

การประยุกต์ใช้โปรโตคอล ONVIF เพื่อรองรับการจัดการกล้องไอพี  
และการสตรีมวิดีโอผ่านเว็บไซต์

Application of the ONVIF Protocol to support IP camera  
management and video streaming through websites.

นายนันทกร โสภาพ

รายงานสรุปโครงการฉบับนี้เป็นส่วนหนึ่งของ  
การศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต  
วิศวกรรมระบบสมองกลฝังตัว คณะวิศวกรรมศาสตร์  
มหาวิทยาลัยบูรพา  
ปีการศึกษา 2565

การประยุกต์ใช้โปรโตคอล ONVIF เพื่อรองรับการจัดการกล้องไอพี  
และการสตรีมวิดีโอผ่านเว็บไซต์

Application of the ONVIF Protocol to support IP camera  
management and video streaming through websites.

นายนันทกร โสภาพ

รายงานสรุปโครงการฉบับนี้เป็นส่วนหนึ่งของ  
การศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต  
วิศวกรรมระบบสมองกลฝังตัว คณะวิศวกรรมศาสตร์  
มหาวิทยาลัยบูรพา  
ปีการศึกษา 2565

เรื่อง	การประยุกต์ใช้protoคอล ONVIF เพื่อรองรับการจัดการกล้องไอพี และการสตรีมวิดีโอผ่านเว็บไซต์
สาขาวิชา	วิศวกรรมระบบสมองกลฝังตัว
ผู้เขียน	นายนันทกร ไสภพ รหัสนิสิต 62050553
อาจารย์ที่ปรึกษา	รองศาสตราจารย์วิรุฬห์ ศรีบริรักษ์
ปีการศึกษา	2565

---

ภาควิชาวิศวกรรมระบบสมองกลฝังตัว คณะวิศวกรรมศาสตร์มหาวิทยาลัยบูรพา อนุมัติรายงานสรุปโครงการฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์ บัณฑิต สาขาวิชาวิศวกรรมระบบบวชิกรรมระบบสมองกลฝังตัว

..... อาจารย์ที่ปรึกษา

(รองศาสตราจารย์วิรุฬห์ ศรีบริรักษ์)

..... หัวหน้าภาควิชาวิศวกรรมไฟฟ้า

(อาจารย์ ดร.ณัฐพันธ์ ถนนสัตย์)

## บทคัดย่อ

ในปัจจุบัน, ธุรกิจในกลุ่มระบบปรักษาความปลอดภัยมีความสำคัญอย่างมากเนื่องจากมีความต้องการในความปลอดภัยสูงขึ้นทั้งในเรื่องชีวิตและทรัพย์สิน ในอดีต, การใช้โปรแกรมของบริษัทผู้ผลิตกล้อง IP เพื่อคุ้มภาพเป็นที่ยอมรับ แต่หากผู้ใช้งานต้องการใช้กล้อง IP ยี่ห้ออื่น ๆ จะต้องติดตั้งโปรแกรมเพิ่มเติม ส่งผลให้ผู้ใช้งานที่ต้องการเลือกใช้กล้อง IP ตามงบประมาณที่เหมาะสมต้องมีโปรแกรมที่ต้องติดตั้งมากขึ้น ซึ่งส่งผลกระทบต่อผู้ใช้งานที่ต้องการความยืดหยุ่นในการเลือกใช้กล้อง IP ตามความต้องการในช่วงงบประมาณนั้น และที่กำลังได้รับความนิยมในปัจจุบันคือการใช้ AI ร่วมกับกล้อง IP เพื่อเพิ่มความสามารถในการตรวจจับ โดยที่ผู้ใช้งานจะได้รับรายงานเพียงข้อมูลพื้นฐาน หากแต่ธุรกิจระบบปรักษาความปลอดภัยมีความต้องการเพิ่มความสามารถในการให้บริการในการตรวจจับให้มากขึ้น โดยการเพิ่มจำนวน AI ที่ใช้ร่วมกับกล้อง IP ทำให้ข้อมูลที่ได้รับมีมากขึ้นตามและการใช้งานในรูปแบบเดิมไม่สามารถตอบสนองต่อความต้องการของผู้ใช้งานได้เต็มที่

งานวิจัยนี้จึงมุ่งเน้นการประยุกต์ใช้โปรโตคอล ONVIF เพื่อรองรับการจัดการกล้อง IP และการสตรีมวิดีโอผ่านเว็บไซต์ เพื่อตอบสนองต่อความต้องการของผู้ใช้งานให้สามารถเข้ามาร่วมต่อและจัดการกล้อง IP และส่งข้อมูลภาพวิดีโอและข้อมูลเมตาจากกล้อง IP ผ่านหน้าเว็บไซต์ได้อย่างสะดวกในการพัฒนาโค้ดการนี้ได้นำเอาแนวคิดรวมถึงเทคนิคการสื่อสารผ่านเครือข่ายเกี่ยวข้อง เช่น การสร้างเว็บแอปพลิเคชันด้วย Golang, การสตรีมวิดีโอ, การจัดการข้อมูลเมตาในรูปแบบ JSON รวมถึงการนำโปรโตคอล ONVIF มาประยุกต์ใช้ในการส่งข้อมูลเมตาจากกล้อง IP เพื่อให้เกิดประโยชน์ในการพัฒนาระบบซอฟต์แวร์ เพื่อสนับสนุนธุรกิจระบบปรักษาความปลอดภัยให้เป็นไปอย่างมีประสิทธิภาพ

**คำสำคัญ :** กล้องไอพี, การสตรีมวิดีโอ, ข้อมูลเมตา, เว็บแอปพลิเคชัน, Open Network Video Interface Forum (ONVIF)

## กิตติกรรมประกาศ

การจัดทำงานนิพนธ์เรื่อง การประยุกต์ใช้protocol ONVIF เพื่อรองรับการจัดการกล้องไอพี และการสตรีมวิดีโอผ่านเว็บไซต์ ครั้งนี้สำเร็จลุล่วงได้ด้วยดีโดยได้รับความช่วยเหลือและการสนับสนุน จากผู้ที่มีความรู้หลายท่าน โดยเฉพาะอย่างยิ่ง ร.ศ.วิรุฬห์ ศรีบริรักษ์ ซึ่งเป็นอาจารย์ที่ปรึกษา ที่ได้เสียสละเวลาให้คำแนะนำ ให้ความรู้ด้านวิชาการ แนวทางการแก้ไขปัญหา และเทคนิคต่างๆ ตลอดจนข้อคิดเห็นที่เป็นประโยชน์ในการแก้ไขปรับปรุงผลงานงานนิพนธ์ฉบับนี้ให้สมบูรณ์มากยิ่งขึ้น ผู้จัดทำซาบซึ้งในความกรุณาของท่านเป็นอย่างยิ่งและขอกราบขอบพระคุณท่านเป็นอย่างสูง

ขอขอบคุณเพื่อนๆ ร่วมรุ่นทุกๆ คน รวมถึงเพื่อนๆ พี่ๆ ณ ที่ทำงาน ที่เคยให้ความช่วยเหลือ ชี้แนะกันและกันมาโดยตลอดในระยะเวลาในการศึกษา

ขอกราบขอบพระคุณคุณพ่อคุณแม่ ครอบครัวและบุคคลอันเป็นที่รักยิ่ง ที่เคยเป็นส่วนหนึ่ง ของกำลังใจที่สำคัญยิ่งในการจัดทำงานนิพนธ์จนประสบความสำเร็จลุล่วงไปได้ด้วยดี ซึ่งทุกท่านจะถูก จดจำไว้ในจิตใจของผู้จัดทำงานนิพนธ์ตลอดไป

ในท้ายที่สุดนี้ผู้จัดทำหวังเป็นอย่างยิ่งว่าผลงานงานนิพนธ์ฉบับนี้จะเป็นประโยชน์กับผู้ที่ ต้องการศึกษาด้านการดูแลระบบบันทึกภาพความปลอดภัยด้วยการสตรีมวิดีโอและส่งข้อมูลเมต้าผ่าน เว็บไซต์ และหากมีข้อผิดพลาดประการใดในงานนิพนธ์ฉบับนี้ ทางผู้จัดทำต้องขอกราบขอภัยเป็น อย่างสูงมา ณ ที่นี้

นายนันทกร โสภาพ

## สารบัญ

หน้า

บทคัดย่อ .....	๑
กิตติกรรมประกาศ.....	๑
<b>สารบัญ.....</b>	<b>๒</b>
สารบัญตาราง.....	๓
สารบัญภาพ .....	๔
บทที่ ๑ .....	๑
บทนำ .....	๑
1.1 ความสำคัญและที่มา.....	๑
1.2 วัตถุประสงค์.....	๒
1.3 ขอบเขตการศึกษา .....	๒
1.4 แผนการดำเนินโครงการ .....	๓
1.5 ประโยชน์ที่คาดว่าจะได้รับ .....	๓
บทที่ ๒ .....	๔
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง .....	๔
แนวคิด ทฤษฎี เทคโนโลยีและงานวิจัยที่เกี่ยวข้อง .....	๔
2.1 Web Application.....	๔
2.2 Hypertext Transfer Protocol (HTTP) .....	๕
2.3 Web Service .....	๘
2.4 Open Network Video Interface Forum (ONVIF).....	๙

2.5 Session .....	11
2.6 Database .....	11
2.7 Internet Protocol Camera (IP Camera) .....	13
2.8 ภาษา Golang.....	14
2.9 ภาษา HTML .....	15
2.10 ภาษา CSS .....	17
2.11 ภาษา JavaScript .....	19
2.12 JavaScript Object Notation (JSON) .....	20
2.13 งานวิจัยที่เกี่ยวข้อง .....	21
บทที่ 3 .....	22
วิธีดำเนินงานโครงงาน .....	22
3.1 ศึกษาปัญหาและความต้องการของระบบ .....	22
3.2 วิเคราะห์และออกแบบระบบ .....	23
3.2.1 สถาปัตยกรรมของระบบ (System Architecture) .....	23
3.2.2 การวิเคราะห์และออกแบบ API .....	25
3.2.3 การอธิบาย Flow Chart การทำงานของระบบ .....	26
3.2.4 การออกแบบหน้าเว็บไซต์ .....	37
3.3 เครื่องมือในการพัฒนาระบบ .....	38
บทที่ 4 .....	39
ผลการดำเนินงาน .....	39
4.1 วิธีการทดสอบการพัฒนาระบบ .....	39
4.1.1 ทดสอบการติดตั้งโปรแกรม NONVIF .....	39

4.1.2 ทดสอบเกี่ยวกับระบบ Login, Logout, Signup และ Forgot Password.....	43
4.1.3 ทดสอบเกี่ยวกับการเรียกสตรีมภาพวิดีโอจากกล้องไอพี .....	50
4.1.4 ทดสอบเกี่ยวกับการเรียกข้อมูลเมตาจากกล้องไอพี .....	60
4.1.5 ทดสอบส่วนอื่น ๆ.....	65
4.2 การพัฒนาระบบ.....	67
4.3 การทดสอบเพิ่มเติม .....	71
บทที่ 5 .....	74
สรุปผลและข้อเสนอแนะ .....	74
5.1 สรุปผลการดำเนินงาน .....	74
5.2 ข้อเสนอแนะเพิ่มเติม .....	75
5.2.1 เกี่ยวกับระบบเชิร์ฟเวอร์.....	75
5.2.2 เกี่ยวกับการออกแบบ .....	75
5.2.3 เกี่ยวกับการพัฒนาเพิ่มเติม.....	75
เอกสารอ้างอิง .....	76
ประวัติของผู้เขียน .....	77

## สารบัญตาราง

หน้า

ตารางที่ 1 ตัวอย่างคำสั่งของโปรโตคอล HTTP .....	6
ตารางที่ 2 ตัวอย่างรหัสสถานะการทำงานของ HTTP .....	6
ตารางที่ 3 ตารางอธิบายของผู้ใช้งานทั่วไป.....	23
ตารางที่ 4 แสดงรายการของ API ทั้งหมดภายในโครงงานนี้.....	25
ตารางที่ 5 อุปกรณ์ฮาร์ดแวร์ที่ใช้ในการพัฒนาระบบ.....	38
ตารางที่ 6 ซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ .....	38
ตารางที่ 7 ตารางแสดงค่าก้อนการทดสอบ.....	71
ตารางที่ 8 ความละเอียด 480p : Standard Definition(SD).....	72
ตารางที่ 9 ความละเอียด 720p : High Definition(HD).....	72
ตารางที่ 10 ความละเอียด 1080p : Full High Definition(Full HD).....	73
ตารางที่ 11 ความละเอียด 2160p : Ultra High Definition(UHD หรือ 4k).....	73

## สารบัญภาพ

หน้า

ภาพที่ 1 ระยะเวลาในการวางแผนและพัฒนาโครงการ .....	3
ภาพที่ 2 ภาพตัวอย่างการทำงานของ Web Application .....	5
ภาพที่ 3 ภาพกระบวนการของ HTTP .....	8
ภาพที่ 4 ภาพตัวอย่างของ ONVIF Profile .....	10
ภาพที่ 5 ภาพรวมของ SQL และอื่น ๆ .....	13
ภาพที่ 6 ภาพรวมของตัวอย่างกล้องไอพี .....	14
ภาพที่ 7 ตัวอย่างการใช้งานภาษา Golang .....	15
ภาพที่ 8 โครงสร้างของภาษา HTML .....	16
ภาพที่ 9 ตัวอย่างการใช้งานภาษา HTML .....	17
ภาพที่ 10 โครงสร้างของภาษา CSS .....	18
ภาพที่ 11 ตัวอย่างการใช้งานภาษา CSS .....	18
ภาพที่ 12 ตัวอย่างการใช้งานภาษา JavaScript .....	19
ภาพที่ 13 ตัวอย่างการแสดงผลข้อมูลรูปแบบ JSON .....	20
ภาพที่ 14 Use Case Diagram .....	22
ภาพที่ 15 สถาปัตยกรรมโดยรวมของระบบ .....	23
ภาพที่ 16 ผังการทำงานการเข้าสู่ระบบ .....	27
ภาพที่ 17 ผังการทำงานการลงทะเบียนผู้ใช้งาน .....	29
ภาพที่ 18 ผังการทำงานการกรอกรหัสผ่านใหม่สำหรับผู้ลืมรหัส .....	31
ภาพที่ 19 ผังการทำงานภายในระบบ .....	32
ภาพที่ 20 ผังการทำงานภายในหน้า Cameras .....	34
ภาพที่ 21 ผังการทำงานภายในหน้า Dashboard .....	36
ภาพที่ 22 แผนผังส่วนของเว็บแอปพลิเคชัน .....	37
ภาพที่ 23 หน้า Github ของ Nuntakorn-Buu/NONVIF .....	39
ภาพที่ 24 ส่วนของการอธิบาย Component หน้า Github ของ Nuntakorn-Buu/NONVIF .....	40
ภาพที่ 25 ส่วนของการอธิบาย Installation หน้า Github ของ Nuntakorn-Buu/NONVIF .....	41
ภาพที่ 26 ภาพของขั้นตอนที่ 1 ในการใช้คำสั่งที่ใช้ในการ Clone NONVIF repository .....	41
ภาพที่ 27 ภาพของขั้นตอนที่ 2 ในการใช้คำสั่ง cd NONVIF .....	42
ภาพที่ 28 ภาพของขั้นตอนที่ 3 ในการใช้คำสั่ง go get .....	42

ภาพที่ 29 ภาพหลังจากการใช้คำสั่ง go run .....	42
ภาพที่ 30 ภาพการเปิด Web Browser แล้วทำการพิมพ์ http://localhost:9000/ .....	43
ภาพที่ 31 ภาพการกรอกข้อมูลเพื่อทำการ Login เข้าสู่ระบบ.....	43
ภาพที่ 32 ภาพของการที่ทำการ Login แล้วไม่สำเร็จ .....	44
ภาพที่ 33 ภาพของการที่ทำการ Login แล้วสำเร็จ .....	44
ภาพที่ 34 ภาพในหน้า Sign up เพื่อทำการลงทะเบียน .....	45
ภาพที่ 35 ภาพของการกรอกข้อมูลผิดแล้วจะมีข้อความแจ้งเตือน .....	45
ภาพที่ 36 ภาพในหน้า Forgot Password .....	46
ภาพที่ 37 ภาพการใส่ Email ผิดในหน้า Forgot Password .....	46
ภาพที่ 38 ภาพหลังการใส่ Email ถูกต้องในหน้า Forgot Password.....	47
ภาพที่ 39 ภาพในหน้า Gmail ที่แสดงการส่งรหัสในการเปลี่ยน Password.....	47
ภาพที่ 40 ภาพการใส่รหัส Verification Code ที่ถูกต้อง .....	48
ภาพที่ 41 ภาพในหน้าการสร้าง Password ใหม่ .....	48
ภาพที่ 42 ภาพการใส่รหัส Verification Code ผิด.....	49
ภาพที่ 43 แอปพลิเคชัน IP Webcam บน Google Play.....	51
ภาพที่ 44 ภาพแสดงแอปพลิเคชัน IP Webcam ที่ติดตั้งเสร็จแล้วที่หน้าจอสมาร์ทโฟน .....	51
ภาพที่ 45 ภาพเมื่อเข้าสู่แอปพลิเคชัน IP Webcam .....	52
ภาพที่ 46 ภาพการเปิดใช้งานการสนับสนุน ONVIF .....	52
ภาพที่ 47 ภาพการตั้งค่ารหัสผ่านของกล้องไอโอพีalongบน IP Webcam .....	53
ภาพที่ 48 ภาพตัวอย่างการใส่รหัสผ่านของกล้องไอโอพีalongบน IP Webcam .....	53
ภาพที่ 49 การเปิดใช้งานการจำลองกล้องไอโอพีด้วย IP Webcam.....	54
ภาพที่ 50 ภาพเมื่อแสดงบนสมาร์ทโฟนเมื่อการจำลองกล้องกำลังทำงาน .....	54
ภาพที่ 51 ภาพการหยุดการจำลองกล้องไอโอพี .....	55
ภาพที่ 52 ภาพหน้า Cameras.....	56
ภาพที่ 53 การเพิ่มที่อยู่ไอโอพีของกล้องไอโอพี.....	56
ภาพที่ 54 ภาพการเพิ่มที่อยู่กล้องไอโอพีสำเร็จ .....	56
ภาพที่ 55 ภาพการเพิ่มที่อยู่ของกล้องไอโอพีมากกว่า 1 ไอโอพี.....	57
ภาพที่ 56 ภาพกล่องข้อความสำหรับใส่รหัสผ่านของกล้องไอโอพี .....	57
ภาพที่ 57 ภาพการคลิกเพื่อรับชมกล้องตัวที่ 1.....	58
ภาพที่ 58 ภาพการคลิกเพื่อรับชมกล้องตัวที่ 2.....	59
ภาพที่ 59 ภาพการเรียกดูภาพจากกล้องไอโอพีไม่สำเร็จ .....	59

ภาพที่ 60 ภาพแสดงปุ่ม Remove.....	60
ภาพที่ 61 ภาพแสดงข้อมูลจากฟังก์ชัน GetDeviceInformation.....	60
ภาพที่ 62 ภาพแสดงข้อมูลจากฟังก์ชัน GetUser.....	61
ภาพที่ 63 ภาพแสดงข้อมูลจากฟังก์ชัน GetSystemDateAndTime.....	61
ภาพที่ 64 ภาพแสดงข้อมูลจากฟังก์ชัน GetCapabilities.....	62
ภาพที่ 65 ภาพแสดงข้อมูลจากฟังก์ชัน GetHostname.....	62
ภาพที่ 66 ภาพแสดงข้อมูลจากฟังก์ชัน GetNetworkProtocols .....	63
ภาพที่ 67 ภาพแสดงข้อมูลจากฟังก์ชัน GetDiscoveryMode .....	63
ภาพที่ 68 ภาพแสดงข้อมูลจากฟังก์ชัน GetServiceCapabilities .....	64
ภาพที่ 69 ภาพการแสดงของหน้า About.....	65
ภาพที่ 70 ภาพการแสดงของหน้า Portfolio .....	66
ภาพที่ 71 ภาพการแสดงของหน้า 404 Not Fount .....	66
ภาพที่ 72 ภาพของการเขียนเรียกแสดงข้อมูลเมต้าแบบเก่า ในส่วนของ JavaScript.....	67
ภาพที่ 73 ภาพของการเขียนเรียกแสดงข้อมูลเมต้าแบบเก่า ในส่วนของ HTML .....	68
ภาพที่ 74 ภาพของการเขียนเรียกแสดงข้อมูลเมต้าแบบใหม่ ในส่วนของ JavaScript.....	68
ภาพที่ 75 ภาพของการเขียนเรียกแสดงข้อมูลเมต้าแบบใหม่ ในส่วนของ HTML .....	69
ภาพที่ 76 ภาพแสดงการเรียกข้อมูลแบบใหม่จากฟังก์ชัน GetHostname, GetNetworkProtocols, GetDeviceInformation และ GetUser .....	69
ภาพที่ 77 ภาพแสดงการเรียกข้อมูลแบบใหม่จากฟังก์ชัน GetDiscoveryMode และ GetSystemDateAndTime .....	70
ภาพที่ 78 ภาพแสดงการเรียกข้อมูลแบบใหม่จากฟังก์ชัน GetServiceCapabilities .....	70

## บทที่ 1

### บทนำ

#### 1.1 ความสำคัญและที่มา

ในยุคสมัยที่เทคโนโลยีมีการพัฒนาอย่างรวดเร็วเพิ่มขึ้น การดูแลและจัดการระบบรักษาความปลอดภัยก็ได้รับความสนใจมากขึ้น เช่นกัน การใช้งานกล้องไอพีเพื่อจัดการกับระบบกล้องไอพีในปัจจุบันต้องสามารถสตรีมวิดีโอและข้อมูลเมต้าไปยังแพลตฟอร์มที่แตกต่างกันได้อย่างรวดเร็วและมีประสิทธิภาพ การจัดการกล้อง IP, การสตรีมวิดีโอ และการส่งข้อมูลเมต้าผ่านเว็บไซต์เป็นสิ่งสำคัญในหลายสาขาอุตสาหกรรม เช่น ระบบกล้องวงจรปิด (CCTV), การตรวจจับและติดตามวัตถุ, การตรวจสอบความปลอดภัย และอื่น ๆ เพื่อตอบสนองต่อความต้องการในการตรวจจับการบุกรุกและป้องกันความเสี่ยงที่เกี่ยวข้องกับความปลอดภัย แต่ปัญหาที่พบคือบริษัทผู้ผลิตกล้องวงจรปิดใช้โปรโตคอลของตนเองในการควบคุมกล้อง ซึ่งทำให้ผู้ใช้ที่ต้องการใช้กล้องไอพีต่างยื่ห้อต้องใช้ระบบที่แตกต่างกันในการจัดการกล้องตามผู้ผลิตกล้องนั้น ๆ

ดังนั้นการเลือกใช้กล้องที่มีมาตรฐานprotoคอล ONVIF เป็นสิ่งสำคัญ เนื่องจากมาตรฐานนี้รองรับกล้องไอพีมากถึง 80 เปอร์เซ็นต์ในตลาด การใช้มาตรฐาน ONVIF จะทำให้การจัดการกล้องไอพีสามารถสื่อสารกับแพลตฟอร์มที่แตกต่างกันได้อย่างมีประสิทธิภาพโดยไม่จำเป็นต้องพึ่งซอฟต์แวร์ของผู้ผลิตกล้องแต่ละยี่ห้อ การสร้างซอฟต์แวร์ที่รองรับการเรียกกล้องด้วยมาตรฐาน ONVIF จะช่วยลดปัญหาการติดตั้งระบบที่แตกต่างกันตามผู้ผลิตกล้องนั้น ๆ

โครงการนี้จึงมุ่งเน้นการประยุกต์ใช้protoคอล ONVIF เพื่อรองรับการจัดการกล้องไอพี และการสตรีมวิดีโอผ่านเว็บไซต์ เพื่อตอบสนองต่อความต้องการของผู้ใช้ในการนำไปใช้ในทางอุตสาหกรรมหรือผู้ใช้ทั่วไปที่ต้องการจัดการกล้อง IP, การสตรีมวิดีโอ และการส่งข้อมูลเมต้าผ่านเว็บไซต์อย่างมีประสิทธิภาพ โดยทำให้สามารถเชื่อมต่อรับส่งข้อมูลกับเว็บไซต์ได้อย่างสะดวก โดยโครงการนี้มีการพัฒนาโดยนำแนวคิดรวมถึงเทคโนโลยีที่เกี่ยวข้อง เช่น การทำ Web application ด้วย Golang, การสตรีมวิดีโอ, การจัดการข้อมูลเมต้าด้วยรูปแบบข้อมูล JSON รวมถึงการนำ ONVIF Protocol มาประยุกต์ใช้เพื่อให้เกิดประโยชน์ในการพัฒนาระบบซอฟต์แวร์เพื่อสนับสนุนการดำเนินธุรกิจประเภทบริษัทความปลอดภัยอย่างมีประสิทธิภาพ โครงการนี้สามารถเป็นซอฟต์แวร์ที่ช่วยให้ผู้ใช้ได้ความสะดวกสบายและประหยัดเวลาในการจัดการกล้องวงจรปิดและการสตรีมวิดีโอผ่านเว็บไซต์ได้อย่างมีประสิทธิภาพ นอกจากนี้ยังสามารถนำร่องศึกษานำไปเป็นแนวทางในการพัฒนาและปรับปรุงระบบซอฟต์แวร์อื่น ๆ ที่มีลักษณะคล้ายคลึงกันได้ต่อไป

## 1.2 วัตถุประสงค์

1. เพื่อศึกษาและพัฒนาซอฟต์แวร์ที่สามารถสนับสนุนการจัดการกล้องไอพี ได้อย่างมีประสิทธิภาพ เช่น การเพิ่มหรือลบที่อยู่ IP ของกล้องและการเรียกคุ้มภารการสตรีมจากกล้องที่ได้ถูกเพิ่มเข้าสู่ระบบ
2. เพื่อสนับสนุนและอำนวยความสะดวกให้ผู้ใช้สามารถแสดงภาพวิดีโอและข้อมูลเมตาจากกล้อง IP ผ่านหน้าเว็บไซต์
3. เพื่อศึกษาและเข้าใจกระบวนการรับส่งข้อมูลเมตาโดยใช้รูปแบบมาตรฐานของ ONVIF
4. เพื่อให้เป็นแนวทางและแรงบันดาลใจในการพัฒนาต่อไปสำหรับผู้ที่สนใจและต้องการใช้ข้อมูลจากโครงงานนี้ในอนาคต

## 1.3 ขอบเขตการศึกษา

1. การออกแบบระบบ: เน้นการออกแบบระบบสำหรับการจัดการกล้องไอพี การสตรีมวิดีโอและข้อมูลเมตาผ่านเว็บไซต์
  - a. เพื่อให้สามารถรับส่งข้อมูลกล้องไอพี ด้วยมาตรฐาน HTTP
  - b. เพื่อให้สามารถรับส่งข้อมูลเมตาจากกล้องไอพีด้วยมาตรฐาน ONVIF โดยจะรองรับทั้งหมด 8 ฟังก์ชัน คือ 1) GetDeviceInformation, 2) GetUser, 3) GetSystemDateAndTime, 4) GetCapabilities, 5) GetHostname, 6) GetNetworkProtocols, 7) GetDiscoveryMode และ 8) GetServiceCapabilities
  - c. เพื่อให้สามารถจัดการควบคุมการ เพิ่มหรือลด ที่อยู่กล้องไอพีผ่านหน้าเว็บไซต์
2. ฐานข้อมูล: โครงงานนี้เลือกใช้ SQLite เป็นฐานข้อมูล เพื่อจัดเก็บข้อมูลเกี่ยวกับที่อยู่กล้องไอพี และข้อมูลของผู้ใช้ที่จะเข้าใช้งานภายในระบบ
3. ภาษาโปรแกรม: โครงงานนี้ใช้ GoLang เป็นภาษาหลักในการพัฒนาโครงงาน
4. การพัฒนาเว็บ: หน้าเว็บเลือกพัฒนาด้วยภาษา HTML, CSS และ JavaScript

## 1.4 แผนการดำเนินโครงการ

TASK	July 2022	August 2022	September 2022	October 2022	November 2022	January 2023	February 2023	March 2023
Stage Gate-1 : website page creation								
1.ศึกษาเกี่ยวกับภาษา Golang เพื่อใช้ในการสร้าง server และ Func อื่นๆที่เกี่ยวข้องกับหน้าตาในโครงการ								
2.ศึกษาเกี่ยวกับภาษา HTML, CSS และ Framework ที่เกี่ยวข้อง เพื่อใช้เป็นพื้นฐานในการสร้างหน้าเว็บไซต์								
Stage Gate-2 : building a security system								
1.ศึกษาเกี่ยวกับ SQLite เพื่อใช้ในการทำฐานข้อมูลเพื่อจัดเก็บข้อมูลของผู้ที่จะเข้าใช้งาน								
2.ศึกษาเกี่ยวกับระบบบันทึกตัวตนในการเข้าใช้งาน								
3.ศึกษาเกี่ยวกับการทำ sessions เพื่อรักษา session ของผู้ใช้งาน								
Stage Gate-3 : Video streaming and metadata with ONVIF								
1.ศึกษาเกี่ยวกับการสร้าง API สำหรับการเรียกการสตรีมวิดีโอและ การเรียกขอข้อมูลตามสเปคของ ONVIF								
2.ศึกษาเกี่ยวกับ ONVIF protocol เพื่อใช้ในการรับขอข้อมูลมา จากกล้องไอพี								
3.ศึกษาเกี่ยวกับ JavaScipt เพื่อแปลงข้อมูลที่ได้รับมาจาก API ONVIF ให้อยู่ในรูปแบบของ JSON								
4.จัดการเพิ่มการเรียกการสตรีมวิดีโอด้วยข้อมูลมาต้าที่หน้าเว็บไซต์								
5.เพิ่มความสามารถในการจัดการกล้องไอพี ในการเพิ่มหรือลดกล้องไอพี การล็อกอัตโนมัติและการสตรีมขอข้อมูลตามต้องที่สนใจ								
Stage Gate-4 : Bug fixes and improvements								
1.ทดสอบระบบโดยรวมและจัดเก็บบันทึกข้อมูล								
2.สรุปผลการทดลอง								
3.จัดทำรายงานสรุปผลรายงานการท่าโครงการ								

ภาพที่ 1 ระยะเวลาในการวางแผนและพัฒนาโครงการ

อ้างอิงจากภาพที่ 1 แสดงแผนการดำเนินงานประจำเดือนต่อไปนี้

- 1) เรียนรู้และพัฒนาการสร้างหน้าเว็บไซต์
- 2) สร้างและพัฒนาเกี่ยวกับการทำฐานข้อมูลเพื่อจัดเก็บข้อมูลผู้ที่จะเข้าใช้เว็บไซต์เพื่อสร้างความปลอดภัยต่อระบบในระดับหนึ่ง
- 3) เรียนรู้และพัฒนาเกี่ยวกับการสร้าง API เพื่อใช้ในการเรียกดูภาพการสตรีมและข้อมูลมาต้าด้วย ONVIF Protocol จากกล้องไอพี และการจัดการกับกล้องไอพีต่างๆ เช่น การเพิ่มหรือลบกล้องไอพี
- 4) การทดสอบระบบโดยรวมแล้วบันทึกผล การสรุปผลการทดลองและการเขียนเล่มรายงานสรุปผลของโครงการ

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. โครงการสามารถเพิ่มที่อยู่ของกล้องไอพีให้กับผู้ใช้งานที่มีกล้องไอพีมากกว่า 1 อุปกรณ์
2. โครงการสามารถทำการสตรีมข้อมูลวิดีโอและข้อมูลมาต้าไปยังหน้าเว็บไซต์
3. หน้าเว็บไซต์สามารถทำการจัดการเพิ่มหรือลดและเลือกคุณภาพจากกล้องไอพีได้

## บทที่ 2

### ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้มีการกล่าวถึงเนื้อหา ทฤษฎี และเทคโนโลยีที่เกี่ยวข้องกับการทำเนินโครงการในครั้งนี้ และจะเป็นการกล่าวถึงงานวิจัยที่เกี่ยวข้องโดยจะมุ่งเน้นไปที่งานที่เกี่ยวข้องกับการสตรีมภาพวิดีโอ และข้อมูลเมต้าของกล้องไอพีไปยังหน้าเว็บไซต์ และการจัดเก็บข้อมูลผู้ที่เข้าใช้งานเว็บไซต์ และข้อมูลที่อยู่กล้องไอพีลิงในฐานข้อมูล โดยเนื้อหาที่ผู้วิจัยศึกษามีดังนี้

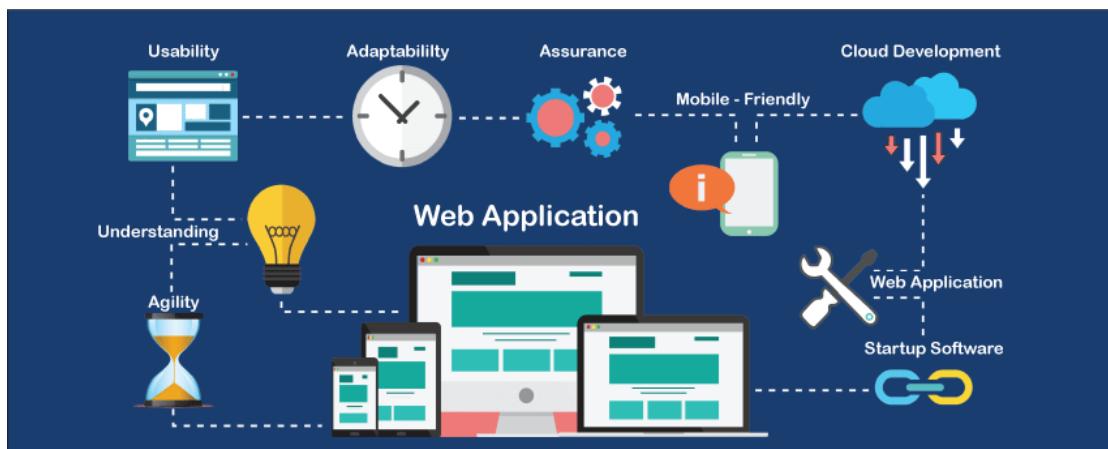
1. Web Application
2. HTTP
3. Web Service
4. Open Network Video Interface Forum (ONVIF)
5. Session
6. Database
7. Internet Protocol Camera (IP Camera)
8. ภาษา Golang
9. ภาษา HTML
10. ภาษา CSS
11. ภาษา JavaScript
12. JavaScript Object Notation (JSON)
13. งานวิจัยที่เกี่ยวข้อง

### แนวคิด ทฤษฎี เทคโนโลยีและงานวิจัยที่เกี่ยวข้อง

#### 2.1 Web Application

เว็บแอปพลิเคชัน (Web Application) เป็นแอปพลิเคชันที่ถูกสร้างขึ้นโดยใช้เทคโนโลยีเว็บเบสต์ (Web-based) และถูกเข้าถึงผ่านทางเว็บเบราว์เซอร์บนอินเทอร์เน็ต โดยผู้ใช้สามารถเข้าถึงและใช้งานได้ผ่านอุปกรณ์ที่เชื่อมต่อกับเน็ตเวิร์ค เว็บแอปพลิเคชันนั้นสามารถให้ผู้ใช้ทำการกรอกแบบฟอร์ม ทำการคำนวณ ดำเนินการธุกรรมการชำระเงิน แสดงผลข้อมูล หรือดำเนินการตามธุกรรมอื่น ๆ ได้ เท่าที่เกี่ยวข้องกับเทคโนโลยีเว็บ การสร้างเว็บแอปพลิเคชันมักใช้ภาษาโปรแกรมมิ่งเว็บต่าง ๆ เช่น HTML (HyperText Markup Language), CSS (Cascading Style Sheets), และ

