreenivas³, S. Sai kishore⁴
Gudlavalluru Engineering College, India

Abstract: Security systems are very important in present Society as there is a increase in criminal activities every day. With the technological advancements a individual doesn't have to worry about providing a security to his/her home or property. The aim of project is implemented to develop the system to detect the intruder by using motion sensor and send message to owner and capture image of intruder.

PIR sensor finds the motion by using the difference between the infrared or radiation level emitted by the surrounding objects. The PIR sensor outputs goes high if detects any motion, then the sensor sends a high pulse to camera shutter module. By using camera shutter module the wireless camera captures the image of intruder. GSM sends a message to the owner that intruder alert.

KEY WORDS: Arduino Uno, PIR Sensor, GSM SIM900a, Bluetooth camera shutter, Buzzer.

L. INTRODUCTION

The project focuses on Intruder alert and security system and it is designed to provide security for home/office/private property. The recent activities of robbers by shutting down the power devised us to develop a wireless surveillance using battery backup.

The main aim of project is to develop a security system to detect intruder and send a message to owner. The Sensor to detect the motion and send a signal to Arduino Uno. Arduino Uno receives the signal from motion sensor and it detects the motion and sends a high pulse to the Arduino Uno. Arduino Uno sends a High pulse to Camera shutter module and GSM module, then the camera shutter module captures the image of intruder and GSM sends a message to owner as Intruder breach.

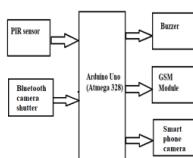


Fig. No. 1 : Block Diagram

2. Design Components :

A) **Arduino Uno :**
Arduino Uno is an open source proto-type platform based on easy to use hardware and software. It doesn't require additional software or hardware for programming board. It consists of 14 digital input/output pins out of which 6 pins can be used for PWM pins, 6 analog pins, reset button and ICSB header.



Fig. No. 2 : Arduino Uno

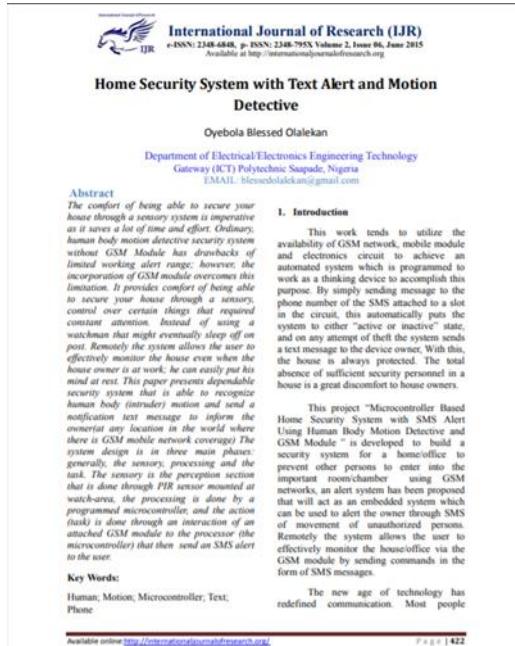
B) **PIR Sensor :**
PIR Sensor is a Passive infrared sensor which can be used to motion by using difference between radiation or infrared.

IJRAR19J4167 | International Journal of Research and Analytical Reviews (IJRAR) www.ijrar.org | 469

รูปภาพ 1 ภาพประกอบ INTRUDER ALERT AND SECURITY SYSTEM

1.1.2. Home Security System with Text Alert and Motion Detective

บทความนี้นำเสนอรหบบรักษาความปลอดภัยที่ใช้อีเมลได้ซึ่งสามารถรับการเคลื่อนไหวของร่างกายมนุษย์ (ผู้บุกรุก) และส่งข้อความแจ้งเตือนเพื่อแจ้งให้เจ้าของทราบ (ณ ที่ใด ๆ ในโลกที่มีเครือข่ายมือถือ GSM ครอบคลุม) [2] (ดูรูปภาพที่ 2)



รูปภาพ 2 ภาพประกอบ Home Security System with Text Alert and Motion Detective

1.1.3. Development of Cloud Integrated Internet of Things Based Intruder Detection System

บทความนี้มุ่งเน้นไปที่การสร้างระบบรักษาความปลอดภัยภายในบ้านแบบไร้สายอัจฉริยะซึ่งจะส่งการแจ้งเตือนไปยังเจ้าของโดยใช้อินเทอร์เน็ตในการณ์ที่มีการบุกรุกและส่งเสียงเตือนเป็นทางเลือกปัจจุบันการรักษาความปลอดภัยและความปลอดภัยเป็นที่นิยมมากขึ้น ระบบที่นำเสนอมีความสามารถในการปรึกษาตัวของมนุษย์ด้วยเซ็นเซอร์ PIR ซึ่งจะแจ้งให้ผู้ใช้ทราบผ่านวิดีโอที่ถ่ายด้วยกล้องและส่งไปยังแอปพลิเคชัน Android โดยใช้อินเทอร์เน็ตจากนั้นยังส่งข้อความผ่าน GSM

1.1.4. Design and implementation of embedded network video monitoring terminal with SMS alert system

จุดมุ่งหมายของโครงการของเราคือการพัฒนาเทอร์มินัลการตรวจสอบวิดีโอเครือข่ายแบบฝังซึ้งเป็นระบบการส่งวิดีโอที่คุ้มค่าและมีประสิทธิภาพ ระบบตรวจสอบวิดีโอเครือข่ายดิจิทัลซึ่งมีข้อดีของ การควบคุมระยะทางที่ไม่จำกัด การขยายที่ยืดหยุ่นและอื่น ๆ กำลังกลายเป็นมาตรฐานใหม่ของ ระบบรักษาความปลอดภัย เนื่องจากข้อจำกัด ของแบบวิดีโอเครือข่ายสัญญาณวิดีโอที่รวม จะต้องได้รับการบีบอัดเพื่อให้สามารถรับส่งวิดีโอบนเครือข่ายได้ เพื่อรักษาคุณภาพ ของสัญญาณบีบอัดและเพื่อลดจำนวนข้อมูลให้เหลือน้อยที่สุด วิดีโอที่ถ่ายผ่านกล้องจะถูก ประมวลผลและส่งไปยังอินเทอร์เน็ต เช็นเซอร์ PIR ใช้เพื่อตรวจจับการเคลื่อนไหวที่อยู่ด้านหน้าของ กล้องและเรียกใช้โมเด็ม GSM เพื่อส่งการแจ้งเตือนไปยังผู้ใช้ที่ได้รับอนุญาต

1.1.5. Development of PIR sensor based indoor location detection system for smart home

การตระหนักถึงบ้านอัจฉริยะคือวิธีตรวจจับตำแหน่งของผู้อยู่อาศัยปัจจุบันความพยายามในการวิจัย มุ่งเน้นไปที่สองแนวทาง: วิธีการที่ใช้เทอร์มินัลและไม่ใช้เทอร์มินัล วิธีการที่ใช้เทอร์มินัลใช้อุปกรณ์ ประเภทหนึ่งที่ผู้อยู่อาศัยควรนำติดตัวไปในขณะที่วิธีการที่ไม่ใช้เทอร์มินัลไม่จำเป็นต้องใช้อุปกรณ์ ดังกล่าว บทความนี้นำเสนอแนวทางใหม่ที่ไม่ใช้เทอร์มินัลโดยใช้เช็นเซอร์อินฟราเรดไฟโรอิเล็กทริก (เช็นเซอร์ PIR) ที่สามารถตรวจจับผู้อยู่อาศัยได้

1.1.6. Low cost smart security camera with night vision capability using Raspberry Pi and OpenCV

ระบบตรวจสอบโทรทัศน์วงจรปิดภายในบ้าน เป็นอุปกรณ์ที่ขาดไม่ได้ในสังคมปัจจุบันซึ่งเปอร์มาრ์เก็ต โรงงานโรงพยาบาลโรงพยาบาลและ บริษัท ต่างๆ มีระบบกล้องวงจรปิดของตัวเองสำหรับการ ตรวจสอบตลอด 24 ชั่วโมงทุกวัน ให้การตรวจสอบแบบเรียลไทม์ให้ภาพการเฝ้าระวังและอนุญาต ให้เจ้าหน้าที่มีหลักฐานต่อต้านกิจกรรมที่ผิดกฎหมาย นักวิจัยได้พัฒนากล้องวงจรปิดแบบแอกทีฟที่ มีความสามารถในการระบุบริบทของภาพที่กำลังเฝ้าติดตามและสามารถแจ้งเตือนหรือแจ้งเตือนเมื่อ เหตุการณ์เกิดขึ้น

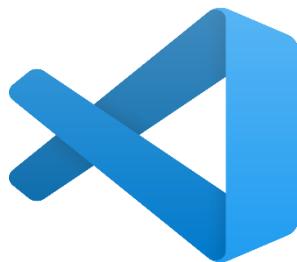
2. ความรู้พื้นฐาน เทคนิค เทคโนโลยีและเครื่องมือที่ใช้

แบ่งส่วนความรู้พื้นฐาน 3 ส่วน คือความรู้พื้นฐานเกี่ยวกับโปรแกรมที่ใช้และความรู้พื้นฐานเกี่ยวกับ อุปกรณ์และความรู้พื้นฐานเกี่ยวกับเครื่อข่าย

ส่วนที่ 1 ความรู้พื้นฐานเกี่ยวกับโปรแกรม

1. Visual Studio Code

เป็นโปรแกรมประเภท Editor ใช้ในการแก้ไขโค้ดที่มีขนาดเล็ก แต่มีประสิทธิภาพสูง เป็น Open Source โปรแกรมจึงสามารถนำมาใช้งานได้โดยไม่มีค่าใช้จ่าย เหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานหลายแพลตฟอร์ม รองรับการใช้งานทั้งบน Windows , macOS และ Linux [3]



รูปภาพ 3 VS Code

2. Python Language

ภาษาโปรแกรม Python คือภาษาโปรแกรมคอมพิวเตอร์ระดับสูง โดยถูกออกแบบมาให้เป็นภาษาสคริปต์ที่อ่านง่าย โดยตัดความซับซ้อนของโครงสร้างและไวยกรณ์ของภาษาออกไป ในส่วนของการแปลงชุดคำสั่งที่เราเขียนให้เป็นภาษาเครื่อง [4]



รูปภาพ 4 Python

3. C Language

ภาษาคอมพิวเตอร์ใช้สำหรับพัฒนาโปรแกรมทั่วไป ถูกพัฒนาครั้งแรกเพื่อใช้เป็นภาษาสำหรับพัฒนาระบบปฏิบัติการยูนิกซ์ (Unix Operating System) แทนภาษาแอสเซมบลี [5]



รูปภาพ 5 C Language

4. Node.js

Cross Platform Runtime Environment สำหรับฝั่ง Server และเป็น Open Source ซึ่งเขียนด้วยภาษา JavaScript สรุปรวมๆ NodeJS ก็คือ Platform ตัวหนึ่งที่เขียนด้วย JavaScript สำหรับเป็น Web Server [6]



รูปภาพ 6 Node JS

5. JavaScript

JavaScript คือ เป็นภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต ซึ่งเป็นภาษาสคริปต์เชิงวัตถุ สามารถใช้ร่วมกับภาษา HTML เพื่อการสร้างและพัฒนาเว็บไซต์ ทำให้เว็บไซต์มีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะการเปลี่ยนความและดำเนินงานไปทีละคำสั่ง เรียกว่า (Object Oriented Programming) [7]

JavaScript

รูปภาพ 7 JavaScript

6. Express Framework

Express Framework คือ web Application Framework ชื่อดังที่ได้รับความนิยมมาก สำหรับทำงานบน platform ของ Node.js ซึ่งเป็น Server ตัวหนึ่ง โดยทั้ง Express.js และ Node.js ต่างก็ใช้ภาษา Java Script ในการพัฒนา [8]



รูปภาพ 8 Express Framework

7. PostgreSQL

เป็นระบบการจัดการฐานข้อมูลเชิงวัตถุ-สัมพันธ์ (object-relational) แบบ ORDBMS โดยสามารถใช้รูปแบบคำสั่งของภาษา SQL ได้เกือบทั้งหมด [9]



รูปภาพ 9 PostgreSQL

8. Flutter

Framework ที่ใช้สร้าง UI สำหรับ mobile application ที่สามารถทำงานข้ามแพลตฟอร์มได้ทั้ง iOS และ Android [10]



รูปภาพ 10 Flutter

9. Dart

Dart นั้นเป็นภาษาโปรแกรมที่เอาไว้สำหรับสร้างแอพพลิเคชันบนแพลตฟอร์มที่หลากหลายโดยได้ทั้ง mobile, desktop, server และก็ web สิ่งที่เป็นที่นิยมที่สุดที่ทำให้คนสนใจมาเรียนภาษา Dart กัน ก็คือเพื่อที่จะเอาไปใช้ร่วมกับ Flutter [11]

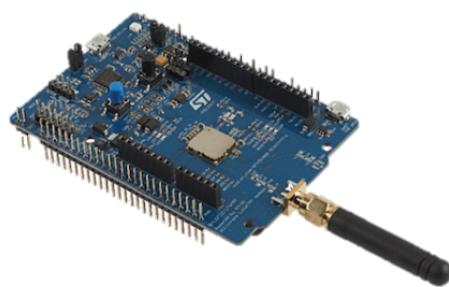


รูปภาพ 11 Dart

ส่วนที่ 2 ความรู้พื้นฐานเกี่ยวกับอุปกรณ์

10. STM32 B-L072Z-LRWAN1

เป็นเครื่องมือพัฒนาสำหรับการเรียนรู้และพัฒนาโซลูชันพื้นฐานเทคโนโลยี LoRa™ และ/หรือ FSK/OOK เครื่องมือนี้นำเสนอโซลูชันโมดูลอเนกประสงค์แบบเปิด (โดย Murata) เพื่อแก้ปัญหา เครือข่ายที่ครอบคลุมพื้นที่กว้างแต่ใช้พลังงานต่ำ (Low-Power Wide Area Network - LPWAN) และรองรับโปรโตคอลไร้สายระยะไกล LoRaWAN [12]



รูปภาพ 12 STM32 B-L072Z-LRWAN1

11. Raspberry Pi4

บอร์ดคอมพิวเตอร์ขนาดเล็กที่สามารถเชื่อมต่อกับจอมอนิเตอร์ คีย์บอร์ด และมาส์ได้ สามารถนำมาประยุกต์ใช้ในการทำโครงการทางด้านอิเล็กทรอนิกส์ การเขียนโปรแกรม หรือเป็นเครื่องคอมพิวเตอร์ตั้งโต๊ะขนาดเล็ก [13]



รูปภาพ 13 Raspberry pi4

12. Pir sensor

อุปกรณ์ Sensor ชนิดหนึ่งที่ใช้ตรวจจับคลื่นรังสี Infrared ที่แพร่จากมนุษย์ หรือสัตว์ ที่มีการเคลื่อนไหว ทำให้มีการนำเอา PIR มาประยุกต์ใช้งานกันเป็นอย่างมากใช้เพื่อตรวจจับการเคลื่อนไหวของสิ่งมีชีวิต หรือ ตรวจจับการบุกรุกในงานรักษาความปลอดภัย [14]



รูปภาพ 14 PIR sensor

13. Raspberry Pi 8MP Camera Module V2



รูปภาพ 15 Raspberry Pi 8MP Camera Module V2

Raspberry Pi Camera V2 ไม่คลอกล้องเวอร์ชั่นที่ 2 สำหรับใช้งานร่วมกับบอร์ด Raspberry Pi ใช้เซ็นเซอร์ Sony IMX219 และเลนส์แบบพิกเซลฟิกส์ ถ่ายภาพนิ่งได้ความละเอียดสูงสุด 8 ล้านพิกเซล (3280×2464) เชื่อมต่อผ่านพอร์ต CSI ใช้งานร่วมกับ Raspberry Pi [15]

14. Wireless USB adapter



รูปภาพ 16 Wireless USB adapter

อุปกรณ์ usb ที่ใช้เป็นอุปกรณ์ต่อสัญญาณ wireless ให้กับคอมพิวเตอร์พีซีหรือโน๊ตบุ๊ค เป็นอุปกรณ์ขนาดโตกว่าแฟลชไดร์ฟเกินน้อย สามารถเสียบเข้ากับพอร์ท usb ที่คอมพิวเตอร์พีซีหรือน็ตบุ๊คได้เด [16]

15. ChirpStack

ChirpStack เป็น LoRaWAN Network Server stack แบบ open-source เป็น solution ที่ทำงานนอกกรอบและเนื่องจากมีโครงสร้างแบบแยกส่วนจึงรวมเข้ากับโครงสร้างพื้นฐานที่มีอยู่อย่างเรียบร้อย [17]



รูปภาพ 17 Chirpstack

16. MQTT

เป็นโปรโตคอลสำหรับใช้ในสื่อสารข้อมูลระหว่าง Machine to Machine (M2M) ออกแบบมาเพื่อใช้สื่อสารในระบบเครือข่ายที่มีทรัพยากรค่อนข้างจำกัด ใช้งานแบบดิวิตต์ สามารถ publish-subscribe ข้อมูลระหว่าง Device เพื่อสื่อสารกันระหว่างอุปกรณ์ [18]



รูปภาพ 18 MQTT

17. LoRaWAN

LPWAN (Low-Power WAN) คือการวางโครงข่ายไร้สายความเร็วต่ำ แต่กินพลังงานต่ำ มีพื้นที่ครอบคลุมเป็นวงกว้าง ใช้สำหรับอุปกรณ์ IoT ที่ต้องสื่อสารตลอดเวลา แต่ส่งข้อมูลไม่เยอะในแต่ละครั้ง [19]



รูปภาพ 19 LoRaWAN

บทที่ 3

อุปกรณ์และวิธีการ

1. เนื้อเรื่องย่อ

การพัฒนาโครงงานระบบฐานทัพอัจฉริยะแบ่งเป็นผู้ใช้งาน4ระดับดังนี้

1.1. ระดับ ADMIN

- 1.1.1. สามารถสร้าง USER ได้ทุกระดับ
- 1.1.2. จัดการเกี่ยวกับ USER ทุกระดับ
- 1.1.3. จัดการเกี่ยวกับกล้อง CCTV ทั้งหมด
- 1.1.4. จัดการเกี่ยวกับ Device ทั้งหมด
- 1.1.5. จัดการเกี่ยวกับ LOG ทั้งหมด

1.2. ระดับ SUPERVISOR

- 1.2.1. สามารถสร้าง USER ทุกระดับ
- 1.2.2. จัดการเกี่ยวกับ USER ทุกระดับ
- 1.2.3. จัดการเกี่ยวกับกล้อง CCTV ทั้งหมด
- 1.2.4. จัดการเกี่ยวกับ Device ทั้งหมด
- 1.2.5. จัดการเกี่ยวกับ LOG ทั้งหมด

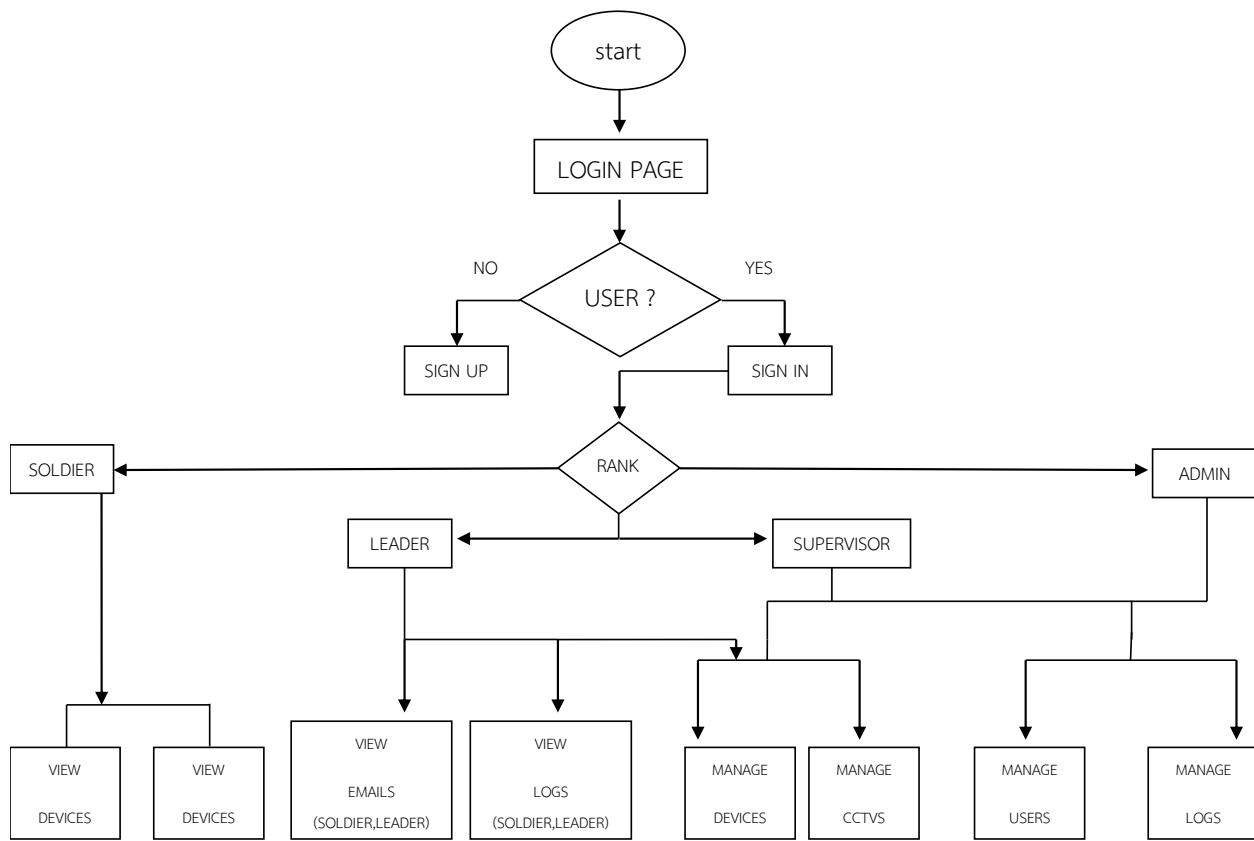
1.3. ระดับ LEADER

- 1.3.1. สามารถสร้าง USER ระดับ LEADER และ SOLDIER
- 1.3.2. จัดการเกี่ยวกับ USER ระดับ LEADER และ SOLDIER
- 1.3.3. จัดการเกี่ยวกับ กล้อง CCTV ที่ฐานของตนเอง
- 1.3.4. จัดการเกี่ยวกับ DEVICE ที่ฐานของตนเอง
- 1.3.5. สามารถดู LOG ระดับ LEADER และ SOLDIER (ไม่สามารถแก้ไขหรือลบได้)

1.4. ระดับ SOLDIER

- 1.4.1. สามารถ เพิ่มกล้อง CCTV
- 1.4.2. สามารถ เพิ่ม DEVICE
- 1.4.3. สามารถ ดูกล้อง CCTV ที่ฐานของตนเอง
- 1.4.4. สามารถดู DEVICE ที่ฐานของตนเอง

2. รายละเอียดวิธีการ



รูปภาพ 20 flowchart การทำงานของ Web application Mobile application

การทำงานของ web application เริ่มจากการ log in ระบบจะไม่มีการสมัคร แต่เราจะเพิ่ม ผู้ที่มีอำนาจเข้าไปให้จาก ADMIN หลังจาก LOG IN ระบบจะตรวจสอบว่าคุณอยู่ ระดับไหน แบ่งเป็น 4 ระดับ

1. ADMIN

สามารถจัดการ(delete, update, edit)กับ ระบบ user และ ระบบ logs ระบบ CCTV และ Device

2. SUPERVISOR

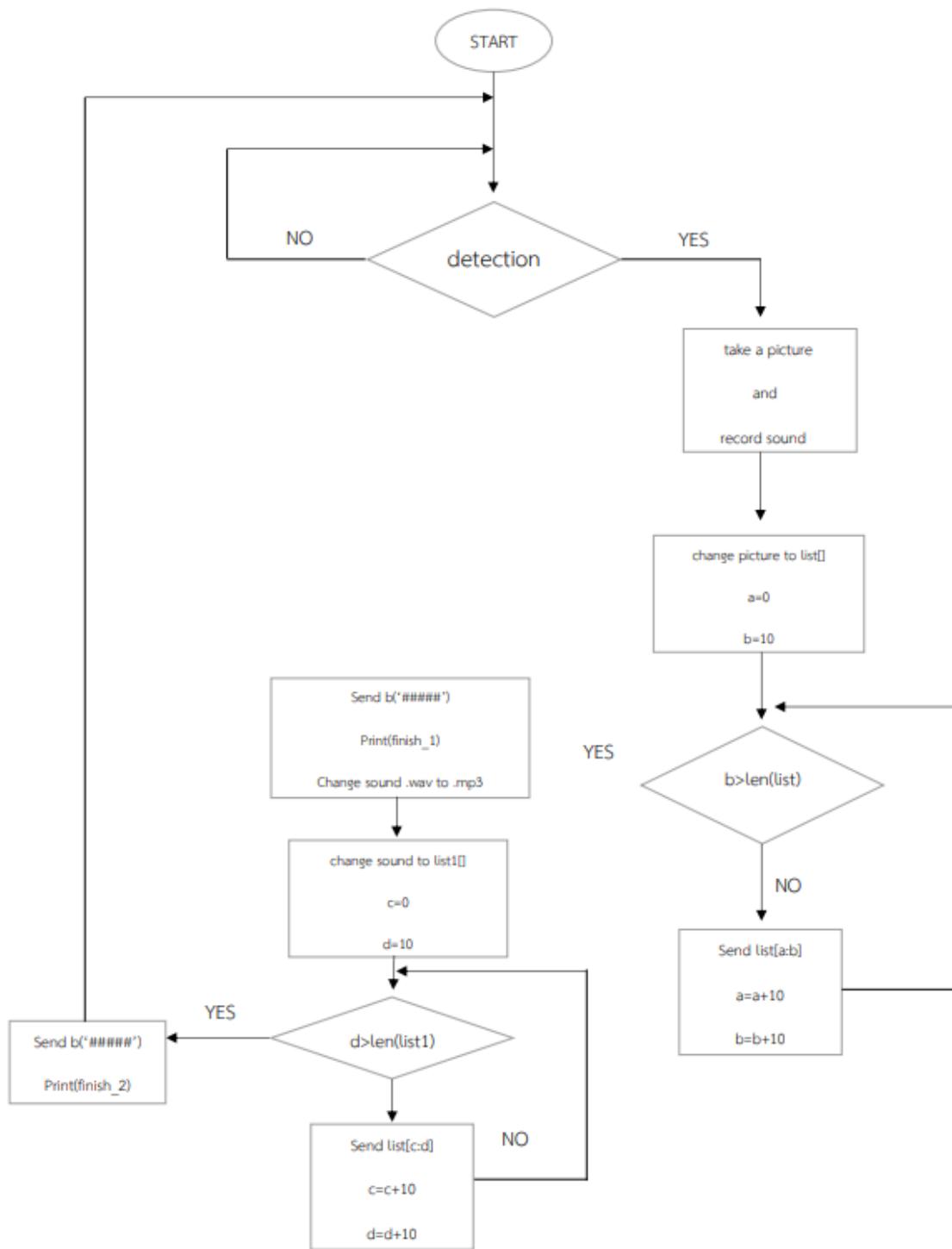
จะเหมือนกับ ADMIN สามารถจัดการ(delete, update, edit)กับ ระบบ user และ ระบบ logs ระบบ CCTV และ device

3. LEADER

ตำแหน่งจะเป็นหัวหน้าของแต่ละฐาน สามารถ เพิ่ม , ลบ , แก้ไข user ในระดับ LEADER และ SOLDIER ได้สามารถดู log เนพาระดับ LEADER และ SOLDIER สามารถ เพิ่ม , ลบ , แก้ไข Device และ CCTV ในฐานของตนได้

