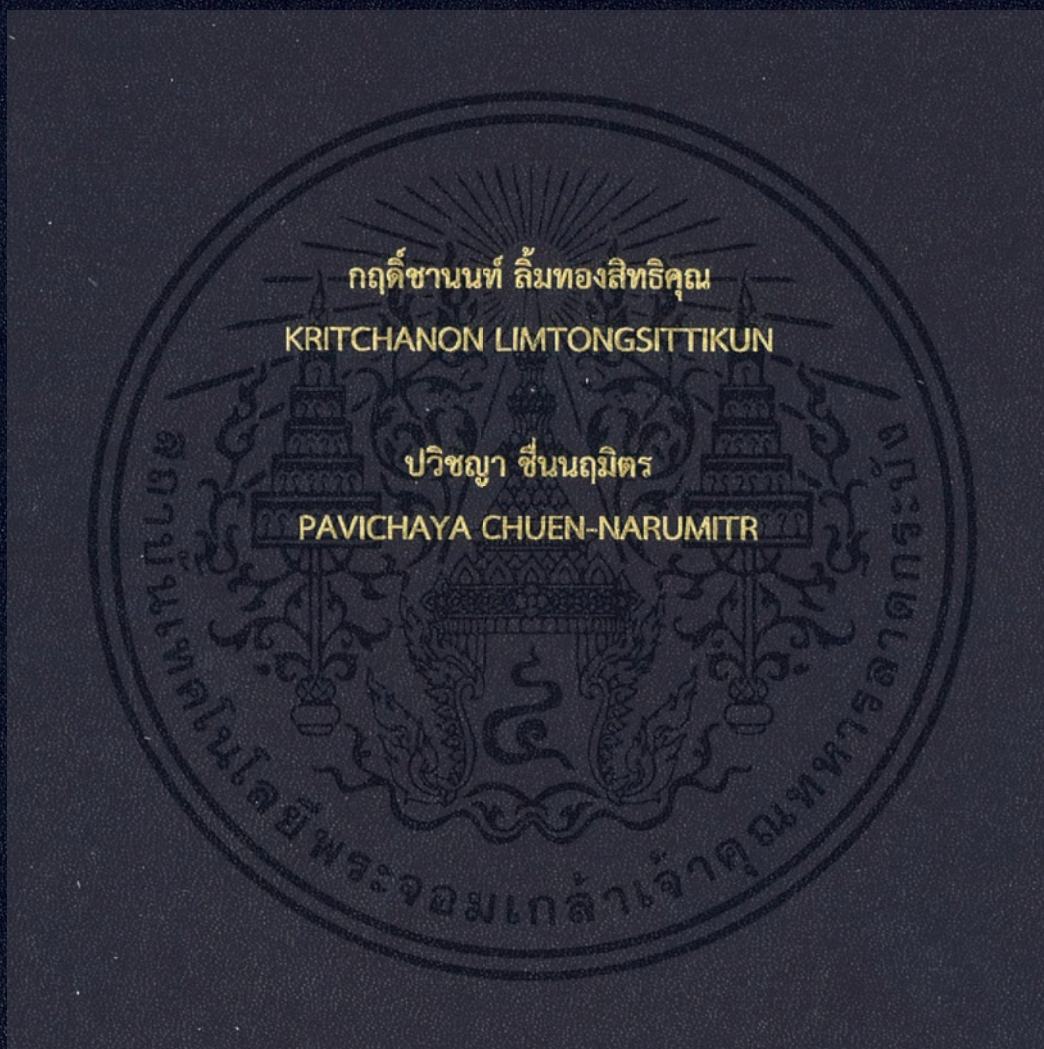


ระบบตรวจสินค้าในชั้นวางสินค้าของห้างสรรพสินค้า
DEPARTMENT STORE SHELF PRODUCT CHECKING SYSTEM



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2561

ระบบตรวจสินค้าในชั้นวางสินค้าของห้างสรรพสินค้า
DEPARTMENT STORE SHELF PRODUCT CHECKING SYSTEM



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DEPARTMENT STORE SHELF PRODUCT CHECKING SYSTEM



THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INFORMATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2018

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญาบัตร	ระบบตรวจสินค้าในชั้นวางสินค้าของห้างสรรพสินค้า	
รายนามนักศึกษา	นายกฤดีชานนท์ ลิ้มทองสิทธิคุณ	รหัสนักศึกษา 58010027
	นางสาวปวิษฐา ชื่นนฤมิตร	รหัสนักศึกษา 58010759
ปริญญา	วิศวกรรมศาสตรบัณฑิต	
สาขาวิชา	วิศวกรรมสารสนเทศ	
พ.ศ.	2561	
อาจารย์ที่ปรึกษาปริญญาบัตร	รศ.ดร.ชวลิต เบญจางคประเสริฐ	

ปริญญาบัตรฉบับนี้ ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรวิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการ	ระบบตรวจสอบสินค้าในชั้นวางสินค้าของห้างสรรพสินค้า		
รายชื่อนักศึกษา	นายกฤตวิชานนท์ ลิ้มทองสิทธิคุณ	รหัสนักศึกษา	58010027
	นางสาวปวิษฐา ชื่นนฤมิตร	รหัสนักศึกษา	58010759
ระดับปริญญา	วิศวกรรมศาสตรบัณฑิต		
สาขาวิชา	วิศวกรรมสารสนเทศ		
ภาควิชา	วิศวกรรมคอมพิวเตอร์		
ปีการศึกษา	2561		
อาจารย์ที่ปรึกษาปริญญานิพนธ์	รศ.ดร.ชวลิต เบญจางคประเสริฐ		

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ศึกษาและพัฒนาแอปพลิเคชันช่วยออกแบบระบบแจ้งเตือนเมื่อสินค้าในชั้นวางของหมด เพื่ออำนวยความสะดวกและเป็นแนวทางการออกแบบให้แก่ผู้ที่ต้องการเปิดร้านสะดวกซื้อขนาดเล็กถึงขนาดใหญ่ แต่ต้องการลดจำนวนพนักงานที่ใช้ตรวจสอบสินค้าในชั้นวางของ เพื่อลดต้นทุนของร้าน และเพิ่มความสะดวกให้กับลูกค้าเวลาสินค้าบนชั้นวางของหมด ลูกค้าไม่จำเป็นต้องเดินไปแจ้งพนักงาน โดยนำเทคโนโลยี IoT (Internet of things) มาประยุกต์ใช้ในการจัดการภายในร้าน เพื่อให้เกิดความสะดวกรวดเร็ว และมีประสิทธิภาพมากขึ้น โดยระบบจะมีการติดตั้งเซนเซอร์ไว้ที่ชั้นวางของ เพื่อตรวจจับสินค้าเมื่อสินค้าหมด จะมีการแจ้งเตือนผ่านแอปพลิเคชัน เพื่อให้พนักงานไปเติมสินค้าโดยไม่จำเป็นต้องเดินตรวจสอบสินค้าตลอดเวลา

Project Title	Department Store Shelf Product Checking System.		
Student	Mr.Kritchanon Limtongsittikun	Student ID.	58010027
	Ms.Pavichaya Chuen-narumitr	Student ID.	58010759
Degree	Bachelor of Engineering		
Program	Information Engineering		
Department	Computer Engineering		
Academic Year	2018		
Project Advisor	Assoc.Prof. Dr.Chawalit Benjangkprasert		

ABSTRACT

The objective of the project is to study and develop a web application by design a notification system when the product runs out of shelves. It facilitates and design a solution for those who want to open a small to large convenience store with less number of employees for checking the product on the shelves, reduce costs, and more convenient for customers. When the products on the shelves are empty, the customers do not need to walk to the staff. IoT (Internet of things) is applied in the shop management for quick, easy access, and also more effective. The system will be equipped with sensors on the shelves to detect when the products are exhausted. There will alert through the application to allow employees to refill without walking to check the product.

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา รศ.ดร.ชวลิต เบญจางคประเสริฐ ที่ให้ความช่วยเหลือ ให้คำชี้แนะช่วยแก้ปัญหาตลอดจน ให้ความรู้และถ่ายทอดประสบการณ์ที่ดีแก่คณะผู้จัดทำในตลอดระยะเวลาการศึกษา

ขอกราบขอบพระคุณ ผศ.มยุรี เลิศเวชกุล และ ผศ.ชลชัย สุขเจริญผล ที่คอยให้คำปรึกษาและให้คำชี้แนะ แนวทางการทำโครงการ

ขอขอบคุณ เพื่อน ๆ และพี่ ห้อง 1216 ร่วมทุกข์ร่วมสุข และการช่วยเหลือต่าง ๆ ตลอดการทำโครงการ

สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ คณะผู้จัดทำขอมอบให้กับบิดามารดา ซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านได้ประสิทธิ์ประสาทวิชาความรู้และถ่ายทอดประสบการณ์แก่ข้าพเจ้า



นายกฤดีชานนท์ ลีมหองสิทธิคุณ

นางสาวปวิษฐา ชีนนฤมิตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ.....	i
ABSTRACT.....	ii
กิตติกรรมประกาศ.....	iii
สารบัญตาราง.....	vii
สารบัญรูป.....	viii
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบเขตของโครงการ.....	1
1.4 ขั้นตอนการดำเนินงานโครงการ.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	3
บทที่ 2 ทฤษฎีและเครื่องมือที่เกี่ยวข้อง.....	4
2.1 Internet of Things.....	4
2.1.1 ความหมาย.....	4
2.1.2 ประโยชน์ของ Internet of Things.....	5
2.1.3 องค์ประกอบพื้นฐาน.....	5
2.2 อุปกรณ์ที่ใช้ในโครงการ.....	6
2.2.1 Arduino mega 2560.....	6
2.2.2 Load cell sensor.....	8
2.2.3 arduino node esp8266.....	9
2.2.4 Ultrasonic sensor HC-SR04.....	10
2.3 Arduino.....	11
2.3.1 ส่วนที่เป็น Hardware.....	11
2.3.2 ส่วนที่เป็น Software.....	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.4 Firebase.....	13
2.4.1 Firebase มีบริการอะไรให้บ้าง	13
2.5 android studio.....	14
2.6 Node JS.....	15
2.6.1 Node.js คืออะไร.....	15
2.6.2 Node.js เอาไปต่อยอดอะไรได้บ้าง.....	15
2.6.3 วิธีติดตั้ง Node.js	15
2.7 Heroku.....	16
2.8 Github.....	17
2.8.1 Version Control สำคัญยังไง?.....	17
2.8.2 Git ทำงานยังไง?.....	19
บทที่ 3 การออกแบบ.....	23
3.1 หลักการทำงานและภาพรวมของระบบ.....	23
3.2 ผังงาน (Flowchart).....	24
3.2.1 ผังการทำงานของแอปพลิเคชัน.....	24
3.2.2 ผังการทำงานของ Arduino Mega 2560.....	25
3.2.3 ผังการทำงานของ Function getDistance ของ Arduino Mega 2560.....	26
3.2.4 ผังการทำงานของ Function getWeight ของ Arduino Mega 2560.....	27
3.2.5 ผังการทำงานของ Node MCU ESP8266.....	28
3.2.6 ผังการทำงานของ Function initial ของ Node MCU ESP8266.....	30
3.2.7 ผังการทำงานของ Function setupWifi ของ Node MCU ESP8266.....	30
3.2.8 ผังการทำงานของ Function setupFirebase ของ Node MCU ESP8266	31
3.2.9 ผังการทำงานของ Function setDataFirebase ของ Node MCU ESP8266.....	32
บทที่ 4 การทำงานของโปรแกรม.....	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
4.1 การเข้าสู่หน้าหลัก.....	33
4.2 การเข้าสู่เมนู Admin	35
4.2.1 การเข้าสู่ระบบเมื่อไม่ได้เป็นสมาชิก.....	36
4.2.2 เมื่อ log in เข้าสู่ระบบได้แล้ว.....	37
4.3 การเข้าสู่เมนู Staff.....	42
บทที่ 5 สรุปผลการดำเนินโครงการ	45
5.1 สรุปผลการทดลอง.....	45
5.2 ปัญหาและอุปสรรคของการดำเนินโครงการ.....	45
5.3 แนวทางการพัฒนาโครงการ.....	45
เอกสารอ้างอิง.....	46
ภาคผนวก	48
ภาคผนวก ก.....	49
ภาคผนวก ข.....	51
ภาคผนวก ค.....	60
ภาคผนวก ง.....	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
1.1 แผนการดำเนินงานโครงการ โปรเจกต์ 1.....	2
1.2 แผนการดำเนินงานโครงการ โปรเจกต์ 2.....	3



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 ตัวอุปกรณ์ Arduino mega 2560.....	6
2.2 ตัวอุปกรณ์ load cell.....	8
2.3 ตัวอย่างการทำงานของ load cell.....	8
2.4 ตัว node mcu.....	9
2.5 หลักการตรวจจับและวัดระยะห่างระหว่างวัตถุด้วยคลื่นเสียง.....	10
2.6 การต่อใช้งานโมดูลอัลตราโซนิก HC-SR04.....	11
2.7 ตัวอย่าง Arduino.....	12
2.8 หน้าโปรแกรม Arduino.....	12
2.9 4 Developer กับ 4 Feature.....	17
2.10 การรวมโค้ดของนาย A และ B เข้าด้วยกันก่อน ส่วนนาย C กับ D นั่งรอ.....	18
2.11 การทำงานแบบ Distributed Version Control Systems.....	19
2.12 Server กลางตัวหนึ่งที่คอยเก็บข้อมูลจากผู้ใช้แต่ละเครื่องก่อน.....	19
2.13 แต่ละเครื่องทำงานเป็น VCS.....	20
2.14 Server หลายตัวทำเป็น VCS.....	20
2.15 ใครไววกว่าก็สบายไป.....	21
2.16 การดึงโค้ดมารวมกับโค้ดของตัวเองให้เรียบร้อยก่อน ถึงจะ Sync ขึ้น Remote ได้.....	22
3.1 ภาพรวมการทำงาน.....	23
3.2 ผังการทำงานของแอปพลิเคชัน.....	24
3.3 ผังการทำงานของ Arduino Mega 2560.....	25
3.4 ผังการทำงานของ Function getDistance ของ Arduino Mega 2560.....	26
3.5 ผังการทำงานของ Function getWeight ของ Arduino Mega 2560.....	27
3.6 ผังการทำงานของ Node MCU ESP8266.....	29
3.7 ผังการทำงานของ Function initial ของ Node MCU ESP8266.....	30
3.8 ผังการทำงานของ Function sertupWifi ของ Node MCU ESP8266.....	30
3.9 ผังการทำงานของ Function setupFirebase ของ Node MCU ESP8266.....	31
3.10 ผังการทำงานของ Function setDataFirebase ของ Node MCU ESP8266.....	32
4.1 หน้าแอปพลิเคชัน.....	33
4.2 หน้าหลักของโปรแกรม.....	34
4.3 หน้าเมนูย่อยเมื่อเข้ามาสู่เมนู Admin.....	35
4.4 หน้าสมัครสมาชิกของ Admin.....	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.5 หน้าโปรแกรมเมื่อเข้าสู่ระบบ Admin	37
4.6 หน้าการแก้ไขข้อมูลสินค้า	38
4.7 หน้าเมนูย่อยของระบบ Admin.....	39
4.8 หน้าหน้าต่าง Manage Staff.....	40
4.9 หน้าหน้าต่าง Manage Staff และการลบข้อมูล Staff.....	41
4.10 หน้าระบบ Staff.....	42
4.11 หน้าโปรแกรมเมื่อเข้าสู่ระบบ Staff.....	43
4.12 หน้าเมนูย่อยของระบบ Staff ที่ไม่มีหน้าต่าง Manage Staff.....	44
ข.1 หน้า ค้นหาการ Download Arduino.....	52
ข.2 หน้า เว็บไซต์ Download Arduino.....	52
ข.3 หน้า Download Arduino.....	53
ข.4 หน้า Download เสร็จ.....	53
ข.5 หน้า ไฟล์สำหรับติดตั้งโปรแกรม Arduino IDE.....	54
ข.6 หน้า หน้าต่างสำหรับติดตั้ง.....	54
ข.7 หน้า install option.....	55
ข.8 หน้า เลือกที่ติดตั้งโปรแกรม.....	55
ข.9 หน้า รอกการติดตั้ง.....	56
ข.10 หน้า Install เพื่อติดตั้ง Adafruit Industries LLC Ports.....	56
ข.11 หน้า Install เพื่อติดตั้ง Arduino USB Driver srl.....	57
ข.12 หน้า Install เพื่อติดตั้ง Arduino USB Driver LLC.....	57
ข.13 หน้า Completed	58
ข.14 หน้าไอคอน Arduino.....	58
ข.15 หน้า ข้างในโปรแกรม Arduino	59
ค.1 หน้า Download Java JDK.....	61
ค.2 Install JDK.....	62
ค.3 Install JDK Fin.....	62
ค.4 Set PATH.....	63
ค.5 Set Java Home.....	63
ค.6 Finish setup JDK.....	64
ค.7 Android Studio	64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
ค.8 Install Android Studio Install Android Studio(1).....	65
ค.9 Install Android Studio Install Android Studio(2).....	65
ค.10 Install Android Studio Install Android Studio(3).....	66
ค.11 Install Android Studio Install Android Studio(4).....	66
ค.12 Completed.....	67
ค.13 Error can't find JDK.....	67
ค.14 Install Component.....	68
ค.15 Run As Administrator.....	68
ง.1 Default rules in Firebase Realtime Database.....	70
ง.2 Run default rules ด้วย simulator.....	70
ง.3 Run public rules ด้วย simulator.....	71
ง.4 ตัวอย่างการ setValue().....	72
ง.5 ตัวอย่างการ push().....	73
ง.6 write ข้อมูลเข้าไปถึง 2 objects.....	74
ง.7 update ข้อมูลบางส่วนทั้ง 2 objects.....	74



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

ในปัจจุบัน ทุกสิ่งทุกอย่างส่วนใหญ่ถูกเชื่อมต่อด้วยอินเทอร์เน็ตทั้งสิ้น และโทรศัพท์มือถือมีบทบาทกับวิถีชีวิตมากขึ้น จึงได้เกิดแนวคิดการนำ Internet of thing มาประยุกต์ใช้ร่วมกับธุรกิจการค้าขาย เพื่อเพิ่มความสะดวกสบายในการบริหารและการบริการเพิ่มมากขึ้น

นิยามของ Internet of thing คือการที่สิ่งต่าง ๆ ถูกเชื่อมโยงด้วยอินเทอร์เน็ต สามารถนำมาประยุกต์ใช้ได้กับธุรกิจการค้าได้หลากหลาย เช่น ในห้างสรรพสินค้า เมื่อสินค้าในชั้นวางสินค้าหมด ก็ไม่สามารถรับรู้ได้ว่าหมดตอนไหน หรือเวลาใด ถ้านำแนวคิดของ Internet of thing เข้ามาใช้ เมื่อสินค้าในชั้นวางสินค้าใกล้จะหมดจะมีการแจ้งเตือนและแสดงตำแหน่งไปที่แอปพลิเคชันทันที และนอกจากนั้นยังทำให้เราสามารถวิเคราะห์พฤติกรรมของลูกค้าได้อีก เมื่อสินค้าชนิดนั้นหมดไปอย่างรวดเร็ว สามารถส่งไปให้นักวิเคราะห์การตลาดต่อยอดไปได้เรื่อย ๆ

จึงได้เกิดการคิดค้นแอปพลิเคชันในการแจ้งเตือน และเก็บข้อมูลขึ้นมา เพื่อนำไปใช้ประโยชน์กับธุรกิจการค้าขายต่อไป

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อสร้างโครงการที่เป็นประโยชน์ต่อร้านสะดวกซื้อ และ ห้างสรรพสินค้า
2. เพื่อเพิ่มความสะดวกให้กับพนักงานขายในร้านสะดวกซื้อให้ไม่ต้องเดินตรวจสินค้าตลอดเวลา
3. เพื่อให้มีสินค้ากับลูกค้า

1.3 ขอบเขตของโครงการ

1. สร้างชั้นวางของที่สามารถตรวจสอบได้ว่ามีสินค้าเหลืออยู่ในชั้นวางของหรือไม่
2. สร้างแอปพลิเคชันแจ้งเตือนเมื่อสินค้าในชั้นวางของหมด
3. โปรแกรมสามารถบอกตำแหน่งและชื่อสินค้าที่พนักงานต้องเอาสินค้าชนิดนั้นไปเติมที่ชั้นวางของได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขั้นตอนการดำเนินงานโครงการงาน

โปรเจกต์ 1

1. ศึกษาค้นคว้าข้อมูลที่ใช้ในการออกแบบระบบ
2. ทำการออกแบบระบบวางแผนการ
3. คิดค้นและพัฒนาส่วนจัดเก็บข้อมูลของ

ตารางที่ 1.1 แผนการดำเนินงานโครงการงาน โปรเจกต์ 1

ลำดับ	ขั้นตอนการดำเนินงาน	ปี 2561				
		ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.
1	เสนอโครงการ และหาอุปกรณ์ที่ต้องใช้ทำ					
2	ศึกษาโปรแกรม Arduino					
3	ศึกษา sensor ที่ใช้งาน					
4	ทำโครงชั้นวางสินค้า					
5	ทำให้ค่าจาก sensor ส่งขึ้น sever ได้					
6	ทดลองและประเมินผลการทำงานของโปรแกรม					
7	เขียนโปรแกรมคำสั่งเชื่อมต่อระบบ					
8	ตรวจสอบและแก้ไขปัญหา					
9	ทำการทดลองและตรวจสอบผลอีกครั้ง					
10	สรุปผลการทดสอบของระบบและทำโครงการวิจัย					

โปรเจกต์ 2

1. พัฒนาอัลกอริทึมที่ใช้ในระบบวางแผนการ
2. ทดสอบการทำงานของระบบวางแผนการ
3. ปรับปรุง แก้ไข ทดสอบการทำงานในส่วนต่าง ๆ
4. สรุปผลการทดสอบ

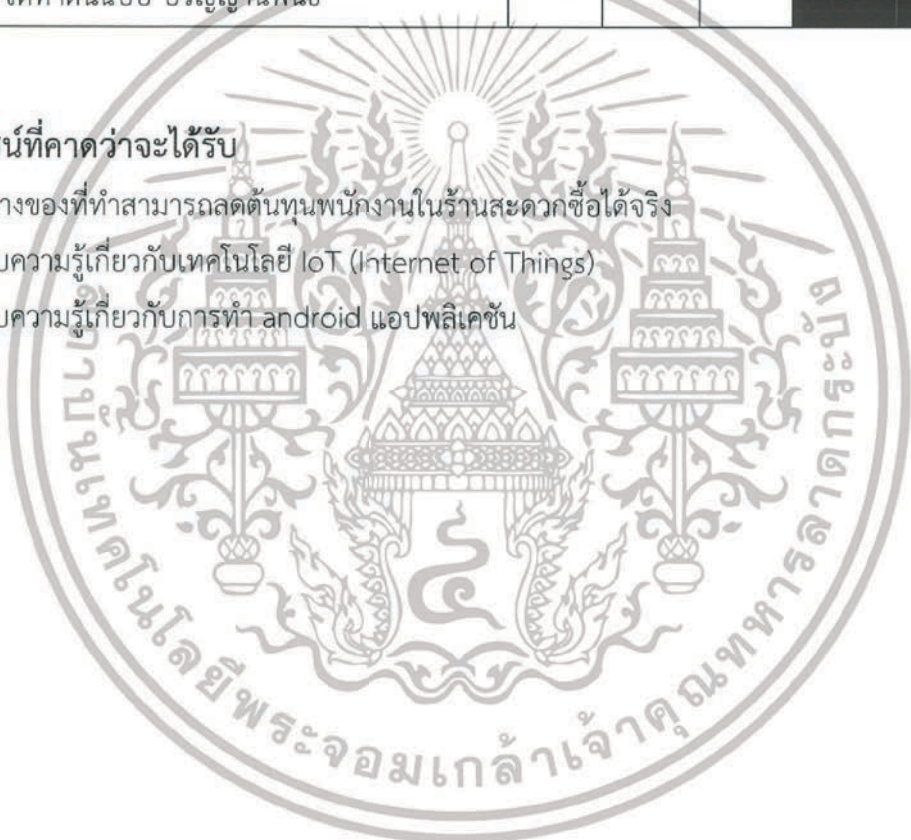
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1.2 แผนการดำเนินงานโครงการ โปรเจกต์ 2

ลำดับ	ขั้นตอนการดำเนินงาน	ปี 2562				
		ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.
1	ศึกษาและทำ app android					
2	ดึงค่าจาก server เข้า app android					
3	ตกแต่งแก้ไข bug ต่าง ๆ					
4	ปรับปรุงและแก้ไขปัญหา					
5	ทำการทดลองและตรวจสอบผลอีกครั้ง					
6	สรุปผลการทดสอบ					
7	จัดทำต้นฉบับ ปรินตยูนิฟอนท์					

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ชีวภาพของที่สามารถลดต้นทุนพนักงานในร้านสะดวกซื้อได้จริง
2. ได้รับความรู้เกี่ยวกับเทคโนโลยี IoT (Internet of Things)
3. ได้รับความรู้เกี่ยวกับการทำ android แอปพลิเคชัน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและเครื่องมือที่เกี่ยวข้อง

ในการจัดสร้างโครงการ ระบบตรวจสินค้าในชั้นวางสินค้าของห้างสรรพสินค้า จำเป็นจะต้องศึกษาทฤษฎีที่เกี่ยวข้อง ดังต่อไปนี้

2.1 Internet of Things

2.1.1 ความหมาย

สำหรับอินเทอร์เน็ตเพื่อสรรพสิ่งนั้นมีคำสำคัญสองคำ คือคำว่า “Internet” คือ ระบบเครือข่ายคอมพิวเตอร์ขนาดใหญ่ที่เชื่อมต่อและสื่อสารจากคอมพิวเตอร์เครื่องหนึ่งไปยังอีกเครื่องหนึ่งได้ หรือจากเครือข่ายคอมพิวเตอร์หนึ่งไปยังอีกเครือข่ายคอมพิวเตอร์หนึ่งได้ ส่วนคำว่า “Thing” นั้นหมายถึง สรรพสิ่งทุกอย่าง วัตถุหรือสิ่งของ อุปกรณ์ต่าง ๆ เช่น เครื่องปรับอากาศ โทรทัศน์ โต้ะ เก้าอี้ ปากกา ดินสอ เสื้อผ้า รองเท้า ฯลฯ

Rajkumar Buyya, Amir Vahid Dastjerdi (2016) อธิบาย เกี่ยวกับ Internet of Things สรุปได้ว่าเป็นกระบวนการหนึ่ง (วิธีคิด วิธีปฏิบัติ ตัวแบบ รูปแบบ กรอบแนวความคิด และ แนวทางการศึกษา) ที่ว่าด้วยการนำอุปกรณ์อิเล็กทรอนิกส์ มาใช้ประโยชน์ให้สามารถเชื่อมต่อกับมนุษย์ได้ โดยอาศัยโครงสร้างพื้นฐานทางการสื่อสารโทรคมนาคมหรืออินเทอร์เน็ต เพื่อการใช้ประโยชน์จากทรัพยากรให้เกิดความคุ้มค่าสูงสุดรวมทั้งการบริการและเสริมสร้างคุณภาพชีวิตของมนุษย์ เช่น อุปกรณ์ทางการแพทย์ ตู้เย็น กล้องถ่ายภาพและเซ็นเซอร์ต่าง ๆ ที่เชื่อมผ่านเครือข่ายอินเทอร์เน็ต ซึ่งกระบวนการนี้จะนำไปสู่การสร้างสรรคนวัตกรรมที่สามารถทำให้มีปฏิสัมพันธ์ระหว่างสิ่งต่าง ๆ กับมนุษย์ สามารถทำได้ ง่ายขึ้น สะดวกขึ้น

Robert Lutz (2016) ได้อธิบายเกี่ยวกับ Internet of Things ว่าเป็นระบบที่จะเปลี่ยนแปลงวิถีชีวิตของมนุษย์ โดยเป็นระบบที่ให้วัตถุหรือสิ่งของสามารถสื่อสาร เชื่อมต่อ กับอุปกรณ์หรือเครื่องจักรอื่น ๆ ได้ Padraig Scully, Knud Lasse Luet (2016) โดยพื้นฐานแล้ว Internet of Things คือ แนวความคิดที่อธิบายการเชื่อมต่อ (Connecting) กับวัตถุทางกายภาพใด ๆ หรือ “สิ่ง (Thing)” ผ่านเครือข่ายอินเทอร์เน็ต ซึ่งการเชื่อมต่อกับ วัตถุต่าง ๆ แบบนี้ ส่งผลกระทบต่อสำคัญในการจัดการข้อมูล หรืออุปกรณ์จำนวนมากมาย ที่ต้องปรับเปลี่ยนให้สามารถเชื่อมต่อหรือสื่อสารกันได้ ดังนั้น Internet of Things จึงเป็นการนำอุปกรณ์อิเล็กทรอนิกส์มาฝังไว้ในสิ่งต่าง ๆ เพื่อเก็บรวบรวมและแลกเปลี่ยนข้อมูลต่าง ๆ สามารถสื่อสารหรือเชื่อมโยงผ่านเครือข่ายอินเทอร์เน็ต โดยจะไม่ติดต่อกับมนุษย์โดยตรง แต่จะมีอยู่ในสิ่งแวดล้อม อาคาร สถานที่ ต้นไม้ รถยนต์ ทุกอย่างสามารถเชื่อมต่อได้ ซึ่งบางครั้งเรียกว่า “Smart Objects”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 ประโยชน์ของ Internet of Things