JavaScript เพื่อสร้างโครงสร้างของเว็บและกำหนดรูปแบบการแสดงผล นอกจากนี้ เว็บแอปพลิเคชันอาจใช้ภาษาโปรแกรมมิ่งเว็บผู้เชิญฟล็อก เช่น Python, Ruby, PHP, Golang หรือภาษาอื่น ๆ ใน การประมวลผลและจัดการกับข้อมูล เว็บแอปพลิเคชันสามารถให้ประสบการณ์ผู้ใช้แบบพิเศษได้ โดยสามารถเข้าถึงผ่านอินเทอร์เฟซกราฟิก (Graphical User Interface) ที่สวยงามและใช้งานง่าย รวมถึงสามารถปรับปรุงและอัปเดตได้อย่างต่อเนื่องผ่านการอัปเดตเวอร์ชันใหม่ ๆ และการพัฒนาเพิ่มเติม เว็บแอปพลิเคชันมีความสามารถในการทำงานแบบ real-time และอัปเดตข้อมูลในเวลาจริง โดยใช้เทคโนโลยีที่เชื่อมต่อกับเซิร์ฟเวอร์ในเบื้องหลัง ซึ่งช่วยให้ผู้ใช้ได้รับข้อมูลอัปเดตและการแจ้งเตือนในเวลาที่เกิดขึ้น เว็บแอปพลิเคชันได้รับความนิยมอย่างมากในการพัฒนาและใช้งานใน หลากหลายรูปแบบ เช่น เว็บไซต์อีคอมเมิร์ซ, เครื่องมือการบริหารจัดการ, เว็บแอปพลิเคชันสังคม ออนไลน์ และแอปพลิเคชันบนอุปกรณ์เคลื่อนที่



ภาพที่ 2 ภาพตัวอย่างการทำงานของ Web Application

(ที่มา: <https://www.javatpoint.com/web-application>)

## 2.2 Hypertext Transfer Protocol (HTTP)

Hypertext Transfer Protocol หรือ (HTTP) เป็นโปรโตคอลที่ใช้ในการสื่อสารและ แลกเปลี่ยนข้อมูลระหว่างเว็บเบราว์เซอร์และเว็บเซิร์ฟเวอร์ในรูปแบบของข้อความ โดยทั่วไปแล้วใช้ พอร์ต 80 เป็นพอร์ตสำหรับการสื่อสารของ HTTP ซึ่งถูกออกแบบมาเพื่อใช้ในการร้องขอและ ตอบสนองสำหรับเว็บไซต์และแอปพลิเคชันเว็บ โดยการร้องขอจะถูกส่งจากเว็บเบราว์เซอร์ไปยังเว็บ เซิร์ฟเวอร์เพื่อขอข้อมูล และเว็บเซิร์ฟเวอร์จะตอบสนองกลับด้วยข้อมูลที่ร้องขอ ซึ่งจะถูกส่งกลับไปยัง

เว็บเบราว์เซอร์เพื่อแสดงผลหรือดำเนินการต่อไป HTTP มีลักษณะการทำงานแบบส่งคำขอ (request) และรับคำตอบ (response) โดยคำขอจะประกอบด้วยเส้นทาง (method) เช่น GET, POST, PUT, DELETE เป็นต้น และข้อมูลเสริม เช่น ส่วนหัว (header) และเนื้อหา (content) ซึ่งเป็นข้อมูลที่ส่งพร้อมกับคำขอ สรุปได้ว่า HTTP เป็นโปรโตคอลที่ใช้กันอย่างแพร่หลายในการสื่อสารในอินเทอร์เน็ต และเป็นพื้นฐานสำคัญที่อยู่ในฐานของการทำงานของเว็บไซต์และแอปพลิเคชันออนไลน์ทั่วไปในปัจจุบัน

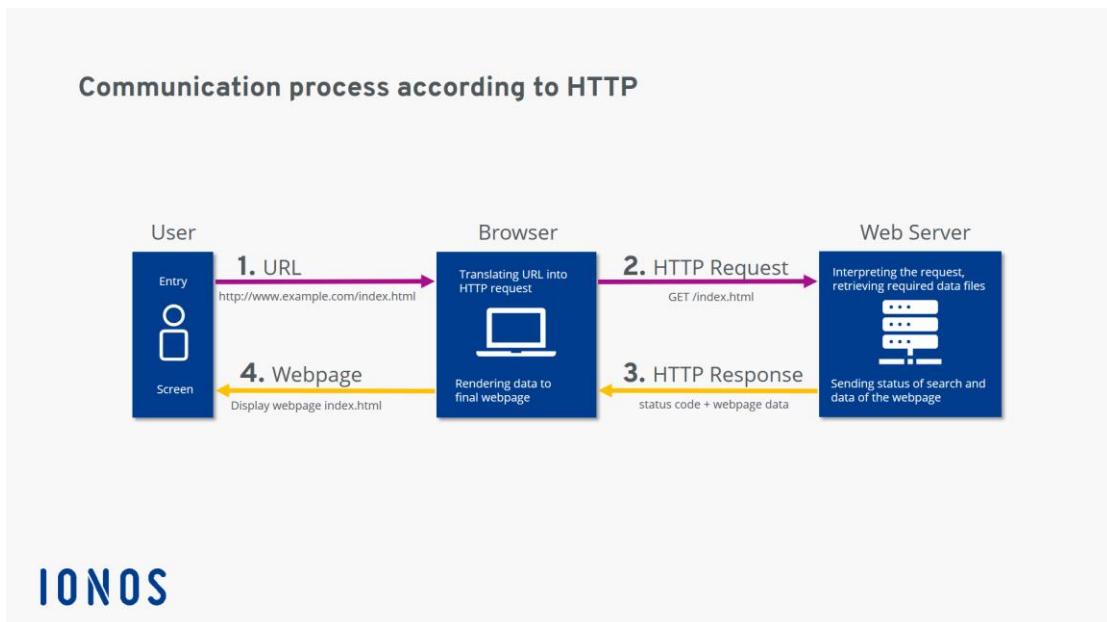
ตารางที่ 1 ตัวอย่างคำสั่งของโปรโตคอล HTTP

คำสั่ง	รายละเอียด
GET	ใช้สำหรับร้องขอข้อมูลจากเซิร์ฟเวอร์ โดยระบุ URI (Uniform Resource Identifier) ของข้อมูลที่ต้องการ
POST	ใช้สำหรับส่งข้อมูลจากไคลเอนต์ไปยังเซิร์ฟเวอร์ โดยใส่ข้อมูลในส่วนของเนื้อหา (body) ของข้อความ HTTP
PUT	ใช้สำหรับอัปโหลด หรือ อัปเดตข้อมูลที่อยู่ในเซิร์ฟเวอร์
DELETE	ใช้สำหรับลบข้อมูลที่อยู่ในเซิร์ฟเวอร์

ตารางที่ 2 ตัวอย่างรหัสสถานะการทำงานของ HTTP

ประเภท	รหัสสถานะ	รายละเอียด
Successful	200 OK	การร้องขอเรียกสมบูรณ์และได้รับการตอบสนองที่ถูกต้อง
	201 Created	แสดงว่าการร้องขอสร้างแหล่งข้อมูลใหม่เรียกสมบูรณ์

	202 Accepted	ได้รับคำสั่งให้ทำงานเรียบร้อย แต่ไม่ต้องมีการตอบกลับ
	204 NO CONTENT	การร้องขอเสร็จสมบูรณ์แล้ว ไม่มีเนื้อหาที่จะส่งกลับ
Redirection	301 Moved Permanently	แสดงว่าที่อยู่ของข้อมูลถูกเปลี่ยนไปและจะต้องร้องขอที่อยู่ใหม่
	302 Moved Temporarily	แสดงว่าที่อยู่ของข้อมูลถูกเปลี่ยนไปชั่วคราวและจะต้องร้องขอที่อยู่ใหม่
Client Error	400 Bad Request	การร้องขอไม่ถูกต้องหรือไม่สามารถตีความได้
	401 Unauthorized	การร้องขอต้องมีการรับรองความถูกต้องก่อนที่จะเข้าถึงข้อมูล
	403 Forbidden	เซิร์ฟเวอร์ไม่อนุญาตให้ใช้งาน หรือคุณลénontมีสิทธิในการใช้งานไม่เพียงพอ
	404 Not Found	ไม่พบข้อมูลที่ร้องขอ
Server Error	500 Internal Server Error	เกิดข้อผิดพลาดที่ไม่ระบุได้ในเซิร์ฟเวอร์
	501 Not Implemented	เซิร์ฟเวอร์ไม่รองรับคำสั่งที่สั่งไป
	502 Bad Gateway	Proxy Server รับคำสั่งไม่ถูกต้องจากเว็บเซิร์ฟเวอร์
	503 Service Unavailable	เซิร์ฟเวอร์ไม่สามารถตอบสนองการร้องขอได้ในขณะนั้น



ภาพที่ 3 ภาพกระบวนการของ HTTP

## 2.3 Web Service

เว็บเซอร์วิส (Web Service) เป็นโครงสร้างที่ใช้ในการสื่อสารและแลกเปลี่ยนข้อมูลระหว่างแอปพลิเคชันต่าง ๆ ผ่านทางเครือข่ายอินเทอร์เน็ต โดยใช้โปรโตคอลและภาษามาตราฐานเพื่อทำให้แอปพลิเคชันสามารถเชื่อมต่อและทำงานร่วมกันได้อย่างราบรื่น เว็บเซอร์วิสส่วนใหญ่ใช้โปรโตคอล HTTP (Hypertext Transfer Protocol) ในการสื่อสาร และใช้ภาษา XML (eXtensible Markup Language) หรือ JSON (JavaScript Object Notation) เพื่อรูปแบบข้อมูลที่จะถูกส่งและรับรู้ในรูปแบบที่เป็นระเบียบ เว็บเซอร์วิสมีความยืดหยุ่นและความสามารถในการทำงานร่วมกันกับแอปพลิเคชันอื่น ๆ ที่อยู่ในพื้นที่ต่าง ๆ หรือแม้แต่ในองค์กรที่แตกต่างกัน นอกจากนี้ เว็บเซอร์วิสยังช่วยให้สามารถเชื่อมต่อและแลกเปลี่ยนข้อมูลกับระบบฐานข้อมูล ระบบการชำระเงิน และบริการอื่น ๆ ได้ เว็บเซอร์วิสเป็นเทคโนโลยีสำคัญในการสร้างและพัฒนาระบบอินเทอร์เน็ตแบบเครื่องอื่น ๆ เช่น บริการเว็บแอปพลิเคชัน (Web Application), คลาวด์คอมพิวติ้ง (Cloud Computing), และเครือข่ายสัญญาณ (Internet of Things) เพื่อให้การสื่อสารและการทำงานร่วมกันระหว่างแอปพลิเคชันนั้นสามารถเกิดขึ้นได้อย่างมีประสิทธิภาพ

## 2.4 Open Network Video Interface Forum (ONVIF)

Open Network Video Interface Forum (ONVIF) Protocol เป็นมาตรฐานโปรโตคอลที่ถูกพัฒนาขึ้นโดยองค์กร ONVIF เพื่อสนับสนุนการสื่อสารและการทำงานร่วมกันระหว่างอุปกรณ์วิดีโอดิจิตอล (IP-based) ที่ใช้ในระบบธุรกษาความปลอดภัยและการดูแลรักษาความปลอดภัย โดยปกติแล้ว ONVIF Protocol ถูกใช้ในกล้องไอพี (IP Camera), NVR (Network Video Recorder) และอุปกรณ์ระบบธุรกษาความปลอดภัยอื่น ๆ เพื่อให้สามารถเชื่อมต่อและทำงานร่วมกันได้อย่างราบรื่น ONVIF Protocol มีประโยชน์ที่สำคัญในการทำงานร่วมกันระหว่างอุปกรณ์จากผู้ผลิตที่แตกต่างกัน โดยสามารถรองรับความสามารถเบื้องต้น เช่น การส่งสัญญาณวิดีโอและเสียง การควบคุมการปรับแต่งภาพ การจัดการการบันทึกวิดีโอ และการจัดการอุปกรณ์รอบข้างอื่น ๆ การใช้งาน ONVIF Protocol ช่วยให้ผู้ใช้สามารถเลือกใช้อุปกรณ์และระบบธุรกษาความปลอดภัยที่ต้องการ โดยไม่ต้องพึ่งพาบริษัทผู้ผลิตเดียว กัน และเพิ่มความยืดหยุ่นในการออกแบบและการเชื่อมต่อระหว่างอุปกรณ์ในระบบธุรกษาความปลอดภัยอินเทอร์เน็ต

ONVIF Profile เป็นคำอธิบายแนวทางการใช้งานของมาตรฐาน ONVIF Protocol สำหรับอุปกรณ์วิดีโอดิจิตอล (IP-based) ที่ใช้ในระบบธุรกษาความปลอดภัยและการดูแลรักษาความปลอดภัย ONVIF Profile จะกำหนดคุณสมบัติและฟังก์ชันที่ต้องมีสำหรับอุปกรณ์แต่ละประเภท เช่น Profile S ใช้สำหรับกล้องไอพี (IP Camera), Profile T ใช้สำหรับอุปกรณ์ระบบธุรกษาความปลอดภัยแบบ IP ที่รองรับการถ่ายทอดสด (live streaming), ระบบการบันทึกภาพ (media recording), และอื่น ๆ การใช้งาน ONVIF Profile ช่วยให้ผู้ใช้สามารถเลือกใช้อุปกรณ์และระบบธุรกษาความปลอดภัยที่ต้องการโดยมีความสอดคล้องกับคุณสมบัติและการทำงานร่วมกันตามคำแนะนำของ ONVIF Profile ที่เลือกใช้ เช่น Profile S หรือ Profile T การใช้งาน ONVIF Profile ช่วยลดความซับซ้อนในการติดตั้งและการปรับแต่งระบบธุรกษาความปลอดภัย และช่วยให้ผู้ใช้สามารถเลือกใช้อุปกรณ์จากผู้ผลิตที่แตกต่างกัน โดยมีการรับรองว่าอุปกรณ์ที่ใช้งานร่วมกันสามารถทำงานได้อย่างมีประสิทธิภาพและรองรับมาตรฐานร่วมกันได้

## ONVIF Profiles



ภาพที่ 4 ภาพตัวอย่างของ ONVIF Profile

(ที่มา: <https://cctvapp.net/en/2022/09/09/what-is-onvif-learn-about-onvif-standard-in-surveillance-cameras/>)

ใน ONVIF Protocol นั้นมี XML, SOAP, และ WSDL เป็นส่วนสำคัญในการสื่อสารระหว่างอุปกรณ์เครือข่ายวิดีโอที่เข้าร่วมมาตรฐาน ONVIF ซึ่งเป็นมาตรฐานที่ช่วยให้อุปกรณ์เหล่านี้สามารถสื่อสารและทำงานร่วมกันได้อย่างราบรื่น โดย

1. XML (eXtensible Markup Language): XML เป็นภาษาที่ใช้ในการกำหนดรูปแบบข้อมูลที่เข้าร่วมในการสื่อสาร โดย XML ใช้โครงสร้างแบบต้นแบบ (structured data) ซึ่งใช้แท็ก (tag) เพื่ออธิบายและแบ่งกลุ่มข้อมูลในรูปแบบที่เป็นข้อความ ใน ONVIF Protocol, ข้อมูลที่ส่งหรือรับระหว่างอุปกรณ์เครือข่ายวิดีโอด้วย XML เพื่อเข้าใจและแลกเปลี่ยนข้อมูลระหว่างกัน
2. SOAP (Simple Object Access Protocol): SOAP เป็นโปรโตคอลสื่อสารที่ใช้ในการแพ็คเกจข้อมูล XML และส่งผ่านเครือข่าย โดย SOAP เป็นตัวกลางในการสื่อสารระหว่างอุปกรณ์เครือข่ายวิดีโอด้วย ONVIF Protocol ซึ่งเป็นมาตรฐานสำหรับการสื่อสารระหว่างแอปพลิเคชันต่าง ๆ ผ่านทางเครือข่าย ข้อมูล XML จะถูกแพ็คเกจเป็นข้อมูล

SOAP และส่งผ่านโครงข่าย ข้อดีของ SOAP คือสามารถสื่อสารระหว่างระบบที่ไม่เหมือนกันได้ เนื่องจากใช้ภาษา XML เป็นรูปแบบฐานข้อมูล

3. WSDL (Web Services Description Language): WSDL เป็นภาษาที่ใช้ในการบรรยายและระบุเซอร์วิสวีบเซอร์วิสที่ให้บริการใน ONVIF Protocol โดย WSDL จะระบุโครงสร้างของข้อมูล XML ที่ใช้ในการสื่อสารระหว่างอุปกรณ์ เช่น รายละเอียดของเมธอด (methods) ที่สามารถเรียกใช้ได้และข้อมูลที่ต้องรับหรือส่ง นอกจากนี้ WSDL ยังระบุวิธีการเชื่อมต่อ ที่ใช้ในการสื่อสารระหว่างอุปกรณ์ เช่น HTTP หรือ SOAP

ดังนั้น ใน ONVIF Protocol จะใช้ XML เพื่อกำหนดรูปแบบข้อมูล ใช้ SOAP เป็นโปรโตคอลสื่อสารและแพ็กเกจข้อมูล XML และใช้ WSDL เป็นภาษาที่ใช้ในการบรรยายและระบุเซอร์วิสวีบเซอร์วิสที่ให้บริการใน ONVIF Protocol ความสัมพันธ์ระหว่างเทคโนโลยีเหล่านี้ช่วยให้อุปกรณ์เครือข่ายวิดีโอที่เข้าร่วมมาตรฐาน ONVIF สามารถสื่อสารและทำงานร่วมกันได้อย่างมีประสิทธิภาพ และสอดคล้องกันได้

## 2.5 Session

Session ในเว็บแอปพลิเคชันเป็นตัวที่ใช้ในการเก็บข้อมูลและสถานะของผู้ใช้ระหว่างที่ผู้ใช้เข้าถึงเว็บไซต์หรือแอปพลิเคชัน ภาษาที่ใช้สร้างเว็บ เช่น Golang หรือ PHP สามารถใช้งาน Session ได้ เมื่อผู้ใช้เข้าสู่ระบบหรือทำการติดต่อกับเว็บไซต์ แอปพลิเคชันจะสร้าง Session ใหม่ขึ้นและกำหนดตัวระบุเซสชัน (session ID) ให้กับผู้ใช้ ซึ่ง session ID จะถูกเก็บไว้ในคุกกี้ (cookie) หรือส่งผ่านพารามิเตอร์ของ URL Session ทำหน้าที่เก็บข้อมูลที่เกี่ยวข้องกับผู้ใช้ เช่น ข้อมูลการเข้าสู่ระบบ ค่าตัวแปร เมนูที่เลือก หรือสถานะการทำงาน ข้อมูลเหล่านี้จะถูกเก็บไว้ในเซิร์ฟเวอร์หรือฐานข้อมูลที่เกี่ยวข้องกับเว็บไซต์ Session ช่วยให้เว็บไซต์สามารถระบุและจดจำข้อมูลของผู้ใช้ได้ระหว่างเว็บเข้าถึง

## 2.6 Database

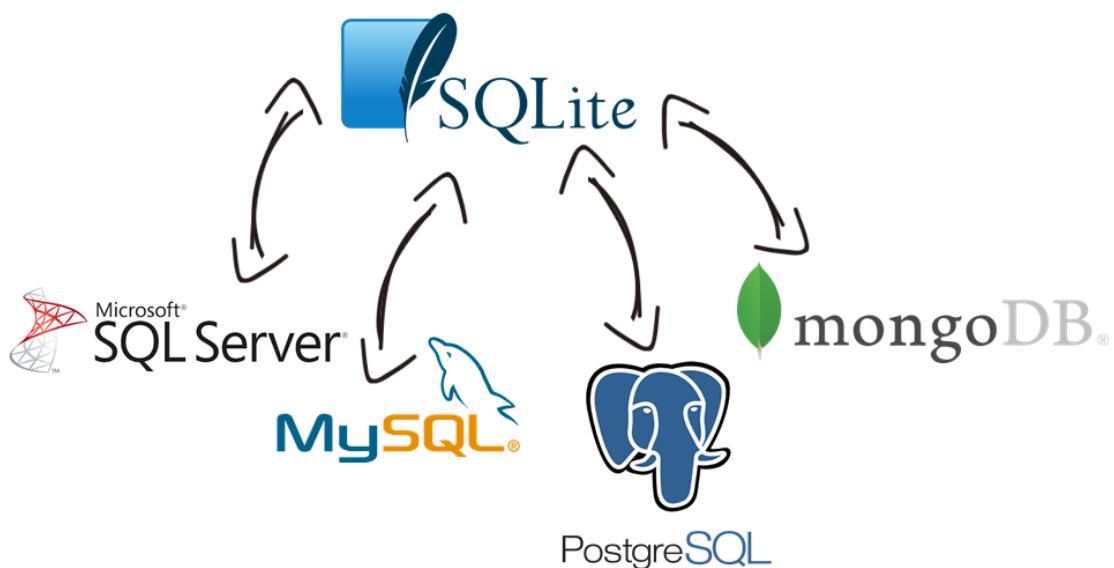
ฐานข้อมูล (Database) เป็นคอลเลกชันข้อมูลที่อยู่ในรูปแบบที่สามารถจัดเก็บและจัดระเบียบได้อย่างเป็นระบบเรียบร้อย เป็นแหล่งที่เก็บข้อมูลที่สามารถเข้าถึงและประมวลผลได้ตามความต้องการ ฐานข้อมูลมีการโครงสร้างอย่างเป็นระบบ เช่น ใช้ตาราง (Table) เพื่อเก็บข้อมูลใน

รูปแบบของแ苦难และคอลัมน์ ฐานข้อมูลที่ใช้ระบบการจัดเก็บแบบนี้เรียกว่า ฐานข้อมูลแบบสัมพันธ์ (Relational Database) โดยจะทำการใช้ภาษา SQL (Structured Query Language) เพื่อเข้าถึง และดำเนินการกับข้อมูลในฐานข้อมูล ฐานข้อมูลสามารถใช้ในการเก็บข้อมูลที่หลากหลาย และเก็บข้อมูลที่มีความสัมพันธ์ต่อกัน โดยสามารถสร้างความสัมพันธ์ระหว่างตารางหรือองค์ประกอบข้อมูลที่แตกต่างกันในฐานข้อมูล ซึ่งช่วยให้สามารถดึงข้อมูลจากตารางหรือรวมข้อมูลจากหลายตารางได้อย่างมีประสิทธิภาพ การใช้งานฐานข้อมูลสามารถทำได้ในหลายแบบ เช่น ฐานข้อมูลขนาดเล็ก (Small-scale Database) ที่ใช้ในแอปพลิเคชันเดียวจะใช้เป็น SQLite หรือ ฐานข้อมูลขนาดใหญ่ (Large-scale Database) ที่ใช้ในระบบองค์กรหรือเว็บไซต์ขนาดใหญ่จะใช้เป็น MySQL เป็นต้น

ความสัมพันธ์ของ SQL, SQLite และ MySQL คือ SQL เป็นภาษาที่ใช้ในการจัดการข้อมูลในระบบฐานข้อมูลทั้งสาม โดยทั้ง SQLite และ MySQL เป็นระบบฐานข้อมูลที่ใช้ SQL เป็นภาษาในการจัดการข้อมูล แต่มีความแตกต่างในลักษณะและการใช้งานดังนี้:

1. SQL (Structured Query Language): SQL เป็นภาษาamuraตறฐานที่ใช้ในการสื่อสารกับระบบฐานข้อมูล ภาษา SQL มีคำสั่งที่ใช้ในการสร้างฐานข้อมูล (CREATE), เพิ่มข้อมูล (INSERT), แก้ไขข้อมูล (UPDATE), ลบข้อมูล (DELETE), และดึงข้อมูล (SELECT) เป็นต้น ภาษา SQL เป็นภาษาที่เข้าใจง่ายและเป็นมาตรฐานที่ใช้ได้กับหลายระบบฐานข้อมูล
  2. SQLite: SQLite นั้นเป็นระบบฐานข้อมูลแบบฝังตัว (embedded database) ซึ่งหมายความว่ามันเป็นไลบรารีที่ทำหน้าที่เป็นฐานข้อมูลและเครื่องมือ SQL ที่มีอยู่ในรูปแบบของไฟล์เดียวกับแอปพลิเคชัน ฐานข้อมูล SQLite ถูกสร้างและเก็บไว้ในไฟล์เดียวกับแอปพลิเคชันที่ใช้งาน มันเหมาะสมสำหรับโปรแกรมที่ต้องการใช้งานฐานข้อมูลบนอุปกรณ์เคลื่อนที่หรือแอปพลิเคชันที่ไม่ต้องการระบบฐานข้อมูลที่มีขนาดใหญ่
  3. MySQL: MySQL เป็นระบบจัดการฐานข้อมูลแบบเซิร์ฟเวอร์ (server-based database) ที่ใช้ SQL เป็นภาษาในการจัดการข้อมูล มันเป็นระบบฐานข้อมูลที่แพร่หลายและได้รับความนิยมอย่างแพร่หลายในการพัฒนาเว็บแอปพลิเคชัน MySQL มีความสามารถในการรองรับการทำงานขนาดใหญ่ โดยมีการจัดเก็บข้อมูลและการดำเนินการบนเซิร์ฟเวอร์ที่แยกออกจากแอปพลิเคชัน

โดยสรุปแล้ว SQLite เป็นฐานข้อมูลแบบฝังตัวที่ใช้ในแอปพลิเคชันเดียวหรือโปรแกรมที่มีขนาดเล็ก ส่วน MySQL เป็นระบบจัดการฐานข้อมูลแบบเซิร์ฟเวอร์ที่มีความสามารถในการรองรับการทำงานขนาดใหญ่ และ SQL เป็นภาษา命令ภาษาที่ใช้ในการจัดการฐานข้อมูลและสื่อสารกับระบบฐานข้อมูลต่าง ๆ ฐานข้อมูลมีบทบาทสำคัญในการจัดเก็บข้อมูลในรูปแบบที่เป็นระเบียบและสามารถเข้าถึงได้ตามความต้องการ นอกจากนี้ยังมีความสามารถในการบันทึกและดึงข้อมูลอย่างรวดเร็ว และให้การรับประกันความถูกต้องและความปลอดภัยของข้อมูลที่ถูกจัดเก็บ



ภาพที่ 5 ภาพรวมของ SQL และอื่น ๆ

(ที่มา: <https://medium.com/arnoldcode-academy/the-differences-between-sql-vs-sqlite-massive-centralized-versus-lightweight-a16aa8f5109c>)

## 2.7 Internet Protocol Camera (IP Camera)

IP Camera (Internet Protocol Camera) เป็นกล้องวงจรปิดที่สามารถส่งสัญญาณวิดีโอผ่านเครือข่ายอินเทอร์เน็ตได้ โดยใช้โปรโตคอล IP (Internet Protocol) ในการสื่อสารและการรับส่งข้อมูล รูปแบบนี้ช่วยให้ผู้ใช้สามารถติดตั้งและใช้งานกล้องได้อย่างง่ายและสะดวกในระบบเครือข่ายที่มีอินเทอร์เน็ต IP Camera มีความสามารถในการบันทึกภาพนิ่งหรือวิดีโอที่ความละเอียดสูง และสามารถถ่ายทอดสดสัญญาณวิดีโอผ่านเครือข่ายอินเทอร์เน็ต ผู้ใช้นั้นสามารถเข้าถึงกล้อง IP ได้จากที่ไหนก็ได้ผ่านอุปกรณ์ เช่น คอมพิวเตอร์ สมาร์ทโฟน หรือแท็บเล็ตที่เชื่อมต่อกับเครือข่ายอินเทอร์เน็ต

การติดตั้งและใช้งาน IP Camera ไม่ต้องใช้สายสัญญาณที่ซับซ้อนเหมือนกับกล้องวงจรปิดแบบอื่น ๆ เนื่องจากกล้องเชื่อมต่อเข้ากับเครือข่ายผ่านสายเคเบิลเครือข่ายหรือระบบไร้สาย (Wi-Fi) ซึ่งทำให้ง่าย ต่อการติดตั้งในพื้นที่ต่าง ๆ และสามารถควบคุมการกำหนดค่าและดูภาพผ่านเว็บเบราว์เซอร์ (Web Browser) หรือแอปพลิเคชันที่รองรับได้ IP Camera มีความหลากหลายในรูปแบบและคุณสมบัติต่าง ๆ ซึ่งรวมถึงความละเอียดของภาพ การตรวจจับการเคลื่อนไหว การบันทึกข้อมูล พังก์ชันกล้องระบบอัจฉริยะ เป็นต้น โดยสามารถใช้กล้อง IP Camera ในการรักษาความปลอดภัยในบ้าน ที่ทำงาน หรือสถานที่อื่น ๆ หรือใช้เพื่อการตรวจสอบหรือระบุปัญหาในสถานที่ต่าง ๆ ได้



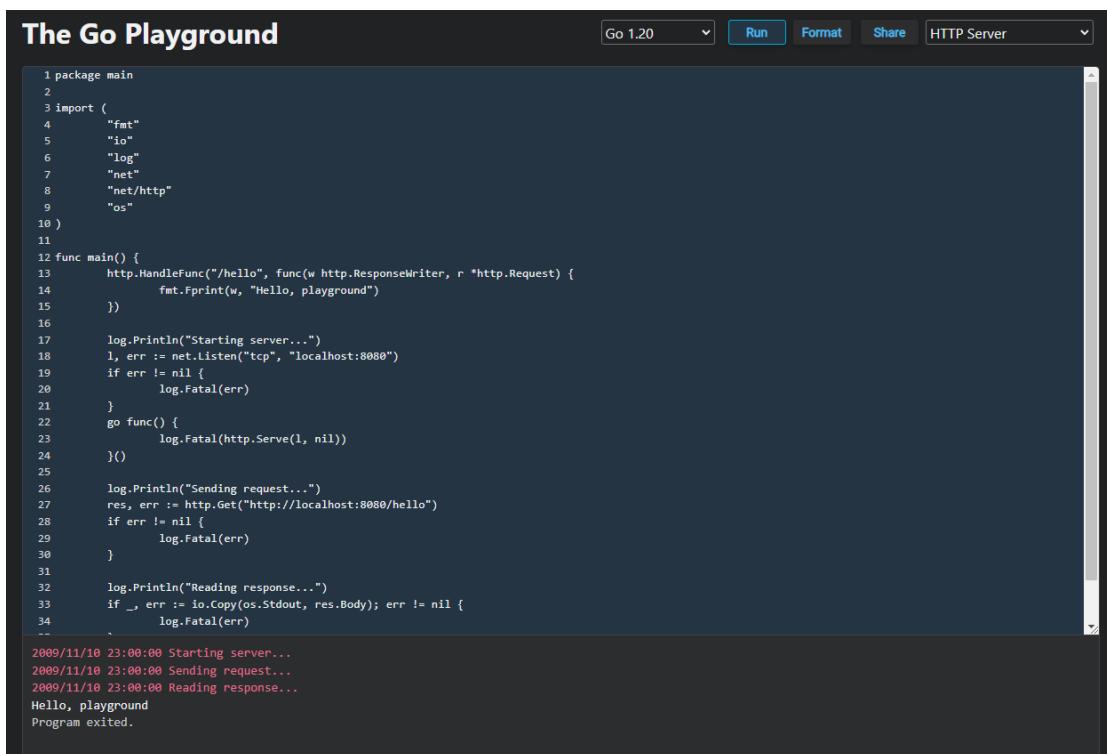
ภาพที่ 6 ภาพรวมของตัวอย่างกล้องไอพี

(ที่มา: <https://www.tp-link.com/th/blog/665/>)

## 2.8 ภาษา Golang

ภาษา Go (Golang) เป็นภาษาโปรแกรมที่พัฒนาโดยทีมผู้พัฒนาของ Google โดยมีวัตถุประสงค์เพื่อสร้างภาษาโปรแกรมที่มีประสิทธิภาพสูง ง่ายต่อการอ่านและเขียน ภาษา Golang เน้นไปที่ความเร็ว ความปลอดภัย และความเป็นระเบียบในการเขียนโปรแกรม Golang มีสัญลักษณ์เฉพาะที่ช่วยให้โค้ดอ่านและเขียนได้ง่าย เช่น การใช้งาน `go routines` เพื่อรับการประมวลผลแบบพร้อมกัน และใช้งาน `channels` เพื่อสื่อสารระหว่าง `go routines` แบบปลอดภัย ภาษา Golang จึงเหมาะสมสำหรับการพัฒนาโปรแกรมที่ต้องการประสิทธิภาพสูง เช่น การพัฒนาเว็บแอปพลิเคชัน

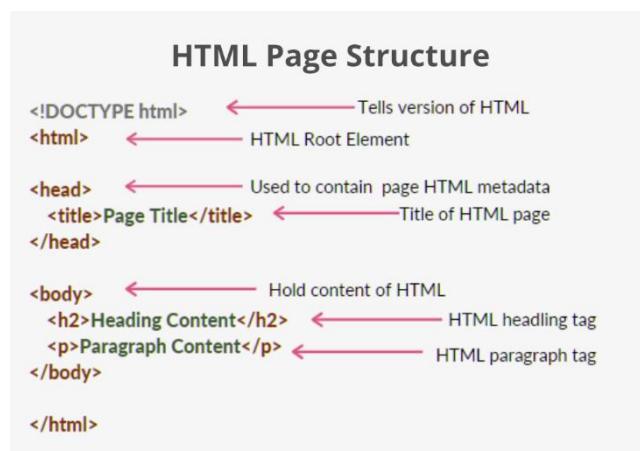
ระบบเครือข่าย ระบบคลังข้อมูล และแอปพลิเคชันที่ต้องการการจัดการแบบพร้อมกัน (concurrency) อีกทั้งยังสนับสนุนการพัฒนาโปรแกรมแบบเป็นเชิร์ฟเวอร์อย่างดี นอกจากนี้ Golang ยังเป็นภาษาที่มีเครื่องมือและไลบรารีมากมายที่ช่วยในการพัฒนาโปรแกรมได้อย่างรวดเร็วและมีประสิทธิภาพ เช่น ไลบรารีสำหรับการทำงานกับเว็บเซิร์ฟเวอร์ (web server) หรือไลบรารีสำหรับการทำงานกับฐานข้อมูล ทำให้ Golang เป็นภาษาที่ได้รับความนิยมสูงในงานพัฒนาเว็บแอปพลิเคชัน และบริการเว็บต่าง ๆ ในปัจจุบัน



```

1 package main
2
3 import (
4     "fmt"
5     "io"
6     "log"
7     "net"
8     "net/http"
9     "os"
10 )
11
12 func main() {
13     http.HandleFunc("/hello", func(w http.ResponseWriter, r *http.Request) {
14         fmt.Fprintf(w, "Hello, playground")
15     })
16
17     log.Println("Starting server...")
18     l, err := net.Listen("tcp", "localhost:8080")
19     if err != nil {
20         log.Fatal(err)
21     }
22     go func() {
23         log.Fatal(http.Serve(l, nil))
24     }()
25
26     log.Println("Sending request...")
27     res, err := http.Get("http://localhost:8080/hello")
28     if err != nil {
29         log.Fatal(err)
30     }
31
32     log.Println("Reading response...")
33     if _, err := io.Copy(os.Stdout, res.Body); err != nil {
34         log.Fatal(err)
35     }
36 }
37
38 log.Println("Program exited.")
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
559
560
561
562
563
564
565
566
567
568
569
569
570
571
572
573
574
575
576
577
578
579
579
580
581
582
583
584
585
586
587
587
588
589
589
590
591
592
593
594
595
596
597
597
598
599
599
600
601
602
603
604
605
606
607
607
608
609
609
610
611
612
613
614
615
615
616
617
617
618
619
619
620
621
622
623
623
624
625
625
626
627
627
628
629
629
630
631
631
632
633
633
634
635
635
636
637
637
638
639
639
640
641
641
642
643
643
644
645
645
646
647
647
648
649
649
650
651
651
652
653
653
654
655
655
656
657
657
658
659
659
660
661
661
662
663
663
664
665
665
666
667
667
668
669
669
670
671
671
672
673
673
674
675
675
676
677
677
678
679
679
680
681
681
682
683
683
684
685
685
686
687
687
688
689
689
690
691
691
692
693
693
694
695
695
696
697
697
698
699
699
700
701
701
702
703
703
704
705
705
706
707
707
708
709
709
710
711
711
712
713
713
714
715
715
716
717
717
718
719
719
720
721
721
722
723
723
724
725
725
726
727
727
728
729
729
730
731
731
732
733
733
734
735
735
736
737
737
738
739
739
740
741
741
742
743
743
744
745
745
746
747
747
748
749
749
750
751
751
752
753
753
754
755
755
756
757
757
758
759
759
760
761
761
762
763
763
764
765
765
766
767
767
768
769
769
770
771
771
772
773
773
774
775
775
776
777
777
778
779
779
780
781
781
782
783
783
784
785
785
786
787
787
788
789
789
790
791
791
792
793
793
794
795
795
796
797
797
798
799
799
800
801
801
802
803
803
804
805
805
806
807
807
808
809
809
810
811
811
812
813
813
814
815
815
816
817
817
818
819
819
820
821
821
822
823
823
824
825
825
826
827
827
828
829
829
830
831
831
832
833
833
834
835
835
836
837
837
838
839
839
840
841
841
842
843
843
844
845
845
846
847
847
848
849
849
850
851
851
852
853
853
854
855
855
856
857
857
858
859
859
860
861
861
862
863
863
864
865
865
866
867
867
868
869
869
870
871
871
872
873
873
874
875
875
876
877
877
878
879
879
880
881
881
882
883
883
884
885
885
886
887
887
888
889
889
890
891
891
892
893
893
894
895
895
896
897
897
898
899
899
900
901
901
902
903
903
904
905
905
906
907
907
908
909
909
910
911
911
912
913
913
914
915
915
916
917
917
918
919
919
920
921
921
922
923
923
924
925
925
926
927
927
928
929
929
930
931
931
932
933
933
934
935
935
936
937
937
938
939
939
940
941
941
942
943
943
944
945
945
946
947
947
948
949
949
950
951
951
952
953
953
954
955
955
956
957
957
958
959
959
960
961
961
962
963
963
964
965
965
966
967
967
968
969
969
970
971
971
972
973
973
974
975
975
976
977
977
978
979
979
980
981
981
982
983
983
984
985
985
986
987
987
988
989
989
990
991
991
992
993
993
994
995
995
996
997
997
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
1551
1552
1552
1553
1553
1554
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1560
1561
1561
1562
1562
1563
1563
1564
1564
1565
1565
1566
1566
1567
1567
1568
1568
1569
1569
1570
1570
1571
1571
1572
1572
1573
1573
1574
1574
1575
1575
1576
1576
1577
1577
1578
1578
1579
1579
1580
1580
1581
1581
1582
1582
1583
1583
1584
1584
1585
1585
1586
1586
1587
1587
1588
1588
1589
1589
1590
1590
1591
1591
1592
1592
1593
1593
1594
1594
1595
```