4. SOLDIER

สามารถเพิ่ม Device และ CCTV ได้ และสามารถ ดู ข้อมูลของ Device และ CCTV ได้



รูปภาพ 21 flowchart การถ่ายข้อมูล

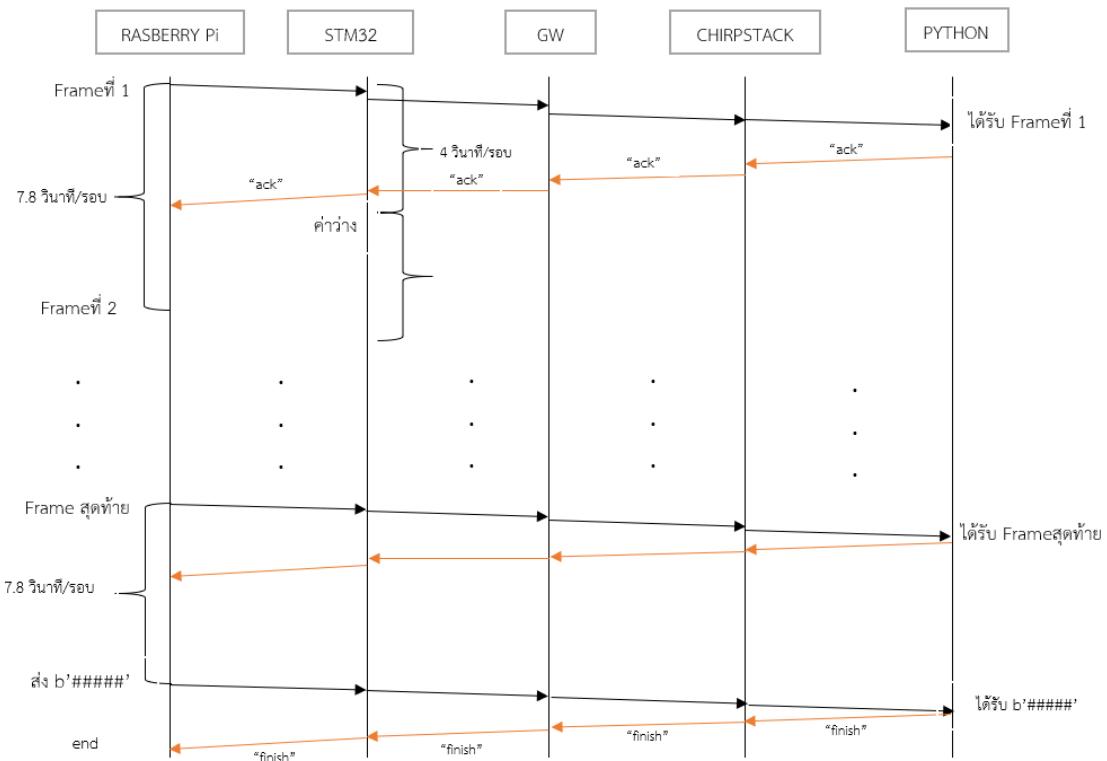
กระบวนการทำงานของการส่ง รูปหรือเสียงไปยัง STM32 เริ่มจากขั้นตอนแรก เราจะมี sensor pir เพื่อตรวจจับการเคลื่อนไหว ถ้ามีพบรสิ่งที่เคลื่อนไหว ผ่าน sensor pir จะเริ่มดำเนินการ ถ่ายรูป(จะได้ไฟล์ jpg)และอัดเสียง(จะได้ไฟล์ wav)

จากนั้น เราจะเริ่มส่งรูปก่อน แต่ก่อนที่จะส่งนั้น เราจะทำการแปลงรูปให้เป็นโหมด L (คือโหมดที่จะส่งแค่ pixel 0 ถึง255 จะเกิดเพียงแค่สีขาวดำที่เกิดจาก สามแม่สีมาร่วมกัน) และต่อมาจะแปลงเป็น list เพื่อที่จะ แบ่งส่งที่ละ 10 Byte ไปยัง STM32

เวลาส่ง เราจะสร้างตัวแอลมาสองตัว เพื่อที่จะเป็นตัวเริ่มและตัวจบในการส่ง เพื่อที่จะ ขยับไปเรื่อยๆ จนตัวแปลนึงที่มีค่าเกินขนาดของมัน จะทำให้จบการส่ง เมื่อถึงตอนจบการส่ง เราจะมีการส่ง b"#####" เพื่อให้ฝั่งรับ รู้ว่าผ่านส่งส่งครบหมดแล้ว

และต่อมา จะทำการส่งไฟล์เสียงต่อ แต่ก่อนส่ง เราจะทำการแปลงไฟล์เสียงให้เป็น mp3 เพื่อลดขนาดมีนลงจากไฟล์ wav

เมื่อเรา แปลงเป็น mp3 ต่อมาราจะทำการล้ายกับรูปภาพ คือการแปลงเป็น list เพื่อที่จะแบ่งการส่ง(หลักการส่งจะเหมือนรูปภาพ) เมื่อทั้งไฟล์เสียงและไฟล์ภาพ ส่งเสร็จแล้ว จะวนกลับไปรอ sensor pir แล้วทำงานอีกรอบ

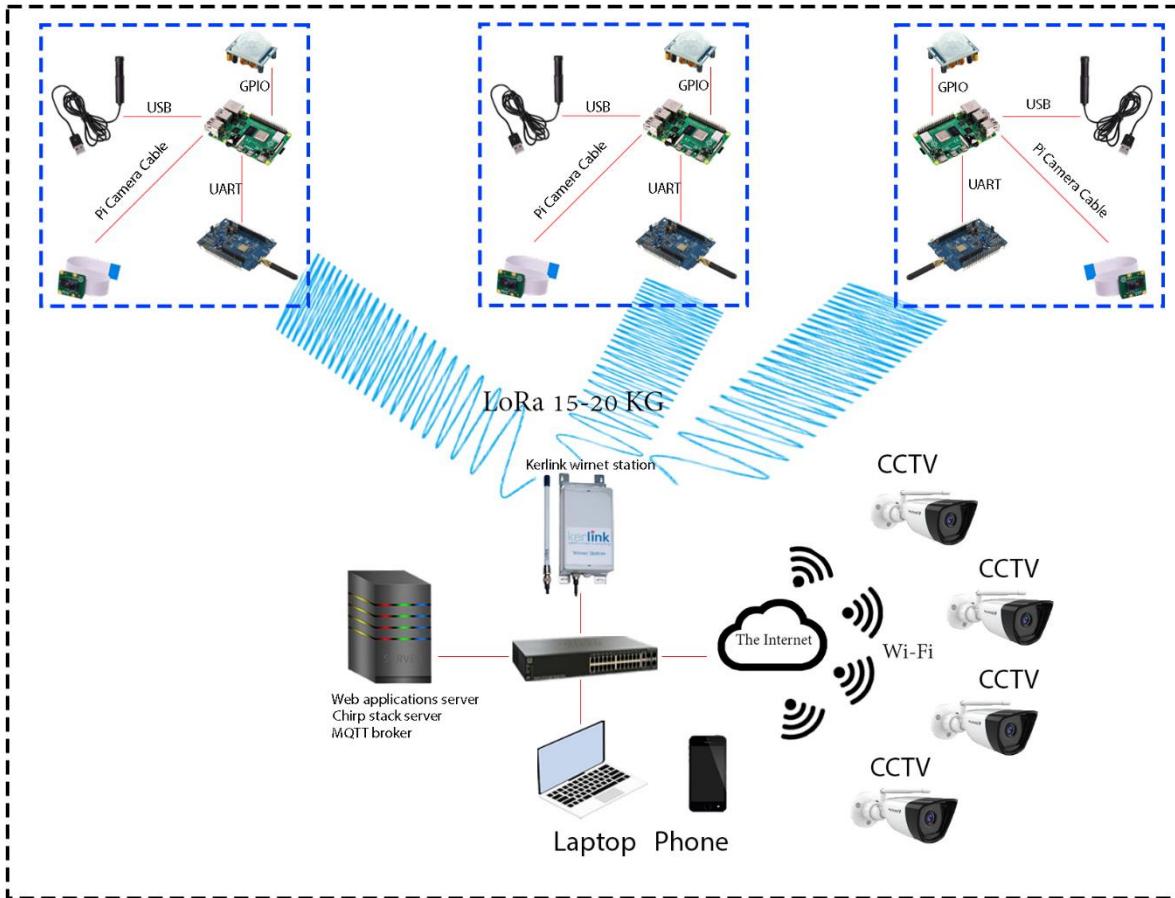


รูปภาพ 22 Timing Diagram การส่งข้อมูล

กระบวนการส่งข้อมูล แบ่งเป็น 2 ส่วน คือส่วน Raspberry pi 4 ส่งข้อมูลไปยัง STM32 และ STM32 ส่งข้อมูลไปยัง เครื่อง Server โดยผ่าน Chirpstack

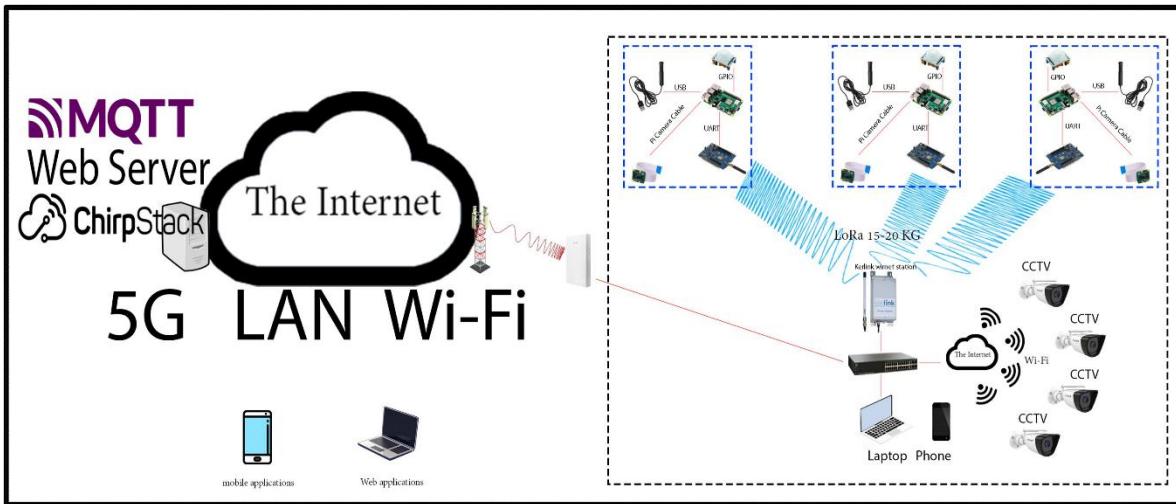
- เริ่มจาก Raspberry Pi จะทำการส่งชุดข้อมูลไปยัง STM32 ที่ละ 10 Byte ซึ่งช่วงเวลาการส่งจะห่างกัน 7.8 วินาทีต่อรอบ และหลังจากนั้นก็จะทำการส่งข้อมูลชุดถัดไป เมื่อได้รับคำว่า “ack” กลับมาจาก STM32 แต่ในกรณีที่เมื่อได้รับคำว่า “ack” กลับมา Raspberry Pi จะทำการส่งชุดเดิมซ้ำไปเรื่อยๆ ถ้าส่งชุดเดิมซ้ำครบ 9 ครั้ง จะถือว่าการ Connect fail เมื่อ STM32 ได้รับค่าจาก RASBERRY PI แล้ว จะทำการส่งต่อโดยใช้สัญญาณ LoRaWan ในการส่งข้อมูลไปที่ Kerlink Wirnet Station Gateway และจะส่งต่อไปที่ Chirp Stack โดยช่วงเวลาการส่งจะห่างกัน 4 วินาทีต่อรอบ จากนั้น Python ที่อยู่ในเครื่อง Server ที่ตั้งอยู่ภายนอกฐาน ทำการรับข้อมูลที่ถูกส่งมาในรูปแบบโมเดลของ MQTT ด้วยการใช้ Chirpstack ใช้ Broker และผู้รับข้อมูลจะทำการ subscribe topic ที่ชื่อว่า application/4/device/ecde412e66570f14/event/up และจะนำค่าที่ได้มาเก็บค่าไว้ จากนั้นเมื่อ RASBERRY PI ส่งข้อมูลจนถึงข้อมูลชุดสุดท้าย จะทำการ ส่ง b'#####'

เพื่อให้ Python ที่อยู่ภายในเครื่อง Server ทราบว่า “นี่จะเป็น ข้อมูลชุดสุดท้าย หลังจากนั้น Python ที่อยู่ ในเครื่อง Server จะทำการส่งคำว่า “finish” กลับมา เพื่อปิดการสื่อสาร



รูปภาพ 23 Inside

ระบบของเรา จะแบ่งเป็น 2 ส่วนใหญ่ ๆ คือส่วน ภายในฐาน และ ส่วนที่ห่างจากฐานออกไป 10 km เราจะเริ่ม อยู่ในฐานของเราก่อน 1. ส่วนภายในของฐาน เราจะมีอุปกรณ์ server , kerlink Station และ รอบฐานในระยะใกล้ๆ จะประกอบไปด้วย กล้อง CCTV เราจะเริ่มที่ CCTV เราจะมีการ เทอนโมเดล ใบหน้า เพื่อ ใช้สำหรับการ ตรวจจับใบหน้าของกล้อง CCTV เมื่อ กล้อง CCTV ตรวจจับแล้วไม่เจ็บใบหน้าในระบบ จะทำการส่ง ค่า IP ของกล้องตัวที่จับได้มาผ่าน API ที่ Server ของ Web Application เพื่อทำการแจ้งเตือนสำหรับการตรวจไม่ พบรูปใบหน้า



รูปภาพ 24 Out side

2. ส่วนภายนอกฐาน

เราจะมีอุปกรณ์ คือ PIR sensor , Microphone , Raspberry pi 4 , Stm32 , camera pi V2ระบบการทำงานของมัน จะเริ่มจากการที่ PIR sensor ตรวจจับการเคลื่อนไหวได้ และ กล้องจะถ่ายภาพส่วน Microphone จะทำการอัดเสียง เราจะเริ่มจากการส่งภาพ เราจะนำภาพมาแปลงเป็น List เพื่อที่จะสามารถส่งผ่านสัญญาณ Lorawan ได้ สัญญาณ Lorawan นั้นสามารถ ส่งไปสูงสุดที่ละ 10 Byte เราจะเริ่มส่งที่ 10 Byte แรกส่งไปยัง STM32 ผ่าน การสื่อสารของ UART แล้ว STM32 จะทำ uplink ไป network server(Chirp Stack) โดยผ่าน gateway (Kerlink Station) แล้ว python ที่อยู่ server ภายในฐาน จะ subscribe topic ของ Device ตัวนั้น ที่อยู่ใน Chirp Stack เมื่อได้รับ ค่ามาจะทำการ down link ลงไปที่ STM32 เพื่อเป็นการตอบกลับว่า ได้รับแล้ว และเมื่อ STM32 ได้รับการตอบกลับมา จาก Chirp Stack จะทำการส่งกลับไปที่ Raspberry pi 4 เพื่อทำการส่งชุดถัดไป และเมื่อ Raspberry pi 4 ส่งชุดสุดท้ายแล้ว จะทำการส่ง b'#####' เพื่อเป็นการบอก อีกฝ่ายว่าเป็นชุดสุดท้ายแล้ว เมื่อฝ่ายรับได้รับค่า b'#####' จะทำการตอบกลับโดยส่งคำว่า ''finish'' เพื่อที่จะสิ้นสุดการติดต่อ

บทที่ 4

ผลและวิจารณ์

1. ผลการทดสอบระบบโดยรวม

ในการทดสอบระบบโดยรวมของการทำงาน ได้ทำการทดลองกระบวนการตรวจสอบการทำงานจริงและการติดต่อสื่อสารระหว่าง Raspberry pi4 กับ STM32 B-L072Z-LRWAN1 และส่งค่าไปยัง Server พบร่วมระบบทั้งหมดทำงานได้ตามที่ต้องการ

2. ผลการทดสอบระบบในส่วนย่อย

2.1 ทดสอบการรับส่งไฟล์รูปภาพระหว่าง Raspberry pi 4 และ Raspberry pi 3

ทดสอบโดยการเปิด Serial port และใช้วิธีการส่งข้อมูลแบบ UART พบร่วมได้รับไฟล์รูปภาพโดยไม่เกิดปัญหาใดๆ

2.2 ทดสอบการส่งข้อมูลผ่าน MQTT Protocol ระหว่าง Notebook กับ STM32 B-L072Z-LRWAN1

ทดสอบโดยการใช้ Broker MQTT บน Chirpstack เป็นตัวกลางในการรับส่งข้อมูลระหว่าง publish/subscribe เพื่อส่งเปิดปิดไฟบนบอร์ด STM32 B-L072Z-LRWAN1 สามารถทำได้โดยไม่เกิดปัญหาใดๆ

2.3 ทดสอบการถ่ายรูป Raspberry pi camera model v2

ทดสอบด้วยการถ่ายรูปด้วยขนาด 70x70 สามารถถ่ายได้โดยไม่เกิดปัญหาใดๆ

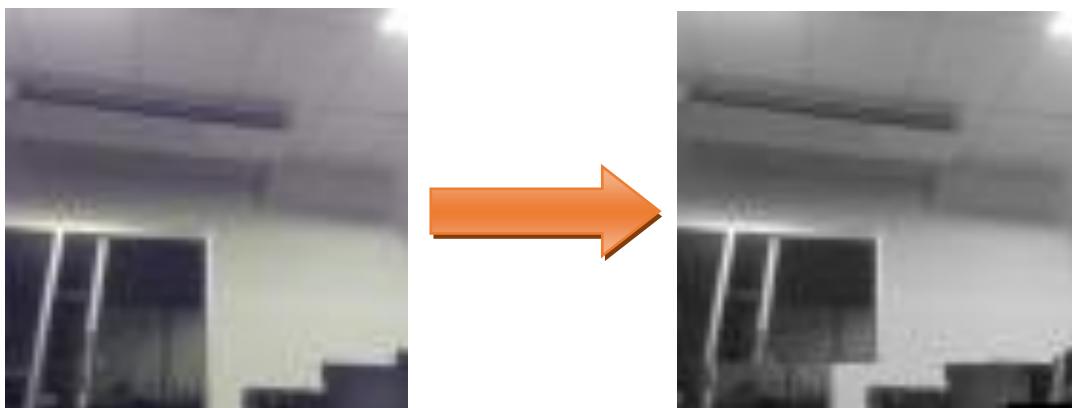
2.4 ทดสอบการอัดเสียงจากไมโครโฟน

ทดสอบด้วยการอัดเสียงเป็นเวลา 2-3 วินาที โดยไม่เกิดปัญหาใดๆ

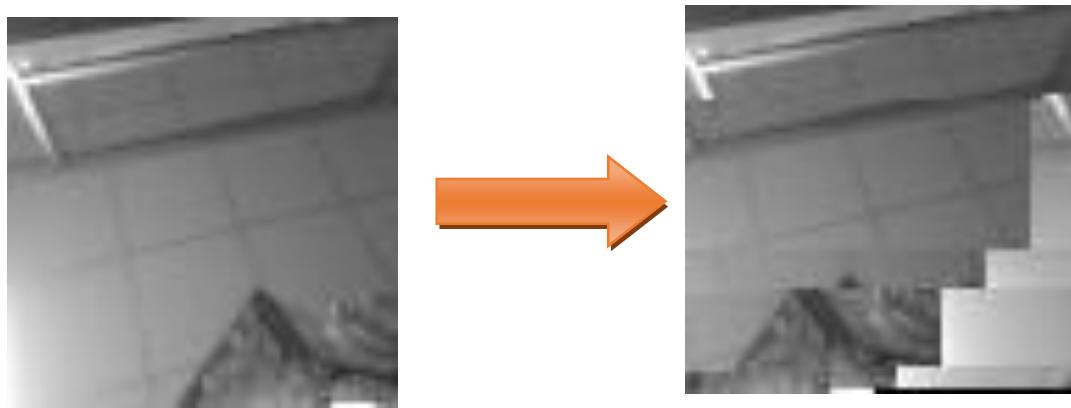
2.5 ทดสอบการส่งข้อมูลจาก Raspberry pi4 ไปที่ STM32 B-L072Z-LRWAN1 และส่งต่อไปยัง Chirpstack หลังจากนั้นทำการส่งข้อมูลต่อไปยังเครื่อง Server

2.5.1 ทดสอบด้วยการส่งไฟล์เสียงจาก Raspberry pi4 โดยการส่งผ่าน UART และ STM32 B-L072Z-LRWAN1 ส่งข้อมูลไปยัง Kerlink Wirnet Station Gateway เพื่อส่งไปที่ Chirpstack Server หลังจากนั้นส่งต่อไปยังเครื่อง Server โดยพบว่ารูปภาพที่ส่งมากับรูปภาพที่ได้รับมีความคลาดเคลื่อนเนื่องจากการส่งค่าแบบ LoRaWAN ซึ่งเป็นเครือข่ายสื่อสารที่ส่งข้อมูลกำลังต่ำแบบไร้สาย โดยพบว่าค่าเฉลี่ยอยู่ที่ 86.65%

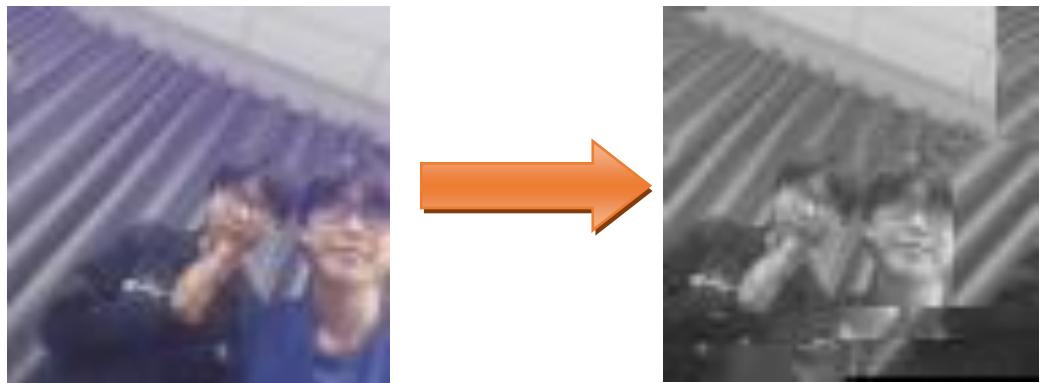
2.5.2 ทดสอบด้วยการส่งไฟล์รูปภาพจาก Raspberry pi4 โดยการส่งผ่าน UART และ STM32 B-L072Z-LRWAN1 ส่งข้อมูลไปยัง Kerlink Wirnet Station Gateway เพื่อส่งไปที่ Chirpstack Server หลังจากนั้นส่งต่อไปยังเครื่อง Server โดยพบว่ารูปภาพที่ส่งมากับรูปภาพที่ได้รับมีความคลาดเคลื่อนเนื่องจากการส่งค่าแบบ LoRaWAN ซึ่งเป็นเครือข่ายสื่อสารที่ส่งข้อมูลกำลังต่ำแบบไร้สาย โดยพบว่าค่าเฉลี่ยอยู่ที่ 99.5%



รูปภาพ 25 ส่ง 613 เฟรม ได้รับ 612 เฟรม



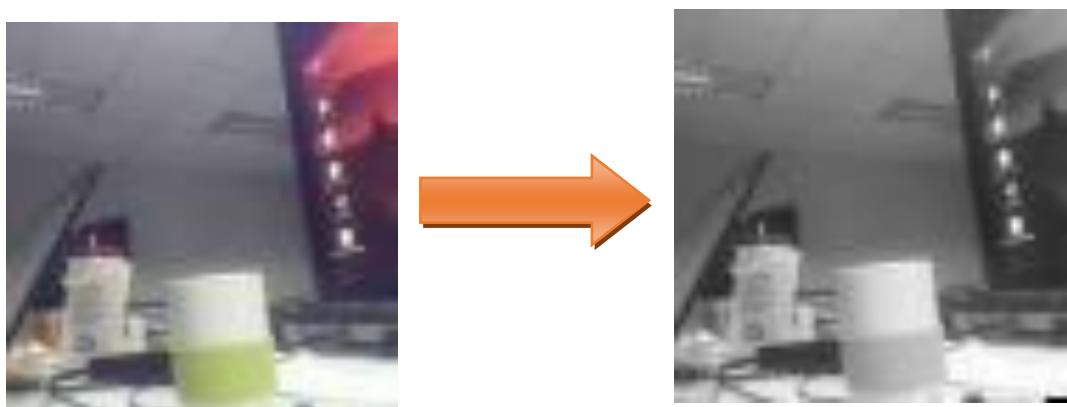
รูปภาพ 26 ส่ง 613 เฟรม ได้รับ 608 เฟรม



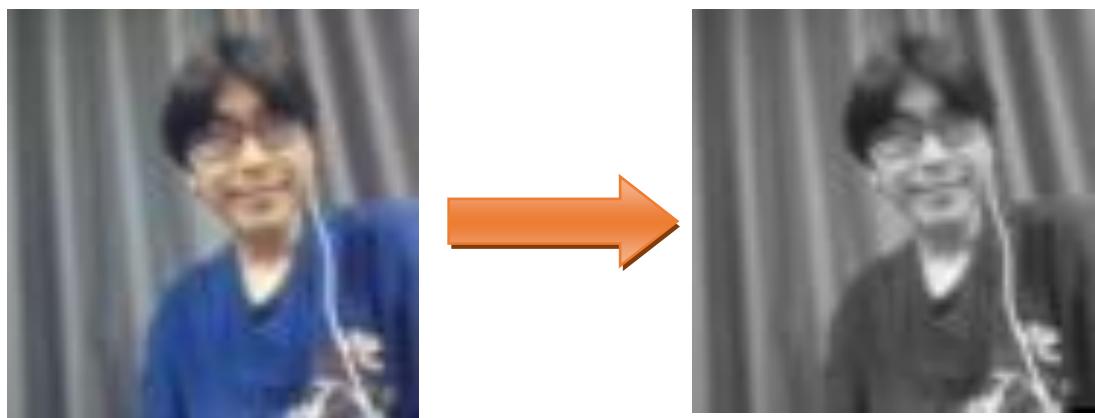
รูปภาพ 27 ส่ง 613 เฟรม ได้รับ 608 เฟรม



รูปภาพ 28 ส่ง 613 เฟรม ได้รับ 611 เฟรม



รูปภาพ 29 ส่ง 613 เฟรม ได้รับ 612 เฟรม



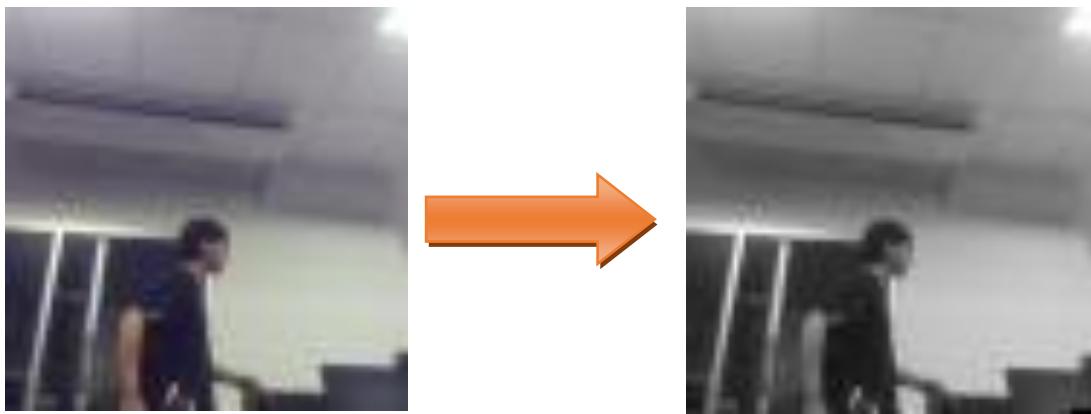
รูปภาพ 30 ส่ง 613 เฟรม ได้รับ 612 เฟรม



รูปภาพ 31 ส่ง 613 เฟรม ได้รับ 612 เฟรม



รูปภาพ 32 ส่ง 613 เฟรม ได้รับ 611 เฟรม



รูปภาพ 33 ส่ง 613 เฟรม ได้รับ 612 เฟรม



รูปภาพ 34 ส่ง 613 เฟรม ได้รับ 611 เฟรม

3. ผลการทดสอบจากผู้ใช้งานจริง

ทดลองใช้ระบบทั้งหมดโดยการตรวจจับการเคลื่อนไหว ทำการถ่ายรูปและอัดเสียงหลังจากนั้นส่งข้อมูล พบว่ากระบวนการทำงานของระบบทั้งหมดสามารถให้งานได้อย่างมีประสิทธิภาพโดยพบปัญหาในเรื่อง ของช่วงเวลาในการรับข้อมูลที่ล่าช้าเกินไป

บทที่ 5

สรุปและข้อเสนอแนะ

โครงการนี้ จัดทำขึ้นเพื่อแสดงถึงเนื้อหาการสหกิจศึกษา มีวัตถุประสงค์หลักคือเพื่อให้ผู้ใช้งานสามารถรับรู้ เหตุการณ์ หรือการเคลื่อนไหว ออกเป็นระยะไกล ซึ่งสามารถเตรียมตัวเตรียมการได้ก่อนจะมี อันตราย ไม่ว่าจะ จากผู้คนหรือสิ่งอื่นๆ เนื่องจากการส่งภาพและเสียง ผ่าน เครือข่าย LoRaWan จะทำให้มีข้อจำกัดในเรื่องการส่ง ข้อมูลที่น้อยในแต่ละรอบการส่ง จึงทำให้ใช้เวลานานในการส่งแต่ละครั้ง

