



เครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลี้ยงขนาดเล็ก

FEED ME PLEASE

นางสาวไอรดา พฤษภา

นางสาวสายธาร งามขำ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี

พ.ศ. 2561

เครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลี้ยงขนาดเล็ก

นางสาวไอรดา พฤษภา

นางสาวสายธาร งามคำ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี

พ.ศ. 2561

FEED ME PLEASE

MISS IRADA PRUETSAPA

MISS SAITAN NGAMKAM

THIS PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE BACHELOR DEGREE OF ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING

FACULTY OF ENGINEERING

RAJAMENGALA UNIVERSITY OF TECHNOLOGY THANYABURI

YEAR 2018

หัวข้อปริญญานิพนธ์	เครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลี้ยงขนาดเล็ก
นักศึกษา	นางสาวไอรดา พฤษภา นางสาวสายธาร งามคำ
อาจารย์ที่ปรึกษา	อาจารย์มานิช ประชา

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
อนุมัติให้ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

.....หัวหน้าภาควิชาฯ
(อาจารย์มานิช ประชา)

คณะกรรมการสอบปริญญานิพนธ์

.....ประธานกรรมการ
(อาจารย์เจษฎา อรุณฤกษ์)

.....กรรมการ
(อาจารย์พัฒนพงศ์ สุนันทพจน์)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.พศยน นินทนาวงศา)

.....กรรมการและอาจารย์ที่ปรึกษา
(อาจารย์มานิช ประชา)

ลิขสิทธิ์ของภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี

หัวข้อปริญญานิพนธ์	เครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลี้ยงขนาดเล็ก		
นักศึกษา	นางสาวไอรดา พฤษภา	รหัส	115710400160-4
	นางสาวสายธาร งามขำ	รหัส	115710400943-3
อาจารย์ที่ปรึกษา	อาจารย์มานิช ประชา		
ปีการศึกษา	2561		

บทคัดย่อ

วัตถุประสงค์ของโครงการเครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลี้ยงขนาดเล็ก เพื่อช่วยอำนวยความสะดวกแก่ผู้มีสัตว์เลี้ยง ซึ่งอาจเกิดกรณีผู้เลี้ยงไม่สะดวกให้อาหารด้วยตนเอง เช่น อยู่นอกบ้าน หรือจำเป็นต้องปล่อยสัตว์เลี้ยงไว้ที่บ้านโดยไม่มีผู้ดูแล อุปกรณ์ที่ควบคุมระบบการทำงานของเครื่องให้อาหาร คือ โหนดเอ็มซียู (NodeMCU) ทำหน้าที่เชื่อมต่อกับเซิร์ฟเวอร์ (Server) ผ่านพีเอชพี (PHP) เพื่อบันทึกข้อมูล และดึงข้อมูลจากฐานข้อมูล (Database) โดยทำงานร่วมกับอาร์ดิวโน ซึ่งทำหน้าที่ส่งผ่านคำสั่งทำงานจากโหนดเอ็มซียู ให้กับอุปกรณ์ต่อร่วม และรับข้อมูลจากอุปกรณ์เพื่อส่งให้โหนดเอ็มซียู ในส่วนของโมบายแอปพลิเคชัน (Mobile Application) นั้น มีระบบบันทึกข้อมูลสัตว์เลี้ยง บันทึกรูปแบบการให้อาหารสัตว์เลี้ยง ซึ่งจะมีรูปแบบการให้ตามเวลา และการให้เมื่ออาหารในถาดเหลือปริมาณตามที่กำหนด แสดงปริมาณในถาดแบบเรียลไทม์ (Real time) สัตว์เลี้ยงที่อยู่ใกล้เครื่องให้อาหาร สถิติปริมาณการกินอาหารของสัตว์เลี้ยง จะมีการแจ้งเตือน วันนัดหมายต่าง ๆ ที่เกี่ยวข้องกับสัตว์เลี้ยง และในกรณีอาหารในถาดใกล้หมด

ผลการดำเนินงานโครงการเครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลี้ยงขนาดเล็ก สามารถทำงานได้ตามที่คาดหวัง โดยเครื่องให้อาหารสามารถให้อาหารในปริมาณที่กำหนดได้ใกล้เคียงปริมาณจริง ซึ่งผู้ใช้งานสามารถใช้เครื่องให้อาหารได้ผ่านโมบายแอปพลิเคชัน และยังสามารถดูข้อมูลตามข้างต้นได้อีกด้วย

คำสำคัญ เครื่องให้อาหาร โมบายแอปพลิเคชัน ฐานข้อมูล

กิตติกรรมประกาศ

โครงการ “เครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลี้ยงขนาดเล็ก” นี้สำเร็จตามความคาดหวังของคณะผู้จัดทำได้ เนื่องจากได้รับความกรุณาจากอาจารย์มานะ ประชา อาจารย์ที่ปรึกษาโครงการที่ได้ให้คำแนะนำ แนวคิด ตลอดจนการแก้ไขข้อบกพร่องต่าง ๆ มาโดยตลอด รวมทั้งท่านอาจารย์ในภาควิชาวิศวกรรมคอมพิวเตอร์ที่คอยช่วยเหลือ และชี้แนะแนวทางในการดำเนินงานอันเป็นประโยชน์ อีกทั้งบุคลากรในภาควิชาวิศวกรรมคอมพิวเตอร์ที่อำนวยความสะดวกเรื่องอุปกรณ์ในการจัดทำชิ้นงาน จนโครงการนี้เสร็จสมบูรณ์

สุดท้ายนี้ขอกราบขอบพระคุณบิดา มารดาของคณะผู้จัดทำที่สนับสนุนส่งเสริมการเรียนรู้ คอยให้คำปรึกษาในเรื่องต่าง ๆ รวมทั้งเป็นกำลังใจที่ดีเสมอมา ขอกราบขอบพระคุณครูบาอาจารย์ทุกท่านที่เคยอบรมสั่งสอนมาตั้งแต่แรกเริ่มจนถึงปัจจุบัน และมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี จนทำให้คณะผู้จัดทำโครงการได้มีวันนี้

คณะผู้จัดทำ

สารบัญ

	หน้า
บทคัดย่อ	ง
กิตติกรรมประกาศ	จ
สารบัญ	ฉ
สารบัญตาราง	ซ
สารบัญรูป	ณ
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขต	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	4
2.1 งานวิจัยที่เกี่ยวข้อง	4
2.2 RFID	4
2.3 Load cell	8
2.4 Arduino Mega 2560 R3	9
2.5 NodeMCU ESP8266	12
2.6 RC Servo Motor	13
2.7 Mobile Application	16
บทที่ 3 วิธีการดำเนินงาน	18
3.1 แผนการดำเนินงาน	18
3.2 การออกแบบ / เครื่องมือ	19
3.3 Flowchart Diagram	20
3.4 Sequence Diagram	23
3.5 การออกแบบหน้าจอบริการแอปพลิเคชัน	32
3.6 Entity Relationship	46
3.7 Data Dictionary	46
3.8 การออกแบบอุปกรณ์	52
3.9 วัสดุ / อุปกรณ์	64

สารบัญ (ต่อ)

	หน้า
บทที่ 4 ผลการดำเนินงานและการวิเคราะห์	66
4.1 ผลการประกอบเครื่องให้อาหาร	66
4.2 ผลการทดสอบการใช้งานโมบายแอปพลิเคชัน	67
4.3 ผลการทดสอบเครื่องให้อาหาร	77
4.4 ผลการทดสอบรันโปรแกรมบนไมโครคอนโทรลเลอร์	82
บทที่ 5 สรุปผลและข้อเสนอแนะ	88
5.1 สรุปผลการดำเนินงาน	88
5.2 ปัญหาและอุปสรรค	88
5.3 ข้อเสนอแนะ	89
บรรณานุกรม	90
ภาคผนวก ก	92
ขั้นตอนการติดตั้งโมบายแอปพลิเคชัน	93
ภาคผนวก ข	94
คู่มือการใช้งานโมบายแอปพลิเคชัน	95
ภาคผนวก ค	96
Source code	97
ประวัติผู้ทำปริญญานิพนธ์	98

สารบัญตาราง

ตารางที่		หน้า
3.1	แผนการดำเนินงาน	18
3.2	ตาราง pet	46
3.3	ตาราง select_type	47
3.4	ตาราง feed_type1	48
3.5	ตาราง feed_type2	49
3.6	ตาราง food_alert	49
3.7	ตาราง date_alert	50
3.8	ตาราง food_tray	51
3.9	ตาราง tray	51
3.10	รายละเอียดของโครงสร้างเครื่องให้อาหาร	54
3.11	การต่อขาสัญญาณวงจร I2C	57
3.12	การต่อขาสัญญาณวงจรตรวจวัดน้ำหนัก	58
3.13	การต่อขาสัญญาณวงจรควบคุมเซอร์โวมอเตอร์ปล่อยอาหาร	60
3.14	การต่อขาสัญญาณวงจรควบคุมอัลตราโซนิกตรวจวัดระยะ	61
3.15	การต่อขาสัญญาณวงจรควบคุมเครื่องอ่านอาร์เอฟไอดี	62
3.16	การต่อขาสัญญาณวงจรควบคุมการเล่นเสียง	64
3.17	รายการวัสดุ และอุปกรณ์	64
4.1	ผลการบันทึกข้อมูลปริมาณอาหารที่ปล่อย	77
4.2	ผลการบันทึกข้อมูลปล่อยอาหารตามเวลาที่กำหนด	77
4.3	ผลการบันทึกข้อมูลปล่อยอาหารตามปริมาณอาหารในถาดที่กำหนด	78
4.4	ผลการบันทึกข้อมูลปริมาณอาหารในถาดเข้าสู่ฐานข้อมูล	79
4.5	ผลการบันทึกรหัสประจำตัวสัตว์เลี้ยงจากแทคประจำตัวสัตว์เลี้ยงเข้าสู่ฐานข้อมูล	80
4.6	ผลการบันทึกข้อมูลการแจ้งเตือนอาหารในถังใกล้หมดเข้าสู่ระบบ	81

สารบัญรูป

รูปที่		หน้า
2.1	การทำงานของระบบ RFID	5
2.2	RFID Tag รูปแบบต่าง ๆ	6
2.3	RFID reader และ RFID Tag	7
2.4	โหนดเซลล์แบบสเตรนเกจ (Strain Gauge Load cell)	8
2.5	Arduino Mega 2560	9
2.6	NodeMCU ESP8266	13
2.7	ส่วนประกอบภายนอกของอาร์ชีเซอร์ไวโมเตอร์	14
2.8	ส่วนประกอบภายในของอาร์ชีเซอร์ไวโมเตอร์	14
2.9	เซอร์ไวโมเตอร์บล็อกไดอะแกรม	15
2.10	องศาการหมุนของอาร์ชีเซอร์ไวโมเตอร์	16
2.11	ความกว้างพัลส์เมื่อกำหนดให้หมุน 45 องศา	16
3.1	ภาพรวมระบบ	19
3.2	Flowchart Diagram การทำงานของเครื่องให้อาหาร	20
3.3	Flowchart Diagram การทำงานของเครื่องให้อาหาร (ส่วนที่ 1 ต่อ)	21
3.4	Flowchart Diagram การตรวจสอบสัต์ว์เลี้ยงใกล้เครื่อง และปริมาณอาหารในถาดแบบเรียลไทม์	22
3.5	Sequence Diagram แสดงข้อมูลปริมาณอาหารในถาดอาหาร	23
3.6	Sequence Diagram แสดงรายชื่อสัตว์เลี้ยงใกล้บริเวณเครื่องให้อาหาร	23
3.7	Sequence Diagram แสดงรายชื่อสัตว์เลี้ยง	24
3.8	Sequence Diagram แสดงข้อมูลสัตว์เลี้ยง	25
3.9	Sequence Diagram บันทึกข้อมูลสัตว์เลี้ยง	25
3.10	Sequence Diagram แก้ไขข้อมูลสัตว์เลี้ยง	26
3.11	Sequence Diagram ลบข้อมูลสัตว์เลี้ยง	27
3.12	Sequence Diagram บันทึก และแสดงข้อมูลตารางนัดหมาย	27
3.13	Sequence Diagram แก้ไขข้อมูลตารางนัดหมาย	28
3.14	Sequence Diagram ลบข้อมูลตารางนัดหมาย	29
3.15	Sequence Diagram บันทึกรูปแบบการปล่อยอาหาร	29
3.16	Sequence Diagram บันทึกการปล่อยอาหาร	30

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.17	Sequence Diagram แสดงข้อมูลสถิติ	31
3.18	Mobile Application Sitemaps	32
3.19	ออกแบบหน้าจอในส่วนของหน้า Home	33
3.20	ออกแบบหน้าจอในส่วนเมนูหลักของระบบ	34
3.21	ออกแบบหน้าจอในส่วนของหน้า Pet	35
3.22	ออกแบบหน้าจอในส่วนของหน้าเพิ่มสัตว์เลี้ยง	36
3.23	ออกแบบหน้าจอในส่วนของการแสดงข้อมูลสัตว์เลี้ยง	37
3.24	ออกแบบหน้าจอในส่วนของหน้าแก้ไขข้อมูลสัตว์เลี้ยง	38
3.25	ออกแบบหน้าจอในส่วนของหน้า Appointment	39
3.26	ออกแบบหน้าจอในส่วนของหน้าเพิ่มการนัดหมาย	40
3.27	ออกแบบหน้าจอในส่วนของหน้า Stat	41
3.28	ออกแบบหน้าจอในส่วนของหน้าแสดงข้อมูลสถิติ	42
3.29	ออกแบบหน้าจอในส่วนของหน้า Food	43
3.30	ออกแบบหน้าจอในส่วนของหน้ารูปแบบที่ 1	44
3.31	ออกแบบหน้าจอในส่วนของหน้ารูปแบบที่ 2	45
3.32	Entity Relationship	46
3.33	โครงสร้างเครื่องให้อาหาร ด้านหน้า	52
3.34	โครงสร้างเครื่องให้อาหาร ด้านข้าง	53
3.35	Hardware architecture ภาพรวมอุปกรณ์	55
3.36	ภาพรวมการประกอบวงจรเครื่องให้อาหาร	56
3.37	วงจรรับ-ส่งข้อมูลระหว่าง Master (NodeMCU) และ Slave (Arduino Mega) แบบ I2C	56
3.38	การประกอบวงจรรับ-ส่งข้อมูลแบบ I2C	57
3.39	วงจรตรวจวัดน้ำหนัก	58
3.40	การประกอบวงจรตรวจวัดน้ำหนัก	58
3.41	วงจรควบคุมเซอร์โวมอเตอร์ปล่อยอาหาร	59
3.42	การประกอบวงจรควบคุมเซอร์โวมอเตอร์ปล่อยอาหาร	59
3.43	วงจรควบคุมอัลตราโซนิกตรวจวัดระยะ	60

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.44	การประกอบวงจรควบคุมอัลตราโซนิกตรวจวัดระยะ	60
3.45	วงจรควบคุมเครื่องอ่านอาร์เอฟไอดี	61
3.46	การประกอบวงจรควบคุมเครื่องอ่านอาร์เอฟไอดี	62
3.47	วงจรควบคุมการเล่นเสียง	63
3.48	การประกอบวงจรควบคุมการเล่นเสียง	63
4.1	ภาพรวมเครื่องให้อาหาร	66
4.2	หน้าแรกเมื่อเปิดเข้าใช้งานระบบ	67
4.3	หน้าเมนูหลัก	68
4.4	หน้าหลักของระบบ	68
4.5	หน้าแสดงรายชื่อสัตว์เลี้ยง	69
4.6	หน้าเพิ่มสัตว์เลี้ยง	69
4.7	หน้าข้อมูลสัตว์เลี้ยง	70
4.8	หน้าแก้ไขข้อมูลสัตว์เลี้ยง	70
4.9	หน้าแสดงตารางการนัดหมาย	71
4.10	หน้าเพิ่มการนัดหมาย	71
4.11	แก้ไขแล้วลบข้อมูลการนัดหมาย	72
4.12	หน้าเมนูรูปแบบสถิติ	72
4.13	หน้าแสดงข้อมูลสถิติรายวัน	73
4.14	หน้าแสดงข้อมูลรายอาทิตย์	73
4.15	หน้าแสดงข้อมูลสถิติรายเดือน	74
4.16	หน้าแสดงข้อมูลรูปแบบการปล่อยอาหาร	74
4.17	หน้าการปล่อยอาหารรูปแบบที่ 1	75
4.18	หน้าเมนูการปล่อยอาหารรูปแบบที่ 2	75
4.19	หน้าการปล่อยอาหารรูปแบบที่ 2	76
4.20	หน้าแจ้งเตือน	76
4.21	มอนิเตอร์เมื่อไมโครคอนโทรลเลอร์เริ่มทำงาน	82
4.22	มอนิเตอร์เมื่อมีการส่งค่าน้ำหนักในถาดและรหัสประจำตัวสัตว์เลี้ยง	83
4.23	มอนิเตอร์เมื่อมีการส่งข้อมูลไปบันทึกยังฐานข้อมูล	84

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.24	มอนิเตอร์เมื่อมีการให้อาหารรูปแบบที่ 1	85
4.25	มอนิเตอร์เมื่อมีการให้อาหารรูปแบบที่ 2	86
4.26	มอนิเตอร์เมื่อมีการส่งคำสั่งเล่นเสียงและข้อมูลระยะห่างในถึงบรรจอาหาร	87

ฉบับร่าง

บทที่ 1

บทนำ

เนื่องจากสัตว์เลี้ยงมีบทบาทช่วยในการเสริมสร้างความมั่นคงทางอารมณ์ให้กับผู้เลี้ยง สัตว์เลี้ยงจึงเป็น "เพื่อน" ที่จำเป็นมากสำหรับคนบางคน สัตว์เลี้ยงเป็นเพื่อนในยามเหงา เพราะว่าพวกมันมีความรักให้กับเราอย่างเต็มที่ การเลี้ยงสัตว์จะทำให้เรามีพัฒนาการในการเรียนรู้มากขึ้น ไม่ว่าจะเป็นเด็ก หรือผู้ใหญ่ สัตว์เลี้ยงถือเป็นนักบำบัดให้กับเราเป็นอย่างดี ทำให้เราลดความเครียด โดยเฉพาะตอนที่เรากลับจากทำงานมาเหนื่อย ๆ หรือตอนที่เรามีความเครียด ซึ่งการที่เราแบ่งเวลาไปเล่นกับสัตว์เลี้ยงของตัวเองจะช่วยลดอาการความกดดัน และความเหนื่อยล้าได้ ทั้งยังช่วยกระตุ้นให้เรามีความกระฉับกระเฉงมากขึ้นด้วย

1.1 ความเป็นมาและความสำคัญ

จากปัญหาที่บางครั้งผู้เลี้ยงอาจจะจะมีเหตุให้ไม่สามารถอยู่กับสัตว์เลี้ยงตลอดเวลาได้ เช่น ไปทำงาน กลับบ้านต่างจังหวัด หรือเดินทางไปต่างประเทศ เป็นต้น จึงทำให้ผู้เลี้ยงเกิดความกังวลที่จะต้องปล่อยให้ผู้เลี้ยงอยู่เพียงลำพัง

ทางคณะผู้จัดทำได้มองเห็นถึงปัญหาทางด้านโภชนาการของสัตว์เลี้ยงเมื่อผู้เลี้ยงไม่อยู่ จึงคิดประดิษฐ์สิ่งประดิษฐ์ ที่จะช่วยเพิ่มความสะดวกในการที่ผู้เลี้ยงไม่จำเป็นต้องคอยให้อาหารสัตว์เลี้ยงทุกวัน แต่เป็นการเพิ่มเติมอาหารเป็นครั้งคราวในเครื่องให้อาหาร โดยทำงานผ่านระบบโปรแกรมประยุกต์บนโทรศัพท์เคลื่อนที่ การประดิษฐ์สิ่งประดิษฐ์นี้เพื่อให้สัตว์เลี้ยงไม่เหงาเวลาที่ผู้เลี้ยงไม่อยู่ และยังลดความกังวลเมื่อไม่มีคนดูแลเรื่องโภชนาการของสัตว์เลี้ยงได้

1.2 วัตถุประสงค์

- 1.2.1 เพื่อให้อาหารสัตว์เลี้ยงได้เมื่อไม่อยู่บ้าน
- 1.2.2 เพื่อกำหนดปริมาณอาหารของสัตว์เลี้ยง
- 1.2.3 เพื่อการแสดงผลการวิเคราะห์ข้อมูลผ่านแอปพลิเคชัน
- 1.2.4 เพื่อควบคุมการทำงานผ่านแอปพลิเคชัน
- 1.2.5 เพื่อให้สัตว์เลี้ยงสนใจเครื่องให้อาหารเมื่อถึงเวลาให้อาหาร
- 1.2.6 เพื่อทราบปริมาณการกินอาหารของสัตว์เลี้ยง

1.3 ขอบเขต

1.3.1 ข้อมูลทางเทคนิคของเครื่องให้อาหารเม็ดสัตว์เลี้ยงขนาดเล็ก

- 1) ใช้บอร์ดไมโครคอนโทรลเลอร์สำหรับควบคุมการทำงานทั้งระบบ
- 2) ใช้เซนเซอร์ตรวจจับน้ำหนักโหลดเซลล์ ในการตรวจจับน้ำหนักของอาหารในถาด
- 3) ใช้อาร์เอฟไอดี (RFID: Radio Frequency Identification) เพื่อระบุถึงสัตว์เลี้ยงตัว

ใดที่อยู่ใกล้บริเวณตัวเครื่องในระยะที่กำหนด

- 4) สามารถแจ้งเตือนปริมาณอาหารในถังบรรจุได้ เมื่อปริมาณอาหารลดลงในระดับที่กำหนดไว้

1.3.2 ข้อมูลการทำงานของโปรแกรมการให้อาหารสัตว์เลี้ยงขนาดเล็ก

- 1) มีการทำงานของระบบผ่านทางโปรแกรมประยุกต์บนโทรศัพท์เคลื่อนที่
- 2) ฐานข้อมูลของระบบจัดเก็บในระบบคลาวด์เซิร์ฟเวอร์
- 3) สามารถกำหนดข้อมูลของสัตว์เลี้ยง ต่อไปนี้เป็นอย่างน้อย
 - แทคไอดี (Tag ID) ของอาร์เอฟไอดี
 - ชื่อสัตว์เลี้ยง
 - ประเภทของสัตว์เลี้ยง
 - ช่วงเกิดของสัตว์เลี้ยงแบบ วัน-เดือน-ปี หรือ เดือน-ปี
 - โรคประจำตัว
 - วันฉีดวัคซีน
 - วันพบสัตวแพทย์
- 4) การให้อาหาร และปริมาณอาหารของระบบ ทำได้ 2 รูปแบบ และตั้งค่าได้ 5 ชุด

ข้อมูล

- แบบตั้งค่าช่วงปริมาณอาหาร (เมื่ออาหารในถาดเหลือปริมาณตามที่กำหนด)
- แบบตั้งค่าเวลา และปริมาณอาหารรายวัน

5) เสียงบันทึก

• สามารถบันทึกเสียงพูดได้ 3 แทรค ความยาวสูงสุด 10 วินาที จะเล่นเสียงทุกครั้งแบบตั้งค่า หรือสุมเสียงเมื่อใช้โหมดการให้อาหารแบบตั้งค่าเวลา และปริมาณอาหารรายวัน

• สามารถบันทึกเสียงเรียก และเสียงไล่ได้ อย่างละ 1 แทรค ต่อ 1 ตัว ความยาวสูงสุด 10 วินาที

6) มีระบบเรียกสัตว์เลี้ยงอัตโนมัติเมื่อสัตว์เลี้ยงไม่มากินอาหาร 2 มื้อขึ้นไป โดยแจ้งเตือนผ่านแอปพลิเคชัน และเล่นเสียงเรียกสัตว์เลี้ยงตัวนั้น ๆ

- 7) สามารถแสดงข้อมูลสัตว์เลี้ยงในบริเวณเครื่องให้อาหารได้

8) สามารถส่งเสียงที่ตั้งค่าไว้ให้สัตว์เลี้ยงได้

9) สามารถแสดงปริมาณอาหารคงเหลือแบบเรียลไทม์

10) สามารถรายงานข้อมูลปริมาณการกินอาหารของสัตว์เลี้ยงแต่ละตัว หรือรวมทั้งหมด
แบบรายวัน รายสัปดาห์ และรายเดือน ได้ 2 รูปแบบ

- แบบตารางตัวเลข
- แบบกราฟแท่ง

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 ผู้เลี้ยงมีความสะดวกในการให้อาหาร และมีเวลาว่างมากขึ้น ไม่จำเป็นต้องให้อาหาร
สัตว์เลี้ยงด้วยตัวเองทุกครั้ง เพียงแค่เติมอาหารในถังบรรจุเป็นครั้งคราว

1.4.2 ช่วยควบคุมปริมาณอาหารที่สัตว์เลี้ยงกิน

1.4.3 สัตว์เลี้ยงรู้จักกินอาหารเป็นเวลา

1.4.4 ผู้เลี้ยงไม่ต้องห่วงเรื่องให้อาหารสัตว์เลี้ยงเมื่อผู้เลี้ยงไม่อยู่บ้าน

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

การจัดทำโครงการนี้ได้พิจารณา ค้นคว้าทฤษฎี หลักการ แนวคิด และงานวิจัยที่เกี่ยวข้อง เพื่อนำมาประยุกต์ใช้ พัฒนาระบบเครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลี้ยงขนาดเล็ก คณะผู้จัดทำได้ทำการศึกษาหัวข้อ ดังต่อไปนี้

2.1 งานวิจัยที่เกี่ยวข้อง

2.1.1 ระบบเครื่องให้อาหารเม็ดอัตโนมัติ [1]

ระบบเครื่องให้อาหารเม็ดอัตโนมัติ พัฒนาโดย เกียรติศักดิ์ อยู่ดี เนื้อหาที่เกี่ยวข้องกับระบบเครื่องให้อาหารเม็ดอัตโนมัติ การให้อาหารแก่สัตว์เลี้ยง นับเป็นปัจจัยหนึ่งในการทางธุรกิจสัตว์เลี้ยงเพื่อการส่งออกในประเทศและต่างประเทศซึ่ง การให้อาหารสัตว์ต้องใช้ แรงงานคนจำนวนมาก อันเป็นภาระของผู้ประกอบการในการจัดหาแรงงานและค่าใช้จ่าย รวมทั้งผู้ที่ชอบเลี้ยงสัตว์ เช่น ปลา สวงาม สุนัข แมว อีกด้วย

เครื่องให้อาหารสัตว์ประกอบไปด้วยส่วนของมอเตอร์ไฟฟ้ากระแสตรง เป็นมอเตอร์ที่ต้องใช้ไฟฟ้ากระแสตรงผ่านเข้าไปในขดลวดอาร์เมเจอร์ เพื่อทำให้เกิดการดูด และผลักกันของแม่เหล็กถาวรกับแม่เหล็กไฟฟ้าที่เกิดจากขดลวด มอเตอร์จึงหมุนได้ หม้อแปลงที่ใช้ในการส่งผ่านพลังงาน จากวงจรไฟฟ้าหนึ่งไปยังอีก วงจรโดยอาศัยหลักการของแม่เหล็กไฟฟ้าโดยปกติจะใช้เชื่อมโยงระหว่างระบบไฟฟ้าแรงสูง และไฟฟ้าแรงต่ำ มีการใช้สวิตช์และอุปกรณ์ควบคุม (Switch Controller) ทำหน้าที่ควบคุมการเปลี่ยนแปลงขนาดของแรงเคลื่อนไฟฟ้า และมีอุปกรณ์ป้องกันไฟฟ้าชนิดต่าง ๆ มี Timer เป็นอุปกรณ์ในการตั้งเวลาเปิด-ปิดของโครงการ โดยใช้วงจรควบคุมมอเตอร์ PWM 12/24 V วงจรนี้เป็นวงจรที่ใช้ไอซีไมโครคอนโทรลเลอร์เข้ามาควบคุมการทำงาน และมีระบบ Soft Start

2.2 RFID [2]

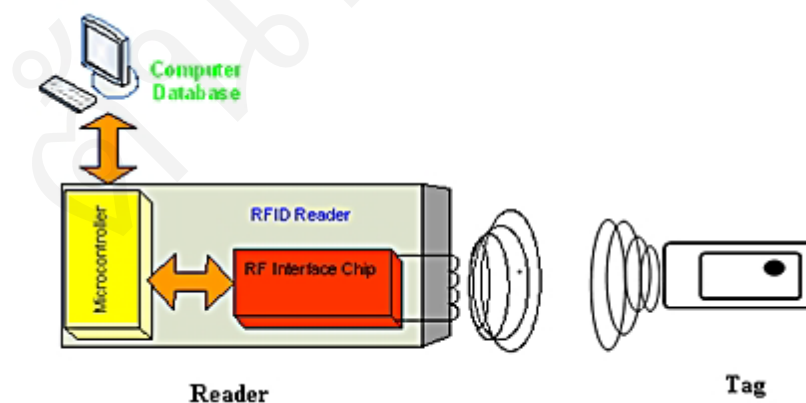
อาร์เอฟไอดี (RFID) ย่อมาจาก Radio Frequency Identification เป็นระบบฉลากที่ได้ถูกพัฒนา ตั้งแต่ปี ค.ศ. 1980 ในเวลานั้นทำหน้าที่เป็นเครื่องมือดักจับสัญญาณ ไม่ได้ทำหน้าที่เป็นตัวระบุเอกลักษณ์อย่างที่ใช้กันอยู่ในปัจจุบัน แต่อาร์เอฟไอดีปัจจุบันมีลักษณะเป็นป้ายอิเล็กทรอนิกส์ (RFID Tag) ที่สามารถอ่านค่าโดยผ่านคลื่นวิทยุจากระยะห่าง เพื่อตรวจสอบติดตาม และบันทึกข้อมูลที่ติดอยู่กับป้าย ซึ่งนำไปฝังไว้ข้างใน หรือติดอยู่กับวัตถุต่าง ๆ เช่น ผลิตภัณฑ์ กล่อง หรือสิ่งของใด ๆ สามารถติดตามข้อมูลของวัตถุ 1 ชิ้นว่า คืออะไร ผลิตที่ไหน ใครเป็นผู้ผลิต ผลิตอย่างไร ผลิตวันไหน และเมื่อไหร่ ประกอบไปด้วยชิ้นส่วนที่ขึ้น และแต่ละชิ้นมาจากที่ไหน รวมทั้งตำแหน่ง