ด้านการแพทย์ : ปัจจุบันวิทยาการทางการแพทย์ ได้มีการนำเทคโนโลยีเข้ามาช่วยด้วยมากขึ้น เช่น กล้องขนาดเล็กที่ส่งเข้าไปภายในร่างกายของคนไข้ ทำให้สามารถเห็นอวัยวะภายในได้ โดยไม่ต้องทำการผ่าตัด ซึ่งช่วยลดความเจ็บปวด และเวลาในการรักษาให้สั้นลง ซึ่งถ้าหากมีการนำ Internet of Thing เข้ามามีส่วนร่วมด้วย จะช่วยในด้านความสะดวกรวดเร็วในการติดต่อระหว่างแพทย์และคนไข้ได้อย่างรวดเร็วมากยิ่งขึ้นเช่น การฝังชิปไว้ในร่างกายผู้ป่วย ที่สามารถติดต่อแพทย์ให้อัตโนมัติเมื่อมีเหตุการณ์ผิดปกติ

ด้านการโฆษณา : การทำโฆษณาบนเครือข่ายอินเทอร์เน็ตนั้น นอกจากจะเข้าถึงกลุ่มเป้าหมายได้หลากหลายชั้นแล้ว ยังช่วยประหยัดต้นทุนในการเข้าพื้นที่โฆษณาได้อีกด้วย แต่การที่จะดูโฆษณาบนเครือข่ายอินเทอร์เน็ตได้นั้นหมายความว่า จะต้องอยู่หน้าจอกอมพิวเตอร์ หรือโทรศัพท์มือถือ แต่ถ้าหากนำแนวคิด Internet of Thing เข้ามาเสริมนั้น ระหว่างที่เดินผ่านหน้าร้านสินค้า ก็จะมีโฆษณาแสดงขึ้นมาโดยอัตโนมัติ ตัวอย่างเช่น หากมีผู้คนเดินผ่านหน้าร้านสินค้าของเรา (ซึ่งถูกตรวจจับได้โดยระบบเซ็นเซอร์) ก็จะปรากฏภาพโฆษณาขึ้นให้ผู้คนที่เดินผ่านไปมาได้เห็นทันที ซึ่งจะส่งผลให้สามารถเข้าถึงกลุ่มลูกค้าได้หลากหลายยิ่งขึ้น

ด้านการลดต้นทุน : เช่นการลดต้นทุนให้กับกรไฟฟ้า การที่ต้องมีพนักงานมาคอยตรวจเช็ค และจดมิเตอร์ไฟฟ้าในทุกเดือน ถือเป็นต้นทุนอย่างหนึ่งที่มีการไฟฟ้าต้องจ่าย เพื่อจ้างพนักงานให้คอยทำหน้าที่นี้ หากมีการนำแนวคิด Internet of Thing มาใช้ จะส่งผลให้สามารถตัดค่าใช้จ่ายในส่วนนี้ได้ เนื่องจากมิเตอร์จะทำการส่งข้อมูลไปยังระบบที่คอยบันทึกข้อมูลการใช้ไฟของการไฟฟ้าเอง โดยไม่ต้องใช้คนจด อีกทั้งยังช่วยลดการใช้ไฟฟ้าได้อีกด้วย จากการที่สามารถบอกอัตราการใช้ไฟของเครื่องใช้ไฟฟ้าแต่ละชนิด

2.1.3 องค์ประกอบพื้นฐาน

2.1.3.1 เทคโนโลยีไร้สาย (Wireless technologies) เป็นเทคโนโลยีที่มีการรับ-ส่งข้อมูลผ่านกระบวนการ ผสมสัญญาณคลื่นแม่เหล็กไฟฟ้าเป็นคลื่นพาห์ (Radio Frequency: RF) หรือคลื่นความถี่วิทยุเป็นพาห์และ คลื่นอินฟราเรด (Infrared) เป็นตัวกลางในการรับ-ส่ง ข้อมูลระหว่างคอมพิวเตอร์กับคอมพิวเตอร์ หรือระหว่าง คอมพิวเตอร์กับอุปกรณ์อื่น ๆ

2.1.3.2 ไมโครเทคโนโลยี (Micro Electro-Mechanical) เป็นเทคโนโลยีที่เกี่ยวข้องกับสิ่งที่มีขนาดเล็กมาก ๆ ไมโครเทคโนโลยีประกอบด้วยชิ้นส่วนที่มีขนาดระหว่าง 0.001 ถึง 0.1 มิลลิเมตร และจะมีหน่วยประมวลผลกลาง อุปกรณ์อื่น ๆ และไมโครเซนเซอร์ (Jean-Baptiste Waldner, 2008)

2.1.3.3 ไมโครเซอร์วิส (Microservices) เป็นการออกแบบสถาปัตยกรรมการออกแบบซอฟต์แวร์ โดยจะแบ่งซอฟต์แวร์ออกเป็นบริการหรือหน่วยย่อย ๆ แยกส่วน มีกระบวนการทำงานจัดเก็บ แก๊ซ ประมวลผลข้อมูล ตามความเหมาะสมของหน่วยย่อย ๆ นั้น และสามารถเชื่อมต่อหรือสื่อสารกับหน่วยย่อยอื่น ๆ ได้ ซึ่งคุณสมบัติสำคัญของไมโครเซอร์วิส คือ เนื่องด้วยในแต่ละหน่วยหรือในแต่ละ services ของซอฟต์แวร์มีอิสระต่อกัน จึงง่ายต่อการเปลี่ยนแปลง ปรับปรุง แก้ไข หากหน่วยใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกิดความเสียหาย ก็ยังมีหน่วยอื่น ๆ ที่สามารถทำงานได้และสามารถใช้งานได้หากมีการใช้ภาษาคอมพิวเตอร์ ฐานข้อมูล ฮาร์ดแวร์ และซอฟต์แวร์ หรือสภาพแวดล้อมอื่น ๆ ที่แตกต่างกัน รองรับผู้ใช้งานจำนวนมากทั้งนี้ขึ้นอยู่กับความเหมาะสมในการ เลือกใช้งาน

2.1.3.4 อินเทอร์เน็ต (Internet) คือ เครือข่ายคอมพิวเตอร์ที่มีการเชื่อมต่อสื่อสารรับ-ส่งข้อมูลระหว่าง เครือข่ายจำนวนมากจากทั่วโลกที่ทุกคนใช้งานกันอยู่ในปัจจุบันนั่นเอง ใช้เป็นแหล่งข้อมูล แหล่งรับ-ส่งข่าวสาร เป็นแหล่งให้ความบันเทิง ทำธุรกิจต่าง ๆ และเพื่อตอบสนองชีวิตประจำวันในด้านต่าง ๆ

2.2 อุปกรณ์ที่ใช้ในโครงการ

2.2.1 Arduino mega 2560

Arduino Mega 2560 คือบอร์ดไมโครคอนโทรลเลอร์ที่พัฒนาจาก ATmega2560 มี 54 digital input/output โดยมี 14 ขา สามารถใช้เป็น output แบบ PWM ได้ มี analog inputs 16 ขา มี UARTs(hardware serial ports) 4 ขา ทำงานที่ความถี่ 16 MHz สามารถเชื่อมต่อกับคอมพิวเตอร์ด้วยสายเคเบิล USB หรือใช้ adaptor AC-to-DC เพื่อเริ่มต้นใช้งาน และมีปุ่ม reset สามารถต่อเข้ากับ shields ที่ออกแบบเพื่อใช้งานกับ Arduino Duemilanove หรือ Diecimila.



รูปที่ 2.1 ตัวอุปกรณ์ Arduino mega 2560

(อ้างอิงโดย: https://www.gravitechthai.com/product_detail.php?d=55)

2.2.1.1 Pin ทั่วไป

1. VIN เป็น input voltage ของบอร์ด Arduino โดยใช้แหล่งจ่ายจากภายนอก
2. 5V เป็น output pin ที่ควบคุม 5 V จากบอร์ด
3. 3V3 เป็น 3.3 volt supply ที่สร้างขึ้นจาก regulator บนบอร์ด และให้กระแสไฟฟ้าสูงสุด 50 mA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. GND เป็น ground pin
5. IOREF เป็น pin ที่ให้ voltage reference กับไมโครคอนโทรลเลอร์ เพื่อเลือกค่าแรงดันให้กับ shield ที่มาเชื่อมต่อกับบอร์ด

2.2.1.2 หน่วยความจำ

ATmega2560 มีหน่วยความจำ 256 KB (8 KB ใช้สำหรับ bootloader) นอกจากนี้ยังมีอีก 8 KB สำหรับ SRAM และ 4 KB สำหรับ EEPROM

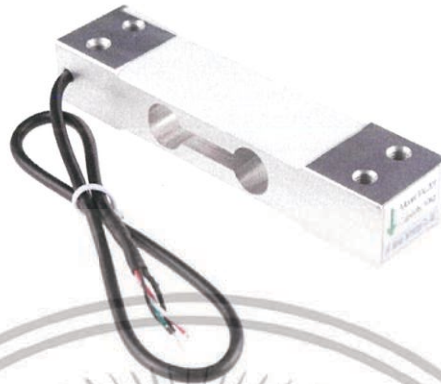
2.2.1.3 Input and Output

ในแต่ละ digital pins ทั้ง 54 pins บนบอร์ด Arduino Uno สามารถเป็นได้ทั้ง input และ output โดยจะทำงานที่แรงดัน 5 V และให้กระแสไฟฟ้าสูงสุด 40 mA

2.2.1.4 ฟังก์ชันอื่น ๆ

1. External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), 21 (interrupt 2). pins เหล่านี้สามารถที่จะกำหนดค่าที่เรียก interrupt ในค่าต่ำ ๆ ขอบขาขึ้นและลง หรือเปลี่ยนแปลงค่า
2. PWM: 2 ถึง 13 และ 44 ถึง 46 ให้ output PWM output 8-bits
3. SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS) ใช้สำหรับรองรับการสื่อสารแบบ SPI โดยที่ไม่เกี่ยวข้องกันกับ ICSP header ซึ่งจะมีลักษณะคล้ายกับ Uno, Duemilanove และ Diecimila
4. LED 13 : เป็น build-in LED ที่เชื่อมต่อกับ digital pin 13 เมื่อ pin มีค่าเป็น HIGH LED จะติด แต่เมื่อ pin เป็น LOW LED จะดับ
5. TWI : 20 (SDA) and 21 (SCL). รองรับการเชื่อมต่อแบบ TWI(I2C)
6. บอร์ด Mega2560 มี 16 analog inputs แต่ละ pins ให้ความละเอียด 10 bits
7. AREF. แรงดันอ้างอิง สำหรับ analog input
8. Reset ใช้ในการ reset ไมโครคอนโทรลเลอร์ โดยทั่วไปจะใช้โดยการเพิ่มปุ่ม reset ไว้บน shield เพื่อป้องกันปุ่มที่อยู่บนบอร์ด

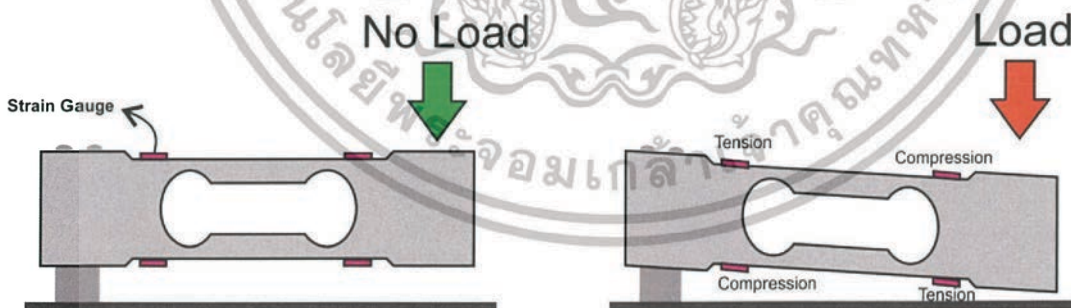
2.2.2 Load cell sensor



รูปที่ 2.2 ตัวอย่าง load cell

(อ้างอิงโดย: <https://www.thaieasyelec.com/article-wiki/review-product-article/how-to-use-load-cell-and-hx711-amplifier-module.html>)

Load Cell คือ Sensor สำหรับตรวจวัดน้ำหนัก แรงกระทำทางกล หรือปริมาณของ Load ที่ต้องการทราบค่า โดยใช้ Strain Gauge มาติดตั้งในบริเวณที่มีการเปลี่ยนแปลงรูปทรงของ Load Cell เมื่อมีแรงมากระทำกับตัว Load Cell จะทำให้ Strain Gauge ที่ติดตั้งในบริเวณที่มีการเปลี่ยนแปลงรูปทรง ยืด หรือ หดตัว ทำให้ค่าความต้านทานที่ตัว Strain Gauge เปลี่ยนไป



รูปที่ 2.3 ตัวอย่างการทำงานของ load cell

(อ้างอิงโดย: <https://www.thaieasyelec.com/article-wiki/review-product-article/how-to-use-load-cell-and-hx711-amplifier-module.html>)

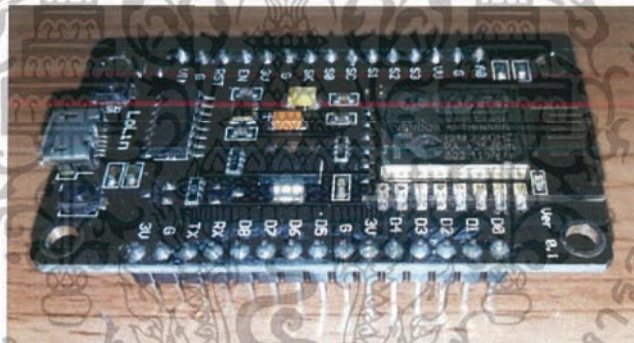
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตามรูปที่ 2.3 ในจุดที่ Strain Gauge ได้รับแรงกด (Compression) จะทำให้ Strain Gauge หดตัวเข้าหากัน และในจุดที่ได้รับแรงดึง (tension) จะทำให้ strain gauge ถูกยืดออก จึงทำให้ค่าความต้านทานของ Strain Gauge เปลี่ยนแปลงไป Strain Gauge ทั้ง 4 ตัวที่อยู่บน Load Cell แบบ Straight Bar จะถูกต่ออยู่ด้วยกันในลักษณะของวงจร Wheatstone Bridge

2.2.3 arduino node esp8266

NodeMCU (โนนด เอ็มซียู) คือ บอร์ดคล้าย Arduino ที่สามารถเชื่อมต่อกับ WiFi ได้ สามารถเขียนโปรแกรมด้วย Arduino IDE ได้เช่นเดียวกับ Arduino และบอร์ดก็มีราคาถูกมาก ๆ เหมาะแก่ผู้ที่จะเริ่มต้นศึกษา หรือทดลองใช้งานเกี่ยวกับ Arduino, IoT อิเล็กทรอนิกส์ หรือแม้แต่การนำไปใช้จริงในโปรเจกต์ต่าง ๆ ก็ตาม เพราะราคาไม่แพง

ภายในบอร์ดของ NodeMCU ประกอบไปด้วย ESP8266 (ไมโครคอนโทรลเลอร์ที่สามารถเชื่อมต่อ WiFi ได้) พร้อมอุปกรณ์อำนวยความสะดวกต่าง ๆ เช่น พอร์ต micro USB สำหรับจ่ายไฟ/อัปโหลดโปรแกรม ซิปสำหรับอัปโหลดโปรแกรมผ่านสาย USB ชิปแปลงแรงดันไฟฟ้า และขาสำหรับเชื่อมต่ออุปกรณ์ภายนอก เป็นต้น



รูปที่ 2.4 ตัว node mcu

(อ้างอิงโดย: <https://poundxi.com/nodemcu-คืออะไร>)

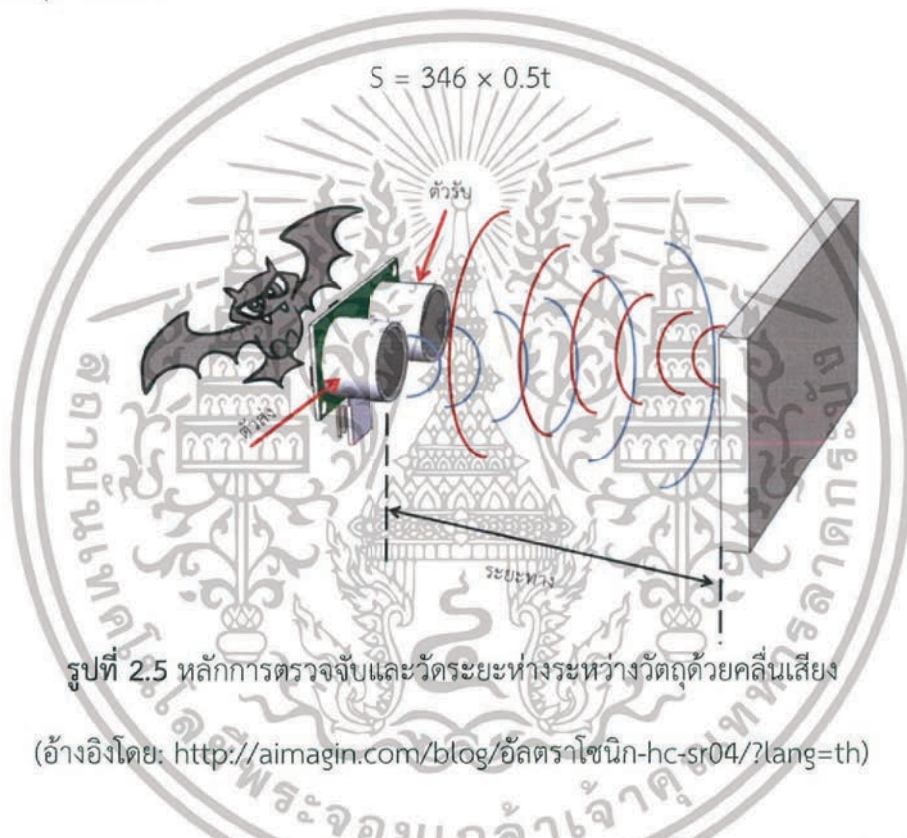
2.2.3.1 จุดเด่นของ NodeMCU

1. สามารถเชื่อมต่อกับ WiFi ได้โดยไม่ต้องติดตั้งโมดูล WiFi เพิ่มเติม
2. ราคาถูกมาก เมื่อเทียบกับบอร์ดที่มี WiFi ในตัวรุ่นอื่น ๆ (ราคาในไทยประมาณ 200 บาท)
3. สามารถเขียน และอัปโหลดโปรแกรมลงบอร์ดด้วยโปรแกรม Arduino IDE ผ่านสาย USB แบบเดียวกับที่ใช้ซาร์จโทรศัพท์ได้
4. ตัวบอร์ดมีขนาดเล็ก (ประมาณ 5.5 x 3 cm.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 Ultrasonic sensor HC-SR04

HC-SR04 เป็นเซนเซอร์โมดูลสำหรับตรวจจับวัตถุและวัดระยะทางแบบไม่สัมผัส [1-2] โดยใช้คลื่นอัลตราโซนิก ซึ่งเป็นคลื่นเสียงความถี่สูงเกินกว่าการได้ยินของมนุษย์ วัดระยะได้ตั้งแต่ 2–400 เซนติเมตร หรือ 1–156 นิ้ว สามารถทำงานกับไมโครคอนโทรลเลอร์ได้ง่าย ใช้พลังงานต่ำ เหมาะกับการนำไปประยุกต์ใช้งานด้านระบบควบคุมอัตโนมัติ หรืองานด้านหุ่นยนต์ หลักการทำงาน จะเหมือนกันกับการตรวจจับวัตถุด้วยเสียงของค้างคาว ตามรูปที่ 2.5 โดยจะประกอบไปด้วยตัวรับ-ส่งอัลตราโซนิก ตัวส่งจะส่งคลื่นความถี่ 40 kHz ออกไปในอากาศด้วยความเร็วประมาณ 346 เมตรต่อวินาที และตัวรับจะคอยรับสัญญาณที่สะท้อนกลับจากวัตถุ เมื่อทราบความเร็วในการเคลื่อนที่ของคลื่น เวลาที่ใช้ในการเดินทางไป-กลับ (t) ก็จะสามารถคำนวณหาระยะห่างของวัตถุ (S) ได้จาก



เพื่อให้การคำนวณหาระยะเป็นไปด้วยความง่าย โมดูลเซนเซอร์นี้จึงได้ประมวลผลให้เรียบร้อยแล้ว และส่งผลลัพธ์ของการคำนวณเป็นสัญญาณพัลส์ที่มีความกว้างสัมพันธ์กับระยะทางที่วัดได้ โมดูลนี้มีจุดต่อใช้งานทั้งหมด 4 จุด การใช้งานบอร์ด STM32F4DISCOVERY การทดลองในเบื้องต้นสามารถต่อวงจรอย่างง่ายได้โดยใช้โปรโตบอร์ดและสายไฟต่อวงจรตามรูปที่ 2 ทั้งนี้ต้องตรวจสอบคุณสมบัติของพอร์ตของไมโครคอนโทรลเลอร์จากดาต้าชีท [3] ว่าสามารถทนระดับแรงดันลอจิก High (5V) ได้

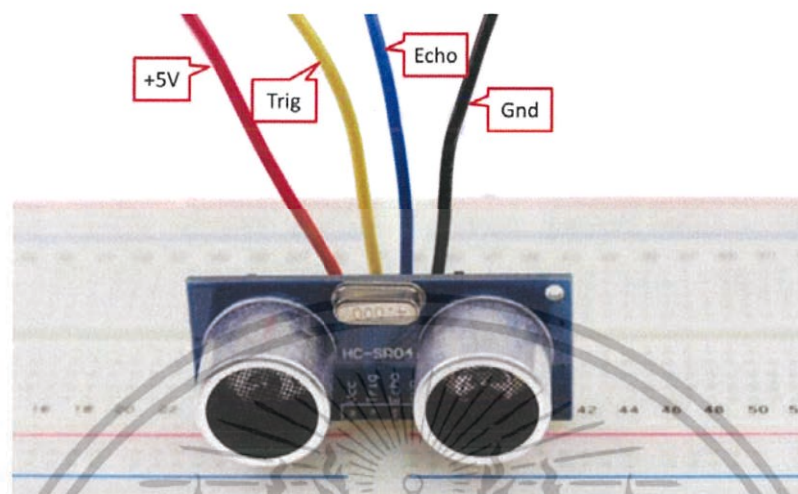
2.2.4.1 ขา VCC สำหรับต่อแรงดันไฟเลี้ยงไม่เกิน 5V

2.2.4.2 ขา Trig เป็นขาอินพุตรับสัญญาณพัลส์ความกว้าง 10 ไมโครวินาทีเพื่อกระตุ้นการสร้างคลื่นอัลตราโซนิกความถี่ 40 kHz ออกสู่อากาศจากตัวส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4.3 ขา Echo เป็นขาเอาต์พุตสำหรับส่งสัญญาณพัลส์ออกจากโมดูลไปยังไมโครคอนโทรลเลอร์ เพื่อตรวจจับความกว้างของสัญญาณพัลส์และคำนวณเป็นระยะทาง

2.2.4.4 ขา GND สำหรับต่อจุดกราวด์ร่วมแรงดันและสัญญาณ



รูปที่ 2. 6 การต่อใช้งานโมดูลอัลตราโซนิก HC-SR04

(อ้างอิงโดย: <http://aimagin.com/blog/อัลตราโซนิก-hc-sr04/?lang=th>)

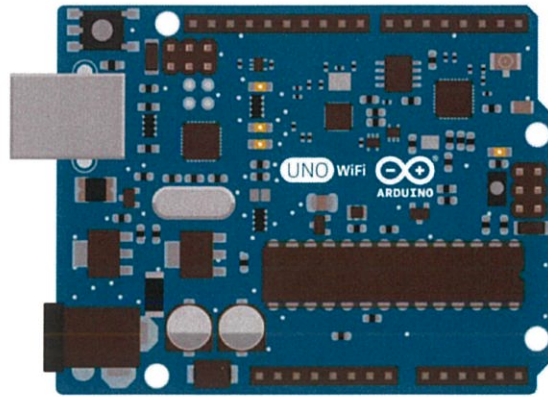
2.3 Arduino

Arduino เป็นภาษาอิตาลีอ่านว่า "อาดูอีโน" หรือจะเรียกว่า "อาดูยโน" ก็ได้ Arduino เป็น Open-Source Platform (แพลตฟอร์มสาธารณะ) โดยเป็นบอร์ดไมโครคอนโทรลเลอร์ตระกูล AVR สำหรับการสร้างต้นแบบทางอิเล็กทรอนิกส์ที่ง่ายต่อการใช้งาน โดยประกอบด้วย

2.3.1. ส่วนที่เป็น Hardware

ไมโครคอนโทรลเลอร์ (MCU: Microcontroller Unit) เป็นการร่วมกับอุปกรณ์อิเล็กทรอนิกส์ต่าง ๆ ประกอบเป็นบอร์ดอิเล็กทรอนิกส์ขนาดเล็กเพื่อให้ง่ายต่อการใช้งาน หรือที่เรียกกันว่า บอร์ด Arduino โดยบอร์ด Arduino เองก็มีหลายรุ่นให้เลือกใช้ โดยในแต่ละรุ่นอาจมีความแตกต่างกันในเรื่องของขนาดหรือสเปก เช่น จำนวนของขารับส่งสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ตัวอย่าง Arduino

(อ้างอิงโดย: <https://www.arduinospro.com/article/7/arduino-basic-ep0-arduino-คืออะไร-ทำอะไร-ได้-มีกี่แบบ>)

2.3.2. ส่วนที่เป็น Software คือ

ภาษา C / C++ เป็นภาษาสำหรับเขียนโปรแกรมควบคุม Arduino IDE เป็นเครื่องมือสำหรับเขียนโปรแกรมด้วยภาษา Arduino คอมไพล์โปรแกรม (Compile) และอัปโหลดโปรแกรมลงบอร์ด (Upload)



รูปที่ 2.8 หน้าโปรแกรม Arduino

(อ้างอิงโดย: <https://www.arduinospro.com/article/7/arduino-basic-ep0-arduino-คืออะไร-ทำอะไร-ได้-มีกี่แบบ>)

ได้-มีกี่แบบ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 Firebase

Firebase คือ โปรเจกต์ที่ถูกออกแบบมาให้เป็น API และ Cloud Storage สำหรับพัฒนา Realtime Application รองรับหลาย Platform ทั้ง iOS App, Android App, Web App



Firebase ถูกสร้างขึ้นจากคุณสมบัติเสริมว่านักพัฒนาสามารถผสมและจับคู่เพื่อให้พอดีกับความ ต้องการของตน บริษัท ก่อตั้งขึ้นในปี 2011 โดยแอนดรูส์และเจมส์ เหมปลิน สินค้าเริ่มต้น Firebase ขึ้น ฐานข้อมูลเรียลไทม์ซึ่งมี API ที่ช่วยให้นักพัฒนาในการจัดเก็บและซิงค์ข้อมูล โดย Google Firebase 2.0 กูเกิลได้ซื้อกิจการ Firebase และมีการพัฒนาให้สามารถ จากบริการ backend เก็บข้อมูลอย่างเดียว มาเป็น แพลตฟอร์ม ครบวงจรสำหรับนักพัฒนาแอปพลิเคชัน รองรับบริการแทบทุกอย่างที่นักพัฒนาแอปพลิเคชัน ต้องใช้งาน

2.4.1 Firebase มีบริการอะไรให้บ้าง

2.4.1.1 **Firebase Analytics** บริการวิเคราะห์ข้อมูล ดึงเทคโนโลยีมาจาก Google Analytics แคมเปญเปิดให้ใช้ฟรีแบบไม่จำกัดปริมาณข้อมูลใด ๆ

2.4.1.2 **Firebase Cloud Messaging (FCM)** ระบบส่งข้อความแจ้งเตือน ใช้งานฟรีไม่ จำกัดปริมาณข้อความ

2.4.1.3 **Firebase Storage** บริการพื้นที่เก็บข้อมูล เอาไว้เก็บภาพ วิดีโอ หรือไฟล์ขนาดใหญ่จากแอปพลิเคชันของผู้ใช้ สร้างอยู่บน Google Cloud Storage