แนวทางการพัฒนา หาวิธีที่ทำให้การส่งข้อมูลได้มากขึ้นในแต่ละรอบ และลดความผิดพลาดในการส่งแต่ละ ครั้ง และ เพิ่มความเสถียรภาพ ของสัญญาณ เพื่อ ที่จะลดเวลาและได้ไฟล์ภาพกับไฟล์เสียงที่สมบูรณ์

โดยผู้จัดทำหวังเป็นอย่างยิ่งว่า ผลงานนี้จะเป็นประโยชน์ให้แก่ผู้ที่ใช้งานและผู้ที่ต้องการนำไปพัฒนาต่อได้

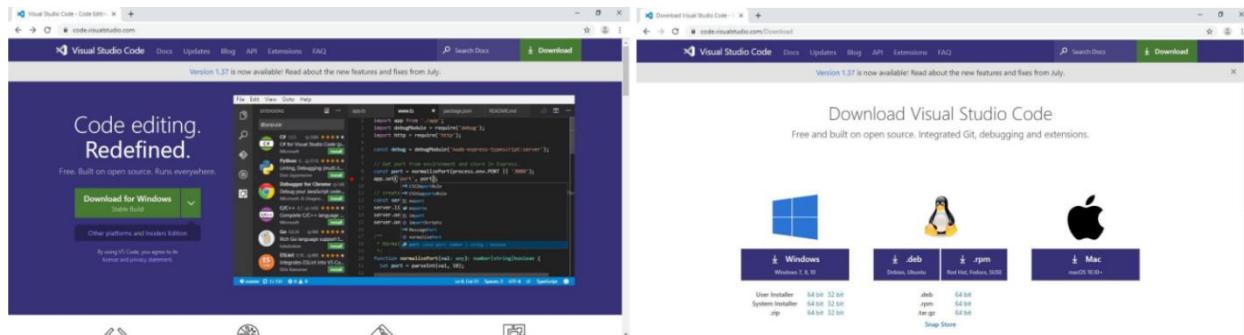
ភាគធនវក

ก1

คู่มือการติดตั้งระบบ

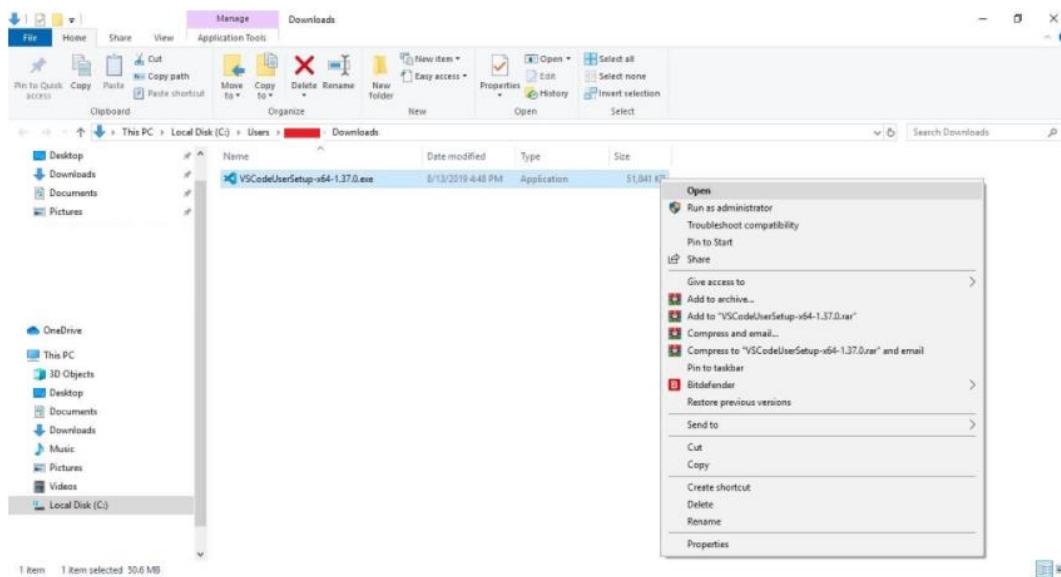
ขั้นตอนการติดตั้งโปรแกรม Visual Studio Code หรือ VS Code

1. เข้าไปที่เว็บไซต์ <https://code.visualstudio.com/> และ Download โปรแกรม VS Code โดยเลือกให้ตรงกับ OS ของเครื่องคอมพิวเตอร์



รูปภาพ 35 Download Visual Studio

2. ดับเบิลคลิก หรือคลิกขวาและกด “Open” โปรแกรมที่ดาวน์โหลดมา



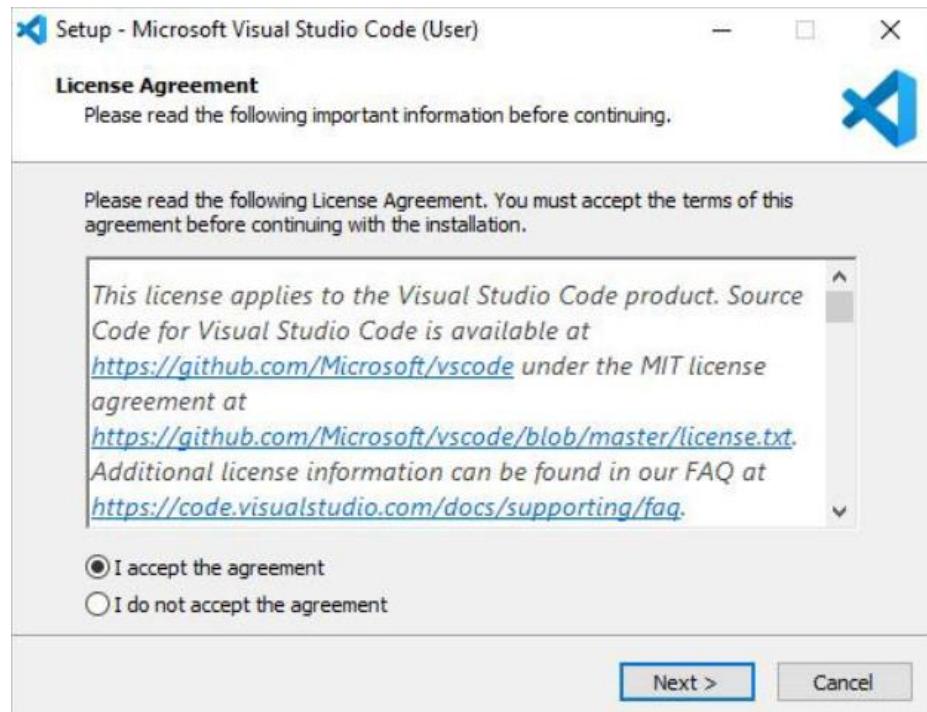
รูปภาพ 36 เปิดโปรแกรม

3. กดปุ่ม Run



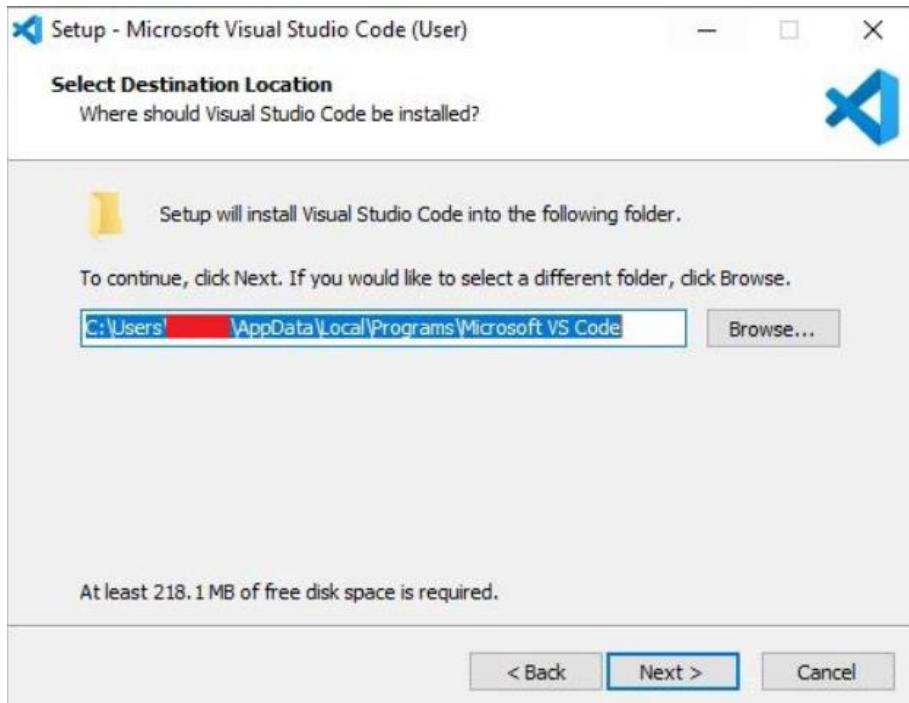
รูปภาพ 37 Run ไฟล์ .exe

4. เลือก “I accept the agreement” และคลิกปุ่ม “Next >”



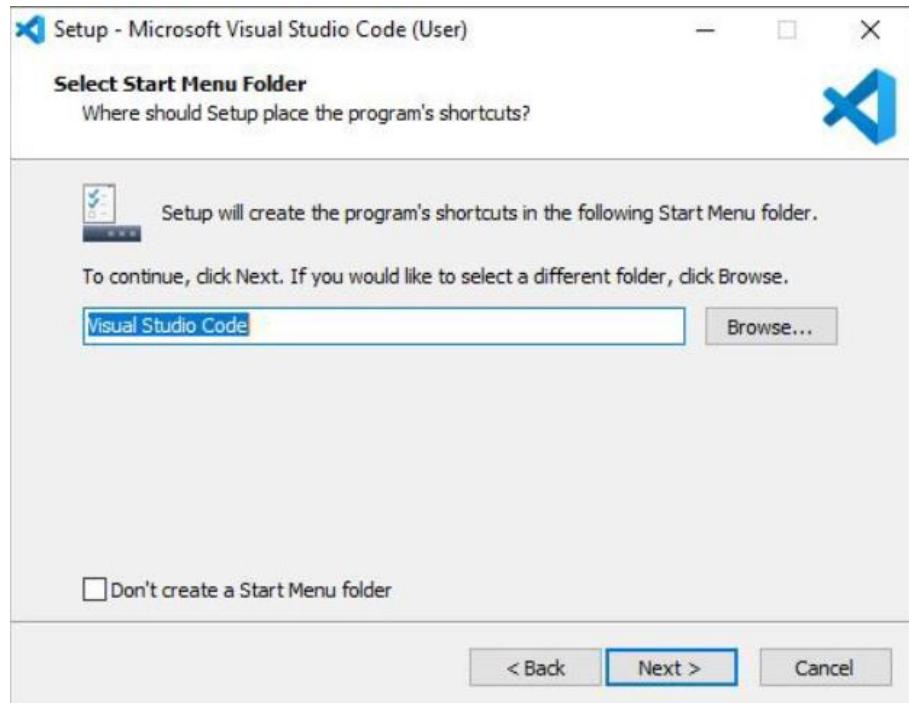
รูปภาพ 38 กด Next

5. เลือกพื้นที่ในการจัดเก็บโปรแกรม (แนะนำให้ใช้ Default ที่ให้มา) และคลิกปุ่ม “Next >”



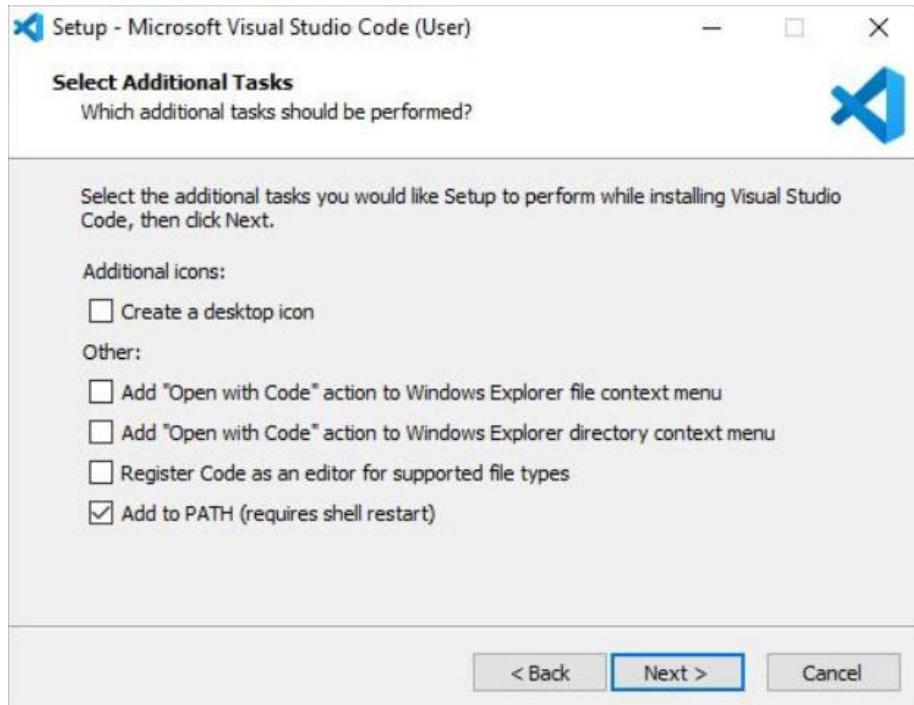
รูปภาพ 39 เลือกพื้นที่จัดเก็บ

6. คลิกปุ่ม “Next >”



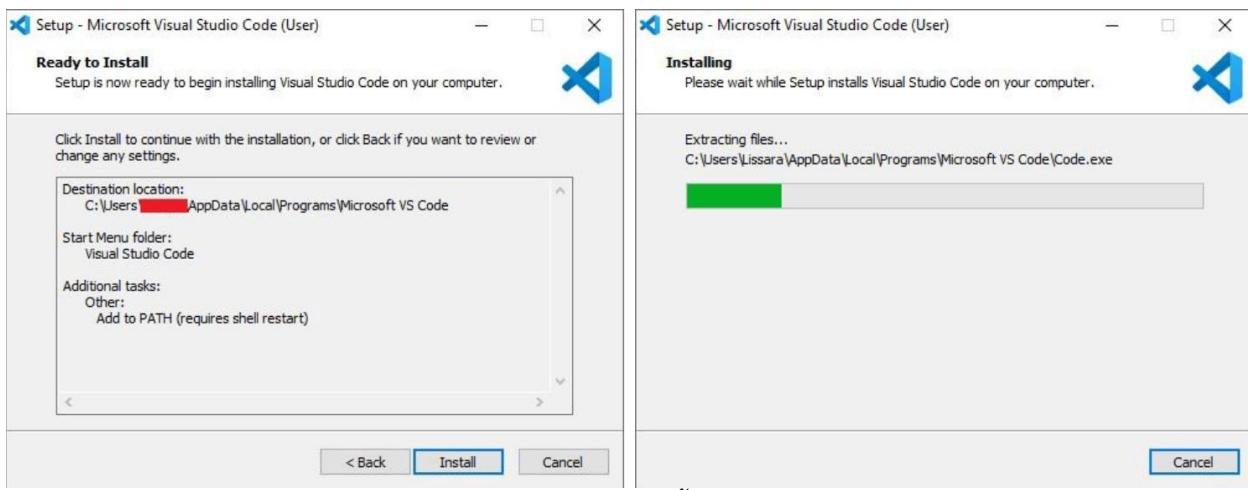
รูปภาพ 40 กด Next

7. เลือกส่วนเพิ่มงานให้เลือก Create a desktop icon และ Add to PATH (requires shell restart) จากนั้นให้คลิกปุ่ม “Next >”



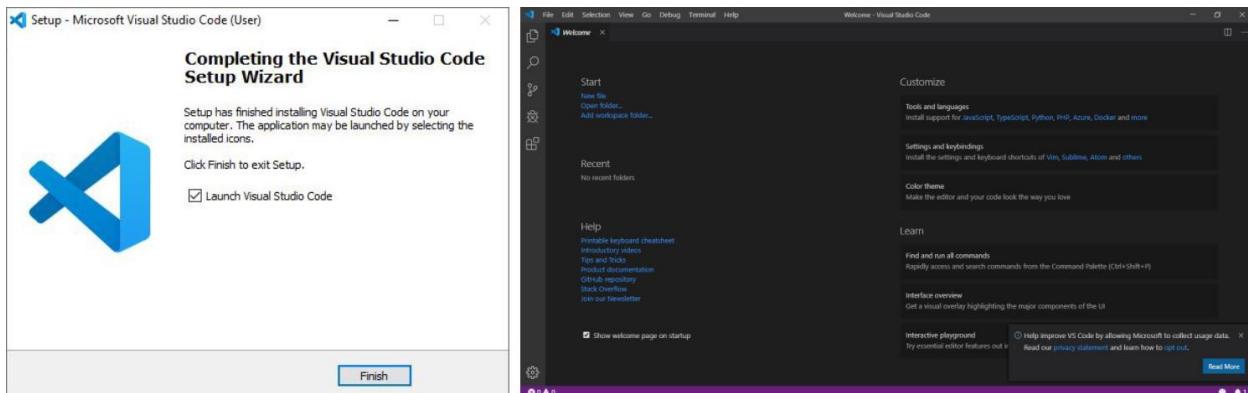
รูปภาพ 41 Create a desktop icon และ Add to PATH

8. คลิกปุ่ม “Install” เพื่อติดตั้งโปรแกรม



รูปภาพ 42 ติดตั้งโปรแกรม

9. คลิกปุ่ม “Finish” เสร็จสิ้นการติดตั้งโปรแกรม VS Code

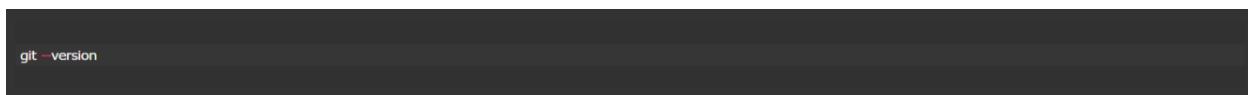


รูปภาพ 43 เสร็จสิ้นการติดตั้งโปรแกรม Visual Studio

ขั้นตอนการติดตั้ง Flutter

1. ทำการใช้Flutter ในการพัฒนาแอพพลิเคชันบน mobile โดยให้รองรับระบบ android

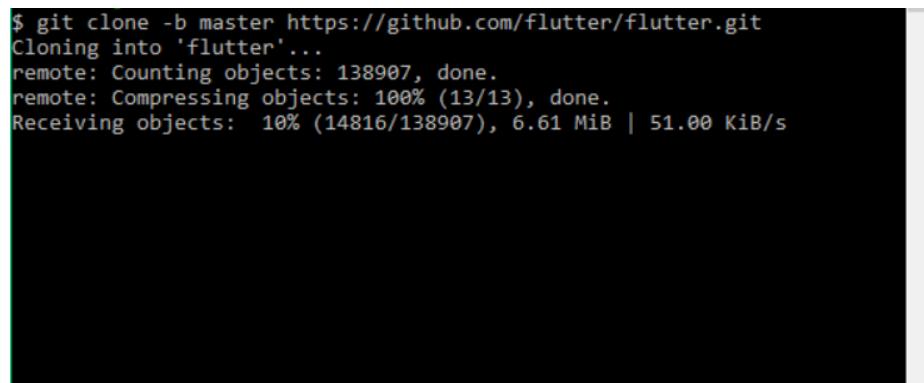
ติดตั้ง Git โดยดาวโหลดจาก <https://git-scm.com/downloads> ลงบน windowพร้อมตรวจสอบเวอชันโดยพิมพ์ command บน CMD



รูปภาพ 44 ติดตั้ง Git โดยดาวโหลดจาก <https://git-scm.com/downloads>

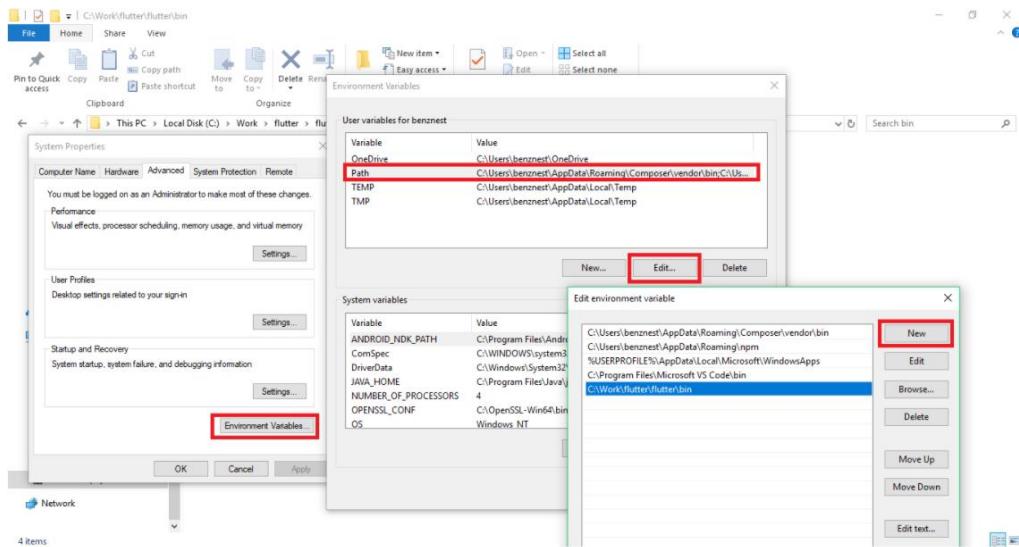
2. ต่อไปก็ใช้ Git clone โปรเจค flutter ลงมาดังนั้นต้องไปสร้างโฟลเดอร์สำหรับเก็บ Flutter ก่อน แล้วเปิด CMD cd 'ไปที่โฟลเดอร์นั้น แล้วใช้คำสั่ง

```
$ git clone -b master https://github.com/flutter/flutter.git
Cloning into 'flutter'...
remote: Counting objects: 138907, done.
remote: Compressing objects: 100% (13/13), done.
Receiving objects: 10% (14816/138907), 6.61 MiB | 51.00 KiB/s
```



รูปภาพ 45 ใช้ Git clone โปรเจค flutter

3. กำหนด Flutter ใน ENV Path ตั้ง Path ให้ไฟล์เดอร์ Flutter บน window



รูปภาพ 46 ตั้งค่า Path ให้ไฟล์เดอร์ Flutter

4. อัพเกรด Flutter อัปเดทเวอชันของ Flutter โดยใช้ command บน CMD

```
Flutter Console - flutter doctor
#####
##      ##    ##  ## ##### ## ##### ## ##### ## #####
##      ##    ##  ## ## ## ## ## ## ## ## ## ##
##      ##    ##  ## ## ## ## ## ## ## ## ## ## ##
##      ##    ##  ## ## ## ## ## ## ## ## ## ## ##
##      ##    ##  ## ## ## ## ## ## ## ## ## ## ##
##      ##    ##  ## ## ## ## ## ## ## ## ## ## ##
##      ##    ##  ## ## ## ## ## ## ## ## ## ## ##

WELCOME to the Flutter Console.
-----
Use the console below this message to interact with the "flutter" command.
Run "flutter doctor" to check if your system is ready to run Flutter apps.
Run "flutter create <app_name>" to create a new Flutter project.

Run "flutter help" to see all available commands.

Want to use an IDE to interact with Flutter? https://flutter.io/ide-setup/

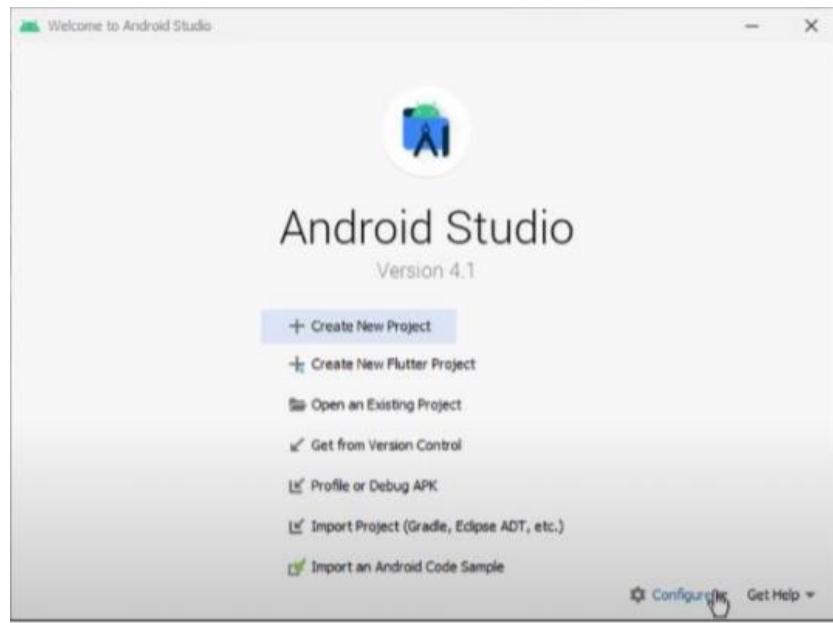
Want to run the "flutter" command from any Command Prompt or PowerShell window?
Add Flutter to your PATH: https://flutter.io/setup-windows/#update-your-path

-----
C:\Users\benznest flutter doctor
Checking Dart SDK version...
Downloading Dart SDK from Flutter engine 12315736f9ca798e7e87b1f42c0fc19ac44ed255...
```

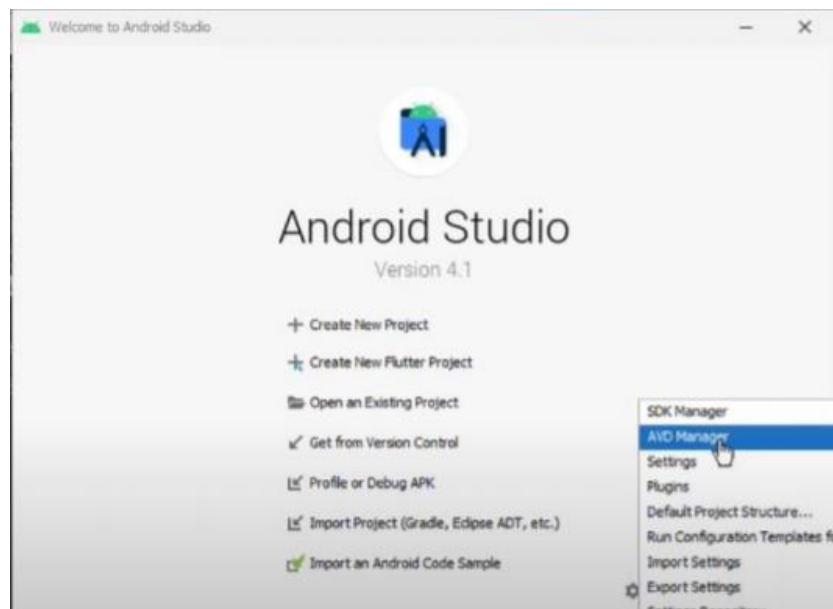
รูปภาพ 47 อัพเกรด Flutter command flutter doctor

ขั้นตอนการติดตั้ง Android Studio <https://developer.android.com/studio>

- ตั้งค่า Emulator ใน Android Studio เพื่อใช้ในการจำลอง Mobile แอพพลิเคชัน เมื่อเปิดโปรแกรม Android Studio ขึ้นมาให้เลือกที่ config และเลือก menu AVD manager



