

ระบบกริ่งอัจฉริยะ

Doorbell Intelligent System



ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Doorbell Intelligent System



SPECIAL PROBLEM SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF BACHELOR OF SCIENCE
IN COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE
FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้า
ACADEMIC YEAR 2558
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ ระบบกริ่งอัจฉริยะ

ชื่อนักศึกษา นายณัฐวุฒิ พิทักษ์นรเศรษฐ รหัส 55050299
 นายทศพล อนันต์สินชัย รหัส 55050309
 นายธีระวัฒน์ ธีระวัฒนสุข รหัส 55050337

ปริญญา วิทยาศาสตร์บัณฑิต

สาขาวิชา วิทยาการคอมพิวเตอร์

ปีการศึกษา 2558

อาจารย์ที่ปรึกษา ผศ.ธีระ ศิริธีรากุล

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้
 ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา
 วิทยาศาสตร์บัณฑิต ประจำปีการศึกษา 2558

คณะกรรมการสอบ	ลายมือชื่อ
ดร.สายชล ใจเย็น ประธานกรรมการ	
อ.สันธนะ อุ่อดมยิ่ง กรรมการ	
ผศ.ธีระ ศิริธีรากุล กรรมการและอาจารย์ที่ปรึกษา	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ลิขสิทธิ์ของคณะวิทยาศาสตร์ อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง รทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	ระบบกริ่งอัจฉริยะ		
ชื่อนักศึกษา	นายณัฐวุฒิ พิทักษ์นรเศรษฐ	รหัส	55050299
	นายทศพล อนันต์สินชัย	รหัส	55050309
	นายธีระวัฒน์ ธีระวัฒนสุข	รหัส	55050337
ปริญญา	วิทยาศาสตร์บัณฑิต		
สาขาวิชา	วิทยาการคอมพิวเตอร์		
ปีการศึกษา	2558		
อาจารย์ที่ปรึกษา	ผศ.ธีระ ศิริธีรากล		

บทคัดย่อ

ระบบกริ่งอัจฉริยะ (Doorbell Intelligent System) จัดทำขึ้นเพื่อป้องกันและลดความเสี่ยงอันตรายที่จะเกิดขึ้นกับบุคคลภายในบ้านเมื่อมีผู้มากดกริ่ง โดยระบบจะใช้เทคโนโลยี Streaming Technology ส่งภาพและเสียงไปยังอุปกรณ์สมาร์ตโฟนของบุคคลในบ้านผ่านทางระบบเครือข่ายไร้สาย (Wireless) เพื่อพูดคุยและซักถามประกอบกับการตัดสินใจที่จะเลือกเปิดประตูต้อนรับผู้ที่มากดกริ่ง หากต้องการเปิดประตูต้อนรับผู้ที่มากดกริ่ง ผู้ใช้สมาร์ตโฟนสามารถกดคำสั่งเปิดประตูอัตโนมัติผ่านทางสมาร์ตโฟน เพื่ออำนวยความสะดวกให้แก่บุคคลภายในบ้านได้มากยิ่งขึ้น

ทั้งนี้ระบบกริ่งอัจฉริยะยังสามารถบันทึกภาพและเวลาของผู้ที่กดกริ่งได้ ซึ่งเป็นประโยชน์ต่อการตรวจสอบเวลาและบุคคลที่มากดกริ่งเมื่อเกิดเหตุการณ์ไม่คาดคิดเกิดขึ้นกับบุคคลภายในบ้าน

คำสำคัญ : ไอโอที ราวเซอร์พาย กริ่ง แอนดรอยด์แอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title Doorbell Intelligent System

Students Mr. Natthawut Pitaknorasate 55050299
Mr. Tosapon Anantasinchai 55050309
Mr. Thirawat Thirawatthanasuk 55050337

Degree Bachelor of Science

Department Computer Science

Academic Year 2558

Advisor Asst. Prof. Teera Siriteerakul

Abstract

Doorbell Intelligent System was prepared for preventing and reducing risk of house members, when someone ring the bell .The system will use Streaming Technology to transmit picture and voice to house members Smartphone device through Wireless connection, for talking and deciding whether to open the door .If house members want to open the door for the guest, house members can use their Smartphone to trigger the door, which is much more convenient.

In addition, Doorbell Intelligent System can record a picture and visiting time of the guest, so it is easy to investigate the individual time and person when unexpected things happen to house members.

Keyword : IoT, Raspberry Pi, Doorbell, Android Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

รายงานเล่มนี้สำเร็จลงได้ด้วยความช่วยเหลืออย่างดียิ่งจาก ผศ.ธีระ ศิริธีรากุล อาจารย์ที่ปรึกษา ซึ่งท่านได้ให้คำแนะนำและข้อคิดเห็นต่างๆ ในการทำรายงานด้วยดีตลอดมา

ขอขอบพระคุณ ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ สำนักพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ กระทรวงวิทยาศาสตร์และเทคโนโลยี ที่เป็นผู้สนับสนุนเงินอุดหนุนในการจัดทำโครงการระบบกริ่งอัจฉริยะ (Doorbell Intelligent System) ภายใต้โครงการการแข่งขันพัฒนาโปรแกรมคอมพิวเตอร์แห่งประเทศไทย ครั้งที่ 18 อันเป็นประโยชน์ต่อคณะผู้จัดทำในการทำรายงานฉบับนี้

นาย ธีระวุฒิ พัทธ์ธีรเศรษฐ์

นาย ทศพล อนันต์สินชัย

นาย ธีระวัฒน์ ธีระวัฒนสุข



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ช
สารบัญรูป.....	ซ
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของปัญหาพิเศษ.....	1
1.3 ขอบเขตของปัญหาพิเศษ.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.5 ขั้นตอนการดำเนินการ.....	2
1.6 อุปกรณ์ที่ใช้ในการทำปัญหาพิเศษ.....	3
1.7 งบประมาณ.....	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	4
2.1 การควบคุมอุปกรณ์ภายในบ้านจาก Android Application และ Raspberry Pi.....	4
2.2 ทฤษฎีที่เกี่ยวข้องกับสตรีมมิ่งมีเดีย.....	6
2.2.1 ความเป็นมาและความหมายของสตรีมมิ่งมีเดีย.....	6
2.2.2 ความแตกต่างระหว่างการดาวน์โหลดและการส่งแบบสตรีมมิ่งมีเดีย.....	7
2.2.3 ลักษณะการส่งสตรีมมิ่งมีเดีย.....	8
2.2.4 องค์ประกอบพื้นฐานของระบบสตรีมมิ่งมีเดีย.....	9
2.4 โพรโทคอล (Protocol).....	10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.4.1 รูปแบบการส่งข้อมูลของโพรโทคอลในระบบเครือข่าย	11
2.5 ซ็อกเกต (Socket).....	13
2.5.1 Working with Sockets in Java	15
บทที่ 3 ขั้นตอนการออกแบบและพัฒนาซอฟต์แวร์.....	17
3.1 การออกแบบฮาร์ดแวร์	17
3.2 ขั้นตอนการทำงานของ Application บน Raspberry Pi.....	18
3.2.1 ฟังก์ชันสตรึมมิ่งข้อมูลและรอรับการเชื่อมต่อ	18
3.2.4 ตารางฐานข้อมูล.....	21
3.3 ขั้นตอนการทำงานของ Application บน Android Smartphone.....	21
3.4 การออกแบบส่วนติดต่อกับผู้ใช้ (User Interface) บน Android Application.....	30
บทที่ 4 ผลการดำเนินงานและอภิปรายผล.....	35
4.1 การทดสอบฟังก์ชันบนอุปกรณ์สมาร์ตโฟนที่เชื่อมต่อกับเครื่องอัจฉริยะ	35
4.1.1 ฟังก์ชันการค้นหาอุปกรณ์กริ่งบนวงแลนเดียวกัน.....	35
4.1.2 ฟังก์ชันสำหรับการโต้ตอบระหว่างผู้ใช้งานบนอุปกรณ์สมาร์ตโฟนกับบุคคลที่มากดกริ่ง อัจฉริยะ	37
4.1.3 ฟังก์ชันการรับชมรูปที่ถูกสตรีมมาจากกริ่ง.....	38
4.1.4 ฟังก์ชันการเก็บบันทึกข้อมูลของบุคคลที่กดกริ่งและผู้ที่ได้รับสายจากกริ่ง	39
4.1.5 ฟังก์ชันการปลดล๊อคประตู.....	39
4.1.6 ฟังก์ชันการตั้งค่าอุปกรณ์.....	41
4.3 ตัวอย่างโค้ดแอปพลิเคชันอุปกรณ์กริ่งอัจฉริยะที่พัฒนาด้วยภาษา Python	42
4.3.1 ตัวอย่างโค้ดค้นหา Wi-Fi และทำการเชื่อมต่ออุปกรณ์ Wi-Fi.....	42
4.3.2 ตัวอย่างโค้ดการจัดการกับรหัสผ่านทั้งการเชื่อมต่อมายังอุปกรณ์และระบบล๊อคประตู.....	44

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
4.3.3 ตัวอย่างโค้ดการสตรีมมิ่งภาพ เสียง และการรับเสียงจากแอปพลิเคชันมาแสดงผล.....	45
4.3.4 ตัวอย่างโค้ดการทำงานเมื่อมีการกดกริ่ง	48
4.3.5 ตัวอย่างโค้ดควบคุมการทำงานของระบบลือคประตู่.....	49
4.4 ตัวอย่างการเชื่อมต่อวงจรรูปกรณ์กริ่งอัจฉริยะผ่าน GPIO	52
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ.....	54
5.1 สรุปผลการดำเนินงาน	54
5.2 ขอบเขตและข้อจำกัดของการพัฒนา	54
5.3 ข้อเสนอแนะ.....	55
เอกสารอ้างอิง	56
ภาคผนวก.....	56
ภาคผนวก ก. วิธีการติดตั้งและคู่มือการใช้งานแอปพลิเคชัน Doorbell Intelligent.....	58
ภาคผนวก ข. ขั้นตอนการติดตั้ง “Java JDK 7u75”	70
ภาคผนวก ค. การติดตั้งโปรแกรม Android Studio.....	73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่		หน้า
2.1	เปรียบเทียบการส่งผ่านข้อมูลแบบสตรีมมิ่งมีเดียกับการดาวน์โหลดไฟล์.....	8
2.2	แบบจำลองเพื่อการต่อเชื่อมระบบแบบเปิด	11
3.1	ตารางบันทึกการใช้งานอุปกรณ์เครื่องอัจฉริยะ.....	22
3.2	คำอธิบายยูสเคส กดกริ่ง.....	22
3.3	คำอธิบายยูสเคส รับสายจากอุปกรณ์	23
3.4	คำอธิบายยูสเคส ตั้งรหัสผ่าน.....	23
3.5	คำอธิบายยูสเคส ยืนยันตัวตน	24
3.6	คำอธิบายยูสเคส รีเซ็ตรหัสผ่าน.....	24
3.7	คำอธิบายยูสเคส เลือกเชื่อมต่ออุปกรณ์.....	25
3.8	คำอธิบายยูสเคส เปลี่ยนรหัสผ่านจากกล้อง.....	25
3.9	คำอธิบายยูสเคส จัดการ WiFi.....	26
3.10	คำอธิบายยูสเคส ลบอุปกรณ์จากแอปพลิเคชัน	26
3.11	คำอธิบายยูสเคส เรียกดูกล้องจากอุปกรณ์	27
3.12	คำอธิบายยูสเคส ดูประวัติการกดกริ่งย้อนหลัง	28
3.13	คำอธิบายยูสเคส ดูประวัติสายไม่ได้รับ	28
3.14	คำอธิบายยูสเคส ปลดล็อคประตู.....	28
3.15	คำอธิบายยูสเคส ลบข้อมูลประวัติย้อนหลัง	29
3.16	คำอธิบายยูสเคส เรียกดูกล้องจากอุปกรณ์	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1	รหัสเบอร์รีพาย	4
2.2	การแก้ปัญหาโดยภาพรวม	5
2.3	การส่งข้อมูลของโพรโทคอลแบบ Connection-Oriented	12
2.4	การส่งข้อมูลของโพรโทคอลแบบ Connectionless	13
2.5	การเชื่อมต่อของซอคเกตแบบ Connection-Oriented	14
2.6	การเชื่อมต่อของซอคเกตแบบ Connectionless	14
2.7	Diagram ขั้นตอนการสร้าง Passive Sockets	15
2.8	Diagram ขั้นตอนการสร้าง Active Sockets	16
3.1	การเชื่อมต่ออุปกรณ์ภายในระบบ	17
3.2	Flowchart การทำงานของการสตรีมมิ่งข้อมูลและรับการเชื่อมต่อ	18
3.3	Flowchart การทำงานเมื่อมีการกดกริ่งและส่งสัญญาณให้ Raspberry Pi	19
3.4	Flowchart การทำงานเมื่อมีการส่งข้อมูลเสียงมายัง Raspberry Pi	20
3.5	Use Case Diagram	21
3.6	หน้าจอแสดงการ Register	30
3.7	หน้าจอแสดงการยืนยันรหัส PIN 4 หลัก	31
3.8	แสดงการปลดล็อค PIN	31
3.9	หน้าจอแสดงการค้นหาอุปกรณ์	32
3.10	หน้าจอแสดงการ Login เข้าสู่ระบบของอุปกรณ์	32
3.11	หน้าจอแสดงภาพวิดีโอจาก อุปกรณ์ที่เชื่อมต่อ	33
3.12	หน้าจอแสดง Log ของอุปกรณ์	33
3.13	หน้าจอแสดง Miss Call	34
3.14	หน้าจอแสดง Setting	34
4.1	หน้าจอแสดงการค้นหาอุปกรณ์ในวงแลนเดียวกัน	35
4.2	หน้าจอแสดงการเชื่อมต่ออุปกรณ์แบบ Manually	36
4.3	การเรียกดู ip จากกริ่งอัจฉริยะ	36
4.4	ภาพแสดงเมื่อกดปุ่มที่อุปกรณ์กริ่งและส่งภาพไปยังอุปกรณ์สมาร์ทโฟน (ขณะยังไม่รับสาย)	37
4.5	แสดงเมื่อกดปุ่มที่อุปกรณ์กริ่งและส่งภาพไปยังอุปกรณ์สมาร์ทโฟน(ขณะรับสาย)	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.6	หน้าจอหลัก.....	38
4.7	หน้าจอการรับชมรูปที่ถูกสตรีมมาจากกริ่ง.....	38
4.8	หน้าจอแสดงการเก็บบันทึกข้อมูลของบุคคลที่กดกริ่งและผู้ที่ได้รับสายจากกริ่ง.....	39
4.9	การแสดงการปลดล็อคโดยแอปพลิเคชัน.....	39
4.10	การแสดงการปลดล็อคโดยหน้าจอสายเรียกเข้า.....	40
4.11	การแสดงการปลดล็อคโดยกดรหัสที่อุปกรณ์.....	40
4.12	หน้าจอการตั้งค่าอุปกรณ์ต่างๆ.....	41
4.13	แสดงตัวอย่าง class ที่ใช้ในการส่งข้อมูล SSID ที่ค้นหาได้ไปยังแอปพลิเคชัน.....	42
4.14	แสดงตัวอย่าง class ที่ใช้ในการรับข้อมูล SSID ที่ผู้ใช้งานเลือกที่จะเชื่อมต่ออุปกรณ์เข้ากับ Wi-Fi มาทำการแก้ไขไฟล์ config บนอุปกรณ์กริ่งอัจฉริยะ.....	43
4.15	แสดงตัวอย่างโค้ดที่ใช้ในการตรวจสอบรหัสผ่านเมื่อผู้ใช้งานเชื่อมต่อมายังอุปกรณ์.....	44
4.16	แสดงตัวอย่างการแก้ไขรหัสผ่านที่ใช้สำหรับเชื่อมต่อมายังอุปกรณ์กริ่ง.....	44
4.17	แสดงตัวอย่างการแก้ไขรหัสผ่านที่ใช้สำหรับการปลดล็อคประตู.....	45
4.18	แสดงตัวอย่างการใช้งาน VLC Library ในการสตรีมมิงภาพ.....	45
4.19	แสดงตัวอย่างการสตรีมมิงเสียงเมื่อมีการร้องขอจากแอปพลิเคชันมายังตัวอุปกรณ์กริ่ง....	46
4.20	แสดงตัวอย่างการรับข้อมูลเสียงจากแอปพลิเคชันมาแสดงผลบนอุปกรณ์กริ่ง.....	47
4.21	แสดงตัวอย่างขั้นตอนการทำงานเมื่อมีการกดกริ่ง.....	48
4.22	แสดงตัวอย่างฟังก์ชันที่ถูกเรียกใช้เมื่อมีการกดกริ่ง.....	48
4.23	แสดงตัวอย่างขั้นตอนการทำงานอื่นๆในฟังก์ชันการถ่ายภาพของผู้ที่มากดกริ่ง.....	49
4.24	แสดงตัวอย่างโค้ด PHP ที่รับค่าสถานะประตูจากแอปพลิเคชัน.....	49
4.25	แสดงตัวอย่างขั้นตอนการส่งงานรีเลย์ที่ใช้สำหรับควบคุมการล็อคประตู.....	50
4.26	แสดงตัวอย่างขั้นตอนการรับค่าปุ่มกดแบบตัวเลขเพื่อใช้ในการปลดล็อคประตูและแสดงข้อความสถานะประตูผ่านจอ LCD และหลอดไฟ LED.....	51
4.27	ภาพแสดงตัวอย่างวงจรที่เชื่อมต่อกับอุปกรณ์ภายนอกผ่าน GPIO Pin บนอุปกรณ์กริ่งอัจฉริยะ.....	52
ก.1	ตรวจสอบไฟล์ .apk ที่ใช้ติดตั้งบนสมาร์ตโฟนผ่านโปรแกรม File Commander.....	58
ก.2	หน้าจอแสดงตัวเลือกการติดตั้งไฟล์ .apk.....	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

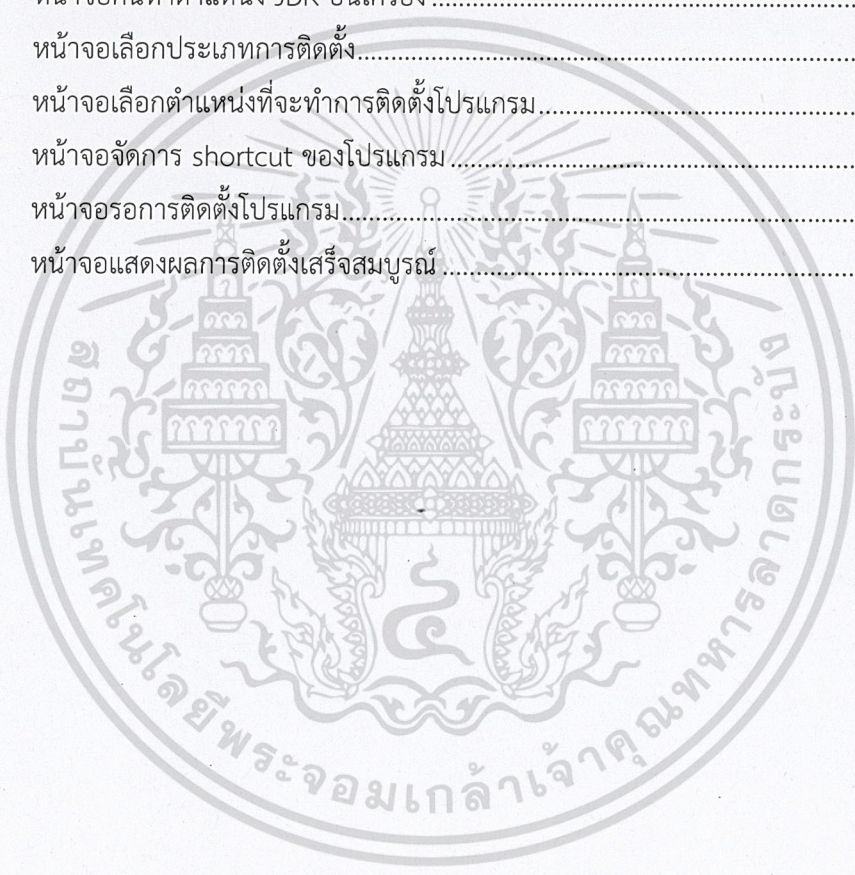
สารบัญรูป (ต่อ)

รูปที่		หน้า
ก.3	หน้าจอแสดงรายละเอียดการเข้าถึงข้อมูลของแอปพลิเคชัน	59
ก.4	แสดงการแจ้งเตือนเมื่อติดตั้งแอปพลิเคชันสมบูรณ์	59
ก.5	ไอคอนแอปพลิเคชัน	60
ก.6	หน้าจอ loading ของแอปพลิเคชัน	60
ก.7	หน้าจอสร้าง PIN Code และการเข้าใช้งานผ่าน PIN Code ของแอปพลิเคชัน	61
ก.8	หน้าจอ scan หาอุปกรณ์ในวงแลนเดียวกัน	61
ก.9	หน้าจอ scan หาอุปกรณ์ในวงแลนเดียวกัน	62
ก.10	หน้าจอเชื่อมต่อไปยังอุปกรณ์กริ่ง	62
ก.11	หน้าเมนูหลักของแอปพลิเคชัน	63
ก.12	ฟังก์ชันรับชมภาพจากอุปกรณ์กริ่ง	63
ก.13	หน้าจอฟังก์ชันการรับชมภาพจากอุปกรณ์กริ่ง	64
ก.14	ฟังก์ชันดูข้อมูลการกดกริ่งที่ไม่มีการตอบรับ	64
ก.15	หน้าจอแสดงข้อมูลการกดกริ่งที่ไม่มีการตอบรับ	65
ก.16	ฟังก์ชันปลดล็อคประตูผ่านแอปพลิเคชัน	65
ก.17	เมื่อเลือกฟังก์ชันปลดล็อคประตู	66
ก.18	ฟังก์ชันดูบันทึกรายการของผู้ที่มากดกริ่ง	66
ก.19	หน้าจอฟังก์ชันแสดงรายการข้อมูลของผู้ที่มากดกริ่ง	67
ก.20	ฟังก์ชันตั้งค่าอุปกรณ์	67
ก.21	หน้าจอฟังก์ชันตั้งค่าอุปกรณ์	68
ก.22	หน้าจอแจ้งเตือนเมื่อมีการกดกริ่ง	68
ก.23	หน้าจอเมื่อตอบรับผู้ที่มากดกริ่ง	69
ก.24	อุปกรณ์กริ่งอัจฉริยะ	69
ข.1	ไฟล์ติดตั้ง “Java JDK 7u75”	70
ข.2	เริ่มต้นการติดตั้ง “Java JDK 7u75”	70
ข.3	คำอธิบายต่างๆเกี่ยวกับ “Java7 JDK 7u75”	71
ข.4	เลือก Directory “Java7 JDK 7u75”	71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
ข.5	เริ่มการติดตั้ง “Java7 JDK 7u75”	72
ข.6	เสร็จสิ้นการติดตั้ง “Java7 JDK 7u75”	72
ค.1	ตัวอย่างไฟล์โปรแกรม android studio หลังจากทำการดาวน์โหลด	73
ค.2	หน้าจอตอนรับเมื่อเริ่มทำการติดตั้งโปรแกรม	73
ค.3	หน้าจอค้นหาตำแหน่ง JDK บนเครื่อง	74
ค.4	หน้าจอเลือกประเภทการติดตั้ง	74
ค.5	หน้าจอเลือกตำแหน่งที่จะทำการติดตั้งโปรแกรม	75
ค.6	หน้าจอจัดการ shortcut ของโปรแกรม	75
ค.7	หน้าจอรอการติดตั้งโปรแกรม	76
ค.8	หน้าจอแสดงผลการติดตั้งเสร็จสมบูรณ์	76



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของปัญหาพิเศษ

- 1.3.1 อุปกรณ์เครื่องอัจฉริยะสามารถส่งสัญญาณในรูปแบบของเสียงและวิดีโอ ไปยังสมาร์ตโฟนที่เชื่อมต่อกับเครื่องอัจฉริยะได้
- 1.3.2 อุปกรณ์เครื่องอัจฉริยะสามารถบันทึกภาพบุคคลที่มากดเครื่องได้
- 1.3.3 แอปพลิเคชันที่ถูกพัฒนาขึ้น เพื่อใช้งานร่วมกับสมาร์ตโฟนจะสามารถเห็นใบหน้าและได้ยินเสียงของผู้ที่มากดเครื่องได้แบบ Real-Time
- 1.3.4 เมื่อมีผู้มากดเครื่อง แอปพลิเคชันที่ถูกติดตั้งบนสมาร์ตโฟนจะแจ้งเตือน พร้อมทั้งแสดงภาพและเสียงของผู้ที่มากดเครื่อง
- 1.3.5 ผู้ใช้งานแอปพลิเคชันบนสมาร์ตโฟนสามารถเลือกที่จะตอบรับหรือไม่ตอบรับเมื่อมีคนมากดเครื่อง
- 1.3.6 อุปกรณ์เครื่องอัจฉริยะ ไม่สามารถส่งสัญญาณภาพและวิดีโอไปยังสมาร์ตโฟนหลายๆ เครื่องได้ในเวลาเดียวกัน
- 1.3.7 ผู้ใช้งานแอปพลิเคชันบนสมาร์ตโฟนสามารถตรวจสอบข้อมูลภาพ วัน และเวลา ของผู้ที่มากดเครื่องได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1.4.1 อุปกรณ์เครื่องอัจฉริยะสามารถช่วยอำนวยความสะดวกด้านความปลอดภัยให้แก่ผู้ใช้งานได้อย่างแท้จริง
- 1.4.2 อุปกรณ์เครื่องอัจฉริยะสามารถจดจำและตรวจสอบข้อมูลภาพ วัน และเวลา ของผู้ที่มากดเครื่องได้เพื่อป้องกันอันตรายที่อาจจะเกิดขึ้น
- 1.4.3 ผู้ใช้งานสามารถเป็นผู้เลือกพิจารณาเปิดประตูต้อนรับ ผู้ที่มากดเครื่องได้ตามดุลพินิจการของผู้ใช้งานเอง

1.5 ขั้นตอนการดำเนินการ

วิธีการดำเนินงานประกอบด้วยขั้นตอนต่อไปนี้

- 1.5.1 วางแผนการดำเนินงาน
- 1.5.2 เก็บรวบรวมข้อมูล และวิเคราะห์ความต้องการของระบบ
- 1.5.3 ออกแบบอุปกรณ์ที่จะใช้เชื่อมต่อกับเครื่อง
- 1.5.4 พัฒนาโปรแกรมที่สามารถรับและส่งค่าไปยังอุปกรณ์ได้
- 1.5.5 ตรวจสอบและแก้ไขข้อผิดพลาดของอุปกรณ์และโปรแกรม
- 1.5.6 ประเมินผล
- 1.5.7 จัดทำรูปเล่มปัญหาพิเศษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 อุปกรณ์ที่ใช้ในการทำปัญหาพิเศษ

1.6.1 Hardware

- คอมพิวเตอร์ หรือ คอมพิวเตอร์โน้ตบุ๊ก
 - ระบบปฏิบัติการ วินโดวส์ 7 ขึ้นไป
 - CPU 1.6 GHz หรือสูงกว่า
 - RAM 4 GB ขึ้นไป (ถ้ารันบน Virtual machine ต้องใช้ RAM ตั้งแต่ 1.5 GB ขึ้นไป)
 - พื้นที่วางของหน่วยความจำต้องมีพื้นที่ไม่ต่ำกว่า 20 GB
- โทรศัพท์มือถือที่ใช้ระบบปฏิบัติการ Android
- Raspberry pi 2 Model B
- Raspberry pi Camera Noir Module
- กริ่งไร้สาย
- Electromagnet Door Lock
- Expansion Board (X200)

1.6.2 Software

- Android Studio
- Python 3
- Apache 2
- Php5
- Mysql

1.7 งบประมาณ

- 5,000 บาท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

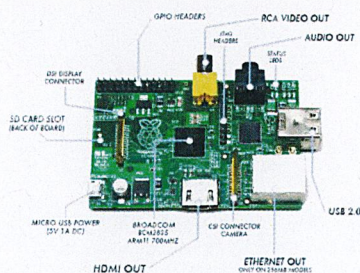
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

กริ่งหน้าบ้านถือเป็นอุปกรณ์อย่างหนึ่งที่ทุกบ้านนิยมนำมาติดไว้ใช้งานเพื่อเป็นเครื่องมือสำหรับแจ้งให้ทราบเมื่อมีบุคคลมาเยือนที่บ้าน หากกล่าวถึงกริ่งธรรมดาทั่วไปแล้วจะทำงานโดยอาศัยการกดกริ่งและส่งเสียงร้องออกมาเท่านั้น โดยกริ่งที่มีขายตามท้องตลาดโดยส่วนมากจะมีให้เลือก 2 ลักษณะ คือ กริ่งแบบมีสายและกริ่งแบบไร้สาย ในการศึกษาทฤษฎีและงานวิจัยต่างๆที่เกี่ยวข้องต่อจากนี้เป็นส่วนหนึ่งในการพัฒนาระบบกริ่งที่มีขายกันตามท้องตลาดให้มีประสิทธิภาพมากยิ่งขึ้น

