

**กล้องถ่ายภาพตรวจตราโดยใช้โมดูลอัจฉริยะและ AMAZON WEB
SERVICES
REMOTE CAMERA USING SMART MODULE AND AMAZON
WEB SERVICES**

**วชิรวิทย์ เตรียมวิทยานนท์
วทัณญ์ คำรงค์โกศลภัณฑ์**

**ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560**

กล้องถ่ายภาพตรวจตราโดยใช้โมดูลอัจฉริยะและ AMAZON WEB
SERVICES
REMOTE CAMERA USING SMART MODULE AND AMAZON
WEB SERVICES

วชิรวิทย์ เตรียมวิทยานนท์
วทัณญ์ ดำรงโภคภัณฑ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

ปริญญาานิพนธ์ปีการศึกษา 2560

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง กล้องถ่ายภาพตรวจตราโดยใช้โมดูลอัจฉริยะและ AMAZON WEB SERVICES

REMOTE CAMERA USING SMART MODULE AND AMAZON WEB SERVICES

ผู้จัดทำ

นายวชิรวิทย์ เตรียมวิทยานนท์

รหัสนักศึกษา 57011087

นายวทัณญ์ ดำรงโกศลภณช์

รหัสนักศึกษา 57011089



อาจารย์ที่ปรึกษา

(รองศาสตราจารย์ ดร.สมศักดิ์ มิตะธา)



อาจารย์ที่ปรึกษาร่วม

(อาจารย์สรยุทธ กลมกล่อม)

กล้องถ่ายภาพตรวจตราโดยใช้โมดูลอัจฉริยะและ AMAZON

WEB SERVICES

นายวชิรวิทย์ เตรียมวิทยานนท์ 57011087

นายวัทฒญ์ คำรงค์ภักดิ์ 57011089

รศ.ดร.สมศักดิ์ มิตะธา อาจารย์ที่ปรึกษา

อาจารย์สรยุทธ กลมกล่อม อาจารย์ที่ปรึกษาร่วม

ปีการศึกษา 2560

บทคัดย่อ

โครงการนี้จัดทำขึ้นเพื่อศึกษา ออกแบบ และพัฒนาชิ้นงานบนสมาร์ตโมดูล SC20-E และ AMAZON WEB SERVICES ต่าง ๆ เพื่อนำมาพัฒนาต่อขยายออกมาใช้งานในเชิงพาณิชย์ โดยนำมาสร้างระบบกล้องถ่ายภาพตรวจตรา ด้วยการถ่ายภาพนิ่งเป็นระยะ ๆ และส่งภาพที่ถ่ายได้ขึ้นไปบนระบบคลาวด์สาธารณะเพื่อจัดเก็บรูปภาพ แล้วทำการวิเคราะห์หาวัตถุในรูปภาพ และหากมีวัตถุในรูปภาพใด ตรงกับวัตถุที่ได้ระบุไว้ก่อนหน้า เช่น มนุษย์ ไฟไหม้ หรือน้ำท่วม ระบบจะทำการแจ้งเตือนไปยังบุคคลที่เกี่ยวข้อง

นอกจากนี้ยังสามารถตั้งค่าความละเอียด และความถี่ในการถ่ายภาพ ของกล้องถ่ายภาพตรวจตราผ่านเว็บแอปพลิเคชันได้ รวมถึงสามารถดูภาพถ่ายย้อนหลัง และสร้างวิดีโอแบบ Time-Lapse จากเว็บแอปพลิเคชันได้

Remote Camera Using Smart Module and Amazon Web Services

Mr.Wachirawit	Triamwittayanon	57011087
Mr.Watanyou	Damrongphokaphan	57011089
Assoc.Prof.Dr.Somsak	Mitatha	Advisor
Mr.Sorayut	Glomglome	Co-Advisor

Academic Year 2560

ABSTRACT

This project aims to study, design and implement smart module SC20-E and AMAZON WEB SERVICES for using in commercial. The remote camera takes pictures periodically and sends them to public cloud system for storage and analytic. Objects inside each picture are detected by cloud service. It detected object's match pre-defined interested objects such as human, fire and flood, the system will notify relevance people, Moreover, the remote camera can be setting resolution or time interval remotely from web application. Pictures that taken by the remote camera can be watched or made a Time-Lapse video through the web application.

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีด้วยความช่วยเหลือจากหลายฝ่ายทั้งในทางตรงและทางอ้อม ปริญญาานิพนธ์ฉบับนี้จะสำเร็จลงไม่ได้หากปราศจากความช่วยเหลือของบุคคลเหล่านี้ ขอขอบพระคุณ อาจารย์ที่ปรึกษาคือ รศ.ดร.สมศักดิ์ มิตะดาและอาจารย์ที่ปรึกษาร่วมคือ อาจารย์สรยุทธ กลมกล่อม เป็นผู้ให้คำแนะนำ คำปรึกษาและให้ความช่วยเหลือตลอดการทำโครงการ ซึ่งทำให้การทำงานต่าง ๆ เป็นไปได้อย่างราบรื่นและสำเร็จลุล่วงไปได้ด้วยดี

ขอขอบพระคุณอาจารย์และบุคลากรต่าง ๆ ในสาขาวิชาวิศวกรรมคอมพิวเตอร์ที่ได้ให้คำแนะนำและตั้งสอนความรู้ต่าง ๆ มาโดยตลอด รวมถึงห้องห้องปฏิบัติการ HCRL (The Hybrid Computing Research Laboratory) ที่ได้เอื้อเฟื้อสถานที่ในการทำวิจัยและพัฒนาโครงการ

ขอขอบคุณรุ่นพี่และเพื่อนหลาย ๆ คนในภาควิชาวิศวกรรมคอมพิวเตอร์ที่ได้ให้คำแนะนำ คำปรึกษาและแบ่งปันความรู้ในทุก ๆ ด้าน

ในสุดท้ายนี้ ขอขอบพระคุณ บิดา มารดา และครอบครัวที่ได้เลี้ยงดู ตั้งสอน และให้การสนับสนุน พร้อมทั้งให้โอกาสในการศึกษาและให้กำลังใจเสมอมา

วชิรวิทย์ เตรียมวิทยานนท์
วทัณญ์ ดำรงภักดิ์

สารบัญ

	หน้า
บทคัดย่อ.....	I
ABSTRACT.....	II
กิตติกรรมประกาศ.....	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญรูป	VII
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์โครงการ	3
1.3 ประโยชน์ที่คาดว่าจะได้รับ.....	4
1.4 ขอบเขตของโครงการ	4
1.5 ข้อจำกัดของโครงการ	5
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	6
2.1 EYE TRAX The Ranger.....	6
2.2 Amazon Web Service.....	7
2.3 MQTT.....	19
2.4 SC20-E	22
2.5 งานวิจัยที่เกี่ยวข้อง	25
บทที่ 3 การออกแบบและพัฒนา.....	26
3.1 ภาพรวมการทำงานภายในระบบ	26
3.2 ก่อร่างตรวจสอบตราระยะไกล.....	27
3.3 โครงสร้างของระบบ.....	28
3.4 Architecture Diagram	29

สารบัญ (ต่อ)

	หน้า
3.5 Flowchart ของกล้องตรวจตรา.....	30
3.6 Use case Diagram.....	31
3.7 Business Process Model and Notation (BPMN)	38
บทที่ 4 ผลการทดลอง.....	43
4.1 การทดสอบคุณภาพของภาพที่ถ่ายได้จากกล้องตรวจตรา.....	43
4.2 การทดสอบการดูรูปภาพบนเว็บแอปพลิเคชัน.....	48
4.3 การทดสอบการแจ้งเตือนเมื่อเจอวัตถุ หรือสถานการณ์ที่กำหนดไว้.....	49
4.4 การทดสอบการสร้างวิดีโอแบบ Time-Lapse	54
4.5 การคำนวณค่าใช้จ่ายของตัวอุปกรณ์ และค่าบริการของ AWS	59
บทที่ 5 บทสรุปและข้อเสนอแนะ	60
5.1 บทสรุป.....	60
5.2 ปัญหาและอุปสรรค	61
5.3 แนวทางการพัฒนาต่อ.....	61

สารบัญตาราง

ตาราง	หน้า
3.1 Use case สมัครสมาชิก.....	32
3.2 Use case อัปโหลดภาพนิ่ง.....	32
3.3 Use case เพิ่ม/ลบ Remote Camera.....	33
3.4 ดูภาพถ่ายย้อนหลัง.....	33
3.5 Use case สร้างภาพเคลื่อนไหว Time-Lapse.....	34
3.6 Use case ลบภาพถ่าย.....	34
3.7 Use case ตั้งค่าอุปกรณ์ถ่ายภาพ.....	35
3.8 Use case เก็บภาพนิ่งในเมมโมรี่.....	35
3.9 Use case แจ้งเตือนเมื่อพบวัตถุที่ค้นพบ.....	36
3.10 Use case ค้นหาภาพถ่าย.....	37
3.11 Use case ถ่ายภาพนิ่ง.....	37
4.1 แสดงราคาของ EVB SC20-E และ Smart Module SC20-E.....	59
4.2 แสดงราคาค่าใช้จ่ายจาก AWS.....	59

สารบัญรูป

รูป	หน้า
1.1 EYE TRAX The Ranger	1
1.2 QUECTEL SC20-E.....	2
1.3 ภาพรวมของระบบ	3
1.4 Evaluation Board ของ Smart Module SC20-E	5
2.1 Eyetrax The Ranger	6
2.2 รูปภาพแสดงตัวอย่างการทำเว็บแอปพลิเคชันเชื่อมต่อกับ AWS Lambda ซึ่งทำตัวเป็น API และ จะเกิดการรันก็ต่อเมื่อมี Request เข้ามาที่ตัวระบบ	10
2.3 ยกตัวอย่างการใช้งานร่วมกับ S3 เมื่อมีรูปถูกอัปโหลดขึ้นมาที่ S3 ก็จะมี Event ตัวหนึ่งไป trigger เพื่อรัน lambda function ไปทำการ resize รูป.....	11
2.4 ตัวอย่าง Visual Workflow ของ AWS Step Functions.....	12
2.5 ตัวอย่างการเชื่อมต่ออุปกรณ์ด้วย MQTT.....	13
2.6 ตัวอย่างการตรวจสอบการเชื่อมต่อระหว่างเซ็นเซอร์ อุปกรณ์ และแอปพลิเคชัน	13
2.7 ตัวอย่างการเพิ่มความปลอดภัยในการจับชี้ด้วย AWS IoT และรถที่สามารถเชื่อมต่อกับเครือข่ายได้	14
2.8 ตัวอย่างการวิเคราะห์สารเคมีด้วยเครื่อง Spectrometer บนเครือข่ายที่ไม่เสถียร	14
2.9 แสดงการวิเคราะห์ข้อความในรูปภาพด้วย Amazon Rekognition.....	15
2.10 รูปภาพแสดงการวิเคราะห์วัตถุ และฉากหลังในรูปภาพด้วย Amazon Rekognition.....	16
2.11 รูปภาพแสดงการวิเคราะห์ใบหน้าด้วย Amazon Rekognition	16
2.12 แสดงการใช้งาน Amazon API Gateway	17
2.13 แสดงการประยุกต์ใช้งาน Amazon Cognition	18
2.14 รูปแสดงสถาปัตยกรรมของ MQTT	19
2.15 รูปแสดงการทำงานของ QoS ระดับ 0	20

สารบัญรูป (ต่อ)

รูป	หน้า
2.16 รูปแสดงการทำงานของ QoS ระดับ 1	21
2.17 รูปแสดงการทำงานของ QoS ระดับ 2	21
2.18 Chipset SC20 series	22
2.19 แสดง Evaluation Board ของ Smart Module SC20-E.....	23
2.20 ภาพแสดงการเปรียบเทียบแต่ละรุ่นในตระกูล SC20 series.....	24
3.1 ก่อโครงสร้างตรา.....	27
3.2 System Diagram.....	28
3.3 Architecture Diagram.....	29
3.4 Flowchart ของก่อกโครงสร้างตรา.....	30
3.5 Use case Diagram.....	31
3.6 เมื่อ Amazon S3 ได้รับรูปภาพ และสั่งให้ Amazon Step Functions ทำงาน	38
3.7 การ Login เข้าสู่ระบบ	39
3.8 การแสดงรูปภาพทั้งหมดที่ Remote Camera ถ่ายได้ผ่านเว็บแอปพลิเคชัน.....	40
3.9 การสร้างภาพ Time-Lapse จากช่วงเวลาของภาพถ่ายที่เลือกผ่านเว็บแอปพลิเคชัน.....	41
3.10 การลบรูปภาพที่กำหนดผ่านเว็บแอปพลิเคชัน.....	42
4.1 บรรยากาศขณะทำการทดสอบคุณภาพของภาพถ่ายที่ได้จากก่อกโครงสร้างตรา.....	43
4.2 แสดงภาพที่ถ่ายได้จาก Smart Module SC 20-E ถ่ายช่วงเวลา 12 นาฬิกา 23 นาที	44
4.3 เมื่อขยายภาพที่ 4.2 ที่ 500%.....	44
4.4 แสดงภาพที่ถ่ายได้จาก Smart Module SC 20-E ถ่ายช่วงเวลา 18 นาฬิกา 41 นาที	45
4.5 เมื่อขยายภาพที่ 4.4 ที่ 500%.....	45
4.6 แสดงภาพที่ถ่ายได้จาก Smart Module SC 20-E ถ่ายช่วงเวลา 18 นาฬิกา 46 นาที	46

สารบัญรูป (ต่อ)

รูป	หน้า
4.7 เมื่อขยายภาพที่ 4.6 ที่ 500%.....	46
4.8 แสดงภาพที่ถ่ายได้จาก Smart Module SC 20-E ถ่ายช่วงเวลา 19 นาฬิกา 51 นาที	47
4.9 เมื่อขยายภาพที่ 4.8 ที่ 500%.....	47
4.10 หน้า Dashboard แสดงข้อมูลพื้นฐานของผู้ใช้งาน และกล้องตรวจตราทั้งหมดที่มี	48
4.11 หน้าเว็บแอปพลิเคชัน แสดงรูปภาพที่ได้จากกล้องตรวจตรา	49
4.12 บรรยากาศขณะทำการทดสอบการแจ้งเตือนเมื่อเจอวัตถุ หรือสถานการณ์ที่กำหนดไว้.....	50
4.13 ภาพแสดงตัวอย่างรูปภาพที่ใช้ในการทดลอง	50
4.14 การแจ้งเตือนผ่าน Line Notify เมื่อเจอเหตุการณ์ไฟไหม้	51
4.15 การแจ้งเตือนผ่าน Line Notify เมื่อเจอเหตุการณ์น้ำท่วม.....	52
4.16 การแจ้งเตือนผ่าน Line Notify เมื่อเจอบุคคลบุกรุก.....	53
4.17 หน้า Dashboard แสดงข้อมูลพื้นฐานของผู้ใช้งาน และกล้องตรวจตราทั้งหมดที่มี	54
4.18 หน้าต่างการตั้งค่าของกล้องตรวจตราหน้าแรก.....	54
4.19 หน้าต่างการตั้งค่าการสร้างวิดีโอ Time-Lapse.....	55
4.20 หน้าต่างเลือกช่วงเวลาของการทำวิดีโอ Time-Lapse.....	55
4.21 หน้าต่างเลือกความละเอียดของการทำวิดีโอ Time-Lapse	56
4.22 หน้าต่างเลือกความเร็วทำวิดีโอ Time-Lapse	56
4.23 การแจ้งเตือนว่ากำลังดาวน์โหลดรูปภาพเพื่อทำวิดีโอ Time-Lapse	56
4.24 การแจ้งเตือนว่ากำลังทำวิดีโอ Time-Lapse	57
4.25 การแจ้งเตือนการทำวิดีโอ Time-Lapse เสร็จสมบูรณ์	57
4.26 ไฟล์ที่ได้จากการทำวิดีโอ Time-Lapse.....	57
4.27 ตัวอย่างวิดีโอของภาพ Time-Lapse.....	58

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ปัจจุบัน กล้องวงจรปิดได้เข้ามามีบทบาทในการสำรวจ ตรวจสอบ และบันทึกเหตุการณ์ต่าง ๆ มากขึ้น ไม่ว่าจะเป็นเหตุร้าย อุบัติเหตุ หรือการเปลี่ยนแปลงของสภาพแวดล้อม แต่การติดตั้งกล้องวงจรปิดนั้น มีข้อเสียคือต้องมีการวางสายไฟ มีศูนย์กลางที่คอยควบคุม รวมถึงค่าติดตั้งและซ่อมบำรุงที่สูงมาก นอกจากนี้ ยังไม่เหมาะกับการนำไปติดตั้งในสภาพแวดล้อมที่ไม่มีไฟฟ้า ยากต่อการเข้าถึงไฟฟ้า หรือยากต่อการเข้าถึงอินเทอร์เน็ต broadband รวมไปถึงการติดตั้งเพื่อเฟิร์มแวร์เพียงชั่วคราวที่ไม่ได้ต้องการภาพวิดีโอ หรือการมีงบประมาณที่จำกัด ไม่สามารถวางระบบได้ และความต้องการเฝ้าดูการเปลี่ยนแปลงเป็นระยะ ๆ

ในงานที่ต้องมีการติดตาม บันทึกภาพเป็นระยะ ๆ เพื่อดูการเปลี่ยนแปลง เช่น การก่อสร้าง หรือปรับปรุงอาคาร บ้าน สถานที่สาธารณะ หรือการเปลี่ยนแปลงของสภาพแวดล้อม เช่น การสำรวจป่า สถานที่ท่องเที่ยวตามธรรมชาติ การสำรวจทางวิทยาศาสตร์ หรือการสำรวจสัตว์ป่า การสำรวจสภาพอากาศ ด้วยการติดตั้งกล้องวงจรปิด และการวางระบบอาจมีค่าใช้จ่ายที่สูงเกินไป จึงต้องมีการพัฒนาอุปกรณ์ที่สามารถทำงานตอบรับในสถานการณ์ข้างต้นได้



รูปที่ 1.1 EYE TRAX The Ranger¹

ในปัจจุบัน มีผู้ผลิตบางรายได้พัฒนาอุปกรณ์ที่ตอบรับการใช้งาน โดยมีความสามารถในการถ่ายภาพและส่งข้อมูลผ่าน 4G/LTE มีกล้องความละเอียดสูงสองตัวสำหรับใช้ในตอนกลางวันและ

¹ EYETRAX [ออนไลน์] <https://eyetrax.net/> (สืบค้นข้อมูล : ตุลาคม 2560)

กลางคืน มีระบบวิเคราะห์ภาพ และมีโซล่าเซลล์ เพื่อใช้งานในสถานที่ห่างไกล แต่ภาพที่ได้จะถูกเก็บไว้ในเว็บไซต์ของบริษัท ทำให้ยากต่อการจัดการ

แนวทางการพัฒนาอุปกรณ์ สามารถใช้ไมโครคอนโทรลเลอร์ที่เชื่อมต่อกับกล้องถ่ายรูปได้ผ่านช่องทาง CSI (Camera Serial Interface) หรือ I²C หรือไมโครคอนโทรลเลอร์ที่มีการฝังระบบปฏิบัติการลินุกซ์ แต่ในสภาพการทำงานที่ไม่ต้องการใช้สายไฟเพื่อส่งข้อมูล การเลือกใช้ไมโครคอนโทรลเลอร์หรือ Smart Module ที่มีไมโครคอนโทรลเลอร์ พร้อมชิพเซตสำหรับการเชื่อมต่อผ่านเครือข่ายไร้สาย Wi-Fi หรือเครือข่ายเซลลูลาร์ เช่น 3G/4G/LTE ก็จะช่วยย่นระยะเวลาในการออกแบบ และพัฒนาฮาร์ดแวร์ลงไปได้



รูปที่ 1.2 QUECTEL SC20-E²

ดังนั้น ผู้จัดทำจึงได้เลือกใช้ Smart Module ของบริษัท QUECTEL ในรุ่น SC20-E ดังรูป 1.2 ซึ่งมีความสามารถ เช่น การติดตั้งระบบปฏิบัติการ Android เวอร์ชัน 6.1 (Marsh Mallow) มาให้ในตัว รองรับ LTE Cat.4 รองรับ Wi-Fi มาตรฐาน 802.11 b/g/n และ Bluetooth เวอร์ชัน 4.1 LE รองรับเทคโนโลยี MIMO รองรับ GPS/GLONASS ความไวสูงและรองรับการเชื่อมต่อด้วยอินเทอร์เฟซที่หลากหลาย เช่น LCM, Camera (MIPI-CSI), UART, USB, I²C และ GPIO

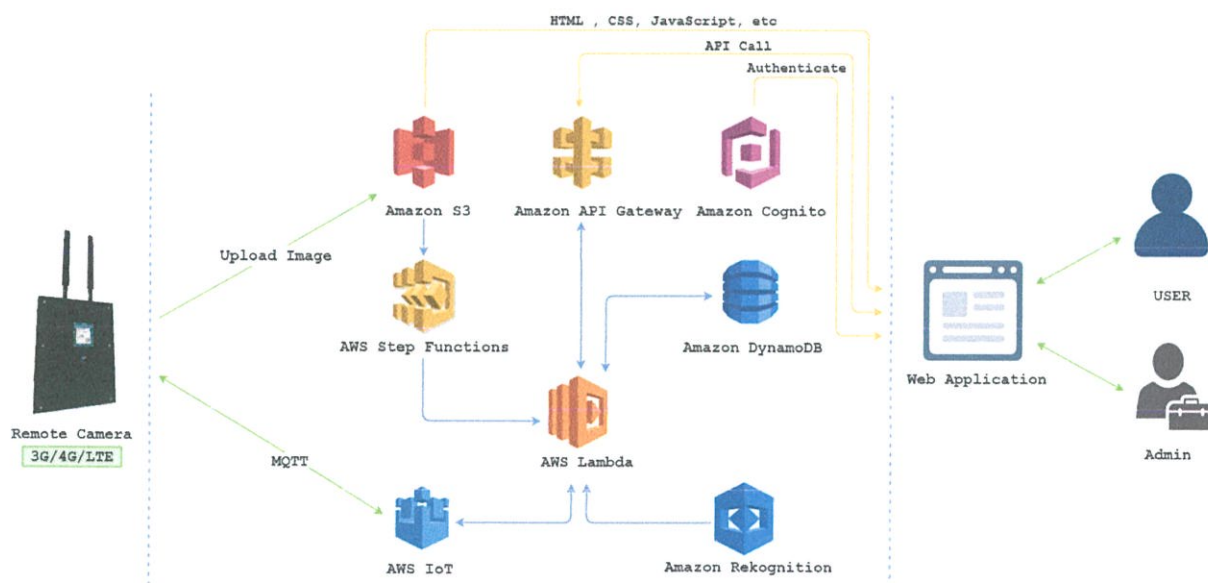
ตัวอุปกรณ์จะเป็น Internet of Things เพื่อให้การเพิ่มจำนวนอุปกรณ์เมื่อต้องนำไปใช้งานจริง หรือการใช้งานในเชิงพาณิชย์เป็นไปได้ โดยจะใช้ Public Cloud ของ Amazon Web Services โดยการใช้ Service ต่าง ๆ คือ

- AWS IoT เป็น Service ที่เกี่ยวกับ Internet of Things ที่มีความปลอดภัยสูงมาก และใช้โปรโตคอล MQTT ในการสื่อสารระหว่าง IoT Device และ AWS IoT

² LTE SC20 Smart Module [ออนไลน์] <http://www.quectel.com/product/sc20.htm> (สืบค้นข้อมูล: ตุลาคม 2560)

- AWS Lambda นับเป็น Service ที่น่าสนใจ คือเป็น Service ที่ทำการประมวลผลตามชุดคำสั่งที่ได้เขียนไว้เมื่อเหตุการณ์ (Event) ที่กำหนดไว้เกิดขึ้น โดยไม่จำเป็นต้องใช้ Server เพื่อใช้งาน
- Amazon Step Functions เป็นบริการที่ทำให้นักพัฒนาสร้าง ใช้งาน ปรับแต่งงานเบื้องหลัง ให้ทำงานในแบบขนาน (Parallel) หรือแบบลำดับขั้น (Sequential) หรืออาจกล่าวได้ว่า คือบริการที่ติดตาม และจัดการกระบวนการทำงานในคลาวด์
- Amazon Rekognition เป็นบริการ Deep Learning ที่ช่วยในการตรวจสอบ วิเคราะห์ และจำแนกรูปภาพเพื่อหาวัตถุที่อยู่ในภาพ เช่น มนุษย์ สัตว์ ต้นไม้ และอาจวิเคราะห์ได้ถึงสายพันธุ์ของสัตว์และอารมณ์ของมนุษย์
- AWS S3 ใช้สำหรับการเก็บข้อมูล รูปภาพและเว็บไซต์