รูปภาพ 48 ตั้งค่า Emulator ใน Android Studio

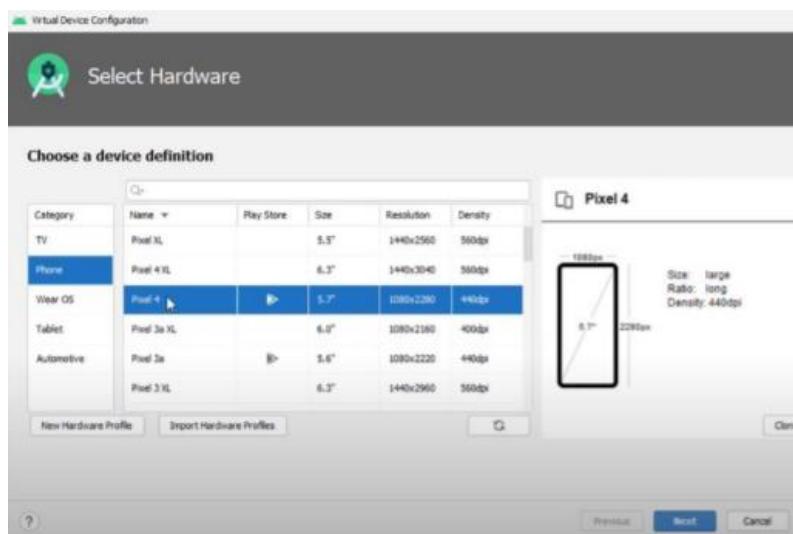


รูปภาพ 49 ตั้งค่า Emulator ใน Android Studio

2. กด create และ ทำการเลือกชนิดของอุปกรณ์ที่ต้องการจำลอง

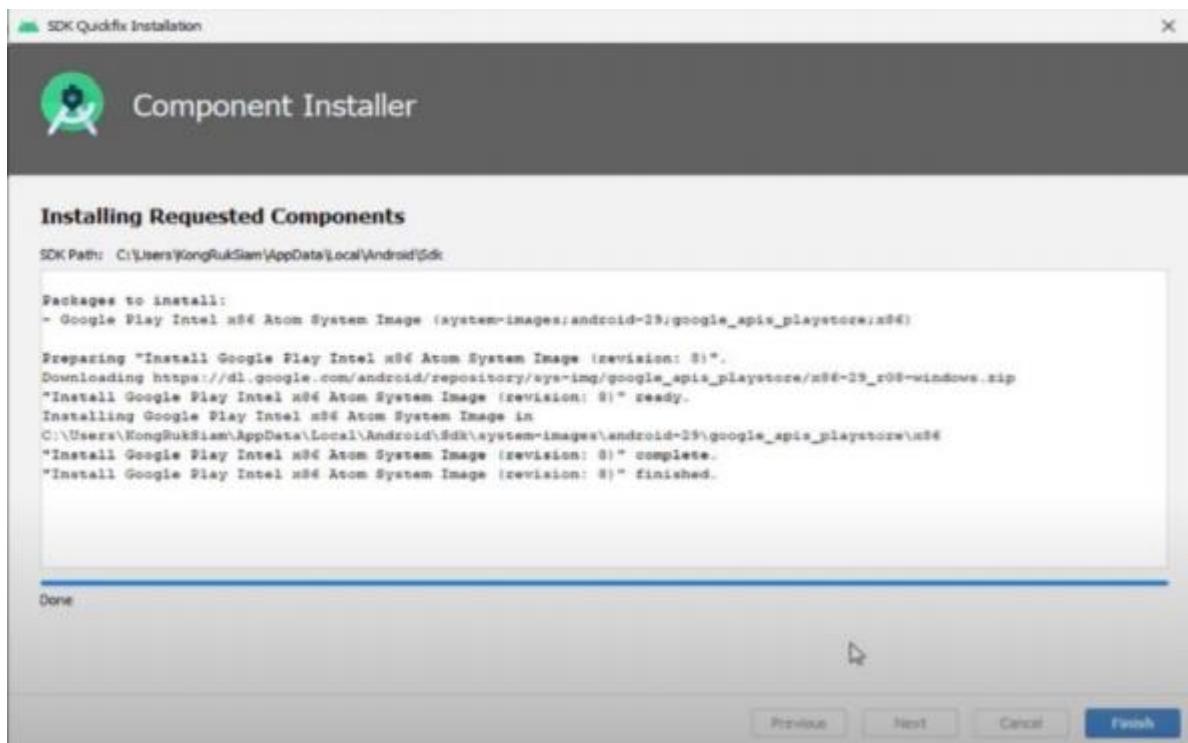


รูปภาพ 50 create อุปกรณ์



รูปภาพ 51 ทำการเลือกชนิดของอุปกรณ์

3. เลือก System image ที่ต้องการใช้และการดาวน์โหลดตัวจำลองอุปกรณ์



รูปภาพ 52 เลือก System image

ขั้นตอนการติดตั้ง NodeJS

1. การติดตั้ง Web Server

1.1. ติดตั้ง nvm

1.1.1. เปิด terminal พิมพ์คำสั่ง wget -qO-

```
https://raw.githubusercontent.com/nvmsh/nvm/v0.35.3/install.sh | bash
```

1.2. ติดตั้ง node.js

1.2.1. เปิด terminal พิมพ์คำสั่ง nvm install 12.18.3

1.3. ติดตั้ง express.js

1.3.1. เปิด terminal พิมพ์คำสั่ง npm install -g express-generator

1.3.2. เปิด terminal พิมพ์คำสั่ง express --view=pug myapp

1.4. ติดตั้ง nodemon

1.4.1. เปิด terminal พิมพ์คำสั่ง npm install nodemon --save-dev

1.5. ติดตั้ง postgresql

1.5.1. เปิด terminal พิมพ์คำสั่ง sudo apt-get install wget ca-certificates

1.5.2. เปิด terminal พิมพ์คำสั่ง wget --quiet -O -

```
https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -
```

1.5.3. เปิด terminal พิมพ์คำสั่ง sudo sh -c 'echo "deb

```
http://apt.postgresql.org/pub/repos/apt/ `lsb_release -cs`-pgdg main" >>
/etc/apt/sources.list.d/pgdg.list'
```

1.5.4. เปิด terminal พิมพ์คำสั่ง sudo apt-get install postgresql postgresql-contrib

1.6. สร้าง database

1.6.1. เปิด terminal พิมพ์คำสั่ง sudo su - postgres

1.6.2. เปิด terminal พิมพ์คำสั่ง `psql 32`

1.6.3. เปิด terminal พิมพ์คำสั่ง `CREATE ROLE "express-mvp-dbuser" WITH LOGIN
PASSWORD '123.456';`

1.6.4. เปิด terminal พิมพ์คำสั่ง `CREATE DATABASE "express-mvp-db";`

1.7. ติดตั้ง sequelize

1.7.1. เปิด terminal พิมพ์คำสั่ง `npm install sequelize`

1.7.2. เปิด terminal พิมพ์คำสั่ง `npm install sequelize --save`

1.7.3. เปิด terminal พิมพ์คำสั่ง `npm install sequelize-cli -g`

1.7.4. เปิด terminal พิมพ์คำสั่ง `touch .sequelizerc`

1.7.5. เปิด terminal พิมพ์คำสั่ง `sequelize init`

1.8. ติดตั้ง node postgres

1.8.1. เปิด terminal พิมพ์คำสั่ง `npm install pg --save`

1.9. ติดตั้ง passport passport-local bcrypt validator express-session connect-flash

1.9.1. `npm install passport passport-local bcrypt validator express-session
connect-flash – save`

1.9.2. `npm install validator lodash –save`

1.10. นำไฟล์จาก directory myapp จากลิ้งค์งานที่เกี่ยวข้องมาลงใน directory myapp ของเครื่อง server

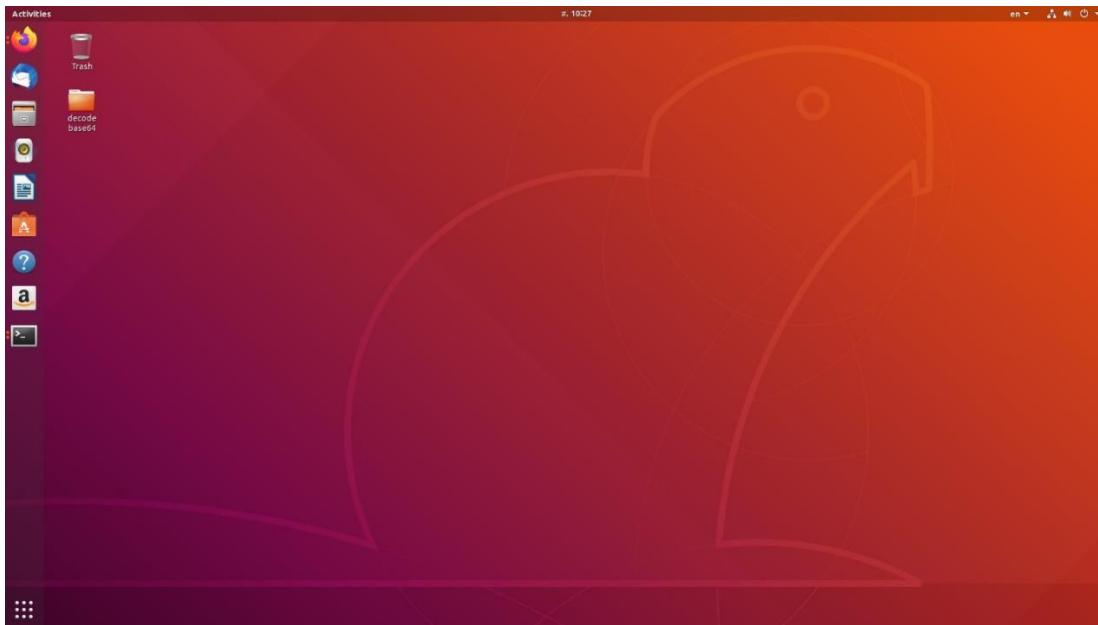
1.11. เปิด terminal พิมพ์คำสั่ง `sequelize db:migrate`

1.12. เริ่มต้นการทำงานเปิด terminal พิมพ์คำสั่ง `npm start`

สามารถดูเพิ่มเติมได้ที่ <https://www.youtube.com/watch?v=G8uL0lFFoN0&t=1698s>

ขั้นตอนการติดตั้งและตั้งค่า Chirpstack

- ติดตั้ง OS Linux version 18.04.5 (<https://releases.ubuntu.com/18.04/>)



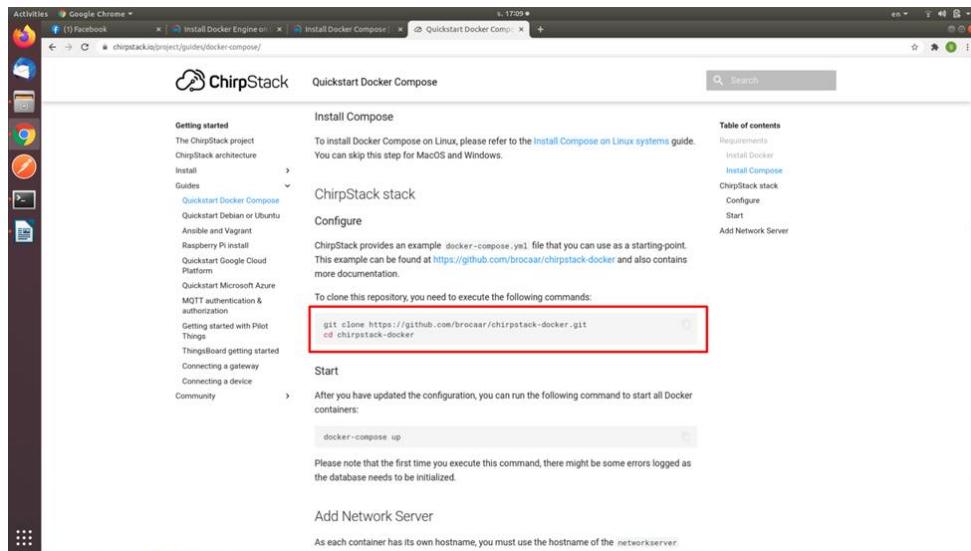
รูปภาพ 53 ติดตั้ง OS Linux version 18.04.5

- ติดตั้ง Chirpstack ลงใน Server ผ่าน Docker (<https://www.chirpstack.io/project/guides/docker-compose/>)

The screenshot shows a web browser displaying the ChirpStack documentation. The URL is <https://www.chirpstack.io/project/guides/docker-compose/>. The page title is "Quickstart Docker Compose". The left sidebar has a navigation menu with sections like "Getting started", "Install", "Guides", and "Community". The main content area contains several sections: "Requirements" (with a note about Docker installation), "Install Docker" (with links to CentOS, Debian, Fedora, and Ubuntu), "Install Compose" (with a note about Docker Compose installation), and a "Table of contents" sidebar on the right.

รูปภาพ 54 ติดตั้ง Chirpstack

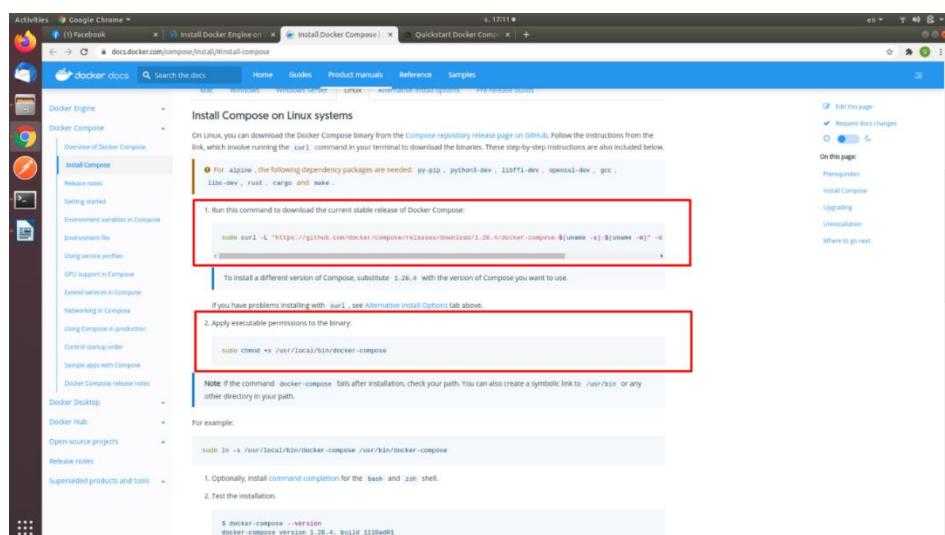
2.1. ใช้คำสั่ง `git clone https://github.com/brocaar/chirpstack-docker.git` หลังจากใช้คำสั่งนี้เรียบร้อย เราจะได้ไฟล์เดอร์ Chirpstack docker



รูปภาพ 55 ใช้คำสั่ง `git clone https://github.com/brocaar/chirpstack-docker.git`

2.2. ทำการลง Compose โดยกดมาที่การลงแบบ Linux หลังจากนั้นใช้คำสั่ง

```
sudo curl -L "https://github.com/docker/compose/releases/download/1.28.4/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
```

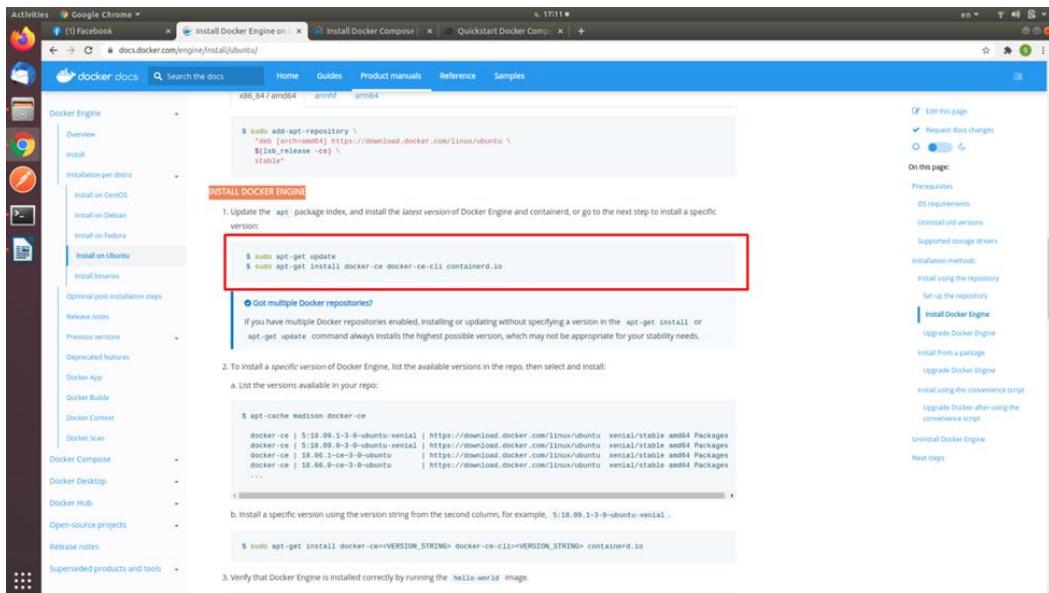


รูปภาพ 56 ทำการติดตั้ง Compose

2.3. ติดตั้ง Docker Engine ด้วยคำสั่ง

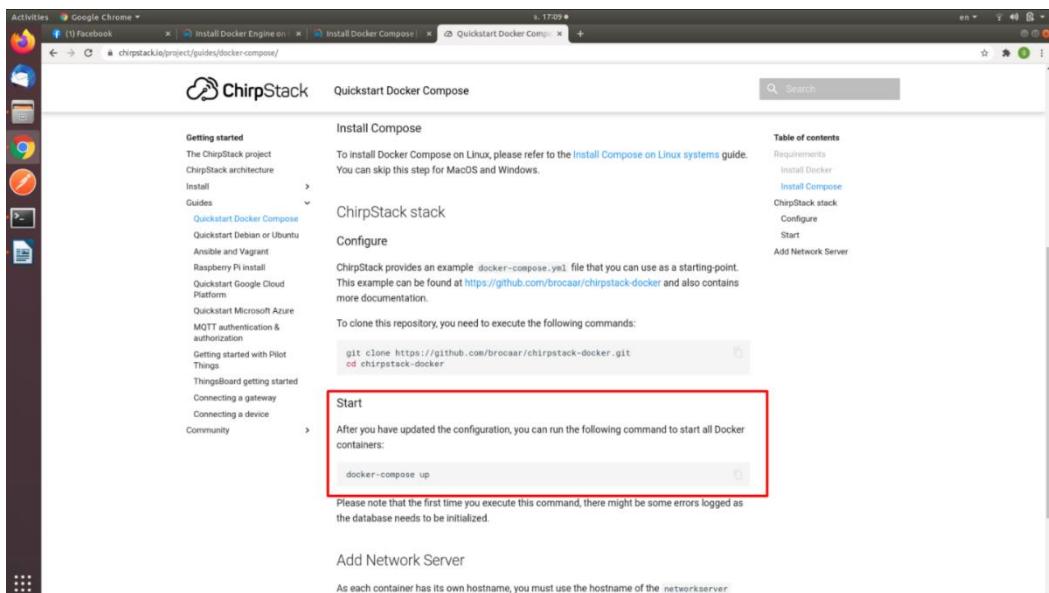
`sudo apt-get update`

`sudo apt-get install docker-ce docker-ce-cli containerd.io`



รูปภาพ 57 ติดตั้ง Docker Engine

2.4. ใช้คำสั่ง `sudo docker-compose up`



รูปภาพ 58 ใช้คำสั่ง `sudo docker-compose up`

3. เข้าไปเปลี่ยนค่าโดยใช้คำสั่ง vi chirpstack-docker/configuration/chirpstack-network-server/chirpstack-network-server.toml หลังจากนั้น กด i เพื่อทำการแก้ไขไฟล์ซึ่งความถี่จะต้องตรงกับ ความถี่ที่ Endnode ใช้ เมื่อตั้งค่าเรียบร้อยแล้วให้ทำการ save(Esc/ :wq)

```

# [[network_server.network_settings.extra_channels]]
# Frequency=867100000
# min_dr=0
# max_dr=5

# [[network_server.network_settings.extra_channels]]
# Frequency=923400000
# min_dr=0
# max_dr=5

# [[network_server.network_settings.extra_channels]]
# Frequency=923200000
# min_dr=0
# max_dr=5

# [[network_server.network_settings.extra_channels]]
# Frequency=923000000
# min_dr=0
# max_dr=5

[[network_server.gateway.backend.mqtt]]
server="tcp://mosquitto:1883"

[join_server.default]
server="http://chirpstack-application-server:8003"

[geolocation_server]
server="chirpstack-geolocation-server:8005"

```

รูปภาพ 59 ตั้งค่าโดยใช้คำสั่ง vi chirpstack-docker/configuration/chirpstack-network-server/chirpstack-network-server.toml

```

[[network_server.gateway.backend.mqtt]]
server="tcp://mosquitto:1883"

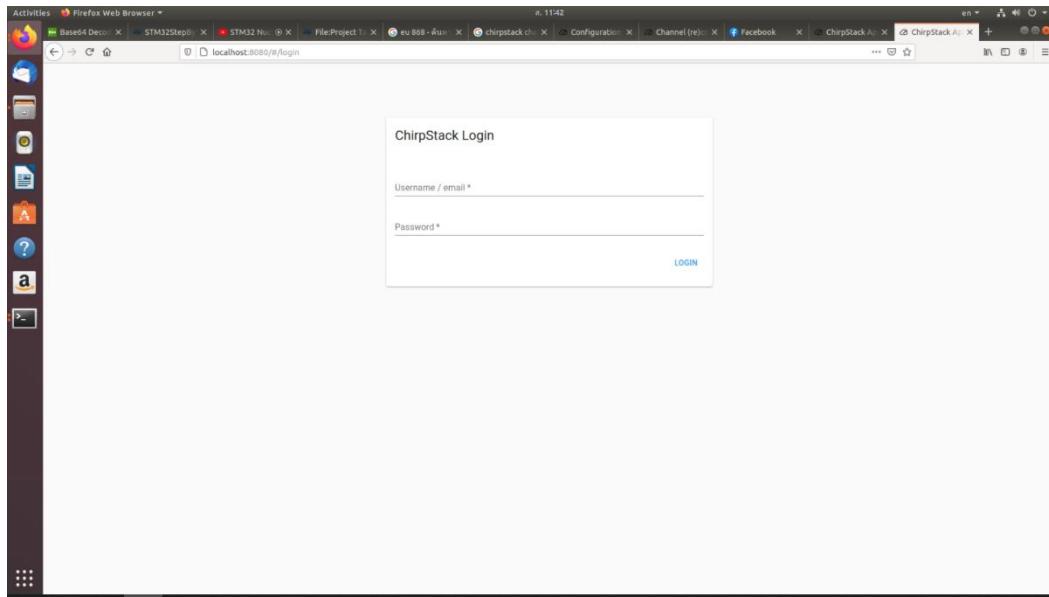
[join_server.default]
server="http://chirpstack-application-server:8003"

[geolocation_server]
server="chirpstack-geolocation-server:8005"

```

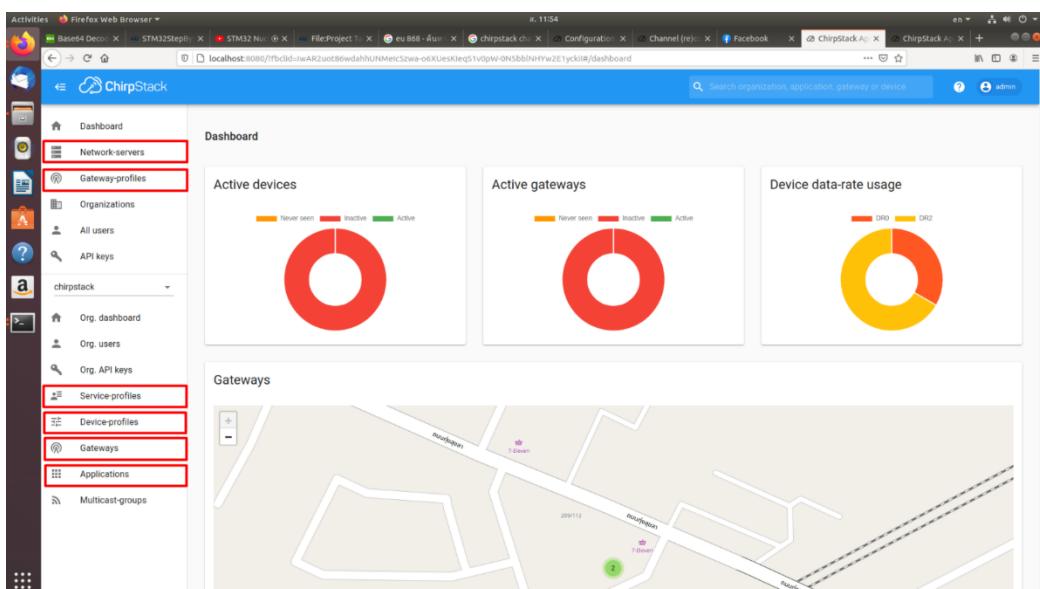
รูปภาพ 60 ตั้งค่าความถี่

4. เข้าไปที่โฟลเดอร์ chirpsatck-docker และใช้ command sudo docker-compose up เพื่อเรียกใช้ server chirpstack ให้ Chirpstack Application ทำงานจากนั้นเข้าไปที่ <http://localhost:8080/>



รูปภาพ 61 login เพื่อเข้าไปตั้งค่าต่างๆ

5. ทำการตั้งค่าในส่วนต่างๆ



รูปภาพ 62 ตั้งค่าในส่วนต่างๆ ภายใน Web application ของ Chirp stack

5.1. ทำการ Add Network-server

The screenshot shows the ChirpStack application's Network-servers page. On the left is a sidebar with various navigation options like Dashboard, Network-servers, Gateway-profiles, etc. The main area displays a table with one row: 'ns1' and 'chirpstack network-server:8000'. In the top right corner of the main area, there is a red-bordered button labeled '+ ADD'.

รูปภาพ 63 กด ADD เพื่อทำการเพิ่ม Network-server

จากนั้นเมื่อใส่ข้อมูลครบแล้วกดไปที่ UPDATE NETWORK-SERVER

The screenshot shows the ChirpStack application's Network-server configuration page for 'ns1'. The page has tabs for GENERAL, GATEWAY DISCOVERY, and TLS CERTIFICATES. Under the GENERAL tab, there are fields for 'Network-server name' (set to 'ns1') and 'Network-server server' (set to 'chirpstack-network-server:8000'). At the bottom right of the form, there is a blue-bordered button labeled 'UPDATE NETWORK SERVER' with an orange arrow pointing to it.

รูปภาพ 64 กด UPDATE NETWORK-SERVER หลังจากใส่ข้อมูลครบ

5.2. ทำการ Create Gateway-profiles

The screenshot shows the ChirpStack web interface. On the left is a sidebar with various navigation options like Dashboard, Network-servers, and Gateway-profiles. The main area is titled 'Gateway-profiles' and shows a single entry: 'gw1' under 'Name' and 'ns1' under 'Network-server'. In the top right corner of this table, there is a red box around the '+ CREATE' button.

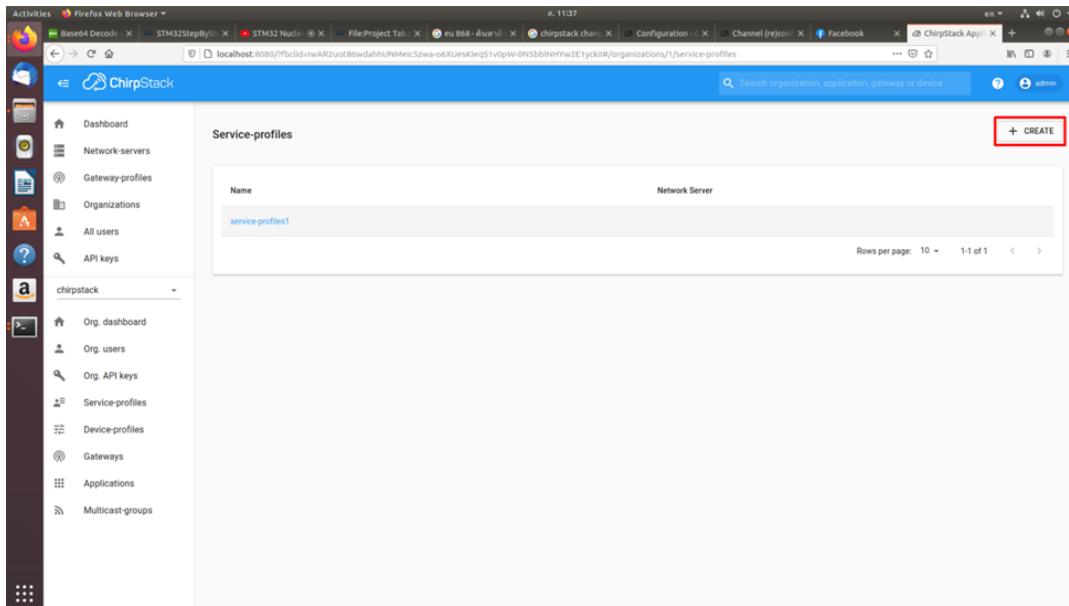
รูปภาพ 65 สร้าง Gateway-profiles

จากนั้นเมื่อใส่ข้อมูลครบแล้วกดไปที่ UPDATE GATEWAY-PROFILE

The screenshot shows the ChirpStack web interface with the URL 'localhost:8080/gateway-profiles/gw1'. The left sidebar shows the 'Gateway-profiles' option selected. The main area is titled 'Gateway-profiles / gw1' and contains several configuration fields: 'Name * gw1', 'Stats interval (seconds) * 20', and 'Enabled channels * 0, 1, 2, 3'. At the bottom right, there is a blue button labeled 'UPDATE GATEWAY-PROFILE' with an orange arrow pointing towards it.

