รายงานการวิจัย

เรื่อง การพัฒนาระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็ค

A Development of Rack Server Security Surveillance System.

โดย

นายจันตพงษ์ บุตรลักษณ์

แหล่งทุน

สำนักคอมพิวเตอร์และเทคโนโลยีสารสนเทศ

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

พ.ศ. 2560 (ปีงบประมาณที่ได้รับทุน)

หัวข้องานวิจัย : การพัฒนาระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็ค
โดย : นายจันตพงษ์ บุตรลักษณ์
ชื่อทุน : ทุนสนับสนุนการทำวิจัยและเสนอผลงานวิจัย
ปีงบประมาณ : 2560

บทคัดย่อ

การให้บริการระบบสารสนเทศที่มีคุณภาพการให้บริการ (SLA) ที่ดีนั้น ผู้รับบริการจะต้องได้รับการบริการอย่างต่อเนื่องและปลอดภัย โดยตัวแปรที่ส่งผลต่อสภาพแวดล้อมของศูนย์ข้อมูล เช่น อุณหภูมิ ความชื้น ระบบทำความเย็น ระบบตรวจจับควันและระบบไฟฟ้าเป็นต้น เพราะสิ่งเหล่านี้สามารถก่อให้เกิดความเสียหายกับข้อมูลและอุปกรณ์ต่างๆที่อยู่ภายในศูนย์ข้อมูลได้

ดังนั้นงานวิจัยนี้ต้องการเสนอการพัฒนาระบบเฝ้าระวังความปลอดภัยของเซิร์ฟเวอร์แร็ค โดยพัฒนาระบบตรวจสอบสภาพแวดล้อมต่าง ๆ ของศูนย์ข้อมูลและมีการแจ้งเตือนเมื่อเกิดเหตุการณ์ผิดปกติ ซึ่งจะช่วยป้องกันอุปกรณ์ต่าง ๆ ในศูนย์ข้อมูลไม่ให้เกิดความเสียหายร่วมถึงการบูรณาการกับอุปกรณ์บางส่วนที่อยู่ในศูนย์ข้อมูลเพื่อคุณภาพการให้บริการ (Service Level Agreement) รวมถึงลดระยะเวลาในการกู้คืนระบบของศูนย์ข้อมูล

ผลการดำเนินงานวิจัย ระบบสามารถแสดงข้อมูลสถานะอุณหภูมิ ความชื้น ตรวจจับควันไฟและการใช้งานกระแสไฟฟ้าแบบเรียลไทม์ รวมถึงมีการแจ้งเตือนผ่านโปรแกรม Line และอีเมล์หากตรวจพบความผิดปกติ ผลการประเมินระบบจากผู้เชี่ยวชาญ ด้านประสิทธิภาพของระบบ, ด้านความสะดวก สวยงาม, ด้านคุณภาพของระบบ ค่าเฉลี่ยเท่ากับ 4.2 อยู่ในระดับมาก

Research Title : A Development of Rack Server Security Surveillance System.
Researcher : Mr. Jantapong Boodluck
Funding Sources : Institute of Computer and Information Technology
Funding Category : Grants for research projects and presentation
Year : 2017

Abstract

In the information technology services, the service level agreement (SLA) plays an important role to ensure the quality of service that the provider can provide to the users. The provided services must be continuous and secure. However, some factors affect the data center environment and must be controlled and monitored such as temperature, humidity, cooling system, smoke detection, and electrical systems. Because these factors may cause the damage to the data and equipment in the data center.

Therefore, this research proposes the development of rack server security surveillance system. This system monitors the various data center environments and alerts the administrators when the abnormal event happened. When the administrators get the notification from the system, they can react fast to the data center environment problems. Thus, it will help protect data center equipment from damage and reduce time to recover the data center system.

From the testing, our system can display the temperature, humidity, smoke detector status information, and electrical usage in the data center in real-time. Moreover, the system can notify the administrators via LINE application and email when the system detects the abnormal events in the data center. The system evaluation from the experts in the system performance, convenience, and quality, we found that the mean score is 4.2 which is at a high level.

กิตติกรรมประกาศ

ขอขอบพระคุณ คณะเจ้าหน้าที่และสำนักคอมพิวเตอร์และเทคโนโลยีสารสนเทศที่ได้เอื้อเฟื้ออุปกรณ์ เครื่องมือ สถานที่และให้ความช่วยเหลือในด้านต่าง ๆ เป็นอย่างดี ขอบคุณ เพื่อน ๆ พี่ ๆ น้อง ๆ ทุกคนที่ช่วยทดสอบระบบ ให้คำปรึกษา ข้อเสนอแนะและช่วยเป็นกำลังใจมาโดยตลอด ขอขอบพระคุณ ทุกท่านและผู้ที่มีส่วนร่วมเกี่ยวข้องกับความสำเร็จ แต่มิได้เอ่ยนามทุกท่านมา ณ ที่นี้ด้วย

ท้ายนี้ผู้จัดทำใคร่ขอกราบขอบพระคุณบิดา-มารดา คณะผู้บริหารที่คอยเป็นแรงผลักดันเป็นกำลังใจ อีกทั้งให้การสนับสนุนในทุก ๆ ด้าน ตลอดจนคณะครู-อาจารย์ทุกท่านที่ได้เคยประสิทธิ์ประสาทวิชาความรู้ รวมทั้งความหวังดีอื่น ๆ กระทั่งผู้จัดทำประสบความสำเร็จในการทำงานวิจัย

ผู้จัดทำ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญตาราง	ง
สารบัญภาพ	ช
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการวิจัย	2
1.3 สมมติฐานของการวิจัย	2
1.4 ขอบเขตของการวิจัย	2
1.5 นิยามศัพท์เฉพาะ	2
1.6 ประโยชน์ที่คาดว่าจะได้รับ	3
บทที่ 2 ผลงานวิจัยและทฤษฎีที่เกี่ยวข้อง	4
2.1 ทฤษฎีที่เกี่ยวข้อง	4
2.2 ผลงานวิจัยที่เกี่ยวข้อง	9
บทที่ 3 วิธีดำเนินการวิจัย	11
3.1 หลักการทำงานของโปรแกรม	11
3.2 การออกแบบอุปกรณ์ฮาร์ดแวร์และการเขียนโปรแกรม	13
3.3 การเขียนโปรแกรม PHP ติดต่อกับ PDU (Power Distribution Unit)	17
3.4 การออกแบบส่วนติดต่อกับผู้ใช้งานบนเว็บด้วย Grafana	21
บทที่ 4 ผลการวิจัย	26
4.1 ผลการวัดประสิทธิภาพของอุปกรณ์	26
4.2 หน้าเว็บไซต์ระบบเฝ้าระวังความปลอดภัยของเซิร์ฟเวอร์แร็ค	28
4.3 การแจ้งเตือนความผิดปกติผ่านโปรแกรม Line และอีเมล์	31
4.4 ผลการประเมินประสิทธิภาพและความพึงพอใจจากผู้เชี่ยวชาญ	32
บทที่ 5 สรุป อภิปรายผล และข้อเสนอแนะ	33
5.1 สรุป	33
5.2 ปัญหาและแนวทางแก้ไขปัญหา	33
5.3 ข้อเสนอแนะและแนวทางพัฒนา	34
บรรณานุกรม	35
ภาคผนวก ก	36
รายนามผู้เชี่ยวชาญประเมินระบบ	37
ภาคผนวก ข	38
แบบสอบถาม	39

ภาคผนวก ค 40
คู่มือการใช้งาน 41
ประวัติผู้วิจัย 46

สารบัญตาราง

ตารางที่		หน้า
4-1	ประสิทธิภาพของอุปกรณ์วัดอุณหภูมิและความชื้น	27
4-2	ประสิทธิภาพของอุปกรณ์ตรวจจับควันไฟ	27
4-3	ผลการประเมินประสิทธิภาพและความพึงพอใจ	31

สารบัญภาพ

ภาพที่		หน้า
2-1	โหนดเอ็มซียู (Node MCU ESP8266)	5
2-2	ตำแหน่งขาของโหนดเอ็มซียู (Node MCU ESP8266)	5
2-3	DHT21 (AM2301)	6
2-4	โมดูลตรวจวัดแก๊สไวไฟ (MQ-2)	6
3-1	บล๊อกไดอะแกรมของระบบ	11
3-2	การเชื่อมต่อ ESP8266 กับเซ็นเซอร์วัดอุณหภูมิและความชื้น	14
3-3	การเชื่อมต่อ ESP8266 กับเซ็นเซอร์ตรวจจับควัน (MQ2)	15
3-4	การเชื่อมต่อ ESP8266 กับจอ OLED	17
3-5	รูปแบบโครงสร้างการเชื่อมต่อของ PDU	18
3-6	OID โหลดกำลังไฟฟ้า	19
3-7	OID การใช้งานกระแสไฟฟ้า	19
3-8	OID ความถี่	19
3-9	โครงสร้างหน้าแสดงผล	21
3-10	โครงสร้างหน้าแสดงผลแบบ Real Time	22
3-11	การสร้างกราฟ	22
3-12	การกำหนดค่าแจ้งเตือน	23
3-13	โครงสร้างหน้าแสดงกำลังไฟฟ้า PDU	24
3-14	โครงสร้างหน้าแสดงอุณหภูมิตู้แร็ค	24
3-15	โครงสร้างหน้าแสดงความชื้นตู้แร็ค	25
3-16	โครงสร้างหน้าแสดงกำลังไฟฟ้า Input	25
3-17	โครงสร้างหน้าแสดงความถี่ Input	25
3-18	โครงสร้างหน้าแสดงการใช้กำลังไฟฟ้า	25
3-19	โครงสร้างหน้าแสดงสถานะควันไฟ	26
4-1	อุปกรณ์ที่ติดตั้งสำหรับวัดค่าอุณหภูมิ ความชื้น และควันไฟ	27
4-2	หน้า Login	29
4-3	หน้า Home	29
4-4	หน้าจอเว็บไซต์บนอุปกรณ์พกพาในแนวตั้ง (iPhone X)	30
4-5	หน้าจอเว็บไซต์บนอุปกรณ์พกพาในแนวนอน (iPhone X)	31
4-6	การแจ้งเตือนผ่านโปรแกรม Line และอีเมล์	31
4-7	กำหนดค่าการแจ้งเตือน	32

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การให้บริการระบบสารสนเทศที่มีคุณภาพการให้บริการ (SLA) ที่ดีนั้น ผู้รับบริการจะต้องได้รับการบริการอย่างต่อเนื่องและปลอดภัย ปัจจัยที่มีความสำคัญต่อคุณภาพในการให้บริการคือ การป้องกันไวรัส สปายแวร์ ภัยคุกคามเครือข่ายต่าง ๆ และการรักษาสภาพแวดล้อมของศูนย์ข้อมูลให้มีความเหมาะสมอยู่ตลอดเวลา โดยตัวแปรที่ส่งผลต่อสภาพแวดล้อมของศูนย์ข้อมูล ได้แก่ ความร้อน ความชื้น ระบบทำความเย็น ระบบตรวจจับควันและระบบไฟฟ้า เพราะสิ่งเหล่านี้สามารถก่อให้เกิดความเสียหายกับข้อมูลและอุปกรณ์ต่าง ๆ ที่อยู่ภายในศูนย์ข้อมูลได้

ผลกระทบที่เห็นอย่างชัดเจน เช่น กรณีของอุณหภูมิความร้อน โดยปกติคอมพิวเตอร์เซิร์ฟเวอร์จะปล่อยอุณหภูมิความร้อนสูงออกมา ซึ่งความร้อนส่งผลโดยตรงต่อประสิทธิภาพการทำงานของคอมพิวเตอร์เซิร์ฟเวอร์ อุณหภูมิที่เพิ่มขึ้นแต่ละองศามีผลต่อการทำงานของคอมพิวเตอร์เซิร์ฟเวอร์ ซึ่งอาจจะก่อให้เกิดการทำงานที่ผิดพลาดได้ โดยการเพิ่มอุณหภูมิจาก 20 องศาเซลเซียส ไปเป็น 30 องศาเซลเซียส ในระยะยาวส่งผลให้คอมพิวเตอร์เซิร์ฟเวอร์หรืออุปกรณ์อิเล็กทรอนิกส์ต่าง ๆ มีอายุการใช้งานลดลงถึง 50 เปอร์เซ็นต์ (Miller, 2009) และหากระบบทำความเย็นขัดข้องไม่สามารถทำงานได้อุณหภูมิอาจจะสามารถเพิ่มขึ้นมากกว่า 30 องศาเซลเซียสได้ภายในเวลาไม่ถึง 1 นาที ผลกระทบของความร้อนที่เกิดขึ้นในชิ้นส่วนอิเล็กทรอนิกส์ เมนบอร์ด และซิปหน่วยความจำ จะไม่เสียในทันที แต่อายุการใช้งานจะน้อยลงและเสียหายในระยะเวลาที่สั้นกว่าที่ควรจะเป็น (Hughes, 2009) นอกจากนี้กรณีของความชื้นอาจทำให้วงจรอิเล็กทรอนิกส์ลัดวงจร หรือการทำงานของระบบทำความเย็นที่อาจจะก่อให้เกิดปัญหาน้ำรั่วซึมเป็นผลกระทบให้เกิดอุณหภูมิและความชื้นผิดปกติหรืออาจทำให้ระบบไฟฟ้าลัดวงจร กระแสไฟฟ้าในศูนย์ข้อมูลต้องมีความเสถียรและสามารถจ่ายกระแสไฟฟ้าให้อุปกรณ์ต่าง ๆ ในศูนย์ข้อมูลได้อย่างต่อเนื่อง บางครั้งอาจมีการนำคอมพิวเตอร์เซิร์ฟเวอร์มาติดตั้งโดยที่ไม่ได้ผ่านการตรวจสอบหรือในกรณีคอมพิวเตอร์เซิร์ฟเวอร์บางเครื่องทำงานผิดปกติอาจทำให้ใช้งานกระแสไฟฟ้าเพิ่มมากขึ้น ส่งผลกระบทต่อเสถียรภาพระบบจ่ายกระแสฟ้าได้ ระบบตรวจสอบควันไฟที่ควรต้องมีเฉพาะจุด เช่น เซิร์ฟเวอร์แร็ค เนื่องจากคอมพิวเตอร์เซิร์ฟเวอร์มีการทำงานตลอดเวลา ซึ่งอาจมีบางกรณีที่คอมพิวเตอร์เซิร์ฟเวอร์อุปกรณ์ภายในเสียหายเกิดควันไฟขึ้นหากปล่อยไว้นานอาจเกิดไฟไหม้ได้ ปัจจุบันคอมพิวเตอร์เซิร์ฟเวอร์และอุปกรณ์บางส่วนในศูนย์ข้อมูลสามารถรายงานสถานะอุณหภูมิและการใช้งานกระแสไฟฟ้าได้ ทั้งนี้ไม่รวมถึงคอมพิวเตอร์เซิร์ฟเวอร์ของหน่วยงานต่าง ๆ ที่นำมาติดตั้งที่ศูนย์ข้อมูล ซึ่งไม่มีสิทธิ์เข้าไปตรวจสอบสถานะได้ ดังนั้นข้อมูลที่ได้จึงไม่เพียงพอสำหรับการเฝ้าระวังความปลอดภัยของศูนย์ข้อมูล

จากปัญหาดังกล่าวข้างต้น งานวิจัยนี้ต้องการเสนอการพัฒนาระบบเฝ้าระวังความปลอดภัยของเซิร์ฟเวอร์แร็ค โดยพัฒนาระบบตรวจสอบสภาพแวดล้อมต่างๆ ของศูนย์ข้อมูลและมีการแจ้งเตือนเมื่อเกิดเหตุการณ์ผิดปกติ เพื่อป้องกันอุปกรณ์ต่างๆ ในศูนย์ข้อมูลไม่ให้เกิดความเสียหาย รวมถึงการบูรณาการกับระบบรายงานสถานะของคอมพิวเตอร์เซิร์ฟเวอร์และอุปกรณ์บางส่วนที่อยู่ใน

ศูนย์ข้อมูล เพื่อคุณภาพการให้บริการ (Service Level Agreement) และช่วยลดระยะเวลาในการกู้คืนระบบของศูนย์ข้อมูลให้ได้เร็วที่สุด รวมถึงบันทึกข้อมูลการใช้งานทรัพยากรและสภาพแวดล้อมต่างๆ เพื่อเป็นข้อมูลนำไปสู่การวิเคราะห์ด้านความคุ้มค่าของการลงทุนและการบริหารจัดการระบบศูนย์ข้อมูลอย่างคุ้มค่าต่อไปในอนาคต

1.2 วัตถุประสงค์ของการวิจัย

1.2.1 เพื่อพัฒนาระบบเฝ้าระวังความปลอดภัยของเซิร์ฟเวอร์แร็ค

1.2.2 เพื่อประเมินคุณภาพของระบบที่พัฒนาขึ้น

1.2.3 เพื่อหาประสิทธิภาพของอุปกรณ์ตรวจจับสภาพแวดล้อมในระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็ค