ที่ตั้งของวัตถุนั้น ๆ ในปัจจุบันว่าอยู่ส่วนใดในโลก โดยไม่จำเป็นต้องอาศัยการสัมผัส (Contact-Less) หรือต้องเห็นวัตถุนั้น ๆ ก่อนทำงานโดยใช้เครื่องอ่านที่สื่อสารกับป้ายด้วยคลื่นวิทยุในการอ่าน และเขียนข้อมูล

2.2.1 ประเภทของอาร์เอฟไอดี

คลื่นวิทยุที่ใช้ในเทคโนโลยีอาร์เอฟไอดี มีความยาวคลื่นระหว่าง 0.1 ซม. ถึง 1,000 กม. จะอยู่ในช่วงความถี่ระหว่าง 30 เฮิรท์ส และ 300 กิกะเฮิรท์ส จากช่วงความถี่ดังกล่าวทำให้สามารถแบ่งคลื่นวิทยุได้เป็น 4 ประเภท ได้แก่

- 1) ความถี่ต่ำ (Low Frequency: LF) ในเทคโนโลยีอาร์เอฟไอดีจะใช้คลื่น 125 กิโลเฮิรท์ส ถึง 134 กิโลเฮิรท์ส
- 2) ความถี่สูง (High Frequency: HF) ความถี่นี้จะอยู่ในช่วง 3 เมกะเฮิรท์ส ถึง 30 เมกะเฮิรท์ส ความถี่ 13.56 เมกะเฮิรท์ส
- 3) ความถี่สูงยิ่ง (Ultra-High Frequency: UHF) ความถี่นี้จะอยู่ในช่วง 300 เมกะเฮิรท์ส ถึง 1 กิกะเฮิรท์ส
- 4) คลื่นความถี่ไมโครเวฟ (Microwave Frequency) ความถี่นี้คือความถี่ที่สูงกว่า 1 กิกะเฮิรท์ส ขึ้นไป



รูปที่ 2.1 การทำงานของระบบ RFID [2]

2.2.2 องค์ประกอบในระบบอาร์เอฟไอดี

ลักษณะการทำงานของระบบอาร์เอฟไอดี หลักสำคัญของเทคโนโลยีอาร์เอฟไอดี ได้แก่ อินเลย์ (Inlay) ที่บรรจุอุปกรณ์และวงจรอิเล็กทรอนิกส์กับโลหะที่ยึดหยุ่นได้สำหรับการติดตามหรือทำหน้าที่เป็นเสาอากาศนั่นเอง อินเลย์มีความหนาสูงสุดอยู่ที่ 0.375 มิลลิเมตร สามารถทำเป็น

แผ่นบางอัดเป็นชั้น ๆ ระหว่างกระดาษ, แผ่นฟิล์ม หรือพลาสติกก็ได้ ซึ่งเป็นการผลิตเครื่องหมายหรือฉลาก จึงทำให้ง่ายต่อการติดเป็นป้ายชื่อ หรือฉลากของชิ้นงานหรือวัตถุนั้น ๆ ได้สะดวก

องค์ประกอบในระบบอาร์เอฟไอดี จะมีหลัก ๆ อยู่ 2 ส่วนด้วยกัน ส่วนแรกคือ ฉลากหรือป้ายขนาดเล็กที่จะถูกผนึกอยู่กับวัตถุที่เราสนใจ โดยฉลากนี้จะทำการบันทึกข้อมูลเกี่ยวกับวัตถุชิ้นนั้นๆ เอาไว้ ฉลากดังกล่าวมีชื่อเรียกว่า ทรานสปอนเดอร์ (Transponder, Transmitter & Responder) หรือที่เรียกกันโดยทั่วไปว่า แทค (Tag) ส่วนที่สอง เป็นอุปกรณ์สำหรับอ่าน หรือเขียนข้อมูลภายในแทค มีชื่อเรียกว่า ทรานสซิฟเวอร์ (Transceiver, Transmitter & Receiver) หรือที่เรียกกันโดยทั่วไปว่า เครื่องอ่าน (Reader)



รูปที่ 2.2 RFID Tag รูปแบบต่าง ๆ [2]

1) อาร์เอฟไอดีแทค [3] เป็นส่วนประกอบหลักของระบบอาร์เอฟไอดี อาร์เอฟไอดีแทคเป็นอุปกรณ์ที่ใช้เก็บข้อมูล และส่งข้อมูลไปให้เครื่องอ่านโดยผ่านคลื่นวิทยุ อาร์เอฟไอดีแทคสามารถแบ่งเป็นประเภทต่าง ๆ ได้ดังต่อไปนี้

- พาสซีฟอาร์เอฟไอดีแทค (Passive RFID Tag) ไม่มีแหล่งพลังงานในตัวเอง ในการส่งข้อมูลนั้น อาร์เอฟไอดีแทค ประเภทนี้จะอาศัยพลังงานจากเครื่องอ่าน เพื่อให้ตนเองมีพลังงานในการส่งข้อมูลกลับไปให้กับเครื่องอ่าน
- แอคทีฟอาร์เอฟไอดีแทค (Active RFID Tag) จะมีแหล่งพลังงาน หรือแบตเตอรี่เป็นของตนเอง โดยส่วนใหญ่แอคทีฟจะมีอายุการทำงานประมาณ 2 ถึง 7 ปี ขึ้นอยู่กับประเภทของ

แบตเตอรี่ ในการติดต่อกันระหว่างแทค กับเครื่องอ่านสำหรับอาร์เอฟไอดีแทคประเภทนี้ แทคจะเป็นส่วนที่เริ่มการติดต่อก่อน ประเภทนี้มีแหล่งพลังงานของตนเอง ทำให้ส่งข้อมูลได้ในระยะไกล

- อาร์เอฟไอดีแทคแบบ Semi-active/Semi-passive เป็นอาร์เอฟไอดีแทคที่มีแหล่งพลังงานเป็นของตนเอง ทำหน้าที่เหมือนแบตเตอรี่ในแอคทีฟแทคในการส่งข้อมูลนั้น อาร์เอฟไอดีแทคประเภทนี้จะอาศัยพลังงานจากเครื่องอ่าน แต่เมื่อแหล่งพลังงานหมดแทคประเภทนี้จะทำงานในลักษณะเหมือนพาสซีฟแทค ต่างจากแอคทีฟแทคที่เมื่อแหล่งพลังงานหมดจะไม่สามารถทำงานได้



รูปที่ 2.3 RFID reader และ RFID Tag [3]

2) เครื่องอ่านอาร์เอฟไอดี เป็นอุปกรณ์ที่ใช้ในการอ่านและเขียนข้อมูลลงในอาร์เอฟไอดีแทค ในการเขียนข้อมูลนั้นเป็นการเชื่อมโยงอาร์เอฟไอดีแทคกับสิ่งใดสิ่งหนึ่ง การสื่อสารกันระหว่างเครื่องอ่านกับอาร์เอฟไอดีแทคนั้นขึ้นอยู่กับประเภทของแทคลักษณะการสื่อสารข้อมูลระหว่างแทคกับเครื่องอ่านมี 3 ลักษณะคือ มอดูเลตแบคสแกตเตอร์ (Modulated backscatter), ทรานซ์มิเตอร์ไทป์ (Transmitter type) และ ทรานซ์พอนเดอร์ไทป์ (Transponder type)

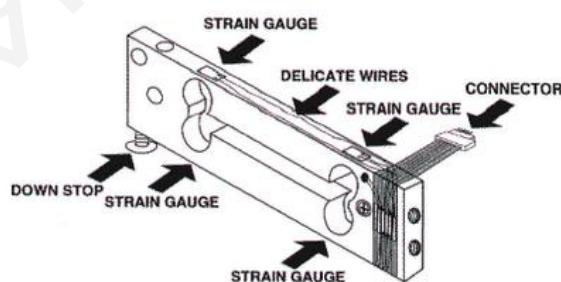
- มอดูเลตแบคสแกตเตอร์ การสื่อสารลักษณะนี้ เครื่องอ่านจะส่งคลื่นวิทยุในลักษณะต่อเนื่อง (Continuous wave) ซึ่งจะส่งออกมาในลักษณะกระแสวิกฤต ผ่านเสาอากาศที่อยู่ในอาร์เอฟไอดีแทค เมื่ออาร์เอฟไอดีแทคได้รับกระแสจากเครื่องอ่าน เสาอากาศก็จะส่งพลังงานให้กับไมโครชิปที่อยู่ในแทค เพื่อให้แทคมีกำลังไฟในการทำงาน หลังจากนั้นอาร์เอฟไอดีแทคก็จะทำการส่งข้อมูลกลับไปให้แก่เครื่องอ่าน เพื่อทำงานต่อไป

- ทรานซ์มิเตอร์ การสื่อสารลักษณะนี้จะใช้กับแอคทีฟแทคเท่านั้น การสื่อสารในลักษณะนี้ แทคจะส่งข้อมูลเป็นช่วงเวลาที่กำหนดไว้ โดยไม่สนว่ามีเครื่องอ่านอยู่หรือไม่ ดังนั้นการสื่อสารแบบนี้แทคจะเป็นอุปกรณ์ที่เริ่มการสื่อสารก่อนเสมอ

- ทรานซ์พอนเดอร์ การสื่อสารแบบนี้แทคจะไม่ทำงาน หรืออยู่ในโหมดหลับ (sleep) เมื่อไม่มีการติดต่อสื่อสารกับเครื่องอ่าน ในช่วงที่แทคอยู่ในโหมดหลับ อาจจะส่งข้อมูลออกมาเป็นระยะเพื่อตรวจสอบว่า มีเครื่องอ่านอยู่ในบริเวณดังกล่าวหรือไม่ เมื่อเครื่องอ่านได้รับสัญญาณดังกล่าว เครื่องอ่านก็จะส่งคำสั่งไปปลุก (wake up) ให้แทคทำงาน เมื่อแทคได้รับสัญญาณนี้จากเครื่องอ่าน ก็จะเริ่มทำการส่งข้อมูล ในการสื่อสารแบบนี้แทคส่งข้อมูลเมื่อได้รับสัญญาณจากเครื่องอ่าน เท่านั้น เสาอากาศของเครื่องอ่านอาร์เอฟไอดี เครื่องอ่านติดต่อสื่อสารกับแทคโดยผ่านทางเสาอากาศของเครื่องอ่าน ซึ่งอาจจะเป็นอุปกรณ์ที่แยกออกจากเครื่องอ่าน หรือเป็นลักษณะที่รวมเข้ากับเครื่องอ่านเป็นอุปกรณ์เดียวกัน ขอบข่ายของเสาอากาศจะเป็นตัวกำหนดอาณาเขตการอ่าน (Read Zone) เมื่อแทคเข้ามาอยู่ในบริเวณเขตการอ่าน กระบวนการติดต่อสื่อสารระหว่างเครื่องอ่านกับแทคก็จะเริ่มทำงาน

2.3 Load cell [4]

โหลดเซลล์ (Load cell) คือ เซนเซอร์ที่สามารถแปลงค่าแรงกด หรือแรงดึง เป็นสัญญาณทางไฟฟ้าได้ เหมาะสำหรับการทดสอบคุณสมบัติทางกลของชิ้นงาน (Mechanical Properties of Parts) โหลดเซลล์ถูกนำไปใช้ในอุตสาหกรรมหลากหลายประเภท ได้แก่ การชั่งน้ำหนัก การทดสอบแรงกดของชิ้นงาน การทดสอบความแข็งแรงของชิ้นงาน การทดสอบการเข้ารูปชิ้นงาน (Press fit) ใช้สำหรับงานทางด้านวัสดุโลหะ ทดสอบโลหะ ชิ้นส่วนรถยนต์ วิศวกรรมโยธา ทดสอบคอนกรีต ทดสอบไม้ เป็นต้น



รูปที่ 2.4 โหลดเซลล์แบบสเตรนเกจ (Strain Gauge Load cell) [4]

2.3.1 ประเภทของ Load cell

1) โหลดเซลล์แบบสเตรนเกจ (Strain Gauge Load cell) หลักการของโหลดเซลล์ประเภทนี้ก็คือ เมื่อมีน้ำหนักมากระทำ ความเครียด (Strain) จะเปลี่ยนเป็นความต้านทานทางไฟฟ้า

ในสัดส่วนโดยตรงกับแรงที่มากระทำ ปกติแล้วมักจะใช้เกจวัดความเครียด 4 ตัว (วงจร Wheatstone Bridge Circuit) ในการวัดโดยเกจตัวด้านทานทั้งสี่จะเชื่อมต่อเข้าด้วยกันเพื่อใช้แปลงแรงที่กระทำกับตัวของมัน ไม่ว่าจะเป็นแรงกดหรือแรงดึงส่งสัญญาณออกมาเป็นแรงดันไฟฟ้า

2) โหลดเซลล์แบบไฮดรอลิก (Hydraulic Load Cell) [5] โหลดเซลล์ประเภทนี้อาศัยหลักการทำงานของแรงดันน้ำมัน เมื่อมีน้ำหนักมากระทำจะทำให้เกิดแรงดันภายในต้นน้ำมันเราจะเอาแรงที่วัดแรงดันจากเกจ นี้เพื่อเทียบออกมาเป็นน้ำหนักแรงกด โหลดเซลล์ชนิดนี้ใช้กับงานชั่งน้ำหนักชั่งถัง โหลดเซลล์ชนิดนี้ไม่นิยมใช้งานทั่ว ๆ ไปเนื่องจากมีความยุ่งยากในการติดตั้งและราคาค่อนข้างสูงมาก เหมาะกับการใช้งานในพื้นที่อันตรายเนื่องจากไม่ต้องใช้ไฟฟ้าในการทำงาน

3) นิวเมตริกโหลดเซลล์ (Pneumatic Load Cell) เป็นโหลดเซลล์ที่อาศัยหลักการทำงานของแรงดันลม โดยทั่วไปจะใช้กับการชั่งน้ำหนักแบบน้อย ๆ

2.4 Arduino Mega 2560 R3 [6]

Arduino (อ่านว่า อา-ตุ-อิ-โน้ หรือ อา-ดุย-โน้) เป็นบอร์ดไมโครคอนโทรเลอร์ตระกูลเอวีอาร์ (AVR) ที่มีการพัฒนาแบบโอเพนซอร์ส (Open Source) คือมีการเปิดเผยข้อมูลทั้งด้านฮาร์ดแวร์ และซอฟต์แวร์ ตัว บอร์ดอาร์ดุยโน้ ถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้ผู้ใช้งานยังสามารถดัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ด หรือโปรแกรมต่อได้อีกด้วย

ความง่ายของบอร์ดอาร์ดุยโน้ในการต่ออุปกรณ์เสริมต่าง ๆ คือผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิกส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ขาอินพุต/เอาต์พุตของบอร์ด หรือเพื่อความสะดวกสามารถเลือกต่อกับบอร์ดเสริม (Arduino Shield) ประเภทต่าง ๆ เช่น อาร์ดุยโน้มิวสิคชิลด์ (Arduino Music Shield), อาร์ดุยโน้รีเลย์ชิลด์ (Arduino Relay Shield), อาร์ดุยโน้ไวร์เลสชิลด์ (Arduino Wireless Shield), อาร์ดุยโน้จีพีเอสชิลด์ (Arduino GPRS Shield) เป็นต้น มาเปรียบกับบอร์ดอาร์ดุยโน้แล้วเขียนโปรแกรมพัฒนาต่อได้เลย



รูปที่ 2.5 Arduino Mega 2560 [7]

2.4.1 ภาพรวม [8]

อาร์ดุยโนเมกะ (Arduino Mega) เป็นบอร์ดไมโครคอนโทรลเลอร์ (Microcontroller board) ที่ใช้ชิพเอทีเมกะ 1280 (ATmega1280) ประกอบไปด้วย พินอินพุต/เอาต์พุต 54 พิน (14 พินสามารถใช้เป็น พัดดับเบิลยูเอ็มเอต์พุต (PWM output) ได้), แอนะล็อกอินพุต 16 พิน, ยูอาร์ท (UART: hardware serial ports) 4 พิน, คริสตัลออสซิลเลเตอร์ (crystal oscillator) 16 เมกะเฮิรตซ์, พอร์ตเชื่อมต่อยูเอสบี, ช่องเสียบแหล่งจ่าย (power jack), ไอซีเอสพีเฮดเดอร์ (ICSP header) คือ ส่วนที่เป็นเอวีอาร์ ขนาดเล็กสำหรับการโปรแกรมอาร์ดุยโน ซึ่งประกอบด้วย เอ็มไอเอสไอ (MOSI), เอ็มไปเอสไอ (MISO), เอสซีเค (SCK), รีเซ็ต, วีซีซี (VCC), กราวนด์ (GND), และปุ่มรีเซ็ต เมกะเชื่อมต่อกับคอมพิวเตอร์ได้ง่ายโดยใช้สายยูเอสบี หรือใช้แหล่งจ่ายไฟผ่านอะแดปเตอร์เอซีทูดีซี (AC-to-DC) หรือแบตเตอรี่ เพื่อเริ่มทำงาน

โดยบอร์ดนี้มีทุกสิ่งที่ไม่โครคอนโทรลเลอร์จำเป็นต้องใช้อย่างการต่อไฟเลี้ยงสามารถทำได้ทั้งการเชื่อมต่อเข้ากับสายยูเอสบี หรือใช้แหล่งจ่ายไฟผ่านอะแดปเตอร์เอซีทูดีซี (AC-to-DC adaptor) หรือการใช้แบตเตอรี่ ซึ่งเมกะเป็นบอร์ดที่เข้ากันได้กับชิลด์ (shield) ที่ออกแบบมาเพื่ออาร์ดุยโนดูมิลานอฟ (Arduino Duemilanove) หรือ ดิอีซีมิล่า (Diecimila)

เมกะ 2560 นี้มีความแตกต่างจากบอร์ดก่อนหน้านี้ตรงที่ไม่ใช้ เอฟทีดีไอ ยูเอสบีทูซีเรียล ไดรเวอร์ชิพ (FTDI USB-to-serial driver chip) แต่จะมีเอทีเมกะ 16U2 (ATmega16U2) เข้ามาเป็นโปรแกรมแปลงยูเอสบีทูซีเรียล

อาร์ดุยโนเมกะ 2560 รีวิชัน 2 (Arduino Mega2560 Revision 2) มีเอทีเมกะ 8U2 (ATmega8U2) ทำให้อัพเดทเฟิร์มแวร์ ผ่านยูเอสบีโปรโตคอลที่เรียกว่าดีเอฟยู (DFU: Device Firmware Update) ได้ง่ายขึ้น ส่วนอาร์ดุยโนเมกะรีวิชัน 3 (Arduino Mega Revision 3) มีฟีเจอร์ใหม่ ๆ เพิ่มขึ้นมาดังนี้

1) 1.0 พินเอาต์ เพิ่มเอสดีเอ (SDA) และเอสซีแอล (SCL) และอีกสองพินใหม่คือ IOREF เป็นพินที่ใช้ในการเชื่อมต่อกับชิลด์เพื่อแปลงเป็นแรงดันที่ได้จากบอร์ด ส่วนอีก 1 พินที่เหลือมีไว้สำหรับใช้ร่วมกับเอวีอาร์ในอนาคต

2) วงจรรีเซ็ตที่ดีขึ้น

3) ใช้ เอทีเมกะ 16U2 แทน 8U2

2.4.2 แหล่งพลังงาน [9]

อาร์ดุยโนเมกะสามารถเชื่อมรับพลังงานโดยการเชื่อมต่อไมโครยูเอสบี หรือจากพาวเวอร์ซัพพลายจากภายนอกได้ โดยแหล่งพลังงานจะถูกเลือกโดยอัตโนมัติ

แหล่งจ่ายจากภายนอกสามารถมาได้จาก AC-to-DC adapter หรือจากแบตเตอรี่ โดยต่อเข้ากับ 2.1 mm center-positive plug ไปยังช่องเสียบแหล่งจ่าย และการต่อเข้ากับแบตเตอรี่สามารถทำได้โดยการต่อเข้ากับกราวด์ และ Vin pin header ของ power connector

บอร์ดสามารถทำงานได้ในช่วงแรงดัน 6 ถึง 20 โวลต์ ถ้าแหล่งจ่ายมีค่าต่ำกว่า 7 โวลต์ อาจส่งผลให้พิน 5 โวลต์ มีแรงดันที่ต่ำกว่า 5 โวลต์ และบอร์ดอาจจะไม่เสถียร แต่ถ้าหากแรงดันมีค่าสูงกว่า 12 โวลต์ อาจส่งผลให้บอร์ดโอเวอร์ฮีท (overheat) และอาจทำให้เสียหายได้ ดังนั้นช่วงแรงดันที่เหมาะสมกับบอร์ดคือ 7 โวลต์ ถึง 12 โวลต์

- 1) VIN เป็นอินพุตโวลต์เทจของบอร์ดอาร์ดยูโนโดยใช้แหล่งจ่ายจากภายนอก
- 2) 5V เป็นเอาต์พุตที่ควบคุม 5 โวลต์จากบอร์ด
- 3) 3V3 เป็น 3.3 volt supply ที่สร้างขึ้นจากเรกกูเลเตอร์ (regulator) บนบอร์ด และให้กระแสได้สูงสุด 50 มิลลิแอมป์
- 4) GND เป็นกราวด์พิน
- 5) IOREF เป็นพินที่ให้โวลต์เทจเรฟเฟอเรนซ์ (voltage reference) กับไมโครคอนโทรลเลอร์ เพื่อเลือกค่าแรงดันให้กับชิปด์ที่มาเชื่อมต่อกับบอร์ด

2.4.3 หน่วยความจำ

เอทีเมกะ 2560 มีหน่วยความจำ 256 กิโลไบต์ (8 กิโลไบต์ ใช้สำหรับ บูทโหลดเดอร์ (bootloader)) นอกจากนี้ยังมีอีก 8 กิโลไบต์ สำหรับแอสแรม (SRAM) และ 4 กิโลไบต์ สำหรับอีอีพรอม (EEPROM)

2.4.4 อินพุต และเอาต์พุต

ในแต่ละดิจิทัลอลทั้ง 54 พินบนบอร์ดอาร์ดยูโนเมกะสามารถเป็นได้ทั้งอินพุต และเอาต์พุตโดยจะทำงานที่แรงดัน 5 โวลต์ และให้กระแสสูงสุด 40 มิลลิแอมป์

2.4.5 ฟังก์ชันอื่น ๆ เพิ่มเติม

- 1) Serial: 0 (Rx) และ 1(Tx), Serial 1: 19(Rx) และ 18 (Tx), Serial 2: 17 (Rx) และ 16(Tx), Serial 3:15 (Rx) และ 14 (Tx) ใช้สำหรับรับ (Rx) และส่ง (Tx) TTL serial data โดยพิน 0 และ 1 จะถูกเชื่อมต่อไปยัง corresponding pins ของเอทีเมกะ 16U2 USB-to-TTL serial chip
- 2) External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), 21 (interrupt 2). พินเหล่านี้สามารถที่จะกำหนดค่าที่เรียกอินเตอร์รัพในค่าต่ำ ๆ, ขอบขาขึ้น และลง หรือเปลี่ยนแปลงค่า
- 3) PWM: 2 ถึง 13 และ 44 ถึง 46 ให้เอาต์พุต PWM output 8-bits

4) SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS) ใช้สำหรับรองรับการสื่อสารแบบ SPI โดยที่ไม่เกี่ยวข้องกันกับ ICSP header ซึ่งจะมีลักษณะคล้ายกับ Uno, Duemilanove และ Diecimila

5) LED 13: เป็น build-in แอลอีดีที่เชื่อมต่อกับดิจิตอลพิน 13 เมื่อพินมีค่าเป็น HIGH แอลอีดีจะติด, แต่เมื่อพินเป็น LOW แอลอีดีจะดับ

6) TWI: 20 (SDA) and 21 (SCL). รองรับการเชื่อมต่อแบบ TWI(I2C)

7) บอร์ดเมกะ 2560 มีนาฬิกาอินพุต 16 พินแต่ละพินให้ความละเอียด 10 บิต

8) AREF แรงดันอ้างอิง สำหรับอนาล็อกอินพุต

9) Reset ใช้ในการรีเซ็ตไมโครคอนโทรลเลอร์ โดยทั่วไปจะใช้โดยการเพิ่มปุ่มรีเซ็ตไว้บนชิปเพื่อป้องกันปุ่มที่อยู่บนบอร์ด

2.4.6 การสื่อสาร

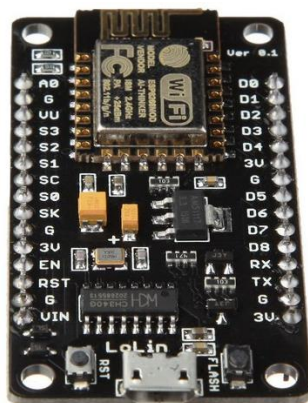
อาร์ดุยโนเมกะสามารถสื่อสารกับคอมพิวเตอร์อาร์ดุยโนตัวอื่น ๆ หรือไมโครคอนโทรลเลอร์ได้ โดยที่ไมโครคอนโทรลเลอร์บนบอร์ด คือเอทีเมกะ 32U4 จะให้การสื่อสารแบบอนุกรม UART TTL (5 โวลต์) ซึ่งมีอยู่ในพิน 0 (Rx) และ 1 (Tx) นอกจากนี้ 32U4 สามารถใช้การสื่อสารแบบอนุกรมผ่านยูเอสบี และจะปรากฏเป็น COM port เสมือนไปยังซอฟต์แวร์ แต่อย่างไรก็ตามต้องใช้ ไฟล์ .inf บนระบบปฏิบัติการ Windows แต่ OSX และ Linux สามารถ recognize ได้โดยอัตโนมัติ

2.4.7 โปรแกรมมิ่ง

อาร์ดุยโนเมกะสามารถรองรับการโปรแกรมด้วยอาร์ดุยโนซอฟต์แวร์ โดยสามารถใช้ได้ทั้งในระบบปฏิบัติการ Windows, Mac OS X และ Linux

2.5 NodeMCU ESP8266 [10]

NodeMCU (อ่านว่า โหนดเอ็มซียู) เป็นแพลตฟอร์มที่ออกแบบทุกอย่างเป็นสถานีเชื่อมต่อ (Node) การทำงานย่อย ๆ และ ใช้ภาษาแล้ว (Lua) ในการเขียนโปรแกรม แต่ด้วยแพลตฟอร์มที่สะดวกในการใช้งาน ทางกลุ่มนักพัฒนาของอีเอสพี (ESP8266) ก็เลยนำ โหนดเอ็มซียู (อีเอสพี 8266) มานับรวมเป็นบอร์ดหนึ่งของอาร์ดุยโนไอดีอี (Arduino IDE-ESP8266) ด้วยเลย ได้จึงได้มีการพัฒนาต่อให้สามารถเขียนในภาษาซีพลัสพลัส หากเป็นผู้ที่นิยมเล่นไมโครคอนโทรลเลอร์อยู่ก่อนจะนิยมเล่นเป็นภาษาซี/ซีพลัสพลัส ซึ่งภาษานี้สามารถไปได้กว้างเล่นได้หลายอย่างกว่าภาษาแล้ว



รูปที่ 2.6 NodeMCU ESP8266 [11]

2.5.1 ข้อดีของบอร์ดอาร์ดูโน้ ไอเอสพี 8266

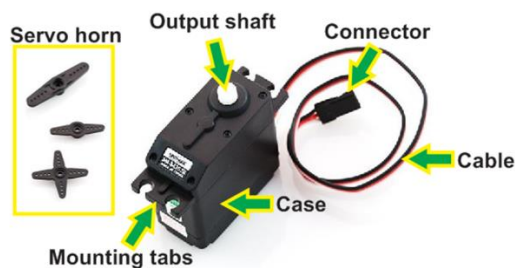
- 1) เป็นแบบโอเพนซอร์สโปรเจก (Open Source Project) มีซอร์สโค้ด (Source code) ให้ได้เรียนรู้หลากหลายแหล่ง
- 2) สามารถอัปโหลดสเกตช์ (Upload sketch) ได้ เชื่อมต่อบอร์ดกับยูเอสบีคอมพิวเตอร์ใช้งานง่าย ขนาดของบอร์ดต่อลงโปรโตบอร์ด (Protoboard) ได้
- 3) ชิปภายในไอเอสพี 8266 มี CPU ขนาด 32 บิต แตกต่างจากอาร์ดูโน้ ที่เป็นซีพียู 8 บิต
- 4) ถึงแม้ว่า I/O จะไม่มากเท่าของอาร์ดูโน้แต่เราสามารถเขียนโปรแกรมลงบนขา GPIO ได้ทุกขาพอ ๆ กัน เป็นข้อดีที่เพิ่มมาจากความต้องการใช้ Wi-Fi เชื่อมต่อเมื่อต้องการเล่นอาร์ดูโน้ทำให้ต้องซื้อโมดูลไวไฟ (Module Wi-Fi) เพิ่ม นั่นคือ NodeMCU (ESP8266) มีต้นทุนต่ำกว่ามาก
- 5) มีอุปกรณ์หลายอย่างที่ใช้งานที่แรงดัน 3.3 โวลต์ เป็นส่วนใหญ่ ดังนั้นเราสามารถนำ NodeMCU (ESP8266) มาใช้เชื่อมต่อได้โดยตรง