ด้วยการใช้งาน Smart Module (SC20-E) และ Amazon Web Service (AWS) จะช่วยให้การพัฒนาอุปกรณ์ และ Cloud Services เป็นไปได้อย่างรวดเร็ว และมีประสิทธิภาพมากขึ้น



รูปที่ 1.3 ภาพรวมของระบบ

1.2 วัตถุประสงค์โครงการ

- 1) เพื่อสร้างระบบตรวจสอบการเปลี่ยนแปลงของสภาพแวดล้อม
- 2) เพื่อศึกษาการใช้งาน Smart Module (SC20-E) เพื่อใช้ในเชิงพาณิชย์
- 3) เพื่อศึกษาและใช้งาน IoT Platform และ Services บน AWS

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ระบบที่สามารถดูการเปลี่ยนแปลงของสภาพแวดล้อมย้อนหลังในรูปแบบของภาพนิ่ง
- 2) ระบบที่สามารถดูการเปลี่ยนแปลงของสภาพแวดล้อมย้อนหลังในรูปแบบของภาพเคลื่อนไหว (Time -lapse)
- 3) ระบบที่คัดกรองวัตถุที่สนใจในรูปภาพ เพื่อการค้นหา และแจ้งเตือน
- 4) เข้าใจการทำงานของ Smart Module (SC20-E)
- 5) เข้าใจการทำงานของ IoT Platform และ Cloud Services บน AWS
- 6) ชื่นงานประยุกต์ใช้ บน Smart Module เองโดยใช้ IoT Platform ที่รองรับ ทำให้สะดวกต่อการพัฒนาชิ้นงาน

1.4 ขอบเขตของโครงการ

มีอุปกรณ์ถ่ายภาพด้วย Smart Module (SC20-E) เพื่อส่งข้อมูลไปเก็บที่ Cloud Service ของ AWS โดยการทำงานเป็นแบบอัตโนมัติผ่านทาง 3G/4G/LTE เพื่อเก็บข้อมูลลงระบบ วิเคราะห์ภาพถ่าย และเรียกดูย้อนหลัง ซึ่งมีการทำงานดังนี้

1) การทำงานด้านอุปกรณ์ถ่ายภาพ

1.1 อุปกรณ์สามารถถ่ายรูป และส่งข้อมูลไปยัง Cloud Service ของ Amazon Web Service(AWS)

1.2 ส่งข้อมูลภาพนิ่งที่บันทึกได้ไปยัง Amazon Web Service (AWS) ผ่านเครือข่าย 3G/4G/LTE

1.3 พัฒนา Application บนระบบปฏิบัติการ Android เวอร์ชัน 6.1 บน Smart Module (SC20-E) เพื่อให้ถ่ายภาพนิ่ง

2) การทำงานด้าน Amazon Web Service (AWS)

2.1 รับข้อมูลรูปภาพจาก Smart Module (SC20-E)

2.2 วิเคราะห์ จำแนกวัตถุในภาพ และจัดเก็บลงฐานข้อมูล

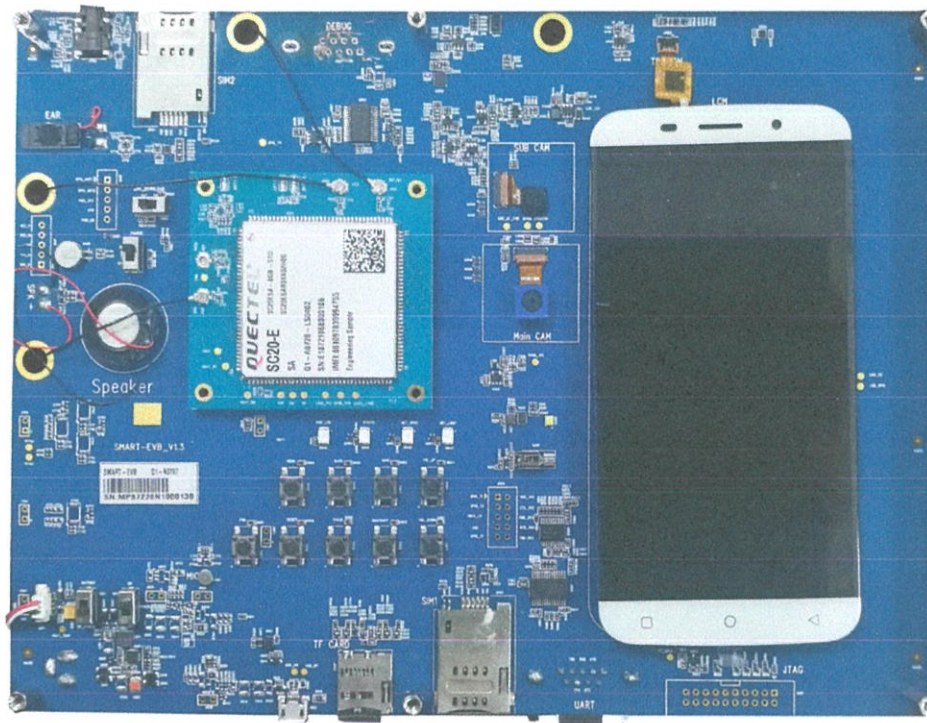
3) ผู้ใช้งานดูภาพย้อนหลังผ่านทาง เว็บแอปพลิเคชัน

3.1 เลือกดูภาพย้อนหลัง ตามช่วงเวลา

3.2 รองรับการใช้งานพร้อมกันหลายคน

1.5 ข้อจำกัดของโครงการ

Smart Module (SC20-E) ใช้บน Evaluation Board ที่มี Sensor ซึ่งทำให้สะดวกในการพัฒนา
ชิ้นงาน เพื่อศึกษาการพัฒนาบนอุปกรณ์ทำเป็นต้นแบบ แต่ไม่เหมาะสำหรับนำไปใช้งานจริง



รูปที่ 1.4 Evaluation Board ของ Smart Module SC20-E

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 EYE TRAX The Ranger



รูปที่ 2.1 Eyetrax The Ranger³

Eye Trax The Ranger คือ กล้องถ่ายภาพนิ่งควบคุมทางไกลผ่านทางเครือข่ายไร้สาย ตัวกล้องมีความละเอียด 2 ล้านพิกเซล (1920 x 1080) สามารถเปลี่ยนเลนส์ได้ พร้อมอินฟราเรดที่ส่องได้ไกล 30 เมตร เชื่อมต่อกับ 3G หรือ 4G พร้อมกับค่าบริการรายเดือน ที่ไม่จำกัดอัตราการรับส่งข้อมูล และสามารถนำไปใช้ได้ทั่วโลก

Eye Trax มาพร้อมกับโซล่าเซลล์ และแบตเตอรี่ ที่สามารถใช้งานได้นาน 7 วัน โดยไม่มีแสงแดด และสามารถชาร์จโดยใช้โซล่าเซลล์ ขนาด 30 วัตต์ ที่แถมมากับตัวกล้องได้ นอกจากนี้ ตัวกล้องยังสามารถเชื่อมต่อกับอุปกรณ์เสริมต่าง ๆ เช่น ไซเรน หรือเซ็นเซอร์อินฟราเรด เพิ่มเติมได้

กล้อง Eye Trax ใช้เซ็นเซอร์อินฟราเรดแบบ Passive Infrared Sensors (PIR) ซึ่งใช้เทคโนโลยีการตรวจจับความร้อนเพื่อแจ้งเตือนการบุกรุกบนไซต์งาน ในขณะที่กล้องส่วนใหญ่ในตลาด ใช้การตรวจจับการเคลื่อนไหวผ่านการเปลี่ยนแปลงในแต่ละพิกเซล ซึ่งจะก่อให้เกิดสัญญาณเตือนภัยปลอม (false alarm) เซ็นเซอร์ Eye Trax PIR สามารถกำจัดสัญญาณเตือนภัยปลอม โดยการใช้ตาราง

³ EYETRAX [ออนไลน์] <https://eyetrax.net/> (สืบค้นข้อมูล : ตุลาคม 2560)

ความร้อน ซึ่งจะแจ้งเตือนเมื่อพบรูปแบบความร้อนของมนุษย์ หรือยานพาหนะ ที่มารบกวพื้นที่ การตรวจจับ เมื่อมีการตรวจพบการรบกวนในพื้นที่ กล้อง Eye Trax จะตอบสนอง และจับภาพ วิดีโอภายในครึ่งวินาที

นอกจากนี้ Eye Trax ยังมาพร้อมกับคลาวด์แบบส่วนตัว ที่จ่ายค่าบริการแบบรายเดือน พร้อมพื้นที่เก็บข้อมูลไม่จำกัด และสามารถเข้าถึงผ่านเว็บเบราว์เซอร์ หรือแอปพลิเคชันบนสมาร์ตโฟน ได้ โดยสามารถดูภาพย้อนหลังได้ สามารถดูภาพ Live View ได้ สามารถสร้างภาพเคลื่อนไหว Time-Lapse ได้ และสามารถตั้งค่าต่าง ๆ ของกล้องได้

กล้อง Eye Trax มีราคาเริ่มต้นที่ 2,590 ดอลลาร์สหรัฐ (ราคารวมกล้อง แผงโซล่าเซลล์ รีโมต Mounting แบตเตอรี่ และที่ชาร์จ) และค่าบริการคลาวด์รายเดือนที่ 80 ดอลลาร์สหรัฐ

2.2 Amazon Web Service

Amazon Web Service (AWS) เป็นบริษัทในเครือของ Amazon.com ที่ให้บริการคลาวด์ คอมพิวติง แบบ on-demand สำหรับบุคคลทั่วไป บริษัท หรือรัฐบาล โดยการสมัครใช้บริการ และ มาพร้อม free-tier ให้ใช้บริการฟรี 12 เดือน

เทคโนโลยี AWS ถูกติดตั้งบนเซิร์ฟเวอร์ฟาร์มทั่วโลก และดูแลโดย บริษัท ในเครือของ Amazon ค่าธรรมเนียมในการใช้งาน จะขึ้นอยู่กับทางเลือกใช้งาน ฮาร์ดแวร์ / ระบบปฏิบัติการ / ซอฟต์แวร์ / เน็ตเวิร์ค ของผู้ให้บริการ

ในปี พ.ศ. 2560 AWS มีบริการมากกว่า 90 รายการ ครอบคลุมหลากหลายสาขา ทั้งการ ประมวลผล ระบบจัดเก็บข้อมูล ระบบเครือข่าย ฐานข้อมูล การวิเคราะห์ การจัดการอุปกรณ์ เคลื่อนที่ เครื่องมือสำหรับนักพัฒนาซอฟต์แวร์ และเครื่องมือสำหรับ IoT

บริการที่ได้รับความนิยมมากที่สุด ได้แก่ Amazon Elastic Compute Cloud (EC2) และ Amazon Simple Storage Service (S3) ในขณะที่บริการส่วนใหญ่ของ AWS ไม่ได้ถูกเปิดเผย โครงสร้างภายในแก่ผู้ใช้งาน แต่ให้ฟังก์ชันการทำงานผ่าน API สำหรับนักพัฒนาซอฟต์แวร์ เพื่อใช้ ในแอปพลิเคชัน

2.2.1 Amazon Simple Storage Service (Amazon S3)

Amazon S3 ย่อมาจาก Amazon Simple Storage Service คือ บริการการพื้นที่จัดเก็บ ข้อมูลบนอินเทอร์เน็ต ซึ่งออกแบบมาให้ผู้พัฒนา สามารถขยายขนาดได้ง่าย

Amazon S3 มีอินเตอร์เฟซการเชื่อมต่อที่ง่ายในการเก็บ และเรียกข้อมูลออกมา ใน ขนาดใดก็ได้ ทุกที่ ทุกเวลา และมอบความสามารถในการขยายขนาดได้สูง มีความน่าเชื่อถือ มีความรวดเร็ว และประหยัด

2.2.1.1 องค์ประกอบของ Amazon S3

1) ตะกร้า (Buckets) คือ สิ่งที่ใช้ในการเก็บวัตถุ หรือ Object ใน Amazon S3 ซึ่งทุก ๆ Object จะต้องถูกจัดเก็บไว้ใน Bucket ยกตัวอย่างเช่น Object ที่ชื่อ photos/puppy.jpg ที่ถูกเก็บไว้ใน Bucket ที่ชื่อ johnsmith การที่จะเข้าถึง Object นี้ สามารถทำได้ด้วยการเข้าไปที่ <http://johnsmith.s3.amazonaws.com/photos/puppy.jpg>

Bucket สามารถนำมาประยุกต์ใช้งานได้อย่างหลากหลาย ยกตัวอย่างเช่น การทำ namespace การแบ่งสิทธิในการเข้าถึง และจัดการข้อมูล

2) วัตถุ (Objects) คือ Entities ที่ถูกเก็บไว้ใน Amazon S3 ภายใน Object จะประกอบไปด้วยข้อมูลของตัว Object และ Metadata

ข้อมูลของ Object เดี่ยว ๆ อาจทำให้เกิดความไม่ชัดเจนขึ้นใน Amazon S3 จึงต้องมี metadata ซึ่งเป็นคู่ข้อมูล ประกอบด้วย ชื่อ และ ค่า และโดยปกติ ข้อมูลใน metadata จะประกอบไปด้วยข้อมูลที่บอกว่า Object ถูกแก้ไขล่าสุดเมื่อใด และยังสามารถกำหนด metadata เองได้

Object จะถูกระบุด้วย Key (โดยใช้ ชื่อ) และเวอร์ชัน ใด

3) กุญแจ (Keys) คือ สิ่งที่ใช้ระบุถึง Object ต่าง ๆ โดยจะต้องไม่ซ้ำกัน และแต่ละ Object จะมี Key เพียงอันเดียวเท่านั้น โดย Key คือการนำ bucket มารวมกับ key หรือชื่อ Object และ เลขเวอร์ชันของ Object มาต่อกัน (ถ้ามี) ยกตัวอย่างเช่น <http://doc.s3.amazonaws.com/2006-03-01/AmazonS3.wsdl>

“doc” คือชื่อของ Bucket และ “2006-03-01/AmazonS3.wsdl” ก็คือ Key

4) ภูมิภาค (Regions) คือพื้นที่บน โลก ที่ Amazon S3 ตั้งอยู่ การเลือก regions ที่เหมาะสม จะสามารถลดค่าใช้จ่าย หรือลด latency ลงไปได้ ในส่วนของ Amazon S3 มี Region ให้เลือก 14 ที่ คือ

- US East (N. Virginia)
- US East (Ohio)
- US West (N. California)
- US West (Oregon)
- Canada (Central)
- Asia Pacific (Mumbai)
- Asia Pacific (Seoul)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)

- Asia Pacific (Tokyo)
- EU (Frankfurt)
- EU (Ireland)
- EU (London)
- South America (São Paulo)

5) รูปแบบการเก็บข้อมูลบน Amazon S3 (Amazon S3 data Consistency model) Amazon S3 ใช้รูปแบบการบันทึกแบบ read-after-write consistency ซึ่งเวลาแก้ไขหรือเพิ่มข้อมูล จะทำเป็น atomic ที่จะช่วยป้องกันในกรณีที่เกิดการแก้ไขข้อมูลในขณะที่อ่านข้อมูลที่ถูกอ่าน อาจจะเป็นข้อมูลเก่า หรือข้อมูลใหม่ก็ได้ แต่จะไม่ใช่ข้อมูลที่ผสมระหว่างข้อมูลใหม่ และเก่าแน่นอน

2.2.2 AWS Lambda

AWS Lambda คือ บริการสำหรับประมวลผล ที่สามารถรันโค้ดได้ โดยไม่ต้องสร้าง Virtual Machine หรือลงระบบปฏิบัติการ โดย AWS Lambda จะทำงานก็ต่อเมื่อเราต้องการสั่งโปรแกรมให้ทำงาน และมีการปรับขนาดของระบบที่ใช้ทำงานโดยอัตโนมัติ สามารถรองรับการเรียกใช้วันละไม่กี่ครั้ง จนถึงหลักพันครั้งต่อวินาทีได้ และคิดค่าใช้จ่ายแค่เวลาทั้งหมดที่ทำงานเท่านั้น หากไม่มีการเรียกใช้งาน ระบบจะไม่คิดเงิน

ด้วย AWS Lambda เราสามารถรันโค้ดของแอปพลิเคชัน หรือเซอร์วิสเบื้องหลังที่หลากหลาย โดยไม่ต้องมีผู้ดูแลระบบ เพราะ AWS Lambda เอง รันอยู่บน infrastructure ที่มีความสามารถสูง และจะคอยจัดการระบบทุกอย่างให้เรา ทั้งเซิร์ฟเวอร์ ระบบปฏิบัติการ การซ่อมบำรุง และการขยายขนาด สิ่งที่คุณพัฒนาต้องทำกับ AWS Lambda คือเขียนโค้ดที่ AWS Lambda รองรับ เช่น Node.js, Java, Python และ C#

2.2.2.1 Architecture

การรันโปรแกรมบน AWS Lambda นั้น ตัว Lambda จะแปลงโค้ดที่เราจะต้องการรัน หรือเรียกสั้น ๆ ว่าเป็น Lambda Function เป็น Container เมื่อมีการรันโปรแกรมผ่านช่องทางที่เราผูก Event ไว้ ตัว Lambda เองก็จะรันตัวโปรแกรมนั้นให้เราจบ ๆ ไปเป็น event และเมื่อไม่มีการเรียกใช้งานก็จะอยู่ในสถานะ freeze ตัวเองไป ไม่มีการรันและใช้งานทรัพยากรใด ๆ ของตัวประมวลผล



รูปที่ 2.2 รูปภาพแสดงตัวอย่างการทำเว็บแอปพลิเคชันเชื่อมต่อกับ AWS Lambda ซึ่งทำตัวเป็น API และจะเกิดการรันก็ต่อเมื่อมี Request เข้ามาที่ตัวระบบ⁴

2.2.2.2 Lambda Function

การเขียนโปรแกรมบน AWS Lambda นั้นนักพัฒนา สามารถเขียนฟังก์ชันที่เราต้องการจะรันในการเกิด Event แต่ละครั้ง โดยเราสามารถเขียนบนภาษาที่ AWS Lambda รองรับได้ คือ Node.js, Java, Python และ C# โดยเมื่อมี event เกิดขึ้น ตัว Lambda จะเข้ามารันฟังก์ชันที่ได้เชื่อมต่อไว้กับตัว event นั้น

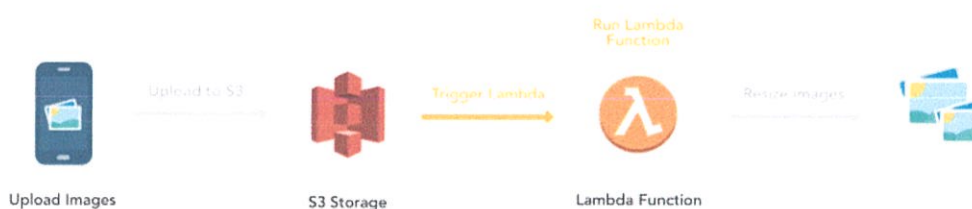
2.2.2.3 Event Driven

การรัน Lambda Function ที่เราเขียนไว้นั้น จะเกิดขึ้นก็ต่อเมื่อมี Event มาทำการสั่งรันตัวฟังก์ชัน ซึ่งมีด้วยกันหลาย Trigger โดยแบ่งเป็น 2 แบบหลัก ๆ คือ

- 1) Streaming Event เป็นอีเวนต์ที่เข้ามาตลอดเวลาในช่วงระยะเวลาหนึ่ง ๆ เช่น ดึงข้อมูลจาก social network เป็นต้น โดยอีเวนต์นี้สามารถที่จะเชื่อมต่อกับเซอร์วิสของ AWS 2 ตัว คือ
 - DynamoDB Streaming
 - AWS Kinesis

⁴ AWS Lambda [ออนไลน์] <https://aws.amazon.com/lambda/> (สืบค้นข้อมูล : ตุลาคม 2560)

- 2) Non-Streaming Event เป็นอีเวนต์ปกติที่สามารถ trigger ขึ้นมาด้วยการกระทำบางอย่าง เช่น การเรียก HTTP Request / IoT Event เป็นต้น โดยสามารถเชื่อมต่อกับ AWS Services ได้ดังนี้
- API Gateway สำหรับสร้าง API ให้สามารถเรียกได้ผ่าน HTTP Request
 - AWS IoT
 - Cloudfront
 - Cloudwatch
 - S3
 - SNS



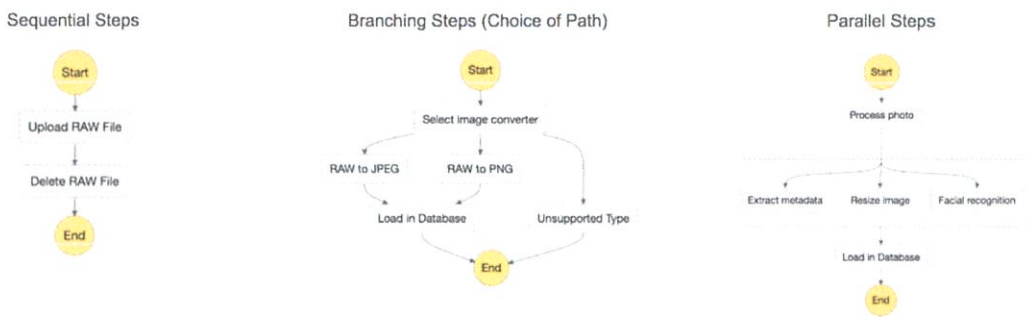
รูปที่ 2.3 ยกตัวอย่างการใช้งานร่วมกับ S3 เมื่อมีรูปถูกอัปโหลดขึ้นมาที่ S3 ก็จะมี Event ตัวหนึ่งไป trigger เพื่อรัน lambda function ไปทำการ resize รูป⁵

⁵ AWS Lambda [ออนไลน์] <https://aws.amazon.com/lambda/> (สืบค้นข้อมูล : ตุลาคม 2560)

2.2.3 Amazon Step Function

AWS Step Functions เป็นหนึ่งใน AWS Serverless Platform ซึ่งช่วยจัดการ Micro service ที่เป็นส่วนประกอบของแอปพลิเคชัน โดยใช้ Visual Workflow ทำให้การสร้างแอปพลิเคชันจากการนำ Micro Service ต่าง ๆ มาประกอบกันได้ง่ายขึ้น

AWS Step Functions เป็นบริการที่มีความน่าเชื่อถือ ในการจัดระเบียบการทำงานของแต่ละส่วนประกอบของแอปพลิเคชัน พร้อมทั้งสามารถจัดการในรูปแบบกราฟิก เพื่อให้ง่ายต่อการจัดวางการทำงาน และช่วยให้การสร้างแอปพลิเคชันที่มีความซับซ้อน สามารถทำได้ง่าย และสามารถติดตามการทำงานในแต่ละฟังก์ชันได้ พร้อมทั้งสามารถตรวจจับ และรองรับได้หากเกิดข้อผิดพลาดระหว่างการทำงาน เช่นให้ลองทำใหม่อีกครั้ง และทำการบันทึก (log) การทำงานแต่ละขั้นตอนไว้ เพื่อให้ นักพัฒนาสามารถแก้ไขปัญหาได้อย่างรวดเร็ว