2.4.1.4 **Firebase Remote Config** ตัวช่วยอัปเดตคอนฟิกของแอปพลิเคชัน สำหรับ ปรับแต่งค่าต่าง ๆ ในแอปพลิเคชัน จากระยะไกล (เช่น เกมที่อยากปรับสมดุลของเกมตลอดเวลา) สามารถใช้ร่วมกับ Firebase Analytics เพื่อกำหนดผู้ใช้งานแยกเป็นกลุ่ม ๆ ได้

2.4.1.5 **Firebase Crash Reporting** ตัวรายงานการแครชของแอปพลิเคชัน รองรับทั้ง iOS และ Android

2.4.1.6 **Firebase Test Lab for Android** บริการทดสอบแอปพลิเคชันบนฮาร์ดแวร์จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.7 Firebase Notifications เป็นคอนโซลสำหรับนักพัฒนา เพื่อส่งข้อความผ่าน FCM ไปยังผู้ใช้ สำหรับโปรโมทหรือกระตุ้นให้ผู้ใช้กลับมาเปิดแอปพลิเคชันของเรา (เช่น แจกของในเกม)

2.4.1.8 Firebase Dynamic Links บริการ URL กลางที่สามารถชี้ทางไปยังเพจต่าง ๆ แปรผันตามอุปกรณ์หรือคุณสมบัติของผู้ใช้ (เช่น แต่ละประเทศคลิกเดียวกัน เข้าคนละเพจกัน)

2.4.1.9 Firebase Invites ระบบเชิญเพื่อนมาใช้แอปพลิเคชัน มีพีเจอร์ referral คนชวนได้สิทธิประโยชน์

2.4.1.10 Firebase App Indexing เปลี่ยนชื่อมาจาก Google App Indexing ที่ช่วยให้ Google Search ค้นเจอเนื้อหาภายในแอปพลิเคชัน

2.5 android studio



Android Studio เป็น IDE Tool จาก Google ไว้พัฒนา Android สำหรับ Android Studio เป็น IDE Tools ล่าสุดจาก Google ไว้พัฒนาโปรแกรม Android โดยเฉพาะ โดยพัฒนาจากแนวคิดพื้นฐานมาจาก IntelliJ IDEA คล้าย ๆ กับการทำงานของ Eclipse และ Android ADT Plugin โดยวัตถุประสงค์ของ Android Studio คือต้องการพัฒนาเครื่องมือ IDE ที่สามารถพัฒนา App บน Android ให้มีประสิทธิภาพมากขึ้น ทั้งด้านการออกแบบ GUI ที่ช่วยให้สามารถ Preview ตัว App มุมมองที่แตกต่างกันบน Smart Phone แต่ละรุ่นสามารถแสดงผลบางอย่างได้ทันที โดยไม่ต้องทำการรัน App บน Emulator รวมทั้งยังแก้ไขปรับปรุงในเรื่องของความเร็วของ Emulator ที่ยังเจอปัญหากันในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 Node JS



2.6.1 Node.js คืออะไร

JavaScript เดิมทีออกแบบมาให้ทำงานกับ HTML มีขอบเขตอยู่ใน Web Browser ผู้คนอยากให้ JavaScript ทำงานนอก Web Browser ได้เพื่อเพิ่มความสามารถของมัน จึงมีคนคิดค้นสร้างสิ่งที่เรียกว่า Node

Node ใช้เป็น Runtime สำหรับ JavaScript ทำให้เมื่อติดตั้ง Node ลงไปในระบบต่าง ๆ แล้ว JavaScript สามารถทำงานได้ คล้ายกับ Java ที่มี Runtime ตามหลักการพัฒนาระบบที่ว่า Write once, run anywhere

เนื่องจาก Node.js นั้นขึ้นชื่อในด้านความเร็วของการประมวลผล จึงทำให้ application ที่เขียนด้วย Node.js นั้นมีจำนวนเพิ่มขึ้นอย่างรวดเร็ว ซึ่งรวมไปถึง application ที่จะช่วยให้การพัฒนาเว็บไซต์เป็นไปอย่างรวดเร็วมากขึ้นด้วย

2.6.2 Node.js เอาไปต่อยอดอะไรได้บ้าง

2.6.2.1 สร้าง Mobile App แบบข้ามระบบ iOS และ Android ได้ โดยใช้เครื่องมือที่ชื่อว่า PhoneGap/Cordova หรือ Ionic Framework

2.6.2.2 สร้างฝั่ง Web Server ก็ได้ โดยใช้ Framework อย่าง MEAN Stack หรือ Meteor

2.6.2.3 สร้างระบบฝั่ง IoT ก็ได้ โดยใช้ Particle ที่ชื่อเดิมคือ Spark JS, Cylon JS, หรือ Johnny Five

2.6.3 วิธีติดตั้ง Node.js

2.6.3.1 การจะนำ application ที่เขียนด้วย Node.js มาใช้งาน จะต้องติดตั้ง Node.js เสียก่อน download ได้ที่ <https://nodejs.org>

2.6.3.2 เลือก download แบบ installer เพราะขั้นตอนการติดตั้งจะไม่ยุ่งยาก

2.6.3.3 เมื่อติดตั้งเสร็จแล้ว ให้เปิด Command-line Interface(Command Prompt, Terminal) ขึ้นมา แล้วลองพิมพ์คำสั่ง `node -v`

2.6.3.4 หากเลขเวอร์ชันของ Node.js แสดงขึ้นมา ก็แปลว่าได้ติดตั้ง Node.js เรียบร้อยแล้ว

2.7 Heroku

Heroku เป็น Platform as a Service (Paas) ที่ให้ใช้งานได้ฟรี (มีแบบเสียเงินด้วย) โดยรองรับภาษาโปรแกรมที่หลากหลาย เช่น Ruby, PHP, Node.js, Python, Java, Clojure, Scala และยังสามารถสร้าง buildpack สำหรับภาษาอื่น ๆ ได้ เช่น Lua ที่รันอยู่บน OpenResty ได้อีกด้วย



heroku

Heroku เหมาะกับใครบ้าง

จริง ๆ แล้ว มันเหมาะกับทุกคน เช่น นักศึกษาอยากลองเขียนเว็บด้วย PHP แต่ไม่ได้เช่า Hosting ก็สามารถใช้ Heroku ได้ หรือแม้แต่บริษัท Start up ที่ไม่ยักวางเครื่องเอง คอนฟิเอง ก็ใช้ได้ เพราะมันสามารถ scale ให้รองรับผู้ใช้เยอะ ๆ ได้โดยง่าย

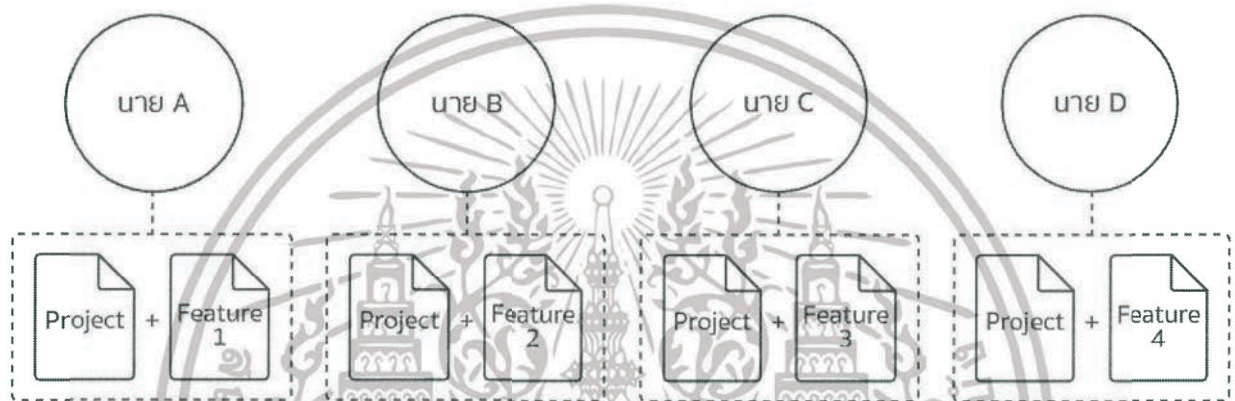
นอกจากรองรับภาษาโปรแกรมที่หลากหลายแล้ว ตัว Heroku มี App Store ของมันด้วยเรียกว่า add-ons สำหรับเพิ่มเติมบริการอื่น ๆ เข้าไปในแอปพลิเคชัน เช่น PostgreSQL, MongoDB, Redis เป็นต้น ซึ่งก็มีทั้งฟรี และไม่ฟรีให้เลือกใช้งาน

2.8 Github

มาพูดถึง Version Control Systems (VCS) กันก่อน ในยุคนี้ต้องบอกเลยว่า Version Control ถือว่าเป็นหนึ่งในคุณสมบัติพื้นฐานของ Developer ก็ว่าได้ โดยเฉพาะอย่างยิ่งบริษัทที่ต้องทำงานร่วมกันเป็นทีม (ถึงจะพัฒนาคนเดียว ก็ควรจะใช้เหมือนกันนะ)

2.8.1 Version Control สำคัญยังไง?

ลองนึกดูว่าถ้าคุณต้องพัฒนาโปรเจกต์ขนาดใหญ่ที่มี Developer 4 คนที่กำลังเขียนโค้ดในโปรเจกต์นี้อยู่ คุณจะใช้วิธีไหนเพื่อเอาโค้ดที่แต่ละคนเขียนมารวมเข้าด้วยกันในโปรเจกต์



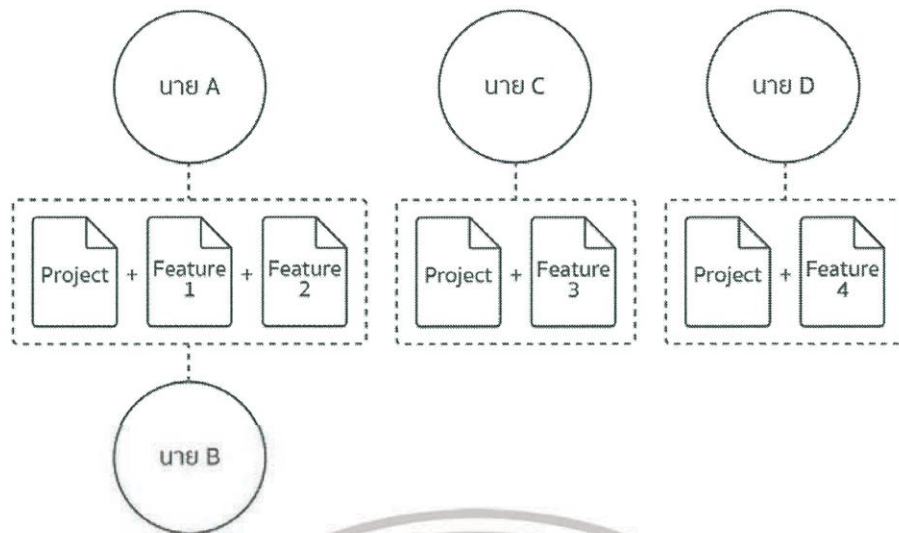
รูปที่ 2.9 4 Developer กับ 4 Feature

(อ้างอิงโดย: <https://blog.nextzy.me/มาเรียนรู้-git-แบบง่ายๆกันเถอะ-427398e62f82>)

วิธีสุดเก่าแก่ที่ใช้กันก็คือ — คัดลอกโปรเจกต์จากแต่ละคนมารวมไว้ในเครื่องเดียวกัน แล้วนั่งทำด้วยกัน มีคนหนึ่งทำหน้าที่เปิดโค้ดของแต่ละคนขึ้นมา สมมติว่าคน ๆ นั้นคือนาย A และนาย B เป็นคนเขียนโค้ดที่กำลังจะรวมไว้ในโปรเจกต์เดียวกัน นาย A ก็เลยต้องถามนาย B ว่าเขียนโค้ดตรงไหนเพิ่มบ้าง แล้วค่อยเอาไปปะรวมไว้ในโปรเจกต์หลัก

ปัญหาเก่า ๆ ที่เกิดขึ้นอยู่ประจำก็คือ โค้ดที่นาย B ไปแก้ไข ดันไปทับซ้อนกับนาย C เพราะนาย C ก็แก้ไขโค้ดตรงจุดนั้นเหมือนกัน กลายเป็นว่านาย B ก็ต้องไปเรียกนาย C มานั่งคุยด้วย เพื่อบอกให้นาย A แก้ไขให้โค้ดของนาย B และนาย C ทำงานร่วมกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 การรวมโค้ดของนาย A และ B เข้าด้วยกันก่อน ส่วนนาย C กับ D นั่งรอ

(อ้างอิงโดย: <https://blog.nextzy.me/มาเรียนรู้-git-แบบง่ายๆกันเถอะ-427398e62f82>)

นี่ยังไม่รวมไปถึงกรณีที่นาย B จำไม่ได้ว่าตัวเองแก้ไขโค้ดตรงจุดไหนไปบ้าง เพราะพีเจอร์ท่านั้นใช้เวลาหลายวันและเขียนโค้ดหลายบรรทัด กลายเป็นว่าต้องนั่งไล่ดูโค้ดว่าพลาดตรงไหนไปหรือบ้าง ซึ่งรวมไปถึงปัญหาอื่น ๆ ที่จะตามมาอีกมากมาย ทำให้เกิดสิ่งที่เรียกว่า Version Control ขึ้นมาเพื่อควบคุมการเปลี่ยนแปลงของโค้ดในโปรเจกต์ โดยประโยชน์ที่เห็นได้ชัดของ Version Control จะมีดังนี้

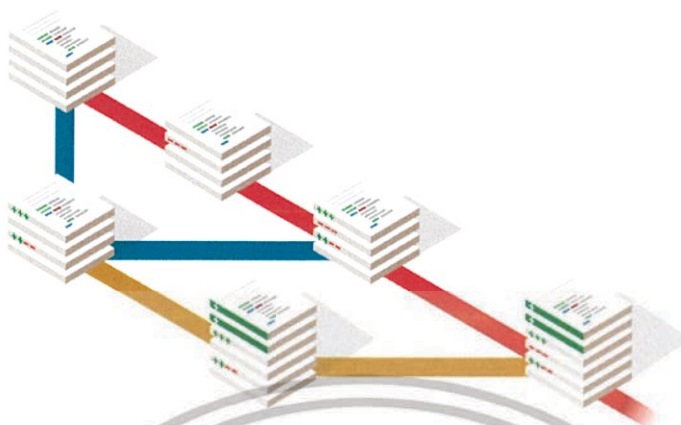
- 2.8.1.1 เก็บประวัติการแก้ไขโค้ดไว้ทุกครั้ง และรู้ว่าโค้ดตรงไหนใครเป็นคนเพิ่มเข้ามาหรือแก้ไข
- 2.8.1.2 ช่วยรวมโค้ดจากหลาย ๆ คนเข้าด้วยกันให้ง่ายขึ้น ดูได้ว่าโค้ดเดิมคืออะไร และแก้ไขเป็นอะไร
- 2.8.1.3 เมื่อเกิดปัญหาก็สามารถติดตามดูประวัติการแก้ไขโค้ดในแต่ละไฟล์แต่ละบรรทัดได้ง่าย
- 2.8.1.4 ช่วยให้สามารถจัดการโปรเจกต์ได้อย่างเป็นระบบ ไม่แก้ไขโค้ดสะเปะสะปะ มองโค้ดแต่ละส่วนเป็นพีเจอร์ท ไม่เขียนโค้ดข้ามพีเจอร์ทไปมาในโค้ดชุดเดียวกัน
- 2.8.1.5 เป็น Backup ไปในตัว ไม่ต้องกลัวเวลาโค้ดมีปัญหาแล้วต้อง Rollback กลับไปใช้โค้ดชุดเก่า และใช้พื้นที่ในการเก็บข้อมูลน้อยเมื่อเทียบกับการ Backup แบบเก็บทั้งโปรเจกต์ไว้ทุกครั้งที่ทำกร Backup
- 2.8.1.6 สามารถ Track การทำงานของทุกคนภายในทีมได้จาก History

โดย Git นั้นเป็นหนึ่งใน Version Control แบบ Distributed Version Control Systems ที่นิยมใช้งานกันในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.2 Git ทำงานยังไง?

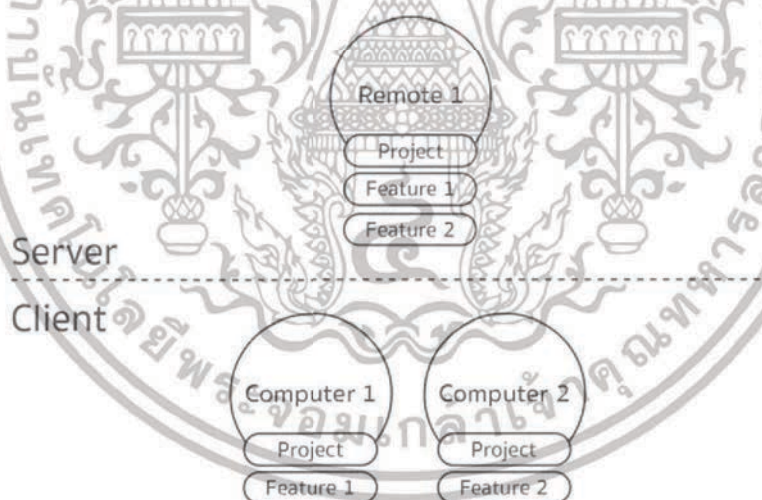
มันก็คือการทำงานแบบ Distributed Version Control Systems



รูปที่ 2.11 การทำงานแบบ Distributed Version Control Systems

(อ้างอิงโดย: <https://git-scm.com/images/branching-illustration@2x.png>)

ให้ลองนึกถึงภาพว่ามี Server กลางตัวหนึ่งที่คอยเก็บข้อมูลจากผู้ใช้แต่ละเครื่องก่อนนะครับ



รูปที่ 2.12 Server กลางตัวหนึ่งที่คอยเก็บข้อมูลจากผู้ใช้แต่ละเครื่องก่อน

(อ้างอิงโดย: <https://blog.nextzy.me/มาเรียนรู้-git-แบบง่ายๆกันเถอะ-427398e62f82>)

Server กลาง เรียกมันว่า Remote 1 ส่วน Developer ที่ทำโปรเจกต์นี้อยู่จะเรียกว่า Computer 1 และ 2 ซึ่งแต่ละคนก็ทำคนละ Feature อยู่ ซึ่ง Feature ของแต่ละคนนั้นก็จะถูกเก็บไว้ที่ Remote 1

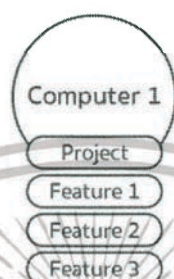
อาจจะดูเหมือนว่ามันเป็นระบบ Server กลางที่คอย Backup ข้อมูลของทุกคนอยู่ตลอดเวลา แต่ในความเป็นจริง Git นั้นมีอะไรมากกว่านี้อีกเยอะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Git ออกแบบมาให้ทำงานกระจายแบบไม่มีศูนย์กลาง ทุกเครื่องทำงานเป็น VCS ด้วยตัวเองได้ นั้นหมายความว่าไม่จำเป็นต้องมี Server กลางก็ได้ สามารถใช้เครื่องส่วนตัวทำเป็น VCS ได้เลย แต่ถ้าต้องทำงานร่วมกันหลาย ๆ เครื่อง ก็ต้องใช้ Server เป็นตัวกลางในการรวมข้อมูล

Server

Client

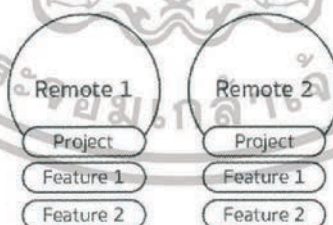


รูปที่ 2.13 แต่ละเครื่องทำงานเป็น VCS

(อ้างอิงโดย: <https://blog.nextzy.me/มาเรียนรู้-git-แบบง่ายๆกันเถอะ-427398e62f82>)

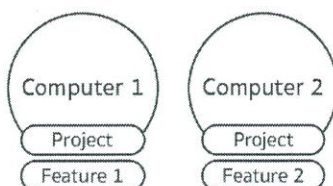
ซึ่งจะทำให้ข้อมูลไม่ผูกขาดกับ Server จนเกินไป ในเวลาที่ Server กลางมีปัญหาหรือว่าทำงานแบบ Offline ก็ยังคงทำงานได้อยู่ โดยใช้ข้อมูลจาก VCS ภายในเครื่องตัวเอง พอเชื่อมต่อกับ Server กลางก็ค่อย Sync ข้อมูลทีหลังได้

และเมื่อทุก ๆ เครื่องทำงานเป็น VCS อยู่แล้ว จึงทำให้สามารถมี Server กลางมากกว่า 1 ตัวได้เช่นกัน (แต่ไม่ค่อยเจอการใช้งานแบบนี้มากนัก)



Server

Client



รูปที่ 2.14 Server หลายตัวทำเป็น VCS

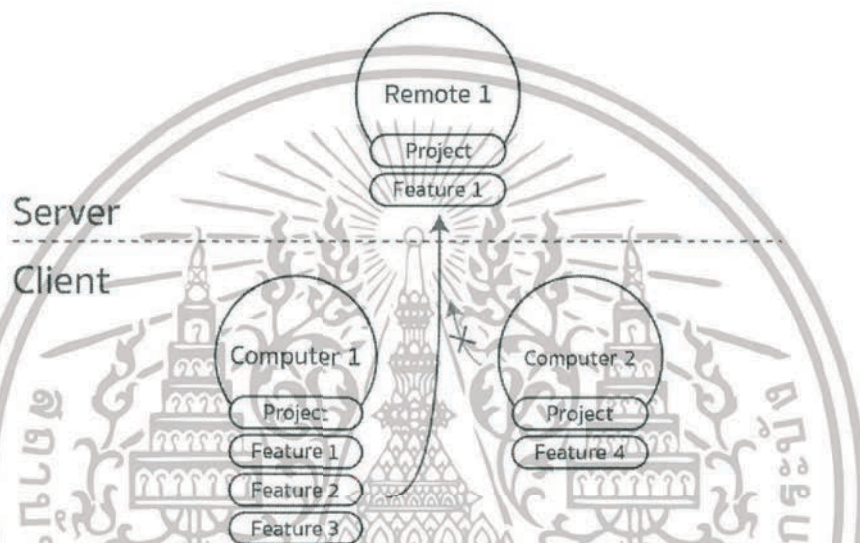
(อ้างอิงโดย: <https://blog.nextzy.me/มาเรียนรู้-git-แบบง่ายๆกันเถอะ-427398e62f82>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Git จะ Sync ข้อมูลเมื่อสั่งเท่านั้น

ขั้นตอนการ Sync ข้อมูลของ Git เดียวจะพูดถึงในภายหลัง ขอเน้นไปที่ภาพรวมของการทำงานก่อน เวลาที่อยากจะ Sync ข้อมูลจาก Remote มาลงเครื่อง หรือจากเครื่องส่งขึ้นไปยัง Remote ก็จะต้องเป็นคนจัดการเอง ไม่ได้ทำการ Sync ให้โดยอัตโนมัติ เพราะว่ามันจะมีเรื่องจุกจิกที่ต้องเป็นคนตัดสินใจและเลือกเอง ไม่สามารถให้ตัว Git จัดการให้อัตโนมัติได้

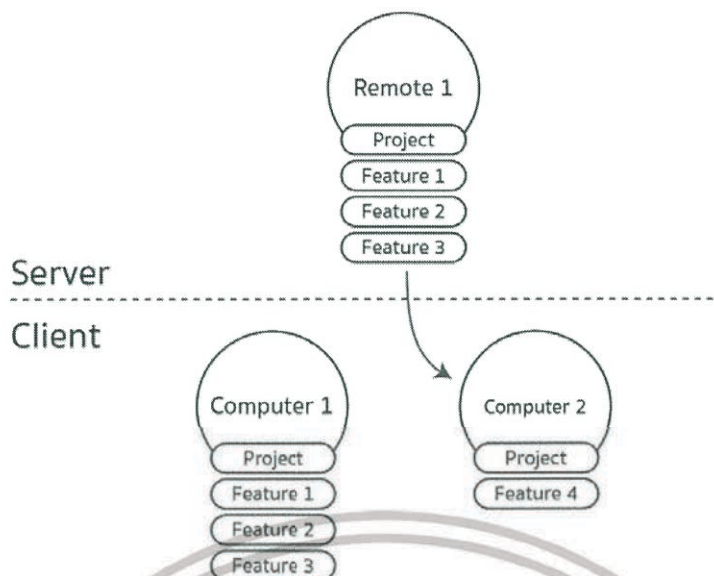
ยกตัวอย่างเช่น โค้ดจากนักพัฒนา 2 คนที่ดันไปแก้ไขที่บรรทัดเดียวกัน แต่คำสั่งต่างกัน และจุดประสงค์ของโค้ดต่างกัน แล้ว Git จะรู้ได้ยังไงว่าต้องเอาโค้ดแบบไหน?