2.6 RC Servo Motor [12]

เซอร์โวมอเตอร์ (Servo Motor) ก็คือ มอเตอร์ที่เราสามารถสั่งงาน หรือตั้งค่าแล้วตัวมอเตอร์จะหมุนไปยังตำแหน่งองศาที่เราสั่งได้เองอย่างถูกต้อง โดยใช้การควบคุมแบบป้อนกลับ (Feedback Control) ในบทความนี้จะกล่าวถึงอาร์ชีเซอร์โวมอเตอร์ ซึ่งนิยมนำมาใช้ในเครื่องเล่นที่บังคับด้วยคลื่นวิทยุ (RC = Radio Controlled) เช่น เรือบังคับวิทยุ รถบังคับวิทยุ เฮลิคอปเตอร์บังคับวิทยุ เป็นต้น

การควบคุมแบบป้อนกลับ คือระบบควบคุมที่มีการวัดค่าเอาต์พุตของระบบนำมาเปรียบเทียบกับค่าอินพุตเพื่อควบคุม และปรับแต่งให้ค่าเอาต์พุตของระบบให้มีค่าเท่ากับ หรือใกล้เคียงกับค่าอินพุต

2.6.1 ส่วนประกอบภายนอกของอาร์ซีเซอร์โวมอเตอร์



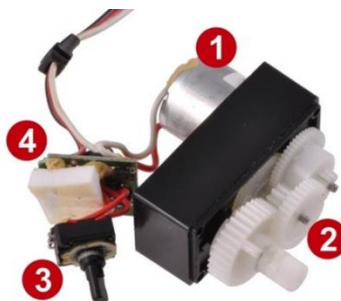
รูปที่ 2.7 ส่วนประกอบภายนอกของอาร์ซีเซอร์โวมอเตอร์ [12]

- 1) ตัวถัง (Case) หรือกรอบของตัวเซอร์โวมอเตอร์
- 2) ส่วนจับยึดตัวเซอร์โวกับชิ้นงาน (Mounting Tab)
- 3) เฟืองส่งกำลัง (Output Shaft)
- 4) เซอร์โวฮอร์น (Servo Horns) ส่วนเชื่อมต่อกับเฟืองส่งกำลังเพื่อสร้างกลไก
- 5) สายเชื่อมต่อ (Cable) เพื่อจ่ายไฟฟ้า และควบคุม เซอร์โวมอเตอร์จะประกอบด้วยสายไฟ 3 เส้น และ ในอาร์ซีเซอร์โวมอเตอร์จะมีสีของสายแตกต่างกันไปดังนี้

- สายสีแดง คือ ไฟเลี้ยง (4.8 – 6 โวลต์)
- สายสีดำ หรือ น้ำตาล คือ กราวด์
- สายสีเหลือง (ส้ม ขาว หรือฟ้า) คือ สายส่งสัญญาณพัลส์ควบคุม (3 – 5 โวลต์)

- 6) จุดเชื่อมต่อสายไฟ (Connector)

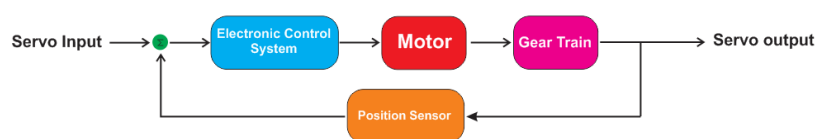
2.6.2 ส่วนประกอบภายในของอาร์ซีเซอร์โวมอเตอร์



รูปที่ 2.8 ส่วนประกอบภายในของอาร์ซีเซอร์โวมอเตอร์ [12]

- 1) Motor เป็นส่วนของตัวมอเตอร์
- 2) Gear Train หรือ Gearbox เป็นชุดเกียร์ทดแรง
- 3) Position Sensor เป็นเซ็นเซอร์ตรวจจับตำแหน่งเพื่อหาค่าองศาในการหมุน
- 4) Electronic Control System เป็นส่วนที่ควบคุมและประมวลผล

2.6.3 เซอร์โวมอเตอร์บล็อกไดอะแกรม



รูปที่ 2.9 เซอร์โวมอเตอร์บล็อกไดอะแกรม [12]

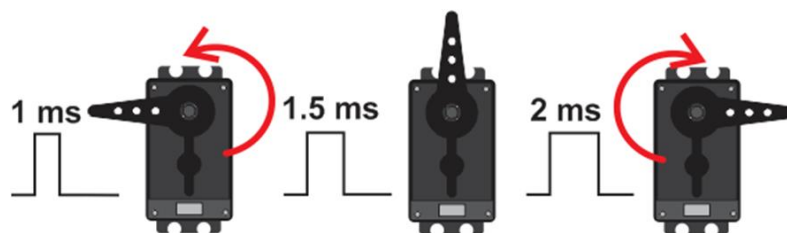
หลักการทำงานของอาร์ซีเซอร์โวมอเตอร์ คือ เมื่อจ่ายสัญญาณพัลส์เข้ามายังอาร์ซีเซอร์โวมอเตอร์ ส่วนวงจรควบคุม (Electronic Control System) ภายในเซอร์โวจะทำการอ่าน และประมวลผลค่าความกว้างของสัญญาณพัลส์ที่ส่งเข้ามาเพื่อแปลค่าเป็นตำแหน่งองศาที่ต้องการให้มอเตอร์หมุนเคลื่อนที่ไปยังตำแหน่งนั้น แล้วส่งคำสั่งไปทำการควบคุมให้มอเตอร์หมุนไปยังตำแหน่งที่ต้องการ โดยมีตำแหน่งเซ็นเซอร์ (Position Sensor) เป็นตัวคอยวัดค่ามุมที่มอเตอร์กำลังหมุน เป็นการป้อนกลับ (Feedback) มาให้วงจรควบคุมเปรียบเทียบกับค่าอินพุตเพื่อควบคุมให้ได้ตำแหน่งที่ต้องการอย่างถูกต้องแม่นยำ

2.6.4 สัญญาณอาร์ซีในรูปแบบ PWM

ตัวอาร์ซีเซอร์โวมอเตอร์ออกแบบมาไว้สำหรับรับคำสั่งจากรีโมทคอนโทรล ที่ใช้ควบคุมของเล่นด้วยสัญญาณวิทยุต่าง ๆ เช่น เครื่องบินบังคับ รถบังคับ เรือบังคับ เป็นต้น ซึ่งรีโมทจำพวกนี้ที่ภาครับจะแปลงความถี่วิทยุออกมาในรูปแบบสัญญาณ PWM (การปรับความกว้างพัลส์ Pulse Width Modulation)

มุมหรือองศาจะขึ้นอยู่กับความกว้างของสัญญาณพัลส์ ซึ่งโดยส่วนมากความกว้างของพัลส์ที่ใช้ในอาร์ซีเซอร์โวมอเตอร์ จะอยู่ในช่วง 1-2 มิลลิวินาที หรือ 0.5-2.5 มิลลิวินาที

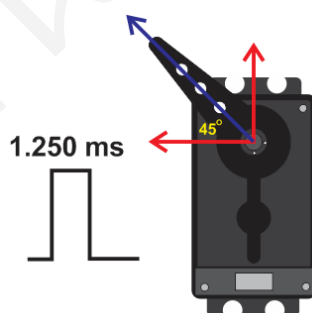
ยกตัวอย่างเช่น หากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 1 มิลลิวินาที ตัวเซอร์โวมอเตอร์จะหมุนไปทางซ้ายสุด ในทางกลับกันหากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 2 มิลลิวินาที ตัวเซอร์โวมอเตอร์จะหมุนไปยังตำแหน่งขวาสุด แต่หากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 1.5 มิลลิวินาที ตัวเซอร์โวมอเตอร์ก็จะหมุนมาอยู่ที่ตำแหน่งตรงกลางพอดี



รูปที่ 2.10 องศาการหมุนของอาร์ชีเซอร์โวมอเตอร์ [12]

ดังนั้นสามารถกำหนดองศาการหมุนของอาร์ชีเซอร์โวมอเตอร์ได้โดยการเทียบค่า เช่น อาร์ชีเซอร์โวมอเตอร์สามารถหมุนได้ 180 องศา โดยที่ 0 องศา ใช้ความกว้างพัลส์เท่ากับ 1000 ไมโครวินาที ที่ 180 องศาความกว้างพัลส์เท่ากับ 2000 ไมโครวินาที เพราะฉะนั้นค่าที่เปลี่ยนไป 1 องศาจะใช้ความกว้างพัลส์ต่างกัน $(2000-1000)/180$ เท่ากับ 5.55 ไมโครวินาที

จากการหาค่าความกว้างพัลส์ที่มุม 1 องศาข้างต้น หากต้องกำหนดให้อาร์ชีเซอร์โวมอเตอร์หมุนไปที่มุม 45 องศาจะหาค่าพัลส์ที่ต้องการได้จาก 5.55×45 เท่ากับ 249.75 ไมโครวินาที แต่ที่มุม 0 องศาเราเริ่มที่ความกว้างพัลส์ 1 มิลลิวินาที หรือ 1000 ไมโครวินาที เพราะฉะนั้นความกว้างพัลส์ที่ใช้กำหนดให้อาร์ชีเซอร์โวมอเตอร์หมุนไปที่ 45 องศา คือ $1000 + 249.75$ เท่ากับ ประมาณ 1250 ไมโครวินาที



รูปที่ 2.11 ความกว้างพัลส์เมื่อกำหนดให้หมุน 45 องศา [12]

2.7 Mobile Application [13]

โมบายแอปพลิเคชัน เป็นโปรแกรมคอมพิวเตอร์ ที่ถูกออกแบบให้ สามารถใช้งานได้บนสมาร์ตโฟน หรือแท็บเล็ต ได้อย่างรวดเร็ว สะดวก และเรียบง่าย ดังนั้นจึงจะเห็นว่า ในปัจจุบันมีโมบายแอปพลิเคชัน ต่าง ๆ ที่ถูกพัฒนาออกมาอย่างมากมาย ทั้งแอปพลิเคชันที่เกี่ยวกับการท่องเที่ยว, แอป

พลีเคชันการทำธุรกรรมออนไลน์, ความบันเทิง, แอปพลิเคชันเกมส์ต่าง ๆ เป็นต้น ซึ่งจากที่กล่าวมาจึงปฏิเสธไม่ได้ว่า ในตอนนี้แอปพลิเคชันได้กลายเป็นจุดสำคัญ ในการทำธุรกิจ หรือกิจกรรมต่าง ๆ ไปแล้ว

2.7.1 ประเภทของโมบายแอปพลิเคชัน

1) Native App คือแอปพลิเคชันที่ถูกพัฒนามาด้วย Library (ชุดคำสั่ง) หรือ SDK (เครื่องมือที่เอาไว้สำหรับพัฒนาโปรแกรมหรือแอปพลิเคชัน) ของ OS Mobile นั้น ๆ โดยเฉพาะ อาทิ แอนดรอยด์ (Android) ใช้ Android SDK, iOS ใช้ Objective c, Windows Phone ใช้ C# เป็นต้น

2) Hybrid Application คือแอปพลิเคชันที่ถูกพัฒนาขึ้นมาด้วยจุดประสงค์ ที่ต้องการ ให้สามารถ รันบนระบบปฏิบัติการได้ทุก OS โดยใช้ Framework (ชุดคำสั่ง) เข้าช่วย เพื่อให้สามารถ ทำงานได้ทุกระบบปฏิบัติการ

3) Web Application คือแอปพลิเคชันที่ถูกเขียนขึ้นมาเพื่อเป็น Browser สำหรับการ ใช้งานเว็บเพจต่าง ๆ ซึ่งถูกปรับแต่งให้แสดงผลแต่ส่วนที่จำเป็น เพื่อเป็นการลดทรัพยากรในการ ประมวลผล ของตัวเครื่องสมาร์ทโฟน หรือแท็บเล็ตทำให้โหลดหน้าเว็บไซต์ได้เร็วขึ้น อีกทั้งผู้ใช้งานยัง สามารถใช้งานผ่าน อินเทอร์เน็ตและอินทราเน็ตในความเร็วต่ำได้

2.7.2 ข้อดีของโมบายแอปพลิเคชันแต่ละประเภท

1) Native App มีข้อดีคือผู้ใช้งานสามารถเข้าถึงได้ง่าย จาก Google Play หรือ Apple's App Store รวมถึงการทำงานแบบไม่ต้อง เชื่อมต่ออินเทอร์เน็ตในบางแอป ทำให้ผู้ใช้งาน สะดวกในการใช้งานแอปได้ทุกที่หากไม่มีสัญญาณอินเทอร์เน็ต รวมถึงสะดวกในการใช้ที่ผู้ใช้งาน ใช้กล้องดิจิทัล, GPS และรายชื่อผู้ติดต่อ ในระหว่างที่ใช้งานแอปได้อีกด้วย

2) Hybrid Application เป็นประเภทแอป ที่ถูกออกแบบมาให้รองรับระบบปฏิบัติการ ได้หลายแพลตฟอร์มในแอปเดียวจึงมีข้อดีคือ ทำให้ผู้พัฒนาไม่ต้องเสียเวลาในการทำ เพราะเขียน ชุดคำสั่งครั้งเดียวสามารถใช้ได้ทุกแพลตฟอร์ม และเสียค่าใช้จ่ายน้อย

3) Web Application ใช้งานง่ายได้สะดวกทุกที่ ทุกเวลา ถ้าหากไม่มีเครื่อง คอมพิวเตอร์ แต่ต้องการใช้ Web browser ก็สามารถใช้อุปกรณ์นี้ได้ รวมถึงมีการอัปเดต แก้ไข ข้อผิดพลาดต่าง ๆ อยู่ตลอดเวลา และใช้งานได้ทุกแพลตฟอร์ม

บทที่ 3 วิธีการดำเนินงาน

การดำเนินงานการจัดทำโครงงานเครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลี้ยงขนาดเล็ก ผู้จัดทำได้กำหนดแนวทางในการดำเนินงาน เพื่อให้โครงงานสำเร็จตามวัตถุประสงค์ และขอบเขต ตามที่กำหนดไว้

3.1 แผนการดำเนินงาน

เพื่อให้การทำงานดำเนินเป็นไปอย่างราบรื่น แผนการดำเนินงานจึงเป็นสิ่งสำคัญที่ช่วยให้ ทำผลงานตามเป้าหมายกิจกรรมที่กำหนดไว้ สำเร็จตามกรอบระยะเวลาที่กำหนด

ตารางที่ 3.1 แผนการดำเนินงานของโครงการ

กิจกรรม	พ.ศ. 2560			พ.ศ. 2561						
	ก.พ.	มี.ค.	ธ.ค.	ม.ค.	มี.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.
1. รวบรวมความต้องการที่เกี่ยวข้องโครงงาน	— — .									
2. ศึกษาวิเคราะห์ข้อมูลของระบบ	— —									
3. กำหนดขอบเขตของระบบ		—								
4. ศึกษาค้นคว้าทฤษฎีที่เกี่ยวข้อง			— — .		— — — —					
5. ออกแบบเครื่องให้อาหารและประกอบเครื่องให้อาหาร			— — — —				— — — —			
6. ออกแบบและเขียนโมบายแอปพลิเคชัน			— — — —				— — — —			
7. ออกแบบและเขียนระบบของเครื่องให้อาหาร			— — — —				— — — —			
8. ติดตั้งและทดสอบระบบ			— — —					— — — —		
9. แก้ไขปรับปรุงระบบ				— —					— — —	

ตารางที่ 3.1 แผนการดำเนินงานของโครงการ (ต่อ)

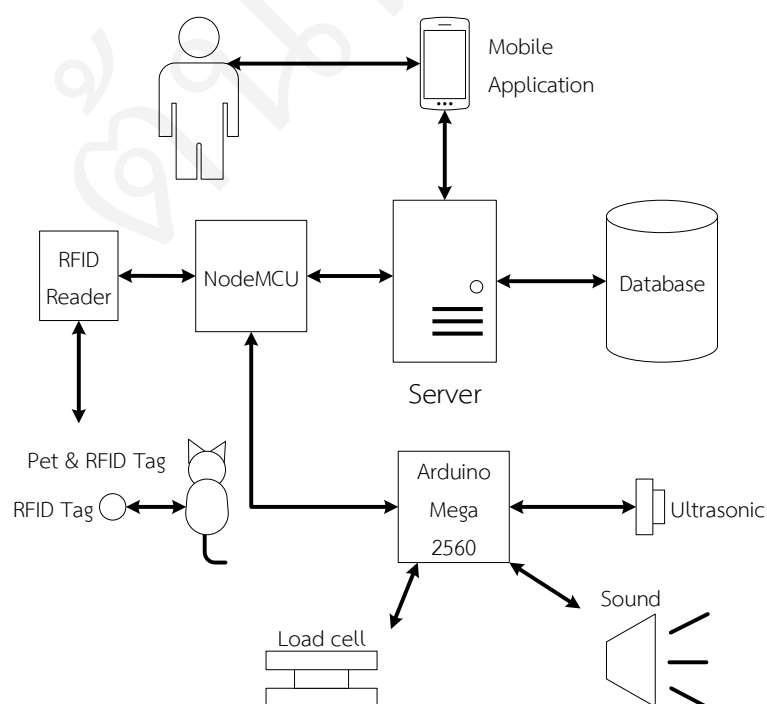
กิจกรรม	พ.ศ. 2560			พ.ศ. 2561						
	ก.พ.	มี.ค.	ธ.ค.	ม.ค.	มี.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.
10. จัดทำรูปเล่มโครงงานและส่งโครงงาน			—	—						

— — — . แสดงแผนการดำเนินงาน

————— แสดงการดำเนินงานจริง

3.2 การออกแบบ / เครื่องมือ

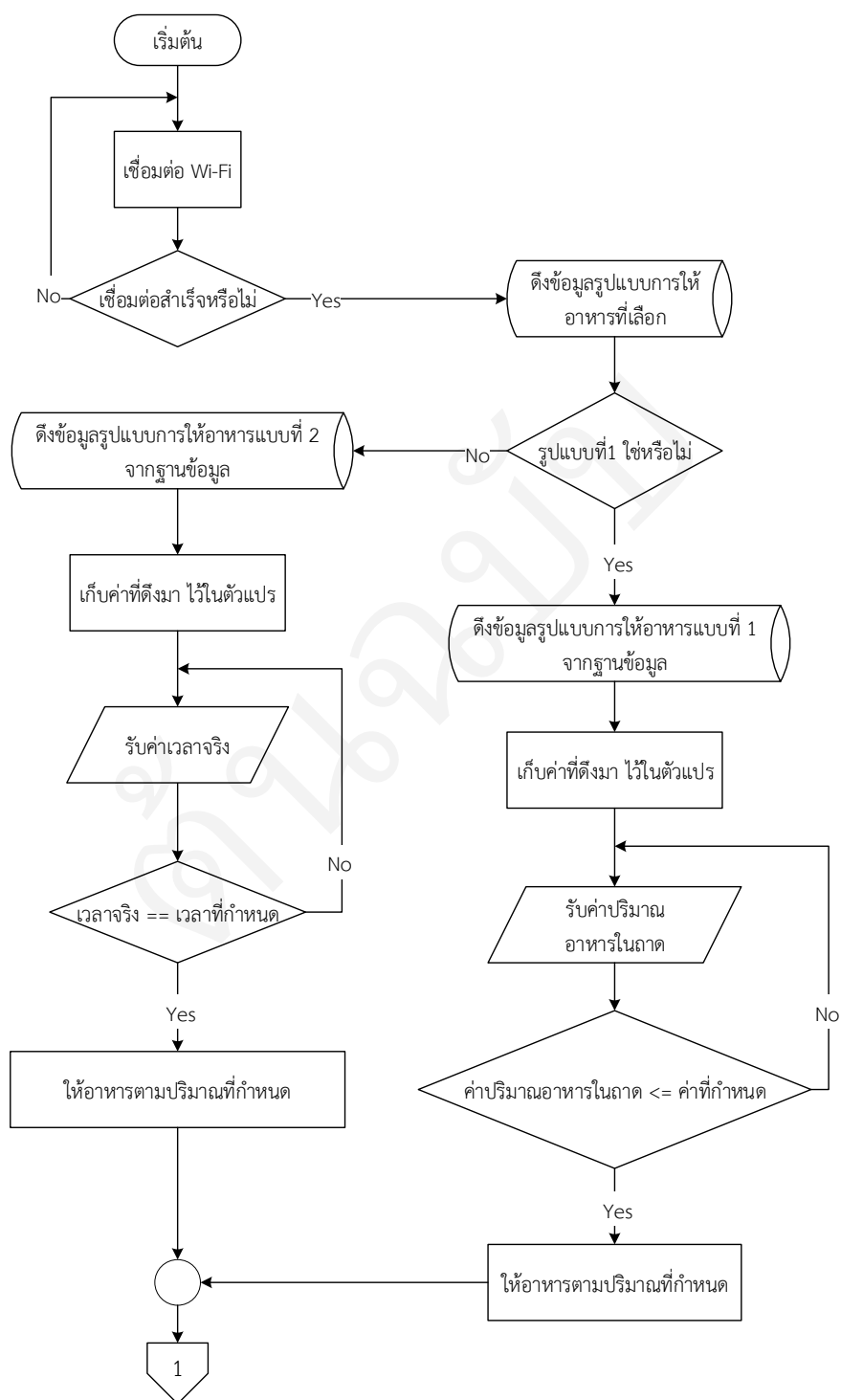
การออกแบบแยกออกเป็นสองส่วนหลัก คือ ส่วนของตัวเครื่อง โดยตัวเครื่องนั้นใช้ไมโครคอนโทรลเลอร์เป็นตัวหลักในการควบคุมการทำงานระหว่างอุปกรณ์ที่ต่อร่วม และส่วนที่สองโมบายแอปพลิเคชัน เป็นส่วนบันทึกรูปแบบการให้อาหาร บันทึกข้อมูลสัตว์เลี้ยง แสดงผลข้อมูลต่าง ๆ ของสัตว์เลี้ยง เช่น สถิติการกินของสัตว์เลี้ยงแต่ละตัวในรูปแบบตารางตัวเลข



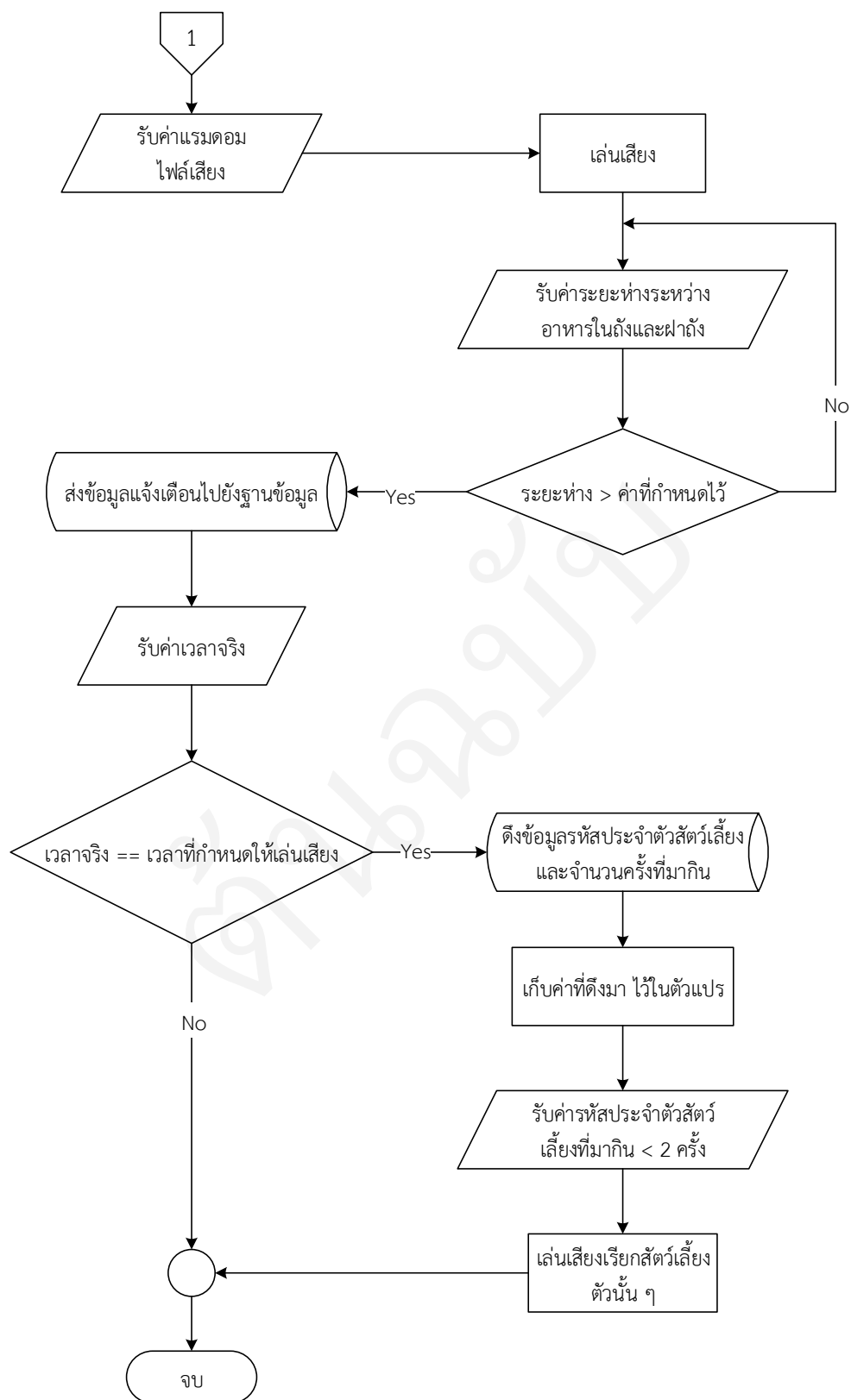
รูปที่ 3.1 ภาพรวมระบบ

3.3 Flowchart Diagram

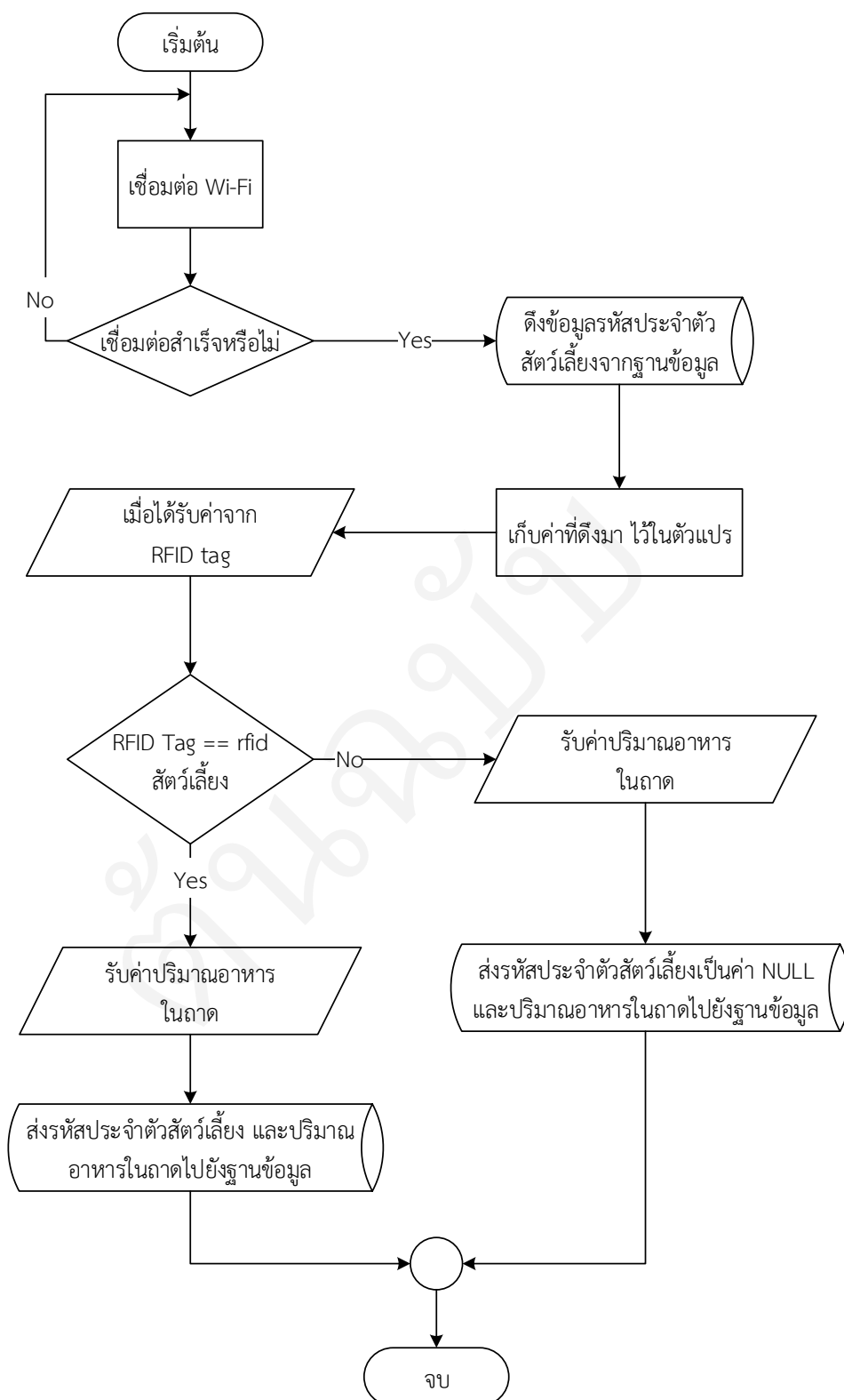
3.3.1 Flowchart Diagram เครื่องให้อาหาร



รูปที่ 3.2 Flowchart Diagram การทำงานของเครื่องให้อาหาร



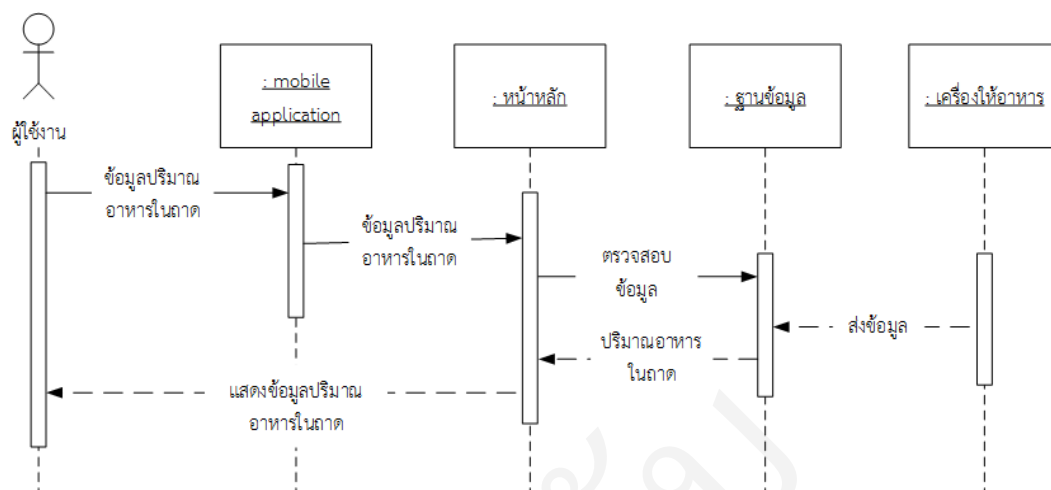
รูปที่ 3.3 Flowchart Diagram การทำงานของเครื่องให้อาหาร (ส่วนที่ 1 ต่อ)