1.3 สมมติฐานของการวิจัย

การรวบรวมข้อมูลสภาพแวดล้อมต่าง ๆ ภายในเซิร์ฟเวอร์แร็ค แล้วนำมาวิเคราะห์พร้อมแจ้งเตือนจะช่วยลดระยะเวลาในการกู้คืนระบบได้

1.4 ขอบเขตของการวิจัย

1.4.1 ประชากรและกลุ่มตัวอย่าง

1.4.1.1 ประชากร คือ ผู้เชี่ยวชาญที่มีประสบการณ์ในการทำงานด้านเครือข่ายคอมพิวเตอร์ การให้บริการศูนย์ข้อมูลคอมพิวเตอร์ และวิศวกรด้านคอมพิวเตอร์แม่ข่ายอย่างน้อย 3 ปี

1.4.1.2 กลุ่มตัวอย่าง คือ ผู้เชี่ยวชาญจำนวน 5 คน ที่ได้มาจากการคัดเลือกแบบเจาะจงจากประชากรทั้งหมด

1.4.2 ความสามารถของระบบ

1.4.2.1 ระบบสามารถตรวจสอบอุณหภูมิ ความชื้น ปริมาณการใช้งานกระแสไฟฟ้า และควันไฟภายในตู้แร็คเซิร์ฟเวอร์จำนวน 2 ตู้

1.4.2.2 ระบบสามารถแจ้งเตือนความผิดปกติของอุณหภูมิ ความชื้น ปริมาณการใช้งานกระแสไฟฟ้า และควันไฟผ่านอีเมล์ได้

1.4.2.3 ระบบสามารถแสดงผลค่าที่วัดได้ และการแจ้งเตือนผ่านหน้าเว็บไซต์รองรับการใช้งานผ่าน Smart Phone แบบ Real Time ได้

1.4.2.4 ระบบสามารถบันทึกค่าอุณหภูมิ ความชื้น ปริมาณการใช้งานกระแสไฟฟ้า การแจ้งเตือนควันไฟสำหรับดูย้อนหลังได้ไม่น้อยกว่า 90 วัน

1.4.2.5 ระบบสามารถบูรณาการร่วมกับระบบการรายงานสถานะของคอมพิวเตอร์เซิร์ฟเวอร์และอุปกรณ์บางส่วนที่อยู่ในศูนย์ข้อมูลได้

1.5 นิยามศัพท์เฉพาะ

เฝ้าระวังความปลอดภัยของเซิร์ฟเวอร์แร็ค หมายถึง เฝ้าระวังระดับอุณหภูมิ ความชื้น ควันไฟ และปริมาณการใช้งานกระแสไฟฟ้า

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1.6.1 ได้ระบบตรวจสอบอุณหภูมิ ความชื้น การใช้งานกระแสไฟฟ้า ควันไฟ เพื่อการบริหารจัดการที่สะดวกยิ่งขึ้น

1.6.2 ได้ระบบแจ้งเตือนปัญหาความผิดปกติของอุณหภูมิ ความชื้น การใช้งานกระแสไฟฟ้า ควันไฟ เพื่อการสนับสนุนการกู้คืนระบบที่รวดเร็วและสนับสนุนคุณภาพการให้บริการ (Service Level Agreement)

1.6.3 ได้ระบบจัดเก็บข้อมูลอุณหภูมิ ความชื้น การใช้งานกระแสไฟฟ้า ควันไฟ เพื่อประโยชน์ในการการนำข้อมูลไปวิเคราะห์ต่างๆ เช่น การวิเคราะห์ความคุ้มค่าของการใช้งานคอมพิวเตอร์เซิร์ฟเวอร์ เป็นต้น

1.6.4 ได้ระบบตรวจสอบอุณหภูมิ ความชื้น การใช้งานกระแสไฟฟ้า ควันไฟ ที่เป็นศูนย์กลาง

1.6.5 เพิ่มคุณภาพการให้บริการและลดระยะเวลาในการกู้คืนระบบ

1.6.6 ได้แนวทางในการคัดเลือกอุปกรณ์ตรวจจับสภาพแวดล้อมที่เหมาะสมสำหรับเซิร์ฟเวอร์แร็ค

บทที่ 2
ผลงานวิจัยและทฤษฎีที่เกี่ยวข้อง

การทำวิจัยในครั้งนี้เพื่อการพัฒนาระบบเฝ้าระวังความปลอดภัยของเซิร์ฟเวอร์แร็ค ซึ่งแบ่งออกเป็น 2 หัวข้อใหญ่ ดังนี้

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 โหนดเอ็มซียู (NodeMCU)

โหนดเอ็มซียู (NodeMCU) คือแพลตฟอร์มหนึ่งที่ช่วยในการพัฒนาอินเทอร์เน็ตออฟธิง (Internet of Things หรือ IoT) ที่ประกอบไปด้วยตัวบอร์ด (Development Kit) และเฟิร์มแวร์ (Firmware Software) ที่เป็นโอเพนซอร์ซ ทำให้ใช้งานได้ง่ายขึ้นพร้อมกับโมดูลไวไฟ (ESP8266) ซึ่งเป็นหัวใจสำคัญในการใช้เชื่อมต่อกับอินเทอร์เน็ต ตัวโมดูลอีเอสพี 8266 นั้นมีหลายรุ่น ตั้งแต่เวอร์ชันแรกที่เป็น อีเอสพี-01 จนปัจจุบันมี อีเอสพี-12 และโหนดเอ็มซียู (NodeMCU) เวอร์ชั่นแรกนั้นเป็น อีเอสพี-12 แต่ในเวอร์ชั่น 2 นั้นจะใช้เป็น อีเอสพี-12อี แทน ซึ่งการใช้งานโดยรวมก็ไม่แตกต่างกันมากนัก โหนดเอ็มซียู (NodeMCU) นั้นมีลักษณะคล้ายกับอาร์ดูโน่ คือมีพอร์ตอินพุตเอาต์พุตในตัว สามารถเขียนโปรแกรมควบคุมอุปกรณ์ อินพุต/เอาต์พุต ได้โดยไม่ต้องผ่านอุปกรณ์อื่น ๆ และได้มีนักพัฒนาที่สามารถทำให้อาร์ดูโน่ไอดีอีใช้งานร่วมกับโหนดเอ็มซียู (NodeMCU) ได้ทำให้ใช้ภาษา C/C++ ในการเขียนโปรแกรมได้สามารถใช้งานได้หลากหลายมากยิ่งขึ้น เช่นทำเว็บเซิร์ฟเวอร์ขนาดเล็ก, การควบคุมการเปิด-ปิดไฟผ่านไวไฟและอื่น ๆ อีกมากมาย

2.1.1.1 คุณสมบัติของโหนดเอ็มซียู (NodeMCU(ESP8266) Specification)

2.1.1.1.1 ชุดพัฒนานี้เป็นไมโครคอนโทรเลอร์ที่มีโมดูลไวไฟในตัว ทำให้มีความสะดวกในการเชื่อมต่อเครือข่ายไร้สาย แสดงดังภาพที่ 2-1 และตำแหน่งขา แสดงดังภาพที่ 2-2

2.1.1.1.2 มีจีพีไอโอ, พีดับบลิวเอ็ม, ไอสแควร์ซี, วันไวร์ และเอดีซี (GPIO PWM, I2C, 1-wire และ ADC) รวมมาอยู่บนบอร์ดเดียว

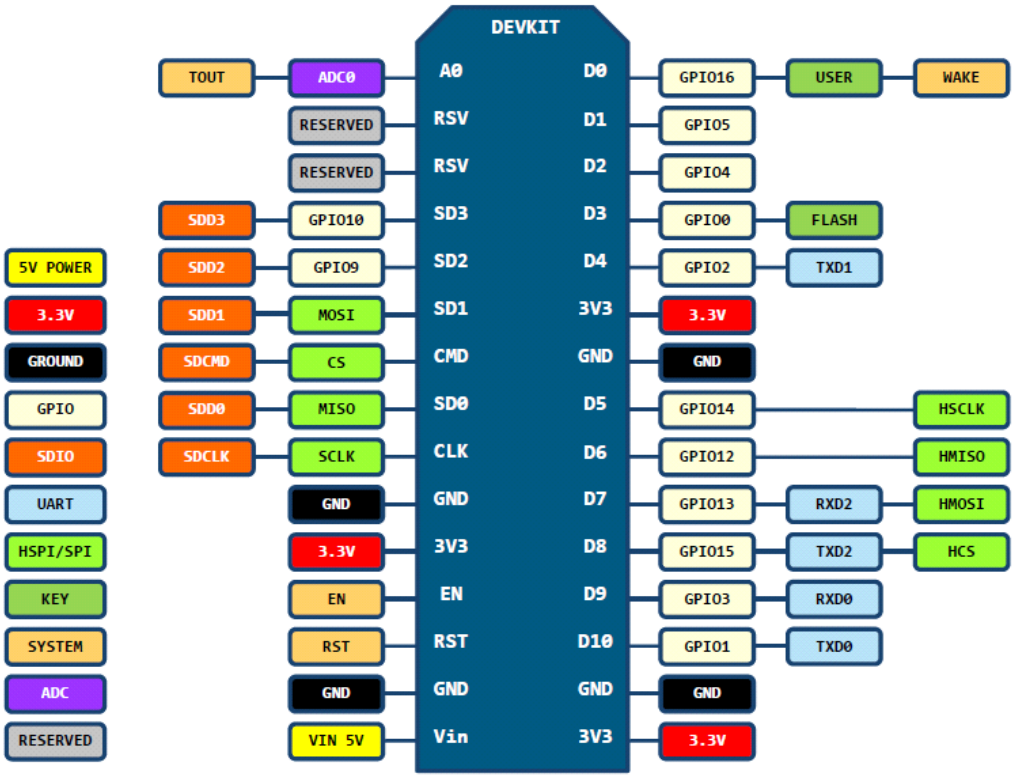
2.1.1.1.3 มียูเอสบี-ทีทีแอล (USB-TTL) ในตัวทำให้ใช้งานได้สะดวก

2.1.1.1.4 มีขาจีพีไอโอ (GPIO) ขาทุก ๆ ขาสามารถเป็นพีดับเบิลบูเอ็ม, ไอสแควร์ซี (PWM, I2C) และวันไวร์ (1-wire) ได้

2.1.1.1.5 สายอากาศ (PCB antenna) สำหรับรับส่งสัญญาณไร้สาย

2.1.1.1.6 ใช้คอนเนกเตอร์แบบไมโครยูเอสบี (micro USB) สำหรับจ่ายแรงดันไฟเลี้ยงหรือเท่ากับ +5 โวลต์ และสำหรับดาวน์โหลดเฟิร์มแวร์

ภาพที่ 2-1 โหนดเอ็มซียู (NodeMCU ESP8266)



ภาพที่ 2-2 ตำแหน่งขาของโหนดเอ็มซียู (Node MCU ESP8266)

2.1.2 โมดูลวัดอุณหภูมิและความชื้น DHT21 (AM2301)

เซนเซอร์วัดอุณหภูมิและความชื้นสัมพัทธ์แบบดิจิทัลโมเดล AM2301 / DHT21 เป็นเซนเซอร์วัดอุณหภูมิและความชื้นสัมพัทธ์แบบดิจิทัล และเชื่อมต่อด้วยสัญญาณเพียงเส้นเดียวแบบสองทิศทาง (bidirectional) ใช้แรงดันไฟเลี้ยงได้ในช่วง 3.3V ถึง 5.2V สามารถวัดค่าอุณหภูมิได้ในช่วง -40 ถึง 80°C ความละเอียดในการวัดอุณหภูมิและความชื้น คือ 0.5°C และ 0.1%RH และมีความแม่นยำ ±0.5°C และ ±3%RH ตามลำดับ ใช้ขาเชื่อมต่อเพียง 3 ขา ได้แก่ VCC, GND และ SDA (Serial Data)

ในการอ่านข้อมูลแต่ละครั้ง จะอ่านข้อมูลทั้งหมด 40 บิต แบ่งเป็น 16 บิต สำหรับค่าความชื้น 16 บิต สำหรับค่าอุณหภูมิ และ 8 บิต สำหรับตรวจสอบค่า Parity Bits เพื่อดูว่าอ่านค่าได้ถูกต้องหรือไม่ ลักษณะของเซ็นเซอร์ แสดงดังภาพที่ 2-3

ภาพที่ 2-3 DHT21 (AM2301)

2.1.2 โมดูลตรวจวัดแก๊สไวไฟ (MQ-2)

เซ็นเซอร์ก๊าซไวไฟและควัน สามารถตรวจจับความเข้มข้นของก๊าซที่ติดไฟในอากาศได้และให้ output ออกมาเป็น analog ตัวเซ็นเซอร์สามารถวัดความเข้มข้นของก๊าซไวไฟได้ระหว่าง 300 ถึง 10,000 ppm และทำงานได้ในอุณหภูมิ -20 ถึง 50 องศาเซลเซียส และกินไฟแค่ 150 mA 5V โมดูลตรวจวัดแก๊สไวไฟ (MQ-2) MQ-2 Combustible Gas Sensor Module เป็นโมดูลตรวจวัดแก๊ส ที่ไวต่อแก๊สไวไฟในกลุ่ม LPG, i-butane,propane, methane, alcohol, Hydrogen รวมไปถึงควันไฟที่เกิดจากการเผาไหม้ ลักษณะของเซ็นเซอร์ แสดงดังภาพที่ 2-4

ภาพที่ 2-4 โมดูลตรวจวัดแก๊สไวไฟ (MQ-2)

2.1.3 InfluxDB

InfluxDB เป็นฐานข้อมูลแบบอนุกรมเวลาคือฐานข้อมูลที่ถูกพัฒนาและปรับปรุงมาเพื่อเก็บข้อมูลในงานทางด้านอนุกรมเวลาโดยเฉพาะ ฐานข้อมูลแบบอนุกรมเวลาส่วนใหญ่แล้วจะถูกปรับแต่งให้เหมาะกับข้อมูลที่มีปริมาณมากและสามารถที่จะจัดเก็บจุดข้อมูลของอนุกรมเวลาและคำอธิบายเพิ่มเติมของข้อมูลนั้น ๆ เท่านั้น

ดังนั้นฐานข้อมูลแบบอนุกรมเวลาสามารถจัดการข้อมูลแบบอนุกรมเวลาได้เหมาะสมกว่าฐานข้อมูลอเนกประสงค์ทั่ว ๆ ไป เช่น ฐานข้อมูลแบบมีความสัมพันธ์ (Relational database) หรือฐานข้อมูลแบบไม่มีความสัมพันธ์ (Non-relational database)

อีกทั้งฐานข้อมูลแบบนี้ยังสามารถค้นหาข้อมูลแบบอนุกรมเวลา เลื่อนช่วงของเวลาที่ต้องการสอบถามรวมหลาย ๆ อนุกรมเวลาเข้ามาเป็นหนึ่งอนุกรมเวลา หรือคำนวณเพื่อหาค่าของจุดที่หายไปของอนุกรมเวลา (interpolation)

2.1.4 MQTT Protocol

MQTT (Message Queue Telemetry Transport) เป็นโปรโตคอลระหว่างเครื่องจักรกับเครื่องจักรสื่อสารกัน (machine-to-machine(M2M)/Internet of Things) โปรโตคอลนี้ใช้วิธีการเช่นเดียวกับ Message Queue แต่พัฒนาให้รองรับงานด้าน Internet of Things นอกจากจะรอรับและอ่านค่าแล้ว ยังสามารถสั่งงานอุปกรณ์ที่ใช้ MQTT ได้ด้วย การใช้งานโปรโตคอลนี้ต้องจะต้องมีอุปกรณ์ที่ฝัง MQTT Broker มาเพื่อทำหน้าที่รับส่งข้อมูลไปยังอุปกรณ์อื่นที่ใช้ MQTT เหมือนกัน MQTT Protocol จะประกอบไปด้วย Broker , Publisher และ Subscriber ซึ่งมีหน้าที่ดังนี้

2.1.4.1 Broker จะทำหน้าที่เป็นตัวกลางคอยจัดการ message โดยอ้างอิงด้วย topic ซึ่งโปรแกรมที่ใช้ทำหน้าที่เป็น MQTT Broker มีหลากหลายให้ใช้งาน เช่นโปรแกรม Mosquitto เป็น OpenSource MQTT Broker สนับสนุน MQTT Broker v3.1/3.1.1

2.1.4.2 Subscriber จะทำหน้าที่คอยดูการเปลี่ยนแปลงของ message ที่อ้างอิงด้วย topic เช่น หากหัวข้อที่สนใจมีการเปลี่ยนแปลงก็จะดึง data มาใช้งาน