รูปที่ 2.15 ใครไวกว่าก็สบายไป

(อ้างอิงโดย: <https://blog.nextzy.me/มาเรียนรู้-git-แบบง่ายๆกันเถอะ-427398e62f82>)

และนั่นก็คือหน้าที่ของนักพัฒนาที่จะต้องเป็นคนตัดสินใจหรือแก้ไขโค้ดนั่นเองครับ โดยคนที่ Sync ข้อมูลขึ้น Remote ก่อนจะไม่ต้องทำอะไร และคนที่ Sync ข้อมูลทีหลังจะต้องเป็นคนจัดการเอง ซึ่งสามารถตัดสินใจได้ว่าจะใช้โค้ดของคนแรก หรือจะใช้โค้ดของตัวเอง หรือว่าจะแก้ไขโค้ดเพื่อให้รองรับโค้ดของทั้งสองคนก็ได้เช่นกัน



รูปที่ 2.16 การดึงโค้ดมารวมกับโค้ดของตัวเองให้เรียบร้อยก่อน ถึงจะ Sync ขึ้น Remote ได้

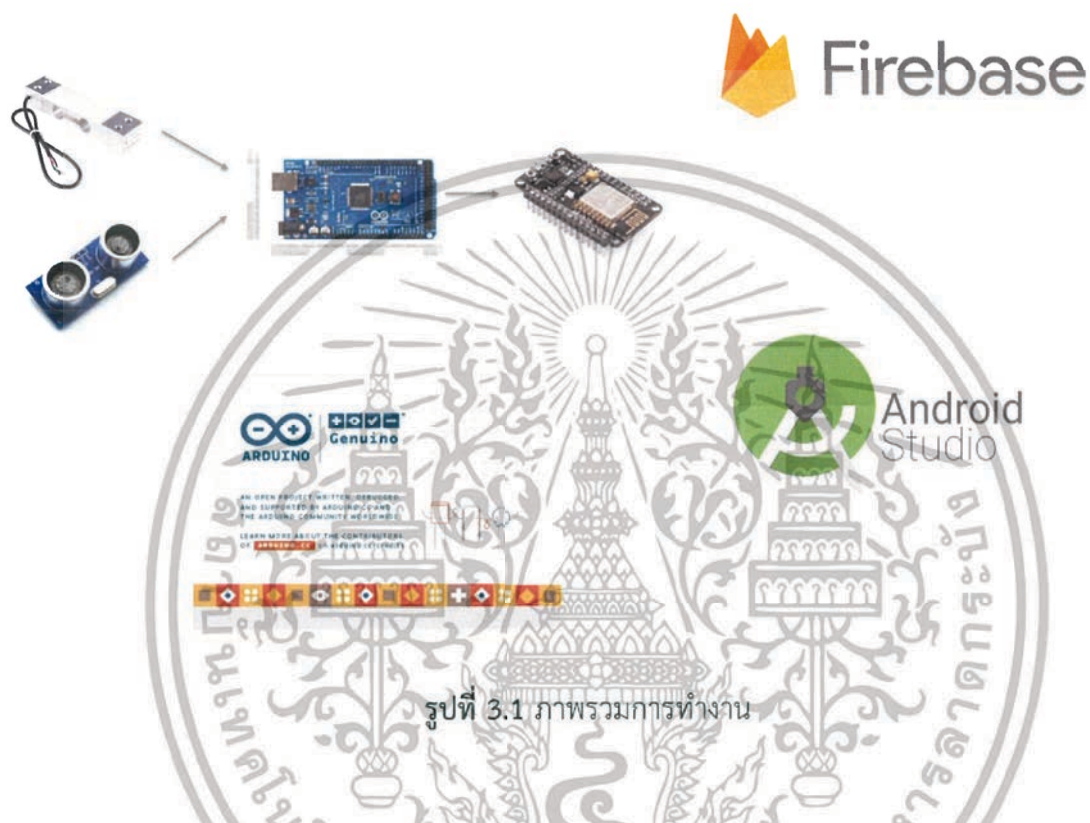
(อ้างอิงโดย: <https://blog.nextzy.me/มาเรียนรู้-git-แบบง่ายๆกันเถอะ-427398e62f82>)

ดังนั้น Git จะไม่ได้ทำงานอัตโนมัติโดยสมบูรณ์แบบ เพราะมันยังมีเรื่องละเอียดละอ่อนอีกหลาย ๆ อย่างที่ผู้ใช้อย่าง ๆ ต้องเข้าใจเพื่อที่จะได้จัดการกับโค้ดในโปรเจกต์ได้อย่างเหมาะสม

บทที่ 3

การออกแบบ

3.1 หลักการทำงานและภาพรวมของระบบ



รูปที่ 3.1 ภาพรวมการทำงาน

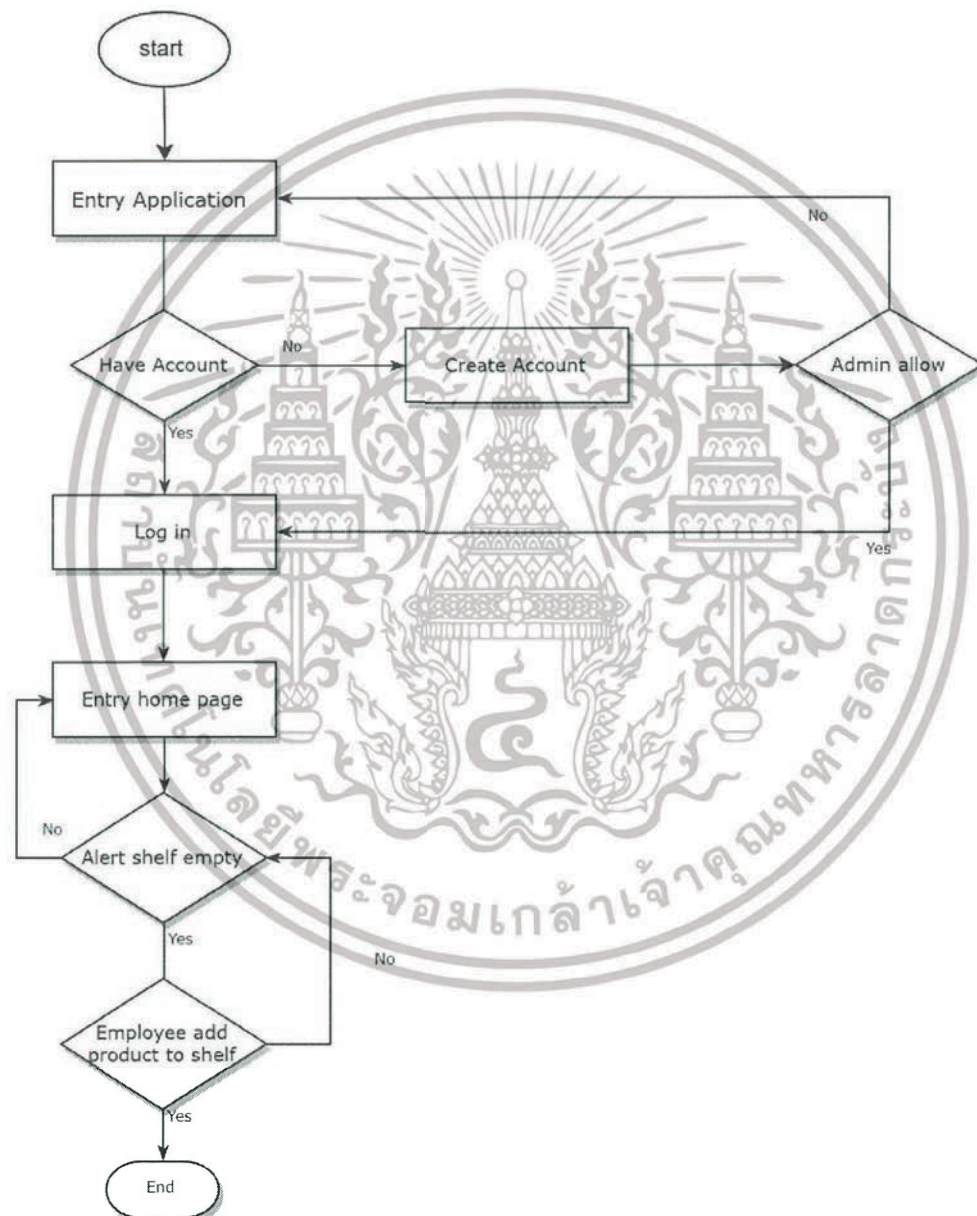
จากรูปที่ 3.1 เป็นขั้นตอนการทำงานของ sensor ทั้ง 2 ตัว คือ Load cell sensor กับ Ultrasonic sensor HC-SR04 ทำการตรวจจับค่าต่าง ๆ ตามหน้าที่ของ sensor นั้น ๆ แล้วส่งค่ามาที่บอร์ด Arduino mega 2560 ในการประมวลผล เมื่อค่าที่ได้ตรงตามเงื่อนไขที่ตั้งไว้ บอร์ด Arduino ก็จะทำการส่งค่านั้นไปยังบอร์ด Node mcu เมื่อ บอร์ด Node mcu รับค่ามา ก็จะเอาค่านั้นส่งขึ้นไปเก็บใน server ที่ชื่อ firebase และเมื่อค่ามีการเปลี่ยนแปลง sever จะส่งค่ามาที่แอปพลิเคชัน เพื่อแจ้งเตือนให้ผู้ใช้ได้ทราบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ผังงาน (Flowchart)

3.2.1 ผังการทำงานของแอปพลิเคชัน

ในรูปที่ 3.2 เมื่อเข้า Application ถ้ามี account แล้ว ให้ทำการ login แต่ถ้ายังไม่มีให้ทำการสมัครสมาชิก หลังจากสมัครสมาชิกเสร็จต้องรอ admin ยอมรับ เมื่อ admin ทำการยอมรับเสร็จก็จะสามารถ login ได้เข้ามาหน้า homepage เมื่อมีการแจ้งเตือนเมื่อสินค้าหมดพนักงานก็จะมาทำการเติมสินค้า แต่ถ้ายังไม่มีการแจ้งเตือนก็จะรออยู่หน้า homepage ต่อไป

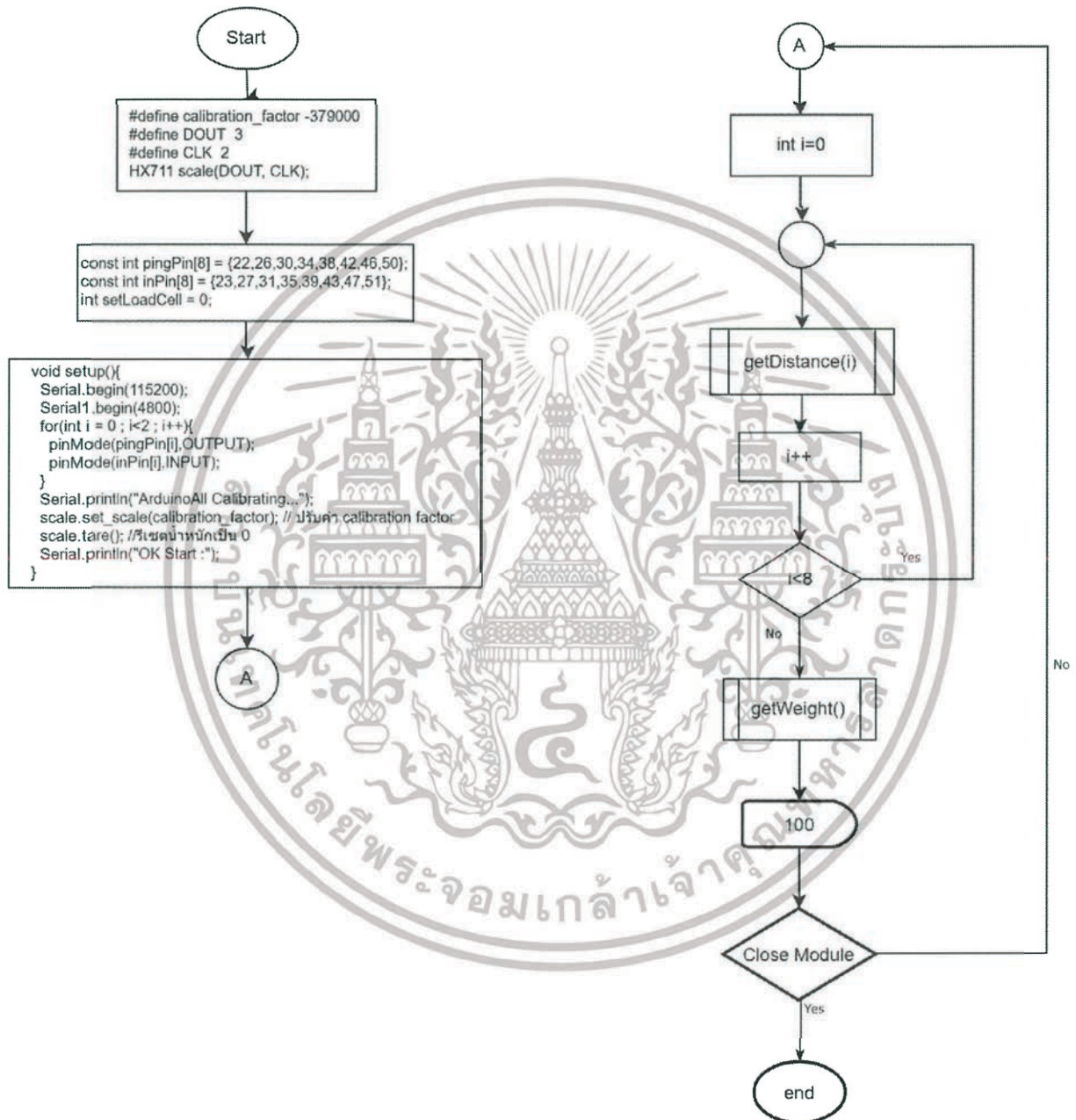


รูปที่ 3.2 ผังการทำงานของแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 ผังการทำงานของ Arduino Mega 2560

รูปที่ 3.3 เริ่มต้นด้วยกำหนดตัวแปรที่เป็นค่าของเซนเซอร์ที่ใช้งานแต่ละตัวต้องการ แล้ว setup โดยนำตัวแปรที่กำหนดไว้มาใส่ให้กับเซนเซอร์ต่าง ๆ ที่ต่อกับ board Arduino mega 2560 ในส่วน A จะเป็น loop การทำงานโดยเริ่มจากรับค่าระยะทางจากเซนเซอร์ตัวที่ 1 ถึง 8 แล้วจากนั้นก็รับค่าน้ำหนักจากเซนเซอร์ตัวที่ 9 แล้วก็กลับไปทำงานในส่วนเซนเซอร์รับค่าระยะทางใหม่วนไปเรื่อย ๆ จนกว่าจะทำการปิดอุปกรณ์

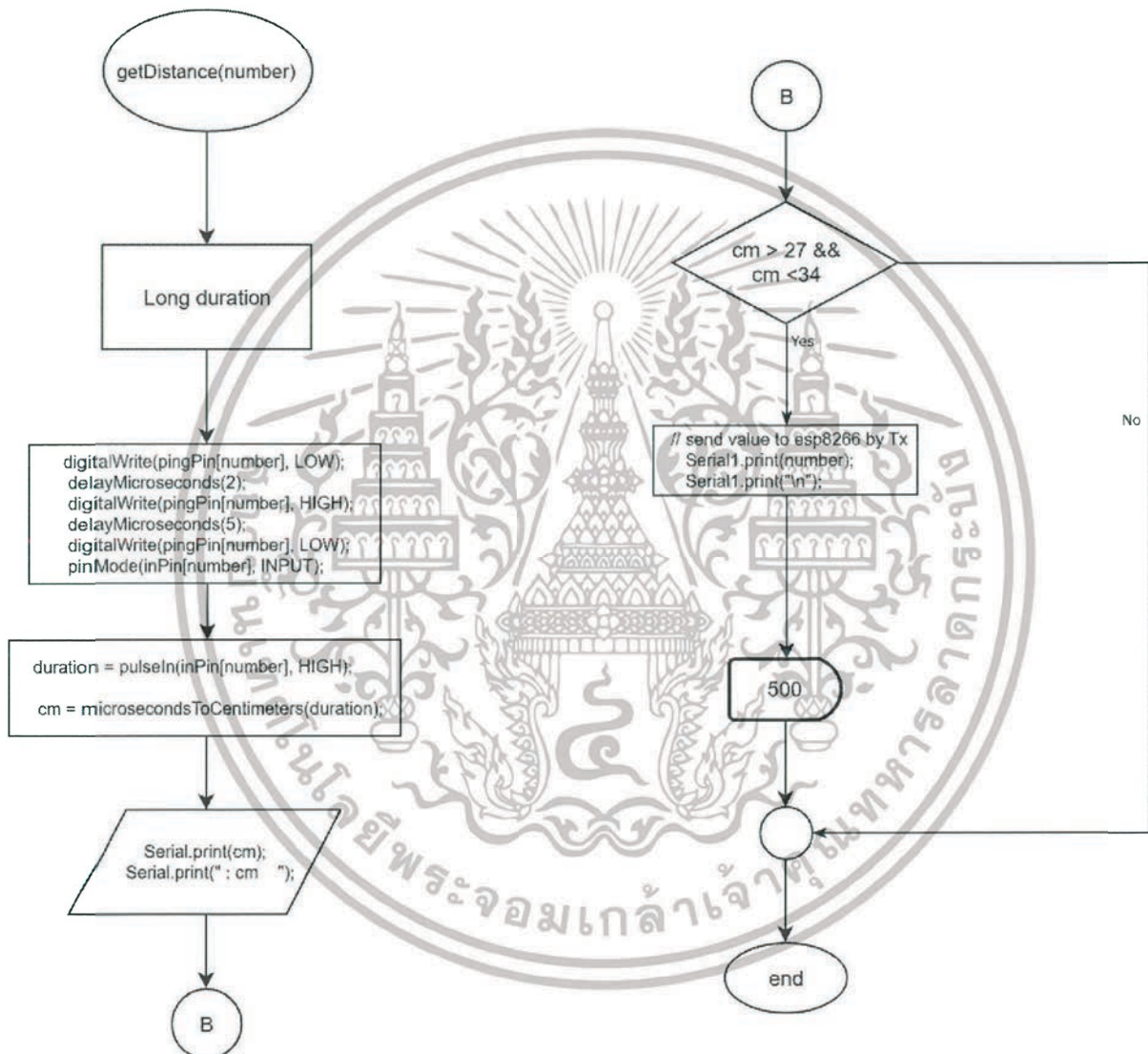


รูปที่ 3.3 ผังการทำงานของ Arduino Mega 2560

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 ผังการทำงานของ Function getDistance ของ Arduino Mega 2560

รูปที่ 3.4 เป็นการทำงานภายใน function getDistance หรือการทำงานรับค่ารับระยะทางเริ่มด้วยรับตัวแปร number เพื่อดูว่าเป็นเซนเซอร์ตัวที่เท่าไร กำหนดตัวแปร duration ต่อด้วยการคำนวณระยะทางโดยแปลงเวลาการเดินทางของคลื่นความถี่เป็นระยะทาง จากนั้นนำค่าระยะทางที่มีหน่วยเป็นเซนติเมตรมาตรวจสอบ ถ้าค่ามากกว่า 27 เซนติเมตร และ น้อยกว่า 34 เซนติเมตร ให้ส่งค่าตำแหน่งเซนเซอร์ไปยัง NodeMcu เพื่อทำการส่งค่าขึ้น server ต่อไป

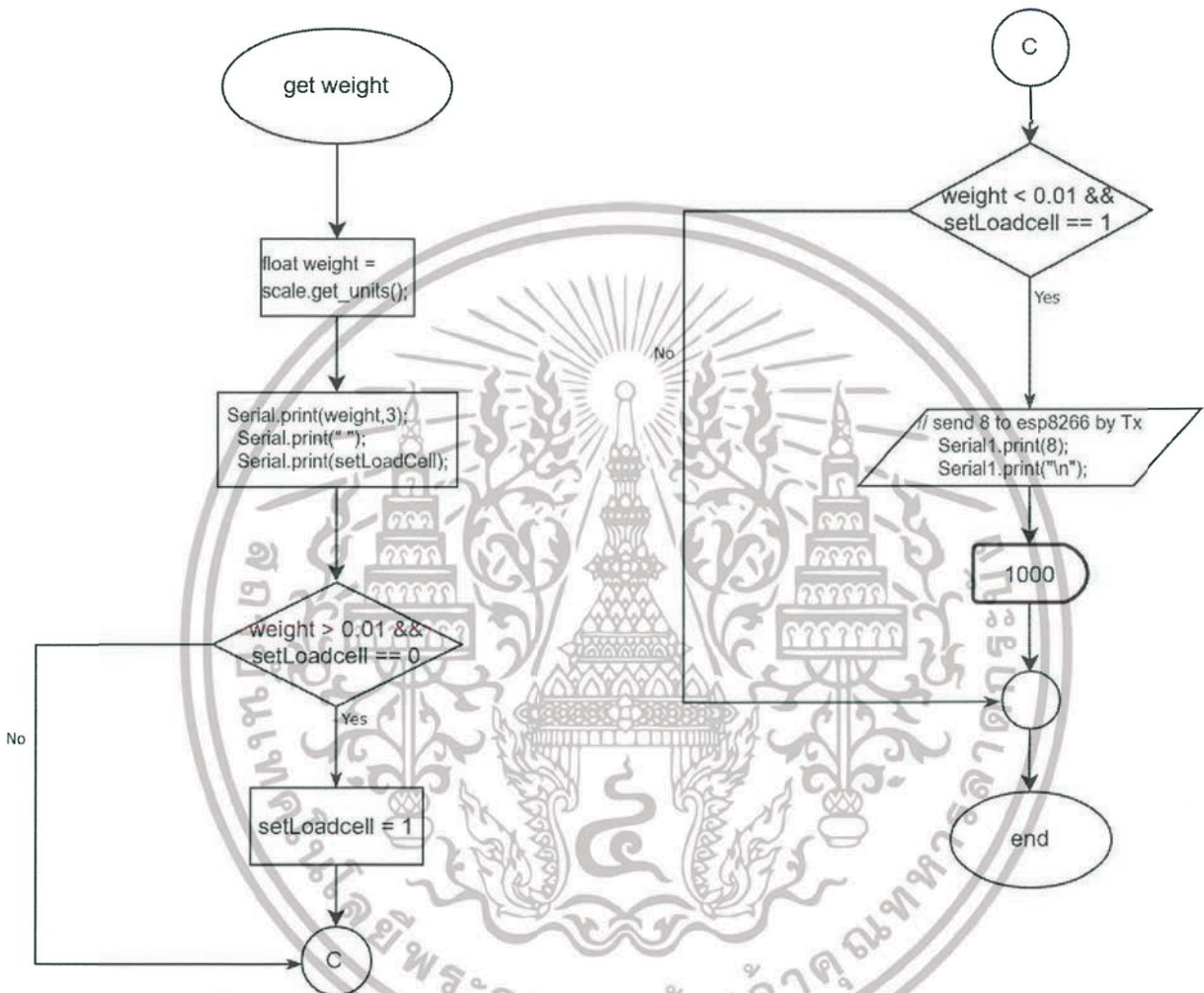


รูปที่ 3.4 ผังการทำงานของ Function getDistance ของ Arduino Mega 2560

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 ผังการทำงานของ Function getWeight ของ Arduino Mega 2560

รูปที่ 3.5 เป็นการทำงานภายใน function getWeight ประกาศตัวแปรรับค่าน้ำหนักมาจาก load cell แสดงค่าน้ำหนักเป็นทศนิยม 3 ตำแหน่ง หน่วยกิโลกรัม เมื่อเริ่มวางสินค้าจะทำการตรวจจับน้ำหนัก เมื่อสินค้าหมดหรือน้ำหนักเหลือ 0 กิโลกรัม จะทำการส่งค่าเซนเซอร์ไปบอก NodeMcu ว่าเป็นเซนเซอร์ที่ใช้ชั่งน้ำหนัก



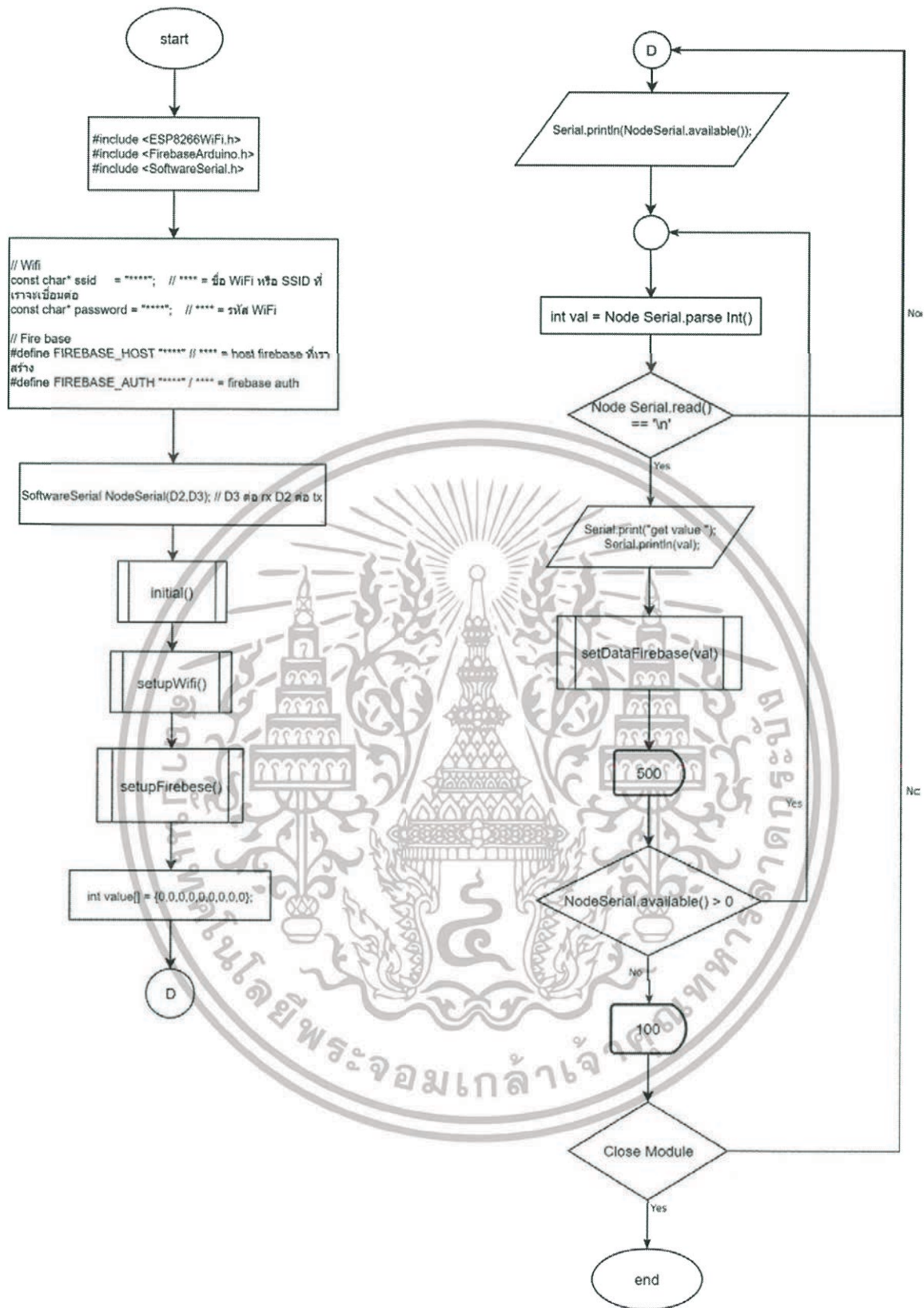
รูปที่ 3.5 ผังการทำงานของ Function getWeight ของ Arduino Mega 2560

3.2.5 ฝั่งการทำงานของ Node MCU ESP8266

รูปที่ 3.6 เป็นการทำงานของ NodeMcu เริ่มด้วยนำ library ที่จำเป็นเข้ามาใช้งาน ต่อด้วยประกาศตัวแปรเพื่อกำหนดค่าให้กับเซนเซอร์ NodeMcu ต่อไปกำหนดช่อง Serial เพิ่ม ใช้ในการส่งค่า กำหนดค่าเซนเซอร์ใน function initial กำหนดค่าให้ NodeMcu ต่อ Wifi ด้วย function setupWifi แล้วกำหนดการต่อ server ด้วย function setupFirebase กำหนดตัวแปรในการรับค่าจาก Arduino mega

ในส่วนของ D เริ่มด้วยแสดงสถานะว่า Arduino mega มีการส่งค่ามาหรือไม่ ถ้าไม่มีการรับค่าจะทำการอ่านค่าไปเรื่อย ๆ เมื่อมีการรับค่าเข้ามาจะแสดงผลค่าที่ได้รับเข้ามาและเข้าไปส่งค่าไปเก็บที่ server ด้วย function setDataFirebase แล้วตรวจสอบว่ามีการส่งค่ามาอีกหรือไม่ ถ้ามีทำการส่งค่าไป server เหมือนเดิมแต่ถ้าไม่ก็รอค่าที่ส่งมาจาก Arduino mega จบการทำงานเมื่อปิดอุปกรณ์



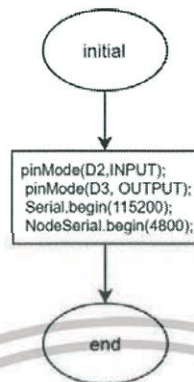


รูปที่ 3.6 ผังการทำงานของ Node MCU ESP8266

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6 ฟังก์ชันการทำงานของ Function initial ของ Node MCU ESP8266

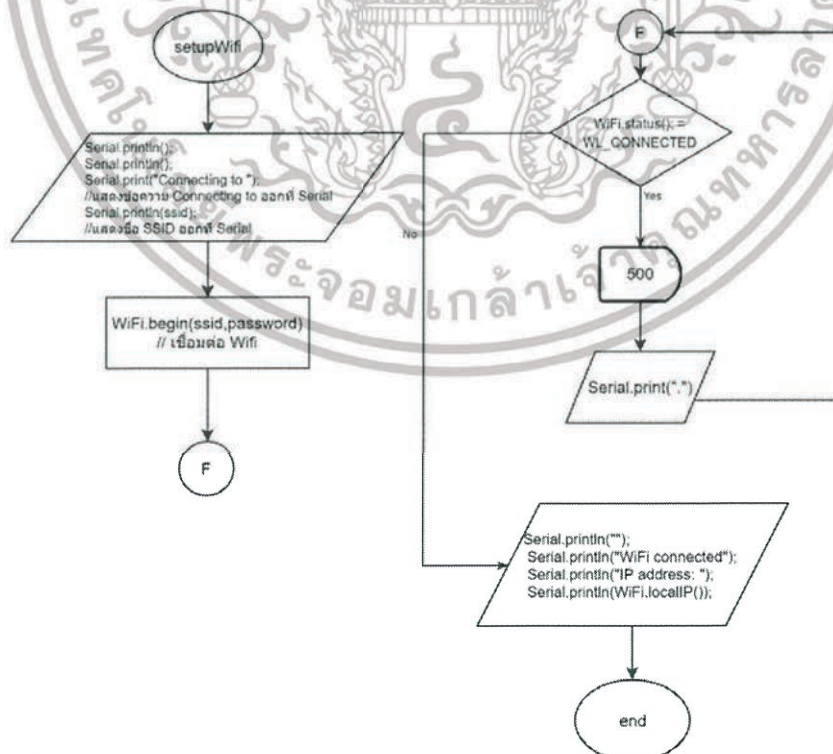
รูปที่ 3.7 เป็นการทำงานกำหนดค่าให้กับตัว node mcu โดยกำหนดพวกรา serial ที่ใช้ส่งข้อมูล ความเร็วในการส่ง ภายใน function initial



รูปที่ 3.7 ฟังก์ชันการทำงานของ Function initial ของ Node MCU ESP8266

3.2.7 ฟังก์ชันการทำงานของ Function setupWifi ของ Node MCU ESP8266

รูปที่ 3.8 เป็นการกำหนดค่าในการเชื่อมต่อ wifi ให้กับ NodeMcu เริ่มต้นด้วยแสดงค่าว่ากำลังต่อ wifi ตามด้วย setup username และ password ที่ใช้ในการต่อ ถ้าต่อสำเร็จ แสดง ip address ของ wifi ที่ต่อไว้ ถ้าต่อไม่สำเร็จจะแสดง . ไปเรื่อย ๆ จนกว่าจะต่อ wifi ได้ เมื่อต่อสำเร็จก็จะจบการทำงานของ function setupWifi