รูปที่ 3.4 Flowchart Diagram การตรวจสอบสัตว์เลี้ยงใกล้เครื่อง และปริมาณอาหารในถาดแบบเรียลไทม์

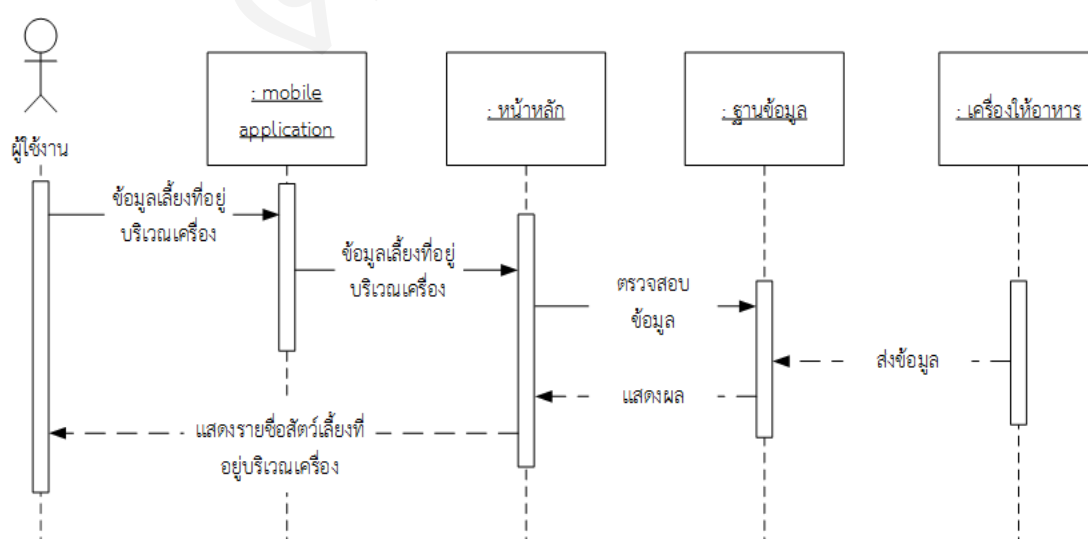
3.4 Sequence Diagram

3.4.1 Sequence Diagram Mobile Application



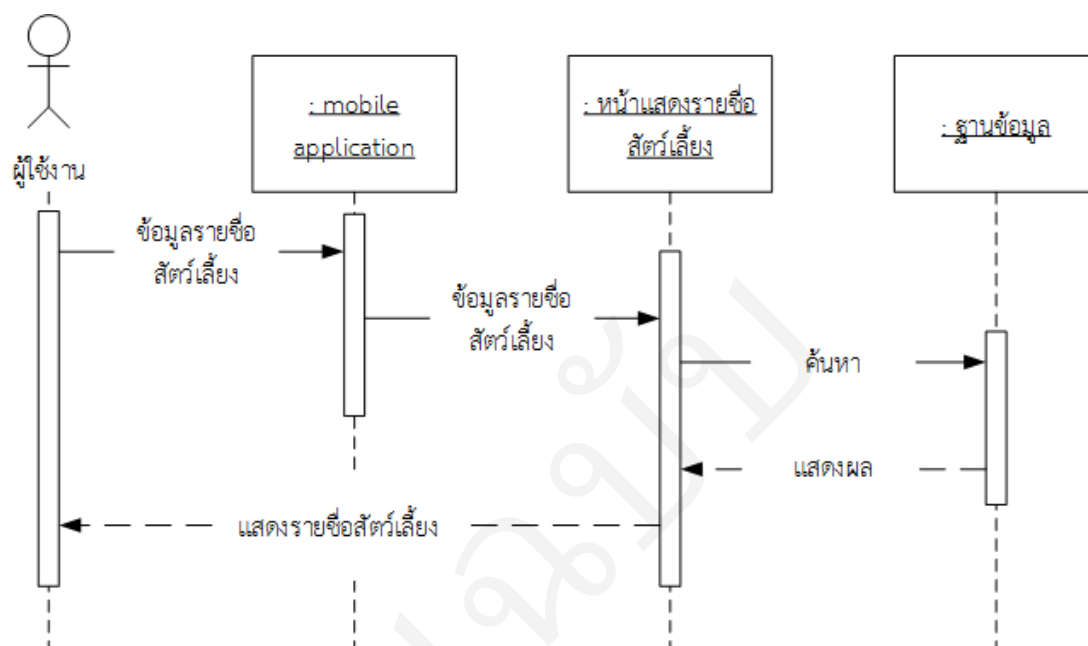
รูปที่ 3.5 Sequence Diagram แสดงข้อมูลปริมาณอาหารในถาดอาหาร

ผู้ใช้งานสามารถดูปริมาณอาหารในถาดผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าหลักของการทำงาน โมบายแอปพลิเคชันจะทำการตรวจสอบข้อมูลกับฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ ซึ่งเครื่องให้อาหารจะมีการส่งข้อมูลมายังฐานข้อมูล และหน้าหลักจะแสดงข้อมูลปริมาณอาหารในถาดให้กับผู้ใช้งาน



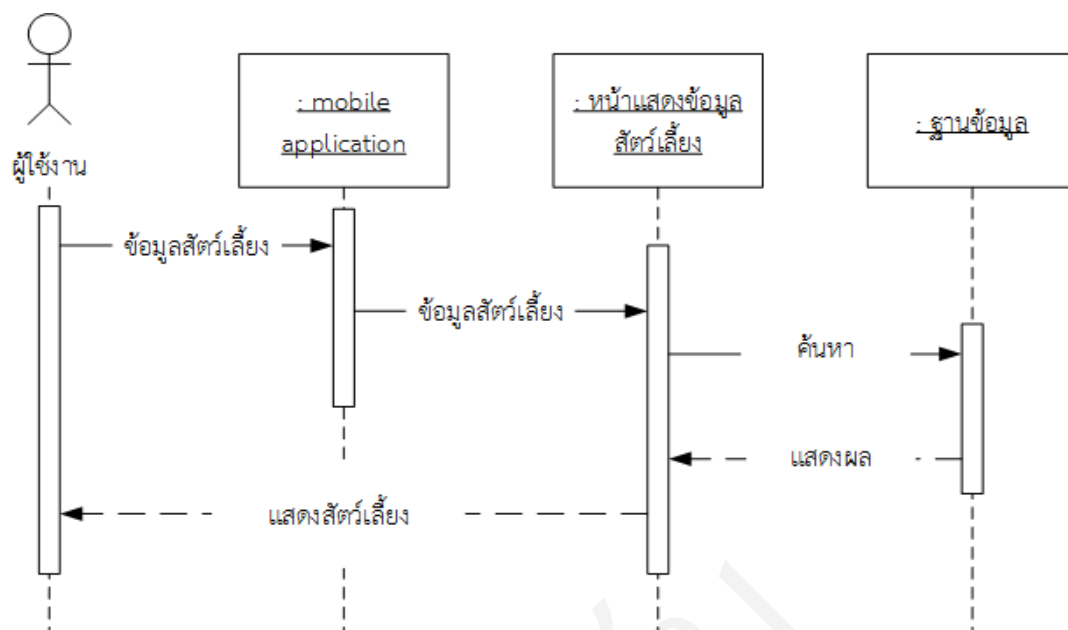
รูปที่ 3.6 Sequence Diagram แสดงรายชื่อสัตว์เลี้ยงใกล้บริเวณเครื่องให้อาหาร

ผู้ใช้งานสามารถดูข้อมูลสัตว์เลี้ยงที่อยู่บริเวณเครื่องให้อาหารผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าหลักของการทำงาน โมบายแอปพลิเคชันจะทำการตรวจสอบข้อมูลกับฐานข้อมูล โดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ ซึ่งเครื่องให้อาหารจะมีการส่งข้อมูลมายังฐานข้อมูล และหน้าหลักจะแสดงข้อมูลสัตว์เลี้ยงที่อยู่บริเวณเครื่องให้อาหารให้กับผู้ใช้งาน



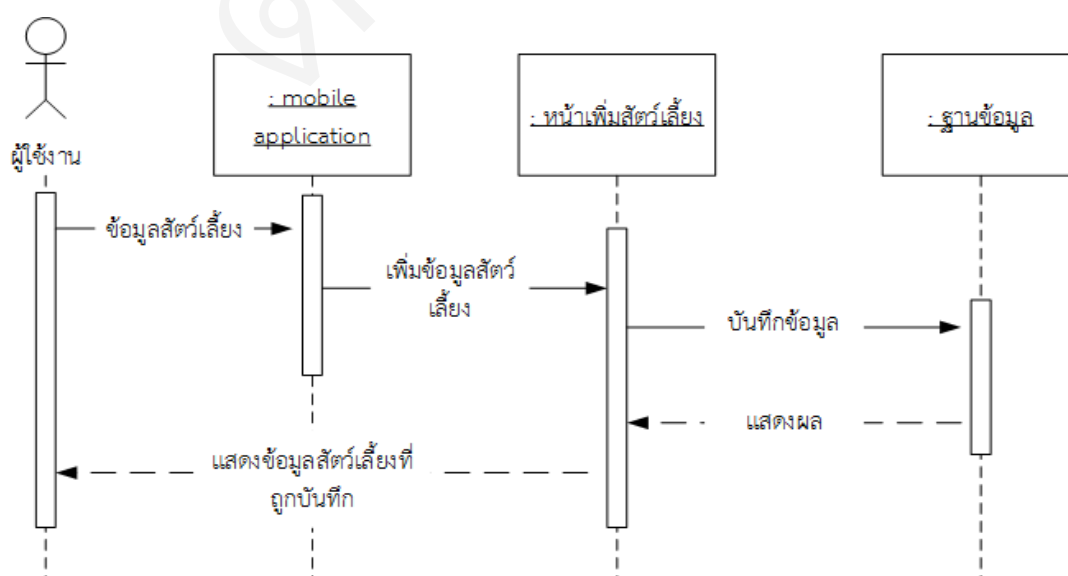
รูปที่ 3.7 Sequence Diagram แสดงรายชื่อสัตว์เลี้ยง

ผู้ใช้งานสามารถดูรายชื่อสัตว์เลี้ยงที่มีการลงทะเบียนไว้ในระบบผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าแสดงรายชื่อสัตว์เลี้ยง โมบายแอปพลิเคชันจะทำการค้นหาข้อมูลจากฐานข้อมูล โดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ และหน้าแสดงรายชื่อสัตว์เลี้ยงจะทำการแสดงรายชื่อสัตว์เลี้ยงให้กับผู้ใช้งาน



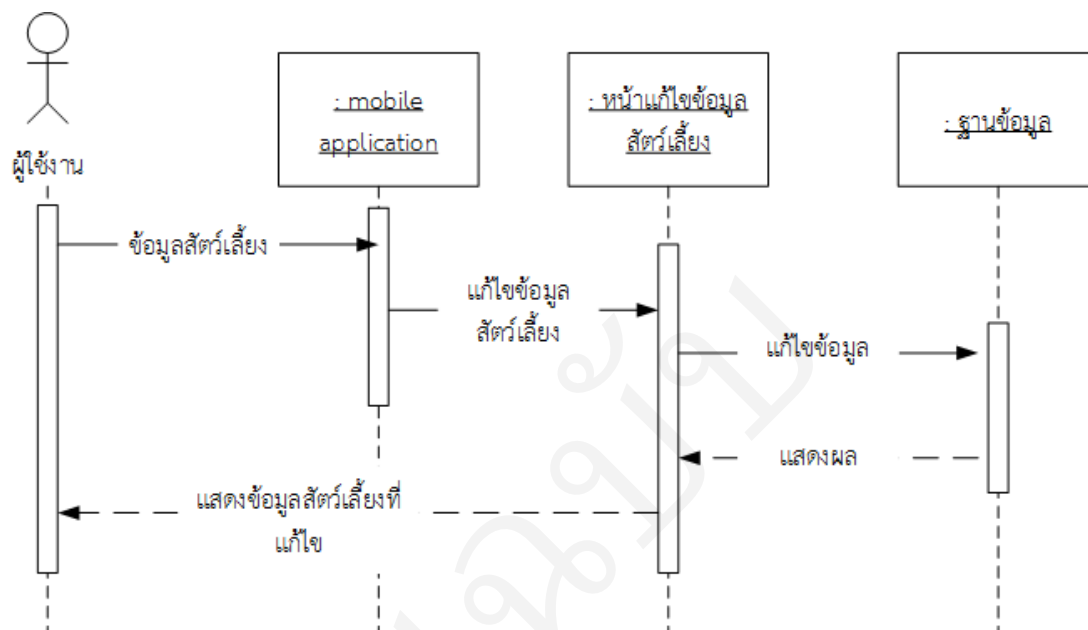
รูปที่ 3.8 Sequence Diagram แสดงข้อมูลสัตว์เลี้ยง

ผู้ใช้งานสามารถดูข้อมูลสัตว์เลี้ยงแต่ละตัวที่มีการลงทะเบียนไว้ในระบบผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าแสดงข้อมูลสัตว์เลี้ยง โมบายแอปพลิเคชันจะทำการค้นหาข้อมูลจากฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ และหน้าแสดงข้อมูลสัตว์เลี้ยงจะทำการแสดงข้อมูลสัตว์เลี้ยงให้กับผู้ใช้งาน



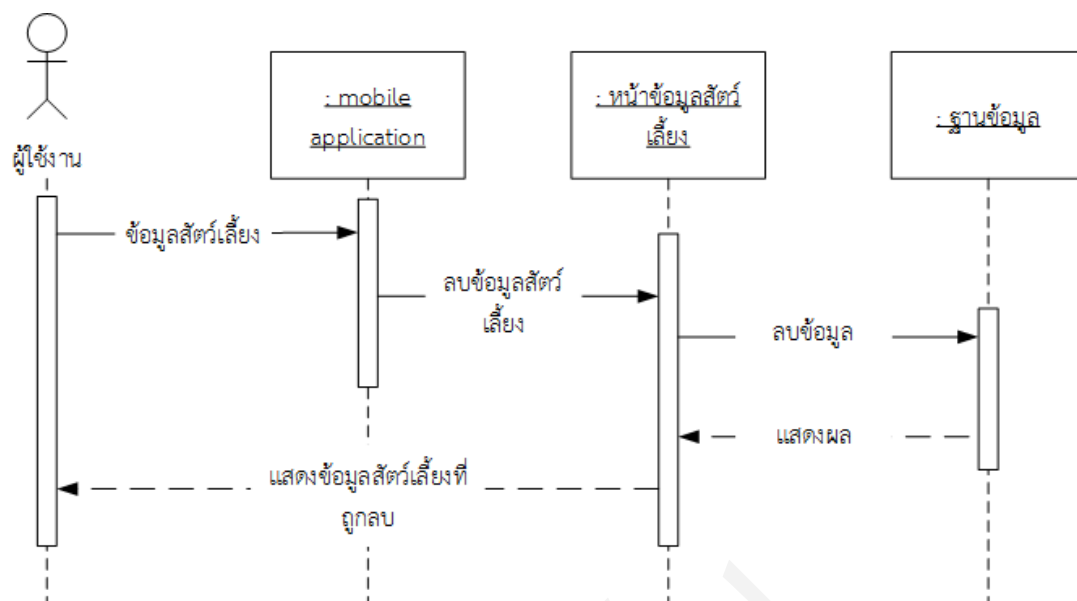
รูปที่ 3.9 Sequence Diagram บันทึกข้อมูลสัตว์เลี้ยง

ผู้ใช้งานสามารถเพิ่มสัตว์เลี้ยงเข้าสู่ระบบผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าเพิ่มข้อมูลสัตว์เลี้ยงทำการกรอกข้อมูลสัตว์เลี้ยง และบันทึกข้อมูล โมบายแอปพลิเคชันจะทำการเพิ่มข้อมูลสัตว์เลี้ยงไปยังฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ และหน้าเพิ่มข้อมูลสัตว์เลี้ยงจะทำการแสดงข้อมูลสัตว์เลี้ยงที่ถูกบันทึกให้กับผู้ใช้งาน



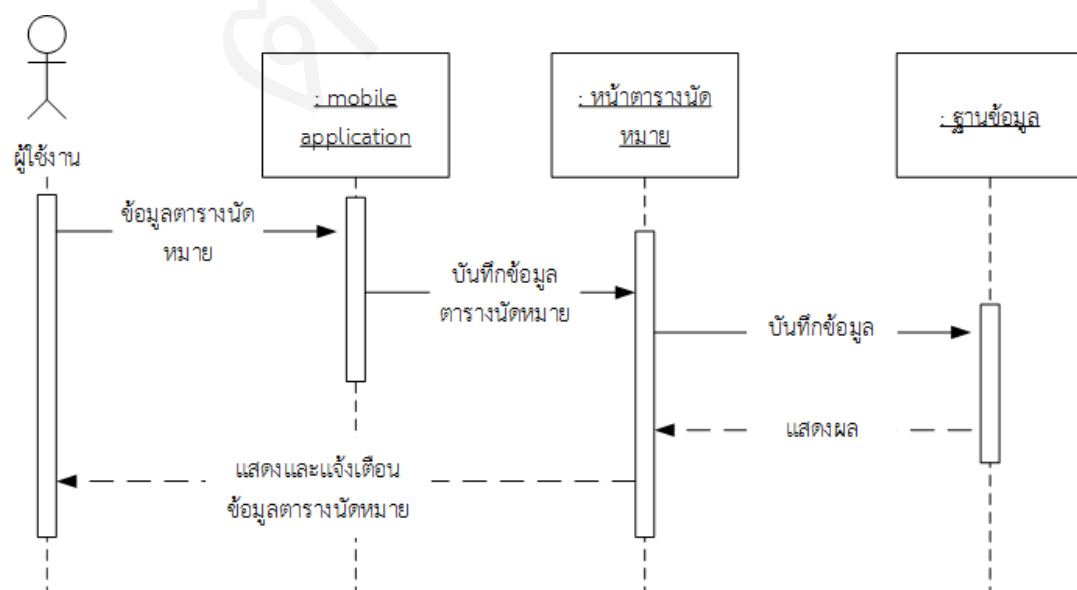
รูปที่ 3.10 Sequence Diagram แก้ไขข้อมูลสัตว์เลี้ยง

ผู้ใช้งานสามารถแก้ไขข้อมูลสัตว์เลี้ยงผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าแก้ไขข้อมูลสัตว์เลี้ยงทำการแก้ไขข้อมูลสัตว์เลี้ยง และบันทึกข้อมูล โมบายแอปพลิเคชันจะทำการแก้ไขข้อมูลสัตว์เลี้ยงในฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ และหน้าแก้ไขข้อมูลสัตว์เลี้ยงจะทำการแสดงข้อมูลสัตว์เลี้ยงที่ถูกแก้ไขให้กับผู้ใช้งาน



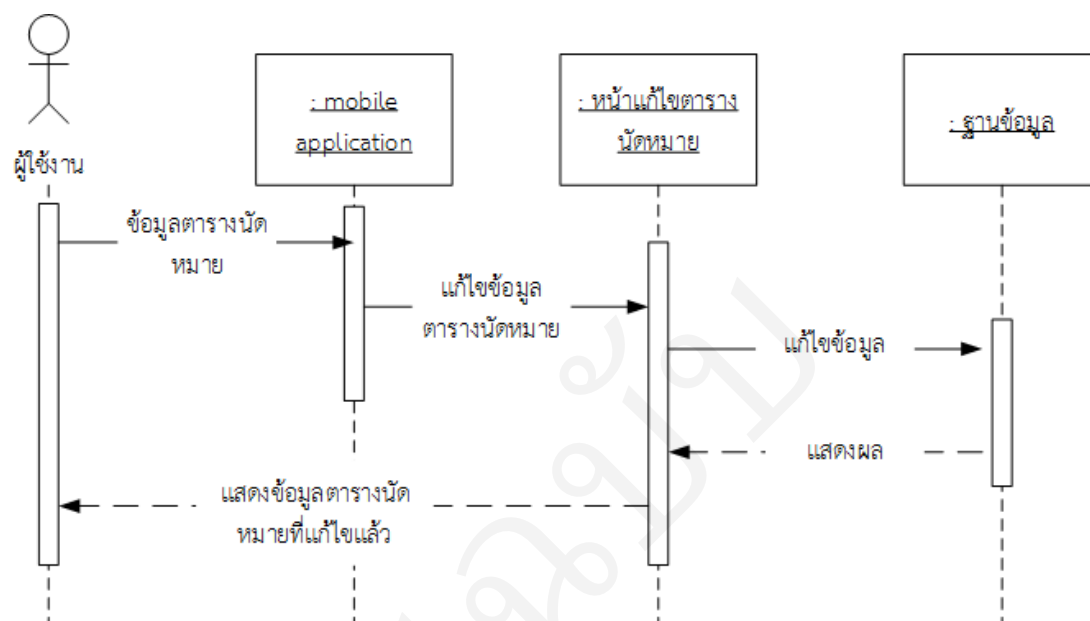
รูปที่ 3.11 Sequence Diagram ลบข้อมูลส่วนตัวเลี้ยง

ผู้ใช้งานสามารถลบส่วนตัวเลี้ยงที่มีอยู่ในระบบผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าข้อมูลส่วนตัวเลี้ยงทำการลบข้อมูลส่วนตัวเลี้ยง โมบายแอปพลิเคชันจะทำการลบข้อมูลส่วนตัวเลี้ยงออกจากฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ และหน้าข้อมูลส่วนตัวเลี้ยงจะไม่แสดงข้อมูลส่วนตัวเลี้ยงที่ถูกลบ



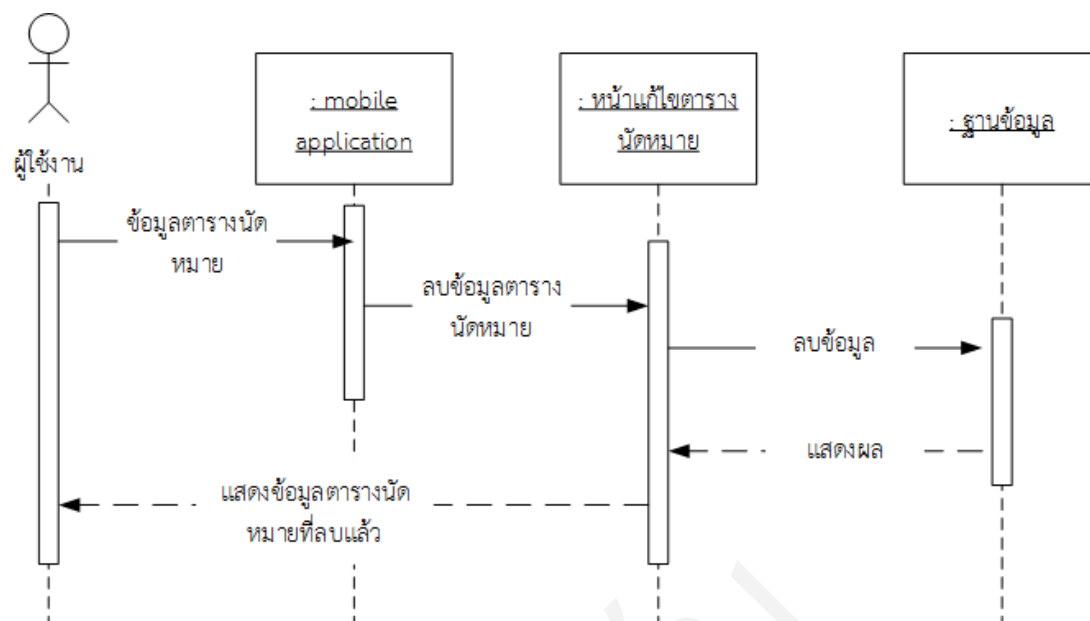
รูปที่ 3.12 Sequence Diagram บันทึก และแสดงข้อมูลตารางนัดหมาย

ผู้ใช้งานสามารถเพิ่มข้อมูลการนัดหมายได้ ผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าตารางนัดหมาย และทำการบันทึกข้อมูลการนัดหมาย โมบายแอปพลิเคชันจะทำการเพิ่มข้อมูลการนัดหมายไปยังฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ และหน้าตารางนัดหมาย จะทำการแสดงข้อมูลการนัดหมายให้กับผู้ใช้งาน



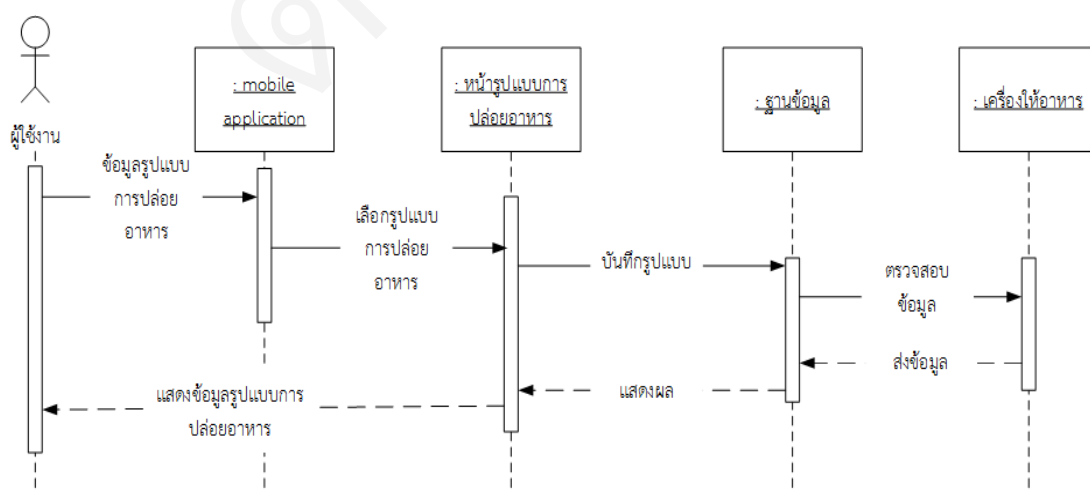
รูปที่ 3.13 Sequence Diagram แก้ไขข้อมูลตารางนัดหมาย

ผู้ใช้งานสามารถแก้ไขข้อมูลการนัดหมายได้ ผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าแก้ไขตารางนัดหมาย และทำการแก้ไขข้อมูลการนัดหมาย โมบายแอปพลิเคชันจะทำการแก้ไขข้อมูลการนัดหมายในฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ และหน้าแก้ไขตารางนัดหมายจะทำการแสดงข้อมูลการนัดหมายที่มีการแก้ไขให้กับผู้ใช้งาน



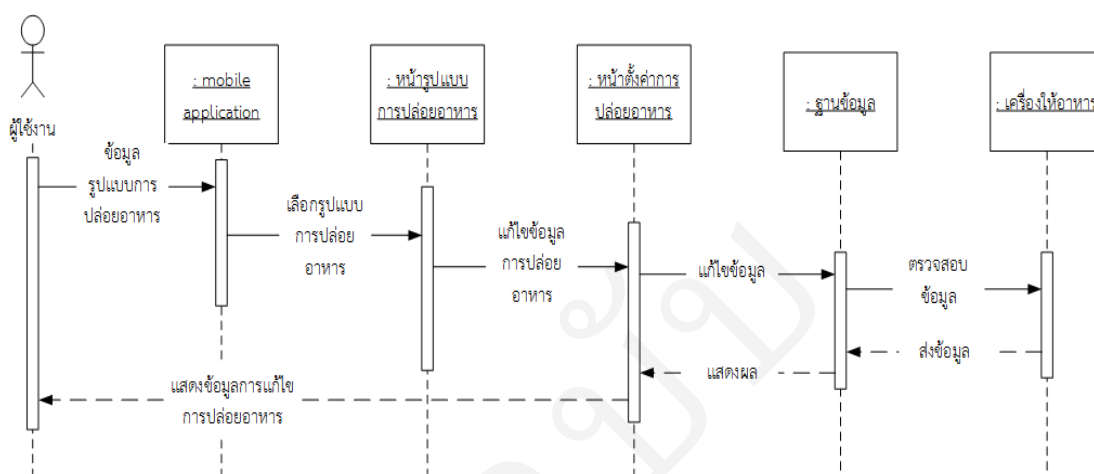
รูปที่ 3.14 Sequence Diagram ลบข้อมูลตารางนัดหมาย