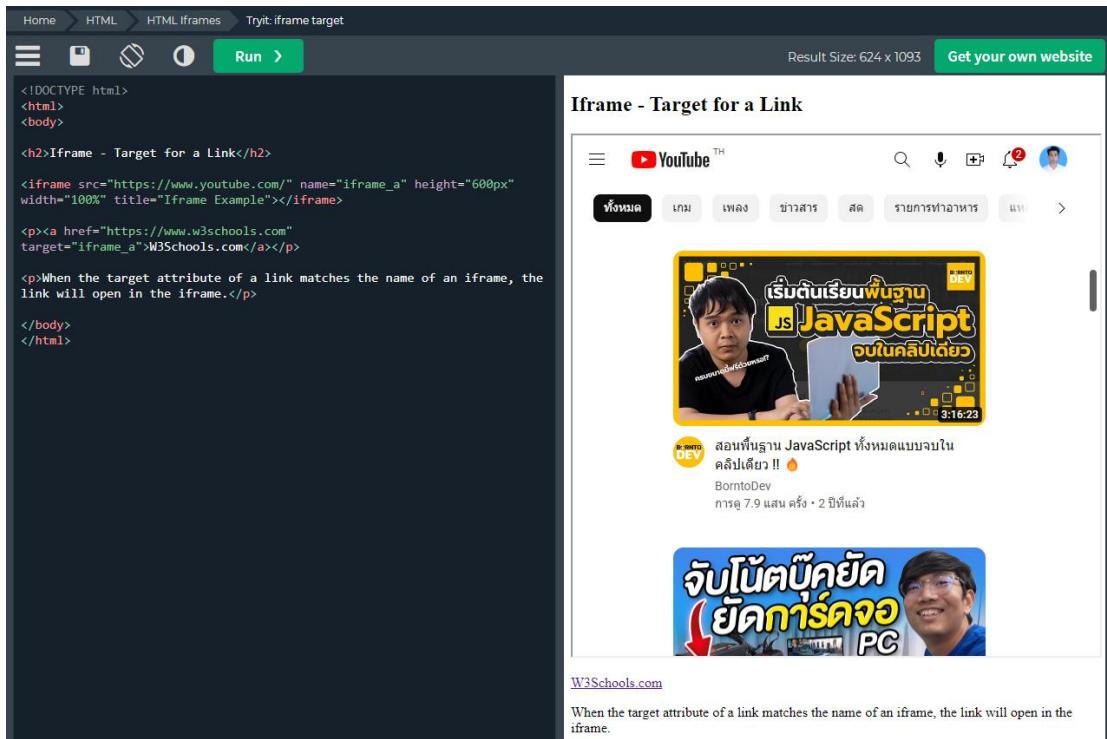
สนับสนุนในการสร้างโครงสร้าง HTML อื่น เช่น VScode เป็นต้น นอกจากนั้น HTML ยังใช้โครงสร้าง ๆ ที่เรียกว่า "แท็ก" (tags) เพื่อกำหนดรูปแบบและประเภทของเนื้อหาในหน้าเว็บ แท็กจะถูกใช้ในรูปแบบของคู่ เช่น <tag> และ </tag> โดยรูปแบบที่เปิด <tag> จะใช้ในการเริ่มต้นขององค์ประกอบ และรูปแบบที่ปิด </tag> จะใช้ในการสิ้นสุดขององค์ประกอบ HTML ใช้แท็ก (tags) เป็นตัวกำหนดส่วนประกอบต่าง ๆ ในหน้าเว็บ เช่น แท็ก <html> ใช้เพื่อกำหนดบริบทและเริ่มต้นของหน้าเว็บ แท็ก <head> ใช้เพื่อกำหนดข้อมูลที่เกี่ยวข้องกับหน้าเว็บ เช่น การเชื่อมต่อไฟล์ CSS หรือสคริปต์ JavaScript และแท็ก <body> ใช้เพื่อกำหนดเนื้อหาที่จะแสดงในหน้าเว็บ



ภาพที่ 8 โครงสร้างของภาษา HTML

(ที่มา: <https://www.geeksforgeeks.org/html/>)

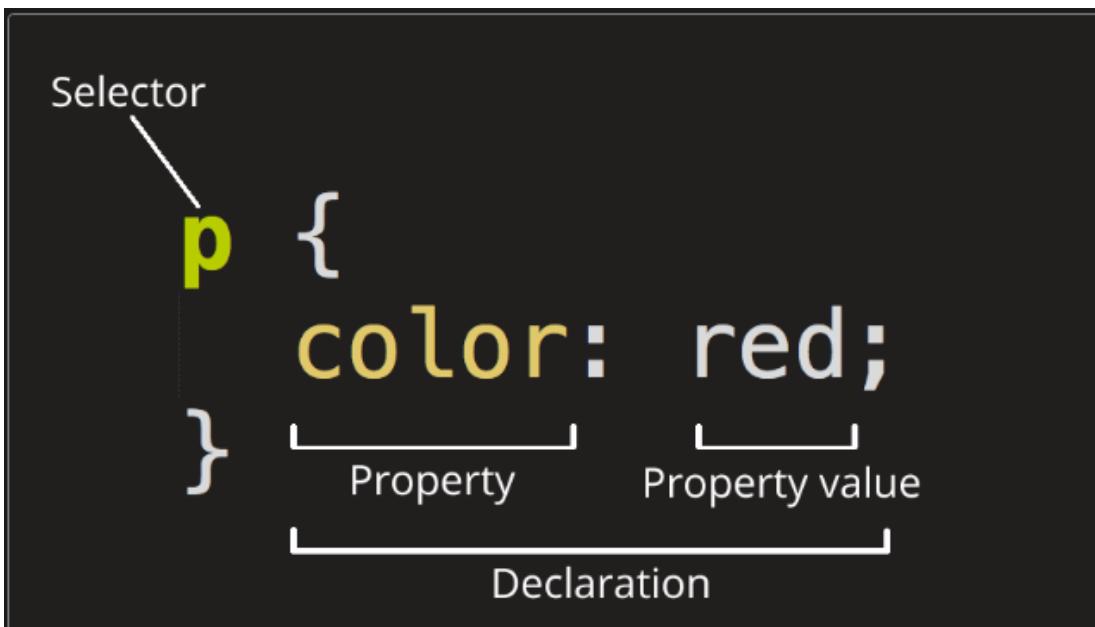
HTML เป็นภาษาที่เปิดเผยแพร่และมีมาตรฐานสากล ซึ่งหมายความว่าเว็บเบราว์เซอร์สามารถอ่านและแสดงผลหน้าเว็บที่เขียนด้วย HTML ได้ทั่วไป ซึ่งทำให้เป็นภาษาหลักในการพัฒนาเว็บไซต์ และแอปพลิเคชันเว็บ



ภาพที่ 9 ตัวอย่างการใช้งานภาษา HTML

## 2.10 ภาษา CSS

ภาษา CSS (Cascading Style Sheets) เป็นภาษาที่ใช้ในการกำหนดรูปแบบและสไตล์สำหรับเอกสาร HTML (Hypertext Markup Language) หรือ XML (eXtensible Markup Language) ภาษานี้ใช้เพื่อกำหนดสีพื้นหลัง ขนาดตัวอักษร ขอบเขตของเนื้อหา รูปแบบการจัดวาง และอื่น ๆ เพื่อสร้างประสบการณ์การแสดงผลที่สวยงามและสมดุลในเว็บไซต์หรือเอกสารต่าง ๆ CSS ทำงานร่วมกับ HTML โดยผู้ใช้สามารถกำหนดรูปแบบให้กับองค์ประกอบต่าง ๆ ในเอกสาร HTML ได้ซึ่งการกำหนดรูปแบบใน CSS เป็นแบบลำดับ (cascade) ซึ่งหมายความว่ารูปแบบที่ถูกกำหนดให้กับองค์ประกอบในระดับสูงสุดจะมีผลบังคับบัญชาลงมาไปยังระดับต่ำกว่านั้น รูปแบบใน CSS ประกอบด้วยชีเล็กเตอร์ (selectors) คุณสมบัติ (properties) และค่า (value) โดยชีเล็กเตอร์ใช้ในการเลือกองค์ประกอบที่ต้องการกำหนดรูปแบบ ส่วนคุณสมบัติใช้ในการกำหนดรูปแบบที่ต้องการให้กับองค์ประกอบที่ถูกเลือก และค่าใช้ในการกำหนดระดับภายใต้คุณสมบัตินั้น ๆ ดังภาพที่ 10



ภาพที่ 10 โครงสร้างของภาษา CSS

CSS เป็นภาษาที่เปิดเผยแพร่สามารถนำไปใช้งานบนหลายแพลตฟอร์มและเว็บเบราว์เซอร์ได้ซึ่งช่วยให้ผู้พัฒนาสามารถสร้างเว็บไซต์ที่มีรูปแบบสวยงาม สื่อความหมาย และสามารถปรับแต่งได้อย่างยืดหยุ่นตามความต้องการของผู้ใช้ได้อย่างมีความสะดวกและหลากหลาย

```

<!DOCTYPE html>
<html>
<head>
<style>
.nav {
  background-color: yellow;
  list-style-type: none;
  text-align: center;
  margin: 0;
  padding: 0;
}

.nav li {
  display: inline-block;
  font-size: 20px;
  padding: 20px;
}
</style>
</head>
<body>

<h1>Horizontal Navigation Links</h1>
<p>By default, list items are displayed vertically. In this example we use display: inline-block to display them horizontally (side by side).</p>
<p>Note: If you resize the browser window, the links will automatically break when it becomes too crowded.</p>

<ul class="nav">
<li><a href="#">Home</a></li>
<li><a href="#">About Us</a></li>
<li><a href="#">Our Clients</a></li>
<li><a href="#">Contact Us</a></li>
</ul>

</body>
</html>
  
```

**Horizontal Navigation Links**

By default, list items are displayed vertically. In this example we use display: inline-block to display them horizontally (side by side).

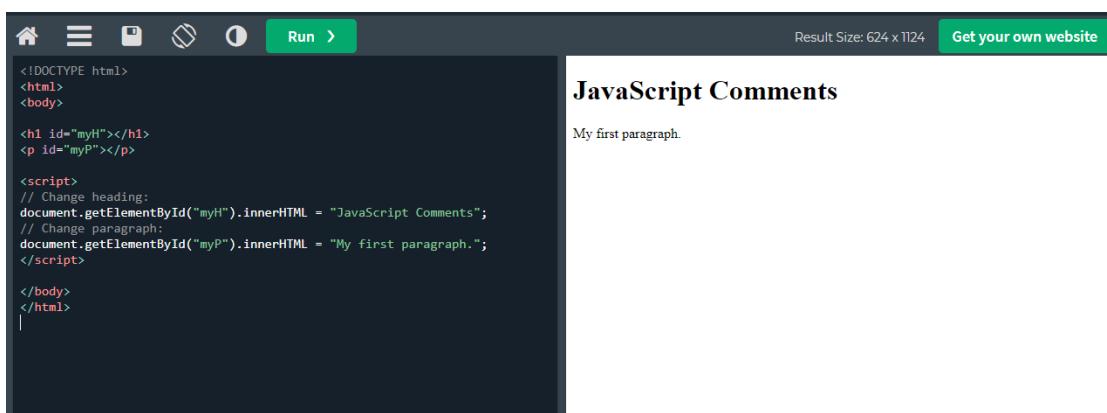
Note: If you resize the browser window, the links will automatically break when it becomes too crowded.

[Home](#) [About Us](#) [Our Clients](#) [Contact Us](#)

ภาพที่ 11 ตัวอย่างการใช้งานภาษา CSS

## 2.11 ภาษา JavaScript

ภาษา JavaScript (ECMAScript) เป็นภาษาโปรแกรมที่ถูกพัฒนาขึ้นเพื่อใช้ในการเขียนโปรแกรมที่ทำงานบนเว็บเบราว์เซอร์ ภาษานี้ได้รับความนิยมอย่างแพร่หลายเนื่องจากความสามารถที่หลากหลายและความยืดหยุ่นในการใช้งานกับเว็บไซต์และแอปพลิเคชันต่าง ๆ JavaScript เป็นภาษาโปรแกรมแบบสคริปต์ (scripting language) ที่มีลักษณะการทำงานแบบอินเทอร์เพรต์ (interpreted) ซึ่งหมายความว่าโค้ด JavaScript สามารถรันได้ทันทีโดยไม่ต้องผ่านขั้นตอนคอมไพล์ ก่อน ซึ่งทำให้การพัฒนาและทดสอบโปรแกรมเป็นไปได้อย่างรวดเร็วและมีความยืดหยุ่นสูง JavaScript เป็นภาษาที่ใช้ในการควบคุมพฤติกรรมของเว็บไซต์หรือแอปพลิเคชัน ได้แก่ การจัดการกับเหตุการณ์ต่าง ๆ เช่น การคลิกที่ปุ่ม, การกรอกแบบฟอร์ม, การแสดงผลข้อมูลแบบแสดงผลเรียลไทม์ (real-time), การส่งข้อมูลผ่านเครือข่าย และอื่น ๆ นอกจากนี้ JavaScript ยังมีความสามารถในการเข้าถึงและปรับแต่งองค์ประกอบในหน้าเว็บ (DOM - Document Object Model) ซึ่งช่วยให้เราสามารถแก้ไขหรือเปลี่ยนแปลงเนื้อหาและการแสดงผลบนหน้าเว็บได้อย่างได้ผล โดยรวมแล้ว JavaScript เป็นภาษาโปรแกรมที่มีความสำคัญมากในการพัฒนาเว็บไซต์และแอปพลิเคชัน และมีการใช้งานและการพัฒนาอย่างต่อเนื่องเพื่อตอบสนองต่อความต้องการและความเปลี่ยนแปลงในอุตสาหกรรมเทคโนโลยีสารสนเทศและการสื่อสารในปัจจุบัน



The screenshot shows a web-based code editor interface. On the left, there is a dark-themed code editor window containing the following JavaScript code:

```
<!DOCTYPE html>
<html>
<body>

<h1 id="myH"></h1>
<p id="myP"></p>

<script>
// Change heading:
document.getElementById("myH").innerHTML = "JavaScript Comments";
// Change paragraph:
document.getElementById("myP").innerHTML = "My first paragraph.";
</script>

</body>
</html>
|
```

On the right, there is a preview window titled "JavaScript Comments" showing the resulting HTML output:

**JavaScript Comments**

My first paragraph.

At the top of the interface, there are icons for file, edit, and refresh, followed by a "Run" button. At the bottom right, there is a green button that says "Get your own website".

ภาพที่ 12 ตัวอย่างการใช้งานภาษา JavaScript

## 2.12 JavaScript Object Notation (JSON)

JSON (JavaScript Object Notation) เป็นรูปแบบที่ใช้ในการแลกเปลี่ยนข้อมูลคอมพิวเตอร์ที่มีความยืดหยุ่นและอยู่ในรูปแบบข้อความธรรมดា (Plain Text) ซึ่งทั้งมนุษย์และโปรแกรมคอมพิวเตอร์สามารถอ่านและเข้าใจได้ง่าย JSON ถูกสร้างขึ้นจากชุดข้อมูลในภาษาจาวาสคริปต์ (JavaScript) ที่เป็นภาษาโปรแกรมที่ใช้กันอย่างแพร่หลายในการพัฒนาเว็บแอปพลิเคชันและเว็บไซต์ รูปแบบของ JSON ถูกออกแบบให้ใช้งานลึกซึ้ง ([ ]) เพื่อแทนแผลงลำดับ (Array) และใช้งานลึกเป็นค่า ({}) เพื่อแทนแผลงคู่ (Associative Array) หรือที่เรียกว่า Object ข้อมูลใน JSON ถูกจัดเรียงเป็นสมาชิกแต่ละตัวและคู่นัดด้วยเครื่องหมายจุลภาค (,) ซึ่งชี้อ้างอิงข้อมูลจะถูกค้นด้วยเครื่องหมายทวีภาค (:). เพื่อแสดงความสัมพันธ์ระหว่างชื่อและค่าของข้อมูลในแต่ละสมาชิก JSON มีคุณสมบัติที่มีประสิทธิภาพและใช้งานง่าย ทำให้เป็นรูปแบบที่ได้รับความนิยมในการส่งข้อมูลในเว็บแอปพลิเคชัน ในอดีต XML เป็นรูปแบบที่ใช้ในการแลกเปลี่ยนข้อมูลในเว็บ แต่ปัจจุบัน JSON ได้รับความนิยมมากกว่าเนื่องจากมีขนาดที่เล็กกว่าและเข้าใจได้ง่ายกว่า การอ่านและสร้าง JSON ก็สะดวกและมีความกระชับ นอกจากนี้ JSON ยังสามารถนำไปประยุกต์ใช้ในการรับส่งข้อมูลในรูปแบบอื่น ๆ ได้ เช่นการจัดเก็บข้อมูลผ่านการให้บริการเว็บไซต์ (Web Service) หรือการสื่อสารกับแหล่งข้อมูลอื่น ๆ ดังนั้น JSON จึงเป็นรูปแบบที่มีความยืดหยุ่น กระชับและเข้าใจง่ายในการแลกเปลี่ยนข้อมูลทางคอมพิวเตอร์ ซึ่งได้รับความนิยมในเว็บแอปพลิเคชัน เนื่องจากสามารถนำไปใช้กับหลายแพลตฟอร์มและการสื่อสารกับระบบอื่น ๆ ได้โดยง่าย

```
Response(5ms) ×

1 HTTP/1.1 200 OK
2 Content-Type: application/json; charset=utf-8
3 Date: Wed, 08 Dec 2021 03:19:41 GMT
4 Content-Length: 251
5 Connection: close
6
7 ~ [
8 ~ {
9 ~ "id": "1",
10 ~ "title": "Harry Potter",
11 ~ "author": "J. K. Rowling"
12 ~ },
13 ~ {
14 ~ "id": "2",
15 ~ "title": "The Lord of the Rings",
16 ~ "author": "J. R. R. Tolkien"
17 ~ },
18 ~ {
19 ~ "id": "3",
20 ~ "title": "The Wizard of Oz",
21 ~ "author": "L. Frank Baum"
22 ~ },
23 ~ {
24 ~ "id": "4",
25 ~ "title": "The Godfather",
26 ~ "author": "Mario Puzo"
27 ~ }
28 ]
```

ภาพที่ 13 ตัวอย่างการแสดงของข้อมูลรูปแบบ JSON

## 2.13 งานวิจัยที่เกี่ยวข้อง

Phan, C. C., Nguyen, T. B. and Chung, S.-T. (2015). พยายามนำบริการ ONVIF Event มาทำการสร้างข้อความแจ้งเตือนเมื่อมีเหตุการณ์เกิดขึ้น มีการใช้ชุดเครื่องมือ gSOAP เพื่อสร้างโค้ดไลบรารี C เพื่อจัดการข้อความ SOAP กำหนดโครงสร้างข้อมูลสำหรับคำขอต่าง ๆ มีการสร้าง Event Collector เพื่อรับเหตุการณ์ที่มาจากการตัวโมดูลอย่างต่อเนื่อง มีการใช้งาน API เพื่อทำการจัดส่งข้อความระหว่างกันโดยทุกข้อความจะมี ID เป็นของตัวเอง ช่วยให้เซิร์ฟเวอร์จัดการการทำงานได้อย่างราบรื่น

Wenfeng, L. and Yuan, G. (2018). นำเอกสารล้องที่มีมาตรฐาน ONVIF มาใช้กับการจับภาพวิดีโອในที่มีดี โดยมาตรฐาน ONVIF มีบทบาทที่ดีในการตรวจสอบความปลอดภัยแบบเรียลไทม์ และสามารถแจ้งเตือนเหตุการณ์ฉุกเฉินในที่มีดีได้

Lopes, S. F., and et al. (2013). ได้พัฒนาไลบรารี ONVIF เพื่อพัฒนาไคลเอนต์ของกล้องวิดีโอ ซึ่งวิธีที่งานวิจัยนี้นำเสนอส่งผลให้จำนวนพารามิเตอร์ของฟังก์ชันนั้นลดลง ซึ่งช่วยให้เพิ่มความสามารถของ API ของไลบรารีได้อย่างมากและได้ผ่านการใช้งานจริงในเชิงอุตสาหกรรม

Tsai, Y.-H., Hsu, J.-K., Wu, Y.-E. and Huang, W.-F. (2011). ได้เสนอการเพิ่มประสิทธิภาพการกระจายการจัดการเนื้อหาวิดีโอด้วยการควบคุมสมดุลของโหลดในระบบผ่านวังโดยใช้อินเตอร์เฟซ ONVIF กับระบบผลลัพธ์ของการดำเนินการทำให้มีประสิทธิภาพที่ดีในการสตรีมวิดีโอ

Senst, T. and et al. (2011). ได้มีการสร้างเครือข่ายการผ่านวังแบบกระจายอำนาจซึ่งประกอบด้วยกล้องไอพี อุปกรณ์วิเคราะห์และโหนດกลางที่รวบรวมข้อมูล โดยการแลกเปลี่ยนของข้อมูลทั้งหมดในเครือข่ายจะเกิดขึ้นตามข้อกำหนด ONVIF ดังนั้นจึงมั่นใจได้ว่าการทำงานร่วมกันของทุกอุปกรณ์มีความเข้ากันได้และมีความยืดหยุ่นต่ออุปกรณ์ใหม่ที่จะอัพเกรดเข้ามาในระบบ ซึ่งจะช่วยลดต้นทุนและอำนวยความสะดวกต่อการบำรุงรักษาของระบบ

Lin, C.-F. and et al. (2018). บทความนี้พยายามเสนอวิธีแก้ไขข้อผิดพลาดของ ONVIF โดยพยายามขยายข้อกำหนดมาตรฐาน ONVIF ในปัจจุบันด้วยการแก้ไขเพียงเล็กน้อยและเสนอวิธีแก้ไขปัญหาที่ทนต่อความผิดพลาด เพื่อจัดการกับความล้มเหลวของเครื่องบันทึกวิดีโอด้วย

## บทที่ 3

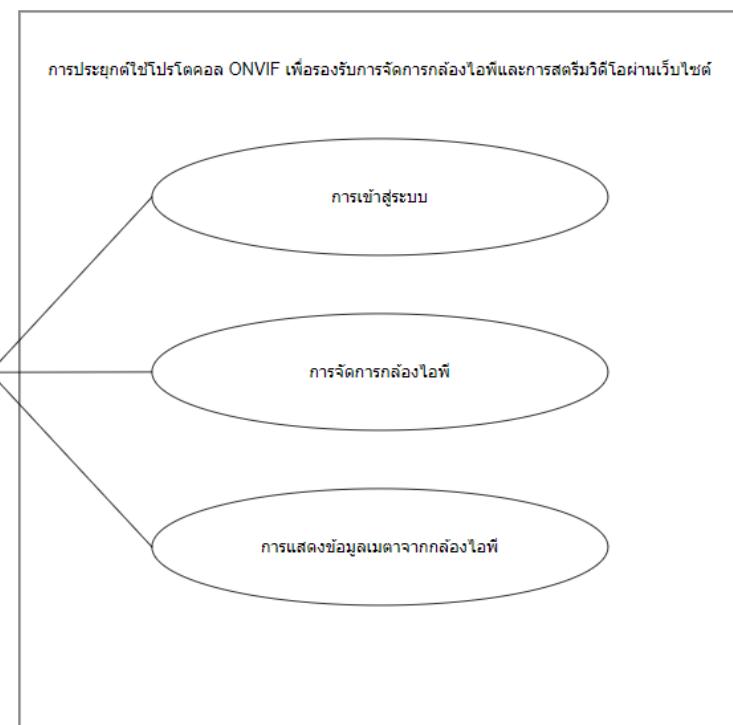
### วิธีดำเนินงานโครงการ

ในบทนี้ทางผู้วิจัยจะกล่าวถึงกระบวนการและขั้นตอนในการพัฒนาการประยุกต์ใช้โพรโทคอล ONVIF เพื่อรับการจัดการกล้องไอพีและการสตรีมวิดีโอผ่านเว็บไซต์ ด้วยเทคโนโลยีที่เกี่ยวข้องต่าง ๆ ซึ่งจะแบ่งออกเป็น 3 หัวข้อดังนี้

- 1) ศึกษาปัญหาและความต้องการของระบบ
- 2) วิเคราะห์และออกแบบระบบ
- 3) เครื่องมือการพัฒนาระบบ

#### 3.1 ศึกษาปัญหาและความต้องการของระบบ

ทางผู้วิจัยได้ทำการเก็บข้อมูลความต้องการของระบบจากผู้ใช้งาน จำนวน 6 ข้อมูลที่ได้มาวิเคราะห์เพื่อการพัฒนาระบบ สามารถออกแบบแผนภาพยูสเคส (Use Case) และสรุปผลได้ดัง ภาพที่ 14 และตารางแสดงการใช้งานในส่วน ๆ ของผู้ใช้งานได้ตามตารางที่ 3



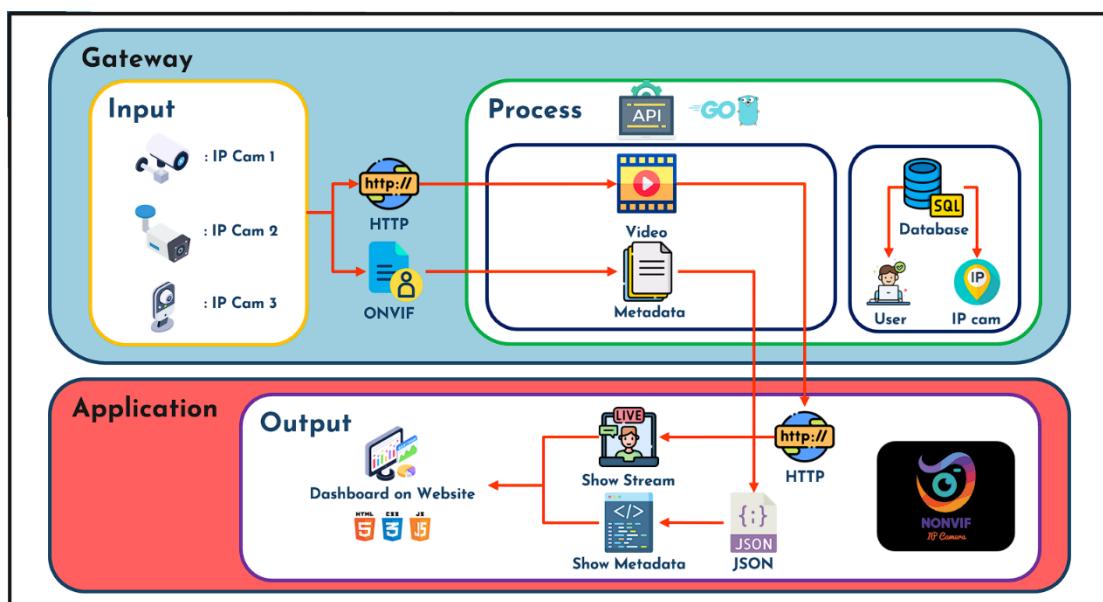
ภาพที่ 14 Use Case Diagram

ตารางที่ 3 ตารางอธิบายของผู้ใช้งานทั่วไป

ลำดับ	รายการ
การเข้าสู่ระบบ	
1	ผู้ใช้งานสามารถสร้างบัญชีเพื่อใช้เข้าสู่ระบบ
2	ผู้ใช้งานสามารถเข้าสู่ระบบด้วยอีเมลและรหัสผ่าน
3	ผู้ใช้งานสามารถกดลีมรหัสผ่านและทำการสร้างรหัสผ่านใหม่
4	ผู้ใช้งานสามารถเลือกตุณ้ำของการสตรีมภาพหรือหน้าการแสดงข้อมูลเมตาได้
5	ผู้ใช้งานสามารถกดออกจากระบบ
การจัดการกล้องไอโอพี	
1	ผู้ใช้งานสามารถเพิ่มกล้องไอโอพี
2	ผู้ใช้งานสามารถลบกล้องไอโอพี
3	ผู้ใช้งานสามารถเลือกรับชมภาพการสตรีมจากกล้องไอโอพีที่เพิ่มเข้ามาในระบบได้
การแสดงข้อมูลเมตาจากกล้องไอโอพี	
1	ผู้ใช้งานสามารถดูข้อมูลเมตาจากกล้องไอโอพีได้

### 3.2 วิเคราะห์และออกแบบระบบ

#### 3.2.1 สถาปัตยกรรมของระบบ (System Architecture)



ภาพที่ 15 สถาปัตยกรรมโดยรวมของระบบ

อ้างอิงจากภาพที่ 15 ภาพสถาปัตยกรรมของการประยุกต์ใช้โปรโตคอล ONVIF เพื่อรับการจัดการกล้องไอพีและการสตรีมวิดีโอผ่านเว็บไซต์ นั่นเองค์ประกอบดังนี้

1. Input จะเป็นการรับค่าของมูลจากกล้องไอพี โดยจะเรียกผ่านมาตรฐาน 2 มาตรฐานคือ 1. HTTP และ 2. ONVIF
2. Process จะเป็นส่วนของการประมวลผลโดยเป็นการเรียกใช้งานผ่าน API ที่เขียนด้วยภาษา Golang โดยจะมี 2 ส่วนการทำงานหลัก ๆ คือ

### 2.1 ส่วนของการเรียกภาพและเรียกข้อมูลเมตาจากกล้องไอพี

โดยจะเป็นการเรียกภาพการสตรีมจากกล้องไอพีด้วยมาตรฐานของ HTTP โดยข้อมูลภาพจะถูกส่งต่อไปยังส่วนของ Output เพื่อแสดงภาพการสตรีมวิดีโอด้วยกล้องไอพีต่อไป และส่วนของการเรียกข้อมูลเมตาจากกล้องไอพีจะทำการเรียกข้อมูลผ่านมาตรฐาน ONVIF โดยจะมีการใช้งานทั้งหมด 8 พัฟก์ชัน คือ

- 1) GetDeviceInformation
- 2) GetUser
- 3) GetSystemDateAndTime
- 4) GetCapabilities
- 5) GetHostname
- 6) GetNetworkProtocols
- 7) GetDiscoveryMode
- 8) GetServiceCapabilities

โดยข้อมูลเมตาที่เรียกได้จากการสตรีมจะถูกส่งต่อไปยังส่วนของ Output เพื่อแสดงข้อมูลเมตาจากกล้องไอพีต่อไป

### 2.2 ส่วนของฐานข้อมูล

โดยจะเป็นการเก็บข้อมูลลงในฐานข้อมูล แบ่งเป็น 2 ส่วน คือ

- 1) เก็บข้อมูลของผู้ใช้งานที่ได้ทำการลงทะเบียนไว้ เพื่อใช้ในการเข้าสู่ระบบ
- 2) เก็บข้อมูลของที่อยู่ไอพีของกล้องไอพีที่เพิ่มเข้ามาภายในระบบ

3. Output จะเป็นส่วนของการแสดงผลออกที่หน้าเว็บไซต์ โดยจะเป็นการแสดงภาพจากการสตรีมวิดีโอด้วยกล้องไอพีผ่านมาตรฐาน HTTP และแสดงข้อมูลเมตาจากมาตรฐาน ONVIF ที่ถูกแปลงค่าข้อมูลให้อยู่ในรูปของ JSON

### 3.2.2 การวิเคราะห์และออกแบบ API

จากการศึกษาถึงปัญหาและความต้องการของระบบและการวิเคราะห์ ผู้ใช้งานสามารถออกแบบ API สำหรับเว็บเซอร์วิส ได้ดังตารางที่ 4

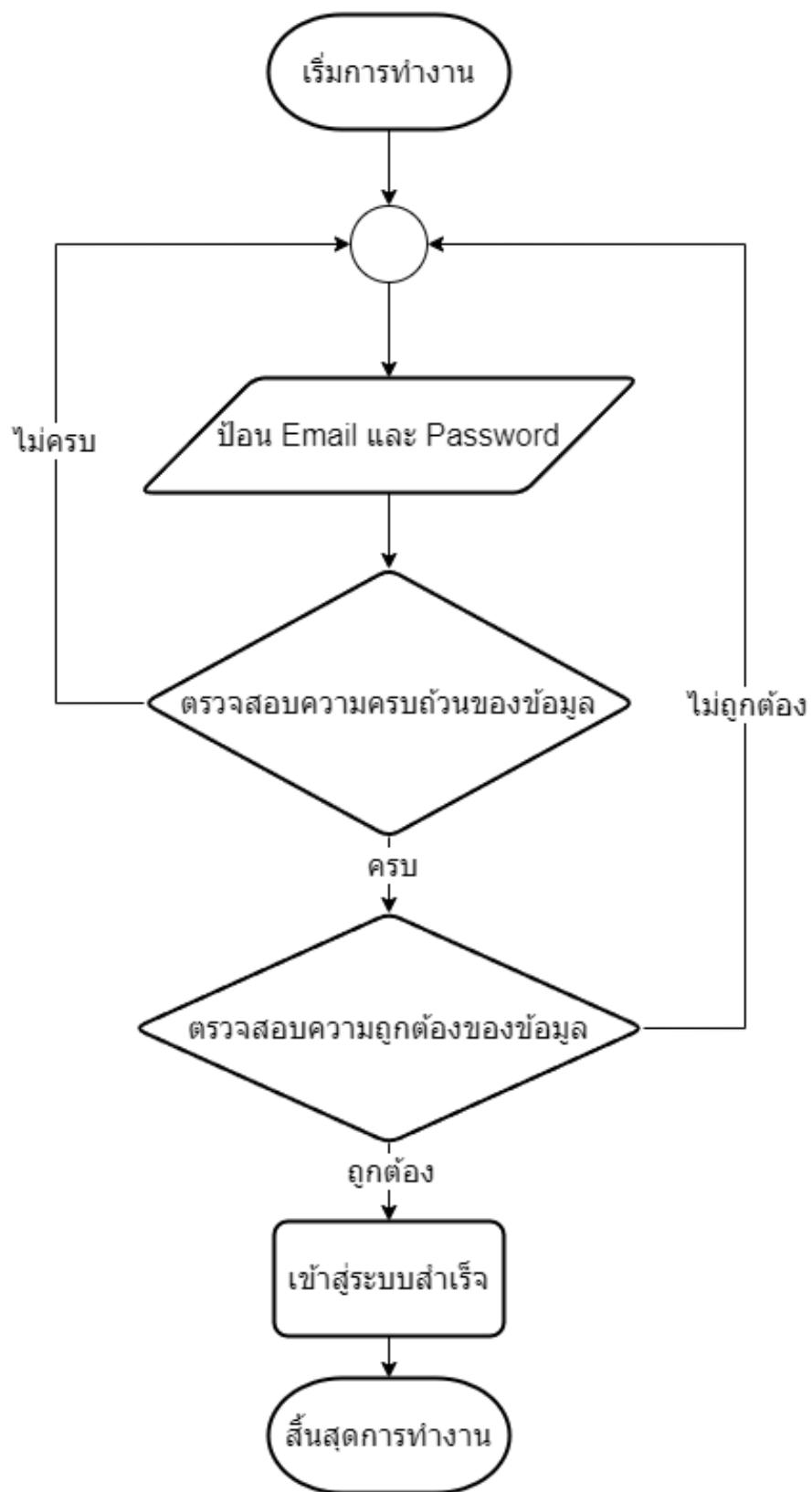
ตารางที่ 4 แสดงรายการของ API ทั้งหมดภายในโครงงานนี้