รูปที่ 2.4 ตัวอย่าง Visual Workflow ของ AWS Step Functions⁶

2.2.4 AWS IoT

AWS IoT คือ บริการคลาวด์ที่ช่วยให้การจัดการกับอุปกรณ์ IoT ได้ง่าย มีความปลอดภัย และยังสามารถเชื่อมต่อกับบริการอื่น ๆ ของ Amazon ได้ และ AWS IoT ยังมีความสามารถในการรองรับอุปกรณ์นับล้านตัว และรองรับการส่งข้อความนับพันล้านข้อความ และสามารถประมวลผล และส่งต่อไปยังบริการอื่น ๆ ได้อย่างปลอดภัย และมีความน่าเชื่อถือ นอกจากนี้ AWS IoT ยังทำให้แอปพลิเคชัน สามารถติดต่อกับอุปกรณ์ IoT ได้ตลอดเวลา แม้ว่าอุปกรณ์นั้น จะไม่ได้เชื่อมต่ออยู่ก็ตาม

AWS IoT ถูกออกแบบให้สามารถเชื่อมต่อระหว่างคลาวด์ กับอุปกรณ์ IoT ได้ง่าย โดยรองรับการเชื่อมต่อด้วยโปรโตคอลที่หลากหลาย เช่น HTTP, WebSocket หรือ MQTT ซึ่งเป็นโปรโตคอลที่มีน้ำหนักเบา และถูกออกแบบให้ทนต่อการสื่อสารที่ไม่เสถียร ใช้งานง่าย และกินแบนด์วิธต่ำ

⁶ AWS Step Functions [ออนไลน์] <https://aws.amazon.com/step-functions/> (สืบค้นข้อมูล : กุมภาพันธ์ 2561)

นอกจากนี้ AWS IoT ยังรองรับโปรโตคอลอื่น ๆ อีกมาก สามารถสร้างโปรโตคอลเองได้ และสามารถเชื่อมต่ออุปกรณ์ IoT ที่ใช้โปรโตคอลต่างกันได้

2.2.4.1 ตัวอย่างการเชื่อมต่ออุปกรณ์ด้วย MQTT



ก) อุปกรณ์นับล้านสามารถ
กำลังรับหรือส่งข้อมูล

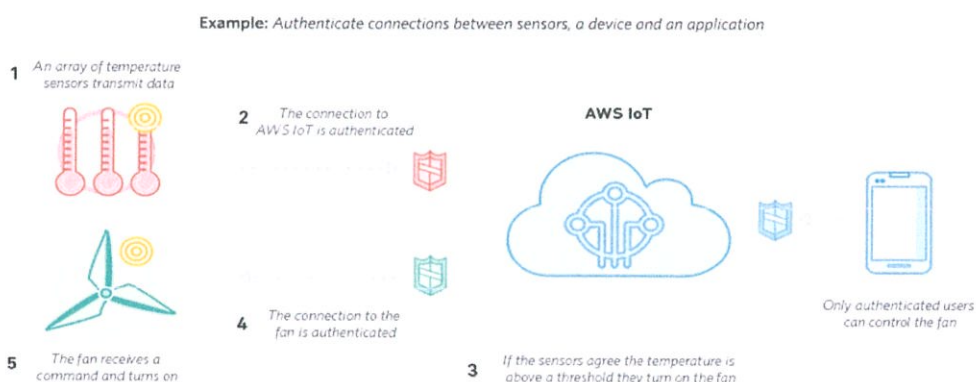
ข) ข้อความมีการรับและส่งผ่านโปรโตคอล
MQTT ที่มีขนาดเล็ก และประหยัดแบนด์วิธ

ค) AWS IoT เปิดช่องทางให้อุปกรณ์
สามารถสื่อสารกับ AWS Service

รูปที่ 2.5 ตัวอย่างการเชื่อมต่ออุปกรณ์ด้วย MQTT⁷

ในเรื่องของความปลอดภัย AWS IoT มีการตรวจสอบความถูกต้อง และการเข้ารหัสแบบ End-to-End ในทุก ๆ จุดของการเชื่อมต่อ นอกจากนี้ ยังสามารถรักษาความปลอดภัยด้วยกำหนดสิทธิ์ในการเข้าถึงได้

2.2.4.2 ตัวอย่างการตรวจสอบการเชื่อมต่อระหว่างเซ็นเซอร์ อุปกรณ์ และแอปพลิเคชัน



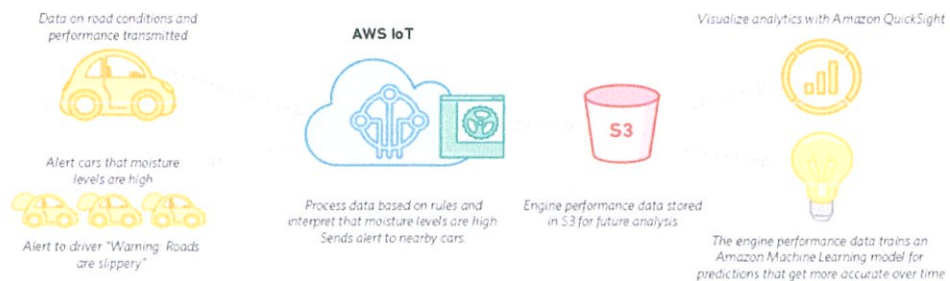
รูปที่ 2.6 ตัวอย่างการตรวจสอบการเชื่อมต่อระหว่างเซ็นเซอร์ อุปกรณ์ และแอปพลิเคชัน⁸

⁷ AWS IoT Services Overview [ออนไลน์] <https://aws.amazon.com/iot/> (สืบค้นข้อมูล : ตุลาคม 2560)

⁸ เรื่องเดียวกัน

ด้วย AWS IoT ออกแบบให้สามารถรองรับ เปลี่ยนแปลงข้อมูล หรือ ทำงานในอุปกรณ์ได้ โดยไม่ต้องปิด AWS IoT ก่อน นอกจากนี้ ยังสามารถเชื่อมต่อไปยัง AWS Service อื่น ๆ ได้ เช่น AWS Lambda, Amazon Kinesis, Amazon S3, Amazon Machine Learning, Amazon DynamoDB, Amazon CloudWatch และ Amazon Elasticsearch

2.2.4.3 ตัวอย่างการเพิ่มความปลอดภัยในการขับขี่



รูปที่ 2.7 ตัวอย่างการเพิ่มความปลอดภัยในการขับขี่ด้วย AWS IoT และรถที่สามารถเชื่อมต่อกับเครือข่ายได้

ความสามารถอีกอย่างหนึ่งของ AWS IoT คือ การเก็บสถานะล่าสุดของ อุปกรณ์ไว้ (Shadow) ซึ่งสามารถอ่าน หรือเขียนทับลงไปได้ ทำให้แอปพลิเคชัน มองเห็นอุปกรณ์ ตลอดเวลา และสามารถอ่านหรือเขียนคำสั่งให้อุปกรณ์นั้นได้ แม้การเชื่อมต่อจะขาดหายไปก็ตาม ซึ่งตัว Device จะมีสถานะตาม Shadow ของมันทันที เมื่อสามารถเชื่อมต่อกับ AWS IoT ได้

2.2.4.4 ตัวอย่างแสดงการวิเคราะห์ข้อมูลทางเคมีบนเครื่อง Spectrometer



รูปที่ 2.8 ตัวอย่างการวิเคราะห์สารเคมีด้วยเครื่อง Spectrometer บนเครือข่ายที่ไม่เสถียร¹⁰

⁹ AWS IoT Services Overview [ออนไลน์] <https://aws.amazon.com/iot/> (สืบค้นข้อมูล : ตุลาคม 2560)

¹⁰ เรื่องเดียวกัน

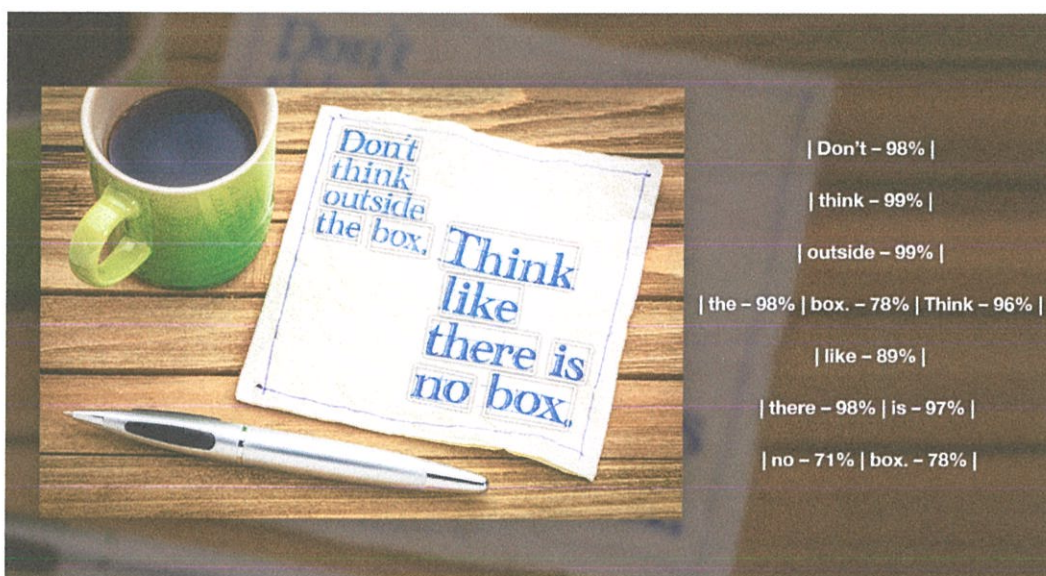
2.2.5 Amazon Rekognition

Amazon Rekognition คือ บริการที่ช่วยวิเคราะห์รูปภาพให้กับแอปพลิเคชันที่เรียกใช้ สามารถตรวจจับวัตถุ ฉากพื้นหลัง ข้อความ ใบหน้า นอกจากนี้ ยังสามารถเปรียบเทียบความคล้ายของใบหน้าได้ด้วย

Amazon Rekognition ใช้เทคโนโลยีการเรียนรู้ Deep Learning ที่ได้รับการพิสูจน์แล้ว ซึ่งถูกพัฒนาขึ้นโดยนักวิทยาศาสตร์ด้านการมองเห็นในคอมพิวเตอร์ของ Amazon เพื่อวิเคราะห์ภาพนับพันล้านภาพต่อวันจาก Prime Photos

Amazon Rekognition ใช้แบบจำลอง Deep Neural Network เพื่อตรวจจับวัตถุ ฉากนับพัน ๆ ภาพ ในรูปภาพ และพัฒนาการจำแนกแบบใหม่ ๆ อย่างต่อเนื่อง

Amazon Rekognition สามารถตรวจจับ และรับรู้เนื้อหาต้นฉบับจากรูปภาพ เช่น ชื่อถนน คำอธิบายภาพ ข้อผลิตภัณฑ์ และป้ายทะเบียนรถยนต์



รูปที่ 2.9 แสดงการวิเคราะห์ข้อความในรูปภาพด้วย Amazon Rekognition¹¹

Amazon Rekognition สามารถแยกวัตถุ นับพันวัตถุ เช่น ยานพาหนะ สัตว์เลี้ยง เฟอ์นเจอร์ และยังให้คะแนนความมั่นใจของการวิเคราะห์ด้วย

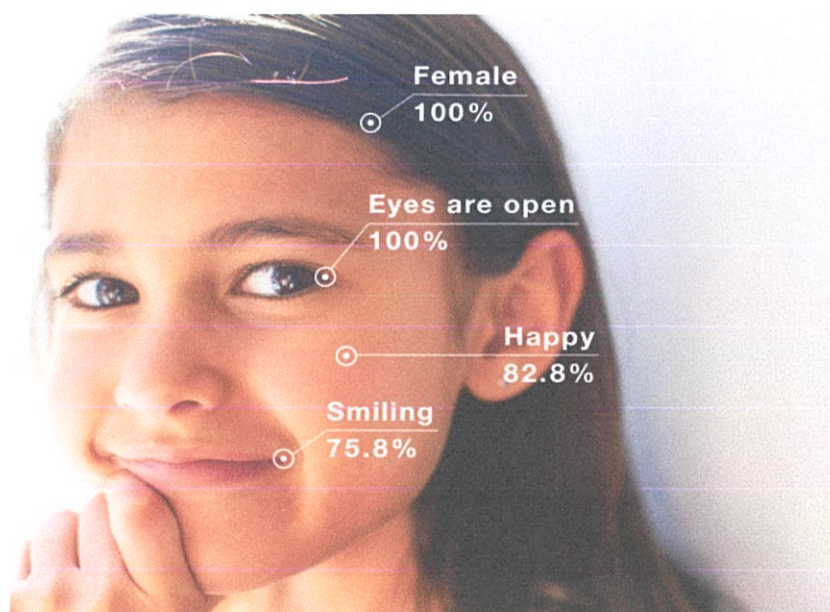
¹¹ Amazon Rekognition [ออนไลน์] <https://aws.amazon.com/rekognition/> (สืบค้นข้อมูล : ตุลาคม 2560)

นอกจากนี้ Amazon Rekognition ยังสามารถวิเคราะห์ฉากหลัง เช่น ชายหาด พระอาทิตย์ตกได้



รูปที่ 2.10 รูปภาพแสดงการวิเคราะห์วัตถุ และฉากหลังในรูปภาพด้วย Amazon Rekognition¹²

Amazon Rekognition สามารถวิเคราะห์รูปภาพใบหน้า และบอกได้ว่าใบหน้ากำลังยิ้ม หรือตาเปิดอยู่ ด้วยพีเจอร์รี่ ทำให้สามารถวิเคราะห์ความรู้สึกของบุคคลในภาพได้อย่างง่ายดาย



รูปที่ 2.11 รูปภาพแสดงการวิเคราะห์ใบหน้าด้วย Amazon Rekognition¹³

¹² Amazon Rekognition [ออนไลน์] <https://aws.amazon.com/rekognition/> (สืบค้นข้อมูล : ตุลาคม 2560)

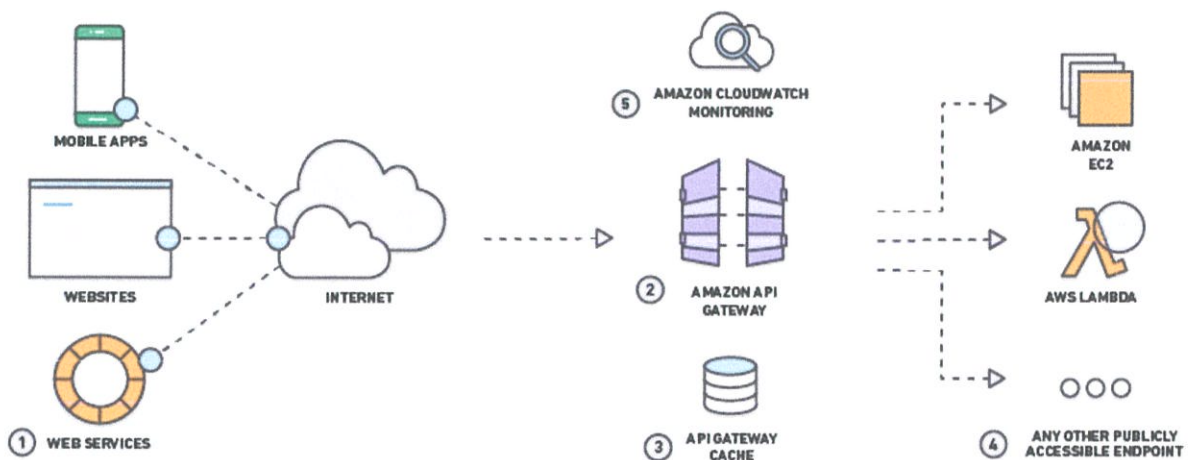
¹³ เรื่องเดียวกัน

2.2.6 Amazon API Gateway

Amazon API Gateway คือ บริการที่ช่วยนักพัฒนาสร้าง ดูแล ปรับปรุง ตรวจสอบ และรักษาความปลอดภัยให้กับ API ของตนเอง ซึ่ง Amazon API Gateway จะทำหน้าที่เหมือนประตูหน้าด่านของแอปพลิเคชัน เพื่อที่จะเข้าไปใช้งานข้อมูล หรือบริการอื่น ๆ ที่อยู่เบื้องหลัง เช่น โค้ดที่ทำงานอยู่บน AWS Lambda หรือการทำงานบน Amazon Elastic Compute Cloud (Amazon EC2)

นอกจากนี้ Amazon API Gateway ยังช่วยจัดการกราฟฟิค การยืนยันตัวตน การควบคุมการเข้าถึง การมอนิเตอร์ และการจัดการเวอร์ชันของ API และรองรับการเรียกใช้งาน API ตั้งแต่หลักร้อยถึงหลักพันครั้งต่อวินาที

Amazon API Gateway คิดค่าบริการตามจำนวนครั้งที่เรียกใช้งาน และการค่าบริการสำหรับการโอนถ่ายข้อมูล



รูปที่ 2.12 แสดงการใช้งาน Amazon API Gateway¹⁴

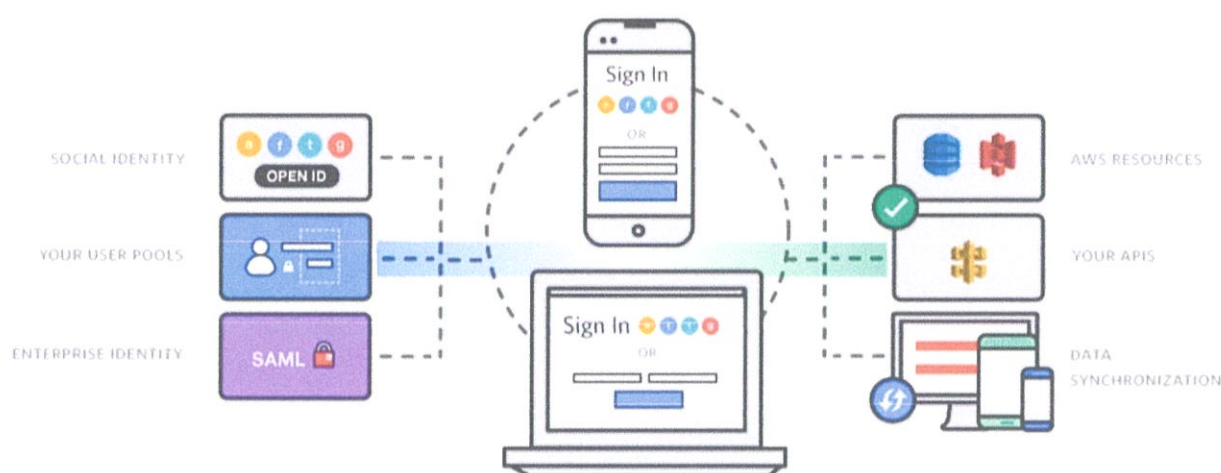
จากรูป 2.12 เมื่อแอปพลิเคชันบนโทรศัพท์ เว็บไซต์ หรือ เว็บเซอร์วิส อื่น ๆ ที่ต้องการจะใช้ทรัพยากรภายในคลาวด์ เช่น AWS Lambda และ Amazon EC2 จะต้องเรียกใช้งานผ่าน API Gateway

¹⁴ Amazon API Gateway [ออนไลน์] <https://aws.amazon.com/api-gateway/> (สืบค้นข้อมูล : ตุลาคม 2560)

2.2.7 Amazon Cognito

Amazon Cognito ช่วยให้นักพัฒนาสามารถสร้างฟังก์ชันการสมัครสมาชิก หรือการล็อกอินเข้าใช้งานผ่านเว็บหรือแอปพลิเคชันได้อย่างง่ายดาย และยังสามารถใช้การยืนยันตัวตนผ่านโซเชียลมีเดีย เช่น Facebook Twitter หรือ Amazon ได้ นอกจากนี้ Amazon Cognito ยังเก็บข้อมูลลงบนพื้นที่ของผู้ใช้งาน ทำให้แอปพลิเคชันสามารถทำงานต่อไปได้ แม้จะออฟไลน์อยู่ก็ตาม และสามารถชิงโครไนซ์ข้อมูลของผู้ใช้งานได้ โดยไม่ต้องห่วงว่าผู้ใช้งาน จะใช้อุปกรณ์ตัวใดอยู่

สิ่งที่นักพัฒนาต้องทำ คือ การสร้างแอปพลิเคชันที่ดี โดยไม่ต้องกังวลเกี่ยวกับการสร้างการรักษาความปลอดภัย การจัดการผู้ใช้งาน การตรวจสอบความถูกต้อง



รูปที่ 2.13 แสดงการประยุกต์ใช้งาน Amazon Cognito¹⁵

2.2.8 Amazon DynamoDB

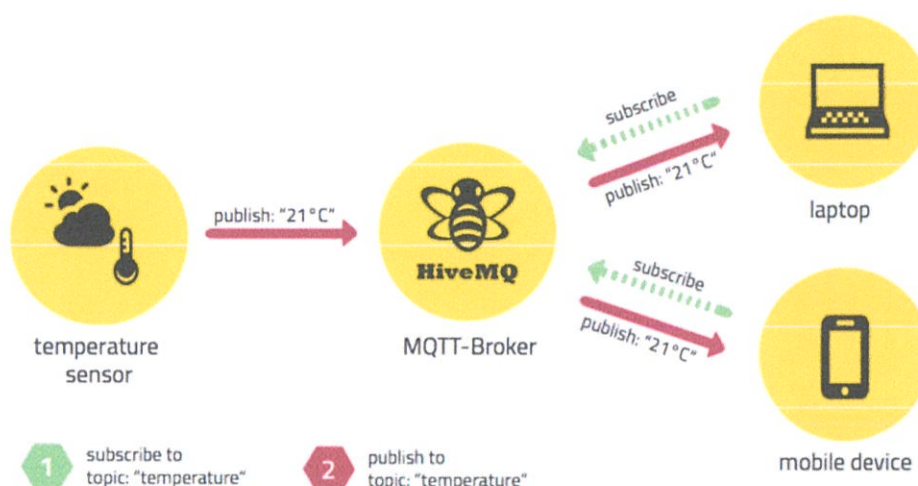
Amazon DynamoDB เป็นบริการฐานข้อมูลแบบ NoSQL ที่มีความรวดเร็ว ยืดหยุ่น และมีประสิทธิภาพสูง เหมาะกับแอปพลิเคชันที่ต้องการความแน่นอน และมีความล่าช้าเพียงไม่กี่มิลลิวินาที สามารถเก็บข้อมูลได้ทั้งแบบ Document และ Key-Value นอกจากนี้ ยังมี Data model ที่ยืดหยุ่น มีความน่าเชื่อถือสูง และมีการขยายระบบได้อัตโนมัติ ทั้งในเรื่อง Throughput และความจุ ทำให้เหมาะสมกับการนำไปใช้กับแอปพลิเคชันบนโทรศัพท์มือถือ บนเว็บไซต์ เกม หรืออุปกรณ์ IoT

¹⁵ Amazon Cognito – Simple and Secure User Sign Up & Sign In [ออนไลน์] <https://aws.amazon.com/api-gateway/> (สืบค้นข้อมูล : ตุลาคม 2560)

2.3 MQTT

MQTT (Message Queuing Telemetry Transport) เป็นมาตรฐานการรับส่งข้อความแบบมาตรฐาน ISO (ISO / IEC PRF 20922) ที่ทำงานบนโปรโตคอล TCP / IP และเป็นโปรโตคอลสำหรับการสื่อสารแบบ M2M (Machine-to-Machine) ซึ่ง M2M (Machine to Machine) นั้นคือเทคโนโลยีที่ทำให้อุปกรณ์สามารถทำการติดต่อสื่อสารระหว่างอุปกรณ์และอุปกรณ์ได้ โดยกลไกการทำงานของ M2M นั้นเริ่มจากเครื่องที่มีตัวเซ็นเซอร์ หรือมิเตอร์ที่ใช้วัดค่าบางอย่าง ซึ่งข้อมูลจะถูกส่งผ่านเครือข่ายไปยังแอปพลิเคชัน ซึ่งทำหน้าที่แปลงข้อมูลดิบให้เป็นข้อมูลที่มีความหมายและใช้ในการตัดสินใจได้

2.3.1 MQTT – Architecture



รูปที่ 2.14 รูปแสดงสถาปัตยกรรมของ MQTT¹⁶

โครงสร้างการทำงานของ MQTT - Architecture มีขั้นตอนดังนี้

- 1) ผู้ที่ต้องการรับข้อมูล หรือ subscriber จะทำการขอรับข้อมูล Topic: "Temperature"
- 2) ผู้ที่ต้องการส่งข้อมูล หรือ publisher จะทำการขอส่งข้อมูล Topic: "Temperature"