รูปภาพ 66 กด UPDATE GATEWAY-PROFILE หลังจากใส่ข้อมูลครบ

5.3. Create Service-profiles



รูปภาพ 67 สร้าง Service-profiles

จากนั้นเมื่อใส่ข้อมูลครบแล้วกดไปที่ CREATE SERVICE-PROFILE

รูปภาพ 68 กด CREATE SERVICE- PROFILE เมื่อใส่ข้อมูลครบ

5.4. Create Device-profiles

The screenshot shows the ChirpStack Application interface with the URL `localhost:8080/#/organizations/1/device-profiles`. On the left is a sidebar with various navigation options like Dashboard, Network-servers, Gateway-profiles, Organizations, All users, API keys, and chirpstack. The main area is titled 'Device-profiles' and shows a single entry: 'device-profile1' under 'Name' and 'ns1' under 'Network Server'. At the top right of this table is a red-bordered '+ CREATE' button.

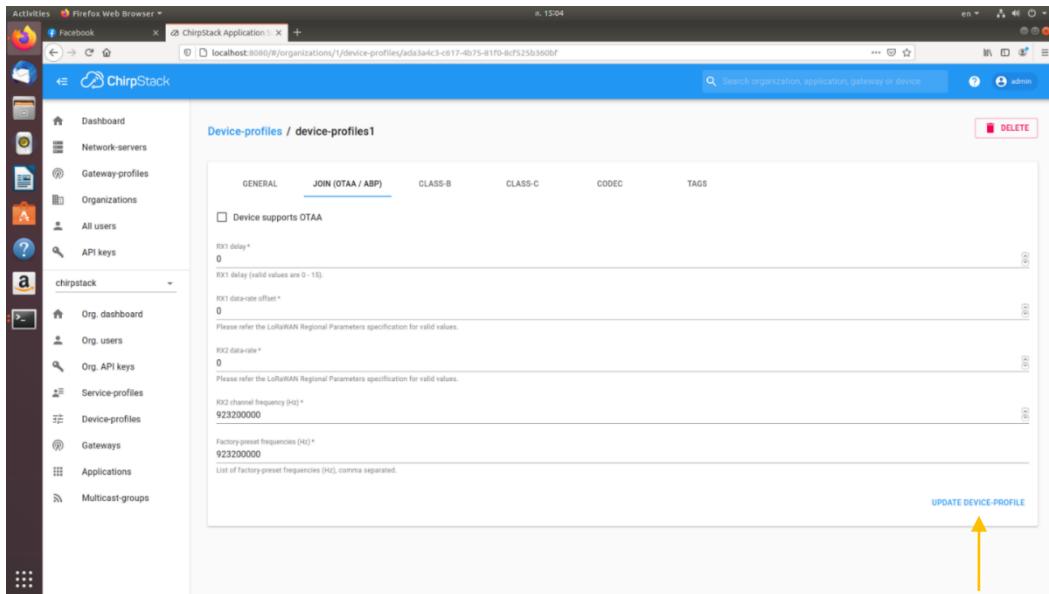
รูปภาพ 69 กดสร้าง Device-profiles

ใส่ข้อมูลครบแล้วจากนั้นกดไปที่ UPDATE DEVICE-PROFILE

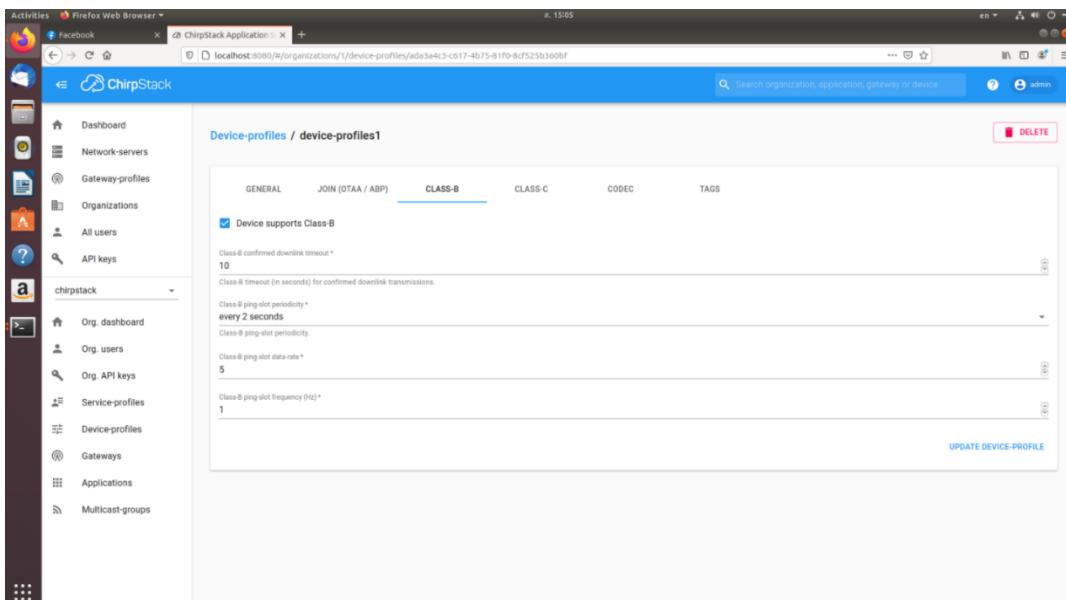
The screenshot shows the ChirpStack Application interface with the URL `localhost:8080/#/organizations/1/device-profiles/1/device-profiles1`. The sidebar and main navigation are identical to the previous screenshot. The main area is titled 'Device-profiles / device-profiles1' and contains several tabs: GENERAL, JOIN (OTAA / ABP), CLASS-B, CLASS-C, CODEC, and TAGS. Under the GENERAL tab, there are fields for 'Device-profile name' (set to 'device-profile1'), 'LoRaWAN MAC version' (set to '1.0.2'), 'LoRaWAN Regional Parameter revision' (set to 'B'), 'ADR algorithm' (set to 'Default ADR algorithm'), 'Max ERIP' (set to '0'), and 'Uplink interval (seconds)' (set to '10'). At the bottom right of this form is a red-bordered 'UPDATE DEVICE-PROFILE' button, which has an orange arrow pointing to it from below.

รูปภาพ 70 กด UPDATE DEVICE-PROFILE เมื่อใส่ข้อมูลครบ

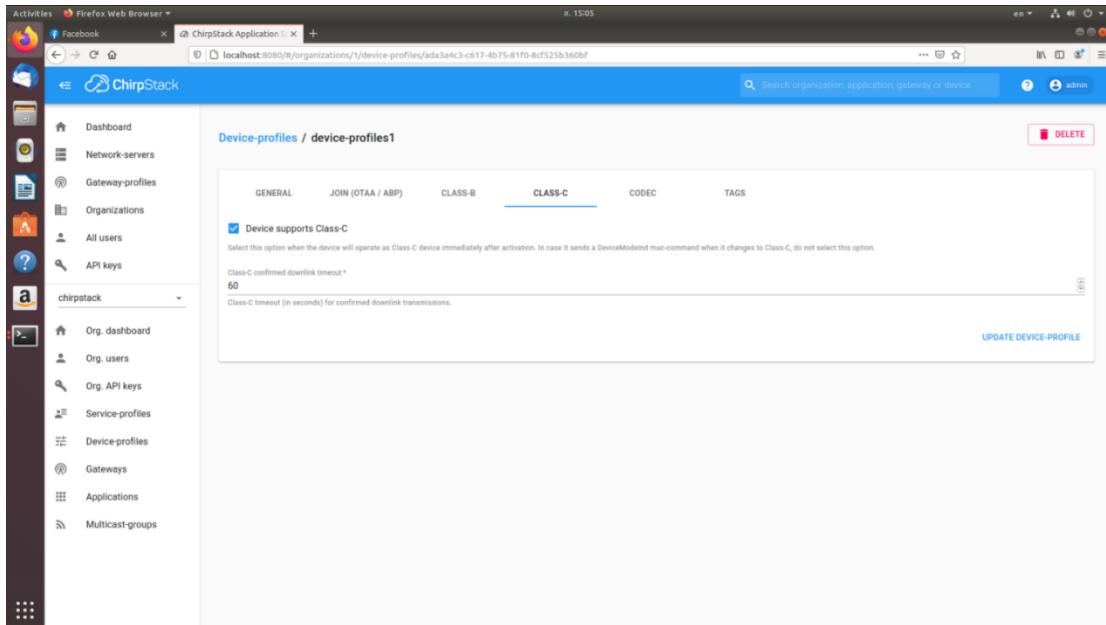
ใส่ข้อมูลครบแล้วจากนั้นกดไปที่ UPDATE DEVICE-PROFILE



รูปภาพ 71 กด UPDATE DEVICE-PROFILE เมื่อใส่ข้อมูลครบ

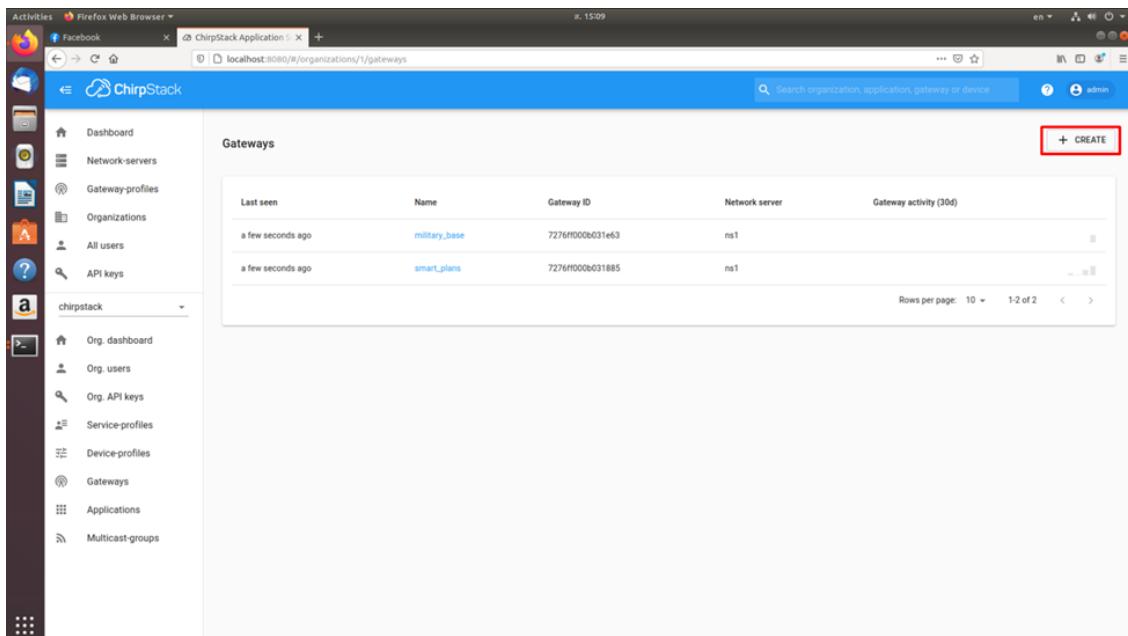


รูปภาพ 72 กด UPDATE DEVICE-PROFILE เมื่อใส่ข้อมูลครบ



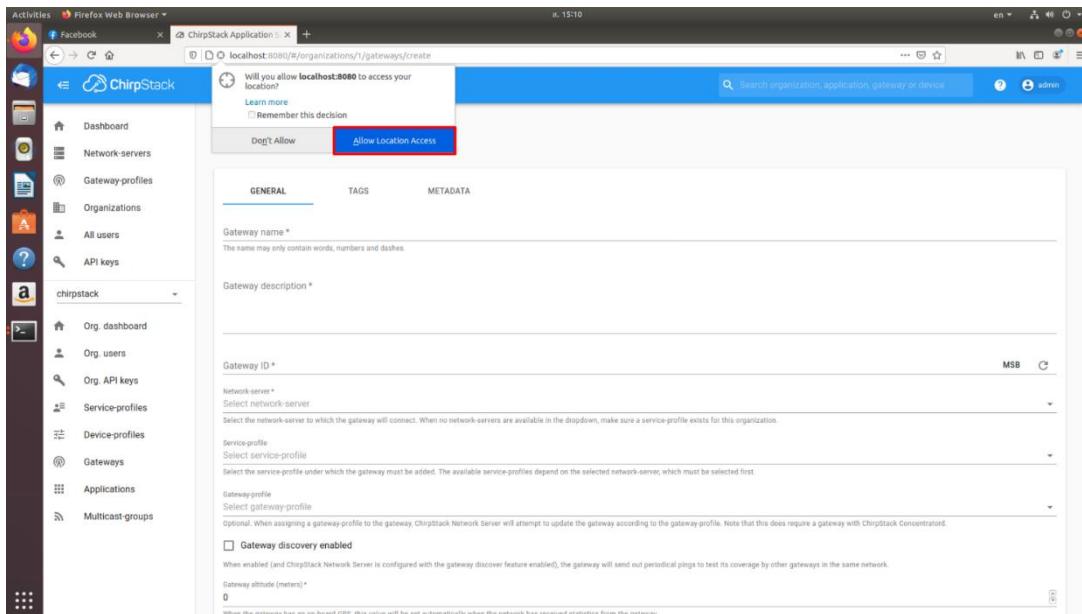
รูปภาพ 73 กด UPDATE DEVICE-PROFILE เมื่อใส่ข้อมูลครบ

5.5. Create Gateways

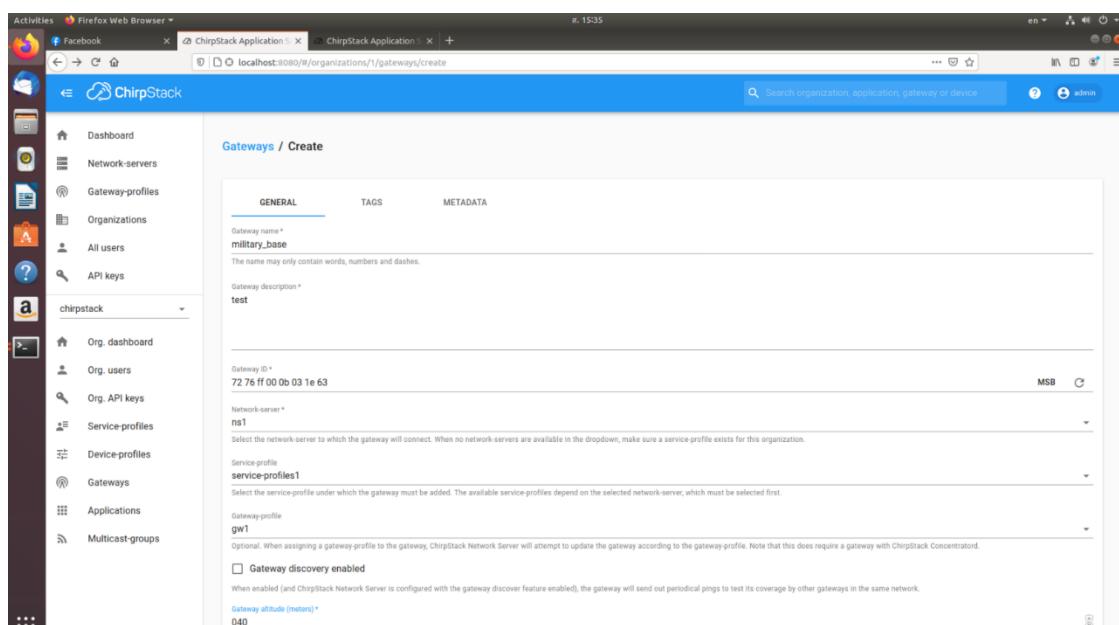


รูปภาพ 74 กดสร้าง Gateways

กดที่ Allow Location Access

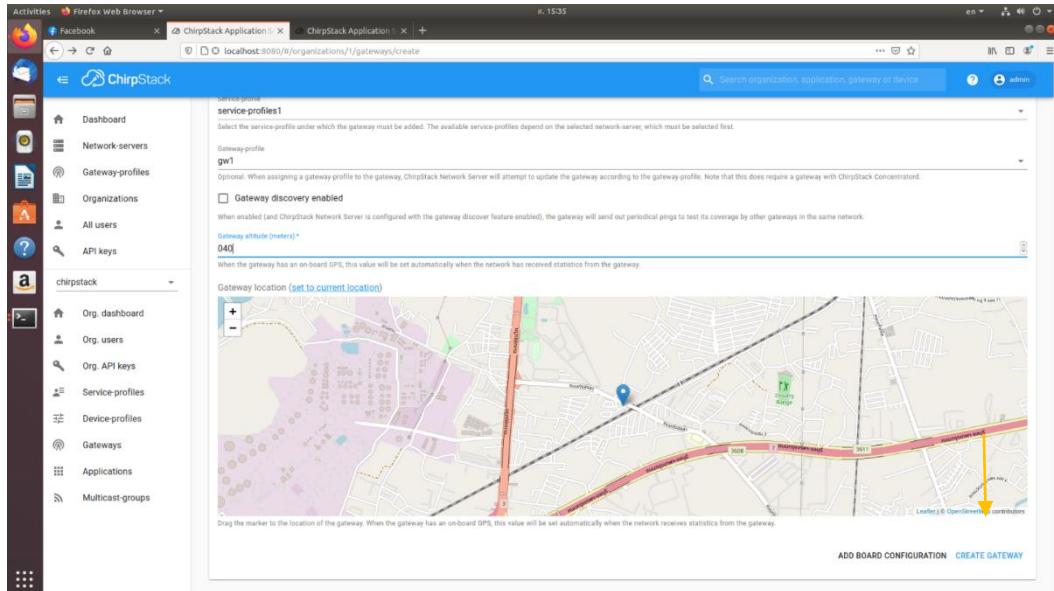


ຮັບການ 75 ກວດອຸນ່ມາຕໍ່ທີ່ເຂົ້າຄື່ງຕໍ່ແຫ່ນໆ



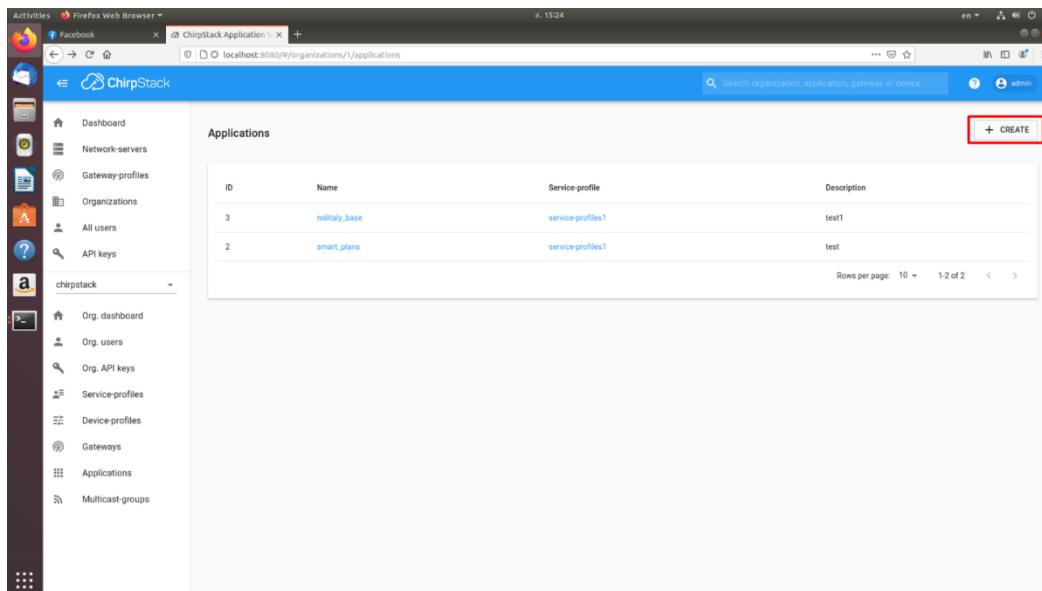
ຮັບການ 76 ໄລ້ຂໍ້ມູນໃຫ້ຮຽນ

ใส่ข้อมูลครบแล้วจากนั้นกดไปที่ CREATE GATEWAY



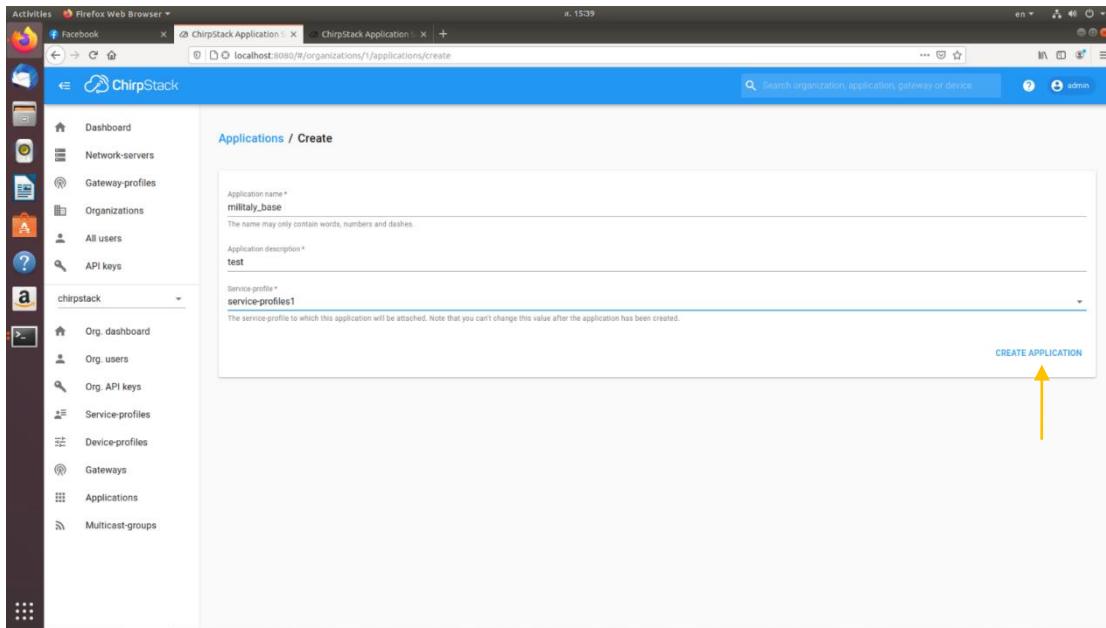
รูปภาพ 77 กด CREATE GATEWAY เมื่อใส่ข้อมูลครบ

5.6. Create Application



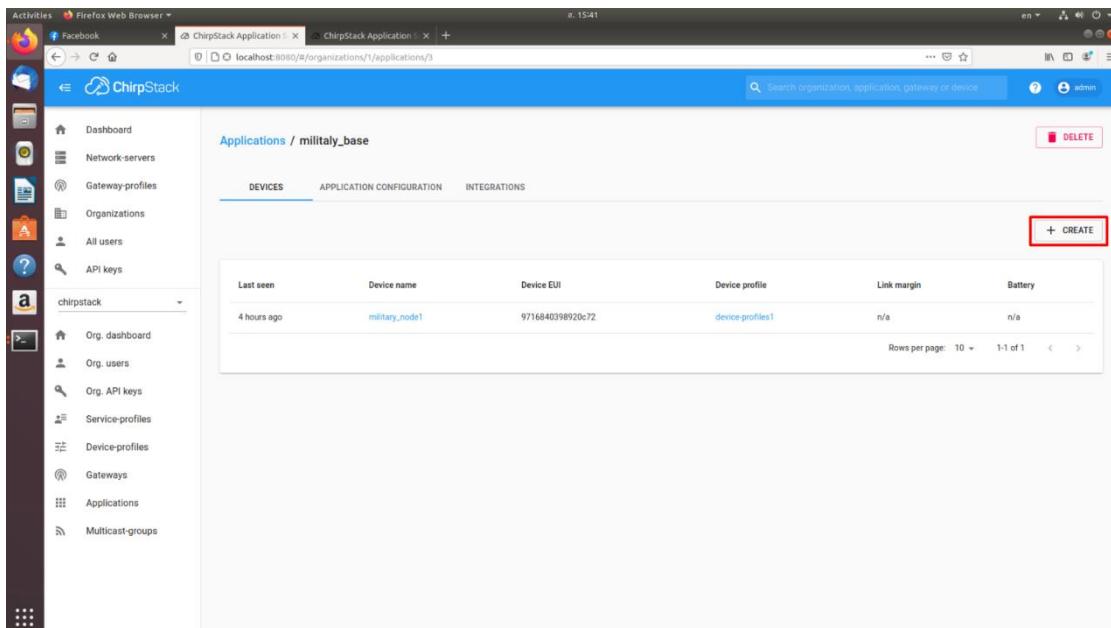
รูปภาพ 78 กดสร้าง Application

ใส่ข้อมูลครบแล้วจากนั้นกดไปที่ CREATE APPLICATION



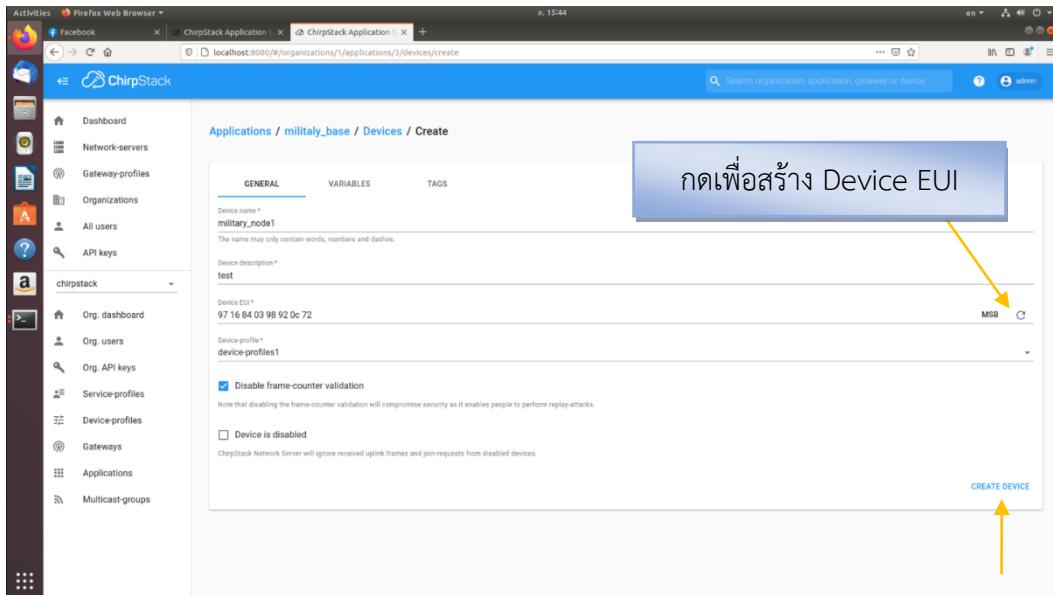
รูปภาพ 79 กด CREATE APPLICATION เมื่อใส่ข้อมูลครบ

5.7. Create Device



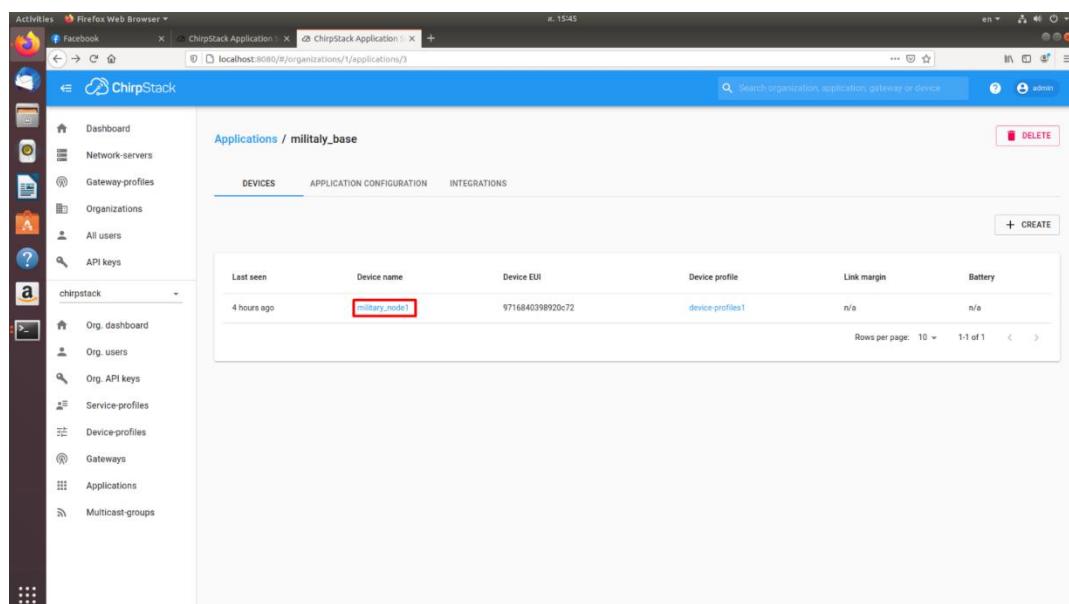
รูปภาพ 80 กดสร้าง Device

เมื่อใส่ข้อมูลครบแล้วจากนั้นกดไปที่ CREATE DEVICE



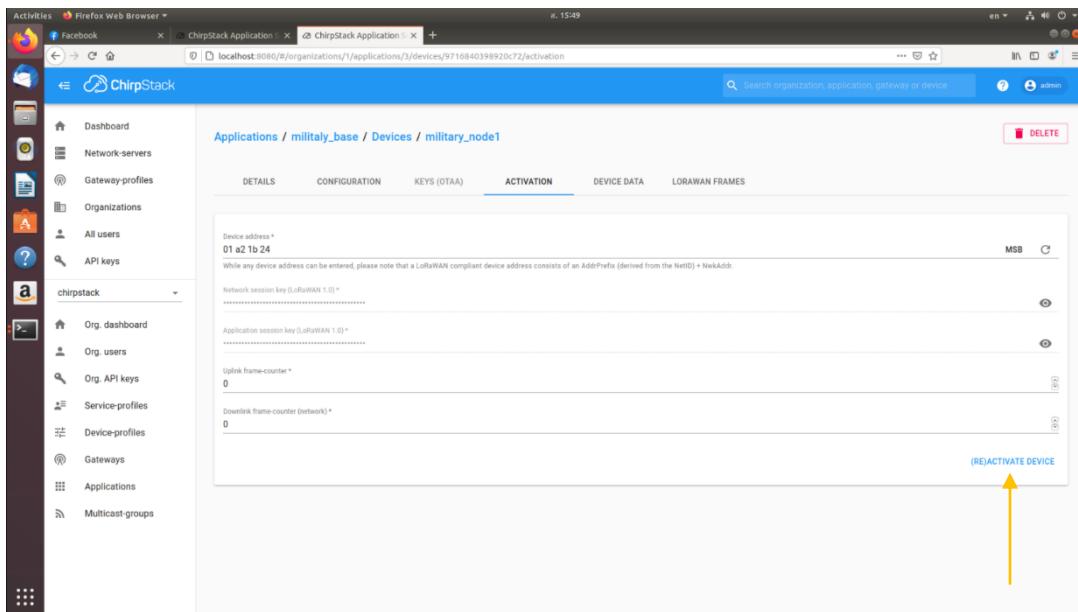
รูปภาพ 81 กด CREATE DEVICE เมื่อใส่ข้อมูลครบ