ผู้ใช้งานสามารถลบข้อมูลการนัดหมายออกจากระบบผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าแก้ไขตารางนัดหมายและทำการลบข้อมูลการนัดหมาย โมบายแอปพลิเคชันจะทำการลบข้อมูลการนัดหมายออกจากฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ และหน้าแก้ไขตารางนัดหมายจะทำการแสดงข้อมูลการนัดหมายที่ถูกลบให้กับผู้ใช้งาน



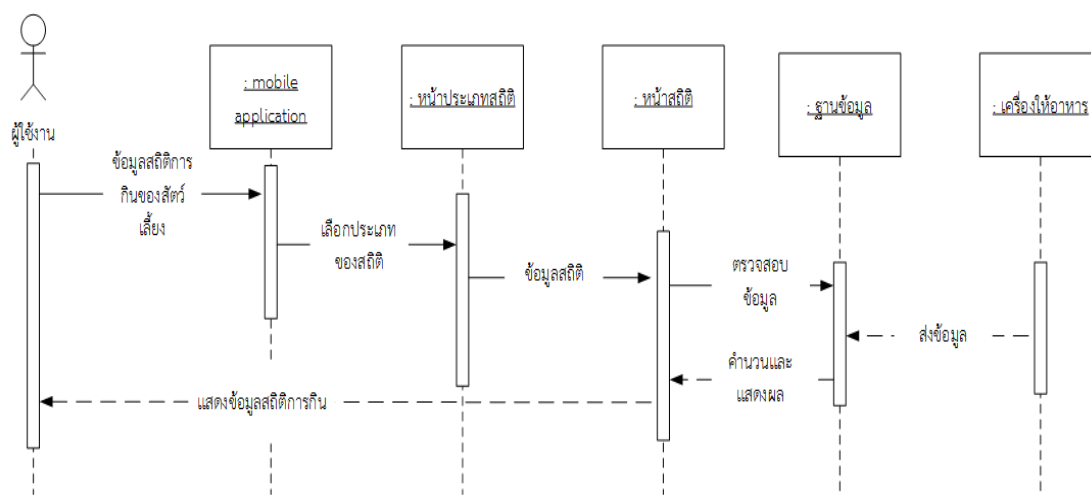
รูปที่ 3.15 Sequence Diagram บันทึกรูปแบบการป้อนอาหาร

ผู้ใช้งานสามารถบันทึกรูปแบบการให้อาหารผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้ารูปแบบการปล่อยอาหาร ทำการเลือกรูปแบบการปล่อยอาหาร โมบายแอปพลิเคชันจะทำการบันทึกข้อมูลรูปแบบการปล่อยอาหาร ไปยังฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ ซึ่งเครื่องให้อาหารมีการเชื่อมต่อกับฐานข้อมูลไว้เพื่อทำการรับส่งข้อมูลกับฐานข้อมูล และหน้ารูปแบบการปล่อยอาหารจะทำการแสดงรูปแบบการให้อาหารที่กำลังทำงานให้กับผู้ใช้งาน



รูปที่ 3.16 Sequence Diagram บันทึกการปล่อยอาหาร

ผู้ใช้งานสามารถบันทึกข้อมูลการปล่อยอาหารผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้ารูปแบบการปล่อยอาหาร ทำการเลือกรูปแบบการปล่อยอาหาร และไปยังหน้าตั้งค่าการปล่อยอาหารเพื่อบันทึกข้อมูล โมบายแอปพลิเคชันจะทำการบันทึกข้อมูลการปล่อยอาหาร ไปยังฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ ซึ่งเครื่องให้อาหารมีการเชื่อมต่อกับฐานข้อมูลไว้เพื่อทำการรับส่งข้อมูลกับฐานข้อมูล หน้าตั้งค่าการปล่อยอาหารจะทำการแสดงข้อมูลการปล่อยอาหารที่บันทึกให้กับผู้ใช้งาน

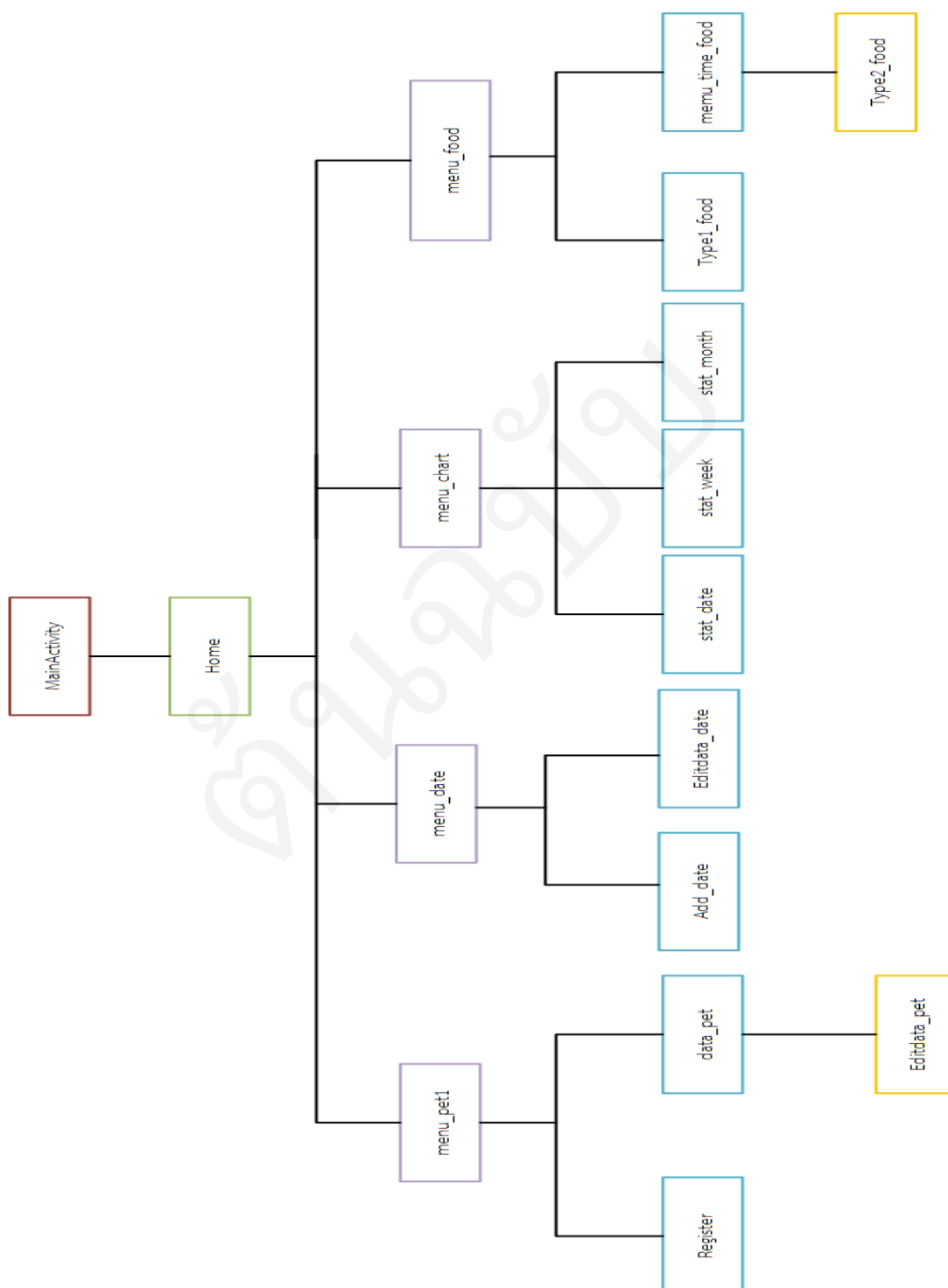


รูปที่ 3.17 Sequence Diagram แสดงข้อมูลสถิติ

ผู้ใช้งานสามารถดูข้อมูลสถิติการกินอาหารของสัตว์ผ่านทางโมบายแอปพลิเคชัน โดยการเปิดไปที่หน้าประเภทสถิติ ทำการเลือกประเภทสถิติ แล้วจะเข้าสู่หน้าสถิติ โมบายแอปพลิเคชัน จะทำการตรวจสอบข้อมูลการกินอาหารของสัตว์เลี้ยง กับฐานข้อมูลโดยใช้ไฟล์ PHP ในการเชื่อมต่อกับเซิร์ฟเวอร์ ซึ่งเครื่องให้อาหารมีการเชื่อมต่อกับฐานข้อมูลไว้เพื่อทำการรับส่งข้อมูลกับฐานข้อมูล หน้าสถิติจะทำการแสดงข้อมูลสถิติการกินให้กับผู้ใช้งาน

3.5 การออกแบบหน้าจอบายแอปพลิเคชัน

3.5.1 Mobile Application Sitemaps



รูปที่ 3.18 Mobile Application Sitemaps

3.5.2 หน้าจอบริยายแอปพลิเคชัน

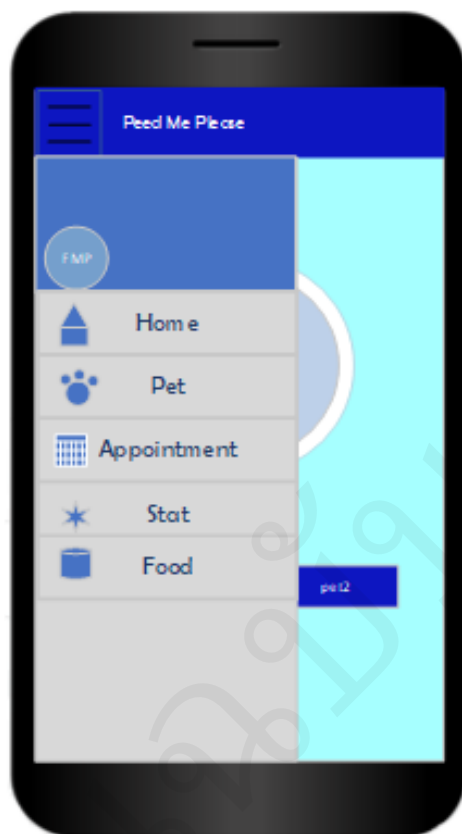
1) หน้า Home



รูปที่ 3.19 ออกแบบหน้าจอในส่วนของหน้า Home

ในส่วนนี้ทำการออกแบบหน้าต่างของโปรแกรมหน้า Home ของระบบเครื่องให้อาหารสัตว์เลี้ยงขนาดเล็ก โดยจะมีการแสดงปริมาณอาหารที่คงเหลือในถาดอาหาร และสัตว์เลี้ยงที่อยู่ใกล้บริเวณเครื่องให้อาหาร

2) หน้าเมนูหลัก



รูปที่ 3.20 ออกแบบหน้าจอในส่วนเมนูหลักของระบบ

ในส่วนของเมนูหลักในการเข้าถึงข้อมูลในส่วนต่าง ๆ ของระบบประกอบไปด้วย หน้า Home หน้า Pet หน้า Appointment หน้า Stat และหน้า Food

3) หน้ารายชื่อสัตว์ แสดงรายชื่อสัตว์เลี้ยงที่ทำการลงทะเบียนกับระบบ



รูปที่ 3.21 ออกแบบหน้าจอในส่วนของหน้า Pet

เมื่อทำการเลือกหน้า Pet จากเมนูหลักจะมีการแสดงรายชื่อของสัตว์เลี้ยงทั้งหมดของระบบ และมีปุ่ม ADD สำหรับเพิ่มสัตว์เลี้ยงเข้าสู่ระบบ

4) หน้าเพิ่มสัตว์เลี้ยง ทำการเพิ่มสัตว์เลี้ยงเข้าสู่ระบบ

รูปที่ 3.22 ออกแบบหน้าจอในส่วนของหน้าเพิ่มสัตว์เลี้ยง

เมื่อกดปุ่ม ADD เข้ามาจะมีแบบฟอร์มให้กรอกข้อมูลสัตว์เลี้ยง เมื่อต้องการบันทึกให้กดปุ่ม SAVE หากต้องการยกเลิกการทำงานให้กดปุ่ม CANCEL

5) หน้าข้อมูลสัตว์เลี้ยง แสดงรายละเอียดของสัตว์เลี้ยง และทำการลบสัตว์เลี้ยงออกจากระบบ



รูปที่ 3.23 ออกแบบหน้าจอในส่วนของการแสดงข้อมูลสัตว์เลี้ยง

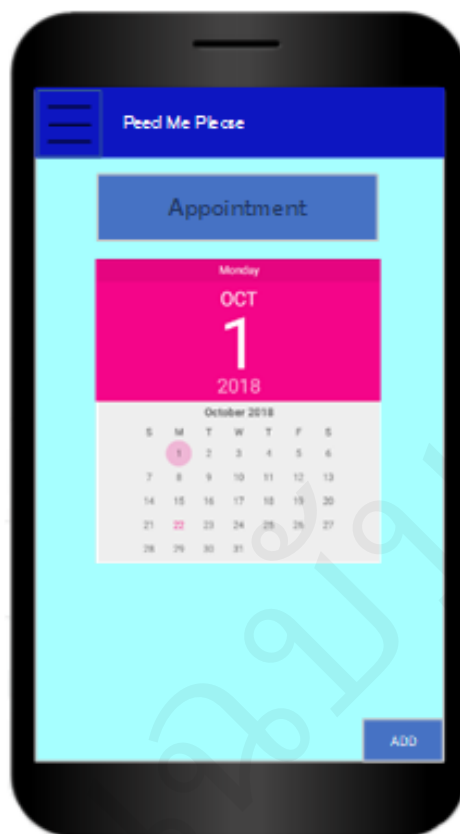
เมื่อกดเลือกชื่อสัตว์เลี้ยงจากหน้า Pet จะมีการแสดงข้อมูลสัตว์เลี้ยงตัวที่เลือก หากต้องการแก้ไขข้อมูลให้กดปุ่ม EDIT หากต้องการลบสัตว์เลี้ยงออกจากระบบก็ให้ทำการกดปุ่ม DELETE

6) หน้าแก้ไขข้อมูลสัตว์เลี้ยง ทำการแก้ไขข้อมูลของสัตว์เลี้ยง

รูปที่ 3.24 ออกแบบหน้าจอในส่วนของหน้าแก้ไขข้อมูลสัตว์เลี้ยง

เมื่อกดปุ่ม EDIT เข้ามาจะมีการแสดงข้อมูลต่างของสัตว์ และสามารถแก้ไขข้อมูลได้ หากต้องการบันทึกข้อมูลสัตว์เลี้ยงให้กดปุ่ม SAVE หากต้องการยกเลิกการเปลี่ยนแปลงให้กดปุ่ม CANCEL

7) หน้าตารางปฏิทิน แสดงรายละเอียดวันที่และการนัดหมาย



รูปที่ 3.25 ออกแบบหน้าจอในส่วนของหน้า Appointment

เมื่อทำการเลือกหน้า Appointment จากเมนูหลัก จะมีการแสดงปฏิทินที่มีการนัดหมายไว้ และหากต้องการเพิ่มการนัดหมายให้กดปุ่ม ADD

8) หน้าเพิ่มการนัดหมาย ทำการบันทึกรายละเอียดการนัดหมาย

รูปที่ 3.26 ออกแบบหน้าจอในส่วนของหน้าเพิ่มการนัดหมาย

เมื่อกดปุ่ม ADD เข้ามาแล้วจะมีแบบฟอร์มเพิ่มตารางการนัดหมายให้กรอกข้อมูล หากต้องการบันทึกให้กดปุ่ม SAVE

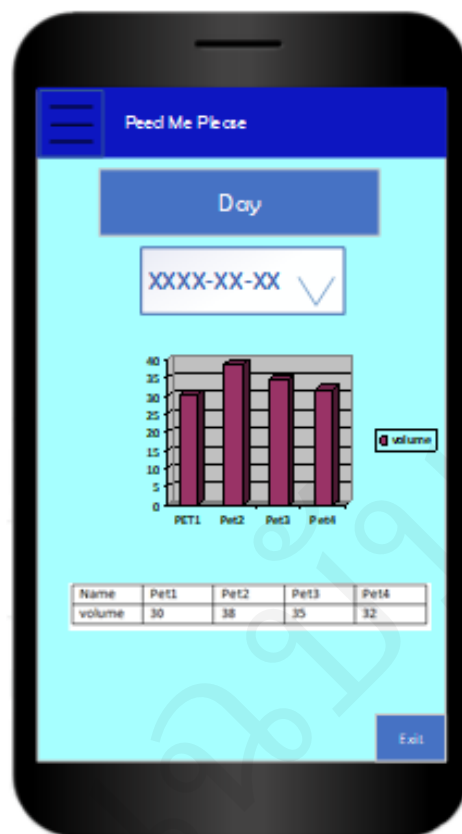
9) หน้าสถิติการกิน



รูปที่ 3.27 ออกแบบหน้าจอในส่วนของหน้า Stat

เมื่อทำการเลือกหน้า Appointment จากเมนูหลักจะมีการแสดงเมนูของสถิติ
ในแบบของรายวัน รายอาทิตย์ และรายเดือน

10) หน้าแสดงข้อมูลสถิติ



รูปที่ 3.28 ออกแบบหน้าจอในส่วนของหน้าแสดงข้อมูลสถิติ

เมื่อกดเข้ามาจากเมนูหน้า Stat หน้านี้จะมีการแสดงข้อมูลสถิติในรูปแบบกราฟ และรูปแบบตารางข้อมูล

11) หน้าการปล่อยอาหาร



รูปที่ 3.29 ออกแบบหน้าจอในส่วนของหน้า Food

เมื่อทำการเลือกหน้า Food จากเมนูหลักเข้ามาหน้านี้นี้จะมีการแสดงเมนูการตั้งค่าการให้อาหาร ซึ่งมีทั้งหมด 2 รูปแบบให้เลือก

12) หน้าตั้งค่าการป้อนอาหารรูปแบบที่ 1

รูปที่ 3.30 ออกแบบหน้าจอในส่วนของหน้ารูปแบบที่ 1

เมื่อทำการเลือกรูปแบบการให้อาหารรูปแบบที่ 1 หน้านี้จะมีแบบฟอร์มสำหรับการตั้งค่าการให้อาหารที่ต้องระบุปริมาณอาหารคงเหลือในถาดอาหาร และปริมาณอาหารที่ต้องการให้

13) หน้าเมนูการปล่อยอาหารรูปแบบที่ 2

The screenshot shows a mobile application interface titled "Feed Me Please". Below the title bar, there is a section labeled "Foodtype 2". This section contains three numbered entries, each with a "Time" field and a "Food" field. The fields are pre-filled with placeholder text "xx:xx:xx" for time and "xx g" for food amount. A "Save" button is located at the bottom right of the form.

Section	Time	Food
No.1	xx:xx:xx	xx g
No.2	xx:xx:xx	xx g
No.3	xx:xx:xx	xx g

รูปที่ 3.31 ออกแบบหน้าจอในส่วนของหน้ารูปแบบที่ 2

เมื่อทำการเลือกรูปแบบการให้อาหารรูปแบบที่ 2 หน้านี้จะมีแบบฟอร์มสำหรับการตั้งค่าการให้อาหาร ซึ่งต้องระบุเวลา และปริมาณอาหารที่ต้องการให้

ตารางที่ 3.2 ตาราง pet (ต่อ)

No.	Attribute Name	Description	Data Type	Key Type	Constraints	Reference
3	birth_day	วันเกิด	DATE			
4	pet_type	ชนิดสัตว์เลี้ยง	VARCHAR (30)		Not Null	
5	disease	โรคประจำตัว	VARCHAR (40)			
6	rfid	รหัส RFID	VARCHAR (8)		Not Null	

ตาราง pet เป็นตารางที่เก็บข้อมูลเกี่ยวกับรายละเอียดของสัตว์เลี้ยง ประกอบไปด้วยคอลัมน์ต่าง ๆ ดังนี้

- 1) pet_id ใช้เก็บข้อมูลรหัสสัตว์เลี้ยง เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 1 ตัว
- 2) pet_name ใช้เก็บข้อมูลชื่อของสัตว์เลี้ยง เป็นข้อมูลชนิด VARCHAR เก็บข้อมูลเป็นตัวอักษร ได้ 20 ตัวอักษร
- 3) birth_day ใช้เก็บข้อมูลวันเกิดของสัตว์เลี้ยง เป็นข้อมูลชนิด DATE
- 4) pet_type ใช้เก็บข้อมูลชนิดของสัตว์เลี้ยง เป็นข้อมูลชนิด VARCHAR เก็บข้อมูลเป็นตัวอักษร ได้ 30 ตัวอักษร
- 5) disease ใช้เก็บข้อมูลเกี่ยวกับโรคประจำตัวของสัตว์เลี้ยง เป็นข้อมูลชนิด VARCHAR เก็บข้อมูลเป็นตัวอักษร ได้ 40 ตัวอักษร
- 6) rfid ใช้เก็บข้อมูลรหัสอาร์เอฟไอดีประจำตัวสัตว์เลี้ยง เป็นข้อมูลชนิด VARCHAR เก็บข้อมูลเป็นตัวอักษร ได้ 8 ตัวอักษร

ตารางที่ 3.3 ตาราง select_type

No.	Attribute Name	Description	Data Type (Size)	Key Type	Constraints	Reference
1	select_type_id	รหัสเลือกรูปแบบ	INT (1)	PK	Not Null	
2	type	รูปแบบที่เลือก	INT (1)		Not Null	

ตาราง select_type เป็นตารางที่เก็บข้อมูลเกี่ยวกับรูปแบบการให้อาหารที่ผู้ใช้งานเลือก ประกอบด้วยคอลัมน์ต่าง ๆ ดังนี้

1) select_type_id ใช้เก็บข้อมูลรหัสเลือกรูปแบบ เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 1 ตัว

2) type ใช้เก็บข้อมูลรหัสรูปแบบที่เลือก เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 1 ตัว

ตารางที่ 3.4 ตาราง feed_type1

No.	Attribute Name	Description	Data Type	Key Type	Constraints	Reference
1	type1_id	รหัสการปล่อยอาหาร รูปแบบที่1	INT (1)	PK	Not Null	
2	t1_quantity	ปริมาณอาหารที่ต้องการ ปล่อย	INT (4)		Not Null	
3	t1_weight	ปริมาณอาหารที่เหลือใน ถาด	INT (4)		Not Null	

ตาราง feed_type1 เป็นตารางที่เก็บข้อมูลเกี่ยวกับรายละเอียดของการปล่อยอาหารในรูปแบบที่ 1 เป็นการปล่อยตามปริมาณคงเหลือในถาด ประกอบไปด้วยคอลัมน์ต่าง ๆ ดังนี้

1) type1_id ใช้เก็บข้อมูลรหัสการปล่อยอาหารรูปแบบที่1 เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 1 ตัว

2) t1_quantity ใช้เก็บข้อมูลเกี่ยวกับปริมาณอาหารที่ต้องการปล่อย เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 4 ตัว

3) t1_weight ใช้เก็บข้อมูลปริมาณอาหารที่เหลือในถาด เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 4 ตัว

ตารางที่ 3.5 ตาราง feed_type2

No.	Attribute Name	Description	Data Type	Key Type	Constraints	Reference
1	type2_id	รหัสการปล่อยอาหารรูปแบบที่ 2	INT (1)	PK	Not Null	
2	t2_quantity	ปริมาณอาหารที่ต้องการปล่อย	INT (4)		Not Null	
3	t2_time	เวลาปล่อยอาหาร	TIME		Not Null	

ตาราง feed_type2 เป็นตารางที่เก็บข้อมูลเกี่ยวกับรายละเอียดของการปล่อยอาหารในรูปแบบที่ 2 เป็นการปล่อยตามปริมาณและเวลาที่ตั้งไว้ ประกอบไปด้วยคอลัมน์ต่าง ๆ ดังนี้

- 1) type2_id ใช้เก็บข้อมูลรหัสการปล่อยอาหารรูปแบบที่ 2 เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 1 ตัว
- 2) t2_quantity ใช้เก็บข้อมูลปริมาณอาหารที่ต้องการปล่อย เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 4 ตัว
- 3) t2_time ใช้เก็บข้อมูลเวลาปล่อยอาหาร เป็นข้อมูลชนิด TIME

ตารางที่ 3.6 ตาราง food_alert

No.	Attribute Name	Description	Data Type	Key Type	Constraints	Reference
1	food_alert_id	รหัสแจ้งเตือนอาหารใกล้หมดถัง	INT (5)	PK	Not Null, Auto increment	
2	dt_alert	วันเวลาที่แจ้งเตือน	DATETIME		Not Null	

ตาราง food_alert เป็นตารางที่เก็บข้อมูลการแจ้งเตือนเมื่ออาหารในถังใกล้หมด ประกอบด้วยคอลัมน์ต่าง ๆ ดังนี้

- 1) food_alert_id ใช้เก็บรหัสการแจ้งเตือนอาหารใกล้หมดถึง เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข 5 ตัว
- 2) dt_alert เก็บวันที่แจ้งเตือน เป็นข้อมูลชนิดวันเวลา

ตารางที่ 3.7 ตาราง date_alert

No.	Attribute Name	Description	Data Type	Key Type	Constraints	Reference
1	date_alert_id	รหัสนัดหมาย	INT (5)	PK	Not Null, Auto increment	
2	information	ข้อมูลการนัดหมาย	VARCHAR (40)		Not Null	
3	date	วันทีนัดหมาย	DATE		Not Null	

ตาราง date_alert เป็นตารางที่เก็บข้อมูลเกี่ยวกับรายละเอียดของตารางนัดหมาย ประกอบด้วยคอลัมน์ต่าง ๆ ดังนี้

- 1) date_alert_id ใช้เก็บข้อมูลรหัสการนัดหมาย เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลขได้ 5 ตัว
- 2) information ใช้เก็บข้อมูลการนัดหมาย เป็นข้อมูลชนิด VARCHAR เก็บข้อมูลเป็นตัวอักษรได้ 40 ตัวอักษร
- 3) date ใช้เก็บข้อมูลเกี่ยวกับวันทีนัดหมาย เป็นข้อมูลชนิด DATE

ตารางที่ 3.8 ตาราง food_tray

No.	Attribute Name	Description	Data Type	Key Type	Constraints	Reference
1	food_tray_id	รหัสอาหารในถาด	INT (10)	PK	Not Null, Auto increment	
2	pet_id	รหัสสัตว์เลี้ยง	INT (1)	FK		pet
3	ft_quantity	ปริมาณอาหาร	INT (4)		Not Null	
4	ft_date_time	วันที่และเวลา	DATETIME		Not Null	

ตาราง food_tray เป็นตารางที่เก็บข้อมูลเกี่ยวกับปริมาณอาหารที่อยู่ในถาดและสัตว์เลี้ยงที่ใกล้เคียง ประกอบด้วยคอลลัมน์ต่าง ๆ ดังนี้

- 1) food_tray_id ใช้เก็บข้อมูลรหัสอาหารในถาด เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 10 ตัว
- 2) pet_id ใช้เก็บข้อมูลรหัสสัตว์เลี้ยง เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 1 ตัว
- 3) ft_quantity ใช้เก็บข้อมูลปริมาณอาหาร เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 4
- 4) ft_date_time ใช้เก็บข้อมูลวันที่และเวลา เป็นข้อมูลชนิด DATETIME