2.1 การควบคุมอุปกรณ์ภายในบ้านจาก Android Application และ Raspberry Pi

เทคโนโลยีในปัจจุบันมีความก้าวหน้ามากขึ้น ตอบสนองต่อความต้องการของผู้ใช้มากขึ้น และมีความเป็นอัตโนมัติมากขึ้นในการทำงาน ทำให้เกิดความคิดเกี่ยวกับระบบอัตโนมัติต่างๆ ภายในบ้านเพียงแค่นี้โทรศัพท์มือถือเครื่องเดียวในการควบคุม ไม่ว่าจะผู้ใช้จะอยู่ส่วนใดของโลกก็ตาม เพียงแค่มืออินเทอร์เน็ต และอุปกรณ์ที่ติดตั้งแอปพลิเคชันสำหรับการควบคุม ผู้ใช้ก็สามารถควบคุมอุปกรณ์ต่างๆภายในบ้านได้ โดยการควบคุมอุปกรณ์ต่างๆ ภายในบ้านจากแอนดรอยด์แอปพลิเคชัน และราสเบอร์รี่พาย เป็นที่นิยมมากในปัจจุบัน ระบบดังกล่าวจะถูกพัฒนาให้ ผู้ใช้ เว็บเซอร์วิส ราสเบอร์รี่พาย และอุปกรณ์ต่างๆ ภายในบ้าน สามารถติดต่อสื่อสารกันได้และสามารถทำงานร่วมกันได้ เช่น การปรับแสง การปรับอุณหภูมิ การเปิดหรือปิดบานหน้าต่าง เป็นต้น

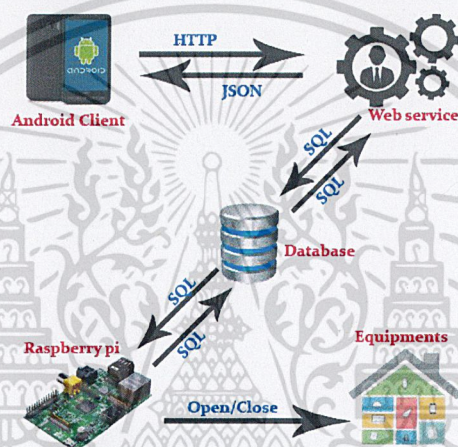
ราสเบอร์รี่พาย เป็นเหมือนคอมพิวเตอร์ขนาดเล็กที่มีขนาดเท่ากับบัตรเครดิต ราคาถูก แต่สามารถทำงานได้เหมือนกับคอมพิวเตอร์เดสก์ท็อปทั่วไป ซึ่งราสเบอร์รี่พายจะรองรับระบบปฏิบัติการ Linux ได้หลายระบบ โดยที่นิยมใช้กันมากที่สุดคือ Raspbian



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ **รูปที่ 2.1** ราสเบอร์รี่พาย ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 1 จะเห็นว่า มีช่องต่างๆ ให้ใช้งานมากมาย ทั้ง GPIO (General Purpose Input Output) เพื่อควบคุมฮาร์ดแวร์ภายนอกด้วยค่า 0 กับ 1 เหมือนกับ Microcontroller ทั่วไป และ Ethernet Port สำหรับเชื่อมต่อกับสาย LAN เพื่อใช้งานอินเทอร์เน็ต และ USB Port สำหรับเชื่อมต่ออุปกรณ์ใช้งานอื่นๆ เช่น เมาส์ แป้นพิมพ์ เป็นต้น และ HDMI สำหรับเชื่อมต่อกับมอนิเตอร์ เพื่อแสดงผล

เนื่องจากความแตกต่างของอุปกรณ์และโครงสร้างการทำงานของอุปกรณ์ ทำให้การติดต่อสื่อสารโดยตรงนั้นเป็นเรื่องยาก โดยวิธีแก้ปัญหาคือจะเสนอเป็นดังรูปต่อไปนี้



รูปที่ 2.2 การแก้ปัญหาโดยภาพรวม

จากรูปที่ 2 เริ่มจาก Android Client ซึ่งเป็นสมาร์ทโฟนหรือแท็บเล็ต ที่ติดตั้งแอปพลิเคชัน สำหรับการควบคุมอุปกรณ์ต่างๆ ภายในบ้านไว้ โดยจะเป็นส่วนที่ติดต่อกับผู้ใช้โดยตรงเพื่อให้ผู้ใช้ดำเนินการส่งคำสั่งไปยังเว็บเซอร์วิส (Web Service) ในรูปแบบ HTTP เพื่อให้เว็บเซอร์วิสใช้คำสั่ง SQL บันทึกคำสั่งที่ผู้ใช้ส่งมาลงในฐานข้อมูล (Database) และทำการตอบสนองกลับไปยังผู้ใช้ว่าการดำเนินการสำเร็จหรือไม่ ในรูปแบบของ JSON ในขณะเดียวกันเว็บเซอร์วิสจะทำการใช้คำสั่ง SQL เพื่อค้นหาคำสั่งใหม่จากผู้ใช้ในฐานข้อมูล ที่ยังไม่ได้ทำการประมวลผล ออกมาประมวลผลกับอุปกรณ์ต่างๆ ภายในบ้าน ตามคำสั่งที่ได้รับมา เพื่อเปิดหรือปิดอุปกรณ์นั้นๆ จากนั้นเว็บเซอร์วิสจะทำการแก้ไขสถานะของคำสั่งที่ประมวลผลแล้วลงในฐานข้อมูลอีกครั้ง เพื่อระบุว่าคำสั่งได้ดำเนินการเสร็จสิ้นแล้ว

ผลการทดลองของการควบคุมอุปกรณ์ต่างๆ ภายในบ้านจากแอนดรอยด์แอปพลิเคชัน และเราเอกสารนี้เว็บเซอร์วิส นั้นประสบความสำเร็จสามารถใช้คำสั่งจากแอนดรอยด์แอปพลิเคชัน เพื่อควบคุมอุปกรณ์ต่างๆ ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่างๆภายในบ้านได้จริง ผู้ใช้หรือผู้พัฒนาสามารถนำไปประยุกต์กับการทำงานของอุปกรณ์ต่างๆภายในบ้านได้อย่างหลากหลาย

2.2 ทฤษฎีที่เกี่ยวข้องกับสตรีมมิ่งมีเดีย

ปัจจุบันเทคโนโลยีทางการสื่อสารมีความเจริญก้าวหน้ามากขึ้นจนการสื่อสารผ่านระบบเครือข่ายนั้นสามารถที่จะเชื่อมโยงถึงกันได้ทั่วทั้งโลกการสื่อสารผ่านเครือข่ายที่ทำได้เพียงส่งข้อความถึงกันจึงไม่เพียงพอกับความต้องการของผู้ใช้งานทำให้ผู้ผลิตหลายรายได้ทำการพัฒนาการส่งข้อมูลให้มีความเร็วมากขึ้นจนสามารถใช้งานมัลติมีเดียผ่านระบบเครือข่ายได้ดังปัจจุบันด้วยการส่งข้อมูลคล้ายการไหลของกระแสน้ำ (Streaming) คือมีลักษณะการส่งข้อมูลผ่านระบบเครือข่ายอย่างต่อเนื่องเหมือนการไหลของกระแสน้ำซึ่งพัฒนาขึ้นเพื่อใช้ในการส่งข้อมูลข่าวสารเพื่อเผยแพร่แพร่ภาพหรือแสดงผลผ่านทางระบบเครือข่ายต่างๆและอินเทอร์เน็ตเรียกสื่อที่มีลักษณะการส่งข้อมูลดังกล่าวว่า “สตรีมมิ่งมีเดีย (Streaming Media)” หรือ “สื่อประสมสายธาร” เว็บไซต์ต่างๆในปัจจุบันส่วนใหญ่มักจะมีส่วนประกอบที่เป็นสตรีมมิ่งมีเดียทั้งสิ้นเช่นการฟังวิทยุการฟังตัวอย่างเพลงการโปรโมทภาพยนตร์การเรียนทางไกลและการค้าขายสินค้าผ่านทางอินเทอร์เน็ต เป็นต้น

2.2.1 ความเป็นมาและความหมายของสตรีมมิ่งมีเดีย

สตรีมมิ่งมีเดียได้พัฒนาสืบเนื่องจากการพัฒนาเว็บเบราว์เซอร์ในปี ค.ศ. 1993 โดยมีการปรับปรุงการใช้งานของเว็บเบราว์เซอร์ให้ง่ายขึ้นเพิ่มความสามารถในการใช้งานทรัพยากรร่วมกันและเชื่อมโยงแหล่งข้อมูลเข้าด้วยกันผู้ใช้งานสามารถเพิ่มรูปแบบที่ต้องการเข้าไปในเว็บไซต์และฟังเสียงผ่านอินเทอร์เน็ตได้แต่ไฟล์เสียงที่แลกเปลี่ยนหรือส่งถึงกันจะมีขนาดใหญ่กว่าไฟล์ข้อความเนื่องจากความสามารถในการส่งผ่านข้อมูลที่ไม่มากเพียงพอและความต้องการสื่อสารข้อมูลที่เพิ่มมากขึ้นเป็นผลให้ผู้ใช้งานต้องรอคอยเป็นระยะเวลาเวลานานเพื่อดาวน์โหลดและส่งไฟล์ถึงกันโดยการรอคอยนี้จะมีผลเฉพาะกับผู้ที่ต้องการรับฟังไฟล์เสียงเนื่องจากไฟล์ดังกล่าวมีขนาดใหญ่ดังนั้นการฟังเสียงความยาวหนึ่งนาทีโดยใช้การเชื่อมต่อความเร็วต่ำ (โมเด็ม 14.4 kbps) จะต้องรอการดาวน์โหลดไฟล์ไม่ต่ำกว่า 5 นาทีส่วนการรับฟังเสียงที่มีคุณภาพระดับเดียวกับซีดีเพลงจะต้องรอประมาณ 2 ชั่วโมงโดยที่ไม่สามารถทำอะไรกับไฟล์ที่กำลังดาวน์โหลดได้เลยจนกว่าการดาวน์โหลดจะเสร็จสิ้นปัญหาหลักที่ทำให้ต้องรอคอยเพื่อฟังข้อมูลเสียงที่ต้องการเกิดจากการทำงานของเครื่องคอมพิวเตอร์ที่ต้องจัดเก็บข้อมูลลงบนฮาร์ดดิสก์ก่อนนำข้อมูลทั้งหมดไปแสดงผลบวกกับโปรโตคอลและเว็บเบราว์เซอร์ที่ใช้งานในช่วงนั้นไม่สนับสนุนการแสดงผลไฟล์เสียงทำให้การแสดงผลไฟล์เสียงนั้นๆต้องกระทำหลังจากเบราว์เซอร์ดาวน์โหลดข้อมูลเสร็จสิ้นแล้วด้วยโปรแกรมซอฟต์แวร์ที่สนับสนุนการเล่นไฟล์เสียงเช่นโปรแกรม Media Player และ AudioPlayer เป็นต้นนอกจากนี้การถ่ายทอดสัญญาณผ่านทางอินเทอร์เน็ตยังต้องอาศัยความรู้ความชำนาญและการเชื่อมต่อในระดับที่สูงกว่าที่เป็นอยู่ทำให้สตรีมมิ่งมีเดียถูก

พัฒนาขึ้นมาเพื่อใช้แก้ปัญหาดังกล่าวต่อมาสตรีมมิ่งมีเดียได้ส่งผลให้เกิดสื่อรูปแบบใหม่บน

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินเทอร์เน็ตโดยเปลี่ยนแปลงจากลักษณะที่ต้องรอการดาวน์โหลดข้อมูลลงสู่เครื่องคอมพิวเตอร์ก่อน แล้วจึงจะแสดงผลได้นั้นพัฒนาไปสู่การแสดงผลไปพร้อมๆกับการส่งผ่านข้อมูลในช่วงเวลาเดียวกัน รวมถึงสนับสนุนให้ผู้ใช้งานสามารถควบคุมการแสดงผลผ่านสตรีมในขณะที่กำลังแสดงผลอยู่ได้ และสามารถถ่ายทอดสัญญาณได้ตรงกับความต้องการของผู้ชม (On-Demand) ซึ่งผู้ชมกลุ่มที่พลาดการถ่ายทอดสดสามารถรับชมการถ่ายทอดนั้นๆได้ในภายหลังสตรีมมิ่งในปัจจุบันจะเป็นการส่งข้อมูลมาในรูปแบบที่บีบอัดแล้วผ่านทางเครือข่ายอินเทอร์เน็ตและแสดงผลไปยังเครื่องคอมพิวเตอร์ปลายทาง โดยที่ผู้ใช้ไม่ต้องรอดาวน์โหลดข้อมูลขนาดใหญ่ทั้งหมดก่อนที่จะรับชมหรือรับฟังแต่สตรีมมิ่งจะดาวน์โหลดข้อมูลเพียงบางส่วนแล้วนำมาแสดงผลได้ทันทีทำให้ผู้ใช้ไม่เกิดความเบื่อหน่ายระหว่างรอชมหรือฟัง

2.2.2 ความแตกต่างระหว่างการดาวน์โหลดและการส่งแบบสตรีมมิ่งมีเดีย

การดาวน์โหลดไฟล์มีลักษณะการส่งที่ไม่เป็นแบบเรียลไทม์เมื่อผู้ชมคลิกบนลิงค์เพื่อดาวน์โหลดไฟล์จะต้องรอนกว่าข้อมูลทั้งหมดถูกส่งไปถึงยังเครื่องของผู้ชมเรียบร้อยแล้วจึงจะสามารถแสดงผลได้โดยในขณะที่ทำการดาวน์โหลดอยู่นั้นจะไม่สามารถทำอะไรกับข้อมูลได้เลยนอกจากการยกเลิกการดาวน์โหลดเท่านั้นและไฟล์ที่ได้จากการดาวน์โหลดนี้จะถูกคัดลอกได้ง่ายทำให้ไม่สามารถป้องกันการละเมิดลิขสิทธิ์ได้แต่ในปัจจุบันสามารถใช้การเข้ารหัสข้อมูลหรือที่เรียกว่า Digital Rights Management (DRM) เข้ามาใช้เพื่อป้องกันการละเมิดลิขสิทธิ์ดังกล่าวได้โดยส่วนใหญ่พบว่า การดาวน์โหลดจะกระทำผ่านทางบราวเซอร์ส่วนสตรีมมิ่งมีเดียมีลักษณะการส่งที่เป็นแบบเรียลไทม์โดยเมื่อผู้ชมเข้าไปเยี่ยมชมเว็บไซต์แล้วคลิกที่ลิงค์เพื่อต้องการชมหรือฟังสตรีมมิ่งมีเดียที่ได้จัดเตรียมไว้เพียงไม่กี่วินาทีต่อมาผู้ชมก็จะได้รับชมหรือรับฟังสตรีมมิ่งมีเดียที่ต้องการโดยไม่ต้องรออีกต่อไป ทั้งนี้เนื่องจากข้อมูลที่ถูกส่งไปยังเครื่องคอมพิวเตอร์ของผู้ชมแบบสตรีมมิ่งจะถูกลบทิ้งไปหลังจากแสดงผล ดังนั้นจึงไม่มีการจัดเก็บข้อมูลลงบนฮาร์ดดิสก์ทำให้สามารถป้องกันการละเมิดลิขสิทธิ์ได้นอกจากนี้ เซิร์ฟเวอร์ที่ให้บริการสตรีมมิ่งมีเดียยังอนุญาตให้ผู้ชมสามารถควบคุมการแสดงผลผ่านสตรีมได้เสมือนกำลังรับชมรับฟังด้วยโปรแกรมแสดงผลทั่วไปเช่นเล่นเพลงก่อนหน้า (Previous) เล่นซ้ำ (Repeat) เป็นต้นแต่ไฟล์ที่ได้จากการดาวน์โหลดจะควบคุมการแสดงผลได้ก็ต่อเมื่อทำการดาวน์โหลดไฟล์ทั้งหมดเสร็จสิ้นแล้วด้วยความสามารถในการโต้ตอบกับผู้ชมขณะรับชมสตรีมมิ่งมีเดียส่งผลให้สตรีมมิ่งมีเดียได้รับความนิยมในปัจจุบันเป็นอย่างมากนอกจากนี้ข้อมูลสตรีมมิ่งมีเดียยังได้รับการป้องกันการละเมิดลิขสิทธิ์ด้วยขั้นตอนการเผยแพร่ที่ซับซ้อนทำให้ไม่สามารถคัดลอกข้อมูลที่กำลังเผยแพร่อยู่ได้สตรีมมิ่งมีเดียจึงเป็นที่นิยมของผู้ผลิตมีเดียอีกด้วยจากที่กล่าวมาสามารถสรุปความแตกต่างของการส่งผ่านไฟล์สตรีมมิ่งมีเดียและการดาวน์โหลดไฟล์ได้ดังตารางนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 เปรียบเทียบการส่งผ่านข้อมูลแบบสตรีมมิ่งมีเดียกับการดาวน์โหลดไฟล์

การส่งผ่านไฟล์สตรีมมิ่งมีเดีย	การดาวน์โหลดไฟล์
1. สามารถแสดงผลแบบเรียลไทม์ได้	1. ไม่สามารถแสดงผลแบบเรียลไทม์ได้
2. ควบคุมการแสดงผลได้ในขณะที่ทำการส่งผ่านข้อมูล	2. ต้องรอดาวน์โหลดไฟล์เสร็จสิ้นก่อนจึงจะแสดงผลได้
3. ป้องกันการละเมิดลิขสิทธิ์ได้อย่างครอบคลุม	3. ต้องทำการป้องกันการละเมิดลิขสิทธิ์ด้วย DRM
4. ไม่มีการจัดเก็บข้อมูลลงบนฮาร์ดดิสก์	4. ต้องจัดเก็บข้อมูลลงบนฮาร์ดดิสก์
5. ขั้นตอนการเผยแพร่ไฟล์ข้อมูลซับซ้อน	5. เตรียมไฟล์สำหรับดาวน์โหลดได้ง่ายมาก

2.2.3 ลักษณะการส่งสตรีมมิ่งมีเดีย

ลักษณะการส่งสตรีมมิ่งมีเดียที่ได้รับความนิยมในปัจจุบันนี้คือโปรแกรมสซีพดาวน์โหลด (Progressive Download) ออนดีมานด์ (On-Demand Files) และการถ่ายทอดสด (Live Broadcasting) ดังรายละเอียดต่อไปนี้

2.2.3.1 โปรแกรมสซีพดาวน์โหลด (Progressive Download)

โปรแกรมสซีพดาวน์โหลดเป็นเทคโนโลยีที่เกิดจากการผสมผสานวิธีการส่งข้อมูลแบบสตรีมและการดาวน์โหลดเข้าด้วยกันวิธีการนี้เป็นการดาวน์โหลดข้อมูลลงบนเครื่องคอมพิวเตอร์ของผู้ชมซึ่งในระหว่างที่ดาวน์โหลดอยู่นั้นผู้ชมสามารถที่จะเล่นหรือแสดงผลไฟล์ได้ก่อนที่การดาวน์โหลดจะเสร็จสิ้นสมบูรณ์เนื่องจากระบบได้มีการนำพื้นที่บางส่วนภายในหน่วยความจำชั่วคราวของเครื่องคอมพิวเตอร์ที่เรียกว่า “บัฟเฟอร์ (Buffer)” มาใช้งานเพื่อเก็บพักข้อมูลแต่วิธีการนี้มักนิยมใช้กับไฟล์มัลติมีเดียที่ไม่ใหญ่มากนักเหมาะสำหรับผู้ที่ต้องการถ่ายทอดและเผยแพร่ไฟล์ข้อมูลที่มีคุณภาพสูงกว่าไฟล์สตรีมมิ่งมีเดียทั่วไปโดยผ่านทางช่องสัญญาณ (Bandwidth) ที่มีขนาดจำกัดเช่นการถ่ายทอดและเผยแพร่ไฟล์ตัวอย่างภาพยนตร์ผู้ชมยินยอมที่จะดาวน์โหลดไฟล์และรอคอยเพื่อที่จะเล่นชมภาพยนตร์คุณภาพสูงที่ตนสนใจ

2.2.3.2 ไฟล์ออนดีมานด์ (On-Demand Files)

ไฟล์ออนดีมานด์เป็นไฟล์ที่สามารถเรียกใช้งานได้ทันทีเมื่อต้องการโดยไฟล์เหล่านี้จะถูกเข้ารหัสในรูปแบบที่เหมาะสมต่อการแสดงผลแบบสตรีมมิ่งแล้วนำไปจัดเก็บไว้บนเซิร์ฟเวอร์เพื่อให้ทุกคนสามารถเรียกใช้งานพร้อมกันได้หลายคนในเวลาเดียวกันโดยแต่ละคนสามารถควบคุมฟังก์ชันการทำงานได้อิสระไม่ว่าจะเป็นหยุดการแสดงผลชั่วคราว (Pause) แสดงผลย้อนกลับ (Rewind) หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาคือเท่านั้น ไม่อนุญาตให้เผยแพร่ไปเชิงพาณิชย์ในการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แม้แต่การแสดงผลซ้ำ(Replay) ซึ่งได้รับความนิยมใช้งานกันอย่างแพร่หลายตัวอย่างเช่นระบบการศึกษาทางไกลผ่านเว็บ (Web-Based Training System : WBTS) โดยที่ผู้เรียนสามารถเข้าถึงข้อมูลหรือบทเรียนได้ตลอดเวลาที่ต้องการผ่านทางเครือข่ายอินเทอร์เน็ต

2.2.3.3 การถ่ายทอดสด (Live Broadcasting)

การถ่ายทอดสดบนอินเทอร์เน็ตเป็นการถ่ายทอดเหตุการณ์ที่เกิดขึ้น ณ ขณะนั้นโดยที่ผู้ชมได้รับชมและฟังเหตุการณ์ต่างๆได้เป็นปัจจุบันและทันท่วงทีด้วยวิธีการแปลงสัญญาณนำเข้าข้อมูลจากกล้องวิดีโอไปเป็นข้อมูลดิจิทัลแล้วส่งผ่านข้อมูลเหล่านี้ในรูปแบบของสตรีมไปยังเครื่องเซิร์ฟเวอร์ซึ่งได้ทำการติดตั้งระบบบริหารจัดการไว้แล้วจากนั้นเครื่องเซิร์ฟเวอร์จะทำการถ่ายทอดสด (Live Broadcast) ไปยังเครื่องของผู้ชมปลายทางได้คราวละพร้อมๆกันเป็นจำนวนมากวิธีการนี้อาจจำเป็นต้องมีเครื่องเซิร์ฟเวอร์สำรองฉุกเฉินเพื่อแบ่งเบาภาระของเครื่องเซิร์ฟเวอร์หลักที่ได้ให้บริการมากเกินไปหรือกรณีที่เครื่องเกิดล่มลงในระหว่างการทำงาน

2.2.4 องค์ประกอบพื้นฐานของระบบสตรีมมิ่งมีเดีย

สำหรับสตรีมมิ่งมีเดียนั้นองค์ประกอบพื้นฐานมีด้วยกัน 3 ส่วนได้แก่ 1) เครื่องเข้ารหัส 2) เครื่องเซิร์ฟและ 3) เครื่องผู้ชมซึ่งทั้ง 3 ส่วนนี้จะมีการติดตั้งซอฟต์แวร์สำหรับใช้ทำงานร่วมกันและแสดงผลสตรีมมิ่งมีเดียเพื่อให้ผู้ชมได้รับชมหรือรับฟังสื่อต่างๆได้ตามความต้องการโดยมีหลักการทำงานเริ่มจากไฟล์สตรีมมิ่งมีเดียที่ส่งมาจากกล้องวิดีโอแล้วนำมาเข้ารหัสด้วยตัวเอ็นโค้ดเดอร์จากนั้นก็ส่งไฟล์สตรีมมิ่งมีเดียที่ได้ทำการเข้ารหัสไว้แล้วส่งผ่านไปยังพื้นที่สำหรับจัดเก็บข้อมูลไว้บนเซิร์ฟเวอร์เพื่อให้เซิร์ฟเวอร์บริหารจัดการข้อมูลเหล่านี้ก่อนที่จะทำการถ่ายทอดหรือเผยแพร่ไปยังเครือข่ายอินเทอร์เน็ตเมื่อได้มีการร้องขอจากเครื่องของผู้ชม

2.2.4.1 เครื่องเข้ารหัส (Encoder)

เป็นเครื่องมัลติมีเดียพีซี (Multimedia PC) ที่ได้ทำการติดตั้งซอฟต์แวร์หรือโปรแกรมไว้สำหรับใช้แปลงไฟล์เสียงและวิดีโอให้อยู่ในรูปแบบของสตรีมมิ่งจากนั้นจึงนำมาทำการเข้ารหัสข้อมูลหรือเรียกกันสั้นๆว่า “เอ็นโค้ด” (Encode) ด้วยการใส่รหัสพร้อมทั้งบีบอัดข้อมูลให้มีขนาดของไฟล์ลดลงแต่ยังคงคุณภาพเดิมให้มากที่สุดเท่าที่จะทำได้เช่นการเข้ารหัสข้อมูลวิดีโอแบบ MPEG หรือ Windows Media File (WMF) เป็นต้น

2.2.4.2 เครื่องเซิร์ฟเวอร์ (Servers)

เป็นเครื่องเซิร์ฟเวอร์ที่ได้ทำการติดตั้งซอฟต์แวร์หรือโปรแกรมสำหรับใช้บริหารจัดการกับสตรีมมิ่งมีเดียที่ได้จากการเข้ารหัสของเครื่องเข้ารหัสเพื่อเตรียมจัดส่งต่อไปยังเครื่องของผู้ชมตามคำร้องขอมาเรียกการทำงานลักษณะนี้ว่า “สตรีมมิ่งเซิร์ฟเวอร์” (Streaming Servers) แม้ว่าหลักการ

การทำงานจะคล้ายคลึงกับกลไกการทำงานของเว็บเซิร์ฟเวอร์ก็ตามแต่ด้วยคุณสมบัติที่แตกต่างกันออกไป

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก็คือสตรีมมิ่งเซิร์ฟเวอร์จะคอยดูแลการติดต่อและเชื่อมต่อของทั้งสองฝั่งตลอดระยะเวลาการทำงาน โดยที่ผู้ชมสามารถที่จะหยุดการเล่นชั่วคราวหรือเลือกเล่นชมในช่วงที่ต้องการได้โดยไม่ติดขัด นอกจากนี้สตรีมมิ่งเซิร์ฟเวอร์ยังสามารถทำงานได้หลายอย่างพร้อมๆกันไม่ว่าจะเป็นการให้บริการแก่เครื่องผู้ชมพร้อมกันหลายๆเครื่องในคราวเดียวการจัดลำดับความสำคัญในการให้บริการกับเครื่องผู้ชมที่ได้ทำการติดต่อและร้องขอมาหรือแม้แต่การกำหนดมาตรการรักษาความปลอดภัยให้กับระบบ ได้อย่างมีประสิทธิภาพอีกด้วย

2.2.4.3 เครื่องผู้ชม (Player)

เป็นเครื่องมัลติมีเดียพีซี (Multimedia PC) ที่ได้ทำการติดตั้งซอฟต์แวร์หรือโปรแกรมสำหรับใช้แสดงผลเพื่อเล่นชมหรือฟังสตรีมมิ่งมีเดียโดยการติดต่อสื่อสารและรับสตรีมมิ่งมีเดียจากสตรีมมิ่งเซิร์ฟเวอร์มาทำการถอดรหัสข้อมูลหรือเรียกกันว่า“ดีโค้ด” (Decode) ก่อนที่จะแสดงผลอีกครั้งโดยที่ผู้ชมสามารถควบคุมการแสดงผลสตรีมมิ่งมีเดียเหล่านี้ได้ไม่ว่าจะเป็นการเล่นการหยุดเล่นการเล่นซ้ำหรือแม้แต่การเล่นในช่วงถัดไป ทั้งนี้ขึ้นอยู่กับขีดความสามารถของโปรแกรมที่เลือกติดตั้งไว้ใช้งาน สำหรับโปรแกรมบางประเภทสามารถจัดเก็บพักข้อมูลสตรีมมิ่งมีเดียไว้ได้รวมไปถึงฟังก์ชันงานที่ช่วยสนับสนุนในการปรับแต่งเสียงและภาพวิดีโอได้ตามความต้องการอีกด้วยตัวอย่างโปรแกรมประเภทนี้ที่นิยมนำมาใช้กันอย่างแพร่หลายเช่นโปรแกรม RealPlayer , Windows Media Player และ QuickTime Player