กดไปที่ Device ที่เราเพิ่งสร้าง



รูปภาพ 82 กด Device ที่สร้างไว้

จากนั้นเมื่อใส่ข้อมูลครบแล้วกดไปที่ (RE)ACTIVATE DEVICE



รูปภาพ 83 กด (RE)ACTIVATE DEVICE เมื่อใส่ข้อมูลครบ

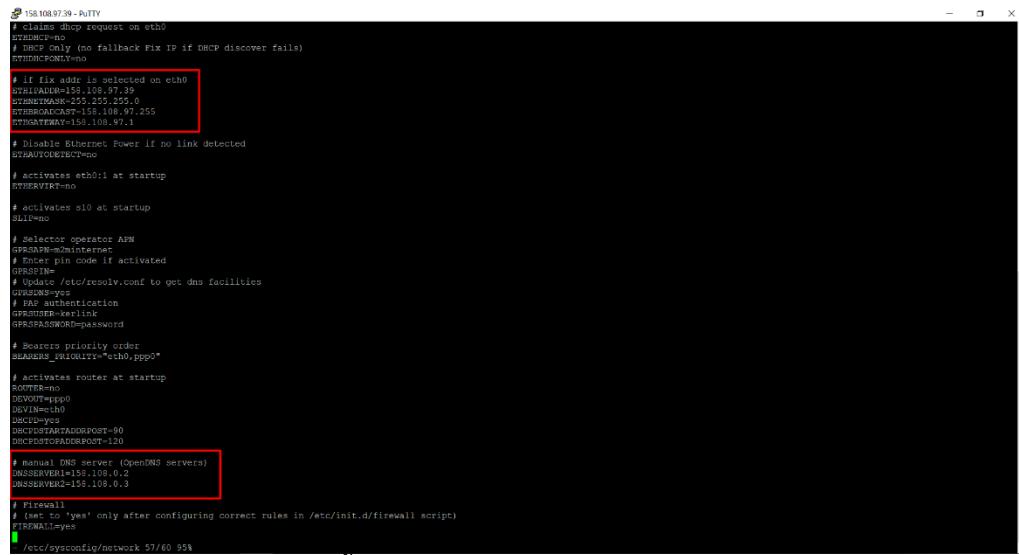
6. Setup Kerlink Wirnet Station Gateway

เราใช้ Kerlink Wirnet Station Gateway <https://www.kerlink.com/wp-content/uploads/2020/10/Whitepaper-Outdoor-LoRaWAN-Gateways-Introduction-to-radio-coexistence-issues-and-solutions.pdf> (เอกสารอ้างอิง)
วิธีการตั้งค่าโดยอ้างอิงจาก <https://github.com/dandavid3000/KerlinkLoRaLoTStationToTTN>



รูปภาพ 84 Setup Kerlink Wirnet Station Gateway

6.1. เสียบสาย UTP เข้าที่คอมพิวเตอร์และ Kerlink Wirnet Station Gateway จากนั้นเข้าไปตั้งค่า IP Address ให้เป็นแบบ Static โดยเข้าผ่าน SSH ผ่าน Default IP 192.168.4.155 และใช้คำสั่ง vi /etc/sysconfig/network



```
# Claims DHCP request on eth0
#DHCP=auto
# DHCP Only (no fallback Fix IP if DHCP discover fails)
#DHCPFPONLY=no

# If fix addr is selected on eth0
#DHCPADDR=192.168.97.39
#DHCPMASK=255.255.255.0
#DHCPBROADCAST=192.168.97.255
#DHCPSTATION=192.168.97.1

# Disable Ethernet Power if no link detected
#ETHERAUTOPWRDOWN=no

# activates eth0:1 at startup
#ETHERVIRT=no

# activates s10 at startup
#SLLW=no

# Selects operator APN
#APN=operator/internet
# Enter pin code if activated
#GPRSPIN=
# Update /etc/resolv.conf to get dns facilities
#DNS=auto
# TAP authentication
#GRPSERVER=kerlink
#GRPSPASSWORD=password

# Sets priority order
#MANAGERS_PRIORITY="eth0_pppo"
# Activates router at startup
#ROUTER=no
#DEVOUT=ppp0
#DEVIN=eth0
#DEVIP=192.168.97.2
#DEVMTU=1500
#DEVMTUADJPOST=90
#DEVCEDSTADJPOST=120

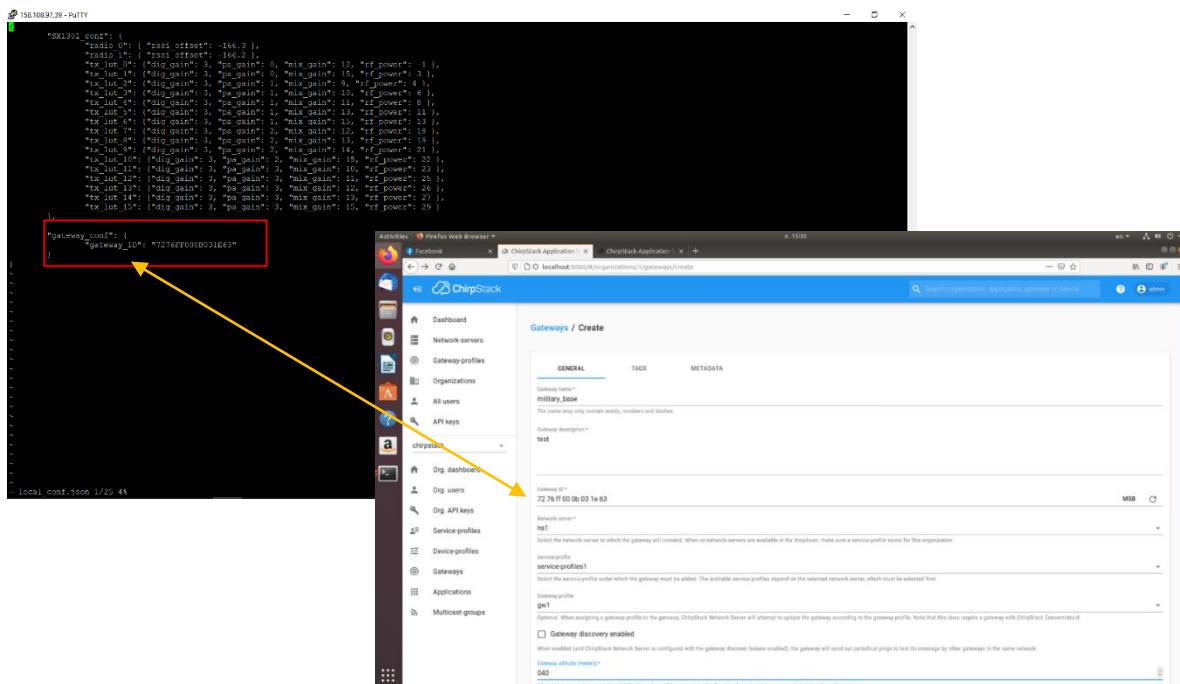
# Manually DNS server (OpenDNS servers)
#DNSERVER1=153.109.0.2
#DNSERVER2=153.109.0.3

# Firewall
# (Ask to 'yes' only after configuring correct rules in /etc/init.d/firewall script)
#FIREWALL=yes

/etc/sysconfig/network 57/60 95%
```

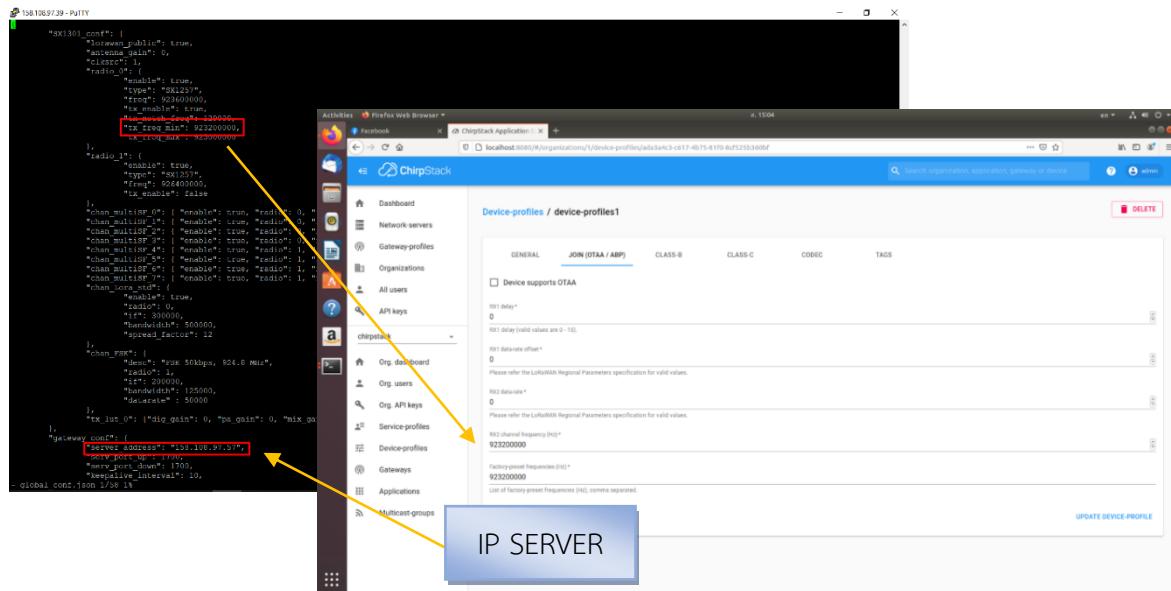
รูปภาพ 85 ตั้งค่า IP Address ให้เป็นแบบ Static โดยเข้าผ่าน SSH

6.2. เข้าไปแก้ไขไฟล์โดยคำสั่ง vi /mnt/fsuser-1/spf/etc/local_conf.json และแก้ไข Gateway_ID ให้ตรงกับ Gateway ที่ Chirpstack Web application ที่เคยสร้างไว้ก่อนหน้า



รูปภาพ 86 ตั้งค่า Gateway ID ให้ตรงกับ Web application ของ Chirpstack

6.3. ไปที่ไฟล์เดอ cd /mnt/fsuser-1/spf/etc/ เพื่อแก้ไขไฟล์ global_conf.json ให้เป็น AS923 และตั้งค่าความถี่โดยจะต้องตรงกับ ChirpStack Web Applications ทั้ง 2 ค่าและ Server_address เป็น IP Server



รูปภาพ 87 ตั้งค่าความถี่ให้ตรงกับ Web application ของ Chirpstack

เพิ่มเติม เมื่อมีการส่งค่าผ่าน Kerlink Wirnet Station Gateway สามารถดู Log ได้จาก tail -f /mnt/fsuser-1/spf/var/log/spf.log

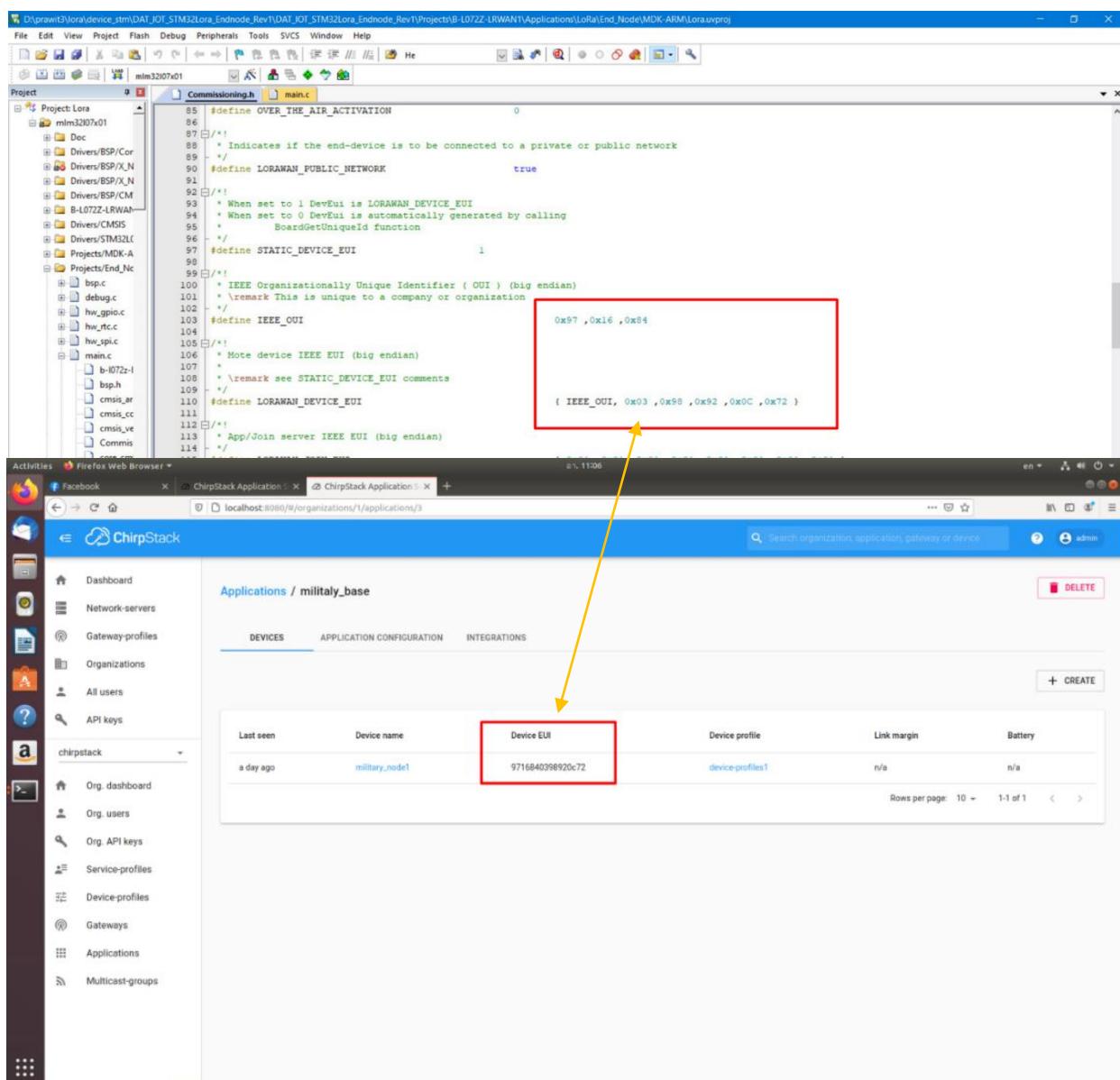
```
[root@Wirnet_0B031E65 etc]# tail -f /mnt/fsuser-1/spf/var/log/spf.log
Feb 27 14:51:06 Wirnet local.notice spt: # BEACON rejected: 0
Feb 27 14:51:08 Wirnet local.notice spt: ## [JITI] ####
Feb 27 14:51:08 Wirnet local.notice spt: # Invalid time reference: age: 1614437468 sec
Feb 27 14:51:08 Wirnet local.notice spt: # No valid GPS coordinates available yet
Feb 27 14:51:08 Wirnet local.notice spt: ##### END #####
Feb 27 14:51:08 Wirnet local.notice spt: JSON up: {"stat":{"time":"2021-02-27 14:51:08 GMT","rxnb":0,"rxok":0,"rxfw":0,"ackr":100.0,"dwnb":0,"txnb":0,"ping":1}}
Feb 27 14:51:10 Wirnet local.notice spt: INFO: [down] PULL ACK received in 1.01 ms
Feb 27 14:51:15 Wirnet local.notice spt: INFO: [down] PULL ACK received in 1.01 ms
Feb 27 14:51:25 Wirnet local.notice spt: INFO: [down] PULL ACK received in 0.98 ms
Feb 27 14:51:35 Wirnet local.notice spt: INFO: [down] PULL ACK received in 0.98 ms
Feb 27 14:51:38 Wirnet local.notice spt: ##### 2021-02-27 14:51:38 GM #####
Feb 27 14:51:38 Wirnet local.notice spt: ## [UPSTREAM] ####
Feb 27 14:51:38 Wirnet local.notice spt: RF packets received by concentrator: 1
Feb 27 14:51:38 Wirnet local.notice spt: # CRC errors: 0, CRC FAIL: 100.0%, NO_CRC: 0.0%
Feb 27 14:51:38 Wirnet local.notice spt: RF packets forwarded: 0 (0 bytes)
Feb 27 14:51:38 Wirnet local.notice spt: PUSH DATA datagrams sent: 1 (122 bytes)
Feb 27 14:51:38 Wirnet local.notice spt: # PUSH DATA acknowledged: 100.0%
Feb 27 14:51:38 Wirnet local.notice spt: ##### [DOWNSTREAM] #####
Feb 27 14:51:38 Wirnet local.notice spt: # FULL DATA sent: 3 (100.0% acknowledged, ping 0.98 ms)
Feb 27 14:51:38 Wirnet local.notice spt: # RF packets sent to concentrator: 0 (0 bytes)
Feb 27 14:51:38 Wirnet local.notice spt: # TX errors: 0
Feb 27 14:51:38 Wirnet local.notice spt: # BEACON queued: 0
Feb 27 14:51:38 Wirnet local.notice spt: # BEACON sent so far: 0
Feb 27 14:51:38 Wirnet local.notice spt: ## [JITI] ####
Feb 27 14:51:38 Wirnet local.notice spt: # Invalid time reference: age: 1614437498 sec
Feb 27 14:51:38 Wirnet local.notice spt: # No valid GPS coordinates available yet
Feb 27 14:51:38 Wirnet local.notice spt: ##### END #####
Feb 27 14:51:38 Wirnet local.notice spt: JSON up: {"stat":{"time":"2021-02-27 14:51:38 GMT","rxnb":1,"rxok":0,"rxfw":0,"ackr":100.0,"dwnb":0,"txnb":0,"ping":1}}
Feb 27 14:51:38 Wirnet local.notice spt: INFO: [up] PUSH ACK received in 0 ms
```

รูปภาพ 88 คำสั่งดู Log เมื่อมีค่าส่งผ่าน Kerlink Wirnet Station Gateway

7. Setup B-L072Z-LRWAN1 Device (End node) เพื่อที่จะให้อุปกรณ์สื่อสารกับ Chirpstack application server โดยจะต้องกำหนดค่า Device EUI, Device Address, Network Session, Application session key ให้เหมือนกับใน Chirpstack application server ซึ่งไฟล์ที่จะเข้าไปแก้ไขจะอยู่ที่ Project/End_Node Code:

<https://drive.google.com/file/d/1T9y70psbDs6A79qzLv8pshPNMOeOMO5/view?usp=sharing>

Device EUI



รูปภาพ 89 Setup B-L072Z-LRWAN1 Device เพื่อที่จะให้อุปกรณ์สื่อสารกับ Chirpstack application server

Device Address

The screenshot illustrates the configuration of a device address in a LoRaWAN application. In the top window, a C code editor shows the definition of the device address:

```

127 #define LORAWAN_NWK_KEY { 0x2B, 0x7E, 0x15, 0x16, 0x2B, 0xE6, 0xD3, 0xA6, 0xAB, 0xF7, 0x15, 0x66, 0x09, 0xCF, 0x9F, 0xA3 }
128 #if( OVER_THE_AIR_ACTIVATION == 0 )
129     * Current network ID
130     */
131     + Current network ID
132     */
133     #define LORAWAN_NETWORK_ID ( uint32_t )0
134
135     */
136     * When set to 0 DevAddr is automatically generated using
137     * a pseudo random generator seeded with a value derived from
138     * BoardUniqueId value
139     */
140     #define STATIC_DEVICE_ADDRESS
141     */
142     + Device address on the network (big endian)
143     */
144     + remark see STATIC_DEVICE_ADDRESS comments
145     */
146     + BoardUniqueId
147     */
148     #define LORAWAN_DEVICE_ADDRESS
149     */
150
151 //const char *devAddr = "260110AD";
152 //const char *nwkSKey = "7AEAF43A8E88AB11C327C07E491FA6A";
153 //const char *appSKey = "6E8533E9A3B8395A3D7194B309BB17C0";
154
155 */
156     + Forwarding Network session integrity key
157     + WARNING: NWK_S_KEY FOR 1.0.x DEVICES
158     */
159     #define LORAWAN_F_NWK_S_INT_KEY
160

```

A red box highlights the line `(uint32_t)0x1a21b24`, which is the device address being defined.

An arrow points from this highlighted line in the code editor to the corresponding field in the ChirpStack Application Management interface below:

ChirpStack Application Management Interface (localhost:8080):

- Left sidebar: Applications / military_base / Devices / military_node1
- Details tab: Device address * **01 a2 1b 24** (highlighted in red)
- Activation tab: Network session key (LoRaWAN 1.0) * df ef 93 f5 c4 ac 17 76 cc 63 0c 5b 48 8d 9f 1a (highlighted in red)
- Activation tab: Application session key (LoRaWAN 1.0) * 20 47 31 c6 be 11 3b 34 9a 58 58 56 30 55 4c 47 (highlighted in red)
- Activation tab: Uplink frame-counter * 1402
- Activation tab: Downlink frame-counter (network) * 3315
- Bottom right: (REACTIVATE DEVICE)

ສັບກາງ 90 ເພີ່ມ Device

Network Session and Application session key

The screenshot displays two main windows. The top window is a code editor showing a C file named `Commissioning.h`. It contains several defines for LORAWAN keys:

```

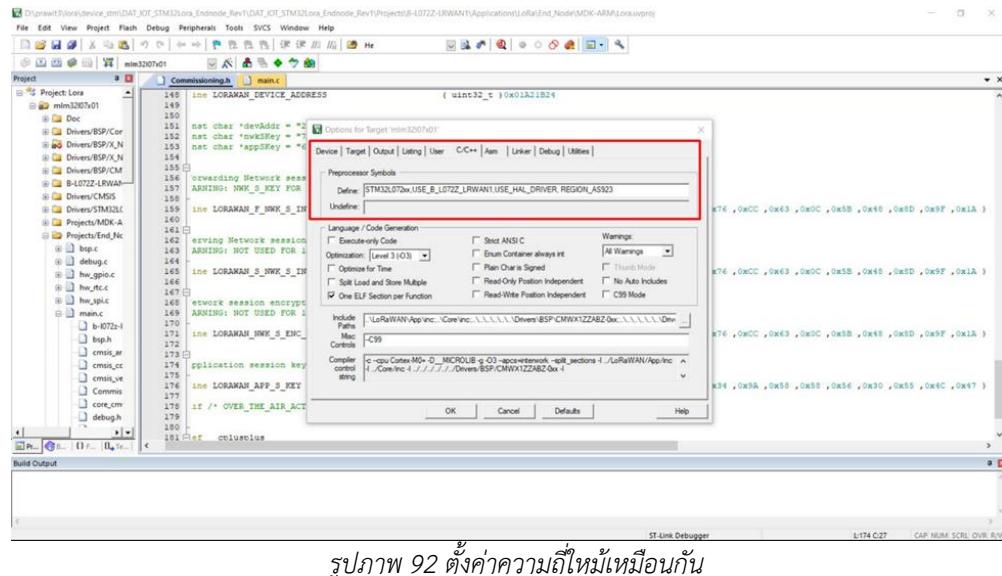
148 #define LORAWAN_DEVICE_ADDRESS (0x01a21b24)
149
150
151 #define devAddr "+06011030"
152 #define nwkKey "77EAF445A9E88AB11C327C07E4D1F46A"
153 #define appKey "E68533E9A3B0395A3D7194B3D90B17C0"
154
155
156 #define NWK_S_KEY 0x01a21b24
157 #define APP_S_KEY 0x01a21b24
158
159 #define LORAWAN_F_NWK_S_INT_KEY 0x01a21b24
160
161 #define LORAWAN_F_NWK_S_ENC_KEY 0x01a21b24
162
163 #define LORAWAN_APP_S_KEY 0x01a21b24
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181

```

The bottom window is a web browser showing the ChirpStack application interface. It is displaying the activation details for a device with address `01 a2 1b 24`. The Network session key (Lorawan 1.0) is listed as `df e1 93 f5 c4 ac 17 76 cc 63 0c 5b 48 8d 9f 1a`, and the Application session key (Lorawan 1.0) is listed as `20 47 31 c6 be 11 3b 34 9a 58 56 30 55 4c 47`.

