



การออกแบบและพัฒนาระบบการจัดการโรงงานขยะรีไซเคิลด้วย internet of things

นายภูพาน ผลแก้ว

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์
มหาวิทยาลัยนครพนม
พุทธศักราช 2563



Design and Development of Internet of Things Based Systems
for Garbage Recycle Plant Management

Mr. Pupan Poulkaew

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF BACHELOR OF ENGINEERING
IN COMPUTER ENGINEERING
NAKHON PHANOM UNIVERSITY 2020

ภูพาน ผลแก้ว 2563: การออกแบบและพัฒนาระบบการจัดการโรงงานขยะรีไซเคิลด้วย internet of things
 ปริญญาวิศวกรรมศาสตรบัณฑิต (วิศวกรรมคอมพิวเตอร์) สาขาวิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัย
 นครพนม

อาจารย์ที่ปรึกษาหลัก: ผู้ช่วยศาสตราจารย์ ดร. คมกฤษณ์ ชูเรือง

อาจารย์ที่ปรึกษาร่วม: อาจารย์กฤษณ์ วัฒนบุญ

บทคัดย่อ

งานวิจัยนี้นำเสนอ การออกแบบและพัฒนาระบบการจัดการโรงงานขยะรีไซเคิลด้วย internet of things ซึ่งเป็นการนำอุปกรณ์เก่าที่มีในโรงงานมาทำการออกแบบและพัฒนาขึ้นใหม่และควบคุมระบบใหม่ การทำงานประกอบด้วย 4 ส่วนหลักคือส่วนของตราซึ่งมีการปรับปรุงและออกแบบให้มีความเหมาะสมและทันสมัยมากขึ้นโดยจะปรับเปลี่ยนตราซึ่งเดิมให้มีการเชื่อมต่ออินเทอร์เน็ตโดยจะนำค่าน้ำหนักที่ได้มาเก็บไว้ที่เว็บเซิร์ฟเวอร์ หลังจากนั้นก็จะให้ เว็บแอปพลิเคชัน มาทำการดึงข้อมูลขึ้นไปแสดงข้อมูลเพื่อที่จะใช้ในการตรวจจับน้ำหนักและสามารถดูได้ว่าวันนี้มีลูกค้ามาขายของกี่ประเภท ส่วนของตราซึ่งใหญ่ก็จะพัฒนาของเดิมของโรงงานให้สามารถเชื่อมต่ออินเทอร์เน็ตได้โดยจะใช้ RS232 ในการรับข้อมูลและในส่วนของ Finger Print ออกแบบมาเพื่อที่จะเป็นตัวเรียกข้อมูลผู้ขายและใช้เป็นฟังก์ชันสมัครสมาชิกโดยจะพัฒนาจากของเดิมที่จะต้องกรอกชื่อทะเบียนรถเลขบัตรประจำตัวประชาชนดังนั้นฟังก์ชัน Finger Print จะใช้แค่ลายนิ้วมือก็จะสามารถดึงข้อมูลมาโชว์บน เว็บแอปพลิเคชัน ได้และส่วนสุดท้ายคือส่วนบันทึกภาพผู้มาติดต่อขายขยะรีไซเคิลจากที่เมื่อก่อนต้องมีพนักงานอีกคนคอยที่จะกดถ่ายรูปหลังจากติดต่อขายขยะเสร็จดังนั้นจึงได้ออกแบบให้ฟังก์ชันถ่ายภาพได้ทำการบันทึกภาพหลังจากที่สแกนลายนิ้วมือเสร็จทันทีโดยใช้ Raspberry pi และ USB-cam ในส่วนของการทำงานนั้นจะใช้ Python ในการเขียนเพื่อที่จะใช้สั่งให้อุปกรณ์รอคำสั่งการทำงานจากเว็บแอปพลิเคชัน โดย เว็บแอปพลิเคชัน นั้นได้ถูกออกแบบและพัฒนาให้สามารถทำงานร่วมกับอุปกรณ์ที่พัฒนาขึ้นทั้งหมด

Pupan Poulkaew 2020: Design and Development of Internet of Things Based Systems for Garbage Recycle Plant Management,
Bachelor of Engineering (Computer Engineering), Department of Computer Engineering Nakhon Phanom University,
Advisors: Assistant Professor Dr. Komkrit Chooruang
Advisors: Kritchanat Ruamboon

ABSTRACT

This research proposed the design and development of recycling waste management system using Internet of things (IoT). The existing instruments and devices had been modified and upgraded with new controller. The developed system comprises of 4 main parts: weight scales in which it had been modified to be able to connect to the Internet and can upload weight scales data onto webserver. The web application can display data of product's weight and customer transactions in real-time. The vehicle weight scales were also upgraded for Internet connectivity via RS232. Customer registration and inquire processes, such as new member enrolls, inquire data, these processes was implemented using fingerprint reader and then display data on web application. The last part is an automatic image capture, this part uses the Raspberry pi, USB Camera and Python script to automatically capture customer images. The web application is the main center for communicating and controlling of the developed devices and used for interaction with users.

กิตติกรรมประกาศ

งานวิจัยฉบับนี้สำเร็จลุล่วงได้ด้วยดีเนื่องจากได้รับความกรุณาอย่างสูงจากอาจารย์ ผศ.ดร.คมกิต ชูเรือง อาจารย์กฤษณัท ร่วมบุญ และอาจารย์สาขาวิชาวิศวกรรมคอมพิวเตอร์ ที่กรุณาให้คำแนะนำปรึกษาตลอดจนปรับปรุงแก้ไขข้อบกพร่องต่าง ๆ ด้วยความเอาใจใส่เป็นอย่างดี ผู้วิจัยตระหนักถึงความจริงใจ ความทุ่มเทของอาจารย์และขอกราบขอบพระคุณเป็นอย่างสูง พร้อมกับขอขอบพระคุณ บริษัท จีจิบเซียง รีไซเคิลจำกัด จังหวัดอุดรธานี ที่ได้ให้โอกาสได้ออกแบบและพัฒนาโรงงานขยะรีไซเคิล อนึ่งผู้วิจัยหวังว่างานวิจัยฉบับนี้จะมีประโยชน์อยู่ไม่น้อยจึงขอมอบส่วนที่ดีทั้งหมดนี้ให้แก่เหล่าคณาจารย์ที่ได้ประสพวิชาจนทำให้ผลงานวิจัยเป็นประโยชน์ต่อผู้ที่เกี่ยวข้องและขอมอบความกตัญญูกตเวทิตาคุณแด่บิดา มารดา และผู้มีพระคุณทุกท่าน สำหรับข้อบกพร่องต่าง ๆ ที่อาจจะเกิดขึ้นนั้น ผู้วิจัยขอน้อมรับไว้

ผู้วิจัย

ภูพาน ผลแก้ว

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญต่อ	จ
สารบัญตาราง	ฉ
สารบัญรูปภาพ	ช
สารบัญรูปภาพต่อ	ซ
ใบรับรองวิทยานิพนธ์	ณ

บทที่

1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	2
1.4 สถานที่ทำโครงการ	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
1.6 แผนการดำเนินงาน	2
2 ทฤษฎีและผลงานวิจัยที่เกี่ยวข้อง	3
2.1 Embedded system	3
2.2 RS232	4
2.3 การตรวจสอบลายนิ้วมือ	5
2.4 Python	8
2.5 OpenCV (Open source Computer Vision)	8
2.6 Numpy	9
2.7 JavaScript	9
2.8 PHP	9
2.9 งานวิจัยที่เกี่ยวข้อง	10

สารบัญ (ต่อ)

บทที่	หน้า
3 วิธีดำเนินงานโครงการ	11
3.1 ภาพรวมของระบบการจัดการโรงขยะรีไซเคิล	11
3.2 ออกแบบลายวงจรอุปกรณ์สแกนลายนิ้วมือแบบออนไลน์	12
3.3 ขั้นตอนการออกแบบและการเชื่อมต่ออุปกรณ์สแกนลายนิ้วมือแบบออนไลน์	12
3.4 ขั้นตอนการบันทึกโปรแกรมเพื่อใช้แสดงเว็บไซต์สำหรับอุปกรณ์สแกนลายนิ้วมือ	17
3.5 ขั้นตอนการติดต่อสื่อสารเครื่องชั่งน้ำหนักแบบออนไลน์	18
3.6 ขั้นตอนการออกแบบและการเชื่อมต่อของเครื่องบันทึกภาพแบบออนไลน์	20
3.7 ขั้นตอนการประกอบเครื่องบันทึกภาพแบบออนไลน์และการเชื่อมต่อ	20
3.8 ขั้นตอนการออกแบบแอปพลิเคชันบนเว็บและระบบฐานข้อมูล	25
3.9 ขั้นตอนขั้นตอนการออกแบบแอปพลิเคชันบนเว็บและระบบ	25
4 ออกแบบการทดสอบ	32
4.1 ออกแบบการทดลอง	32
4.2 ผลทดสอบความพึงพอใจในการใช้งานเว็บไซต์	32
4.3 อุปกรณ์ที่ทำการทดสอบ	32
4.4 การหาค่าความแม่นยำของอุปกรณ์สแกนลายนิ้วมือ และความเร็วในการรับส่งข้อมูล	34
4.5 การหาค่าความแม่นยำของเครื่องชั่งน้ำหนัก และความเร็วในการรับส่งข้อมูล	34
4.6 การทดลองการทำงานของเว็บแอปพลิเคชัน	38
5 สรุปผลและข้อเสนอแนะ	43
5.1 สรุปผล	43
5.2 ข้อเสนอแนะ	43
5.3 อุปสรรคการทำงาน	43
ภาคผนวก	44
บรรณานุกรม	99
ประวัติผู้เขียน	100

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 1.1 แผนดำเนินงาน	2
ตารางที่ 4.1 ผลการทดลองค่าความแม่นยำของอุปกรณ์สแกนลายนิ้วมือที่ 1	35
ตารางที่ 4.2 ผลการทดลองค่าความแม่นยำของอุปกรณ์สแกนลายนิ้วมือที่ 2	35
ตารางที่ 4.3 ผลการทดลองค่าความแม่นยำและการส่งข้อมูลของอุปกรณ์สแกนลายนิ้วมือที่ 3	36
ตารางที่ 4.4 ผลการทดลองค่าความแม่นยำของอุปกรณ์สแกนลายนิ้วมือที่ 4	36
ตารางที่ 4.5 ผลการทดลองค่าความแม่นยำและการส่งข้อมูลของอุปกรณ์สแกนลายนิ้วมือที่ 5	37
ตารางที่ 4.6 ผลการทดลองค่าความแม่นยำ 1kg	38
ตารางที่ 4.7 ผลการทดลองค่าความแม่นยำ 10kg	39
ตารางที่ 4.8 ผลการทดลองค่าความแม่นยำของอุปกรณ์สแกนลายนิ้วมือที่ 20kg	39
ตารางที่ 4.9 ผลการทดลองบันทึกภาพแบบออนไลน์ร่วมกับเว็บแอปพลิเคชันที่สร้างขึ้น	40
ตารางที่ 4.10 ผลการทดลองเครื่องสแกนลายนิ้วมือทำงานร่วมกับเว็บแอปพลิเคชันที่สร้างขึ้น	41
ตารางที่ 4.11 ผลการทดลองเครื่องสแกนลายนิ้วมือทำงานร่วมกับเว็บแอปพลิเคชันที่สร้างขึ้น	42

สารบัญรูปภาพ

รูปที่	หน้า
รูปที่ 2.1 บล็อกไดอะแกรมส่วนประกอบของระบบสมองกลฝังตัว	3
รูปที่ 2.2 การทำงานระหว่าง RS-232 กับ Node MCU	4
รูปที่ 2.3 ลักษณะเส้นขนานและเส้นร่องของนิ้วมือ	5
รูปที่ 2.4 ลักษณะลายนิ้วมือแบบรูปก้นหอย	6
รูปที่ 2.5 มัดหวายปิดซ้ายลักษณะลายนิ้วมื้อมัดหวายปิดขวา	6
รูปที่ 2.6 ลักษณะลายนิ้วมือเส้นโค้งราบ	6
รูปที่ 2.7 ลักษณะลายนิ้วมือเส้นโค้งกระโจม	7
รูปที่ 2.8 เส้นโค้งกระโจม	7
รูปที่ 3.1 รูปที่ 3.1 ภาพรวมที่ทำการออกแบบทั้งหมด	11
รูปที่ 3.2 ลายวงจรที่ใช้สำหรับการออกแบบเครื่องสแกนลายนิ้วมือ	12
รูปที่ 3.3 ติดตั้ง Platformio จาก Visual Studio Code	13
รูปที่ 3.4 ติดตั้ง ไลบรารีลงในโฟลเดอร์	14
รูปที่ 3.5 อัปโหลดโปรแกรมเพื่อใช้งาน	14
รูปที่ 3.6 อุปกรณ์พร้อมสำหรับการใช้งาน	15
รูปที่ 3.7 การทำงานของ Fingerprint	16
รูปที่ 3.8 ออกแบบลายวงจรที่ใช้สำหรับอุปกรณ์รับค่าจากซิงก์น้ำหนักระบบออนไลน์	17
รูปที่ 3.9 อัปโหลดโปรแกรมเพื่อใช้งาน	18
รูปที่ 3.10 อุปกรณ์ที่ออกแบบกับเครื่องชั่ง	19
รูปที่ 3.11 การทำงานของเครื่องชั่ง	19
รูปที่ 3.12 อุปกรณ์ที่ใช้สำหรับบันทึกภาพแบบออนไลน์	20
รูปที่ 3.13 ติดตั้ง ส่วนขยาย OpenCV ที่จำเป็นต่อการใช้งาน	21
รูปที่ 3.14 ติดตั้ง ส่วนขยาย NumPy ที่จำเป็นต่อการใช้งาน	21
รูปที่ 3.15 ขั้นตอนการใช้งาน	22
รูปที่ 3.16 ขั้นตอนการใช้งานอุปกรณ์บันทึกภาพแบบออนไลน์	23

รูปที่	สารบัญรูปภาพ (ต่อ)	หน้า
รูปที่ 3.17	Flow Chart การทำงาน	24
รูปที่ 3.18	โปรแกรมที่ใช้ในการออกแบบแอปพลิเคชันบนเว็บและระบบฐานข้อมูล	25
รูปที่ 3.19	ภาพรวมของเว็บแอปพลิเคชัน	26
รูปที่ 3.20	การทำ Normalization	27
รูปที่ 3.21	ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้เก็บข้อมูลการลงชื่อเข้าใช้	27
รูปที่ 3.22	ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้บ่งบอกระดับการใช้งาน	28
รูปที่ 3.23	ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้เก็บข้อมูลสมาชิกที่ทำการลงทะเบียน	28
รูปที่ 3.24	ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้เก็บข้อมูลสินค้าและราคาสินค้า	29
รูปที่ 3.25	ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้เก็บข้อมูลการรับซื้อขยะรีไซเคิล	29
รูปที่ 3.26	ออกแบบหน้าเว็บให้รองรับระบบสื่อคอิน	30
รูปที่ 3.27	แสดงค่าน้ำหนักลงในแอปพลิเคชันบนเว็บที่พัฒนาขึ้น	30
รูปที่ 3.28	ทดสอบสั่งการทำงานอุปกรณ์สแกนลายนิ้วมือแบบออนไลน์	31
รูปที่ 3.29	ทดสอบสั่งการทำงานอุปกรณ์บันทึกภาพแบบออนไลน์จากแอปพลิเคชันบนเว็บ	31
รูปที่ 4.1	ความพึงพอใจในการใช้งานเว็บไซต์ด้านการออกแบบและจัดการรูปแบบเว็บไซต์	31
รูปที่ 4.2	ความพึงพอใจในการใช้งานเว็บไซต์ด้านการติดต่อกับอุปกรณ์	33
รูปที่ 4.3	ความพึงพอใจในการใช้งานเว็บไซต์ด้านเนื้อหา	33
รูปที่ 4.4	สรุปแบบสอบถามความพึงพอใจ	34



ใบรับรองวิทยานิพนธ์
มหาวิทยาลัยนครพนม
หลักสูตร
วิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์

ชื่อวิทยานิพนธ์: การออกแบบและพัฒนาระบบการจัดการโรงงานขยะรีไซเคิลด้วย internet of things

ชื่อผู้ทำวิทยานิพนธ์: นายภูพาน ผลแก้ว

คณะกรรมการสอบวิทยานิพนธ์	อาจารย์ชาญวิทย์ สุวรรณพงศ์	ผู้รับผิดชอบวิชา
	อาจารย์อภิวดี แก้วส่อง	กรรมการ
	อาจารย์อภิวัตร บุญกอง	กรรมการ
	อาจารย์ทรงฤทธิ์ กิตติศรีวรพันธุ์	กรรมการ

อาจารย์ที่ปรึกษาวิทยานิพนธ์:

.....
(ผู้ช่วยศาสตราจารย์ ดร. คมกฤษณ์ ชูเรือง)
อาจารย์ที่ปรึกษา

.....
(อาจารย์กฤษณ์ วัฒนบุญ)
อาจารย์ที่ปรึกษาร่วม

.....
(อาจารย์อภิวัตร บุญกอง)
หัวหน้าสาขา

ลิขสิทธิ์ของมหาวิทยาลัยนครพนม

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบัน บริษัท จีจิบเชียง รีไซเคิล 2008 จำกัด คือผู้ดำเนินการธุรกิจรีไซเคิล โดยมีกิจกรรมของบริษัทประกอบไปด้วย กระบวนการรับซื้อขยะ โดยการจัดทำประวัติและจัดลำดับคิวของผู้ขายแต่ละราย รวมไปถึงขั้นตอนการออกบิลเพื่อแจ้งข้อมูลและค่าใช้จ่ายที่ผู้ขายแต่ละรายจะได้รับ จากนั้นจึงดำเนินการชั่งน้ำหนักโดยใช้ตราชั่งใหญ่ที่ได้รับมาตรฐานจากสำนักงานกลางชั่งตวงวัด กรมการค้าภายใน เก็บขยะเข้าคลังเพื่อรอการคัดแยก กระบวนการคัดแยกขยะแต่ละประเภท อาทิเช่น ขยะประเภทพลาสติก แก้ว กระดาษ เพื่อเตรียมเข้าสู่กระบวนการถัดไป เป็นต้น กระบวนการทำความสะอาดและคัดแยกสิ่งสกปรก รวมถึงขั้นตอนที่ไม่ต้องการออกจากขยะแต่ละประเภท กระบวนการย่อยขนาด หรือบดอัด เพื่อเตรียมจัดส่งให้กับลูกค้า

บริษัทมีปริมาณการรับซื้อเศษขยะหรือวัสดุต่าง ๆ เฉลี่ยอยู่ที่ประมาณ 20 ตันต่อวัน ซึ่งเป็นการรับซื้อจากผู้ประกอบการรายย่อย ทั้งในรูปแบบบริษัทและแบบปัจเจก นอกจากนี้ทางบริษัทยังมีสำนักงานสาขาซึ่งทำหน้าที่เป็นจุดรับซื้ออีกจำนวน 3 สาขา และมีพนักงานรวมกันมากกว่า 30 คน โดยทางบริษัทแบ่งส่วนงานหลักๆ ออกเป็น 2 ส่วน ประกอบด้วยฝ่ายธุรการและฝ่ายการผลิต แต่การบริหารงานหรือกิจกรรมที่เกิดขึ้นทั้งหมดจะเป็นไปในลักษณะของผู้บริหารคนเดียวเป็นผู้มีอำนาจในการดูแลและตัดสินใจ ส่งผลให้การบริหารและจัดการกิจกรรมต่าง ๆ ที่เกิดจากทั้ง 3 สาขา มีความยากลำบากและขาดการดูแลอย่างทั่วถึง นอกจากนี้เครื่องชั่งน้ำหนักที่ในจุดรับซื้อของแต่ละสาขายังเป็นลักษณะแบบ offline

ซึ่งหากผู้บริหารต้องการทราบปริมาณขยะรีไซเคิลที่แต่ละสาขารับซื้อในแต่ละวัน ทางผู้บริหารจำเป็นต้องนำข้อมูลการซื้อขายในรอบสัปดาห์จากทั้ง 3 สาขา มาทำการรวมด้วยโปรแกรมไมโครซอฟท์ Excel ด้วยตนเอง เพื่อจัดทำรายงานสรุปต่าง ๆ ส่งผลให้ผู้บริหารไม่ทราบสถานะในการซื้อขาย และข้อมูลต่าง ๆ ของแต่ละสาขาได้อย่างทันถ่วงที

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อออกแบบและพัฒนาอุปกรณ์ที่ใช้วัดน้ำหนักของขยะรีไซเคิลผ่านเครือข่ายอินเทอร์เน็ต
- 1.2.2 เพื่อออกแบบและพัฒนาระบบพิสูจน์ตัวตนลูกค้าผ่านระบบสแกนลายนิ้วมือ
- 1.2.3 เพื่อออกแบบและพัฒนาเว็บแอปพลิเคชันเบื้องต้นเพื่อใช้งานร่วมกับอุปกรณ์ที่สร้างขึ้น
- 1.2.4 เพื่อออกแบบและพัฒนาระบบบันทึกภาพแบบออนไลน์เพื่อใช้ร่วมกับโปรแกรมที่พัฒนาขึ้น

1.3. ขอบเขตของโครงการ

- 1.3.1 ได้ตราชั่งที่สามารถเชื่อมต่อกับอินเทอร์เน็ตได้
- 1.3.2 ได้อุปกรณ์ที่ใช้สื่อสารระหว่างตราชั่งใหญ่ ผ่านเครือข่ายอินเทอร์เน็ตจำนวน 1 อุปกรณ์
- 1.3.3 ได้อุปกรณ์สแกนลายนิ้วมือแบบออนไลน์ที่ทำงานร่วมกับโปรแกรม

1.4. สถานที่ทำโครงการ

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนครพนม

1.5. ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 ได้ตราซังที่สามารถทำงานแบบออนไลน์ในราคาที่ถูกลง