โดยในขั้นตอนที่ 1 และ 2 นี้ จำเป็นจะต้องมีตัวกลางในการรับ-ส่งข้อมูล ซึ่งเรียกว่า MQTT-Broker ทำหน้าที่เป็นตัวกลาง จัดการคิว รับ - ส่ง ข้อมูลระหว่างอุปกรณ์

¹⁶ MQTT 101 – How to Get Started with the lightweight IoT Protocol [ออนไลน์] <https://www.hivemq.com/blog/how-to-get-started-with-mqtt> (สืบค้นข้อมูล : ตุลาคม 2560)

2.3.2 องค์ประกอบของ MQTT Protocol

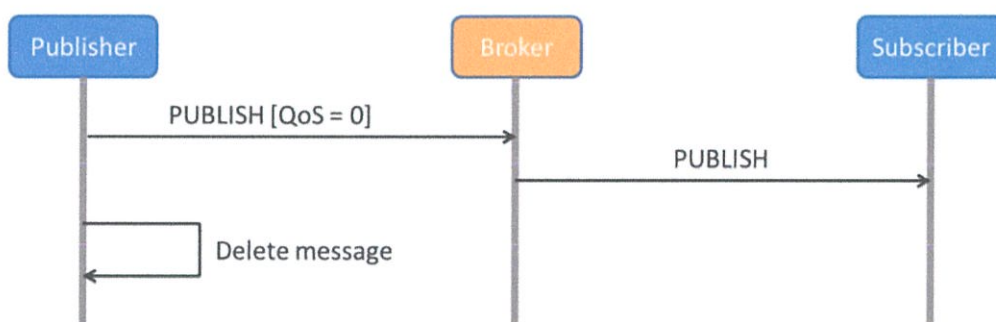
จะประกอบไปด้วย Broker, Publisher และ Subscriber แต่ละอย่างก็จะทำหน้าที่แตกต่างกันออกไปดังนี้

- Broker ทำหน้าที่เป็นตัวกลางจัดการกับข้อความ โดยอ้างอิงจาก Topic
- Publisher จะทำหน้าที่ส่งข้อมูลไปยังหัวข้อ Topic ที่ต้องการ
- Subscriber จะทำหน้าที่ดูแลการเปลี่ยนแปลงของ message ที่อ้างอิงด้วย Topic เช่น ถ้ามีหัวข้อที่น่าสนใจและมีการเปลี่ยนแปลงก็จะทำการดึงข้อมูลนั้นมาใช้งาน

2.3.3 MQTT - QoS

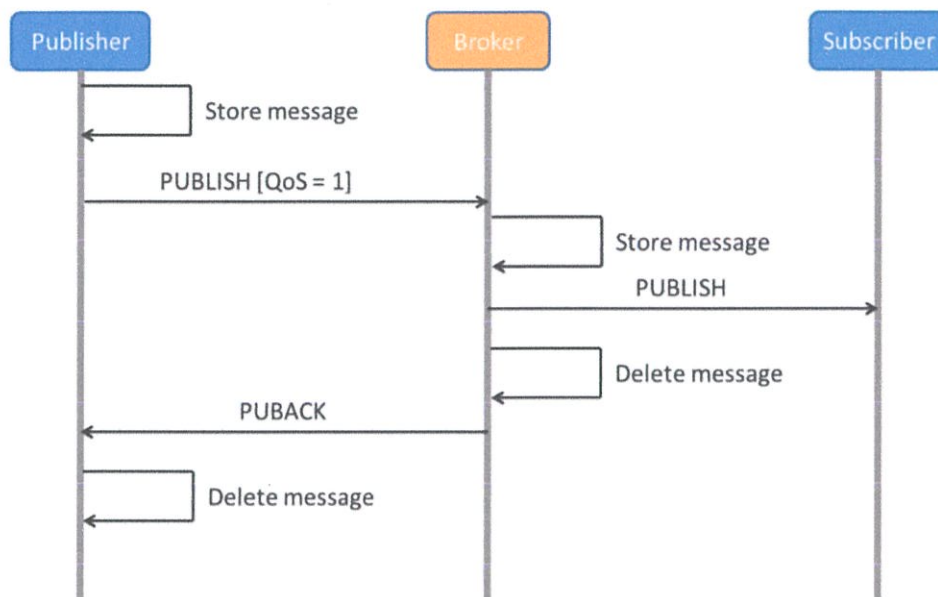
MQTT มีระดับของการให้บริการ 3 ระดับ ดังนี้

QoS 0: At most once หรืออย่างมาก 1 ครั้ง เป็นระดับการให้บริการที่ต่ำที่สุด คือ ไม่มีการรับประกันว่าข้อความจะส่งถึงผู้รับ เมื่อ Publisher ส่งข้อความด้วย



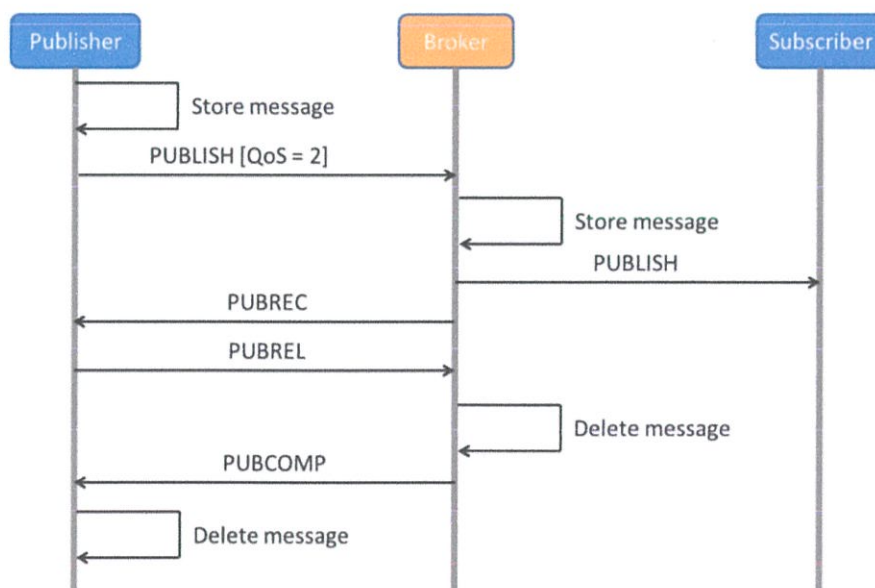
รูปที่ 2.15 รูปแสดงการทำงานของ QoS ระดับ 0

QoS 1: Exactly once หรือ 1 ครั้งเท่านั้น การส่งถึงขณะนี้ Publisher จะเก็บข้อความเอาไว้ จนกว่าจะได้รับ PUBACK จากผู้รับ



รูปที่ 2.16 รูปแสดงการทำงานของ QoS ระดับ 1

QoS 2: At least once หรืออย่างน้อย 1 ครั้ง จะรับประกันว่าข้อความจะส่งถึงผู้รับ 1 ครั้ง เท่านั้น



รูปที่ 2.17 รูปแสดงการทำงานของ QoS ระดับ 2

2.4 SC20-E

SC20-E series เป็นรุ่นใหม่ของ Quectel ที่มาพร้อมกับโมดูล Smart LTE Cat4 ที่มีหลายโหมดการทำงาน มาพร้อมกับระบบปฏิบัติการ Android ในตัว มีทั้งหมด 5 รุ่น คือ SC20-CE, SC20-E, SC20-A, SC20-AU และ SC20-J ซึ่ง SC20 series สามารถมีอัตราการดาวน์โหลดข้อมูลถึง 150Mbit/s และมีอัตราการอัปโหลดข้อมูลถึง 50Mbit/s ซึ่งเหมาะสำหรับการใช้งานในภาคอุตสาหกรรม และผู้ใช้งานที่ต้องการอัตราการส่งข้อมูลที่รวดเร็วและอินเทอร์เน็ตความเร็วสูง



รูปที่ 2.18 Chipset SC20 series¹⁷

SC-20 series เป็น LTE Smart Module รุ่นใหม่ของ Quectel ที่มาพร้อมกับโมดูล Smart LTE Cat4 ที่มีหลายโหมดการทำงาน โดยมีทั้งหมด 5 รุ่น คือ SC20-CE, SC20-E, SC20-A, SC20-AU และ SC20-J ซึ่ง SC20 series มีอัตราการดาวน์โหลดข้อมูลถึง 150Mbit/s และมีอัตราการอัปโหลดข้อมูลถึง 50Mbit/s เหมาะสำหรับการใช้งานในภาคอุตสาหกรรม และผู้ใช้งานที่ต้องการอัตราการส่งข้อมูลที่รวดเร็วและอินเทอร์เน็ตความเร็วสูง

SC20 ยังสามารถใช้งานร่วมกับ EDGE and GSM/GPRS เพื่อให้มั่นใจได้ว่าสามารถเชื่อมต่ออินเทอร์เน็ตได้แม้ในพื้นที่ห่างไกล ที่ไม่มี 3G หรือ LTE เข้าถึง นอกจากนี้ยังสามารถสื่อสารแบบไร้สายระยะใกล้ผ่าน IEEE 802.11 b / g / n (SC20-E / -A / -AU / -J รองรับ IEEE 802.11 a / b / g / n) และ BT4.1 LE ได้

¹⁷ LTE SC20 Smart Module [ออนไลน์] <http://www.quectel.com/product/sc20.htm> (สืบค้นข้อมูล : ตุลาคม 2560)

SC20 มีเทคโนโลยี Multiple-Input Multiple-Output (MIMO) ซึ่งเป็นเทคโนโลยีเสาอากาศที่ทันสมัยซึ่งสามารถส่งข้อมูลหลาย ๆ ข้อมูลไปยังเครื่องรับส่งสัญญาณหลายเครื่องได้ โมดูลยังรวมการเชื่อมต่อไร้สายความเร็วสูงเข้ากับ Multi-Constellation และเครื่องรับสัญญาณ GPS + GLONASS + BeiDou ที่มีความเร็วสูง



รูปที่ 2.19 แสดง Evaluation Board ของ Smart Module SC20-E

Evaluation Board ของ Smart Module SC20-E ดังรูปที่ 2.19 มาพร้อมชุดอินเตอร์เฟซที่หลากหลาย เช่น LCM, กล้องถ่ายภาพหน้า-หลัง, หน้าจอสัมผัส, ไมโครโฟน, ลำโพง, UART, USB และ I²C พร้อมไดรเวอร์ที่มีอยู่มากมาย ช่วยให้โมดูลสามารถใช้งานแอปพลิเคชัน M2M (Machine to Machine) ได้อย่างหลากหลาย เช่น CPE, Wireless POS, Smart metering, Router, Data Card, Automotive, Smart Phone, Digital Signage, Alarm Panel, Security และ Industry PDA

Specification

Frequency	SC20-E	SC20-A	SC20-J	SC20-AU	SC20-CE	SC20-W
LTE	FDD LTE	B1/ B3/ B5/ B7/ B8/ B20	B2/ B4/ B5/ B7/ B12/ B13/ B25/ B26	B1/ B3/ B8/ B18/ B19/ B26	B1/ B3/ B5/ B7/ B8/ B28	B1/ B3/ B6
	TDD LTE	B38/ B40/ B41		B41	B40	B38/ B39/ B40/ B41
3G	WCDMA	B1/ B5/ B8	B1/ B2/ B4/ B5/ B6	B1/ B6/ B8/ B19	B1/ B2/ B5/ B8	B1/ B6
	TD-SCDMA					B34/39
	EVDO/ CDMA2000 1x					BC0
GSM/EDGE	Quad Band	B2/ B5 (850/1900MHz)	B3/8	850/900/1800/1900	B3/ B8	
Embedded GNSS	Y	Y	Y	Y	Y	
Wi-Fi Interface	Y	Y	Y	Y	Y	Y
Region	EMEA, Korea, Thailand, India	America	Japan	Australia/Taiwan/ Brazil	China (CMCC, CUCC)	
Certification	CE/ GCF/ NCC/ KC/ SKT/ ICASA	FCC	TBD	TBD	CCC/ SRRC/ CTA	

รูปที่ 2.20 ภาพแสดงการเปรียบเทียบแต่ละรุ่นในตระกูล SC20 series¹⁸

¹⁸ LTE SC20 Smart Module [ออนไลน์] <http://www.quectel.com/product/sc20.htm> (สืบค้นข้อมูล : ตุลาคม 2560)

2.5 งานวิจัยที่เกี่ยวข้อง

2.5.1 Google Vision vs. Amazon Rekognition: A Vendor-neutral Comparison

งานวิจัยชิ้นนี้เป็นการเปรียบเทียบความสามารถในด้านต่าง ๆ ราคาของ Google Vision และ Amazon Rekognition

ในส่วนของ Amazon Rekognition ยังคงเป็นบริการใหม่อยู่ในด้านของ Artificial Intellegent แต่มีความสามารถ และราคาที่น่าสนใจ มีความสามารถในการตรวจจับความรู้สึก และสามารถวิเคราะห์ภาพได้ แม้ภาพจะถูกหมุนไป ซึ่งโดดเด่นกว่า Google มาก

นอกจากนี้ Amazon Rekognition ยังมาพร้อมกับความสามารถขั้นสูงต่าง ๆ เช่น การเปรียบเทียบใบหน้า การค้นหาใบหน้า เป็นต้น

ส่วน Google Vision นั้น มาพร้อมกับ API ที่มีความยืดหยุ่น รองรับไฟล์รูปภาพได้หลายฟอร์เมต ฟังก์ชันการตรวจจับวัตถุมีความแม่นยำ และผลลัพธ์ที่วิเคราะห์ออกมาได้ มีความเกี่ยวข้องกับวัตถุที่อยู่ในภาพมากกว่า ส่วนการค้นหาใบหน้าทำออกมาได้ดี แม้จะยังไม่สมบูรณ์เท่าไร

สิ่งที่ดูเหมือนจะเป็นปัญหาใน Google Vision คือการหมุนรูปภาพ แต่อาจจะได้รับการแก้ไขในอนาคต ส่วนการวิเคราะห์อารมณ์ ยังวิเคราะห์ได้ไม่หลากหลายนัก

บริการทั้งสองตัว ยังคงขาดความสามารถในการประมวลผลแบบ Batch และการประมวลผลวิดีโอ และคุณสมบัติอื่น ๆ เช่น การค้นหาวัตถุในรูปภาพ การติดตามวัตถุในวิดีโอ รวมไปถึงการวิเคราะห์รูปจากแหล่งภายนอก เช่น จาก URL เป็นต้น

บทที่ 3

การออกแบบและพัฒนา

3.1 ภาพรวมการทำงานภายในระบบ

3.1.1 Specification

3.1.1.1 Android

- 1) อุปกรณ์สามารถถ่ายภาพนิ่ง
- 2) อุปกรณ์สามารถส่งภาพนิ่งไปยัง Amazon Web Services
- 3) อุปกรณ์สามารถบันทึกภาพ ไปยังหน่วยความจำ
- 4) พัฒนาแอปพลิเคชัน บนระบบปฏิบัติการ Android เวอร์ชัน 6.1 บน Smart Module (SC20-E)

3.1.1.2 Amazon Web Services

- 1) สามารถข้อมูลภาพนิ่งไปเก็บไว้ยัง Amazon S3
- 2) สามารถวิเคราะห์ภาพถ่ายโดยใช้ Amazon Rekognition
- 3) สามารถจัดการ Remote camera โดยใช้ Amazon IoT ได้
- 4) สามารถทำเว็บแอปพลิเคชันได้
- 5) สามารถสร้างวิดีโอ Time-Lapse ได้
- 6) สามารถค้นหารูปภาพตามวัตถุที่อยู่ในภาพได้
- 7) สามารถลบรูปถ่ายที่ถ่ายมาแล้วได้

3.1.1.3 เว็บแอปพลิเคชัน

- 1) สามารถจัดการกล้องของแต่ละผู้ใช้งานได้
- 2) สามารถดูรูปภาพที่ถ่ายไว้ได้ และสามารถดูภาพถ่ายแยกในแต่ละวันได้
- 3) สามารถตั้งค่ากล้องถ่ายภาพระยะไกลได้
- 4) สามารถตั้งค่าการแจ้งเตือนเมื่อพบวัตถุได้
- 5) สามารถดาวน์โหลดภาพถ่ายจาก AWS ได้
- 6) สามารถทำวิดีโอ Time-Lapse ในขนาดต่าง ๆ ได้

3.2 กล้องตรวจตราระยะไกล

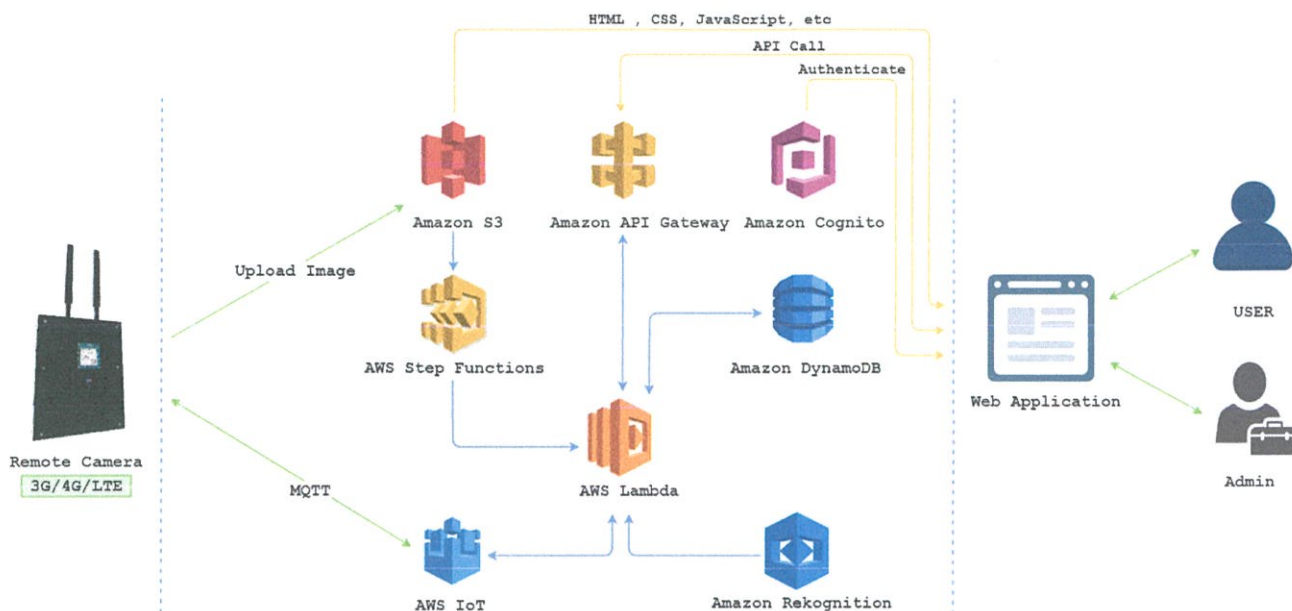
ตัวกล้องมีขนาด กว้าง x ยาว x สูง อยู่ที่ 225 x 200 x 32.6 มิลลิเมตร (ไม่รวมเสาสัญญาณ) วัสดุทำมาจากอะคริลิกสีดำเงา ด้านหลังมีช่องสำหรับยึดกับขาตั้งกล้องขนาดมาตรฐานได้

เซ็นเซอร์รับภาพที่ใช้ มีขนาด 8 ล้านพิกเซลล์ ขนาดภาพสูงสุด 3264 x 2448 พิกเซลล์ เสาสัญญาณมีทั้งหมด 3 เสา โดยเป็นเสาสำหรับสัญญาณมือถือ 2 เสา ติดตั้งอยู่ภายนอก และเสาสำหรับ Wi-Fi ซ่อนอยู่ในตัวกล้อง



รูปที่ 3.1 กล้องตรวจตรา

3.3 โครงสร้างของระบบ



รูปที่ 3.2 System Diagram

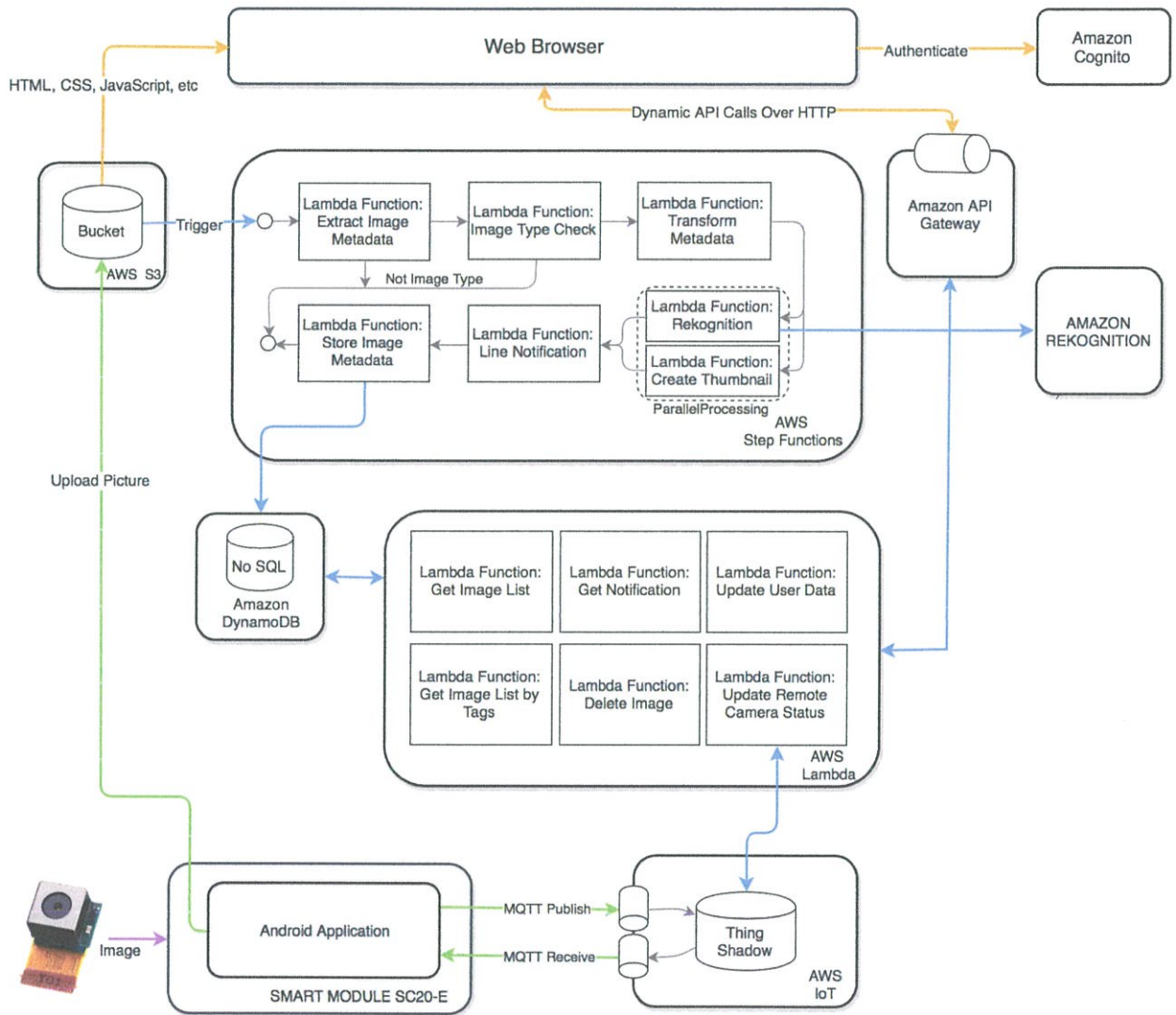
จากรูปที่ 3.2 ระบบแบ่งออกเป็น 3 ส่วน คือ กล้องถ่ายภาพตรวจตรา Amazon Web Services และเว็บแอปพลิเคชัน