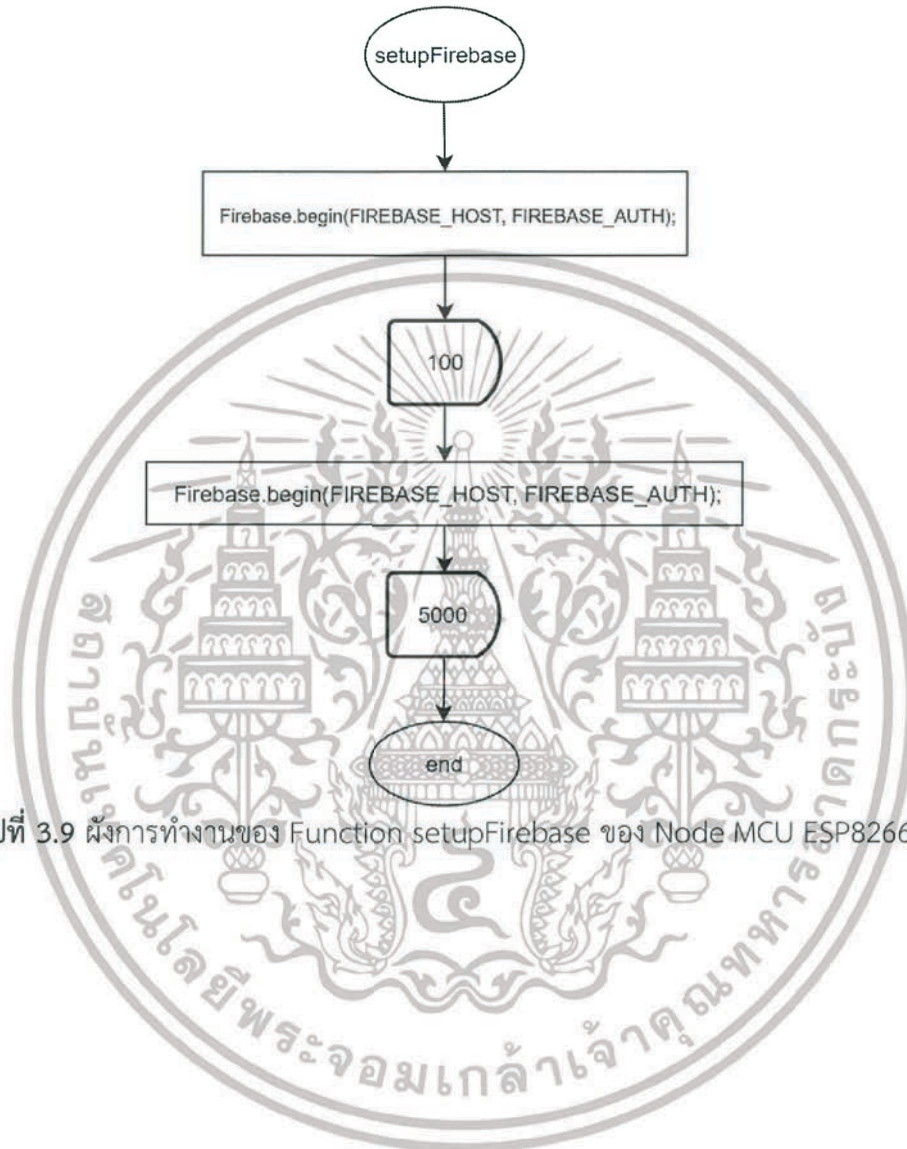


รูปที่ 3.8 ฟังก์ชันการทำงานของ Function setupWifi ของ Node MCU ESP8266

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.8 ผังการทำงานของ Function setupFirebase ของ Node MCU ESP8266

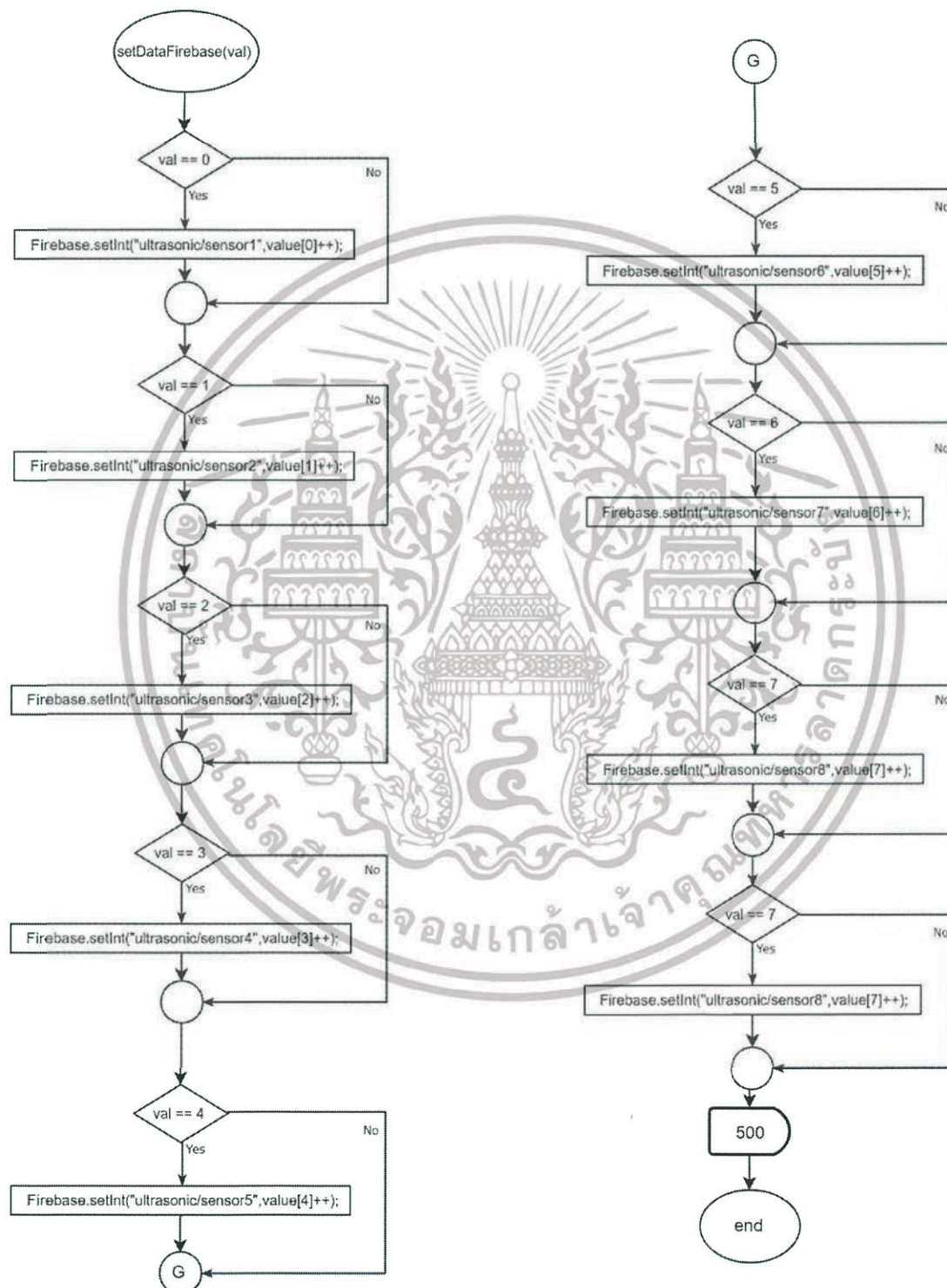
รูปที่ 3.9 เป็นการกำหนดค่าในการเชื่อมต่อ firebase ให้กับ NodeMcu เริ่มต้นด้วย setup firebase ด้วยชื่อ host กับรหัส authentication ของ firebase จบการทำงาน function setupFirebase



รูปที่ 3.9 ผังการทำงานของ Function setupFirebase ของ Node MCU ESP8266

3.2.9 ผังการทำงานของ Function setDataFirebase ของ Node MCU ESP8266

รูปที่ 3.10 เป็นการรับค่า val ซึ่งเป็นค่าของตัวเลขของ sensor ที่ต่อกับ Arduino mega ที่ส่งมาว่าเป็น sensor ตัวไหนเมื่อค่า val ตรงกับเลขไหนที่ได้ set ไว้ในเงื่อนไขก็ให้ไป set ค่าในที่ตำแหน่งของ firebase ที่ได้ set ไว้



รูปที่ 3.10 ผังการทำงานของ Function setDataFirebase ของ Node MCU ESP8266

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทำงานของโปรแกรม

จากการวิเคราะห์และออกแบบในบทที่ 3 ในบทนี้ จะแสดงถึงขั้นตอนการทำงานของโปรแกรม ตามที่ได้วิเคราะห์และออกแบบไว้ดังกล่าว

การทำงานของระบบการแจ้งเตือนสินค้าเมื่อของหมด

4.1 การเข้าสู่หน้าหลัก

เข้าโปรแกรมโดยการไปที่แอปพลิเคชันที่มีรูปรถเข็นสีเขียว



รูปที่ 4.1 หน้าแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเปิดโปรแกรมขึ้นมาจะได้หน้าหลักดังรูปที่ 4.2 โดยจะมีเมนูหลักรวมกัน 2 เมนู คือ

1. Admin
2. Staff

โดยระบบ Staff จะรวบรวมการเก็บข้อมูลโดยพื้นฐานทั้งหมดได้แก่ ข้อมูลของสินค้า และข้อมูลของ Staff ซึ่งเมื่อผู้ใช้งานคนไหนไม่ได้ทำงานแล้วสามารถลบข้อมูลของผู้ใช้งานคนนั้นออกไปได้ โดยไม่ต้องกังวลว่าจะมีการนำข้อมูลไปใช้ที่อื่น และระบบ Staff ใช้เพื่อแจ้งเตือนพนักงานว่าสินค้าที่ได้ตั้งค่าไว้หมดแล้ว

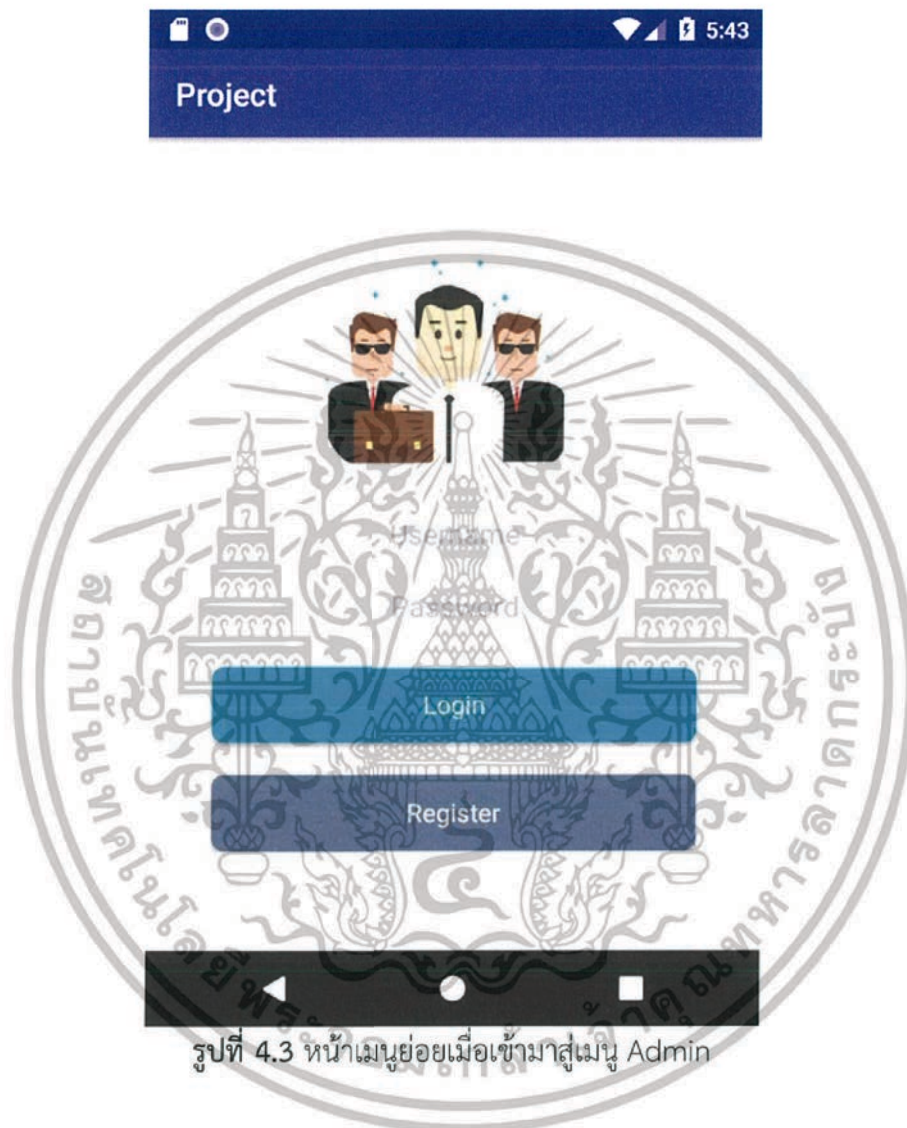


รูปที่ 4.2 หน้าหลักของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การเข้าสู่เมนู Admin

เมื่อเข้าสู่เมนู Admin ดังรูปที่ 4.3 จะต้องมีการ log in เมื่อเป็นสมาชิกแล้วสามารถใส่ Username และ Password แล้วกด log in เข้าไปได้เลย ถ้ายังไม่ได้เป็นสมาชิก สามารถกดสมัครได้ที่ Register แต่การใช้งานของระบบ Admin นั้นจะต้องเป็นผู้จัดการเท่านั้น ผู้ใช้งานที่เป็นพนักงานธรรมดาจะใช้ไม่ได้



รูปที่ 4.3 หน้าเมนูย่อยเมื่อเข้ามาสู่เมนู Admin

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 การเข้าสู่ระบบเมื่อไม่ได้เป็นสมาชิก

เมื่อไม่ได้เป็นสมาชิกต้องกดเข้าสู่เมนู Register จะพบกับหน้า สมัครสมาชิก ดังรูปที่ 4.4

Project

Register

Name Surname
Your Name Your Surname

Age
Age

Username
Username

Password Re-Password
Password Re-Password

OK

รูปที่ 4.4 หน้าสมัครสมาชิกของ Admin.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 เมื่อ log in เข้าสู่ระบบได้แล้ว

เมื่อกด log in เข้ามาในระบบได้แล้ว จะพบกับหน้าโปรแกรม ดังรูปที่ 4.5



รูปที่ 4.5 หน้าโปรแกรมเมื่อเข้าสู่ระบบ Admin

ในส่วนการทำงานของระบบ Admin สามารถแก้ไขข้อมูลของสินค้าได้ โดยการกดไปที่ตัวสินค้า จะพบกับหน้าการแก้ไขข้อมูลสินค้า ทั้งเปลี่ยนชื่อสินค้า หรือรูปสินค้า ดังรูปที่ 4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 หน้าการแก้ไขข้อมูลสินค้า

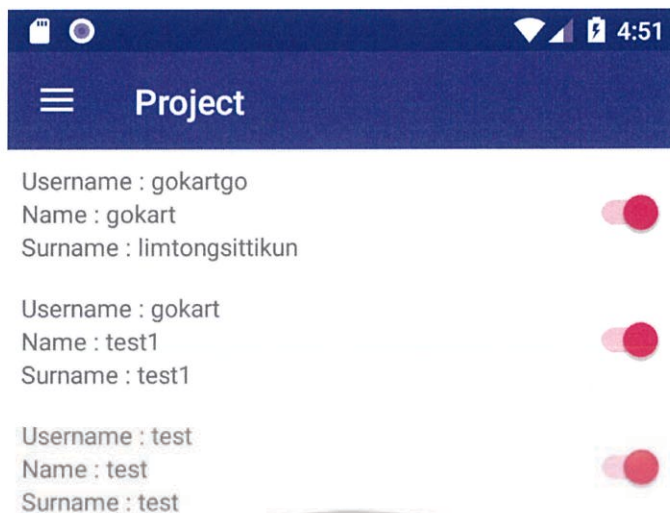
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้ในส่วนการทำงานของ Admin ยังมีการลบข้อมูลของ Staff ออก เมื่อ Staff คนนั้นไม่ได้ทำงานให้กับห้างสรรพสินค้า นั้น ๆ แล้ว เพื่อให้เกิดความปลอดภัยแก่ข้อมูลของห้างสรรพสินค้า สามารถกดได้ที่ปุ่มขีด 3 แถบเรียงกันที่มุมฝั่งซ้ายบนสุดของแอปพลิเคชัน ดังที่วงสี่เหลี่ยมในรูปที่ 4.5 จะเห็นเมนูย่อย Manage Staff และ log out ดังรูปที่ 4.7 และเมื่อกดเข้าไปแล้วจะเห็นหน้าต่างที่ดูรายชื่อ Staff และสามารถลบได้ ดังรูปที่ 4.8 และ 4.9



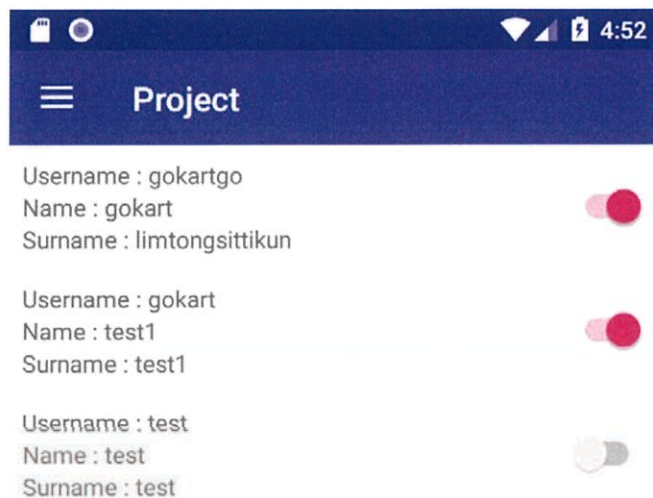
รูปที่ 4.7 หน้าเมนูย่อยของระบบ Admin

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 หน้าหน้าตา Manage Staff

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 หน้าหน้าต่าง Manage Staff และการลบข้อมูล Staff

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

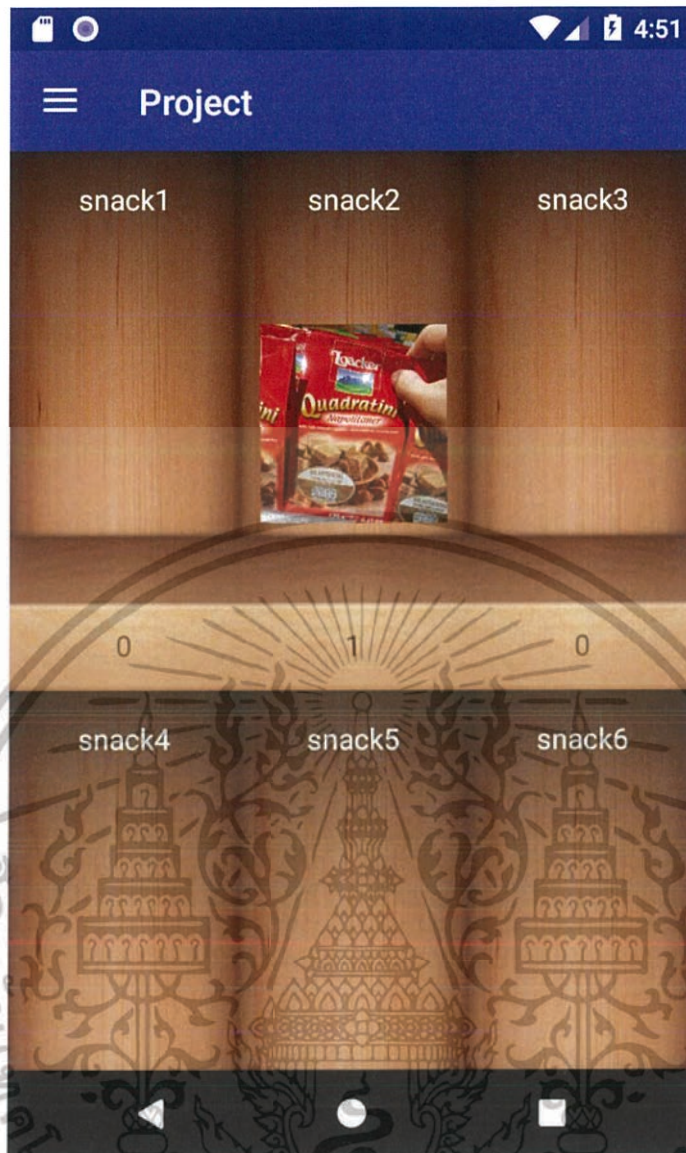
4.3 การเข้าสู่เมนู Staff

เมื่อเข้าสู่เมนู Staff จะเห็นหน้าต่างดังรูปที่ 4.10 ให้ทำการใส่ Username และ Password แล้วกด log in เมื่อเป็นสมาชิกอยู่แล้ว แต่ถ้ายังไม่เป็นสมาชิก ให้กด Register และทำการสมัครเหมือนการสมัครของระบบ Admin ในรูปที่ 4.4 ในส่วนของระบบ Staff นั้น การทำงานจะคล้าย ๆ ระบบ Admin เพียงแต่จะไม่มีหน้าต่าง Manage Staff เหมือนระบบ Admin



รูปที่ 4.10 หน้าระบบ Staff

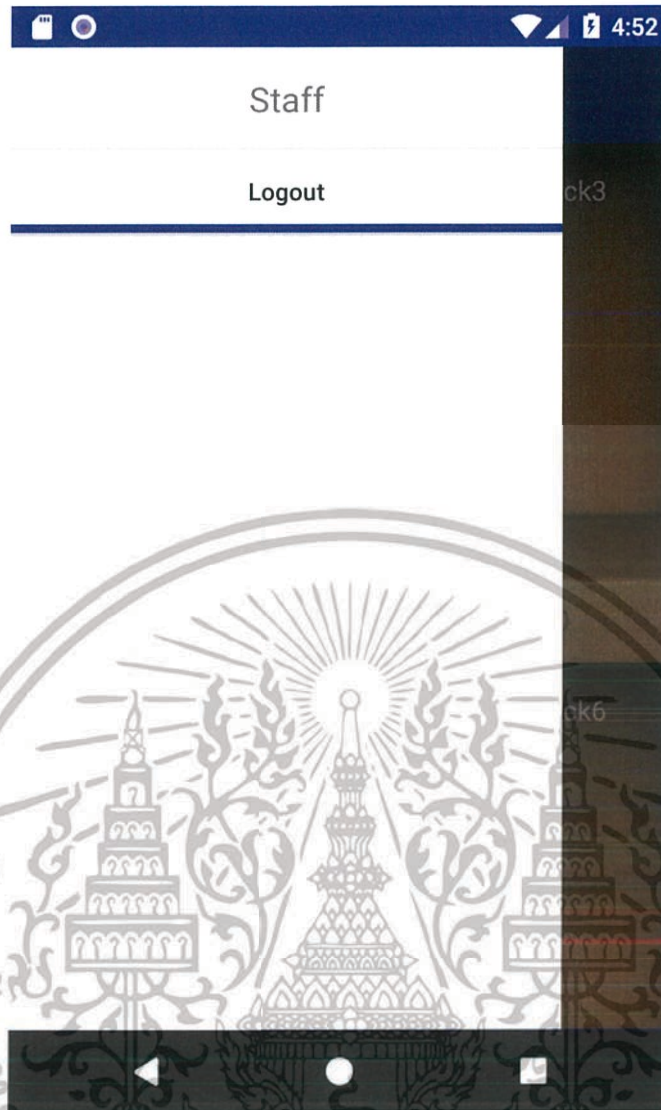
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 หน้าโปรแกรมเมื่อเข้าสู่ระบบ Staff

เมื่อเข้าสู่ระบบมาแล้วจะเห็นหน้าต่างข้อมูลของสินค้าต่าง ๆ ที่ได้ตั้งค่าไว้ ดังรูปที่ 4.11 จะมีการแจ้งเตือนเมื่อสินค้าหมด ในส่วนของเมนูย่อยของระบบ Staff เมื่อกดขีดสามขีดมุมบนซ้าย ของแอปพลิเคชันนั้น จะไม่พบหน้าต่าง Manage Staff ตามที่ได้กล่าวไว้ ดังรูปที่ 4.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 หน้าเมนูย่อยของระบบ Staff ที่ไม่มีหน้าต่าง Manage Staff

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการดำเนินโครงการ

5.1 สรุปผลการทดลอง

โครงการเรื่องระบบตรวจสอบสินค้าในชั้นวางสินค้าของห้างสรรพสินค้า (Department store shelf product checking system) นำเทคโนโลยี IoT (Internet of things) มาประยุกต์ใช้ในการจัดการภายในร้านเพื่อให้เกิดความสะดวกรวดเร็ว และมีประสิทธิภาพมากขึ้น โดยระบบจะมีการติดตั้งเซนเซอร์ ไร้ที่ชั้นวางของเพื่อตรวจสอบสินค้าเป็นจำนวน 9 ช่อง ซึ่งสินค้าในชั้นวางสินค้ามีหลากหลายชนิด เมื่อสินค้าหมดจะมีการแจ้งเตือนผ่านแอปพลิเคชัน

จากการทดลองใช้ระบบตรวจสอบสินค้าในชั้นวางสินค้าของห้างสรรพสินค้านั้น เมื่อสินค้าหมดจะมีการแจ้งเตือนไปที่แอปพลิเคชัน ซึ่งถือว่าสำเร็จไปได้ด้วยดีตามที่คาดหวังไว้

5.2 ปัญหาและอุปสรรคของการดำเนินโครงการ

ในการทำโครงการตรวจสอบสินค้าในชั้นวางสินค้าของห้างสรรพสินค้า พบปัญหาและอุปสรรคของการดำเนินโครงการที่เกิดขึ้นคือ

5.2.1 ชั้นวางสินค้ายังไม่เป็นแบบมาตรฐานที่ใช้ตามห้างสรรพสินค้าทั่วไป

5.2.2 การแจ้งเตือนของแอปพลิเคชันไม่สามารถแจ้งเตือนเมื่อปิดแอปพลิเคชัน แต่สามารถแก้ไขได้แล้ว

5.2.3 ยังไม่สามารถตรวจสอบว่าเป็นสินค้าชิ้นไหนได้ ข้อจำกัดเลยสูง เมื่อเอาสินค้าอื่นมาแทนเลยยังตรวจสอบไม่ได้ว่าเป็นสินค้าอะไร

5.3 แนวทางการพัฒนาโครงการ

ระบบที่พัฒนาขึ้นจะมีการใช้ เทคโนโลยีการประมวลผลภาพ (Image processing) ในการเพิ่มประสิทธิภาพการประมวลผลภาพและวิเคราะห์ข้อมูลที่ได้จากภาพในระบบ เพื่อแก้ปัญหาสินค้าในชั้นวางไม่ตรงกันกับที่ตั้งไว้แต่แรกหรือเป็นของชนิดอื่น และพัฒนาให้มีการเก็บข้อมูลนับจำนวนสินค้าทั้งหมด หรือนำมาเติมของในชั้นวางบ่อย ๆ เพื่อนำข้อมูลไปใช้งานในด้านการตลาดว่าสินค้าชนิดไหนขายดี สามารถต่อยอดหรือเพิ่มฟังก์ชันในการทำงานตามความต้องการของผู้ใช้งานเพื่อให้เกิดประโยชน์และเพิ่มความสะดวกสบายในชีวิตประจำวันให้แก่ผู้ใช้งานโดยสารได้อีกในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] วิวัฒน์ มีสุวรรณ. (2559). อินเทอร์เน็ตเพื่อสรรพสิ่ง (Internet of Things) กับการศึกษา Internet of Things on Education. วารสารวิชาการนวัตกรรมสื่อสารสังคม, 4(2), 84-87.
<https://www.tci-thaijo.org/index.php/jcosci/article/download/93106/72931/>
- [2] Arduino Mega 2560, เข้าถึงได้จาก :
https://www.gravitechthai.com/product_detail.php?d=55
- [3] วิธีการใช้งาน Load Cell กับ HX711 Amplifier Module, เข้าถึงได้จาก :
<https://www.thaieasyelec.com/article-wiki/review-product-article/how-to-use-load-cell-and-hx711-amplifier-module.html>
- [4] NodeMCU คืออะไร, เข้าถึงได้จาก : <https://poundxi.com/nodemcu-คืออะไร>
- [5] การใช้ไมโครตรวจจับสัญญาณและเซนเซอร์อัลตราโซนิก HC-SR04, เข้าถึงได้จาก :
<http://aimagin.com/blog/อัลตราโซนิก-hc-sr04/?lang=th>
- [6] Arduino Basic [EP0] : Arduino คืออะไร ? ทำอะไรได้ ? มีกี่แบบ ?, เข้าถึงได้จาก :
<https://www.arduinospro.com/article/7/arduino-basic-ep0-arduino-คืออะไร-ทำอะไรได้-มีกี่แบบ>
- [7] Firebase คืออะไร และมีข้อดีอย่างไรบ้าง ?, เข้าถึงได้จาก :
<https://www.softmelt.com/article.php?id=588>
- [8] เริ่มต้นสร้าง Android Application พื้นฐานด้วย Android Studio (Lab 3SB04), เข้าถึงได้จาก :
<https://medium.com/@palmez/เริ่มต้นสร้าง-android-application-พื้นฐานด้วย-android-studio-lab-3sb04-3fda43b07a1>
- [9] Node.js คืออะไร ?, เข้าถึงได้จาก : <https://www.softmelt.com/article.php?id=582>
- [10] รู้จัก และเริ่มต้นกับ Heroku, เข้าถึงได้จาก : <http://thaiopensource.org/รู้จัก-และเริ่มต้นกับ-heroku/>
- [11] มาเรียนรู้ Git แบบง่าย ๆ กันเถอะ, เข้าถึงได้จาก : <https://blog.nextzy.me/มาเรียนรู้-git-แบบง่าย ๆ กันเถอะ-427398e62f82>
- [12] Arduino Basic [EP1] : Arduino IDE วิธีติดตั้ง (Windows), เข้าถึงได้จาก :
<https://www.arduinospro.com/article/2/arduino-basic-ep1-arduino-ide-วิธีติดตั้ง-windows-2>
- [13] ขั้นตอนการติดตั้ง Android Studio 1.0 สำหรับมือใหม่, เข้าถึงได้จาก :
<https://devahoy.com/posts/how-to-setup-and-install-android-studio/>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [14] รู้จัก Firebase Realtime Database ตั้งแต่ Zero จนเป็น Hero, เข้าถึงได้จาก :
<https://medium.com/firebasethailand/รู้จัก-firebase-realtime-database-ตั้งแต่-zero-จนเป็น-hero-5d09210e6fd6>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

Poster



Department of Computer Engineering
(Information Engineering)

CE 6132


Title: Department Store Shelf Product Checking System

Mr.Kritchanon Limtongsittikun, and Ms.Pavichaya Chuen-narumitr
Advisor: Assoc.Prof.Dr.Chawalit Benjangkprasert

Abstract

The objective of the project is to study and develop a web application by design a notification system when the product runs out of shelves. It facilitates and design a solution for those who want to open a small to large convenience store with less number of employees for checking the product on the shelves, reduce costs, and more convenient for customers. When the products on the shelves are empty, the customers do not need to walk to the staff. IoT (internet of things) is applied in the shop management for quick, easy access, and also more effective. The system will be equipped with sensors on the shelves to detect when the products are exhausted. There will alert through the application to allow employees to refill without walking to check the product.