ตารางที่ 3.9 ตาราง tray

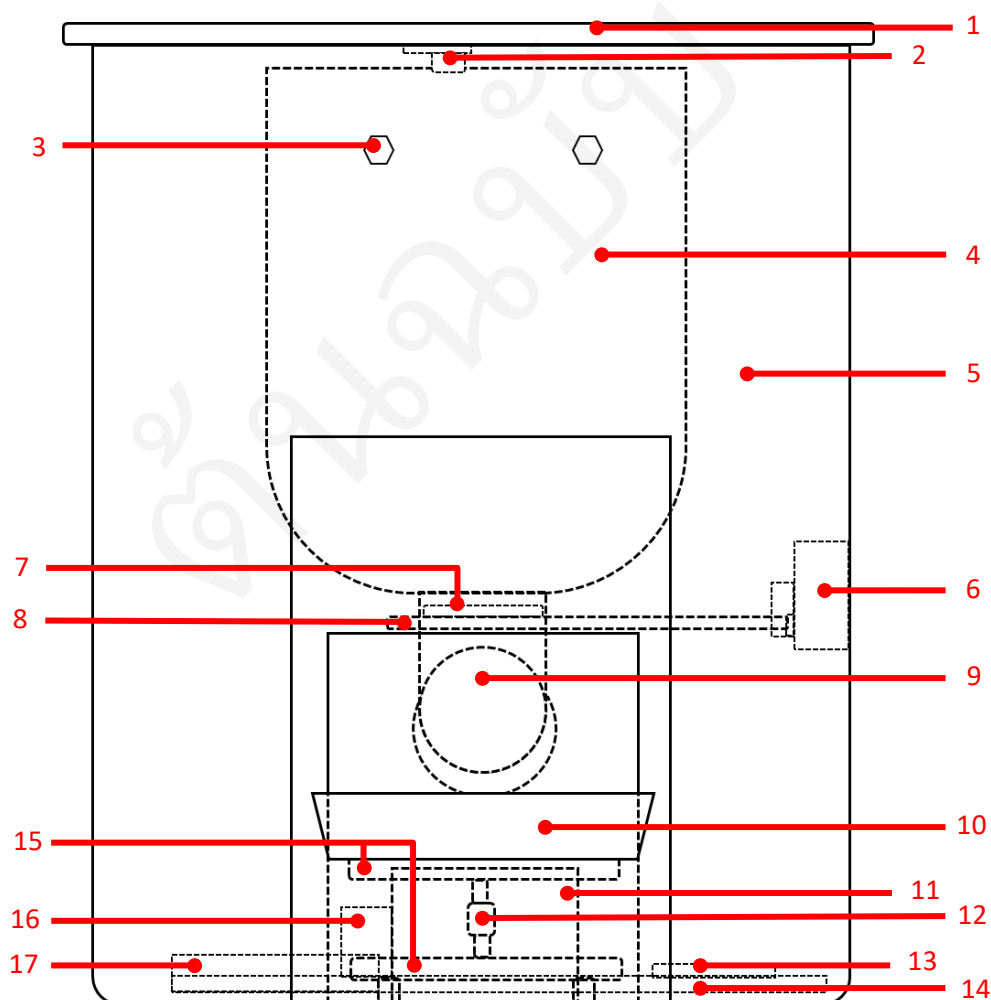
No.	Attribute Name	Description	Data Type	Key Type	Constraints	Reference
1	tray_id	รหัสอาหารในถาด	INT (4)	PK	Not Null, Auto increment	
2	pet_id	รหัสสัตว์เลี้ยง	INT (1)	FK	Not Null	pet
3	t_quantity	ปริมาณอาหาร	INT (4)		Not Null	
4	t_date	วันที่	DATE		Not Null	

ตาราง tray เป็นตารางที่เก็บข้อมูลเกี่ยวกับปริมาณอาหารที่สัตว์เลียกิน ประกอบไปด้วยคอลัมน์ต่าง ๆ ดังนี้

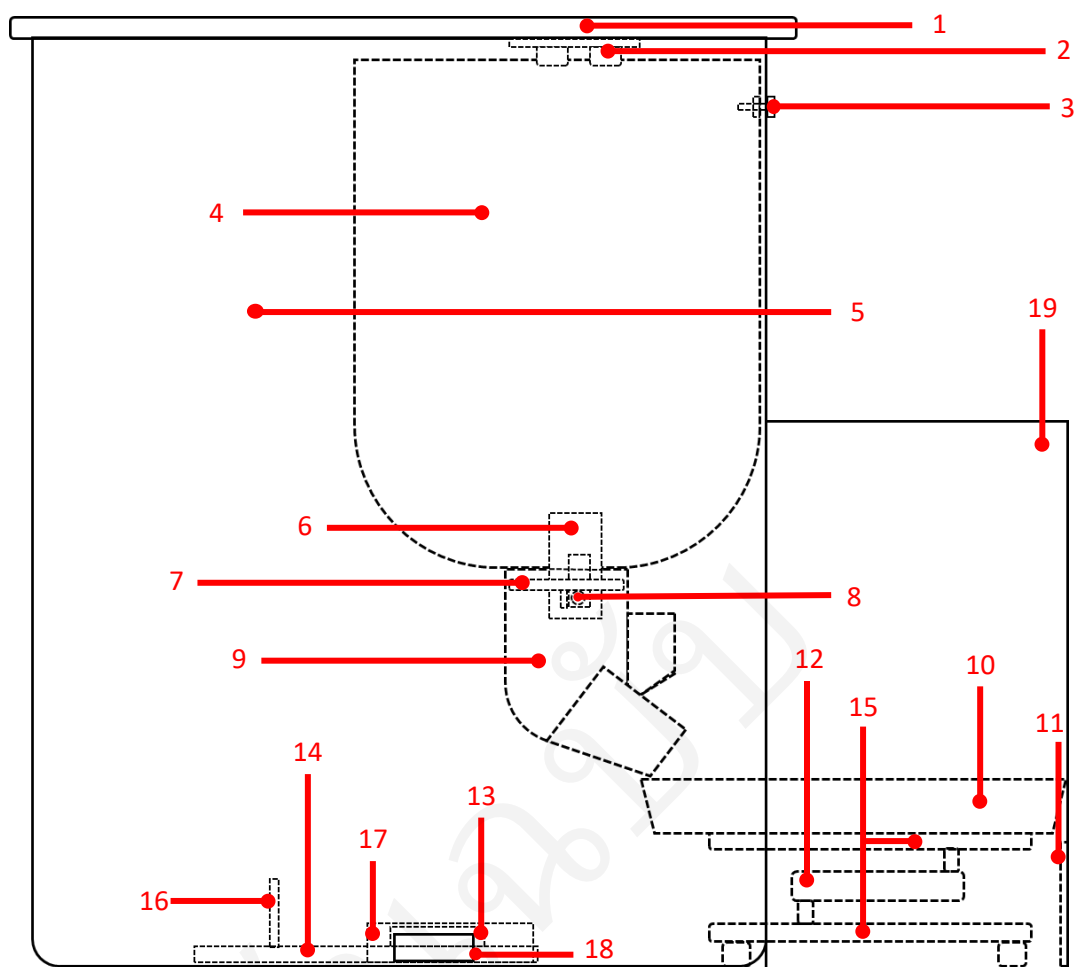
- 1) tray_id ใช้เก็บข้อมูลรหัสอาหารในถาด เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 4 ตัว
- 2) pet_id ใช้เก็บข้อมูลรหัสสัตว์เลีย เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 1 ตัว
- 3) t_quantity ใช้เก็บข้อมูลปริมาณอาหาร เป็นข้อมูลชนิด INT เก็บข้อมูลเป็นตัวเลข ได้ 4 ตัว
- 4) t_date ใช้เก็บข้อมูลวันที่ และเวลา เป็นข้อมูลชนิด DATE

3.8 การออกแบบอุปกรณ์

3.8.1 โครงสร้างเครื่องให้อาหาร



รูปที่ 3.33 โครงสร้างเครื่องให้อาหาร ด้านหน้า

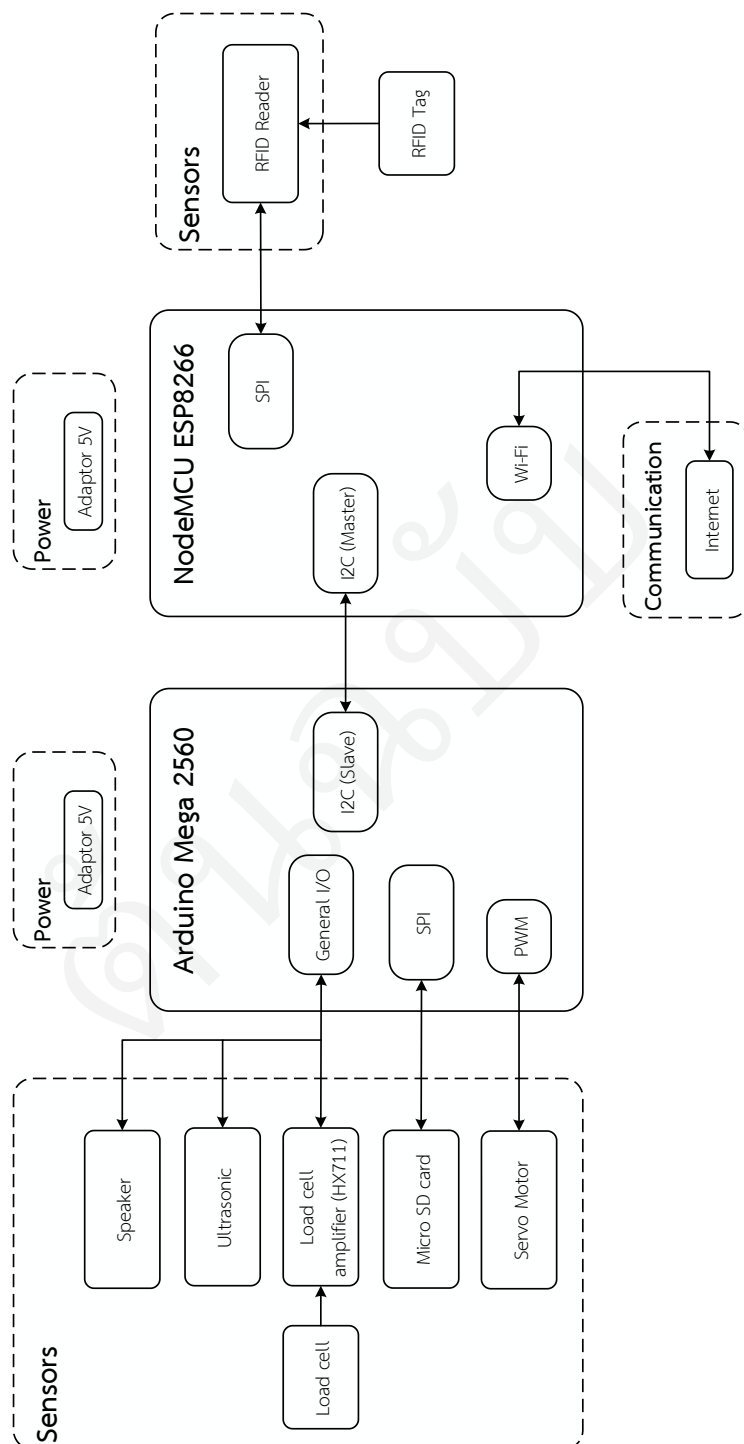


รูปที่ 3.34 โครงสร้างเครื่องให้อาหาร ด้านข้าง

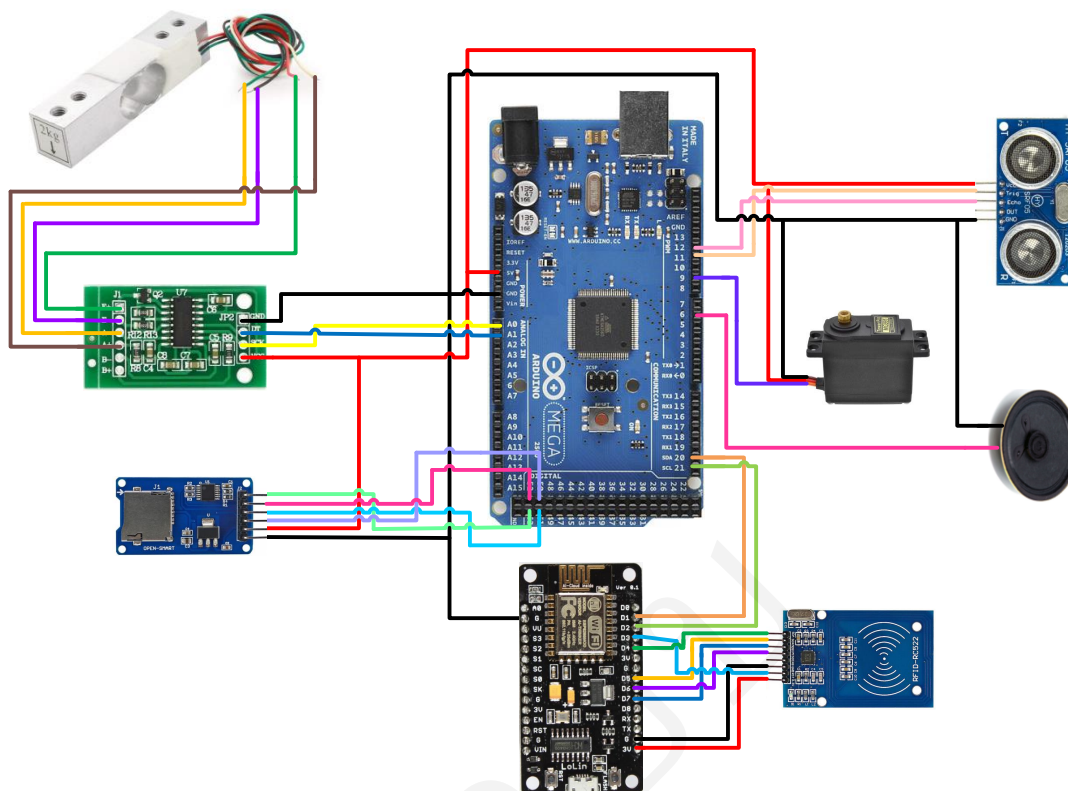
ตารางที่ 3.10 รายละเอียดของโครงสร้างเครื่องให้อาหาร

ลำดับที่	รายละเอียด
1	แผ่นอะคริลิกฝ้าถึง วงกลมหนา 4 มม. ขนาดเส้นผ่าศูนย์กลาง 40 ซม.
2	เซนเซอร์วัดระยะ อัลตราโซนิก รุ่น HY-SRF05
3	น็อตหกเหลี่ยม ขนาด 5 มม.
4	ถังน้ำขนาด 6 ลิตร ดัดแปลงเป็นถังบรรจุอาหาร
5	ถังพลาสติกโครงตัวเครื่อง ทรงทอกลม สูง 35.5 ซม. กว้าง 28 ซม.
6	ดีซีเกียร์เซอร์โวมอเตอร์ รุ่น Tower Pro MG995
7	แผ่นพลาสติก (ตัดเป็นรูปวงกลม สำหรับเป็นตัวเปิด-ปิดการให้อาหาร)
8	ปลอกปากกา (สำหรับทำแกนหมุน)
9	ท่อพีวีซี 90 องศา เส้นผ่าศูนย์กลาง 1.5 นิ้ว (1 นิ้ว 2 หุน)
10	ถาดอาหาร
11	เครื่องอ่านอาร์เอฟไอดี
12	เซนเซอร์ตรวจวัดน้ำหนักกรองรับน้ำหนักไม่เกิน 2 กิโลกรัม
13	ไมโครคอนโทรลเลอร์ รุ่น NodeMCU ESP8266
14	โปรโตบอร์ด
15	แผ่นอะคริลิกประกอบกับโพลีคาร์บอเนต หนา 4 มม. ยาว 12 ซม. กว้าง 10 ซม.
16	Micro SD card Adapter (Catalex)
17	ไมโครคอนโทรลเลอร์ รุ่น Arduino Mega R3
18	ช่องสำหรับใส่สายไฟ กว้าง 1 ซม. ยาว 3 ซม.
19	อะคริลิกสำหรับทำเป็นอุโมงค์ วางโพลีคาร์บอเนต ถาดอาหาร และเครื่องอ่านอาร์เอฟไอดี หนา 4 มม. สูง 20 ซม. กว้าง 14 ซม. ลึก 14 ซม.

3.8.2 ภาพรวมการเชื่อมต่ออุปกรณ์



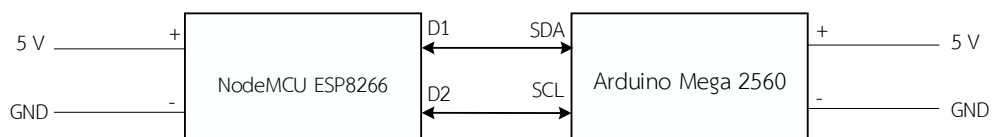
รูปที่ 3.35 Hardware architecture ภาพรวมอุปกรณ์



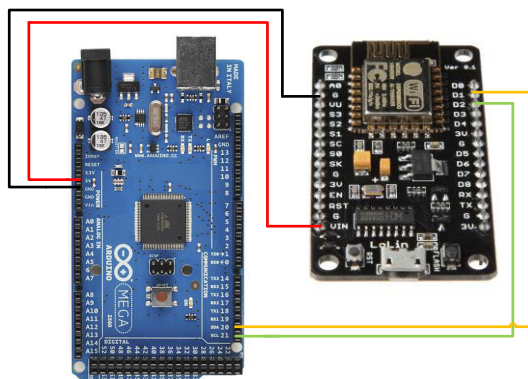
รูปที่ 3.36 ภาพรวมการประกอบวงจรเครื่องให้อาหาร

การเชื่อมต่อวงจรโดยรวมประกอบไปด้วย วงจรตรวจจับน้ำหนัก, วงจรควบคุมเซอร์โวมอเตอร์ปล่อยอาหาร, วงจรควบคุมอัลตราโซนิกตรวจจับระยะ, วงจรควบคุมการเล่นเสียง และวงจรควบคุมเครื่องอ่านอาร์เอฟไอดี หลักการทำงานคือ NodeMCU ติดต่อกับฐานข้อมูลผ่านอินเทอร์เน็ตเป็นตัวควบคุมวงจรเครื่องอ่านอาร์เอฟไอดี และเป็นมาสเตอร์ (Master) สั่งให้ Arduino Mega ซึ่งเป็นสเลฟ (Slave) ส่งคำสั่งทำงานไปยังอุปกรณ์อื่น ๆ อีกต่อหนึ่ง โดยมีหลักการทำงานแต่ละวงจรดังต่อไปนี้

3.8.3 วงจรรับ-ส่งข้อมูลระหว่าง Master (NodeMCU) และ Slave (Arduino Mega) แบบ I2C



รูปที่ 3.37 วงจรรับ-ส่งข้อมูลระหว่าง Master (NodeMCU) และ Slave (Arduino Mega) แบบ I2C



รูปที่ 3.38 การประกอบวงจรรับ-ส่งข้อมูลแบบ I2C

ตารางที่ 3.11 การต่อขาสัญญาณวงจร I2C

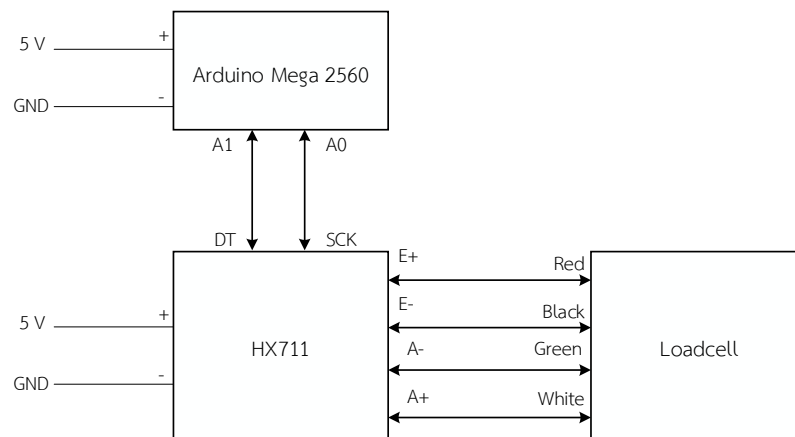
เส้นสี	Arduino Mega 2560	NodeMCU ESP8266
ส้ม	D1	SDA
เขียว	D2	SCL
แดง	5V	VIn
ดำ	GND	GND

หลักการทำงานของวงจรรับ-ส่งข้อมูลแบบ I2C คือ เมื่อไมโครคอนโทรลเลอร์ หรือก็คือ NodeMCU ส่งคำสั่งไปยังสเลฟ หรือก็คือ Arduino Mega ให้ทำงาน หรือส่งข้อมูลกลับมาให้ไมโครคอนโทรลเลอร์

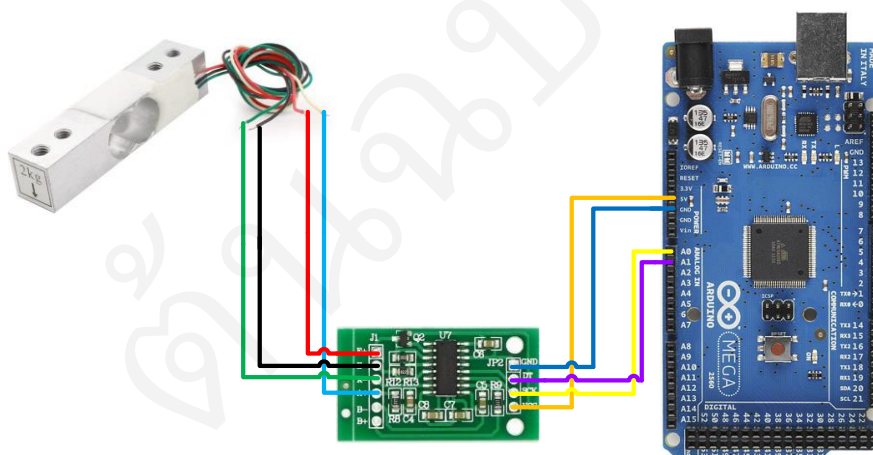
คำสั่งที่ NodeMCU ส่งไปยัง Arduino Mega ได้แก่

- 1) เมื่อต้องการทราบค่าระยะห่างระหว่างฝ่าถ้ง-อาหาร (หน่วยเป็นเซนติเมตร) โดยส่งคำสั่งทำงานไปยัง Ultrasonic เมื่อ Arduino Mega ได้รับค่าระยะแล้ว จะส่งค่านั้นไปยัง NodeMCU
- 2) เมื่อต้องการทราบค่าน้ำหนักอาหารในถาด (หน่วยเป็นกรัม) โดย Arduino Mega รับค่าน้ำหนักจาก Load Cell แล้วส่งค่านั้นไปยัง NodeMCU
- 3) เมื่อต้องการให้เซอร์โวมอเตอร์ทำการหมุนเพื่อปล่อยอาหารตามปริมาณอาหารที่กำหนด
- 4) เมื่อต้องการให้เล่นเสียง
- 5) เมื่อฟังก์ชันตรวจสอบหาสัตว์เลี้ยงที่มากินอาหารน้อยกว่า 2 ครั้ง จะส่งไอดีของสัตว์เลี้ยงตัวนั้น ๆ ไปยัง Arduino Mega เพื่อให้เล่นเสียงประจำตัวสัตว์เลี้ยง

3.8.4 วงจรตรวจวัดน้ำหนัก



รูปที่ 3.39 วงจรตรวจวัดน้ำหนัก



รูปที่ 3.40 การประกอบวงจรตรวจวัดน้ำหนัก

ตารางที่ 3.12 การต่อขาสัญญาณวงจรตรวจวัดน้ำหนัก

เส้นสี	Loadcell 2 Kg	Loadcell Amplifier HX711	Arduino Mega 2560
เขียว	เขียว	E+	-
ดำ	ดำ	E-	-
แดง	แดง	A-	-

ตารางที่ 3.12 การต่อขาสัญญาณวงจรตรวจวัดน้ำหนัก (ต่อ)

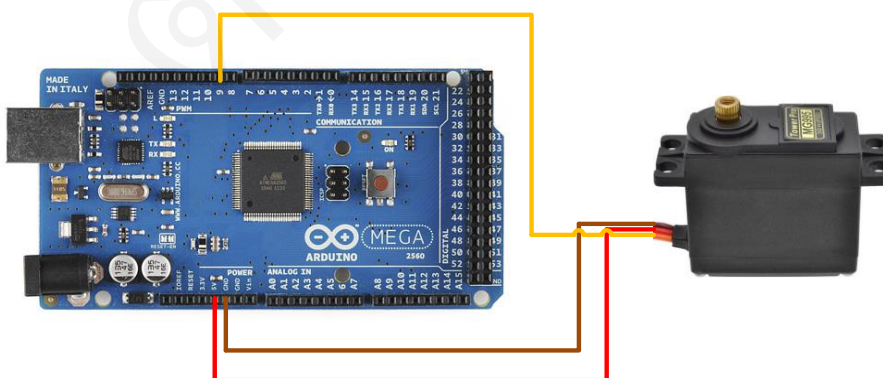
ฟ้า	ขาว	A+	-
ม่วง	-	DT	A1
เหลือง	-	SCK	A0
ส้ม	-	VCC	5V
น้ำเงิน	-	GND	GND

หลักการทำงานคือ เมื่อ Load Cell ได้รับค่าจากแรงกด หรือก็คือน้ำหนักจากถาดอาหาร จะแปลงค่านั้นเป็นสัญญาณทางไฟฟ้า HX711 จะทำการขยายค่าเพื่อให้ Arduino Mega สามารถนำไปใช้ได้

3.8.5 วงจรควบคุมเซอร์โวมอเตอร์ปล่อยอาหาร



รูปที่ 3.41 วงจรควบคุมเซอร์โวมอเตอร์ปล่อยอาหาร



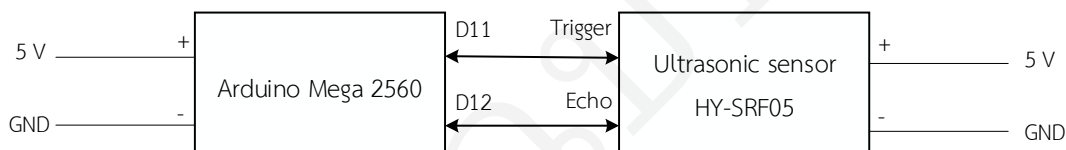
รูปที่ 3.42 การประกอบวงจรควบคุมเซอร์โวมอเตอร์ปล่อยอาหาร

ตารางที่ 3.13 การต่อขาสัญญาณวงจรควบคุมเซอร์โวมอเตอร์ปล่อยอาหาร

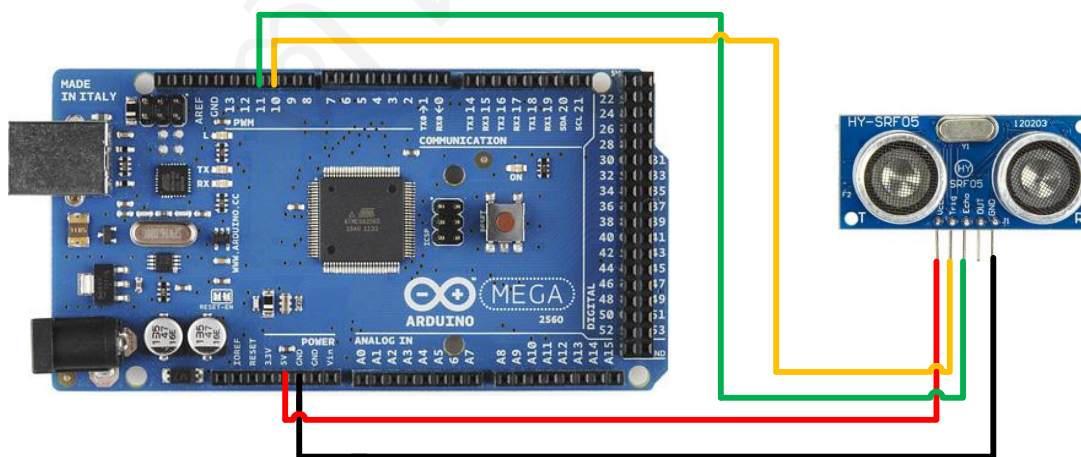
เส้นสี	Arduino Mega 2560	Servo motor
น้ำตาล	GND	น้ำตาล
แดง	5V	แดง
ส้ม	D9	ส้ม

หลักการทำงานคือ เมื่อเซอร์โวมอเตอร์ได้รับคำสั่งให้ทำงานจาก Arduino Mega จะทำการเปรียบเทียบค่าองศาสุดท้ายที่หมุนไปหากเป็น 0 องศา จะหมุนไปยัง 90 องศา แต่หากเป็น 90 องศา จะหมุนไปยัง 0 องศา เพื่อปล่อยอาหารตามปริมาณที่กำหนดไว้

3.8.6 วงจรควบคุมอัลตราโซนิกตรวจวัดระยะ



รูปที่ 3.43 วงจรควบคุมอัลตราโซนิกตรวจวัดระยะ



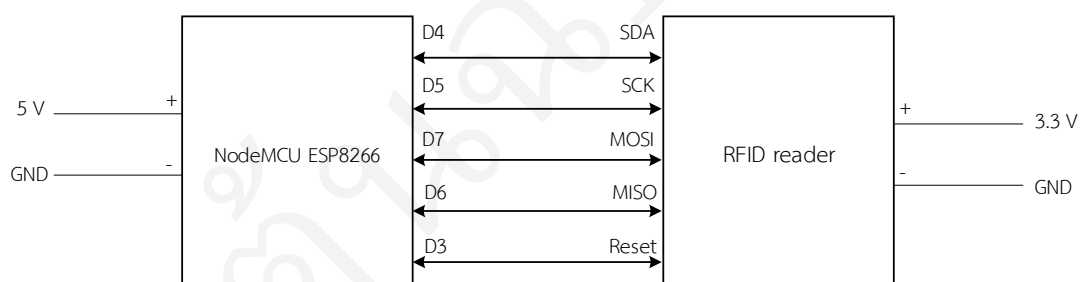
รูปที่ 3.44 การประกอบวงจรควบคุมอัลตราโซนิกตรวจวัดระยะ

ตารางที่ 3.14 การต่อขาสัญญาณวงจรควบคุมอัลตราโซนิกตรวจวัดระยะ

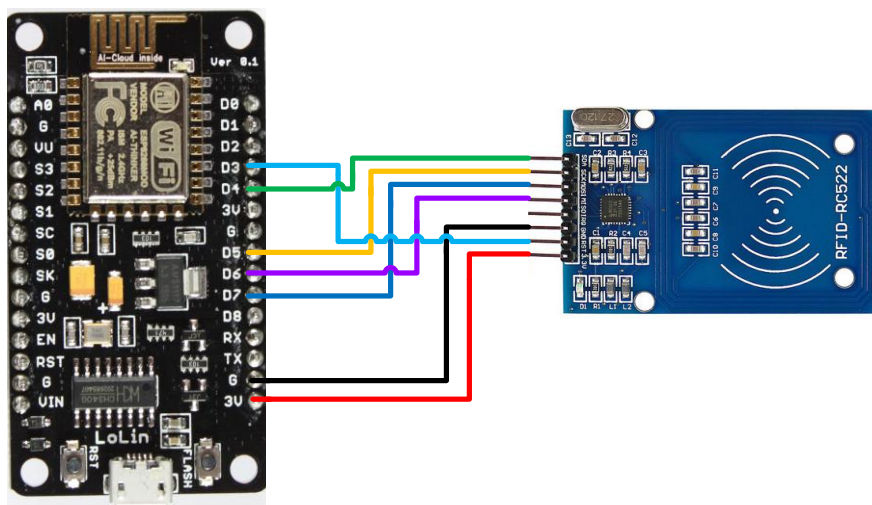
เส้นสี	Arduino Mega 2560	Ultrasonic
แดง	5V	VCC
ส้ม	D11	Trigger
เขียว	D12	Echo
ดำ	GND	GND