1.5.2 เข้าใจหลักการทำงานของเว็บแอปพลิเคชัน

1.5.3 ได้ระบบพิสูจน์ตัวตนที่ทันสมัยขึ้น

1.6. แผนการดำเนินงาน

ลำดับ	กิจกรรม	ระยะเวลา (เดือน)
1	เก็บรวบรวมข้อมูล	สิงหาคม
2	ออกแบบอุปกรณ์ที่ใช้สื่อสารระหว่างตราซังใหญ่กับโปรแกรมที่สร้างขึ้น	กันยายน
3	ออกแบบและพัฒนาตราซังเล็กที่สื่อสารระหว่างตราซังกับโปรแกรมผ่านเครือข่ายอินเทอร์เน็ต	กันยายน
4	ออกแบบและสร้างระบบสแกนลายนิ้วมือแบบออนไลน์	กันยายน
5	ออกแบบและพัฒนาระบบอุปกรณ์บันทึกภาพแบบออนไลน์ร่วมกับโปรแกรมที่พัฒนาขึ้นบนเว็บ	ตุลาคม
6	ออกแบบระบบฐานข้อมูล	ธันวาคม
7	ออกแบบและพัฒนาระบบเว็บแอปพลิเคชันสำหรับติดต่อกับอุปกรณ์ต่าง ๆ ที่สร้างขึ้น	ธันวาคม
8	ทำการทดลองอุปกรณ์ต่าง ๆ ที่พัฒนาขึ้น	มกราคม
9	สรุปผล	กุมภาพันธ์

ตารางที่ 1.1 แผนการดำเนินงาน

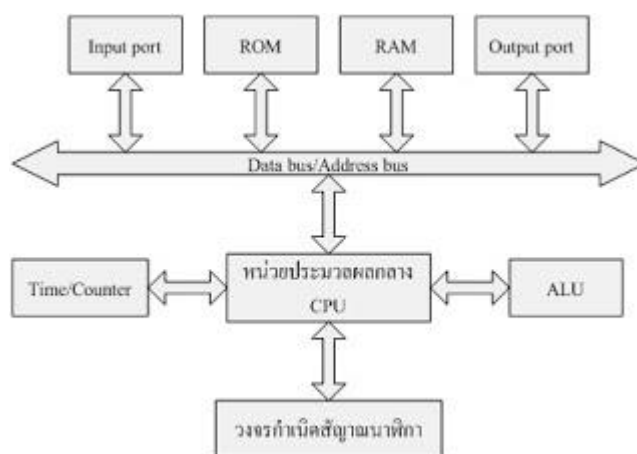
บทที่ 2

ทฤษฎีและผลงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎีและหลักการที่เกี่ยวข้องกับโครงงานวิจัยเรื่องการออกแบบและพัฒนาระบบการจัดโรงงานขยะรีไซเคิลด้วย internet of things ผู้จัดทำโครงงานได้ศึกษาสืบค้นข้อมูลและผลงานวิจัยที่เกี่ยวข้อง มีเนื้อหาตามลำดับดังต่อไปนี้

2.1 Embedded system

Embedded System คือระบบประมวลผลขนาดเล็ก ซึ่งเปรียบเทียบกับคอมพิวเตอร์ส่วนบุคคลขนาดเล็กที่ถูกย่อขนาดมาเหลือเพียงแผงวงจรอิเล็กทรอนิกส์แนวคิดและหลักการเกี่ยวกับระบบฝังตัวหรือสมองกลฝังตัว (Embedded system) นั้นเป็นที่นิยมมากขึ้นในปัจจุบัน มีการใช้งานกันอย่างแพร่หลายเช่น ในงานอิเล็กทรอนิกส์กำลัง ระบบการวัดและควบคุม ระบบการแปลงผันพลังงาน และในระบบการขับเคลื่อนไฟฟ้า เป็นต้น เนื่องจากมีราคาถูก การใช้งานง่ายสามารถแยกทำงานได้อย่างอิสระไม่จำเป็นต้องเชื่อมต่อกับคอมพิวเตอร์ ระบบฝังตัวหรือสมองกลฝังตัวนั้นเป็นระบบประมวลผลขนาดเล็กที่ฝังไว้ในอุปกรณ์เพื่อเพิ่มความฉลาดและความสามารถให้กับอุปกรณ์เหล่านั้นผ่านซอฟต์แวร์ ซึ่งจะแตกต่างจากระบบประมวลผลที่ใช้เครื่องคอมพิวเตอร์ทั่วไป ภายในระบบสมองกลฝังตัวจะมีไมโครคอนโทรลเลอร์หรือไมโครโพรเซสเซอร์เป็นตัวประมวลผลและสั่งงาน หลักการทำงานและส่วนประกอบพื้นฐานของระบบสมองกลฝังตัวนั้นจะเหมือนกับคอมพิวเตอร์ ซึ่งมีส่วนประกอบดังแสดงเป็นบล็อกไดอะแกรมในรูปที่ 2.1



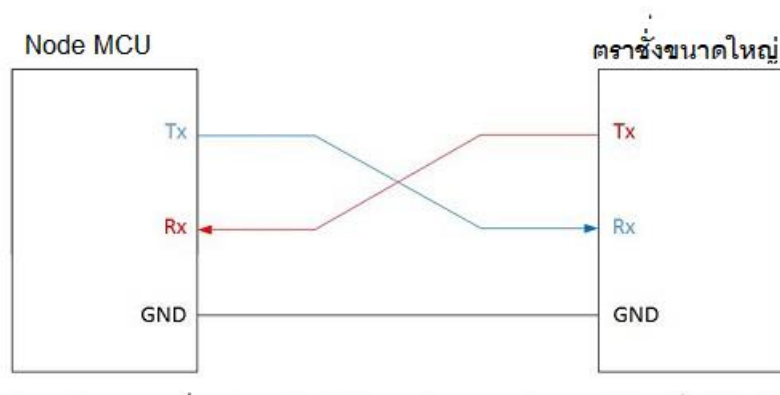
รูปที่ 2.1 บล็อกไดอะแกรมส่วนประกอบของระบบสมองกลฝังตัว

จากรูประบบสมองกลนั้นจะประกอบด้วย หน่วยประมวลผลกลาง (CPU), หน่วยประมวลผลทางวงจรกำเนิดสัญญาณนาฬิกา (Oscillator), อินพุต (Input), เอาต์พุต (Output), หน่วยความจำโปรแกรม (ROM), หน่วยความจำข้อมูล (RAM) และบัส (Bus) ทำหน้าที่เหมือนกับคอมพิวเตอร์ขนาดเล็ก ระบบสมองกลฝังตัวนั้นมีประโยชน์มากช่วยให้วิศวกรหรือผู้ออกแบบสามารถสร้างวงจรการทำงานได้เล็กลง และมีความซับซ้อนน้อยลงเป็นอย่างมาก อีกทั้งยังมีการพัฒนาอย่างต่อเนื่องไม่ว่าจะเป็นในส่วนของฮาร์ดแวร์และซอฟต์แวร์ ทำให้การใช้งานและการพัฒนาโปรแกรมทำได้ง่าย นอกจากนี้จะมีส่วนประกอบดังรูปที่ 1.1 แล้วในปัจจุบันยังมีการเพิ่มส่วนการทำงานด้านการประมวลผลสัญญาณดิจิทัล (DSP) เพื่อใช้ในการคำนวณทางคณิตศาสตร์ การแปลงสัญญาณแอนะล็อกเป็นดิจิทัล (ADC) การสร้างสัญญาณพัลส์วิดท์มอดูเลชัน (PWM) จากคุณสมบัติที่เพิ่มเข้ามาทั้งหมดนั้น ทำให้ระบบสมองกลฝังตัวสามารถตอบสนองการใช้งานได้อย่างกว้างขวาง

วัตถุประสงค์เพื่อสามารถสั่งการทำงานอุปกรณ์ที่ต้องการให้สามารถทำงานได้เองโดยอัตโนมัติ ดังนั้นโครงการวิจัยนี้จึงได้นำระบบฝังตัวมาพัฒนาใช้ในระบบรับส่งข้อมูลต่าง ๆ ในเครื่องชั่งและเครื่องตรวจสอบลายนิ้วมือ

2.2. RS232

คือมาตรฐานการสื่อสารข้อมูลดิจิทัลแบบอนุกรมโดยถูกกำหนดมาจาก EIA (Electronic Industries Association) หรือ สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ของอเมริกา ซึ่งในยุคแรก RS232 เป็นที่นิยมมาก ขนาดที่คอมพิวเตอร์ทุกเครื่องจะต้องมี Serial port สำหรับการสื่อสารมาตรฐานนี้และเชื่อว่าคอมพิวเตอร์ของผู้ใช้หลายๆท่านก็ยังมี Port เชื่อมต่อนี้อยู่ แต่ในปัจจุบันได้มี USB ซึ่งเป็นมาตรฐานสื่อสารที่รับ/ส่งข้อมูลได้เร็วกว่าเข้ามาแทนที่ ทำให้มาตรฐานการสื่อสารอย่าง RS232 ก็ค่อยๆมีอุปกรณ์ที่รองรับน้อยลงเรื่อย ๆ ตามการเวลาโดยกระบวนการจะเป็นไปตามดังรูปที่ 2.2



รูปที่ 2.2 การทำงานระหว่าง RS-232 กับ Node MCU

จากรูปที่ 2.2 เป็นตัวอย่างการเชื่อมต่อแบบ RS232 ของตราซังขนาดใหญ่กับ Node MCU เพื่อรับค่าน้ำหนักของตราซังขนาดใหญ่ผ่าน Software

Tx (เครื่องวัด) จะถูกต่อเข้ากับ Rx (NODE MCU) เพื่อส่งข้อมูลจากเครื่องวัดไปยังตัวรับของคอมพิวเตอร์

Rx (เครื่องวัด) จะถูกต่อเข้ากับ Tx (NODE MCU) เพื่อรับข้อมูลที่ส่งมาจากคอมพิวเตอร์

GND (เครื่องวัด) จะถูกต่อเข้ากับ GND (NODE MCU) เพื่อเทียบสัญญาณแรงดัน 0V

2.3 การตรวจสอบลายนิ้วมือ

ในปัจจุบันการตรวจสอบลายนิ้วมือเพื่อระบุตัวบุคคลนั้นกำลังได้รับความนิยมอย่างแพร่หลาย เนื่องจากมีความแม่นยำและมีความรวดเร็วในการระบุตัวบุคคล อีกทั้งลักษณะลายนิ้วมือนั้นยังมีความซับซ้อนยากต่อการปลอมแปลงและซ้ำซ้อนกับบุคคลอื่น โดยได้มีการศึกษาพบว่าโอกาสที่คนสองคนจะมีลายนิ้วมือเหมือนกันนั้นมีความน่าจะเป็น $1/64,000,000,000$ (เสาวณี ธาตุอินจันทร์, 2553) ซึ่งเป็นการประเมินค่าโดยใช้การแบ่งรายละเอียดรูปแบบของลายนิ้วมือออกเป็นส่วนๆ และหาความน่าจะเป็นของการซ้ำ กันของแต่ละส่วนนั้น นอกจากนี้แล้วแม้แต่คนภายในครอบครัวเดียวกันหรือคู่แฝดแท้ก็ยังมีลักษณะลายนิ้วมือ ที่แตกต่างกัน ลักษณะของลายนิ้วมือจะประกอบด้วยลายเส้นนูน (Ridge) ที่เกิดจากรอยนูนที่อยู่สูงขึ้นมาจากผิวส่วน นอกและอีกชนิดหนึ่งเรียกว่ารอยร่องหรือเส้นร่อง (Valley) ที่มีลักษณะของรอยลึกอยู่ต่ำกว่าระดับของเส้นนูน (เอกรินทร์ ชื้อธนูวงศ์, 2548) ดังรูปที่ 2.2



รูปที่ 2.3 ลักษณะเส้นนูนและเส้นร่องของนิ้วมือ

ลักษณะลายนิ้วมือสามารถแบ่งเป็นลักษณะรูปแบบได้ 3 รูปแบบ (สมปอง ศรีลักษณ์, 2549) คือแบบลายก้นหอย (Whorl), แบบลายมัดหวาย (Loop) และแบบลายโค้ง (Arch) โดยแต่ละแบบมีรายละเอียดดังนี้

2.3.1 แบบลายก้นหอย (Whorl) เป็นลายลายนิ้วมือที่มีลักษณะเป็นรูปเส้นเวียนรอบตรง กลางหรือมีลักษณะคล้ายลายก้นหอยดังรูปที่ 2.3 ซึ่งบุคคลที่มีลายนิ้วมือนี้นี้อยู่ประมาณ 30-35 เปอร์เซ็นต์ โดยสามารถแบ่งออกได้เป็น 2 ประเภทคือ

2.3.1.1 ก้นหอยธรรมดา (Plain Whorl) เป็นลายนิ้วมือที่มีลักษณะลายเส้นเวียน เป็นรูปไข่หรือวงกลม ก้นหอยจะมีสันตอนสองสันตอนถ้าลากเส้นจากสันตอนหนึ่งไปยังอีกสันตอนหนึ่งเส้นที่ลากจะตัดหรือแตะวงกลมในก้นหอยอย่างน้อย 1 วง

2.3.1.2 ก้นหอยกระเป๋ากลาง (Central Pocket) มีลักษณะคล้ายก้นหอยธรรมดา แต่ลากเส้นจากสันตอนหนึ่งไปยังอีกสันตอนหนึ่ง เส้นจะไม่ตัดกับวงกลมในก้นหอย



รูปที่ 2.4 ลักษณะลายนิ้วมือแบบรูปก้นหอย

2.3.2 แบบลายมัดหวาย (Loop) เป็นลายนิ้วมือที่พบได้เป็นส่วนใหญ่ ซึ่งบุคคลที่มีลายนิ้วมือ ชนิดนี้มีอยู่ประมาณ 60-65 เปอร์เซ็นต์ โดยลักษณะลายนิ้วมือแบบลายมัดหวายสามารถแบ่งออกได้เป็น 2 ประเภทคือ

2.3.2.1 มัดหวายปิดซ้าย (Left Loop) เป็นเส้นลายนิ้วมือที่ลายเส้นวิ่งจากด้าน ซ้ายมือไปยังกึ่งกลางและวกกลับมายังด้านเดิมอย่างน้อย 1 เส้นมีสันตอนเพียงจุดเดียวดังรูปที่ 2.4

2.3.2.2 มัดหวายปิดขวา (Right Loop) เป็นเส้นลายนิ้วมือที่ลายเส้นวิ่งจากด้าน ขวามือไปยังกึ่งกลางและวกกลับมายังด้านเดิมอย่างน้อย 1 เส้นดังรูปที่ 2.5



รูปที่ 2.5 มัดหวายปิดซ้าย



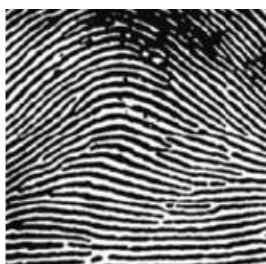
รูปที่ 2.6 มัดหวายปิดขวา

2.3.3 แบบลายโค้ง (Arch) เป็นลายนิ้วมือที่มีลักษณะพิเศษและมีอยู่น้อยมากจากลายนิ้วมือ ที่มีอยู่ ซึ่งลายนิ้วมือชนิดนี้มีประมาณ 5 เปอร์เซ็นต์ โดยลักษณะลายนิ้วมือแบบลายโค้งสามารถแบ่งออกได้ เป็น 2 ประเภทคือ

2.3.3.1 เส้นโค้งราบ (Plain Arch) เป็นลักษณะลายนิ้วมือที่วิ่งจากขอบด้านหนึ่งไป ยังขอบอีกด้านหนึ่งแล้ววิ่งหรือไหลออกไปอีกด้านหนึ่ง ลายนิ้วมือแบบโค้งจัดเป็นลักษณะลายนิ้วมือที่มีลายเส้น ชนิดที่

ดูได้ง่ายที่สุดกว่าลายนิ้วมือทุกชนิด ไม่มีเส้นเกือกม้าและไม่เกิดมุมแหลมคมที่เห็นได้ชัดตรงกลางหรือไม่มี เส้นพุ่งสูงขึ้นตรงกลาง ดังรูปที่ 2.6

2.3.3.2 เส้นโค้งกระโจม (Tented Arch) เป็นลักษณะลายนิ้วมือชนิดโค้งราบแต่มี ลักษณะแตกต่างกับชนิดโค้งราบคือ มีลายเส้นหนึ่งเส้นหรือมากกว่าที่อยู่ตรงกลางของลายนิ้วมือพุ่งขึ้นจาก แนวนอน เป็นมุมแหลมคมหรือมุมฉาก ดังรูปที่ 2.7



รูปที่ 2.7 เส้นโค้งราบ



รูปที่ 2.8 เส้นโค้งกระโจม

จากรูปร่างลักษณะของลายนิ้วมื่อดังกล่าวนั้นสามารถนำมาวิเคราะห์ลายนิ้วมือเพื่อทำการตรวจสอบลายนิ้วมือของแต่ละบุคคลได้ โดยกระบวนการตรวจสอบลายนิ้วมือจะแบ่งออกเป็น 2 กระบวนการ (สมปอง ศรีลักษณ์, 2549) คือกระบวนการลงทะเบียนลายนิ้วมือ (Enrollment Procedure) และกระบวนการตรวจสอบลายนิ้วมือ (Verification Procedure) ดังรูปที่ 2.7

กระบวนการลงทะเบียน (Enrollment Procedure) ขั้นตอนจะเป็นการรับลายนิ้วมือของ แต่ละบุคคลซึ่งจะถูกเก็บไว้เป็นภาพผ่านตัวตรวจสอบลายนิ้วมือ จากนั้นจะทำการประมวลผลวิเคราะห์และผ่านขั้นตอนการดึงลักษณะเด่น (Feature Extraction) ที่เป็นเอกลักษณ์ของภาพลายนิ้วมือในแต่ละบุคคล มา แปลงเป็นโครงสร้างข้อมูลที่เหมาะสม จากนั้นจะเก็บข้อมูลที่ได้ลงในระบบฐานข้อมูลพร้อมรหัสชื่อของแต่ละ บุคคลที่ลงทะเบียนเพื่อเป็นแม่แบบ (Template) สำหรับนำไปเปรียบเทียบพิสูจน์ตัวตนต่อไป กระบวนการตรวจสอบลายนิ้วมือ (Verification Procedure) เป็นกระบวนการที่ระบบ ตรวจสอบลายนิ้วมือในฐานข้อมูลว่าเป็นลายข้อมูลเดียวกันหรือไม่ โดยจะทำการประมวลผลภาพวิเคราะห์และแยกลักษณะเด่นออกมาแปลงเป็นโครงสร้างข้อมูลที่เหมาะสม แล้วทำการเปรียบเทียบคู่ลักษณะเด่น (Feature Matcher) ที่ได้กับข้อมูลที่มีอยู่ในฐานข้อมูลเพื่อพิสูจน์ตัวบุคคลนั้น ๆ ซึ่งถ้าข้อมูลทั้งคู่ตรงกันจึงจะสามารถเข้าไปยังระบบได้

2.4 Python

Python คือชื่อภาษาที่ใช้ในการเขียนโปรแกรมภาษาหนึ่ง ซึ่งถูกพัฒนาขึ้นมาโดยไม่ยึดติดกับแพลตฟอร์ม กล่าวคือสามารถรันภาษา Python ได้ทั้งบนระบบ Unix, Linux , Windows NT, Windows 2000, Windows XP หรือแม้แต่ระบบ FreeBSD อีกอย่างหนึ่งภาษาตัว นี้เป็น Opensource เหมือนอย่าง PHP ทำให้ทุกคนสามารถที่จะนำ Python มาพัฒนาโปรแกรมของเราได้ฟรีๆโดยไม่ต้องเสียค่าใช้จ่าย และความเป็น Open Source ทำให้มีคนเข้ามาช่วยกันพัฒนาให้ Python มีความสามารถสูงขึ้น และใช้งานได้ครอบคลุมกับทุกลักษณะงานไวยากรณ์ของภาษา Python ภาษา Python นั้นถูกพัฒนาขึ้นมาโดยมีความตั้งใจว่าจะให้เป็นภาษาที่อ่านง่าย มันถูกออกแบบมาให้มีโครงสร้างที่มองเห็นได้โดยไม่ซับซ้อน โดยมักจะใช้คำในภาษาอังกฤษในขณะที่ภาษาอื่นใช้เครื่องหมายวรรคตอน นอกจากนี้ Python มีข้อยกเว้นของโครงสร้างทางภาษาน้อยกว่าภาษา C และ Pascal Python Interpreter Python interpreter นั้นเป็นตัวแปลภาษาของภาษา Python เพื่อให้สามารถรันโค้ด Python ได้ ซึ่งได้มากับไลบรารีมาตรฐานที่สามารถใช้งานได้ฟรี ซึ่งดาวน์โหลดได้ที่ <https://www.python.org/> เป็นโปรแกรมแบบ source และ binary สำหรับแพลตฟอร์มที่ได้รับความนิยม นอกจากนี้ Interpreter ยังสนับสนุนการเขียนโปรแกรมกับ Interactive shell ซึ่งเป็นการเขียนโค้ดของภาษา Python ลงไปและเห็นผลลัพธ์การทำงานของคำสั่งได้ในทันที Python Interpreter นั้นยังสามารถนำเพิ่มความสามารถกับฟังก์ชันใหม่ที่ถูกพัฒนามาจากภาษา C และ C++ Python นั้นเหมาะสำหรับเป็นภาษาในการสร้าง Extension และแอปพลิเคชันที่ปรับแต่งได้

2.5 OpenCV (Open source Computer Vision)

เป็นไลบรารีฟังก์ชันการเขียนโปรแกรม (Library of Programming Functions) โดยส่วนใหญ่จะมุ่งเป้าไปที่การแสดงผลด้วยคอมพิวเตอร์แบบเรียลไทม์ (Real-Time Computer Vision) เดิมทีแล้วถูกพัฒนาโดย Intel แต่ภายหลังได้รับการสนับสนุนโดย Willow Garage ตามมาด้วย Itseez (ซึ่งต่อมาถูกเข้าซื้อโดย Intel) OpenCV เป็นไลบรารีแบบข้ามแพลตฟอร์ม (Cross-Platform) และใช้งานได้ฟรีภายใต้ลิขสิทธิ์ของ BSD แบบโอเพ่นซอร์ส (Open-Source BSD License)