2.1.4.3 Publisher จะทำหน้าที่ส่งข้อมูลไปยัง Topic นั้น ๆ

2.1.5 Telegraf

เป็นเอเจนต์ใช้ในการรับข้อมูลจาก Source ต่าง ๆ เข้ามาไม่ว่าจะเป็น Database อื่น ๆ หรือแม้กระทั่ง MQTT Message และรวมถึงดึงข้อมูลจาก InfluxDB และส่งกลับไปยังระบบอื่น

2.1.6 Grafana

เป็นซอฟต์แวร์สำหรับการทำ Data visualization & Monitoring และการจัดการเกี่ยวกับการแจ้งเตือนเหตุการณ์ต่าง ๆ เช่น การแจ้งเตือนผ่านอีเมล์ หรือ Line เป็นต้น โดยสามารถดึงข้อมูลออกมาได้ในระดับ Real time สามารถดึงข้อมูลมาจาก datasource ได้หลากหลาย เช่น influxdb, Prometheus, elasticsearch, AWS CloudWatch และ datasource ต่าง ๆ เป็นต้น

2.1.7 SNMP

โปรโตคอล SNMP ได้ถูกพัฒนาขึ้นในปี พ.ศ.2531 เนื่องจากมีความเจริญเติบโตในการใช้อุปกรณ์ที่สนับสนุนโปรโตคอล TCP/IP อย่างสูง โปรโตคอล SNMP ถูกออกแบบให้มีฟังก์ชั่นและการทำงานแบบง่าย ๆ โดยมีจุดประสงค์หลักเพื่อให้ผู้ดูแลระบบเครือข่ายสามารถเข้ามาจัดการอุปกรณ์เครือข่ายได้จากระยะไกลโดยง่ายโดยทั่วไปแล้วโปรโตคอล SNMP ใช้จัดการกับอุปกรณ์ประเภทเรา-เตอร์ หรือสวิตช์เท่านั้น แต่อันที่จริงแล้วโปรโตคอล SNMP ใช้ได้กับอุปกรณ์บนเครือข่ายทุกประเภท ไม่ว่าจะเป็น ตัวเซิร์ฟเวอร์ประเภทต่าง ๆ เช่น UNIX หรือกับคอมพิวเตอร์ทั่วไปที่ใช้ระบบปฏิบัติการ

Microsoft Windows เครื่องใช้สำนักงาน เช่น ปริ๊นเตอร์, โมเด็ม รวมแม้กระทั่งเครื่องสำรองไฟฟ้า UPS

สำหรับมาตรฐานโปรโตคอล SNMP จะมีหน่วยงานสากลในชื่อ IETF (InternetEngineering Task Force) คอยกำกับดูแล โดยโปรโตคอล SNMP มีเวอร์ชันดังนี้

2.1.7.1 SNMP Version 1 (SNMPv1) เป็นมาตรฐานปัจจุบัน และเป็นที่นิยมเพราะความง่ายของโปรโตคอล SNMP ซึ่งถูกระบุใน RFC1157 และได้รับอนุมัติให้เป็นมาตรฐานที่สมบูรณ์ ระดับความปลอดภัย SNMPv1 จะขึ้นอยู่กับคอมมิวนิตี้สตริง (Community String) ที่ทำหน้าที่เหมือนรหัสผ่านหรือพาสเวิร์ด (Password) โดยที่จริงแล้วเป็นเพียงข้อความแบบธรรมดา(Plain Text) ที่บ่งบอกถึงสิทธิการเข้าไปจัดการอุปกรณ์เครือข่าย โดยปกติคอมมิวนิตี้จะมีสามประเภทนั้นคือ อ่านอย่างเดียว (Read-only), อ่านเขียน (Read-write) และแทรป (Trap)

2.1.7.2 SNMP Version 2 (SNMPv2) คือ เวอร์ชันที่ทำงานบนคอมมิวนิตี้ที่ได้รับการปรับปรุง ในทางเทคนิคเรียกว่า SNMPv2c ซึ่งระบุใน RFC1905, RFC1906 และ RFC1907 และอยู่ในขั้นตอนทดสอบใช้งาน แต่ก็มีบางผู้ผลิตได้นำมาใช้งานในอุปกรณ์ SNMPv2 ออกแบบมาเพื่อแก้ไขข้อด้อยของ SNMPv1 ในเรื่องการร้องข้อมูลปริมาณมากและปัญหาในการส่งข้อมูล แบบแทรป

2.1.7.3 SNMP Version 3 (SNMPv3) เป็นเวอร์ชันถัดไปของโปรโตคอล SNMP ที่ถูกคาดหวังให้เป็นมาตรฐานที่สมบูรณ์ ซึ่งในปัจจบันอยู่ในสถานะเสนอระบุใน RFC1905, RFC1906, RFC1907, RFC2571, RFC2572, RFC2573, RFC2574 และ RFC2575 โดยมุ่งเน้นการเพิ่มระดับความปลอดภัยของโปรโตคอล SNMP

2.1.8 Line API

Line API คือ Application Program Interface (API) ที่อนุญาตให้โปรแกรมสามารถสื่อสารระหว่างกันได้ หรือเป็นช่องทางสำหรับขอใช้บริการคำสั่ง จากระบบปฏิบัติการ(OS) หรือโปรแกรมอื่น ๆ ซึ่งใช้งานโดยติดตั้งฟังก์ชันและเรียกใช้งานตามเอกสารที่เขียนไว้

2.1.9 ข้อตกลงระดับการให้บริการ (Service Level Agreement : SLA) คือ ข้อตกลงระหว่างผู้ให้บริการและลูกค้า ประกอบไปด้วยลักษณะเฉพาะของการบริการ คุณภาพความพร้อมและความรับผิดชอบ ดังนั้นปัญหาหลักสำหรับผู้ให้บริการคือทำอย่างไรให้ลูกค้ามั่นใจในการให้บริการ ซึ่งโดยทั่วไปจะใช้วิธีตรวจสอบ Service Availability, MTBF และ MTTR ผู้ให้บริการต้องมีวิธีการตรวจสอบความผิดปกติที่แม่นยำและรวดเร็ว โดยต้องทราบรายละเอียดโครงสร้างที่ขึ้นต่อกันของบริการต่าง ๆ (ปัญญา, 2556)

2.1.9.1 ความพร้อมใช้งาน (Availability) ความสามารถที่ระบบสามารถใช้งานได้ ซึ่งเกิดจากการทำงานของบริการต่าง ๆ ร่วมกัน เช่น Web Application เกิดจากการทำงานร่วมกันของ DNS Service, Web application firewall, Authentication service, Storage service, ระบบจ่ายกระแสไฟฟ้า สามารถคำนวณหาความพร้อมใช้งานได้ดังนี้

$$\text{Availability} = \frac{\text{MTTF}}{\text{MTTR} + \text{MTTF}} = \frac{\text{MTTF}}{\text{MTBF}} = \frac{\text{Service Availability}}{\text{Total time}} \qquad (2.1)$$

โดยที่

MTTF (Mean Time To Failure) คือ เวลาเฉลี่ยก่อนการเกิด Failure

MTTR (Mean Time To Repair) คือ เวลาเฉลี่ยในการซ่อมแซมระบบ

MTBF (Mean Time Between Failure) = (MTTR+MTTF) คือเวลาเฉลี่ยระหว่างการเสียหาย

Service Availability คือ ช่วงเวลาที่ระบบสามารถใช้งานได้

Total time คือ เวลารวมทั้งหมดของระบบ

2.2 ผลงานวิจัยที่เกี่ยวข้อง

ทัศนะ, สุรสิทธิ์, วิโรจน์และศุภชัย (2547) นำเสนอการออกแบบสร้างตัวควบคุมอุณหภูมิและความชื้นแบบฟัซซีลอจิกสำหรับตู้ฟักไข่ขนาดเล็ก โดยใช้ไมโครคอนโทรลเลอร์ 8 บิต เบอร์ PIC16F84 เป็นหน่วยประมวลผลกลางรับค่าอุณหภูมิและความชื้นภายในตู้ฟักผ่านเซนเซอร์เข้ามาที่หน่วยประมวลผลกลาง เพื่อตัดสินใจ ส่งสัญญาณควบคุมให้รีเลย์ตัดต่อหลอดไฟและพัดลมเพื่อรักษาอุณหภูมิและความชื้นภายในตู้ให้อยู่ในย่านที่กำหนดไว้

ธวัชชัย (2549) นำเสนอโปรแกรมเฝ้ามองและแจ้งเตือนเครือข่ายในเวลาจริง ซึ่งโปรแกรมทำหน้าที่ดักจับข้อมูลต่างๆ ที่วิ่งอยู่ในเครือข่ายมาแสดงผลในลักษณะที่ผู้ดูแลระบบเข้าใจ เช่น แสดงการรับส่งข้อมูลระหว่างตอนทางและปลายทาง แสดงรายละเอียดของข้อมูลที่รับส่งกัน แสดงกราฟ สามารถป้อนข้อมูลของเว็บไซต์ที่ไม่พึงประสงค์เพื่อตรวจสอบเครื่องลูกข่ายว่าใช้งานเว็บไซต์ที่ไม่พึงประสงค์หรือไม่ สามารถตรวจสอบได้ว่าเครื่องลูกข่ายผิดปกติหรือไม่ ในกรณีที่โดนไวรัสสามารถส่งข้อความไปบอกเครื่องลูกข่ายได้และสามารถสั่งปิดพอร์ตได้ การประเมินประสิทธิภาพของระบบโดยผู้เชี่ยวชาญระบบ พบว่ามีระดับของความพึงพอใจในระดับดีและสามารถที่จะนำไปใช้ในงานจริงได้อย่างเหมาะสม

ประทีป (2556) นำเสนอการการใช้เทคโนโลยีเข้ามาสนับสนุนการดำเนินงานเพื่อช่วยในการให้บริการลูกค้า ซึ่งจะนำกรอบการทำงานไอทิล (ITIL Framework) ไปพัฒนากระบวนการทำงานให้มีความรวดเร็วและมีประสิทธิภาพมากขึ้น โดยใช้ซอฟต์แวร์ที่เป็นมาตรฐานสากลที่สามารถใช้งานได้จริงของบริษัทไอบีเอ็ม ที่ชื่อ “IBM Tivoli Service Request Manager” เวอร์ชัน 7.2 ในการจำลองเหตุการณ์ต่างๆ ทางธุรกิจ โดยมีสมมุติฐานกระบวนการทางธุรกิจที่จะครอบคลุมกระบวนการทางธุรกิจ 4 กระบวนการ คือ 1) การปฏิบัติตามคำร้องขอ (Request Fulfilment) 2) การบริหารจัดการเหตุการณ์ (Incident Management) 3) การบริหารจัดการปัญหา (Problem Management) และ 4) การบริหารจัดการระดับการให้บริการ (Service Level Management) พบว่า ช่วยปรับปรุงกระบวนการทำงานให้เป็นมาตรฐานสากลและสามารถให้การบริการอินเทอร์เน็ตที่มีคุณภาพแก่ลูกค้าตามที่สัญญาไว้

เอกลักษณ์และธานินทร์ (2557) นำเสนอการพัฒนาแอพพลิเคชันแอนดรอยด์ร่วมกับระบบสมองกลฝังตัว เพื่อควบคุมระบบไฟฟ้าและอุณหภูมิภายในโรงเรือน โดยการทำงานแบ่งเป็น 3 ส่วน คือ 1) สามารถควบคุมผ่านบอร์ดควบคุมของโรงเรือนโดยตรง 2) การควบคุมผ่านบลูทูธแบบไร้สายบนแอพพลิเคชันและ 3) ระบบส่งข้อความสั้นที่สามารถแจ้งเตือนผู้ใช้ในระยะไกลได้ ผลจากการ

ทดลองทำให้ได้ระบบควบคุมระบบไฟฟ้าและอุณหภูมิในโรงเรือนที่สามารถควบคุมอุณหภูมิภายในแบบอัตโนมัติ สามารถเก็บข้อมูล เรียกดูข้อมูล และควบคุมอุปกรณ์ผ่านแอพพลิเคชันแอนดรอยด์ด้วยบลูทูธในระยะใกล้ 20 เมตรและระยะไกลด้วยข้อความสั้น

บทที่ 3
วิธีดำเนินงานวิจัย

การดำเนินการโครงงานวิจัย “ระบบเฝ้าระวังความปลอดภัยของเซิร์ฟเวอร์แร็ค” แบ่งออกเป็นส่วนต่างๆ ดังนี้

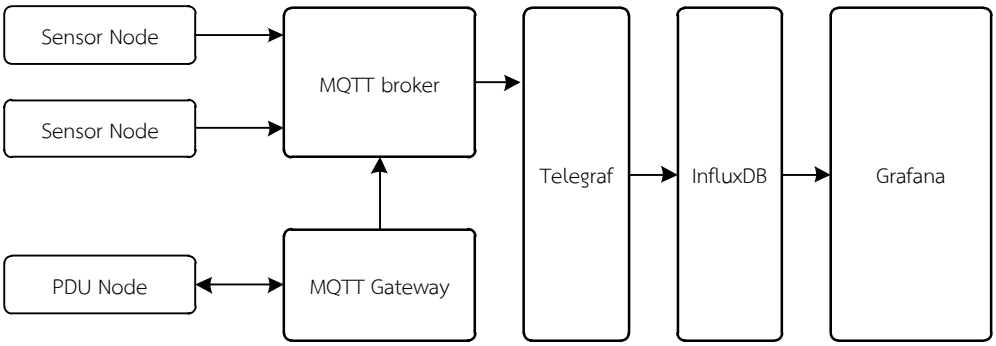
3.1 หลักการทำงานของโปรแกรม

3.2 การออกแบบอุปกรณ์ฮาร์ดแวร์และการเขียนโปรแกรม

3.3 การเขียนโปรแกรม PHP ติดต่อกับ PDU (Power Distribution Unit)

3.4 การออกแบบส่วนติดต่อกับผู้ใช้งานบนเว็บด้วย Grafana

3.1 หลักการทำงานของโปรแกรม



ภาพที่ 3-1 บล๊อกไดอะแกรมของระบบ

จากภาพที่ 3-1 Sensor node จะทำการวัดค่าอุณหภูมิและความชื้น, PDU node หรือปลั้กไฟจะถูกอ่านค่าโดย MQTT Gateway แล้วจัดการข้อมูลให้อยู่ในรูปแบบสำหรับส่งเข้า MQTT broker หลังจากนั้นข้อมูลจะส่งเข้า MQTT broker เพื่อให้ Telegraf ส่งต่อไปที่ฐานข้อมูล InfluxDB และแสดงผลในรูปแบบตัวเลขและกราฟด้วย Grafana ส่วนประกอบของระบบประกอบด้วย Sensor node, PDU node, MQTT Gateway, MQTT broker, Telegraf, InfluxDB และ Grafana ซึ่งส่วนประกอบต่างๆ มีหน้าที่ดังนี้

Sensor Node ทำหน้าที่วัดอุณหภูมิ ความชื้น และตรวจจับควันไฟ ภายในเซิร์ฟเวอร์แร็ค

PDU Node ทำหน้าที่ในการตรวจวัดการใช้พลังงานไฟฟ้า เช่น แรงดันไฟฟ้า การใช้งานกระแสไฟฟ้า กำลังไฟฟ้า เป็นต้น

MQTT Gateway ทำหน้าที่เป็นตัวกลางระหว่าง MQTT broker และ PDU โดยติดตั้งบน Linux OS พร้อมติดตั้ง Mosquitto Library, PHP และโปรแกรมต่าง ๆ ที่เกี่ยวข้อง เพื่ออ่านค่าการใช้งานพลังงานไฟฟ้าต่าง ๆ ผ่าน SNMP แล้วจึง publish ไปยัง MQTT broker โค้ดโปรแกรมแสดงดังนี้

```
$COMMUNITY_STRING = "public"; 1
$PDU_MASTER_IP = "192.168.66.200"; 2
$PDUCURRENT1 = explode(" ",snmpget( $PDU_MASTER_IP,
$COMMUNITY_STRING,"LIEBERT-GP-PDU-MIB::lgpPduPsEntryEcNeutral.1.1")); 3
$PDUObj->PduCurrent1 = $PDUCURRENT1[1]*0.1; 4
define('BROKER', '202.44.33.90'); 5
define('PORT', 1883); 6
$client = new Mosquitto\Client();
$client->setCredentials('admin','xxxxx'); 7
$client->connect(BROKER, PORT);
$client->publish('DC/RACK/A/SENSOR/PDUCurrent',$PDUObj->PduCurrent1, 0, false); 8
sleep(1);
```