หลักการทำงานคือ NodeMCU สั่งการทำงานผ่าน Arduino Mega เมื่ออัลตราโซนิกได้รับคำสั่งให้ทำงาน ตัวส่งคลื่น (Trigger) จะสร้างคลื่นเสียงออกไป เมื่อไปกระทบอาหารในถังคลื่นจะถูกสะท้อนกลับมายังตัวรับ (Echo) แล้วประมวลผลเพื่อให้ได้ค่าระยะห่างในถัง Arduino Mega จะทำการส่งค่าระยะห่างนั้นไปยัง NodeMCU เพื่อตรวจสอบว่าอาหารในถังใกล้หมดหรือยัง

3.8.7 วงจรควบคุมเครื่องอ่านอาร์เอฟไอดี



รูปที่ 3.45 วงจรควบคุมเครื่องอ่านอาร์เอฟไอดี



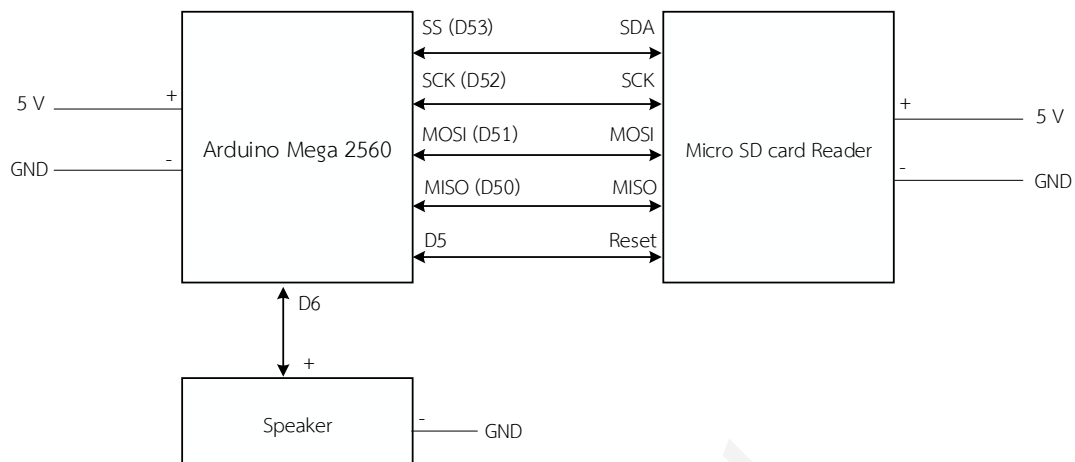
รูปที่ 3.46 การประกอบวงจรควบคุมเครื่องอ่านอาร์เอฟไอดี

ตารางที่ 3.15 การต่อขาสัญญาณวงจรควบคุมเครื่องอ่านอาร์เอฟไอดี

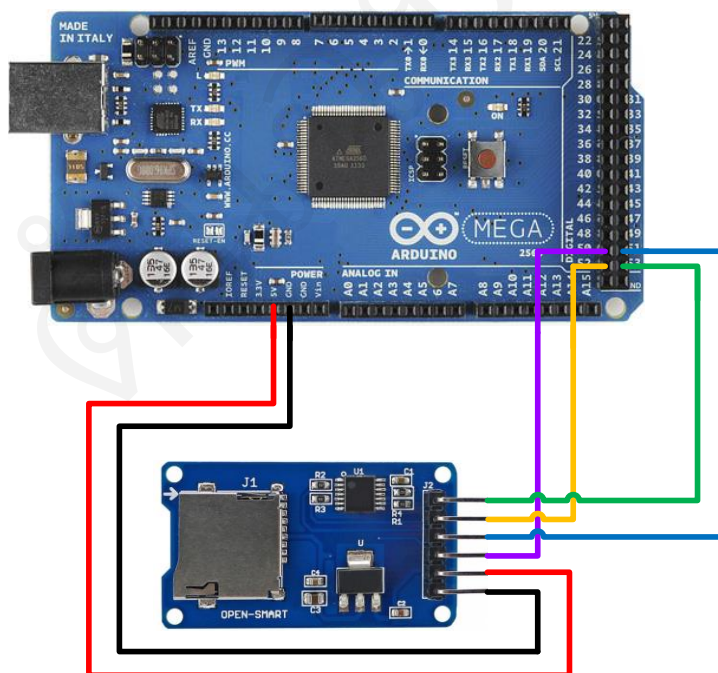
เส้นสี	RFID-RC522	NodeMCU ESP8266
เขียว	SDA	D4
ส้ม	SCK	D5
น้ำเงิน	MOSI	D7
ม่วง	MISO	D6
ดำ	GND	GND
ฟ้า	RST	D3
แดง	3.3V	3.3V

หลักการทำงานคือ เมื่อเครื่องอ่านอาร์เอฟไอดีได้รับค่าจากแทคซึ่งใช้ระบุตัวสัตว์เลี้ยง แล้วนั้น จะส่งรหัสของแทคนั้นไปยัง NodeMCU แล้ว NodeMCU จะทำการประมวลผลรหัสจากแทคเปรียบเทียบกับรหัสของสัตว์เลี้ยงว่าเป็นไอดีของสัตว์เลี้ยงตัวใด แล้วส่งข้อมูลไปบันทึกในฐานข้อมูล

3.8.8 วงจรควบคุมการเล่นเสียง



รูปที่ 3.47 วงจรควบคุมการเล่นเสียง



รูปที่ 3.48 การประกอบวงจรควบคุมการเล่นเสียง

ตารางที่ 3.16 การต่อขาสัญญาณวงจรควบคุมการเล่นเสียง

เส้นสี	Micro SD card Adapter (Catalex)	Arduino Mega 2560
เขียว	CS	D53 (SS)
ส้ม	SCK	D52 (SCK)
น้ำเงิน	MOSI	D51 (MOSI)
ม่วง	MISO	D50 (MISO)
แดง	VCC	5V
ดำ	GND	GND

หลักการทำงานคือ หลังจากที่ยังวงจรควบคุมเซอร์โวมอเตอร์ปล่อยอาหารทำงานแล้ว NodeMCU จะส่งคำสั่งให้ SD Card เล่นเสียงผ่านทาง Arduino Mega โดย SD Card จะแรนดอมไฟล์เสียง แล้ว Arduino Mega ส่งสัญญาณเสียงนั้นออกทางลำโพง

3.9 วัสดุ / อุปกรณ์

วัสดุ และอุปกรณ์ที่ใช้ในการจัดทำโครงงานเครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลี้ยงขนาดเล็ก มีรายละเอียดดังตารางที่ 3.17 นี้

ตารางที่ 3.17 รายการวัสดุและอุปกรณ์

ลำดับที่	รายการ	จำนวน
1	ไมโครคอนโทรลเลอร์ รุ่น Arduino Mega R3	1
2	ไมโครคอนโทรลเลอร์ รุ่น NodeMCU ESP8266	2
3	เซนเซอร์ตรวจจับน้ำหนักกรงรับน้ำหนักไม่เกิน 2 กิโลกรัม และ Load cell Amplifier-HX711	1
4	RFID card reader/Detractor Module kit (RC522)	1
5	RFID tag	2
6	ดีซีเกียร์เซอร์โวมอเตอร์ รุ่น Tower Pro MG995	1
7	เซนเซอร์วัดระยะ อัลตราโซนิก รุ่น HY-SRF05	1

ตารางที่ 3.17 รายการวัสดุและอุปกรณ์ (ต่อ)

8	Micro SD card Adapter (Catalex)	1
9	Micro SD card 1 GB	1
10	ลำโพง	1
11	สายไฟ Jumper ตัวผู้-ตัวเมีย, ตัวผู้-ตัวผู้, ตัวเมีย-ตัวเมีย	30
12	หม้อแปลง 9 โวลต์ 1 แอมป์	1
13	โปรโตบอร์ด	2
14	ถังน้ำขนาด 6 ลิตร ดัดแปลงเป็นถังบรรจุอาหาร	2
15	ท่อพีวีซี 90 องศา	1
16	แผ่นอะคริลิกปิดฝาถัง	1
17	ถาดอะคริลิก (สำหรับวางโหลดเซลล์ และ RFID card reader)	1
18	บานพับประตูขนาดเล็ก	2
19	พลาสติกโครงตัวเครื่อง	1
20	ปลอกปากกา (สำหรับทำแกนหมุน)	1
21	แผ่นพลาสติก (ตัดเป็นรูปวงกลม สำหรับเป็นตัวเปิด-ปิดอาหาร)	1

บทที่ 4

ผลการดำเนินงานและการวิเคราะห์

การจัดทำโครงงานเครื่องให้อาหารเมื่อดัตโนมิติสำหรับสัตว์เลี้ยวขนาดเล็ก เป็นการพัฒนาระบบเครื่องให้อาหารสัตว์ โดยมีการจัดทำในส่วนของเครื่องให้อาหารสัตว์ และส่วนของระบบการทำงานผ่านโมบายแอปพลิเคชัน จากผลการทดสอบพบว่าน่าพึงพอใจในระดับปานกลาง แต่ยั้งต้องพัฒนาส่วนอื่น ๆ เพิ่มเติม เพื่อให้ระบบมีประสิทธิภาพมากขึ้น

4.1 ผลการประกอบเครื่องให้อาหาร



รูปที่ 4.1 ภาพรวมเครื่องให้อาหาร

เครื่องให้อาหารประกอบไปด้วยวงจรรับ-ส่งข้อมูลแบบ I2C, วงจรตรวจจับน้ำหนัก, วงจรควบคุมเซอร์โวมอเตอร์ปล่อยอาหาร, วงจรควบคุมอัลตราโซนิกตรวจจับระยะ, วงจรควบคุมการเล่นเสียง และวงจรควบคุมเครื่องอ่านอาร์เอฟไอดี หลักการทำงานคือ NodeMCU ติดต่อกับฐานข้อมูลผ่านอินเทอร์เน็ตเป็น Master สั่งให้ Arduino Mega ซึ่งเป็น Slave ส่งคำสั่งทำงานไปยังอุปกรณ์อื่น ๆ อีกต่อหนึ่ง ในส่วนของรูปแบบการให้อาหารแบบที่ 1 คือ ให้อาหารเมื่อปริมาณอาหารในถาด เหลือปริมาณตามที่กำหนด วงจรตรวจจับน้ำหนักจะทำการส่งค่าน้ำหนักปัจจุบันไปยัง NodeMCU

เมื่อเปรียบเทียบเงื่อนไขน้ำหนักน้อยกว่าที่กำหนดไว้เป็นจริง วงจรเซอร์โวจึงจะทำงาน และในรูปแบบที่ 2 เมื่อ NodeMCU เปรียบเทียบเวลาที่กำหนดกับเวลาจริงตรงกันเป็นจริง วงจรเซอร์โวจะทำงานเพื่อให้อาหารตามเวลาในปริมาณที่กำหนด ทุกครั้งหลังจากเซอร์โวทำงานเสร็จสิ้น วงจรควบคุมการเล่นเสียงจะแรนดอมไฟล์เสียงเพื่อเล่นเสียง และวงจรตรวจสอบระยะจะทำงานเพื่อตรวจวัดระยะปริมาณอาหารในถัง หากถึงระยะที่กำหนดก็จะส่งข้อมูลการแจ้งเตือนไปยังแอปพลิเคชัน เมื่อ RFID Card Reader อ่านรหัสจาก RFID Tag ประจำตัวสัตว์เลี้ยง NodeMCU จะทำการส่งรหัสประจำตัวสัตว์เลี้ยงตัวนั้น และข้อมูลที่เกี่ยวข้องไปยังแอปพลิเคชัน

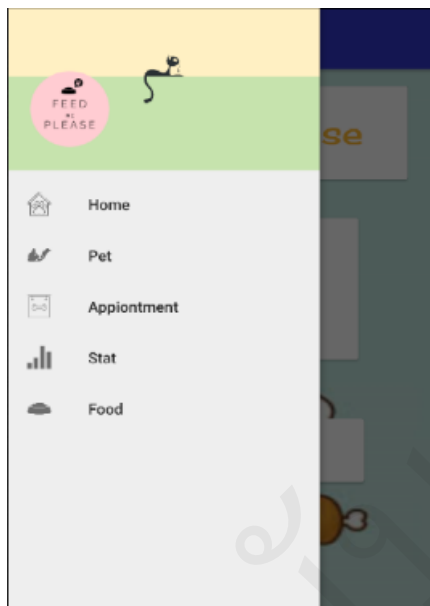
4.2 ผลการทดสอบการใช้งานโมบายแอปพลิเคชัน

4.2.1 หน้าแรก



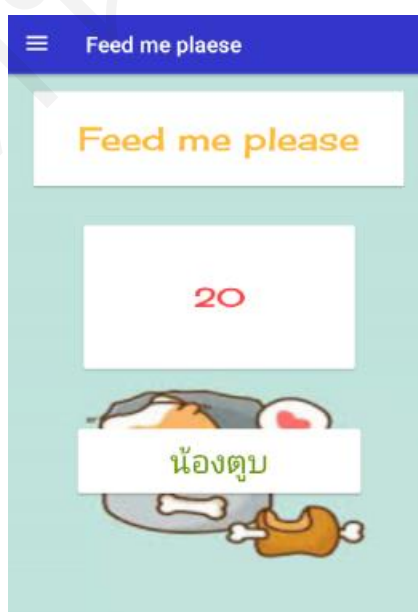
รูปที่ 4.2 หน้าแรกเมื่อเปิดเข้าใช้งานระบบ

4.2.2 หน้าเมนูหลัก แสดงข้อมูลต่าง ๆ เมื่อเข้าใช้งานระบบ โดยมีรูปเมเนื่อดังรูปที่ 4.1



รูปที่ 4.3 หน้าเมนูหลัก

4.2.3 หน้าหลัก



รูปที่ 4.4 หน้าหลักของระบบ

4.2.4 สัตว์เลี้ยง

ในรายการนี้จะเป็นการจัดการข้อมูลสัตว์เลี้ยง โดยจะมามีการทำงานทั้งหมด 4 ขั้นตอน ดังนี้

1) รายชื่อสัตว์เลี้ยง

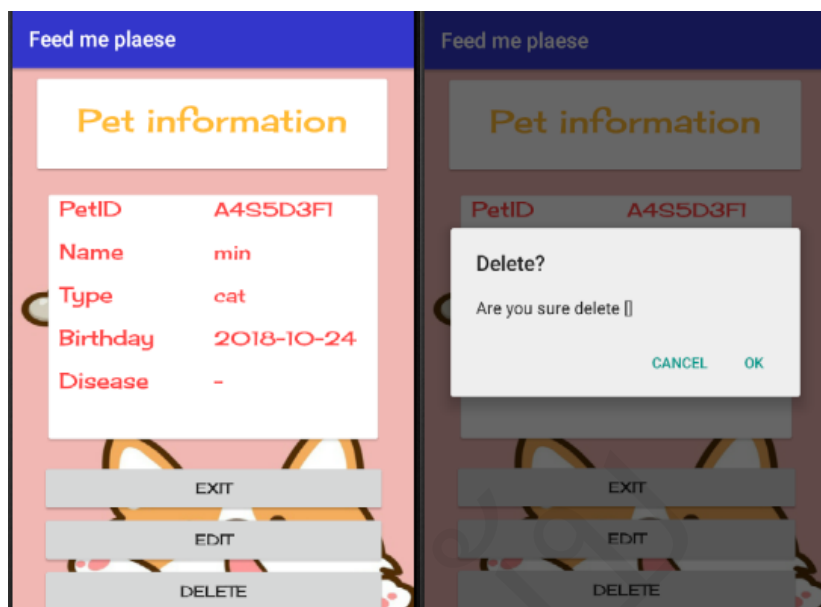


รูปที่ 4.5 หน้าแสดงรายชื่อสัตว์เลี้ยง

2) เพิ่มสัตว์เลี้ยง

รูปที่ 4.6 หน้าเพิ่มสัตว์เลี้ยง

3) แสดงข้อมูลสัตว์เลี้ยงและลบข้อมูล



รูปที่ 4.7 หน้าข้อมูลสัตว์เลี้ยง

4) แก้ไขข้อมูลสัตว์เลี้ยง



รูปที่ 4.8 หน้าแก้ไขข้อมูลสัตว์เลี้ยง

4.2.5 ข้อมูลตารางนัดหมาย

ในรายการนี้จะมีการแสดงข้อมูลและจัดการตารางนัดหมายของสัตว์เลี้ยง โดยจะมีการทำงานทั้งหมด 3 ขั้นตอน ดังนี้

1) แสดงตารางการนัดหมาย



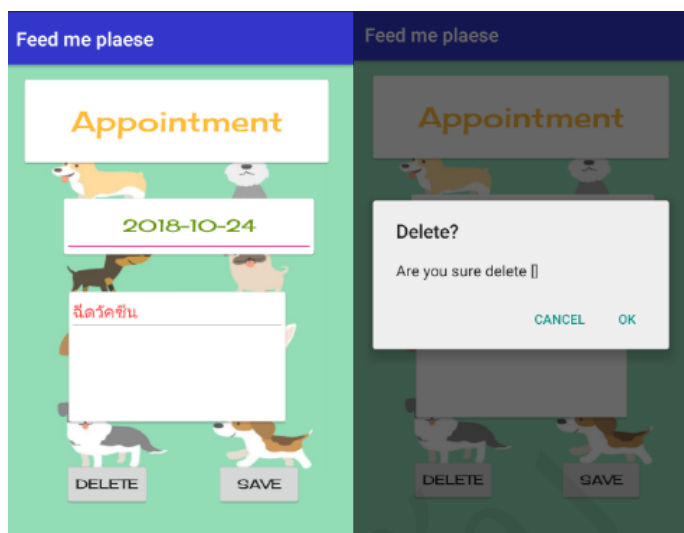
รูปที่ 4.9 หน้าแสดงตารางการนัดหมาย

2) เพิ่มการนัดหมาย



รูปที่ 4.10 หน้าเพิ่มการนัดหมาย

3) แก้ไขและลบข้อมูลการนัดหมาย

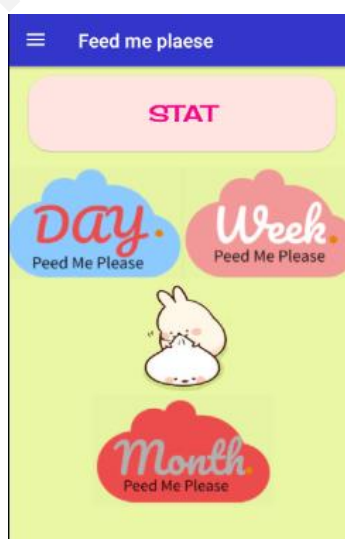


รูปที่ 4.11 แก้ไขแล้วลบข้อมูลการนัดหมาย

4.2.6 ข้อมูลสถิติ

ในรายการนี้จะมีการแสดงข้อมูลการกินอาหารของสัตว์เลี้ยง โดยจะมีการแสดงข้อมูลทั้งหมด 3 แบบ ดังนี้

1) เมนูรูปแบบการแสดงข้อมูลสัตว์เลี้ยง



รูปที่ 4.12 หน้าเมนูรูปแบบสถิติ

2) แสดงข้อมูลสถิติการกินอาหารของสัตว์เลี้ยงรายวัน



Feed me please

2018-11-11

นั่งดูบ

Date	quantity(g)
06.00	23
07.00	55
08.00	63
09.00	45
14.00	67

รูปที่ 4.13 หน้าแสดงข้อมูลสถิติรายวัน

3) แสดงข้อมูลสถิติการกินอาหารของสัตว์เลี้ยงรายอาทิตย์



Feed me please

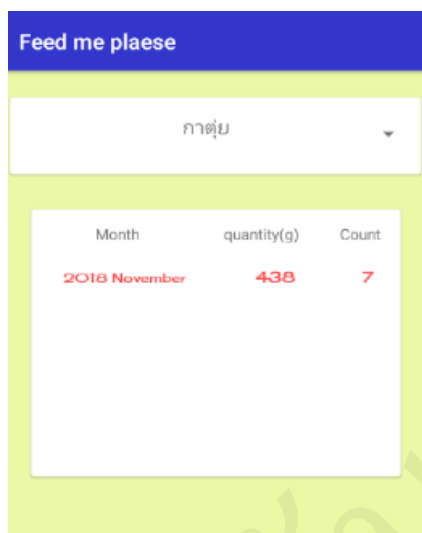
5 November 2018

กาดูบ

Date	quantity(g)
2018-11-05	70
2018-11-06	52
2018-11-07	28
2018-11-08	122
2018-11-09	166

รูปที่ 4.14 หน้าแสดงข้อมูลรายอาทิตย์

4) แสดงข้อมูลสถิติการกินอาหารของสัตว์เลี้ยงรายเดือน



Month	quantity(g)	Count
2018 November	438	7

รูปที่ 4.15 หน้าแสดงข้อมูลสถิติรายเดือน

4.2.7 แสดงข้อมูลและจัดการข้อมูลของรูปแบบการปล่อยอาหาร

โดยมีการทำงานทั้งหมด 4 ขั้นตอน ดังนี้

- 1) แสดงข้อมูลการเลือกรูปแบบการปล่อยอาหาร และสามารถเลือกรูปแบบการปล่อยอาหารได้



รูปที่ 4.16 หน้าแสดงข้อมูลรูปแบบการปล่อยอาหาร

2) แสดงและแก้ไขการปล่อยอาหารรูปแบบที่ 1

Feed me please

TYPE 1

ปริมาณอาหารที่ต้องการปล่อย 30

ปริมาณที่คงเหลือในถาดอาหาร 50

OK

รูปที่ 4.17 หน้าการปล่อยอาหารรูปแบบที่ 1

3) แสดงข้อมูลชุดคำสั่งการปล่อยอาหารทั้งหมดของรูปแบบที่ 2

Feed me please

TYPE 2

15:58:00	20
15:59:10	50
15:49:47	20
15:51:20	30
15:50:58	20

EXIT

รูปที่ 4.18 หน้าเมนูการปล่อยอาหารรูปแบบที่ 2

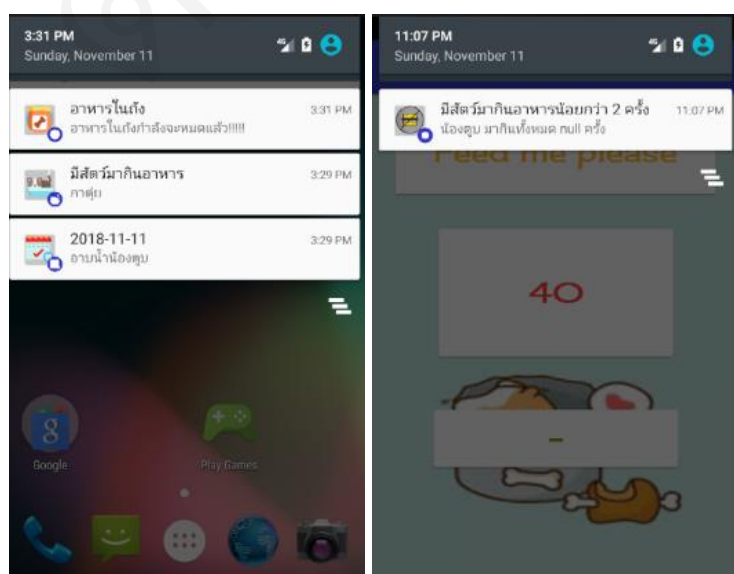
4) แสดงและแก้ไขข้อมูลการตั้งค่าการปล่อยอาหารรูปแบบที่ 2



รูปที่ 4.19 หน้าการปล่อยอาหารรูปแบบที่ 2

4.2.8 แสดงข้อมูลแจ้งเตือน

ในรายการนี้จะมีการแสดงการแจ้งเตือนวันนัดหมาย ปริมาณอาหารใกล้เคียง และ สัตว์เลี้ยงที่มากินอาหารน้อยกว่า 2 ครั้ง



รูปที่ 4.20 หน้าแจ้งเตือน

4.3 ผลการทดสอบเครื่องให้อาหาร

4.3.1 ปล่อยอาหารตามปริมาณที่กำหนด สามารถสรุปได้ดังนี้

ตารางที่ 4.1 ผลการบันทึกข้อมูลปริมาณอาหารที่ปล่อย

ครั้งที่	ปริมาณที่กำหนด (กรัม)	ปริมาณจริง (กรัม)
1	20	17
2	20	19
3	20	15
4	20	20
5	20	16
6	20	15
7	20	18
8	20	19
9	20	17
10	20	20
ปริมาณอาหารโดยเฉลี่ย		17.6

จากตารางที่ 4.1 สามารถสรุปผลการทดสอบการปล่อยอาหาร โดยทำการทดสอบการปล่อยจำนวน 10 ครั้ง ปัจจัยในการทดสอบอาหารเม็ดต้องมีขนาดที่เหมาะสม และความหนาแน่นในถังที่ไม่มากเกินไป ซึ่งเฉลี่ยแล้วในการปล่อยอาหารแต่ละครั้งจะมีปริมาณ 17.6 กรัม

4.3.2 ปล่อยอาหารตามเวลาที่กำหนด สามารถสรุปได้ดังนี้

ตารางที่ 4.2 ผลการบันทึกข้อมูลปล่อยอาหารตามเวลาที่กำหนด

ครั้งที่	ปล่อยอาหารตามเวลาที่กำหนดอย่างถูกต้อง	เกิดข้อผิดพลาดในการปล่อยอาหารตรงตามเวลาที่กำหนด
1	✓	✗
2	✓	✗

ตารางที่ 4.2 ผลการบันทึกข้อมูลปล่อยอาหารตามเวลาที่กำหนด (ต่อ)

ครั้งที่	ปล่อยอาหารตามเวลาที่กำหนดอย่างถูกต้อง	เกิดข้อผิดพลาดในการปล่อยอาหารตรงตามเวลาที่กำหนด
3	✓	✗
4	✓	✗
5	✓	✗
6	✓	✗
7	✓	✗
8	✓	✗
9	✓	✗
10	✓	✗

จากตารางที่ 4.2 สามารถสรุปผลการปล่อยอาหารตามเวลาที่กำหนด จำนวน 10 ครั้งได้ว่าเครื่องให้อาหารสามารถปล่อยอาหารตามเวลาที่ระบบกำหนดได้ได้อย่างถูกต้อง โดยไม่มีข้อผิดพลาดแม้แต่ครั้งเดียว

4.3.3 ปล่อยอาหารตามปริมาณอาหารในถาดที่กำหนด สามารถสรุปได้ดังนี้

ตารางที่ 4.3 ผลการบันทึกข้อมูลปล่อยอาหารตามปริมาณอาหารในถาดที่กำหนด

ครั้งที่	ปล่อยอาหารตามตามปริมาณอาหารในถาดที่กำหนดอย่างถูกต้อง	เกิดข้อผิดพลาดในการปล่อยอาหารตามปริมาณอาหารในถาดที่กำหนด
1	✓	✗
2	✓	✗
3	✓	✗
4	✗	✓
5	✗	✓
6	✓	✗
7	✓	✗

ตารางที่ 4.3 ผลการบันทึกข้อมูลปล่อยอาหารตามปริมาณอาหารในภาตที่กำหนด (ต่อ)

ครั้งที่	ปล่อยอาหารตามตามปริมาณอาหาร ในภาตที่กำหนดอย่างถูกต้อง	เกิดข้อผิดพลาดในการปล่อยอาหารตาม ปริมาณอาหารในภาตที่กำหนด
8	✓	✗
9	✓	✗
10	✓	✗

จากตารางที่ 4.3 สามารถสรุปผลการปล่อยอาหารตามปริมาณอาหารในภาตที่กำหนดจำนวน 10 ครั้ง ได้ว่าเครื่องให้อาหารสามารถปล่อยอาหารตามปริมาณอาหารในภาตที่กำหนดได้สำเร็จ 8 ครั้ง และผิดพลาด 2 ครั้ง ทั้งนี้ขึ้นอยู่กับความเร็วในการตรวจสอบปริมาณอาหารในภาต

4.3.4 บันทึกข้อมูลปริมาณอาหารในภาตเข้าสู่ฐานข้อมูล สามารถสรุปได้ดังนี้

ตารางที่ 4.4 ผลการบันทึกข้อมูลปริมาณอาหารในภาตเข้าสู่ฐานข้อมูล

ครั้งที่	บันทึกข้อมูลปริมาณอาหารในภาตเข้าสู่ ฐานข้อมูลได้ถูกต้อง	เกิดข้อผิดพลาดในการบันทึกข้อมูลปริมาณ อาหารในภาตเข้าสู่ฐานข้อมูล
1	✓	✗
2	✓	✗
3	✓	✗
4	✓	✗
5	✓	✗
6	✓	✗
7	✓	✗
8	✓	✗
9	✓	✗
10	✓	✗

จากตารางที่ 4.4 สามารถสรุปผลการบันทึกข้อมูลปริมาณอาหารในภาตเข้าสู่ฐานข้อมูลจำนวน 10 ครั้ง ได้ว่าเครื่องให้อาหารสามารถบันทึกข้อมูลปริมาณอาหารในภาตไปยังฐานข้อมูลได้อย่างถูกต้อง โดยไม่มีข้อผิดพลาดแม้แต่ครั้งเดียว