ลำดับ	API		คำอธิบาย
<b>การเข้าสู่ระบบ</b>			
1	GET	/	แสดงหน้าหลักในระบบ
2	GET	/cameras	แสดงหน้าจัดการกล้องไอพี
3	GET	/dashboard	แสดงหน้าแสดงข้อมูลเมตาจากกล้องไอพี
4	GET	/about	แสดงหน้าข้อมูลตารางการทำงานของโครงงาน
5	GET	/portfolio	แสดงหน้าข้อมูลของนักพัฒนา
6	GET	/login	แสดงหน้าการเข้าสู่ระบบ
7	POST	/loginauth	การใส่ข้อมูล email และ password เพื่อเข้าระบบ
8	GET	/signup	แสดงหน้าการลงทะเบียนสร้างบัญชีผู้ใช้
9	POST	/signupauth	การใส่ข้อมูลเพื่อลงทะเบียนสร้างบัญชีผู้ใช้
10	GET	/forgot_pass	แสดงหน้าลืมรหัสผ่าน
11	POST	/code_verify	ตรวจสอบรหัสยืนยัน เพื่อเปลี่ยนรหัสผ่านใหม่
12	POST	/forgot_pass_auth	ตรวจสอบอีเมลที่ใช้ในการส่งรหัสยืนยัน
13	POST	/checkpass	สร้างและตรวจสอบรหัสผ่านใหม่
14	-	/ “เรียก API ที่ไม่มีอยู่”	เมื่อเขียนเรียก API ที่ไม่ได้มีการสร้างไว้ จะแสดงหน้า 404 Not Found แจ้งเตือน
<b>การจัดการกล้องไอพี</b>			
1	POST	/add_camera	การเพิ่มกล้องไอพี
2	POST	/remove_camera	การลบกล้องไอพี
<b>การแสดงข้อมูลเมตาจากกล้องไอพี</b>			
1	POST	/metedatas	การแสดงข้อมูลเมตาจากกล้องไอพี

### 3.2.3 การอธิบาย Flow Chart การทำงานของระบบ

อธิบายผังงานจากภาพที่ 16 ผังการทำงานการเข้าสู่ระบบ

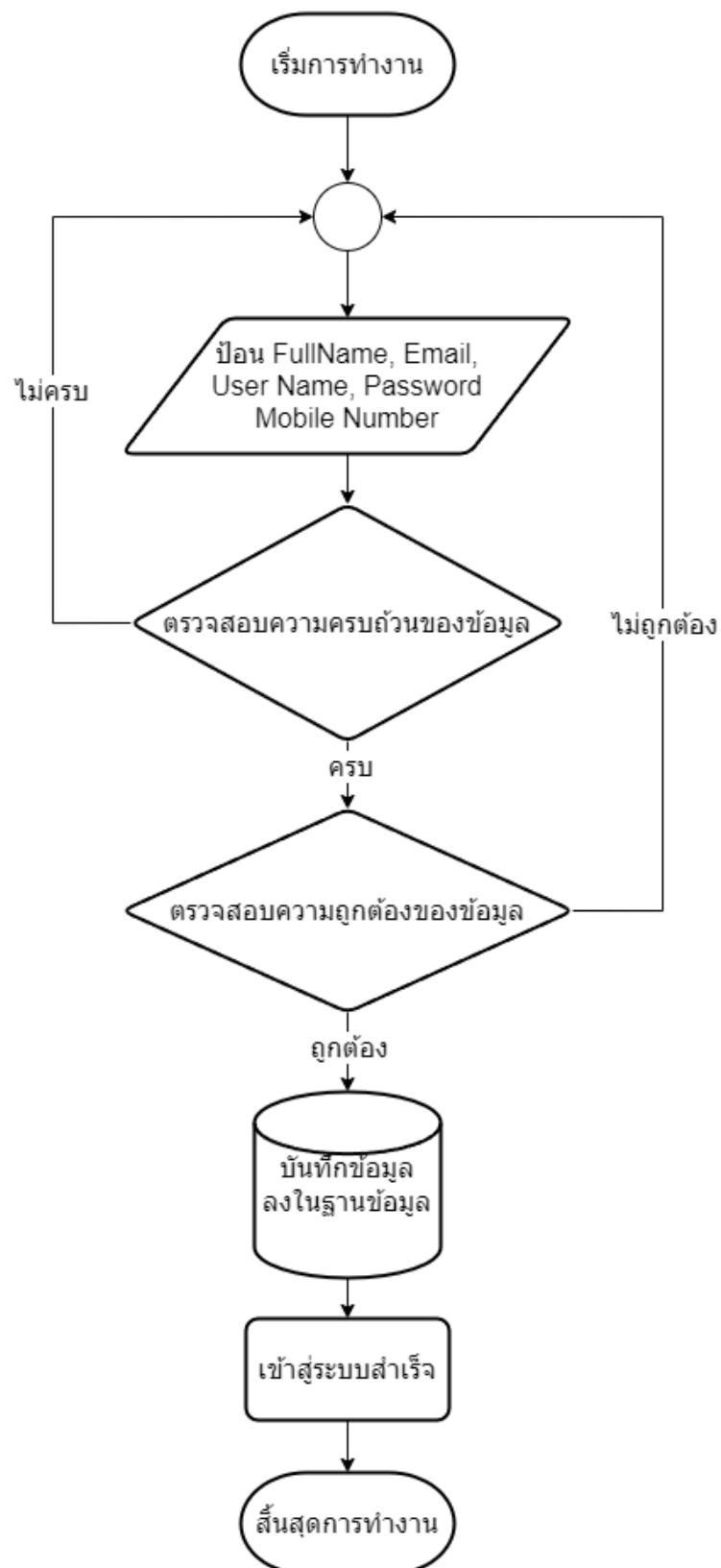
1. เมื่อเริ่มต้นการทำงานที่หน้าเว็บไซต์ ระบบจะแสดงหน้า Login เพื่อให้ผู้ใช้ทำ การกรอกข้อมูลเพื่อเข้าใช้งานภายในระบบ
2. ระบบจะให้ผู้ใช้ทำการป้อนอีเมลและรหัสผ่าน
3. ระบบจะทำการตรวจสอบความครบถ้วนของข้อมูลที่ผู้ใช้ได้กรอก
4. ระบบตรวจสอบว่าข้อมูลยังไม่ครบผู้ใช้จะต้องทำการกรอกข้อมูลใหม่
5. ระบบตรวจสอบว่าข้อมูลครบจะมีการตรวจสอบว่ามีข้อมูลผู้ใช้ตรงกับในฐานข้อมูลหรือไม่ถ้าตรงก็จะทำการพาผู้ใช้เข้าสู่ระบบ
6. ระบบตรวจสอบว่าข้อมูลไม่มีในฐานข้อมูลผู้ใช้จะต้องทำการกรอกข้อมูลใหม่
7. สิ้นสุดการทำงานของการเข้าสู่ระบบ



ภาพที่ 16 ผังการทำงานการเข้าสู่ระบบ

อธิบายผังงานจากภาพที่ 17 ผังการทำงานการลงทะเบียนผู้ใช้งาน

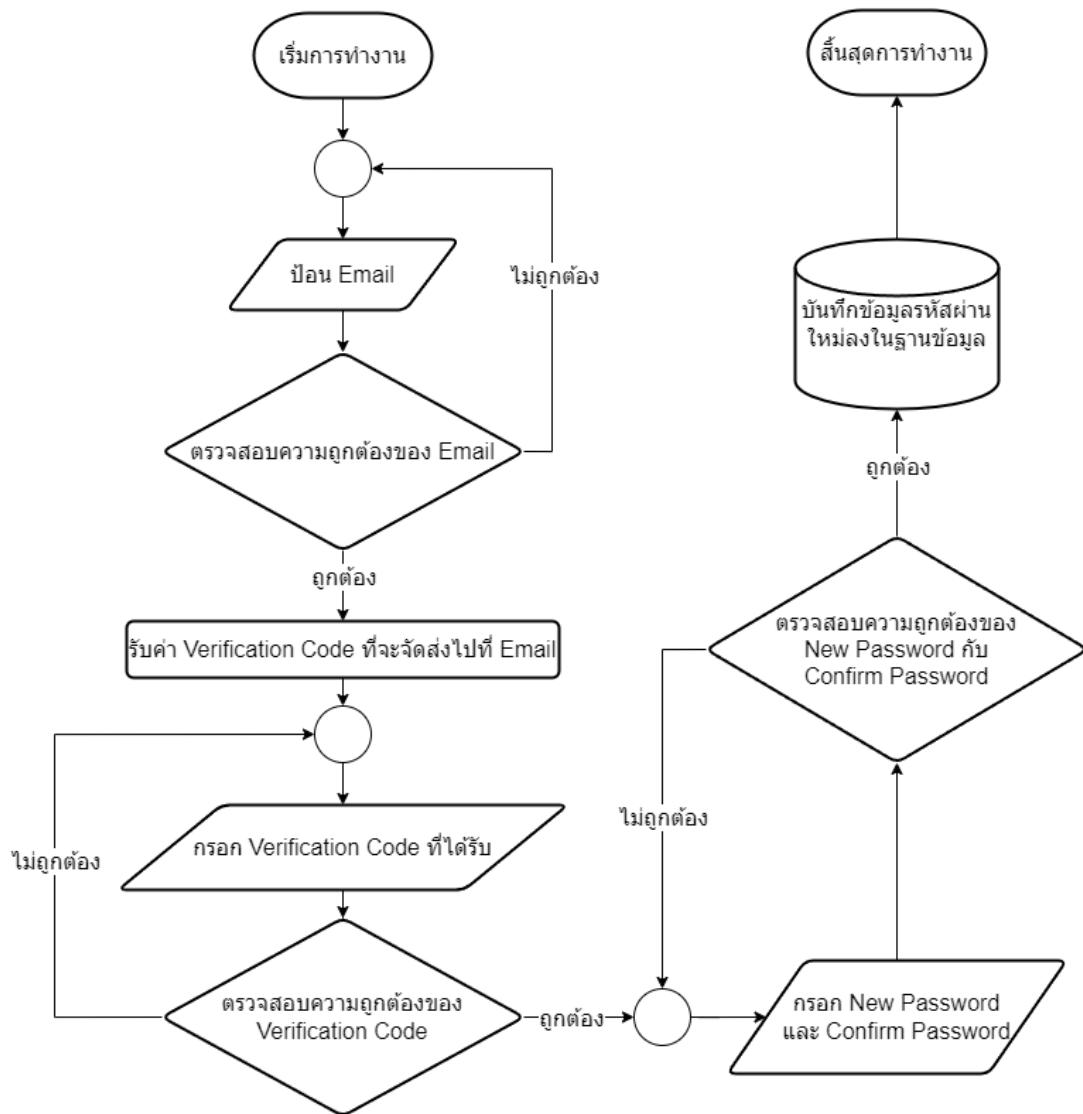
1. เมื่อเริ่มต้นการทำงานระบบจะแสดงหน้า Sign up เพื่อให้ผู้ใช้งานใหม่ได้ทำการลงทะเบียนเพื่อใช้ Login ในการเข้าสู่ระบบ
2. ระบบจะให้ผู้ใช้ใหม่ทำการกรอกข้อมูลของ ชื่อ, อีเมล, ชื่อผู้ใช้งาน, รหัสผ่าน และเบอร์โทรศัพท์
3. ระบบจะทำการตรวจสอบความครบถ้วนของข้อมูลที่ผู้ใช้ได้กรอก
4. ระบบตรวจพบว่าข้อมูลยังไม่ครบผู้ใช้จะต้องทำการกรอกข้อมูลใหม่
5. ระบบตรวจพบว่าข้อมูลครบจะมีการตรวจสอบว่ามีข้อมูลผู้ใช้ตรงกับในฐานข้อมูลหรือไม่ถ้าตรงกับที่มีอยู่ในระบบผู้ใช้จะต้องทำการกรอกใหม่
6. ระบบตรวจพบว่าข้อมูลที่กรอกมาถูกต้องไม่ซ้ำกับในฐานข้อมูลจะทำการบันทึกข้อมูลผู้ใช้ใหม่ลงในฐานข้อมูลและพาผู้ใช้เข้าสู่ระบบ
7. สิ้นสุดการทำงาน



ภาพที่ 17 ผังการทำงานการลงทะเบียนผู้ใช้งาน

อธิบายผังงานจากภาพที่ 18 ผังการทำงานการกรอกรหัสผ่านใหม่สำหรับผู้ล็อกอินรหัส

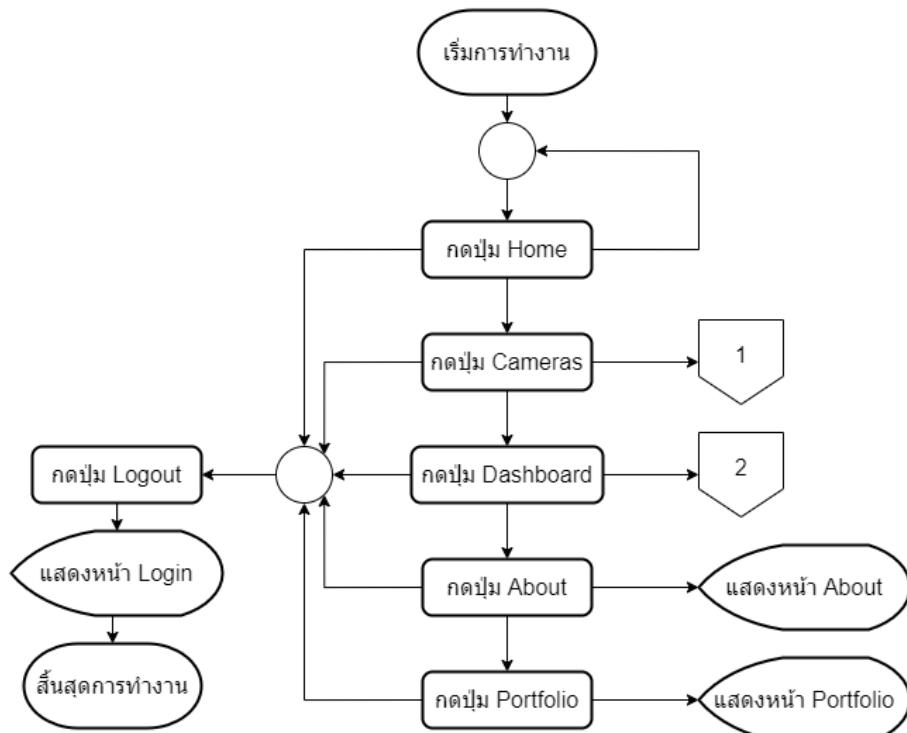
1. เมื่อเริ่มต้นการทำงานระบบจะแสดงหน้า Forgot Password เพื่อให้ผู้ใช้งานได้ทำการสร้างรหัสผ่านใหม่เพื่อใช้ Login ในการเข้าสู่ระบบ
2. ระบบจะให้ผู้ใช้ทำการกรอกข้อมูลของ อีเมล เพื่อใช้ในการส่งรหัส Verification Code ไปยังอีเมลที่ได้ระบุเอาไว้
3. ระบบจะทำการตรวจสอบความถูกต้องของอีเมลที่ผู้ใช้ได้ทำการกรอกหากพบว่าไม่ถูกต้องผู้ใช้จะต้องทำการกรอกอีเมลใหม่ หากกรอกถูกต้องตรงกับที่มีในฐานข้อมูลระบบจะทำการส่งรหัสไปยังอีเมลนั้น
4. เปิดดูที่อีเมลที่ได้ทำการระบุจะพบรหัส Verification Code
5. ให้ผู้ใช้งานนำรหัส Verification Code ที่ได้รับนำมากรอก
6. ระบบจะทำการตรวจสอบความถูกต้องของรหัส Verification Code กับระบบว่าตรงกันหรือไม่หากไม่ตรงกันผู้ใช้จะต้องทำการกรอกใหม่ หากตรงกันระบบจะพาไปหน้าให้กรอกรหัสผ่านใหม่
7. ผู้ใช้ทำการกรอกรหัสผ่านใหม่ โดยจะกรอกรหัสผ่านใหม่และกรอกรหัสครั้งที่สองเพื่อยืนยันรหัสผ่านใหม่ที่ผู้ใช้จะใช้งาน
8. ระบบจะทำการตรวจสอบว่ารหัสผ่านใหม่ที่ผู้ใช้กรอกกับรหัสที่ยืนยันนั้นตรงกันหรือไม่ หากไม่ตรงกันผู้ใช้จะต้องกรอกรหัสผ่านใหม่อีกครั้ง
9. หากรหัสผ่านกับรหัสที่ยืนยันนั้นตรงกันระบบจะทำการบันทึกรหัสผ่านใหม่ลงในฐานข้อมูลและพาผู้ใช้งานเข้าสู่ระบบ
10. สิ้นสุดการทำงาน



ภาพที่ 18 ผังการทำงานการกรอกรหัสผ่านใหม่สำหรับผู้ลืมรหัส

อธิบายผังงานจากภาพที่ 19 ผังการทำงานภายในระบบ

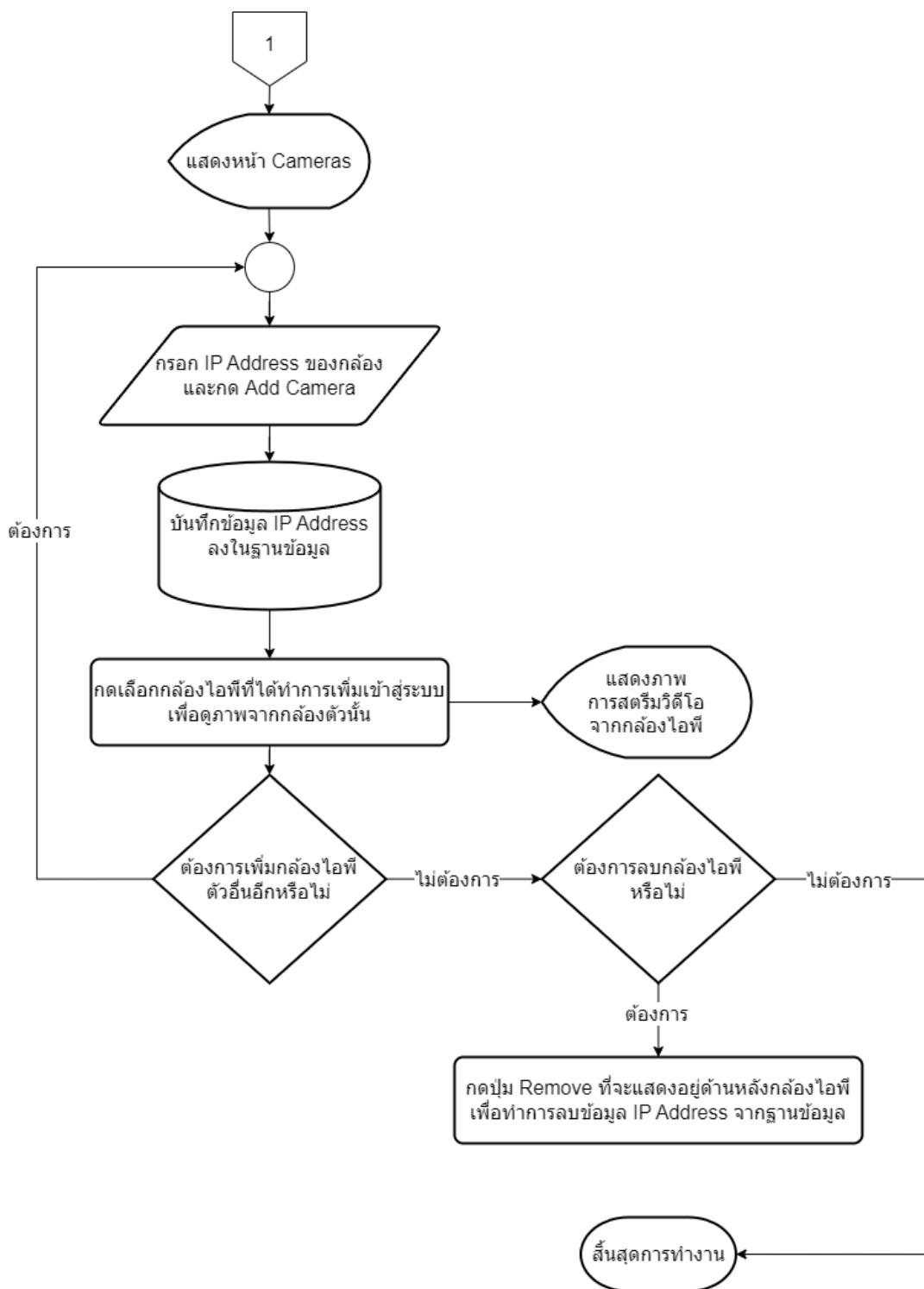
1. เมื่อเริ่มต้นการทำงานระบบจะแสดงหน้า Home ให้กับผู้ใช้งาน หลังจากที่ผู้ใช้งานได้ทำการ Login เข้าสู่ระบบเรียบร้อยแล้ว
2. ระบบจะมีการแสดงปุ่มทั้งหมด 6 ปุ่มที่แถบนำทางในหน้า Home
3. เมื่อผู้ใช้งานกดปุ่ม Home ไม่ว่าจะในหน้าไหน ระบบจะทำการพาผู้ใช้กลับมาในหน้า Home เสมอ
4. เมื่อผู้ใช้งานกดปุ่ม Cameras ไม่ว่าจะในหน้าไหน ระบบจะทำการพาผู้ใช้กลับมาในหน้า Cameras เสมอและดูคำอธิบายของหน้านี้ต่อได้ในหน้าที่ 33
5. เมื่อผู้ใช้งานกดปุ่ม Dashboard ไม่ว่าจะในหน้าไหน ระบบจะทำการพาผู้ใช้กลับมาในหน้า Dashboard เสมอและดูคำอธิบายของหน้านี้ต่อได้ในหน้าที่ 35
6. เมื่อผู้ใช้งานกดปุ่ม About ไม่ว่าจะในหน้าไหน ระบบจะทำการพาผู้ใช้กลับมาในหน้า About เสมอ
7. เมื่อผู้ใช้งานกดปุ่ม Portfolio ไม่ว่าจะในหน้าไหน ระบบจะทำการพาผู้ใช้กลับมาในหน้า Portfolio เสมอ
8. เมื่อผู้ใช้งานกดปุ่ม Logout ไม่ว่าจะในหน้าไหน ระบบจะทำการพาผู้ใช้ออกจากระบบกลับมาในหน้า Login เสมอ



ภาพที่ 19 ผังการทำงานภายในระบบ

อธิบายผังงานจากภาพที่ 20 ผังการทำงานภายในหน้า Cameras

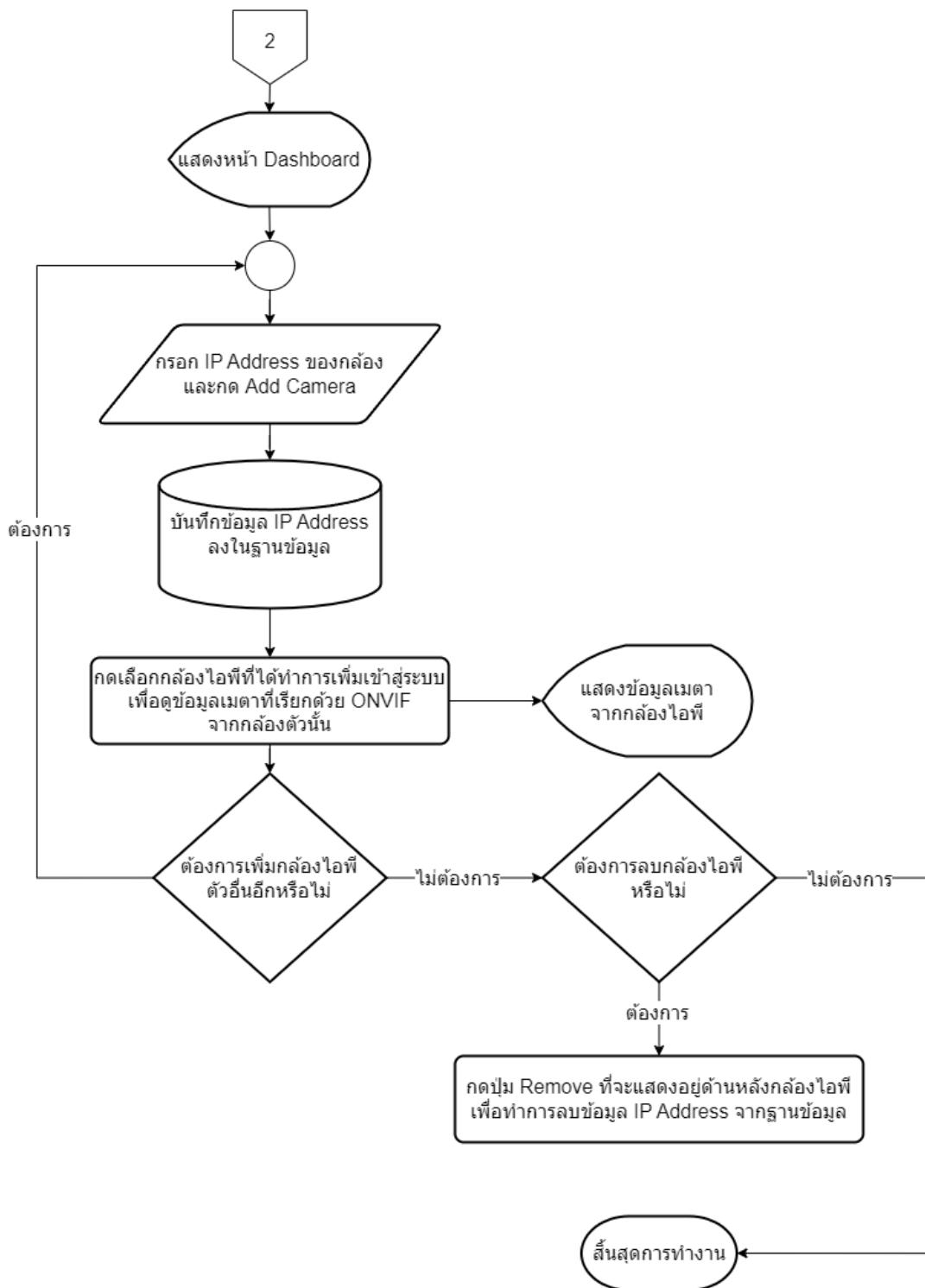
1. เมื่อเริ่มต้นการทำงานระบบจะแสดงหน้า Cameras ให้กับผู้ใช้งาน หลังจากที่ผู้ใช้งานได้ทำการกดปุ่ม Cameras ที่แถบนำทางแล้วในภาพที่ 19
2. ระบบจะทำการแสดงหน้า Cameras ให้กับผู้ใช้
3. ระบบจะแสดงกล่องให้ผู้ใช้ได้สามารถกรอกที่อยู่ IP Address ของกล้องไอพีจากการนั้นให้ผู้ใช้กดปุ่ม Add Camera เพื่อเพิ่มข้อมูลที่อยู่ IP Address
4. ระบบจะทำการเพิ่มข้อมูลของที่อยู่ IP Address ที่ผู้ใช้ได้กรอกลงในฐานข้อมูล
5. ระบบจะแสดงที่อยู่ IP Address ที่ผู้ใช้งานได้กรอก
6. หากผู้ใช้งานได้ทำการกดที่ที่อยู่ IP Address ที่แสดงอยู่ระบบจะแสดงภาพการสตรีมวิดีโอจากกล้องตัวนั้น
7. หากผู้ใช้งานมีความต้องการจะทำการเพิ่มที่อยู่ IP Address ของกล้องตัวอื่นผู้ใช้สามารถเริ่มทำซ้ำได้ที่ข้อ 3
8. หากผู้ใช้งานมีความต้องการในการลบที่อยู่ IP Address ของกล้องไอพีตัวใดผู้ใช้งานสามารถกดปุ่ม Remove ที่จะแสดงอยู่ด้านหลังของเลขที่อยู่ IP Address นั้น ๆ ได้เลย
9. หากผู้ใช้งานไม่ต้องการเพิ่มหรือลบที่อยู่ IP Address ของกล้องจะเป็นอันสิ้นสุดการทำงาน



ภาพที่ 20 ผังการทำงานภายในหน้า Cameras

อธิบายผังงานจากภาพที่ 21 ผังการทำงานภายในหน้า Dashboard

1. เมื่อเริ่มต้นการทำงานระบบจะแสดงหน้า Dashboard ให้กับผู้ใช้งาน หลังจากที่ผู้ใช้งานได้ทำการกดปุ่ม Dashboard ที่แนบมาทางแล้วในภาพที่ 19
2. ระบบจะทำการแสดงหน้า Dashboard ให้กับผู้ใช้
3. ระบบจะแสดงกล่องให้ผู้ใช้ได้สามารถกรอกที่อยู่ IP Address ของกล้องไอพีจากการนั้นให้ผู้ใช้กดปุ่ม Add Camera เพื่อเพิ่มข้อมูลที่อยู่ IP Address
4. ระบบจะทำการเพิ่มข้อมูลของที่อยู่ IP Address ที่ผู้ใช้ได้กรอกลงในฐานข้อมูล
5. ระบบจะแสดงที่อยู่ IP Address ที่ผู้ใช้งานได้กรอก
6. หากผู้ใช้งานได้ทำการกดที่ที่อยู่ IP Address ที่แสดงอยู่ระบบจะแสดงข้อมูลเมตาที่เรียกผ่าน ONVIF จากกล้องตัวนั้น มาแสดงในผู้ใช้ได้เห็นข้อมูล
7. หากผู้ใช้งานมีความต้องการจะทำการเพิ่มที่อยู่ IP Address ของกล้องตัวอื่นผู้ใช้สามารถเริ่มทำซ้ำได้ที่ข้อ 3
8. หากผู้ใช้งานมีความต้องการในการลบที่อยู่ IP Address ของกล้องไอพีตัวใดผู้ใช้งานสามารถกดปุ่ม Remove ที่จะแสดงอยู่ด้านหลังของเลขที่อยู่ IP Address นั้น ๆ ได้เลย
9. หากผู้ใช้งานไม่ต้องการเพิ่มหรือลบที่อยู่ IP Address ของกล้องจะเป็นอันสิ้นสุดการทำงาน



ภาพที่ 21 ผังการทำงานภายในหน้า Dashboard

### 3.2.4 การออกแบบหน้าเว็บไซต์

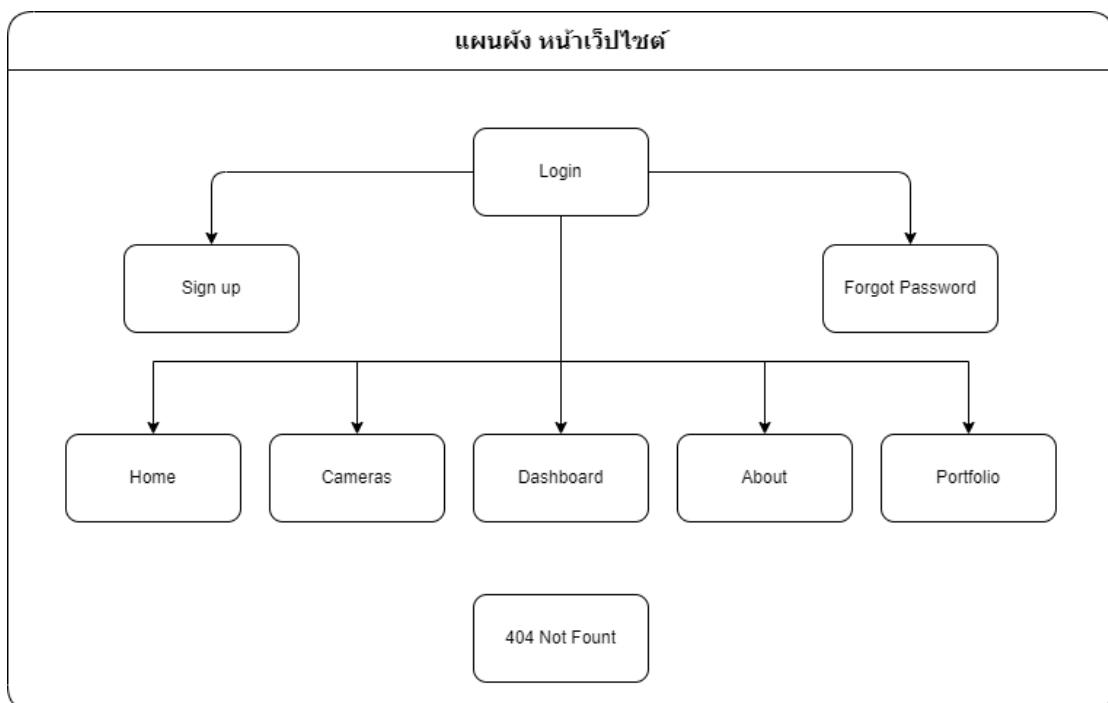
แผนผังของระบบในเว็บแอปพลิเคชัน จะประกอบไปด้วย 4 ส่วน คือ

ส่วนที่ 1 ส่วนของหน้าเข้าสู่ระบบ (Login)

ส่วนที่ 2 ส่วนของหน้าการลงทะเบียนผู้ใช้งาน (Sign up) ใช้ในการลงทะเบียนผู้ใช้งานเพื่อเข้าใช้งานภายในระบบ และ ส่วนของหน้าลืมรหัสผ่าน (Forgot Password) ใช้สำหรับผู้ที่ลืมรหัสผ่านสำหรับเข้าใช้งานภายในระบบ

ส่วนที่ 3 ส่วนที่เข้าสู่ภายในระบบแล้ว จะมีหน้า Home ซึ่งจะเป็นหน้าหลักในการอธิบายโครงงาน, หน้า Cameras ซึ่งจะเป็นหน้าในการจัดการกล้องไอพีใช้เลือกคุณภาพจากกล้อง เพิ่มกล้อง ลบกล้อง, หน้า Dashboard ใช้สำหรับแสดงข้อมูลเมตาจากกล้องไอพี, หน้า About ใช้แสดงข้อมูลตารางการทำงานของโครงงาน, หน้า Portfolio ใช้แสดงข้อมูลของนักพัฒนา

ส่วนที่ 4 ส่วนของหน้าแสดง 404 Not Fount คือ เชิร์ฟเวอร์หาสิ่งที่ร้องขอไม่พบ



ภาพที่ 22 แผนผังส่วนของเว็บแอปพลิเคชัน

### 3.3 เครื่องมือในการพัฒนาระบบ

การประยุกต์ใช้โปรโตคอล ONVIF เพื่อร้องรับการจัดการกล้องไอพีและการสตรีมวิดีโอผ่านเว็บไซต์ พัฒนาโดยจะใช้เครื่องมือต่าง ๆ ดังตารางที่ 5 และ 6

ตารางที่ 5 อุปกรณ์ฮาร์ดแวร์ที่ใช้ในการพัฒนาระบบ

ลำดับ	รายการ	รายละเอียด
1.	เครื่องคอมพิวเตอร์ส่วนบุคคล จำนวน 1 เครื่อง ซึ่งมีคุณสมบัติต่อไปนี้  จำนวน 1 เครื่อง ซึ่งมีคุณสมบัติต่อไปนี้	คอมพิวเตอร์โน้ตบุ๊ค  CPU : AMD Ryzen 5 3550H @ 2.10 GHz  RAM : 16.00 GB  Hard disk : 500GB  OS : Windows 11 Pro 64 bit ที่ลง WSL หรือ  Ubuntu 20.04.5 LTS
2.	สมาร์ทโฟนแอนดรอยด์ จำนวน 1-2 เครื่อง  ซึ่งมีคุณสมบัติต่อไปนี้	เป็นสมาร์ทโฟนแอนดรอยด์ที่ใช้ระบบปฏิบัติการ มากกว่า Android 10 ขึ้นไป เพื่อใช้งาน  Google Play Store ในการติดตั้งแอปพลิเคชัน  จำลองกล้องไอพี