หมายเลข 1 Community String สำหรับสื่อสารเชื่อมต่อข้อมูลกับ PDU ผ่าน SNMP
หมายเลข 2 หมายเลขไอพีแอดเดรสของ PDU
หมายเลข 3 เชื่อมต่อ SNMP เพื่ออ่านค่ากระแสไฟฟ้าจาก PDU
หมายเลข 4 แปลงหน่วยกระแสไฟฟ้าให้มีหน่วยเป็น แอมป์
หมายเลข 5 กำหนด MQTT broker ในที่นี้คือ 202.44.33.90
หมายเลข 6 กำหนดหมายเลขพอร์ตที่จะเชื่อมต่อ
หมายเลข 7 กำหนดค่าต่าง ๆ และเชื่อมต่อไปที่ MQTT broker
หมายเลข 8 ส่งค่ากระแสไฟฟ้าที่อ่านได้จาก PDU ไปยัง MQTT broker โดยผ่าน Topic

MQTT broker ทำหน้าที่เป็นตัวกลางให้เซ็นเซอร์ *Publish* ข้อมูล และให้ Telegraf มาทำการ *Subscribe* ผ่าน Topic ซึ่งกำหนดรูปแบบสำหรับใช้งานดังนี้

```
DC/RACK/หมายเลขตู้แร็ค/SENSOR/เซ็นเซอร์ที่ต้องการวัดค่า
```

หมายเลขตู้แร็ค คือ A, B
เซ็นเซอร์ที่ต้องการวัดค่า คือ Temperature, Smoke, Humidity, Voltage เป็นต้น

Telegraf ทำหน้าที่อ่านข้อมูลจากเซ็นเซอร์ผ่านการ *Subscribe* จาก MQTT broker โดยระบุ Topic เป็น ***DC/RACK/#*** หมายถึงอ่านข้อมูลจาก Topic ที่ขึ้นต้นด้วย DC/RACK/ ทั้งหมด จากนั้นจึงส่งต่อข้อมูลไปเก็บไว้ที่ InfluxDB การคอนฟิกค่า teletgraf.conf แสดงดังนี้

```
[[inputs.mqtt_consumer]]
servers = ["tcp://localhost:1883"]
topics = [ "DC/RACK/#", ]

[[outputs.influxdb]]
urls = ["http://localhost:8086"]
database = "telegraf"
username = "admin"
password = "xxxx"
```

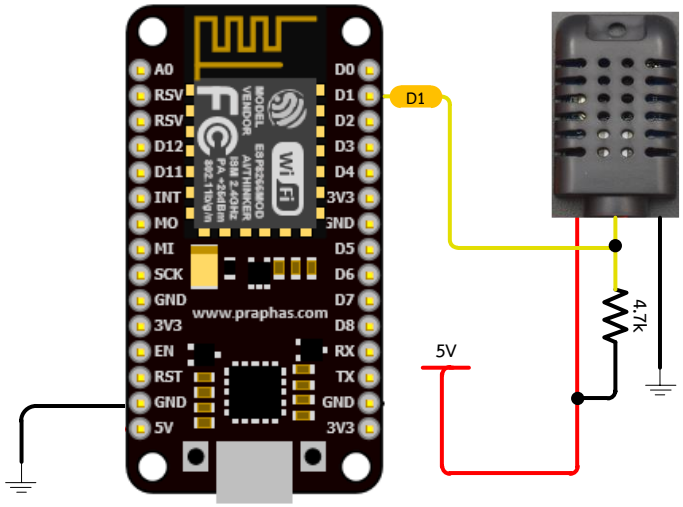
InfluxDB ทำหน้าที่เป็นฐานข้อมูลใช้จัดเก็บข้อมูลจากเซ็นเซอร์ต่าง ๆ ข้อมูลที่จัดเก็บจะแยกเป็นแต่ละเซ็นเซอร์ (Topic) และให้ Grafana นำข้อมูลไปวิเคราะห์และแสดงผล ตัวอย่างข้อมูลที่ถูกจัดเก็บไว้แสดงดังนี้

```
time                host topic                           value
----                ---- -----                           -----
1521797594386376393 iOT  DC/RACK/A/SENSOR/TEMPERATURE/1 5
1521797594578864857 iOT  DC/RACK/A/SENSOR/TEMPERATURE/1 5
```

Grafana ทำหน้าที่ Data Visualize และ Alert โดยดึงข้อมูลที่เป็นลักษณะ Time-Series จาก InfluxDB ไปวิเคราะห์และแสดงผลเป็นกราฟต่าง ๆ แบบ Real Time

3.2 การออกแบบอุปกรณ์ฮาร์ดแวร์และการเขียนโปรแกรม

3.2.1 เซ็นเซอร์วัดอุณหภูมิและความชื้นสัมพัทธ์ (Digital Temperature and Humidity Sensor) เซ็นเซอร์วัดอุณหภูมิและความชื้นสัมพัทธ์ใช้ไฟเลี้ยงวงจรอยู่ที่ 3 ถึง 5 โวลต์ กระแสสูงสุดที่ 2.5 มิลลิแอมป์ เหมาะสำหรับวัดความชื้นระดับ 20 ถึง 80 เปอร์เซ็นต์ โดยมีความผิดพลาดการวัดไม่เกิน 5 เปอร์เซ็นต์ เหมาะสำหรับวัดอุณหภูมิ 0 ถึง 50 องศาเซลเซียส โดยมีความผิดพลาดในการวัดไม่เกินบวกลบ 2 องศาเซลเซียส แสดงการเชื่อมต่อดังภาพที่ 3-2



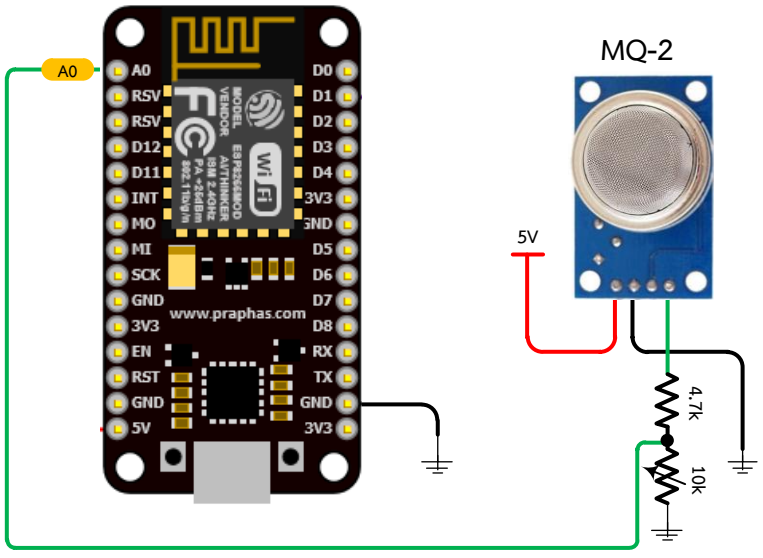
ภาพที่ 3-2 การเชื่อมต่อ ESP8266 กับเซ็นเซอร์วัดอุณหภูมิและความชื้น

ตัวอย่างโค้ดโปรแกรมอ่านค่าอุณหภูมิและความชื้น แล้ว *Publish* ไปยัง MQTT broker

```
#include "DHTesp.h"
#include <PubSubClient.h>
#define MQTT_SERVER   "mqtt.icit.kmutnb.ac.th"     ; เซิร์ฟเวอร์ MQTT
#define MQTT_PORT     1883                         ; หมายพอร์ตที่ใช้
#define MQTT_USERNAME "admin"                      ; ยูสเซอร์สำหรับเชื่อมต่อ
#define MQTT_PASSWORD "xxxx"                       ; รหัสผ่านสำหรับเชื่อมต่อ
#define MQTT_NAME     "ESP_NODE"                   ; ชื่อ MQTT ใช้เชื่อมต่อ

DHTesp dht;                                        ; ประกาศตัวแปร dht
PubSubClient mqtt(client);                         ; ประกาศตัวแปร mqtt
dht.setup(5);                                      ; กำหนดขา 5 สื่อสารข้อมูล

char StrTemp[5];                                   ; ตัวแปรเก็บอุณหภูมิ
char StrHumi[5];                                   ; ตัวแปรเก็บความชื้น

float humidity = dht.getHumidity();                ; อ่านค่าความชื่น
float temperature = dht.getTemperature();          ; อ่านค่าอุณหภูมิ

humidity = humidity+(humidity*0.01);               ; Calibrate
temperature = temperature-(temperature*0.081);     ; Calibrate
```

```
dtostrf(temperature,4,1,StrTemp);              ; แปลงไปข้อความ
dtostrf(humidity,4,1,StrHumi);                 ; แปลงไปข้อความ

mqtt.setServer(MQTT_SERVER, MQTT_PORT);        ; กำหนดค่า
mqtt.connect(MQTT_NAME, MQTT_USERNAME, MQTT_PASSWORD)
; เชื่อมต่อไปยัง MQTT broker
mqtt.publish("DC/RACK/A/SENSOR/TEMPERATURE1",StrTemp);   ; Publish ค่า
delay(1000);                                             ; หน่วงเวลา
mqtt.publish("DC/RACK/A/SENSOR/HUMIDITY1",StrHumi);      ; Publish ค่า
delay(1000);                                             ; หน่วงเวลา
```

3.2.2 เซ็นเซอร์ตรวจจับควัน (MQ-2) MQ-2 เป็น Sensor ตรวจสอบปริมาณก๊าซไวไฟ และควันไฟ เช่น LPG, i-butane, propane, methane, alcohol, hydrogen, smoke ในอากาศที่มีความเข้มข้น 300 ถึง 10000 ppm (part per million) ใช้ไฟเลี้ยงวงจรอยู่ที่ 5 โวลต์ โดยเมื่อมีกลุ่มควัน ก๊าซ แก๊ส Sensor จะมีการส่งสัญญาณ analog ไปยัง ESP8266 ที่ขา A0 หากมีความหนาแน่นของ ก๊าซ แก๊ส ควัน ก็จะมีค่า analog ที่สูงตามไปด้วย ทั้งนี้หากค่า CO มากกว่า 300 ppm อาจก่อให้เกิดเปลวไฟได้ แสดงดังภาพที่ 3-3

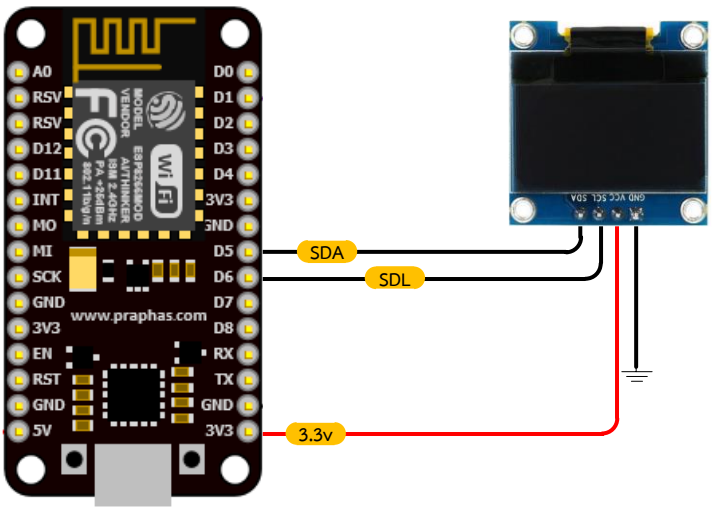


ภาพที่ 3-3 การเชื่อมต่อ ESP8266 กับเซ็นเซอร์ตรวจจับควัน (MQ2)

ตัวอย่างโค้ดโปรแกรมอ่านค่าอุณหภูมิและความชื้น แล้ว *Publish* ไปยัง MQTT broker

```
#include <PubSubClient.h>
#define MQTT_SERVER   "mqtt.icit.kmutnb.ac.th"     ; เซิร์ฟเวอร์ MQTT
#define MQTT_PORT     1883                         ; หมายพอร์ตที่ใช้
#define MQTT_USERNAME "admin"                      ; ยูสเซอร์สำหรับเชื่อมต่อ
#define MQTT_PASSWORD "xxxx"                       ; รหัสผ่านสำหรับเชื่อมต่อ
#define MQTT_NAME     "ESP_NODE"                   ; ชื่อ MQTT ใช้เชื่อมต่อ

PubSubClient mqtt(client);                         ; ประกาศตัวแปร mqtt
char StrSmoke[5];
int SmokePin = A0;
float SmokeDetec = 0;

SmokeDetec = (analogRead(SmokePin)*10000)/1023;          ; อ่านค่าควัน (PPM)
dtostrf(SmokeDetec,4,1,StrSmoke);
mqtt.setServer(MQTT_SERVER, MQTT_PORT);                  ; กำหนดค่า
mqtt.connect(MQTT_NAME, MQTT_USERNAME, MQTT_PASSWORD)
; เชื่อมต่อไปยัง MQTT broker
mqtt.publish("DC/RACK/A/SENSOR/SMOKE1",StrSmoke);        ; Publish ค่า
delay(1000);                                             ; หน่วงเวลา
```

3.2.3 OLED (Organic Light Emitting Diodes) คือจอภาพที่มีลักษณะคล้ายแผ่นฟิล์ม ซึ่งมีส่วนประกอบเป็นสารอินทรีย์ที่สามารถเปล่งแสงเองได้เมื่อได้รับพลังงานไฟฟ้า เรียกว่า กระบวนการอิเล็คโทรลูมิเนเซนส์ (Electroluminescence) โดยที่ไม่ต้องพึ่งพาแสง Backlight และจะไม่มีการเปล่งแสดงในบริเวณที่เป็นภาพสีดำ ส่งผลให้สีดำนั้นดำสนิทอีกทั้งยังช่วยพลังงาน จอแสดงผลแบบ OLED LCD หน้าจอ 128x64 ขนาด 0.96" เชื่อมต่อแบบ I2C ใช้ไฟได้ทั้ง 3.3V หรือ 5V ให้จอสว่างแสดงผลเห็นได้อย่างชัดเจนและประหยัดไฟ สามารถวาดภาพกราฟฟิกส์เป็นรูปต่าง ๆ หรือเมนูตามที่ต้องการได้ แสดงการเชื่อมต่อกับจอ OLED กับ ESP8266 ภาพที่ 3-4



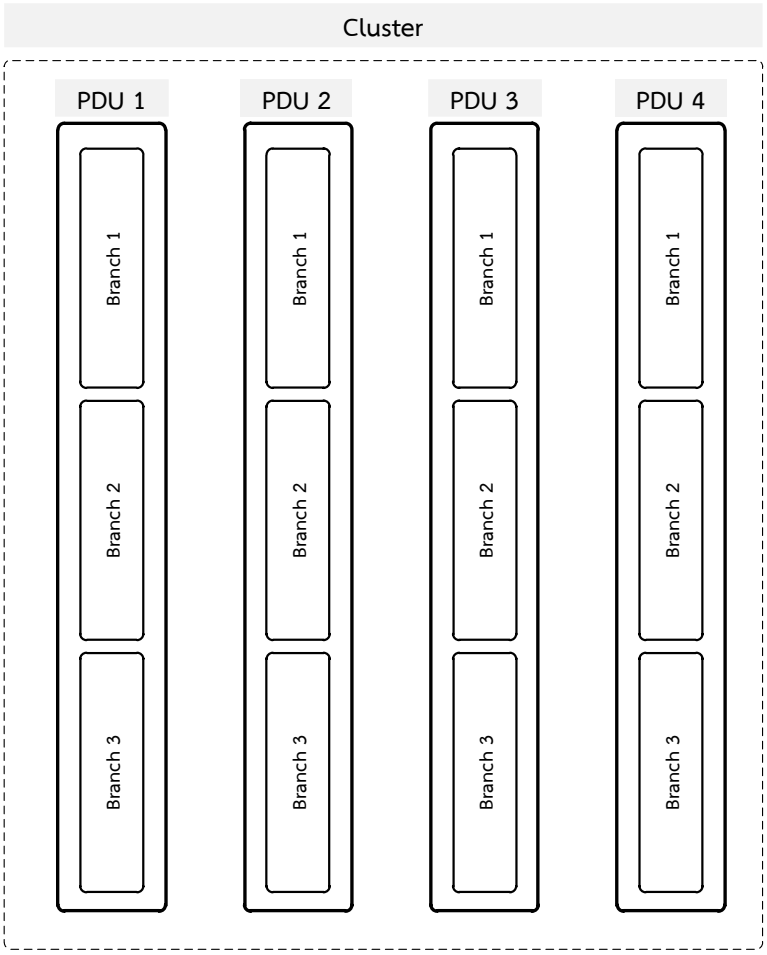
ภาพที่ 3-4 การเชื่อมต่อ ESP8266 กับจอ OLED

ตัวอย่างโค้ดโปรแกรมแสดงผลผ่านจอ OLED

```
display.clearDisplay();                 ; เคลียหน้าจอ
display.setFont(&FreeSansBold9pt7b);    ; กำหนดตัวอักษร
display.setTextSize(2);                 ; กำหนดขนาดอักษร
display.setTextColor(WHITE);            ; กำหนดสีตัวอักษร
display.setCursor(0,55);                ; กำหนดตำแหน่งที่จะแสดง
display.println(temperature,1);         ; นำค่าจากตัวแปร temperature
display.display();                      ; แสดงผล
delay(5000);                            ; หน่วงเวลา 5 วินาที
```