รูปภาพ 91 เพิ่มค่า Network Session and Application Session key

หลังจากนั้นกดที่ Options for target และเลือกที่ C/C++ เพื่อที่จะเปลี่ยนการใช้งานความถี่ให้เหมือนกัน



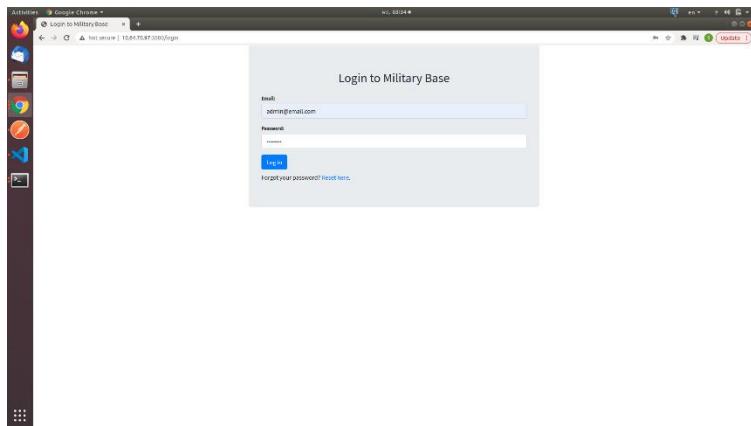
รูปภาพ 92 ตั้งค่าความถี่ใหม่เหมือนกัน

ก2

คู่มือการใช้งาน

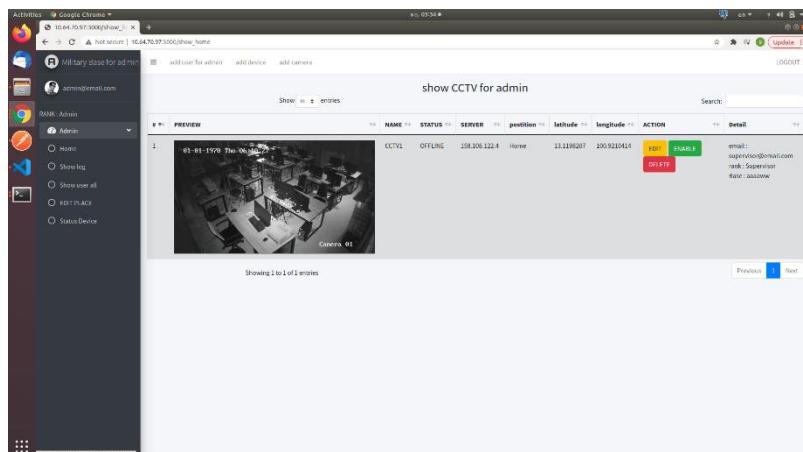
สำหรับ web application เข้าที่ เว็บไซต์ 10.64.70.97:3000/login

จะพบกับหน้านี้ เราจะเริ่ม login ที่ระดับ Admin



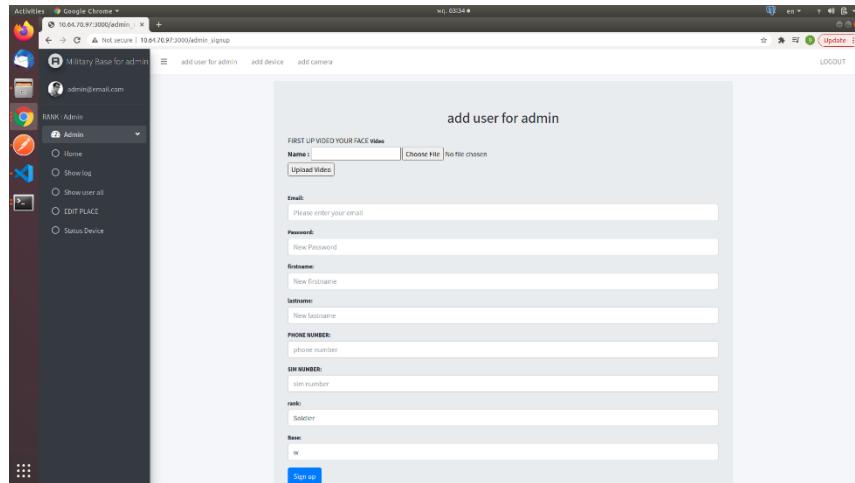
รูปภาพ 93 หน้า login

เมื่อ login เสร็จแล้ว จะพบกับหน้านี้

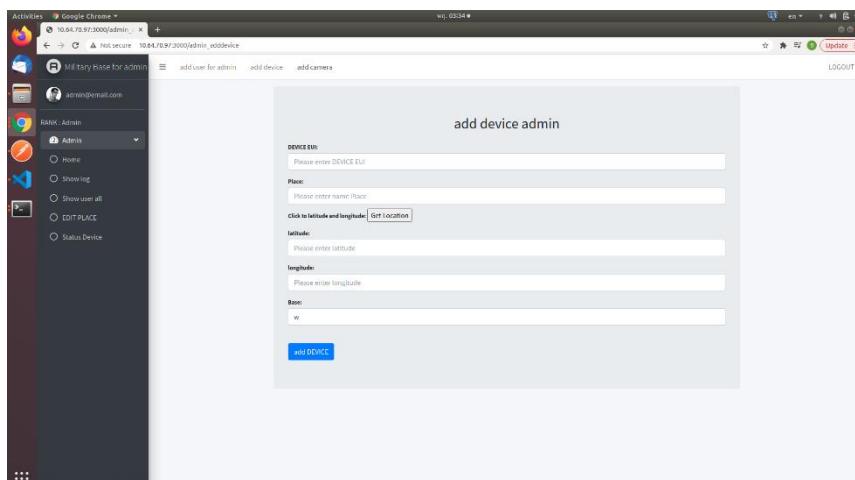


รูปภาพ 94 หน้าแรกของ Admin และ Supervisor

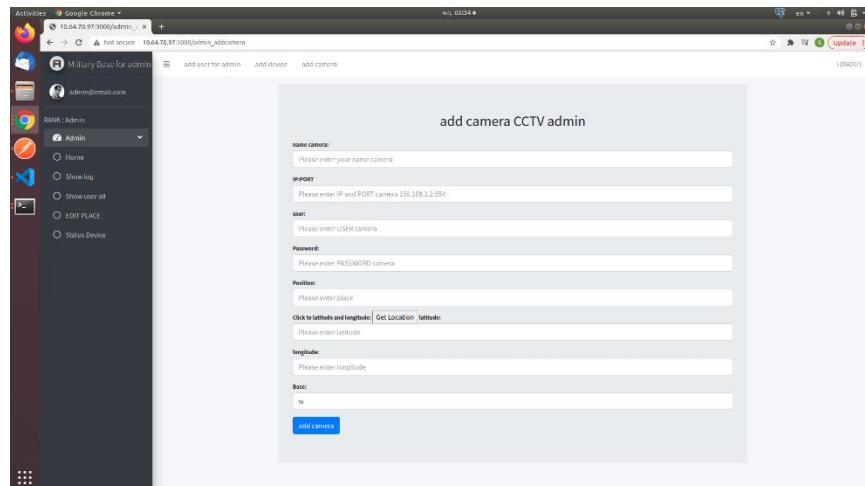
เราจะเริ่มที่ระบบ ADD ของ Admin และ Supervisor ก็จะมี เพิ่ม user ,เพิ่ม Device ,เพิ่ม CCTV



รูปภาพ 95 เพิ่ม user

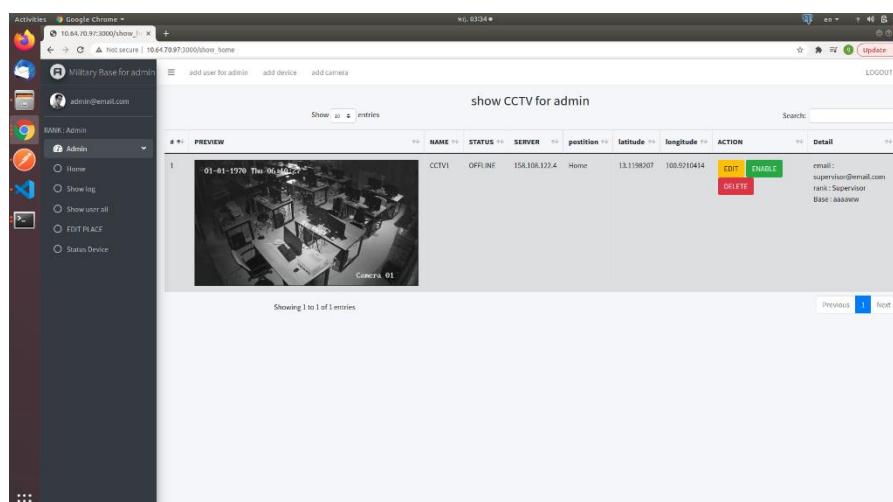


รูปภาพ 96 เพิ่ม device



รูปภาพ 97 เพิ่มกล้อง CCTV

ต่อมาหน้า home ของ Admin และ Supervisor จะสามารถ แก้ไข และ ลบ และ เปิดปิด การใช้งานของกล้องได้



รูปภาพ 98 หน้า home ของ Admin และ Supervisor

EDIT CCTV CCTV1

name camera:
CCTV1

IP/PORT:
192.168.1.234

user:
admin

password:
aaaaaww

position:
Home

Click to latitude and longitude: Get Location | latitude:
13.12392027

longitude:
100.9210414

base:
aaaaaww

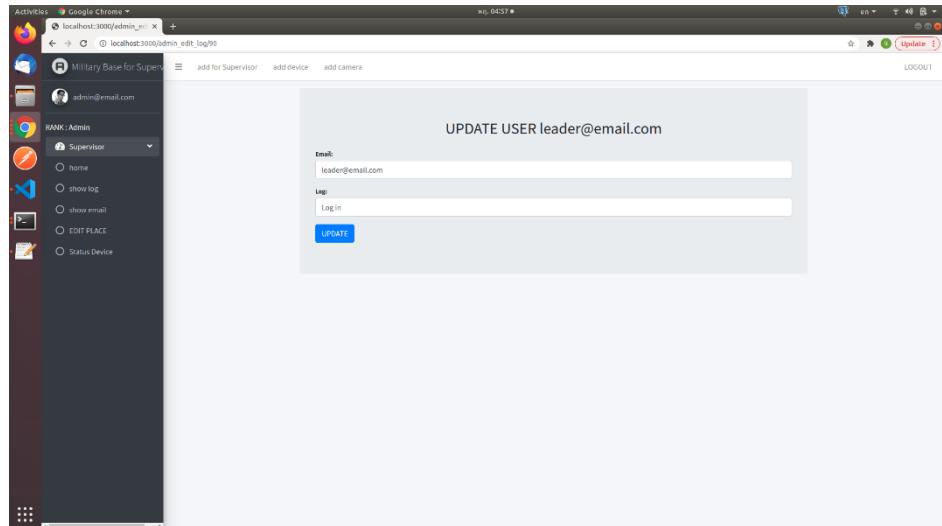
รูปภาพ 99 แก้ไข กล้อง CCTV

ต่อมา แบบง่าย Show log Admin และ Supervisor จะสามารถแก้ไข email และ กิจกรรมของ log และลบได้

show log

ID	Email	Time	Rank	Base	Log	Edit	Delete
1	soldier@email.com	Thu May 20 2021 02:42:13	Soldier	aaaaaww	Log in	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
2	soldier@email.com	Thu May 20 2021 02:42:21	Soldier	aaaaaww	ADD Device	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
3	admin@email.com	Thu May 20 2021 02:42:28	Admin	aaaaaww	Log in	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
4	soldier@email.com	Thu May 20 2021 02:43:00	Soldier	aaaaaww	Log in	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
5	admin@email.com	Thu May 20 2021 03:43:21	Admin	aaaaaww	Log in	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
6	leader@email.com	Thu May 20 2021 02:50:15	Leader	aaaaaww	Log in	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
7	admin@email.com	Thu May 20 2021 03:21:39	Admin	aaaaaww	Log in	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
8	admin@email.com	Thu May 20 2021 03:22:25	Admin	aaaaaww	Log in	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
9	admin@email.com	Thu May 20 2021 03:22:48	Admin	aaaaaww	Log Out	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
10	admin@email.com	Thu May 20 2021 03:22:51	Admin	aaaaaww	Log in	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

รูปภาพ 100 หน้าโชว์ log ของทุกคน



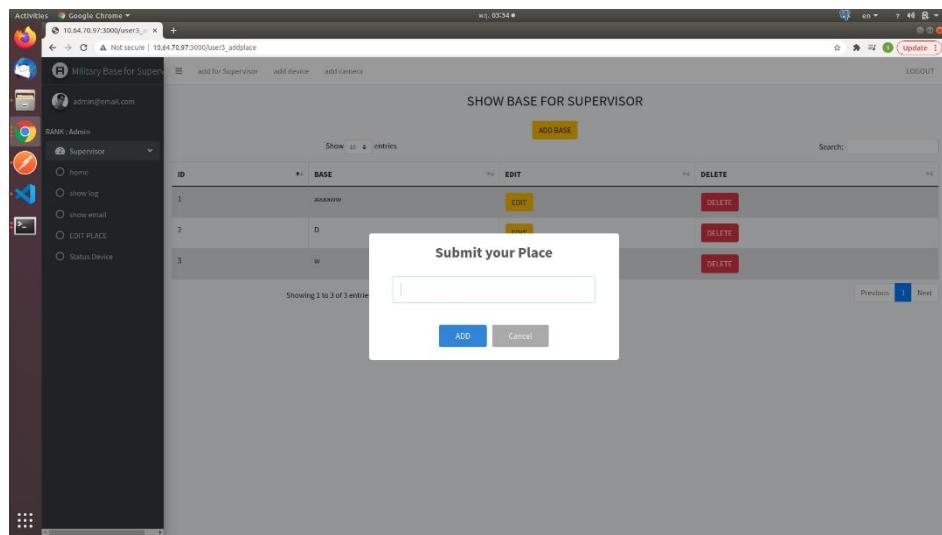
รูปภาพ 101 หน้าแก้ไข log ของ Admin และ Supervisor

ต่อมา จะเป็น หน้า สถานที่ Admin และ Supervisor สามารถ จัดการได้ โดยสามารถ Addและ Delete และ Edit ได้

ID	BASE	EDIT	DELETE
1	aaamewr	EDIT	DELETE
2	D	EDIT	DELETE
3	w	EDIT	DELETE

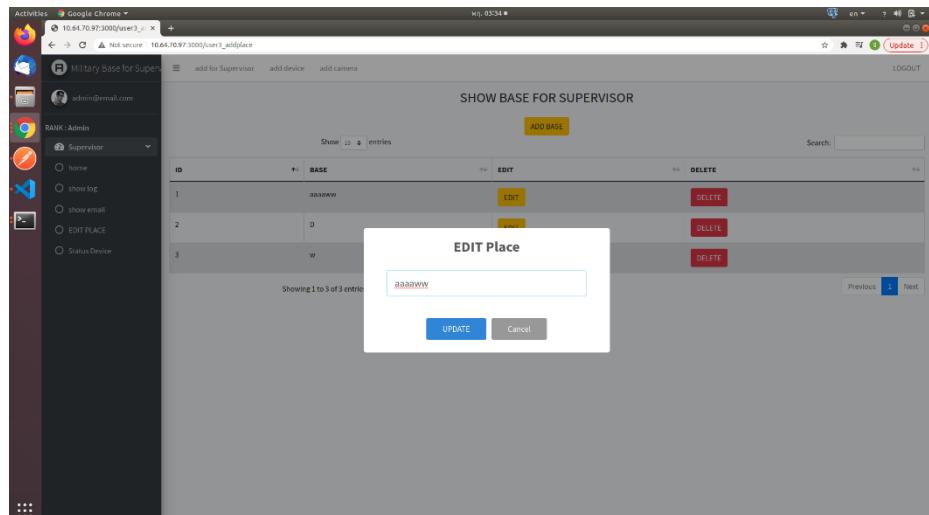
รูปภาพ 102 หน้าโชว์ Base ของ Admin และ Supervisor

ເພີ່ມ



ຮູບການ 103 ເພີ່ມສານທີ່

ແກ້ໄຂ



ຮູບການ 104 ແກ້ໄຂສານທີ່

ต่อมา จะเป็นหน้าระบบ Device Admin และ Supervisor สามารถ Edit, Delete, เปิด ปิด ได้

ID	DEVICE EUI	STATUS	PLACE	latitude	longitude	createdby	Time	edit	OFF	DELETE
1	12312	ONLINE	3213	13.84120319999999	100.5786992	email : soldier@email.com rank : Soldier Base : aaaaaw	Thu May 20 2021 02:42:53	<button>EDIT</button>	<button>OFF</button>	<button>DELETE</button>
2	34444444	ONLINE	4444	13.8575872	100.5786992	email : supervisor@email.com rank : Supervisor Base : w	Mon May 17 2021 07:22:20	<button>EDIT</button>	<button>OFF</button>	<button>DELETE</button>
3	7h0BUnZKdxQn	ONLINE	Home	13.85812919999999	100.5633786	email : adminn@email.com rank : Admin Base : D	Thu May 20 2021 03:35:05	<button>EDIT</button>	<button>OFF</button>	<button>DELETE</button>
4	123	OFFLINE	123	13.1198322	100.9210129	email : admin@email.com rank : Admin Base : w	Thu May 20 2021 02:31:33	<button>EDIT</button>	<button>ON</button>	<button>DELETE</button>

รูปภาพ 105 หน้าจอชื่อ Device ทุกตัว

แก้ไข

EDIT device 12312

DEVICE EUI:
12312

Place:
3213

Click to latitude and longitude:

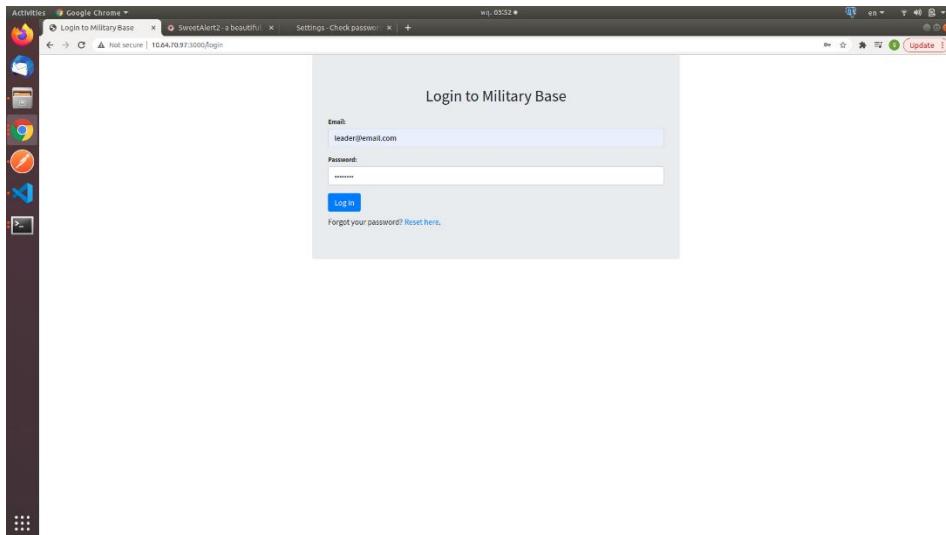
latitude:
13.84120319999999

longitude:
100.5786992

BASE:
aaaaaw

รูปภาพ 106 แก้ไข Device

ต่อมา ระดับ Leader



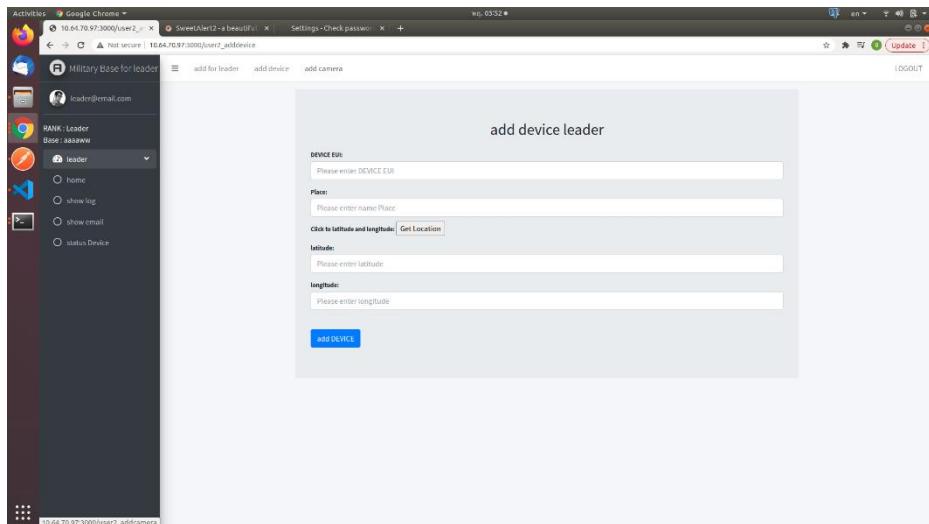
รูปภาพ 107 log in email ระดับ Leader

สามารถ เพิ่ม user ที่ระดับ Leader และ Soldier และสามารถเพิ่ม Device และ CCTV ได้

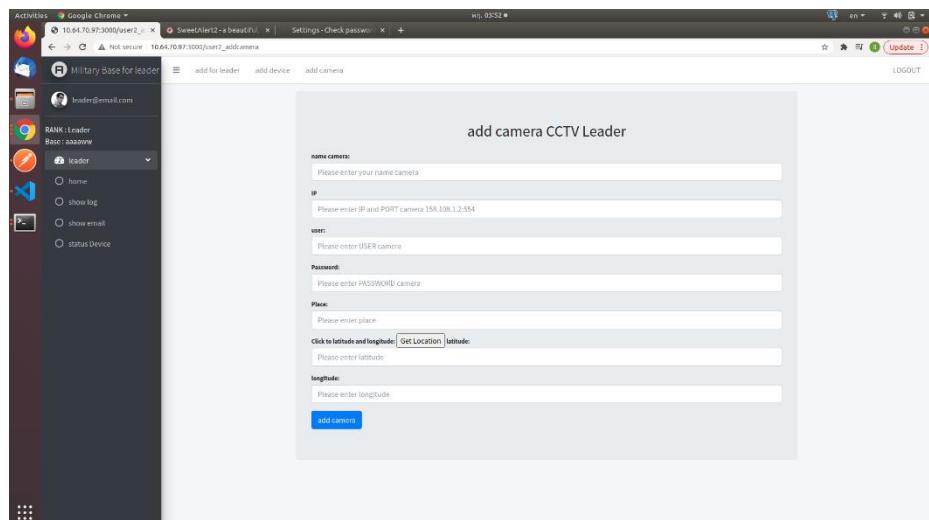
Add user

Name:	<input type="text"/>	<input type="button" value="Choose File"/>	No file chosen
<input type="button" value="Upload Video"/>			
Email:	<input type="text"/>		
Password:	<input type="text"/>		
Firstname:	<input type="text"/>		
Lastname:	<input type="text"/>		
PHONE NUMBER:	<input type="text"/>		
SIM NUMBER:	<input type="text"/>		
rank:	<input checked="" type="radio"/> Soldier <input type="radio"/> Leader <input type="radio"/> Soldier		
<input type="button" value="Sign Up"/>			

รูปภาพ 108 Add user ของ Leader

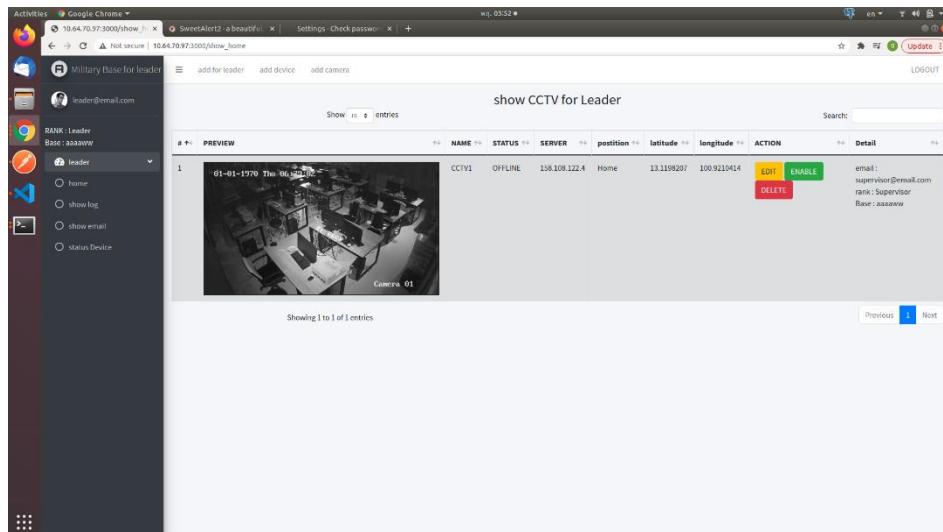


รูปภาพ 109 เพิ่ม Device ของ leader



รูปภาพ 110 เพิ่มกล้องของ Leader

ต่อมา หน้า home ของ Leader จะสามารถ จัดการกับ กล้อง Cctv ได้ Edit , เปิดปิด , Delete



รูปภาพ 111 หน้า Home ของ Leader

แก้ไข กล้อง CCTV

name camera:	CCTV1
IP-PORT:	158.108.122.4
user:	admin
Password:	kusrcl2345
Place:	Home
Click a latitude and longitude:	<input type="button" value="Get Location"/> latitude: 13.1198207 longitude: 100.9210414
<input type="button" value="update camera"/>	

รูปภาพ 112 เพิ่ม กล้อง Cctv ของ Leader

ต่อมาจะเป็นหน้า ดู log ของ Base ตนเอง ดูได้แค่ ระดับ Leader และ Soldier ที่อยู่ใน Base เราเท่านั้น

The screenshot shows a web browser window with a sidebar menu on the left. The sidebar has sections for 'RANK: Leader' and 'Base : aaaaaw'. Under 'leader', there are links for 'home', 'show log', 'show email', and 'status Device'. The main content area is titled 'show log leader'. It contains a table with columns: ID, Email, time, Rank, Base, and log. The table has 7 entries:

ID	Email	time	Rank	Base	log
1	soldier@email.com	Thu May 20 2021 02:42:13	Soldier	aaaaaww	Log in
2	soldier@email.com	Thu May 20 2021 02:42:21	Soldier	aaaaaww	ADD Device
3	soldier@email.com	Thu May 20 2021 02:43:00	Soldier	aaaaaww	Log in
4	leader@email.com	Thu May 20 2021 02:50:15	Leader	aaaaaww	Log in
5	leader@email.com	Thu May 20 2021 03:30:46	Leader	aaaaaww	Log in
6	leader@email.com	Thu May 20 2021 03:33:34	Leader	aaaaaww	Log Out luktaaaa
7	leader@email.com	Thu May 20 2021 03:52:41	Leader	aaaaaww	Log in

Showing 1 to 7 of 7 entries

รูปภาพ 113 หน้า ดู log ของ ระดับ Leader

ต่อมา หน้า Show email ดูได้แค่ ระดับ Leader และ Soldier ที่อยู่ใน Base เราเท่านั้น

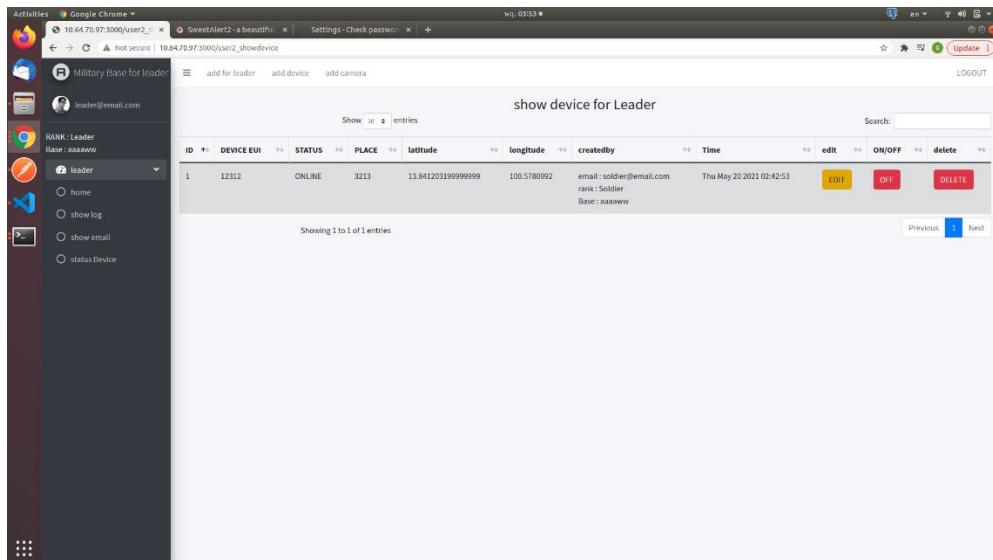
The screenshot shows a web browser window with a sidebar menu on the left. The sidebar has sections for 'RANK: Leader' and 'Base : aaaaaw'. Under 'leader', there are links for 'home', 'show log', 'show email', and 'status Device'. The main content area is titled 'show email for leader'. It contains a table with columns: ID, Email, firstname, lastname, time, rank, Base, and createdby. The table has 2 entries:

ID	Email	firstname	lastname	time	rank	Base	createdby
1	soldier@email.com	soldierXX	nicknameXX	Tue May 18 2021 05:32:41	Soldier	aaaaaww	leader@email.com
2	leader@email.com	leader	leader	Thu May 20 2021 01:40:32	Leader	aaaaaww	admin@email.com

Showing 1 to 2 of 2 entries

รูปภาพ 114 หน้าโชว์ email ของ Leader

ຕໍ່ອມາ ໜ້າ Device Leader ສາມາຮັດ Edit , delete , ເປີດ ປິດ ໄດ້



ຮູບກາພ 115 ໜ້າຈັດກາຮ Device ຂອງ Leader

ແກ້ໄຂ Device

EDIT device 12312

DEVICE EUI:

Place:

click to latitude and longitude:

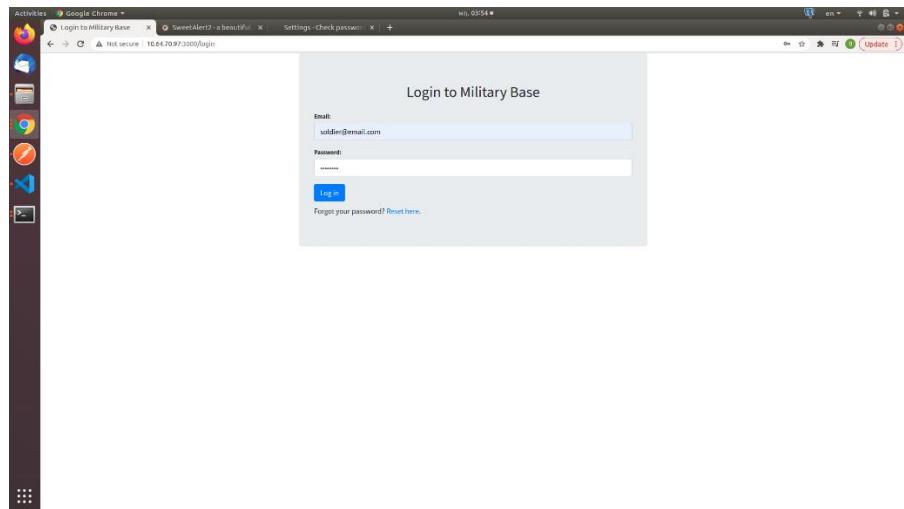
latitude:

longitude:

UPDATE DEVICE

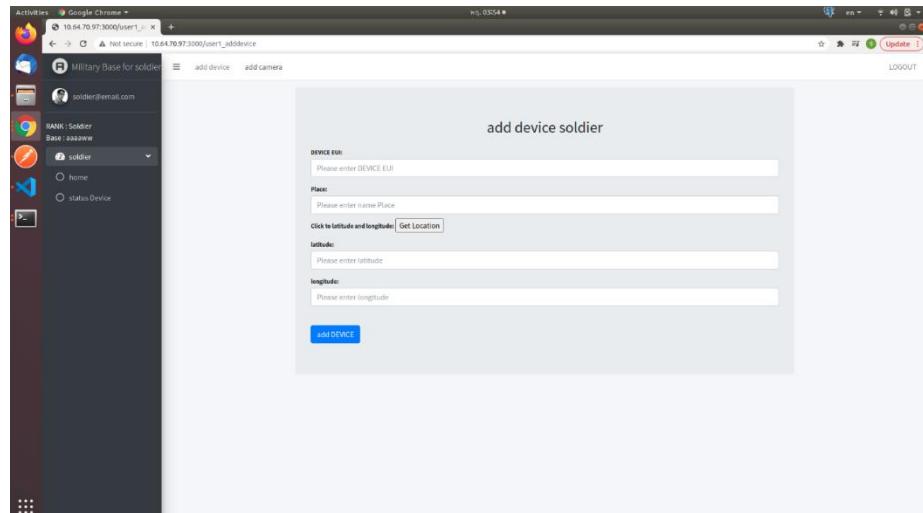
ຮູບກາພ 116 ໜ້າແກ້ໄຂ Device ຂອງ Leader

ต่อมาจะดับ สุดท้าย Soldier

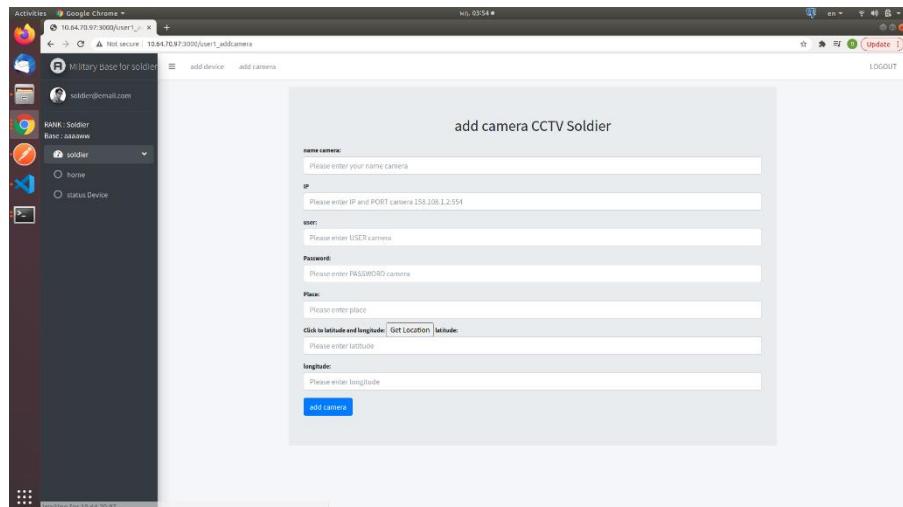


รูปภาพ 117 log in ระดับ Soldier

Soldier นั้น สามารถ เพิ่ม Device และ เพิ่ม กล้อง CCTV ได้

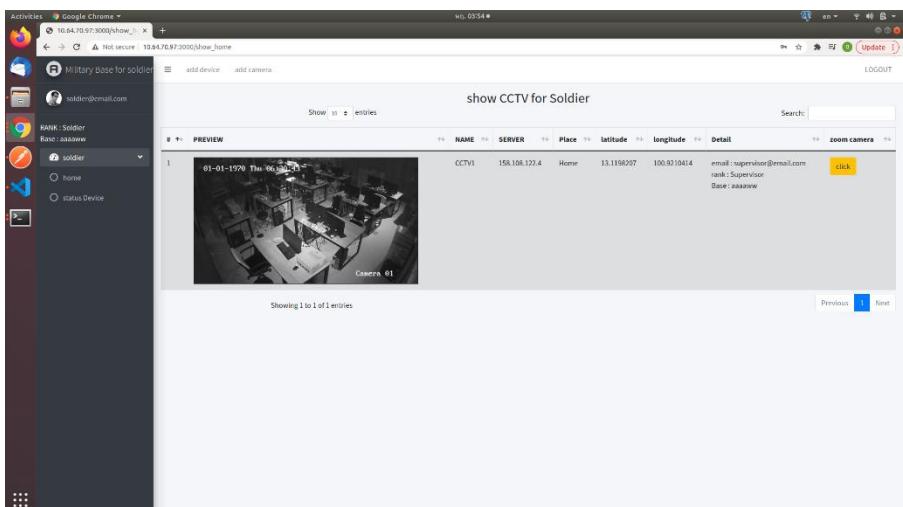


รูปภาพ 118 หน้าเพิ่ม Device ของ Soldier

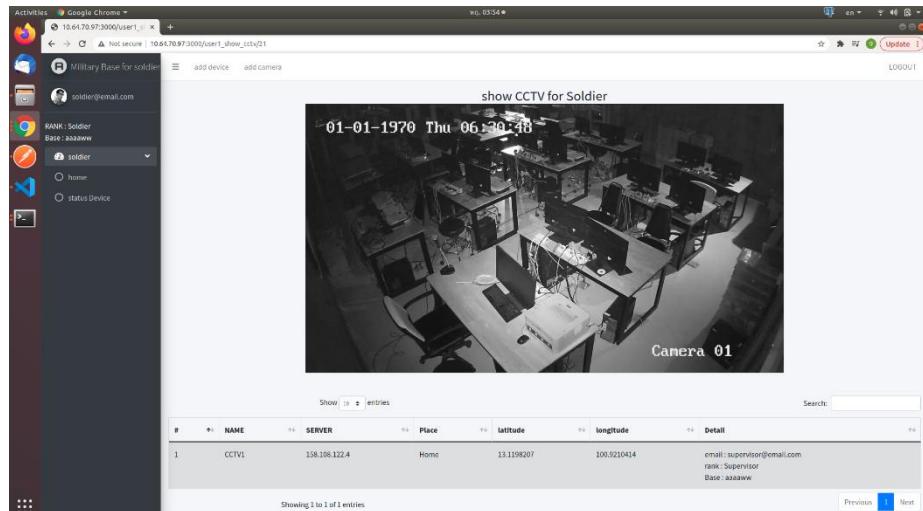


รูปภาพ 119 หน้าเพิ่มกล้อง Cctv ของ Soldier

สามารถดูกล้อง Cctv ได้

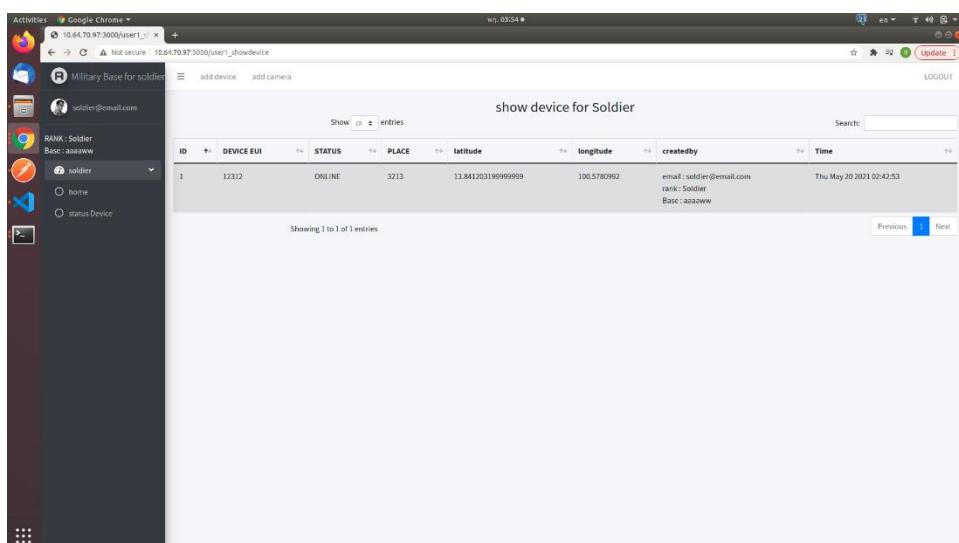


รูปภาพ 120 หน้า home ของ Soldier



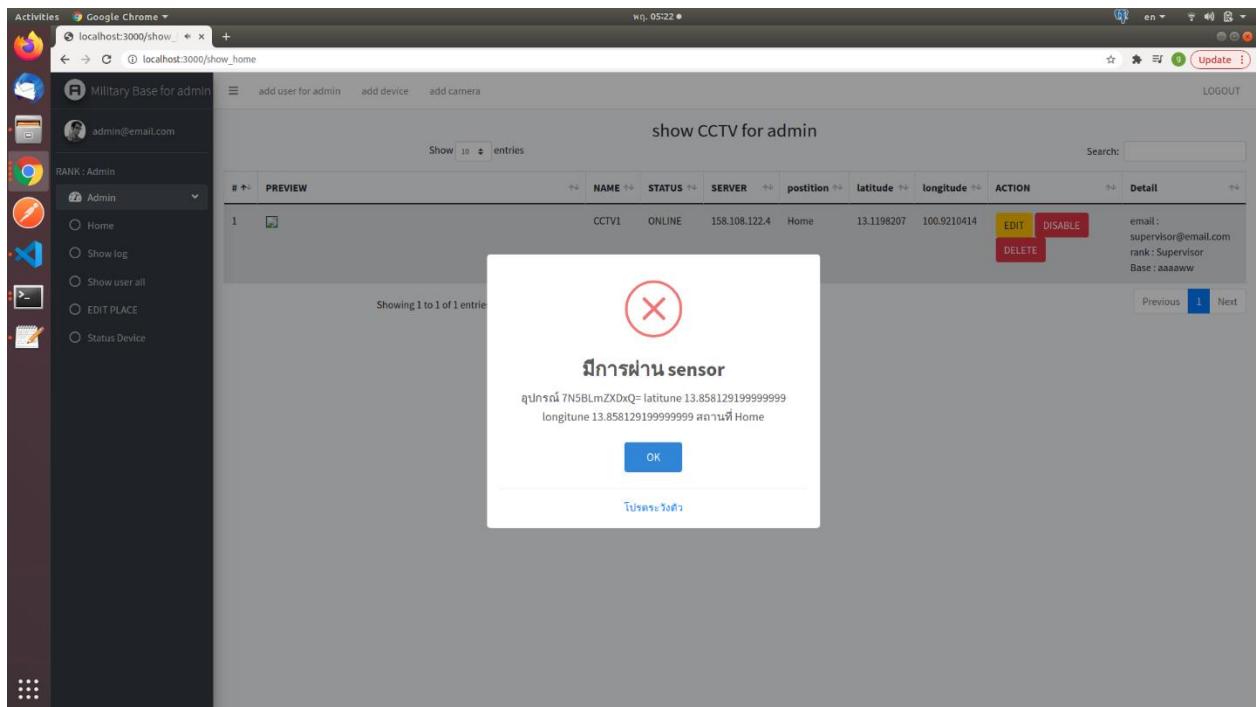
รูปภาพ 121 สามารถดูกล้องแบบขยายให้ใหญ่ได้

และสามารถดู สถานะ Device ได้



รูปภาพ 122 ดูข้อมูลของอุปกรณ์

ทุกรอบดับ ถ้ามีการผ่าน Sensor จะมีการแจ้งเตือน และมีเสียงออกมา และแจ้งว่า ตำแหน่งไหน



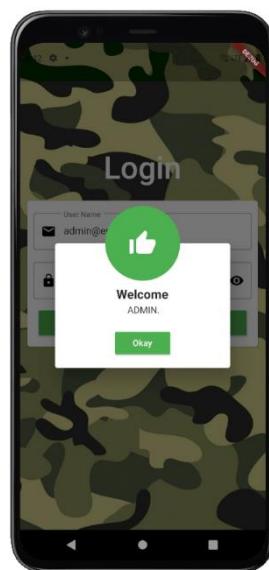
รูปภาพ 123 โชว์การแจ้งเตือน เมื่อมีคนผ่านที่ PIR

สำหรับ mobile application



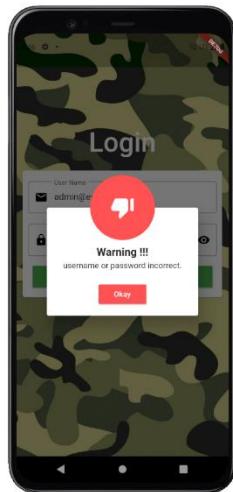
รูปภาพ 124 หน้า Login สำหรับ mobile application

ใส่ Username และ Password ถูกต้อง



รูปภาพ 125 การแจ้งเตือนเมื่อ Login สำเร็จ

ใส่ Username และ Password ผิดพลาด



รูปภาพ 126 การแจ้งเตือนเมื่อ Login ไม่สำเร็จ

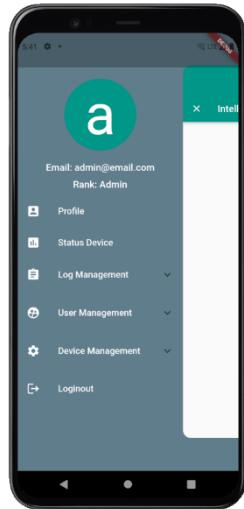
แสดงกล้อง CCTV



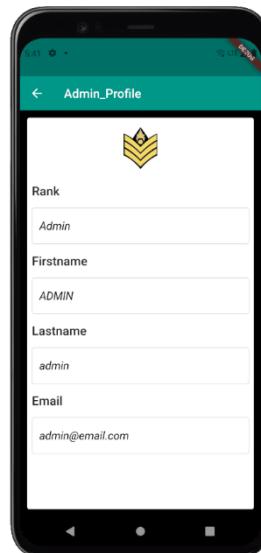
รูปภาพ 127 หน้าแรกสำหรับแสดงกล้อง CCTV

Admin และ Supervisor ระบบจะเหมือนกัน

เราจะเริ่มที่ระบบของ Admin และ Supervisor ก็จะมี เพิ่ม user ,เพิ่ม Device ,เพิ่ม CCTV, ดู Logต่างๆ



รูปภาพ 128 ในส่วนของการ เพิ่ม, ลบ, แสดง log ของ User จะสามารถแสดงเฉพาะตำแหน่งที่เท่ากันหรือต่ำกว่า
แสดงข้อมูลต่างๆของ User เมื่อ Login สำเร็จ



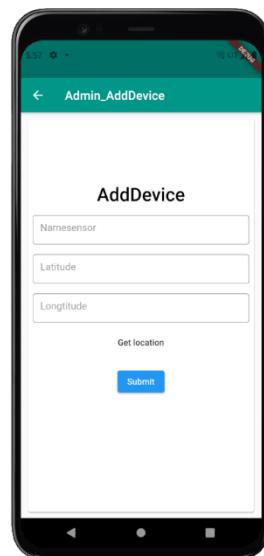
รูปภาพ 129 หน้าแสดงหน้า Profile

เมื่อกดเข้ามาที่เมนู Device Management หลังจากนั้นเลือกเพิ่ม CCTV



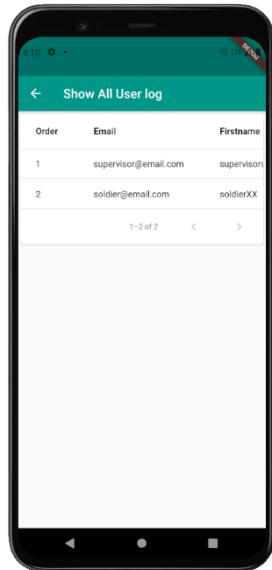
รูปภาพ 130 กรอกรายละเอียดของ CCTV เพื่อเพิ่ม CCTV

เมื่อกดเข้ามาที่เมนู Device Management หลังจากนั้นเลือกเพิ่ม Device



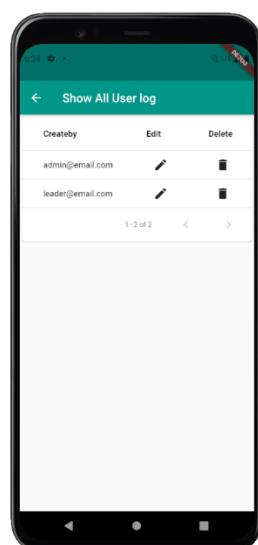
รูปภาพ 131 กรอกรายละเอียดของ Device เพื่อเพิ่ม Device

เมื่อกดเข้ามาที่เมนู User Management หลังจากนั้นเลือกเพิ่ม All User จะแสดง User ทั้งหมด



รูปภาพ 132 แสดง User ทั้งหมดที่อยู่ในระบบ เนื่องไปในการแสดงข้อมูลกับ ตำแหน่ง

สามารถ Delete หรือ Edit



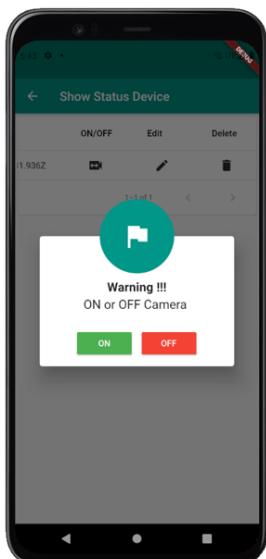
รูปภาพ 133 สำหรับ Delete หรือ Edit User

สำหรับแสดงข้อมูลของ Device



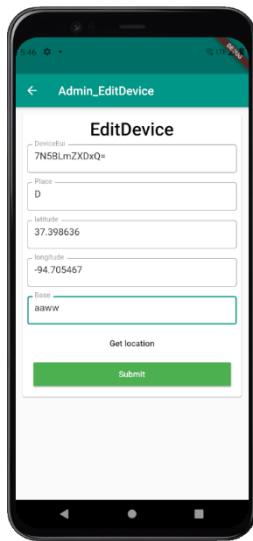
รูปภาพ 134 แสดงรายละเอียดของ Device

สามารถเลือกการเปิดหรือปิดกล้องได้



รูปภาพ 135 เมื่อเลือกแล้วจะแสดงให้เห็น เปิดหรือปิด

ສໍາหารັບການ Edit Device



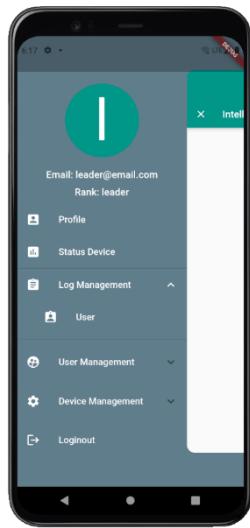
ຮູບກາພ 136 ແລ້ວຂໍ້ຂໍ້ອມຸລ Device

ແສດງຂໍ້ອມຸລຂອງຈານແຕ່ລະຈານສາມາຄັດ Delete ແລະ Edit



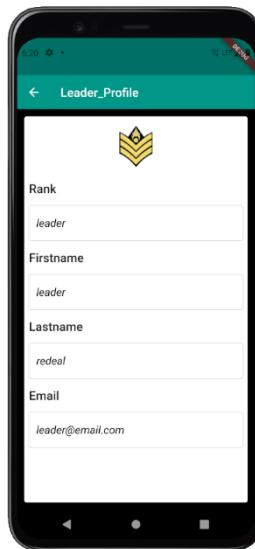
ຮູບກາພ 137 ແສດງຂໍ້ອມຸລຂອງຈານແຕ່ລະຈານ

ระบบของ Leader ก็จะมี เพิ่ม user ,เพิ่ม Device ,เพิ่ม CCTV, ดู Log ต่างๆ



รูปภาพ 138 ในส่วนของการ เพิ่ม, ลบ, แสดง log ของ User จะสามารถแสดงเฉพาะตำแหน่งที่เท่ากันหรือต่ำกว่า

แสดงข้อมูลต่างๆของ User เมื่อ Login สำเร็จ



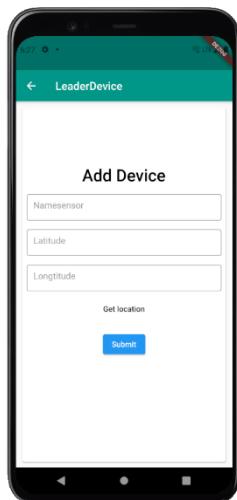
รูปภาพ 139 หน้าแสดงหน้า Profile

เมื่อกดเข้ามาที่เมนู Device Management หลังจากนั้นเลือกเพิ่ม CCTV



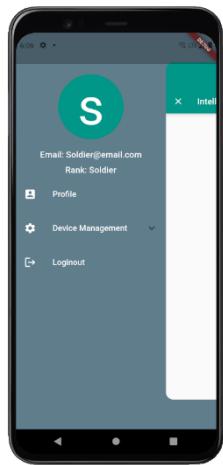
รูปภาพ 140 กรอกรายละเอียดของ CCTV เพื่อเพิ่ม CCTV

เมื่อกดเข้ามาที่เมนู Device Management หลังจากนั้นเลือกเพิ่ม Device



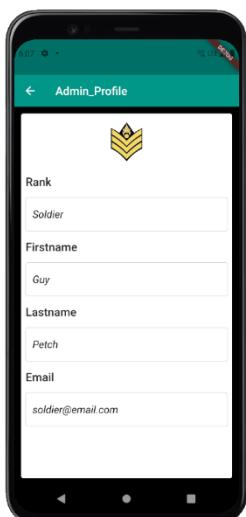
รูปภาพ 141 กรอกรายละเอียดของ Device เพื่อเพิ่ม Device

ระบบของ Leader ก็จะมี ,ดู Profile, เพิ่ม Device ,ดู CCTV,



รูปภาพ 142 แสดง Profile และสามารถ เพิ่มอุปกรณ์ และเพิ่ม CCTV ได้

แสดงข้อมูลต่างๆของ User เมื่อ Login สำเร็จ



รูปภาพ 143 หน้าแสดงหน้า Profile

เอกสารอ้างอิง

- Griffey-Hill , L. (2020, mar 11). <https://pycom.io>. Retrieved may 20, 2021, from
<https://pycom.io/chirpstack-a-lorawan-network-server-to-make-you-sing/>
- SUPPORT THAIEASYELEC. (2018, july 14). [https://blog.thaieeasyelec.com](https://blog.thaieasyelec.com). Retrieved 5 20, 2021, from <https://blog.thaieeasyelec.com/raspberry-pi-programming-with-qt-ch1/>
- aofleejay. (2017, jul 29). <https://medium.com/>. Retrieved 5 30, 2021, from
<https://medium.com/@aofleejay/%E0%B8%AA%E0%B8%A3%E0%B9%89%E0%B8%B2%E0%B8%87-restful-api-%E0%B8%94%E0%B9%89%E0%B8%A7%E0%B8%A2-express-express-101-ee37cc4952b4#:~:text=Express%20%E0%B9%80%E0%B8%9B%E0%B9%87%E0%B8%99%20web%20application%20framework,%E0%B8%AA%>
- Chaivanichanan, T. (2020, january 4). <https://codingthailand.com>. Retrieved may 20, 2021, from
<https://codingthailand.com/blog/%E0%B8%AA%E0%B8%A3%E0%B8%B8%E0%B8%9B%E0%B8%9E%E0%B8%B7%E0%B9%89%E0%B8%99%E0%B8%90%E0%B8%B2%E0%B8%99%E0%B8%A0%E0%B8%B2%E0%B8%A9%E0%B8%B2-dart-%E0%B8%81%E0%B9%88%E0%B8%AD%E0%B8%99%E0%B9%80%E0%B8%82/>
- Hizoka. (2018, Jun 22). <https://medium.com>. Retrieved may 20, 2021, from
<https://medium.com/@hizokaz/%E0%B8%A1%E0%B8%B2%E0%B8%97%E0%B8%B3%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1%E0%B8%A3%E0%B8%B9%E0%B9%89%E0%B8%88%E0%B8%B1%E0%B8%81%E0%B8%81%E0%B8%B1%E0%B8%9A-flutter-%E0%B8%81%E0%B8%B1%E0%B8%99%E0%B9%80%E0%B8%96%E0%B8%AD%E0%B8%80-4>
- <http://csdatasystem.blogspot.com>. (n.d.). Retrieved May 5, 2021, from
<http://csdatasystem.blogspot.com/2016/09/wireless-usb-adapter.html>

<http://www.elfhs.ssru.ac.th>. (n.d.). Retrieved 5 20, 2021, from

http://www.elfhs.ssru.ac.th/wipada_ch/pluginfile.php/868/course/summary/Chapter7-JavaScript.pdf

<https://guru.sanook.com>. (2556, ณ 26). Retrieved 5 20, 2021, from

<https://guru.sanook.com/6394/>

<https://th.mouser.com>. (n.d.). Retrieved may 20, 2021, from

<https://th.mouser.com/new/stmicroelectronics/stmicro-stm32-lorawan-discovery-board/>

<https://www.borntodev.com>. (n.d.). Retrieved 5 20, 2021, from

<https://www.borntodev.com/2020/04/11/%E0%B9%80%E0%B8%A3%E0%B8%B5%E0%B8%A2%E0%B8%99-dart-%E0%B9%81%E0%B8%9A%E0%B8%9A%E0%B8%81%E0%B9%89%E0%B8%B2%E0%B8%A7%E0%B8%81%E0%B8%A3%E0%B8%B0%E0%B9%82%E0%B8%94%E0%B8%94/>

<https://www.borntodev.com>. (n.d.). Retrieved may 20, 2021, from

<https://www.borntodev.com/2020/04/11/%E0%B9%80%E0%B8%A3%E0%B8%B5%E0%B8%A2%E0%B8%99-dart-%E0%B9%81%E0%B8%9A%E0%B8%9A%E0%B8%81%E0%B9%89%E0%B8%B2%E0%B8%A7%E0%B8%81%E0%B8%A3%E0%B8%B0%E0%B9%82%E0%B8%94%E0%B8%94/>

<https://www.mindphp.com>. (2560, october 27). Retrieved 5 20, 2021, from

<https://www.mindphp.com/%E0%B8%9A%E0%B8%97%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1/microsoft/4829-visual-studio-code.html>

<https://www.mindphp.com/>. (2021, april 25). <https://www.mindphp.com>. Retrieved from

<https://www.mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A1%E0%B8%B7%E0%B8%AD/73-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/3872-what-is-postgresql.html>

- https://www.myarduino.net.* (2020, april 14). Retrieved may 20, 2021, from
<https://www.myarduino.net/product/3732/raspberry-pi-camera-v2-8mp#:~:text=Raspberry%20Pi%20Camera%20V2%20%E0%B9%82%E0%B8%A1%E0%B8%94%E0%B8%B9%E0%B8%A5,%E0%B9%84%E0%B8%94%E0%B9%89%E0%B8%97%E0%B8%81%E0%B9%82%E0%B8%A1%E0%B9%80%E0%B8%94%E0%B8%A5%E0%B8%A3>
- Nonsiri, S. (n.d.). *https://www.9experttraining.com.* Retrieved 5 20, 2021, from
<https://www.9experttraining.com/articles/python-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3>
- Oyebola, B. O. (2015, June). *https://www.researchgate.net.* Retrieved 5 20, 2021, from
https://www.researchgate.net/publication/329772538_Home_Security_System_with_Text_Alert_and_Motion_Detective
- Phonbopit, C. (2015, July 31). *https://devahoy.com.* Retrieved 5 20, 2021, from
<https://devahoy.com/blog/2015/07/getting-started-with-nodejs/>
- Ramakrishna, Y. (2020, November). *https://www.researchgate.net.* Retrieved 5 20, 2021, from
https://www.researchgate.net/publication/345709579_INTRUDER_ALERT_AND_SECURITY_SYSTEM
- SUPPORT THAIEASYELEC. (2017, july 14). *https://blog.thaieasyelec.com.* Retrieved may 20, 2021, from [https://blog.thaieeasyelec.com/getting-started-pir-motion-sensor/](https://blog.thaieasyelec.com/getting-started-pir-motion-sensor/)
- SUPPORT THAIEASYELEC. (2019, august 19). *https://blog.thaieeasyelec.com.* Retrieved may 20, 2021, from <https://blog.thaieeasyelec.com/introduction-to-mqtt/>
- พิชิต สีทธิกัน. (2018, Mar 21). *https://artsandculture.mju.ac.th.* Retrieved may 20, 2021, from
<https://artsandculture.mju.ac.th/acticleDetail.aspx?qid=786>