4.3.5 บันทึกข้อมูลรหัสประจำตัวสัตว์เลี้ยงจากแทคประจำตัวสัตว์เลี้ยงเข้าสู่ฐานข้อมูล
สามารถสรุปได้ดังนี้

ตารางที่ 4.5 ผลการบันทึกรหัสประจำตัวสัตว์เลี้ยงจากแทคประจำตัวสัตว์เลี้ยงเข้าสู่ฐานข้อมูล

ครั้งที่	บันทึกข้อมูลรหัสประจำตัวสัตว์เลี้ยง เข้าสู่ฐานข้อมูลได้ถูกต้อง	เกิดข้อผิดพลาดในการบันทึกข้อมูลรหัส ประจำตัวสัตว์เลี้ยงเข้าสู่ฐานข้อมูล
1	✓	✗
2	✓	✗
3	✓	✗
4	✓	✗
5	✓	✗
6	✓	✗
7	✓	✗
8	✓	✗
9	✓	✗
10	✓	✗

จากตารางที่ 4.5 สามารถสรุปผลการบันทึกบันทึกข้อมูลรหัสประจำตัวสัตว์เลี้ยงเข้าสู่
ฐานข้อมูล จำนวน 10 ครั้ง ได้ว่าเครื่องให้อาหารสามารถบันทึกข้อมูลปริมาณอาหารในถาดไปยัง
ฐานข้อมูลได้อย่างถูกต้อง โดยไม่มีข้อผิดพลาดแม้แต่ครั้งเดียว

4.3.6 บันทึกข้อมูลการแจ้งเตือนอาหารในถังใกล้หมดเข้าสู่ระบบ สามารถสรุปได้ดังนี้

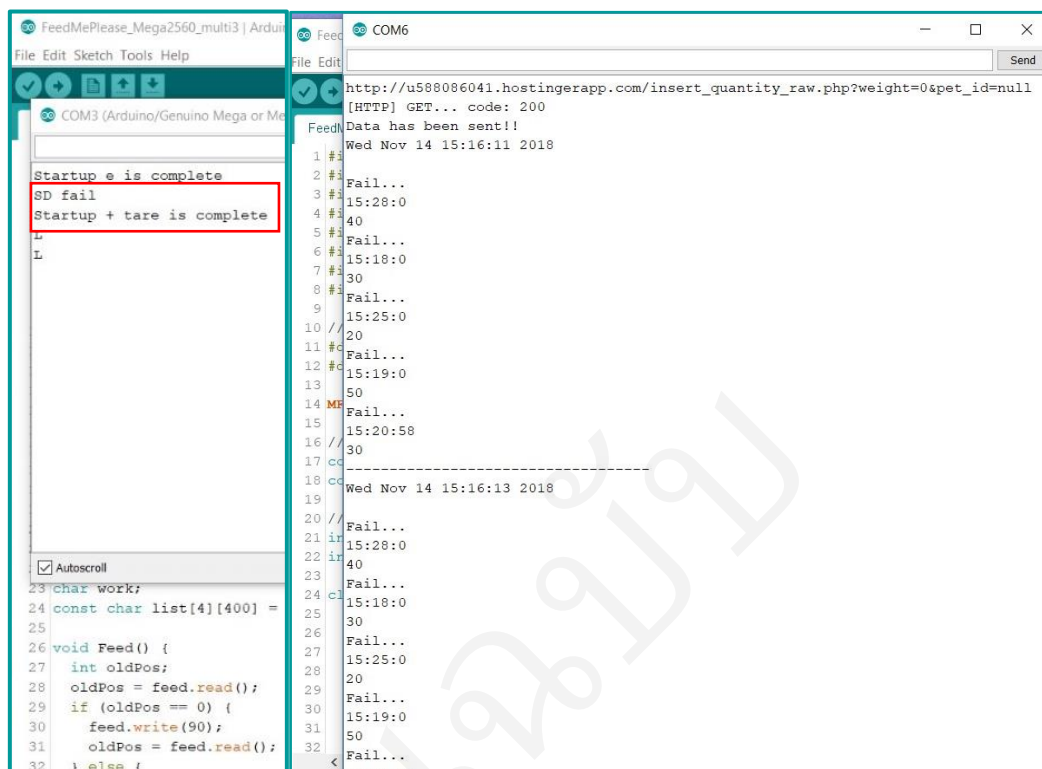
ตารางที่ 4.6 ผลการบันทึกข้อมูลการแจ้งเตือนอาหารในถังใกล้หมดเข้าสู่ระบบ

ครั้งที่	บันทึกข้อมูลการแจ้งเตือนอาหารในถังใกล้หมดเข้าสู่ระบบได้ถูกต้อง	เกิดข้อผิดพลาดในการบันทึกข้อมูลการแจ้งเตือนอาหารในถังใกล้หมดเข้าสู่ระบบ
1	✓	✗
2	✓	✗
3	✓	✗
4	✓	✗
5	✓	✗
6	✓	✗
7	✓	✗
8	✓	✗
9	✓	✗
10	✓	✗

จากตารางที่ 4.6 สามารถสรุปผลการบันทึกบันทึกข้อมูลการแจ้งเตือนอาหารในถังใกล้หมดเข้าสู่ระบบ จำนวน 10 ครั้ง ได้ว่าเครื่องให้อาหารสามารถบันทึกข้อมูลการแจ้งเตือนอาหารในถังใกล้หมดไปยังระบบได้อย่างถูกต้อง โดยไม่มีข้อผิดพลาดแม้แต่ครั้งเดียว

4.4 ผลการทดสอบรันโปรแกรมบนไมโครคอนโทรลเลอร์

4.4.1 ไมโครคอนโทรลเลอร์เริ่มทำงาน



รูปที่ 4.21 มอนิเตอร์เมื่อไมโครคอนโทรลเลอร์เริ่มทำงาน

ฝั่ง Arduino Mega ทำการเซตโหนดเซลล์ให้พร้อมใช้งานเมื่อเสร็จแล้วจะปริ้นท์ “Startup + tare is complete” และเซต SD card ถ้าหาไม่พบจะปริ้นท์ “SD fail” แต่ถ้าหาพบจะไม่ขึ้นข้อความใด ๆ ส่วนฝั่ง NodeMCU จะไม่ขึ้นข้อความใด นอกเสียจากเข้าสู่การให้อาหารจึงจะปริ้นท์ข้อความแสดงการทำงาน จากรูปที่ 4.21 เป็นการให้อาหารรูปแบบที่ 2

4.4.2 การส่งค่าน้ำหนักและรหัสประจำตัวสัตว์ไปยังฐานข้อมูล

```

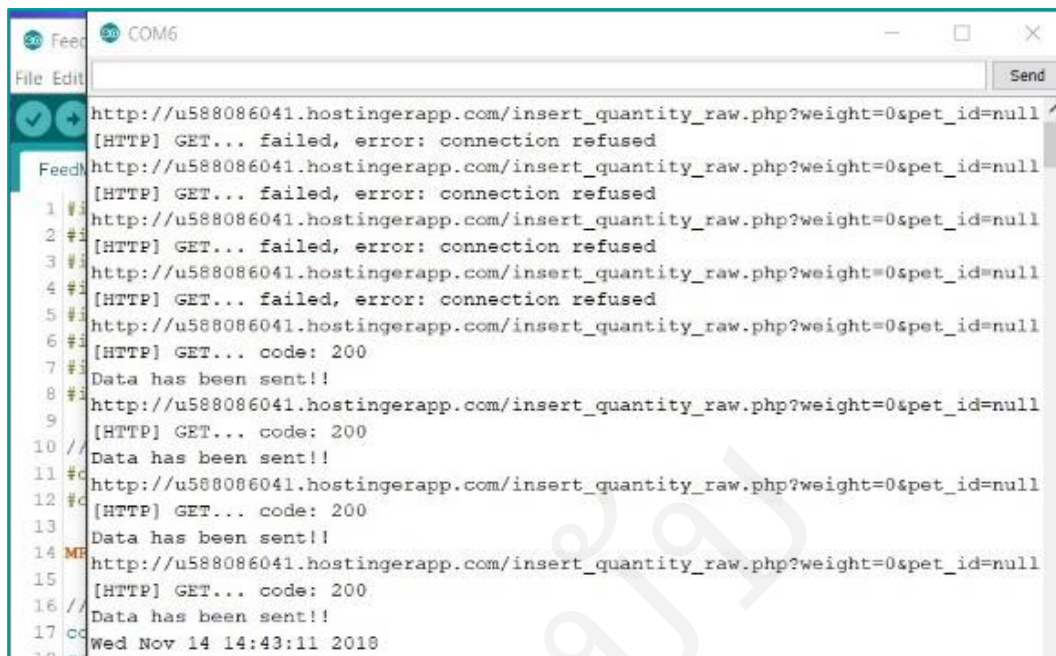
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=null
[HTTP] GET... code: 200
Data has been sent!!
RFID TAG ID:1961CBE
1 5511cc65 2 1961cbe 3 b65951d3
2: RFID = 1961cbe
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=2
[HTTP] GET... code: 200
Data has been sent!!
http://u588086041.hostingerapp.com/insert_to_tray.php
[HTTP] GET... code: 200
null
RFID TAG ID:5511CC65
1 5511cc65 2 1961cbe 3 b65951d3
1: RFID = 5511cc65
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=1
[HTTP] GET... code: 200
Data has been sent!!
http://u588086041.hostingerapp.com/insert_to_tray.php
[HTTP] GET... code: 200
null
RFID TAG ID:5511CC65
1 5511cc65 2 1961cbe 3 b65951d3
1: RFID = 5511cc65
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=1
[HTTP] GET... code: 200
Data has been sent!!
Wed Nov 14 15:58:12 2018
  
```

รูปที่ 4.22 มอนิเตอร์เมื่อมีการส่งค่าน้ำหนักในถาดและรหัสประจำตัวสัตว์เลี้ยง

การส่งค่าน้ำหนักในถาด และรหัสประจำตัวสัตว์เลี้ยงจะทำทุก ๆ 10 วินาที โดย NodeMCU จะส่งคำสั่งไปยัง Arduino Mega ให้ส่งข้อมูลน้ำหนักจากโหลดเซลล์ ณ ขณะนั้นมารวมกับรหัสประจำตัวสัตว์เลี้ยง หาก ณ ขณะนั้นไม่ได้รับรหัสประจำตัวสัตว์เลี้ยง จะส่งเป็นค่า Null ไป (1)

แต่หากได้รับรหัสประจำตัวสัตว์เลี้ยงจากเครื่องอ่านอาร์เอฟไอดี จะทำการตรวจสอบรหัสอาร์เอฟไอดีของสัตว์เลี้ยงตัวนั้นกับรหัสประจำตัว ว่าตรงกับรหัสประจำตัวสัตว์เลี้ยงตัวใด แล้วส่งน้ำหนักในถาดและรหัสประจำตัวสัตว์เลี้ยงตัวนั้นไปยังฐานข้อมูล (2)

4.4.3 การส่งข้อมูลไปบันทึกยังฐานข้อมูล



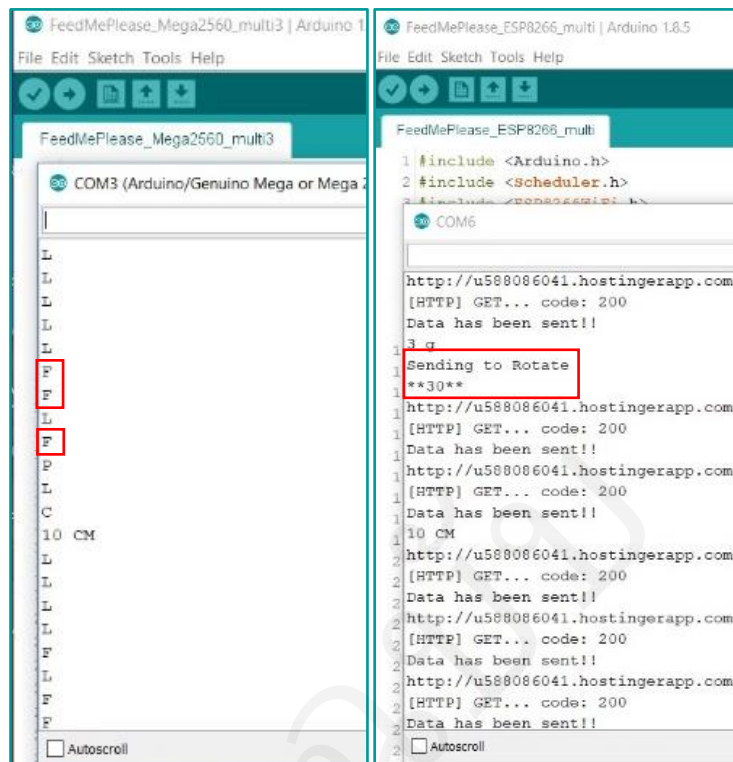
```

COM6
File Edit Send
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=null
[HTTP] GET... failed, error: connection refused
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=null
[HTTP] GET... failed, error: connection refused
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=null
[HTTP] GET... failed, error: connection refused
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=null
[HTTP] GET... failed, error: connection refused
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=null
[HTTP] GET... code: 200
Data has been sent!!
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=null
[HTTP] GET... code: 200
Data has been sent!!
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=null
[HTTP] GET... code: 200
Data has been sent!!
http://u588086041.hostingerapp.com/insert_quantity_raw.php?weight=0&pet_id=null
[HTTP] GET... code: 200
Data has been sent!!
Wed Nov 14 14:43:11 2018
  
```

รูปที่ 4.23 มอนิเตอร์เมื่อมีการส่งข้อมูลไปบันทึกยังฐานข้อมูล

หากการส่งข้อมูลไม่สามารถทำได้ หรือเกิดข้อผิดพลาดขึ้นจะปริ้นท์ “[HTTP] GET... failed, error: connection refused” ต้องทำการเชื่อมต่ออินเทอร์เน็ตอีกครั้ง เมื่อสามารถเชื่อมต่อฐานข้อมูลได้สำเร็จจะปริ้นท์ “[HTTP] GET... code: 200” และสามารถส่งข้อมูลไปบันทึกได้สำเร็จจะปริ้นท์ “Data has been sent!!” แต่ถ้าบันทึกข้อมูลไม่สำเร็จจะขึ้น “Invalid query” ซึ่งอาจจะขึ้นอยู่กับ Query ที่เขียนในไฟล์ PHP หรือรูปแบบข้อมูลที่ส่งไปไม่ถูกต้อง

4.4.4 การให้อาหารรูปแบบที่ 1



รูปที่ 4.24 มอนิเตอร์เมื่อมีการให้อาหารรูปแบบที่ 1

การให้อาหารรูปแบบที่ 1 NodeMCU จะทำการดึงข้อมูลจากฐานข้อมูลมาเปรียบเทียบกับปริมาณอาหารในถาด เมื่ออาหารในถาดเหลือน้อยกว่าค่าที่กำหนดไว้ก็จะปริ้นท์ปริมาณอาหารที่ให้ และส่งคำสั่ง “F” (Feed) ให้เซอร์โวหมุนผ่านทาง Arduino Mega เพื่อให้อาหารในปริมาณที่กำหนด ซึ่งปริมาณที่ให้อันได้แก่ 20, 30, 40 และ 50 กรัม จะมีการสั่งให้เซอร์โวหมุน 2, 3, 4 และ 5 ครั้ง ตามลำดับ

4.4.5 การให้อาหารรูปแบบที่ 2

```

FeedMePlease_Mega2560_multi3 | Arduino 1.8.5
File Edit Sketch Tools Help

COM3 (Arduino/Genuino Mega or Mega 2560)

Startup + tare is complete
L
F
F
L
F
P
L
C
11 CM
L
L
F
L
F
F
L
F
F
L
F
T

COM6

http://u588086041.hostingerapp.co
[HTTP] GET... code: 200
Data has been sent!!
Wed Nov 14 15:18:11 2018

Fail...
15:28:0
40
It's Work!!
15:18:0
30
Sending to Rotate
**30**
http://u588086041.hostingerapp.co
[HTTP] GET... code: 200
Data has been sent!!
http://u588086041.hostingerapp.co
[HTTP] GET... code: 200
Data has been sent!!
11 CM
http://u588086041.hostingerapp.co
[HTTP] GET... code: 200
Data has been sent!!
Fail...
15:25:0
20
Fail...
15:19:0
50
Fail...
15:20:58
30

```

รูปที่ 4.25 มอนิเตอร์เมื่อมีการให้อาหารรูปแบบที่ 2

การให้อาหารรูปแบบที่ 2 NodeMCU จะทำการดึงข้อมูลจากฐานข้อมูลเพื่อเปรียบเทียบเวลาที่ตั้งค่าไว้ กับเวลาจริง โดยโปรแกรมจะทำการตรวจสอบเวลาตรงกันเพียงนาที่ละครั้งเท่านั้น หากเวลาตรงกันจะปริ้นท์ “It’s Work!!” ปริมาณอาหารที่ให้ และส่งคำสั่ง “F” (Feed) ให้เซอร์โวหมุนผ่านทาง Arduino Mega เพื่อให้อาหารในปริมาณที่กำหนด ซึ่งปริมาณที่ให้อันได้แก่ 20, 30, 40 และ 50 กรัม จะมีการสั่งให้เซอร์โวหมุน 2, 3, 4 และ 5 ครั้ง ตามลำดับ

4.4.6 การเล่นเสียง และตรวจสอบอาหารใกล้หมดถึง

```

COM3 (Arduino/Genuino Mega or Mega 2560)
Startup + tare is complete
L
F
F
L
F
P
L
C
11 CM
L
F
L
F
L
F
L
F
L
P
L
C
14 CM

COM6
[HTTP] GET... code: 200
Data has been sent!!
http://u588086041.hostingerapp.com/insert_quantity_raw.php
[HTTP] GET... code: 200
Data has been sent!!
http://u588086041.hostingerapp.com/insert_quantity_raw.php
[HTTP] GET... code: 200
Data has been sent!!
14 CM
http://u588086041.hostingerapp.com/insert_food_alert.php
[HTTP] GET... code: 200
Data has been sent!!
-----
http://u588086041.hostingerapp.com/insert_quantity_raw.php
[HTTP] GET... code: 200
Data has been sent!!
http://u588086041.hostingerapp.com/insert_quantity_raw.php
[HTTP] GET... code: 200
Data has been sent!!
Wed Nov 14 15:21:12 2018
Fail...
15:28:0
40
Fail...
15:18:0
30
Fail...
15:25:0
20
Fail...
15:19:0
50
Fail...
15:20:58
30

```

รูปที่ 4.26 มอนิเตอร์เมื่อมีการส่งคำสั่งเล่นเสียงและข้อมูลระยะห่างในถังบรรจุอาหาร

เมื่อมีคำสั่งการให้อาหาร หลังจากนั้น NodeMCU จะส่งคำสั่ง “P” (Play Sound) เล่นเสียงไปยัง Arduino Mega โดยจะแรนดอมไฟล์เสียงใน Micro SD Card แล้วเล่นเสียงผ่านทางลำโพง เมื่อเล่นเสียงแล้ว NodeMCU ก็ส่งคำสั่ง “C” เพื่อตรวจสอบระยะห่างในถังบรรจุอาหารระหว่างฝาถัง ถึงอาหาร ว่ามีระยะห่างมากกว่าระยะที่กำหนดหรือไม่ หากมากกว่าจะส่งข้อมูลแจ้งเตือนไปยังฐานข้อมูล

บทที่ 5

สรุปผลและข้อเสนอแนะ

ผลสรุปจากการทำโครงงานเครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลื้อยขนาดเล็ก จึงได้ทราบถึงปัญหาที่เกิดขึ้นระหว่างการจัดทำ ติดตั้ง และทดสอบโครงงาน โดยมีผลสรุป และข้อเสนอแนะ เพื่อการพัฒนาแก้ไข ปรับปรุงระบบให้โครงงานมีความสามารถมากขึ้น

5.1 สรุปผลการดำเนินงาน

ผลที่ได้จากการทำโครงงาน จากการทดสอบการทำงานของเครื่องให้อาหาร และโมบายแอปพลิเคชัน ระบบสามารถทำการตั้งค่าการปล่อยอาหาร, ปล่อยอาหาร, ตรวจวัดน้ำหนักปริมาณอาหารในถาด, แจ้งเตือนปริมาณอาหารในถังใกล้หมด, ตรวจสอบสัตว์เลื้อยที่อยู่ใกล้บริเวณ, เพิ่ม แก้ไข และลบข้อมูลสัตว์เลื้อย, เพิ่ม แก้ไข และลบตารางนัดหมาย และตรวจสอบสถิติการกินของสัตว์เลื้อย ทำให้ผู้เลี้ยงสัตว์สามารถให้อาหารสัตว์เลื้อยได้แม้อยู่ในที่อาศัย เกิดความสะดวกสบายมากยิ่งขึ้น สามารถตรวจสอบสถิติการกินอาหารของสัตว์เลื้อยเพื่อควบคุมปริมาณการให้อาหารของสัตว์เลื้อยในแต่ละครั้งได้ แม้ระบบอาจยังไม่สมบูรณ์มากนัก แต่จากผลการทดสอบการใช้งานจริงก็เป็นที่น่าพึงพอใจ

จากการออกแบบ และพัฒนาเครื่องให้อาหารเม็ดอัตโนมัติสำหรับสัตว์เลื้อยขนาดเล็ก ที่ได้กล่าวมาข้างต้น ทำให้คณะผู้จัดทำได้รับความรู้ ความเข้าใจในการวิเคราะห์ และออกแบบระบบโดยการใช้โปรแกรม Android Studio ซึ่งพัฒนาโปรแกรมด้วยภาษา Java ในส่วนของการควบคุมบอร์ดไมโครคอนโทรลเลอร์ใช้โปรแกรม Arduino IDE ซึ่งพัฒนาโปรแกรมด้วยภาษา C และใช้ภาษา PHP ในการเชื่อมต่อกับฐานข้อมูล การจัดทำโครงงานนี้ คณะผู้จัดทำสามารถที่จะนำประสบการณ์ในครั้งนี้ไปใช้ในการทำงานจริงในอนาคต และประยุกต์ใช้กับงานอื่น ๆ ต่อไป

5.2 ปัญหาและอุปสรรค

5.2.1 แรงดันไฟจากบอร์ด NodeMCU ESP8266 ไม่เพียงพอสำหรับอุปกรณ์ตัวอื่น และพอร์ตไม่พอสำหรับต่ออุปกรณ์ทั้งหมด

5.2.2 การระบุตัวสัตว์เลื้อยยังไม่แม่นยำนักหาก RFID tag และไมโดนระยะการอ่านของ RFID Card Reader

5.2.3 ไม่สามารถบันทึกไฟล์เสียงจากโทรศัพท์ไปยังฐานข้อมูลได้

5.3.4 การแสดงข้อมูลสถิติปริมาณการกินในรูปแบบกราฟแท่งไม่สามารถทำได้

5.3 ข้อเสนอแนะ

5.3.1 ศึกษาการขยายพอร์ตของ NodeMCU เพิ่ม หรือเลือกซื้อบอร์ดไมโครคอนโทรลเลอร์ที่มีจำนวนพอร์ตเพียงพอสำหรับการต่ออุปกรณ์อื่นร่วมด้วย

5.3.2 ศึกษาวิธีการระบุตัวต้นสัต์เลี้ยงในรูปแบบอื่น ๆ เพิ่มเติม เช่น การฝังชิพได้ผิวหน้าสัต์เลี้ยง เป็นต้น

5.3.3 การแสดงข้อมูลสถิติปริมาณการกินในรูปแบบกราฟแท่ง อาจขึ้นอยู่กับความเข้ากันได้ของระบบ หรือรุ่นของโปรแกรม Android Studio และไลบรารีของกราฟแท่ง

5.3.4 ควรเพิ่มการตรวจสอบปริมาณอาหารล้นถาด

บรรณานุกรม

- [1] เกียรติศักดิ์ อยู่ดี, **เครื่องให้อาหารเม็ดอัตโนมัติ**, [ออนไลน์] เข้าถึงได้จาก : <http://payaptechno.ac.th> (15 กุมภาพันธ์ 2560).
- [2] Dolporn Prasartkijt, **ความรู้พื้นฐานของ RFID**, [ออนไลน์] เข้าถึงได้จาก : <http://carparkcenter.com> (24 กุมภาพันธ์ 2560).
- [3] Identify Limited, **โครงสร้างของระบบอาร์เอฟไอดี (Tag)**, [ออนไลน์] เข้าถึงได้จาก : <http://rfid.co.th/knowledge/74-rfid-tag?lang=en> (23 กุมภาพันธ์ 2560).
- [4] ปุณณิศา จิตรเจริญ, ภูริทัต ชามาตย์ และ อัษฎาภรณ์ ชูชั้นทอง, **เบา เบา, สารนิพนธ์ วิศวกรรมอัตโนมัติ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**.
- [5] Load Cell Guru, **Load cell ไม่ได้ถูกออกแบบจาก Strain Gauge อย่างเดียว**, [ออนไลน์] เข้าถึงได้จาก : <https://mall.factomart.com/load-cell-design-not-only-from-strain-gauge> (24 กุมภาพันธ์ 2560).
- [6] ThaiEasyElec, **บทความ Arduino คืออะไร? ตอนที่1 แนะนำเพื่อนใหม่ที่ชื่อ Arduino**, [ออนไลน์] เข้าถึงได้จาก : <http://www.thaieasyelec.com/article-wiki/basic-electronics/บทความ-arduino-คืออะไร-เริ่มต้นใช้งาน-arduino.html> (17 ธันวาคม 2560).
- [7] ArduinoAll, **บอร์ด Arduino รุ่น MEGA 2560 R3 พร้อมสาย USB**, [ออนไลน์] เข้าถึงได้จาก : <https://www.arduinoall.com/product/17/บอร์ด-arduino-รุ่น-mega-2560-r3-พร้อมสาย-usb> (16 ตุลาคม 2561).
- [8] AWOL, Aom Noicheen and Siriwimon Sunthon, **Embeded System Design**, [ออนไลน์] เข้าถึงได้จาก : <http://mbeddedweekly.blogspot.com/2014/08/arduino-mega2560.html> (16 ตุลาคม 2561).
- [9] Arduino, **Arduino Mega**, [ออนไลน์] เข้าถึงได้จาก : <https://www.arduino.cc/en/Main/arduinoBoardMega> (16 ตุลาคม 2561).
- [10] Wisdomgoody, **รู้จักกับ Arduino ESP8266 (NodeMCU)**, [ออนไลน์] เข้าถึงได้จาก : <http://www.ayarafun.com/2015/08/introduction-arduino-esp8266-nodemcu> (13 ธันวาคม 2560).
- [11] AUSCOM, **NODEMCU COMPATIBLE ESP8266 CH340 BOARD ESP 8266 WIRELESS IOT LUA WIFI WIRELESS**, [ออนไลน์] เข้าถึงได้จาก : <https://www.auscomtech.com.au/product/nodemcu-compatible-esp8266-ch340-board-esp-8266-wireless-iot-lua-wifi-wireless> (16 ตุลาคม 2561).

บรรณานุกรม (ต่อ)

- [12] ThaiEasyElec, บทความ "ตัวอย่างการควบคุม RC Servo Motor ด้วย Arduino", [ออนไลน์] เข้าถึงได้จาก : <http://www.thaieasyelec.com/article-wiki/review-product-article/บทความตัวอย่างการควบคุม-rc-servo-motor-ด้วย-arduino.html> (17 ธันวาคม 2560).
- [13] สยามโฟน ดอท คอม, รู้หรือไม่!! โมบายแอปพลิเคชันคืออะไร? และวิธีสังเกตแอปฯ ปลอม ทำได้อย่างไร?, [ออนไลน์] เข้าถึงได้จาก : <http://news.siamphone.com/news-17863.html> (1 ตุลาคม 2561).

ภาคผนวก ก
ขั้นตอนการติดตั้งโมบายแอปพลิเคชัน

(อยู่ในแผ่น CD)

ภาคผนวก ข
คู่มือการใช้งานนโยบายแอปพลิเคชัน

(อยู่ในแผ่น CD)

ภาคผนวก ค

Source code

(อยู่ในแผ่น CD)

ประวัติผู้จัดทำปริญญานิพนธ์

ประวัติผู้จัดทำปริญญานิพนธ์



ชื่อ	นางสาวไอรดา พฤษภา รหัส 115710400160-4
สาขาวิชา/ภาควิชา	ภาควิชาวิศวกรรมคอมพิวเตอร์
วัน-เดือน-ปี เกิด	วันที่ 16 กรกฎาคม 2538
สถานที่เกิด	จังหวัดลพบุรี
ที่อยู่	46/268 ซอยนิมิตใหม่ 40 แขวงสามวาตะวันออก เขตคลองสามวา กรุงเทพมหานคร 10210
ประวัติการศึกษา	ประกาศนียบัตรวิชาชีพ (ปวช.) วิทยาลัยเทคนิคัญบุรี ปีที่สำเร็จ 2556

ประวัติผู้จัดทำปริญญานิพนธ์



ชื่อ	นางสาวสายธาร งามขำ รหัส 115710400943-3
สาขาวิชา/ภาควิชา	ภาควิชาวิศวกรรมคอมพิวเตอร์
วัน-เดือน-ปี เกิด	วันที่ 3 ตุลาคม 2538
สถานที่เกิด	จังหวัดนนทบุรี
ที่อยู่	38 หมู่ 4 ตำบลหนองขาว อำเภот่าม่วง จังหวัดกาญจนบุรี 71110
ประวัติการศึกษา	มัธยมศึกษาตอนปลาย (ม.6) โรงเรียนกาญจนาอนุเคราะห์ ปีที่สำเร็จ 2556