2.6 Numpy

เป็นส่วนขยายของภาษาไพทอนเพื่อจัดการเมทริกซ์หรืออาร์เรย์หลายมิติรวมถึงฟังก์ชันทางคณิตศาสตร์ที่ทำงานบนอาร์เรย์เหล่านั้นนอกจากนั้นคลังซอฟต์แวร์โอเพนซอร์สและเสรีนี้มีฟังก์ชันหลายอย่างที่ทำให้ทำได้โดยเฉพาะอย่างยิ่งในการสร้างอาร์เรย์โดยตรงจากไฟล์หรือเพื่อบันทึกอาร์เรย์ในไฟล์เพื่อจัดการเวกเตอร์เมทริกซ์และพหุนาม Numpy เป็นฐานเพิ่มเติมของ SciPy ซึ่งเป็นคอลเลกชันของห้องสมุด Python เกี่ยวกับการคำนวณทางวิทยาศาสตร์

2.7 JavaScript

JavaScript ถูกพัฒนาขึ้นโดย เน็ตสเคปคอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่า Live Script ออกมาพร้อมกับ Netscape Navigator 2.0 เพื่อใช้สร้างเว็บเพจ โดยติดต่อกับเซิร์ฟเวอร์แบบ Live Wire ต่อมาเน็ตสเคปจึงได้ร่วมมือกับ บริษัทซันไมโครซิสเต็มส์ปรับปรุงระบบของบราวเซอร์เพื่อให้สามารถติดต่อกับภาษาจาวาได้ และได้ปรับปรุง LiveScript ใหม่เมื่อปี 2538 แล้วตั้งชื่อใหม่ว่า JavaScript JavaScript สามารถทำให้ การสร้างเว็บเพจ มีลูกเล่น ต่าง ๆ และยัง สามารถโต้ตอบกับผู้ใช้ได้อย่างทันที เช่น การใช้เมาส์คลิก หรือ การกรอกข้อความในฟอร์ม JavaScript คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต ที่กำลังได้รับความนิยมอย่างสูง Java JavaScript เป็น ภาษาสคริปต์เชิงวัตถุ (ที่เรียกกันว่า "สคริปต์" (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปทีละคำสั่ง" (interpret) หรือเรียกว่า อ็อบเจกต์โอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนา โปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์

2.8 PHP

PHP ย่อมาจากคำว่า “Personal Home Page Tool” (ปัจจุบันได้เพิ่มเติมคำย่อใหม่โดยรวมกับตัวย่อเป็น PHP : PHP Hypertext Preprocessor) ซึ่งเป็นภาษาประเภท Script Language ที่ทำงานแบบ Server Side Script กระบวนการทำงานจะทำงานแบบโปรแกรมแปลคำสั่ง interpreter คือแปลภาษา ทุกครั้งที่มีคนเรียกสคริปต์ ข้อดีคือ ไม่ต้องนำไปประมวลผลใหม่ (Compiler) เมื่อนำโปรแกรมไปใช้งาน หรือจะอัปเดตเวอร์ชันของโปรแกรม สามารถอัปเดตขึ้นไปทับไฟล์เดิมแล้วใช้งานได้ทันที ขอเสียที่ต่างกัน อย่างชัดเจนก็คือ กรณี Syntax ผิดจะรู้ก็ต่อเมื่อมีผู้ใช้งานเจออีก

ภาษา PHP จัดอยู่ในประเภท การเขียนโปรแกรมบนเว็บ (Web-based Programming) เพราะเราจะเก็บโค้ดคำสั่ง หรือสคริปต์ทั้งหมดที่เขียนขึ้นมาไว้บนเครื่องเซิร์ฟเวอร์ที่เดียว (Web Server) และให้ผู้ใช้งาน (Client) เรียกใช้งานโปรแกรมผ่านเว็บเบราว์เซอร์ต่างๆ เช่น Internet Explorer, Mozilla Firefox, Google Chrome, Opera, Safari ฯลฯ เพื่อนำข้อมูลมาแสดงผลที่หน้าจอของผู้ใช้แต่ละคนนั่นเอง

2.9 งานวิจัยที่เกี่ยวข้อง

กัลยาณี บรรจงจิตรและคณะ (2551) ได้วิจัย ระบบเช็คชื่อนิสิตด้วยลายนิ้วมือ โดยงานวิจัยนี้ได้พัฒนาขึ้นด้วยซอฟต์แวร์ Microsoft Visual Studio.Net 2005 และ Microsoft SQL Server 2005 เพื่อใช้รวบรวมรายละเอียดเกี่ยวกับการเข้าเรียนและคะแนนสอบของนักศึกษาไว้ในฐานข้อมูล โปรแกรมที่ได้พัฒนาขึ้นมาสามารถบันทึกการเข้าเรียนนักศึกษาด้วยการสแกนลายนิ้วมือซึ่งสามารถตรวจสอบตัวตนของนักศึกษาได้ จากผลการทดสอบการใช้งานพบว่าอาจารย์สามารถประมวลผลการเรียนจากฐานข้อมูลที่สร้างขึ้น และนักศึกษาหรือผู้ปกครองสามารถตรวจสอบการเข้าเรียนของนักศึกษาผ่านทางเว็บไซต์ได้

เสาวณี ธาตุอินจันทร์ (2553) ได้วิจัย ระบบบันทึกการลงเวลาและบริหารงานบุคคลโดยใช้การ ตรวจสอบลายนิ้วมือ โดยระบบที่พัฒนาขึ้นใช้โปรแกรมเดลไฟล์ในการพัฒนาส่วนติดต่อกับผู้ใช้งาน ใช้ระบบ ฐานข้อมูลไฟเบิร์ด (Firebird database) และใช้โปรแกรมไมโครซอฟต์แอคเซส (Microsoft Access) ในการ จัดเก็บลายนิ้วมือ อีกทั้งได้พัฒนาให้เป็นมาตรฐานกระบวนการผลิตซอฟต์แวร์ไอเอสโอ 29110 ผลการทดลอง ใช้พบว่าสามารถบันทึกเวลาการเรียนและการมาปฏิบัติงานของนักเรียนและบุคลากรที่ได้จากการสแกน ลายนิ้วมือ ระบบจะส่งรายงานการมาเรียนและการมาปฏิบัติงานให้กับอาจารย์ ผู้บริหารสถานศึกษาผ่านทาง อีเมลโดยอัตโนมัติ

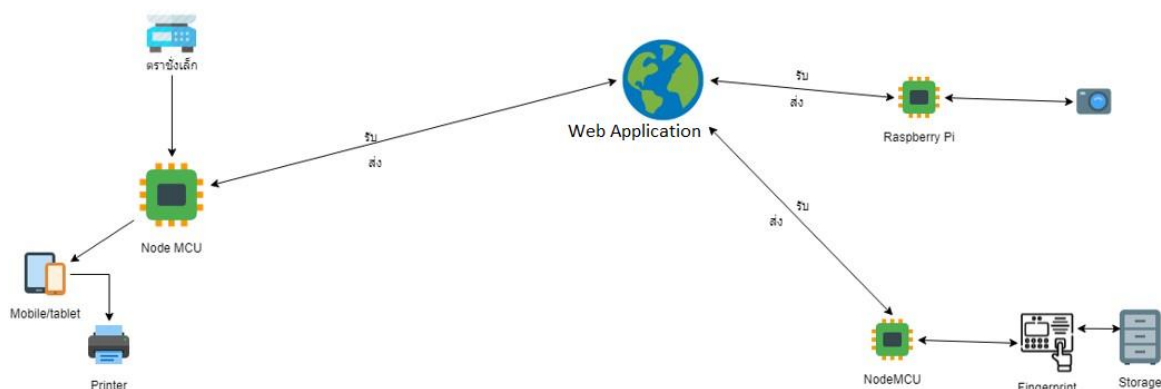
บทที่ 3

วิธีดำเนินงานโครงการ

ในการทำโครงการนี้ได้แบ่งการออกแบบเป็น 4 ส่วน คือ การออกแบบอุปกรณ์ที่ใช้ติดต่อสื่อสารกับเครื่องชั่ง, การออกแบบอุปกรณ์พิสูจน์ตัวตนผ่านอุปกรณ์สแกนลายนิ้วมือ, การออกแบบอุปกรณ์บันทึกภาพแบบออนไลน์, การออกแบบเว็บแอปพลิเคชันเพื่อใช้สำหรับติดต่อสื่อสารกับอุปกรณ์ที่พัฒนาขึ้นดังนี้

3.1 ภาพรวมของระบบการจัดการโรงขยะรีไซเคิล

กล่าวถึงขั้นตอนการทำงานของอุปกรณ์ต่าง ๆ ที่พัฒนาขึ้นโดยใช้ เว็บแอปพลิเคชันที่ออกแบบขึ้นโดยสั่งทำงานอยู่ 3 อุปกรณ์โดยเครื่องชั่งนั้นจะส่งข้อมูลน้ำหนักมายัง Node-MCU โดยใช้ RS232 เป็น Interface ที่ใช้ในการสื่อสาร, อุปกรณ์พิสูจน์ตัวตนผ่านอุปกรณ์สแกนลายนิ้วมือจะใช้ R307 Module Fingerprint กับ Node-MCU โดยเขียนโปรแกรมให้ทำงานอยู่ 2 คำสั่งคำสั่งแรกคือลงทะเบียนลายนิ้วมือ คำสั่งที่ 2 คือ อ่านลายนิ้วมือ, อุปกรณ์บันทึกภาพแบบออนไลน์จะใช้ USB-Cam กับ Raspberry-Pi โดยจะใช้ภาษา Python ในการเขียนโปรแกรมควบคุมให้อุปกรณ์สามารถทำงาน , เว็บแอปพลิเคชันใช้สำหรับสั่งให้อุปกรณ์ต่าง ๆ ทำงานโดยเฟรมเวิร์กที่ใช้คือ Bootstarp4 และ PHP โดยจะทำการออกแบบหน้าเว็บและฟังก์ชันต่าง ๆ ให้สามารถใช้งานได้ขึ้น



รูปที่ 3.1 ภาพรวมที่ทำการออกแบบทั้งหมด

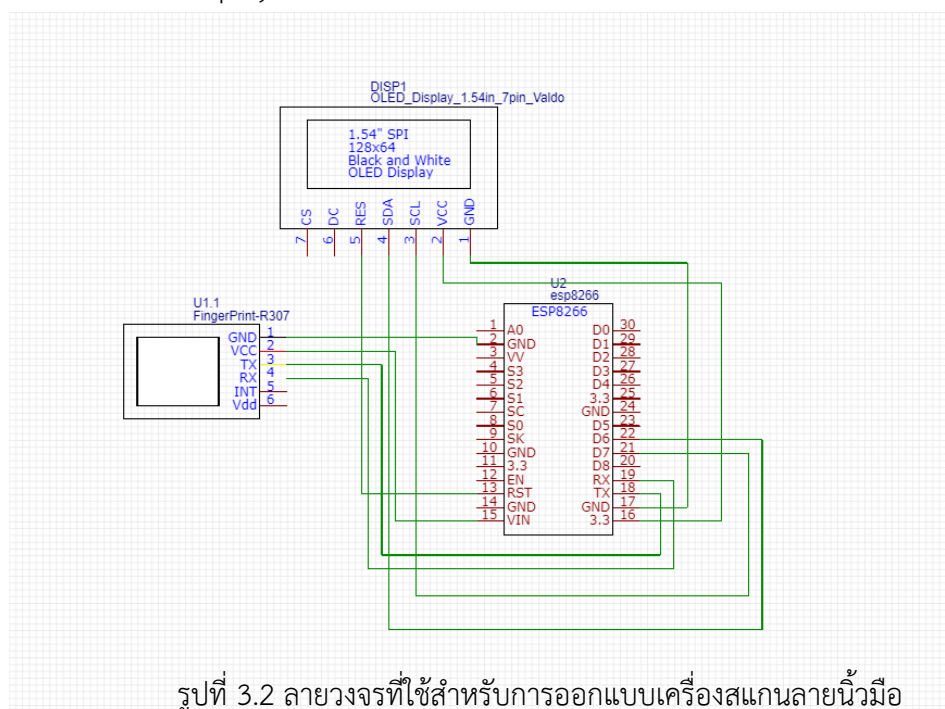
3.2 ออกแบบลายวงจรอุปกรณ์สแกนลายนิ้วมือแบบออนไลน์

กล่าวถึงขั้นตอนการต่อแผงวงจรของอุปกรณ์ที่ใช้สำหรับสแกนลายนิ้วมือ โดยมีอุปกรณ์ 3 ชิ้น

3.2.1 ESP8266

3.2.2 FingerPrint-R307

3.2.3 OLED Display Model SSD1309



รูปที่ 3.2 ลายวงจรที่ใช้สำหรับการออกแบบเครื่องสแกนลายนิ้วมือ

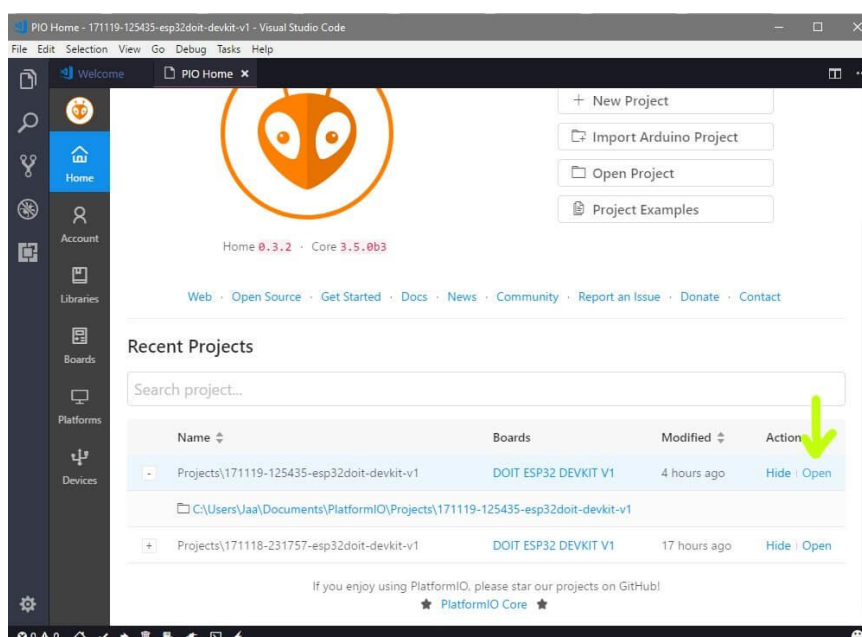
3.3 ขั้นตอนการออกแบบและการเชื่อมต่ออุปกรณ์สแกนลายนิ้วมือแบบออนไลน์

กล่าวถึงขั้นตอนการประกอบในส่วนของอุปกรณ์สแกนลายนิ้วมือแบบออนไลน์ทำหน้าที่ตรวจสอบและลงทะเบียนข้อมูลผู้มาติดต่อขายขยะรีไซเคิล โดยมีขั้นตอนดังนี้

3.3.1 อัปโหลดโปรแกรม

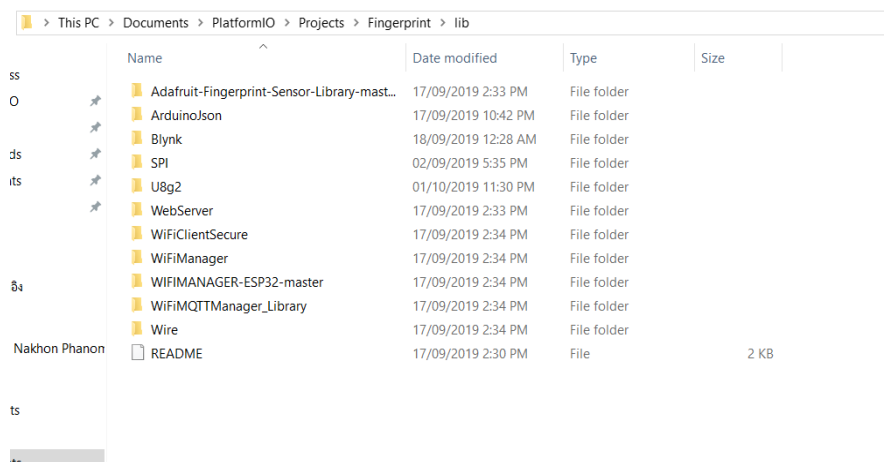
ขั้นตอนการบันทึกโปรแกรมเพื่อใช้แสดงเว็บไซต์สำหรับอุปกรณ์สแกนลายนิ้วมือ
กล่าวถึงขั้นตอนการอัปโหลดโปรแกรมเพื่อใช้สั่งให้อุปกรณ์สแกนลายนิ้วมือทำงานแล้วส่งค่าไปยัง
เว็บเซิร์ฟเวอร์

3.3.1.1 ทำการติดตั้ง Platformio



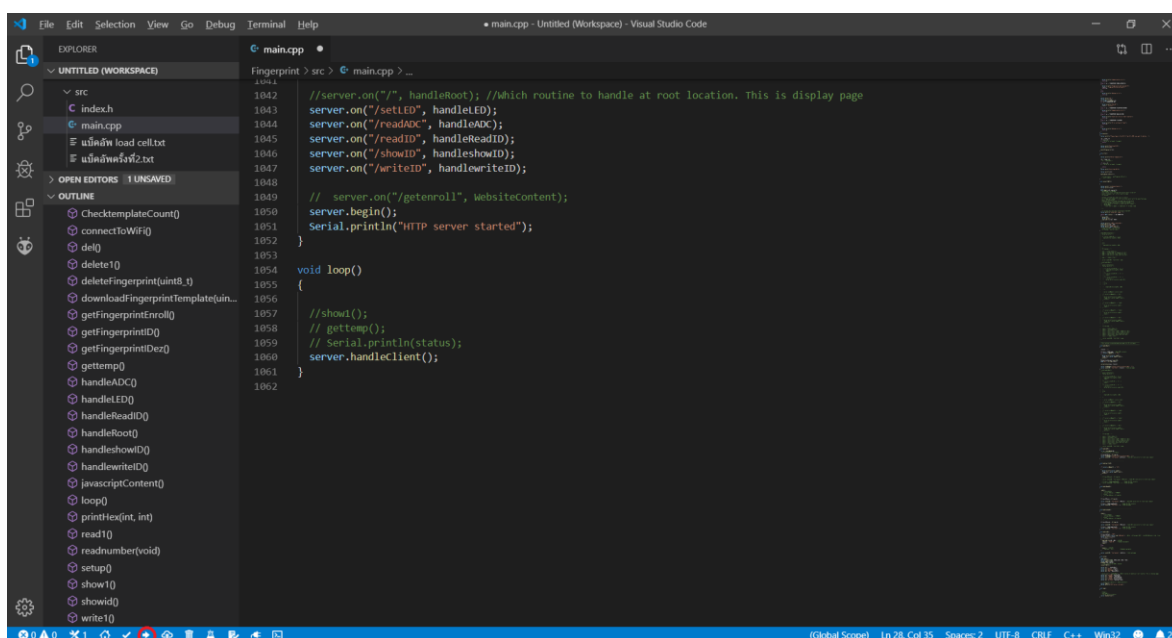
รูปที่ 3.3 ติดตั้ง platformio จาก Visual Studio Code

3.3.1.2 ทำการโหลดไลบรารีจาก Could มาไว้โฟลเดอร์ lib



รูปที่ 3.4 ติดตั้ง ไลบรารีลงในโฟลเดอร์

3.3.1.3 ทำการอัปโหลดโค้ดในภาคผนวก

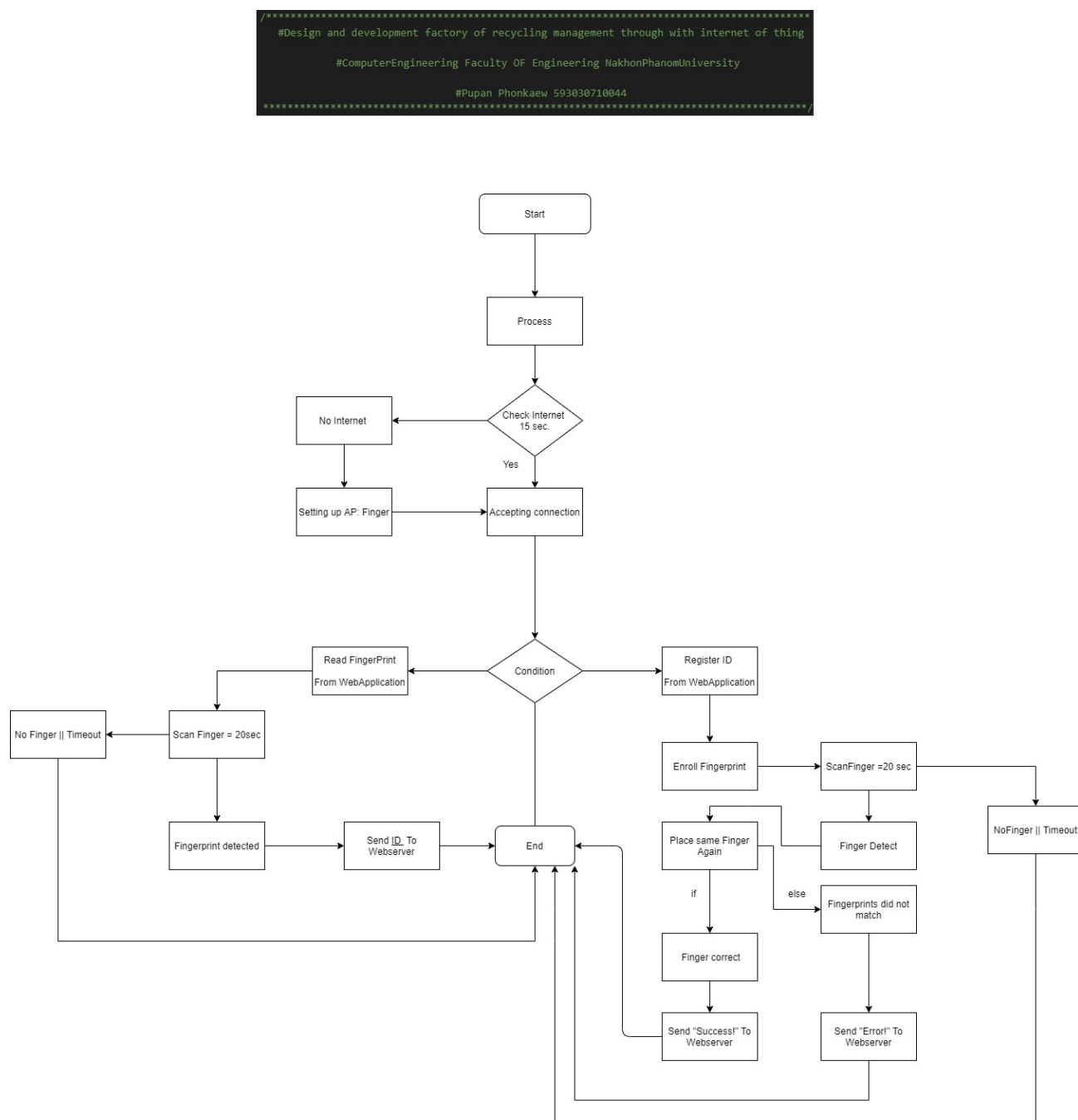


รูปที่ 3.5 อัปโหลดโปรแกรมเพื่อใช้งาน



รูปที่ 3.6 อุปกรณ์พร้อมสำหรับการใช้งาน

3.3.2 Flow Chart การทำงาน



รูปที่ 3.7 การทำงานของ Fingerprint

โดยจะทำงานอยู่สองฟังก์ชันคือฟังก์ชันอ่านลายนิ้วมือกับลงทะเบียนลายนิ้วมือส่วนอ่านจะถูกเขียนให้รอคำสั่งเมื่อมีคำสั่งให้สแกนก็จะทำงานเป็นเวลา 20 วินาที โดยจะรอลายนิ้วมือเมื่อมีลายนิ้วมือก็จะทำการตรวจสอบลายนิ้วมือว่าตรงตามข้อมูลที่ลงทะเบียนหรือไม่เมื่อตรงก็จะส่งข้อมูลไปเก็บไว้ที่ เว็บเซิร์ฟเวอร์เพื่อที่จะให้แอปพลิเคชันบนเว็บนำข้อมูลไปใช้งานแต่ถ้าไม่มีลายนิ้วมือในเวลาที่กำหนดก็จะทำการแจ้งเตือนไปยัง เว็บเซิร์ฟเวอร์ว่า Time Out ส่วนของลงทะเบียนนั้นจะแตกต่างตรงที่ ต้องทำการลงทะเบียนลายนิ้วมือเป็นจำนวน 2 ครั้งเพื่อเป็นการตรวจสอบความถูกต้องโดยจะทำการเขียนให้รอนิ้วมาสแกนเป็นเวลา 20 วินาทีเมื่อมีนิ้วเข้ามาสแกนถ้าข้อมูลตรงกันลายนิ้วมือจะถูกเก็บไว้ใน Database เพื่อที่จะใช้เรียกข้อมูล แต่ถ้าลายนิ้วมือไม่ตรงกันหรือไม่มีลายนิ้วมือสแกนบนเครื่องในเวลาที่กำหนด ก็จะส่ง Error หรือ Time out ไปยัง เว็บเซิร์ฟเวอร์ที่สร้างขึ้น