ตารางที่ 6 ซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ

ลำดับ	รายการ	รายละเอียด
1.	Golang version 1.19	ทำหน้าที่เป็นภาษาหลักให้การพัฒนาระบบ
2.	IP Webcam version 1.16.6.783	ติดตั้งลงบนสมาร์ทโฟนแอนดรอยด์  เพื่อใช้ในการจำลองเป็นกล้องไอพี
3.	Visual Studio Code version 1.76	ทำหน้าที่ในการใช้พัฒนาโปรแกรม

## บทที่ 4

## ผลการดำเนินงาน

ในบทนี้ทางผู้จัยจะกล่าวถึงผลดำเนินการทดลองของการประยุกต์ใช้โปรโตคอล ONVIF เพื่อรับการจัดการกล้องไอพีและการสตรีมวิดีโอผ่านเว็บไซต์

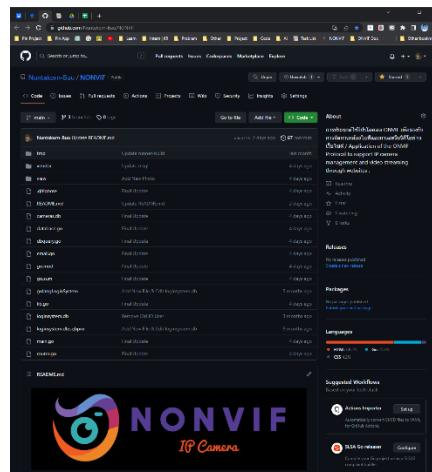
#### 4.1 วิธีการทดสอบการพัฒนาระบบ

โดยจะแบ่งการทดสอบออกเป็นทั้งหมด 4 การทดสอบ คือ

- 1) ทดสอบการติดตั้งโปรแกรม NONVIF
  - 2) ทดสอบเกี่ยวกับระบบ Login, Logout, Sign up และ Forgot Password
  - 3) ทดสอบเกี่ยวกับการเรียกสตรีมภาพวิดีโอจากกล้องไอพี
  - 4) ทดสอบเกี่ยวกับการเรียกข้อมูลเมตาจากกล้องไอพี
  - 5) ทดสอบส่วนอื่น ๆ

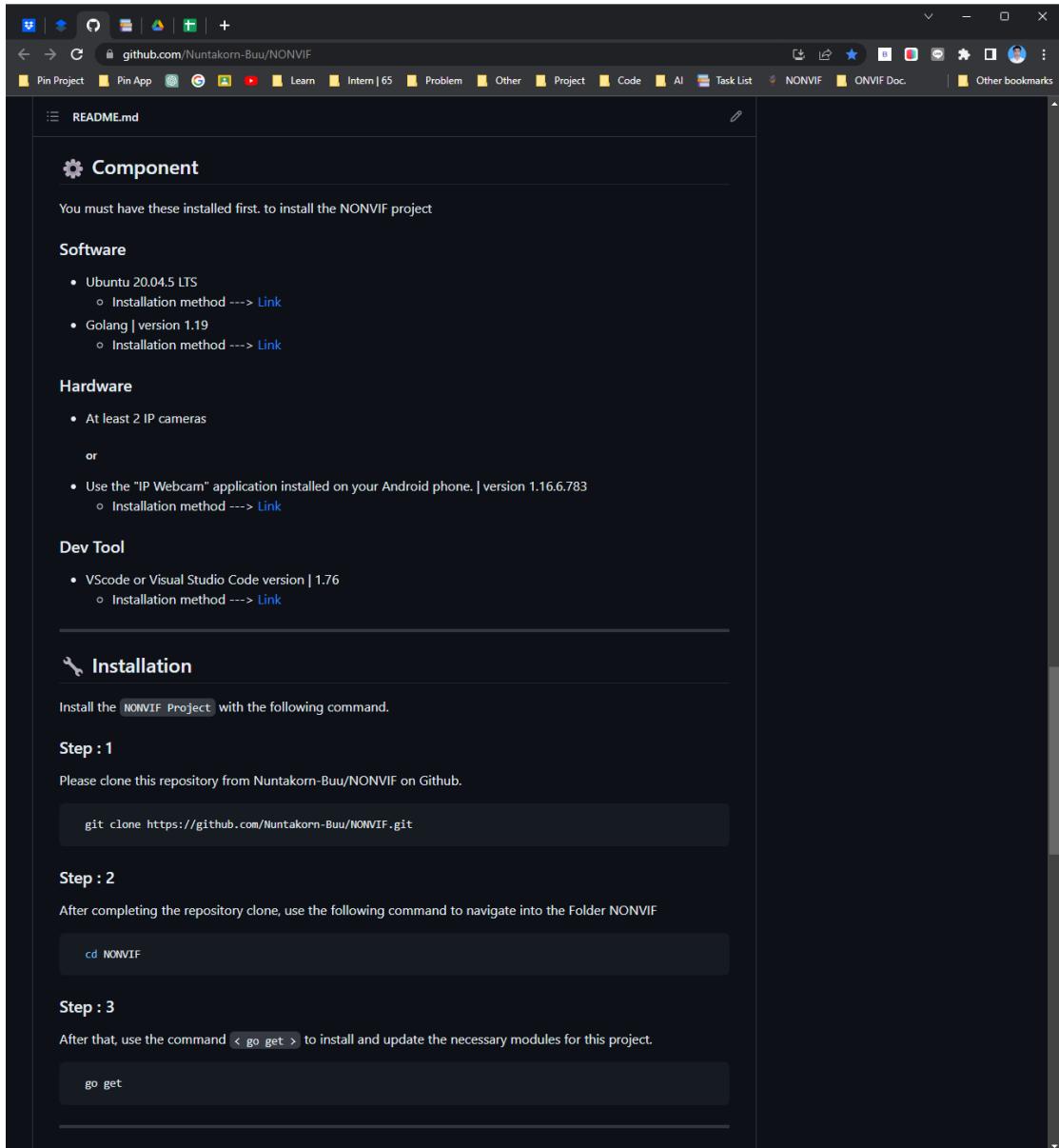
#### 4.1.1 ทดสอบการติดตั้งโปรแกรม NONVIF

ให้ผู้ใช้งานทำการเข้าไปที่ <https://github.com/Nuntakorn-Buu/NONVIF> เพื่อทำการติดตั้ง NONVIF repository เพื่อใช้งานโครงการ การประยุกต์ใช้โปรโตคอล ONVIF เพื่อรับการจัดการกล้องไอพีและการสตรีมวิดีโอผ่านเว็บไซต์



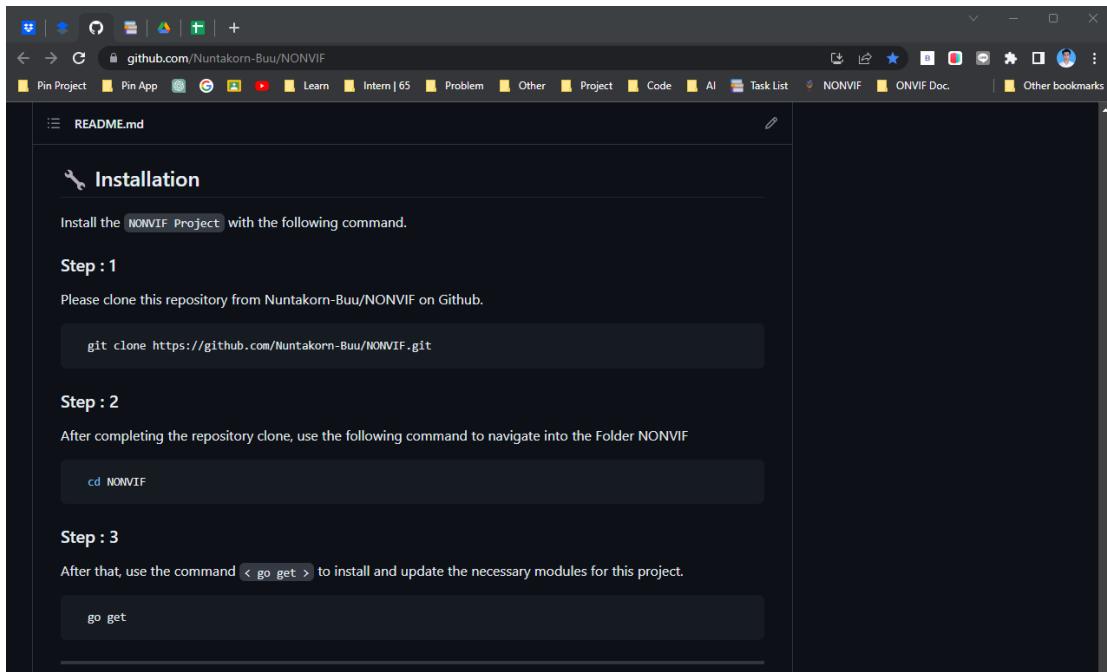
ภาพที่ 23 หน้า Github ของ Nuntakorn-Buu/NONVIF

หลังจากเข้ามาที่หน้าของ <https://github.com/Nuntakorn-Buu/NONVIF> แล้วนั้นให้ทำการเลื่อนลงไปอ่านที่ส่วนของ Component เพื่ออ่านถึงรายละเอียดสิ่งที่ต้องเตรียมก่อนจะติดตั้ง NONVIF repository ซึ่งก็จะตรงกับสิ่งที่ได้มีการอธิบายไว้แล้วในหัวข้อที่ 3.3 เครื่องมือในการพัฒนาระบบ ของบทที่ 3



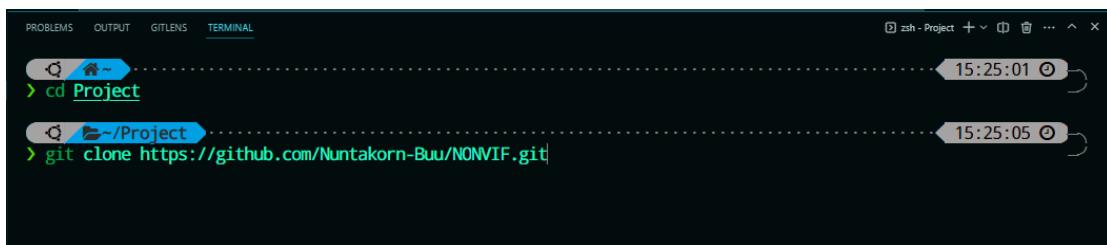
ภาพที่ 24 ส่วนของการอธิบาย Component หน้า Github ของ Nuntakorn-Buu/NONVIF

หลังจากได้อ่านและเตรียมการในส่วนของ Component เสร็จเรียบร้อยแล้ว ให้ทำการเลื่อนลงไปอ่านในส่วนของการ Installation ที่จะทำการติดตั้ง NONVIF นี้เพื่อใช้ทดสอบ



ภาพที่ 25 ส่วนของการอธิบาย Installation หน้า Github ของ Nuntakorn-Buu/NONVIF

จากรูปจะเห็นได้ว่าจะมีขั้นตอนการติดตั้งทั้งหมด 3 ขั้นตอน โดยเริ่มจากให้ทำการเปิดโปรแกรม Visual Studio Code (VScode) และเปิดหน้าต่างของ Terminal ขึ้นมาเพื่อเริ่มในขั้นตอนที่ 1 ที่จะเป็นคำสั่งที่ใช้ในการ Clone NONVIF repository มาลงที่ในเครื่องของเรา ขั้นตอนที่ 2 หลังจากเสร็จสิ้นการ Clone NONVIF repository ก็จะใช้คำสั่งเพื่อเข้าไปในไฟล์เดอร์ที่เราได้ทำการติดตั้ง ขั้นตอนที่ 3 หลังจากเข้าไปยังไฟล์เดอร์ NONVIF แล้วให้เราทำการใช้คำสั่ง go get เพื่อทำการติดตั้งและอัปเดตโมดูลที่จำเป็นของโครงการนี้



ภาพที่ 26 ภาพของขั้นตอนที่ 1 ในการใช้คำสั่งที่ใช้ในการ Clone NONVIF repository

The screenshot shows a terminal window with the following history:

- 15:25:01 cd Project
- 15:25:05 ls
- NONVIF Test
- 15:26:00 cd NONVIF
- 15:26:18 main.l3

ภาพที่ 27 ภาพของขันตอนที่ 2 ในการใช้คำสั่ง cd NONVIF

A screenshot of a terminal window within a code editor interface. The top bar shows tabs for PROBLEMS, OUTPUT, GITLENS, and TERMINAL, with TERMINAL selected. The main area displays two command-line sessions. The first session shows the command `> go get` being typed. The second session shows the command `>` and its output `6s 15:27:27`. Below these sessions, there is a blank line starting with a greater than symbol (`>`). The right side of the window includes standard window control buttons (minimize, maximize, close) and a status bar.

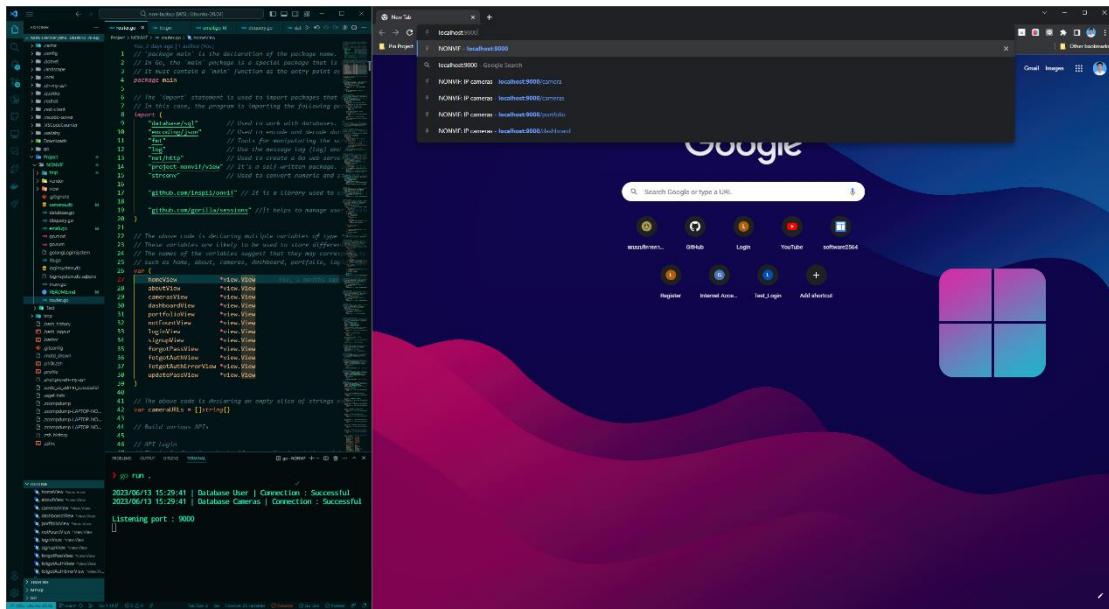
ภาพที่ 28 ภาพของขั้นตอนที่ 3 ในการใช้คำสั่ง go get

หลังจากใช้คำสั่ง go get เสร็จเรียบร้อย ไม่พบข้อความอะไรขึ้นแล้ว ต่อมาจะใช้คำสั่ง go run . การใช้คำสั่งนี้จะเป็นการสั่งที่ให้ภาษา Golang ทำการ Run ทุกไฟล์ที่มีอยู่ในโฟลเดอร์ NONVIF จากนั้นเราจะเห็นข้อความปรากฏดังภาพที่ 29 แปลว่าเซิร์ฟเวอร์ได้มีการทำงานอยู่ที่พอร์ท 9000 ให้ไปที่ Web Browser และทำการพิมพ์ <http://localhost:9000/> จะพบหน้าต่างหน้า Login ปรากฏขึ้นมา

```
~/Project/NONVIF main !15 ?3 14:09:03
> go run .
2023/06/10 14:10:06 | Database User | Connection : Successful
2023/06/10 14:10:06 | Database Cameras | Connection : Successful

Listening port : 9000
```

ภาพที่ 29 ภาพหลังจากการใช้คำสั่ง go run .

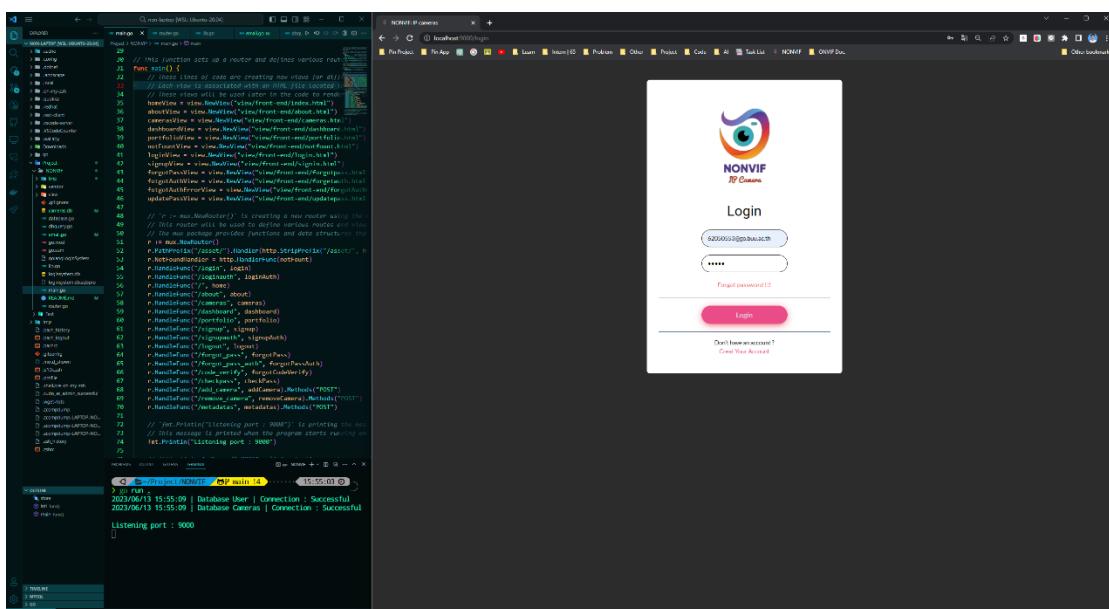


ภาพที่ 30 ภาพการเปิด Web Browser แล้วทำการพิมพ์ <http://localhost:9000/>

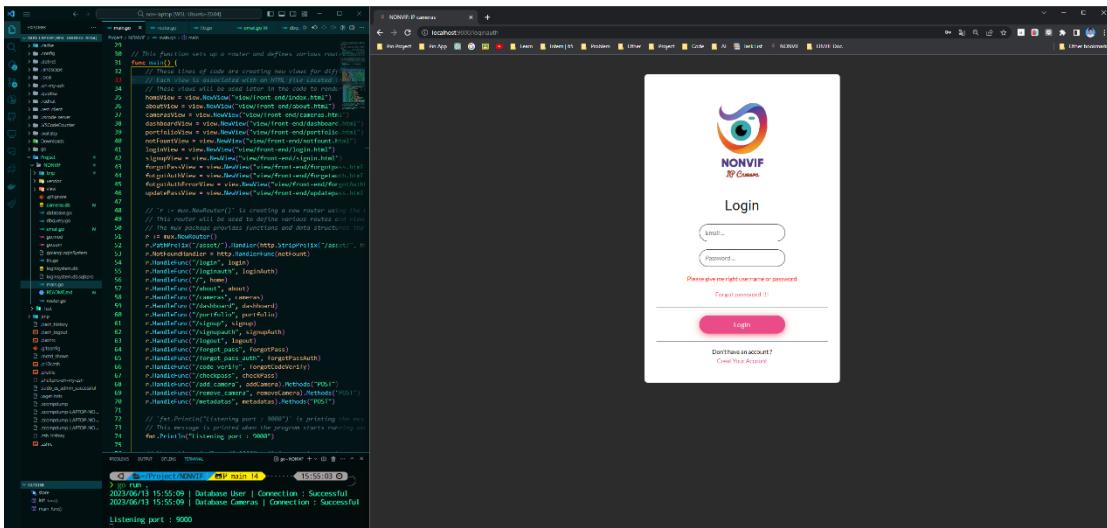
#### 4.1.2 ทดสอบเกี่ยวกับระบบ Login, Logout, Signup และ Forgot Password

ในส่วนนี้จะเป็นการทดสอบเกี่ยวกับระบบของการ Login, Logout, Sign up และ Forgot Password ว่าการทำงานในแต่ละส่วนนั้นได้ผลลัพธ์อย่างไร

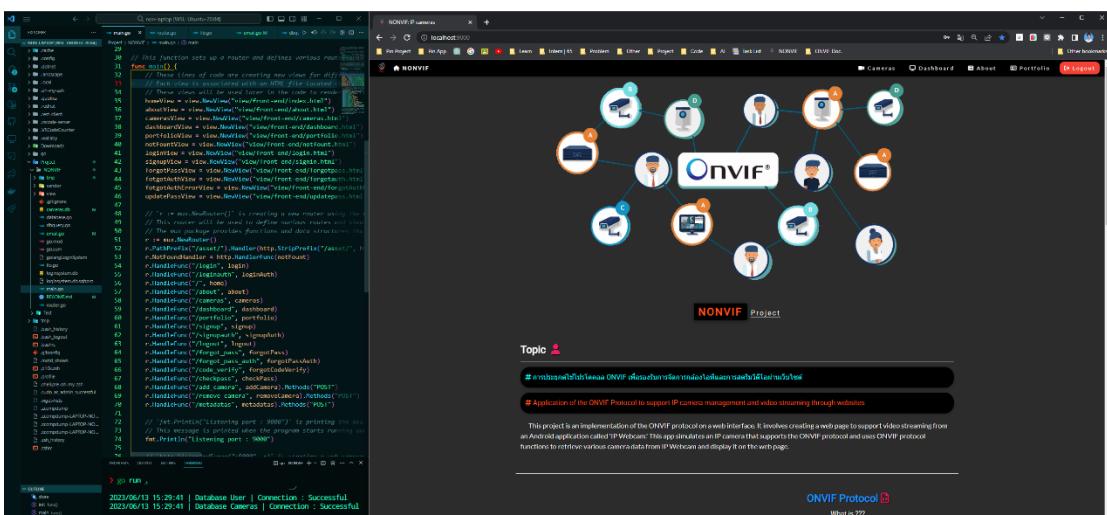
##### 4.1.2.1 ทดสอบเกี่ยวกับการ Login



ภาพที่ 31 ภาพการกรอกข้อมูลเพื่อทำการ Login เข้าสู่ระบบ



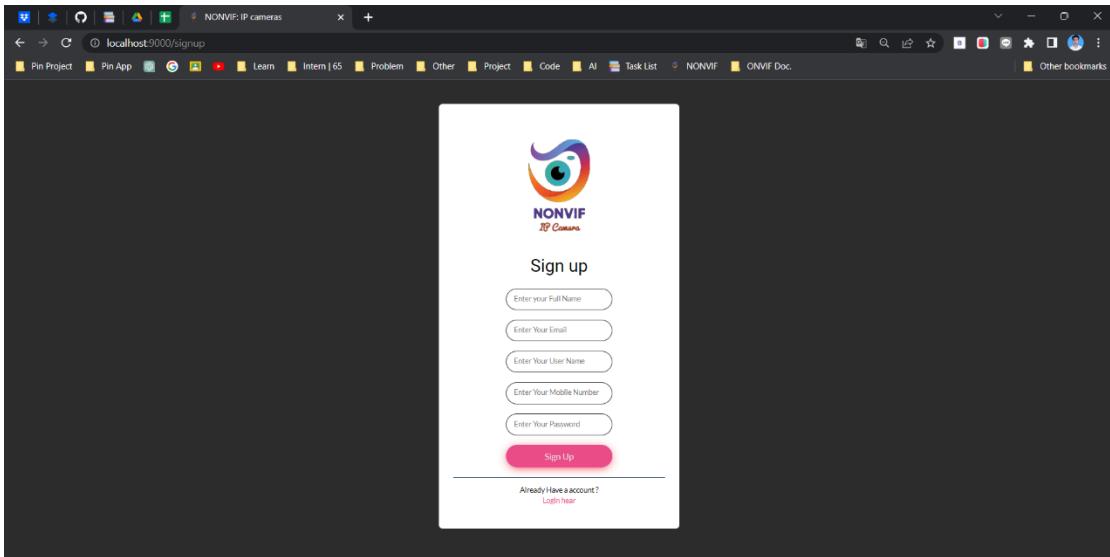
ภาพที่ 32 ภาพของการที่ทำการ Login และไม่สำเร็จ



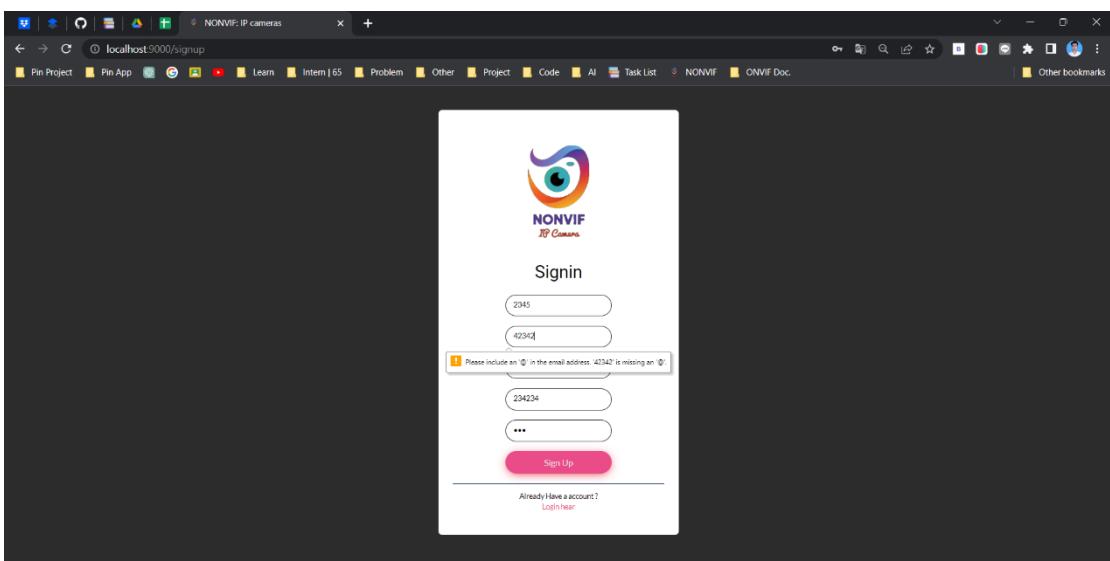
ภาพที่ 33 ภาพของการที่ทำการ Login และสำเร็จ

จากการทดสอบการ Login เข้าสู่ระบบ จากภาพที่ 31 จะเป็นการใช้ข้อมูลของ Email และ Password เพื่อทำการเข้าสู่ระบบ หากใส่ข้อมูลที่ไม่ถูกต้องจะขึ้นข้อความว่า Please give me right username or password ดังภาพที่ 32 ซึ่งเกิดจาก 2 สาเหตุคือ 1) กรอก Email หรือ Password ผิดพลาดให้แก้ไขด้วยการกรอกข้อมูลใหม่อีกครั้ง 2) เกิดจากการที่ไม่มีข้อมูลอยู่ในฐานข้อมูลเนื่องจากยังไม่มีการลงทะเบียนเอาไว้ ให้ทำการคลิกที่ 'ข้อความด้านล่างคำว่า Create Your Account เพื่อทำการสร้างบัญชีแล้วทำการ Login ใหม่อีกครั้ง หากทำการ Login สำเร็จจะพบหน้าต่างดังภาพที่ 33

#### 4.1.2.2 ทดสอบเกี่ยวกับการ Sign up



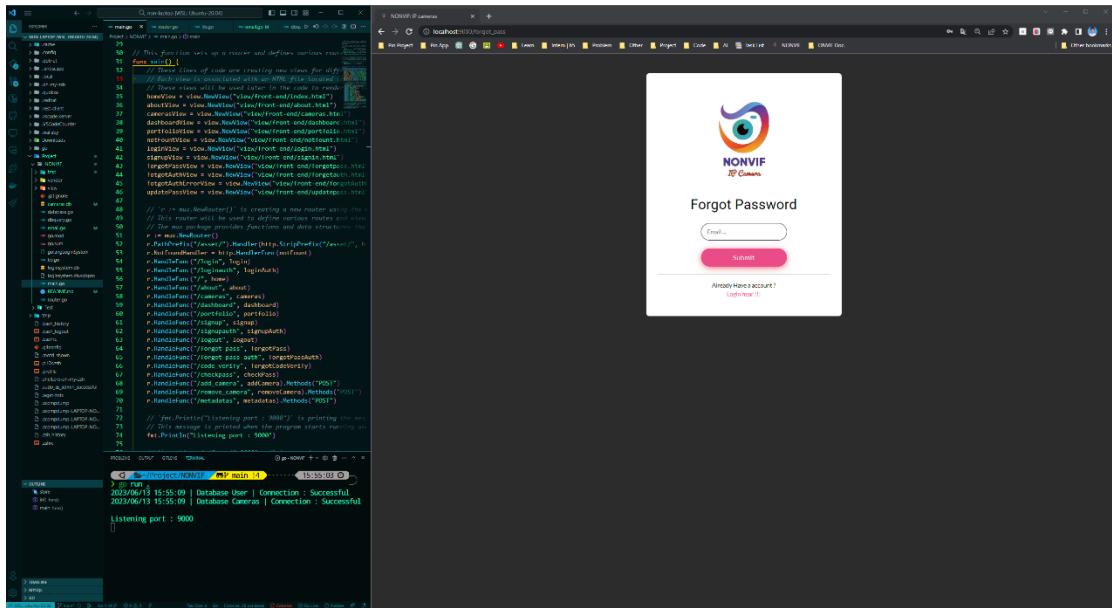
ภาพที่ 34 ภาพในหน้า Sign up เพื่อทำการลงทะเบียน



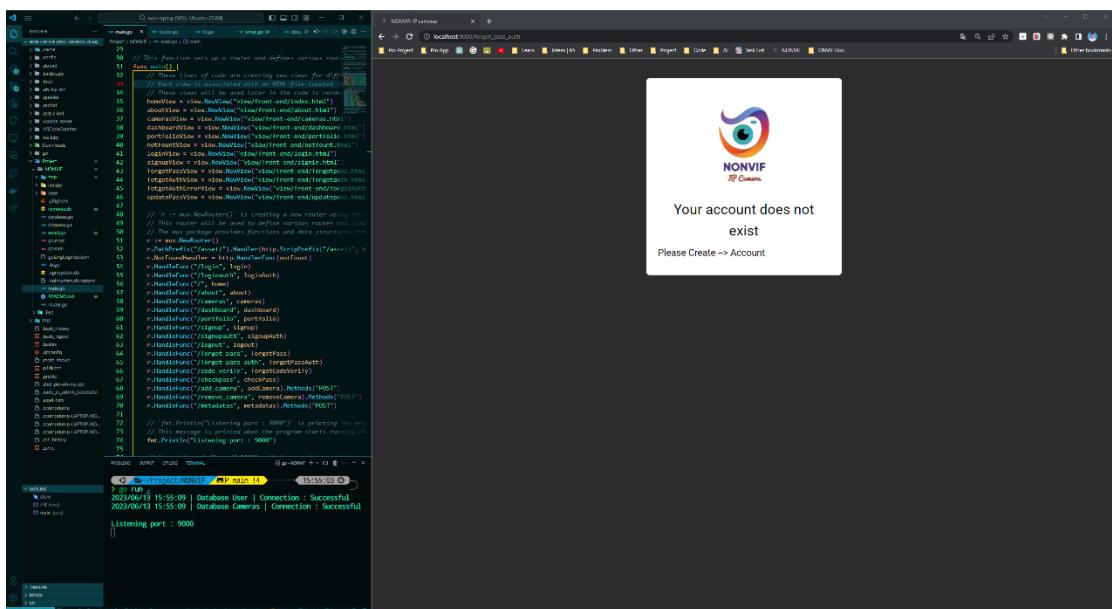
ภาพที่ 35 ภาพของการกรอกข้อมูลผิดแล้วจะมีข้อความแจ้งเตือน

จากการทดสอบการ Sign up เพื่อการลงทะเบียนในการเข้าสู่ระบบ ให้ผู้ใช้ทำการกรอกข้อมูลให้ครบถ้วนในหน้า Sign up ดังภาพที่ 34 แต่หากผู้ใช้ทำการกรอกข้อมูลผิดพลาดจะมีข้อความแจ้งเตือนดังภาพที่ 35 ให้ผู้ใช้ทำการกรอกข้อมูลการลงทะเบียนให้ถูกต้องตามคำแนะนำที่แสดง หากผู้ใช้ทำการกรอกข้อมูลสำเร็จระบบจะทำการพาผู้ใช้เข้าสู่หน้า Home ของระบบ ดังภาพที่ 33

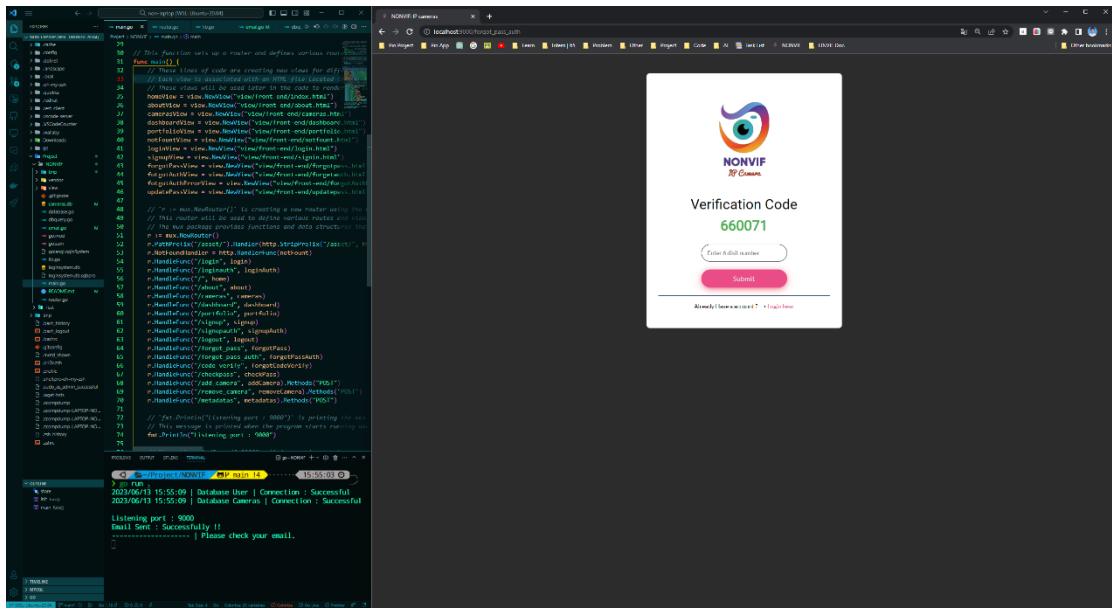
#### 4.1.2.3 ทดสอบเกี่ยวกับการ Forgot Password