3.3 การเขียนโปรแกรม PHP ติดต่อกับ PDU (Power Distribution Unit)

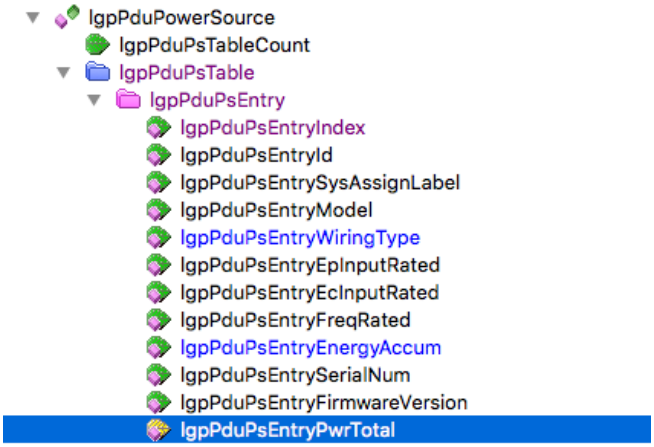
3.3.1 Liebert MPS Switched Rack PDU คืออุปกรณ์ควบคุมและแจกจ่ายกระแสไฟฟ้าให้กับอุปกรณ์ไฟฟ้าต่าง ๆ โดยอุปกรณ์สามารถวัดค่าการใช้งานไฟฟ้ารวมถึงเปิดให้โปรแกรมอื่นเข้ามาอ่านค่าได้โดยผ่านโปรโตคอล SNMP (Simple Network Management Protocol) ในงานวิจัยนี้เขียนโปรแกรมด้วยภาษา PHP เพื่ออ่านค่าดังนี้ เปอร์เซ็นต์การใช้งาน(Percent Current Utilization), ความถี่ (Line Frequency), โหลดกำลังไฟฟ้า (Total Input Power), การใช้พลังงานไฟฟ้าสะสม (Accumulated Energy) และการใช้งานกระแสไฟฟ้า(Neutral Current Measurement) โดยรูปแบบโครงสร้างการเชื่อมต่อของ PDU แสดงดังภาพที่ 3-5



ภาพที่ 3-5 รูปแบบโครงสร้างการเชื่อมต่อของ PDU

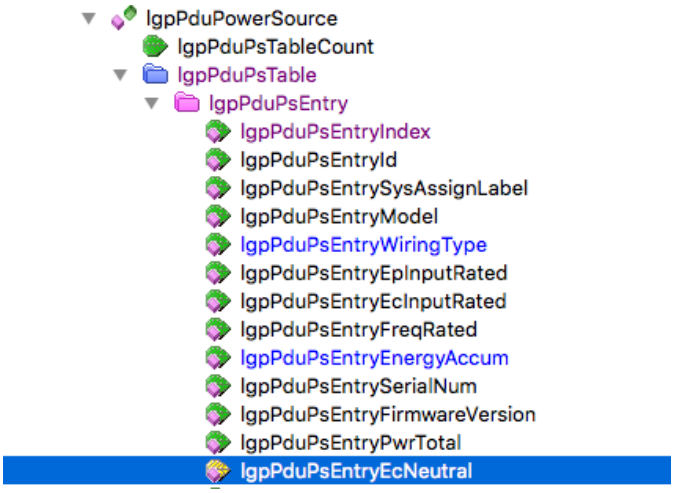
การเขียนโปรแกรมภาษา PHP สำหรับอ่านค่าจาก PDU จะอ่านค่าต่าง ๆ ออกมาได้อย่างถูกต้อง จำเป็นต้องทราบข้อมูลหมายเลข OID โดยสามารถดูได้จาก MIB (Management Information Base) ซึ่งเป็นฐานข้อมูลที่บันทึกตำแหน่งของออบเจ็กต์ทั้งหมด โดยที่ MIB จะจัดเรียงชื่อ, หมายเลข OID, ชนิดข้อมูล, สิทธิการอ่านและเขียนรวมทั้งคำอธิบายสั้น ๆ สำหรับแต่ละออบเจ็กต์ ในงานวิจัยนี้จะดำเนินการติดตั้ง snmp-mibs-downloader สำหรับโหลด MIB ของ PDU เพื่อการเขียนโปรแกรม ตัวอย่างรายละเอียด MIB แสดงได้ดังนี้

- โหลดกำลังไฟฟ้า (Total Input Power) แสดงดังภาพที่ 3-6



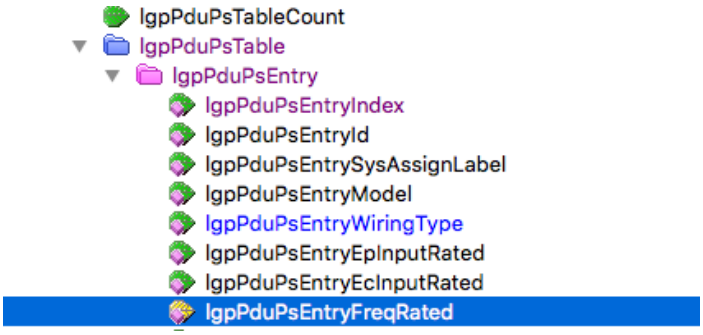
ภาพที่ 3-6 OID โหลดกำลังไฟฟ้า

- การใช้งานกระแสไฟฟ้า (Neutral Current Measurement) แสดงดังภาพที่ 3-7



ภาพที่ 3-7 OID การใช้งานกระแสไฟฟ้า

- ความถี่ (Line Frequency) แสดงดังภาพที่ 3-8



ภาพที่ 3-8 OID ความถี่

- เปอร์เซ็นต์การใช้งาน (Percent Current Utilization)

Percent Current Utilization = (กระแสไฟฟ้า / 32)*100

ตัวอย่าง โค้ดโปรแกรมภาษา PHP ติดต่อ PDU สำหรับตรวจสอบสถานะการเชื่อมต่อ

```
$COMMUNITY_STRING = "xxxx";              ; Community string สำหรับเชื่อมต่อ
 $PDU_MASTER_IP = "192.168.66.200";              ; IP ของ PDU

 snmp_read_mib('MIB/LIEBERT_GP_REG.MIB');              ; โหลด MIB เข้ามาใช้งาน
 snmp_read_mib('MIB/LIEBERT_GP_PDU.MIB');
 snmp_read_mib('MIB/LIEBERT_GP_POWER.MIB');

   $ClusterStatus = snmpget( $PDU_MASTER_IP, $COMMUNITY_STRING,
   'LIEBERT-GP-PDU-MIB::lgpPduGrpSysStatus.0');    ; ตรวจสอบสถานะการเชื่อมต่อ
```

ตัวอย่างโค้ดโปรแกรมสำหรับอ่านค่า โหลดกำลังไฟฟ้า (Total Input Power)

```
$PDUPOWER1 = explode(" ",snmpget( $PDU_MASTER_IP,
                $COMMUNITY_STRING,"LIEBERT-GP-PDU-MIB::lgpPduPsEntryPwrTotal.1.1"));
```

$PDU_MASTER_IP คือ ไอพีแอดเดรสของ PDU
$COMMUNITY_STRING คือ Community string สำหรับเชื่อมต่อ PDU
LIEBERT-GP-PDU-MIB::lgpPduPsEntryPwrTotal.**1**.1 คือ **1**.1 อ่านค่าโหลดกำลังไฟฟ้าอินพุตของ PDU ตัวที่ 1 หากต้องการอ่านค่า PDU ตัวที่ 2 ให้ใช้ **2**.1 เป็นต้น

ตัวอย่างโค้ดโปรแกรมสำหรับอ่านค่า การใช้งานกระแสไฟฟ้า (Neutral Current Measurement)

```
$PDUCURRENT1 = explode(" ",snmpget( $PDU_MASTER_IP,
$COMMUNITY_STRING,"LIEBERT-GP-PDU-MIB::lgpPduPsEntryEcNeutral.1.1"));
```

$PDU_MASTER_IP คือ ไอพีแอดเดรสของ PDU
$COMMUNITY_STRING คือ Community string สำหรับเชื่อมต่อ PDU
LIEBERT-GP-PDU-MIB::lgpPduPsEntryEcNeutral.**1** .1 คือ **1**.1 อ่านการใช้งานกระแสไฟฟ้าของ PDU ตัวที่ 1 หากต้องการอ่านค่า PDU ตัวที่ 2 ให้ใช้ **2**.1 เป็นต้น

ตัวอย่างโค้ดโปรแกรมสำหรับอ่านค่า ความถี่ (Line Frequency)

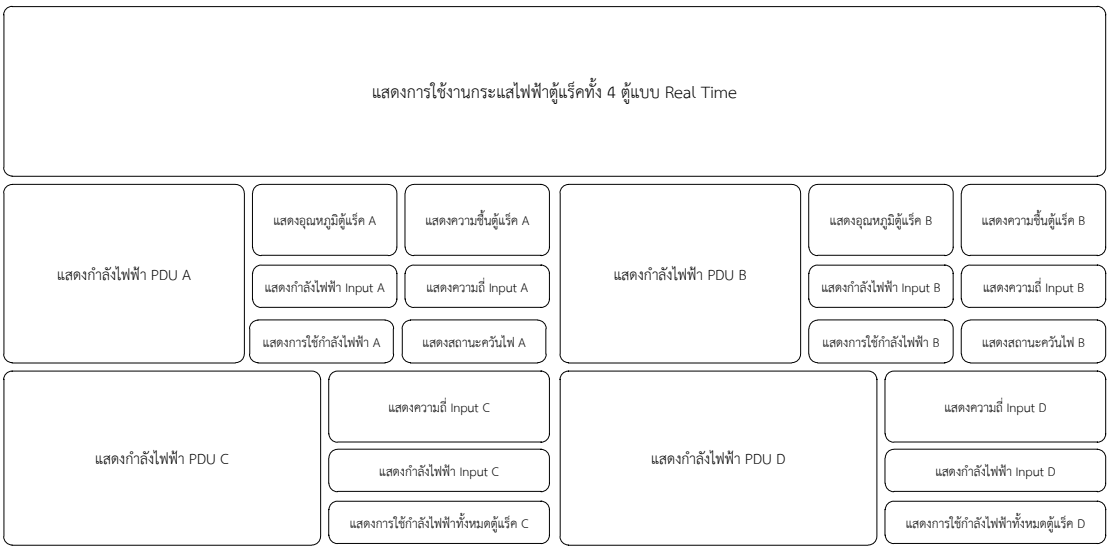
```
$PDUCURRENT1 = explode(" ",snmpget( $PDU_MASTER_IP,
$COMMUNITY_STRING,"LIEBERT-GP-PDU-MIB::lgpPduPSEntryFreqRated.1.1"));
```

$PDU_MASTER_IP คือ ไอพีแอดเดรสของ PDU

$COMMUNITY_STRING คือ Community string สำหรับเชื่อมต่อ PDU

LIEBERT-GP-PDU-MIB::lgpPduPSEntryFreqRated.**1**.1 คือ **1**.1 อ่านการใช้งานความถี่ของ PDU ตัวที่ 1 หากต้องการอ่านค่า PDU ตัวที่ 2 ให้ใช้ **2**.1 เป็นต้น

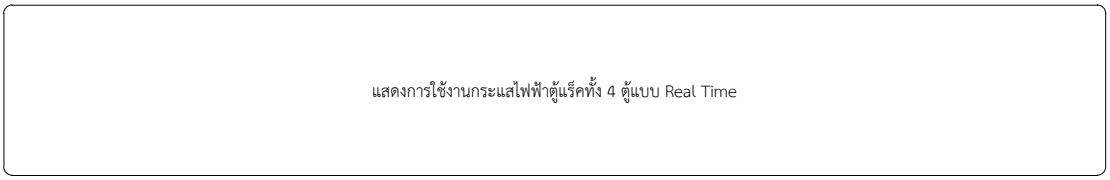
3.4 การออกแบบส่วนติดต่อกับผู้ใช้งานบนเว็บด้วย Grafana



ภาพที่ 3-9 โครงสร้างหน้าแสดงผล

จากภาพที่ 3-9 แสดงโครงสร้างแสดงค่าต่าง ๆ ที่วัดค่าได้ ซึ่งระบบสามารถทำการประมวลผลข้อมูลมาแสดงเป็นกราฟและสถิติต่าง ๆ เช่น ปริมาณการใช้กำลังไฟฟ้าทั้งหมด เป็นต้น ประกอบด้วยส่วนแสดงผลต่าง ๆ ดังนี้คือ แสดงการใช้งานกระแสไฟฟ้าตู้แร็คทั้ง 4 ตู้แบบ Real Time, แสดงกำลังไฟฟ้า PDU, แสดงอุณหภูมิตู้แร็ค, แสดงความชื้นตู้แร็ค, แสดงกำลังไฟฟ้า Input, แสดงความถี่ Input, แสดงการใช้กำลังไฟฟ้า และแสดงสถานะควันไฟ

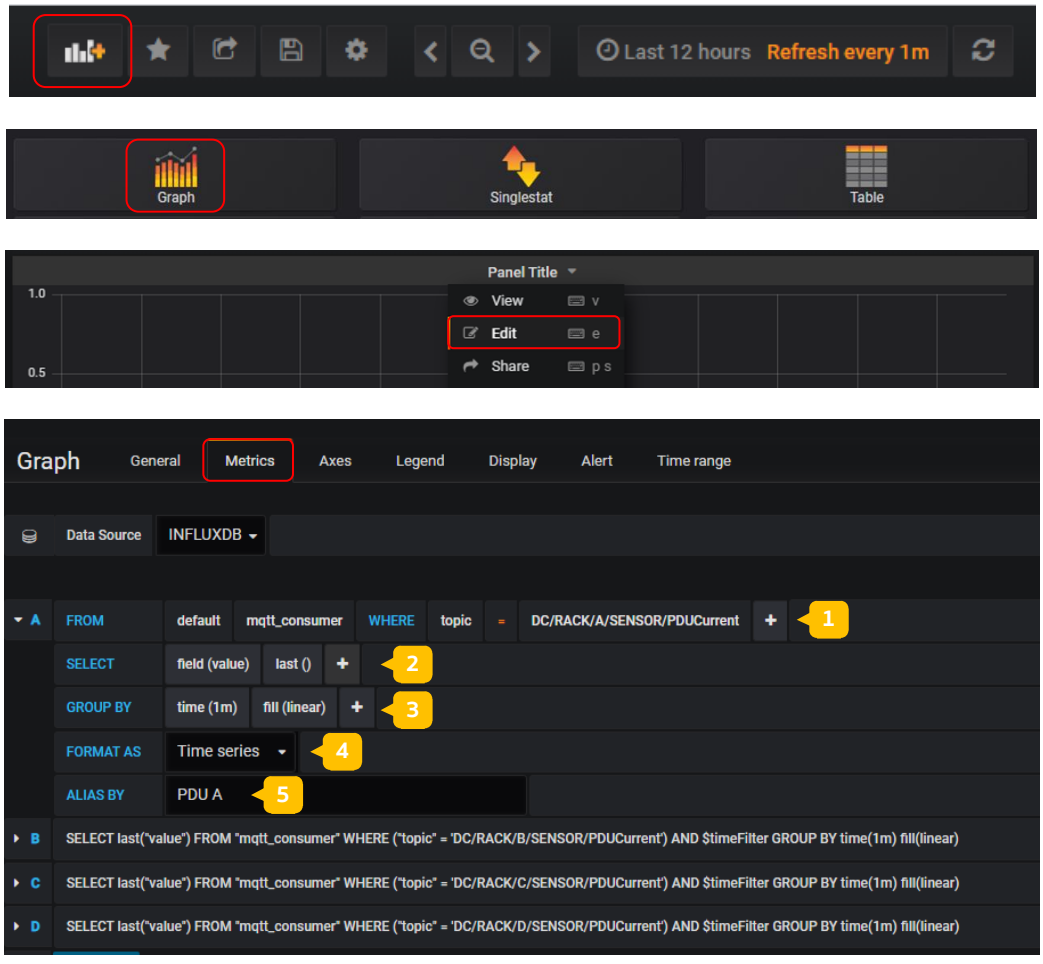
3.4.1 แสดงการใช้งานกระแสไฟฟ้าตู้แร็คทั้ง 4 ตู้แบบ Real Time