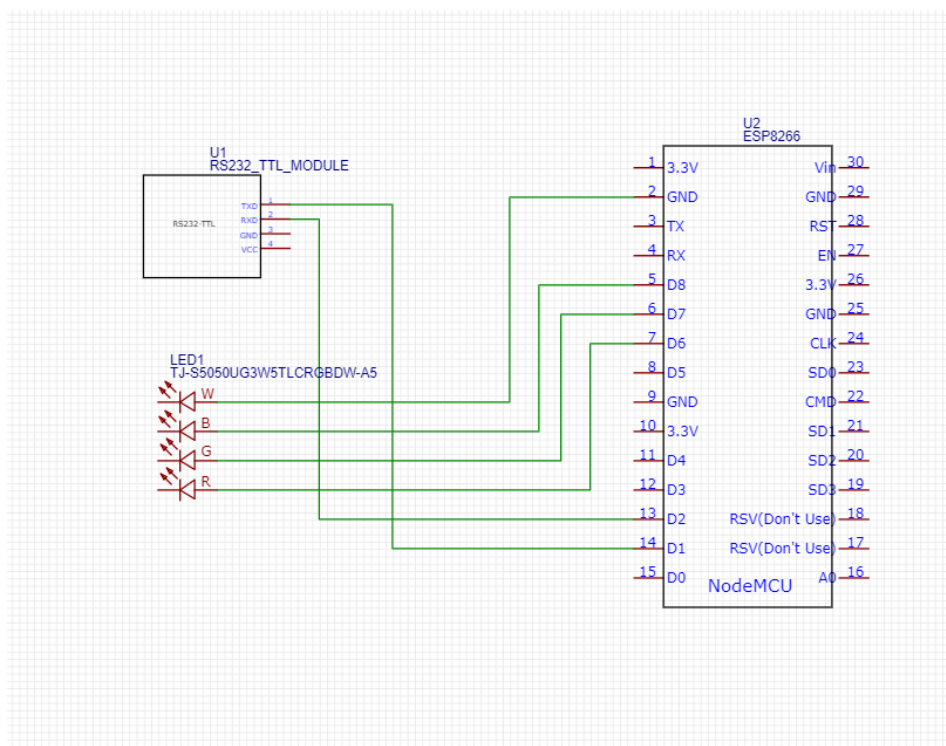
3.4 ขั้นตอนการออกแบบการเชื่อมต่อของเครื่องชั่งน้ำหนักแบบออนไลน์

กล่าวถึงขั้นตอนการต่อแผงวงจรของอุปกรณ์ที่ใช้สำหรับสแกนลายนิ้วมือ โดยมีอุปกรณ์ 3 ชิ้น

3.4.1 ESP8266

3.4.2 Rs232

3.4.3 LED RGB



รูปที่ 3.8 ออกแบบลายวงจรที่ใช้สำหรับอุปกรณ์รับค่าจากชั่งน้ำหนักแบบออนไลน์

3.5 ขั้นตอนการติดต่อสื่อสารเครื่องซังน้ำหนักแบบออนไลน์

เพื่อที่จะใช้สำหรับส่งข้อมูลน้ำหนักจากเครื่องซังไปยังแอปพลิเคชันบนเว็บที่พัฒนาขึ้นโดยมีขั้นตอนดังนี้

3.5.1 อัปโหลดโปรแกรม

ขั้นตอนการบันทึกภาษาโปรแกรมเพื่อใช้แสดงในเว็บไซต์ กล่าวถึงขั้นตอนการบันทึกภาษาโปรแกรมเพื่อใช้ส่งให้อุปกรณ์ซังน้ำหนกแบบออนไลน์ ดึงค่าจากเครื่องซังแล้วส่งค่าไปยัง เว็บเซิร์ฟเวอร์

3.5.1.1 โลบาร์ที่ใช้สำหรับอุปกรณ์ซังน้ำหนกแบบออนไลน์

โดยทำการโหลดมาติดตั้งไว้ในFolder ที่สร้างProject Platform IO

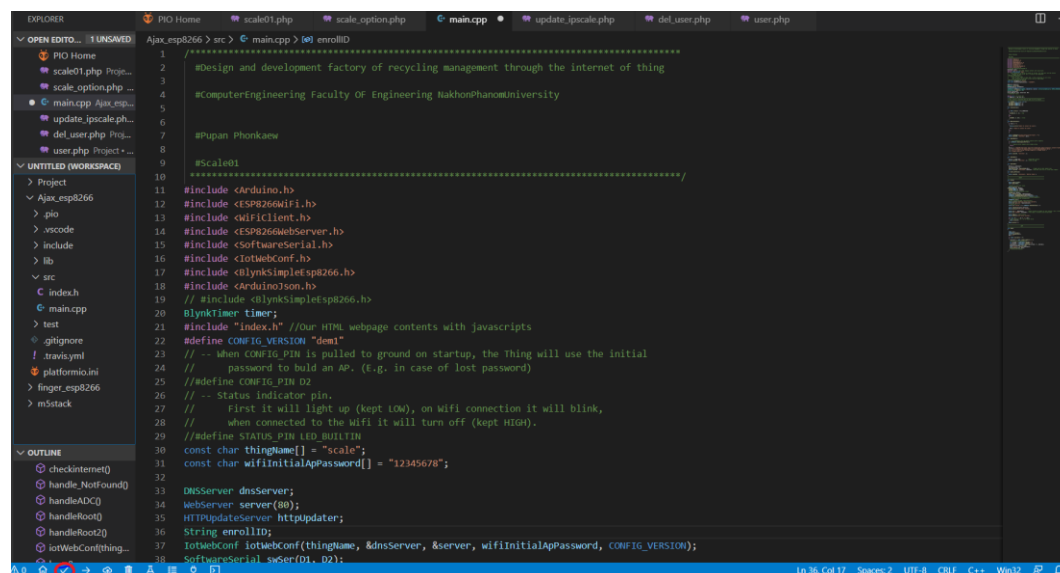
C:\Users\d_do7\Documents\PlatformIO\Projects\Scale_esp8266\lib

<https://github.com/bblanchon/ArduinoJson>

<https://github.com/plerup/espsoftwareserial>

<https://github.com/prampec/lotWebConf>

3.5.1.2 ทำการอัปโหลดโค้ด แสดงในภาคผนวก



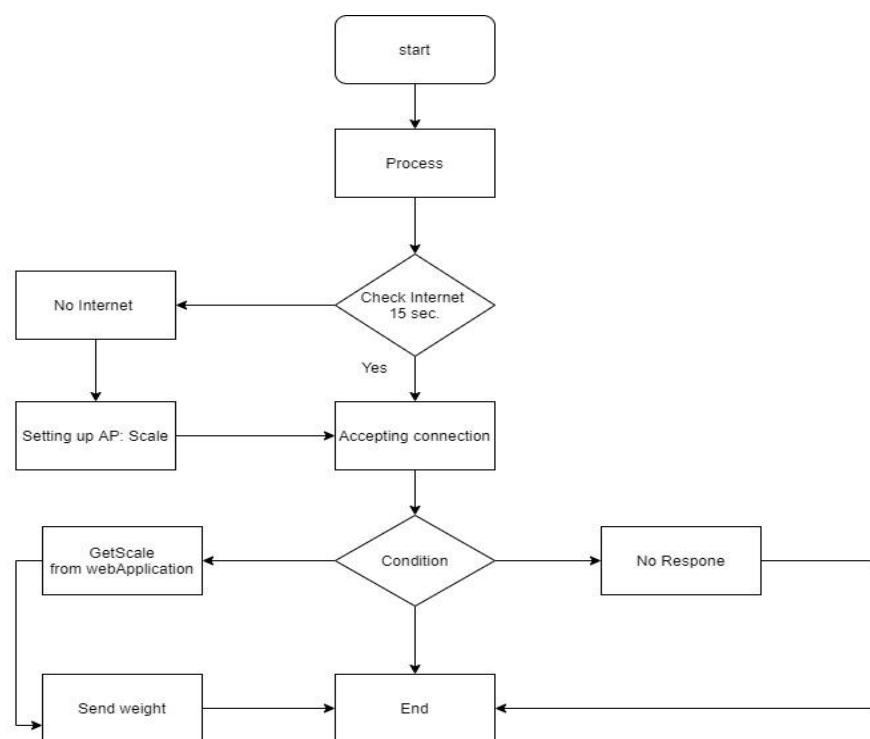
รูปที่ 3.9 อัปโหลดโปรแกรมเพื่อใช้งาน

3.5.1.3 อุปกรณ์พร้อมใช้งาน



รูปที่ 3.10 อุปกรณ์ที่ออกแบบกับเครื่องชั่ง

3.5.2 Flow Chart การทำงานของ เครื่องชั่งน้ำหนักแบบออนไลน์



รูปที่ 3.11 การทำงานของเครื่องชั่ง

หลักการทำงานเริ่มต้นใช้งานจะทำการเช็คว่ามี การเชื่อมต่อ กับอินเทอร์เน็ตหรือไม่ถ้ามีก็จะทำการเช็ค ว่าอุปกรณ์ที่พัฒนาขึ้นสามารถเชื่อมต่อกับเครื่องชั่งได้หรือไม่ถ้าติดต่อก็จะทำการนำข้อมูลที่ได้มาเก็บไว้เว็บเซิร์ฟเวอร์เพื่อที่จะให้ เว็บแอปพลิเคชัน นำข้อมูลน้ำหนักที่ได้ไปโชว์

3.6 ขั้นตอนการออกแบบและการเชื่อมต่อของเครื่องบันทึกภาพแบบออนไลน์

กล่าวถึงขั้นตอนการประกอบเครื่องบันทึกภาพแบบออนไลน์และการใช้งาน โดยมีอุปกรณ์ 2 ชิ้น

- Raspberry pi
- WebCamera



รูปที่ 3.12 อุปกรณ์ที่ใช้สำหรับบันทึกภาพแบบออนไลน์

3.7 ขั้นตอนการประกอบเครื่องบันทึกภาพแบบออนไลน์และการเชื่อมต่อ

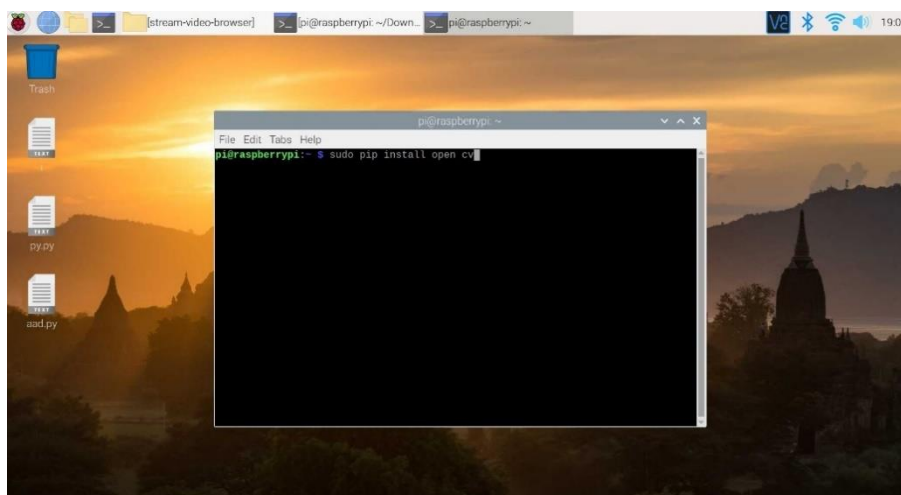
กล่าวถึงขั้นตอนการประกอบเครื่องบันทึกภาพแบบออนไลน์และการใช้งาน

3.7.1 ติดตั้ง ส่วนขยายที่จำเป็นต่อการใช้งาน

3.7.1.1 ติดตั้ง OPEN CV

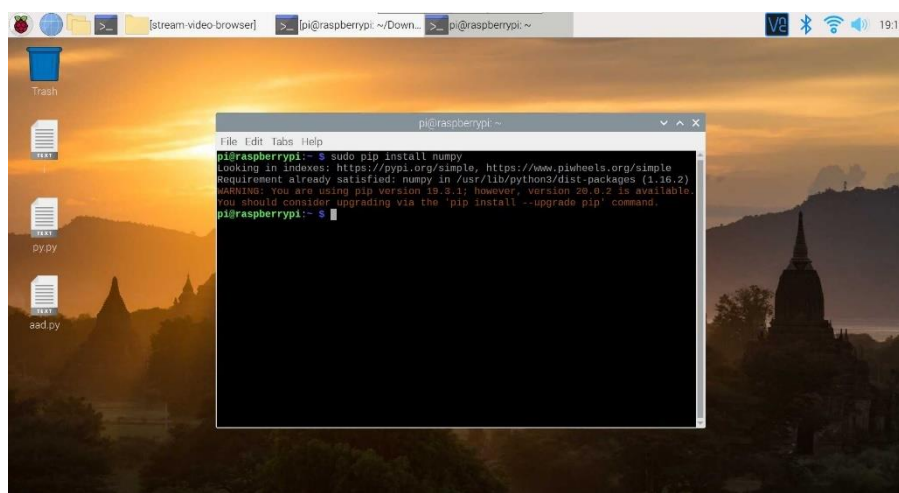
<https://github.com/opencv/opencv>

pip install opencv-python



รูปที่ 3.13 ติดตั้ง ส่วนขยายที่จำเป็นต่อการใช้งาน

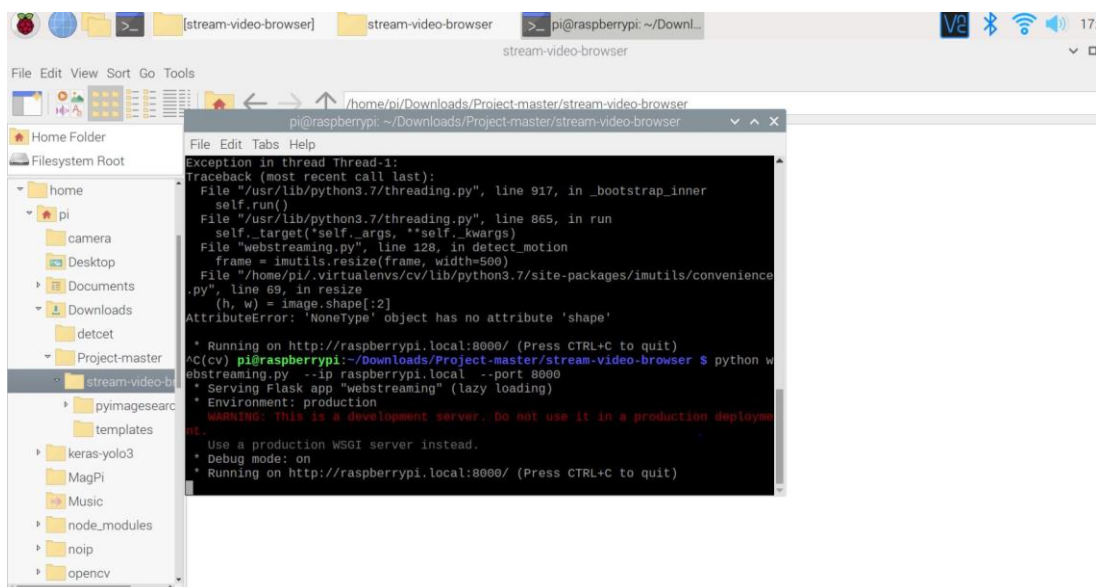
3.7.1.2 ติดตั้ง Numpy <https://github.com/numpy/numpy> pip install numpy



รูปที่ 3.14 ติดตั้ง ส่วนขยายที่จำเป็นต่อการใช้งาน

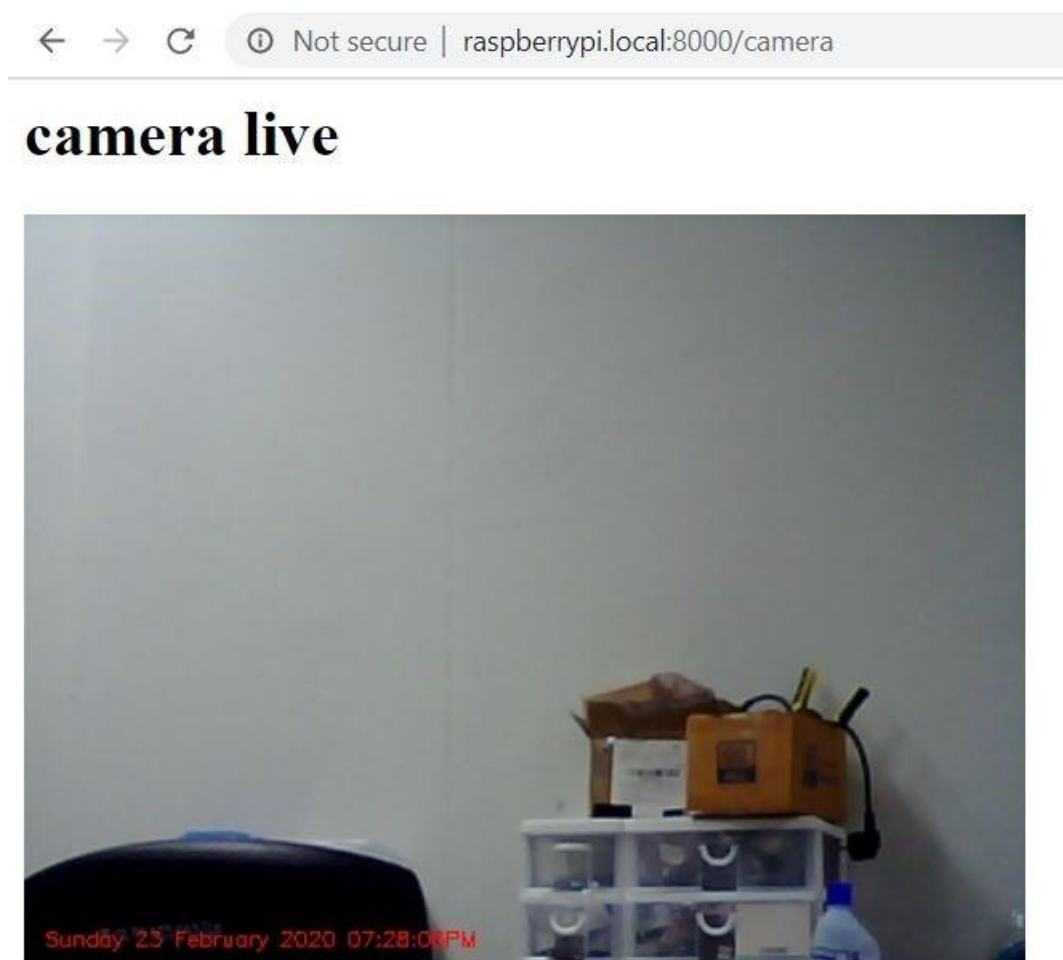
3.7.2 อัปเดตโปรแกรมเพื่อเรียกใช้งานแสดงในภาคผนวก