สำหรับ Amazon Web Services จะมี Service ที่เกี่ยวข้อง 8 ตัว

1. Amazon S3 ใช้สำหรับเก็บรูปภาพที่กล้องตรวจตราถ่ายได้ และไฟล์ html, CSS และ JavaScript สำหรับการทำเว็บแอปพลิเคชัน
2. Amazon API Gateway ใช้สำหรับการทำ API โดยตัว Service จะเป็นตัวเดียวกับ API ไหน ควรใช้ Lambda Function ตัวไหนในการทำงาน รวมถึงการกรอง Request ที่ไม่เกี่ยวข้องออก และป้องกันไม่ให้บุคคลที่ไม่ได้รับอนุญาต (ไม่มี Token) เข้าใช้งาน
3. Amazon Cognito ใช้สำหรับการจัดการระบบสมาชิก ทั้งการสมัครสมาชิก การยืนยันตัวตน การเปลี่ยนรหัสผ่าน และการขอ Token เพื่อเข้าใช้งานระบบ
4. AWS Step Functions ใช้สำหรับจัดการระบบการทำงานของ Lambda Function ให้มีลำดับการทำงานคล้าย Flowchart โดยจะใช้ในขั้นตอนของการรับรูปภาพจาก Remote Camera และประมวลผล
5. AWS Lambda ใช้สำหรับประมวลผลข้อมูลแบบสั้นๆ ใช้สำหรับการประมวลผลข้อมูลรูปภาพ และประมวลผล Request จาก Web Application
6. Amazon DynamoDB เป็นฐานข้อมูลสำหรับเก็บผลการวิเคราะห์รูปภาพ และข้อมูลส่วนตัวของผู้ใช้งาน

7. AWS IoT ใช้เป็นสื่อกลางในการควบคุมกล้องตรวจตรา ทั้งการปรับขนาดรูปภาพ การตั้งเวลาถ่ายภาพ และการรายงานสถานะของกล้องตรวจตรา
8. Amazon Rekognition ใช้สำหรับการจำแนกวัตถุภายในรูปภาพ

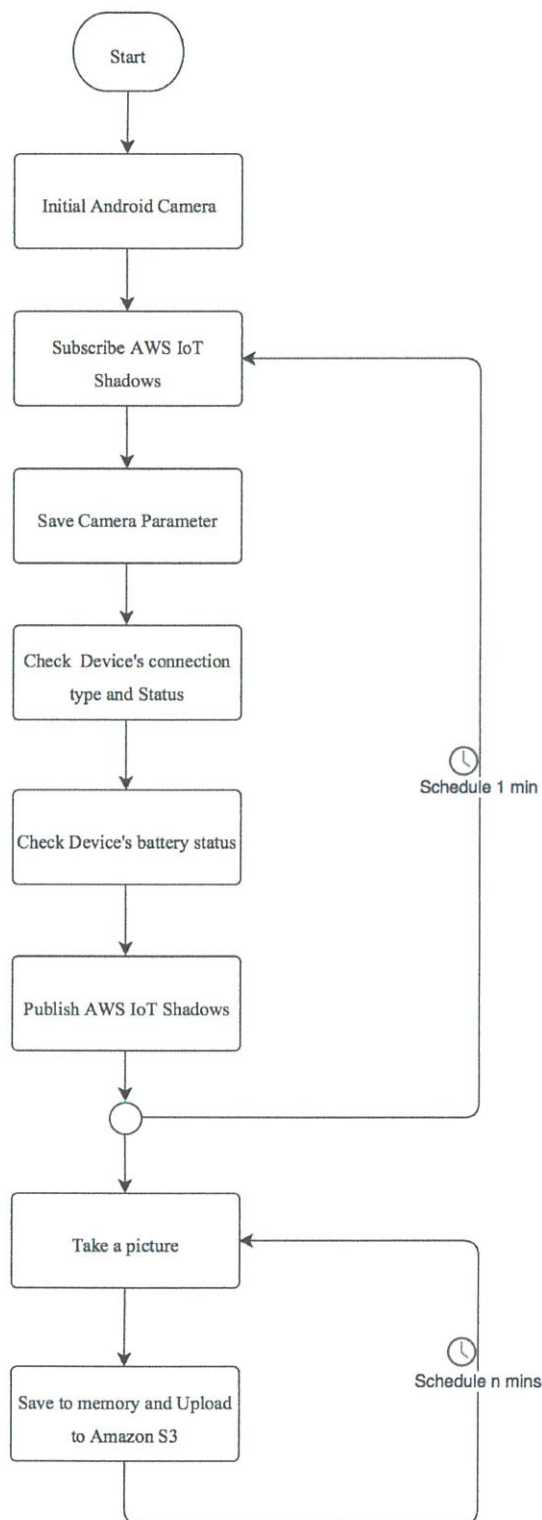
3.4 Architecture Diagram



รูปที่ 3.3 Architecture Diagram

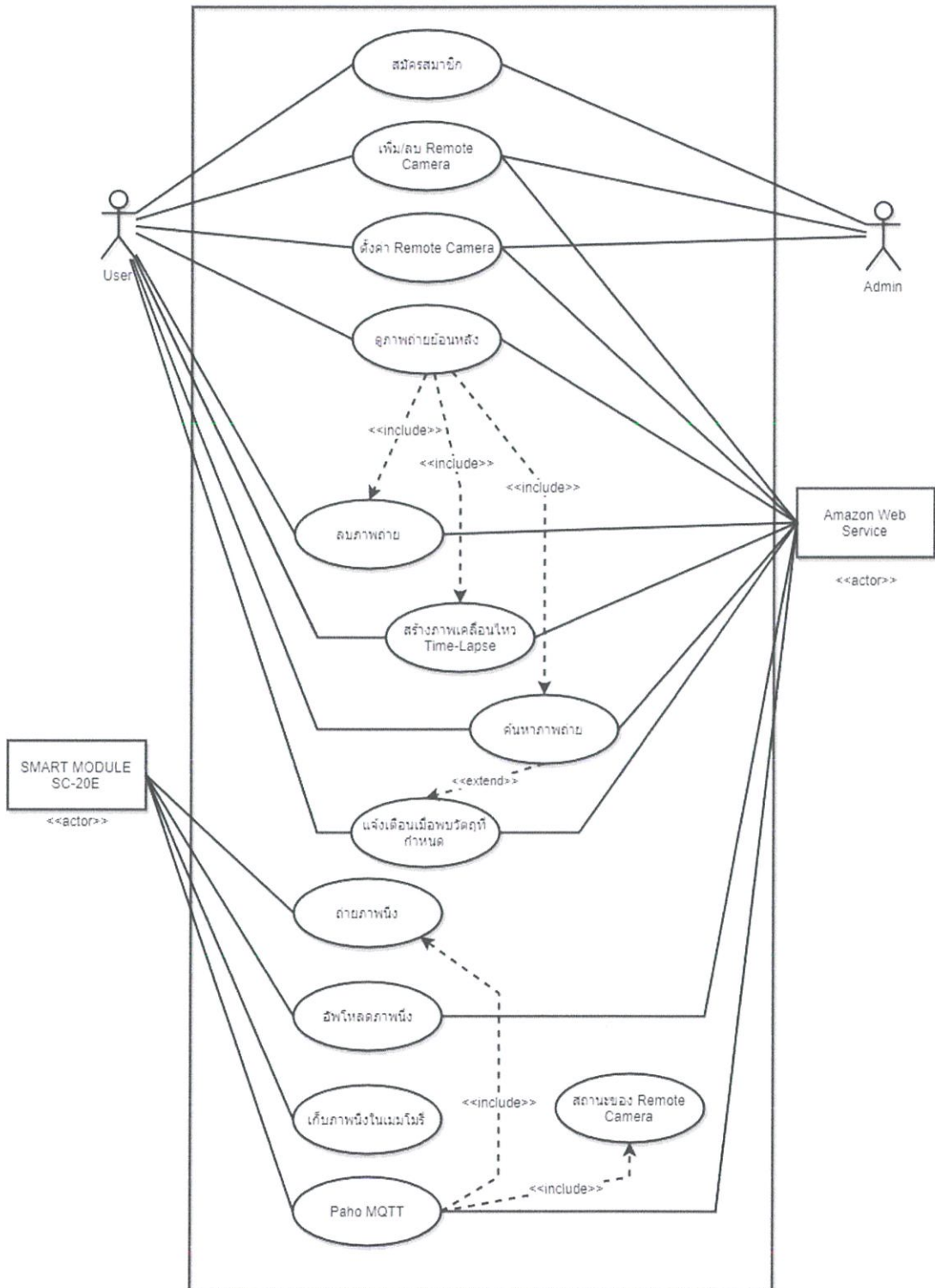
จากรูปที่ 3.3 แสดงถึงความสัมพันธ์ของ Lambda function กับ Service ต่าง ๆ ของ AWS

3.5 Flowchart ของกล้องตรวจตรา



รูปที่ 3.4 Flowchart ของกล้องตรวจตรา

3.6 Use case Diagram



รูปที่ 3.5 Use case Diagram

ตารางที่ 3.1 Use case สมัครสมาชิก

Use Case Name	สมัครสมาชิก
Use Case Purpose	User และ Admin สามารถสมัครสมาชิกได้
Pre-conditions	-
Post-conditions	User และ Admin สามารถสมัครสมาชิกได้สำเร็จ
Limitations	-
Primary Scenario	สมัครสมาชิก 1) เปิด เว็บแอปพลิเคชัน 2) กดปุ่ม "Sign Up" 3) กรอกข้อมูลในแบบฟอร์มที่กำหนด 4) ตรวจสอบข้อมูลที่กรอกเข้ามา 5) หากข้อมูลถูกต้อง 6) บันทึกข้อมูล
Alternative Scenario	หากข้อมูลไม่ถูกต้อง ให้แจ้งเตือน แล้วทำการกรอกข้อมูลใหม่

ตารางที่ 3.2 Use case อัปโหลดภาพนิ่ง

Use Case Name	อัปโหลดภาพนิ่ง
Use Case Purpose	Device สามารถอัปโหลดภาพนิ่งได้
Pre-conditions	Device ต้องเชื่อมต่อไปยัง AWS ได้
Post-conditions	Device สามารถอัปโหลดภาพนิ่ง ไปยัง Amazon Web Service ได้สำเร็จ
Limitations	-
Primary Scenario	อัปโหลดภาพนิ่ง 1) เชื่อมต่อไปยัง Amazon S3 2) Upload รูปภาพพร้อม Metadata 3) รอรับสถานะการอัปโหลด
Alternative Scenario	ถ้าอัปโหลดไม่สำเร็จ เลือกอัปโหลดใหม่อีกครั้ง

ตารางที่ 3.3 Use case เพิ่ม/ลบ Remote Camera

Use Case Name	เพิ่ม/ลบ อุปกรณ์ถ่ายภาพ
Use Case Purpose	User และ Admin สามารถเพิ่ม/ลบ อุปกรณ์ถ่ายภาพได้
Pre-conditions	User และ Admin เป็นสมาชิกของระบบ
Post-conditions	User และ Admin สามารถเพิ่ม/ลบ อุปกรณ์ถ่ายภาพได้สำเร็จ
Limitations	-
Primary Scenario	<p>เพิ่ม/ลบ อุปกรณ์ถ่ายภาพ</p> <p>เพิ่มอุปกรณ์</p> <ol style="list-style-type: none"> 1) กดปุ่ม "+" 2) กรอกเลขประจำตัวอุปกรณ์ 3) ตั้งชื่ออุปกรณ์ <p>ลบอุปกรณ์</p> <ol style="list-style-type: none"> 1) กดปุ่ม "-" 2) เลือกชื่ออุปกรณ์ที่ต้องการลบ 3) ลบอุปกรณ์
Alternative Scenario	-

ตารางที่ 3.4 ดูภาพถ่ายย้อนหลัง

Use Case Name	ดูภาพถ่ายย้อนหลัง
Use Case Purpose	User และ AWS สามารถดูภาพถ่ายย้อนหลังได้
Pre-conditions	User และ AWS สามารถเข้าถึงรูปภาพใน AWS S3 ได้
Post-conditions	User และ AWS สามารถดูภาพถ่ายย้อนหลังได้สำเร็จ
Limitations	-
Primary Scenario	<p>ดูภาพถ่ายย้อนหลัง</p> <ol style="list-style-type: none"> 1) เปิด เว็บแอปพลิเคชัน ขึ้นมา 2) คลิกที่ "รูปภาพ" 3) เลือก "ดูภาพถ่ายย้อนหลัง"
Alternative Scenario	-

ตารางที่ 3.5 Use case สร้างภาพเคลื่อนไหว Time-Lapse

Use Case Name	Time-Lapse
Use Case Purpose	User และ AWS สามารถใช้ระบบ Time-Lapse ได้
Pre-conditions	User และ AWS สามารถเข้าถึง เว็บแอปพลิเคชัน ได้
Post-conditions	User และ AWS สามารถใช้ระบบ Time-Lapse ได้สำเร็จ
Limitations	-
Primary Scenario	Time-Lapse 1) คลิกที่ปุ่ม Time-Lapse 2) เลือกช่วงเวลา (Period Scale) ที่ต้องการ 3) เลือกความละเอียดที่ต้องการ 4) เลือกความเร็วในการแสดงผลที่ต้องการ
Alternative Scenario	-

ตารางที่ 3.6 Use case ลบภาพถ่าย

Use Case Name	ลบภาพถ่าย
Use Case Purpose	User และ AWS สามารถลบภาพถ่ายจาก คูภาพถ่ายย้อนหลังได้
Pre-conditions	User และ AWS สามารถเข้าถึง คูภาพถ่ายย้อนหลังได้
Post-conditions	User และ AWS สามารถลบภาพถ่ายจาก คูภาพถ่ายย้อนหลังได้สำเร็จ
Limitations	-
Primary Scenario	ลบภาพถ่าย 1) เลือก รูปภาพที่ต้องการลบ 2) ลบ
Alternative Scenario	-

ตารางที่ 3.7 Use case ตั้งค่าอุปกรณ์ถ่ายภาพ

Use Case Name	ตั้งค่าอุปกรณ์ถ่ายภาพ
Use Case Purpose	User และ Admin สามารถตั้งค่าอุปกรณ์ถ่ายภาพได้
Pre-conditions	- User และ Admin สามารถเข้าถึงอุปกรณ์ถ่ายภาพได้ - ต้องมีอุปกรณ์ที่ถูกเพิ่มอยู่แล้ว
Post-conditions	User และ Admin สามารถตั้งค่าอุปกรณ์ถ่ายภาพได้สำเร็จ
Limitations	-
Primary Scenario	ตั้งค่าอุปกรณ์ถ่ายภาพ 1) เปิด เว็บแอปพลิเคชัน 2) ใสค่า Parameter ของอุปกรณ์ถ่ายภาพ 3) บันทึก
Alternative Scenario	-

ตารางที่ 3.8 Use case เก็บภาพนิ่งในเมมโมรี

Use Case Name	เก็บภาพนิ่งในเมมโมรี
Use Case Purpose	เก็บภาพนิ่งในเมมโมรีได้
Pre-conditions	Device สามารถเข้าถึงเมมโมรีได้
Post-conditions	เก็บภาพนิ่งในเมมโมรีได้สำเร็จ
Limitations	-
Primary Scenario	เก็บภาพนิ่งในเมมโมรี - บันทึกภาพไปยังเมมโมรี
Alternative Scenario	ถ้าบันทึกภาพนิ่งในเมมโมรีไม่สำเร็จ - ให้บันทึกลงความจำเครื่อง

ตารางที่ 3.9 Use case แจ้งเตือนเมื่อพบวัตถุที่ค้นพบ

Use Case Name	แจ้งเตือนเมื่อพบวัตถุที่ค้นพบ
Use Case Purpose	User และ AWS สามารถแจ้งเตือนเมื่อพบวัตถุที่ค้นพบได้
Pre-conditions	- User และ AWS สามารถเข้าถึงรูปภาพใน Amazon Web Service S3 - Amazon Rekognition ทำการตรวจพบวัตถุ
Post-conditions	User และ AWS สามารถแจ้งเตือนเมื่อพบวัตถุที่ค้นพบได้สำเร็จ ผ่าน Line Notify
Limitations	-
Primary Scenario	แจ้งเตือนเมื่อพบวัตถุที่ค้นพบ 1) เตรียมภาพถ่ายที่ต้องการให้ Amazon Rekognition ตรวจสอบ 2) Amazon Rekognition ตรวจสอบภาพที่ได้รับมา 3) ทำการบันทึกผลลัพธ์การวิเคราะห์ลงบน Metadata ของรูปภาพ 4) Lambda ตรวจสอบ Metadata ของรูปภาพ 5) หากเจอวัตถุที่กำหนดให้ไปเรียกการแจ้งเตือนไปยังผู้ใช้ ผ่าน Line Notify
Alternative Scenario	-

ตารางที่ 3.10 Use case ค้นหาภาพถ่าย

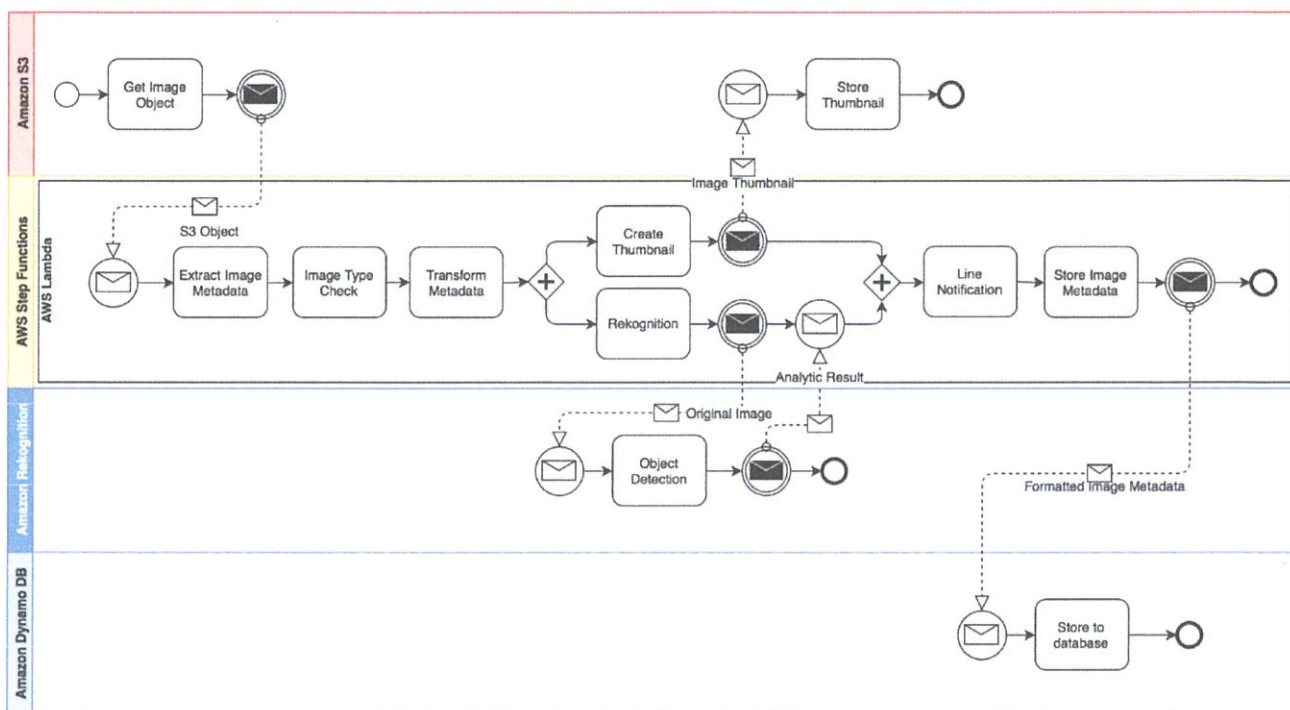
Use Case Name	ค้นหาภาพถ่าย
Use Case Purpose	User และ AWS ต้องสามารถค้นหาภาพถ่ายได้
Pre-conditions	User และ AWS สามารถเข้าถึงรูปภาพใน AMS S3
Post-conditions	User และ AWS ต้องสามารถค้นหาภาพถ่ายได้
Limitations	-
Primary Scenario	ค้นหาภาพถ่าย 1) เปิด เว็บแอปพลิเคชัน ขึ้นมา 2) เข้าไปที่ ภาพถ่ายทั้งหมด 3) เลือก "ค้นหาภาพถ่าย" 4) เลือกช่วงเวลาและใส่ Keyword ของรูปภาพที่ต้องการหา 5) คลิก "ตกลง"
Alternative Scenario	-

ตารางที่ 3.11 Use case ถ่ายภาพนิ่ง

Use Case Name	ถ่ายภาพนิ่ง
Use Case Purpose	Device สามารถถ่ายภาพนิ่งได้
Pre-conditions	- ถ่ายภาพนิ่งเมื่อถึงเวลาที่ต้องถ่ายภาพ - Device สามารถเข้าถึงอุปกรณ์ถ่ายภาพ
Post-conditions	Device สามารถถ่ายภาพนิ่งได้สำเร็จ
Limitations	-
Primary Scenario	ถ่ายภาพนิ่ง 1) เริ่มทำการ Auto Focus แล้ว 2) Lock Focus 3) ทำการ Capture รูปภาพ
Alternative Scenario	-

3.7 Business Process Model and Notation (BPMN)

3.7.1 กล้องตรวจตราส่งภาพมายัง AWS



รูปที่ 3.6 เมื่อ Amazon S3 ได้รับรูปภาพ และสั่งให้ Amazon Step Functions ทำงาน¹⁹

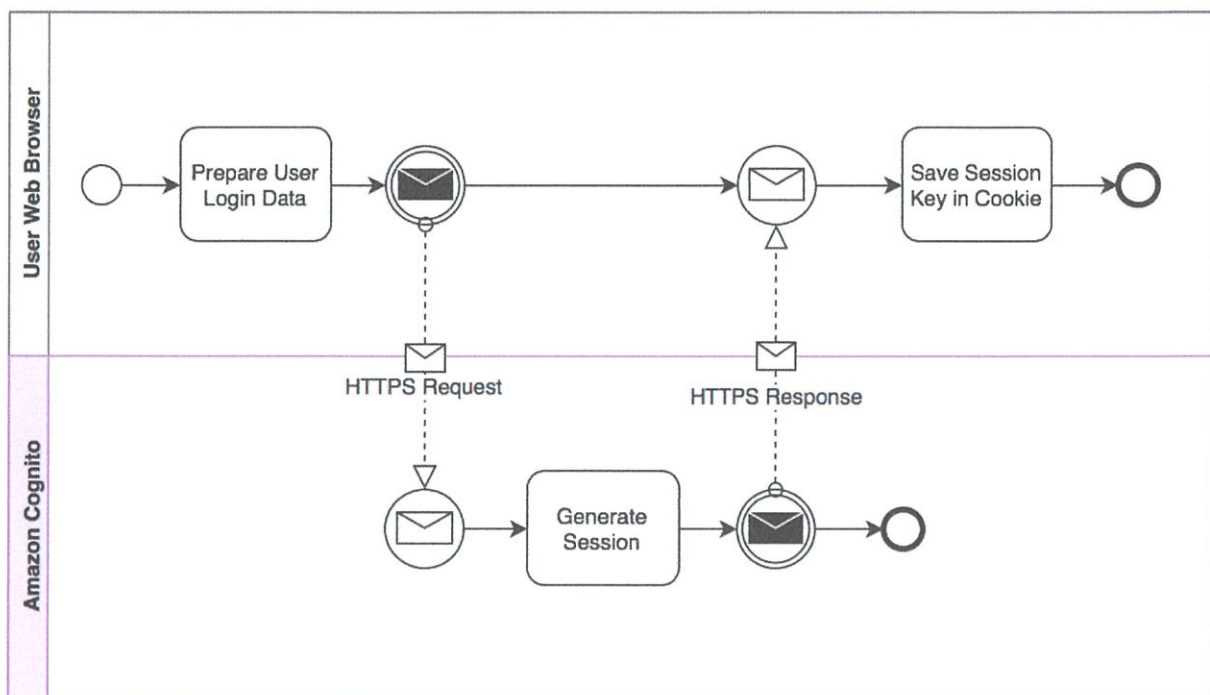
จากรูปที่ 3.6 เมื่อภาพถูกส่งมายัง Amazon S3 ตัว Service จะทำการ Trigger AWS Step Functions เพื่อเรียกใช้งาน AWS Lambda โดยขั้นแรก รูปภาพจะถูกดึงเอา Metadata ทั้งหมดออกมาก่อนด้วยฟังก์ชัน Extract Metadata แล้วส่งไปยังฟังก์ชัน Image Type Check เพื่อตรวจสอบว่า Metadata นั้น เป็นไฟล์รูปภาพใช่หรือไม่ จากนั้น จะส่งไปยัง Transform Metadata เพื่อเลือกเอาข้อมูลที่ต้องการใช้ออกมาจาก Metadata