Results



1. Creating of the shelf model that can verify the remaining products in the rack is successful.
2. The application can notify when the product runs out.
3. The program can identify the location and product's name that the employees have to refill the product to the shelves.


Introduction


At present, most of everything is connected by the internet. And mobile phones began to play a more active role in life, therefore, it has applied the concept of the internet of thing to be used in conjunction with the trading business to increase the convenience of administration and service. The definition of the internet of thing is that things are linked by the internet. It can be applied to a variety of trading businesses, such as in department stores when the product on shelf runs out, was unable to know when or when. If using the concept of the internet of thing to use when the product on the shelves is almost out of stock, there will be a notification and display the location to the application immediately. And also allows us to analyze customer behavior again when the product is gone quickly. It can be sent to the next market analyst. Therefore, it has created an application in the notification and collecting information to be used for further trading business.


Conclusion

Department store shelf product checking system using internet of things to be used in the management of the store to ensure convenience and more efficient. The system has sensors installed and placed on each shelf to detect 9 products. When the product runs out, there will be a notification via the application. From the trial of department store shelf product checking, it is considered successful as well as expected.

Methodology



Arduino IDE


Firebase


Android Studio

References

[1] Internet of Things on Education
Available : <https://www.tci-thaijo.org/index.php/jcsoci/article/download/93106/72931/>
(Accessed November 15, 2018)



E-mail: chawalit.be@kmitl.ac.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

ตัวอย่าง วิธีการ Download และติดตั้ง Arduino IDE

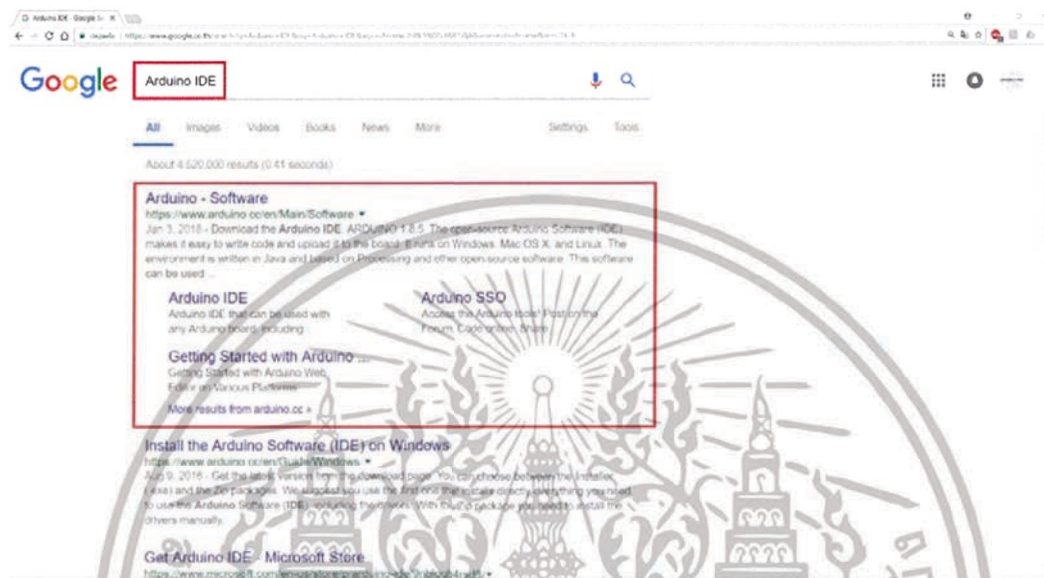


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

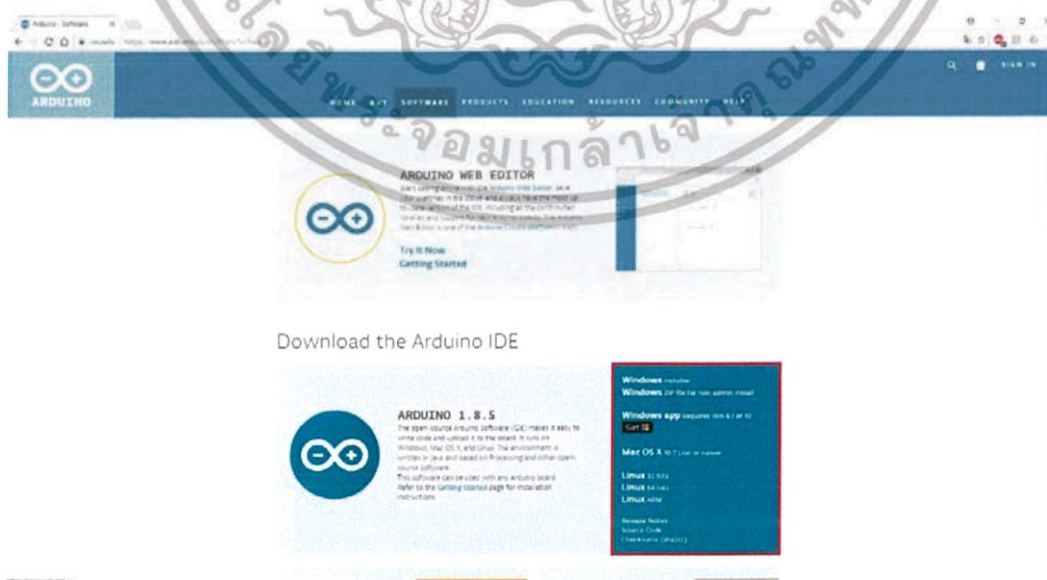
ตัวอย่าง วิธีการ Download และติดตั้ง Arduino IDE

1. ค้นหาว่า Arduino IDE บน Google



รูปที่ ข.1 หน้า ค้นหาการ Download Arduino

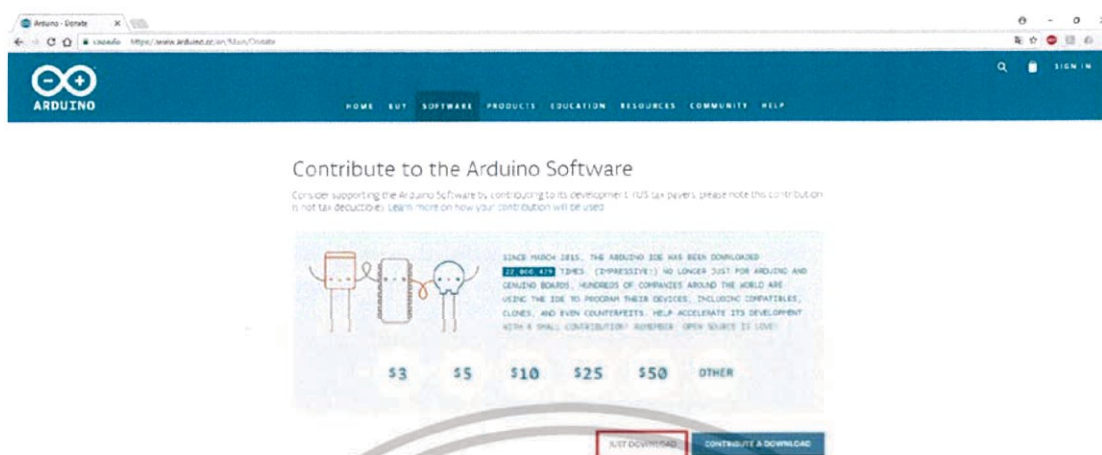
2. จากนั้นเข้าสู่เว็บไซต์ www.arduino.cc จะมีลิงก์สำหรับดาวน์โหลด ให้เลือกตามระบบปฏิบัติการที่ต้องการใช้งาน



รูปที่ ข.2 หน้า เว็บไซต์ Download Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. กวดาวน์โหลดโดยกดที่ JUST DOWNLOAD



รูปที่ ข.3 หน้า Download Arduino

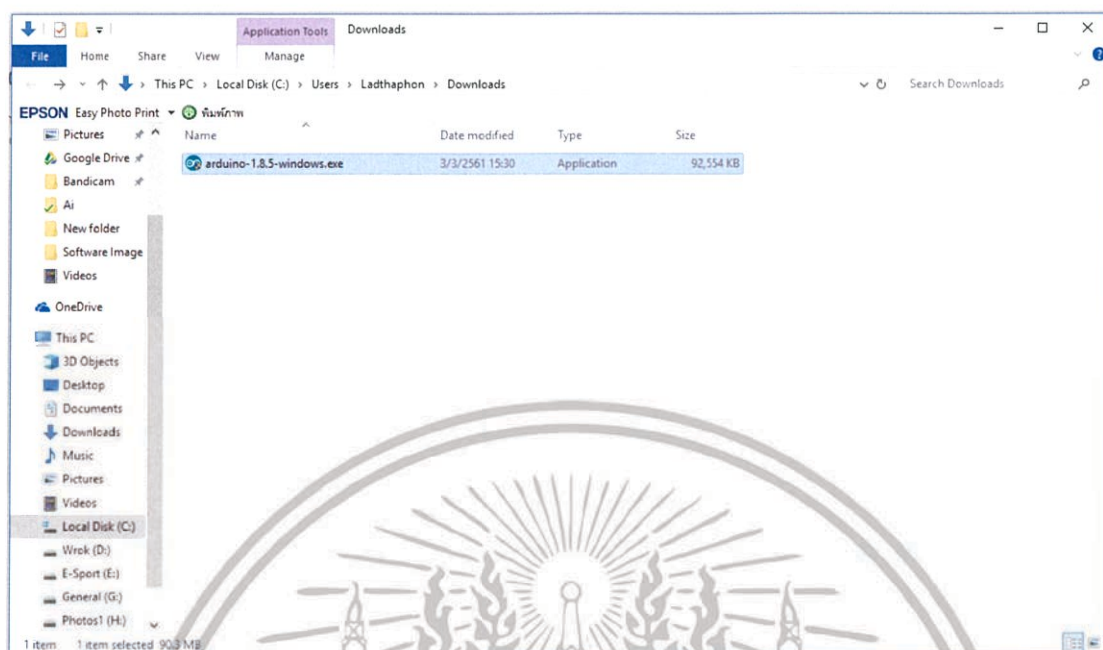
4. รอนจนดาวน์โหลดเสร็จสิ้น



รูปที่ ข.4 หน้า Download เสร็จ

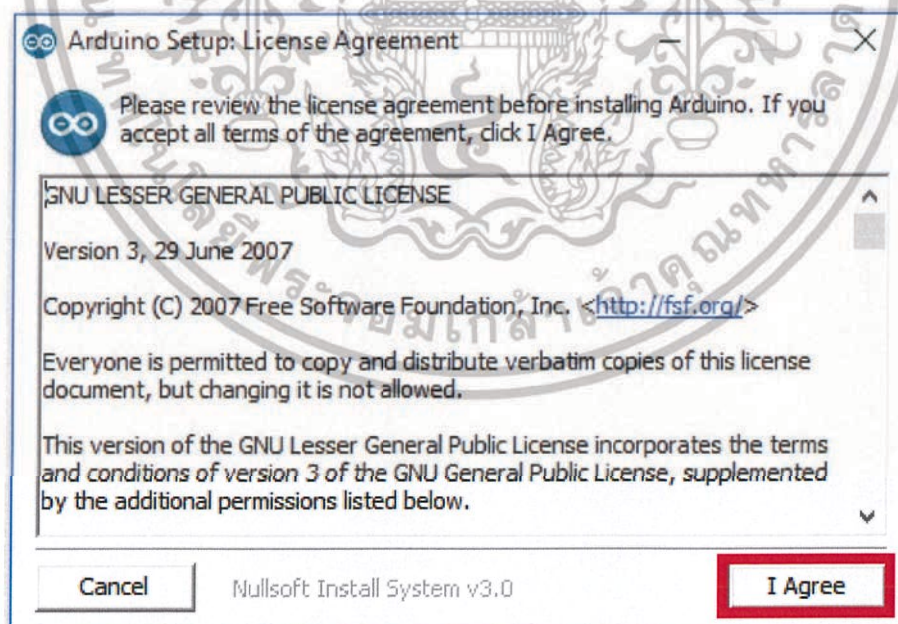
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. จะได้ไฟล์สำหรับติดตั้งโปรแกรม Arduino IDE



รูปที่ ข.5 หน้า ไฟล์สำหรับติดตั้งโปรแกรม Arduino IDE

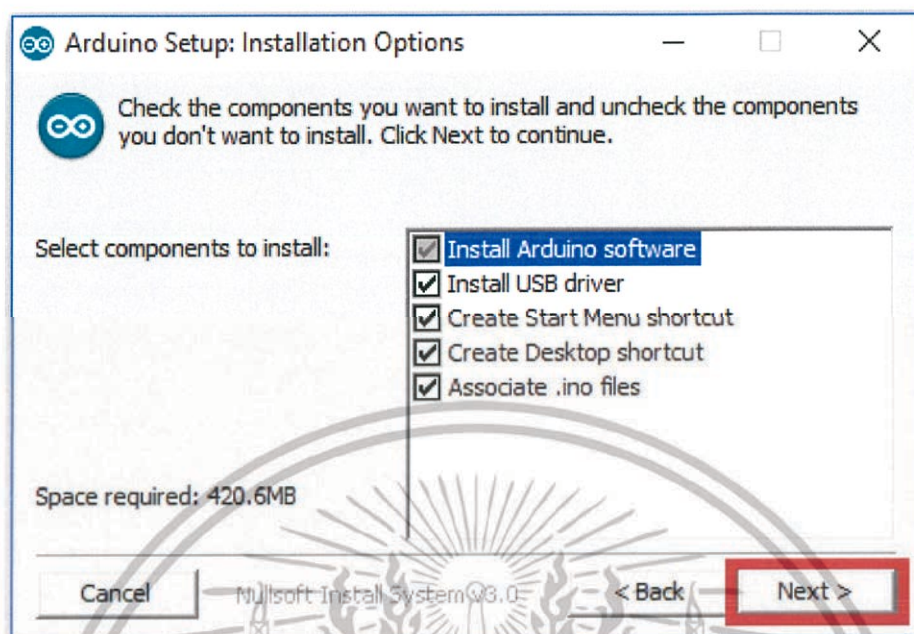
6. เปิดไฟล์ติดตั้งขึ้นมาจะมีหน้าต่างสำหรับติดตั้ง กด I Agree



รูปที่ ข.6 หน้า หน้าต่างสำหรับติดตั้ง

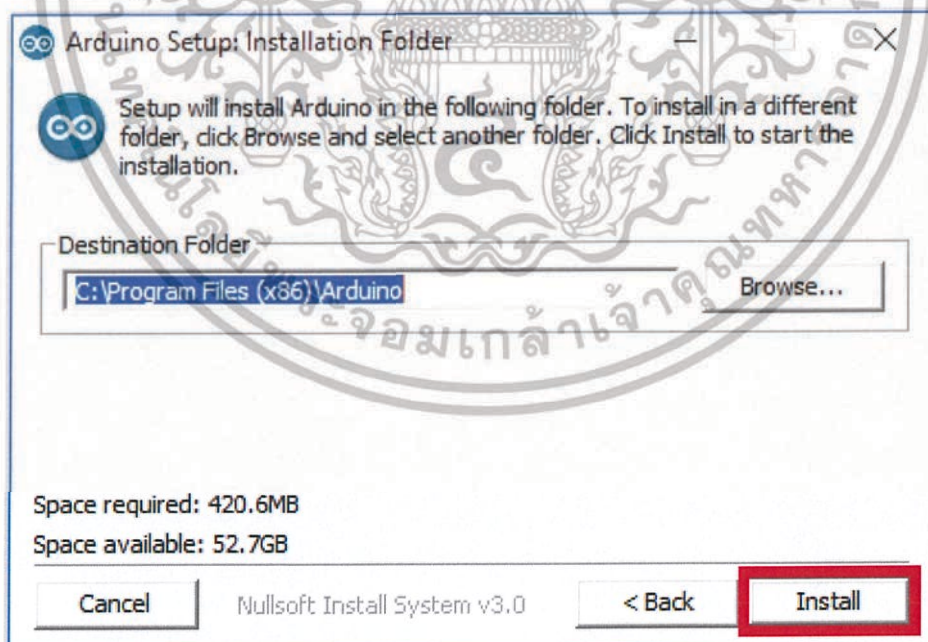
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. กด Next เพื่อไปสู่ขั้นตอนถัดไป



รูปที่ ข.7 หน้า install option

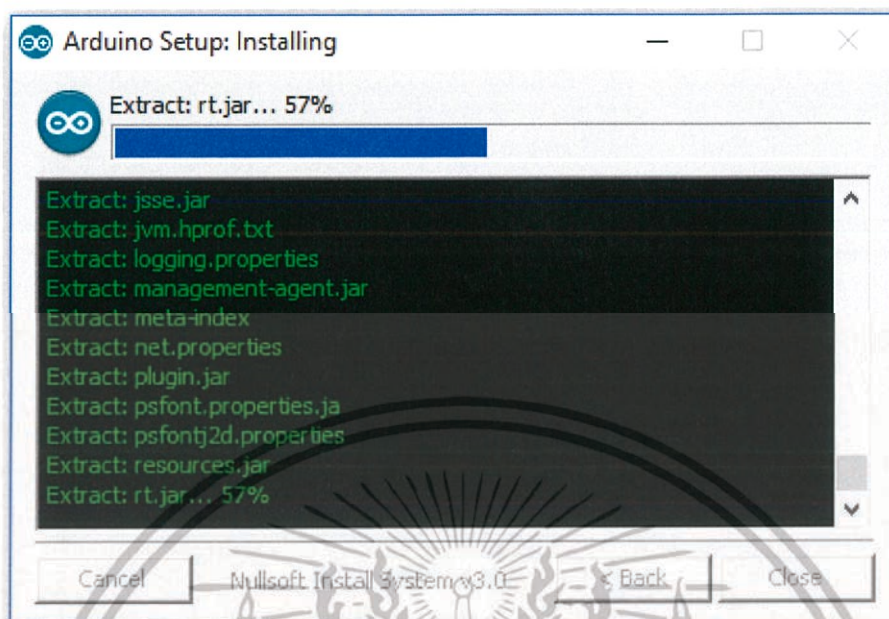
8. เลือกที่ติดตั้งโปรแกรมจากนั้น กด Install



รูปที่ ข.8 หน้า เลือกที่ติดตั้งโปรแกรม

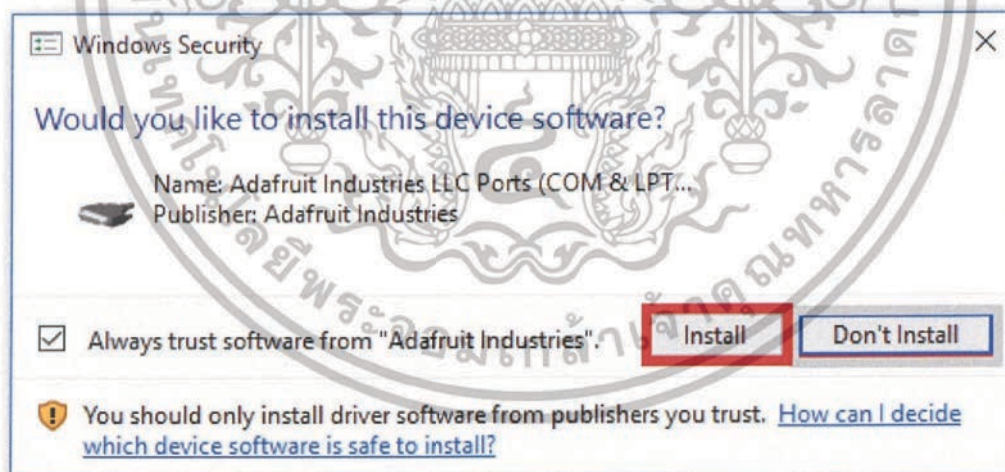
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. รอการติดตั้ง



รูปที่ ข.9 หน้า รอการติดตั้ง

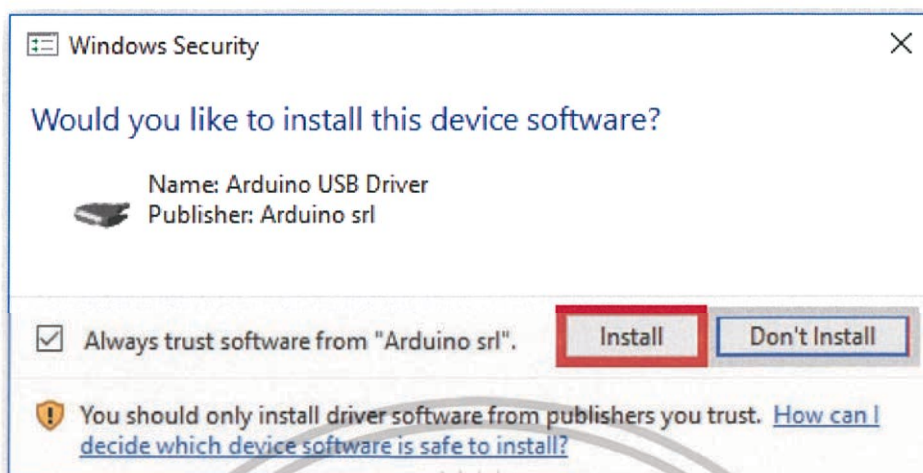
10. กด Install เพื่อติดตั้ง Adafruit Industries LLC Ports



รูปที่ ข.10 หน้า Install เพื่อติดตั้ง Adafruit Industries LLC Ports

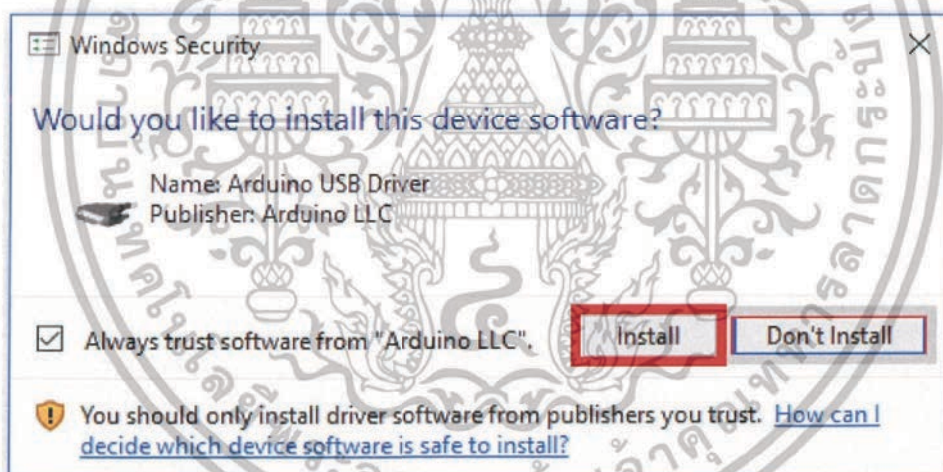
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. กด Install เพื่อติดตั้ง Arduino USB Driver srl



รูปที่ ข.11 หน้า Install เพื่อติดตั้ง Arduino USB Driver srl

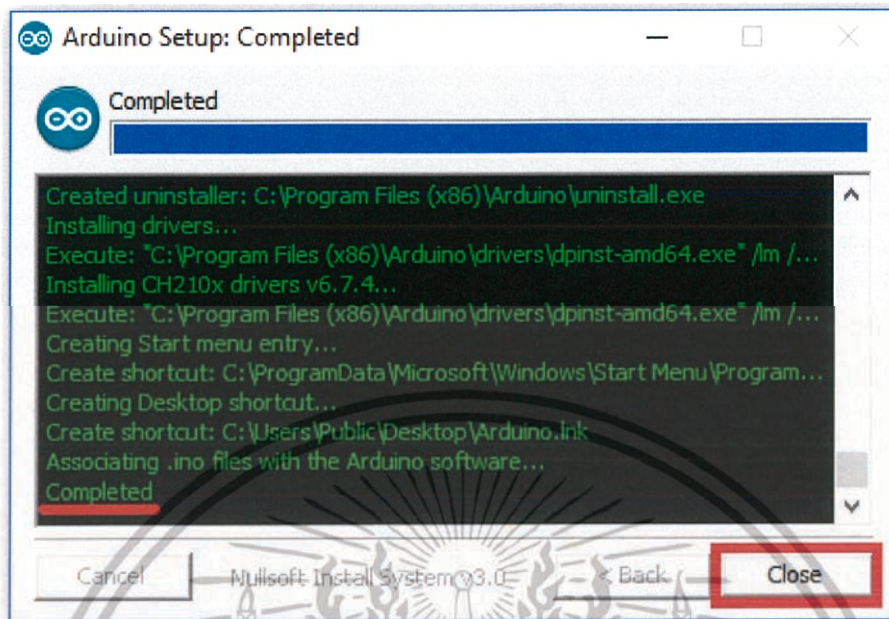
12. กด Install เพื่อติดตั้ง Arduino USB Driver LLC



รูปที่ ข.12 หน้า Install เพื่อติดตั้ง Arduino USB Driver LLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. รอจนแสดงคำว่า Completed เป็นอันเสร็จสิ้นการติดตั้ง จากนั้นกด Close เพื่อปิด



รูปที่ ข.13 หน้า Completed

14. จะได้ไอคอน Arduino บนหน้าจอ



รูปที่ ข.14 หน้าไอคอน Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15. ทดสอบเปิดโปรแกรมจะได้หน้าต่างดังรูป



รูปที่ ข.15 หน้า ช่างในโปรแกรม Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

ตัวอย่าง วิธีการ Download และติดตั้ง Android Studio



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

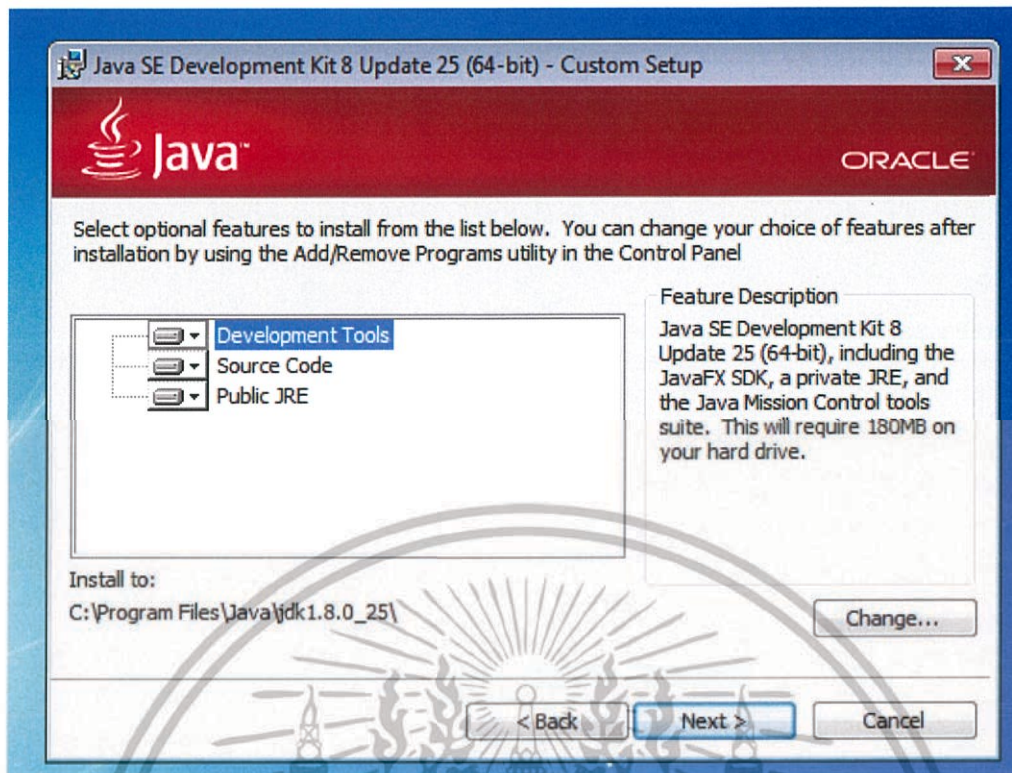
ตัวอย่าง วิธีการ Download และติดตั้ง Android Studio

1. ติดตั้ง Java JDK ถ้าหากยังไม่มี Java อยู่ในเครื่อง หรือมีแล้ว แต่ว่าติดตั้ง Path ยังไม่ถูกต้อง เปิดเว็บ Download Java JDK ขึ้นมา จริง ๆ ค้นหาว่า Download Java JDK ก็มีลิงค์ให้กดแล้ว

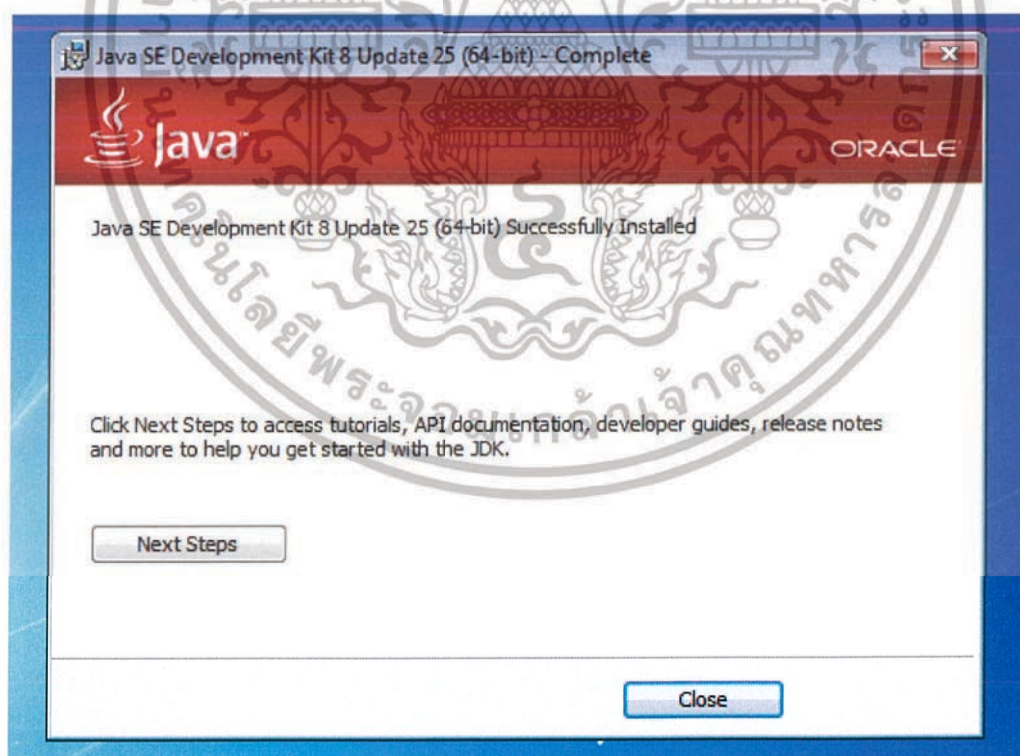


รูปที่ ค.1 หน้า Download Java JDK

2. จากนั้นทำการเลือก Download JDK ปุ่มใหญ่ ๆ และทำการกดยอมรับข้อตกลง License แล้วก็เลือกลิงค์ โหลด 64 Bit (xxx-x64.exe) หรือ 32 Bit (xxx-i586.exe) เมื่อดาวน์โหลดมาแล้ว ก็ทำการติดตั้ง



รูปที่ ค.2 Install JDK

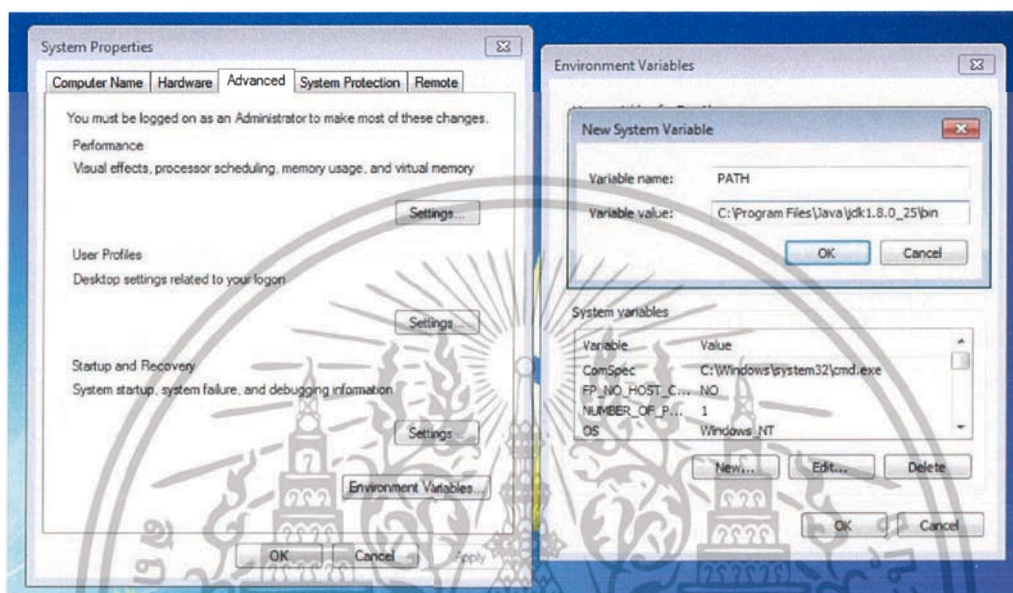


รูปที่ ค.3 Install JDK Fin

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

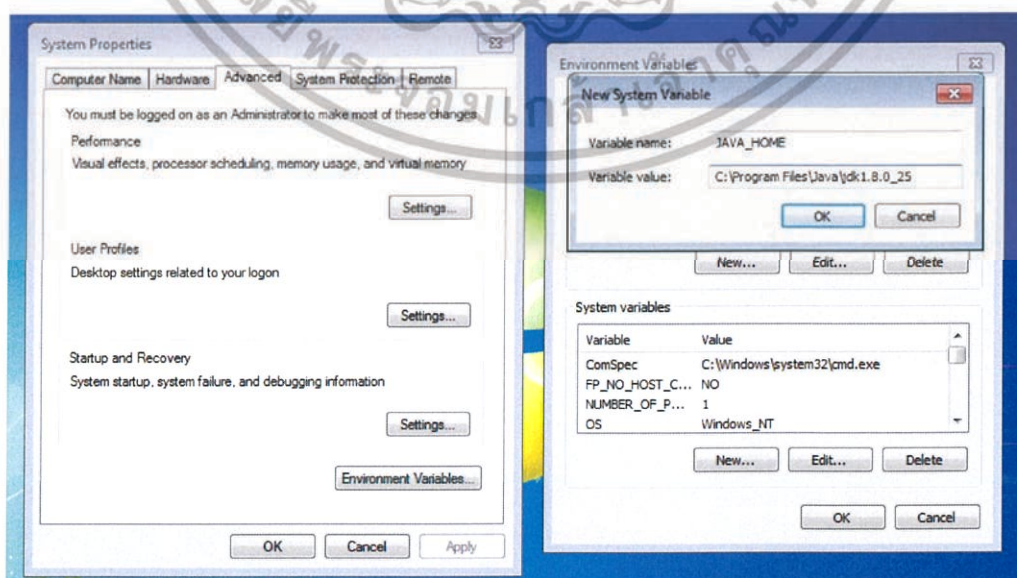
3. เมื่อติดตั้งเสร็จแล้ว ต่อมาทำการ Set Environment Path เพื่อให้เครื่องสามารถเห็น JDK ได้ถูกต้อง โดยทำการคลิกขวาที่ My Computer => เลือก Properties จากนั้นไปที่แท็บ Advanced เลือก Environment Variables...

ให้ทำการเพิ่ม PATH ใหม่ หรือต่อจากของเดิมก็ได้ แต่ต้องใส่ semicolon(;) ก่อนนะ โดยใส่ path ที่อยู่ที่เก็บ Java JDK ไว้ เช่น C:\Program Files\Java\jdk1.8.0_25\bin ตามรูป



รูปที่ ค.4 Set PATH

หรือจะกำหนดเป็น JAVA_HOME และเซต Path เป็น C:\Program Files\Java\jdk1.8.0_25 โดยไม่ต้องมี \bin



รูปที่ ค.5 Set Java Home

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

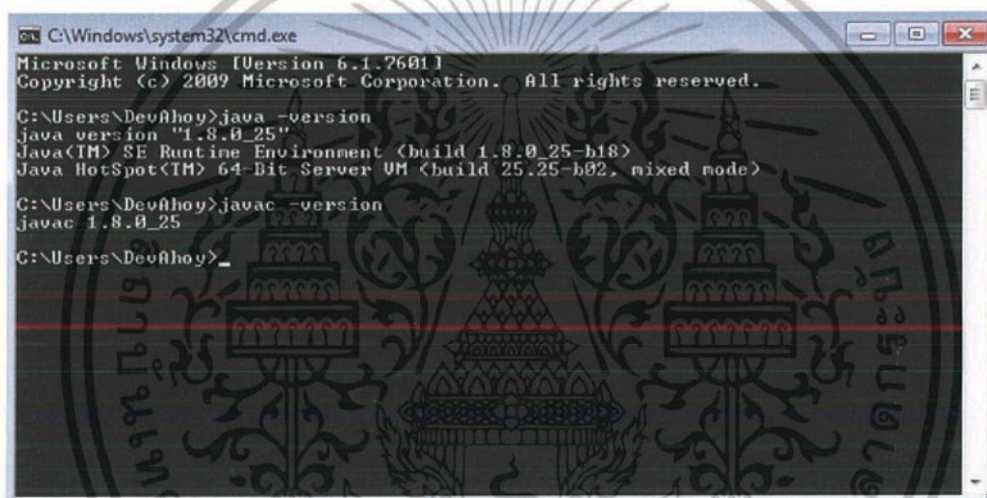
4. เมื่อตั้งค่าเรียบร้อยแล้ว ทดสอบว่าใช้งานได้ โดยการเปิด Command Prompt แล้วพิมพ์

```
java -version
```

และ

```
javac -version
```

ต้องได้ดังภาพ ถ้าไม่ได้แสดงว่ายังไม่ถูก



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

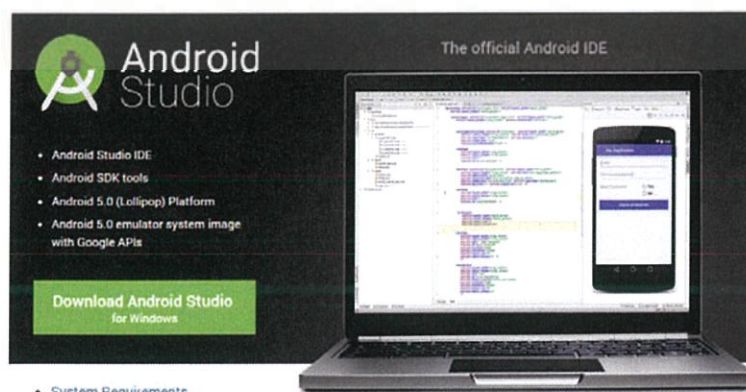
C:\Users\Devahoy>java -version
java version "1.8.0_25"
Java(TM) SE Runtime Environment (build 1.8.0_25-b18)
Java HotSpot(TM) 64-Bit Server VM (build 25.25-b02, mixed mode)

C:\Users\Devahoy>javac -version
javac 1.8.0_25

C:\Users\Devahoy>_
```

รูปที่ ค.6 Finish setup JDK

5. ดาวน์โหลดและติดตั้ง Android Studio เข้าไปหน้าดาวน์โหลดจะเห็นปุ่ม Download อยู่ ก็กดโหลด ไฟล์ ประมาณ 800 MB



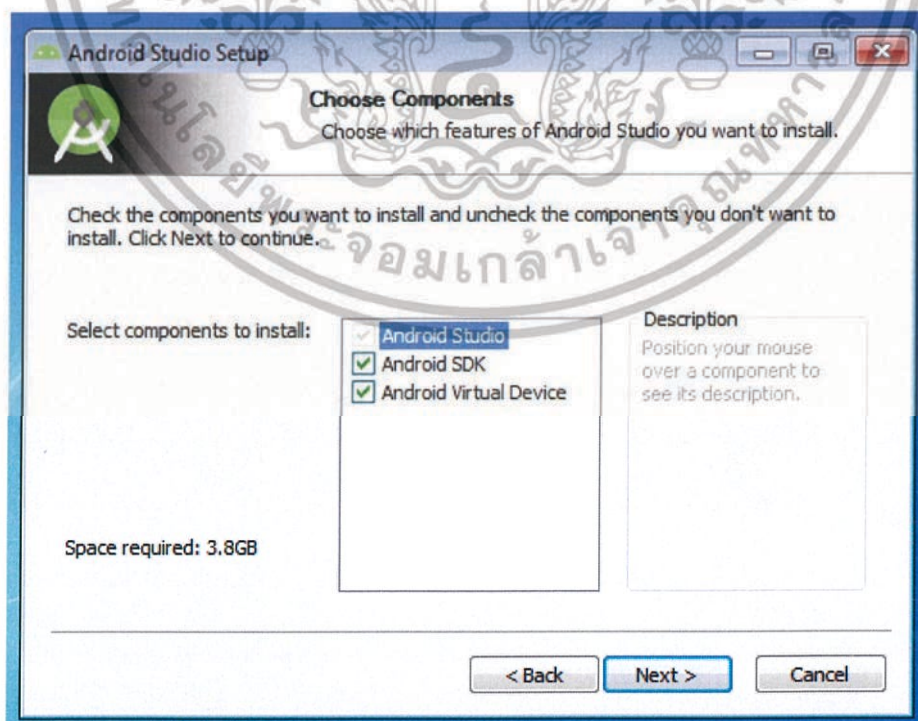
รูปที่ ค.7 Android Studio

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. หลังจากดาวน์โหลดมาแล้ว ก็ทำการติดตั้ง ระหว่างการติดตั้ง จะมีถามด้วย ว่าให้ติดตั้งตัว IDE ไว้ที่ไหน และตัว Android SDK ไว้ที่ไหน ตรงส่วนนี้ก็ใช้ Default ไปก็ได้ หรือจะเลือก Path ที่ต้องการก็ได้ แต่ต้องแน่ใจว่ามี Permission ในการ access read/write ไม่งั้นจะมี Error เวลาเปิด Android Studio



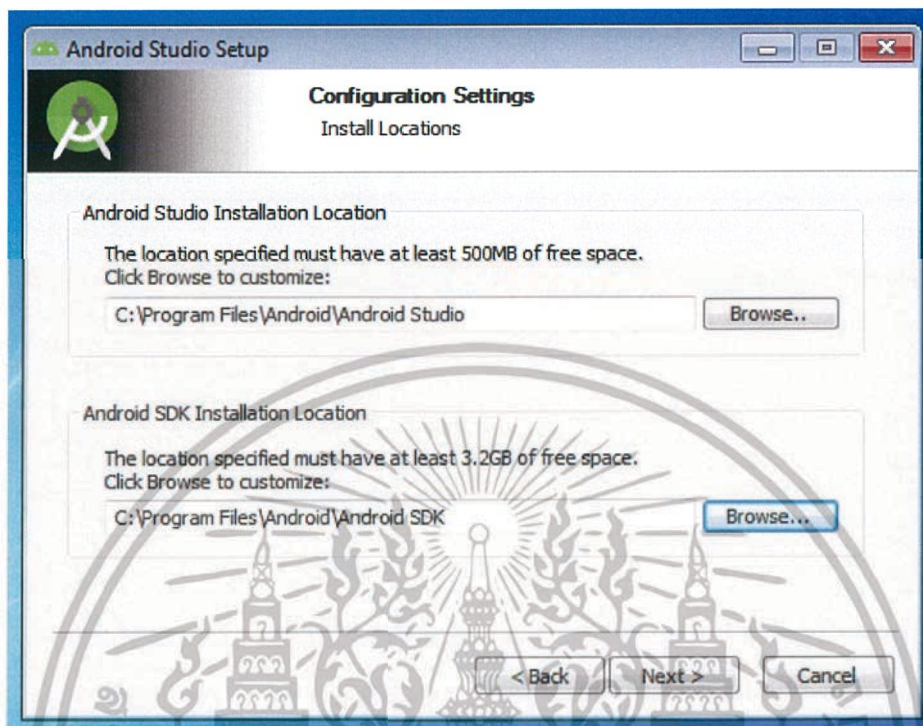
รูปที่ ค.8 Install Android Studio(1)



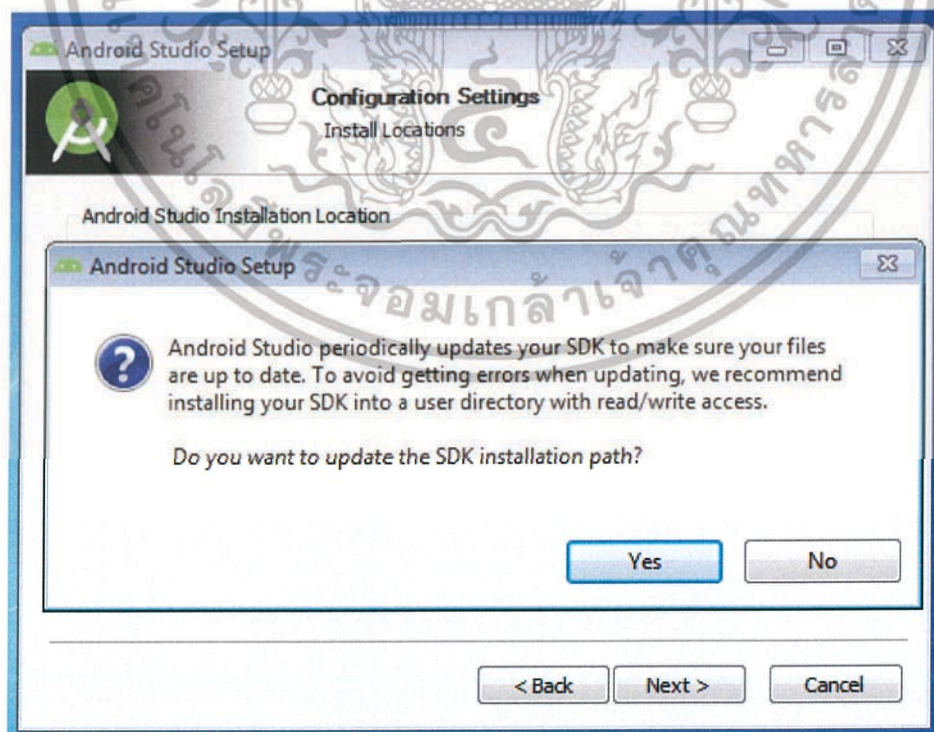
รูปที่ ค.9 Install Android Studio(2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ถ้าเลือกที่เก็บ Android SDK ไว้ที่อื่น จะมี Alert ขึ้นถามว่าแน่ใจหรือเปล่า เพราะบางที่จะเกิดกรณีพวก สิทธิ read/write หรือไม่ก็ต้องรัน Run As Administrator ทุกครั้ง



รูปที่ ค.10 Install Android Studio\Install Android Studio(3)



รูปที่ ค.11 Install Android Studio\Install Android Studio(4)

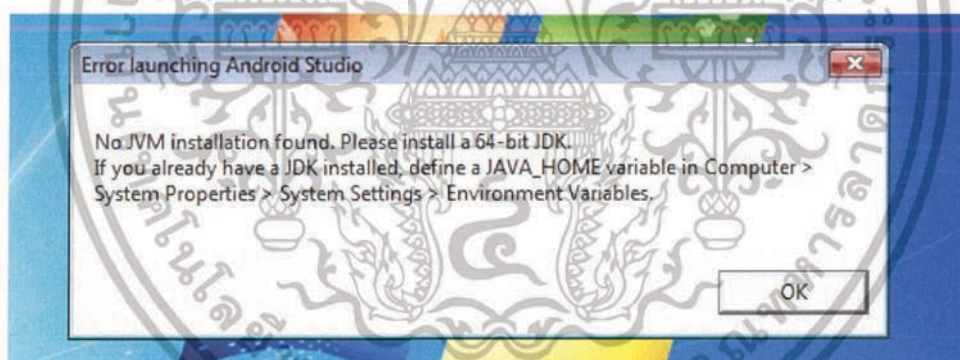
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. เมื่อทำการติดตั้ง Android Studio เสร็จแล้ว เปิดโปรแกรมเลย โปรแกรมจะถามว่าจะให้ Import ไฟล์การตั้งค่าต่าง ๆ ของเวอร์ชันเก่าไว้ไหม ถ้าเพิ่งติดตั้ง ก็ไม่ต้องเลือก แต่ใครที่เคยติดตั้งไว้แล้ว แล้วอยาก Import ก็เลือกได้



รูปที่ ค.12 Completed

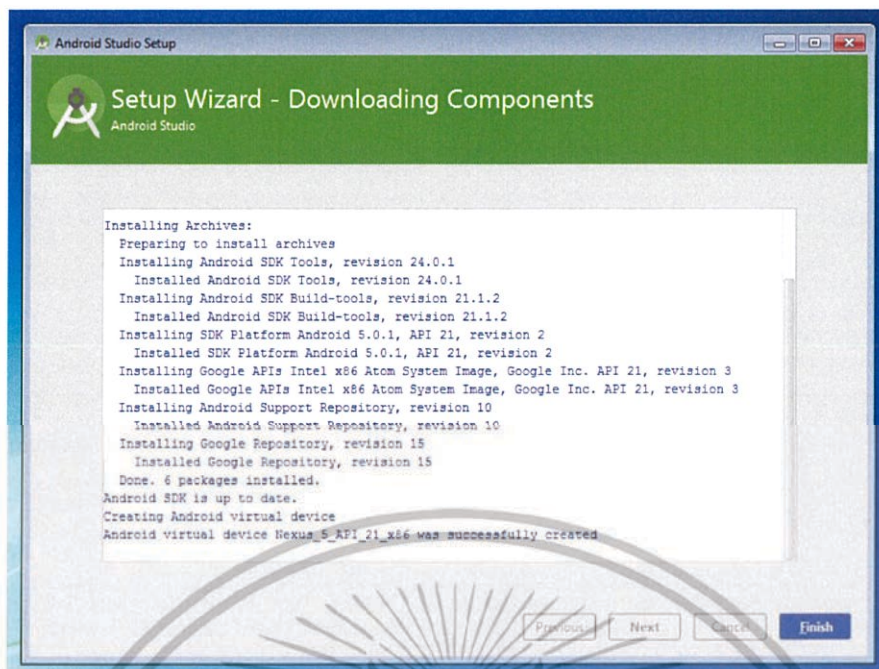
9. หากใครมีปัญหาแบบดั่งภาพ ข้างล่าง เปิด Android Studio ขึ้นมาแล้วเจอแบบนี้ แสดงว่า เซต JAVA_HOME ไม่ถูกต้อง กลับไปอ่าน การติดตั้ง Java JDK อีกรอบ



รูปที่ ค.13 Error can't find JDK

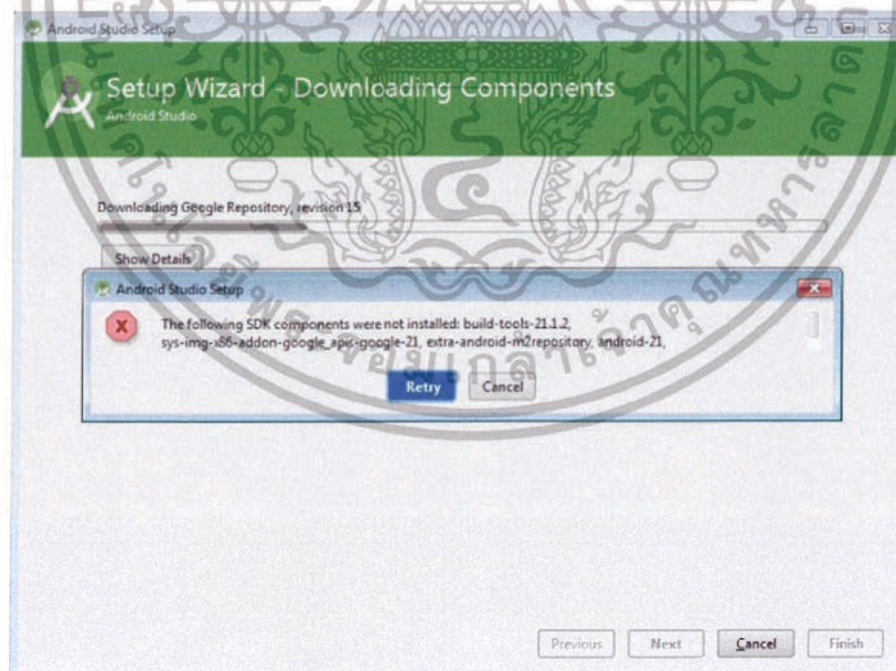
ครั้งแรกตัวโปรแกรมจะทำการดาวน์โหลด Components ต่าง ๆ ที่จำเป็นสำหรับเขียนแอปพลิเคชันให้เองอัตโนมัติ ไม่ต้องทำอะไรเลย มันโหลดให้เองทั้ง Build Tool, API Images แลผมสร้าง Emulator ให้เองด้วยเป็น Nexus 5 Lollipop ช่วงนี้ต้องรอน้อย ความเร็วขึ้นอยู่กับอินเทอร์เน็ตของท่านแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ค.14 Install Component

10. แต่ถ้าหากเจอ Error นี้แทนละ ทำไง ง่าย ๆ ครับ ดูว่า เปิดโปรแกรมด้วย Run As Administrator หรือยัง?



รูปที่ ค.15 Run As Administrator

หากไม่มี Error อะไรเมื่อทำการดาวน์โหลด Components ต่าง ๆ เรียบร้อย ก็กด Finish เป็นอันเสร็จสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง

ตัวอย่าง วิธีการใช้งาน firebase



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง

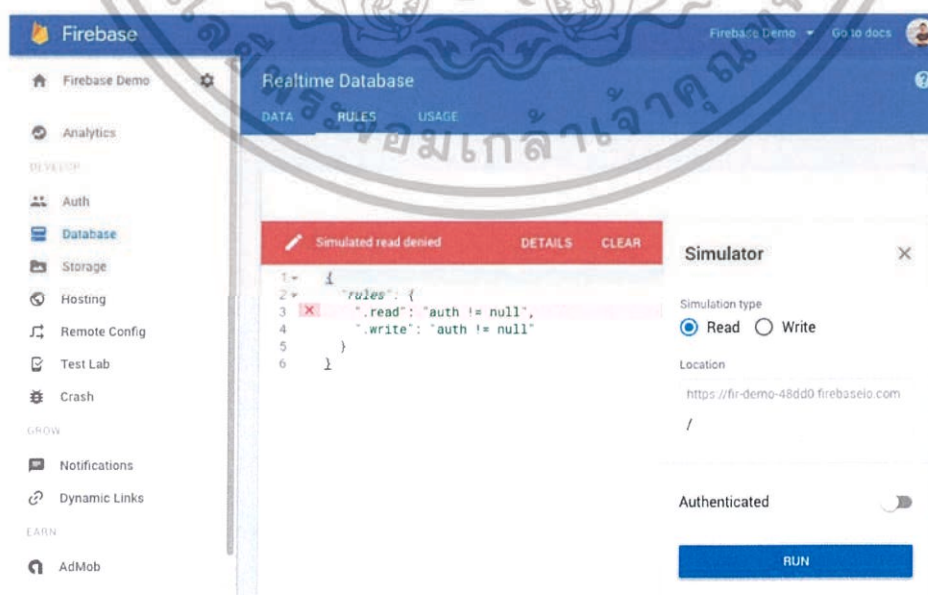
ตัวอย่าง วิธีการใช้งาน firebase

1. เข้าไปที่โปรเจกต์ จากนั้นเลือกเมนู Database แล้วเลือก tab ที่ชื่อว่า RULES จะพบหน้าต่างของประมาณนี้



รูปที่ ง.1 Default rules in Firebase Realtime Database

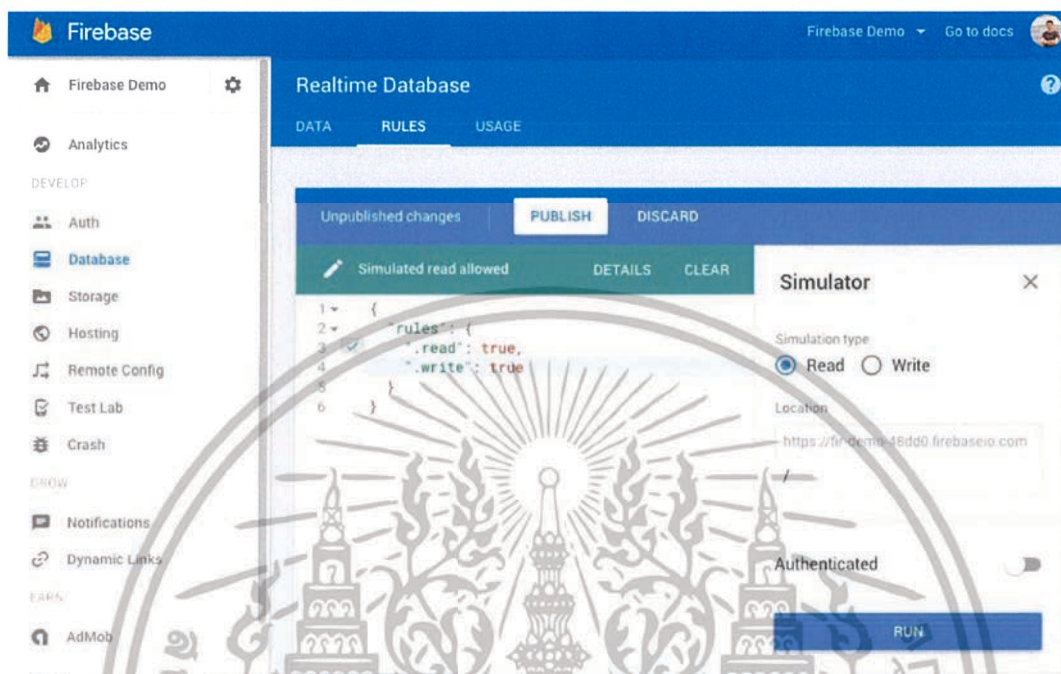
2. ด้านขวามือจะมี simulator ให้ลองทดสอบ rules ที่สร้างขึ้น ทั้งแบบ public หรือแบบ authentication ดังนั้น ลองกด RUN แบบ default rules ก่อนเลย ผลปรากฏว่าถ้าไม่ได้ authentication ก็จะไม่สามารถเข้าถึงข้อมูลได้ ดังรูป ง.2



รูปที่ ง.2 Run default rules ด้วย simulator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ปรับให้เป็น public โดยให้เปลี่ยน rules ตามรูป ง.3 เมื่อกด RUN ก็พบว่า ไม่ต้อง Authentication ก็สามารถเข้าถึงข้อมูลได้แล้ว เมื่อเสร็จแล้วก็กดปุ่ม PUBLISH



รูปที่ ง.3 Run public rules ด้วย simulator

4. การเขียนข้อมูล (Write) เริ่มด้วยการประกาศตัวแปร DatabaseReference รับค่า Instance และอ้างถึง path ที่ต้องการใน database