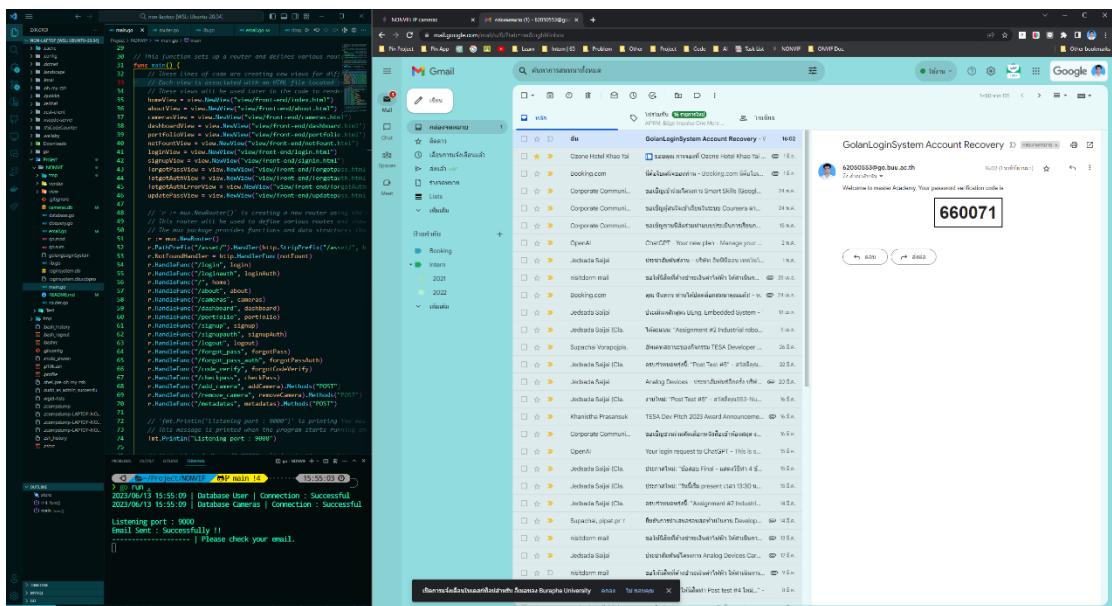
ภาพที่ 36 ภาพในหน้า Forgot Password



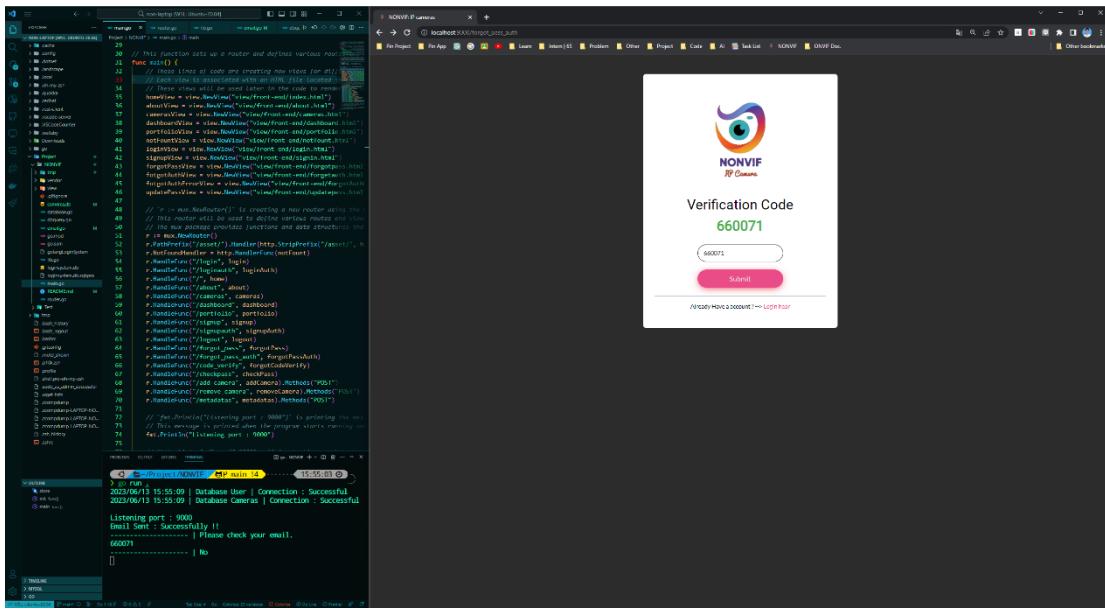
ภาพที่ 37 ภาพการใส่ Email ผิดในหน้า Forgot Password



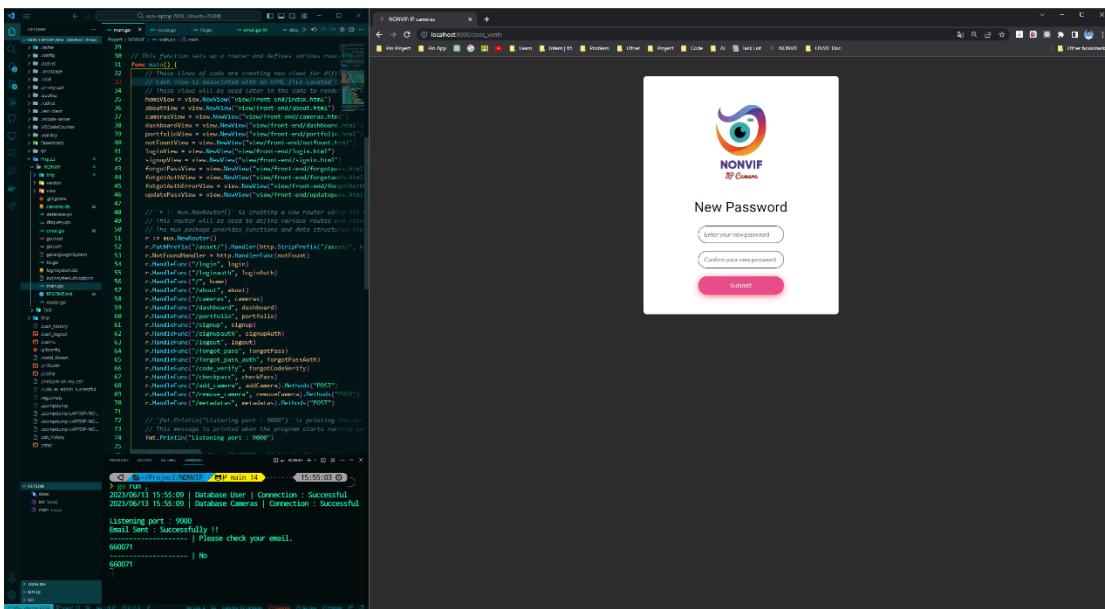
ภาพที่ 38 ภาพหลังการใส่ Email ถูกต้องในหน้า Forgot Password



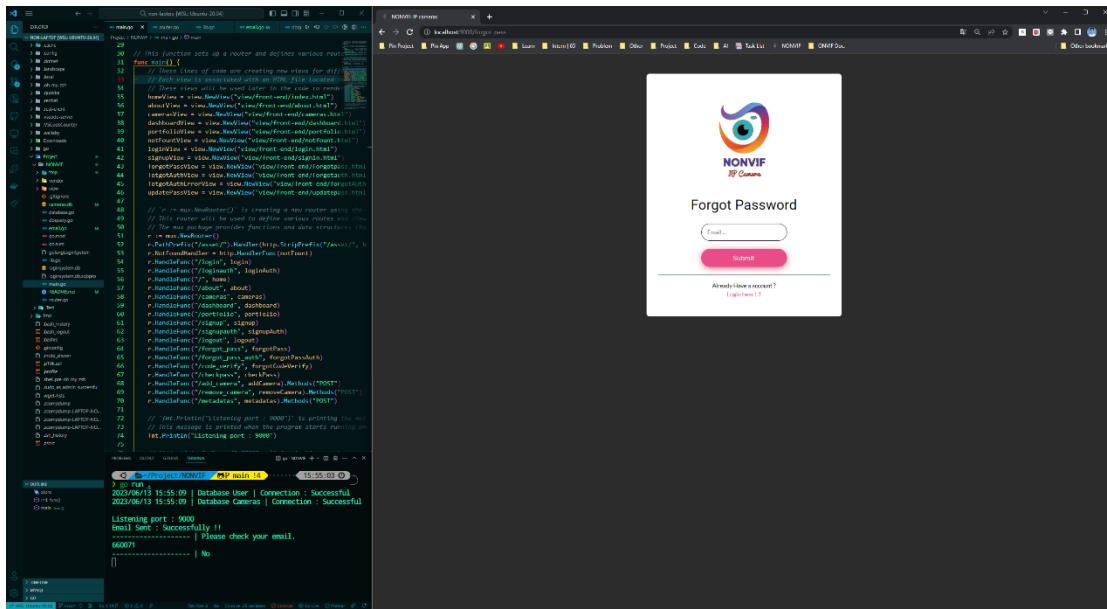
ภาพที่ 39 ภาพในหน้า Gmail ที่แสดงการส่งรหัสในการเปลี่ยน Password



ภาพที่ 40 ภาพการใส่รหัส Verification Code ที่ถูกต้อง



ภาพที่ 41 ภาพในหน้าการสร้าง Password ใหม่



ภาพที่ 42 ภาพการใส่รหัส Verification Code ผิด

จากการทดสอบการ Forgot Password เพื่อการเปลี่ยนรหัสผ่าน สำหรับผู้ที่อาจจะลืมรหัสในการลงทะเบียนในการเข้าสู่ระบบ จากภาพที่ 36 เป็นภาพที่แสดงหน้า Forgot Password สำหรับการกรอก Email ที่ได้ทำการลงทะเบียนไว้ เพื่อส่งรหัส Verification Code ไปยัง Email นั้นหากทำการกรอก Email ผิดจะขึ้นแสดงดังภาพที่ 37 แต่หากผู้ใช้งานกรอก Email ถูกต้องจะขึ้นแสดงดังภาพที่ 38 ผู้ใช้งานสามารถไปคุยกับล่องจดหมายในหน้าที่อยู่ของ Gmail ที่ผู้ใช้ได้ลงทะเบียนไว้ ผู้ใช้จะเห็นจดหมายใหม่ที่เป็น รหัส Verification Code ที่ส่งมาดังภาพที่ 39 ให้ผู้ใช้งานนำรหัส Verification Code ที่ได้รับ นำไปกรอกดังภาพที่ 40 หากกรอกรหัส Verification Code ได้ถูกต้องจะเข้าสู่หน้าสร้างรหัส Password ใหม่ดังภาพที่ 41 แต่หากกรอกรหัส Verification Code ผิดพลาดจะถูกพากลับมาหน้าที่ต้องกรอก Email ใหม่เพื่อส่งรหัส Verification Code ใหม่อีกครั้ง ดังภาพที่ 42

#### **4.1.2.4 ทดสอบเกี่ยวกับการ Logout**

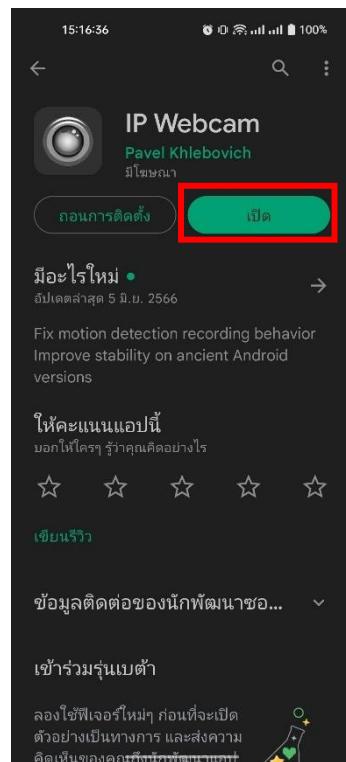
หากผู้ใช้งานสามารถลงทะเบียนเข้าสู่ระบบได้เป็นที่เรียบร้อยแล้วดังภาพที่ 33 เมื่อผู้ใช้งานต้องการออกจากระบบให้ผู้ใช้งานสังเกตมุมบนขวาเมื่อ จะมีคำว่า Logout แสดงอยู่ในแถบนำทางเมื่อกดแล้วผู้ใช้งานจะถูกพาออกจากระบบแล้วอยู่ที่หน้า Login ดังภาพที่ 31 แต่หากผู้ใช้งานไม่ได้วิธีออกจากระบบด้วยวิธีดังกล่าว แต่เลือกเป็นการปิดแท็บที่หน้า Web Browser การเข้าสู่ระบบของผู้ใช้งานจะยังคงบันทึกไว้อยู่ในระบบเนื่องจากระบบที่พัฒนานั้นได้มีการจัดเก็บคุกกี้ของผู้ใช้งานเอาไว้ คุกกี้คือไฟล์ที่เว็บไซต์ต่างๆ ที่ผู้เข้าชมสร้างขึ้น การบันทึกข้อมูลเกี่ยวกับการเข้าชมจะช่วยให้ทำสิ่งต่างๆ ทางออนไลน์ได้ง่ายขึ้น ตัวอย่างเช่น เว็บไซต์จะให้คุณลงชื่อเข้าใช้อยู่เสมอหากผู้ใช้เลือกเปิดแท็บกลับมาอีกครั้งผู้ใช้งานไม่จำเป็นที่จะต้องกรอกรหัสผ่านเพื่อทำการ Login เข้าสู่ใหม่ระบบอีกครั้ง

#### **4.1.3 ทดสอบเกี่ยวกับการเรียกสตรีมภาพวิดีโอจากกล้องไอพี**

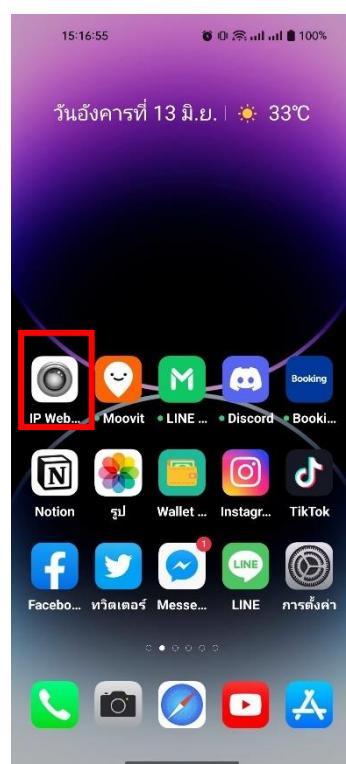
ในส่วนนี้จะเป็นการทดสอบเกี่ยวกับระบบของการติดตั้ง IP Webcam เพื่อใช้จำลองกล้องไอพี, การเพิ่มกล้องไอพี, การเลือกกล้องเพื่อชมภาพวิดีโอจากกล้องไอพี และการลบกล้องไอพี ว่าการทำงานในแต่ละส่วนนั้นได้ผลลัพธ์อย่างไร

##### **4.1.3.1 การติดตั้ง IP Webcam เพื่อใช้จำลองกล้องไอพี**

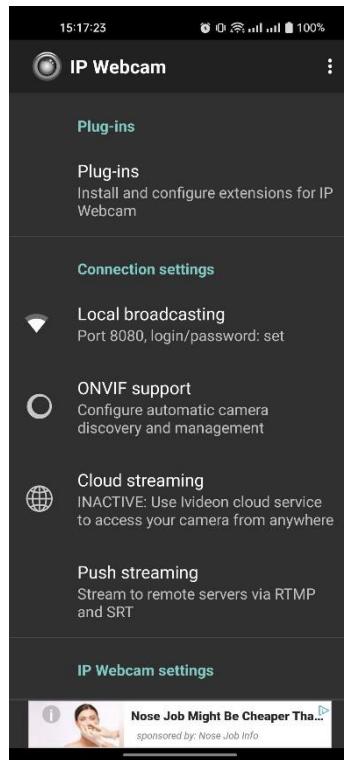
ในการทดสอบของโครงงานนี้นั้น จะใช้เป็นกล้องไอพีจำลอง จากแอปพลิเคชันบนสมาร์ทโฟนที่ชื่อว่า IP Webcam เพื่อใช้ในการจำลองกล้องบนสมาร์ทโฟนให้เป็นกล้องไอพี เมื่อกล้องวงจรปิด ในหัวข้อนี้จึงจะเป็นการติดตั้ง ตั้งค่าและเปิดใช้งาน



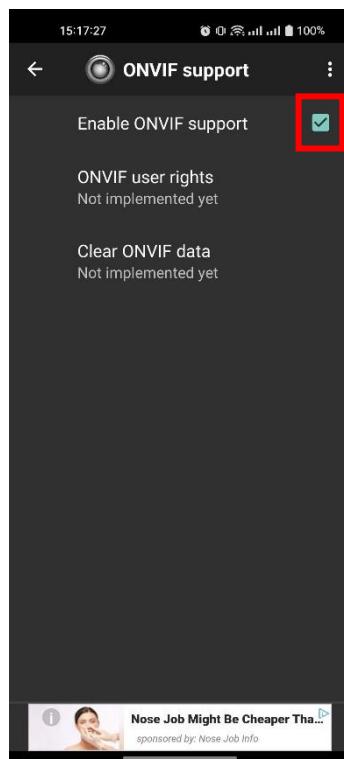
ภาพที่ 43 แอปพลิเคชัน IP Webcam บน Google Play



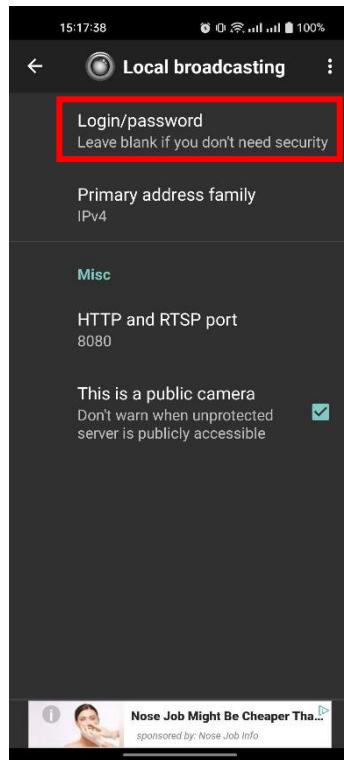
ภาพที่ 44 ภาพแสดงแอปพลิเคชัน IP Webcam ที่ติดตั้งเสร็จแล้วที่หน้าจอสมาร์ทโฟน



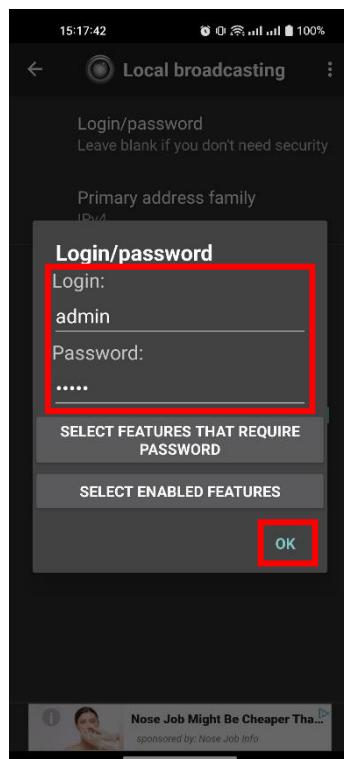
ภาพที่ 45 ภาพเมื่อเข้าสู่แอปพลิเคชัน IP Webcam



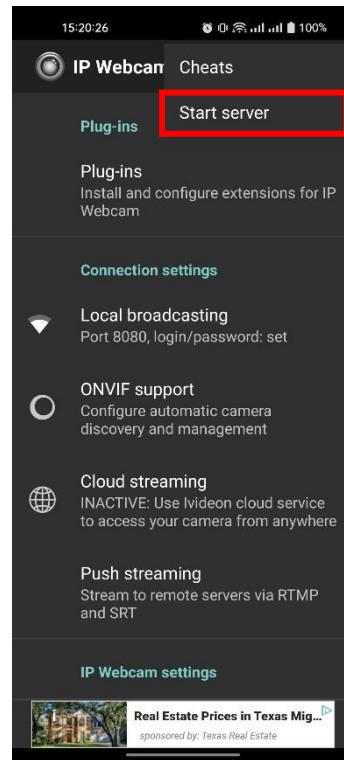
ภาพที่ 46 ภาพการเปิดใช้งานการสนับสนุน ONVIF



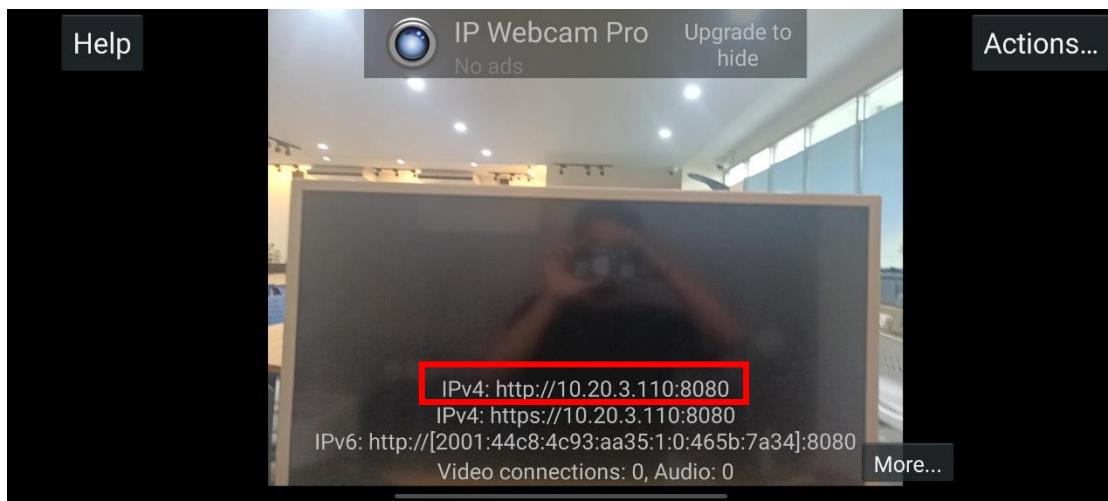
ภาพที่ 47 ภาพการตั้งค่ารหัสผ่านของกล้องไอพีจำลองบน IP Webcam



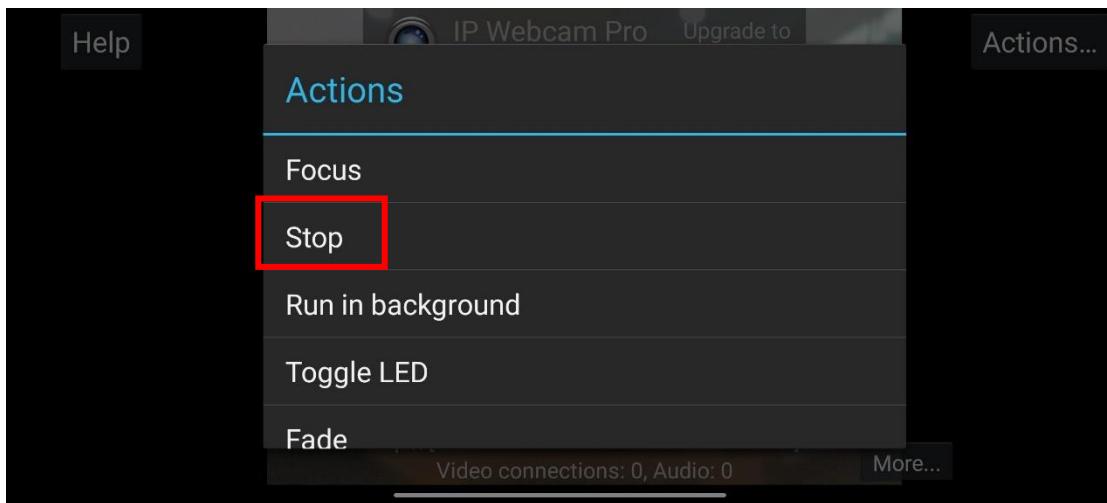
ภาพที่ 48 ภาพตัวอย่างการใส่รหัสผ่านของกล้องไอพีจำลองบน IP Webcam



ภาพที่ 49 การเปิดใช้งานการจำลองกล้องไอพีด้วย IP Webcam



ภาพที่ 50 ภาพเมื่อแสดงบนสมาร์ทโฟนเมื่อการจำลองกล้องกำลังทำงาน

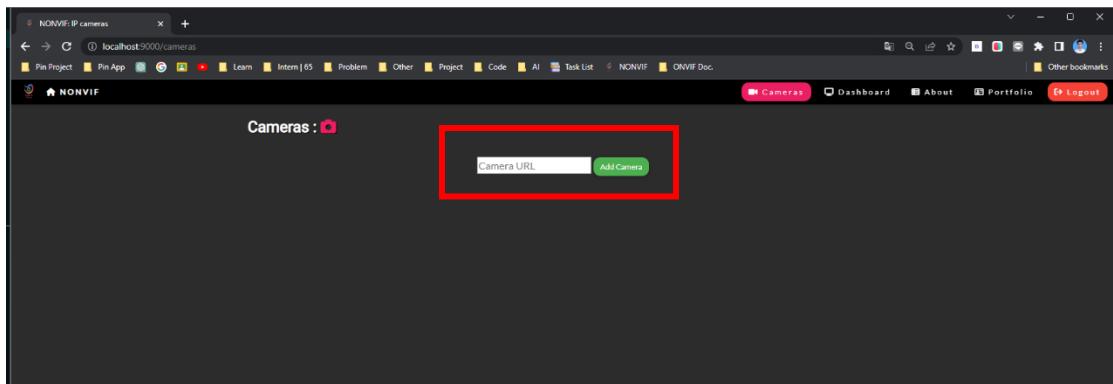


ภาพที่ 51 ภาพการหยุดการจำลองกล้องไอโอพี

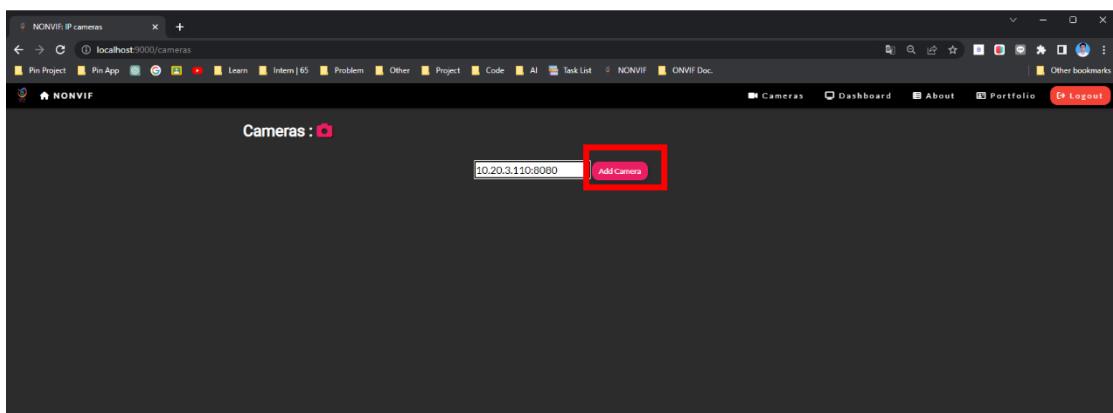
การติดตั้งแอปพลิเคชันจำลองกล้องไอโอพีด้วยสมาร์ทโฟน จะเริ่มจากการเข้าไปที่ Google Play Store บนสมาร์ทโฟน จากนั้นพิมพ์หา IP Webcam ที่แนบการค้นหาจะได้ดังภาพที่ 43 จากนั้นให้ทำการติดตั้ง เมื่อติดตั้งสำเร็จเรียบร้อยจะพบแอปพลิเคชันปรากฏที่หน้าจอスマาร์ทโฟนดังภาพที่ 44 จากนั้นให้ทำการกดเข้าใช้งาน IP Webcam จะพบหน้าต่างภายในแอปพลิเคชันดังภาพที่ 45 ให้ทำการเลือกหัวข้อ ONIVF support และทำการเปิดใช้งานการสนับสนุน ONVIF ดังภาพที่ 46 จากนั้นให้กดย้อนกลับที่ปุ่มลูกศรムุมบนซ้ายมือและเลือกหัวข้อ Local broadcasting เพื่อทำการตั้งค่ารหัสผ่านของกล้องไอโอพีจำลองนี้ดังภาพที่ 47 จากนั้นให้เลือกที่ Login/password และทำการกรอกข้อมูลให้เรียบร้อย ตัวอย่างเช่นให้ทดลองพิมพ์ Login : admin , Password : admin เป็นต้น ดังภาพที่ 48 แล้วกด ok เมื่อเสร็จสิ้นการตั้งค่าต่าง ๆ แล้วจะเป็นการเปิดใช้งานการจำลองกล้องไอโอพี โดยให้คลิกที่จุด 3 จุดมุมบนขวาเมื่อและเลือก Start server ดังภาพที่ 49 เมื่อกล้องทำงานสำเร็จจะเห็นข้อมูลไอโอพีแสดงดังภาพที่ 50 จากตัวอย่างจะแสดงเป็น 10.20.3.110:8080 ให้ทำการจดบันทึกเอาไว้ เพื่อนำไปทดสอบให้หัวข้อต่อไป หากผู้ใช้อยากยกเลิกการจำลองกล้องไอโอพี ให้ผู้ใช้เลือกที่ Actions มุมบนขวาเมื่อและเลือก Stop ดังภาพที่ 51 จะเป็นการหยุดการจำลองกล้องไอโอพีจากสมาร์ทโฟน

#### 4.1.3.2 การเพิ่มกล้องไอพี

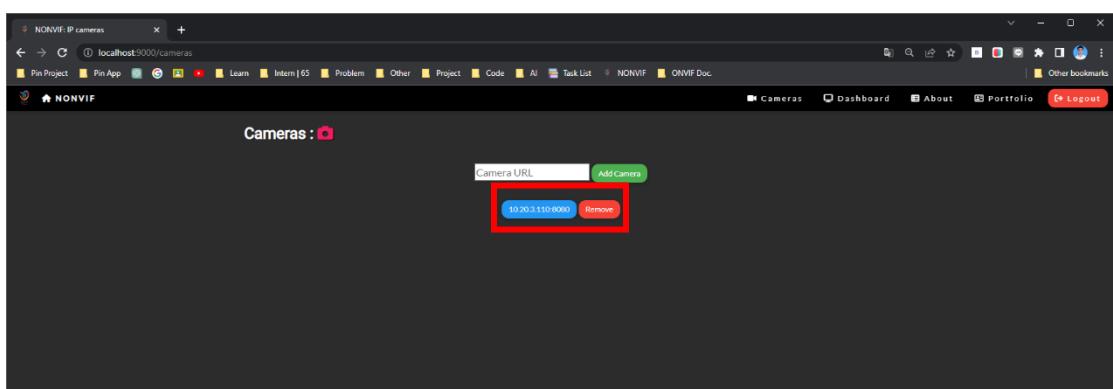
ในหัวข้อนี้จะเป็นการเพิ่มกล้องไอพีจากไอพีที่ได้รับมาจาก IP Webcam โดยจะทดสอบเพิ่มไอพีเป็นจำนวน 2 ไอพี เพื่อทดสอบการทำงานว่าสามารถเพิ่มกล้องได้หรือไม่



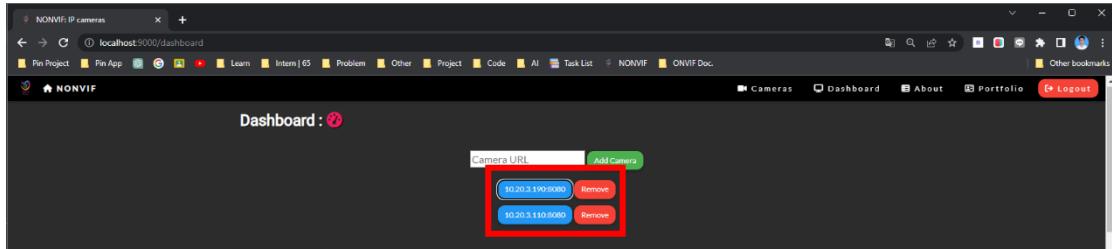
ภาพที่ 52 ภาพหน้า Cameras



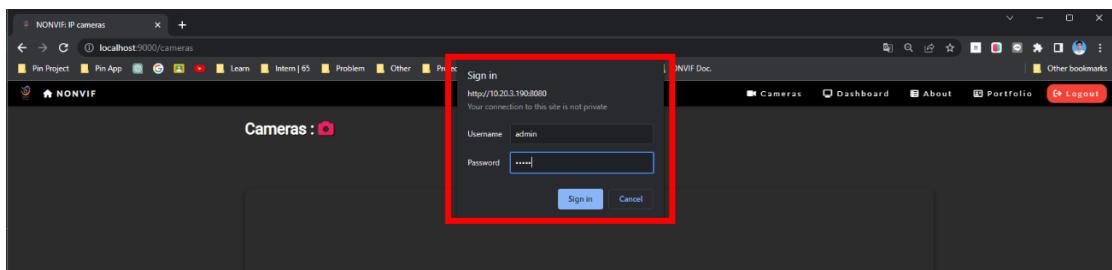
ภาพที่ 53 การเพิ่มที่อยู่ไอพีของกล้องไอพี



ภาพที่ 54 ภาพการเพิ่มที่อยู่กล้องไอพีสำเร็จ



ภาพที่ 55 ภาพการเพิ่มที่อยู่ของกล้องไอพีมากกว่า 1 ไอพี

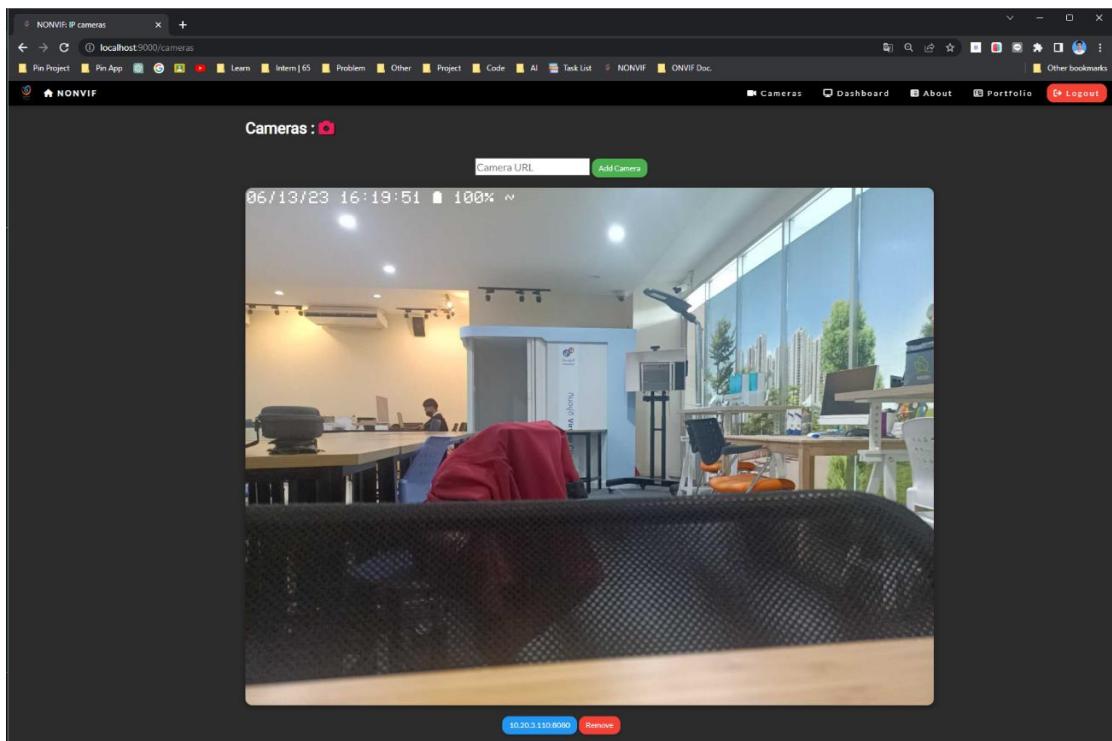


ภาพที่ 56 ภาพกล่องข้อความสำหรับใส่รหัสผ่านของกล้องไอพี

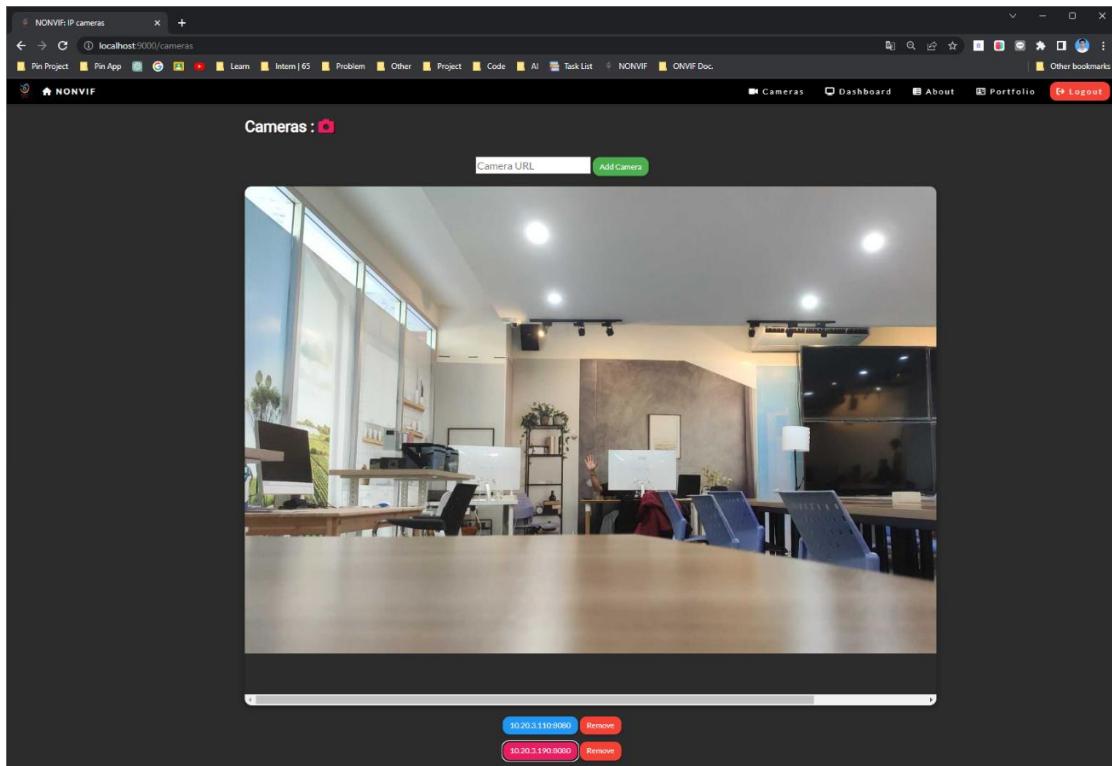
สำหรับการเพิ่มกล้องไอพีนั้นผู้ใช้งานจำเป็นจะต้อง Login เข้าสู่ระบบก่อนและทำการเลือกที่ปุ่ม Cameras ที่แนบตัวบน ผู้ใช้งานจะเห็นหน้าของ Cameras และดังภาพที่ 52 จากนั้นให้ผู้ใช้ นำเอาเลขที่อยู่ของกล้องไอพีที่ได้รับจากหัวข้อที่ 4.1.3.1 นำมากรอกลงในช่องกรอกข้อมูลดังภาพที่ 53 แล้วกดปุ่ม Add Camera เมื่อกรอกสำเร็จจะเห็นกล่องสีฟ้าที่มีเลขไอพีแสดงขึ้นมาดังภาพที่ 54 หากผู้ใช้อยากเพิ่มกล้องไอพีเพิ่มสามารถเพิ่มกล้องไอพีได้อีกตามวิธีก่อนหน้านี้ จะเห็นรายการกล้องไอพีที่เพิ่มเข้ามาแสดงดังภาพที่ 55 เป็นอันเสร็จการทดลองเพิ่มกล้องไอพี แต่หากการเพิ่มกล้องตัวไหนจำเป็นที่จะต้องใส่ Username/Password จะขึ้นกล่องข้อความแสดงดังภาพที่ 56 ให้ผู้ใช้กรอกข้อมูลตามที่ได้ทำการตั้งค่าไว้ดังภาพที่ 56 ได้เลย