ต่อมา รูปภาพจะถูกย่อขนาด เพื่อสร้าง Thumbnail แล้วเก็บผลลัพธ์ลงบน Amazon S3 พร้อมวิเคราะห์หาวัตถุในภาพโดยการใช้ Amazon Rekognition ซึ่งขั้นตอนเหล่านี้ จะทำในรูปแบบขนานกันไป และเมื่อทำเสร็จแล้ว จะส่งผลลัพธ์ไปยังฟังก์ชันต่อไป ที่ชื่อ Line Notification โดยฟังก์ชันนี้ จะดูแลการวิเคราะห์หาวัตถุ และเทียบว่าตรงกับที่ระบุไว้หรือไม่ ถ้าตรง ก็จะทำการแจ้งเตือนผ่าน Line Notify ทันที

ในขั้นตอนสุดท้าย ผลลัพธ์ทั้งหมดจะถูกเก็บลงใน Amazon DynamoDB

¹⁹ AWS Samples. Serverless Reference Architecture: Image Recognition and Processing Backend [ออนไลน์] <https://github.com/aws-samples/lambda-refarch-imagerecognition> (สืบค้นข้อมูล : มีนาคม 2561)

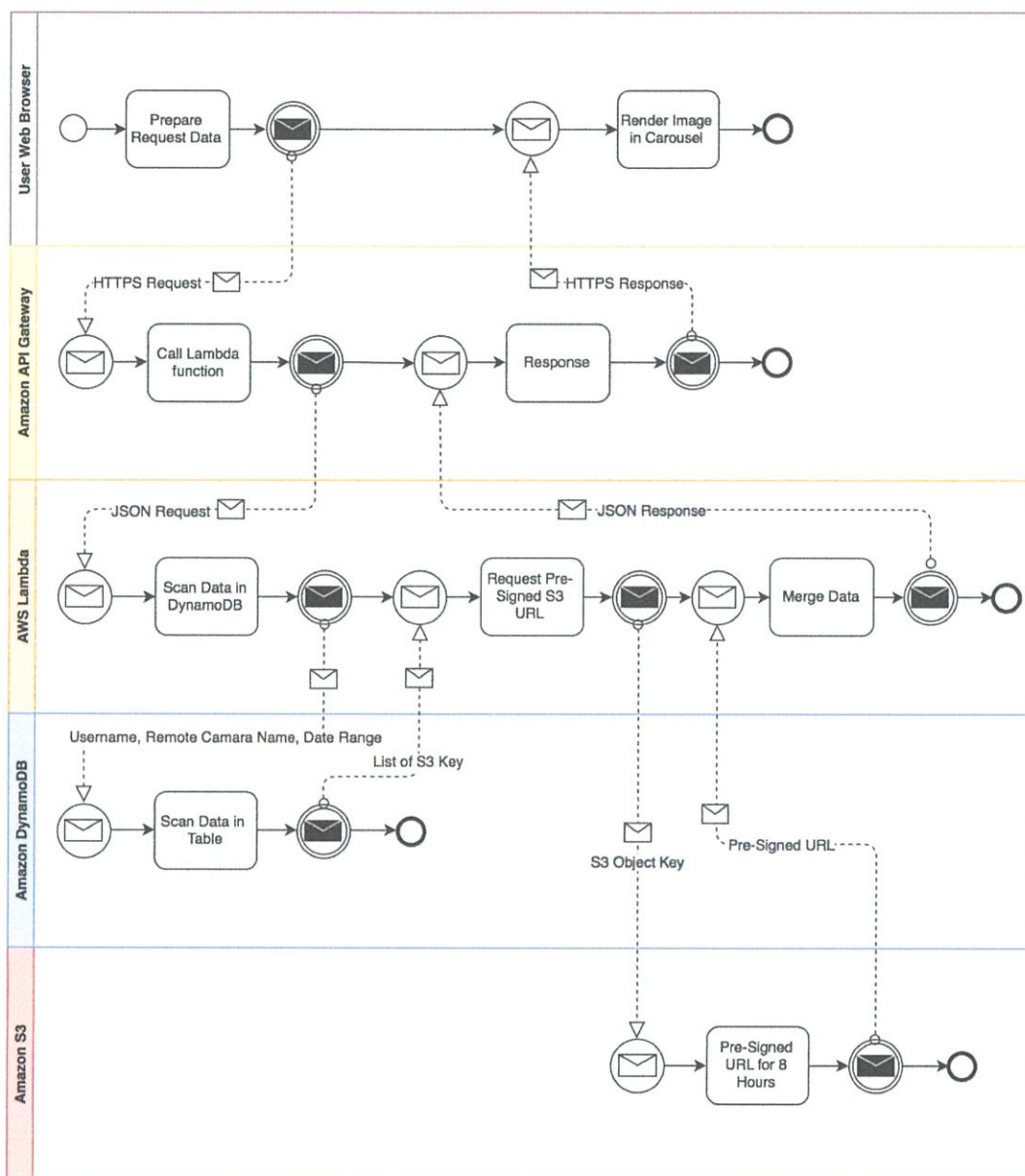
3.7.2 การ Login เพื่อเข้าสู่ Web Application



รูปที่ 3.7 การ Login เข้าสู่ระบบ

จากรูปที่ 3.7 เมื่อผู้ใช้งานทำการเข้าสู่ระบบ ผ่านทางเว็บแอปพลิเคชัน ข้อมูลจะถูกส่งไปยัง Amazon Cognito เพื่อขอ Session key ในรูปแบบ JWT และส่งกลับมายังเว็บแอปพลิเคชัน

3.7.3 การแสดงรูปภาพทั้งหมดที่ Remote Camera ถ่ายได้ผ่านเว็บ แอปพลิเคชัน

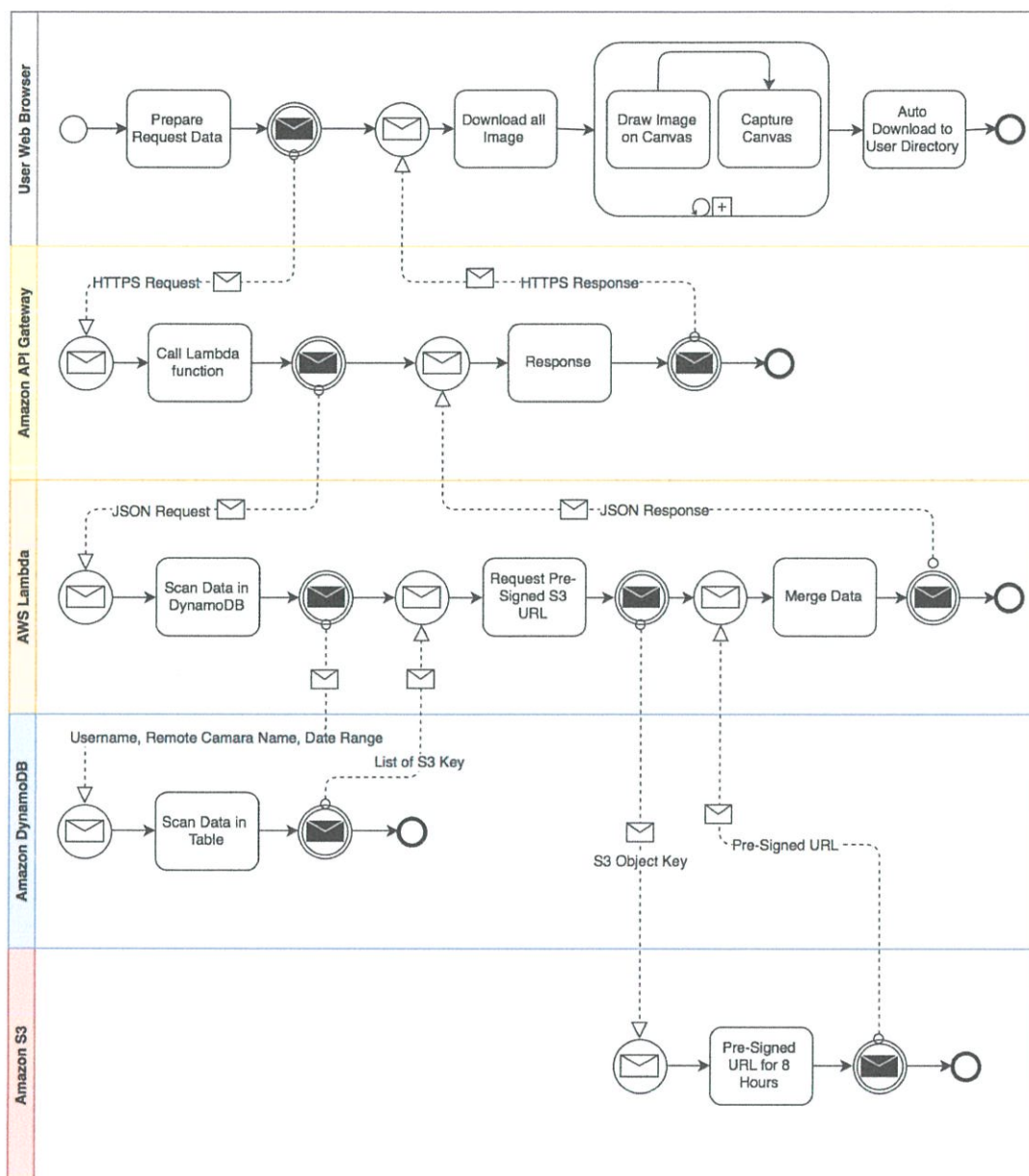


รูปที่ 3.8 การแสดงรูปภาพทั้งหมดที่ Remote Camera ถ่ายได้ผ่านเว็บแอปพลิเคชัน

จากรูปที่ 3.8 เมื่อผู้ใช้งานต้องการดูรูปภาพ เว็บแอปพลิเคชัน จะส่งข้อมูล ประกอบไปด้วย JWT Token และช่วงเวลาของภาพ ไปที่ Amazon API Gateway และ Service นี้ จะทำการตรวจสอบว่า Token นั้น ยังใช้งานได้อยู่หรือไม่ หากไม่ ก็จะตอบกลับด้วย Status 401 ทันที

หาก Token ยังสามารถใช้งานได้ AWS API Gateway จะทำการเรียก Lambda function เพื่อค้นหาข้อมูลใน Amazon DynamoDB โดยจะหารูปภาพที่อยู่ในช่วงเวลานั้น พร้อมผลการวิเคราะห์รูปภาพ และเมื่อได้รายชื่อรูปแล้ว จะทำการ pre-signed รูปภาพเหล่านั้น เพื่ออนุญาตให้ผู้ใช้งานสามารถเข้ามาดูรูปภาพได้ ในเวลาที่กำหนดไว้ จากนั้น จะส่งผลลัพธ์กลับไป เพื่อแสดงผล

3.7.4 การสร้างภาพ Time-Lapse จากช่วงเวลาของภาพถ่ายที่เลือกผ่านเว็บแอปพลิเคชัน

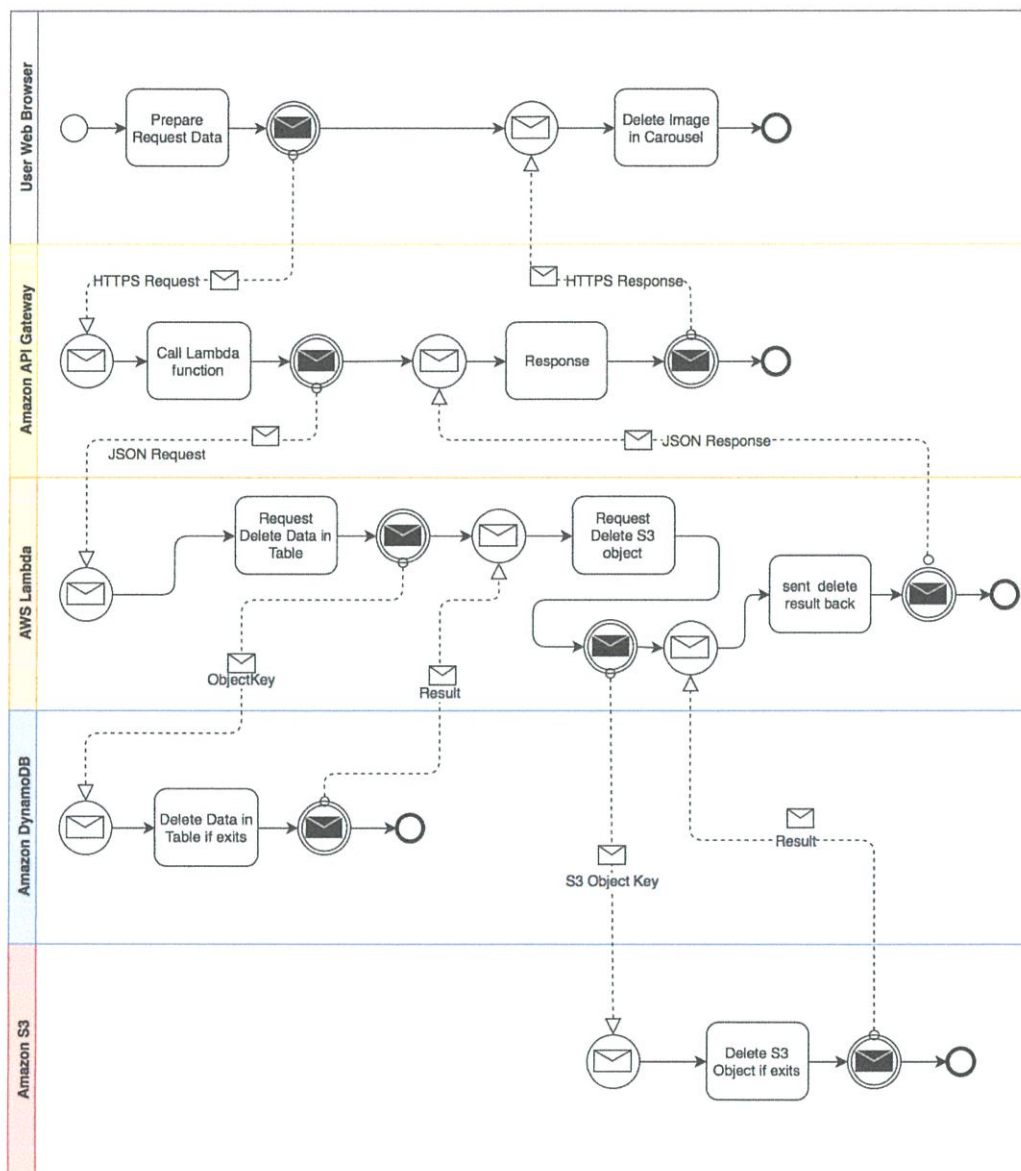


รูปที่ 3.9 การสร้างภาพ Time-Lapse จากช่วงเวลาของภาพถ่ายที่เลือกผ่านเว็บแอปพลิเคชัน

จากรูปที่ 3.9 การสร้างวิดีโอแบบ Time-Lapse จะทำการส่งคำสั่งแบบเดียวกันกับหัวข้อที่ 3.7.3 ต่างกันที่ตอนสุดท้าย ภาพที่ได้จะถูกวาดลงบน Canvas ที่ถูกซ่อนไว้บนเว็บแอปพลิเคชันของผู้ใช้งาน จะบันทึก Canvas นั้น โดยใช้ CCanvas.js²⁰ เข้ามาช่วยในการบันทึกภาพทีละภาพ และบันทึกออกมาเป็นไฟล์เมื่อวาดภาพลงบน Canvas ครบทุกภาพแล้ว

²⁰ Jaime Sanchez. CCanvas.js [ออนไลน์] <https://github.com/spite/ccapture.js> (สืบค้นข้อมูล : มีนาคม 2561)

3.7.5 การลบรูปภาพที่กำหนดผ่านเว็บแอปพลิเคชัน



รูปที่ 3.10 การลบรูปภาพที่กำหนดผ่านเว็บแอปพลิเคชัน

จากรูปที่ 3.10 เมื่อผู้ใช้งานต้องการลบรูปภาพเว็บแอปพลิเคชัน จะส่งข้อมูล ประกอบไปด้วย JWT Token ชื่อของกล้องที่ถ่าย และชื่อของภาพ ไปที่ Amazon API Gateway และ Service นี้ จะทำการตรวจสอบว่า Token นั้น ยังใช้งานได้หรือไม่ หากไม่ ก็จะตอบกลับด้วย Status 401ทันที

หาก Token ยังสามารถใช้งานได้ AWS API Gateway จะทำการเรียก Lambda function เพื่อค้นหาข้อมูลใน Amazon DynamoDB โดยจะหาว่ามีรูปภาพที่ชื่อของกล้อง และชื่อของภาพตรงกับข้อมูลที่มีอยู่ใน Amazon DynamoDB หากมี ก็จะทำการลบทิ้ง และลบข้อมูลภาพใน Amazon S3 ด้วย หากสำเร็จ จะส่งข้อมูลกลับมายังเว็บแอปพลิเคชัน เพื่อแสดงผลให้ผู้ใช้งานทราบ

บทที่ 4

ผลการทดลอง

การทดลองในบทนี้จะเป็นการทดลองใช้ Smart Module SC20-E ในการถ่ายภาพ และทดลองบริการของ Amazon Web Service บางรายการ

- 1) การทดสอบคุณภาพของภาพที่ถ่ายได้จากกล้องตรวจตรา
- 2) การทดสอบการดูภาพบน เว็บแอปพลิเคชัน
- 3) การทดสอบการแจ้งเตือนเมื่อเจอวัตถุ หรือสถานการณ์ที่ระบุไว้
- 4) การทดสอบการสร้างวิดีโอแบบ Time-Lapse
- 5) การคำนวณค่าใช้จ่ายของตัวอุปกรณ์ และค่าบริการของ AWS

4.1 การทดสอบคุณภาพของภาพที่ถ่ายได้จากกล้องตรวจตรา

4.1.1 วัตถุประสงค์

เพื่อทดสอบคุณภาพของภาพถ่ายที่ได้จาก Smart Module ในสภาพแสงต่าง ๆ กัน

4.1.2 วิธีการทดลอง

ตั้งกล้องถ่ายภาพตรวจตรา โดยใช้ขาตั้งกล้องยึดตัวกล้องไว้ ดังรูปที่ 4.1 และปล่อยให้กล้องถ่ายภาพไปตลอดทั้งวัน



รูปที่ 4.1 บรรยากาศขณะทำการทดสอบคุณภาพของภาพถ่ายที่ได้จากกล้องตรวจตรา

4.1.3 ผลการทดลอง

ภาพที่ถ่ายได้เมื่อตอนมีแสงมาก จากรูปที่ 4.2 จะเห็นได้ว่าภาพที่มีความคมชัด มีสีสดใส แต่ยังสามารถพบเห็น Artifacts ที่เกิดจากการบีบอัดรูปภาพได้



รูปที่ 4.2 แสดงภาพที่ถ่ายได้จาก Smart Module SC 20-E ถ่ายช่วงเวลา 12 นาฬิกา 23 นาที

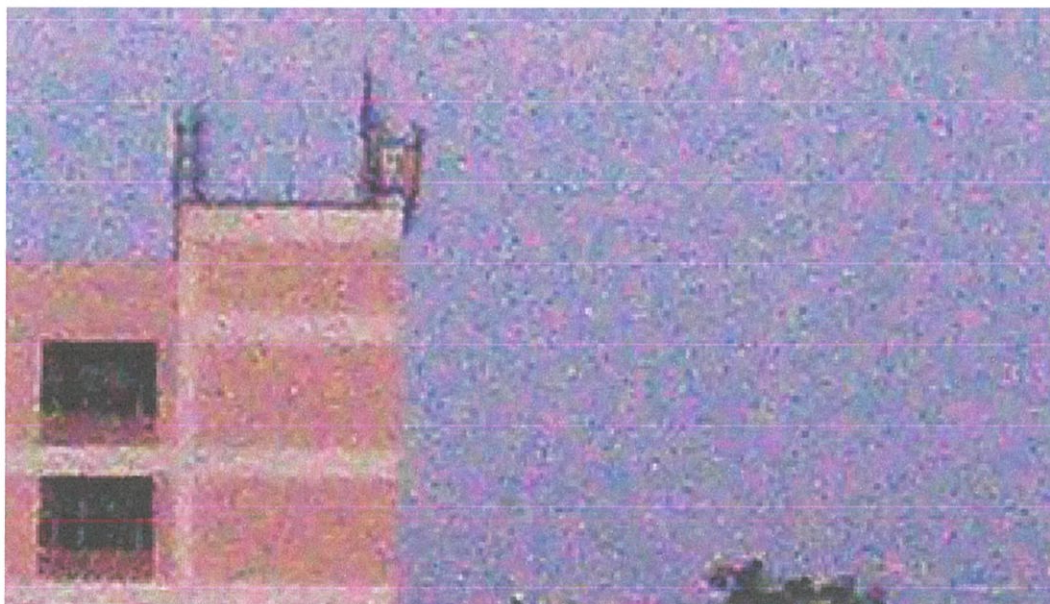


รูปที่ 4.3 เมื่อขยายภาพที่ 4.2 ที่ 500%

ภาพที่ถ่ายได้เมื่อตอนมีแสงน้อย จากรูปที่ 4.4 จะเห็นได้ว่าภาพที่เริ่มมืด และมี Noise รบกวนในระดับปานกลาง แต่ยังมองเห็นรายละเอียดได้

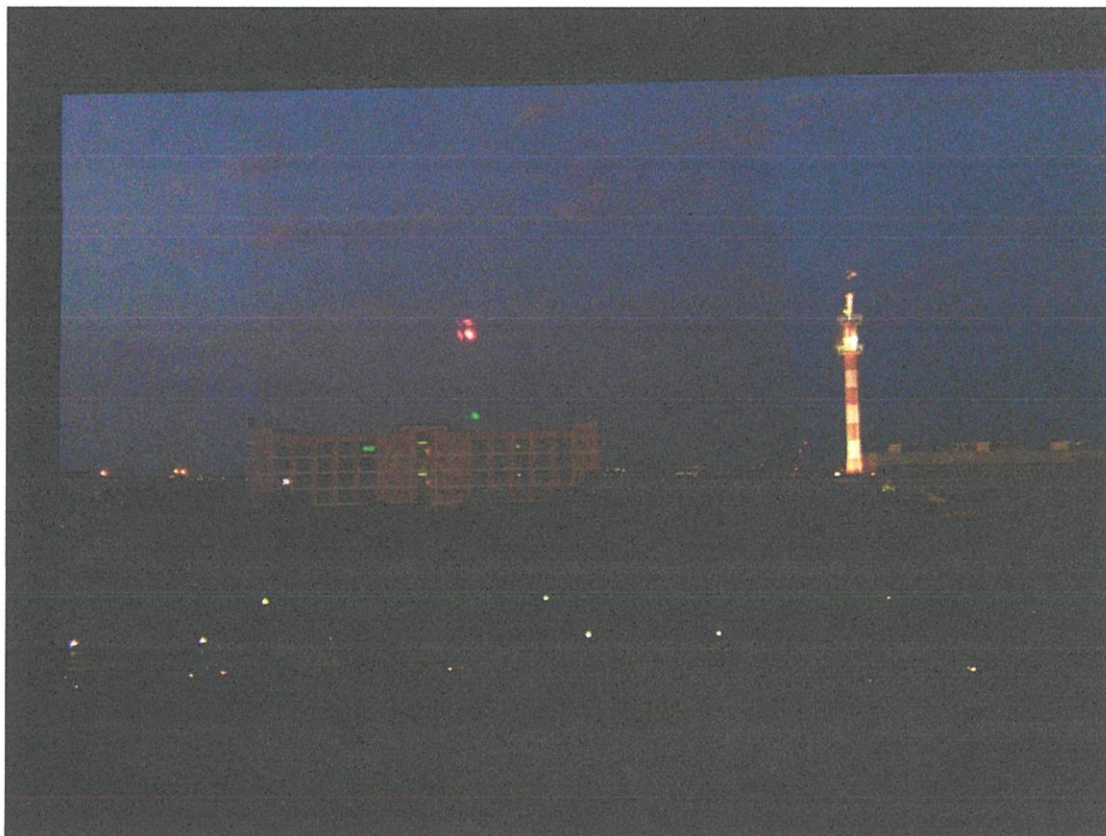


รูปที่ 4.4 แสดงภาพที่ถ่ายได้จาก Smart Module SC 20-E ถ่ายช่วงเวลา 18 นาฬิกา 41 นาที