```
DatabaseReference mRootRef = FirebaseDatabase.getInstance().getReference();
```

จากนั้นก็อ้างอิงไปที่ path ที่ต้องการจะจัดการข้อมูล ตัวอย่างคือ users และ messages

```
DatabaseReference mUsersRef = mRootRef.child("users");
```

```
DatabaseReference mMessagesRef = mRootRef.child("messages");
```

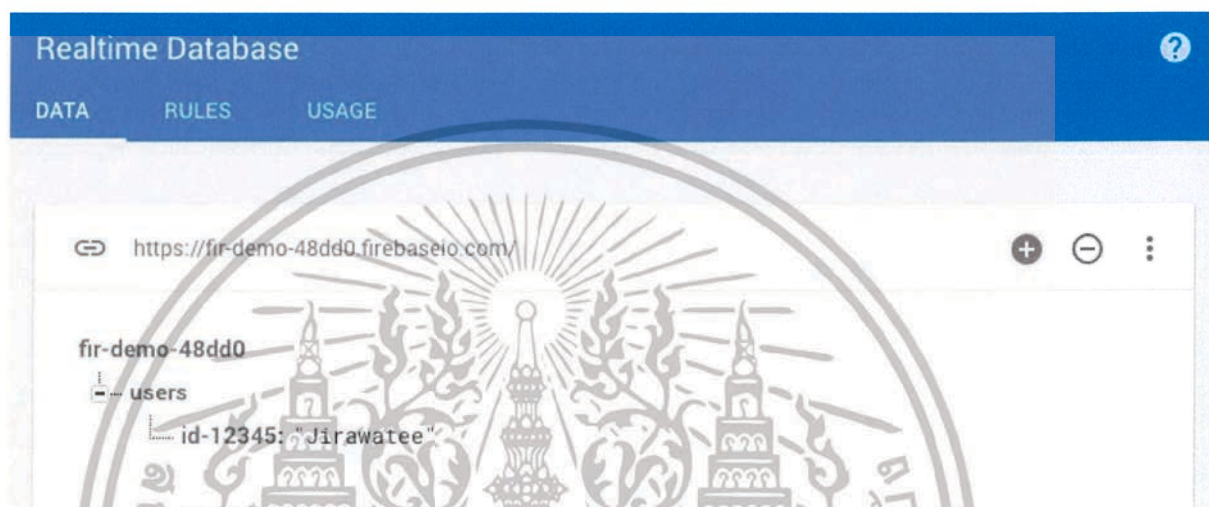
5. การ write, update หรือ delete ข้อมูลใน Firebase Realtime Database จะรองรับค่าหลายประเภท ทั้ง String, Long, Double, Boolean, Map<String, Object> และ List<Object> โดยการ write จะมีด้วยกัน 4 รูปแบบดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1 setValue() เป็นการ write หรือ update ข้อมูล ไปยัง path ที่อ้างอิงได้ เช่น users/<user-id>/<username>

```
mUsersRef.child("id-12345").setValue("Jirawatee");
```

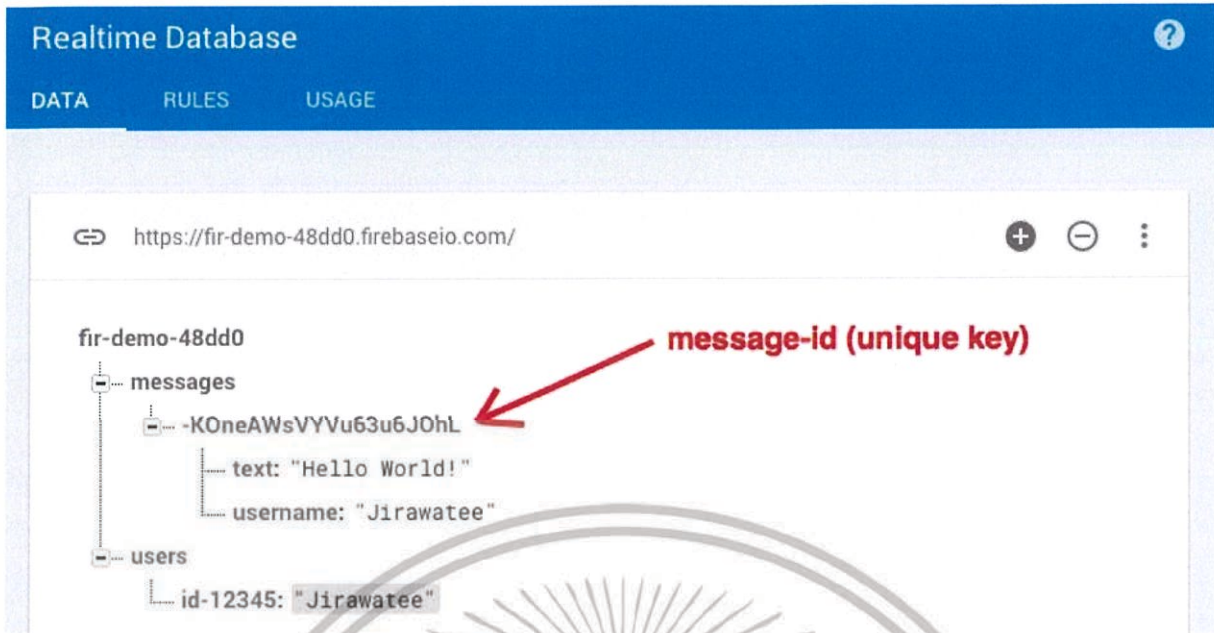
สามารถดูผลลัพธ์แบบ realtime ได้ที่ Firebase Console โดยไปที่เมนู Database แล้วเลือก tab แรก คือ Data ก็จะเห็นข้อมูลทั้งหมด



รูปที่ ๓.๔ ตัวอย่างการ setValue()

5.2 push() เป็นการเพิ่มชุดของข้อมูล ในที่นี้ผมจะสร้าง model object ชื่อ FriendlyMessage ซึ่งจะบรรจุ text และ username ไว้ โดยการ push นั้น Firebase จะสร้าง unique key ของชุดข้อมูลนั้น ๆ เพื่อใช้อ้างอิงต่อไปได้ เช่น messages/<message-id>/<data-model>

```
FriendlyMessage friendlyMessage = new FriendlyMessage("Hello World!", "Jirawatee");
mMessageRef.push().setValue(friendlyMessage);
```



รูปที่ 5.5 ตัวอย่างการ push()

5.3 updateChildren() เป็นการ write หรือ update ข้อมูลบางส่วน(บาง key) ตาม path ที่อ้างถึง โดยไม่ต้องทำการ replace ข้อมูลทั้งหมด และสามารถทำพร้อม ๆ กันได้หลาย object

ตัวอย่างจะเป็นการสร้าง post ใหม่ขึ้นมา โดยจะ write ข้อมูลไป 2 ที่คือ /user-messages/Jirawatee/\$postid และ /messages/\$postid

// push เป็นการ generate \$postid ของ object ชื่อ posts ออกมาก่อนเพื่อใช้ใน // /user-posts/\$userid/\$postid

```
String key = mMessagesRef.push().getKey();
HashMap<String, Object> postValues = new HashMap<>();
postValues.put("username", "Jirawatee");
postValues.put("text", "Hello World!");
```

```
Map<String, Object> childUpdates = new HashMap<>();
childUpdates.put("/messages/" + key, postValues);
childUpdates.put("/user-messages/Jirawatee/" + key, postValues);
```

```
rootRef.updateChildren(childUpdates);
```

สีที่แสดงใน console จะบอกสถานะแตกต่างกันดังนี้

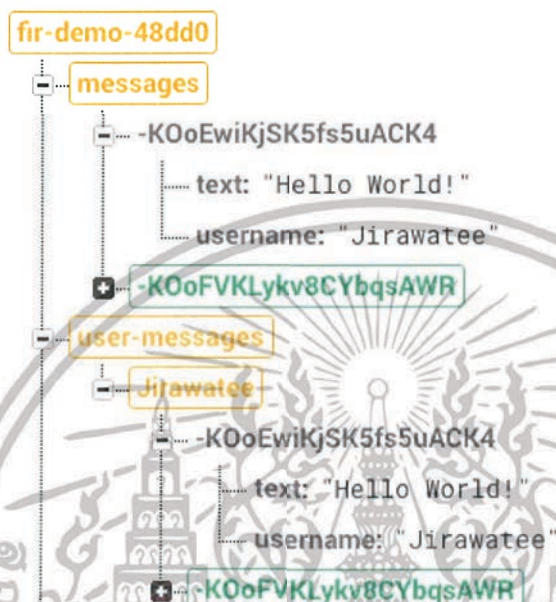
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สีเหลือง: แสดงการปรับปรุง

สีเขียว: แสดงการเพิ่ม

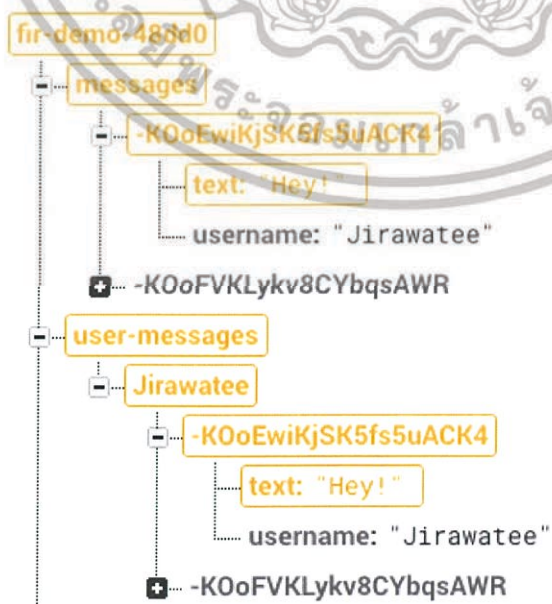
สีแดง: แสดงการลบ

สีน้ำเงิน: แสดงการเคลื่อนย้าย



รูปที่ ๖.6 write ข้อมูลเข้าไปทั้ง 2 objects

ในกรณีที่ต้องการปรับปรุงข้อมูลบางส่วน ก็สามารถทำได้พร้อม ๆ กันได้ โดยจะต้องรู้ username และ message-id เป็นตัวระบุในแต่ละ object



รูปที่ ๖.7 update ข้อมูลบางส่วนทั้ง 2 objects

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 runTransaction() เป็นการอัปเดตข้อมูล ที่มี concurrent เยอะ ๆ ที่อาจเกิดชนกัน เกิดข้อผิดพลาดได้ ตัวอย่างเช่น การกด like และกด unlike ที่โพสต์เดียวกัน เวลาเดียวกัน จะต้องมีการนับยอด like ตลอดเวลา ว่าช่วงเวลานั้นเป็นเท่าไร

```

postRef.runTransaction(new Transaction.Handler() {
    @Override
    public Transaction.Result doTransaction(MutableData mutable) {
        Post p = mutable.getValue(Post.class);
        if (p == null) {
            return Transaction.success(mutable);
        }

        if (p.stars.containsKey(getUid())) {
            // Unlike the post and remove self from likes
            p.starCount = p.starCount - 1;
            p.stars.remove(getUid());
        } else {
            // Like the post and add self to likes
            p.starCount = p.starCount + 1;
            p.stars.put(getUid(), true);
        }

        // Set value and report transaction success
        mutable.setValue(p);
        return Transaction.success(mutable);
    }

    @Override
    public void onComplete(DatabaseError databaseError, boolean b, DataSnapshot
dataSnapshot) {
        if (databaseError != null) {
            databaseError.getMessage()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        Log.w(TAG, databaseError.getMessage());
    } else {
        Log.d(TAG, "Transaction successful");
    }
}
});

```

6. การลบข้อมูล (Delete) การลบข้อมูลนั้น ให้ระบุ path ที่ต้องการจะลบ จากนั้นก็เรียกคำสั่ง `removeValue()` ตัวอย่างเช่น ต้องการลบข้อความทั้งหมดใน object ชื่อ `messages`

```
mMessageRef.removeValue();
```

นอกจากนั้นยังสามารถลบข้อมูล ได้ด้วยการส่งค่า `null` ไปที่ `setValue(null)` และสามารถใช้ค่า `null` กับเทคนิค `updateChildren()` เพื่อลบข้อมูลหลาย ๆ object ได้ด้วย

```

// ลบแบบ setValue()
mMessageRef.setValue(null);
// ลบแบบ updateChildren()
childUpdates.put("/messages/", null);

```

7. การอ่านข้อมูล (Read) เริ่มด้วยการประกาศตัวแปร `DatabaseReference` รับค่า `Instance` และอ้างถึง path ที่ต้องการใน database

```
DatabaseReference mRootRef = FirebaseDatabase.getInstance().getReference();
```

การอ่านข้อมูลใน Firebase Realtime Database จะมี 2 ประเภทแยกตาม Listener ดังนี้

7.1 `ValueEventListener` จะอ่านข้อมูลตั้งแต่เริ่ม และ จะอ่านข้อมูลทุกครั้งที่มีการเปลี่ยนแปลงของข้อมูลทั้งหมดภายใต้ path ที่อ้างถึง วิธีการคือใช้ object ที่อ้างถึงมา `addValueEventListener` โดยจะมี callback 2 แบบ

7.1.1 `onDataChange` จะถูกเรียกตอนเริ่ม และถูกเรียกทุกครั้งที่มีข้อมูลภายใต้ path ที่อ้างถึงมีการเปลี่ยนแปลง

7.1.2 `onCancelled` จะถูกเรียกเมื่อไม่สามารถอ่านข้อมูลจาก database ได้

```
mRootRef.addValueEventListener(new ValueEventListener() {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

@Override
public void onDataChange(DataSnapshot dataSnapshot) {
    String value = dataSnapshot.getValue(String.class);
    mTextView.setText(value);
}

@Override
public void onCancelled(DatabaseError error) {
    mTextView.setText("Failed: " + databaseError.getMessage());
}
});

```

ข้อควรระวังของ ValueEventListener คือไม่ควรอ้างถึง root เนื่องจากมันจะคอยดูข้อมูลทั้งหมดของ database ซึ่งมันอาจมีขนาดใหญ่ และ เปลือง bandwidth โดยใช้เหตุ จึงควรใช้วิธีนี้แบบเจาะจงเฉพาะ หรือ อ่านข้อมูลทั้งหมดครั้งแรก แล้ว removeEventListener() ออกไป

7.2 ChildEventListener จะคอยรับข้อมูลจาก การเพิ่ม, การเปลี่ยนแปลง, การลบ และ การย้าย เฉพาะของ child ที่อ้างถึง วิธีการคือใช้ object ที่อ้างถึงมา addChildEventListener โดยจะมี callback 5 แบบ

- 7.2.1 onChildAdded() จะถูกเรียกเมื่อมีการเพิ่มชุดข้อมูลเข้ามาใน child
- 7.2.2 onChildChanged() จะถูกเรียกเมื่อข้อมูลใน child มีการเปลี่ยนแปลง
- 7.2.3 onChildRemoved() จะถูกเรียกเมื่อข้อมูลใน child ถูกลบ
- 7.2.4 onChildMoved() จะถูกเรียกเมื่อมีการเรียงลำดับของข้อมูลใน child เกิดขึ้น
- 7.2.5 onCancelled() จะถูกเรียกเมื่อโหลดข้อมูลจาก child ไม่สำเร็จ

ตัวอย่างการ addChildEventListener ไปกับการ comment ซึ่งเหมาะมากกับ RecyclerView

```

ChildEventListener childEventListener = new ChildEventListener() {
    @Override
    public void onChildAdded(DataSnapshot dataSnapshot, String previousChildName)
    {
        // A new comment has been added, add it to the displayed list
        Comment comment = dataSnapshot.getValue(Comment.class);
        // ...
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

@Override
public void onChildChanged(DataSnapshot dataSnapshot, String
previousChildName) {

    // A comment has changed, use the key
    // to determine if we are displaying this
    // comment and if so displayed the changed comment.
    Comment newComment = dataSnapshot.getValue(Comment.class);
    String commentKey = dataSnapshot.getKey();
    // ...
}

@Override
public void onChildRemoved(DataSnapshot dataSnapshot) {
    // A comment has changed, use the key
    // to determine if we are displaying this
    // comment and if so remove it.
    String commentKey = dataSnapshot.getKey();
    // ...
}

@Override
public void onChildMoved(DataSnapshot dataSnapshot, String previousChildName)
{
    // A comment has changed position,
    // use the key to determine if we are
    // displaying this comment and if so move it.
    Comment movedComment = dataSnapshot.getValue(Comment.class);
    String commentKey = dataSnapshot.getKey();
    // ...
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

@Override
public void onCancelled(DatabaseError databaseError) {
    Toast.makeText(mContext, "Failed to load comments.",
        Toast.LENGTH_SHORT).show();
};
ref.addChildEventListener(childEventListener);

```

8. การเพิ่ม listener เข้าไปหลายตัว นั้นก็แปลว่าจะต้องมีการ call เกิดขึ้นมากมายตามแต่ event ซึ่ง หากไม่ได้ใช้ หรือออกจากหน้าดังกล่าว ก็ควรจะถอด listener เหล่านั้นออกไปด้วย เพื่อการใช้ bandwidth แบบคุ้มค่ามากที่สุดโดยสามารถถอดออกได้ด้วยคำสั่ง `removeEventListener()`

```

}
@Override
protected void onStop() {
    super.onStop();
    if (mValueEventListener != null) {
        mRootRef.removeEventListener(mValueEventListener);
    }
}

```

9. การอ่านข้อมูลเพียงครั้งเดียว บางครั้งอาจต้องการอ่านข้อมูลแค่ครั้งเดียวและไม่สนใจมันอีก Firebase ได้เตรียม `addListenerForSingleValueEvent()` ที่เมื่อได้ callback แล้วก็จะ remove listener ทิ้งอัตโนมัติ ตัวอย่างเช่น ดึงข้อมูลผู้ใช้งานก่อนทำการโพสต์ว่าเขาผ่านการ sign-in มาหรือยัง หรือยังขาดข้อมูลอะไรที่ต้องการ หากครบถ้วนก็โพสต์ได้ แต่หากไม่ครบอาจพาไปหน้า sign-in หรือพาไปหน้ากรอกข้อมูลให้ครบถ้วน

```

mDatabase.child("users").child(userId).addListenerForSingleValueEvent(new
ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        User user = dataSnapshot.getValue(User.class);
        if (user == null) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        Toast.makeText(NewPostActivity.this, "Error: could not fetch user.",
Toast.LENGTH_LONG).show();
    } else {
        writeNewPost(userId, user.username, title, body);
    }
    finish();
}

@Override
public void onCancelled(DatabaseError databaseError) {
    Log.e(TAG, databaseError.getMessage());
}
});

```

10. การเรียงลำดับและการกรองข้อมูล (Sorting and Filtering)

การ query ข้อมูล ของ Firebase Realtime Database นั้น รองรับการ sort และ filter ได้ อย่างก็ดี การ sort และ filter สามารถทำให้การ query นั้นเข้าได้ ทางที่ศึกษารื่องการทำ index (.indexOn) ไปด้วย จะทำให้การ query นั้นมีประสิทธิภาพมากขึ้น

10.1 การ Sort ข้อมูล มี 3 รูปแบบ ดังนี้ (Ordering Function)

10.1.1 orderByChild() เป็นการเรียงลำดับ value ของ child key ที่ถูกเลือก (การใช้งาน คล้าย WHERE ใน SQL)

10.1.2 orderByKey() เป็นการเรียงลำดับ child key (ใช้เรียง PK เหมาะกับแสดงข้อมูล แบบมี limit หรือแสดง pagination)

10.1.3 orderByValue() เป็นการเรียงลำดับ child value (เหมาะกับการเรียงค่าตัวเลข)

// ตัวอย่างการเรียงลำดับโพสของฉันที่ได้รับคะแนนโหวตมากที่สุด

```
dbReference.child("user-posts").child(getUid()).orderByChild("starCount");
```

10.2 การ Filter ข้อมูล มี 5 รูปแบบ ดังนี้ (Querying Function)

10.2.1 limitToFirst() การระบุจำนวน item ซึ่งจะเรียงลำดับจากแถวแรก

10.2.2 limitToLast() การระบุจำนวน item ซึ่งจะเรียงลำดับจากแถวสุดท้าย

10.2.3 startAt() จะดึงจำนวน item ที่มากกว่า หรือ เท่ากับ ที่ระบุ key หรือ value โดยขึ้นอยู่กับ การ order-by

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.2.4 endAt() จะดึงจำนวน item ที่น้อยกว่า หรือ เท่ากับ ที่ระบุใน key หรือ value โดยขึ้นอยู่กับ การ order-by

10.2.5 equalTo() จะดึงจำนวน item ที่เท่ากับที่ระบุใน key หรือ value โดยขึ้นอยู่กับ การ order-by

```
// ตัวอย่างการดึงข้อมูลโพสต์ 100 อันล่าสุด
databaseReference.child("posts").limitToFirst(100);
```

11. การเปิดใช้งานโหมด offline ลักษณะการทำงาน offline (Persistence Behavior)

Firebase Realtime Database มีการจัดการเรื่องการเชื่อมต่อให้แล้ว เมื่อไม่สามารถเชื่อมต่ออินเทอร์เน็ตหรืออยู่ในโหมด offline ยังสามารถใช้งานแอปพลิเคชันได้ โดยตัว Firebase จะทำการ cache ข้อมูลไว้ทุกการกระทำ และจะทำการ sync ข้อมูลให้อัตโนมัติเมื่อกลับเข้าสู่โหมด online แม้ว่า จะปิดแล้วเปิดแอปพลิเคชันใหม่ก็ตาม มันยอดเยี่ยมมากเลย แค่ประกาศใช้งานง่าย ๆ เพียงบรรทัดเดียว

```
FirebaseDatabase.getInstance().setPersistenceEnabled(true);
*****
```

12. ปัญหาที่หลายคนประกาศ .setPersistenceEnabled แล้วเมื่อออกและเข้ามาใหม่ปรากฏว่า crash โดยแสดง error ประมาณนี้

```
com.google.firebase.database.DatabaseException: Calls to setPersistenceEnabled() must be made before any other usage of FirebaseDatabase instance
```

ผู้เขียนแนะนำให้แก้ปัญหานี้โดย

12.1 สร้าง class ที่ extends Application แล้วไปประกาศ setPersistenceEnabled(true) ที่นั่น

```
class MyApp extends android.app.Application

@Override
public void onCreate() {
    super.onCreate();
    FirebaseDatabase.getInstance().setPersistenceEnabled(true);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12.2 ใน AndroidManifest ให้ไปประกาศ class ในข้อ 1 ใน <application> ซะ

```
android:name="com.example.MyApp"
```

ผู้เขียนทดสอบแล้ว ประกาศทีเดียว จบ

```
*****
```

13. แอป sync ข้อมูลแบบลับ ๆ (Keeping Data Fresh)

Firebase Realtime Database จะ sync และเก็บข้อมูลใน local ของ client เฉพาะข้อมูลที่ตัว listener ทำงานอยู่ แต่สามารถจะ sync ข้อมูลส่วนอื่นที่ยังไม่ได้ active ได้ โดยให้อ้างถึง path ที่ต้องการ sync ตัวอย่างเช่น

```
DatabaseReference scoresRef = FirebaseDatabase.getInstance().getReference("scores");
scoresRef.keepSynced(true);
```

และโดยทั่วไปจะเก็บข้อมูลลง cache ได้ไม่เกิน 10MB แต่หากแอปพลิเคชันมีข้อมูลที่ต้องการ cache มากกว่านั้น Firebase Realtime Database จะ purge cache ออก แต่ข้อมูลที่เก็บแบบ keepSynced จะไม่ถูก purge นะ

14. การ Query ข้อมูลในขณะที่ Offline

เมื่อแอปพลิเคชัน sync ข้อมูลตอน online มาไว้ในเครื่องแล้ว และเมื่อแอปพลิเคชันเข้าสู่โหมด offline ก็ยังสามารถ query ข้อมูลที่ sync มาแล้วได้ ตัวอย่างเช่น

```
// ตอน online ได้ผลลัพธ์ 4 แถว จากการ query
scoresRef.orderByValue().limitToLast(4)
// เมื่อ offline หากต้องการ query ผลลัพธ์ 2 แถว ก็ทำได้
scoresRef.orderByValue().limitToLast(2)
```

15. การ run ตัว Transactions ขณะ Offline

จริง ๆ แล้วก็เหมือนกับกรณี offline ปกติ คือเมื่อ run ตัว transaction ในขณะที่ offline ข้อมูลต่าง ๆ ก็จะถูกเข้าคิวไว้ แต่หากมีการปิดแล้วเปิดแอปพลิเคชันใหม่คิวจะไม่ถูกเก็บไว้ โดยควรแจ้งผู้ใช้ให้ทราบว่า transaction ไม่สำเร็จในกรณีที่ปิดแล้วเปิดแอปพลิเคชันอีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้