ภาพที่ 3-10 โครงสร้างหน้าแสดงผลแบบ Real Time

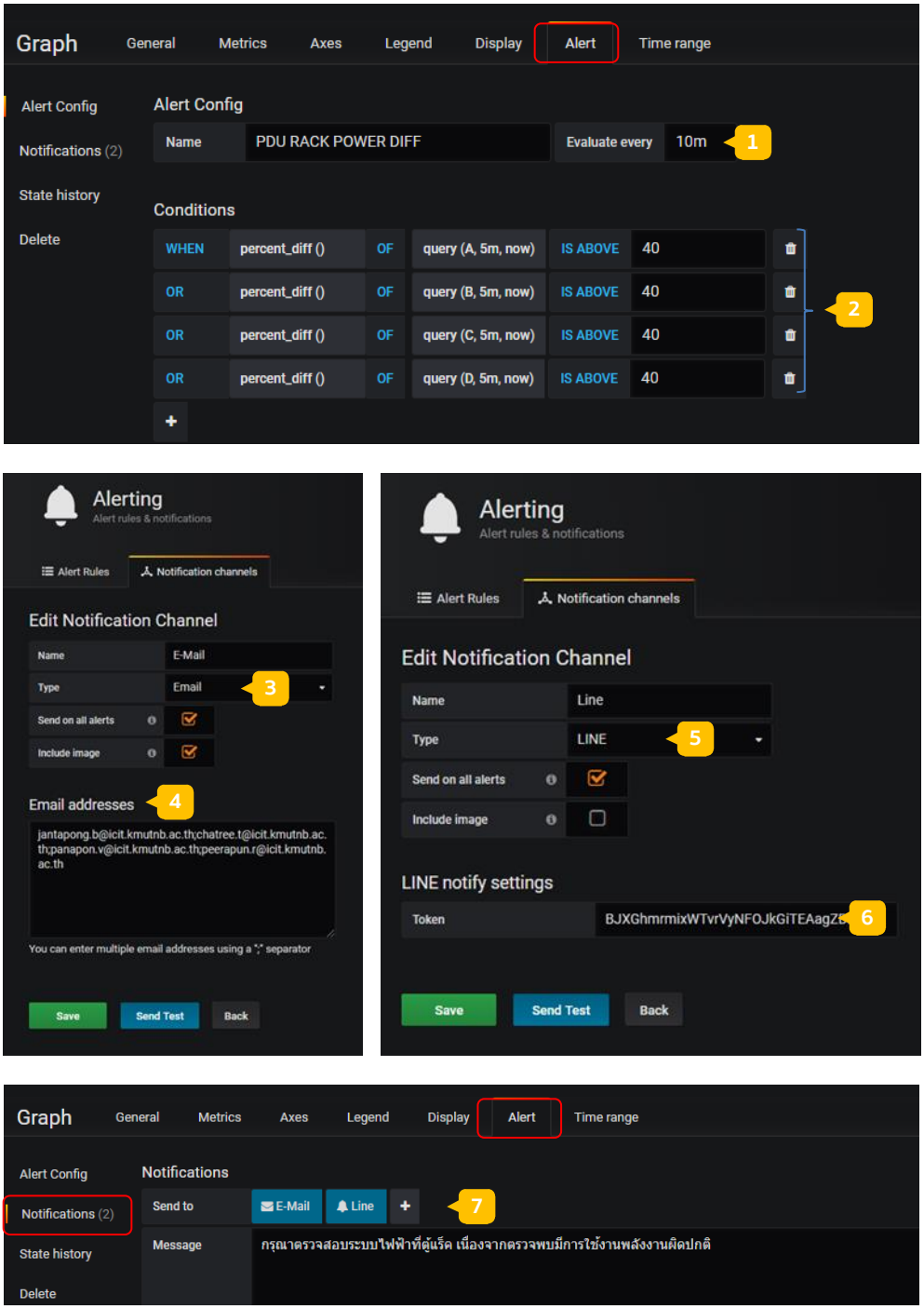
จากภาพที่ 3-10 แสดงหน้าเว็บสำหรับแสดงการใช้กระแสไฟฟ้าจาก PDU ของตู้แร็คทั้ง 4 ตู้ โดยจะแสดงผลแบบ Real Time โดยจะแสดงผลเป็นกราฟ สามารถดูความเปลี่ยนแปลงได้ หากมีการใช้งานกระแสไฟฟ้าที่มากหรือน้อยกว่าปกติ ก็จะมีแจ้งเตือนบนกราฟและ Line

ตัวอย่างการนำค่าที่วัดได้มาแสดงผลเป็นกราฟและการตั้งค่าการแจ้งเตือน สามารถทำได้โดยไปที่เมนู *Add panel* > *Graph* > *Edit* เพื่อกำหนดค่าการแสดงผล แสดงดังภาพที่ 3-11 และภาพที่ 3-12



ภาพที่ 3-11 การสร้างกราฟ

หมายเลข 1 เลือกแหล่งข้อมูล mqtt_consumer และเลือก topic ที่ต้องการ
หมายเลข 2 เลือกข้อมูลล่าสุดจากฟิลด์ value
หมายเลข 3 อ่านและจัดกลุ่มข้อมูลทุก ๆ 1 นาที ให้เป็นลักษณะเชิงเส้น
หมายเลข 4 กำหนดรูปแบบเป็นอนุกรมเวลา
หมายเลข 5 กำหนดชื่อให้กับเส้นกราฟ



ภาพที่ 3-12 การกำหนดค่าแจ้งเตือน

หมายเลข 1 กำหนดให้มีการตรวจสอบทุก ๆ 10 นาที

หมายเลข 2 กำหนดเงื่อนไข หากมีเปลี่ยนแปลงระหว่างข้อมูลเมื่อ 5 นาทีก่อนหน้ากับปัจจุบันแตกต่างกันมากกว่า 40 เปอร์เซ็นต์ ให้แจ้งเตือน

หมายเลข 3 Alerting > Notification Channels คลิ๊ก + New Channel กำหนด Type เป็น Email

หมายเลข 4 ป้อน Email ที่ต้องการส่งการแจ้งเตือน

หมายเลข 5 Alerting > Notification Channels คลิ๊ก + New Channel กำหนด Type เป็น Line

หมายเลข 6 ป้อน Token ของโปรแกรม Line

หมายเลข 7 ให้แจ้งเตือนผ่านอีเมล์และ Line

3.4.2 แสดงกำลังไฟฟ้า PDU

ภาพที่ 3-13 โครงสร้างหน้าแสดงกำลังไฟฟ้า PDU

จากภาพที่ 3-13 แสดงหน้าเว็บสำหรับแสดงกำลังไฟฟ้า PDU ที่มีการใช้งานจริงในปัจจุบัน โดยแสดงผลแบบ Real Time

3.4.3 แสดงอุณหภูมิตู้แร็ค

ภาพที่ 3-14 โครงสร้างหน้าแสดงอุณหภูมิตู้แร็ค

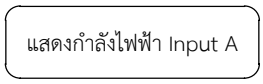
จากภาพที่ 3-14 แสดงหน้าเว็บสำหรับแสดงอุณหภูมิตู้แร็คในปัจจุบัน โดยแสดงผลแบบ Real Time

3.4.4 แสดงความชื้นตู้แร็ค

ภาพที่ 3-15 โครงสร้างหน้าแสดงความชื้นตู้แร็ค

จากภาพที่ 3-15 แสดงหน้าเว็บสำหรับแสดงความชื้นตู้แร็คในปัจจุบัน โดยแสดงผลแบบ Real Time

3.4.5 แสดงกำลังไฟฟ้า Input



ภาพที่ 3-16 โครงสร้างหน้าแสดงกำลังไฟฟ้า Input

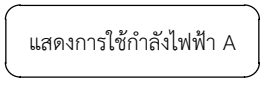
จากภาพที่ 3-16 แสดงหน้าเว็บสำหรับแสดงกำลังไฟฟ้า Input ในปัจจุบัน โดยแสดงผลแบบ Real Time

3.4.6 แสดงความถี่ Input

ภาพที่ 3-17 โครงสร้างหน้าแสดงความถี่ Input

จากภาพที่ 3-17 แสดงหน้าเว็บสำหรับแสดงความถี่ Input ในปัจจุบัน โดยแสดงผลแบบ Real Time

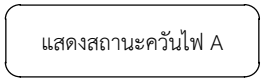
3.4.7 แสดงการใช้กำลังไฟฟ้า



ภาพที่ 3-18 โครงสร้างหน้าแสดงการใช้กำลังไฟฟ้า

จากภาพที่ 3-18 แสดงหน้าเว็บสำหรับแสดงการใช้กำลังไฟฟ้าในปัจจุบัน โดยแสดงผลแบบ Real Time

3.4.8 แสดงสถานะควันไฟ



ภาพที่ 3-19 โครงสร้างหน้าแสดงสถานะควันไฟ

จากภาพที่ 3-19 แสดงหน้าเว็บสำหรับแสดงสถานะควันไฟในปัจจุบัน โดยแสดงผลแบบ Real Time หากค่าเกินกว่าค่ามาตรฐาน ก็จะมีการแสดงแจ้งเตือน

บทที่ 4
ผลการดำเนินงาน

การดำเนินการโครงการวิจัย “ระบบเฝ้าระวังความปลอดภัยของเซิร์ฟเวอร์แร็ค” ระบบมีความสามารถในการทำงานตรงตามขอบเขตที่ได้กำหนดไว้

4.1 ผลการวัดประสิทธิภาพของอุปกรณ์

4.2 หน้าเว็บไซต์ระบบเฝ้าระวังความปลอดภัยของเซิร์ฟเวอร์แร็ค

4.3 การแจ้งเตือนความผิดปกติผ่านโปรแกรม Line และอีเมล์

4.4 ผลการประเมินประสิทธิภาพและความพึงพอใจจากผู้เชี่ยวชาญ

4.1 ผลการวัดประสิทธิภาพของอุปกรณ์

อุปกรณ์ที่จัดทำขึ้นในงานวิจัยแสดงดังภาพที่ 4-1 และการทดสอบวัดประสิทธิภาพของอุปกรณ์นั้นดำเนินการโดยนำอุปกรณ์มาตรฐานจำนวน 1 เครื่องซึ่งวัดค่าออกมาเป็นแบบอนาล็อกทำวัดค่าและเปรียบเทียบกับอุปกรณ์ในงานวิจัย ดังแสดงในตารางที่ 4-1

ภาพที่ 4-1 อุปกรณ์ที่ติดตั้งสำหรับวัดค่าอุณหภูมิ ความชื้น และควันไฟ

ตารางที่ 4-1 ประสิทธิภาพของอุปกรณ์วัดอุณหภูมิและความชื้น

ลำดับ	เวลา	อุปกรณ์มาตรฐาน		อุปกรณ์งานวิจัย		ความแม่นยำ (%)	
		อุณหภูมิ (*C)	ความชื้น (%RH)	อุณหภูมิ (*C)	ความชื้น (%RH)	อุณหภูมิ (*C)	ความชื้น (%RH)
1	10.42	23.00	54.50	23.70	54.40	96.96	99.82
2	11.00	23.00	54.00	23.80	53.50	96.52	99.07
3	11.29	23.00	54.00	23.90	53.40	96.09	98.89
4	11.50	23.00	54.00	23.60	53.10	97.39	98.33
5	13.03	23.00	56.50	23.90	57.30	96.09	98.58
					เฉลี่ย	96.61	98.94

จากตารางที่ 4-1 ผลการวัดประสิทธิภาพของอุปกรณ์ที่ดำเนินการในงานวิจัยเปรียบเทียบกับอุปกรณ์มาตรฐานมีความแม่นยำของการวัดอุณหภูมิอยู่ที่ 96.61% และความแม่นยำในการวัดความชื้นอยู่ที่ 98.94% ซึ่งถือว่ามีความแม่นยำที่สูงและยอมรับได้สำหรับการใช้งานในเซิร์ฟเวอร์แร็ค

ตารางที่ 4-2 ประสิทธิภาพของอุปกรณ์ตรวจจับควันไฟ

ลำดับ	อุปกรณ์มาตรฐาน	อุปกรณ์งานวิจัย	ความแม่นยำ (%)
	การตรวจจับควันไฟ (> 300 ppm)	การตรวจจับควันไฟ (> 300 ppm)	การตรวจจับควันไฟ (> 300 ppm)
1	ตรวจจับได้	ตรวจจับได้	100
2	ตรวจจับได้	ตรวจจับได้	100
3	ตรวจจับได้	ตรวจจับได้	100
4	ตรวจจับได้	ตรวจจับได้	100
5	ตรวจจับได้	ตรวจจับได้	100
		เฉลี่ย	100

จากตารางที่ 4-2 ผลการวัดประสิทธิภาพของอุปกรณ์ตรวจจับควันไฟในงานวิจัยเปรียบเทียบกับอุปกรณ์มาตรฐาน โดยวัดค่ามากกว่า 300 ppm มีความแม่นยำของการตรวจจับอยู่ที่ 100% ซึ่งถือว่ามีความแม่นยำที่สูง

4.2 หน้าเว็บไซต์ระบบเฝ้าระวังความปลอดภัยของเซิร์ฟเวอร์แร็ค

4.2.1 หน้า Login

ภาพที่ 4-2 หน้า Login

จากภาพที่ 4-2 แสดงหน้า Login สำหรับผู้ใช้งานระบบ โดยสามารถเข้าระบบด้วยชื่อ URL http://dc-mon.kmutnb.ac.th:3000 พร้อมกรอกชื่อผู้ใช้และรหัสผ่านเพื่อทำการพิสูจน์ทราบตัวตนก่อนเข้าใช้งานระบบ

4.2.2 หน้า Home

ภาพที่ 4-3 หน้า Home

จากภาพที่ 4-3 มีส่วนประกอบต่าง ๆ ดังนี้

- หมายเลขที่ 1 เป็นส่วนแสดงสถานะการใช้งานกระแสไฟฟ้า ณ ปัจจุบันของ PDU ทั้ง 4 เชิร์ฟเวอร์แร็คแบบ Real Time
- หมายเลขที่ 2 ส่วนแสดงการใช้งาน เปอร์เซ็นต์การใช้งาน (Percent Current Utilization), ความถี่ (Line Frequency), โหลดกำลังไฟฟ้า (Total Input Power), การใช้พลังงานไฟฟ้าสะสม (Accumulated Energy), การใช้งานกระแสไฟฟ้า (Neutral Current Measurement), อุณหภูมิ, ความชื้น และสถานะตรวจสอบควันไฟ (Smoke Detect) ของเซิร์ฟเวอร์แร็ค A

- หมายเลขที่ 3 ส่วนแสดงการใช้งาน เปอร์เซ็นต์การใช้งาน (Percent Current Utilization), ความถี่ (Line Frequency), โหลดกำลังไฟฟ้า (Total Input Power), การใช้พลังงานไฟฟ้าสะสม (Accumulated Energy), การใช้งานกระแสไฟฟ้า (Neutral Current Measurement), อุณหภูมิ, ความชื้น และสถานะตรวจสอบควันไฟ (Smoke Detect) ของเซิร์ฟเวอร์แร็ค B
- หมายเลขที่ 4 ส่วนแสดงการใช้งาน เปอร์เซ็นต์การใช้งาน (Percent Current Utilization), ความถี่ (Line Frequency), โหลดกำลังไฟฟ้า (Total Input Power), การใช้พลังงานไฟฟ้าสะสม (Accumulated Energy) และการใช้งานกระแสไฟฟ้า (Neutral Current Measurement) ของเซิร์ฟเวอร์แร็ค C
- หมายเลขที่ 5 ส่วนแสดงการใช้งาน เปอร์เซ็นต์การใช้งาน(Percent Current Utilization), ความถี่ (Line Frequency), โหลดกำลังไฟฟ้า (Total Input Power), การใช้พลังงานไฟฟ้าสะสม (Accumulated Energy) และการใช้งานกระแสไฟฟ้า(Neutral Current Measurement) ของเซิร์ฟเวอร์แร็ค D

4.2.3 แสดงผลบนอุปกรณ์พกพา

ภาพที่ 4-4 หน้าจอเว็บไซต์บนอุปกรณ์พกพาในแนวตั้ง (iPhone X)

ภาพที่ 4-5 หน้าจอเว็บไซต์บนอุปกรณ์พกพาในแนวนอน (iPhone X)

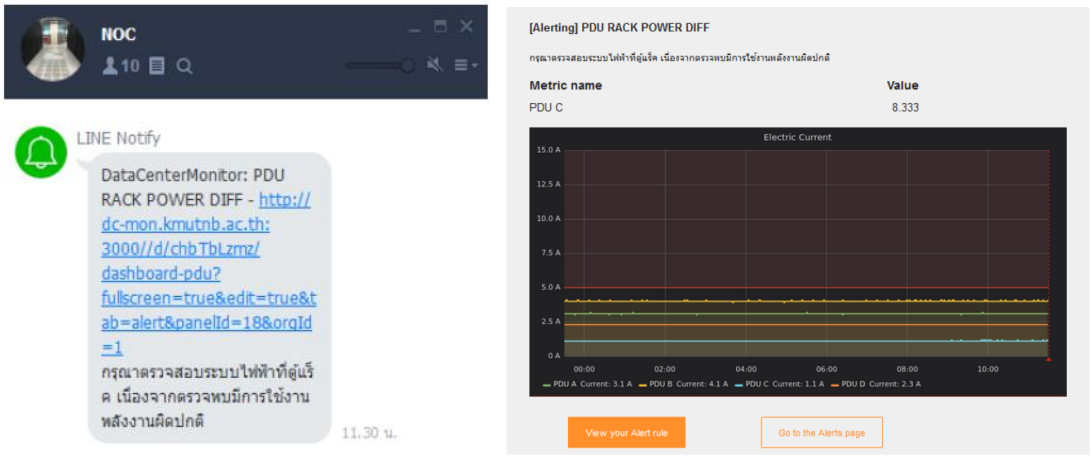
จากภาพที่ 4-4, 4-5 แสดงการทดสอบเข้าใช้งานเว็บไซต์ผ่านอุปกรณ์พกพา iPhone X ทั้งแนวตั้งและแนวนอน โดยสามารถแสดงผลเนื้อหาในลักษณะ Reponsive ได้เป็นอย่างดี พร้อมแสดงผลแบบ Real Time