3.7.2.1 เรียกใช้อุปกรณ์บันทึกภาพแบบออนไลน์



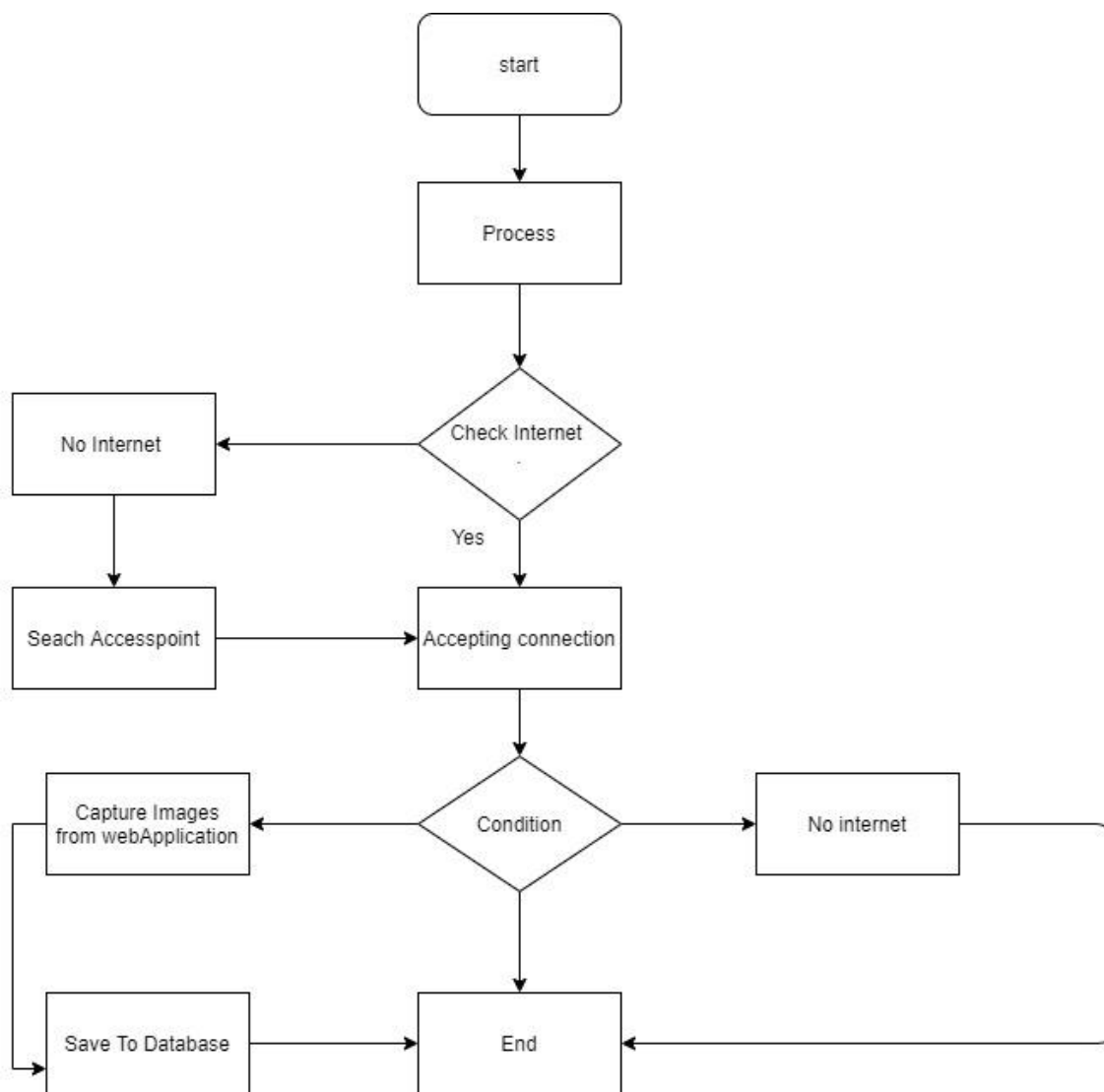
รูปที่ 3.15 ขั้นตอนการใช้งาน

3.7.2.2 เรียกใช้อุปกรณ์บันทึกภาพแบบออนไลน์



รูปที่ 3.16 ขั้นตอนการใช้งานอุปกรณ์บันทึกภาพแบบออนไลน์

3.7.3 Flow Chart การทำงาน



รูปที่ 3.17 Flow Chart การทำงาน

หลักการทำงานโดยจะรอคำสั่งจากแอปพลิเคชันบนเว็บเมื่อมีคำสั่งให้กดถ่ายภาพอุปกรณ์ก็จะทำการถ่ายภาพหลังจากนั้นรูปถ่ายนั้นจะถูกส่งมาเก็บไว้ใน Database

3.8 ขั้นตอนการออกแบบแอปพลิเคชันบนเว็บและระบบฐานข้อมูล

กล่าวถึงโปรแกรมในการออกแบบแอปพลิเคชันบนเว็บและระบบฐานข้อมูลเพื่อที่จะใช้กับอุปกรณ์ที่พัฒนาขึ้น

- Visual Studio
- Mysql



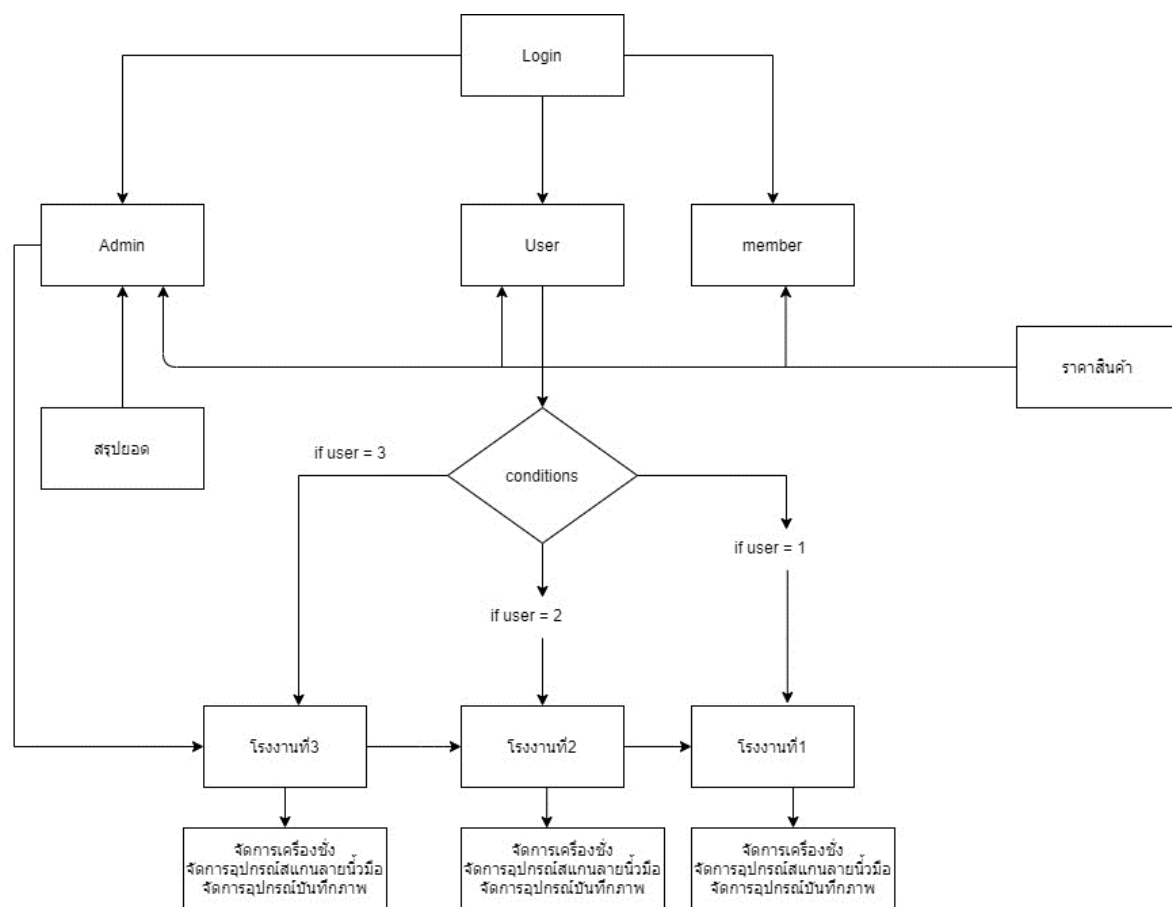
รูปที่ 3.18 โปรแกรมที่ใช้ในการออกแบบแอปพลิเคชันบนเว็บและระบบฐานข้อมูล

3.9 ขั้นตอนขั้นตอนการออกแบบแอปพลิเคชันบนเว็บและระบบฐานข้อมูล

กล่าวถึงขั้นตอนการออกแบบแอปพลิเคชันบนเว็บและระบบฐานข้อมูลเพื่อที่จะใช้รองรับอุปกรณ์ที่พัฒนาขึ้น

3.9.1 ภาพรวมของเว็บแอปพลิเคชัน

โดยได้ทำการออกแบบให้สามารถรองรับการใช้งานฟังก์ชันต่าง ๆ ตามระดับผู้ใช้งานโดย Admin จะสามารถใช้งานฟังก์ชันได้ทุกฟังก์ชันส่วนของ user จะแบ่งการใช้งานโดย user1 จะใช้งานได้ โรงงานที่ 1 user2 จะใช้งานได้ โรงงานที่ 2 user3 จะใช้งานได้ โรงงานที่ 3 ส่วนสุดท้ายคือ memberจะสามารถดูได้แค่ราคาสินค้าเท่านั้น



รูปที่ 3.19 ภาพรวมของเว็บแอปพลิเคชัน

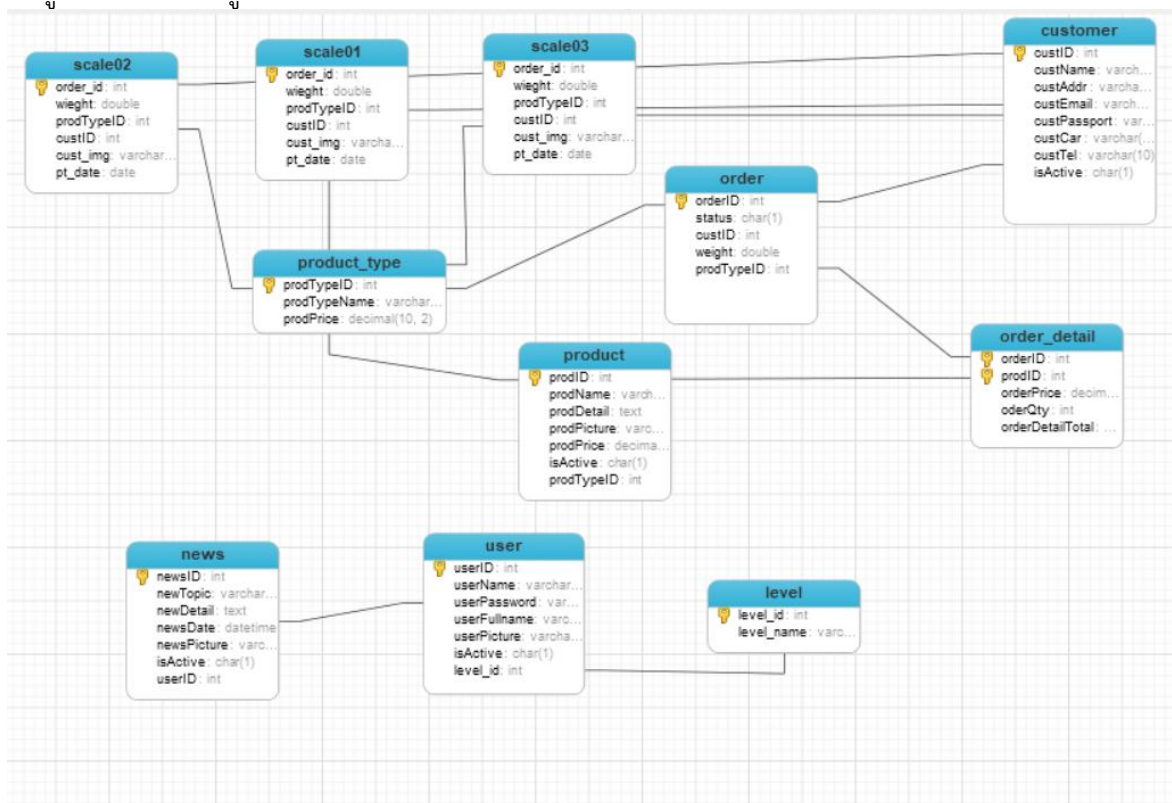
3.9.2 ออกแบบระบบฐานข้อมูลที่ใช้สำหรับเก็บข้อมูลต่าง ๆ ของอุปกรณ์ที่พัฒนาขึ้น

โดยได้ทำการออกแบบฐานข้อมูลสำหรับใช้เก็บข้อมูลสำคัญต่าง ๆ ส่วนหนึ่งก็คือลดความซ้ำซ้อนข้อมูลแถมด้วยลดพื้นที่เก็บไปได้ normalize การนอร์มัลไลเซชัน (Normalization) เป็นทฤษฎีหรือกระบวนการที่ใช้ในการทำให้เอนทิตีและแอททริบิวต์ที่ได้ออกแบบไว้มาจัดกลุ่มตารางให้ความสัมพันธ์กัน ให้อยู่ในรูปแบบที่เรียกว่า รูปแบบบรรทัดฐาน หรือ Normal Form เพื่อให้ความสัมพันธ์ที่ได้อยู่ในรูปแบบบรรทัดฐานที่เหมาะสม โดยจุดมุ่งหมายของการนอร์มัลไลเซชัน คือ

3.9.2.1 ลดความซ้ำซ้อน (Redundancy) ของข้อมูลในตาราง ทำให้ไม่ต้องแก้ไขข้อมูลหลาย ๆ ที่ รวมทั้ง ทำให้ลดเนื้อที่ในการจัดเก็บข้อมูล

3.9.2.2 ทำให้การเปลี่ยนแปลงแก้ไขโครงสร้างของตารางได้ง่ายในภายหลัง

3.9.2.3 ลดปัญหาความไม่ถูกต้องของข้อมูลโดยการปรับปรุงข้อมูลสามารถทำการปรับปรุงข้อมูล จากแหล่งข้อมูลเพียงแหล่งเดียว



รูปที่ 3.20 การทำ Normalization

3.9.2.4 ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้สำหรับฟังก์ชันล็อกอิน

Server: 127.0.0.1 • Database: project • Table: user

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	userID	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	userName	varchar(100)	utf8_general_ci		Yes	NULL			Change Drop More
3	userPassword	varchar(200)	utf8_general_ci		Yes	NULL			Change Drop More
4	userFullname	varchar(200)	utf8_general_ci		Yes	NULL			Change Drop More
5	userPicture	varchar(255)	utf8_general_ci		Yes	NULL			Change Drop More
6	isActive	char(1)	utf8_general_ci		Yes	NULL			Change Drop More
7	level_id	int(11)			Yes	NULL			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after level_id Go

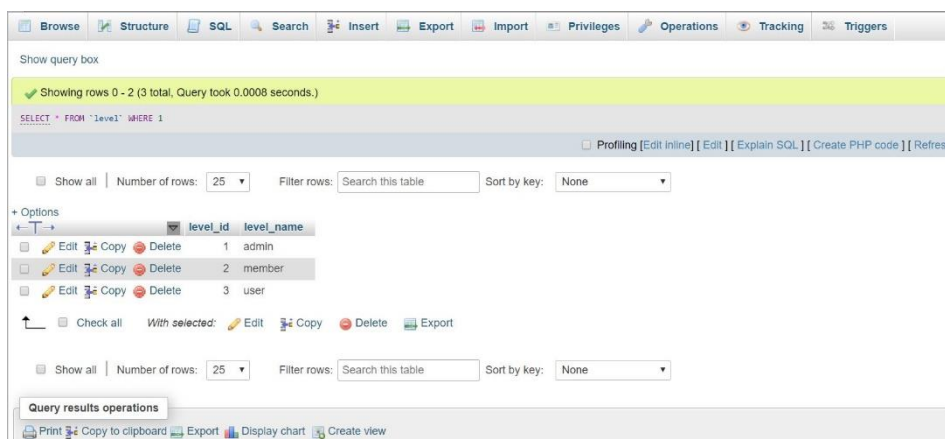
Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	userID	7	A	No	
Edit Drop	level_id	BTREE	No	No	level_id	7	A	Yes	

Create an index on 1 columns Go

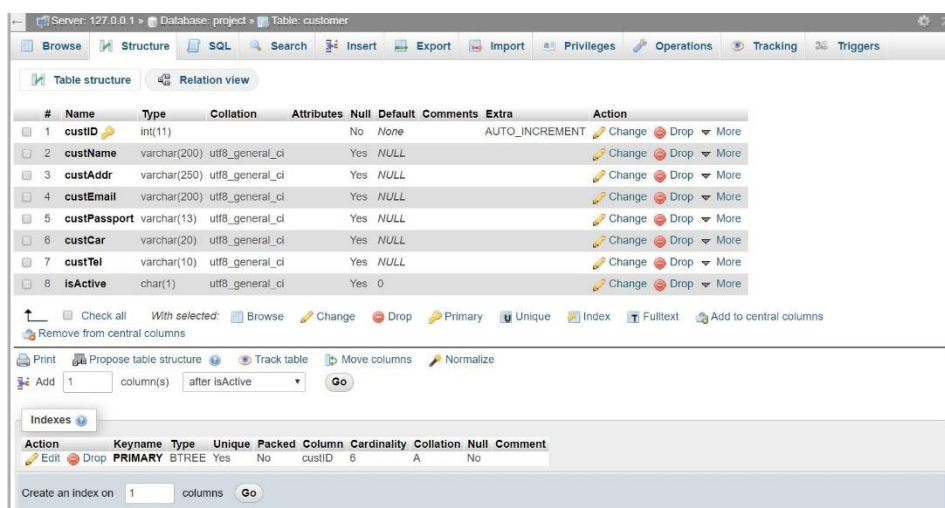
รูปภาพที่ 3.21 ออกแบบระบบฐานข้อมูลเพื่อ

3.9.2.5 ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้สำหรับแบ่งระดับการใช้งาน



รูปที่ 3.22 ออกแบบระบบฐานข้อมูลเพื่อจะใช้งบอกระดับการใช้งาน

3.9.2.6 ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้สำหรับฟังก์ชันข้อมูลลูกค้า



รูปที่ 3.23 ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้เก็บข้อมูลสมาชิกที่ทำการลงทะเบียน

3.9.2.7 ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้สำหรับฟังก์ชันข้อมูลสินค้า

Showing rows 0 - 7 (8 total, Query took 0.0007 seconds.) [prodPrice: 20.30... - 1.50...]

SELECT * FROM product_type ORDER BY prodPrice DESC

Options

	prodTypeID	prodTypeName	prodPrice
<input type="checkbox"/>	49	กระดาษ	20.30
<input type="checkbox"/>	37	ทองแดง	13.50
<input type="checkbox"/>	50	สังกะสี	9.00
<input type="checkbox"/>	40	เหล็ก	7.50
<input type="checkbox"/>	39	กรรมโรงพลาสติก	4.50
<input type="checkbox"/>	1	ขวดแก้ว	3.50
<input type="checkbox"/>	51	เศษเหล็ก	3.50
<input type="checkbox"/>	42	กรรมโรงเหล็ก	1.50

Query results operations

Print Copy to clipboard Export Display chart Create view

รูปที่ 3.24 ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้เก็บข้อมูลสินค้าและราคาสินค้า

3.9.2.8 ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้สำหรับบันทึกข้อมูลการรับซื้อจากเครื่องชั่ง

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	order_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	weight	double(100,2)			Yes	NULL			Change Drop More
3	prodTypeID	int(11)			Yes	NULL			Change Drop More
4	custID	int(11)			Yes	NULL			Change Drop More
5	cust_img	varchar(255)	utf8_general_ci		No	None			Change Drop More
6	pt_date	date			No	None			Change Drop More

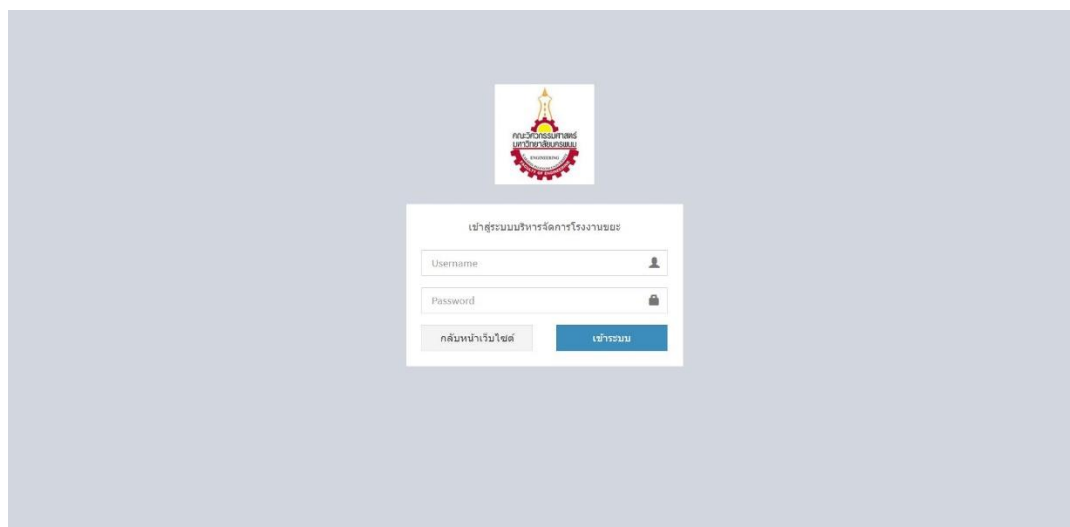
Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	order_id	79	A	No	
Edit Drop	prodTypeID	BTREE	No	No	prodTypeID	19	A	Yes	
Edit Drop	custID	BTREE	No	No	custID	4	A	Yes	

รูปที่ 3.25 ออกแบบระบบฐานข้อมูลเพื่อที่จะใช้เก็บข้อมูลการรับซื้อขยะรีไซเคิล

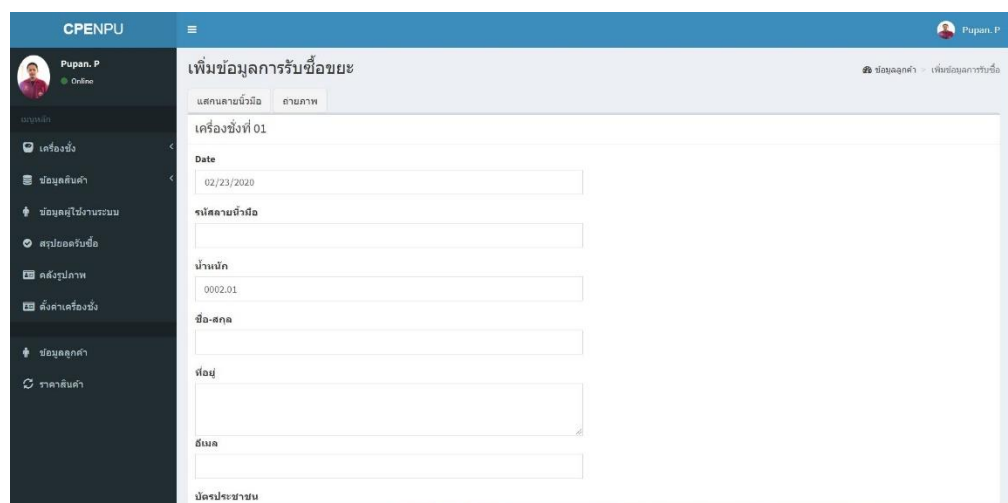
3.9.3 ออกแบบแอปพลิเคชันบนเว็บให้รองรับอุปกรณ์ที่พัฒนาขึ้น