2.4 โพรโทคอล (Protocol)

คือ กฎ ระเบียบ หรือวิธีการที่ใช้เป็นมาตรฐานสำหรับการติดต่อของคอมพิวเตอร์บนระบบเครือข่าย มีจุดประสงค์เพื่อให้คอมพิวเตอร์สามารถสื่อสารกันได้เข้าใจ เนื่องจากในการเชื่อมโยงเครือข่ายคอมพิวเตอร์นั้น คอมพิวเตอร์แต่ละเครื่องอาจมีความเหมือนหรือต่างกันทางด้านฮาร์ดแวร์ และซอฟต์แวร์ โดยคอมพิวเตอร์แต่ละเครื่องต้องใช้โพรโทคอลเดียวกันจึงจะสามารถติดต่อสื่อสารกันได้เข้าใจ

โพรโทคอลที่ใช้กับเครือข่ายท้องถิ่น (Local Area Network) จะอ้างอิงกับมาตรฐาน IEEE 802 (Institute for Electrical and Electronic Engineers) ตามการกำหนดโดยองค์กรมาตรฐานระหว่างประเทศไอเอสโอ (International Organization for Standardization) ซึ่งสามารถแบ่งระดับการสื่อสารข้อมูลออกเป็น 7 ระดับ (Layer) ดังตาราง 2.1 การสื่อสารในระดับต่างๆ จะอาศัยการควบคุมเพื่อให้ระบบการทำงานนั้น เป็นไปอย่างถูกต้อง มีมาตรฐานโดยการสื่อสารข้อมูลแบบชุดข้อมูล(Packet) จะเกี่ยวข้องกับ 3 ระดับล่างซึ่งได้แก่

(1) ระดับฟิสิคัล (Physical Layer) เป็นระดับที่เกี่ยวข้องกับการรับส่งข้อมูลเป็นบิต (Bit) เกี่ยวข้องกับระดับแรงดันไฟฟ้า (Voltage) ช่วงความถี่ (Frequency) และคาบเวลา (Period)

(2) ระดับดาตalink (Data link Layer) เป็นระดับที่ทำการควบคุมลำดับและอัตราการรับส่งข้อมูล โดยการจัดรูปแบบข้อมูลจากบิตเป็นเฟรม (Frame) พร้อมทั้งมีการตรวจสอบข้อผิดพลาดไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีเทคนิคแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(3) ระดับเน็ตเวิร์ค (Network Layer) ทำการส่งข้อมูลที่เป็นชุดข้อมูล (Packet) ผ่านอุปกรณ์ต่างๆ ไปยังเครือข่ายย่อย อย่างถูกต้องตามต้องการ

ตารางที่ 2.2 แบบจำลองเพื่อการต่อเชื่อมระบบแบบเปิด
(Open Systems Interconnection (OSI) reference model)

Level	Layer	Example
Level 7	Application layer	HTTP, SMTP, SNMP, FTP
Level 6	Presentation layer	XNS
Level 5	Session layer	RPC
Level 4	Transport layer	TCP
Level 3	Network layer	IP
Level 2	Data Link layer	Ethernet Frames
Level 1	Physical layer	Voltages

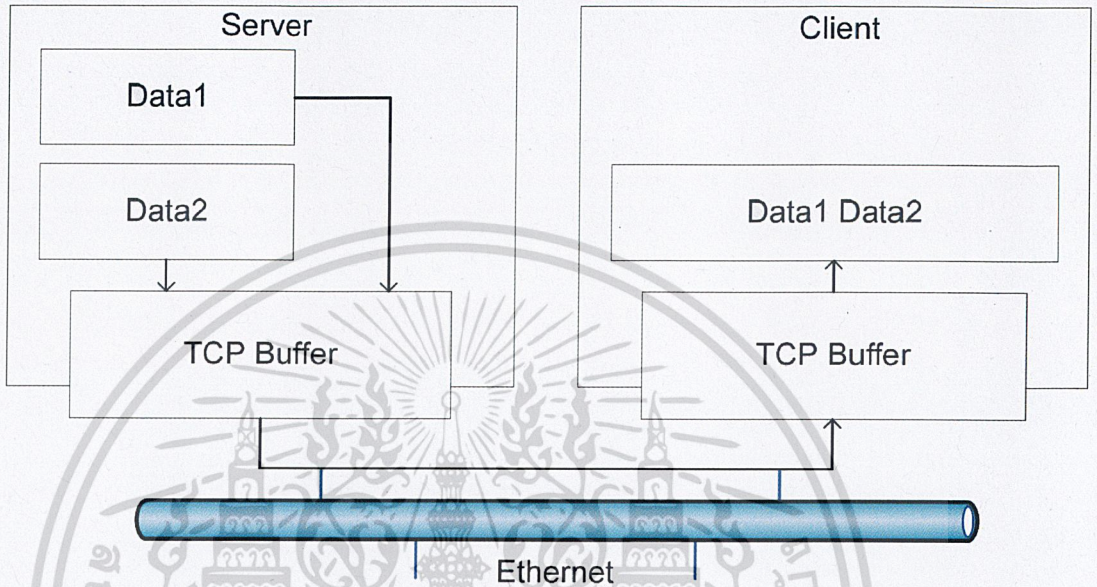
2.4.1 รูปแบบการส่งข้อมูลของโพรโทคอลในระบบเครือข่าย

รูปแบบการส่งข้อมูลของโพรโทคอลในระบบเครือข่าย โดยทั่วไปแบ่งได้ 2 แบบ คือ

- (1) Connection – oriented ต้องมีการสร้างการเชื่อมต่อก่อน จากนั้นจึงทำการส่งข้อมูล พอส่งข้อมูลเสร็จก็ยกเลิกการเชื่อมต่อ ลักษณะเหมือนกับการส่งข้อมูลเสียงทางโทรศัพท์ ต้องมีการสร้างการเชื่อมต่อก่อนโดยการหมุนเบอร์ปลายทางแล้วจึงจะส่งข้อมูลเสียง และเมื่อส่งข้อมูลเสร็จสิ้น ก็จะมีการยุติโดยการวางหูโทรศัพท์ การทำงานของระบบเครือข่ายเอทีเอ็ม (Asynchronous Transfer Mode) และเฟรมรีเลย์ (Frame Relay) ก็เป็นแบบ connection-oriented เช่นกัน ถึงแม้รูปแบบนี้จะค่อนข้างเสียเวลาในตอนเริ่มต้น แต่สามารถป้องกันความผิดพลาดที่เกิดขึ้นในระหว่างการส่งข้อมูล เมื่อได้รับชุดข้อมูลที่ปลายทางแล้ว สามารถเปรียบเทียบค่า Checksum ที่อยู่ในชุดข้อมูล กับ Checksum ที่ได้จากการคำนวณของอัลกอริทึม (Algorithm) เพื่อตรวจสอบว่าข้อมูลผิดพลาดหรือไม่ ถ้าใช่จะส่งชุดข้อมูลนั้นซ้ำอีกครั้ง ดังนั้นโพรโทคอลชนิดนี้จึงต้องมีวิธีการที่ยอมให้เครื่องที่ปลายทางส่งสัญญาณมายังต้นทาง ในกรณีที่ข้อมูลเกิดความผิดพลาดขึ้น เมื่อข้อมูลถูกแบ่งเป็นหลายชุดข้อมูล อาจเดินทางมาถึงปลายทางโดยไม่เรียงตามลำดับของชุดข้อมูล ดังนั้นตัวโพรโทคอลจึงต้องมีความสามารถในการตรวจเช็ค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

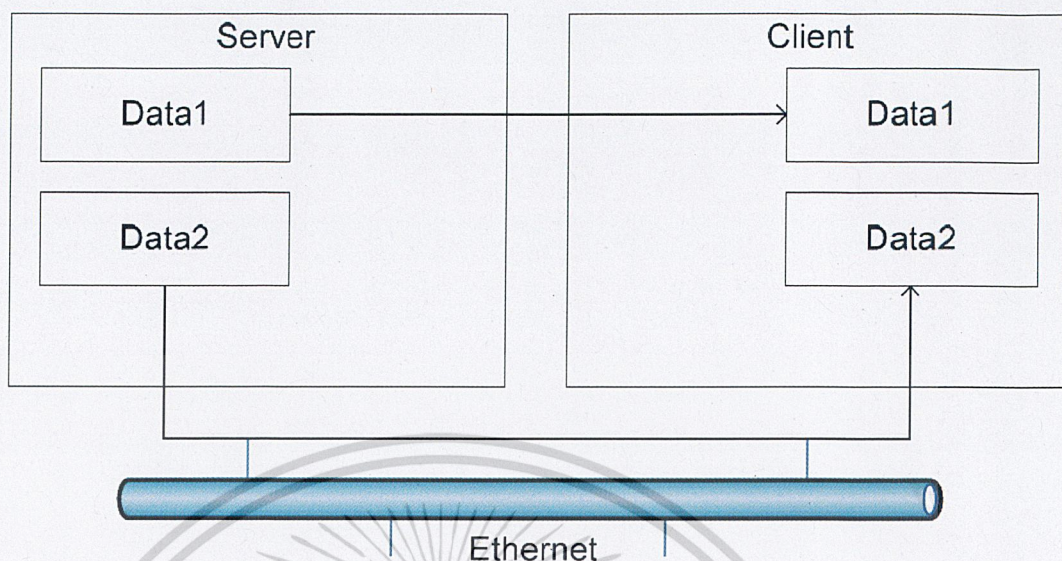
ลำดับของชุดข้อมูล จัดเก็บชุดข้อมูลไว้ในบัฟเฟอร์ (Buffer) และส่งชุดข้อมูลที่เรียงลำดับเรียบร้อยแล้วไปยังระดับแอปพลิเคชันต่อไป อีกทั้งต้องตรวจสอบได้ว่า เมื่อใดมีการส่งข้อมูลซ้ำ และทิ้งข้อมูลซ้ำนั้นไป ในกรณีที่เกิดการผิดพลาดในการสื่อสารโทรทศคอลลอาจ



รูปที่ 2.3 การส่งข้อมูลของโทรทศคอลลแบบ Connection-Oriented

(2) Connectionless ไม่มีการสร้างการเชื่อมต่อ สามารถส่งข้อมูลได้เลย ทำได้รวดเร็ว ไม่เก็บสถานะใดๆของข้อมูลที่ส่งออกไป ข้อมูลแต่ละชิ้นจะเป็นอิสระต่อกัน จะไม่สร้างการเชื่อมต่อผ่านระบบเครือข่ายก่อนที่จะส่งข้อมูล แต่โหนดระหว่างทางแต่ละตัว จะค้นหาเส้นทางในการส่งชุดข้อมูล อย่างเป็นอิสระต่อกัน เมื่อไม่มีการเชื่อมต่อระหว่างต้นทางกับปลายทาง โทรทศคอลลชนิดนี้ จะไม่รับรองความถูกต้องของข้อมูล เมื่อชุดข้อมูลถูกส่งผ่านไปยังระบบเครือข่าย ผู้ส่งจะไม่มีทางทราบว่าชุดข้อมูลไปถึงปลายทาง นอกจากว่าโพรเซสในระดับแอปพลิเคชันจะส่งสัญญาณตอบรับกลับมา การตรวจสอบว่ามีการส่งข้อมูลซ้ำ หรือชุดข้อมูลไม่มาตามลำดับ ก็ไม่สามารถทำได้เช่นกัน แม้ว่าในระบบเครือข่ายยุคใหม่นี้ การสื่อสารแบบนี้จะมีแนวโน้มในการรับส่งข้อมูลผิดพลาดน้อยลงมาก แต่ก็ยังไม่สามารถรับรองความถูกต้องของข้อมูลได้ 100% ดังรูป 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



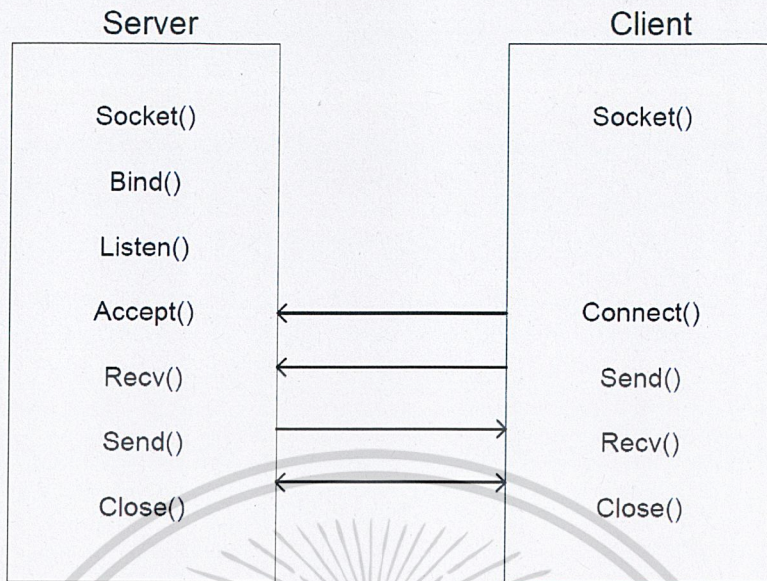
รูป 2.4 การส่งข้อมูลของโพรโทคอลแบบ Connectionless

2.5 ซ็อกเกต (Socket)

ซ็อกเกต คือ การรวมหมายเลขไอพี และหมายเลขพอร์ตเข้าด้วยกัน คำว่า ซ็อกเกต ถูกใช้เปรียบเทียบ เพราะว่าข้อมูลที่บรรจุอยู่ใน ซ็อกเกต สามารถนำไปใช้ได้ซึ่งซ็อกเกตก็คือไอพีแพ็กเก็ตที่อยู่ถัดจากส่วนของไอพีเฮดเดอร์ โดย 16 บิตแรกจะแสดงหมายเลขพอร์ตต้นทาง ส่วน 16 บิตหลังจะเป็นหมายเลขพอร์ตปลายทาง หมายเลขพอร์ตจะช่วยให้เครื่องไอพีโฮสต์สามารถรับส่งข้อมูลหลากหลายชนิด ไปยังไอพีแอดเดรสหนึ่งในเวลาเดียวกันได้หมายเลขไอพีแอดเดรสขนาด 32 บิต และหมายเลขพอร์ตขนาด 16 บิต จะรวมกันเป็นระบบซ็อกเกตที่ใช้กันอยู่ทั่วไปในระบบปฏิบัติการยุคใหม่ ซึ่งซ็อกเกตต้นทาง และซ็อกเกตปลายทาง จะรวมกันเป็นหนึ่งการเชื่อมต่อของทีซีพี จุดสิ้นสุดของเครือข่ายที่เป็นซ็อกเกต เป็นตัวแทนระดับต่ำสุดที่เป็นไปได้ในการสร้างโปรแกรมประยุกต์ ทีซีพี และยูดีพีจะอาศัย ซ็อกเกต เป็นตัวแยกแยะโพรเซสต้นทางและปลายทางที่ติดต่อกันซ็อกเกต มี 3 ชนิด คือ

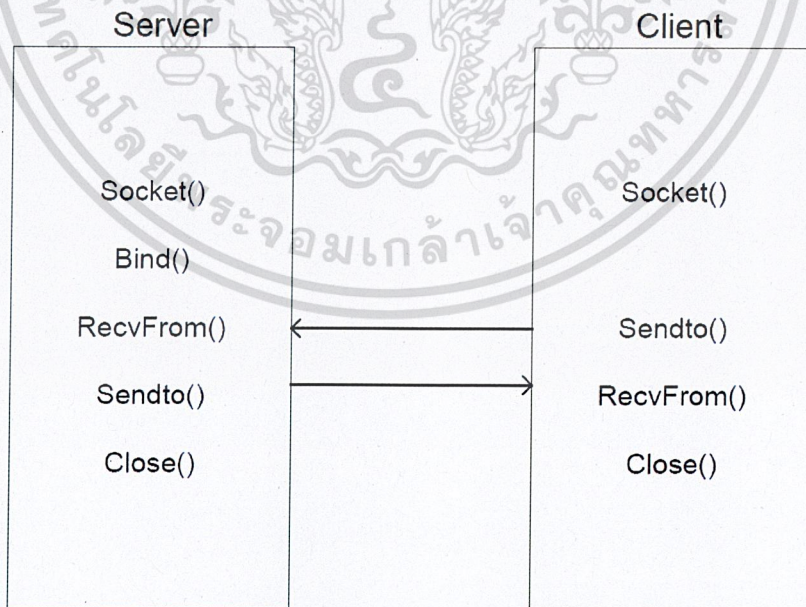
(1) Connection-Oriented Socket การเชื่อมต่อแบบต่อเนื่องที่อนุญาตให้โพรเซสเชื่อมต่อ กับโพรเซสระยะไกล โดยใช้โพรโทคอล ทีซีพี ดังนั้นวิธีนี้จึงทำให้การส่งข้อมูล เชื่อถือได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.5 การเชื่อมต่อของซ็อกเกตแบบ Connection-Oriented

(2) Connectionless Socket การเชื่อมต่อแบบไม่ต่อเนื่องนำมาใช้ในการส่งแบบสั้น ๆ ละ ไม่สำคัญเท่าใดนัก ไม่รับประกันว่าข้อมูลที่ส่งออกไปนั้นจะถูกต้อง เช่น ยูดีพี เมื่อส่งข้อมูลไปแล้ว ไม่สามารถรับรองการได้รับข้อมูลของปลายทาง แต่อย่างไรก็ตามยังคงต้องการ หมายเลขไอพีและ หมายเลขพอร์ตในการระบุปลายทาง



รูป 2.6 การเชื่อมต่อของซ็อกเกตแบบ Connectionless

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(3) Raw Socket เป็นซ็อกเกตที่อนุญาตให้จัดการข้อมูลส่วนไอพีแอดเดรส

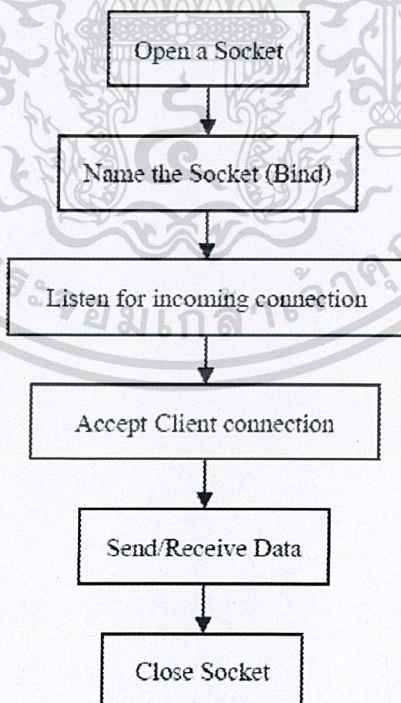
2.5.1 Working with Sockets in Java

ใน Java การทำงานที่เกี่ยวข้องกับ Socket จะต้อง Import Class ที่ชื่อว่า `java.net.Socket` ซึ่งเป็น Class ที่สนับสนุนการทำงานกับ Socket โดย Socket จะแบ่งการทำงานเป็น 2 ลักษณะคือ

(1) Passive Socket หรือ Socket ประเภทที่ใช้งานใน Server Application ซึ่งจะต้อง Import Class ที่ชื่อว่า `java.net.ServerSocket` ใน Class จะประกอบไปด้วย Method ที่สำคัญดังต่อไปนี้

- 1.1 `accept`
- 1.2 `close`
- 1.3 `bind`
- 1.4 `setSoTimeOut`
- 1.5 `setReceiveBufferSize`
- 1.6 `setPerformancePreferences`

ซึ่งมีขั้นตอนในการสร้างดังนี้

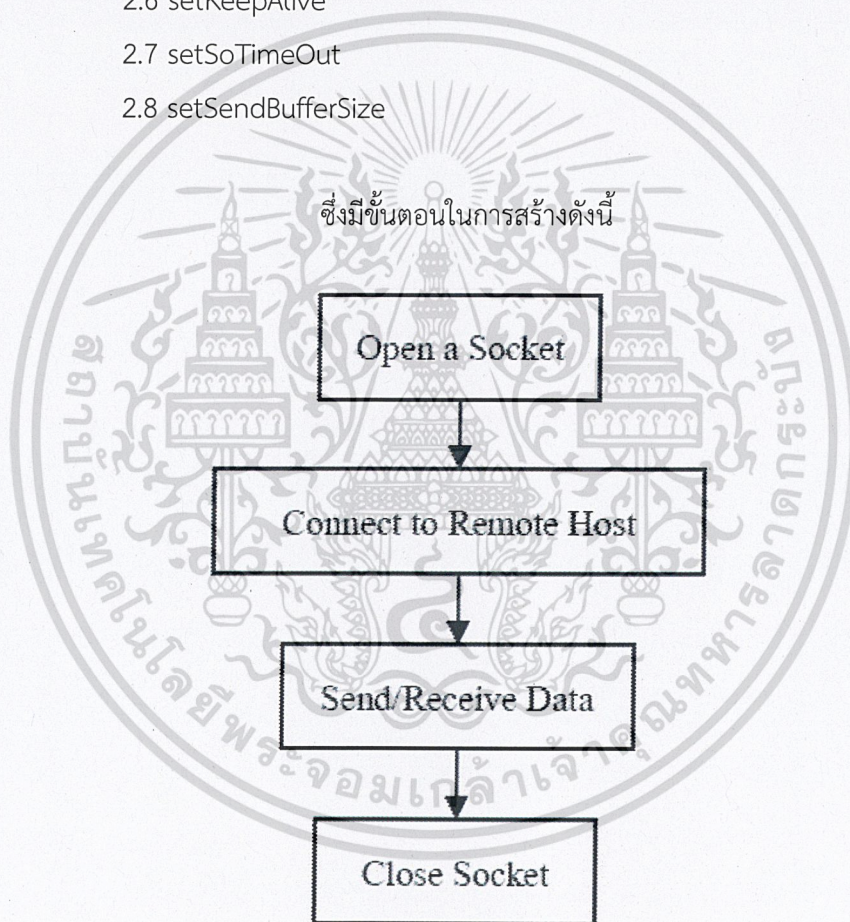


รูป 2.7 Diagram ขั้นตอนการสร้าง Passive Sockets

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการเรียนการสอนเพื่อใช้ประโยชน์ในการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(2) Active Socket หรือ Socket ประเภทที่ใช้งานใน Client Application ซึ่งจะต้อง Import Class ที่ชื่อว่า `java.net.Socket` ใน Class จะประกอบไปด้วย Method ที่สำคัญดังต่อไปนี้

- 2.1 bind
- 2.2 close
- 2.3 getInputStream
- 2.4 getOutputStream
- 2.5 connect
- 2.6 setKeepAlive
- 2.7 setSoTimeOut
- 2.8 setSendBufferSize



รูป 2.8 Diagram ขั้นตอนการสร้าง Active Sockets

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

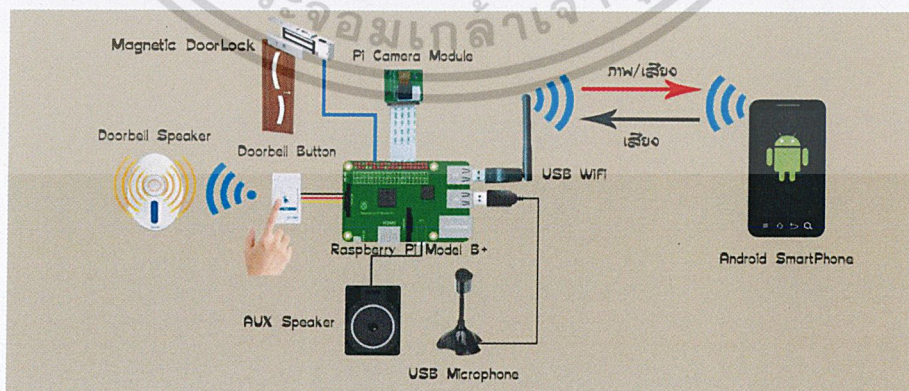
บทที่ 3

ขั้นตอนการออกแบบและพัฒนาซอฟต์แวร์

ในบทนี้จะเป็นการกล่าวถึงการวิเคราะห์และการออกแบบระบบโครงงาน Doorbell Intelligent System หรือกริ่งอัจฉริยะ โดยคณะผู้จัดทำได้ศึกษาทฤษฎีการสตรึมมิ่งมีเดียและหลักการงานที่เกี่ยวกับการควบคุมอุปกรณ์ภายในบ้านผ่านอุปกรณ์ประเภท Embedded Boards จากการศึกษาทำให้คณะผู้จัดทำเกิดแนวคิดที่จะพัฒนาอุปกรณ์รักษาความปลอดภัยเพื่อผู้อยู่อาศัยในครัวเรือนมาประยุกต์ใช้ร่วมกับ SmartPhone โดยมีเป้าหมายในการพัฒนากริ่งที่มีอยู่ในปัจจุบันให้เกิดประสิทธิภาพสูงสุด และสามารถตอบโจทย์ผู้ใช้งานด้านความปลอดภัยได้อย่างแท้จริง โดยนำเอาทฤษฎีการสตรึมมิ่งมีเดียมาเป็นเครื่องมือในการรับชมภาพวิดีโอที่ถ่ายให้เห็นใบหน้าผู้ที่มาดกริ่ง และนำเอาหลักการงานที่เกี่ยวกับการควบคุมอุปกรณ์ภายในบ้านผ่านอุปกรณ์ประเภท Embedded Boards กับหลักการ Network Programming เพื่อเชื่อมต่อกริ่งเข้ากับ SmartPhone ที่มีระบบปฏิบัติการ Android เพื่อรับชมวิดีโอที่สตรึมมิ่งอยู่

3.1 การออกแบบฮาร์ดแวร์

ขั้นตอนการเชื่อมต่ออุปกรณ์ภายในระบบจะเริ่มจากการนำกริ่งที่เป็นระบบไร้สายมาเชื่อมกับ Raspberry Pi โดยจะทำการต่อจากปุ่มกดของกริ่งมายัง Raspberry Pi ผ่านจุดเชื่อมต่อ GPIO และเชื่อมต่อ Pi Camera Module ผ่านจุดต่อกล้องแบบ CSI ในส่วน AUX Speaker จะเชื่อมต่อผ่านช่องเสียบ jack ขนาด 3.5mm ที่เป็นส่วน output ที่ Raspberry Pi มีมาให้ในส่วน Microphone/WiFi จะเชื่อมต่อกับ Raspberry Pi ผ่านทาง USB port และมีการเชื่อมต่อระบบบล็อกประตูเข้ากับบอร์ด Raspberry Pi ดังรูปที่ 3.1



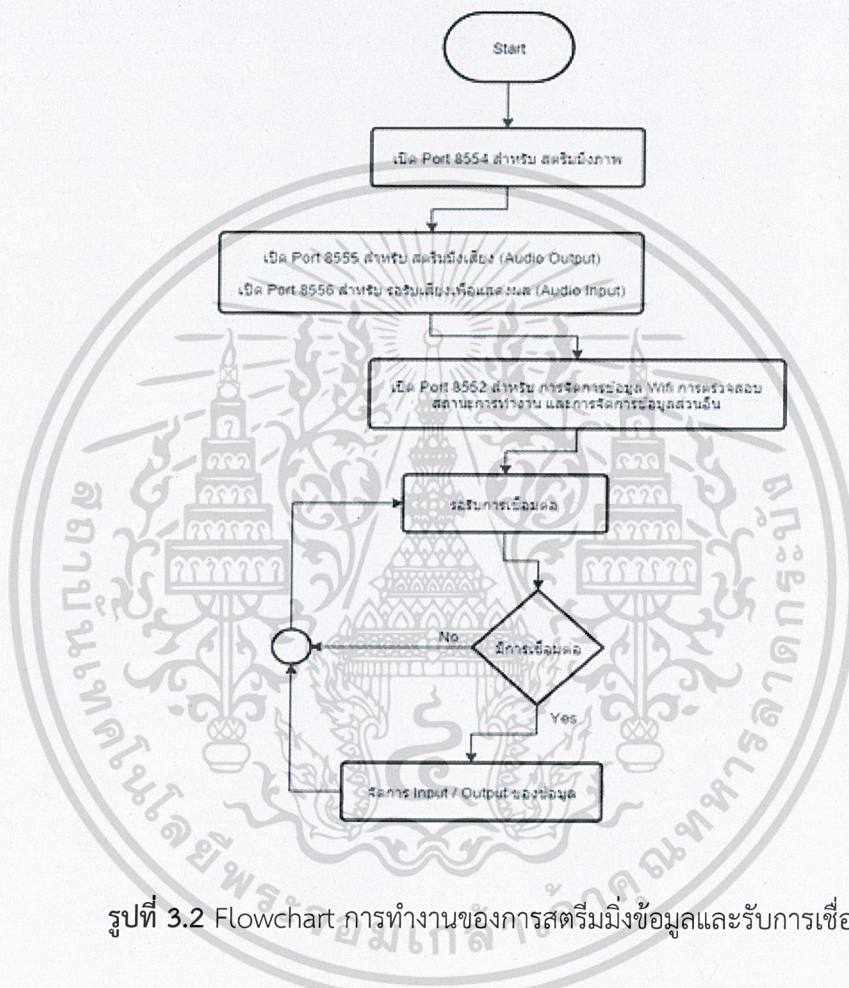
รูปที่ 3.1 การเชื่อมต่ออุปกรณ์ภายในระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ขั้นตอนการทำงานของ Application บน Raspberry Pi

ขั้นตอนการทำงานของ Application บน Raspberry Pi นั้นจะแบ่งออกเป็น 3 ฟังก์ชันการทำงานใหญ่ๆได้ดังนี้

3.2.1 ฟังก์ชันสตรีมมิ่งข้อมูลและรอรับการเชื่อมต่อ

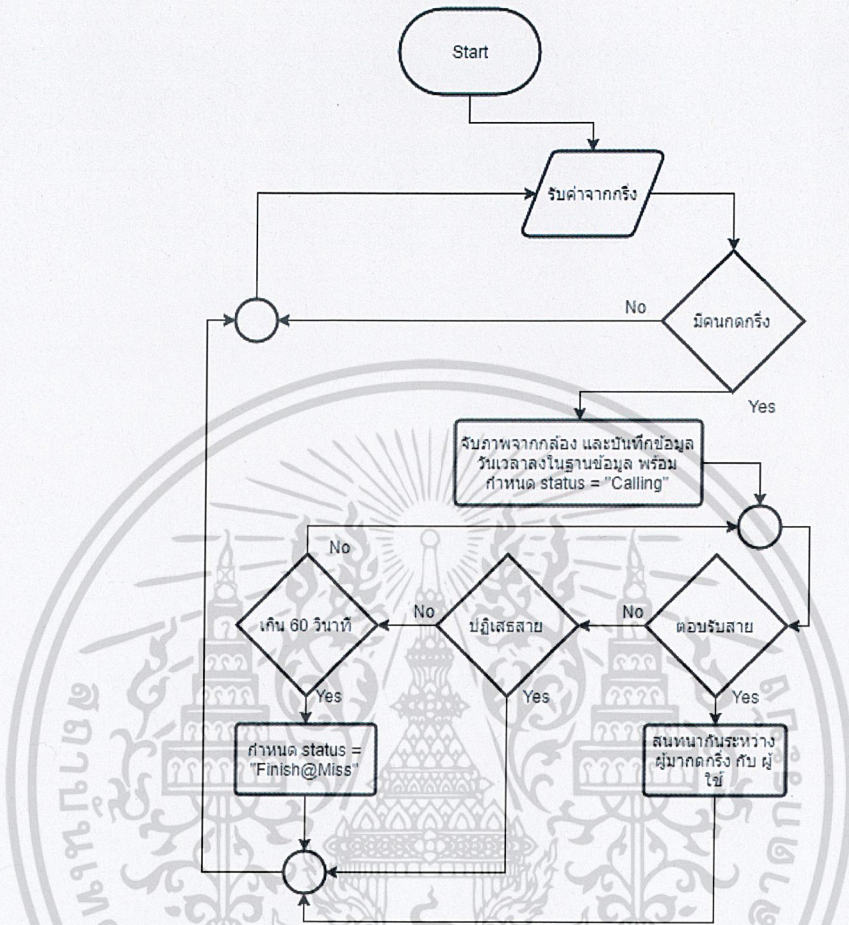


รูปที่ 3.2 Flowchart การทำงานของการสตรีมมิ่งข้อมูลและรับการเชื่อมต่อ

จากรูปที่ 3.2 อธิบายการทำงานของ Raspberry pi เพื่อทำการเปิด Service ให้ Client สามารถเชื่อมต่อหรือร้องขอข้อมูลต่างๆ ได้ โดยจะทำการเปิด Port 8554 เพื่อทำการสตรีมมิ่งภาพจากกล้องของ Raspberry pi ทำการเปิด Port 8555 เพื่อทำการสตรีมมิ่งเสียงจากไมโครโฟนของ Raspberry pi ทำการเปิด Port 8556 เพื่อทำการรอรับเสียงจาก Client เพื่อมาแสดงผลออกทางลำโพงของ Raspberry pi และทำการเปิด Port 8552 เพื่อทำการรอรับข้อมูลสำหรับการจัดการตั้งค่าต่างๆ บนเครื่อง Raspberry pi ไม่ว่าจะเป็นในส่วนของการเชื่อมต่อเน็ตเวิร์ค การตรวจสอบสถานะการออนไลน์ของ Raspberry การจัดการรหัสผ่าน และการเปิดประตูผ่านแอปพลิเคชัน เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 ฟังก์ชันบริการกระทำจากการกดกริ่งบน Raspberry Pi

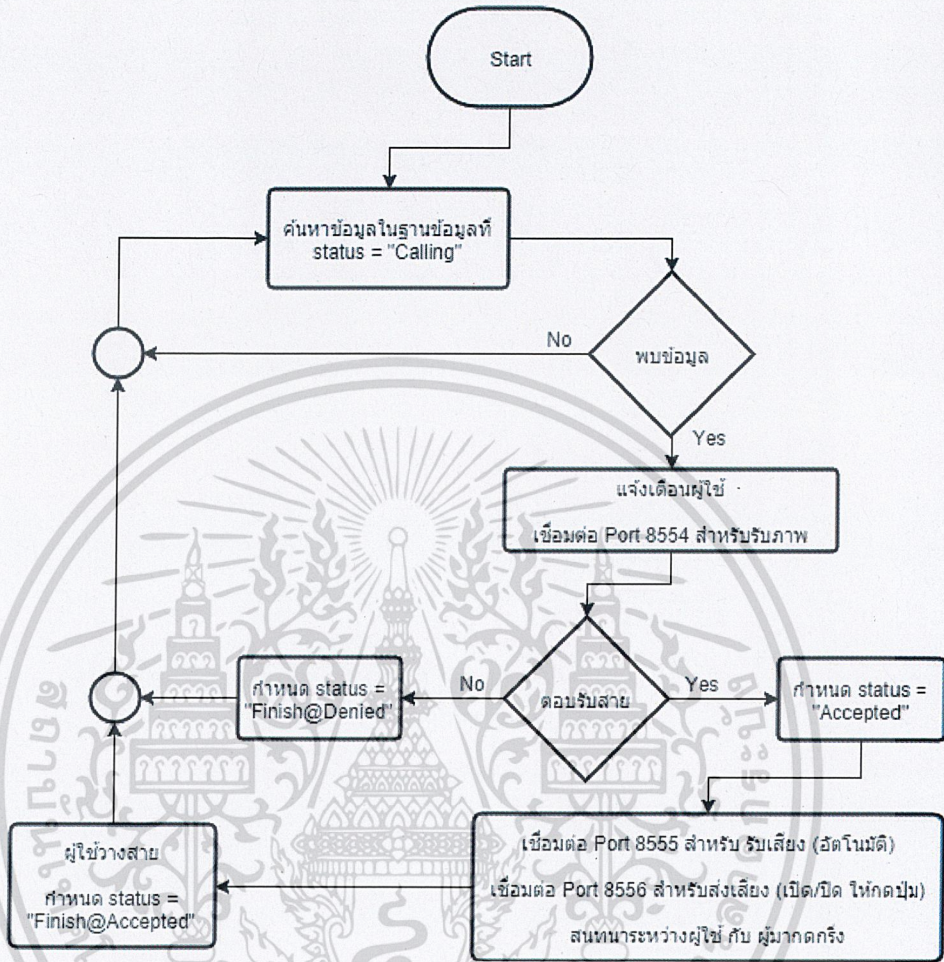


รูปที่ 3.3 Flowchart การทำงานเมื่อมีการกดกริ่งและส่งสัญญาณให้ Raspberry Pi

จากรูปที่ 3.3 อธิบายการทำงานของ Raspberry pi โดยจะเริ่มทำการรอรับค่า เมื่อมีคนมากดกริ่งจะทำการจับภาพจากกล้อง และบันทึกข้อมูลวันเวลาลงในฐานข้อมูลพร้อมกับกำหนดให้ฟิลด์ status = "Calling" จากนั้น Raspberry pi จะทำการรอเพื่อให้ผู้ใช้ Application ของกริ่งอัจฉริยะตัวนั้นรับสายภายใน 60 วินาที หากผู้ใช้ปฏิเสธสาย ระบบจะกลับไปรอรับค่าจากกริ่งใหม่ หากผู้ใช้ไม่มีการตอบสนองนานเกิน 60 วินาที จะกำหนดฟิลด์ status = "Finish@Miss" หากผู้ใช้ตอบรับสายระบบจะทำการเชื่อมต่อเสียง เพื่อให้ผู้ใช้และคนกดกริ่งสามารถสนทนากันได้ เมื่อผู้ใช้วางสาย ระบบจะกลับไปรอรับค่าจากกริ่ง เพื่อขึ้นรอบการทำงานใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 ฟังก์ชันการรับข้อมูลเสียงมาแสดงผลบน Raspberry Pi



รูปที่ 3.4 Flowchart การทำงานเมื่อมีการส่งข้อมูลเสียงมายัง Raspberry Pi

จากรูปที่ 3.4 อธิบายการทำงานของ Android Application ระบบจะทำการค้นหาข้อมูลในฐานข้อมูล เพื่อหาฟิลด์ status = "Calling" หากพบข้อมูลจะทำการแจ้งเตือนผู้ใช้ และแสดงภาพว่าใครที่กดกริ่งอยู่ที่หน้าประตูบ้าน โดยผู้ใช้สามารถ ยอมรับ/ปฏิเสธ สายได้ หากผู้ใช้ปฏิเสธระบบจะกำหนดฟิลด์ status = "Finish@Denied" และกลับไปค้นหาข้อมูลใหม่ หากผู้ใช้อยอมรับสาย ระบบจะทำการกำหนดฟิลด์ status = "Accepted" และเชื่อมต่อเสียงจากตัวอุปกรณ์ Doorbell ให้อัตโนมัติ เพื่อให้ผู้ใช้ได้ยินเสียงจากคนมาดครง และผู้ใช้สามารถกดปุ่ม เปิด/ปิด ไมโครโฟน เพื่อให้ผู้ใช้สามารถตอบเสียงกลับได้ เมื่อผู้ใชวางสายระบบจะกำหนดฟิลด์ status = "Finish@Accepted" และกลับไปค้นหาข้อมูลใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 ตารางฐานข้อมูล

ฐานข้อมูลสำหรับบันทึกการใช้งานของระบบกริ่งอัจฉริยะ ประกอบไปด้วย รหัสรายการ วันเวลา ชื่อผู้ตอบรับ สถานะ ดังแสดงได้ในตารางที่ 3.1

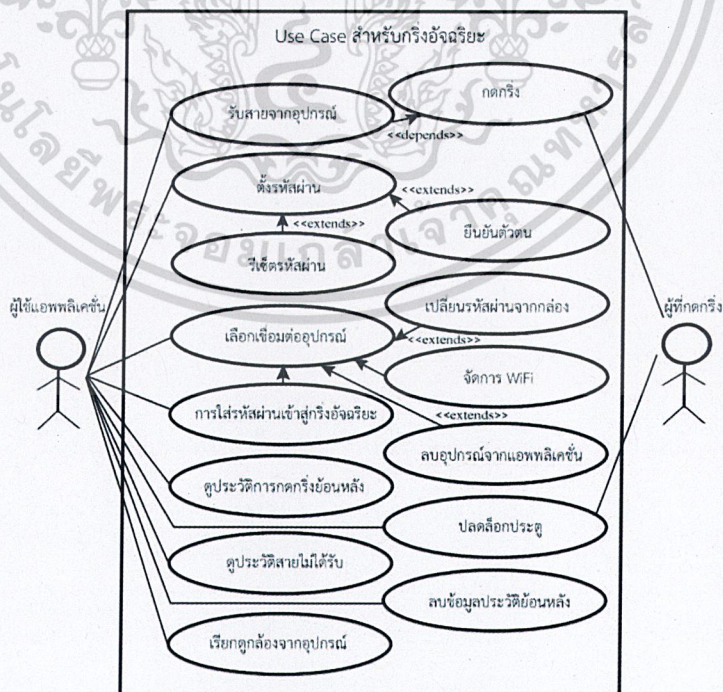
ตารางที่ 3.1 ตารางบันทึกการใช้งานอุปกรณ์กริ่งอัจฉริยะ

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย	คีย์
time	DATETIME	วันเวลาในการทำรายการ	PK
user_accept	VARCHAR	ชื่อผู้ตอบรับรายการ	
status	VARCHAR	สถานะของรายการ	

3.3 ขั้นตอนการทำงานของ Application บน Android Smartphone

Application ที่ถูกพัฒนาขึ้นบนระบบปฏิบัติการ Android จะสามารถแสดงผลภาพและเสียงที่ถูกสตรีมอยู่บนอุปกรณ์และมีฟังก์ชันการทำงานอื่นๆ ที่แสดงอยู่ในยูสเคสไดอะแกรมดังรูปที่

3.5



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.5 แผนภาพ Use Case Diagram
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 คำอธิบายยูสเคส กดกริ่ง

Use case name	กดกริ่ง	
Scenario	กดปุ่มกริ่งเพื่อส่งสายเรียกเข้าภาพและเสียงกริ่งไปหาผู้รับ	
Triggering Event	ต้องการกดกริ่งเพื่อเรียกผู้ที่อยู่ในบ้าน	
Brief Description	กดกริ่ง และทำการถ่ายภาพ 5รูป จากนั้น ส่งสายเรียกเข้าไปหาผู้ใช้แอปพลิเคชัน	
Actors	ผู้ที่มากดกริ่ง	
Related Use Cases	-	
Stakeholders	ผู้ใช้งาน	
Preconditions	-	
Postconditions	เสียงกริ่งดัง และ ถ่ายภาพ 5 รูปเป็นเวลา 5วินาที จากนั้นส่งสายเรียกเข้าไปหาผู้ใช้แอปพลิเคชัน	
Flow of Events	Actor	System
	<ol style="list-style-type: none"> 1. กดกริ่ง 2. ถ่ายรูป 5ภาพ 5 วินาที 	<ol style="list-style-type: none"> 2. เสียงกริ่งดัง 3. ส่งสายเรียกเข้าไปยังผู้ใช้แอปพลิเคชัน
Exception Condition	-	

ตารางที่ 3.3 คำอธิบายยูสเคส รับสายจากอุปกรณ์

Use case name	รับสายจากอุปกรณ์	
Scenario	รับสายหรือเฟกเฉยจากอุปกรณ์	
Triggering Event	ต้องการรับสายจากผู้กดกริ่ง หรือ เฟกเฉย	
Brief Description	มีสายเรียกเข้ามายัง Smart Phone	
Actors	ผู้ใช้งานแอปพลิเคชัน	
Related Use Cases	กดกริ่ง	
Stakeholders	ผู้ใช้งาน	
Preconditions	มีการกดกริ่งจากอุปกรณ์ก่อน	
Postconditions	เสียงกริ่งดัง และ สายเรียกเข้าไปยังผู้ใช้	
Flow of Events	Actor	System
	1. กดรับสาย หรือ เฟกเฉย	ที่นั่น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

		2. ส่งสายเรียกเข้าไปยังผู้ใช้แอปพลิเคชัน
Exception Condition	-	

ตารางที่ 3.4 คำอธิบายยูสเคส ตั้งรหัสผ่าน

Use case name	ตั้งรหัสผ่าน	
Scenario	ตั้งรหัสผ่านสำหรับการเข้าใช้ครั้งแรก	
Triggering Event	เข้าใช้งานครั้งแรก	
Brief Description	เมื่อใช้งานครั้งแรกรหัสผ่านจะถูกตั้งโดยอัตโนมัติ	
Actors	ผู้ใช้งาน	
Related Use Cases	-	
Stakeholders	ผู้ใช้งาน	
Preconditions	-	
Postconditions	รหัสผ่านจะถูกเซต	
Flow of Events	Actor	System
	1. เข้ามายังแอปพลิเคชัน 2. ตั้งรหัสผ่าน	
Exception Condition	ตั้งรหัสผ่านเป็นตัวอักษร	

ตารางที่ 3.5 คำอธิบายยูสเคส ยืนยันตัวตน

Use case name	ยืนยันตัวตน	
Scenario	ทำการยืนยันตัวตนจากรหัสผ่านที่ตั้งมา	
Triggering Event	เข้าใช้งานระบบ	
Brief Description	ยืนยันตัวตนโดยใช้รหัสผ่านที่ตั้งมาครั้งแรกที่ใช้งาน	
Actors	ผู้ใช้งาน	
Related Use Cases	ตั้งรหัสผ่าน	
Stakeholders	ผู้ใช้งาน	
Preconditions	ตั้งรหัสผ่านก่อนจึงจะยืนยันตัวตนได้	
Postconditions	เข้าใช้งานระบบ	
Flow of Events	Actor	System

	1. ใส่รหัสผ่านยืนยันตัวตน	2. ตรวจสอบรหัสผ่าน เข้าใช้งานระบบ
Exception Condition	ใส่รหัสผ่านผิด	

ตารางที่ 3.6 คำอธิบายยูสเคส รีเซ็ตรหัสผ่าน

Use case name	รีเซ็ตรหัสผ่าน	
Scenario	ผู้ใช้งานต้องการตั้งรหัสผ่านใหม่	
Triggering Event	ต้องการรีเซ็ตรหัสผ่านหรือลืมรหัสผ่าน	
Brief Description	รีเซ็ตรหัสผ่านใหม่เมื่อลืมรหัสหรือต้องการล้างข้อมูล	
Actors	ผู้ใช้งาน	
Related Use Cases	ตั้งรหัสผ่าน	
Stakeholders	ผู้ใช้งาน	
Preconditions	ต้องตั้งรหัสผ่านก่อน	
Postconditions	ระบบทั้งหมดจะถูกล้างข้อมูลเพื่อตั้งรหัสผ่านใหม่	
Flow of Events	Actor	System
	1. เข้าใช้งานเมนูรีเซ็ตรหัสผ่าน	2. ล้างข้อมูลทั้งหมดที่เคยตั้งค่า
Exception Condition	-	

ตารางที่ 3.7 คำอธิบายยูสเคส เลือกเชื่อมต่ออุปกรณ์

Use case name	เลือกเชื่อมต่ออุปกรณ์
Scenario	เชื่อมต่อเครื่องอุปกรณ์เข้ากับ Smart Phone และ ทำการ Login
Triggering Event	ต้องการเชื่อมต่อเครื่องอุปกรณ์เข้ากับ Smart Phone
Brief Description	เชื่อมต่ออุปกรณ์เข้ากับ Smart Phone
Actors	ผู้ใช้งาน
Related Use Cases	-
Stakeholders	ผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ต่อสาธารณะและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Preconditions	ต้องการยืนยันตัวตนก่อน	
Postconditions	แอปพลิเคชันบันทึกอุปกรณ์ที่เชื่อมต่อ	
Flow of Events	Actor	System
	1. เข้าสู่ระบบ 2. เลือกเชื่อมต่ออุปกรณ์ 3. Login เข้าสู่อุปกรณ์	4. แอปพลิเคชันบันทึกอุปกรณ์ที่เชื่อมต่อ
Exception Condition	-	

ตารางที่ 3.8 คำอธิบายยูสเคส เปลี่ยนรหัสผ่านจากกล้อง

Use case name	เปลี่ยนรหัสผ่านจากกล้อง	
Scenario	เปลี่ยนรหัสผ่านจากเดิมต้องการเพิ่มความปลอดภัย	
Triggering Event	ต้องการเปลี่ยนรหัสผ่านให้เราทำได้	
Brief Description	เปลี่ยนรหัสผ่านที่ใช้ login เข้าใช้งานอุปกรณ์	
Actors	ผู้ใช้งาน	
Related Use Cases	เลือกเชื่อมต่ออุปกรณ์	
Stakeholders	ผู้ใช้งาน	
Preconditions	ต้อง login และเชื่อมต่ออุปกรณ์เข้ากับแอปพลิเคชัน	
Postconditions	รหัสผ่านจะถูกเปลี่ยนแปลง	
Flow of Events	Actor	System
	1. เข้าสู่ระบบ 2. Login เข้าสู่อุปกรณ์ 3. เปลี่ยนรหัสผ่าน	4. บันทึกรหัสผ่านใหม่
Exception Condition	-	

ตารางที่ 3.9 คำอธิบายยูสเคส จัดการ WiFi

Use case name	จัดการ WiFi
Scenario	เปลี่ยน WiFi ที่เชื่อมต่อกับกล้อง
Triggering Event	ต้องการเปลี่ยน WiFi ที่อยู่ใกล้ที่สุดกับอุปกรณ์
Brief Description	เปลี่ยน WiFi ในการเชื่อมต่อกับอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านอื่น
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Actors	ผู้ใช้งาน	
Related Use Cases	เลือกเชื่อมต่ออุปกรณ์	
Stakeholders	ผู้ใช้งาน	
Preconditions	ต้อง login และเชื่อมต่ออุปกรณ์เข้ากับแอปพลิเคชัน	
Postconditions	เสียงกริ่งดัง และ สายเรียกเข้าไปยังผู้ใช้	
Flow of Events	Actor	System
	1. เข้าสู่ระบบ 2. Login เข้าสู่อุปกรณ์ 3. เปลี่ยนแปลง WiFi	4. บันทึก WiFi ที่เปลี่ยนแปลงใหม่
Exception Condition	ใส่รหัสผ่านใหม่ 2 ครั้งไม่ตรงกัน	

ตารางที่ 3.10 คำอธิบายยูสเคส ลบอุปกรณ์จากแอปพลิเคชัน

Use case name	ลบอุปกรณ์จากแอปพลิเคชัน	
Scenario	ลบอุปกรณ์ที่เชื่อมต่ออยู่กับแอปพลิเคชัน	
Triggering Event	ต้องการลบอุปกรณ์ออกจากแอปพลิเคชันหรือ เปลี่ยนอุปกรณ์เชื่อมต่อ	
Brief Description	ลบและเปลี่ยนอุปกรณ์เชื่อมต่อใหม่	
Actors	ผู้ใช้งาน	
Related Use Cases	เลือกเชื่อมต่ออุปกรณ์	
Stakeholders	ผู้ใช้งาน	
Preconditions	ต้อง login และเชื่อมต่ออุปกรณ์เข้ากับแอปพลิเคชัน	
Postconditions	ลบอุปกรณ์ที่เชื่อมต่ออยู่กับแอปพลิเคชัน	
Flow of Events	Actor	System
	1. เข้าสู่ระบบ 2. Login เข้าสู่อุปกรณ์ 3. เลือกลบอุปกรณ์ที่เชื่อมต่อ	4. ลบอุปกรณ์ที่เชื่อมต่อกับแอปพลิเคชัน
Exception Condition	ใส่รหัสผ่านใหม่ 2 ครั้งไม่ตรงกัน	

ตารางที่ 3.11 การใส่รหัสผ่านเข้าสู่ระบบกริ่งอัจฉริยะ

Use case name	การใส่รหัสผ่านเข้าสู่ระบบกริ่งอัจฉริยะ
---------------	--

Scenario	ผู้ใช้งานต้องการยืนยันตัวตนเพื่อเชื่อมต่อเข้ากับเครื่องอัจฉริยะ	
Triggering Event	ต้องการเชื่อมต่อแอปพลิเคชันกับเครื่องอัจฉริยะ	
Brief Description	เชื่อมแอปพลิเคชันต่อกับเครื่องอัจฉริยะ โดยการใช้ IP และ password	
Actors	ผู้ใช้งาน	
Related Use Cases	-	
Stakeholders	ผู้ใช้งาน	
Preconditions	เลือกเชื่อมต่ออุปกรณ์	
Postconditions	เชื่อมแอปพลิเคชันต่อกับเครื่องอัจฉริยะ	
Flow of Events	Actor	System
	<ol style="list-style-type: none"> เข้าสู่ระบบ เลือกเชื่อมต่ออุปกรณ์ 	<ol style="list-style-type: none"> ใส่ IP และ password เชื่อมแอปพลิเคชันต่อกับเครื่องอัจฉริยะ
Exception Condition	-	

ตารางที่ 3.12 คำอธิบายยูสเคส ดูประวัติการกดกริ่งย้อนหลัง

Use case name	ดูประวัติการกดกริ่งย้อนหลัง	
Scenario	ผู้ใช้งานต้องการเรียกดูประวัติมีใครมากดกริ่งจากอุปกรณ์บ้าง	
Triggering Event	ต้องการเรียกดูประวัติย้อนหลัง	
Brief Description	เรียกดู วันที่ เวลา ภาพ จากผู้ที่มากดกริ่งจากอุปกรณ์	
Actors	ผู้ใช้งาน	
Related Use Cases	-	
Stakeholders	ผู้ใช้งาน	
Preconditions	ต้องการยืนยันตัวตน	
Postconditions	แสดงประวัติจากการกดกริ่งไปยัง Smart Phone	
Flow of Events	Actor	System
	<ol style="list-style-type: none"> เข้าสู่ระบบ ดูประวัติการกดกริ่งย้อนหลัง 	<ol style="list-style-type: none"> แสดงประวัติจากการกดกริ่งไปยัง Smart Phone
Exception Condition	-	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.13 คำอธิบายยูสเคส ดูประวัติสายไม่ได้รับ

Use case name	ดูประวัติสายไม่ได้รับ	
Scenario	ผู้ใช้งานต้องการเรียกดูประวัติสายไม่ได้รับจากอุปกรณ์	
Triggering Event	ต้องการเรียกดูประวัติสายไม่ได้รับ	
Brief Description	เรียกดู วันที่ เวลา ภาพ จากผู้ที่มากดกริ่งจากอุปกรณ์และ ไม่ได้ตอบรับ	
Actors	ผู้ใช้งาน	
Related Use Cases	-	
Stakeholders	ผู้ใช้งาน	
Preconditions	ต้องการยืนยันตัวตน	
Postconditions	แสดงประวัติเฉพาะสายไม่ได้รับไปยัง Smart Phone	
Flow of Events	Actor	System
	<ol style="list-style-type: none"> 1. เข้าสู่ระบบ 2. ดูประวัติสายไม่ได้รับ 	<ol style="list-style-type: none"> 3. แสดงประวัติสายไม่ได้รับไปยัง Smart Phone 4. ลบการแจ้งเตือนสายที่ไม่ได้รับ
Exception Condition	-	

ตารางที่ 3.14 คำอธิบายยูสเคส ปลดล็อคประตู

Use case name	ปลดล็อคประตู	
Scenario	ผู้ใช้งานต้องการปลดล็อคประตู	
Triggering Event	ต้องการจะปลดล็อคประตู	
Brief Description	ปลดล็อคประตู	
Actors	ผู้ใช้งาน , ผู้ที่มากดกริ่ง	
Related Use Cases	-	
Stakeholders	ผู้ใช้งาน	
Preconditions	ต้องการยืนยันตัวตน	
Postconditions	ปลดล็อคประตูให้กับผู้ใช้สมาร์ทโฟนและผู้รู้รหัสการปลดล็อค	
Flow of Events	Actor	System
	<ol style="list-style-type: none"> 1. เข้าสู่ระบบ 	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ...
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	2. กดปุ่มปลดล๊อคประตู	3. แจ้งเตือน Yes หรือ no ในการปลดล๊อคประตู 4. ถ้าตอบ Yes ปลดล๊อคประตูทันที
Exception Condition	-	

ตารางที่ 3.15 คำอธิบายยูสเคส ลบข้อมูลประวัติย้อนหลัง

Use case name	ลบข้อมูลประวัติย้อนหลัง	
Scenario	ผู้ใช้งานต้องลบข้อมูลผู้ที่มาดครั้งย้อนหลัง	
Triggering Event	ต้องการลบข้อมูลประวัติย้อนหลังทั้งหมด	
Brief Description	ลบข้อมูลประวัติย้อนหลังทั้งหมดของผู้ที่มาดครั้ง	
Actors	ผู้ใช้งาน	
Related Use Cases	-	
Stakeholders	ผู้ใช้งาน	
Preconditions	ต้องการยืนยันตัวตน	
Postconditions	ลบข้อมูลประวัติย้อนหลังทั้งหมดในอุปกรณ์กริ่งอัจฉริยะ	
Flow of Events	Actor	System
	1. เข้าสู่ระบบ 2. เลือกเมนู Log 3. กดปุ่มลบข้อมูลย้อนหลัง	3. แจ้งเตือน Yes หรือ no 4. ถ้าตอบ Yes ลบข้อมูลประวัติย้อนหลัง
Exception Condition	-	

ตารางที่ 3.16 คำอธิบายยูสเคส เรียกดูกล้องจากอุปกรณ์

Use case name	รับสายจากอุปกรณ์
Scenario	ผู้ใช้งานต้องการดูกล้องที่ติดจากอุปกรณ์
Triggering Event	ต้องการดูกล้องจากอุปกรณ์
Brief Description	เรียกดูกล้องจากอุปกรณ์ที่เราเลือกเชื่อมต่อ
Actors	ผู้ใช้งาน
Related Use Cases	-

Stakeholders	ผู้ใช้งาน	
Preconditions	ต้องการยืนยันตัวตน	
Postconditions	แสดงภาพจากกล้องที่ติดอยู่กับอุปกรณ์ ไปยัง Smart Phone	
Flow of Events	Actor	System
	1. เข้าสู่ระบบ 2. เรียกดูกล้องจาก อุปกรณ์	3. กล้องส่งภาพไปยัง Smart Phone
Exception Condition	-	

3.4 การออกแบบส่วนติดต่อกับผู้ใช้ (User Interface) บน Android Application

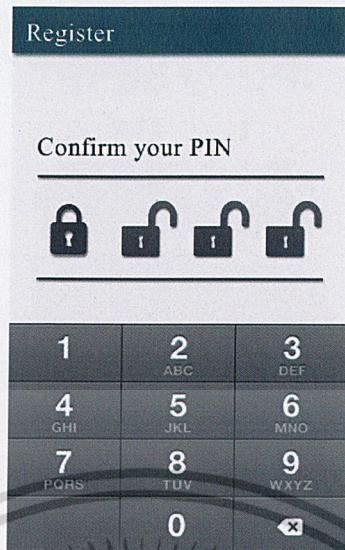
ในส่วนติดต่อประสานงานกับผู้ใช้ ได้ทำการออกแบบหน้าจอหลัก ดังนี้



รูปที่ 3.6 หน้าจอแสดงการ Register

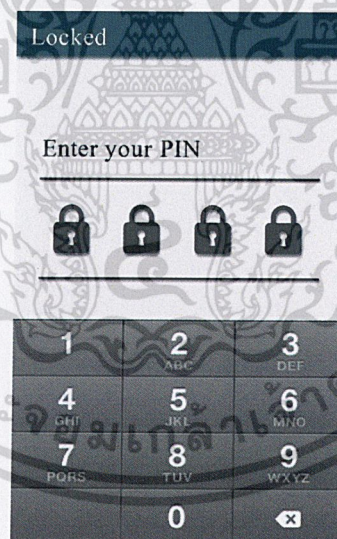
จากรูป 3.6 เป็นการเข้าใช้งานครั้งแรก โดยแอปพลิเคชันจะให้ผู้ใช้งานสร้าง PIN 4 หลัก ซึ่งเป็นระบบป้องกันเบื้องต้น เพื่อป้องกันบุคคลอื่นเข้ามาใช้งานในแอปพลิเคชันนี้ โดยการใส่ PIN 4 หลัก จะบังคับให้ใส่ตัวเลขเท่านั้น เมื่อใส่จนครบ 4 หลักแล้ว แอปพลิเคชันจะให้ทำการให้ยืนยันอีกครั้งในหน้าถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 หน้าจอแสดงการยืนยันรหัส PIN 4 หลัก

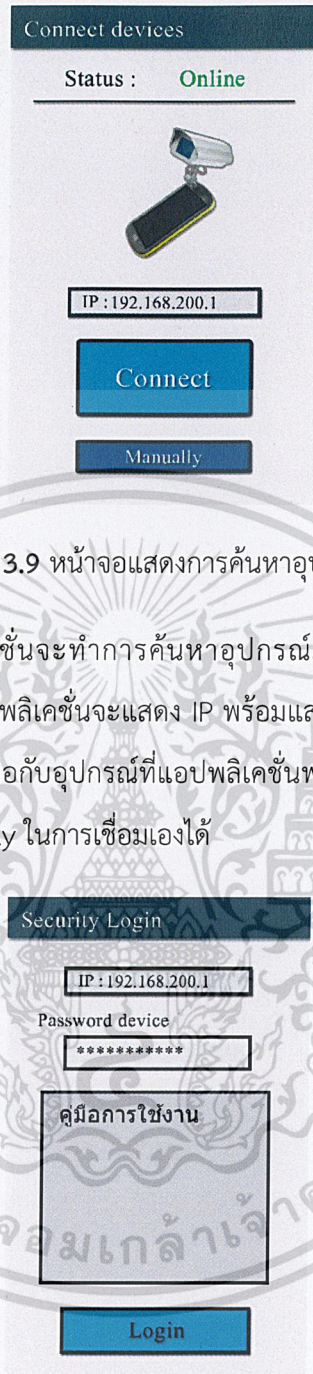
จากรูปที่ 3.7 ผู้ใช้งานจะต้องใส่รหัส PIN 4 หลักอีกครั้ง เพื่อเป็นการยืนยันรหัส PIN หากผู้ใช้งานใส่รหัสไม่ตรงกับรหัสรอบแรกที่กรอก ระบบจะกลับไปหน้า Register ใหม่อีกครั้ง



รูปที่ 3.8 หน้าจอแสดงการปลดล็อค PIN

จากรูปที่ 3.8 แสดงการปลดล็อคของ PIN ที่สมัครเข้ามา ถ้าใส่ PIN 4 หลักถูกต้องแล้ว ระบบจะแสดงหน้าจอถัดไป ถ้ารหัส PIN 4 หลักผิดระบบจะให้ใส่ใหม่อีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Connect devices

Status : **Online**

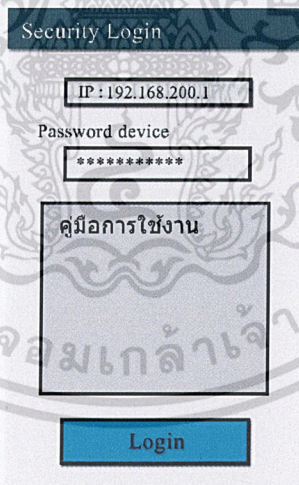
IP : 192.168.200.1

Connect

Manually

รูปที่ 3.9 หน้าจอแสดงการค้นหาอุปกรณ์

จากรูปที่ 3.9 แอปพลิเคชันจะทำการค้นหาอุปกรณ์ภายในวง Lan เดียวกัน เมื่อแอปพลิเคชันพบอุปกรณ์แล้ว แอปพลิเคชันจะแสดง IP พร้อมแสดงสถานะ Online ผู้ใช้งานสามารถเลือกที่เมนู Connect เพื่อเชื่อมต่อกับอุปกรณ์ที่แอปพลิเคชันพบ หากแอปพลิเคชันไม่พบอุปกรณ์ ผู้ใช้งานสามารถเลือกเมนู Manually ในการเชื่อมเองได้



Security Login

IP : 192.168.200.1

Password device

คู่มือการใช้งาน

Login

รูปที่ 3.10 หน้าจอแสดงการ Login เข้าสู่ระบบของอุปกรณ์

จากรูปที่ 3.10 เมื่อค้นหาอุปกรณ์เจอแล้ว ให้ใส่ Password ของอุปกรณ์เพื่อทำการ Login เข้าใช้งาน เป็นการป้องกันความปลอดภัยของอุปกรณ์ โดย Password นั้น จะติดอยู่กับอุปกรณ์ ผู้ใช้งานสามารถเปลี่ยน Password ได้หลังจาก Login เข้าใช้งานแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 หน้าจอแสดงภาพวิดีโอจาก อุปกรณ์ที่เชื่อมต่อ

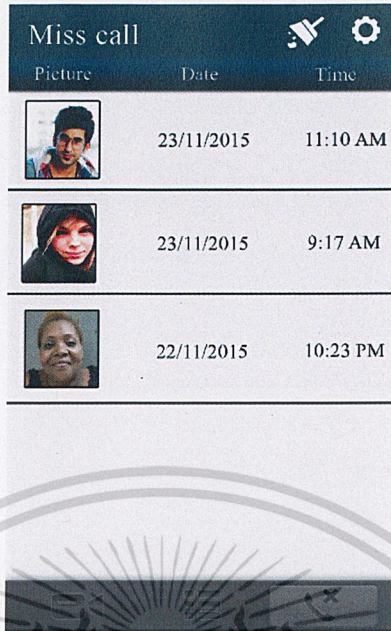
จากรูปที่ 3.11 เป็นการแสดงภาพวิดีโอจากอุปกรณ์ โดยจะเป็นเมนูแรกที่เจอหลังจาก Login เข้าใช้งาน ผู้ใช้สามารถสัมผัสสัญลักษณ์รูปลำโพงเพื่อเปิดเสียงฟังจากอุปกรณ์ได้ และยังมีฟังก์ชันอื่นๆ อาทิเช่น เมนูแก้ไข(Setting) , เมนูดูข้อมูลย้อนหลัง(Log) และเมนูสายไม่ได้รับ(Miss call)

Picture	Date	Time	Status
	23/11/2015	11:10 AM	Misscd
	23/11/2015	9:17 AM	Misscd
	22/11/2015	10:23 PM	Misscd
	21/11/2015	11:13 AM	Accepted
	20/11/2015	11:35 PM	Denied

รูปที่ 3.12 หน้าจอแสดง Log ของอุปกรณ์

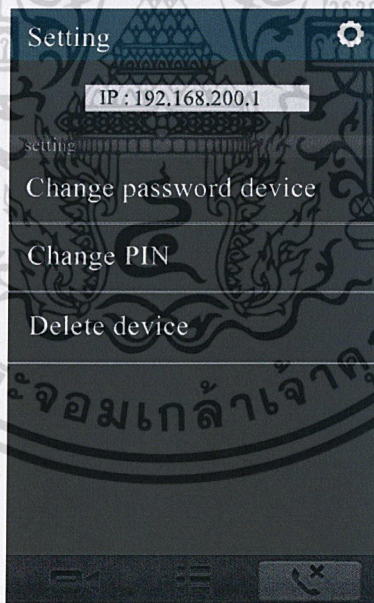
จากรูปที่ 3.12 แสดงการเก็บข้อมูลย้อนหลังของอุปกรณ์โดยจะแสดงผลเป็น รูป , เวลา , วันที่ , สถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 หน้าจอแสดง Miss Call

จากรูปที่ 3.13 แสดงสายที่ไม่ได้รับจากอุปกรณ์ โดยจะแสดงผลเป็น รูป , เวลา , วันที่



รูปที่ 3.14 หน้าจอแสดง Setting

จากรูปที่ 3.14 ผู้ใช้สามารถเปลี่ยนแปลง Password ของอุปกรณ์ และ PIN รวมถึงลบอุปกรณ์ที่เชื่อมต่ออยู่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

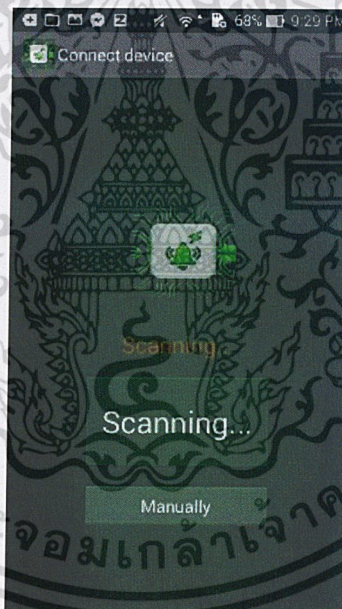
ผลการดำเนินงานและอภิปรายผล

ในบทนี้จะเป็นการกล่าวถึงผลการดำเนินงาน และอภิปรายผลการดำเนินงานของโครงการ กริ่งอัจฉริยะ ที่ทางคณะผู้จัดทำได้ดำเนินการออกแบบการใช้งานกริ่งอัจฉริยะ โดยคำนึงถึงประโยชน์สูงสุดที่กลุ่มเป้าหมายพึงจะได้รับ ซึ่งผลการทดสอบการใช้งานกริ่งอัจฉริยะได้ผลดังนี้

4.1 การทดสอบฟังก์ชันบนอุปกรณ์สมาร์ทโฟนที่เชื่อมต่อกับกริ่งอัจฉริยะ

ฟังก์ชันการใช้งานต่างๆบนอุปกรณ์สมาร์ทโฟนที่เชื่อมต่อกับกริ่งอัจฉริยะ มีการทดสอบรูปแบบการทำงานต่างๆ ดังนี้

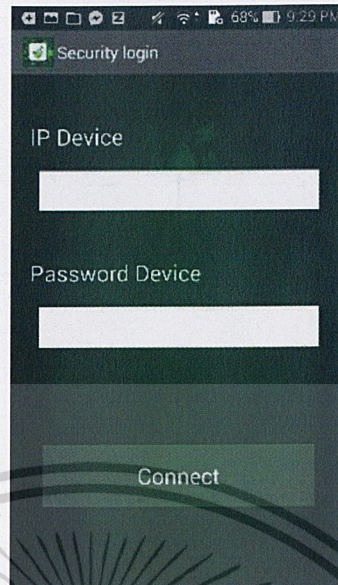
4.1.1 ฟังก์ชันการค้นหาอุปกรณ์กริ่งบนวงแลนเดียวกัน



รูปที่ 4.1 หน้าจอแสดงการค้นหาอุปกรณ์ในวงแลนเดียวกัน

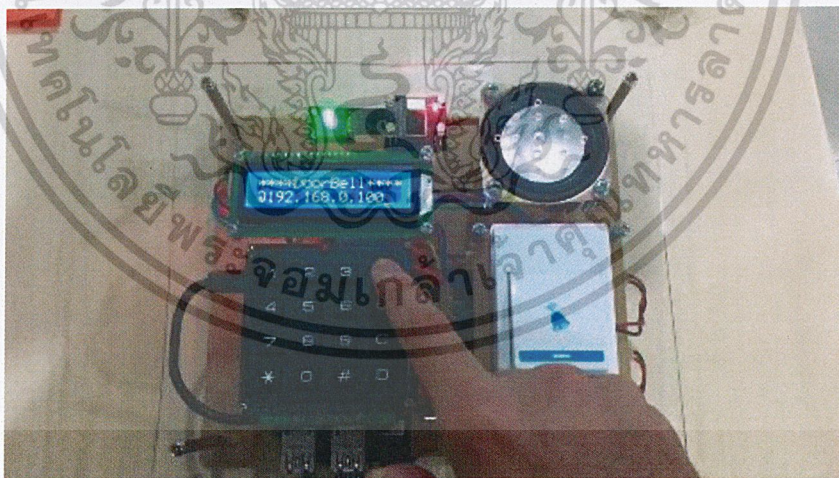
จากรูปที่ 4.1 เมื่อเข้าสู่ขั้นตอนนี้ระบบจะทำการค้นหา IP เพื่อเชื่อมต่อกับอุปกรณ์กริ่งในวงแลนเดียวกันโดยใช้เวลาในการค้นหา IP อุปกรณ์ประมาณ 1 นาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 หน้าจอแสดงการเชื่อมต่ออุปกรณ์แบบ Manually

จากรูปที่ 4.2 ผู้ใช้งานยังสามารถกดปุ่ม Manually เพื่อทำการเชื่อมต่อกับเครื่องโดยไม่ต้องรอการค้นหา แต่ในกรณีที่ระบบได้มีการค้นหาพบ IP ของอุปกรณ์เครื่องอัจฉริยะแล้ว เมื่อกดปุ่ม Connect ระบบจะกรอก IP ลงในช่องของ IP Device ให้และไม่ให้ผู้ใช้แก้ไข IP ที่ได้มา



รูปที่ 4.3 การเรียกดู IP จากเครื่องอัจฉริยะ

จากรูปที่ 4.3 ผู้ใช้งานสามารถดู IP จากอุปกรณ์ได้โดยการกดปุ่ม A ที่อุปกรณ์เครื่อง

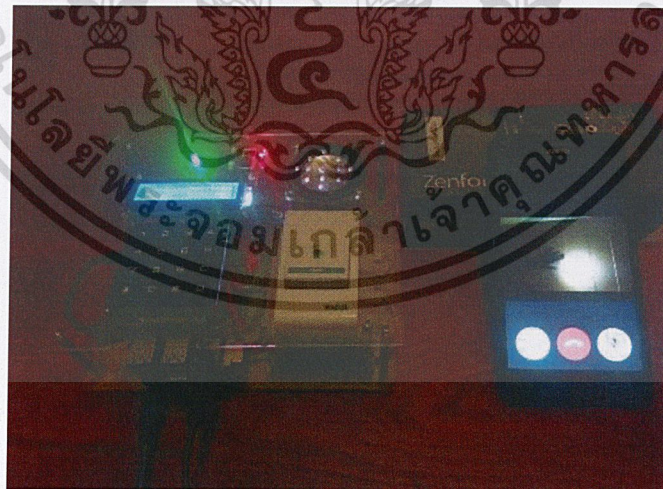
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 ฟังก์ชันสำหรับการโต้ตอบระหว่างผู้ใช้งานบนอุปกรณ์สมาร์ตโฟนกับบุคคลที่มากดกริ่งอัจฉริยะ



รูปที่ 4.4 ภาพแสดงเมื่อกดปุ่มที่อุปกรณ์กริ่งและส่งภาพไปยังอุปกรณ์สมาร์ตโฟน(ขณะยังไม่รับสาย)

จากรูปที่ 4.4 เมื่อมีผู้มากดกริ่ง แสงสัญญาณไฟจะถูกเปลี่ยนเป็นสีน้ำเงิน และระบบจะยกเลิกการสตรีมภาพ จากนั้นระบบจะทำการถ่ายภาพผู้ที่มากดกริ่งจำนวน 5 ภาพ พร้อมทั้งส่งสัญญาณไปยังอุปกรณ์สมาร์ตโฟน เพื่อให้ผู้ใช้งานบนอุปกรณ์สมาร์ตโฟนได้รับชมภาพประกอบการตัดสินใจในการเลือกที่จะรับสายหรือไม่รับสายนั้นๆ ได้



รูปที่ 4.5 ภาพแสดงเมื่อกดปุ่มที่อุปกรณ์กริ่งและส่งภาพไปยังอุปกรณ์สมาร์ตโฟน(ขณะรับสาย)

จากรูปที่ 4.5 เลือกรับสาย สัญญาณไฟสีน้ำเงินจะถูกเปลี่ยนเป็นสีเขียว โดยภาพเคลื่อนไหว จะมีการสตรีมดีเลย์ในการส่งภาพประมาณ 2 วินาที และเสียงมีดีเลย์ประมาณ 0.2-0.5 วินาที เอกสารทั้งหมดนี้ขึ้นอยู่กับสถานที่ติดตั้ง ารใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 ฟังก์ชันการรับชมรูปที่ถูกสตรีมมาจากกริ่ง



รูปที่ 4.6 หน้าจอหลัก

จากรูปที่ 4.6 เมื่อคลิกที่ปุ่ม icon รูปกล้อง ระบบจะทำการสตรีมภาพให้ผู้ใช้งานอุปกรณ์
 สมาร์ทโฟน ดังรูปที่ 4.7



รูปที่ 4.7 หน้าจอการรับชมรูปที่ถูกสตรีมมาจากกริ่ง

จากรูปที่ 4.7 เป็นภาพเคลื่อนไหวจากอุปกรณ์บนกริ่ง ที่ส่งมายังอุปกรณ์บนสมาร์ทโฟน โดย
 ภาพจะมีดีเลย์ประมาณ 2 วินาที ขึ้นอยู่กับสถานที่ติดตั้ง หากอุปกรณ์กริ่งมีปัญหา ภาพด้านซ้ายจะ
 แสดงผลเป็น Offline ทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

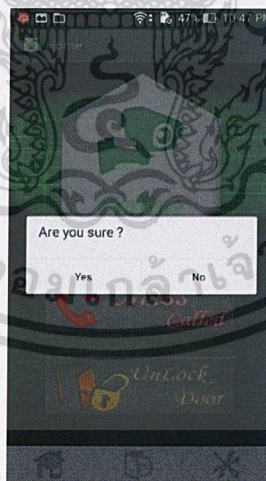
4.1.4 ฟังก์ชันการเก็บบันทึกข้อมูลของบุคคลที่ตกกริ่งและผู้ที่ได้รับสายจากกริ่ง



รูปที่ 4.8 หน้าจอแสดงการเก็บบันทึกข้อมูลของบุคคลที่ตกกริ่งและผู้ที่ได้รับสายจากกริ่ง

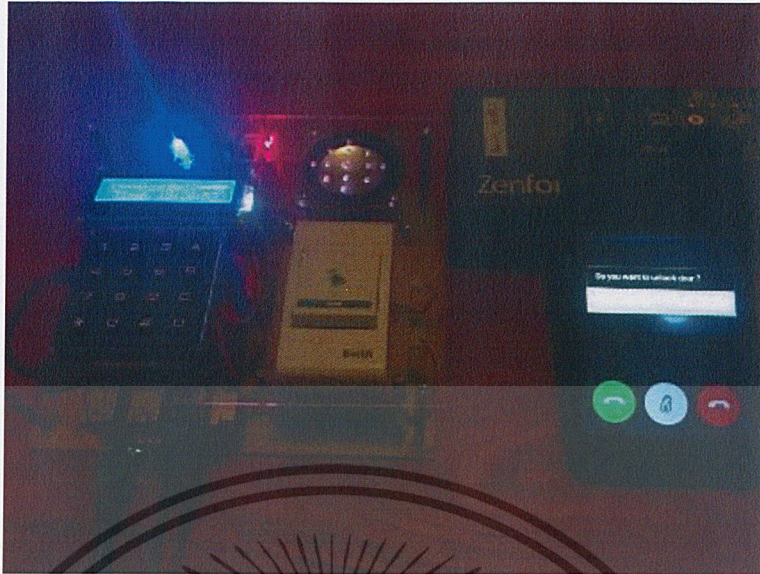
จากรูปที่ 4.8 แสดงหน้าจอการเก็บข้อมูลของบุคคลที่มากดกริ่ง โดยแสดง วันที่, เวลา, สถานะ, รูปภาพจำนวน 5 ภาพ และข้อมูลอุปกรณ์สมาร์ตโฟนที่ได้รับสายจากกริ่ง โดยจะแสดงชื่อของอุปกรณ์สมาร์ตโฟน เช่น ASUS_Z002 และยังสามารถลบบันทึกข้อมูลออกได้ด้วยการกดที่รูปถังขยะ

4.1.5 ฟังก์ชันการปลดล็อคประตู



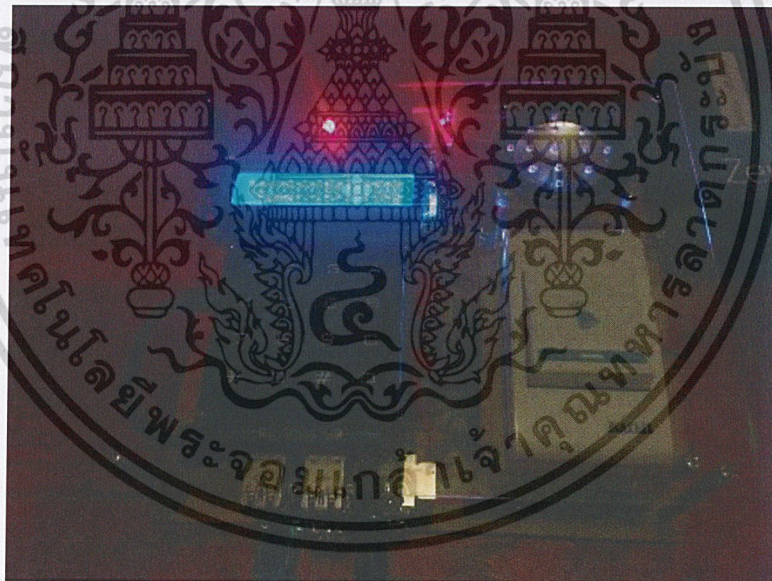
รูปที่ 4.9 การแสดงการปลดล็อคโดยแอปพลิเคชัน

จากรูปที่ 4.9 แสดงการปลดล็อคประตู โดยเมื่อผู้ใช้งานต้องการเปิดประตูให้กับบุคคลภายนอกเข้ามาในระบบจะให้ทำการยืนยันอีกครั้งเพื่อป้องกันการกดผิดพลาด เมื่อกด Yes แล้วระบบจะทำการปลดล็อคประตูทันที และหลังจากที่ปลดล็อคแล้วจะติลเย์ 10 วินาทีเพื่อทำการล็อคอีกครั้ง เอกสารนี้เป็นการสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 การแสดงการปลดล็อคโดยหน้าจอสายเรียกเข้า

จากรูปที่ 4.10 ระบบจะให้ยืนยันอีกครั้ง เมื่อต้องการปลดล็อคประตู เพื่อป้องกันการกดผิดพลาด

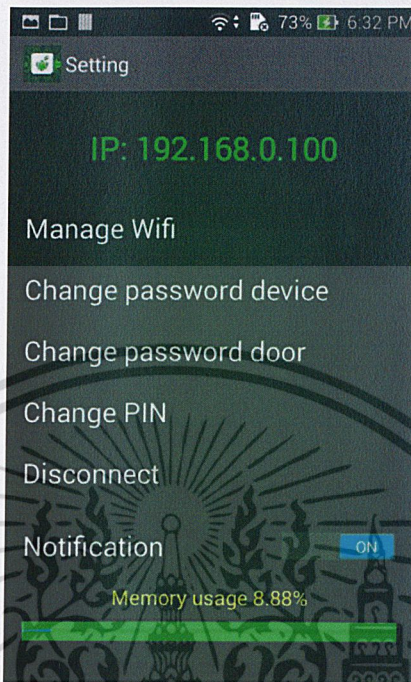


รูปที่ 4.11 การแสดงการปลดล็อคโดยคทรหัสที่อุปกรณ์

จากรูปที่ 4.11 ที่อุปกรณ์กริ่งอัจฉริยะจะมีปุ่มตัวเลขให้กด เพื่อให้ผู้ใช้งานที่รู้รหัสเข้าบ้านได้ โดยไม่ต้องใช้อุปกรณ์สมาร์ทโฟนในการปลดล็อค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.6 ฟังก์ชันการตั้งค่าอุปกรณ์



รูปที่ 4.12 หน้าจอการแสดงผลการตั้งค่าอุปกรณ์ต่างๆ

จากรูปที่ 4.12 หน้าจอจะแสดงผลการตั้งค่าต่างๆ มีดังนี้

1. การจัดการ Wi-Fi ที่เกี่ยวกับอุปกรณ์เครื่องมากที่สุด เพื่อให้เครื่องใช้งานได้อย่างเต็มประสิทธิภาพ
2. การเปลี่ยนพาสเวิร์ดของอุปกรณ์เครื่องอัจฉริยะ
3. การเปลี่ยนพาสเวิร์ดของการปลดล็อคประตูผ่านแป้นพิมพ์บนอุปกรณ์เครื่อง
4. การเปลี่ยนพาสเวิร์ดของอุปกรณ์สมาร์ตโฟน
5. การยกเลิกการเชื่อมต่อของอุปกรณ์เครื่องกับอุปกรณ์สมาร์ตโฟน
6. การปิดการแจ้งเตือนมายังอุปกรณ์สมาร์ตโฟน เมื่อมีบุคคลมากดเครื่อง
7. แสดงสถานะพื้นที่เก็บข้อมูลบนอุปกรณ์เครื่องอัจฉริยะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ตัวอย่างโค้ดแอปพลิเคชันอุปกรณ์กริ่งอัจฉริยะที่พัฒนาด้วยภาษา Python

4.2.1 ตัวอย่างโค้ดค้นหา Wi-Fi และทำการเชื่อมต่ออุปกรณ์ Wi-Fi

1) ค้นหา Wi-Fi ที่ใกล้กับอุปกรณ์กริ่งอัจฉริยะ

```
class _ClientManageWifiScan(_ClientThread):

    def handler(self):
        file = open('/etc/wpa_supplicant/wpa_supplicant.conf', 'r')
        currentWifi = file.readlines()
        if len(currentWifi)>3:
            dataSplit = currentWifi[4].split('ssid="')[1]
            currentSSID = dataSplit.split('"')[0]
        else:
            currentSSID = "Not Connected"
        file.close()
        try:
            data = Cell.all('wlan0')
            self.clientsocket.send((str(len(data))+ "\r\n").encode("UTF-8"))
            self.clientsocket.send((currentSSID+"\r\n").encode("UTF-8"))
            print "Size:"+str(len(data))
            print "Current:"+currentSSID
            for i in range(len(data)):
                self.clientsocket.send(str((data[i].ssid)+"\r\n").encode("UTF-8"))
                print "SSID "+str(i)+" : "+str(data[i].ssid)
                if data[i].encrypted:
                    self.clientsocket.send(str((data[i].encryption_type)+"\r\n").encode("UTF-8"))
                    print "Encryp "+str(i)+" : "+str(data[i].encryption_type)
                else:
                    self.clientsocket.send("\r\n".encode("UTF-8"))
                    print "Encryp "+str(i)+" : None"
                qua = str(data[i].quality.split("/") [0])
                self.clientsocket.send((qua+"\r\n").encode("UTF-8"))
                print "Quality "+str(i)+" : "+str(qua)
            except:
                print "Scan Wifi Error"

        print ('Exit ScanWifi')
```

รูปที่ 4.13 แสดงตัวอย่าง class ที่ใช้ในการส่งข้อมูล SSID ที่ค้นหาได้ไปยังแอปพลิเคชัน

จากรูปที่ 4.13 จะเริ่มต้นด้วยการเปิดคอนฟิกไฟล์เพื่อดึง SSID ของ WiFi ที่เชื่อมต่ออยู่ จากนั้นจะทำการสแกนหาสัญญาณ WiFi รอบๆ และส่งค่า SSID ข้อมูลการเข้ารหัส และคุณภาพของสัญญาณ กลับไปยังสมาร์ตโฟน

2) เชื่อมต่ออุปกรณ์เข้ากับ Wi-Fi ที่ได้จากการค้นหา

```
class _ClientManageWifiModify(_ClientThread):
    def handler(self):
        ssid = str(self.clientsocket.recv(1024).encode("UTF-8").rstrip())
        password = str(self.clientsocket.recv(1024).encode("UTF-8").rstrip())
        encryption = str(self.clientsocket.recv(1024).encode("UTF-8").rstrip())
        print "SSID :"+str(ssid)
        print "Password :"+str(password)
        print "Encryp :"+str(encryption)
        if len(encryption)!=0:
            encryption = encryption+"-PSK"
            print "Encryption :"+str(encryption)
        else :
            print "Encryption : None"
        editFile = open('/etc/wpa_supplicant/wpa_supplicant.conf','w')
        editFile.write('ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev\n')
        editFile.write('update_config=1\n')
        editFile.write('\nnetwork={\n        ssid="'+
        editFile.write(ssid)
        if len(encryption)!=0:
            editFile.write('\n        psk="'+
            editFile.write(password)
            editFile.write('\n        key_mgmt="'+
            editFile.write(encryption)
        else:
            editFile.write('\n        key_mgmt=NONE')
        editFile.write('\n}')
        editFile.close()
        print ('Exit Modify WiFi')
        lcd = pyLcdLib.lcd(0x27,1)
        lcd.lcd_clear()
        lcd.lcd_clear()
        os.system("sudo reboot")
```

รูปที่ 4.14 แสดงตัวอย่าง class ที่ใช้ในการรับข้อมูล SSID ที่ผู้ใช้งานเลือกที่จะเชื่อมต่ออุปกรณ์เข้ากับ Wi-Fi มาทำการแก้ไขไฟล์ config บนอุปกรณ์เครื่องอัจฉริยะ

จากรูปที่ 4.14 เป็นส่วนในการแก้ไข WiFi โดยมีผู้ใช้งานเป็นคนจัดการโดยการทำงานนั้น บนตัวอุปกรณ์จะเปิด socket ไว้สำหรับรับข้อมูลที่ผู้ใช้งานจะส่งกลับมาข้อมูลดังกล่าวได้แก่ SSID รหัสผ่าน และ ข้อมูลการเข้ารหัส โดยเมื่อผู้ใช้งานเลือก WiFi ที่ต้องการจะเชื่อมต่อแล้วระบบจะส่งข้อมูลดังกล่าวมายังอุปกรณ์ จากนั้นจะทำการเปิดคอนฟิกไฟล์ที่ชื่อ wpa_supplicant.conf ขึ้นมาเพื่อทำการแก้ไขข้อมูลการเชื่อมต่อใหม่ หลังจากนั้นจะทำการ reboot ระบบ 1 ครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 ตัวอย่างโค้ดการจัดการกับรหัสผ่านทั้งการเชื่อมต่อมายังอุปกรณ์และระบบ ถือคปรระตุ

1) แอปพลิเคชันกรอกรหัสผ่านเพื่อเชื่อมต่อมายังอุปกรณ์

```
class _ClientPassword(_ClientThread):
    def handler(self):
        getPassword = str(self.clientsocket.recv(1024).encode("UTF-8").rstrip())
        print "Android:"+getPassword
        realPassword = str(self.getPassword().encode("UTF-8").rstrip())
        print "Real:"+realPassword
        print ("Check Password {}".format(getPassword))
        if getPassword == realPassword:
            self.clientsocket.send("true".encode("UTF-8"))
        else:
            self.clientsocket.send("false".encode("UTF-8"))

    def getPassword(self):
        file = open('/home/pi/Desktop/DevicePassword', 'r')
        password = file.readlines()
        print "Real Pass:"+str(password[0])
        return str(password[0])
```

รูปที่ 4.15 แสดงตัวอย่างโค้ดที่ใช้ในการตรวจสอบรหัสผ่านเมื่อผู้ใช้งานเชื่อมต่อมายังอุปกรณ์

จากรูปที่ 4.15 เริ่มต้นการทำงานคือเปิด socket เพื่อรอรับการเชื่อมต่อของสมาร์ทโฟน เมื่อสมาร์ทโฟนเชื่อมต่อมายังอุปกรณ์โดยมีการกรอกรหัสผ่าน ในส่วนนี้จะเป็นส่วนที่นำรหัสผ่านที่ผู้ใช้สมาร์ทโฟนกรอกเข้ามา นำมาทำการตรวจสอบกับรหัสผ่านที่ถูกบันทึกไว้ในไฟล์ที่ชื่อว่า DevicePassword.txt หากรหัสผ่านดังกล่าวตรงกันจะคืนค่า true กลับไปยังสมาร์ทโฟนหากไม่ตรงกันจะคืนค่าเป็น false กลับไป

2) แก้ไขรหัสผ่านเมื่อมีการเชื่อมต่อมายังอุปกรณ์

```
class _ClientChangePasswordDevice(_ClientThread):
    def handler(self):
        file = open('/home/pi/Desktop/DevicePassword.txt', 'r')
        DevicePass = file.readlines()
        DevicePass = str(DevicePass).encode("UTF-8").rstrip()
        print "DevicePass:"+str(DevicePass)
        getOldDevicePass = str(self.clientsocket.recv(1024).encode("UTF-8").rstrip())
        getNewDevicePass = str(self.clientsocket.recv(1024).encode("UTF-8").rstrip())
        print "OldPassword:"+str(getOldDevicePass)
        print "NewPassword:"+str(getNewDevicePass)
        if getOldDevicePass == str(DevicePass):
            print "Password Correct"
            file = open('/home/pi/Desktop/DevicePassword.txt', 'w')
            file.write(getNewDevicePass)
            self.clientsocket.send("true".encode("UTF-8"))
            file.close()
            led = pyled.lib.led(led1,1)
            led.led_clear()
            os.system("sudo reboot")
        else:
            print "Password Wrong"
            self.clientsocket.send("Password Wrong".encode("UTF-8"))
            file.close()
```

รูปที่ 4.16 แสดงตัวอย่างการแก้ไขรหัสผ่านที่ใช้สำหรับเชื่อมต่อมายังอุปกรณ์ครั้ง

จากรูปที่ 4.16 เป็นส่วนการแก้ไขรหัสผ่านที่ใช้สำหรับเชื่อมกับอุปกรณ์โดยจะเปิดไฟล์ที่ชื่อว่า DevicePassword.txt เพื่อดึงรหัสผ่านของตัวอุปกรณ์มาเก็บไว้ในตัวแปร เมื่อผู้ใช้งานทำการแก้ไขรหัสผ่านอุปกรณ์ระบบจะทำการเปิดไฟล์ DevicePassword.txt ขึ้นมาอีกครั้งเพื่อทำการแก้ไขไฟล์เอกสารนี้โดยจะบันทึกที่รหัสผ่านใหม่แทนที่รหัสผ่านเดิมและทำการคืนค่า true กลับไปยังสมาร์ทโฟน โยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) แก้ไขรหัสผ่านเมื่อผู้ใช้งานกรหัสเพื่อปลดล็อคประตู

```
class _ClientChangePasswordDoor (_ClientThread):
    def handler (self):
        file = open ("/home/pi/Desktop/DoorLockPassword","r")
        DoorPass = file.readline ()
        DoorPass = str(DoorPass).encode ("UTF-8").rstrip ()
        print "DoorPassword:"+str(DoorPass)
        getOldDoorPass = str(self.clientsocket.recv(1024).encode("UTF-8").rstrip())
        getNewDoorPass = str(self.clientsocket.recv(1024).encode("UTF-8").rstrip())
        print "OldPassword:"+getOldDoorPass
        print "NewPassword:"+getNewDoorPass
        if getOldDoorPass == str(DoorPass):
            print "Password correct"
            file = open("/home/pi/Desktop/DoorLockPassword","w")
            file.write(getNewDoorPass)
            self.clientsocket.send("true".encode("UTF-8"))
            file.close()
            lcd = pyleclib.lcd(0x27,1)
            lcd.lcd_clear()
            lcd.lcd_clear()
            os.system("sudo reboot")
        else:
            print "Password Wrong"
            self.clientsocket.send("Password Wrong".encode("UTF-8"))
            file.close()
```

รูปที่ 4.17 แสดงตัวอย่างการแก้ไขรหัสผ่านที่ใช้สำหรับการปลดล็อคประตู

จากรูปที่ 4.17 เป็นส่วนในการแก้ไขรหัสผ่านที่ใช้สำหรับการปลดล็อคประตูโดยระบบจะทำการเปิดไฟล์ DoorLockPassword.txt และทำการรับรหัสผ่านที่ผู้ใช้งานต้องการแก้ไขมาทำการแทนที่รหัสผ่านเดิมเมื่อทำเสร็จจะคืนค่า true กลับไปยังสมาร์ทโฟนและทำการรีบูทระบบ หากเกิดข้อผิดพลาดจะคืนค่าเป็น false กลับไป

4.2.3 ตัวอย่างโค้ดการสตรีมมิงภาพ เสียง และการรับเสียงจากแอปพลิเคชันมาแสดงผล

1) สตรีมมิงภาพ

```
def streaming_start (command, delay):
    time.sleep(delay)
    os.system(command)

def prepare():
    s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    gw = os.popen("ip -4 route show default").read().split()
    s.connect((gw[2], 0))
    ip = s.getsockname()[0]
    s.close()
    try:
        thread.start_new_thread(streaming_start, ("raspivid -o - -n -t 0 -w 320 -h 200 --hfliip | cvlc -vvv stream:///dev/stdin "))
        print "Exit Thread Prepare Stream"
    except:
        print "Stream Error"
```

รูปที่ 4.18 แสดงตัวอย่างการใช้งาน VLC Library ในการสตรีมมิงภาพ

จากรูปที่ 4.18 เป็นส่วนโค้ดในส่วนของการสตรีมมิงภาพโดยจะอาศัย Library VLC ในการสตรีมภาพโดยการทำงานคั้นระบบจะทำการดึงไอพีที่อุปกรณ์เชื่อมต่ออยู่นั้นเก็บลงตัวแปร IP และทำการสร้างเทรตใหม่ขึ้นมาสำหรับการประมวลผลคำสั่งการสตรีมมิงโดยจะส่งค่า IP ที่รับมาไปพร้อมเอกสารนี้กับคำสั่งการที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) สตรีมมิ่งเสียง

```

CHUNK = 256
FORMAT = pyaudio.paInt16
CHANNELS = 1
RATE = 44100
p = pyaudio.PyAudio()

def _streamSound(clientsocket):
    streamIn = p.open(format=FORMAT,
                      channels=CHANNELS,
                      rate=RATE,
                      input_device_index=2,
                      input=True,
                      frames_per_buffer=CHUNK)

    isPlay = True
    while isPlay:
        try:
            #print "isPlay: "+str(isPlay)
            dataIn = streamIn.read(CHUNK)
            clientsocket.send(dataIn)
            #isPlay = clientsocket.recv(16).decode("UTF-8").rstrip()
        except IOError as ex:
            # dataIn = None
            # except socket.error as msg:
            isPlay = False
            print 'Socket.error():', msg
        except IOError as ex:
            dataIn = None
            streamIn = p.open(format=FORMAT,
                              channels=CHANNELS,
                              rate=RATE,
                              input_device_index=2,
                              input=True,
                              frames_per_buffer=CHUNK)
            print 'IOError():', ex

    streamIn.stop_stream()
    streamIn.close()

s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
gw = os.popen("ip -4 route show default").read().split()
s.connect((gw[2], 0))
ip = s.getsockname()[0]
s.close()
host = ip#socket.gethostname()
port = 8555
socketserver = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
print 'Open socket {}:{:}'.format(socket.gethostname(), port)
socketserver.bind((host, port))
socketserver.listen(10)

while True:
    try:
        clientsocket, addr = socketserver.accept()
        print 'Connection from {}'.format(str(addr))
        _streamSound(clientsocket)
        #clientsocket.close()
        print 'Disconnect'
        clientsocket.close()
    except socket.error as msg:
        print 'Main Socket.error():', msg
    except IOError as ex:
        print 'Main IOError():', ex

socketserver.close()
p.terminate()
print 'Close socket {}:{:}'.format(socket.gethostname(), port)

```

รูปที่ 4.19 แสดงตัวอย่างการสตรีมมิ่งเสียงเมื่อมีการร้องขอจากแอปพลิเคชันมายังตัวอุปกรณ์เครื่อง

จากรูปที่ 4.19 เป็นโค้ดในส่วนของการสตรีมมิ่งเสียงโดยอาศัย Library PyAudio ในการเปิดการสตรีมมิ่งเสียง และทำการเปิดพอร์ต 8555 เพื่อให้สมาร์ทโฟนสามารถเชื่อมต่อมายังอุปกรณ์ได้ โดยการทำงานหลักๆจะทำการกำหนดคุณสมบัติของการสตรีมมิ่งเสียง หลังจากนั้นจะเข้าสู่กระบวนการรับเสียงจาก Microphone พร้อมกับส่งเสียงดังกล่าวไปยังสมาร์ทโฟนผ่าน socket ที่มีการเชื่อมต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) รับเสียงจากแอปพลิเคชันมาแสดงผลบนอุปกรณ์กริ่ง

```

CHUNK = 2048*2048
FORMAT = pyaudio.paInt16
CHANNELS = 1
RATE = 44100
p = pyaudio.PyAudio()

def _getSound(clientsocket):
    streamOut = p.open(format=FORMAT,
                       channels=CHANNELS,
                       output_device_index=2,
                       rate=RATE,
                       output=True,
                       frames_per_buffer=CHUNK)

    isPlay = True
    dataOut = clientsocket.recv(CHUNK)
    while isPlay and dataOut:
        try:
            streamOut.write(dataOut)
            dataOut = clientsocket.recv(CHUNK)
        except socket.error as msg:
            isPlay = False
            print 'Socket.error{}'.format(msg)
        except IOError as ex:
            dataOut = None
            streamOut = p.open(format=FORMAT,
                               channels=CHANNELS,
                               rate=RATE,
                               output_device_index=2,
                               output=True,
                               frames_per_buffer=CHUNK)
            print 'IOError{}'.format(ex)
            streamOut.close()

    s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    gw = os.popen("ip -4 route show default").read().split()
    s.connect((gw[2], 0))
    ip = s.getsockname()[0]
    s.close()
    host = ip#socket.gethostname()
    port = 8556
    socketserver = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    print 'Open socket {}'.format(socket.gethostbyname(host), port)
    socketserver.bind((host, port))
    socketserver.listen(10)

    while True:
        try:
            clientsocket, addr = socketserver.accept()
            print 'Connection from {}'.format(str(addr))
            _getSound(clientsocket)
            print 'Disconnect'
            clientsocket.close()
        except socket.error as msg:
            print 'Main Socket.error{}'.format(msg)
        except IOError as ex:
            print 'Main IOError{}'.format(ex)

    socketserver.close()
    p.terminate()
    print 'Close socket {}'.format(socket.gethostbyname(host), port)

```

รูปที่ 4.20 แสดงตัวอย่างการรับข้อมูลเสียงจากแอปพลิเคชันมาแสดงผลบนอุปกรณ์กริ่ง

จากรูปที่ 4.20 เป็นโค้ดในส่วนของการรับเสียงจากสมาร์ตโฟนมาแสดงผลที่ตัวอุปกรณ์กริ่งอัจฉริยะโดยการทำงานจะมีลักษณะคล้ายๆกับการส่งเสียงไป ในส่วนแรกของโค้ดจะเป็นส่วนที่ใช้กำหนดคุณสมบัติที่จะใช้แสดงผลเสียงจะต้องมีลักษณะเดียวกับเสียงที่ถูกส่งมาหลังจากนั้นจะเข้าสู่กระบวนการอ่านข้อมูลเสียงที่ถูกส่งมาเพื่อนำไปแสดงผลโดยระบบจะหยุดการทำงานก็ต่อเมื่อการเอกสารนี้เชื่อมต่อที่เชื่อมต่ออยู่ถูกปิดลงการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 ตัวอย่างโค้ดการทำงานเมื่อมีการกดกริ่ง

1) ตัวอย่างโค้ดการตรวจสอบสถานะขณะกดกริ่ง

```

if(not prev_input) and button ):
    queryCheck = MSql()
    str = "SELECT status FROM transaction WHERE status='calling' OR status = 'Accepted'"
    result = queryCheck.select(str)
    data = result.fetchone()
    if data==None :
        call ([["pkill raspivid"], shell=True)
        call ([["pkill vic"], shell=True)
        self.state = "stop"
        print ("-----")
        print ("Start Thread Take Picture")
        self.datetime = time.strftime("%Y-%m-%d")+ " "+time.strftime("%H:%M:%S")
        takePhotoThread = threading.Thread(target = take_photo,args=(self.datetime,))
        takePhotoThread.start()
        time.sleep(5)
        callingThread = threading.Thread(target = calling,args=(self.datetime,))
        callingThread.start()
        $os.system('sudo pkill -f "sudo python /home/pi/CodePython/testPythonTakePhoto.py")
        $s.connect ((gw[2],0))
        $ipaddr = s.getsockname() [0]
        self.datetime = None
        print ("Exit Take Picture")
        print ("-----")
        call ([["pkill raspivid"], shell=True)
        call ([["pkill vic"], shell=True)
        time.sleep(2)
    else:
        print "Busy!"
queryCheck.close()

```

รูปที่ 4.21 แสดงตัวอย่างขั้นตอนการทำงานเมื่อมีการกดกริ่ง

จากรูปที่ 4.21 เป็นโค้ดในส่วนของการถ่ายภาพเมื่อมีคนมากดกริ่งโดยในส่วนแรกจะเป็นการตรวจสอบสถานะว่ามีคนกดกริ่งอยู่หรือไม่ถ้ามีจะไม่มีการถ่ายภาพ หากยังไม่จะมีจะเข้าสู่กระบวนการถ่ายภาพโดยอันดับแรกจะปิดการทำงานของกรสตรีมภาพเนื่องจากการถ่ายภาพใช้ Library เดียวกันกับการสตรีมภาพและดึงเวลาปัจจุบันจากระบบขึ้นมาใช้ในการสร้างไพลเดอร์เพื่อเก็บภาพและใช้เป็น Primary Key ในการค้นหาภาพ เมื่อถ่ายภาพเสร็จก็จะทำการสตรีมภาพกลับเหมือนเดิม

2) ถ่ายภาพผู้ที่มากดกริ่ง

```

def streaming_start(command, delay):
    time.sleep(delay)
    os.system(command)

def take_photo(datetime):
    try:
        thread.start_new_thread(streaming_start, ("sudo python /home/pi/Desktop/CapturePhoto.py"+ " "+datetime,1))
        $os.system("python /home/pi/CodePython/testPythonTakePhoto.py")
        print "Exit Thread TakePhoto"
    except:
        print ("Error TakePhoto")

```

รูปที่ 4.22 แสดงตัวอย่างฟังก์ชันที่ถูกเรียกใช้เมื่อมีการกดกริ่ง

จากรูปที่ 4.22 เป็นตัวอย่างโค้ดในส่วนของการเรียกใช้ฟังก์ชันการสร้างเทรดเพื่อใช้งานในกรณีต้องการทำงานแบบมัลติเทรดเพื่อให้สามารถทำงานหลายๆงานได้ในเวลาเดียวกันโดยส่วนที่

แสดงอยู่ในโค้ดนั้นเป็นการสร้างเทรดเพื่อสั่งให้ประมวลผลไฟล์ CapturePhoto.py ซึ่งเป็นโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการรักษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ที่ใช้จัดการเกี่ยวกับการถ่ายภาพเมื่อมีการกดกริ่งนั่นเอง

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

def capture():
    call(["pkill raspivid"], shell=True)
    call(["pkill vlc"], shell=True)
    conn = MSsql()
    #print "#####Data Pass: "+str(sys.argv[1])+" "+sys.argv[2]+"#####"
    datetime = str(sys.argv[1])+" "+sys.argv[2])
    path = "/var/www/html/img/"+str(datetime)
    os.makedirs(path)
    camera = picamera.PiCamera()
    camera.hflip = True
    pathimgid = str("/var/www/html/img/"+datetime+"/")
    time.sleep(2)
    for i in range(0,5):
        path = pathimgid+str(i)+".jpg"
        camera.capture(path)
    data = "INSERT INTO transaction(time,user_accept,status) VALUES('"+datetime+"','','calling')"
    conn.insert(str(data))
    print "insert Complete"
    conn.close()
    call(["pkill raspivid"], shell=True)
    call(["pkill vlc"], shell=True)

```

รูปที่ 4.23 แสดงตัวอย่างขั้นตอนการทำงานอื่นๆในฟังก์ชันการถ่ายภาพของผู้ที่มากดกริ่ง

จากรูปที่ 4.23 เป็นโค้ดตัวเองของการสร้างโฟลเดอร์ตามวันเวลาที่มีคนมากดกริ่งโดยวันเวลาดังกล่าวจะตรงกันข้อมูลที่ถูกบันทึกในฐานข้อมูล และเมื่อสร้างโฟลเดอร์แล้วจะทำการเรียกใช้ Library Picamera ขึ้นมาเพื่อใช้งานการถ่ายภาพและบันทึกลงไปโฟลเดอร์

4.2.5 ตัวอย่างโค้ดควบคุมการทำงานของระบบลิศคประตุ

1) ตัวอย่างโค้ดสำหรับรับค่าการเปลี่ยนสถานะประตุจากแอปพลิเคชัน โฟล์ PHP

```

<?php
$ctl = $_GET['ctl'];
$command = "sudo python /home/pi/Desktop/testPHFConRelay.py $ctl";
//echo $command;
$result = exec($command);
echo($result);
?>

```

รูปที่ 4.24 แสดงตัวอย่างโค้ด PHP ที่รับค่าสถานะประตุจากแอปพลิเคชัน

จากรูปที่ 4.24 เป็นตัวอย่างโค้ดที่ใช้สำหรับรับค่าจากแอปพลิเคชันบนสมาร์ทโฟนในการควบคุมการเปิดประตุและใช้ PHP ในการสั่งประมวลผลไฟล์ Python

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ตัวอย่างโค้ดควบคุมการเปิดปิดประตูที่ถูกสั่งงานมาจากแอปพลิเคชัน

```
import sys
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setup(25,GPIO.OUT)
GPIO.setwarnings(False)
RED = 22
GREEN = 23
BLUE = 4

GPIO.setup(RED,GPIO.OUT,initial=0)
GPIO.setup(GREEN,GPIO.OUT,initial=0)
GPIO.setup(BLUE,GPIO.OUT,initial=0)
GPIO.setup(27,GPIO.OUT,initial=0)
args = sys.argv
pin= 25
ctl = args[1]

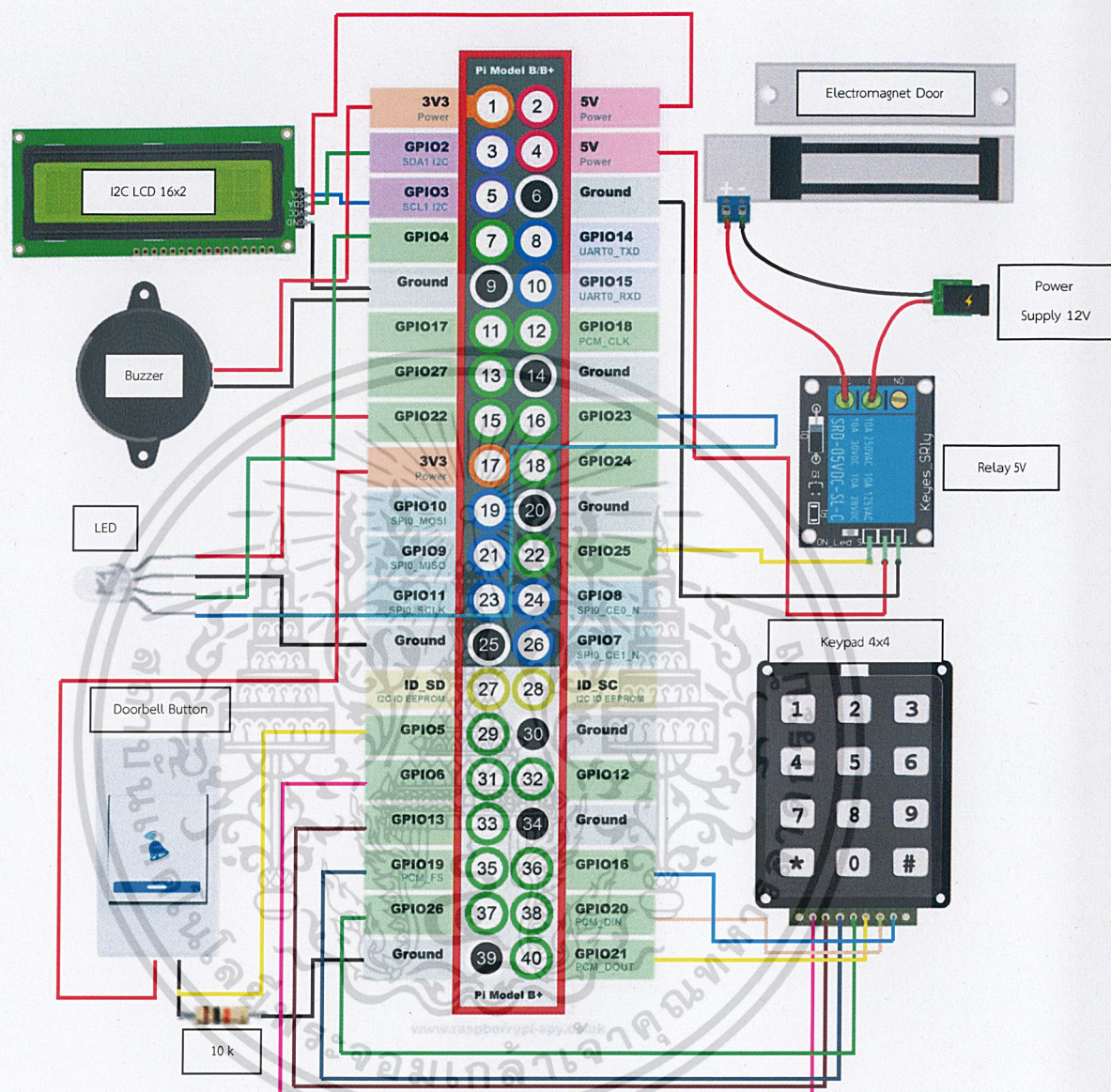
if(int(ctl)==1):
    time.sleep(1)
    GPIO.output(27,GPIO.HIGH)
    time.sleep(0.5)
    GPIO.output(27,GPIO.LOW)
    GPIO.output(GREEN,GPIO.LOW)
    GPIO.output(RED,GPIO.HIGH)
    GPIO.output(25,GPIO.HIGH)
    print "Door Open"
    time.sleep(10)
    GPIO.output(25,GPIO.LOW)
    GPIO.output(RED,GPIO.LOW)
    GPIO.output(GREEN,GPIO.HIGH)
    time.sleep(0.1)
```

รูปที่ 4.25 แสดงตัวอย่างขั้นตอนการสั่งงานรีเลย์ที่ใช้สำหรับควบคุมการล๊อคประตู

จากรูปที่ 4.25 เป็นตัวอย่างโค้ดที่ถูกไฟล์ PHP ส่งประมวลผลโดยค่าที่รับมาจากแอปพลิเคชันบนสมาร์ตโฟนนั้นจะถูกเก็บอยู่ในตัวแปรที่ชื่อ ctl หลังจากนั้นเมื่อ ctl เป็น 1 จะเข้าสู่กระบวนการสั่งอุปกรณ์รีเลย์ควบคุมแม่เหล็กล๊อคประตูที่เชื่อมต่อกับ GPIO ให้ทำงานโดยประตูจะเปิดก็ต่อเมื่อ สถานะการเชื่อมต่อเป็น HIGH พร้อมทั้งแสดงไฟ LED เป็นสีแดง และจะกลับสู่สถานะ LOW หลังจากผ่านไป 10 วินาทีพร้อมทั้งเปลี่ยนสีไฟ LED เป็นสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ตัวอย่างการเชื่อมต่อวงจรอุปกรณ์กริ่งอัจฉริยะผ่าน GPIO



รูปที่ 4.27 ภาพแสดงตัวอย่างวงจรที่เชื่อมต่อกับอุปกรณ์ภายนอกผ่าน GPIO Pin บนอุปกรณ์กริ่งอัจฉริยะ

จากรูปที่ 4.27 เป็น Diagram ของการเชื่อมต่ออุปกรณ์กริ่งอัจฉริยะโดยในครั้งแรกจะเป็นการเชื่อมต่อกริ่งในรูปแบบเดิมๆเข้ากับ GPIO พอร์ตของบอร์ด Raspberry Pi ที่พอร์ต GPIO15 เพื่อรับค่าสถานะการกดกริ่ง ส่วนถัดมาคือส่วนที่รับรหัสผ่านจากผู้ใช้งาน โดยจะเชื่อม Keypad 4x4 เข้ากับ GPIO พอร์ต 6, 13, 19, 26, 21, 20, 16 และในส่วนของการแสดงผลได้แก่ การเชื่อมต่อไฟเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า LED เพื่อแสดงสถานะการทำงานของแม่เหล็กไฟฟ้าประตู (Electromagnet Doorlock) ในส่วนนี้จะไม่ทำการใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เชื่อมต่อเข้ากับ GPIO พอร์ตดังนี้ สีแดง GPIO22 , สีเขียว GPIO4 , สีน้ำเงิน GPIO23 ต่อมาจะเป็นการเชื่อมต่อส่วนการแสดงผลเสียงเตือนเมื่อมีการเปิดประตูจะใช้ Buzzer เป็นตัวแสดงผลเสียงดังกล่าว และในส่วนการแสดงผล ของหน้าจอ LCD 16x2 นั้นจะเชื่อมต่อเข้ากับบอร์ด Raspberry Pi ผ่านตัวแปลงสัญญาณ i2c เข้าที่พอร์ต SDA และ SCL และในส่วนสุดท้ายจะเป็นการเชื่อมต่อรีเลย์ที่ใช้ควบคุมการทำงานของ Electromagnet Doorlock โดยรีเลย์ดังกล่าวนั้นจะต่อเข้ากับ GPIO25 และอีกด้านของรีเลย์จะเชื่อมต่อไปยัง Electromagnet Doorlock กับแหล่งจ่ายไฟที่มีแรงดันไฟฟ้า 12 v



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

ผลของการทดสอบจากโปรแกรมนี้ ผู้ที่อาศัยอยู่ภายในบ้าน สามารถสนทนากับผู้ที่มาดกริงได้ โดยมีทั้งภาพและเสียง และสามารถปลดล็อคประตูได้โดยสั่งผ่านสมาร์ทโฟน รวมถึงสามารถส่งสัญญาณไปได้ทุกเครื่องที่เชื่อมต่อภายในระบบ แต่จะยังมีเสียงที่ส่งระหว่างอุปกรณ์ล่าช้ากว่าปกติ

5.2 ขอบเขตและข้อจำกัดของการพัฒนา

- 5.2.1 อุปกรณ์กริ่งอัจฉริยะสามารถส่งสัญญาณในรูปแบบของเสียงและวิดีโอ ไปยังสมาร์ทโฟนที่เชื่อมต่อกับกริ่งอัจฉริยะได้
- 5.2.2 อุปกรณ์กริ่งอัจฉริยะสามารถบันทึกภาพบุคคลที่มาดกริงได้ 5 ภาพ
- 5.2.3 แอปพลิเคชันที่ถูกพัฒนาขึ้น เพื่อใช้งานร่วมกับสมาร์ทโฟนจะสามารถเห็นใบหน้าและได้ยินเสียงของผู้ที่มาดกริงได้แบบ Real-Time
- 5.2.4 เมื่อมีผู้มาดกริง แอปพลิเคชันที่ถูกติดตั้งบนสมาร์ทโฟนจะแจ้งเตือน พร้อมทั้งแสดงภาพและเสียงของผู้ที่มาดกริง
- 5.2.5 ผู้ใช้งานแอปพลิเคชันบนสมาร์ทโฟนสามารถเลือกที่จะตอบรับหรือไม่ตอบรับเมื่อ มีคนมาดกริง
- 5.2.6 อุปกรณ์กริ่งอัจฉริยะ ไม่สามารถส่งสัญญาณเสียงไปยังแอปพลิเคชันบนสมาร์ทโฟน
หลายๆ เครื่องได้ในเวลาเดียวกัน
- 5.2.7 ผู้ใช้งานแอปพลิเคชันบนสมาร์ทโฟนสามารถตรวจสอบข้อมูลภาพ วัน และเวลา ของผู้ที่มาดกริงได้
- 5.2.8 แอปพลิเคชันบนสมาร์ทโฟนสามารถสั่งงานเปิดประตูบ้านได้

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาและพัฒนาต่อไป ไม่สามารถนำเอกสารนี้ไปใช้ในการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.10 ในกรณีที่อุปกรณ์เครื่องเปลี่ยนเร้าเตอร์ที่เชื่อมต่อ แอปพลิเคชันบนสมาร์ตโฟนจะต้องทำงานเชื่อมต่อกับอุปกรณ์เครื่องใหม่อีกครั้ง

5.3 ข้อเสนอแนะ

5.3.1 พัฒนาให้สามารถเก็บข้อมูลเป็น Video ขณะที่สนทนากันได้

5.3.2 พัฒนาให้เชื่อมต่อจากที่ไหนก็ได้ (ระบบนี้สามารถเชื่อมต่อได้ภายในวงแลนเท่านั้น)

5.3.3 พัฒนาให้เมื่อมีคนผ่านหน้าประตูบ้าน สามารถบันทึกภาพของผู้ที่ผ่านไปมาได้

5.3.4 พัฒนาให้จดจำใบหน้าได้

5.3.5 พัฒนาให้ใช้การสแกนลายนิ้วมือแทนการกรอกรหัส

5.3.6 พัฒนาให้ส่งเสียงขอความช่วยเหลือไปยังนอกร้าน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] Adam. 2013. **Streaming Video Using VLC Player**. [Online]. Available : <http://www.raspberry-projects.com/pi/pi-hardware/raspberry-pi-camera/streaming-video-using-vlc-player>
- [2] Dipto Pratyaksa. 2013. **raspberry pi microphone setup with usb sound card**. [Online]. Available : <http://www.linuxcircle.com/2013/05/08/raspberry-pi-microphone-setup-with-usb-sound-card/>
- [3] Ste Wright. 2015. **Tutorial – Install Apache, PHP and MySQL on a Raspberry Pi 2**. [Online]. Available : <http://raspipress.com/2015/08/tutorial-install-apache-php-and-mysql-on-a-raspberry-pi-2/#comments>
- [4] innovative experiment. 2013. **[Raspberrry Pi] การเชื่อมต่อ WiFi ผ่าน Command Line**. [Online]. Available : <http://doc.inex.co.th/r-pi-wifi-command-line/>
- [5] scottkildall. 2013. **Ultimate Raspberry Pi Configuration Guide**. [Online]. Available : <http://www.instructables.com/id/Ultimate-Raspberry-Pi-Configuration-Guide/?ALLSTEPS>
- [6] PhillyNJ. 2014. **USB card as my default Audio Device**. [Online]. Available : <http://raspberrypi.stackexchange.com/questions/19705/usb-card-as-my-default-audio-device>
- [7] Ioannis Kedros. 2015. **running python script at boot**. [Online]. Available : <http://embeddedday.com/projects/raspberry-pi/a-step-further/running-python-script-at-boot/>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



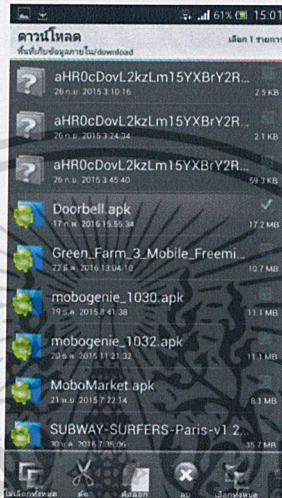
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

ก. วิธีการติดตั้งและคู่มือการใช้งานแอปพลิเคชัน Doorbell Intelligent

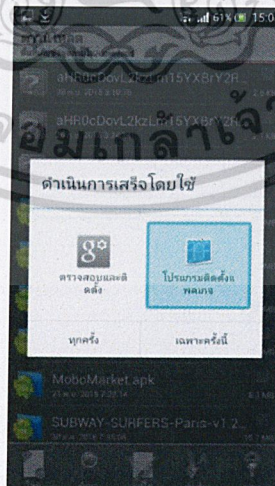
ก.1 วิธีการติดตั้งแอปพลิเคชัน Doorbell Intelligent

1) นำไฟล์แอปพลิเคชัน Doorbell Intelligent ลงบนสมาร์ตโฟน



รูปที่ ก.1 ตรวจสอบไฟล์ .apk ที่ใช้ติดตั้งบนสมาร์ตโฟนผ่านโปรแกรม File Commander

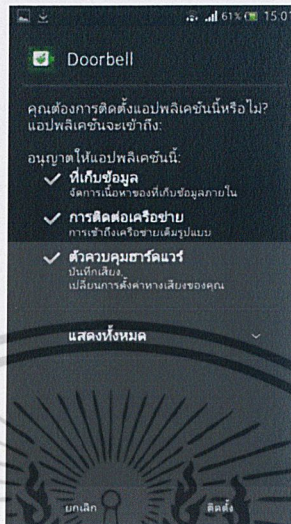
2) เมื่อกดที่ไฟล์ Doorbell.apk จะปรากฏหน้าต่างดังรูปที่ ก.2 และเลือกโปรแกรมติดตั้งแพคเกจหลังจากนั้นกดปุ่ม เฉพาะครั้งนี้เพื่อเริ่มการติดตั้ง



รูปที่ ก.2 หน้าจอแสดงตัวเลือกการติดตั้งไฟล์ .apk

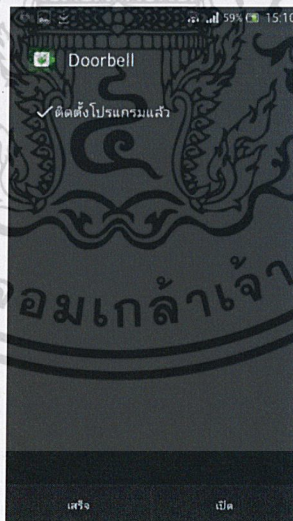
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ก่อนทำการติดตั้งแอปพลิเคชัน จะต้องยอมรับการเข้าถึงข้อมูลต่างๆ ถ้ายอมรับข้อตกลงในการเข้าถึงข้อมูลให้กดติดตั้งเพื่อทำการติดตั้งแอปพลิเคชัน



รูปที่ ก.3 หน้าจอแสดงรายละเอียดการเข้าถึงข้อมูลของแอปพลิเคชัน

4) รอจนกว่าจะติดตั้งเสร็จจะพบหน้าต่างดังรูปที่ ก.4 จะแสดงถึงการติดตั้งแอปพลิเคชันเสร็จสมบูรณ์และสามารถเปิดใช้งานได้ทันที



รูปที่ ก.4 แสดงการแจ้งเตือนเมื่อติดตั้งแอปพลิเคชันสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก.2 คู่มือการใช้งานแอปพลิเคชัน Doorbell Intelligent

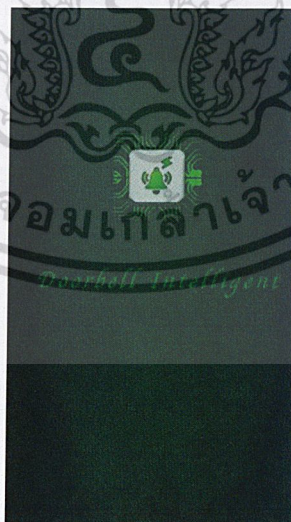
1) เมื่อติดตั้ง Doorbell Intelligent แล้วจะปรากฏไอคอน ดังรูปที่ ก.



รูปที่ ก.5 ไอคอนแอปพลิเคชัน

เมื่อผู้ใช้งานต้องการที่จะใช้งานแอปพลิเคชัน Doorbell Intelligent ให้ผู้ใช้งาน
เลือกที่ ไอคอนของแอปพลิเคชันเพื่อทำการเปิดแอปพลิเคชัน

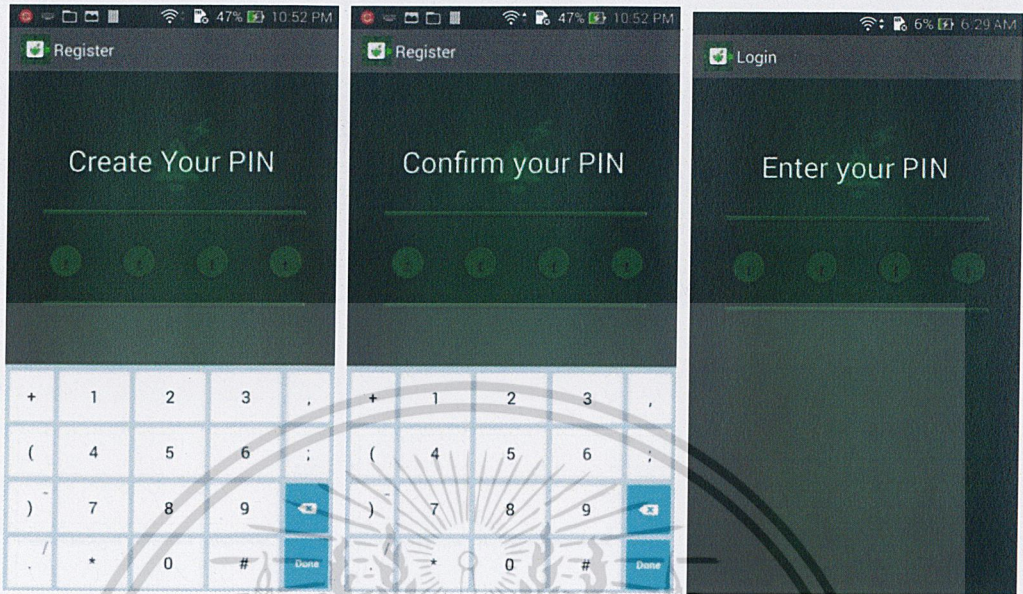
2) หน้าจอ loading เมื่อกดไอคอนแอปพลิเคชัน



รูปที่ ก.6 หน้าจอ loading ของแอปพลิเคชัน

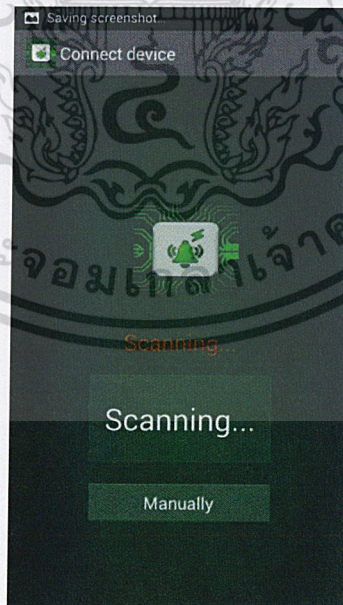
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) เมื่อผู้ใช้เข้าใช้งานแอปพลิเคชันครั้งแรก



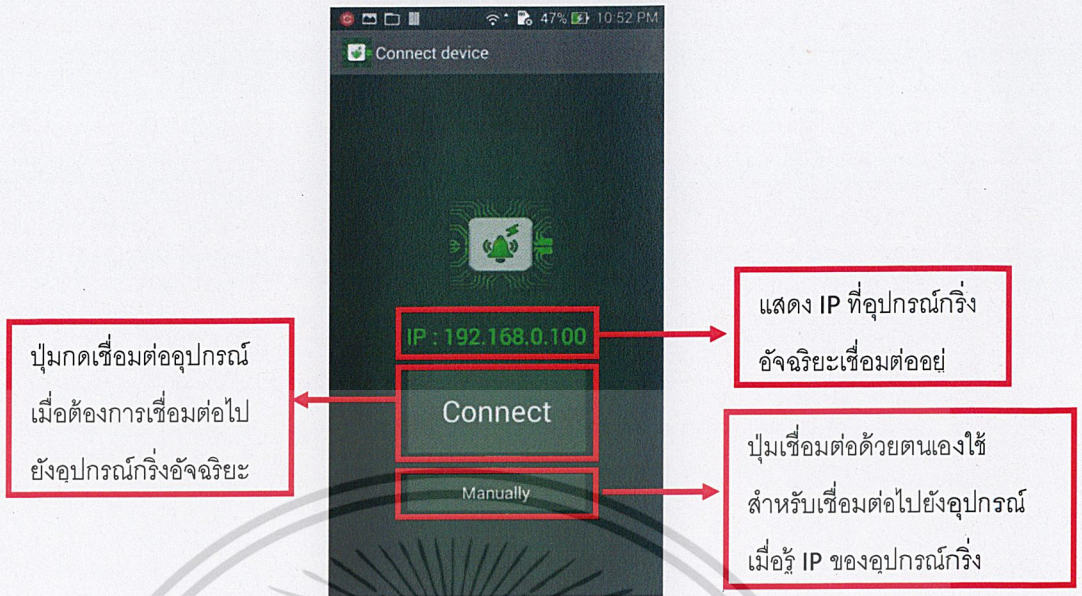
รูปที่ ก.7 หน้าจอสร้าง PIN Code และการเข้าใช้งานผ่าน PIN Code ของแอปพลิเคชัน

4) หน้าจอค้นหาอุปกรณ์เมื่อผู้ใช้งานเข้าใช้งานครั้งแรก

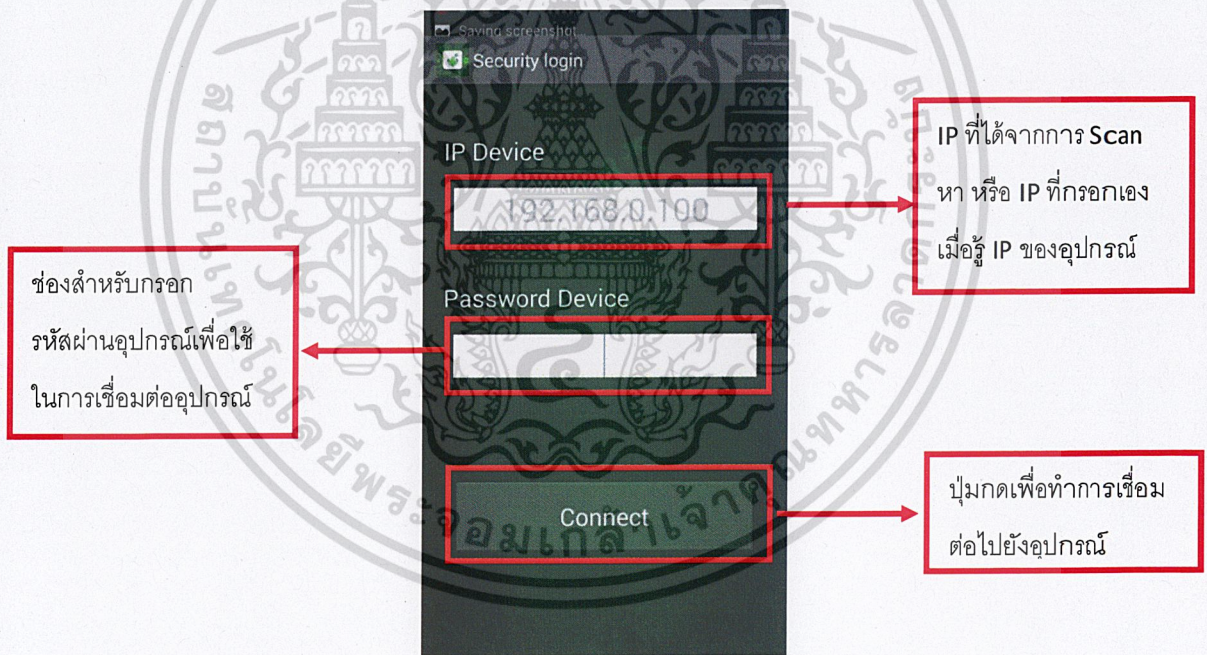


รูปที่ ก.8 หน้าจอ scan หาอุปกรณ์ในวงแลนเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



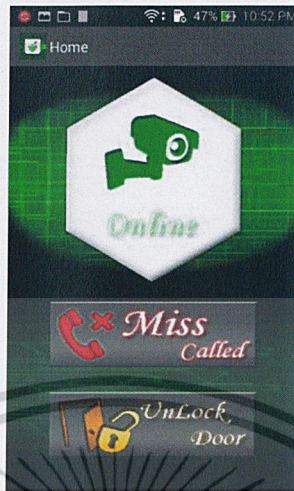
รูปที่ ก.9 หน้าจอ scan หาอุปกรณ์ในวงแลนเดียวกัน



รูปที่ ก.10 หน้าจอเชื่อมต่อไปยังอุปกรณ์เครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

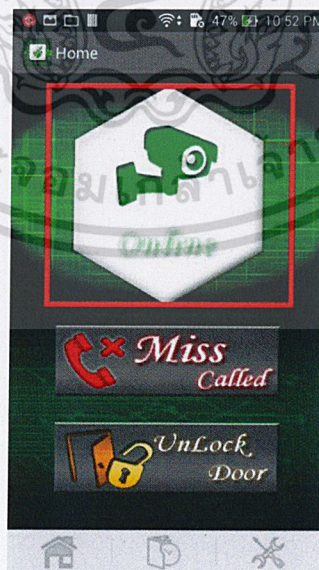
5) หน้าเมนูหลักของ Doorbell Intelligent



รูปที่ ก.11 หน้าเมนูหลักของแอปพลิเคชัน

จากรูปที่ ก. ผู้ใช้งานสามารถเลือกฟังก์ชันการทำงานได้ โดยสามารถกดที่รูปไอคอนต่างๆ เพื่อเข้าสู่การทำงานของฟังก์ชันนั้นๆ

6) เมื่อเลือกฟังก์ชันรับชมภาพจากอุปกรณ์จริง



รูปที่ ก.12 ฟังก์ชันรับชมภาพจากอุปกรณ์จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) หน้าจอแสดงภาพเมื่อเลือกฟังก์ชันรับชมภาพจากอุปกรณ์กริ่ง



รูปที่ ก.13 หน้าจอฟังก์ชันการรับชมภาพจากอุปกรณ์กริ่ง

จากรูปที่ ก. จะปรากฏรูปที่ทำการสตรีมมิ่งมาจากอุปกรณ์กริ่งอัจฉริยะมาแสดงผลบนแอปพลิเคชัน Doorbell Intelligent

8) เมื่อเลือกฟังก์ชันดูข้อมูลการกดกริ่งที่ไม่มีการตอบรับ



รูปที่ ก.14 ฟังก์ชันดูข้อมูลการกดกริ่งที่ไม่มีการตอบรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9) หน้าจอฟังก์ชันดูข้อมูลการกดครั้งที่ไม่มีการตอบรับ



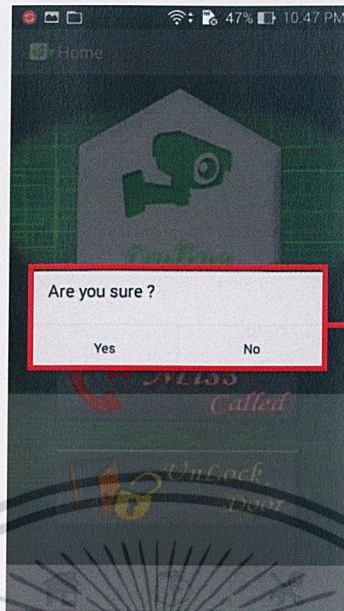
รูปที่ ก.15 หน้าจอแสดงข้อมูลการกดครั้งที่ไม่มีการตอบรับ

10) เมื่อเลือกฟังก์ชันปลดล็อคประตูผ่านแอปพลิเคชัน



รูปที่ ก.16 ฟังก์ชันปลดล็อคประตูผ่านแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



หน้าต่างยืนยันการทำงานเมื่อมีการเรียกใช้ฟังก์ชันปลดล็อคประตู

รูปที่ ก.17 เมื่อเลือกฟังก์ชันปลดล็อคประตู

11) เมื่อเลือกฟังก์ชันดูบันทึกรายการของผู้ที่มากดกริ่ง



รูปที่ ก.18 ฟังก์ชันดูบันทึกรายการของผู้ที่มากดกริ่ง

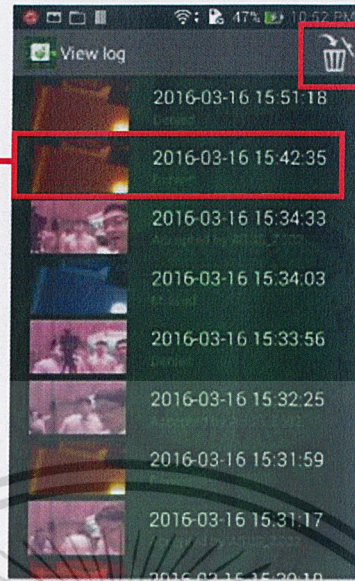
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายการแสดงข้อมูลผู้ที่มาดกริงโดย
แสดง ปี เดือน วัน และเวลา ของผู้ที่มา
ดกริงรวมถึงแสดงสถานะของรายการ
ดังกล่าว ดังนี้

Denied คือ การปฏิเสธการรับสาย

Missed คือ ไม่มีการตอบรับเมื่อมี
การดกริง

Accept By คือ ตอบสายโดยสมบูรณ์



ปุ่มลบข้อมูลผู้
ที่มาดกริง

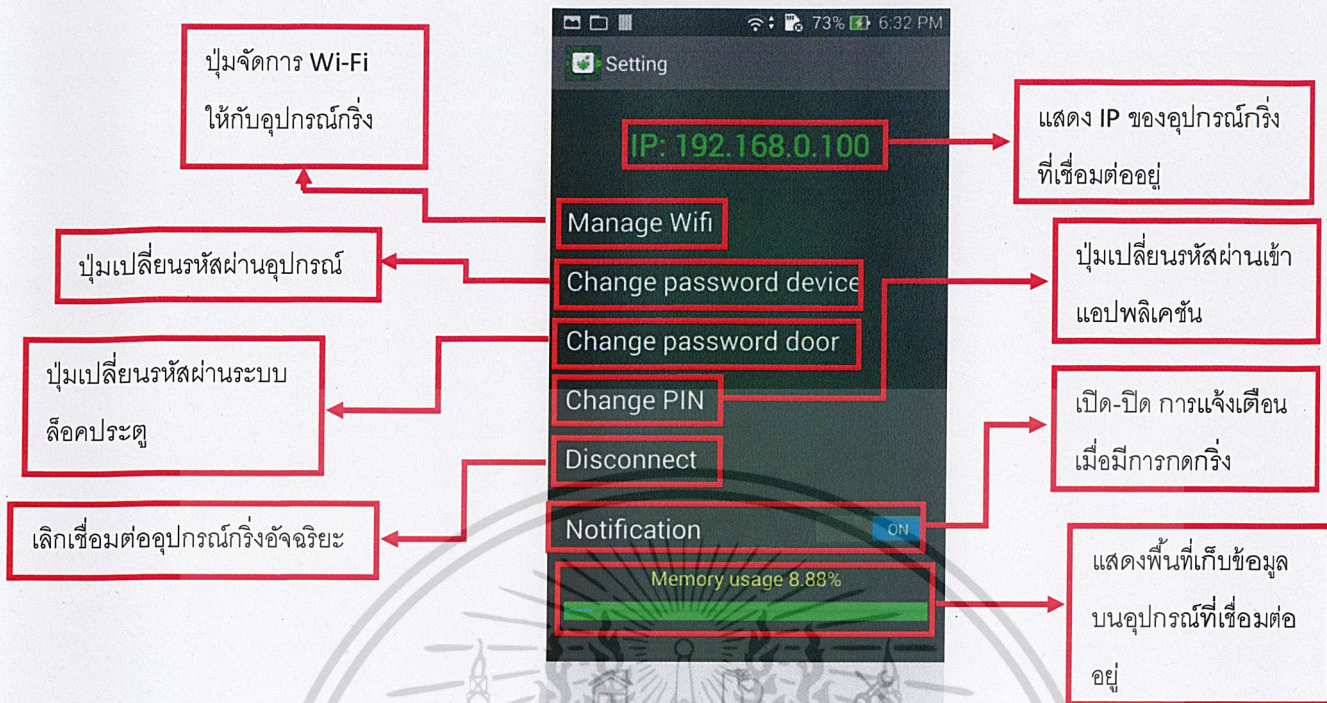
รูปที่ ก.19 หน้าจอฟังก์ชันแสดงรายการข้อมูลของผู้ที่มาดกริง

12) เมื่อเลือกฟังก์ชันตั้งค่าอุปกรณ์



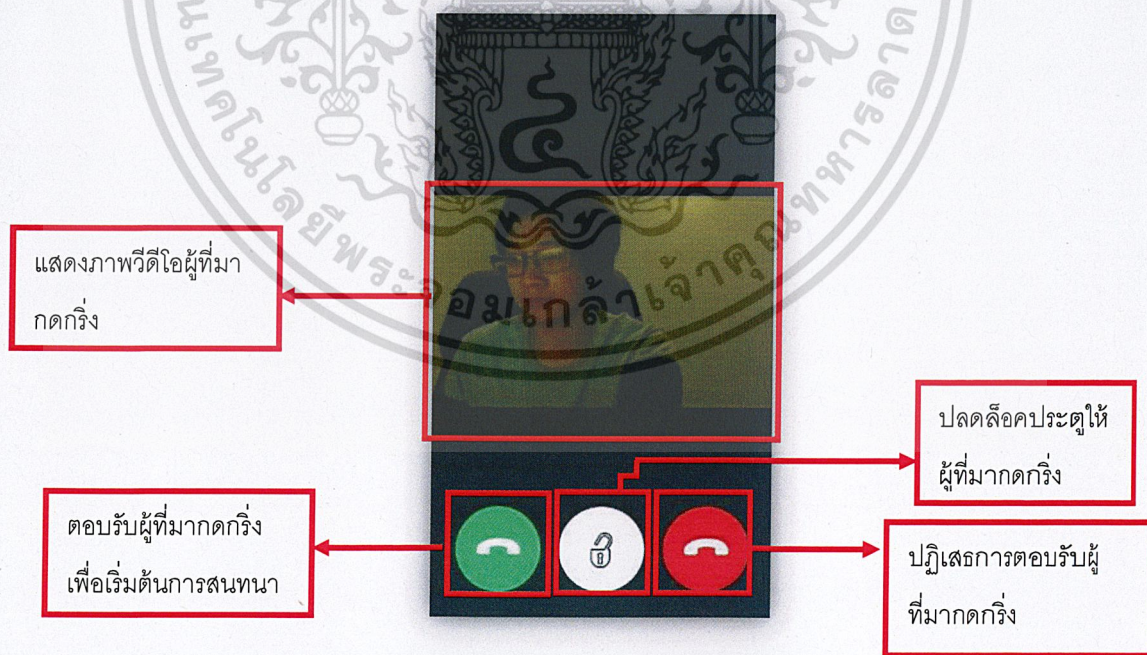
รูปที่ ก.20 ฟังก์ชันตั้งค่าอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



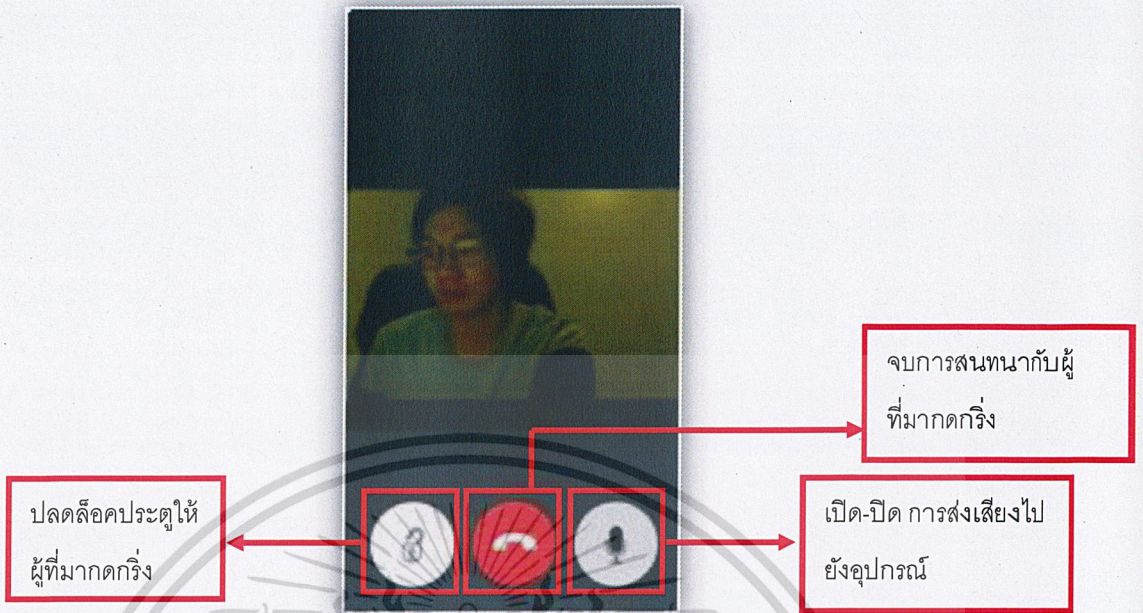
รูปที่ ก.21 หน้าจอฟังก์ชันตั้งค่าอุปกรณ์

13) หน้าจอแสดงผลเมื่อมีการกดกริ่ง



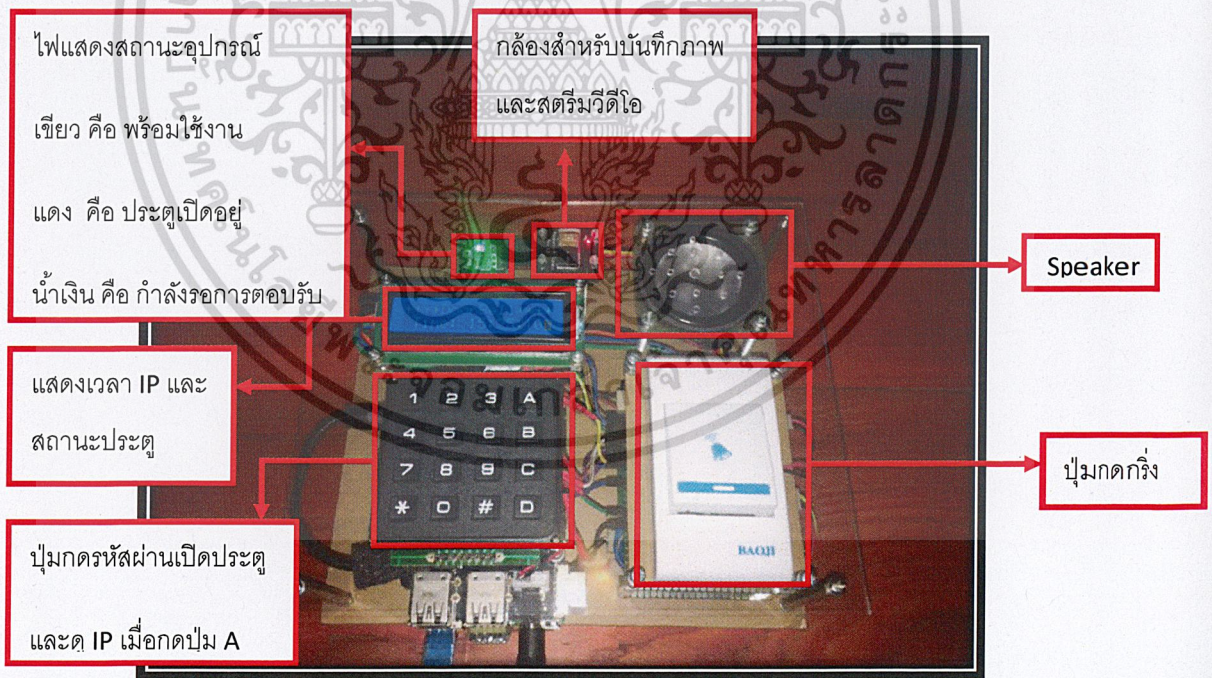
รูปที่ ก.22 หน้าจอแจ้งเตือนเมื่อมีการกดกริ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.23 หน้าจอเมื่อตอบรับผู้ที่มากดกริ่ง

ก.3 คู่มือการใช้งานอุปกรณ์กริ่งอัจฉริยะ



รูปที่ ก.24 อุปกรณ์กริ่งอัจฉริยะ

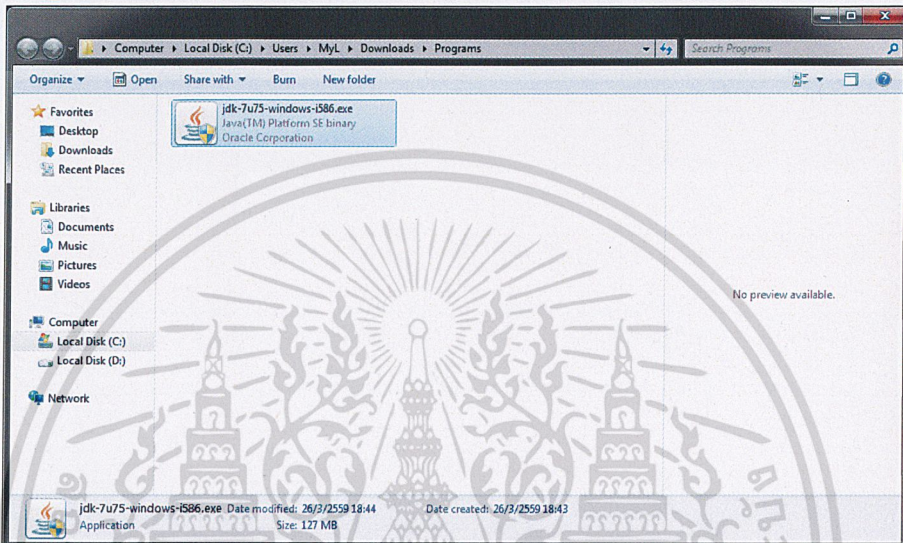
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ข. ขั้นตอนการติดตั้ง “Java JDK 7u75”

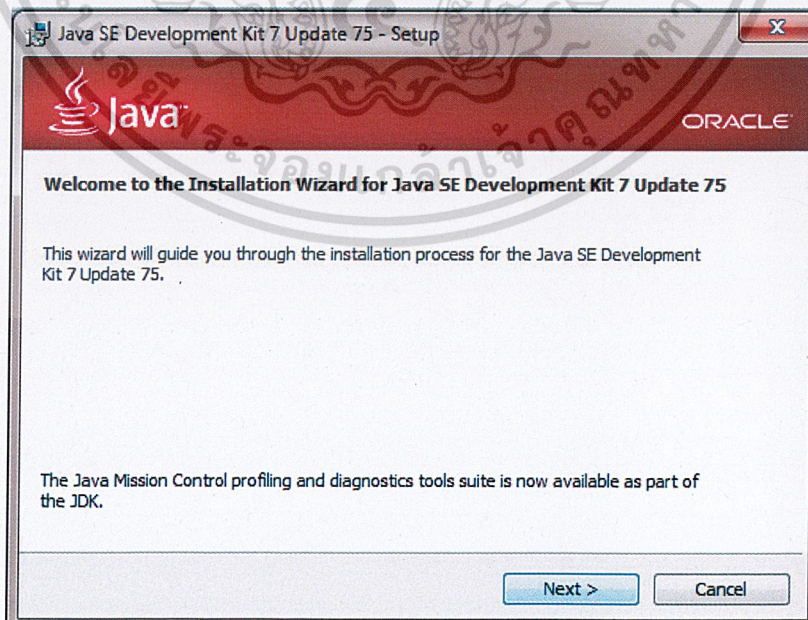
1) ดับเบิลคลิกที่ไฟล์ “jdk-7u75-windows-i586.exe” (ดาวน์โหลดได้จาก Link:

<https://java.com/en/download/>)



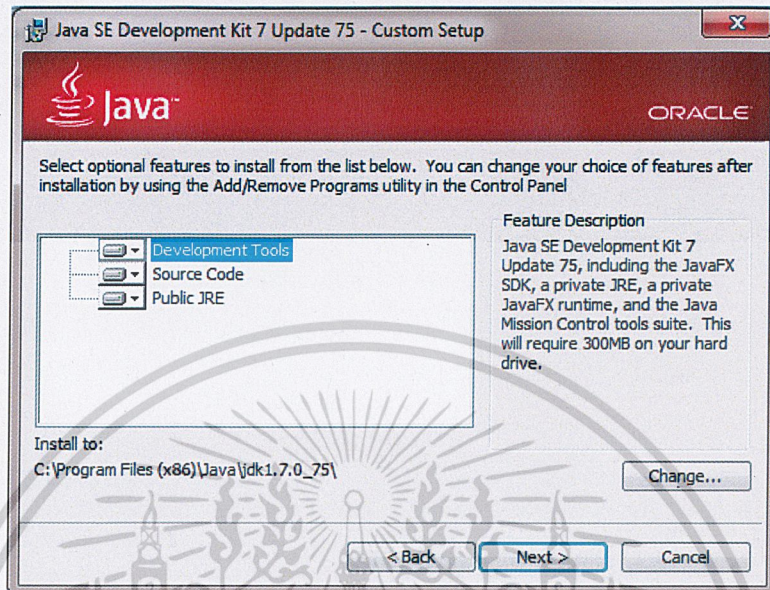
รูปที่ ข.1 ไฟล์ติดตั้ง “Java JDK 7u75”

2) ระบบจะแสดงข้อความต้อนรับ ถ้าต้องการติดตั้งให้กดปุ่ม “Next >”



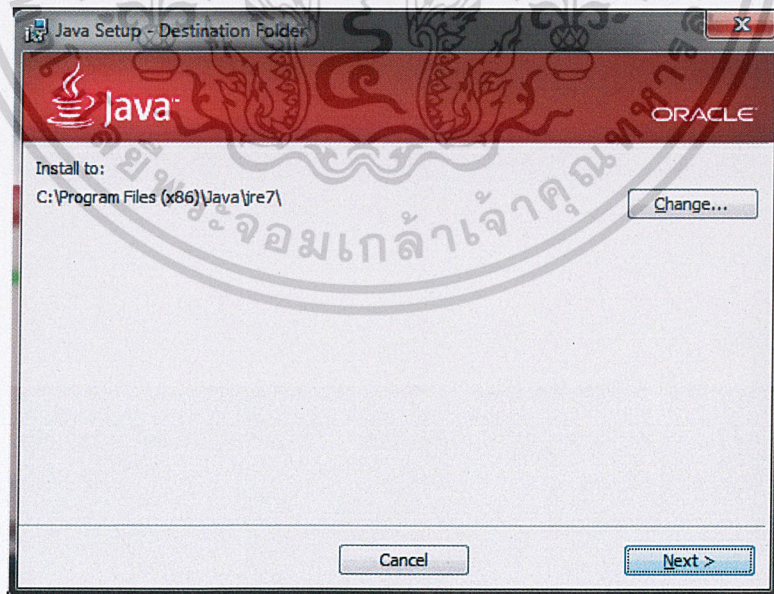
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับคุณเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ ข.2 เริ่มต้นการติดตั้ง “Java JDK 7u75”
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ระบบจะแสดงข้อความอธิบายต่างๆของ Java7 และให้เลือกสิ่งที่ต้องการติดตั้งหลังจากนั้นกดปุ่ม “Next >” ทำขั้นตอนถัดไป



รูปที่ ข.3 คำอธิบายต่างๆเกี่ยวกับ “Java7 JDK 7u75”

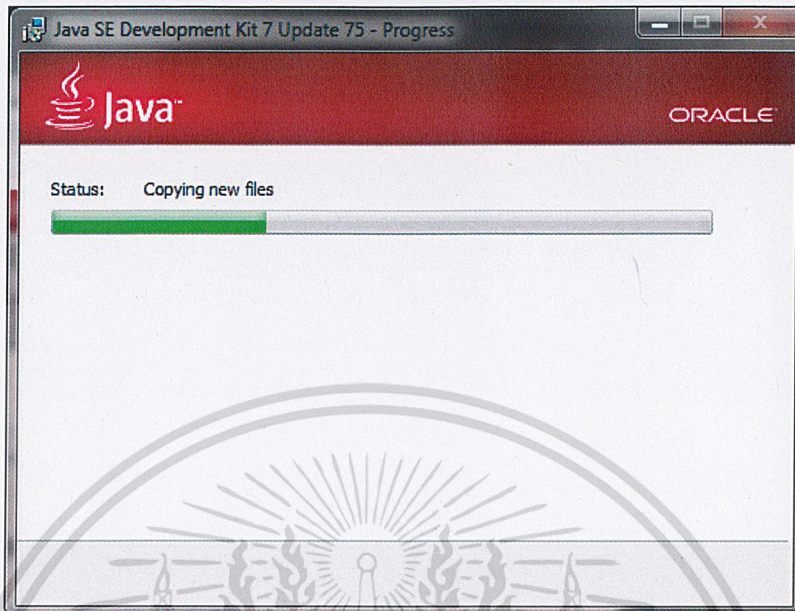
4) ระบบจะให้เลือก Directory “Java7 JDK 7u75” จากนั้นกดปุ่ม “Next >” ทำขั้นตอนถัดไป



รูปที่ ข.4 เลือก Directory “Java7 JDK 7u75”

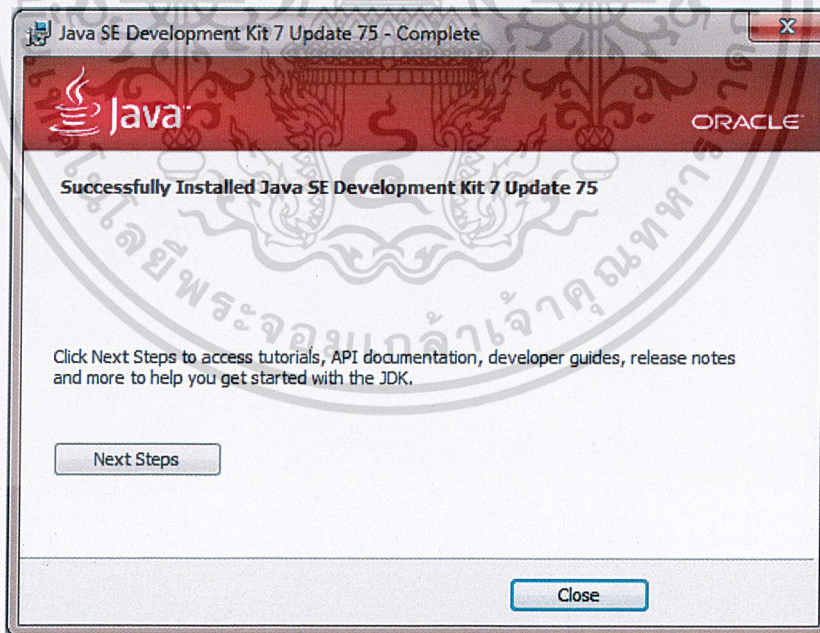
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) เริ่มการติดตั้งโปรแกรม “Java7 JDK 7u75”



รูปที่ ข.5 เริ่มการติดตั้ง “Java7 JDK 7u75”

6) เสร็จสิ้นการติดตั้ง “Java7 JDK 7u75” กดปุ่ม “Close” จบการทำงาน



รูปที่ ข.6 เสร็จสิ้นการติดตั้ง “Java7 JDK 7u75”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

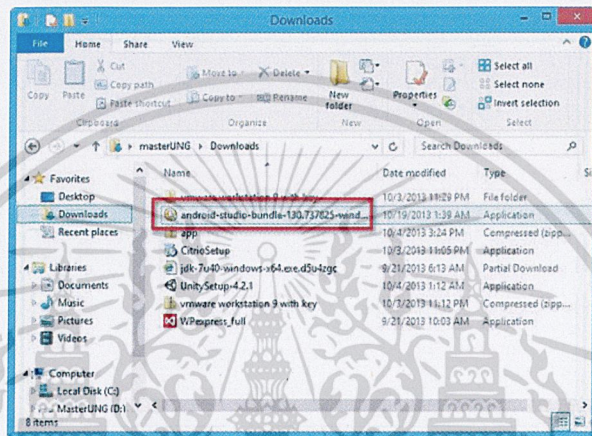
ภาคผนวก ค.

ค. การติดตั้งโปรแกรม Android Studio

1) ดาวน์โหลดโปรแกรมติดตั้ง Android Studio ได้ที่

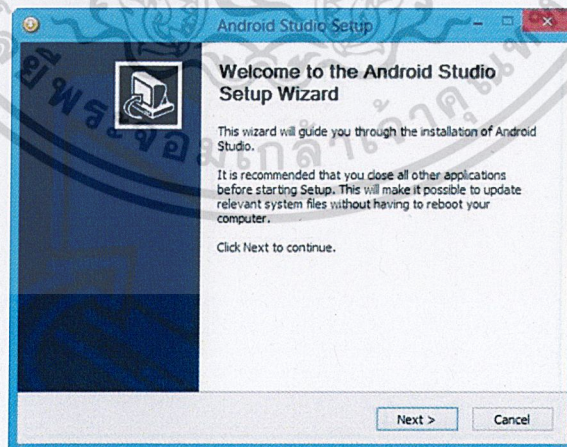
<http://developer.android.com/sdk/index.html>

2) ทำการเปิดโปรแกรมติดตั้ง Android Studio ที่ดาวน์โหลดมา



รูปที่ ค.1 ตัวอย่างไฟล์โปรแกรม android studio หลังจากทำการดาวน์โหลด

3) เมื่อเปิดไฟล์ดังกล่าวจะพบหน้าจอ Welcome ให้กดปุ่ม Next เพื่อดำเนินการต่อ

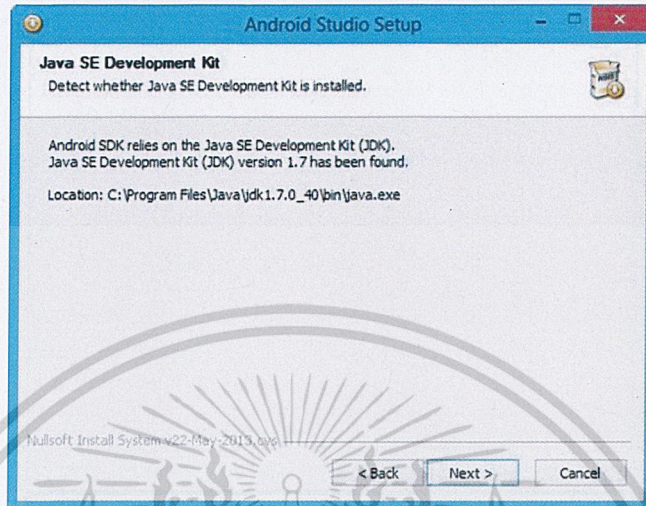


รูปที่ ค.2 หน้าจอตอนรับเมื่อเริ่มทำการติดตั้งโปรแกรม

4) โปรแกรมติดตั้ง Android Studio จะทำการค้นหา JDK (Java Developer Kit) ให้

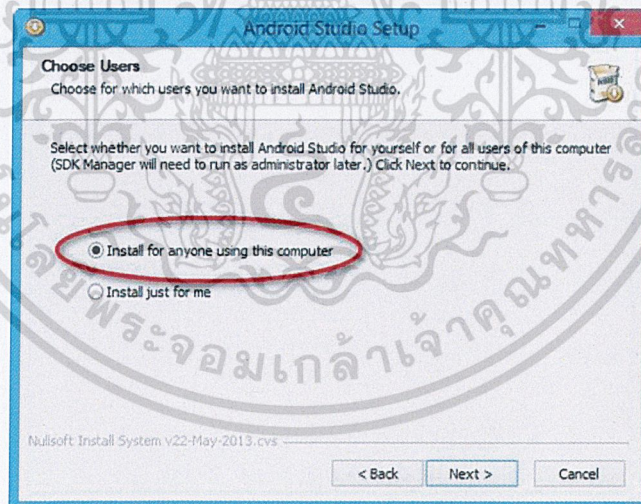
เอกสารนี้เจ้านั้นให้คลิกที่ปุ่ม Next (ในกรณีที่ไม่ได้ติดตั้ง JDK ไว้ จะต้องดาวน์โหลดที่นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html> เพื่อมาติดตั้งบนเครื่องของผู้ใช้ก่อน) จากนั้นกดปุ่ม Next



รูปที่ ค.3 หน้าจอค้นหาตำแหน่ง JDK บนเครื่อง

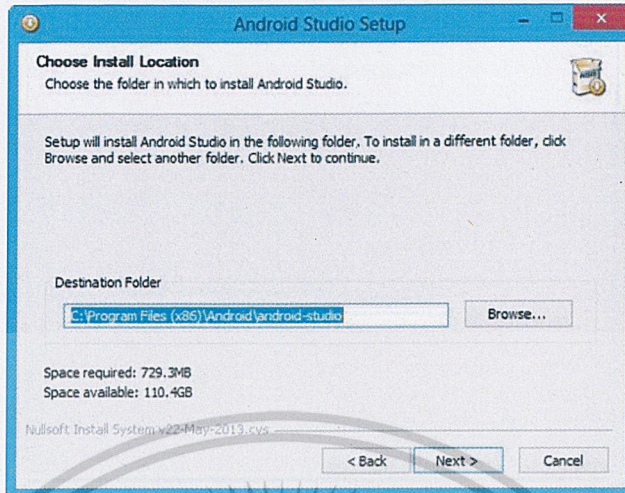
5) เลือก Install for anyone using this computer



รูปที่ ค.4 หน้าจอเลือกประเภทการติดตั้ง

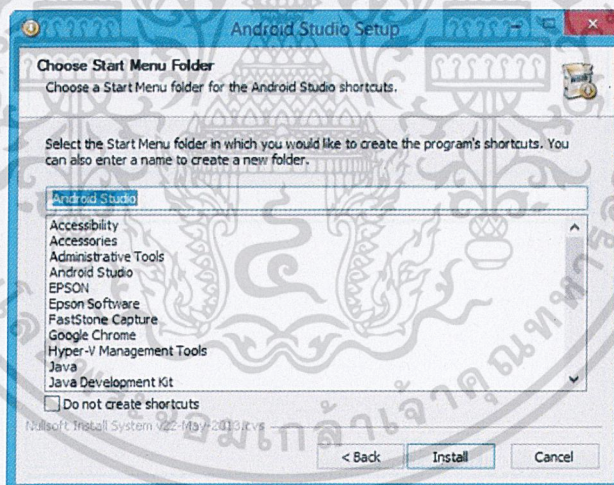
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) เลือกที่อยู่ไฟล์สำหรับติดตั้งโปรแกรม Android Studio จากนั้นคลิกที่ปุ่ม Next



รูปที่ ค.5 หน้าจอเลือกตำแหน่งที่จะทำการติดตั้งโปรแกรม

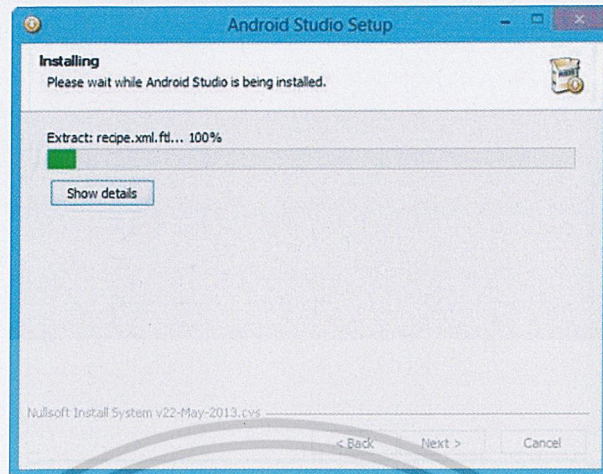
7) กดที่ปุ่ม Install เพื่อเริ่มการติดตั้งโปรแกรม



รูปที่ ค.6 หน้าจอจัดการ shortcut ของโปรแกรม

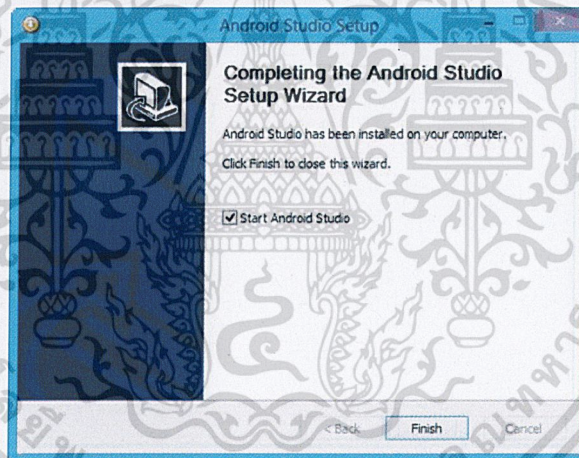
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) รอการติดตั้งโปรแกรม



รูปที่ ค.7 หน้าจอการติดตั้งโปรแกรม

9) คลิกที่ปุ่ม Finish เพื่อสิ้นสุดการติดตั้งโปรแกรม Android Studio



รูปที่ ค.8 หน้าจอแสดงผลการติดตั้งเสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้