#### 4.1.3.3 การเลือกกล้องเพื่อชมภาพวิดีโอด้วยกล้องไอพี

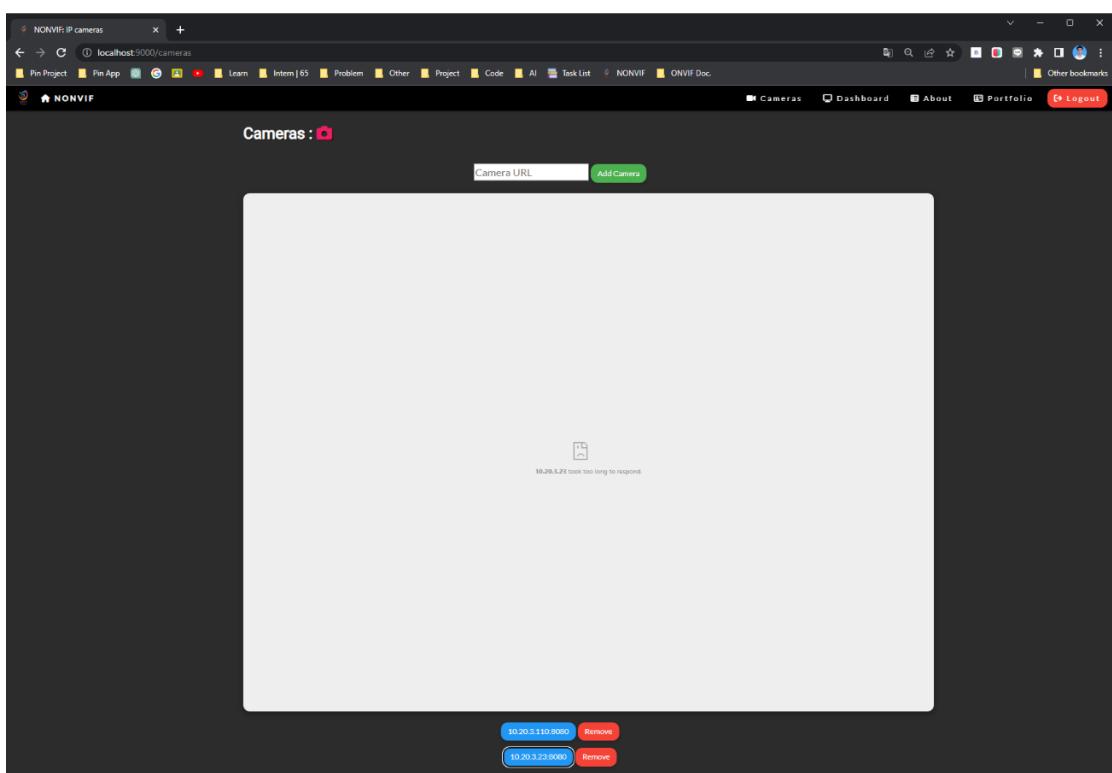
การทดสอบนี้จะเป็นการทดสอบการเลือกชมภาพจากกล้องไอพีที่ได้ทำการเพิ่มเข้ามาภายในระบบ ว่าสามารถทำการเลือกชมภาพแต่ละกล้องไอพีตามที่ได้มีการคลิกเพื่อเลือกชมได้หรือไม่ โดยที่เมื่อผู้ใช้งานสามารถเพิ่มที่อยู่กล้องไอพีสำเร็จตั้งภาพที่ 54 แล้วนั้น หากผู้ใช้งานทำการคลิกที่ปุ่มสีฟ้าที่มีเลขไอพีของกล้องกำกับอยู่จะเป็นการเลือกชมภาพ การสตรีมวิดีโอด้วยกล้องไอพีตัวนั้น ๆ ดังภาพที่ 57 และภาพที่ 58 แต่หากผู้ใช้เพิ่มที่อยู่ของกล้องไอพีผิดพลาดหรือไม่ถูกต้องภาพการสตรีมวิดีโอด้วยกล้องจะไม่ถูกแสดง ดังภาพที่ 59



ภาพที่ 57 ภาพการคลิกเพื่อรับชมกล้องตัวที่ 1



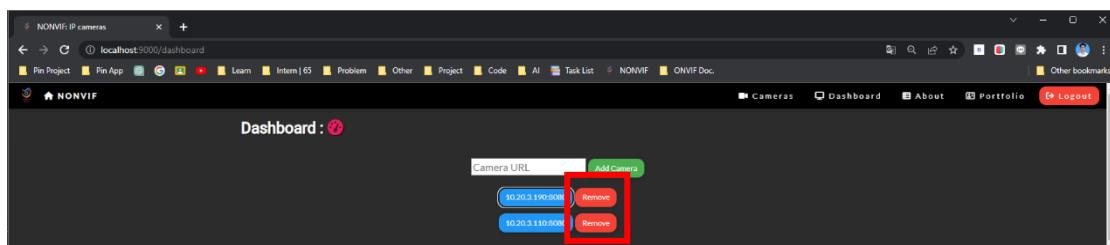
ภาพที่ 58 ภาพการคลิกเพื่อรับชมกล้องตัวที่ 2



ภาพที่ 59 ภาพการเรียกดูภาพจากกล้องไอพีไม่สำเร็จ

#### 4.1.3.4 การลบกล้องไอพี

การทดสอบนี้จะเป็นการทดสอบการลบที่อยู่ของกล้องไอพีออกจากระบบ โดยทางผู้ใช้งานสามารถกดปุ่มสีแดง ที่ชื่อว่า Remove ที่แสดงอยู่หลังปุ่มสีฟ้าที่เป็นปุ่มที่แสดงที่อยู่ของกล้องไอพีได้เลย เมื่อทำการคลิกแล้วตัวที่อยู่ของกล้องไอพินั้น ๆ ก็จะถูกลบออกจากระบบ

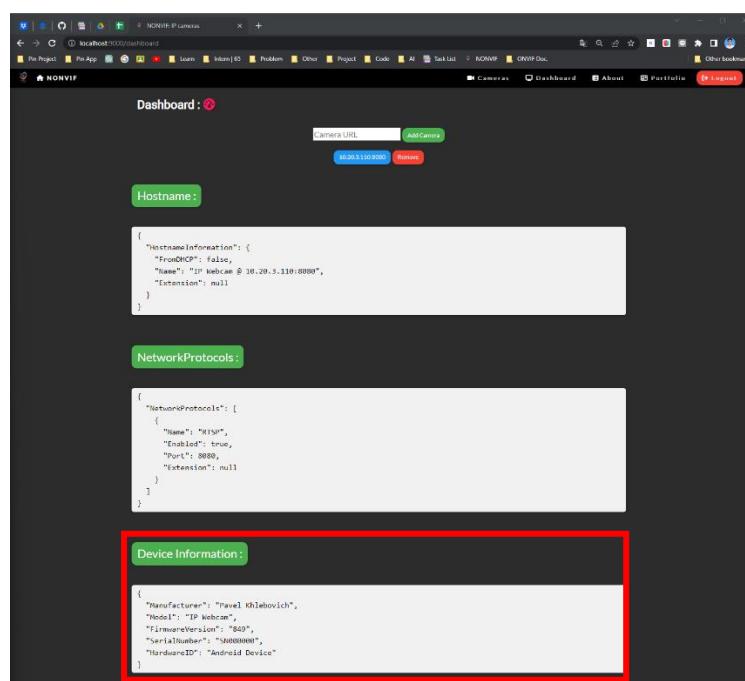


ภาพที่ 60 ภาพแสดงปุ่ม Remove

#### 4.1.4 ทดสอบเกี่ยวกับการเรียกข้อมูลเมตาจากกล้องไอพี

ในส่วนนี้จะเป็นการทดสอบเกี่ยวกับระบบของการแสดงข้อมูลเมตาที่ใช้โปรโตคอล ONVIF เรียกดูข้อมูลจากกล้องไอพี ว่าการทำงานในแต่ละส่วนนั้นได้ผลลัพธ์อย่างไร

##### 4.1.4.1 การแสดงข้อมูลจากฟังก์ชัน GetDeviceInformation



ภาพที่ 61 ภาพแสดงข้อมูลจากฟังก์ชัน GetDeviceInformation

#### 4.1.4.2 การแสดงข้อมูลจากฟังก์ชัน GetUser

```
{
  "User": [
    {
      "Username": "admin",
      "Password": "admin",
      "UserLevel": "Administrator",
      "Extension": null
    }
  ]
}
```

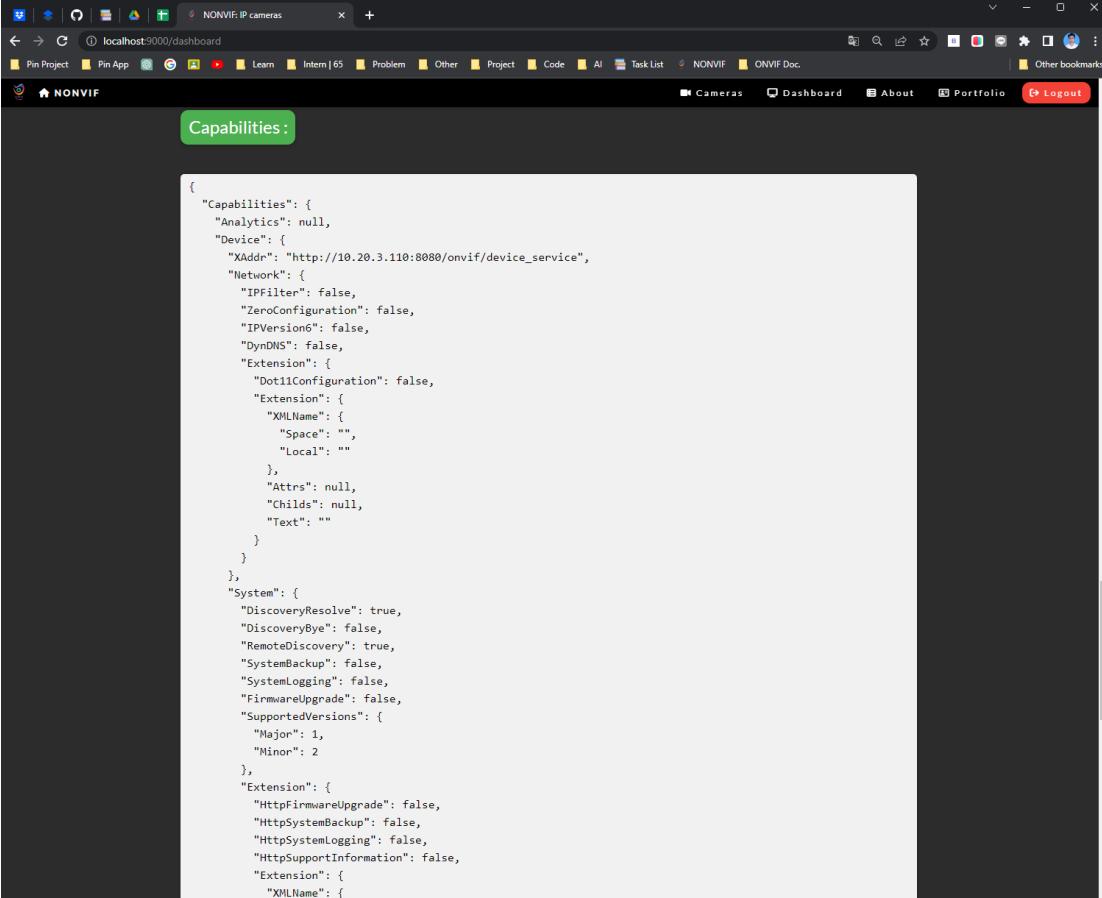
ภาพที่ 62 ภาพแสดงข้อมูลจากฟังก์ชัน GetUser

#### 4.1.4.3 การแสดงข้อมูลจากฟังก์ชัน GetSystemDateAndTime

```
{
  "SystemDateAndTime": {
    "DatetimeType": "Manual",
    "DaylightSavings": false,
    "TimeZone": {
      "TZ": ""
    },
    "UTCDateTime": {
      "Time": {
        "Hour": 5,
        "Minute": 53,
        "Second": 52
      },
      "Date": {
        "Year": 2023,
        "Month": 5,
        "Day": 15
      }
    },
    "LocalDateTime": {
      "Time": {
        "Hour": 5,
        "Minute": 53,
        "Second": 52
      },
      "Date": {
        "Year": 2023,
        "Month": 5,
        "Day": 15
      }
    },
    "Extension": {
      "XMLName": {
        "Space": "",
        "Local": ""
      },
      "Attrs": null,
      "Childs": null,
      "Text": ""
    }
  }
}
```

ภาพที่ 63 ภาพแสดงข้อมูลจากฟังก์ชัน GetSystemDateAndTime

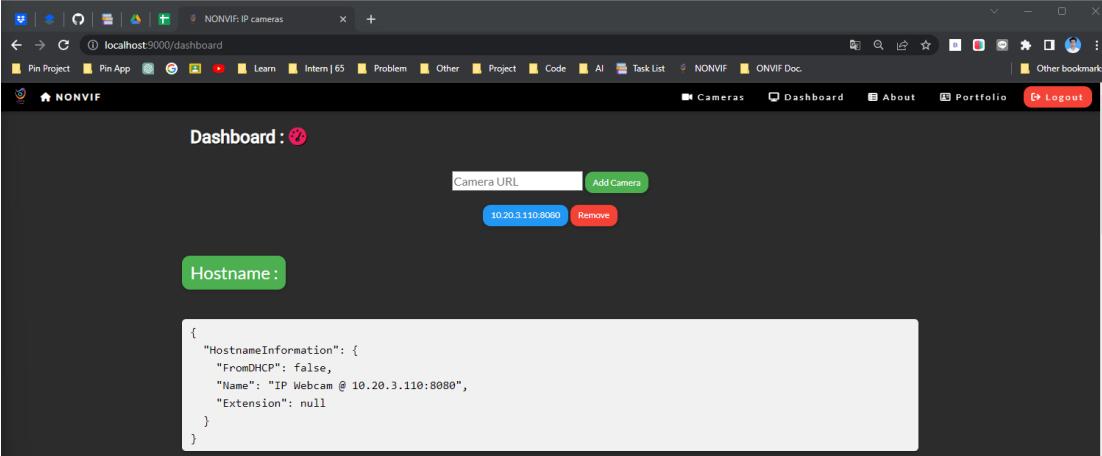
#### 4.1.4.4 การแสดงข้อมูลจากฟังก์ชัน GetCapabilities



```
{
  "Capabilities": {
    "Analytics": null,
    "Device": {
      "XAddr": "http://10.20.3.110:8080/onvif/device_service",
      "Network": {
        "IPFilter": false,
        "ZeroConfiguration": false,
        "IPVersion6": false,
        "DynDNS": false,
        "Extension": {
          "Dot11Configuration": false,
          "Extension": {
            "XMLName": {
              "Space": "",
              "Local": ""
            },
            "Attrs": null,
            "Childs": null,
            "Text": ""
          }
        }
      },
      "System": {
        "DiscoveryResolve": true,
        "DiscoveryBye": false,
        "RemoteDiscovery": true,
        "SystemBackup": false,
        "SystemLogging": false,
        "FirmwareUpgrade": false,
        "SupportedVersions": {
          "Major": 1,
          "Minor": 2
        },
        "Extension": {
          "HttpFirmwareUpgrade": false,
          "HttpSystemBackup": false,
          "HttpSystemLogging": false,
          "HttpSupportInformation": false,
          "Extension": {
            "XMLName": {
              "Space": "",
              "Local": ""
            }
          }
        }
      }
    }
  }
}
```

ภาพที่ 64 ภาพแสดงข้อมูลจากฟังก์ชัน GetCapabilities

#### 4.1.4.5 การแสดงข้อมูลจากฟังก์ชัน GetHostname



```

Dashboard : 🌐

Camera URL  10.20.3.110:8080 Remove

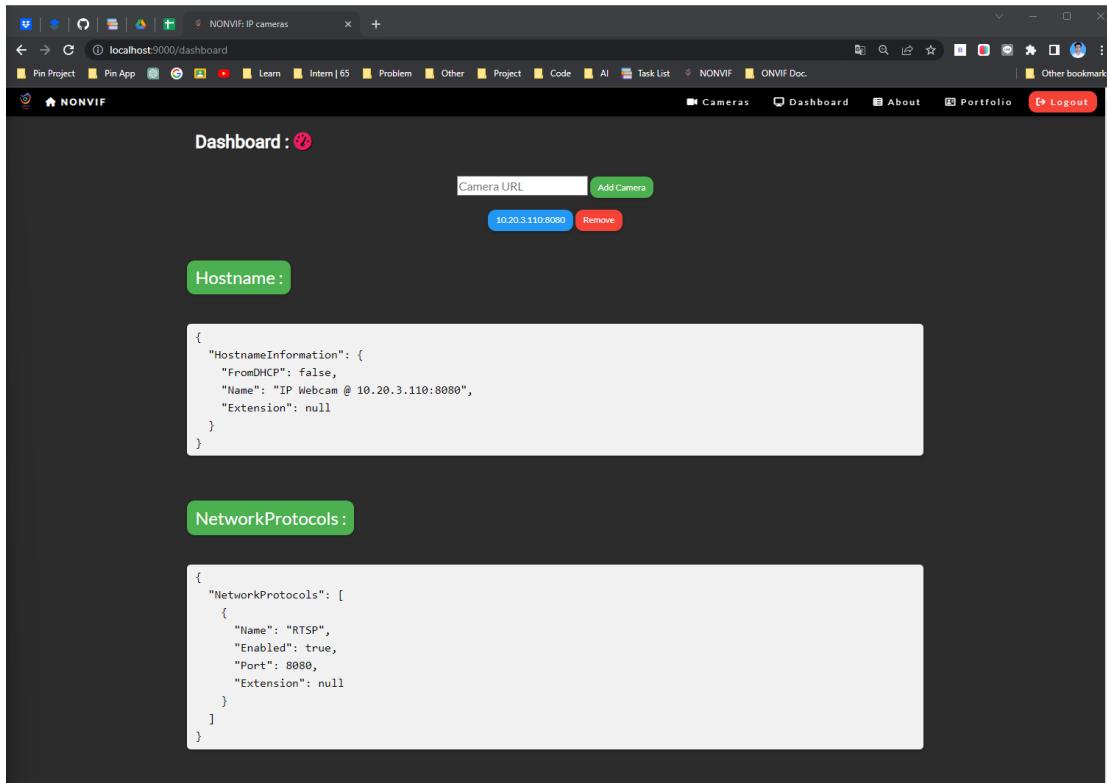
Hostname: 🌐

{
  "HostnameInformation": {
    "FromDHCP": false,
    "Name": "IP Webcam @ 10.20.3.110:8080",
    "Extension": null
  }
}

```

ภาพที่ 65 ภาพแสดงข้อมูลจากฟังก์ชัน GetHostname

#### 4.1.4.6 การแสดงข้อมูลจากฟังก์ชัน GetNetworkProtocols



The screenshot shows a browser window with the URL `localhost:9000/dashboard`. At the top, there's a navigation bar with various project-related links like Pin Project, Pin App, Learn, Intern | 65, Problem, Other, Project, Code, AI, Task List, NONVIF, and ONVIF Doc. Below the navigation bar is a header with the NONVIF logo and a dropdown menu. The main content area has a dark background with several green rounded rectangular boxes containing JSON responses. The first box is labeled "Hostname:" and contains the following JSON:

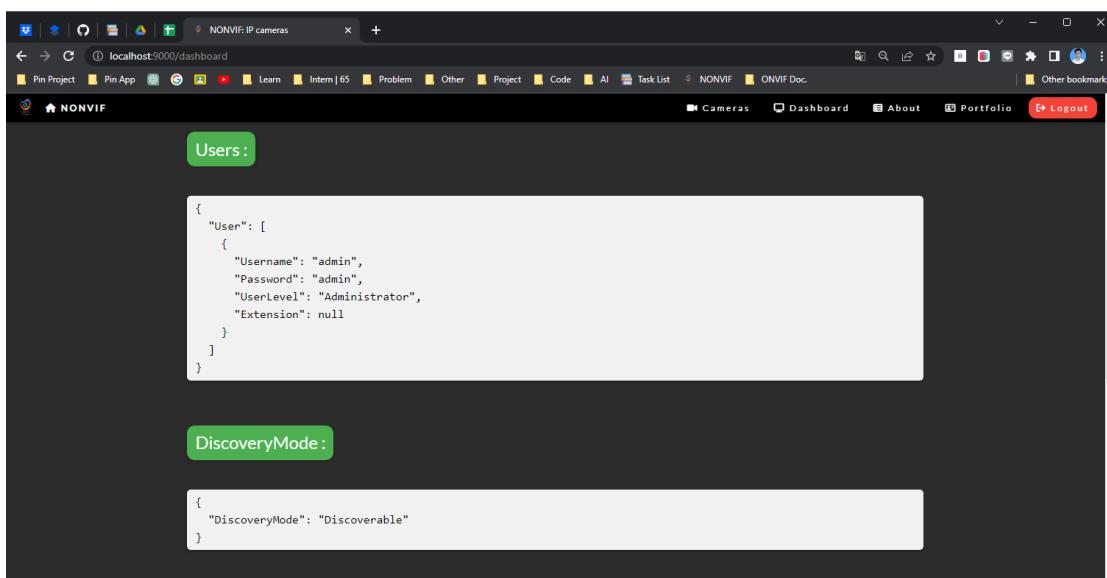
```
{
  "HostnameInformation": {
    "FromDHCP": false,
    "Name": "IP Webcam @ 10.20.3.110:8080",
    "Extension": null
  }
}
```

The second box is labeled "NetworkProtocols:" and contains the following JSON:

```
{
  "NetworkProtocols": [
    {
      "Name": "RTSP",
      "Enabled": true,
      "Port": 8080,
      "Extension": null
    }
  ]
}
```

ภาพที่ 66 ภาพแสดงข้อมูลจากฟังก์ชัน GetNetworkProtocols

#### 4.1.4.7 การแสดงข้อมูลจากฟังก์ชัน GetDiscoveryMode



The screenshot shows a browser window with the URL `localhost:9000/dashboard`. The interface is identical to the previous screenshot, featuring the same navigation bar and header. The main content area displays two green rounded rectangular boxes with JSON responses. The first box is labeled "Users:" and contains the following JSON:

```
{
  "User": [
    {
      "Username": "admin",
      "Password": "admin",
      "UserLevel": "Administrator",
      "Extension": null
    }
  ]
}
```

The second box is labeled "DiscoveryMode:" and contains the following JSON:

```
{
  "DiscoveryMode": "Discoverable"
}
```

ภาพที่ 67 ภาพแสดงข้อมูลจากฟังก์ชัน GetDiscoveryMode

#### 4.1.4.8 การแสดงข้อมูลจากฟังก์ชัน GetServiceCapabilities

```
{
  "Capabilities": {
    "Network": {
      "IPFilter": false,
      "ZeroConfiguration": false,
      "IPVersion6": false,
      "DynDNS": false,
      "Dot11Configuration": false,
      "Dot1XConfigurations": 0,
      "HostnameFromDHCP": false,
      "NTP": 0,
      "DHCPv6": false
    },
    "Security": {
      "TLS1_0": false,
      "TLS1_1": false,
      "TLS1_2": false,
      "OnboardKeyGeneration": false,
      "AccessPolicyConfig": false,
      "DefaultAccessPolicy": false,
      "Dot1X": false,
      "RemoteUserHandling": false,
      "X509Token": false,
      "SAMLToken": false,
      "KerberosToken": false,
      "UsernameToken": false,
      "HTTPDigest": false,
      "RELTOKEN": false,
      "SupportedEAPMethods": {
        "Types": null
      },
      "MaxUsers": 0,
      "MaxUserNameLength": 0,
      "MaxPasswordLength": 0
    },
    "System": {
      "DiscoveryResolve": true,
      "DiscoveryBye": false,
      "RemoteDiscovery": false,
      "SystemBackup": false,
      "SystemLogging": false,
      "FirmwareUpgrade": false,
      "HTTPFirmwareUpgrade": false,
      "HTTPSSystemBackup": false,
      "HTTPSSystemLogging": false,
      "HTTPSSupportInformation": false,
      "StorageConfiguration": false,
      "ManagementInterface": 0
    }
  }
}
```

ภาพที่ 68 ภาพแสดงข้อมูลจากฟังก์ชัน GetServiceCapabilities

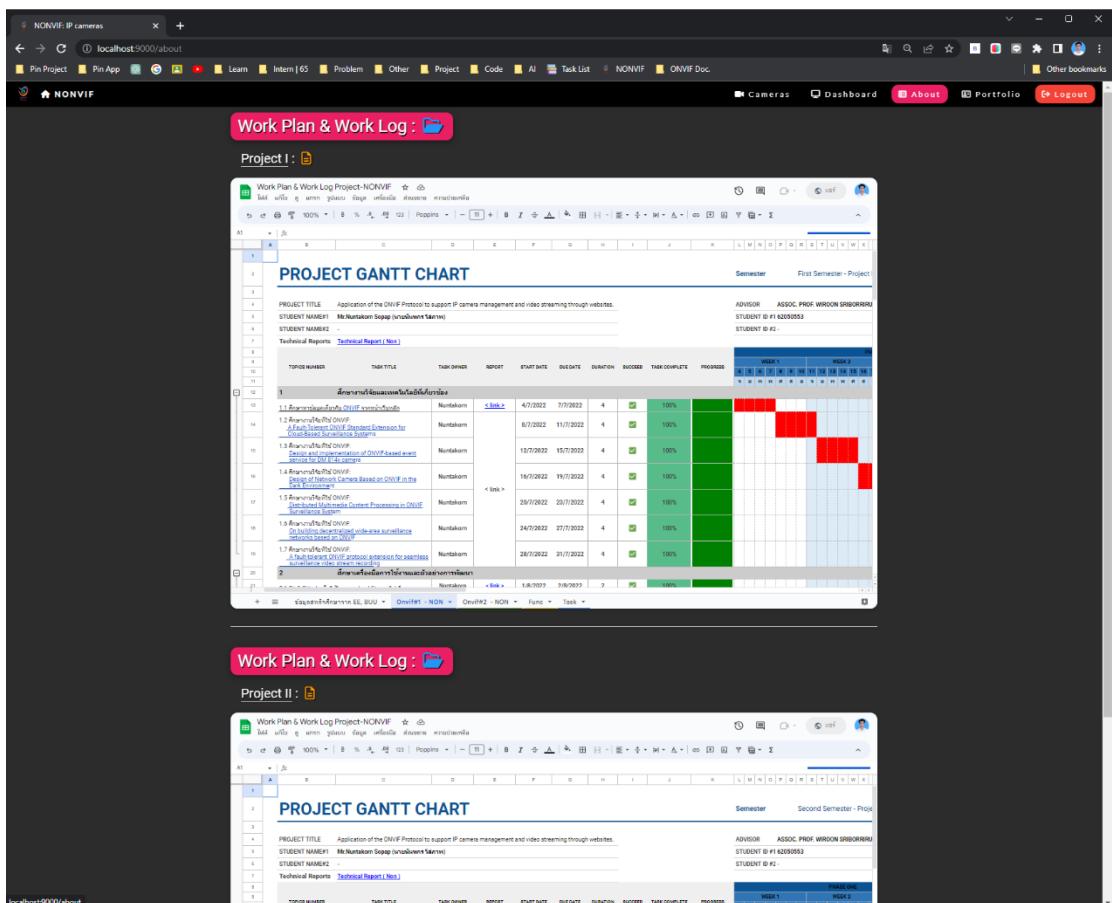
จากภาพที่ 61 ถึงภาพที่ 68 จะเป็นการแสดงข้อมูลเมตาต่าง ๆ ที่สามารถเรียกมาแสดงได้จากกล้องไอพีที่ได้จำลองผ่านแอปพลิเคชัน IP Webcam โดยจะมีการเรียกดูข้อมูลเมตาของกล้องไอพีผ่านโบटคอล ONVIF ซึ่งมีทั้งหมด 8 ฟังก์ชัน คือ 1) GetDeviceInformation, 2) GetUser, 3) GetSystemDateAndTime, 4) GetCapabilities, 5) GetHostname, 6) GetNetworkProtocols, 7) GetDiscoveryMode และ 8) GetServiceCapabilities โดยข้อมูลที่แสดงทั้งหมดที่หน้าเว็บไซต์ จะแสดงอยู่ในรูปแบบของ JSON และหากผู้ใช้งานอยากรู้ข้อมูลของกล้องไอพีตัวอื่น ๆ ให้ผู้ใช้งานทำการเพิ่มกล้องไอพีใหม่ดังหัวข้อที่ 4.1.3.2 หลังจากเพิ่มกล้องตัวใหม่แล้ว ผู้ใช้งานจะสามารถเลือกดูข้อมูลเมตาจากกล้องไอพีตัวอื่นได้เช่นกัน

หากแต่ร้าที่อยู่ของกล้องไอพีเหล่านั้นที่ถูกเพิ่มเข้ามา ไม่ได้มีการกำหนดค่าของ username และ password ของกล้องให้เป็น admin และ admin ตรงตามภายในโค้ดของโครงการนี้ที่ได้มีการกำหนดค่าไว้ หรือเป็นค่าอื่นนอกเหนือจากนี้หรือไม่มีค่า username และ password ผู้ใช้งานจะไม่สามารถเห็นข้อมูลเมตาที่เรียกจากกล้องไอพีได้ ดังนั้นผู้ใช้งานจำเป็นที่จะต้องกำหนดค่าของ username และ password ให้เป็น admin และ admin เพียงเท่านั้น

#### 4.1.5 ทดสอบส่วนอื่น ๆ

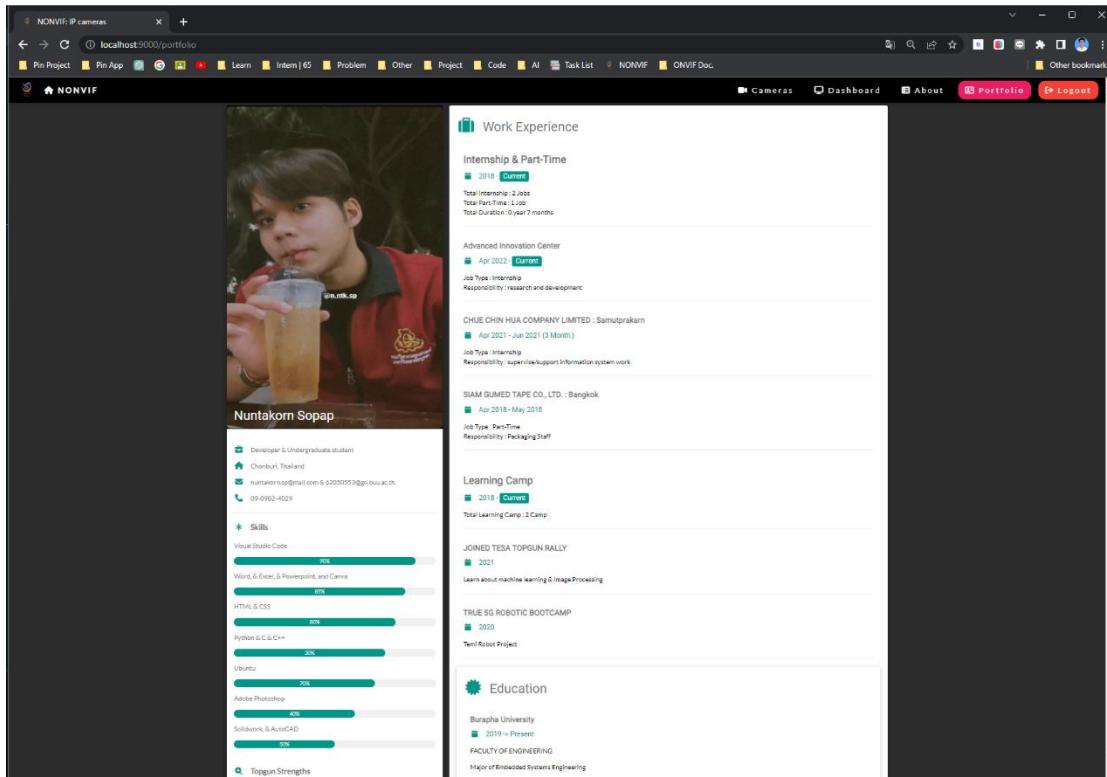
ในส่วนนี้จะเป็นการทดสอบเกี่ยวกับระบบของการแสดงหน้าอื่น ๆ เช่น หน้า About, หน้า Portfolio และหน้า 404 Not Found ว่าการทำงานในแต่ละส่วนนั้นได้ผลลัพธ์อย่างไร

##### 4.1.5.1 การแสดงหน้า About



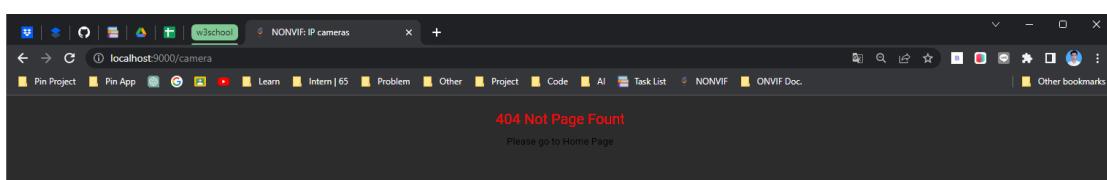
ภาพที่ 69 ภาพการแสดงของหน้า About

#### 4.1.5.2 การแสดงหน้า Portfolio



ภาพที่ 70 ภาพการแสดงของหน้า Portfolio

#### 4.1.5.3 การแสดงหน้า 404 Not Fount



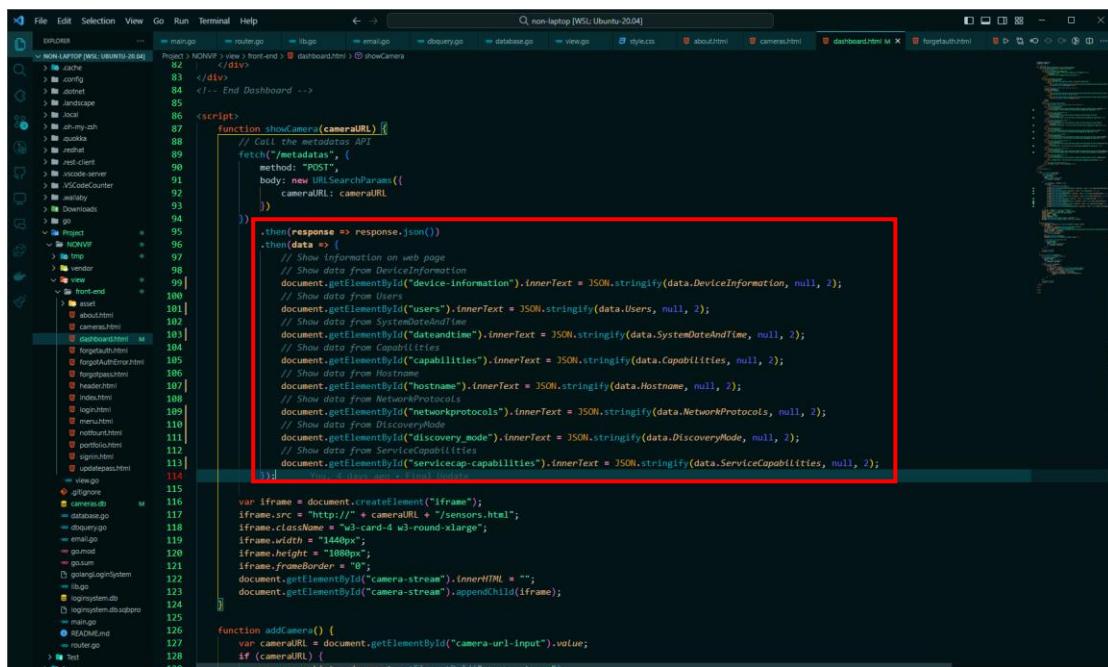
ภาพที่ 71 ภาพการแสดงของหน้า 404 Not Fount

การแสดงการทำงานของส่วนอื่น ๆ นี่จะมีส่วนของการแสดงหน้า About ที่จะแสดงข้อมูลของ Gantt Chart ของการทำงานดังภาพที่ 69 ส่วนของการแสดงหน้า Portfolio ที่แสดงข้อมูลของผู้เขียนโครงการนี้ดังภาพที่ 70 และส่วนของการแสดงหน้า 404 Not Fount ที่จะแสดงแสดงก็ต่อเมื่อมีผู้ใช้เรียก URL ไปแล้วแต่หน้าเว็บไม่สามารถแสดงได้ ตัวอย่างเช่น URL ที่ถูกต้องจะเป็น <http://localhost:9000/cameras> แต่ผู้ใช้เรียกผิดเป็น <http://localhost:9000/camera> ทำให้เกิดการแสดงหน้า 404 Not Fount ดังภาพที่ 70

## 4.2 การพัฒนาระบบ

จากหัวข้อที่ 4.1.4 ที่เป็นการทดสอบการแสดงผลของการเรียกข้อมูลเมตาจากกล้องไว์ด้วยโปรโตคอล ONVIF จะเห็นได้ว่าข้อมูลที่แสดงออกมานางานที่ 61 ถึงภาพที่ 68 จะเป็นการแสดงข้อมูลออกมากทั้งหมดผ่านการเรียกดูข้อมูลของฟังก์ชันนั้น ๆ โดยแสดงอยู่ในรูปแบบของ JSON ทำให้การอ่านค่าข้อมูลเมตาที่ได้รับมาสำหรับผู้ใช้งานทั่วไปที่ไม่ได้มีความรู้เรื่องการแสดงค่าข้อมูลแบบ JSON นั้นอ่านข้อมูลที่แสดงออกมาได้ยาก

ทางผู้พัฒนาได้คำนึงถึงปัญหาในการอ่านค่าข้อมูลเมตาที่ได้รับของผู้ใช้ จึงได้ทำการพัฒนาแก้ไขโค้ดในการเรียกข้อมูลเมตาจากกล้องแบบเก่าดังภาพที่ 72 และภาพที่ 73 ให้เป็นโค้ดแบบใหม่ดังภาพที่ 74 และภาพที่ 75 ซึ่งแก้ไขให้มีการเข้าถึงข้อมูลย่อยที่ได้รับมาจาก การเรียกใช้งานในแต่ละฟังก์ชันของ ONVIF และนำมาเก็บไว้ในค่า id ของแต่ข้อมูลย่อยแล้วจึงนำ id ที่เก็บข้อมูลไว้นำไปเรียกแสดงออกดังภาพที่ 75 โดยทำกับทุกฟังก์ชันของ ONVIF เมื่อทำการแก้ไขครบทั้งหมดแล้วข้อมูลที่แสดงให้ผู้ใช้เห็นแบบใหม่จะเป็นดังภาพที่ 76 ถึงภาพที่ 78



```

function showCamera(cameraURL) {
    // Call the metadata API
    fetch('/metadata', {
        method: 'GET',
        body: new URLSearchParams({
            cameraURL: cameraURL
        })
    }).then(response => response.json())
    .then(data => {
        // Show information on web page
        // Show data from DeviceInformation
        document.getElementById("device-information").innerText = JSON.stringify(data.DeviceInformation, null, 2);
        // Show data from Users
        document.getElementById("users").innerText = JSON.stringify(data.Users, null, 2);
        // Show data from System
        document.getElementById("system").innerText = JSON.stringify(data.System, null, 2);
        // Show data from Date/Time
        document.getElementById("date-time").innerText = JSON.stringify(data.SystemDateAndTime, null, 2);
        // Show data from Capabilities
        document.getElementById("capabilities").innerText = JSON.stringify(data.Capabilities, null, 2);
        // Show data from Hostname
        document.getElementById("hostname").innerText = JSON.stringify(data.Hostname, null, 2);
        // Show data from NetworkProtocols
        document.getElementById("network-protocols").innerText = JSON.stringify(data.NetworkProtocols, null, 2);
        // Show data from DiscoveryMode
        document.getElementById("discovery-mode").innerText = JSON.stringify(data.DiscoveryMode, null, 2);
        // Show data from ServiceCapabilities
        document.getElementById("service-capabilities").innerText = JSON.stringify(data.ServiceCapabilities, null, 2);

        var iframe = document.createElement("iframe");
        iframe.src = "http://" + cameraURL + "/sensors.html";
        iframe.className = "w3-card-4 w3-round-xlarge";
        iframe.width = "1440px";
        iframe.height = "1080px";
        iframe.frameBorder = "0";
        document.getElementById("camera-stream").innerHTML = "";
        document.getElementById("camera-stream").appendChild(iframe);
    });
}

function addCamera() {
    var cameraURL = document.getElementById("camera-url-input").value;
    if (cameraURL) {
        showCamera(cameraURL);
    }
}

```

ภาพที่ 72 ภาพของการเขียนเรียกแสดงข้อมูลเมตาแบบเก่า ในส่วนของ JavaScript

```

<div class="w3-xlarge w3-text-white">
    <!-- Added variable ID for displaying data from /metadata -->
    <!-- Show data from Hostname -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">Hostname :</p>
        <pre id="hostname" class="w3-card-2 w3-round-large w3-padding w3-light-gray"></pre>
    </div>
    <!-- Show data from NetworkProtocols -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">NetworkProtocols :</p>
        <pre id="networkprotocols" class="w3-card-2 w3-round-large w3-padding w3-light-gray"></pre>
    </div>
    <!-- Show data from DeviceInformation -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">Device Information :</p>
        <pre id="device-information" class="w3-card-2 w3-round-large w3-padding w3-light-gray"></pre>
    </div>
    <!-- Show data from Users -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">Users :</p>
        <pre id="users" class="w3-card-2 w3-round-large w3-padding w3-light-gray"></pre>
    </div>
    <!-- Show data from DiscoveryMode -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">DiscoveryMode :</p>
        <pre id="discovery_mode" class="w3-card-2 w3-round-large w3-padding w3-light-gray"></pre>
    </div>
    <!-- Show data from SystemDateAndTime -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">System Date and Time :</p>
        <pre id="dateandtime" class="w3-card-2 w3-round-large w3-padding w3-light-gray"></pre>
    </div>
    <!-- Show data from ServiceCapabilities -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">ServiceCapabilities :</p>
        <pre id="servicecap-capabilities" class="w3-card-2 w3-round-large w3-padding w3-light-gray"></pre>
    </div>
    <!-- Show data from Capabilities -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">Capabilities :</p>
        <pre id="capabilities" class="w3-card-2 w3-round-large w3-padding w3-light-gray"></pre>
    </div>

```

ภาพที่ 73 ภาพของการเขียนเรียกแสดงข้อมูลเมต้าแบบเก่า ในส่วนของ HTML

```

function showCamera(cameraURL) {
    // Call the metadata API
    fetch('/metadata', {
        method: "POST",
        body: new URLSearchParams({
            cameraURL: cameraURL
        })
    })
    .then(response => response.json())
    .then(data => {
        // Show information on web page
        // Show data from DeviceInformation
        document.getElementById("manufacturer").innerText = data.DeviceInformation.Manufacturer;
        document.getElementById("model").innerText = data.DeviceInformation.Model;
        document.getElementById("firmware-version").innerText = data.DeviceInformation.FirmwareVersion;
        document.getElementById("serial-number").innerText = data.DeviceInformation.SerialNumber;
        document.getElementById("hardware-id").innerText = data.DeviceInformation.HardwareID;
        // Show data from Users
        var user = data.Users.User[0];
        document.getElementById("username").innerText = user.Username;
        document.getElementById("password").innerText = user.Password;
        document.getElementById("userlevel").innerText = user.UserLevel;
        // Show data from SystemDateAndTime
        var sysdatetime = data.SystemDateAndTime.SystemDateAndTime;
        document.getElementById("DateTimeType").innerText = sysdatetime.DateTimeType;
        document.getElementById("DaylightSavings").innerText = sysdatetime.DaylightSavings;
        document.getElementById("UTCDateTime-Time-Hour").innerText = sysdatetime.UTCDateTime.Time.Hour;
        document.getElementById("UTCDateTime-Time-Minute").innerText = sysdatetime.UTCDateTime.Time.Minute;
        document.getElementById("UTCDateTime-Time-Second").innerText = sysdatetime.UTCDateTime.Time.Second;
        document.getElementById("UTCDateTime-Date-Day").innerText = sysdatetime.UTCDateTime.Date.Day;
        document.getElementById("UTCDateTime-Date-Month").innerText = sysdatetime.UTCDateTime.Date.Month + 1;
        document.getElementById("UTCDateTime-Year").innerText = sysdatetime.UTCDateTime.Date.Year;
        document.getElementById("LocalDateTime-Time-Hour").innerText = sysdatetime.LocalDateTime.Time.Hour + 7;
        document.getElementById("LocalDateTime-Time-Minute").innerText = sysdatetime.LocalDateTime.Time.Minute;
        document.getElementById("LocalDateTime-Time-Second").innerText = sysdatetime.LocalDateTime.Time.Second;
        document.getElementById("LocalDateTime-Date-Day").innerText = sysdatetime.LocalDateTime.Date.Day;
        document.getElementById("LocalDateTime-Date-Month").innerText = sysdatetime.LocalDateTime.Date.Month + 1;
        document.getElementById("LocalDateTime-Year").innerText = sysdatetime.LocalDateTime.Date.Year;
        // Show data from Capabilities
        document.getElementById("capabilities").innerText = JSON.stringify(data.Capabilities, null, 2);
        // Show data from Hostname
        document.getElementById("hostname-name").innerText = data.Hostname.HostnameInformation.Name;
        document.getElementById("hostname-fromDHCP").innerText = data.Hostname.HostnameInformation.FromDHCP;
        // Show data from NetworkProtocols
        var network = data.NetworkProtocols.NetworkProtocols[0];
        document.getElementById("NetworkProtocols").innerText = `You, 4 days ago * Final Update`;
    })
}

```

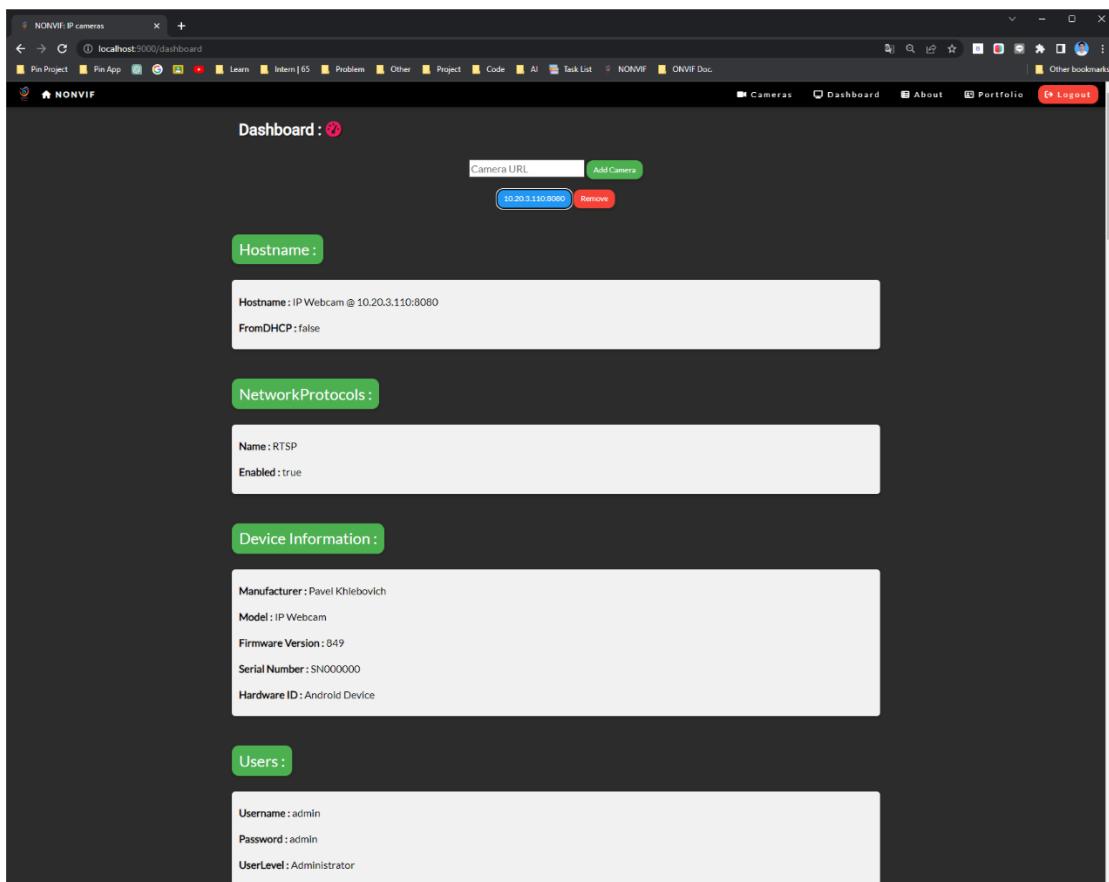
ภาพที่ 74 ภาพของการเขียนเรียกแสดงข้อมูลเมต้าแบบใหม่ ในส่วนของ JavaScript

```

<div class="w3-xlarge w3-text-white">
    <!-- Show data from Hostname -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">Hostname :</p>
        <div id="hostname" class="w3-card-2 w3-round-large w3-padding w3-light-gray">
            <p><strong>Hostname :</strong> <span id="hostname-name"></span></p>
            <p><strong>FromDHCP :</strong> <span id="hostname-fromDHCP"></span></p>
        </div>
    </div>
    <!-- Show data from NetworkProtocols -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">NetworkProtocols :</p>
        <div id="networkprotocols" class="w3-card-2 w3-round-large w3-padding w3-light-gray">
            <p><strong>Name :</strong> <span id="network-name"></span></p>
            <p><strong>Enabled :</strong> <span id="network-enabled"></span></p>
        </div>
    </div>
    <!-- Show data from DeviceInformation -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">Device Information :</p>
        <div id="device-information" class="w3-card-2 w3-round-large w3-padding w3-light-gray">
            <p><strong>Manufacturer :</strong> <span id="manufacturer"></span></p>
            <p><strong>Model :</strong> <span id="model"></span></p>
            <p><strong>Firmware Version :</strong> <span id="firmware-version"></span></p>
            <p><strong>Serial Number :</strong> <span id="serial-number"></span></p>
            <p><strong>Hardware ID :</strong> <span id="hardware-id"></span></p>
        </div>
    </div>
    <!-- Show data from Users -->
    <div class="w3-padding-16">
        <p class="w3-xxlarge w3-bar-item w3-button w3-green w3-round-xlarge w3-card-2">Users :</p>
        <div id="users" class="w3-card-2 w3-round-large w3-padding w3-light-gray">
            <p><strong>Username :</strong> <span id="username"></span></p>
            <p><strong>Password :</strong> <span id="password"></span></p>
            <p><strong>UserLevel :</strong> <span id="userlevel"></span></p>
        </div>
    </div>

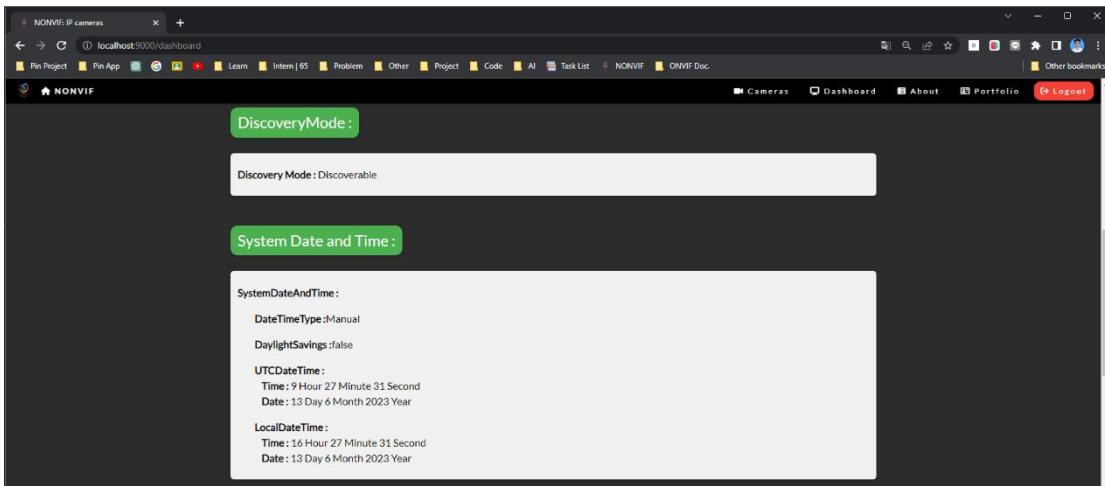
```

ภาพที่ 75 ภาพของการเขียนเรียกแสดงข้อมูลแบบใหม่ ในส่วนของ HTML



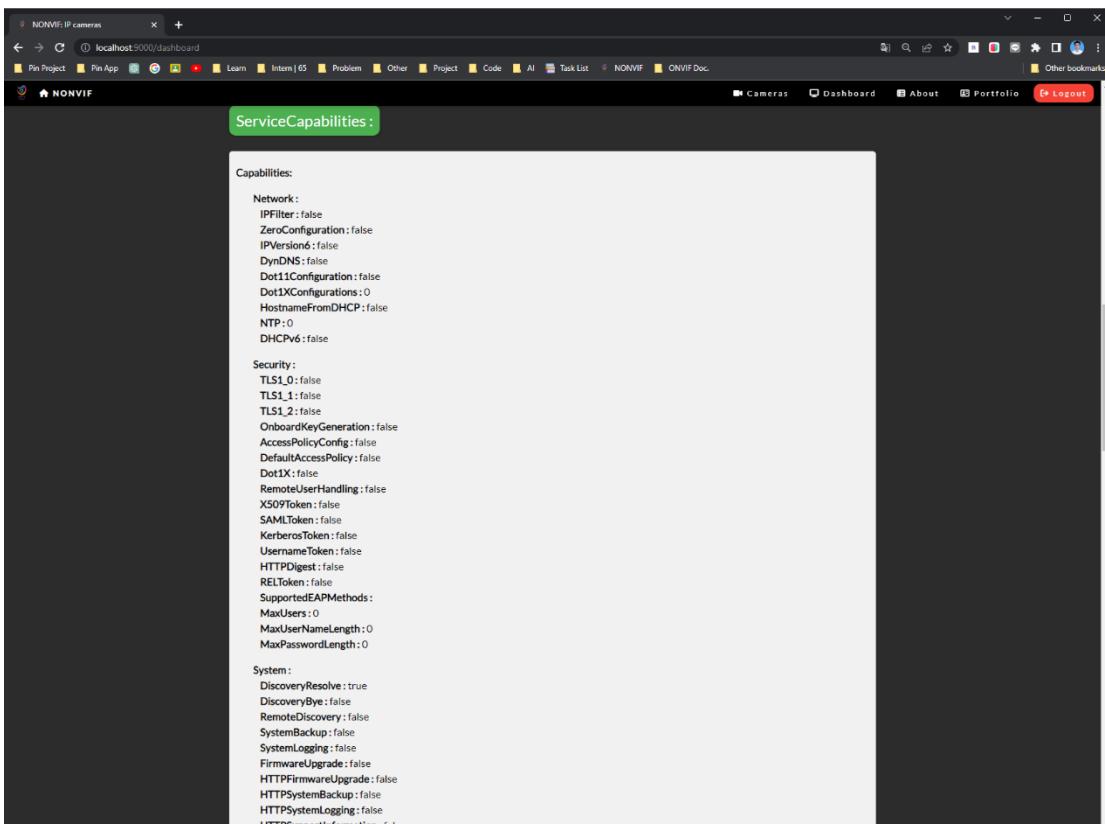
ภาพที่ 76 ภาพแสดงการเรียกข้อมูลแบบใหม่จากฟังก์ชัน GetHostname, GetNetworkProtocols,

GetDeviceInformation และ GetUser



ภาพที่ 77 ภาพแสดงการเรียกข้อมูลแบบใหม่จากฟังก์ชัน GetDiscoveryMode และ

GetSystemDateAndTime



ภาพที่ 78 ภาพแสดงการเรียกข้อมูลแบบใหม่จากฟังก์ชัน GetServiceCapabilities

ผลจากการพัฒนา ผู้ใช้จะเห็นได้ว่าข้อมูลที่แสดงให้เห็นดังภาพที่ 76 ถึงภาพที่ 78 นั้น สามารถอ่านง่ายมากยิ่งขึ้น จึงเป็นข้อสังเกตสำหรับผู้ที่อยากรู้โครงงานนี้นำไปพัฒนาต่อโดยเพิ่มการใช้ฟังก์ชันอื่นของ ONVIF ให้ทำการเข้าถึงข้อมูลด้วยวิธีดังภาพที่ 74 และภาพที่ 75 เพื่อให้แสดง ข้อมูลออกมาให้ผู้ใช้งานรู้ได้โดยง่ายและอ่านได้สะดวกมากยิ่งขึ้น

### 4.3 การทดสอบเพิ่มเติม

ในส่วนนี้จะเป็นการทดสอบเก็บค่าจากการปรับเพิ่มความละเอียดวิดีโอของกล้องไอพีหลังจากนั้นจะทำการเรียกกล้องไอพินั่นไปสตรีมยังหน้าเว็บไซต์และเก็บค่าการทำงานในส่วนต่าง ๆ

โดยจะทำการทดสอบกับตัวกล้องไอพีทั้งหมด จำนวน 5 เครื่อง  
แบ่งเป็นการเก็บค่าทั้งหมด 5 ค่า คือ

1. CPU ของ Google Chrome
2. Memory ของ Google Chrome
3. CPU ของ Visual Studio Code
4. Memory ของ Visual Studio Code
5. Network

แบ่งเป็นการทดสอบเก็บค่าจากความละเอียดของกล้องวิดีโอทั้งหมด 4 ความละเอียด คือ

1. ที่ความละเอียด 480p : Standard Definition(SD)
2. ที่ความละเอียด 720p : High Definition(HD)
3. ที่ความละเอียด 1080p : Full High Definition(Full HD)
4. ที่ความละเอียด 2160p : Ultra High Definition(UHD) หรือ 4k

ตารางที่ 7 ตารางแสดงค่าก่อนการทดสอบ

รายการ	CPU Chrome (%)	Memory Chrome (%)	CPU VScode (%)	Memory VScode (%)	Network (Mbps)
ค่าก่อนการทดสอบ	2.4	5.3	0.8	3.5	0

ตารางที่ 8 ความละเอียด 480p : Standard Definition(SD)

รายการ	CPU Chrome (%)	Memory Chrome (%)	CPU VScode (%)	Memory VScode (%)	Network (Mbps)
กล้องไอพีตัวที่ 1	8.2	5.8	1.2	3.9	6.3
กล้องไอพีตัวที่ 2	8.3	5.7	1.4	4.0	8.4
กล้องไอพีตัวที่ 3	8.8	5.4	1.3	3.8	7.3
กล้องไอพีตัวที่ 4	8.4	5.7	1.1	3.9	8.2
กล้องไอพีตัวที่ 5	8.6	5.6	1.2	3.8	6.9

ตารางที่ 9 ความละเอียด 720p : High Definition(HD)

รายการ	CPU Chrome (%)	Memory Chrome (%)	CPU VScode (%)	Memory VScode (%)	Network (Mbps)
กล้องไอพีตัวที่ 1	14.7	5.2	1.3	4.4	16.3
กล้องไอพีตัวที่ 2	14.2	5.1	1.2	4.3	17.6
กล้องไอพีตัวที่ 3	12.3	4.9	1.1	4.2	16.8
กล้องไอพีตัวที่ 4	13.7	5.3	1.2	4.4	17.2
กล้องไอพีตัวที่ 5	14.5	5.6	1.4	4.3	18.1

ตารางที่ 10 ความละเอียด 1080p : Full High Definition(Full HD)

รายการ	CPU Chrome (%)	Memory Chrome (%)	CPU VScode (%)	Memory VScode (%)	Network (Mbps)
กล้องไอพีตัวที่ 1	37.5	5.8	1.2	4.3	27.8
กล้องไอพีตัวที่ 2	39.6	5.7	1.3	4.2	52
กล้องไอพีตัวที่ 3	38.7	5.6	1.3	4.4	31.6
กล้องไอพีตัวที่ 4	39.2	5.7	1.4	4.3	29.4
กล้องไอพีตัวที่ 5	40.4	6.0	1.2	4.4	34.6

ตารางที่ 11 ความละเอียด 2160p : Ultra High Definition(UHD หรือ 4k)

รายการ	CPU Chrome (%)	Memory Chrome (%)	CPU VScode (%)	Memory VScode (%)	Network (Mbps)
กล้องไอพีตัวที่ 1	49.1	6.3	1.2	4.1	79.6
กล้องไอพีตัวที่ 2	47.7	6.2	1.3	4.2	93.4
กล้องไอพีตัวที่ 3	46.2	6.3	1.1	4.0	77.5
กล้องไอพีตัวที่ 4	47.1	6.4	1.2	4.3	76.4
กล้องไอพีตัวที่ 5	48.9	6.3	1.1	4.2	82.1

จากการทดสอบเก็บค่าจากการปรับเพิ่มความละเอียดของวิดีโอที่ความละเอียดที่ต่างกัน สามารถสังเกตได้ว่า ค่าที่มีการเพิ่มการใช้งานตามการปรับเปลี่ยนความละเอียดของวิดีโอดังนี้ ค่าของ CPU ของ Google Chrome และค่าของ Network เนื่องจากเมื่อมีการเปลี่ยนความละเอียดของวิดีโอดังนั้น จึงส่งผลให้ CPU ของ Google Chrome และค่าของ Network มีการปรับเพิ่มการใช้งานสูงมากขึ้นตามสัดส่วนความละเอียดของวิดีโอที่ถูกปรับเพิ่มขึ้น การทดสอบนี้เป็นการทดสอบที่ความเร็วของอินเทอร์เน็ตที่ 100Mbps ภาพความละเอียดของวิดีโอที่สตรีมแล้วมีความเสถียรมีค่าหน่วงเวลาที่ต่ำและยังคงความชัดอยู่ แนะนำให้ใช้งานคือที่ความละเอียด 1080p: Full High Definition(Full HD)

## บทที่ 5

### สรุปผลและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงข้อสรุปจากการออกแบบและพัฒนาการประยุกต์ใช้โปรโตคอล ONVIF เพื่อรับการจัดการกล้องไอพีและการสตรีมวิดีโอผ่านเว็บไซต์ และข้อเสนอแนะต่างๆ ดังต่อไปนี้

#### 5.1 สรุปผลการดำเนินงาน

ในการดำเนินการประยุกต์ใช้โปรโตคอล ONVIF เพื่อรับการจัดการกล้องไอพีและการสตรีมวิดีโอผ่านเว็บไซต์ จากการวิเคราะห์รวมข้อมูลและออกแบบระบบรวมถึงขั้นตอนในการพัฒนาระบบ ผลงานให้ได้ระบบที่ผู้ใช้สามารถติดตั้งและนำโปรแกรมที่พัฒนาขึ้นมาในโครงการนี้ไปใช้งานกับเครื่องของตนเองได้ เพื่อใช้ในการเรียกดูภาพจากกล้องไอพีรวมถึงการข้อมูลเมตาต่างๆ จากกล้องไอพีผ่านการใช้งานของโปรโตคอล ONVIF แบบเรียลไทม์ได้ที่หน้าเว็บไซต์

จากการออกแบบและพัฒนาระบบรวมถึงการนำมาใช้งานถือว่าประสบความสำเร็จ หน้าเว็บสามารถทำการสตรีมภาพจากกล้องไอพีรวมถึงการเรียกดูข้อมูลเมตาจากกล้องผ่านการเรียกดูข้อมูลด้วยโปรโตคอล ONVIF และสามารถทำงานในส่วนของการจัดการเพิ่มหรือลบกล้องไอพี เลือกดูภาพจากกล้องไอพีแต่ละตัวมีส่วนใจได้ โดยตรงตามวัตถุประสงค์ของการจัดทำโดยมีรายละเอียด ดังนี้

1. ผู้วิจัยได้ศึกษาปัญหาและได้ศึกษาถึงเทคโนโลยีที่เกี่ยวข้องและได้นำเทคโนโลยีที่เหมาะสมกับโครงการมาประยุกต์ใช้กับการพัฒนาการประยุกต์ใช้โปรโตคอล ONVIF เพื่อรับการจัดการกล้องไอพีและการสตรีมวิดีโอผ่านเว็บไซต์
2. ผู้วิจัยได้นำความรู้และทฤษฎีที่เกี่ยวข้องที่ได้ทำการศึกษาตลอดการทำโครงการนี้ นำมาวิเคราะห์และออกแบบระบบงานทำให้หน้าเว็บ การสตรีมภาพจากกล้องไอพี การเรียกดูข้อมูลเมตาจากกล้องไอพีและการจัดการกับกล้องไอพีนั้นสามารถทำงานได้อย่างมีประสิทธิภาพ และได้ระบบที่ตรงต่อ กับความต้องการของผู้ใช้งานที่อยากรีบกล้องไอพีที่ไม่ได้จำกัดที่ยังห้องใดยังห้องนึงในการใช้งาน ซึ่งหากผู้ใช้เลือกใช้หากทลายยี่ห้อในห้องตลาดจะส่งผลให้ต้องมีการติดตั้งโปรแกรมเพื่อใช้งานกล้องยี่ห้อนั้น ๆ มากตามไปด้วย โครงการนี้จึงมีมาเพื่อลดปัญหาในส่วนนี้ โดยผู้ใช้งานนั้นสามารถเรียกดูภาพจากไอพีของกล้องที่หน้าเว็บไซต์ได้โดยเพียงแค่ผู้ใช้รีบกล้องไอพีตัวนั้น ๆ และจะต้องเป็นกล้องที่รู้ที่อยู่ไอพีแบบ HTTP เพียงเท่านั้น

## 5.2 ข้อเสนอแนะเพิ่มเติม

### 5.2.1 เกี่ยวกับระบบเซิร์ฟเวอร์

ระบบนี้ทำงานที่ localhost ผ่าน HTTP ปกติ ซึ่งทำให้เห็นว่าယังขาดเรื่องความปลอดภัยของระบบ แม้จะมีการให้ลิงทะเบียนผู้ใช้ก่อนเข้าใช้งานก็ตาม ซึ่งต่อไปเมื่อ Server มีความพร้อม และพัฒนาไปใช้ Domain Name แทนการใช้งานแบบปัจจุบัน ด้วยการติดตั้ง Certificate และใช้การเข้ามต่อผ่าน HTTPS แทนได้ ซึ่งจะเป็นการเพิ่มระดับความปลอดภัยให้กับระบบมากขึ้น อีกขั้นหนึ่ง

### 5.2.2 เกี่ยวกับออกแบบ

ในการพัฒนาโครงการนี้นั้นบางส่วนของการออกแบบ UX/UI ในบางจุดนั้นอาจยังไม่อำนวยความสะดวกต่อการใช้งานของผู้ใช้ในบางส่วน เนื่องจากเป็นการออกแบบและทดลองใช้แค่เพียงกับตัวผู้พัฒนา ซึ่งผู้ที่สนใจใช้พัฒนาต่อสามารถนำไปทดลองเก็บผลกับผู้ใช้หลาย ๆ คน เพื่อเก็บผลตอบรับและคำแนะนำในการแก้ไขปรับปรุงต่าง ๆ จากผู้ใช้และนำมาปรับปรุงให้ตรงต่อความต้องการของผู้ใช้ให้มากขึ้นต่อไป ซึ่งต้องใช้ความคิดสร้างในการออกแบบสิ่งเหล่านี้มากขึ้นไปด้วย

### 5.2.3 เกี่ยวกับการพัฒนาเพิ่มเติม

จากการดำเนินโครงการในครั้งนี้นั้นในส่วนที่ทำได้ในโครงการ ณ ขณะนี้เป็นการเรียกใช้งาน ONVIF ในบางฟังก์ชันเพื่อเรียกดูข้อมูลต่างๆ จากกล้องไอพีผ่าน ONVIF Protocol แค่เพียงในบางฟังก์ชันของ ONVIF เพียงเท่านั้นผู้ที่สนใจใช้พัฒนาต่อสามารถศึกษาฟังก์ชันอื่น ๆ เพิ่มเติม เช่น ฟังก์ชันเกี่ยวกับ PTZ(Pan-Tilt-Zoom), เกี่ยวกับการจัดส่งเหตุการณ์, เกี่ยวกับการวิเคราะห์และการบันทึกภาพวิดีโอ เป็นต้น และผู้พัฒนาได้มองเห็นถึงโอกาสในการเรียนรู้การต่อยอดและพัฒนารูปแบบของระบบนี้ได้อีกมากมาย เช่น หากอยากรพัฒนาต่อเป็นการตรวจจับวัตถุ ตรวจจับความเร็ว ตรวจจับอุณหภูมิหรือบุคคลก็สามารถพัฒนาต่อได้ โดยอาจจะเลือกใช้งานร่วมกับ OpenCV ที่มี Library อยู่ในภาษา GoLang คือ go-opencv หรือ gocv เป็นต้น โดยผู้สนใจพัฒนาต่อสามารถนำเอา Library เหล่านี้มาพัฒนาต่อยอดจากโครงการนี้ได้ในอนาคต

## ເອກສາຣ້ອງຈົງ

- [1] Phan, C. C., Nguyen, T. B. and Chung, S.-T. (2015). Design and Implementation of ONVIF-based Event Service for DM 814x Camera. **International Conference on Communications, Management and Telecommunications (ComManTel)**. pp. 73-77.
- [2] Wenfeng, L. and Yuan, G. (2018). Design Of Network Camera Based On ONVIF In The Dark Environment. **International Conference on Sensor Networks and Signal Processing (SNSP)**. pp. 84-87.
- [3] Lopes, S. F., and et al. (2013). Development of a library for clients of ONVIF video cameras: challenges and solutions. **International Conference on Industrial Technology (ICIT)**. pp. 1260-1266.
- [4] Tsai, Y.-H., Hsu, J.-K., Wu, Y.-E. and Huang, W.-F. (2011). Distributed Multimedia Content Processing in ONVIF Surveillance System. **International Conference on Future Computer Sciences and Application (ICFCSA)**. pp. 70-73.
- [5] Senst, T. and et al. (2011). On Building Decentralized Wide-Area Surveillance Networks based on ONVIF. **IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)**. pp. 420-423.
- [6] Lin, C.-F. and et al. (2018). A fault-tolerant ONVIF protocol extension for seamless surveillance video stream recording. **Computer Standards & Interfaces**. pp. 55-72.

## ประวัติของผู้เขียน

ชื่อ – สกุล : นายนันทกร โสภาค  
สาขาวิชา : วิศวกรรมระบบสมองกลฝังตัว  
คณะ : วิศวกรรมศาสตร์  
มหาวิทยาลัย : มหาวิทยาลัยบูรพา  
ที่อยู่ : 29/25 ม.3 ช.ร่วมเจริญ ถ.สุขสวัสดิ์ ต.ในคลองบางปลากรด  
อ.พระสมุทรเจดีย์ จ.สมุทรปราการ 10290  
E-mail : [62050553@go.buu.ac.th](mailto:62050553@go.buu.ac.th) / [nuntakorn.sp@gmail.com](mailto:nuntakorn.sp@gmail.com)