3.9.3.1 ออกแบบหน้าเว็บให้รองรับระบบลือคอิน



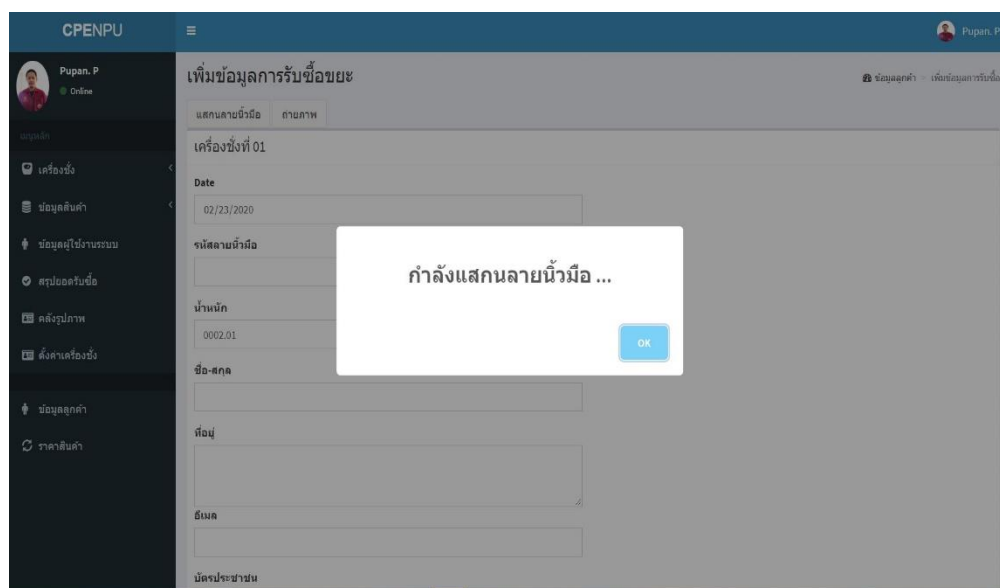
รูปที่ 3.26 ออกแบบหน้าเว็บให้รองรับระบบลือคอิน

3.9.3.2 ออกแบบแอปพลิเคชันบนเว็บให้สามารถทำงานร่วมกับอุปกรณ์ซึ่งนำหนักที่พัฒนาขึ้น



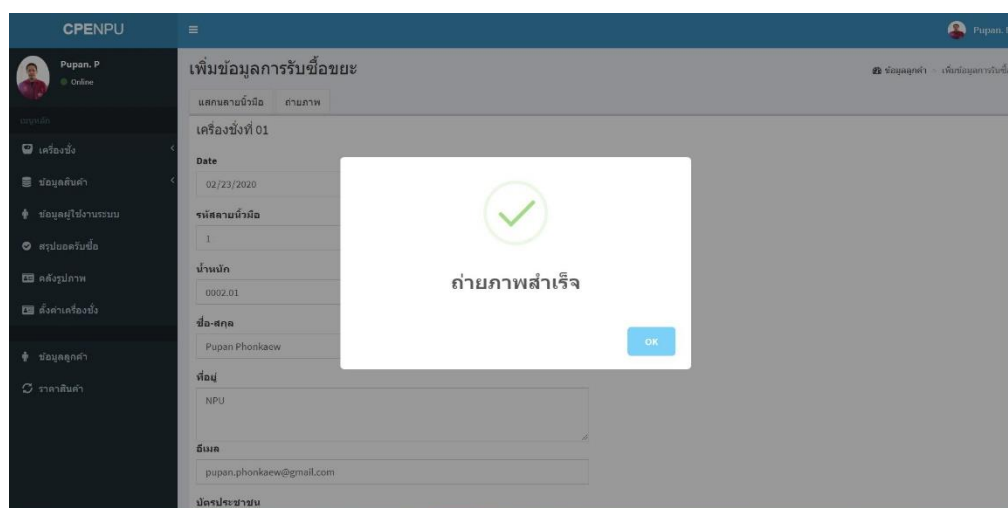
รูปที่ 3.27 แสดงค่าน้ำหนักลงในแอปพลิเคชันบนเว็บที่พัฒนาขึ้น

3.9.3.3 ออกแบบแอปพลิเคชันบนเว็บให้สามารถทำงานร่วมกับอุปกรณ์สแกนลายนิ้วมือแบบออนไลน์ที่พัฒนาขึ้น



รูปที่ 3.28 ทดสอบสั่งการทำงานอุปกรณ์สแกนลายนิ้วมือแบบออนไลน์

3.9.3.4 ออกแบบแอปพลิเคชันบนเว็บให้สามารถทำงานร่วมกับอุปกรณ์บันทึกภาพแบบออนไลน์ที่พัฒนาขึ้น



รูปที่ 3.29 ทดสอบสั่งการทำงานอุปกรณ์บันทึกภาพแบบออนไลน์จากแอปพลิเคชันบนเว็บที่พัฒนาขึ้น

บทที่ 4

ออกแบบการทดสอบ

บทความนี้จะกล่าวถึงแบบสอบถามความพึงพอใจของผู้บริหารและพนักงานจำนวน 10 คนในการใช้งานเว็บไซต์และการหาค่าเฉลี่ยความเร็วในการรับส่งข้อมูลไปยัง เว็บแอปพลิเคชัน จากอุปกรณ์ที่พัฒนาขึ้น เพื่อค้นหาค่าความผิดพลาด เวลาที่ใช้ในการทดสอบ และค่าความแม่นยำของอุปกรณ์

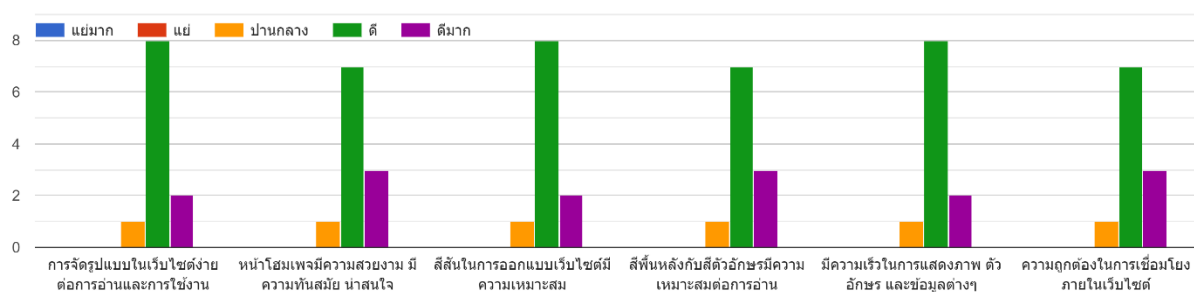
4.1 ออกแบบการทดลองสองแบบดังนี้

แบบที่ 1 โดยจะทำแบบสอบถามความพึงพอใจให้ผู้บริหารและพนักงานโรงงานหลังจากใช้งานว่าก่อนใช้อุปกรณ์ที่ออกแบบและพัฒนาขึ้นเป็นอย่างไรหลังจากใช้เป็นอย่างไร

แบบที่ 2 เป็นการทดสอบเชิงเทคนิคโดยจะทำการทดสอบ Response Time และ Request Time โดยจะนำอุปกรณ์ที่พัฒนาขึ้นทั้งหมดมาหาค่าเฉลี่ยอัตรารับส่งข้อมูลและอัตราตอบสนองของเวลา

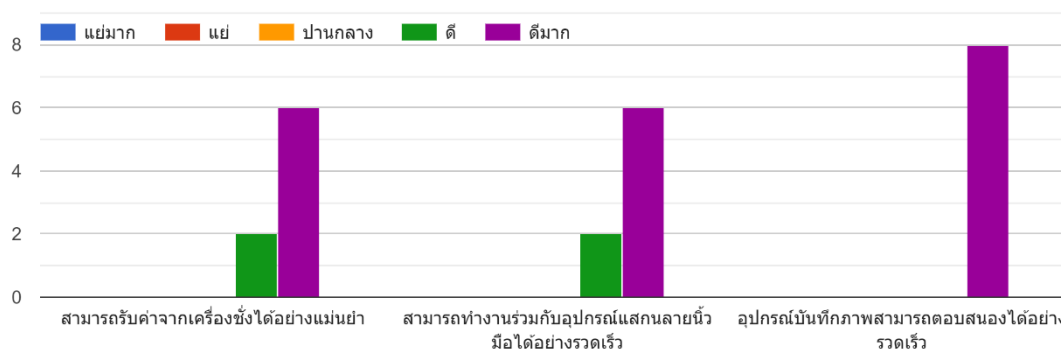
4.2 ผลการทดสอบความพึงพอใจ

ด้านการออกแบบและการจัดรูปแบบเว็บไซต์



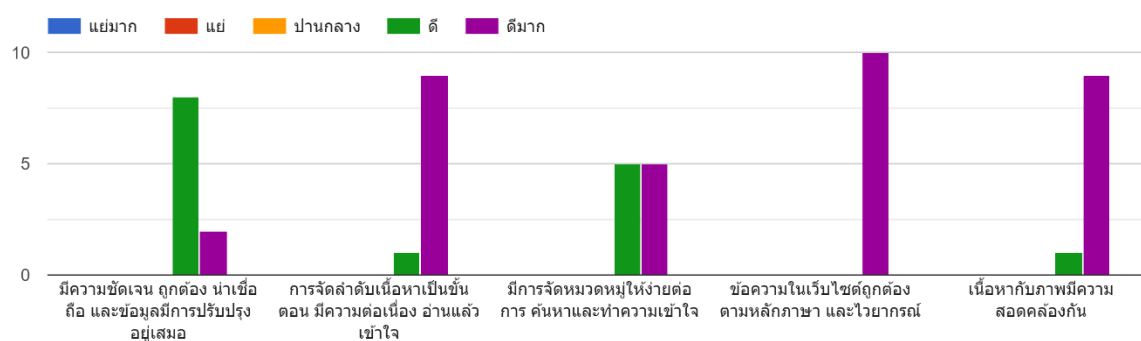
รูปที่ 4.1 ความพึงพอใจในการใช้งานเว็บไซต์ด้านการออกแบบและการจัดการรูปแบบเว็บไซต์

ด้านการติดต่อกับอุปกรณ์



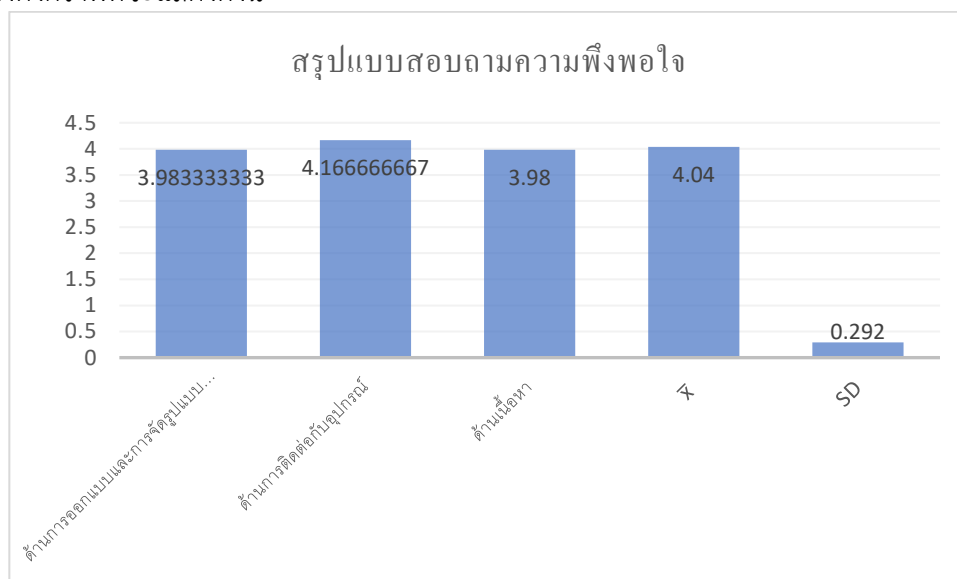
รูปที่ 4.2 ความพึงพอใจในการใช้งานเว็บไซต์ด้านการติดต่อกับอุปกรณ์

ด้านเนื้อหา



รูปที่ 4.3 ความพึงพอใจในการใช้งานเว็บไซต์ด้านเนื้อหา

สรุปแบบสอบถามความพึงพอใจ โดยได้ทำการสำรวจจากผู้บริหารและพนักงานจำนวน 10 คน ผลที่ออกมานั้นค่าเฉลี่ยความพึงพอใจจะอยู่ที่ 4.04 ซึ่งอยู่ในเกณฑ์ ดี โดยค่าเบี่ยงเบนมาตรฐานจะอยู่ที่ 0.292 σ โดยอ้างอิงดังกราฟที่จะแสดงดังนี้



รูปที่ 4.4 สรุปแบบสอบถามความพึงพอใจ

4.3 อุปกรณ์การทดสอบ

- เครื่องชั่ง
- อุปกรณ์สแกนลายนิ้วมือ
- อุปกรณ์บันทึกภาพ
- การทำงานของเว็บแอปพลิเคชันร่วมกับอุปกรณ์ต่าง ๆ

4.4 การหาค่าความแม่นยำของอุปกรณ์สแกนลายนิ้วมือ และความเร็วในการรับส่งข้อมูล

ในส่วนนี้จะเป็นการทดลองเพื่อหาค่าความแม่นยำและเวลาที่ใช้ในการรับส่งข้อมูลของอุปกรณ์สแกนลายนิ้วมือจำนวน 5 ลายนิ้วมือโดยจะเปรียบเทียบค่าความแม่นยำจากตัวโปรแกรมที่ออกแบบขึ้นโดยจะทำการทดลองลายนิ้วมือละ 10 ครั้ง

ผลการทดลองครั้งที่	ค่าที่ได้(%)	ผิดพลาด (%)	เวลาที่ใช้(sec)
1	71	29	3.69
2	69	31	2.31
3	71	29	4.43
4	82	18	5.76
5	67	33	2.37
7	88	12	6.59
8	76	24	7.39
9	67	33	2.97
10	86	14	3.03
เฉลี่ย	75.222	24.777	4.282

ตารางที่ 4.1 ผลการทดลองค่าความแม่นยำของอุปกรณ์สแกนลายนิ้วมือที่ 1

ผลการทดลองครั้งที่	ค่าที่ได้(%)	ผิดพลาด (%)	เวลาที่ใช้(sec)
1	85	15	2.28
2	90	10	2.72
3	83	17	2.53
4	84	16	3.09
5	76	24	3.34
7	84	16	2.68
8	90	10	2.57
9	88	12	2.48
10	78	22	2.67
เฉลี่ย	84.222	15.777	2.706

ตารางที่ 4.2 ผลการทดลองค่าความแม่นยำของอุปกรณ์สแกนลายนิ้วมือที่ 2

ผลการทดลองครั้งที่	ค่าที่ได้(%)	ผิดพลาด (%)	เวลาที่ใช้(sec)
1	80	20	2.95
2	86	14	3.32
3	90	10	2.61
4	75	25	3.09
5	84	16	2.95
7	60	40	5.85
8	87	13	3.18
9	64	36	4.81
10	95	5	2.90
เฉลี่ย	80.111	19.888	3.517

ตารางที่ 4.3 ผลการทดลองค่าความแม่นยำและการส่งข้อมูลของอุปกรณ์สแกนลายนิ้วมือที่ 3

ผลการทดลองครั้งที่	ค่าที่ได้(%)	ผิดพลาด (%)	เวลาที่ใช้(sec)
1	66	34	6.15
2	81	19	3.36
3	57	43	2.92
4	56	44	3.23
5	86	14	3.58
7	62	38	3.26
8	74	26	2.92
9	60	40	2.92
10	52	48	5.15
เฉลี่ย	66	34	3.417

ตารางที่ 4.4 ผลการทดลองค่าความแม่นยำของอุปกรณ์สแกนลายนิ้วมือที่ 4

ผลการทดลองครั้งที่	ค่าที่ได้(%)	ผิดพลาด (%)	เวลาที่ใช้(sec)
1	63	37	6.00
2	62	38	5.54
3	80	20	5.11
4	85	15	2.71
5	77	23	2.77
7	93	7	5.68
8	58	42	2.71
9	83	17	12.98
10	76	24	3.56
เฉลี่ย	75.222	24.777	5.228

ตารางที่ 4.5 ผลการทดลองค่าความแม่นยำและการส่งข้อมูลของอุปกรณ์สแกนลายนิ้วมือที่ 5

สรุปผลการทดลอง 4.4 เครื่องสแกนลายนิ้วมือเมื่อเปลี่ยนวิธีสแกนโดยวางตำแหน่งของนิ้วไม่ตรงตามตำแหน่งที่ลงทะเบียนไว้ค่าแม่นยำที่ได้จะลดลงแต่ถ้าวางตำแหน่งนิ้วให้ตรงตามตำแหน่งที่ลงทะเบียนค่าแม่นยำที่ได้จะสูงโดยค่าผิดพลาดที่ได้ค่อนข้างต่ำถือว่าทำงานได้อย่างมีประสิทธิภาพ

ในส่วนเวลาที่ใช้รับส่งข้อมูลไปยังเซิร์ฟเวอร์นั้นถือว่าค่อนข้างเร็ว

4.5 การหาค่าความแม่นยำของเครื่องชั่งน้ำหนัก และความเร็วในการรับส่งข้อมูล

ในส่วนนี้จะเป็นการทดลองเพื่อหาค่าความแม่นยำของเครื่องชั่งและเวลาที่ใช้ในการรับส่งข้อมูล โดยจะเปรียบเทียบค่าความแม่นยำจากเครื่องชั่งโดยจะทำการทดลองโดยใช้น้ำหนักที่ 1 kg 10 kg 20 kg

ผลการทดลองครั้งที่	น้ำหนักที่ชั่ง(Kg)	ค่าที่ได้ (kg)	เวลาที่ใช้(ms)
1	1.00	1.00	97
2	1.00	1.00	123
3	1.00	1.00	134
4	1.00	1.00	122
5	1.00	1.00	113
7	1.00	1.00	117
8	1.00	1.00	101
9	1.00	1.00	105
10	1.00	1.00	85
เฉลี่ย	1	1	110.778

ตารางที่ 4.6 ผลการทดลองค่าความแม่นยำ 1kg

ผลการทดลองครั้งที่	น้ำหนักที่ชั่ง(Kg)	ค่าที่ได้ (kg)	เวลาที่ใช้(ms)
1	10.00	10.00	91
2	10.00	10.00	93
3	10.00	10.00	122
4	10.00	10.00	91
5	10.00	10.00	86
7	10.00	10.00	141
8	10.00	10.00	144
9	10.00	10.00	152
10	10.00	10.00	123
เฉลี่ย	10	10	115

ตารางที่ 4.7 ผลการทดลองค่าความแม่นยำ 10kg

ผลการทดลองครั้งที่	น้ำหนักที่ชั่ง(Kg)	ค่าที่ได้ (kg)	เวลาที่ใช้(ms)
1	20.00	20.00	64
2	20.00	20.00	64
3	20.00	20.00	65
4	20.00	20.00	63
5	20.00	20.00	63
7	20.00	20.00	63
8	20.00	20.00	64
9	20.00	20.00	63
10	20.00	20.00	63
เฉลี่ย	20	20	63.625

ตารางที่ 4.8 ผลการทดลองค่าความแม่นยำของอุปกรณ์สแกนลายนิ้วมือที่ 20kg

สรุปผลการทดลอง 4.5 จากการทดลองเครื่องชั่งน้ำหนักนั้นไม่มีค่าผิดพลาดที่ชั่งเลยโดยแสดงค่าน้ำหนักได้อย่างแม่นยำจากวัตถุที่ทำการชั่งในส่วนของเวลาที่ทำการรับส่งไปยังเซิร์ฟเวอร์สามารถรับส่งได้อย่างรวดเร็วสามารถรับค่าน้ำหนักที่ได้มาโชว์ที่เว็บแอปพลิเคชันแบบเรียลไทม์

4.6 การทดลองการทำงานของเว็บแอปพลิเคชัน

ในส่วนนี้จะเป็นการทดลองเพื่อหาเวลาที่ใช้ในการรับส่งข้อมูลเวลาที่ใช้ในการตอบสนองระหว่างเว็บแอปพลิเคชันกับ อุปกรณ์ต่าง ๆ ที่พัฒนาขึ้น ว่ามีเวลาในการรับส่งข้อมูลและตอบสนองมากน้อยแค่ไหน โดยจะส่งทำงานทั้ง 3 ระบบพร้อมกัน ได้แก่ เครื่องชั่ง อุปกรณ์สแกนลายนิ้วมือ กล้องบันทึกภาพแบบออนไลน์ เครื่องชั่งน้ำหนัก

4.6.1 ผลการทดลองการบันทึกภาพแบบออนไลน์และการตอบสนอง

ผลการทดสอบครั้งที่	เวลาที่ส่ง (ms)	เวลาที่ใช้ตอบกลับ (ms)	เวลารวม (ms)
1	0.22	137.50	137.72
2	0.078	219.41	219.488
3	0.14	107.32	107.46
4	0.13	109.82	109.95
5	0.18	125.99	126.17
6	0.1	371.72	371.82
7	0.11	66.33	66.44
8	0.096	111.07	111.166
9	0.16	99.57	99.73
10	0.13	110.74	110.87
เฉลี่ย	0.134	149.858	146.081

ตารางที่ 4.9 ผลการทดลองบันทึกภาพแบบออนไลน์ร่วมกับเว็บแอปพลิเคชันที่สร้างขึ้น

4.6.1.2 ผลการทดลองเครื่องสแกนลายนิ้วมือและเวลาที่ใช้ในการตอบสนอง

ผลการทดสอบครั้งที่	เวลาที่ส่ง (sec)	เวลาที่ใช้ในการรอ ข้อมูล (sec)	เวลาที่ใช้ในการ โหลดข้อมูล (sec)	เวลารวม (sec)
1	0.00019	1.34	0.06626	1.406
2	0.00025	1.56	0.18238	1.742
3	0.00017	2.06	0.05757	2.117
4	0.0002	2.07	0.05059	2.120
5	0.00011	1.77	0.05006	1.820
6	0.00014	2.37	0.05582	2.425
7	0.00022	1.35	0.0489	1.399
8	0.00026	2.07	0.05565	2.125
9	0.00012	2.03	0.05945	2.089
10	0.00016	1.25	0.06963	1.319
เฉลี่ย	0.000184	1.846667	0.069631	1.916

ตารางที่ 4.10 ผลการทดลองเครื่องสแกนลายนิ้วมือทำงานร่วมกับเว็บแอปพลิเคชันที่สร้างขึ้น

4.6.1.3 ผลการทดลองเครื่องสแกนลายนิ้วมือและเวลาที่ใช้ในการตอบสนอง

ผลการทดสอบครั้งที่	เวลาที่ส่ง (ms)	เวลาที่ใช้ในการรอ ข้อมูล (ms)	เวลาที่ใช้ในการ โหลดข้อมูล (ms)	เวลารวม (ms)
1	0.21	16.16	52.27	68.64
2	0.18	20.28	55.83	76.29
3	0.21	26.19	60.66	87.06
4	0.16	21.15	49.12	70.43
5	0.21	31.65	70.66	102.52
6	0.089	18.55	44.82	63.459
7	0.22	25.72	64.21	90.15
8	0.18	37.10	67.52	104.8
9	0.23	43.92	56.13	100.28
10	0.27	35.25	54.98	90.5
เฉลี่ย	0.1876	26.746	57.913	84.847

ตารางที่ 4.11 ผลการทดลองเครื่องสแกนลายนิ้วมือทำงานร่วมกับเว็บแอปพลิเคชันที่สร้างขึ้น

สรุปผลการทดลอง 4.6 เมื่อทำการทดลองอุปกรณ์ต่าง ๆ ที่พัฒนาขึ้นให้ทำงานร่วมกับเว็บแอปพลิเคชันใช้เวลาตอบสนองค่อนข้างเร็วและไม่มีข้อผิดพลาดเวลาที่ใช้ในการรับส่งข้อมูลค่อนข้างเร็วมากสามารถรู้ได้ว่าอุปกรณ์ต่าง ๆ ที่สร้างขึ้นสามารถทำงานร่วมกับเว็บแอปพลิเคชันได้อย่างรวดเร็วไม่มีปัญหาเหมาะที่จะบันทึกข้อมูลแบบเรียลไทม์

บทที่ 5

สรุปผลและข้อเสนอแนะ

สำหรับการพัฒนาการทำโครงงานครั้งนี้ซึ่งเกี่ยวกับการออกแบบและพัฒนาระบบการจัดการโรงงานขยะรีไซเคิล มีข้อเสนอแนะตามลำดับ ดังนี้

5.1 สรุปผล

จากผลการทดสอบสามารถสรุปได้ดังนี้

5.1.1 อุปกรณ์ที่ใช้ที่ใช้สื่อสารระหว่างเครื่องซึ่งกับอุปกรณ์ที่พัฒนาขึ้นนั้นสามารถทำงานร่วมกันได้อย่างมีประสิทธิภาพอัตราการรับส่งข้อมูลเฉลี่ยอยู่ที่ 110 ms.