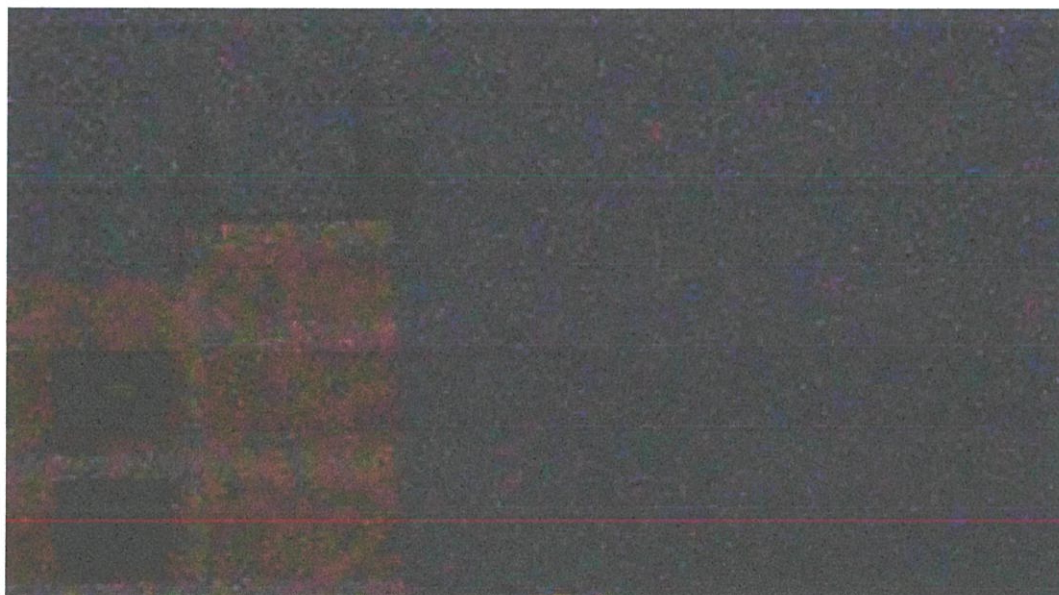


รูปที่ 4.5 เมื่อขยายภาพที่ 4.4 ที่ 500%

ภาพที่ถ่ายได้เมื่อตอนมีแสงน้อยมาก จากรูปที่ 4.6 จะเห็นได้ว่าภาพเริ่มมอง
รายละเอียดไม่ชัดแล้ว เนื่องจากไม่มีแสง และมี Noise พอประมาณ



รูปที่ 4.6 แสดงภาพที่ถ่ายได้จาก Smart Module SC 20-E ถ่ายช่วงเวลา 18 นาฬิกา 46 นาที

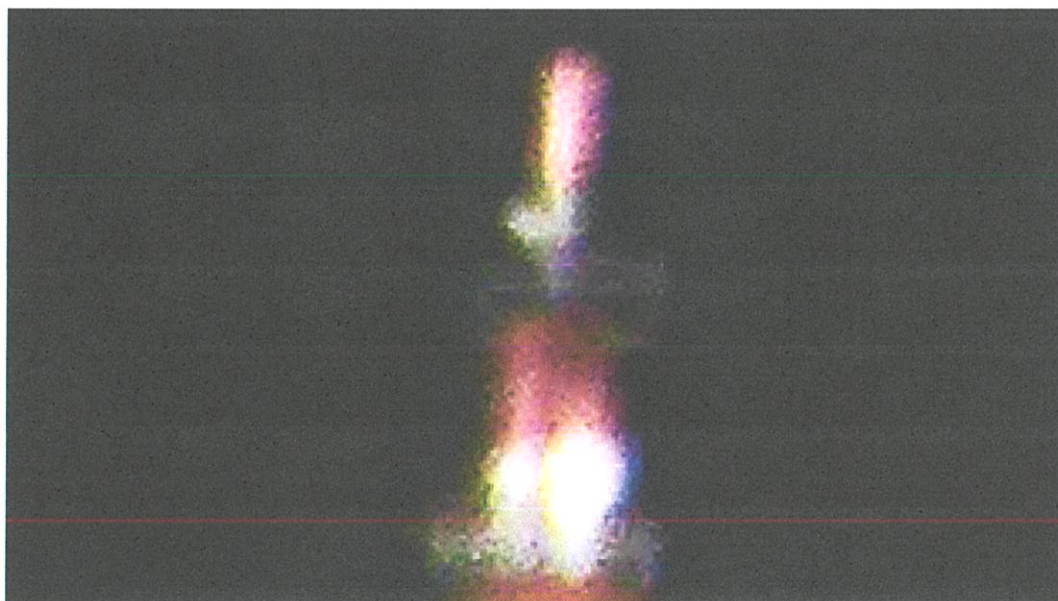


รูปที่ 4.7 เมื่อขยายภาพที่ 4.6 ที่ 500%

ภาพที่ถ่ายได้เมื่อตอนมีแสงมาก จากรูปที่ 4.8 จะเห็นว่าภาพมืดทั้งภาพ และมองไม่เห็นรายละเอียดใด ๆ รวมถึงสูญเสียความสามารถในการโฟกัสอีกด้วย



รูปที่ 4.8 แสดงภาพที่ถ่ายได้จาก Smart Module SC 20-E ถ่ายช่วงเวลา 19 นาฬิกา 51 นาที



รูปที่ 4.9 เมื่อขยายภาพที่ 4.8 ที่ 500%

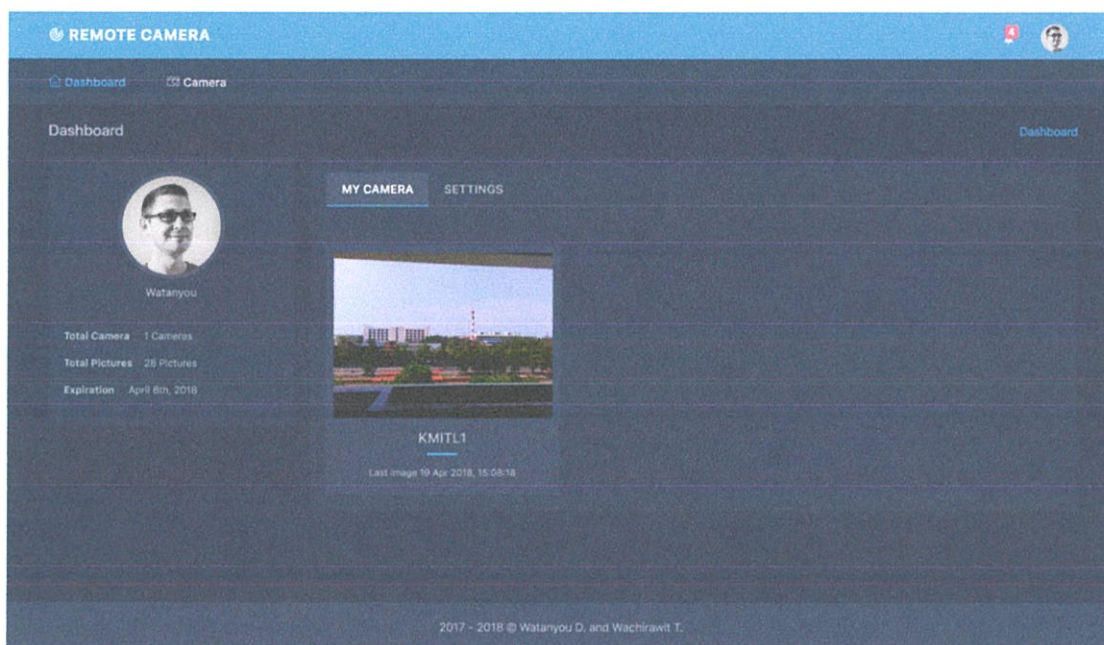
4.2 การทดสอบการดูรูปภาพบนเว็บแอปพลิเคชัน

4.2.1 วัตถุประสงค์

เพื่อทดสอบการแสดงผลรูปภาพบนเว็บแอปพลิเคชัน

4.2.2 วิธีการทดลอง

1) เข้าไปที่ เว็บแอปพลิเคชัน แล้วเลือกกล้องที่ต้องการจะดูรูปภาพ

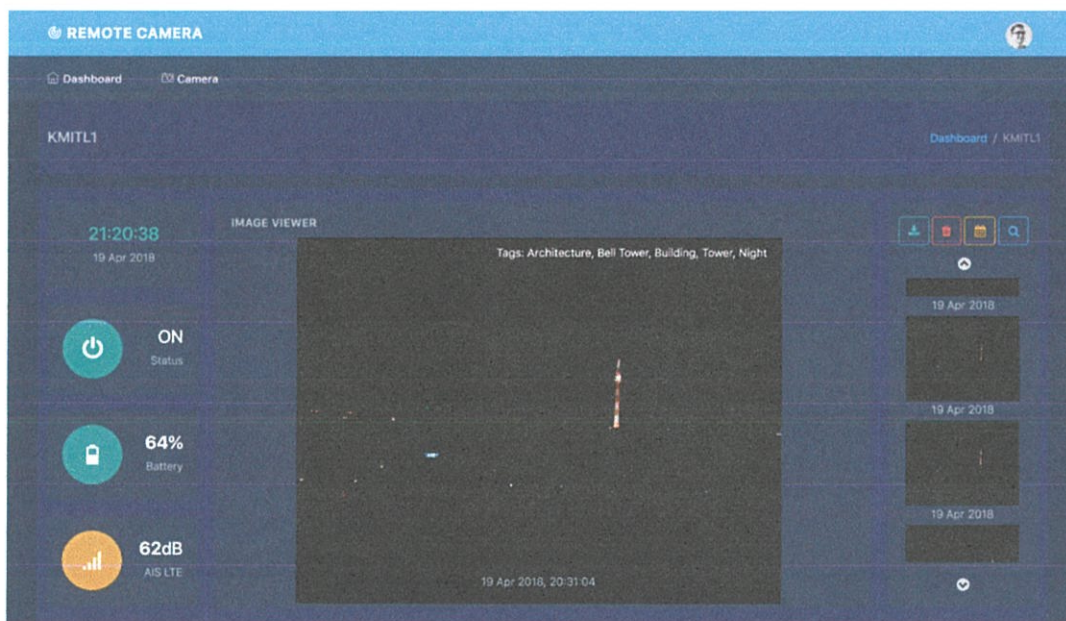


รูปที่ 4.10 หน้า Dashboard แสดงข้อมูลพื้นฐานของผู้ใช้งาน และกล้องตรวจตราทั้งหมดที่มี

4.2.3 ผลการทดลอง

จะได้หน้าต่างแสดงรูปภาพที่กล้องถ่ายได้ ด้านบนขวาของภาพจะบอกถึงวัตถุที่กล้องวิเคราะห์ได้

รูปภาพที่ใหม่กว่า จะอยู่ทางด้านบน และจะอัปเดตภาพอัตโนมัติ หากกล้องทำการถ่ายภาพใหม่ ส่วนแถบทางด้านซ้าย จะเป็นวัน เวลา และสถานะของกล้อง



รูปที่ 4.11 หน้าเว็บแอปพลิเคชัน แสดงรูปภาพที่ได้จากกล้องตรวจตรา

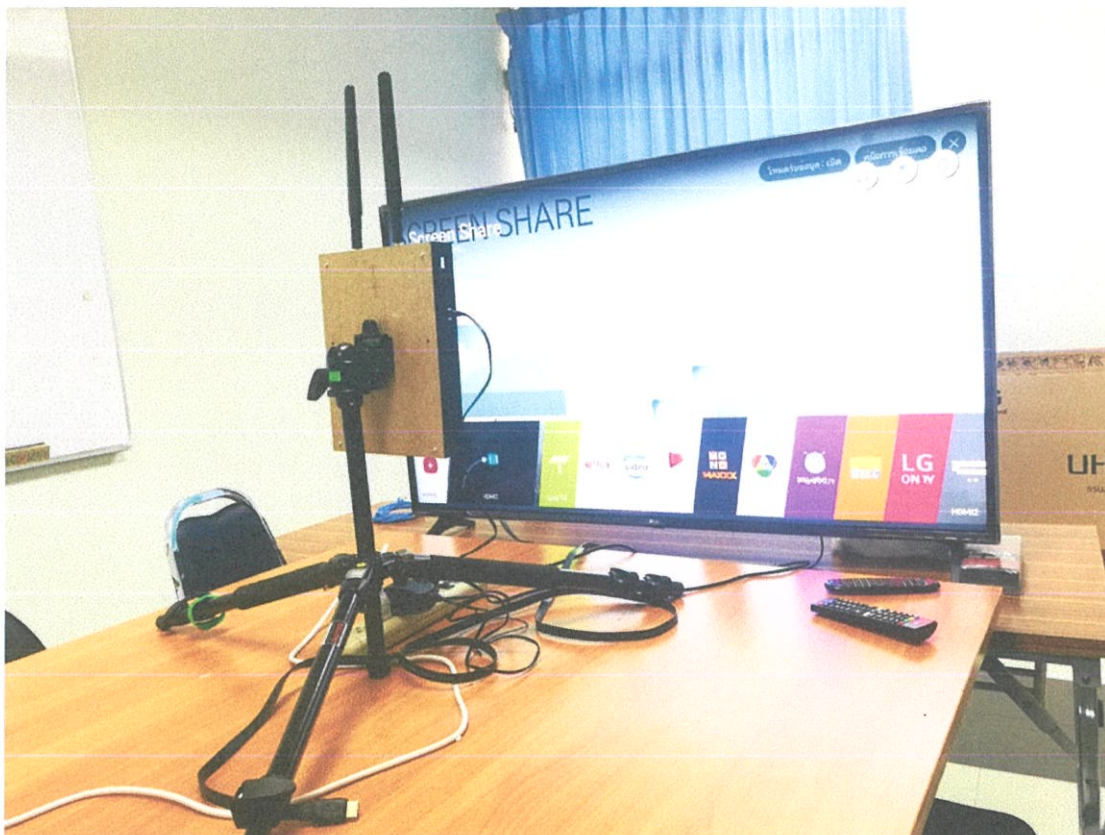
4.3 การทดสอบการแจ้งเตือนเมื่อเจอวัตถุ หรือสถานการณ์ที่กำหนดไว้

4.3.1 วัตถุประสงค์

เพื่อทดสอบการตรวจจับวัตถุ สำหรับการนำไปติดตั้งตามสถานที่ต่าง ๆ และการแจ้งเตือนผ่านแอปพลิเคชัน Line Notify

4.3.2 วิธีการทดลอง

นำกล้องตรวจตรา มาตั้งหน้าโทรทัศน์ ดังรูปที่ 4.12 และเปิดภาพเหตุการณ์ต่าง ๆ 3 เหตุการณ์ ได้แก่ ไฟไหม้ที่ฟักอาศัย น้ำท่วมโรงงาน และภาพบุคคลบุกรุกที่ดิน พร้อมกับตั้งค่าให้ตรวจจับในเว็บแอปพลิเคชัน ด้วยคำว่า “Fire”, “Flood” และ “Person” ดังภาพที่ 4.13



รูปที่ 4.12 บรรยากาศขณะทำการทดสอบการแจ้งเตือนเมื่อเจอวัตถุ หรือสถานการณ์ที่กำหนดไว้



รูปที่ 4.13 ภาพแสดงตัวอย่างรูปภาพที่ใช้ในการทดลอง^{21 22 23}

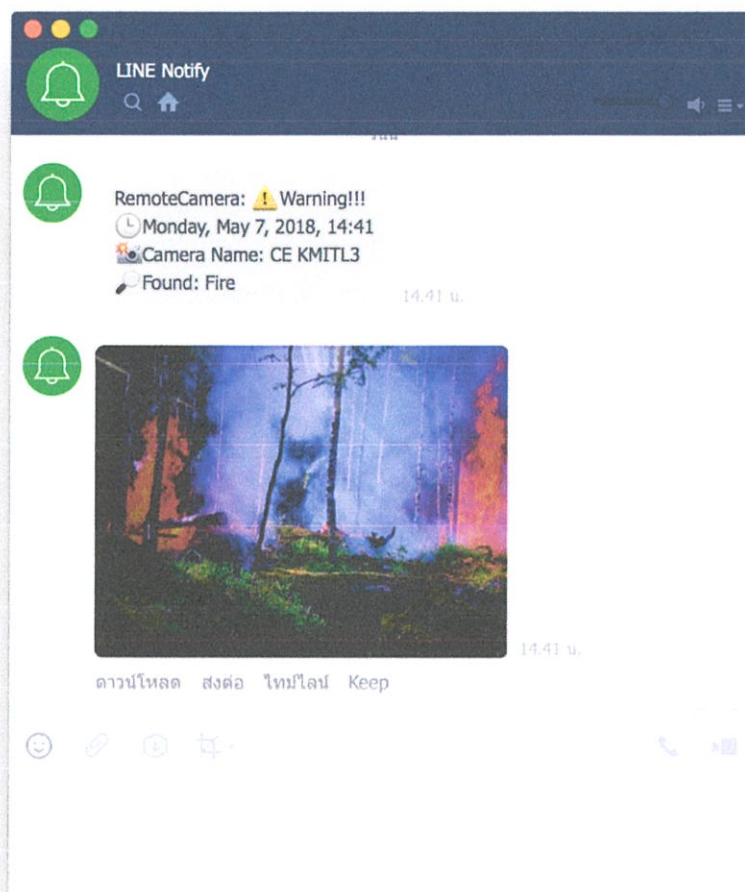
²¹ Fire [ออนไลน์] <https://www.pexels.com> (สืบค้นข้อมูล : เมษายน 2561)

²² Flood [ออนไลน์] <http://pixabay.com> (สืบค้นข้อมูล : เมษายน 2561)

²³ Robert Clark. Brazen thieves waltzed past CCTV to burgle Cheung Sha home last night [ออนไลน์] <https://lantaunews.com/2018/04/19/brazen-thieves-waltzed-past-cctv-to-burgle-cheung-sha-home-last-night/> (สืบค้นข้อมูล : ตุลาคม 2560)

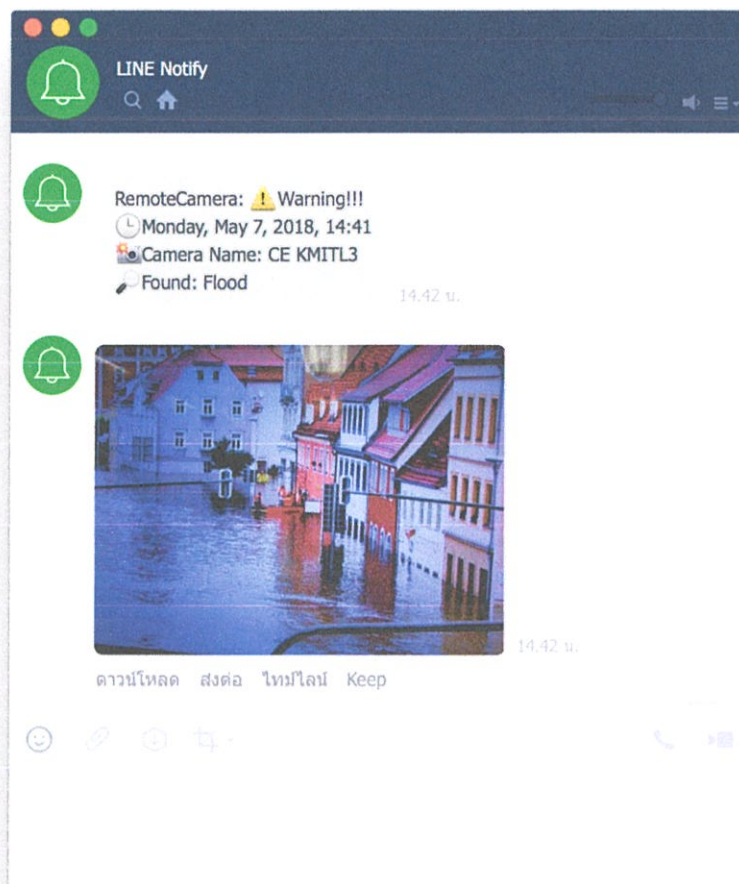
4.3.3 ผลการทดลอง

ในเหตุการณ์ไฟไหม้ กล้องสามารถจำแนกวัตถุได้เป็น Flora, Jar, Plant, Potted Plant, Pottery, Fire และทำการแจ้งเตือนดังรูปที่ 4.14



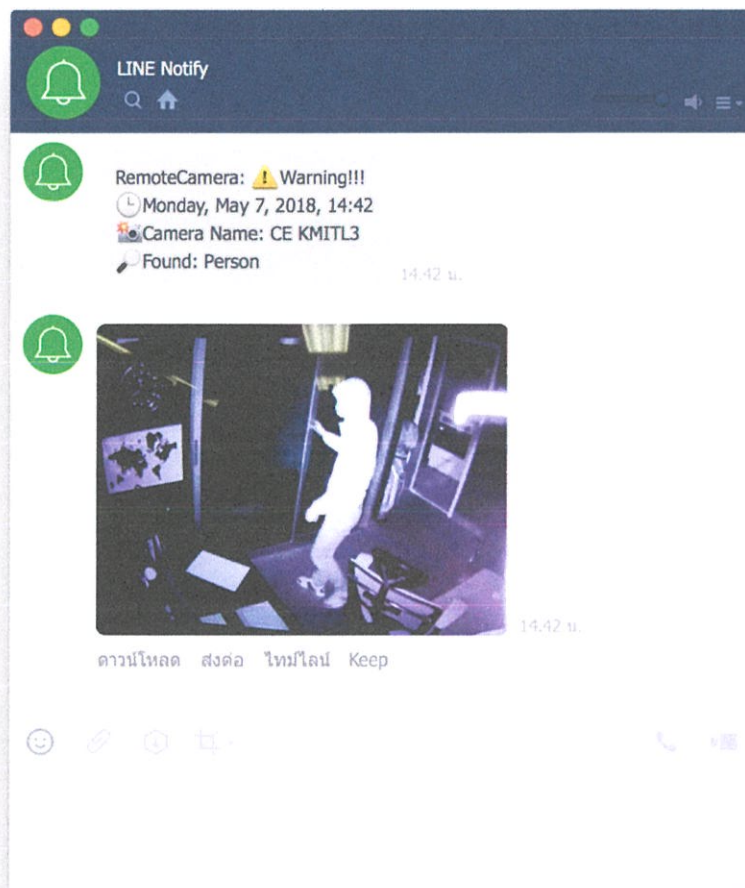
รูปที่ 4.14 การแจ้งเตือนผ่าน Line Notify เมื่อเจอเหตุการณ์ไฟไหม้

ในเหตุการณ์น้ำท่วม กล้องสามารถจำแนกวัตถุได้เป็น Flood, Nature, Building, Housing และทำการแจ้งเตือนดังรูปที่ 4.15



รูปที่ 4.15 การแจ้งเตือนผ่าน Line Notify เมื่อเจอเหตุการณ์น้ำท่วม

ในเหตุการณ์มีผู้บุกรุก กล้องสามารถจำแนกวัตถุได้เป็น Human, American Football, Person, Crash Helmet, Football และทำการแจ้งเตือนดังรูปที่ 4.16



รูปที่ 4.16 การแจ้งเตือนผ่าน Line Notify เมื่อเจอบุคคลบุกรุก

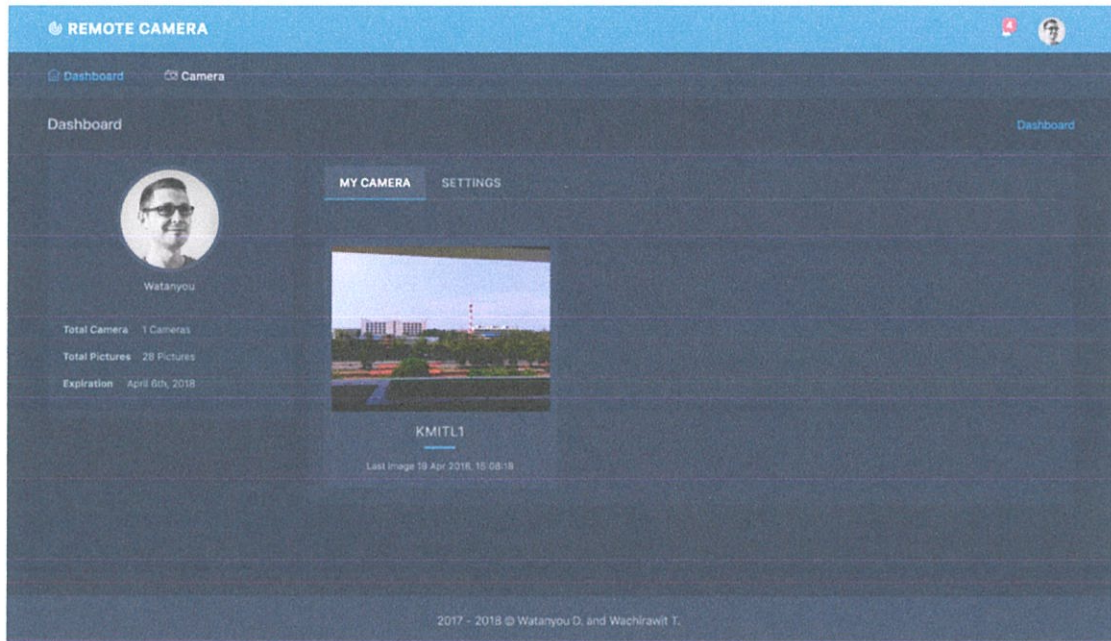
4.4 การทดสอบการสร้างวิดีโอแบบ Time-Lapse

4.4.1 วัตถุประสงค์

เพื่อทดลองสร้างวิดีโอ Time-Lapse จาก เว็บแอปพลิเคชัน

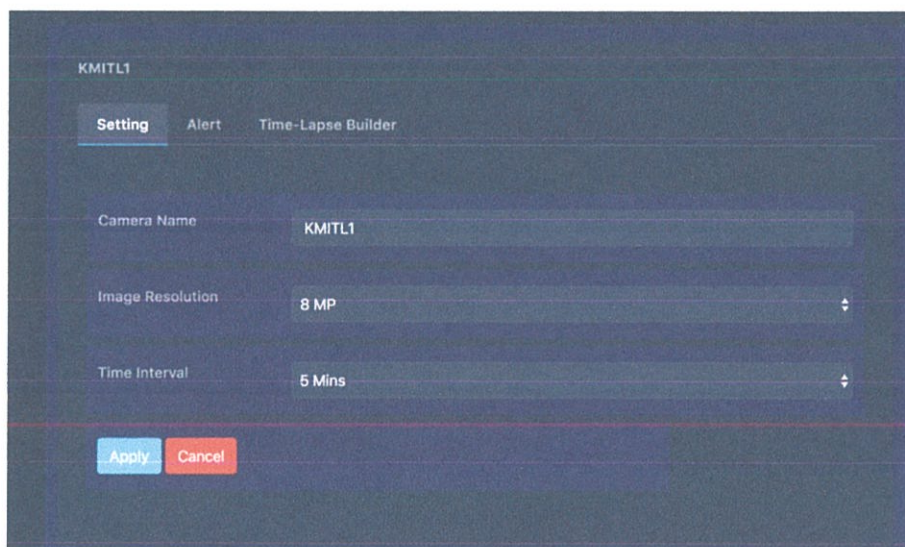
4.4.2 วิธีการทดลอง

- 1) เข้าไปที่ เว็บแอปพลิเคชัน แล้วเลือกกล้องที่ต้องการจะถ่ายวิดีโอ Time-Lapse



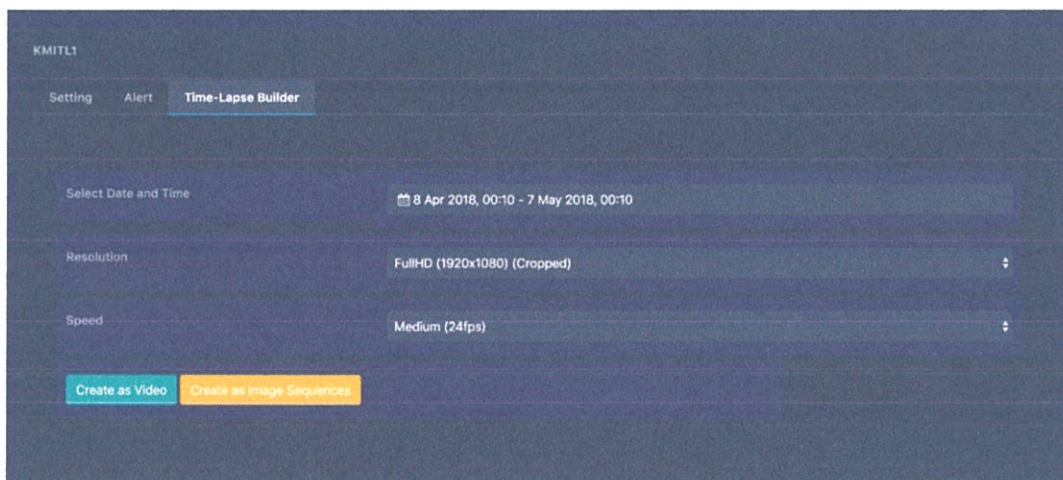
รูปที่ 4.17 หน้า Dashboard แสดงข้อมูลพื้นฐานของผู้ใช้งาน และกล้องตรวจตราทั้งหมดที่มี

- 2) เลือกเมนู Time-Lapse Builder



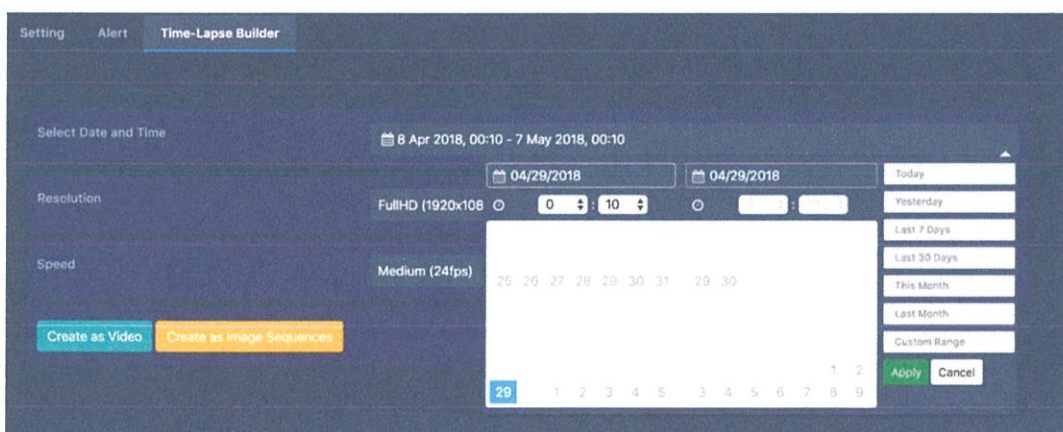
รูปที่ 4.18 หน้าต่างการตั้งค่าของกล้องตรวจตราหน้าแรก

- 3) ตั้งค่าพารามิเตอร์ต่าง ๆ เกี่ยวกับการสร้างวิดีโอ Time-Lapse โดยสามารถเลือกช่วงเวลาที่ให้นำรูปภาพมาทำ Time-Lapse ความละเอียดของวิดีโอ Time-Lapse ความเร็วของวิดีโอ เมื่อเลือกครบแล้ว เว็บแอปพลิเคชัน จะแสดงความยาวของวิดีโอให้ ผู้ใช้งานสามารถเลือกรูปแบบของวิดีโอได้ ในรูปแบบไฟล์วิดีโอ WebM และรูปแบบของภาพนิ่ง ที่บีบอัดอยู่ในรูปแบบ .tar



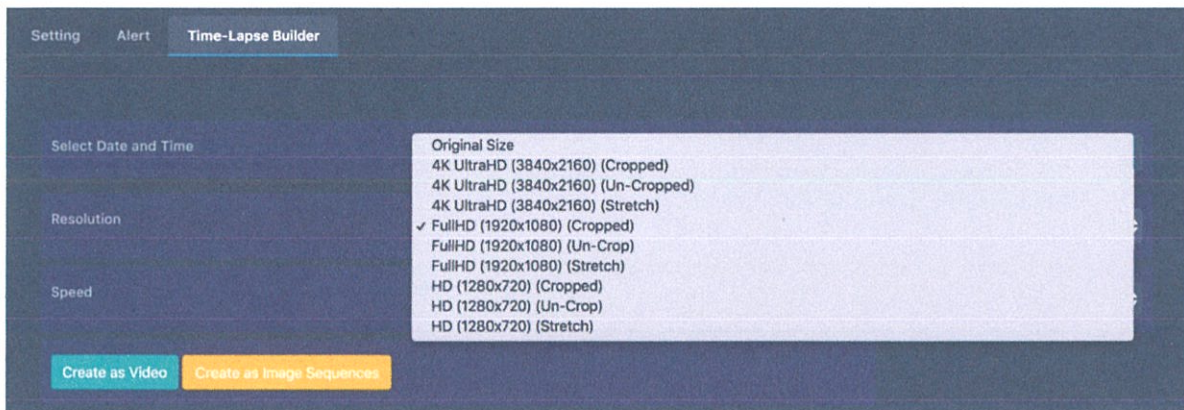
รูปที่ 4.19 หน้าต่างการตั้งค่าการสร้างวิดีโอ Time-Lapse

- ช่วงเวลาที่สร้างวิดีโอ Time-Lapse โดยสามารถเลือกตามช่วงเวลาที่กำหนดไว้ หรือปรับแต่งเองได้



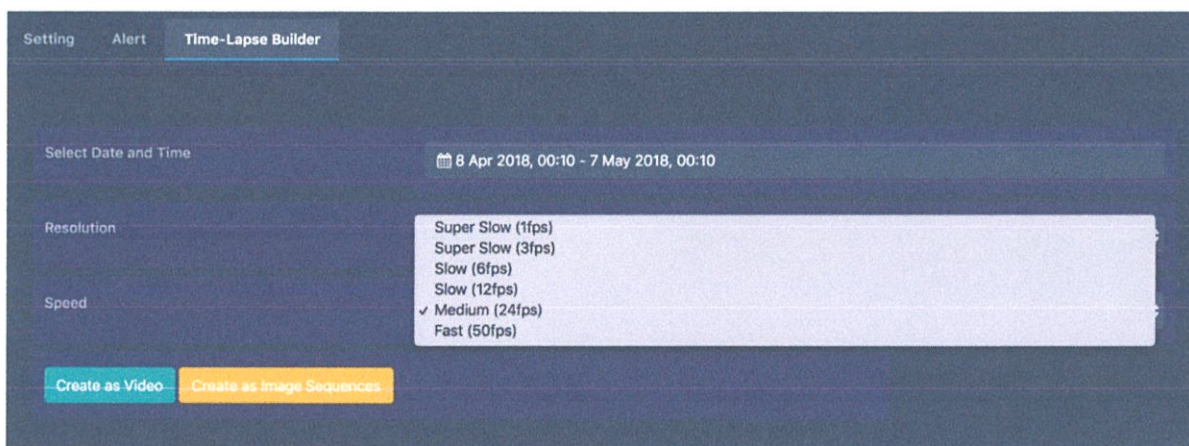
รูปที่ 4.20 หน้าต่างเลือกช่วงเวลาของการทำวิดีโอ Time-Lapse

- ความละเอียดของวิดีโอ สามารถเลือกได้ตั้งแต่ ขนาดเท่ากับรูปต้นฉบับ ขนาด UHD 4K, Full HD และ HD



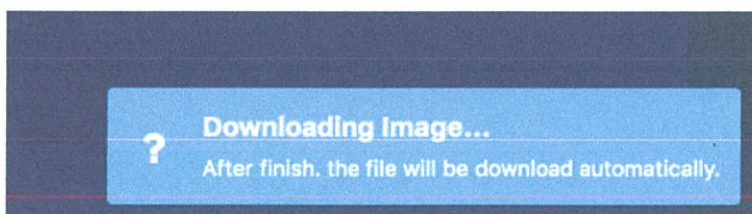
รูปที่ 4.21 หน้าต่างเลือกความละเอียดของการทำวิดีโอ Time-Lapse

- ความเร็วของการแสดงภาพ สามารถปรับได้ 3 ระดับ



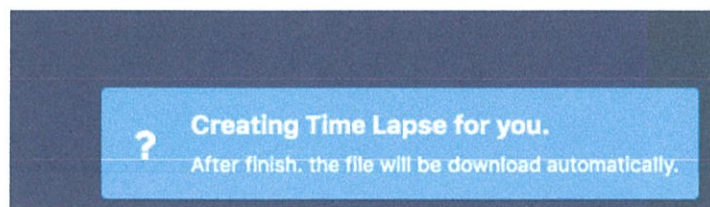
รูปที่ 4.22 หน้าต่างเลือกความเร็วทำวิดีโอ Time-Lapse

- 4) ทดลองกด Create as Video จะขึ้นแจ้งเตือนเมื่อกำลังดาวน์โหลดรูปภาพ



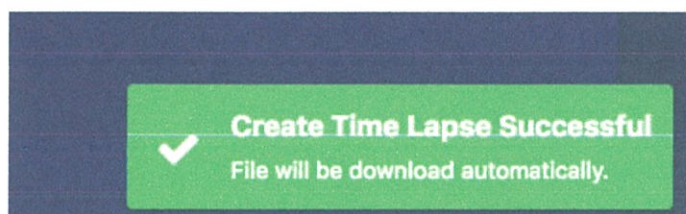
รูปที่ 4.23 การแจ้งเตือนว่ากำลังดาวน์โหลดรูปภาพเพื่อทำวิดีโอ Time-Lapse

- 5) เมื่อคานวน์โหลดเสร็จ เว็บแอปพลิเคชัน จะเริ่มสร้างวิดีโอให้อัตโนมติ



รูปที่ 4.24 การแจ้งเตือนว่ากำลังทำวิดีโอ Time-Lapse

- 6) เมื่อวิดีโอสร้างเสร็จสมบูรณ์ จะขึ้นแจ้งเตือน ว่าจะคานวน์โหลดไฟล์เข้าสู่เครื่องอัตโนมัติ



รูปที่ 4.25 การแจ้งเตือนการทำวิดีโอ Time-Lapse เสร็จสมบูรณ์



รูปที่ 4.26 ไฟล์ที่ได้จากการทำวิดีโอ Time-Lapse

4.4.3 ผลการทดลอง

- 1) ใ้ดูวิดีโอ Time-Lapse ความยาว 1 นาที 12 วินาที



รูปที่ 4.27 ตัวอย่างวิดีโอของภาพ Time-Lapse

4.5 การคำนวณค่าใช้จ่ายของตัวอุปกรณ์ และค่าบริการของ AWS

4.5.1 วัตถุประสงค์

เพื่อหาค่าใช้จ่ายทั้งหมดของกล้องตรวจตรา พร้อมค่าบริการคลาวด์โดย AWS

4.5.2 วิธีการทดลอง

วัดทรัพยากรที่ใช้ใน AWS และราคาของ EVB SC20-E

4.5.3 ผลการทดลอง

หลังจากคำนวณราคาออกมา จะแยกเป็นสองตาราง คือ ตารางที่ 4.1 แสดงราคาของกล้องตรวจตรา และตารางที่ 4.2 แสดงราคาค่าบริการ AWS เมื่อถ่ายภาพเป็นจำนวนต่าง ๆ กันต่อเดือน (ไม่รวมค่าบริการอินเทอร์เน็ตไร้สาย)

ตารางที่ 4.1 แสดงราคาของ EVB SC20-E และ Smart Module SC20-E

ลำดับที่	รายการ	ราคา	หมายเหตุ
1	EVB Board + SC20-E	12,449 บาท	มีอุปกรณ์เกินที่จำเป็นต้องใช้ เช่น จอภาพ และ ลำโพง รวมถึงมีขนาดใหญ่
2	Smart Module SC20-E	3,474 บาท	มีแค่ตัว Smart Module เท่านั้น จำเป็นต้องออกแบบแผงวงจร และติดตั้งส่วนประกอบอื่น ๆ เช่น กล้องถ่ายรูปเอง

ตารางที่ 4.2 แสดงราคาค่าใช้จ่ายจาก AWS

ช่วงเวลาในการถ่ายภาพ	1 วินาที	30 วินาที	1 นาที	15 นาที	1 ชั่วโมง	12 ชั่วโมง
ภาพ/เดือน	2,592,00	86,400	43,200	2,880	720	60
ค่าใช้จ่าย	39,490 บาท	1,970 บาท	1,350 บาท	140 บาท	43 บาท	33 บาท

จะเห็นได้ว่า หากนำ EVB Board + SC20-E มาใช้งาน ราคา 12,449 บาท ถือว่าไม่แพงนัก แต่หากพัฒนาอุปกรณ์เองโดยใช้ Smart Module SC20-E ราคาจะลดลงไปมาก

ในส่วนของค่าบริการรายเดือน หากกำหนดการถ่ายภาพขั้นต่ำที่ 15 นาที หรือมากกว่า จะเป็นช่วงราคาที่สมเหตุสมผล และสามารถทำกำไรจากจุดนี้ได้อีกมาก

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 บทสรุป

ภายในโครงการมีการทำงานร่วมกัน 3 ส่วนหลัก คือ ส่วนกล้องถ่ายภาพ ส่วนของ Amazon Web Service และส่วนของเว็บแอปพลิเคชัน

5.1.1 ส่วนของกล้องถ่ายภาพ

- 1) สามารถถ่ายภาพเป็นช่วงเวลาตามที่กำหนด
- 2) สามารถอัปโหลดรูปภาพเข้าไปยัง Amazon S3 ได้
- 3) สามารถทำงานร่วมกับ Amazon IoT ได้

5.1.2 ส่วนของ Amazon Web Service

- 1) สามารถใช้งาน Amazon S3 ในการเก็บข้อมูล และลบข้อมูลได้
- 2) สามารถใช้งาน Amazon Lambda ในการประมวลผลเพื่อรองรับในแต่ละพีเจเออร์ได้
- 3) สามารถใช้งาน Amazon Step Functions เพื่อควบคุมการทำงานของ Amazon Lambda ได้
- 4) สามารถใช้งาน Amazon Cognito สำหรับการจัดการกับระบบ User และการ Authenticate ของ User และกล้องตรวจตรา
- 5) สามารถใช้งาน Amazon DynamoDB เพื่อจัดเก็บข้อมูลต่าง ๆ
- 6) สามารถใช้งาน Amazon API Gateway เพื่อสร้าง RESTful API สำหรับ เว็บแอปพลิเคชัน ได้
- 7) สามารถใช้งาน AWS IoT เพื่อควบคุม และรายงานผลสถานะของอุปกรณ์ได้
- 8) สามารถใช้งาน Amazon Rekognition เพื่อวิเคราะห์หาวัตถุภายในภาพถ่ายได้

5.1.3 ส่วนของเว็บแอปพลิเคชัน

- 1) สามารถจัดการกล้องของแต่ละผู้ใช้งานได้
- 2) สามารถดูรูปภาพที่ถ่ายไว้ได้ และสามารถดูภาพถ่ายแยกในแต่ละวันได้
- 3) สามารถตั้งค่ากล้องถ่ายภาพระยะไกลได้
- 4) สามารถตั้งค่าการแจ้งเตือนเมื่อพบวัตถุได้
- 5) สามารถดาวน์โหลดรูปภาพถ่ายจาก Cloud ได้
- 6) สามารถทำวิดีโอ Time-Lapse ในขนาดต่าง ๆ ได้

5.2 ปัญหาและอุปสรรค

5.2.1 ส่วนของกล้องถ่ายภาพ

- 1) แบตเตอรี่มีความจุน้อยเกินไป
- 2) ไม่สามารถเข้าถึง GPIO ของ Smart Module

5.2.2 ส่วนของ Amazon Web Service

- 1) AWS Lambda ทำงานค่อนข้างช้า และจำกัดเวลาในการทำงานเพียง 5 นาที ไม่เหมาะกับงานที่เน้นประมวลผลหนัก ๆ เช่น การทำวิดีโอ Time-Lapse
- 2) Library ของบริการ Amazon DynamoDB มีการเปลี่ยนแปลงทำให้บางฟังก์ชันไม่สามารถใช้งานได้

5.2.3 ส่วนของ เว็บแอปพลิเคชัน

- 1) หน้าเว็บแอปพลิเคชันมีการดาวน์โหลดส่วนประกอบค่อนข้างเยอะ ทำให้แสดงผลช้า หากอินเทอร์เน็ตมีความเร็วต่ำ
- 2) การสร้างวิดีโอ Time-Lapse จะทำการ Encode บนเบราว์เซอร์ของผู้ใช้งาน ซึ่งเว็บเบราว์เซอร์จำเป็นต้องรองรับ (รองรับแค่เว็บเบราว์เซอร์ ของ Google Chrome เท่านั้น)

5.3 แนวทางการพัฒนาต่อ

5.3.1 ส่วนของกล้องถ่ายภาพ

- 1) เปลี่ยนกล้องถ่ายภาพให้มีคุณภาพดีขึ้น
- 2) ติดตั้ง PIR (Passive Infrared Sensor) สำหรับตรวจจับการเคลื่อนไหว ให้ถ่ายภาพทันทีที่เกิดการเคลื่อนไหว
- 3) ปรับปรุงการใช้พลังงานของระบบให้น้อยลง
- 4) ติดตั้ง Infrared LED สำหรับถ่ายภาพตอนกลางคืน

5.3.2 ส่วนของ Amazon Web Service

- 1) ปรับปรุงความปลอดภัยของแต่ละส่วนให้รัดกุม และปลอดภัยมากขึ้น
- 2) เพิ่มฟีเจอร์ในการตรวจจับ และจดจำใบหน้า เพื่อเพิ่มความสามารถในการใช้งานให้มากขึ้น

5.3.3 ส่วนของ เว็บแอปพลิเคชัน

- 1) ปรับปรุงความปลอดภัยในการเชื่อมต่อกับ Amazon Web Service ให้มากขึ้น
- 2) ปรับปรุงรูปแบบเว็บไซต์ เพื่อให้แสดงผลบนสมาร์ตโฟนให้สวยงามขึ้น
- 3) เพิ่มความสามารถในการทำ Video Time-Lapse ในเบราว์เซอร์อื่น ๆ

บรรณานุกรม

Eyetrax. **The Ranger**. [Online].Available :

<https://eyetrax.net/ranger>

Amazon.**Amazon Web Services**. [Online].Available :

<https://aws.amazon.com/>

Amazon.**Amazon Simple Storage Service (S3)**. [Online].Available :

<https://aws.amazon.com/s3/>

Amazon.**AWS Lambda**. [Online].Available :

<https://aws.amazon.com/lambda/>

Amazon.**Amazon Simple Workflow Service**. [Online].Available :

<https://aws.amazon.com/swf/>

Amazon.**Amazon Internet of Things**. [Online].Available :

<https://aws.amazon.com/iot/>

Amazon.**Amazon IoT Platform**. [Online].Available :

<https://aws.amazon.com/iot-platform/>

Amazon.**Amazon Rekognition**. [Online].Available :

<https://aws.amazon.com/Rekognition/>

Amazon.**Amazon API Gateway**. [Online].Available :

<https://aws.amazon.com/api-gateway/>

Amazon.**Amazon Cognito**. [Online].Available :

<https://aws.amazon.com/cognito/>

Amazon.**Amazon DynamoDB**. [Online].Available :

<https://aws.amazon.com/dynamodb/>

Quectel. **Quectel LTE SC20 Smart Module**. [Online].Available :

www.quectel.com/product/sc20

Github. **aws-sdk-android-samples**. [Online].Available :

<https://github.com/aws-labs/aws-sdk-android-samples>

Quectel. **Smart EVB user Guide**. [Online].Available :

http://www.quectel.com/UploadImage/Quectel_Smart_EVB_User_Guide_V1.0.pdf