4.2.4 การดูข้อมูลย้อนหลัง

ภาพที่ 4-6 หน้าจอเว็บไซต์เลือกดูข้อมูลย้อนหลัง 90 วัน

4.3 การแจ้งเตือนความผิดปกติผ่านโปรแกรม Line และอีเมล์

ระบบสามารถแจ้งเตือนความผิดปกติของการใช้งานผ่านโปรแกรม Line และอีเมล์ได้ โดยหากค่าที่วัดได้มีค่าที่ผิดจากที่ตั้งไว้(มากหรือน้อยกว่าที่ใช้งานปกติ) ระบบก็จะทำการแจ้งเตือนทันที และเมื่อการใช้งานกลับสู่สภาวะปกติระบบจะดำเนินการแจ้งเช่นกัน แสดงการแจ้งเตือนดังภาพที่ 4-6



ภาพที่ 4-7 การแจ้งเตือนผ่านโปรแกรม Line และอีเมล์

4.4 ผลการประเมินประสิทธิภาพและความพึงพอใจจากผู้เชี่ยวชาญ

ตารางที่ 4-3 ผลการประเมินประสิทธิภาพและความพึงพอใจ

รายการ	ระดับความพึงพอใจ		
	ค่าเฉลี่ย	ส่วนเบี่ยงเบนมาตรฐาน	แปลผล
1. ด้านประสิทธิภาพของระบบ			
1.1 ความถูกต้อง แม่นยำของระบบ	4.20	0.45	มาก
1.2 ตรงตามวัตถุประสงค์ที่ต้องการ	4.20	0.45	มาก
1.3 การออกแบบให้ใช้งานง่าย เมนูไม่ซับซ้อน	4.00	0.00	มาก
2. ด้านความสะดวก สวยงาม			
2.1 ความสะดวกในการใช้งานระบบ	4.20	0.45	มาก
2.2 ความเหมาะสมในการใช้งานระบบ	4.20	0.45	มาก
3. ด้านคุณภาพของระบบ			
3.1 ความพึงพอใจในการใช้งานระบบ	4.40	0.55	มาก
3.2 ความสามารถของระบบในการนำไปใช้ประโยชน์	4.20	0.45	มาก
สรุปโดยรวม	**4.20**	**0.40**	**มาก**

ผลการประเมินระบบจากผู้เชี่ยวชาญ ด้านความถูกต้องแม่นยำของระบบ, ตรงตามวัตถุประสงค์ที่ต้องการ และการออกแบบให้ใช้งานง่าย เมนูไม่ซับซ้อนอยู่ในระดับมาก แสดงให้เห็นว่าระบบที่พัฒนาขึ้นมีประสิทธิภาพในระดับมาก ในด้านความสะดวกในการใช้งานระบบ และความเหมาะสมในการใช้งานระบบอยู่ในระดับมาก แสดงให้เห็นว่าระบบที่พัฒนาขึ้นมีสะดวกต่อการใช้งานในระดับมาก ในด้านความพึงพอใจในการใช้งานระบบและความสามารถของระบบในการนำไปใช้ประโยชน์อยู่ในระดับมาก แสดงให้เห็นว่าระบบที่พัฒนาขึ้นสามารถนำไปใช้ประโยชน์ได้จริง

บทที่ 5
สรุปผลการดำเนินงาน

การจัดทำวิจัยระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็ค สามารถสรุปการดำเนินงาน ปัญหาและอุปสรรคต่าง ๆ ที่เกิดขึ้นระหว่างการทำวิจัย รวมถึงแนวทางแก้ไขปัญหาและข้อเสนอแนะ เพื่อที่จะได้นำวิจัยนี้พัฒนาต่อไปในอนาคต

5.1 สรุป

5.2 ปัญหาและแนวทางแก้ไขปัญหา

5.3 ข้อเสนอแนะและแนวทางพัฒนา

5.1 สรุป

ในการจัดทำวิจัยระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็ค เป็นวิจัยที่ทำเกี่ยวกับการตรวจวัดอุณหภูมิ ความชื้น ตรวจจับควันไฟ และอ่านค่าการใช้งานกระแสไฟฟ้าจาก PDU หลังจากนั้นส่งข้อมูลเข้าไปที่ MQTT Server และใช้ Telegraf เพื่อทำการ Subscribe ข้อมูลจาก MQTT Server แล้วส่งข้อมูลไปเก็บยังฐานข้อมูล InfluxDB และแสดงผลในรูปแบบตัวเลขและกราฟด้วย Grafana ทั้งนี้ระบบสามารถตรวจจับความผิดปกติของการใช้งาน โดยหากค่าที่วัดได้มากหรือน้อยกว่าค่าที่กำหนดไว้ จะมีการแจ้งเตือนผ่านโปรแกรม Line และอีเมล์ทันที เพื่อการแก้ไขปัญหาได้อย่างทันถ่วงทีและหากปัญหาถูกแก้ไขแล้วระบบจะดำเนินการแจ้งสถานะปกติให้ทราบอีกครั้ง อีกทั้งในงานวิจัยนี้ได้ทำการวัดความแม่นยำของอุปกรณ์ที่จัดทำขึ้นเปรียบเทียบกับอุปกรณ์มาตรฐาน ซึ่งพบว่ามีความแม่นยำในการวัดอุณหภูมิและความชื้นอยู่ในระดับมากกว่า 96% และการตรวจจับควันไฟอยู่ในระดับ 100% รวมถึงประเมินประสิทธิภาพของระบบจากผู้เชี่ยวชาญจำนวน 5 ท่าน ซึ่งพบว่าระบบมีประสิทธิภาพอยู่ในระดับมาก

โดยระบบระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็คนี้สามารถทำงานได้ครบถ้วนตามวัตถุประสงค์และขอบเขตของการวิจัย

5.2 ปัญหาและแนวทางแก้ไขปัญหา

5.2.1 หากมีการต่อเซ็นเซอร์หลายตัวและใช้งานกระแสไฟฟ้าจากตัวบอร์ด ESP8266 จะทำให้ไม่สามารถจ่ายกระแสไฟฟ้าได้เพียงพอ ทำให้ไม่สามารถอัพโหลดโปรแกรมได้และอุปกรณ์ทำงานผิดพลาดได้

แนวทางแก้ไข หาแหล่งจ่ายไฟภายนอกเพิ่ม เพื่อต่อใช้งานกับเซ็นต์เซอร์แทนการใช้กระแสไฟฟ้าจากตัวบอร์ด ESP8266

5.2.2 การเขียนโปรแกรมเพื่อติดต่อ PDU เนื่องจาก OID ที่ใช้มีทั้งค่าเฉพาะอุปกรณ์และค่าทั่วไป ทำให้ไม่สามารถดึงข้อมูลบางอย่างได้

แนวทางแก้ไข นำไฟล์ MIB ของ PDU และใช้โปรแกรม MIB Browser เพื่อลองอ่านค่าข้อมูลจาก PDU ให้ทราบว่า OID นั้นคือค่าอะไร หลังจากนั้นโหลด MIB ไปใช้ในการเขียนโปรแกรมภาษา PHP ต่อไป

5.3 ข้อเสนอแนะและแนวทางพัฒนา

5.3.1 ระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็ค ใช้ Grafana เป็นพื้นฐาน ซึ่งมี API สามารถดึงข้อมูลไปใช้งานได้สะดวก ดังนั้นพัฒนาเป็น Mobile Application โดยดึงข้อมูลผ่าน Grafana ช่วยอำนวยสะดวกต่อการใช้งานได้มากขึ้น

5.3.2 ระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็ค แลกเปลี่ยนข้อมูลผ่าน MQTT Server ดังนั้นสามารถเพิ่มเซ็นเซอร์ต่าง ๆ ได้ง่าย เช่น เซ็นเซอร์ตรวจจับการเปิดเซิร์ฟเวอร์แร็ค, เซ็นเซอร์ตรวจจับการเคลื่อนไหว เป็นต้น เพื่อแจ้งเตือนหากมีบุคคลเข้าใกล้เซิร์ฟเวอร์แร็ค

บรรณานุกรม

ไกรสร สืบบุญ. (2562). *#102 Arduino MQ-2 PPM สอนใช้งาน Arduino sensor MQ2 เซนเซอร์ MQ-2 ตรวจจับแก๊สมีเทน LPG, Smoke, CO*. ค้นจาก https://www.arduinoall.com/article/345/102-arduino-mq-2-ppm-สอนใช้งาน-arduino-sensor-mq2-เซนเซอร์ -mq-2-ตรวจจับแก๊สมีเทน -lpg-smoke-co

ไกรสร สืบบุญ. (2562). #61 Arduino สอนใช้งาน Arduino OLED แบบ I2C ไดรเวอร์ SSD1306. ค้นจาก https://www.arduinoall.com/article/315/61-arduino-สอนการใช้-arduino-oled-แบบ-i2c-ไดร์เวอร์-ssd1306

ทัศนะ ถมทอง, สุรสิทธิ์ แสนทอน, วิโรจน์ อยู่เย็นดี และศุภชัย ญาณชาญปรีชา. (2547). การออกแบบสร้างตัวควบคุมอุณหภูมิและความชื้นแบบฟัซซีลอจิกสำหรับตู้ฟักไข่ขนาดเล็ก. *การประชุมทางวิชาการของมหาวิทยาลัยเกษตรศาสตร์ ครั้งที่ 42* (หน้า 247-254), กรุงเทพฯ : มหาวิทยาลัยเกษตรศาสตร์,.

ธวัชชัย บัลลังก์รัตน. (2549). *การพัฒนาโปรแกรมเฝ้ามองและแจ้งเตือนเครือข่ายในเวลาจริง*. (ปัญหาพิเศษปริญญาวิทยาศาสตร์มหาบัณฑิต). สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, กรุงเทพฯ.

ประทีป พึ่งวัฒนาพงศ์. (2556). *การประยุกต์ใช้ไอทิลในการแก้ไขปัญหาของลูกค้าที่ใช้บริการอินเทอร์เน็ตของผู้ให้บริการอินเทอร์เน็ตแห่งหนึ่งในประเทศไทย*. (วิทยานิพนธ์ปริญญาวิทยาศาสตร์มหาบัณฑิต). มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี, กรุงเทพฯ.

ปัญญา กิตติพิพัฒนถาวร. (2556). *การตรวจสอบสภาพพร้อมใช้งานของระบบแบบซอฟท์เรียลไทม์โดยวิธีการวิเคราะห์ล็อกแบบกระจาย*. (วิทยานิพนธ์ปริญญาวิศวกรรมศาสตร์มหาบัณฑิตย์). จุฬาลงกรณ์มหาวิทยาลัย, กรุงเทพฯ.

เอกลักษณ์ สุมนพันธุ์ และธานินท์ สุเชียง. (2557). *แอพพลิเคชันแอนดรอยด์ร่วมกับระบบสมองกลฝังตัวเพื่อควบคุมระบบไฟฟ้าและอุณหภูมิภายในโรงเรือน*. (วิทยานิพนธ์ปริญญาวิศวกรรมศาสตร์บัณฑิต). มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา, เชียงใหม่.

Hughes, J. (2009). Skimp on Server Room Air Conditioning? At Your Peril. Retrieved from https://www.openxtra.co.uk/articles/skimp-server-room-ac

Last Minute Engineers. (2019). Interface OLED Graphic Display Module with ESP32. Retrieved form https://lastminuteengineers.com/oled-display-esp32-tutorial

Miller, J. (2009). The Importance of cabinet-Level power monitoring in a data-center. Retrieved from http://www.bicsi.org/pdf/winter_2010/Jeff_Miller.pdf

ภาคผนวก ก
รายนามผู้เชี่ยวชาญประเมินระบบ

รายนามผู้เชี่ยวชาญที่ประเมินระบบ

1. **ชื่อ** นายพีรธัช หนูชู
   ตำแหน่ง วิศวกร
   สถานที่ปฏิบัติงาน ฝ่ายวิศวกรรมระบบเครือข่าย
   สำนักคอมพิวเตอร์และเทคโนโลยีสารสนเทศ
   ประสบการณ์ทำงาน คอมพิวเตอร์เซิร์ฟเวอร์, Data Center และเครือข่าย 11 ปี

2. **ชื่อ** นายยืนยง ลีจรรยาวัตร์
   ตำแหน่ง System Engineer
   สถานที่ปฏิบัติงาน บริษัท Advance Communication Service Co.,Ltd
   ประสบการณ์ทำงาน Network Operation Center 2 ปี
   System Engineer 2 ปี
   Network Engineer 2 ปี

3. **ชื่อ** นายจีระพล คุ่มเคี่ยม
   ตำแหน่ง นักวิชาการคอมพิวเตอร์ชำนาญการ
   สถานที่ปฏิบัติงาน สำนักหอสมุดกลาง
   มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
   ประสบการณ์ทำงาน คอมพิวเตอร์เซิร์ฟเวอร์, Data Center, ระบบห้องสมุดอัตโนมัติ, IoT, RFID และเครือข่าย 27 ปี

4. **ชื่อ** นายธนพัฒน์ อนุตรกุล
   ตำแหน่ง System Engineer
   สถานที่ปฏิบัติงาน บริษัท เน็ตโปรเทคชั่นคอนเซ็ปต์ อินเตอร์เนชั่นแนล จำกัด
   ประสบการณ์ทำงาน 20 ปี ใน 4 บริษัท
   Computer Technician 1 ปี
   System Engineer 4 ปี
   Network Engineer 10 ปี
   IT Security Consult 6 ปี

5. **ชื่อ** นายอมรชัย ภูหิรัญ
   ตำแหน่ง นักวิชาการคอมพิวเตอร์
   สถานที่ปฏิบัติงาน บัณฑิตวิทยาลัยวิศวกรรมศาสตร์นานาชาติสิรินธร ไทย-เยอรมัน (TGGS)
   มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
   ประสบการณ์ทำงาน IT Support 2 ปี
   System Admin 2 ปี
   Network Engineer 11 ปี

ภาคผนวก ข
แบบสอบถาม

แบบสอบถาม
ประสิทธิภาพและความพึงพอใช้ต่อการใช้งาน
ระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็ค

คำชี้แจง แบบสอบถามชุดนี้จัดทำขึ้นเพื่อประเมินประสิทธิภาพและความพึงพอใช้ต่อการใช้งาน "ระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็ค" ซึ่งเป็นงานวิจัยทุนของสำนักคอมพิวเตอร์และเทคโนโลยีสารสนเทศ ปีงบประมาณ 2560 ทั้งนี้เพื่อการพัฒนาปรับปรุงประสิทธิภาพของระบบในอนาคต กรุณากรอกแบบสอบถามตามความจริง ขอขอบคุณทุกท่านที่ให้ความร่วมมือไว้ ณ โอกาสนี้ด้วย

รายการ	ระดับความคิดเห็น				
	น้อยที่สุด	น้อย	ปานกลาง	มาก	มากที่สุด
1.ด้านประสิทธิภาพของระบบ					
1.1 ความถูกต้อง แม่นยำของระบบ					
1.2 ตรงตามวัตถุประสงค์ที่ต้องการ					
1.3 การออกแบบให้ใช้งานง่าย เมนูไม่ซับซ้อน					
2. ด้านความสะดวก สวยงาม					
2.1 ความสะดวกในการใช้งานระบบ					
2.2 ความเหมาะสมในการใช้งานระบบ					
3. ด้านคุณภาพของระบบ					
3.1 ความพึงพอใจในการใช้งานระบบ					
3.2 ความสามารถของระบบในการนำไปใช้ประโยชน์					

4. อื่นๆ

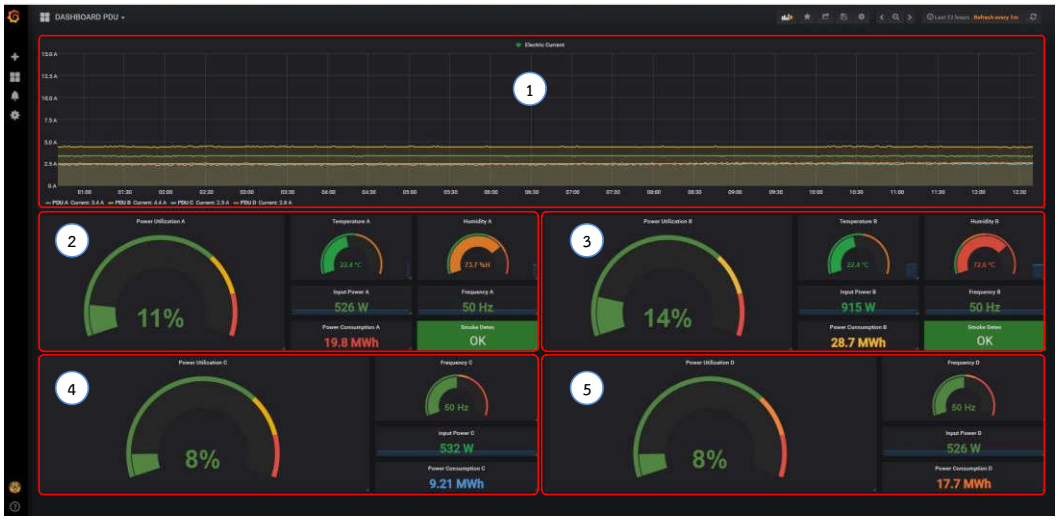
.....
.....
.....
.....

ภาคผนวก ค
คู่มือการใช้งานระบบ

คู่มือการใช้งานระบบ

1. เข้าสู่ระบบผ่านเว็บไซต์ http://dc-mon.kmutnb.ac.th:3000 หลังจากนั้นป้อนชื่อผู้ใช้และรหัสผ่าน เพื่อยืนยันตัวต้นก่อนใช้งานระบบ แสดงดังภาพ

2. เมื่อเข้าสู่ระบบเรียบร้อยแล้ว จะปรากฏหน้าจอแสดงค่าต่าง ๆ ดังนี้

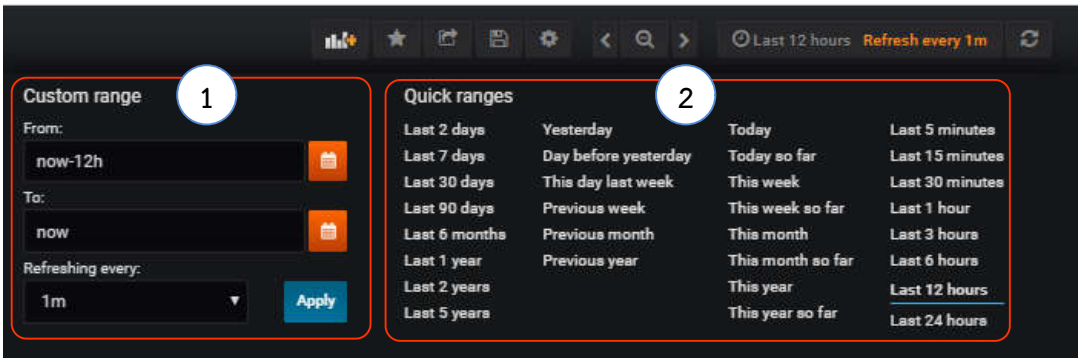


จากภาพมีส่วนประกอบต่าง ๆ ดังนี้

- *หมายเลขที่ 1* เป็นส่วนแสดงสถานะการใช้งานกระแสไฟฟ้า ณ ปัจจุบันของ PDU ทั้ง 4 เซิร์ฟเวอร์แร็คแบบ Real Time
- *หมายเลขที่ 2* ส่วนแสดงการใช้งาน เปอร์เซ็นต์การใช้งาน(Percent Current Utilization), ความถี่ (Line Frequency), โหลดกำลังไฟฟ้า (Total Input Power), การใช้พลังงานไฟฟ้าสะสม (Accumulated Energy), การใช้งานกระแสไฟฟ้า(Neutral Current Measurement), อุณหภูมิ, ความชื้น และสถานะการตรวจสอบควันไฟ (Smoke Detect) ของเซิร์ฟเวอร์แร็ค

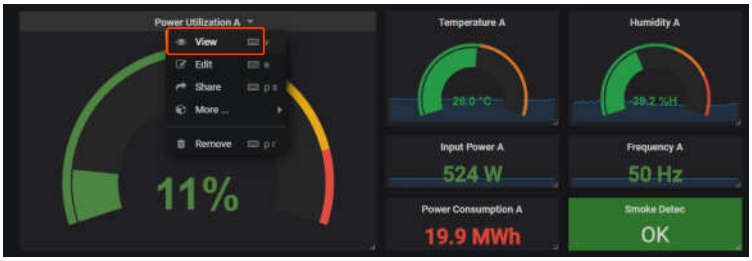
- *หมายเลขที่ 3* ส่วนแสดงการใช้งาน เปอร์เซ็นต์การใช้งาน(Percent Current Utilization), ความถี่ (Line Frequency), โหลดกำลังไฟฟ้า (Total Input Power), การใช้พลังงานไฟฟ้าสะสม (Accumulated Energy), การใช้งานกระแสไฟฟ้า(Neutral Current Measurement), อุณหภูมิ, ความชื้น และสถานะการตรวจสอบควันไฟ (Smoke Detect) ของเซิร์ฟเวอร์แร็ค B
- *หมายเลขที่ 4* ส่วนแสดงการใช้งาน เปอร์เซ็นต์การใช้งาน(Percent Current Utilization), ความถี่ (Line Frequency), โหลดกำลังไฟฟ้า (Total Input Power), การใช้พลังงานไฟฟ้าสะสม (Accumulated Energy) และการใช้งานกระแสไฟฟ้า (Neutral Current Measurement) ของเซิร์ฟเวอร์แร็ค C
- *หมายเลขที่ 5* ส่วนแสดงการใช้งาน เปอร์เซ็นต์การใช้งาน(Percent Current Utilization), ความถี่ (Line Frequency), โหลดกำลังไฟฟ้า (Total Input Power), การใช้พลังงานไฟฟ้าสะสม (Accumulated Energy) และการใช้งานกระแสไฟฟ้า (Neutral Current Measurement) ของเซิร์ฟเวอร์แร็ค C

3. การเลือกการแสดงผลข้อมูลเป็นช่วงเวลา เพื่อดูข้อมูลย้อนหลัง แสดงดังภาพ

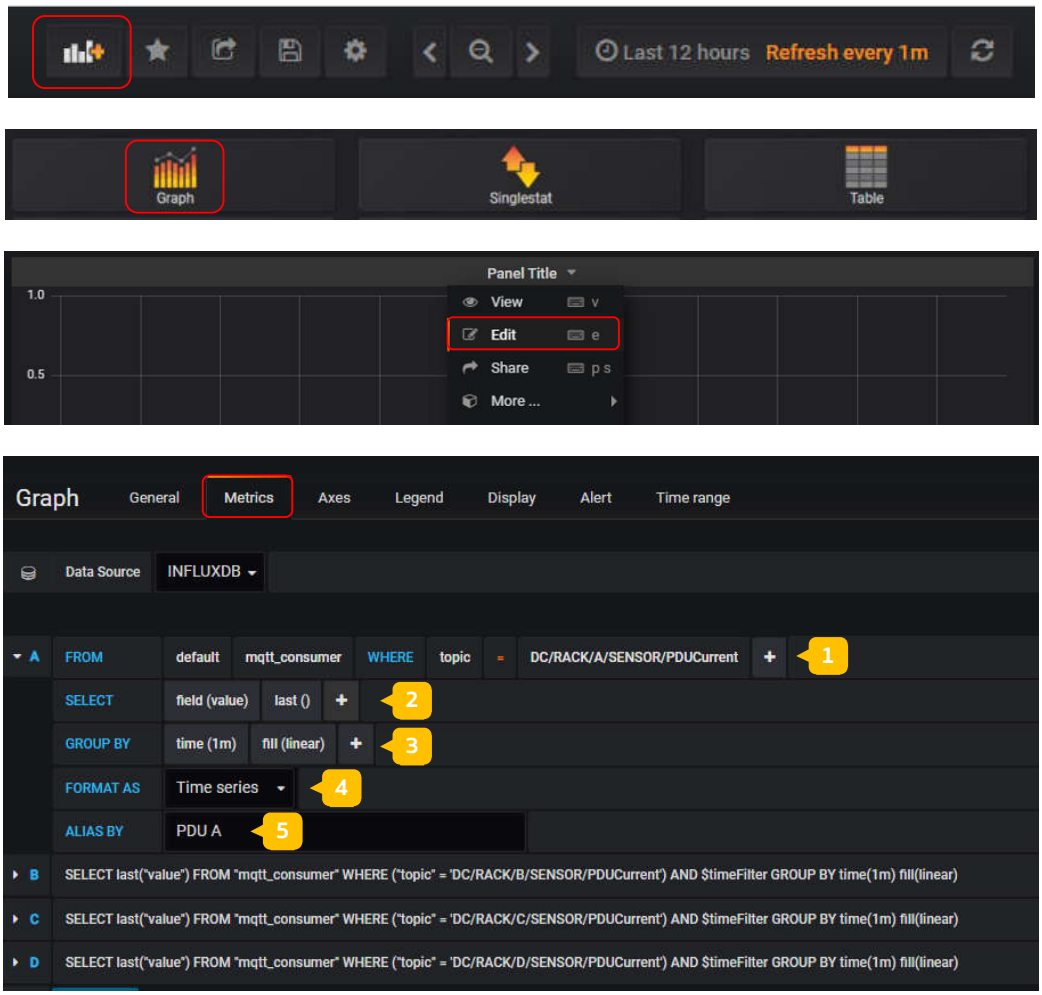


- หมายเลขที่ 1 สามารถเลือกช่วงเวลาในการแสดงผลข้อมูลได้เอง โดยเลือกเวลาเริ่มต้น From และสิ้นสุด เพื่อแสดงข้อมูลที่ต้องการ
- หมายเลขที่ 2 Quick ranges เป็นช่วงเวลาที่กำหนดไว้แล้ว โดยสามารถเลือกใช้งานได้เลย เช่น last 5 minutes คือแสดงข้อมูล 5 นาทีล่าสุด เป็นต้น

4. สามารถซูมเข้าไปดูข้อมูลได้ โดยเลือกที่กราฟหรือข้อมูลที่ต้องการ แล้วเลือกเมนู *View* แสดงดังภาพ

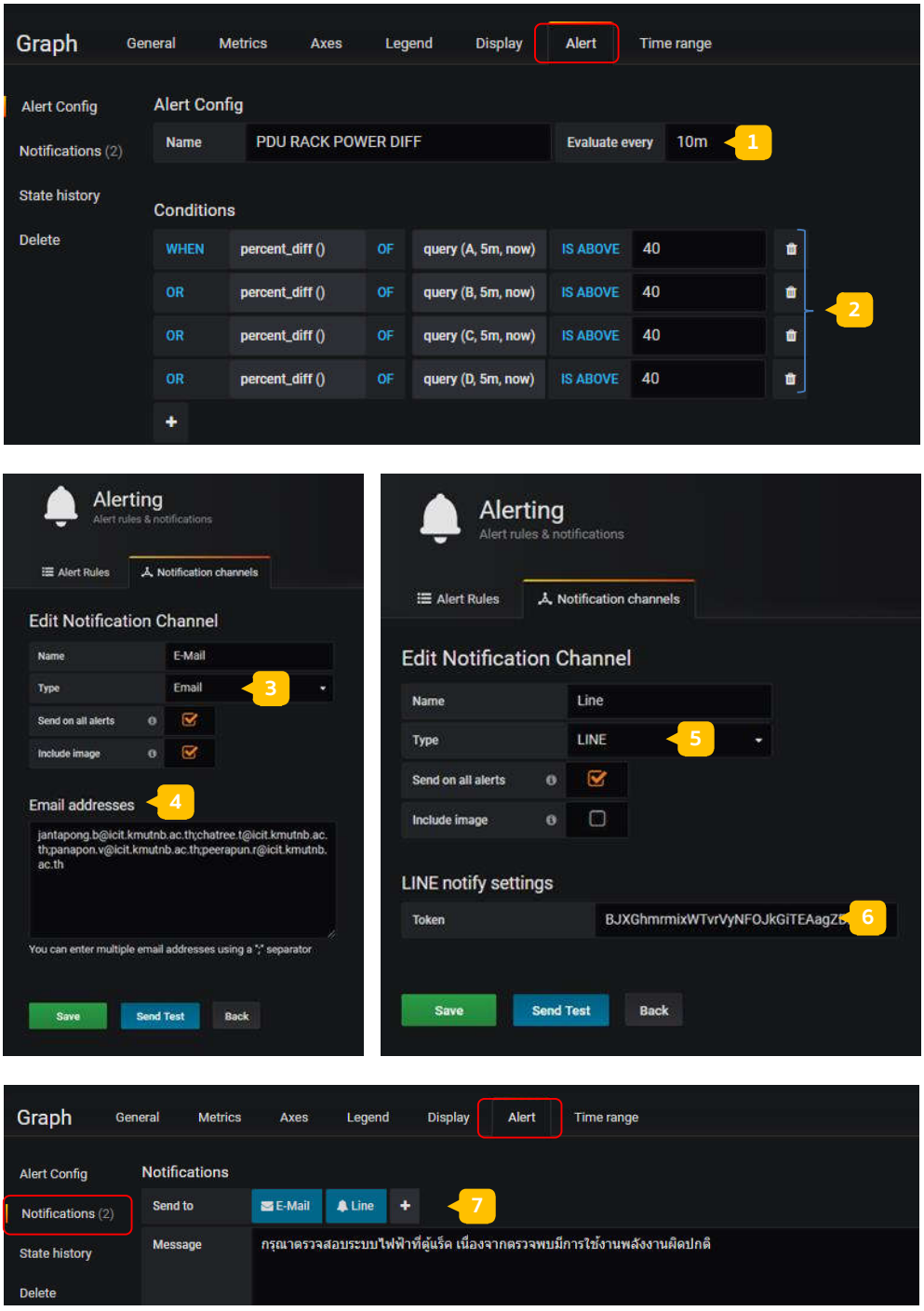


5. การสร้างกราฟและกำหนดค่าแจ้งเตือน สามารถทำได้โดยไปที่เมนู *Add panel* > *Graph* > *Edit* เพื่อกำหนดค่าการแสดงผล แสดงดังภาพที่ 3-11 และภาพที่ 3-12



ภาพที่ 3-11 การสร้างกราฟ

หมายเลข 1 เลือกแหล่งข้อมูล mqtt_consumer และเลือก topic ที่ต้องการ
หมายเลข 2 เลือกข้อมูลล่าสุดจากฟิลด์ value
หมายเลข 3 อ่านและจัดกลุ่มข้อมูลทุก ๆ 1 นาที ให้เป็นลักษณะเชิงเส้น
หมายเลข 4 กำหนดรูปแบบเป็นอนุกรมเวลา
หมายเลข 5 กำหนดชื่อให้กับเส้นกราฟ



ภาพที่ 3-12 การกำหนดค่าแจ้งเตือน

หมายเลข 1 กำหนดให้มีการตรวจสอบทุก ๆ 10 นาที

หมายเลข 2 กำหนดเงื่อนไข หากมีเปลี่ยนแปลงระหว่างข้อมูลเมื่อ 5 นาทีก่อนหน้ากับปัจจุบันแตกต่างกันมากกว่า 40 เปอร์เซ็นต์ ให้แจ้งเตือน

หมายเลข 3 Alerting > Notification Channels คลิ๊ก + New Channel กำหนด Type เป็น Email

หมายเลข 4 ป้อน Email ที่ต้องการส่งการแจ้งเตือน

หมายเลข 5 Alerting > Notification Channels คลิ๊ก + New Channel กำหนด Type เป็น Line

หมายเลข 6 ป้อน Token ของโปรแกรม Line

หมายเลข 7 ให้แจ้งเตือนผ่านอีเมล์และ Line

ประวัติผู้วิจัย

ชื่อ : นายจันตพงษ์ บุตรลักษณ์
ชื่องานวิจัย : การพัฒนาระบบเฝ้าระวังสภาพแวดล้อมของเซิร์ฟเวอร์แร็ค

สถานที่ทำงาน : สำนักคอมพิวเตอร์และเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอม-เกล้าพระนครเหนือ
ตำแหน่ง : นักวิชาการคอมพิวเตอร์
วุฒิการศึกษา : วิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ความชำนาญ/ประสบการณ์ : ระบบแจ้งเตือนการบุกรุก(IDS), ระบบเว็บเซิร์ฟเวอร์และระบบเซิร์ฟเวอร์เสมือน (Virtualization Server), คอมพิวเตอร์เน็ตเวิร์ค, ความมั่นคงปลอดภัยระบบคอมพิวเตอร์