5.1.2 อุปกรณ์ที่ใช้พิสูจน์ตัวตนด้วยลายนิ้วมือนั้นสามารถทำงานได้แม่นยำไม่มีข้อผิดพลาดโดยอัตราในการรับส่งข้อมูลนั้นเร็วสุดอยู่ที่ 1.5 sec. ช้าสุดที่ 2.5 sec.

5.1.3 อุปกรณ์บันทึกภาพแบบออนไลน์อัตราในการตอบสนองถือว่าค่อนข้างเร็วมากเฉลี่ย 84.847ms. โดยสามารถทำงานร่วมกับเว็บแอปพลิเคชันที่พัฒนาขึ้นได้อย่างมีประสิทธิภาพและไม่มีข้อผิดพลาด

5.1.4 เว็บแอปพลิเคชันที่ออกแบบและพัฒนาขึ้นนั้นสามารถทำงานร่วมกับอุปกรณ์ที่พัฒนาขึ้นได้ทุกส่วนที่ออกแบบโดยจะทำหน้าที่ส่งงานและรับข้อมูลจากอุปกรณ์ต่าง ๆ ที่ออกแบบและพัฒนาขึ้น

5.2 ข้อเสนอแนะ

ควรสร้างแอปพลิเคชันสำหรับใช้บนมือถือ

5.3 อุปสรรคการทำงาน

การส่งอุปกรณ์ที่ใช้ในการทำโครงงานมีการจัดส่งที่ล่าช้าเนื่องจากการทำโครงงานมีการใช้ Hardware ค่อนข้างมากทำให้ค่าใช้จ่ายสูงเมื่อส่งของในประเทศควรเลือกซื้ออุปกรณ์สำหรับพัฒนาในต่างประเทศ

ภาคผนวก

โค้ดที่ใช้ในการทำงาน

1. อุปกรณ์สแกนลายนิ้วมือแบบออนไลน์

```
/******
```

This is my project FingerPrint

Pupan Phonkaew | 593030710044

Computer Engineering | Faculty of Engineering NakhonPhanom University

```
*****/
```

```
#include <Arduino.h>
```

```
#include <IoTWebConf.h>
```

```
#include <U8g2lib.h>
```

```
#include <ESP8266WiFi.h>
```

```
#include <Adafruit_Fingerprint.h>
```

```
#include <ArduinoJson.h>
```

```
#include <SPI.h>
```

```
#include <BlynkSimpleEsp8266.h>
```

```
BlynkTimer timer;
```

```
U8G2_SSD1309_128X64_NONAME0_F_4W_SW_SPI u8g2(U8G2_R0, /* clock=*/D5, /* data=*/D7,
/* cs=*/D6, /* dc=*/D8, /* reset=*/D1);
```

```
const char thingName[] = "Finger01";
```

```
int FingerID;
```

```
int cnt1;
```

```
IPAddress ip;
```

```
String send, wifiip;
```

```
String timeout, timeout1;
```

```
// -- Initial password to connect to the Things, when it creates an own Access Point.
```



```

const char wifInitialApPassword[] = "12345678";

// -- Configuration specific key. The value should be modified if config structure was changed.
#define CONFIG_VERSION "dem1"

// -- When CONFIG_PIN is pulled to ground on startup, the Things will use the initial
//    password to build an AP. (E.g. in case of lost password)
#define CONFIG_PIN D2

// -- Status indicator pin.
//    First it will light up (kept LOW), on Wifi connection it will blink,
//    when connected to the Wifi it will turn off (kept HIGH).
#define STATUS_PIN LED_BUILTIN

String JsonDataScale;

String Website, data, XML, Javascript;

String status2, status;

DNSServer dnsServer;

WebServer server(80);

HTTPUpdateServer httpUpdater;

String enrollID;

IoTWebConf iotWebConf(thingName, &dnsServer, &server, wifInitialApPassword, CONFIG_VERSION);

// On Leonardo/Micro or others with hardware serial, use those! #0 is green wire, #1 is white
// uncomment this line:
// #define mySerial Serial1

```

```
// For UNO and others without hardware serial, we must use software serial...
// pin #2 is IN from sensor (GREEN wire)
// pin #3 is OUT from arduino (WHITE wire)
// comment these two lines if using hardware serial
SoftwareSerial mySerial(D3, D4);
```

```
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
uint16_t id;
int cnt = 0;
String sendID;
```

```
// uint16_t readnumber(void)
// {
//   uint16_t num = 0;
```

```
//   while (num == 0)
//   {
//     while (!Serial.available())
//       ;
//     num = Serial.parseInt();
//   }
//   return num;
// }
```

```
void checkinternet()
{
```

```

if (WiFi.status() != WL_CONNECTED)
{
    u8g2.clearBuffer();           // clear the internal memory
    u8g2.setFont(u8g2_font_ncenB08_tr);    // choose a suitable font
    u8g2.drawStr(0, 30, "Waiting for Internet."); // write something to the internal memory
    u8g2.sendBuffer();

    u8g2.clearBuffer();           // clear the internal memory
    u8g2.setFont(u8g2_font_ncenB08_tr);    // choose a suitable font
    u8g2.drawStr(0, 30, "Waiting for Internet..."); // write something to the internal memory
    u8g2.sendBuffer();
}
else
{
    String myip = WiFi.localIP().toString();
    u8g2.clearBuffer();           // clear the internal memory
    u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
    u8g2.drawStr(40, 30, "Ready !");
    u8g2.setFont(u8g2_font_5x7_tr);
    u8g2.drawStr(30, 50, "IP:");
    u8g2.drawStr(30, 50, myip.c_str());

    // write something to the internal memory // write something to the internal memory
    u8g2.sendBuffer();
}
}

void ipshow()
{

```

```

}

//-----enroll -----

uint16_t getFingerprintEnroll()
{
    cnt1 = 0;
    int p = -1;
    Serial.print("Waiting for valid finger to enroll as #");
    Serial.println(id);
    while (p != FINGERPRINT_OK)
    {
        p = finger.getImage();
        switch (p)
        {
            case FINGERPRINT_OK:
                Serial.println("Image taken");
                break;
            case FINGERPRINT_NOFINGER:
                cnt1++;
                Serial.println(".");
                Serial.println(cnt1);
                Serial.println("Scan now #1");

                u8g2.clearBuffer();           // clear the internal memory
                u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
                u8g2.drawStr(30, 30, "Scan Now #1"); // write something to the internal memory

```

```

u8g2.sendBuffer();

if (cnt1 == 30)
{
    Serial.println("Time out");
    status2 = "Time Out";

    u8g2.clearBuffer();          // clear the internal memory
    u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
    u8g2.drawStr(30, 30, "Time Out"); // write something to the internal memory
    u8g2.sendBuffer();
    delay(1500);

    u8g2.clearBuffer();          // clear the internal memory
    u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
    u8g2.drawStr(30, 30, "");    // write something to the internal memory
    u8g2.sendBuffer();

    return p;
}

break;

case FINGERPRINT_PACKETRECEIVEERR:
    Serial.println("Communication error");
    break;

case FINGERPRINT_IMAGEFAIL:
    Serial.println("Imaging error");
    break;

```

```
default:
    Serial.println("Unknown error");
    break;
}
}

// OK success!

p = finger.image2Tz(1);
switch (p)
{
case FINGERPRINT_OK:
    Serial.println("Image converted");
    break;
case FINGERPRINT_IMAGEMESS:
    Serial.println("Image too messy");
    return p;
case FINGERPRINT_PACKETRECIEVEERR:
    Serial.println("Communication error");
    return p;
case FINGERPRINT_FEATUREFAIL:
    Serial.println("Could not find fingerprint features");
    return p;
case FINGERPRINT_INVALIDIMAGE:
    Serial.println("Could not find fingerprint features");
    return p;
default:
```

```
Serial.println("Unknown error");  
return p;  
}  
  
Serial.println("Remove finger");  
delay(2000);  
p = 0;  
while (p != FINGERPRINT_NOFINGER)  
{  
    p = finger.getImage();  
}  
Serial.print("ID ");  
Serial.println(id);  
p = -1;  
Serial.println("Place same finger again");  
while (p != FINGERPRINT_OK)  
{  
    p = finger.getImage();  
    switch (p)  
    {  
        case FINGERPRINT_OK:  
            Serial.println("Image taken");  
            break;  
        case FINGERPRINT_NOFINGER:  
            Serial.print(".");  
            Serial.println("Scan now #2");  
            u8g2.clearBuffer();           // clear the internal memory
```

```

u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
u8g2.drawStr(30, 30, "Scan Now #2"); // write something to the internal memory
u8g2.sendBuffer();

break;
case FINGERPRINT_PACKETRECEIVEERR:
    Serial.println("Communication error");
    break;
case FINGERPRINT_IMAGEFAIL:
    Serial.println("Imaging error");
    break;
default:
    Serial.println("Unknown error");
    break;
}
}

// OK success!

p = finger.image2Tz(2);
switch (p)
{
case FINGERPRINT_OK:
    Serial.println("Image converted");
    break;
case FINGERPRINT_IMAGEMESS:
    Serial.println("Image too messy");

```



```

    return p;
case FINGERPRINT_PACKETRECEIVEERR:
    Serial.println("Communication error");
    return p;
case FINGERPRINT_FEATUREFAIL:
    Serial.println("Could not find fingerprint features");
    return p;
case FINGERPRINT_INVALIDIMAGE:
    Serial.println("Could not find fingerprint features");
    return p;
default:
    Serial.println("Unknown error");
    return p;
}

// OK converted!
Serial.print("Creating model for #");
Serial.println(id);

p = finger.createModel();
if (p == FINGERPRINT_OK)
{
    Serial.println("Prints matched!");
}
else if (p == FINGERPRINT_PACKETRECEIVEERR)
{
    Serial.println("Communication error");
}

```

```

    return p;
}
else if (p == FINGERPRINT_ENROLLMISMATCH)
{
    Serial.println("Fingerprints did not match");
    status2 = "Fingerprints did not match";
    u8g2.clearBuffer();          // clear the internal memory
    u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
    u8g2.drawStr(30, 30, "Error"); // write something to the internal memory
    u8g2.sendBuffer();
    delay(1500);
    u8g2.clearBuffer();          // clear the internal memory
    u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
    u8g2.drawStr(30, 30, "");    // write something to the internal memory
    u8g2.sendBuffer();

    cnt1 = 0;
    return p;
}
else
{
    Serial.println("Unknown error");
    return p;
}

Serial.print("ID ");
Serial.println(id);

```

```

p = finger.storeModel(id);
if (p == FINGERPRINT_OK)
{
    Serial.println("Stored!");
    status2 = "Success!";
    u8g2.clearBuffer();          // clear the internal memory
    u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
    u8g2.drawStr(30, 30, "Success!"); // write something to the internal memory
    u8g2.sendBuffer();
    delay(1500);
    u8g2.clearBuffer();          // clear the internal memory
    u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
    u8g2.drawStr(30, 30, "");    // write something to the internal memory
    u8g2.sendBuffer();

    cnt1 = 0;
}
else if (p == FINGERPRINT_PACKETRECEIVEERR)
{
    Serial.println("Communication error");
    return p;
}
else if (p == FINGERPRINT_BADLOCATION)
{
    Serial.println("Could not store in that location");
    return p;
}

```

```

else if (p == FINGERPRINT_FLASHERR)
{
    Serial.println("Error writing to flash");
    return p;
}
else
{
    Serial.println("Unknown error");
    return p;
}
}
//-----

// void printHex(int num, int precision)
// {
//   char tmp[16];
//   char format[128];

//   sprintf(format, "%%.%dX", precision);

//   sprintf(tmp, format, num);
//   Serial.print(tmp);
// }

// uint8_t downloadFingerprintTemplate(uint16_t id)
// {
//   Serial.println("-----");

```

```

// Serial.print("Attempting to load #");
// Serial.println(id);
// uint8_t p = finger.loadModel(id);
// switch (p)
// {
// case FINGERPRINT_OK:
//   Serial.print("Template ");
//   Serial.print(id);
//   Serial.println(" loaded");
//   break;
// case FINGERPRINT_PACKETRECEIVEERR:
//   Serial.println("Communication error");
//   return p;
// default:
//   Serial.print("Unknown error ");
//   Serial.println(p);
//   return p;
// }

// // OK success!

// Serial.print("Attempting to get #");
// Serial.println(id);
// p = finger.getModel();
// switch (p)
// {

```

```

// case FINGERPRINT_OK:
//   Serial.print("Template ");
//   Serial.print(id);
//   Serial.println(" transferring:");
//   break;
// default:
//   Serial.print("Unknown error ");
//   Serial.println(p);
//   return p;
// }

// // one data packet is 267 bytes. in one data packet, 11 bytes are 'usesless' :D
// uint8_t bytesReceived[534]; // 2 data packets
// memset(bytesReceived, 0xff, 534);

// uint32_t starttime = millis();
// int i = 0;
// while (i < 534 && (millis() - starttime) < 20000)
// {
//   if (mySerial.available())
//   {
//     bytesReceived[i++] = mySerial.read();
//   }
// }

// Serial.print(i);
// Serial.println(" bytes read.");
// Serial.println("Decoding packet...");

```

```

// uint8_t fingerTemplate[512]; // the real template
// memset(fingerTemplate, 0xff, 512);

// // filtering only the data packets
// int uindx = 9, index = 0;
// while (index < 534)
// {
//   while (index < uindx)
//     ++index;
//   uindx += 256;
//   while (index < uindx)
//   {
//     fingerTemplate[index++] = bytesReceived[index];
//   }
//   uindx += 2;
//   while (index < uindx)
//     ++index;
//   uindx = index + 9;
// }
// for (int i = 0; i < 512; ++i)
// {
//   //Serial.print("0x");
//   printHex(fingerTemplate[i], 2);
//   //Serial.print(", ");
// }
// Serial.println("\ndone.");

```

```

// /*
// uint8_t templateBuffer[256];
// memset(templateBuffer, 0xff, 256); //zero out template buffer
// int index=0;
// uint32_t starttime = millis();
// while ((index < 256) && ((millis() - starttime) < 1000))
// {
//   if (mySerial.available())
//   {
//     templateBuffer[index] = mySerial.read();
//     index++;
//   }
// }

// Serial.print(index); Serial.println(" bytes read");

// //dump entire templateBuffer. This prints out 16 lines of 16 bytes
// for (int count= 0; count < 16; count++)
// {
//   for (int i = 0; i < 16; i++)
//   {
//     Serial.print("0x");
//     Serial.print(templateBuffer[count*16+i], HEX);
//     Serial.print(" ");
//   }
//   Serial.println();

```



```
// */
```

```
//}
```

```
//Read fingerprints-----
```

```
uint8_t getFingerprintID()
```

```
{
```

```
    uint8_t p = finger.getImage();
```

```
    switch (p)
```

```
    {
```

```
    case FINGERPRINT_OK:
```

```
        Serial.println("Image taken");
```

```
        break;
```

```
    case FINGERPRINT_NOFINGER:
```

```
        Serial.println("No finger detected");
```

```
        return p;
```

```
    case FINGERPRINT_PACKETRECEIVEERR:
```

```
        Serial.println("Communication error");
```

```
        return p;
```

```
    case FINGERPRINT_IMAGEFAIL:
```

```
        Serial.println("Imaging error");
```

```
        return p;
```

```
    default:
```

```
        Serial.println("Unknown error");
```

```
        return p;
```

```
}
```

```
// OK success!

p = finger.image2Tz();
switch (p)
{
case FINGERPRINT_OK:
    Serial.println("Image converted");
    break;
case FINGERPRINT_IMAGEMESS:
    Serial.println("Image too messy");
    return p;
case FINGERPRINT_PACKETRECIEVEERR:
    Serial.println("Communication error");
    return p;
case FINGERPRINT_FEATUREFAIL:
    Serial.println("Could not find fingerprint features");
    return p;
case FINGERPRINT_INVALIDIMAGE:
    Serial.println("Could not find fingerprint features");
    return p;
default:
    Serial.println("Unknown error");
    return p;
}

// OK converted!
p = finger.fingerFastSearch();
```

```
if (p == FINGERPRINT_OK)
{
    Serial.println("Found a print match!");
}
else if (p == FINGERPRINT_PACKETRECEIVEERR)
{
    Serial.println("Communication error");
    return p;
}
else if (p == FINGERPRINT_NOTFOUND)
{
    Serial.println("Did not find a match");
    return p;
}
else
{
    Serial.println("Unknown error");
    return p;
}

// found a match!
Serial.print("Found ID #");
Serial.print(finger.fingerID);
Serial.print(" with confidence of ");
Serial.println(finger.confidence);

return finger.fingerID;
```

```

}

// returns -1 if failed, otherwise returns ID #
int getFingerprintIDez()
{
    uint16_t p = finger.getImage();
    if (p != FINGERPRINT_OK)
        return -1;

    p = finger.image2Tz();
    if (p != FINGERPRINT_OK)
        return -1;

    p = finger.fingerFastSearch();
    if (p != FINGERPRINT_OK)
        return -1;

    // found a match!
    Serial.print("Found ID #");
    Serial.print(finger.fingerID);
    Serial.print(" with confidence of ");
    Serial.println(finger.confidence);
    return finger.fingerID;
}

uint8_t deleteFingerprint(uint8_t id)
{

```

```
uint16_t p = -1;

p = finger.deleteModel(id);

if (p == FINGERPRINT_OK)
{
    Serial.println("Deleted!");
}
else if (p == FINGERPRINT_PACKETRECEIVEERR)
{
    Serial.println("Communication error");
    return p;
}
else if (p == FINGERPRINT_BADLOCATION)
{
    Serial.println("Could not delete in that location");
    return p;
}
else if (p == FINGERPRINT_FLASHERR)
{
    Serial.println("Error writing to flash");
    return p;
}
else
{
    Serial.print("Unknown error: 0x");
    Serial.println(p, HEX);
}
```

```
        return p;
    }
}

void del1()
{

    Serial.println("Please type in the ID # (from 1 to 127) you want to delete...");
    uint8_t id;

    if (id == 0) { // ID #0 not allowed, try again!
        return;
    }

    Serial.print("Deleting ID #");
    Serial.println(id);

    deleteFingerprint(id);
    status2 = "Delete Success";
}

void read1()
{

    Serial.println("Ready to read a fingerprint!");

    FingerID = -1;
    //id = -1;
```

```
// do
// {

// cnt++;

// id = getFingerprintIDez();
// Serial.println("Scan NOW");
// Serial.println(cnt);
// } while (cnt == 80);
//break;

// while (id == )
// {
// FingerID=getFingerprintIDez();
// cnt++;
// Serial.println(cnt);

// }

while (FingerID == -1)
{
    cnt++;

    FingerID = getFingerprintIDez();
    Serial.println("Scan Now");
    Serial.println(cnt);
```

```

u8g2.clearBuffer();          // clear the internal memory
u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
u8g2.drawStr(30, 30, "Scan Now"); // write something to the internal memory
u8g2.sendBuffer();

u8g2.clearBuffer();          // clear the internal memory
u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
u8g2.drawStr(30, 30, "Scan Now."); // write something to the internal memory
u8g2.sendBuffer();

u8g2.clearBuffer();          // clear the internal memory
u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
u8g2.drawStr(30, 30, "Scan Now..."); // write something to the internal memory
u8g2.sendBuffer();          // transfer internal memory to the display

if (cnt == 30)
{
    break;
}

} //-----๑๗๖

// DynamicJsonDocument root(100);

// root["Scale"] = String(FingerID);

// serializeJson(root, JsonDatascall);
// Serial.println(JsonDatascall);

```



```
// } while (cnt == 65535);
// if (id == 65535)
// {
//   cnt = 0;
//   Serial.println("Time out");
// }
cnt = 0;
sendID = "";
```

```
DynamicJsonDocument root(100);
root["id"] = String(FingerID);
serializeJson(root, sendID);
Serial.println(sendID);
```

```
if (FingerID == -1)
{
  timeout1 = "";
  timeout = "Time Out";
  DynamicJsonDocument root(100);
  root["id"] = String(timeout);
  serializeJson(root, timeout1);
  server.sendHeader("access-control-allow-origin", "*");
  server.send(200, "text/html", String(timeout1));
  Serial.print("Read ID => Time OUT");
  u8g2.clearBuffer();           // clear the internal memory
  u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
  u8g2.drawStr(30, 30, "Time Out"); // write something to the internal memory
```

```

u8g2.sendBuffer();

delay(1000);

u8g2.clearBuffer();          // clear the internal memory
u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
u8g2.drawStr(30, 30, "");    // write something to the internal memory
u8g2.sendBuffer();
}

else
{
  server.sendHeader("access-control-allow-origin", "*");
  server.send(200, "text/html", String(sendID));
  u8g2.clearBuffer();          // clear the internal memory
  u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
  u8g2.drawStr(50, 30, "OK !"); // write something to the internal memory

  //u8g2.drawStr(60, 30, String(FingerID).c_str()); // write something to the internal memory
  u8g2.sendBuffer();
  delay(2000);

  u8g2.clearBuffer();          // clear the internal memory
  u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
  u8g2.drawStr(30, 30, "");    // write something to the internal memory
  u8g2.sendBuffer();
}

//++;

```

```

//Serial.println("Ready to Read a fingerprint!");
// id = getFingerprintIDez(); // check the sensors // wait for sensors to stabilize
// getFingerprintIDez();
// getFingerprintIDez();

// }
// id = finger.fingerID;

// do
// {
//   cnt++;
//   getFingerprintIDez();
//   Serial.println(id);
//   Serial.println(cnt);

//   if (cnt == 80)
//   {
//     cnt = 0;
//     break;
//   }

// } while (id == 65535);
// Serial.println("okpass");
}

void wirte1()
{

```

```

Serial.println("StartRegister ");

getFingerprintEnroll();
}

void javascriptContent()
{
  Javascript = "<SCRIPT>\n";
  Javascript += "var xmlHttp=createXmlHttpRequest();\n";
  Javascript += "function createXmlHttpRequest(){\n";
  Javascript += "if(window.XMLHttpRequest){\n";
  Javascript += "xmlHttp=new XMLHttpRequest();\n";
  Javascript += "}else{\n";
  Javascript += "xmlHttp=new ActiveXObject('Microsoft.XMLHTTP');\n";
  Javascript += "}\n";
  Javascript += "return xmlHttp;\n";
  Javascript += "}\n";
  Javascript += "\n";
  Javascript += "function response(){\n";
  Javascript += "xmlResponse=xmlHttp.responseXML;\n";
  Javascript += "xmldoc = xmlResponse.getElementsByTagName('data');\n";
  Javascript += "message = xmldoc[0].firstChild.nodeValue;\n";
  Javascript += "document.getElementById('div1').innerHTML=message;\n";
  Javascript += "}\n";
  Javascript += "function process(){\n";
  Javascript += "xmlHttp.open('PUT','xml',true);\n";

```

```

Javascript += "xmlHttp.onreadystatechange=response;\n";
Javascript += "xmlHttp.send(null);\n";
Javascript += "setTimeout('process()',200);\n";
Javascript += "}\n";
//server.send(200, "text/html", JsonData);
Javascript += "</SCRIPT>\n";
}
void WebsiteContent()
{
    // javascriptContent();
    // Website = "Access-Control-Allow-Origin: *";
    // Website += "Access-Control-Allow-Methods: POST";
    // Website += JsonDataScale;

    // Website += "<style>\n";
    // Website += "#div1{\n";
    // Website += "width:500px;\n";
    // Website += "margin:0 auto;\n";
    // Website += "margin-top:130px;\n";
    // Website += "font-size:100%;\n";
    // Website += "color:#000100;\n";
    // Website += "}\n";
    // Website += "</style>\n";
    // Website += "<body onload='process()'>";
    // Website += "<div id='div1'>" + JsonDataScale + "</div></body></html>";
    // Website += Javascript;
    server.sendHeader("access-control-allow-origin", "");
}

```

```
server.send(200, "text/plain", "example");
}
```

```
void XMLcontent()
{
```

```
XML = "<?xml version='1.0'?>";
XML += "<data>";
XML += "JsonDatascall";
XML += "</data>";
server.setHeader("access-control-allow-origin", "*");
server.send(200, "text/xml", XML);
}
```

```
void handleRoot2()
```

```
{
// -- Let IoTWebConf test and handle captive portal requests.
if (IoTWebConf.handleCaptivePortal())
{
// -- Captive portal request were already served.
return;
}

String s = "<!DOCTYPE html><html lang='en'><head><meta name='viewport' content='width=device-width, initial-scale=1, user-scalable=no'>";

s += "<title>IoTWebConf 04 Update Server</title></head><body>Hello world!";
s += "Go to <a href='config'>configure page</a> to change values.";
s += "</body></html>\n";
```

```

server.setHeader("access-control-allow-origin", "*");
server.send(200, "text/html", s);
}
void handle_NotFound()
{
    server.send(404, "text/plain", "-----Not found-----");
}
void handleRoot()
{
    status = "";
    String message;

    if (server.arg("del").toInt())
    {
        id = server.arg("id").toInt();
        Serial.println(server.arg("del").toInt());
        del1();
    }

    if (server.arg("id").toInt())
    {
        id = server.arg("id").toInt();
        wirte1();
    }

    DynamicJsonDocument root(100);

```

```

    root["id"] = String(status2);
    serializeJson(root, status);
    server.sendHeader("access-control-allow-origin", "*");
    server.send(200, "text/html", status);
}

void setup()
{
    Serial.begin(115200);
    finger.begin(57600);
    u8g2.begin();
    timer.setInterval(300L, checkinternet);
    iotWebConf.setStatusPin(STATUS_PIN);
    iotWebConf.setConfigPin(CONFIG_PIN);
    iotWebConf.setupUpdateServer(&httpUpdater);
    iotWebConf.getApTimeoutParameter()->visible = true;
    // iotWebConf.setWifiConnectionTimeoutMs(&httpUpdater);
    // -- Initializing the configuration.
    iotWebConf.init();

    // u8g2.drawStr(60, 30, String(ip).c_str()); // write something to the internal memory
    // u8g2.sendBuffer();
    // delay(2000);
    // Serial.println(ip);

    // -- Set up required URL handlers on the web server.
    server.on("/iotweb", handleRoot2);
    server.on("/config", [] { iotWebConf.handleConfig(); });

```



```

server.onNotFound([]() { iotWebConf.handleNotFound(); });

server.on("/readID", read1);

server.on("/", handleRoot);

server.onNotFound(handle_NotFound);

Serial.println("Ready.");


u8g2.clearBuffer();           // clear the internal memory
u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font
u8g2.drawStr(30, 30, "WelCome!"); // write something to the internal memory
u8g2.sendBuffer();           // transfer internal memory to the display
delay(1000);


// set the data rate for the sensor serial port


// if (finger.verifyPassword())
// {
//   Serial.println("Found fingerprint sensor!");
// }
// else
// {
//   Serial.println("Did not find fingerprint sensor :(");
//   while (1)
//   {
//     delay(1);
//   }
// }

```

```
// finger.getTemplateCount();  
// Serial.print("Sensor contains ");  
// Serial.print(finger.templateCount);  
// Serial.println(" templates");  
// Serial.println("Waiting for valid finger...");  
}
```

```
void loop() // run over and over again
```

```
{  
  timer.run();  
  // getFingerprintIDez();  
  iotWebConf.doLoop();  
  server.handleClient();  
  // read1();  
  // getFingerprintIDez();  
}
```

```
//-----  
----
```

2. อุปกรณ์ที่ใช้ติดต่อสื่อสารระหว่างเครื่องขั้

```

/*****
*****

#Design and development factory of recycling management through the internet of thing

# Scale

#ComputerEngineering Faculty OF Engineering NakhonPhanomUniversity

#Pupan Phonkaew 593030710044

*****
*****/

#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
#include <SoftwareSerial.h>
#include <IoTWebConf.h>
#include <BlynkSimpleEsp8266.h>
#include <ArduinoJson.h>

// #include <BlynkSimpleEsp8266.h>

BlynkTimer timer;

#include "index.h" //Our HTML webpage contents with javascripts

#define CONFIG_VERSION "dem1"

// -- When CONFIG_PIN is pulled to ground on startup, the Thing will use the initial
//    password to build an AP. (E.g. in case of lost password)

// #define CONFIG_PIN D2

// -- Status indicator pin.

//    First it will light up (kept LOW), on Wifi connection it will blink,

```

```

//  when connected to the Wifi it will turn off (kept HIGH).
//#define STATUS_PIN LED_BUILTIN
const char thingName[] = "scale";
const char wifInitialApPassword[] = "12345678";

DNSServer dnsServer;
WebServer server(80);
HTTPUpdateServer httpUpdater;
String enrollID;
IotWebConf iotWebConf(thingName, &dnsServer, &server, wifInitialApPassword, CONFIG_VERSION);
SoftwareSerial swSer(D1, D2);
String rs232input1;
String Website, data, Javascript, XML;
// BlynkTimer timer;

#define LED 2 //On board LED
int RGBPin[] = { D5, D6, D7 };
//=====
// This routine is executed when you open its IP in browser
//=====

void SetRGB(int r, int g, int b) {
  analogWrite(RGBPin[2], r);
  analogWrite(RGBPin[1], g);
  analogWrite(RGBPin[0], b);
}

void checkinternet()

```

```

{

  if (WiFi.status() != WL_CONNECTED)
  {
    SetRGB(255, 0, 0); // Red
  }
  else
  {
    SetRGB(0, 0, 255); // Green
  }
}

void WebsiteContent()
{
  if (data == "")
  {
    Serial.println("Unable to contact the scale");

    data = "Unable to contact the scale";
  }
  else
  {
  }

  server.sendHeader("access-control-allow-origin", "*");
  server.send(200, "text/html", data);
}

void handleRoot2()

```

```

{
    // -- Let lotWebConf test and handle captive portal requests.
    if (iotWebConf.handleCaptivePortal())
    {
        // -- Captive portal request were already served.
        return;
    }

    String s = "<!DOCTYPE html><html lang=\"en\"><head><meta name=\"viewport\" content=\"width=device-width, initial-scale=1, user-scalable=no\"/>";

    s += "<title>lotWebConf 04 Update Server</title></head><body>Hello world!";
    s += "Go to <a href='config'>configure page</a> to change values.";
    s += "</body></html>\n";

    server.send(200, "text/html", s);
}

void handleRoot()
{
    String s = MAIN_page;      //Read HTML contents
    server.send(200, "text/html", s); //Send web page
}

void handleADC()
{
    String a = rs232input1;

    String adcValue = a;

    digitalWrite(LED, !digitalRead(LED)); //Toggle LED on data request ajax

```

```

server.send(200, "text/plain", adcValue); //Send ADC value only to client ajax request
}

void handle_NotFound()
{
server.send(404, "text/plain", "404 Not found.");
}

//=====
//      SETUP
//=====

void setup()
{
Serial.begin(115200);
swSer.begin(9600);

//Connect to your WiFi router
Serial.println("");
pinMode(RGBPin[0], OUTPUT);
pinMode(RGBPin[1], OUTPUT);
pinMode(RGBPin[2], OUTPUT);
//Onboard LED port Direction output
pinMode(LED, OUTPUT);
timer.setInterval(300L, checkinternet);
iotWebConf.setupUpdateServer(&httpUpdater);
iotWebConf.getApTimeoutParameter()->visible = true;
// iotWebConf.setWifiConnectionTimeoutMs(&httpUpdater);
// -- Initializing the configuration.
iotWebConf.init();

```

```

//timer.setInterval(300L, WebsiteContent);

Serial.println("\nSoftware serial test started");

server.on("/getscale", WebsiteContent);

// server.on("/", );

server.on("/config", [] { iotWebConf.handleConfig(); });


server.onNotFound(handle_NotFound);

Serial.println("HTTP server started");


server.on("/", handleRoot);    //Which routine to handle at root location. This is display page
server.on("/readADC", handleADC); //This page is called by java Script AJAX


server.begin(); //Start server

Serial.println("HTTP server started");


for (char ch = ' '; ch <= 'z'; ch++)
{ //send serially a to z on software serial
  swSer.write(ch);
}

swSer.println("");
}

//=====
//          LOOP
//=====


void loop()
{

```



```
timer.run();

iotWebConf.doLoop();

server.handleClient();

data = "";

if (swSer.available() > 0)
{
    rs232input1 = swSer.readStringUntil('\n');
    //Serial.println(rs232input1);
    int strindex = rs232input1.indexOf('=');
    int endindex = rs232input1.indexOf(' ');
    rs232input1 = rs232input1.substring(strindex + 1, endindex);
    DynamicJsonDocument root(100);
    root["scale"] = String(rs232input1);
    serializeJson(root, data);
    // Serial.println(data);
}
```

3. อุปกรณ์บันทึกภาพแบบออนไลน์

```
#*****
```

```
*****
```

```
#Design and development factory of recycling management through the internet  
of thing
```

```
# Scale
```

```
#ComputerEngineering Faculty OF Engineering NakhonPhanomUniversity
```

```
#Pupan Phonkaew 593030710044
```

```
#*****
```

```
*****
```

```
from pyimagesearch.motion_detection import SingleMotionDetector
```

```
from imutils.video import VideoStream
```

```
from flask import Response
```

```
from flask import Flask

from flask import render_template

import threading

import argparse

import datetime

import imutils

import time

import cv2

import os

import datetime

from pathlib import Path

import arrow


# filePath = r"/home/pupan/Documents/Savecamera"


# criticalTime = arrow.now().shift(hours=+7).shift(days=-1)


# for item in Path(filePath).glob('*'):

#     if item.is_file():

#         print (str(item.absolute()))
```

```

#     itemTime = arrow.get(item.stat().st_mtime)

#         # if itemTime < criticalTime:

#             #     #remove it

#             # pass


# for filename in os.listdir(path):

#     print(filename)

#     # if os.stat(os.path.join(path, filename)).st_mtime < now - 7 * 86400:

#     if os.path.getmtime(os.path.join(path, filename)) < now - 7 * 86400:

#         if os.path.isfile(os.path.join(path, filename)):

#             #print(filename)

#             os.remove(os.path.join(path, filename))


# initialize the output frame and a lock used to ensure thread-safe

# exchanges of the output frames (useful for multiple browsers/tabs

# are viewing the stream)

```

```
outputFrame = None
```

```
lock = threading.Lock()
```

```
# initialize a flask object
```

```
app = Flask(__name__)
```

```
# initialize the video stream and allow the camera sensor to
```

```
# warmup
```

```
#vs = VideoStream(usePiCamera=1).start()
```

```
vs = VideoStream(src=0).start()
```

```
time.sleep(2.0)
```

```
def delete1():
```

```
    path = r"/home/pupan/Documents/Savecamera"
```

```
    now = time.time()
```

```
    items = os.listdir("/home/pupan/Documents/Savecamera/")
```

```
        newlist = []

        for names in items:

            if names.endswith(".png"):

                os.remove(path+'/'+names)


@app.route("/del")

def delete():

    # return the response generated along with the specific media

    # type (mime type)

    return Response(delete1(),

                    mimetype = "multipart/x-mixed-replace; boundary=frame")


@app.route("/camera")

def index():

    # return the rendered template

    return render_template("index.html")

def generate1():
```

```

        cv2.imwrite('/home/pupan/Documents/Savecamera/'+str(datetime.datetime.now())+'.png', outputFrame)

@app.route("/cap")
def capture():

    # return the response generated along with the specific media

    # type (mime type)

    return Response(generate1(),

                    mimetype = "multipart/x-mixed-replace; boundary=frame")

def detect_motion(frameCount):

    # grab global references to the video stream, output frame, and

    # lock variables

    global vs, outputFrame, lock

    # initialize the motion detector and the total number of frames

    # read thus far

    md = SingleMotionDetector(accumWeight=0.1)

    total = 0

    # loop over frames from the video stream

```

```

while True:

    # read the next frame from the video stream, resize it,

    # convert the frame to grayscale, and blur it

    frame = vs.read()

    frame = imutils.resize(frame, width=1000)

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    gray = cv2.GaussianBlur(gray, (7, 7), 0)


    # grab the current timestamp and draw it on the frame

    timestamp = datetime.datetime.now()

    cv2.putText(frame, timestamp.strftime(

        "%A %d %B %Y %l:%M:%S%p"), (10, frame.shape[0] - 10),

        cv2.FONT_HERSHEY_SIMPLEX, 0.35, (0, 0, 255), 1)


    #cv2.imwrite('/home/pupan/Documents/'+str(datetime.datetime.now())+'.png',
    outputFrame)


    # if the total number of frames has reached a sufficient

```



```
# number to construct a reasonable background model, then
```

```
# continue to process the frame
```

```
if total > frameCount:
```

```
    # detect motion in the image
```

```
    motion = md.detect(gray)
```

```
    # check to see if motion was found in the frame
```

```
    if motion is not None:
```

```
        # unpack the tuple and draw the box surrounding the
```

```
        # "motion area" on the output frame
```

```
        (thresh, (minX, minY, maxX, maxY)) = motion
```

```
        cv2.rectangle(frame, (minX, minY), (maxX, maxY),
```

```
                        (0, 0, 255), 2)
```

```
# update the background model and increment the total number
```

```
# of frames read thus far
```

```
md.update(gray)
```

```
total += 1
```

```
# acquire the lock, set the output frame, and release the
# lock
with lock:
    outputFrame = frame.copy()

def generate():
    # grab global references to the output frame and lock variables
    global outputFrame, lock

    # loop over frames from the output stream
    while True:
        # wait until the lock is acquired
        with lock:
            # check if the output frame is available, otherwise skip
            # the iteration of the loop
            if outputFrame is None:
                continue

            # encode the frame in JPEG format
```

```

(flag, encodedImage) = cv2.imencode(".jpg", outputFrame)

# ensure the frame was successfully encoded
if not flag:
    continue

# yield the output frame in the byte format
yield(b'--frame\r\n' b'Content-Type: image/jpeg\r\n\r\n' +
      bytearray(encodedImage) + b'\r\n')

@app.route("/video_feed")
def video_feed():
    # return the response generated along with the specific media
    # type (mime type)
    return Response(generate(),
                    mimetype = "multipart/x-mixed-replace; boundary=frame")

# check to see if this is the main thread of execution
if __name__ == '__main__':

```

```
# construct the argument parser and parse command line arguments

ap = argparse.ArgumentParser()

ap.add_argument("-i", "--ip", type=str, required=True,

                help="ip address of the device")

ap.add_argument("-o", "--port", type=int, required=True,

                help="ephemeral port number of the server (1024 to 65535)")

ap.add_argument("-f", "--frame-count", type=int, default=32,

                help="# of frames used to construct the background model")

args = vars(ap.parse_args())


# start a thread that will perform motion detection

t = threading.Thread(target=detect_motion, args=(

    args["frame_count"],))

t.daemon = True

t.start()


# start the flask app

app.run(host=args["ip"], port=args["port"], debug=True,

        threaded=True, use_reloader=False)
```

release the video stream pointer

vs.stop()

บรรณานุกรม

- Fingerscan (2019). Fingerprint Recognition สืบค้นเมื่อ 3 สิงหาคม 2562,
 จาก: <http://fingerscan.in.th/fingerprint-scanner/83-fingerprint-recognition>
- Digitalscalesblog (2019). Install the driver for the USB to RS-232 สืบค้นเมื่อ 3 สิงหาคม 2562,
 จาก: <https://www.digitalscalesblog.com/tag/rs-232/>
- Pimylifeup (2019). Build a Raspberry Pi Webcam Server สืบค้นเมื่อ 5 สิงหาคม 2562,
 จาก: <https://pimylifeup.com/raspberry-pi-webcam-server/>
- AB (2019). การเก็บข้อมูลจาก Nodemcu ESP8266 ลงใน ฐานข้อมูล
 Database Mysql (Arduino todatabase) สืบค้นเมื่อ 5 สิงหาคม 2562,
 จาก: bit.ly/2TjjAsS
- Pyimagesearch (2019), Install OpenCV 4 on your Raspberry Pi, สืบค้นเมื่อ 7 กันยายน 2562.
 จาก: <https://www.pyimagesearch.com/2018/09/26/install-opencv-4-on-your-raspberry-pi/>
- Goragod (2018). Ajax และ WebSocket เลือกใช้อะไรดี ความแตกต่างที่สำคัญระหว่าง Ajax และ
 WebSocket มีอะไรบ้าง เมื่อทำการเชื่อมต่อกับ Sever มีวิธีการอย่างไร. สืบค้นเมื่อ 9 พฤศจิกายน
 2562, จาก: https://www.goragod.com/knowledge/ajax_และ_websocket_เลือก_ใช้_อะไรดี.html

ประวัติผู้เขียน



ชื่อ	นายภูพาน ผลแก้ว
วันเกิด	เกิดเมื่อวันเสาร์ที่ 22 มีนาคม พ.ศ. 2540
สถานที่อยู่ปัจจุบัน	17 หมู่ 11 ต.เรณูใต้ อ.เรณูนคร จ.นครพนม 48170
ประวัติการศึกษา	
2551	สำเร็จการศึกษาประถมศึกษาปีที่ 6 โรงเรียนบ้านบ่อสะอาด
2554	สำเร็จการศึกษามัธยมศึกษาตอนต้น โรงเรียนเรณูนครวิทยานุกูล
2559	สำเร็จการศึกษาประกาศนียบัตรวิชาชีพ วิทยาลัยธาตุพนม มหาวิทยาลัยนครพนม
2559	เข้าศึกษาคณะวิศวกรรมศาสตร์สาขาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยนครพนม