

แอปพลิเคชันสแกนอาหาร

Beingwellness

นายพัทธดนย์ จัยสิน

โครงการนี้เป็นส่วนหนึ่งของการศึกษา

หลักสูตรวิทยาศาสตร์บัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

ภาควิชาคณิตศาสตร์ สติติ และคอมพิวเตอร์ คณะวิทยาศาสตร์

มหาวิทยาลัยอุบลราชธานี

ปีการศึกษา 2561

ลิขสิทธิ์ของมหาวิทยาลัยอุบลราชธานี

โครงงาน : แอปพลิเคชันสแกนอาหาร  
Beingwellness

โดย : นายพัทธดนย์ จัยสิน

อาจารย์ที่ปรึกษา : ดร. ไพบูลย์ คงไชย

ระดับการศึกษา : วิทยาศาสตร์บัณฑิต สาขาวิชาการคอมพิวเตอร์

ปีการศึกษา : 2561

---

ได้รับการพิจารณาให้เป็นส่วนหนึ่งของการศึกษา  
ตามหลักสูตรวิทยาศาสตร์บัณฑิต สาขาวิชาการคอมพิวเตอร์

คณะกรรมการสอบประเมินความรู้โครงงานคอมพิวเตอร์

..... อาจารย์ที่ปรึกษา  
(ดร. ไพบูลย์ คงไชย)

..... กรรมการ  
(อาจารย์ วาสนา เหง้าเกษ )

..... กรรมการ  
(ผศ. เกรียงศักดิ์ ตรีประพิน)

..... หัวหน้าภาควิชา  
(ดร. ชัชวิน นามมั่น)

วันที่ ...../...../.....

## กิตติกรรมประกาศ

การพัฒนาโครงงานแอปพลิเคชันบีอิงเวลเนส สำเร็จลุล่วงได้ด้วยความกรุณาและความช่วยเหลือจากหลายๆ ท่าน ข้าพเจ้าขอขอบพระคุณทุกท่าน ที่มีส่วนร่วมในการพัฒนาโครงงานนี้

ขอขอบพระคุณอาจารย์ที่ปรึกษา ดร. ไชย อาจารย์ที่ปรึกษาโครงงานที่ได้แนะนำทฤษฎีและแนวทางในแก้ปัญหาต่าง ๆ ที่เกิดขึ้นระหว่างการพัฒนาแอปพลิเคชัน อีกรังสิยังคงอยู่ตรวจสอบความก้าวหน้าของการทำงานเป็นระยะ ๆ รวมทั้งสร้างกำลังใจให้ผู้พัฒนาอยู่เสมอ

ขอขอบพระคุณเจ้าหน้าที่ภาควิชาคณิตศาสตร์ สถิติและคอมพิวเตอร์ ที่คอยเอื้ออำนวย ความสะดวกทั้งเรื่องอุปกรณ์และสถานที่ต่อการปฏิบัติงานของผู้พัฒนา

ขอกราบขอบพระคุณบิดา มารดา ที่คอยให้กำลังใจ ค่อยให้ความรักและความห่วงใยเสมอมา ตลอดจนคอยช่วยเหลือทุนทรัพย์ทางด้านการศึกษาและอุปกรณ์ในการพัฒนาโครงงาน

นายพัทธดนย์ จัยสิน วันที่ 19 เมษายน 62

โครงการ : แอปพลิเคชันสแกนอาหาร  
 โดย : นายพัทธดนย์ จัยสิน  
 อาจารย์ที่ปรึกษา : ดร. ไพบูลย์ คงไชย  
 ระดับการศึกษา : วิทยาศาสตร์บัณฑิต สาขาวิชาภysics-comพิวเตอร์  
 ปีการศึกษา : 2561

---

## บทคัดย่อ

Being-wellness เป็นแอปพลิเคชันที่วัดแคลลอรี่ ไขมัน น้ำตาล และโซเดียม จากอาหาร ที่ รับประทานเพื่อคำนวณความเสี่ยงของโรคที่เกิดจากการรับประทานอาหาร ผู้ใช้สามารถถ่ายรูปอาหารเพื่อเก็บ ข้อมูลด้วยการใช้เทคนิคการตรวจจับวัตถุแล้วนำผลลัพธ์ที่ได้ไปเทียบกับฐานข้อมูลจากนั้นนำข้อมูลอาหารที่ได้มาแสดง ในแอปพลิเคชันนี้ได้ใช้การเรียนรู้เชิงลึกและใช้ขั้นตอนวิธีMobilenet v1 1.0 เพื่อตรวจจับอาหาร จากรูปภาพ ผู้จัดทำได้เลือกรูปภาพอาหารที่แตกต่างกัน 50 รูปโดยเลือกใช้สินค้าที่มียอดขายดีสุดจากร้านค้า เชเว่น-อีเลฟเว่น เพื่อเป็นชุดข้อมูลฝึกฝน จากการทดลองพบว่ามีค่าความถูกต้องอยู่ที่ร้อยละ 78.8 จาก รูปภาพอาหาร 100 รูป คำสำคัญ: สมาร์ทโฟน, แอปพลิเคชัน, เทคนิคการตรวจจับวัตถุ, การเรียนรู้เชิงลึก, Mobilenet v1 1.0, ฐานข้อมูล

Topic : Beingwellness  
Author : PATTADON JAISIN  
Advisor : PHAICHAYON KONGCHAI, Ph.D.  
Degree : Bachelor of Science (Computer Science)  
Academic Year : 2018

---

## **Abstract**

Being-wellness is an smartphone application that determines calorie, fat, sugar and sodium from the food you are eating daily for predict risk diseases from your food consumption. Users can take a food photo to record data based on the object detection technique And bring the results compared to the database, then bring the food information that has been shown. In this application , we used deep learning algorithm named Mobilenet v1 1.0 for detect foods from photo, we collected 50 different food photos in best seller from seven-eleven supermarket for training model. The experimental results showed that 78.8percent of 100 of food photos were correctly detected.

Keywords: Smartphone Application, Object Detection Technique, Deep Learning Algorithm, Mobilenet v1 1.0, Database

# สารบัญ

	หน้า
กิตติกรรมประกาศ . . . . .	ค
บทคัดย่อภาษาไทย . . . . .	ง
บทคัดย่อภาษาอังกฤษ . . . . .	จ
สารบัญ . . . . .	ฉ
สารบัญตาราง . . . . .	ฉ
สารบัญภาพ . . . . .	ญ
บทที่	
1 บทนำ . . . . .	<b>1</b>
1.1 ที่มาและเหตุผล . . . . .	1
1.2 วัตถุประสงค์ . . . . .	1
1.3 ขอบเขตของโครงการ . . . . .	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ . . . . .	2
1.5 เครื่องมือที่ใช้ในการพัฒนา (Development tools) . . . . .	2
1.5.1 ฮาร์ดแวร์ . . . . .	2
1.5.2 ซอฟต์แวร์ (Software) . . . . .	3
1.5.3 แผนการดำเนินการ . . . . .	4
2 ทฤษฎีที่เกี่ยวข้อง . . . . .	<b>5</b>
2.1 ความรู้พื้นฐานระบบปฏิบัติการแอนดรอยด์ . . . . .	5
2.1.1 ประวัติความเป็นมาของระบบปฏิบัติการแอนดรอยด์ . . . . .	6
2.2 ความรู้พื้นฐาน Java . . . . .	7
2.2.1 ประวัติความเป็นมาของภาษา Java . . . . .	7
2.3 ความรู้พื้นฐาน Python . . . . .	7
2.3.1 ประวัติความเป็นมาของภาษา Python . . . . .	8
2.4 ความรู้พื้นฐาน Tensorflow Lite . . . . .	8
2.5 ความรู้พื้นฐาน Image Processing . . . . .	8
2.6 ความรู้พื้นฐาน ปัญญาประดิษฐ์ . . . . .	12
2.7 ความรู้พื้นฐานเกี่ยวกับการเรียนรู้เชิงลึก (Deep Learning) . . . . .	12
2.8 ความรู้พื้นฐานเกี่ยวกับการเรียนรู้ของเครื่อง (Machine Learning) . . . . .	14

2.9	ความรู้พื้นฐานเกี่ยวกับโครงข่ายประสาทเทียม (Neural Network) . . . . .	16
2.10	ความรู้พื้นฐาน Firebase . . . . .	17
3	การวิเคราะห์และออกแบบระบบ . . . . .	<b>19</b>
3.1	รายละเอียดการออกแบบระบบ . . . . .	20
3.2	System Requirements . . . . .	20
3.2.1	Functional Requirements . . . . .	20
3.2.2	Non-functional Requirements . . . . .	21
3.3	User Interface Design . . . . .	21
3.4	Use Case Diagram . . . . .	26
3.5	Class Diagram . . . . .	31
3.6	Sequence Diagram . . . . .	35
3.7	โครงสร้างฐานข้อมูลไฟร์เบส(Firebase Database Structure) . . . . .	44
3.8	ขั้นตอนการเตรียมข้อมูลและการสร้างโมเดล . . . . .	53
3.8.1	การเก็บข้อมูลรูปอาหาร . . . . .	53
3.8.2	การตั้งค่าขนาดภาพและการใช้รูปแบบการจำแนก . . . . .	54
3.8.3	การสร้างโมเดลด้วย Tensorflow Lite . . . . .	54
3.8.4	การแปลงไฟล์ PB ไปเป็น lite . . . . .	55
4	การพัฒนาระบบ . . . . .	<b>56</b>
4.1	โครงสร้างของการสร้างหน้า MainActivity . . . . .	56
4.2	โครงสร้างของการสร้างหน้า Main2Activity . . . . .	58
4.3	โครงสร้างของการสร้างหน้า DashboardFragment . . . . .	61
4.4	โครงสร้างของการสร้างหน้า ScanFragment . . . . .	66
4.5	โครงสร้างของการสร้างหน้า AddfoodFragment . . . . .	68
4.6	โครงสร้างของคลาส FoodList . . . . .	70
4.7	โครงสร้างของคลาส LoginActivity . . . . .	72
5	การทดสอบระบบ . . . . .	<b>74</b>
5.1	การทดสอบการใช้งานแอปพลิเคชัน . . . . .	74
5.2	การทดสอบความแม่นยำของ Tensorflow Lite . . . . .	80
6	สรุปและขอเสนอแนะ . . . . .	<b>86</b>
6.1	สรุปความสามารถของระบบ . . . . .	86

6.2	ปัญหาและอุปสรรคในการพัฒนา . . . . .	86
6.3	แนวทางการพัฒนาต่อ . . . . .	86
	บรรณานุกรม . . . . .	87
	ภาคผนวก . . . . .	90
ภาคผนวก ก	การติดตั้งเครื่องมือที่ใช้พัฒนาโปรแกรม . . . . .	<b>90</b>
ก.1	การติดตั้ง Android Studio . . . . .	90
ก.2	การติดตั้ง Python3.6 . . . . .	94
ก.3	การติดตั้ง Tensorflow . . . . .	95
ภาคผนวก ข	คู่มือการติดตั้งระบบ(ทำการแก้ไขภายหลังจากการสอบเสร็จ) . . . . .	<b>96</b>
ภาคผนวก ค	คู่มือการใช้งานระบบ . . . . .	<b>99</b>
	ประวัติผู้เขียน . . . . .	110

## สารบัญตาราง

ตารางที่	หน้า
1.1 ขั้นตอนการดำเนินงาน . . . . .	4
3.1 สัญลักษณ์ของ Use case Diagram . . . . .	26
3.2 อธิบาย Use Case หน้าที่ของระบบโดยทุกการทำงานจะต้องอาศัยการเข้าสู่ระบบ ทุกรังสี ในภาพที่ 3.11 . . . . .	28
3.3 Use Case สแกนอาหาร . . . . .	28
3.4 Use Case บันทึกผลลัพธ์ น้ำตาล โซเดียมและไขมัน . . . . .	29
3.5 Use Case เข้าสู่ระบบ . . . . .	29
3.6 Use Case ออกจากระบบ . . . . .	29
3.7 Use Case ดูข้อมูลการบริโภครูปแบบวัน สัปดาห์และเดือน . . . . .	30
3.8 Use Case ลบอาหารที่เพิ่ม . . . . .	30
3.9 Use Case เพิ่มอาหารที่ไม่มีในระบบ . . . . .	30
3.10 สัญลักษณ์ของ Class Diagram . . . . .	31
3.11 อธิบาย Class Diagram ของแอปพลิเคชันบีอิงเวลเนส . . . . .	33
3.12 อธิบาย Class Diagram ของแอปพลิเคชันบีอิงเวลเนส(ต่อ) . . . . .	34
3.13 สัญลักษณ์ของ Sequence Diagram . . . . .	35
3.14 สัญลักษณ์ของโครงสร้างฐานข้อมูลแบบ Firebase . . . . .	44
3.15 อธิบายหนندที่เก็บข้อมูลอาหาร . . . . .	48
3.16 อธิบายหนندเก็บข้อมูลอาหารที่สามารถเพิ่มเข้ามา . . . . .	49
3.17 อธิบายหนندเก็บข้อมูลการบริโภคในรูปแบบวัน . . . . .	50
3.18 อธิบายหนندที่เก็บข้อมูลการบริโภคในรูปแบบสัปดาห์ . . . . .	51
3.19 อธิบายหนندที่เก็บข้อมูลการบริโภคในรูปแบบเดือน . . . . .	52
5.1 ผลการทดสอบเมนูนำทาง . . . . .	75
5.2 ผลการทดสอบหน้าแดชบอร์ด . . . . .	76
5.3 การทดสอบหน้าสแกน . . . . .	77
5.4 ผลการทดสอบหน้าเพิ่มอาหาร . . . . .	78
5.5 ผลการทดสอบหน้าเมนูตั้งค่า . . . . .	79
5.6 ผลการทดสอบความถูกต้องในการทำนาย . . . . .	85

## สารบัญภาพ

รูปที่		หน้า
2.1	การแปลงภาพแอนะล็อกให้เป็นภาพดิจิทัล . . . . .	9
2.2	ภาพแบบ Binary หรือ ภาพขาว-ดำ . . . . .	10
2.3	ภาพแบบ Grayscale . . . . .	10
2.4	ภาพแบบ RGB . . . . .	11
2.5	ภาพ Neural Network โครงข่ายประสาทเทียม . . . . .	11
2.6	ตัวอย่างการทำงานของการเรียนรู้เชิงลึก . . . . .	13
2.7	โมเดลที่ถูกสอนแบบ supervised learning . . . . .	14
2.8	โมเดลที่ถูกสอนแบบ unsupervised learning . . . . .	15
2.9	โมเดลที่ถูกสอนแบบ Reinforcement learning . . . . .	16
2.10	ตัวอย่างของโครงข่ายประสาท . . . . .	17
3.1	โครงสร้างการทำงานขั้นระบบ . . . . .	20
3.2	หน้าจอ เข้าสู่ระบบ . . . . .	21
3.3	หน้าจอ สมัครสมาชิก . . . . .	22
3.4	หน้าจอ แดชบอร์ด . . . . .	22
3.5	หน้าจอ สแกนอาหาร . . . . .	23
3.6	หน้าจอ แสดงข้อมูลอาหาร . . . . .	23
3.7	หน้าจอ แสดงอาหารที่เพิ่มโดยผู้ใช้ . . . . .	24
3.8	หน้าจอ สำหรับเพิ่มอาหาร . . . . .	24
3.9	หน้าจอ แก้ไขอาหาร . . . . .	25
3.10	หน้าจอ จัดการข้อมูลผู้ใช้ . . . . .	25
3.11	Use Case Diagram ของแอปพลิเคชันบีอิงเวลเนส . . . . .	27
3.12	Class Diagram ของแอปพลิเคชันบีอิงเวลเนส . . . . .	32
3.13	Sequence Diagram การเข้าสู่ระบบ . . . . .	36
3.14	Sequence Diagram การสแกน . . . . .	38
3.15	Sequence Diagram ดูข้อมูลการบริโภครูปแบบวัน สัปดาห์และเดือน . . . . .	40
3.16	Sequence Diagram ลบอาหารที่เพิ่ม . . . . .	42
3.17	โครงสร้างฐานข้อมูลแบบ Firebase . . . . .	45
3.18	โครงสร้างฐานข้อมูลแบบ Firebase(ต่อ) . . . . .	46
3.19	โครงสร้างฐานข้อมูลแบบ Firebase(ต่อ) . . . . .	47
3.20	โหลดเก็บข้อมูลประกาศ . . . . .	48
3.21	โหลดเก็บข้อมูลอาหารที่สมาชิกเพิ่มเข้ามา . . . . .	49
3.22	โหลดเก็บข้อมูลการบริโภคในรูปแบบวัน . . . . .	50
3.23	โหลดเก็บข้อมูลการบริโภคในรูปแบบสัปดาห์ . . . . .	51
3.24	โหลดเก็บข้อมูลการบริโภคในรูปแบบเดือน . . . . .	52
3.25	การเก็บข้อมูลภาพอาหาร . . . . .	53

3.26	คำสั่งการตั้งค่าขนาดภาพและการใช้รูปแบบการจำแนก . . . . .	54
3.27	คำสั่งการตั้งค่าขนาดภาพและการใช้รูปแบบการจำแนก . . . . .	54
3.28	คำสั่งในการแปลงไฟล์ PB ไปเป็น lite . . . . .	55
4.1	ตัวแปรในคลาส MainActivity . . . . .	56
4.2	โค้ดส่วนที่ใช้ในการสร้างเมนูนำทางหลักภายในคลาส MainActivity . . . . .	57
4.3	ตัวแปรในคลาส Main2Activity . . . . .	58
4.4	โค้ดส่วนที่ใช้ในการสร้างเมนูนำทางหลักภายในคลาส Main2Activity . . . . .	60
4.5	ตัวแปรในคลาส DashboardFragment . . . . .	61
4.6	โค้ดส่วนที่ใช้ในการสร้างหน้า สัปดาห์และเดือนภายในคลาส Dashbord . . . . .	62
4.7	โค้ดส่วนที่ใช้การตั้งค่า Database . . . . .	63
4.8	โค้ดส่วนที่ใช้ในการแสดงผลหน้าจอ Dashboard . . . . .	65
4.9	ตัวแปรในคลาส ScanFragment . . . . .	66
4.10	โค้ดส่วนที่ใช้ในการสแกน . . . . .	67
4.11	ตัวแปรในคลาส AddfoodFragment . . . . .	68
4.12	โค้ดส่วนที่ใช้แสดงผลหน้าจอเพิ่มอาหาร . . . . .	69
4.13	ตัวแปรในคลาส FoodList . . . . .	70
4.14	โค้ดการทำงานของคลาส Foodlist . . . . .	71
4.15	ตัวแปรในคลาส LoginActivity . . . . .	72
4.16	โค้ดส่วนที่ใช้ในการเข้าสู่ระบบในคลาส LoginActivity . . . . .	73
5.1	การทดสอบความแม่นยำอาหารชื่อ Cokezero . . . . .	80
5.2	การทดสอบความแม่นยำอาหารชื่อ Coke . . . . .	81
5.3	การทดสอบความแม่นยำอาหารชื่อ laysbasilflavor . . . . .	82
5.4	การทดสอบความแม่นยำอาหารชื่อ bentored . . . . .	83
5.5	การทดสอบความแม่นยำอาหารชื่อ hainanesechickenrice . . . . .	84
ก.1	หน้าเว็บดาวน์โหลด Android Studio . . . . .	90
ก.2	หน้าต่างต้อนรับของ Android Studio . . . . .	91
ก.3	หน้าต่างข้อตกลงการใช้งาน Android Studio . . . . .	91
ก.4	หน้าต่างที่จัดเก็บไฟล์ต่างๆ ของ Android Studio . . . . .	92
ก.5	หน้าต่างที่จัดเก็บไฟล์ต่างๆ ของ Android Studio . . . . .	92
ก.6	หน้าต่างผลการติดตั้ง Android Studio . . . . .	93
ก.7	หน้าเว็บดาวน์โหลด Python3.6 . . . . .	94
ก.8	การติดตั้ง Python3.6 . . . . .	94
ก.9	การติดตั้ง Tensorflow สำหรับทำงานบน CPU . . . . .	95
ก.10	การติดตั้ง Tensorflow สำหรับทำงานบน CPU และ GPU . . . . .	95
ข.1	หน้าเว็บดาวน์โหลด ESP installer package . . . . .	96
ข.2	ไฟล์ app-debug.apk บนอุปกรณ์ . . . . .	97
ข.3	หน้าจอต้อนรับการติดตั้งแอปพลิเคชันบนอุปกรณ์เอนดรอย์ . . . . .	97
ข.4	หน้าจอต้อนรับการติดตั้งแอปพลิเคชันบนอุปกรณ์เอนดรอย์ . . . . .	98

ค.1	หน้าจอแสดงการเข้าสู่ระบบ . . . . .	99
ค.2	หน้าจอแสดงการสมัครสมาชิก . . . . .	100
ค.3	หน้าจอแสดงการบริโภครูปแบบวัน . . . . .	101
ค.4	หน้าจอแสดงการบริโภครูปแบบสัปดาห์ . . . . .	102
ค.5	หน้าจอแสดงการบริโภครูปแบบเดือน . . . . .	103
ค.6	หน้าจอแสดงสแกนอาหาร . . . . .	104
ค.7	หน้าจอแสดงสแกนอาหาร . . . . .	104
ค.8	หน้าจอแสดงสแกนอาหาร . . . . .	105
ค.9	หน้าจอแสดงสแกนอาหาร . . . . .	106
ค.10	หน้าจอรายการอาหารที่เพิ่ม . . . . .	107
ค.11	หน้าจอเพิ่มอาหาร . . . . .	108
ค.12	หน้าจออัพเดตอาหารที่เพิ่ม . . . . .	108

## บทที่ 1

### บทนำ

#### 1.1 ที่มาและเหตุผล

ทุกวันนี้ประเทศไทยมีอาหารที่หลากหลาย แต่อาหารบางชนิดยังมีการปั่นอาหารสัดมาก จนเกินไป ทำให้ร่างกายได้รับสารอาหารมากเกินความจำเป็น ส่งผลให้ร่างกายเกิดโรคตามมา เช่น การปั่นอาหารสเค็มจัด ก็จะทำให้ผู้บริโภคมีความเสี่ยงต่อการเป็นโรคไต จึงมีการพัฒนาแอปพลิเคชันมาใช้ในการจัดเก็บข้อมูลการบริโภคทำให้สามารถควบคุมไม่ให้มีการบริโภคเกินความจำเป็น ของร่างกาย แต่แอปพลิเคชันยังมีข้อเสียคือ แอปพลิเคชันจะต้องมีการซื้อเพื่อที่จะสามารถใช้งานได้เต็มประสิทธิภาพและยังมีความซับซ้อนในการบันทึกข้อมูลการบริโภค

แต่ในปัจจุบันมีการนำปัญญาประดิษฐ์(Artificial Intelligence:AI) มาใช้ประโยชน์อย่างแพร่หลาย ไม่ว่าจะเป็นทางด้านการแพทย์เพื่อช่วยในการวินิจฉัยโรค ด้านการคุณภาพช่วยทำการมาช่วยในการควบคุมการทำงานของยานพาหนะหรือด้านการเกษตรที่ช่วยในการควบคุมระบบน้ำและอุณหภูมิ แต่ทางด้านอาหารยังไม่มีการนำมาใช้มากนัก

ผู้พัฒนาจึงเล็งเห็นปัญหาดังกล่าวจึงได้ทำการพัฒนาแอปพลิเคชันจัดเก็บข้อมูลการบริโภค งานวิจัยนี้ได้มีการนำปัญญาประดิษฐ์มาช่วยในการบันทึกข้อมูลการบริโภคโดยระบบจำทำกรเรียนรู้ภาพที่ทำการถ่ายเข้ามาแล้วนำมาประมวลผลเพื่อรับข้อมูลอาหารจากภาพถ่าย งานวิจัยนี้ได้นำเทคนิคทางด้านการประมวลผลภาพ(Image Processing) และเทคนิคการเรียนรู้ของเครื่อง(Machine Learning) มาใช้ในการตรวจจับอาหาร

#### 1.2 วัตถุประสงค์

- เพื่อพัฒนาแอปพลิเคชันสแกนอาหารสำหรับเก็บและแสดงข้อมูลการบริโภคอาหาร
- เพื่อสามารถบอกชื่ออาหารที่ทำการสแกนได้อย่างแม่นยำ

### 1.3 ขอบเขตของโครงการ

1. ใช้ภาพถ่ายอาหาร ที่ขายดีในเซเว่น อีเลฟเว่น จำนวน 50 ชนิด เพื่อใช้ในการฝึกสอน(train)
2. อาหารแต่ละชนิดจะใช้ภาพถ่ายจำนวน 100 รูป

### 1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ผู้ใช้ได้รับความสะดวกในการเก็บข้อมูลการบริโภคของตนเอง
2. ผู้ใช้มีการบริโภคอาหารที่ดีขึ้น
3. เป็นการวิจัยที่สามารถนำไปต่อยอดได้โดยการเพิ่มฟังก์ชันการอุปกรณ์กล้องเพื่อให้มีการดูแลสุขภาพทั้งทางด้านการกินและการออกกำลังกาย

### 1.5 เครื่องมือที่ใช้ในการพัฒนา (Development tools)

#### 1.5.1 ฮาร์ดแวร์

##### 1. สมาร์ทโฟน (Smart phone)

- ระบบปฏิบัติการแอนดรอยด์ 9.0 หรือ API Level 21
- หน่วยประมวลผลกลาง Qualcomm Snapdragon 660 AIE Octa Core ความเร็ว 2.2 กิกะเฮิร์ตซ์ (Gigahertz, GHz)
- หน่วยความจำหลักอย่างน้อย 4 กิกะไบต์ (Gigabyte, GB)
- หน่วยความจำสำรองอย่างน้อย 32 กิกะไบต์ (Gigabyte, GB)
- หน้าจอแสดงผลความละเอียดอย่างน้อย 2160 x 1080 พิกเซล (Pixel)
- หน้าจอแสดงผลขนาดอย่างน้อย 5.99 นิ้ว
- กล้องถ่ายรูปความละเอียดอย่างน้อย 20 เมกกะพิกเซล (Megapixel)

##### 2. เครื่องคอมพิวเตอร์ส่วนบุคคล (Personal computer)

- ระบบปฏิบัติการ macOS mojave
- หน่วยประมวลผลกลาง(cpu) Intel Core i5 ความเร็ว 2.3 กิกะเฮิร์ตซ์ (Gigahertz, GHz)
- หน่วยประมวลผลกราฟฟิก Intel Iris Plus Graphics 640 ความจำ 1.5 กิกะไบต์ (Gigabyte, GB)
- หน่วยความจำหลัก 8 กิกะไบต์ (Gigabyte, GB)

- หน่วยความจำสำรอง 256 กิกะไบต์ (Gigabyte, GB)

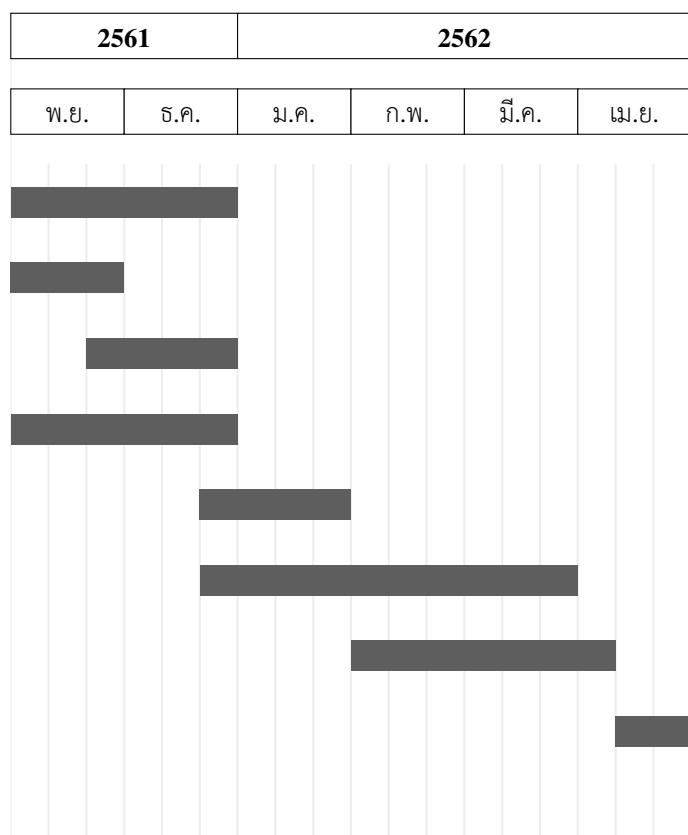
#### 1.5.2 ซอฟต์แวร์ (Software)

1. Java เป็นภาษาโปรแกรมที่ใช้ในการพัฒนา
2. Android Studio เป็น IDE (Integrated Development Environment) ใช้พัฒนาแอปพลิเคชันสำหรับระบบปฏิบัติการแอนดรอยด์
3. Tensorflow Lite เป็น Library ใช้ในการสร้างโมเดล ในการทำ Machine Learning

### 1.5.3 แผนการดำเนินการ

ในการพัฒนาแอปพลิเคชันสแกนอาหาร ผู้พัฒนาได้แบ่งขั้นตอนการดำเนินงานไว้ด้วยกัน 8 ขั้นตอน ดังต่อไปนี้

ตารางที่ 1.1: ขั้นตอนการดำเนินงาน



## บทที่ 2

### ทฤษฎีที่เกี่ยวข้อง

บทนี้จะเป็นรายละเอียดเกี่ยวกับทฤษฎีและงานวิจัยที่เกี่ยวของกับการพัฒนาแอปพลิเคชันในครั้งนี้ โดยแบ่งเนื้อหาออกเป็น ส่วนได้แก่ ส่วนที่ 1 เครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชัน หัวข้อที่ (2.1) ส่วนที่ 2 ความรู้ภาษาที่ใช้ในการพัฒนาระบบ หัวข้อที่ (2.2-2.3) ส่วนที่ 3 ความรู้ของ Library ที่มีพัฒนาแอปพลิเคชัน หัวข้อที่ (2.4-2.6) ส่วนที่ 4 ความรู้ฐานข้อมูลที่ใช้พัฒนาแอปพลิเคชัน หัวข้อที่ (2.7)

#### 2.1 ความรู้พื้นฐานระบบปฏิบัติการแอนดรอยด์

แอนดรอยด์ [1] แอนดรอยด์ (Android) คือระบบปฏิบัติการแบบเปิดเผยแพร่องค์รวม (Open Source) โดยบริษัท กูเกิล (Google Inc.) ที่ได้รับความนิยมเป็นอย่างสูง เนื่องจากอุปกรณ์ที่ใช้ระบบปฏิบัติการแอนดรอยด์ มีจำนวนมาก อุปกรณ์มีหลากหลายระดับ หลากหลายราคা รวมทั้งสามารถทำงานบนอุปกรณ์ที่มีขนาดหน้าจอ และความละเอียดแตกต่างกันได้ ทำให้ผู้บริโภคสามารถเลือกได้ตามต้องการ

หากมองในทิศทางสำหรับนักพัฒนาโปรแกรม (Programmer) แล้วนั้น การพัฒนาโปรแกรมเพื่อใช้งานบนระบบปฏิบัติการแอนดรอยด์ ไม่ใช่เรื่องที่ยาก เพราะมีข้อมูลในการพัฒนาร่วมทั้ง Android SDK (Software Development Kit) เตรียมไว้ให้กับนักพัฒนาได้เรียนรู้ และเมื่อ\_nักพัฒนาต้องการจะเผยแพร่หรือจำหน่ายโปรแกรมที่พัฒนาแล้วเสร็จ แอนดรอยด์ก็ยังมีตลาดในการเผยแพร่โปรแกรม ผ่าน Android Market แต่หากจะกล่าวถึงโครงสร้างภาษาที่ใช้ในการพัฒนานั้น สำหรับ Android SDK จะยึดโครงสร้างของภาษาจาวา (Java language) ในการเขียนโปรแกรม เพราะโปรแกรมที่พัฒนามาได้จะต้องทำงานอยู่ภายใต้ Dalvik Virtual Machine เช่นเดียว กับโปรแกรมจาวา ที่ต้องทำงานอยู่ภายใต้ Java Virtual Machine (Virtual Machine เปรียบได้ กับสภาพแวดล้อมที่โปรแกรมทำงานอยู่)

### 2.1.1 ประวัติความเป็นมาของระบบปฏิบัติการแอนดรอยด์

ถูกพัฒนามาจากบริษัท แอนดรอยด์ (Android Inc.) เมื่อปี พ.ศ 2546 โดยมีนาย แอนดี้ รูบิน (Andy Rubin) ผู้ให้กำเนิดระบบปฏิบัติการนี้ และถูกบริษัท กูเกิล ซื้อกิจการเมื่อเดือนสิงหาคม ปี พ.ศ 2548 โดยบริษัทแอนดรอยด์ ได้กลายเป็นมาบริษัทลูก ของบริษัทกูเกิล และยังมีนาย แอนดี้ รูบิน ดำเนินงานอยู่ในทีมพัฒนาระบบปฏิบัติการต่อไป ระบบปฏิบัติการแอนดรอยด์ เป็นระบบปฏิบัติการที่พัฒนามาจากการนำเอ้า แกนกลางของระบบปฏิบัติการลินุกซ์ (Linux Kernel) ซึ่ง เป็นระบบปฏิบัติการที่ออกแบบมาเพื่อทำงานเป็นเครื่องให้บริการ (Server) มาพัฒนาต่อ เพื่อให้ กลายเป็นระบบปฏิบัติการบนอุปกรณ์พกพา (Mobile Operating System) ต่อมาเมื่อเดือน พฤษภาคม ปี พ.ศ 2550 บริษัทกูเกิล ได้ทำการก่อตั้งสมาคม OHA (Open Handset Alliance) เพื่อ เป็นหน่วยงานกลางในการกำหนดมาตรฐานกลาง ของอุปกรณ์พกพาและระบบปฏิบัติการแอนดรอยด์ โดยมีสมาชิกในช่วงก่อนตั้งจำนวน 34 รายเข้าร่วม ซึ่งประกอบไปด้วยบริษัทชั้นนำที่ดำเนินธุรกิจด้านการสื่อสาร เช่น โรงงานผลิตอุปกรณ์พกพา, บริษัทพัฒนาโปรแกรม, ผู้ให้บริการสื่อสาร และผู้ผลิตอะไหล่ อุปกรณ์ด้านสื่อสาร [2]

## 2.2 ความรู้พื้นฐาน Java

Java [3] เป็นภาษาเขียนโปรแกรมเพื่อวัตถุประสงค์ที่นำไป โดยสามารถทำงานได้พร้อมกัน เป็นภาษาที่สร้างมาจากคลาส และสนับสนุนการเขียนโปรแกรมแบบออบเจ็คอย่างสมบูรณ์ และถูกออกแบบมาให้พร้อมสำหรับการใช้งานมากที่สุด ซึ่งมีเมธอดและคลาสต่าง ๆ อำนวยความสะดวก ให้เข้ากามาย โดยภาษา Java นั้นมีความตั้งใจว่าจะทำให้นักพัฒนาออกแบบและพัฒนาโปรแกรม น้อยลง นั่นคือการเขียนเพียงครั้งเดียว แต่นำไปใช้งานได้ทุกที่หรือทุกแพลตฟอร์ม

แอปพลิเคชันของภาษา Java นั้นโดยปกติแล้วจะคอมไพล์เป็น bytecode ที่สามารถรันได้ใน Java virtual machine (JVM) ขึ้นกับสถานะปัจยกรรมของคอมพิวเตอร์นั้นๆ และในปี 2016 ภาษา Java เป็นภาษาที่ได้รับความนิยมและใช้มากที่สุดในโลก โดยเฉพาะการใช้พัฒนาเว็บแอปพลิเคชัน

### 2.2.1 ประวัติความเป็นมาของภาษา Java

James Gosling Mike Sheridan และ Patrick Naughton ได้เริ่มก่อตั้งโปรเจ็คภาษา Java เมื่อปี 1991 โดยในตอนแรกถูกพัฒนาสำหรับทีวีที่สามารถมีปฏิสัมพันธ์ได้ เช่น เล่นเกมในทีวี แต่ ประสบปัญหาในการที่จะใช้งานกับสายเคเบิลของทีวีดิจิตอล ในตอนแรกภาษา Java ใช้ชื่อว่า Oak เพราะว่ามีต้นโอ๊คยืนอโภกไปยังออฟฟิศของ Gosling ต่อมาก็เปลี่ยนชื่อว่า Green และในตอนท้ายใช้ชื่อว่า Java มีที่มาจากการแฟป Java

ภาษาได้รับการออกแบบให้มีรูปแบบทางภาษาเหมือนภาษา C และ C++ ซึ่งทำให้โปรแกรมเมอร์ส่วนมากนั้นคุ้นเคยกับมันได้ดีขึ้น และ Sun Microsystems เผยแพร่ Java 1.0 ในปี 1995 โดยมีคำกล่าวว่า "Write Once, Run Anywhere" (WORA) เขียนเพียงครั้งเดียวและสามารถนำไปใช้งานได้บนทุกแพลตฟอร์ม

## 2.3 ความรู้พื้นฐาน Python

Python [4] เป็นภาษาเขียนโปรแกรมระดับสูงที่ใช้กันอย่างกว้างขวางในการเขียนโปรแกรม สำหรับวัตถุประสงค์ที่นำไป ภาษา Python นั้นสร้างโดย Guido van Rossum และถูกเผยแพร่ครั้งแรกในปี 1991 Python นั้นเป็นภาษาแบบ interprete ที่ถูกออกแบบโดยมีปรัชญาที่จะทำให้โค้ด อ่านได้ง่ายขึ้น และโครงสร้างของภาษานั้นจะทำให้โปรแกรมเมอร์สามารถเข้าใจแนวคิดการเขียน โค้ดโดยใช้บรรทัดที่น้อยลงกว่าภาษาอย่าง C++ และ Java ซึ่งภาษานั้นถูกกำหนดให้มีโครงสร้างที่ ตั้งใจให้การเขียนโค้ดเข้าใจง่ายทั้งในโปรแกรมเล็กไปจนถึงโปรแกรมขนาดใหญ่

### 2.3.1 ประวัติความเป็นมาของภาษา Python

Python กำเนิดขึ้นตั้งแต่ 1980 และเริ่มต้นใช้งานกันช่วงธันวาคม 1989 โดยนาย Guido Van Rossum ภายใต้ Centrum Wiskunde and Informatica (CWI) ที่ Netherlands[5]

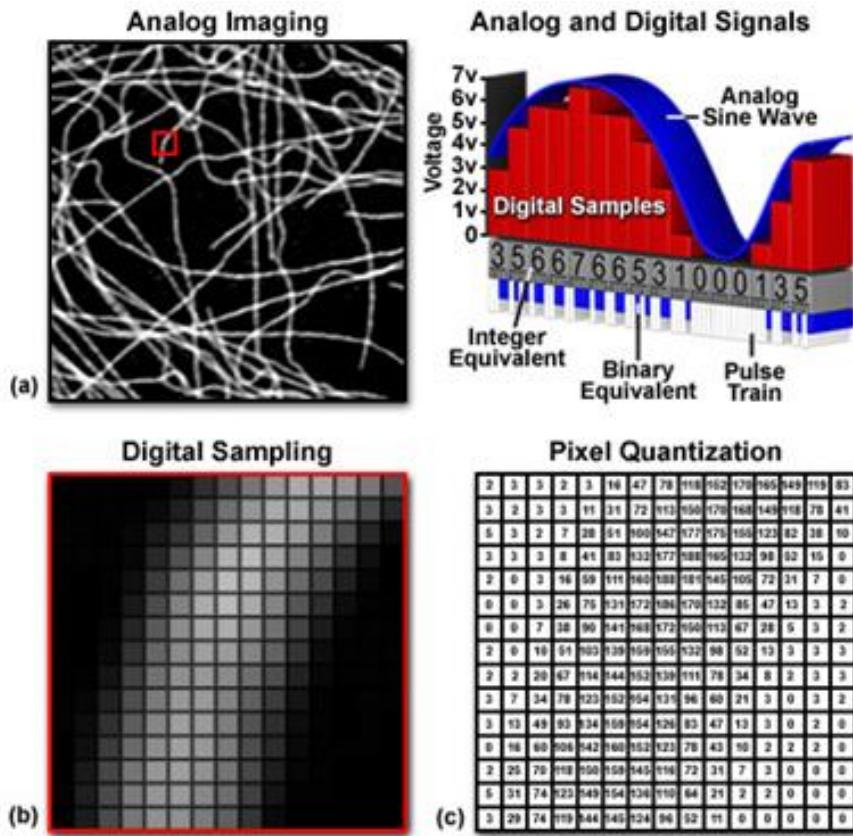
## 2.4 ความรู้พื้นฐาน Tensorflow Lite

Tensorflow Lite คือ Library ที่ถูก optimize จาก Tensorflow ทำให้มีขนาดเล็กลงทำให้สามารถทำงานบนอุปกรณ์พกพาได้ ใช้สำหรับพัฒนา Machine Learning Deep Learning และ Neural Network ได้รับการพัฒนาโดยบริษัท Google ได้ทำการเปิดตัวในงาน I/O ปี 2017 รองรับระบบปฏิบัติการ Android และ IOS ลำดับชั้นในการเรียกใช้ TensorFlow Lite และจะเรียกใช้ งานผ่านエンจิน TensorFlow Lite ที่ฝังมาในตัวแอปพลิเคชัน จากนั้น TensorFlow Lite จะเรียกใช้ไฟล์ Android Neural Networks (Android NN) ของ ระบบปฏิบัติการ ที่เริ่มมีการใส่เข้ามาใน Android 8.1 ลำดับสุดท้าย Android Neural Network จะเรียกใช้ตัวเร่งการประมวลผลที่ระดับฮาร์ดแวร์

## 2.5 ความรู้พื้นฐาน Image Processing

Image Processing [6] หรือ การประมวลผลภาพดิจิทัล เป็นการแปลงข้อมูลรูปที่เป็นสัญญาณอะนาล็อกให้อยู่ในรูปของสัญญาณดิจิทัล เพื่อใช้ในการประมวลผลผ่านทางคอมพิวเตอร์ได้ และ ยังนำมาใช้ในการลดปัญหาของภาพ เช่น ลดสัญญาณรบกวนภายในภาพเป็นต้น

ในการแปลงภาพให้เป็นสัญญาณดิจิทัลนั้น ระบบจะนำรูปที่รับเข้ามาไปคำนวณ โดยกระบวนการ Sampling และ Quantization และส่งข้อมูลออกมาในรูปแบบดิจิทัล คอมพิวเตอร์จะเก็บข้อมูลภาพลงหน่วยความจำ โดยการจองหน่วยความจำภายในเครื่องในรูปแบบของอาร์เรย์ โดยค่าในแต่ละช่องของ อาร์เรย์แสดงถึงคุณสมบัติต่างๆ ของรูปที่จุด พิกเซล นั้นๆ และตำแหน่งของช่อง อาร์เรย์ก็เป็นตัวกำหนดตำแหน่งของจุด พิกเซล ภายในภาพด้วย

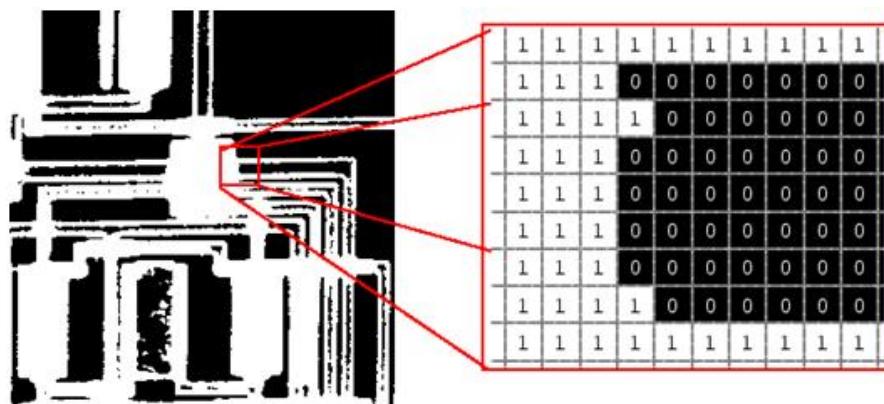
**Figure 1**

รูปที่ 2.1: การแปลงภาพแอนะล็อกให้เป็นภาพดิจิทัล

ที่มา : <https://nextsoftwares.wordpress.com/2014/05/22/>

ภาพดิจิทัลที่ได้จะมีรูปแบบการเก็บเป็นเมทริกซ์ ซึ่งจะมีการจัดเก็บภาพแต่ละชนิดต่างกัน  
ขึ้นอยู่กับระบบสีของภาพดังกล่าว โดยแบ่งชนิดของภาพได้ดังนี้

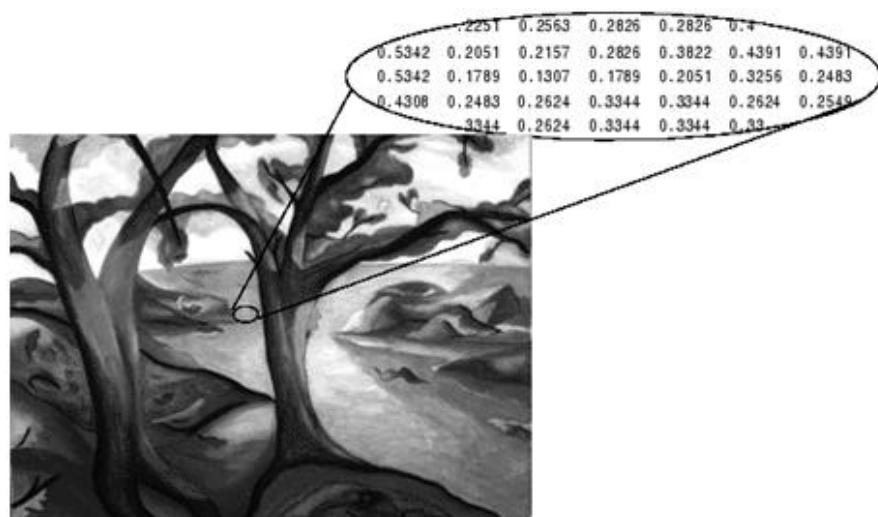
- Binary image หรือ ภาพขาว-ดำ เป็นรูปที่ใช้เนื้อที่เพียง 1 บิต ต่อ พิกเซล โดยค่าสีจะมีแค่ สีดำคือ 0 หรือสีดำ และ 1 หรือสีขาว



รูปที่ 2.2: ภาพแบบ Binary หรือ ภาพขาว-ดำ

ที่มา : <https://nextsoftwares.wordpress.com/2014/05/22/>

- Grayscale Image เป็นรูปที่เก็บโดยใช้รูปแบบของอาร์เรย์ 2 มิติ โดยค่าที่เก็บจะมีค่าอยู่ในช่วงๆหนึ่ง ซึ่งระดับของสีขึ้นอยู่กับขนาดของบิตที่ใช้เก็บค่าสี



รูปที่ 2.3: ภาพแบบ Grayscale

ที่มา : <https://nextsoftwares.wordpress.com/2014/05/22/>

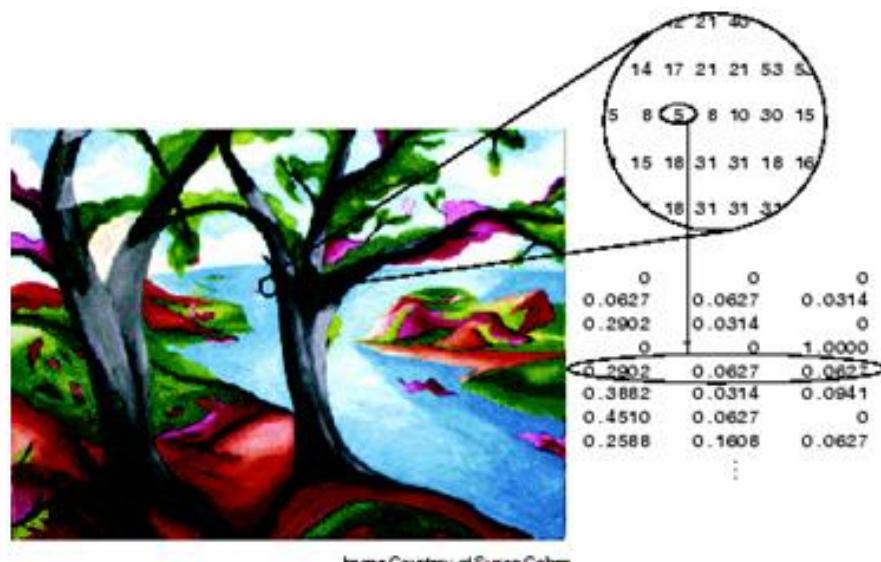
- RGB Image หรือ Truecolor Image เป็นรูปที่เก็บโดยใช้อาร์เรย์ 3 มิติ ขนาด  $m \times n \times 3$  โดยที่  $m$  คือความยาว และ  $n$  คือความกว้างของภาพในหน่วยพิกเซล ส่วนมิติสุดท้ายนั้น ในแต่ละมิติจะเก็บค่าสีแยกกัน คือสีแดง(Red) สีเขียว(Green) และสีน้ำเงิน(Blue)



รูปที่ 2.4: ภาพแบบ RGB

ที่มา : <https://nextsoftwares.wordpress.com/2014/05/22/>

- Indexed Image เป็นรูปที่มีรูปแบบการเก็บแบบ indexed คือ ภาพประเทณนี้จะเก็บค่าสีเป็น indexed และในแต่ละช่องอาร์เรย์ จะเก็บตำแหน่งของสีใน indexed นั้นๆไว้



รูปที่ 2.5: ภาพ Neural Network โครงข่ายประสาทเทียม

ที่มา : <https://nextsoftwares.wordpress.com/2014/05/22/>

## 2.6 ความรู้พื้นฐาน ปัญญาประดิษฐ์

ปัญญาประดิษฐ์ [7] หรือ Artificial Intelligence เป็นศาสตร์แขนงหนึ่งของวิทยาศาสตร์คอมพิวเตอร์ ที่เกี่ยวข้องกับวิธีการทำให้คอมพิวเตอร์มีความสามารถคล้ายมนุษย์หรือเลียนแบบพฤติกรรมมนุษย์ คือโปรแกรม Software (ซอฟแวร์) ต่าง ๆ ที่ใช้กับคอมพิวเตอร์ โดยเฉพาะความสามารถในการคิดเองได้ หรือมีปัญญานั่นเอง ปัญญานี้มนุษย์เป็นผู้สร้างให้คอมพิวเตอร์ จึงเรียกว่า ปัญญาประดิษฐ์ คำนิยาม AI ตามความสามารถที่มนุษย์ต้องการแบ่งได้ 4 กลุ่ม ดังนี้

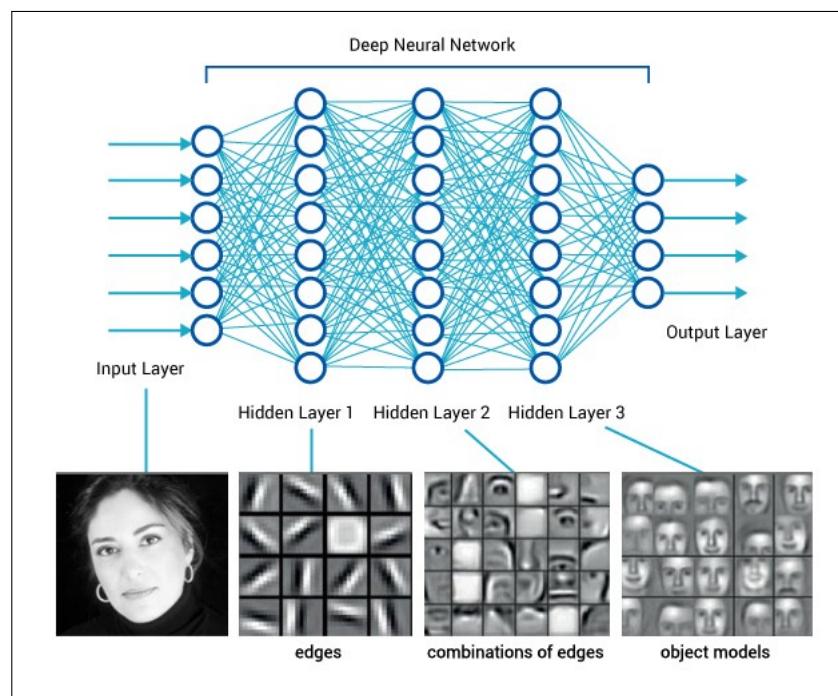
- การกระทำการคล้ายมนุษย์ Acting Humanly การสร้างเครื่องจักรที่ทำงาน, สามารถสื่อสารได้ด้วยภาษาที่มนุษย์ใช้ เช่น ภาษาไทย ภาษาอังกฤษ เช่น การแปลงข้อความเป็นคำพูด และ การแปลงคำพูดเป็นข้อความ, มีประสิทธิรับสัมผัศคล้ายมนุษย์ เช่น คอมพิวเตอร์รับภาพได้โดยอุปกรณ์รับสัมผัส แล้วนำภาพไปประมวลผล, เคลื่อนไหวได้คล้ายมนุษย์ เช่น หุ่นยนต์ช่วยงานต่าง ๆ อย่างการ ดูดฝุ่น เคลื่อนย้ายสิ่งของ และเรียนรู้ได้ โดยสามารถตรวจจับรูปแบบการเกิดของเหตุการณ์ได้ ๆ แล้วปรับตัวสู่สิ่งแวดล้อมที่เปลี่ยนไปได้
- การคิดคล้ายมนุษย์ Thinking Humanly กลไกของกิจกรรมที่เกี่ยวข้องกับความคิดมนุษย์ เช่น การตัดสินใจ การแก้ปัญหา การเรียนรู้
- คิดอย่างมีเหตุผล Thinking rationally การศึกษาความสามารถในด้านสติปัญญาโดยการใช้โมเดลการคำนวณ, การศึกษาวิธีการคำนวณที่สามารถรับรู้ใช้เหตุผล และกระทำ และใช้หลักตรรกศาสตร์ในการคิดหากำตอบอย่างมีเหตุผล เช่น ระบบผู้เชี่ยวชาญ
- กระทำอย่างมีเหตุผล Acting rationally การศึกษาเพื่อออกแบบโปรแกรมที่มีความสามารถในการกระทำ หรือเป็นตัวแทนในระบบอัตโนมัติต่าง ๆ ที่มีปัญญา, พฤติกรรมที่แสดงปัญญาในสิ่งที่มนุษย์สร้างขึ้น และการเพื่อบรรลุเป้าหมายที่ได้ตั้งไว้ เช่น โปรแกรมเล่นเกมหมากรุก ที่จะทำให้คุ้ต่อสู้แพ้ให้ได้

## 2.7 ความรู้พื้นฐานเกี่ยวกับการเรียนรู้เชิงลึก (Deep Learning)

การเรียนรู้เชิงลึก [8] คือโครงข่ายประสาทสมอง (ANN: Artificial Neuron Networks) โดยการเรียนรู้เชิงลึกและโครงข่ายประสาทสมองเป็นอัลกอริทึมที่ถูกสร้างขึ้นมาเพื่อการเรียนรู้ของเครื่อง แต่ความแตกต่างระหว่างการเรียนรู้เชิงลึกกับโครงข่ายประสาทสมองคือชั้นซ่อนตัวที่ในการเรียนรู้เชิงลึกมีชั้นซ่อนตัวมากกว่าในโครงข่ายประสาทสมองโดยโครงข่ายประสาท

สมือนนั้นอาศัยแนวคิดและเทคนิคจากการทำงานของระบบโครงข่ายประสาทในระบบประสาทของมนุษย์ โดยจำลองการทำงานเหมือนกับกลุ่มเซลล์ประสาทที่เชื่อมโยงกันเป็นระบบประสาทที่สามารถรับรู้ได้หลายสิ่งในเวลาเดียวกัน ด้วยการประมวลผลแบบขนาน (Parallel Network) ทำให้ระบบสามารถตัดสินใจได้ใกล้เคียงกับมนุษย์ในการที่เครื่องจะสามารถเข้าใจสิ่งอื่นได้ก็จำเป็นที่จะต้องมีองค์ความรู้ (Knowledge) เสียก่อนจากนั้นก็จะประเมินชุดข้อมูลและนำเสนอหรือแทนองค์ความรู้นั้น โดยมีกระบวนการทำงานดังรูปที่ 2.6

การเรียนรู้เชิงลึกถูกนำมาประยุกต์ใช้ในการทำงาน เช่น การแยกแยะใบหน้าแต่ละคน ตัวอย่างเช่นในการติดแท็กรูปภาพเพื่อนใน Facebook หรือการแยกวัตถุที่ไม่ใช่คน หรือใช้เป็นส่วนหนึ่งในระบบ团圆ตัวรุคขับ เป็นต้น



รูปที่ 2.6: ตัวอย่างการทำงานของการเรียนรู้เชิงลึก

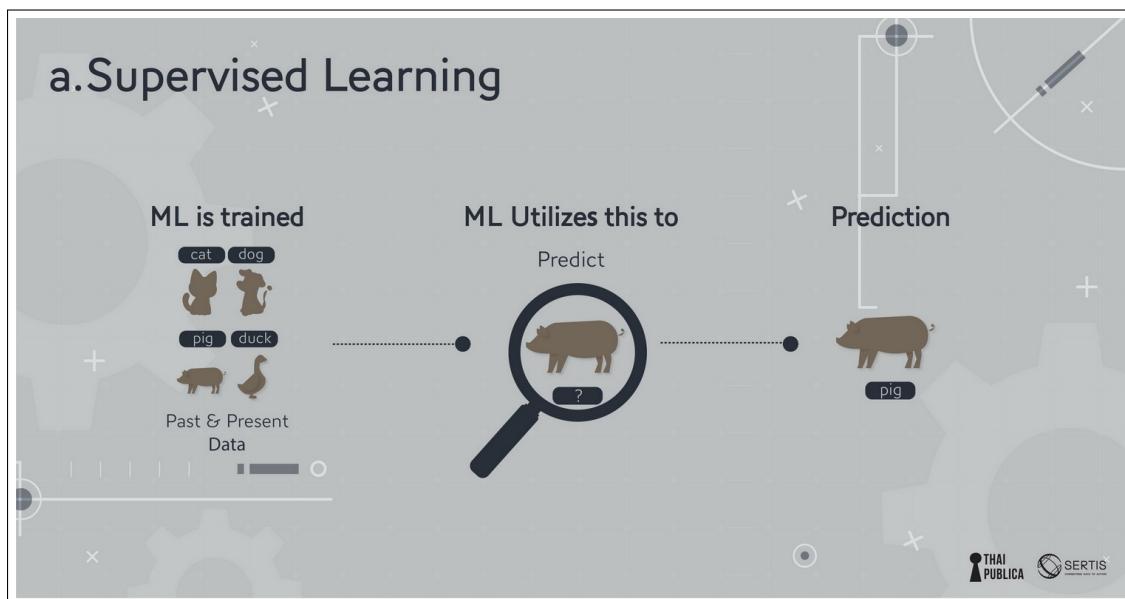
ที่มา: <https://catalystsecure.com/blog/2017/07/deep-learning-in-e-discovery-moving-past-the-hype/>

## 2.8 ความรู้พื้นฐานเกี่ยวกับการเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่อง [9] คือการทำให้คอมพิวเตอร์สามารถเรียนรู้ได้ด้วยตัวเองจากข้อมูลที่มีอยู่ ถ้าให้เบรี่ยบเทียบแบบเห็นภาพชัดเจนคือเราเป็นครู คอมพิวเตอร์เป็นนักเรียนและความรู้เป็นข้อมูล แต่เดิมเรารอฝึกสอนจะไวนักเรียน เราถูกหันหนังสือแล้วถ่ายทอดความรู้ให้กับเด็ก ๆ ซึ่งนักเรียนก็จะเข้าใจความรู้นั้นเป็นก้อน แต่การเรียนรู้ของเครื่องคือการทำให้นักเรียนสามารถใช้ความรู้ (ข้อมูลที่ตัวเองมี) ในการวิเคราะห์ เชื่อมโยง คาดการณ์และประมวลผลได้ด้วยตัวเอง โดยไม่ต้องรอให้เราสอน

การเรียนรู้ของเครื่องมีรูปแบบการเรียนรู้ 3 รูปแบบ ดังนี้

- การเรียนรู้แบบได้รับคำแนะนำ (Supervised learning) เวลาป้อนข้อมูลให้กับคอมพิวเตอร์ (Input) เช่น รูปเสือ แต่คอมพิวเตอร์มันยังไม่รู้ว่าเป็นรูปเสือ เราถูกต้องบอกมันก่อนแล้วคอมพิวเตอร์ก็จะไปวิเคราะห์ (Feature Extraction) ว่า เสือเป็นสัตว์ 4 ขา มี 2 หู 1 หาง เป็นต้น จากนั้นคอมพิวเตอร์ก็นำข้อมูลดังกล่าวไปประมวลผล/จัดหมวดหมู่ (Classification) เพื่อให้หลังจากนี้มันสามารถแยกออกได้ว่าอะไรคือเสือ อะไรไม่ใช่เสือ

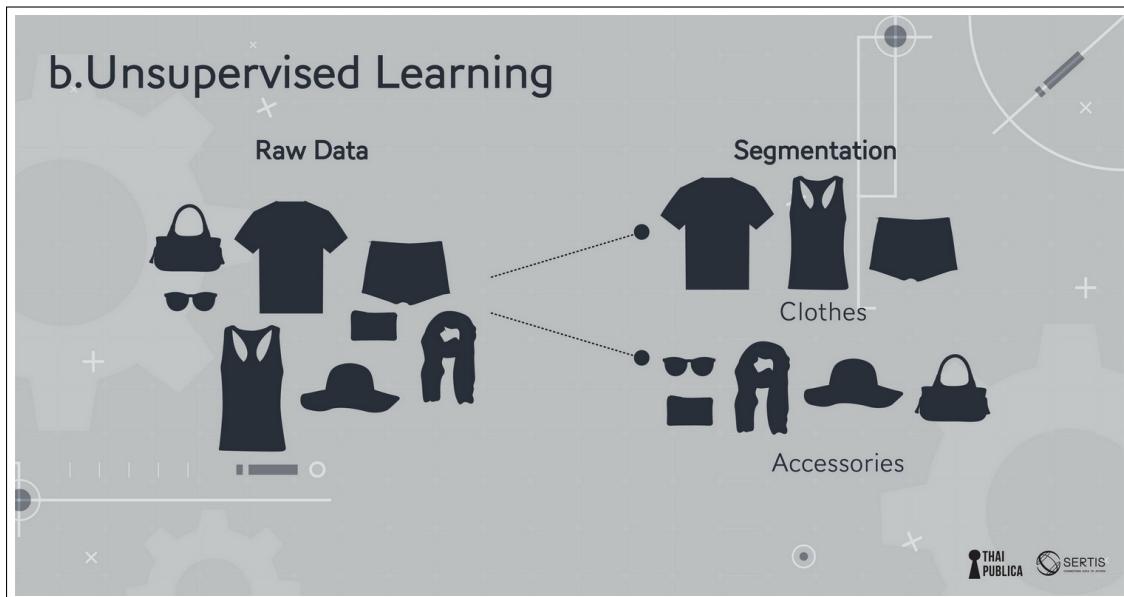


รูปที่ 2.7: โมเดลที่ถูกสอนแบบ supervised learning

ที่มา: <https://thaipublica.org/2017/07/data-driven-society12/>

- การเรียนรู้แบบไม่ได้รับคำแนะนำ (Unsupervised learning) รูปแบบนี้เรียกว่าตrang กัน

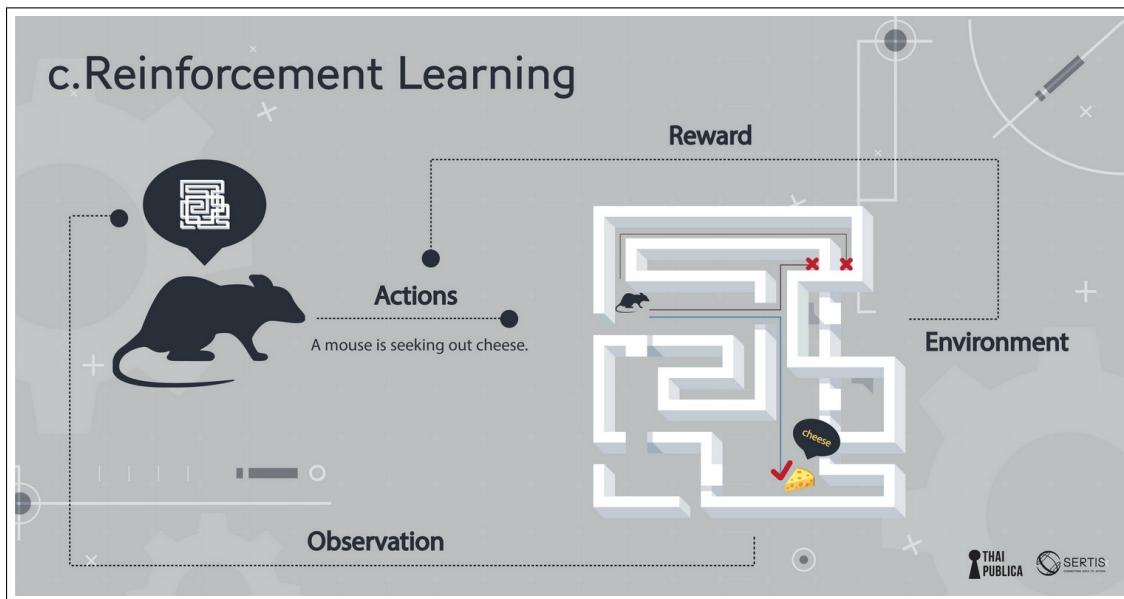
ข้ามกับรูปแบบแรก มันคือการที่เราป้อนข้อมูล (Input) รูปเสือเข้าไป แต่ไม่ได้บอกมันว่า รูปที่ป้อนเข้าไปเป็นรูปเสือ เมื่อคอมพิวเตอร์มันเอาไปวิเคราะห์ (Feature Extraction) มัน ก็วิเคราะห์ได้ว่ารูปที่ใส่เข้าไปมีลักษณะยังไง แต่คราวนี้มันไม่สามารถเอาไปประมวล/จัด หมวดหมู่ (Classification) ได้แล้ว มันจะใช้วิธีการแบ่งกลุ่มแทน (Clustering) ซึ่งคอมพิว- เตอร์มันก็จะเอารูปเสือไปอยู่กับแมว สนุก หรือสัตว์อื่น ๆ ที่มี 4 ขา มี 2 หู 1 หาง เมื่อตอน กัน



รูปที่ 2.8: โมเดลที่ถูกสอนแบบ unsupervised learning

ที่มา: <https://thaipublica.org/2017/07/data-driven-society12/>

- การเรียนรู้แบบเสริมกำลัง (Reinforcement learning) สามารถอธิบายได้ว่า เป็นการที่ เรากำหนดเงื่อนไขบางอย่างให้กับคอมพิวเตอร์ และทำให้คอมพิวเตอร์आชนาะหรือทำการตามเงื่อนไขนั้นให้ได้ ยกตัวอย่างเช่น Alpha Go เงื่อนไขของการเล่นหมากล้อมคือ ใช้มากของ ตนล้อมพื้นที่บันกระดาน เพื่อให้ได้ดินแดนมากกว่าคู่ต่อสู้ ที่นี่ Alpha Go ก็จะเรียนรู้ด้วย ตัวมันเองผ่านการจำลองการแข่งขันเป็นแสน ๆ ล้าน ๆ รอบ เพื่อให้รู้ว่า ถ้าหากคู่ต่อสู้เดิน มากนี้ ตัวมันเองจะเดินมากไปเพื่อให้บรรลุเงื่อนไขที่กำหนดไว้ให้ นั่นคือการยึดพื้นที่ บันกระดานให้ได้มากที่สุด



รูปที่ 2.9: โมเดลที่ถูกสอนแบบ Reinforcement learning

ที่มา: <https://thaipublica.org/2017/07/data-driven-society12/>

## 2.9 ความรู้พื้นฐานเกี่ยวกับโครงข่ายประสาทเทียม (Neural Network)

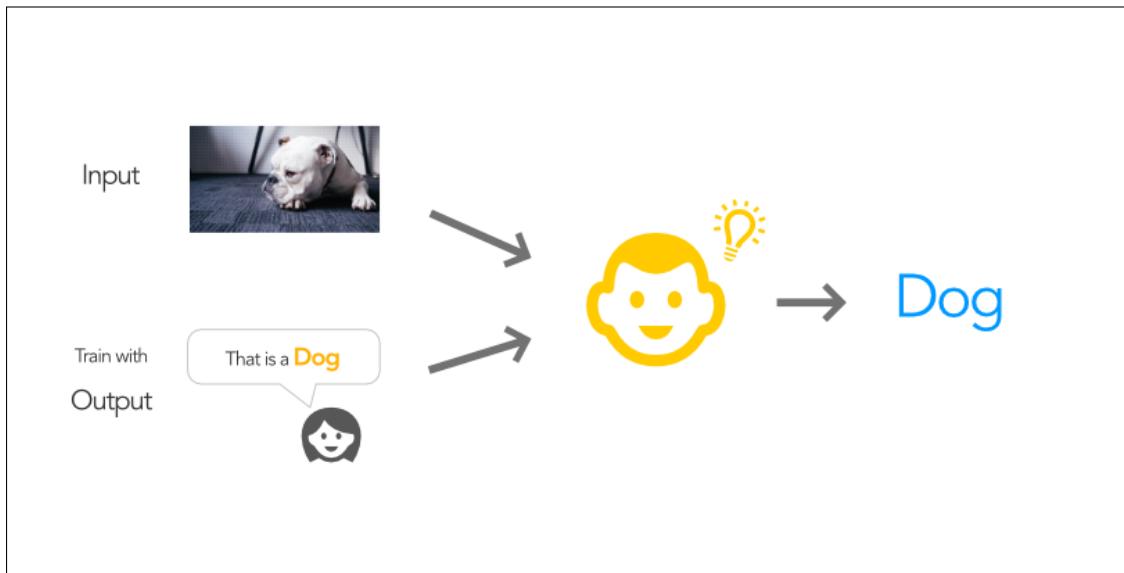
โครงข่ายประสาทเทียมคือโครงข่ายประสมามีการจำลองมาจากสมองของมนุษย์ โดยสมองของเรานั้นจะมีหน่วยประมวลผลขนาดเล็กอยู่เป็นจำนวนมาก และเชื่อมโยงกันด้วยโครงข่ายประสาท ช่วยให้มนุษย์สามารถเรียนรู้และคิดวิเคราะห์ได้อย่างรวดเร็ว แต่ในส่วนคอมพิวเตอร์นั้นไม่ได้มีโครงข่ายที่ซับซ้อนเหมือนกับสมองของมนุษย์ มันมีหน้าที่เพียงรับโปรแกรมตามคำสั่งของมนุษย์เท่านั้น ดังนั้นมีต้องการให้มันทำการเรียนรู้บางอย่าง จึงเป็นเรื่องยากในรูปแบบปกติ จึงเกิดการจำลองแนวทางการเรียนรู้ของมนุษย์ไปสู่คอมพิวเตอร์ด้วยโครงข่ายประสาทเทียม [10]

โดยส่วนที่เล็กที่สุดของโครงข่ายประสาทเทียมคือ Neuron ทำหน้าที่คำนวนข้อมูลรับเข้า เพื่อให้ได้ผลลัพธ์ออกไป โดยในรูปที่ 2.10 มีส่วนประกอบสำคัญดังนี้

- Input หรือค่าที่ส่งเข้ามาที่ Neuron โดยจะมีขาที่เข้ามาได้หลายขา ขึ้นอยู่กับเราจะสร้าง
- Weight เป็นการให้น้ำหนักของขาแต่ละที่ส่งเข้ามา โดยมีค่าระหว่าง 0-1 เมื่อเริ่มต้นจะเป็นการ Random ขึ้นมา จากนั้นตัว Neuron เมื่อทำการเรียนรู้จะเป็นการปรับ weight ตัวเอง เพื่อให้มันได้คำตอบที่ใกล้เคียงที่สุด
- Bias คือค่าที่จะช่วยเข้ามาทำให้ค่าที่เข้ามาอยู่ในระหว่าง 0 - 1 ได้ โดยจะเป็นเลข random

และปรับทุกครั้งที่เรียนรู้

- Output คือผลลัพธ์
- Back Propagation คือการที่ Neuron นำค่า Error ของ Output ที่ได้ กับ Output ที่เราสั่งให้มันเรียนรู้ นำไปปรับ Weight และ Bias ให้เกิดผลลัพธ์ที่ถูกต้องตามที่ได้เรียนรู้มา



รูปที่ 2.10: ตัวอย่างของโครงข่ายประสาท

ที่มา: <https://coladev.com/machine-learning/neural-network/2017/02/22/neural-network-basic>

## 2.10 ความรู้พื้นฐาน Firebase

Firebase [11] คือ Platform ที่รวมเครื่องมือต่าง ๆ สำหรับการจัดการในส่วนของ Backend หรือ Server side ซึ่งทำให้สามารถ Build Mobile Application ได้อย่างมีประสิทธิภาพ และยังลดเวลาและค่าใช้จ่ายของการทำ Server side หรือการวิเคราะห์ข้อมูล โดยมีทั้งเครื่องมือที่ฟรี และเครื่องมือที่มีค่าใช้จ่าย [11] เครื่องมือ Firebase มีดังนี้

1. Cloud Firestore คือ เครื่องมือ Database ที่เป็นลักษณะเป็น NoSQL โดยนำข้อมูลของ Realtime Database ของ Firebase มาพัฒนาต่อ
2. Authentication คือ เครื่องมือ Authentication ซึ่งครอบคลุมทั้ง email-password, phone ไปจนถึง facebook, twitter, github สำหรับการ Login

3. Hosting คือ hosting สำหรับ single-page web app, landing page website ซึ่งจัดการ การ Deploy และมีการติดตั้ง SSL ให้กับในส่วน Custom Domain
4. Crashlytics ช่วยจัดการ Issue ต่าง ๆ และสามารถตรวจสอบ Crash ได้ว่าเกิดขึ้นที่การทำ-งานไหนใน Mobile App
5. Performance Monitoring โดยผู้พัฒนาสามารถดู Performance ของ Code และ Network

## บทที่ 3

### การวิเคราะห์และออกแบบระบบ

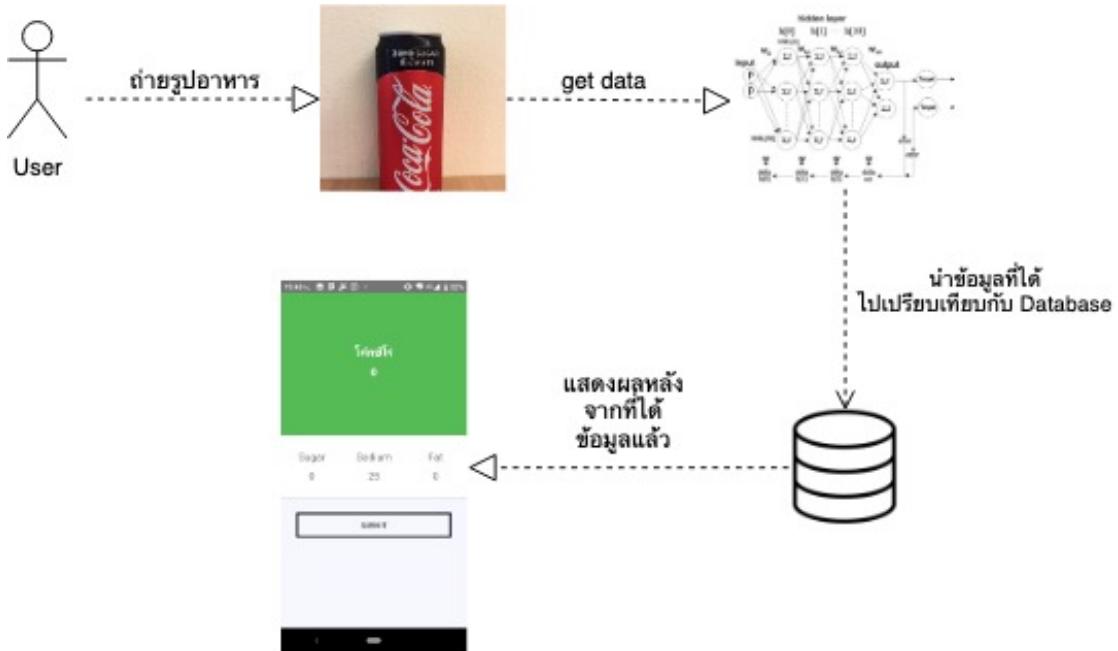
การวิเคราะห์และออกแบบระบบก่อนดำเนินการจริงเป็นอีกหนึ่งขั้นตอนที่มีความสำคัญมาก เพราะการวิเคราะห์และออกแบบระบบนั้นเป็นการกระทำที่ทำให้ผู้พัฒนาเห็นรายละเอียดส่วนย่อยของงานทั้งหมด เพิ่มประสิทธิภาพในการวางแผน การทำงาน และยังช่วยลดปัญหาที่อาจจะเกิดขึ้นในระหว่างพัฒนา เพื่อให้ระบบมีความสมบูรณ์มากยิ่งขึ้น เนื่องจากการวิเคราะห์และออกแบบระบบบัน្តันจะช่วยให้บริการ จัดการทรัพยากรได้อย่างคุ้มค่าและตรงตามความต้องการของระบบ

การวิเคราะห์และออกแบบแอปพลิเคชันสแกนอาหาร ในบทนี้จะแบ่งออกเป็น 6 ขั้นตอน เพื่อให้เห็นการดำเนินงานอย่างมีระบบ ในหัวข้อแรกจะนำเสนอภาพรวมของระบบ ก่อนจะนำเสนอเอกสารแสดงความต้องการของระบบซึ่งจะทำให้เห็นถึงมาของเพจต่าง ๆ ในขั้นตอนของการออกแบบในหัวข้อที่สาม ส่วนหัวข้อที่เหลือจะแสดงแผนภาพการการทำงานของระบบโดยใช้ UML diagram ซึ่งประกอบไปด้วย Use Case, Class และ Sequence Diagram เพื่อแสดงรายละเอียดของระบบก่อนนำไปเขียนคำสั่งด้วยภาษาโปรแกรมในบทต่อไป

- 3.1 โครงสร้างภาพรวมของระบบ (System Architecture) เป็นการออกแบบภาพรวมและเทคโนโลยีของระบบ
- 3.2 System Requirements คือ ความต้องการหรือสิ่งที่ระบบควรจะทำ หรือหน้าที่หลักของระบบที่จะต้องทำ
- 3.3 User Interface Design เป็นการออกแบบส่วนต่อประสานกับผู้ใช้
- 3.4 Use Case Diagram เป็นแผนภาพที่ใช้แสดงให้ทราบว่าระบบทำงานหรือมีหน้าที่ใดบ้าง
- 3.5 Class Diagram เป็นแผนภาพที่ใช้แสดง Class และความสัมพันธ์ระหว่าง Class
- 3.6 Sequence Diagram เป็นแผนภาพที่ใช้แสดงให้เห็นลักษณะการทำงานหรือข้อมูลระหว่างคลาส เรียงตามลำดับของเวลาที่เกิดเหตุการณ์จากน้อยไปมาก

### 3.1 รายละเอียดการออกแบบระบบ

ระบบแอปพลิเคชันสแกนอาหาร มีโครงสร้างการทำงานดังรูปที่ 3.1



รูปที่ 3.1: โครงสร้างการทำงานของระบบ

จากรูปที่ 3.1 สามารถอธิบายโครงสร้างการทำงานของระบบได้ดังนี้ ผู้ใช้งานต้องทำการถ่ายภาพอาหารที่ต้องการ ระบบจะนำค่าที่ได้จากการถ่ายภาพไปเปรียบเทียบกับข้อมูลที่มีอยู่ใน Database จากนั้นระบบจะทำการดึงข้อมูลที่มีอยู่มาแสดงผลเพื่อให้ผู้ใช้สามารถนำไปใช้งานต่อได้

## 3.2 System Requirements

### 3.2.1 Functional Requirements

แอปพลิเคชันสแกนอาหาร แบ่งตามประเภทผู้ใช้งานดังนี้

#### 1. ผู้ใช้งาน

- สามารถทำการเข้าสู่ระบบได้
- สามารถสแกนอาหารได้
- สามารถจัดการแอคเคาท์ของตนเองได้
- สามารถดูข้อมูลการบริโภคในรูปแบบวันได้

- สามารถดูข้อมูลการบริโภคย้อนหลังในรูปแบบสัปดาห์และเดือนได้
- สามารถเพิ่มอาหารที่ไม่มีอยู่ในระบบได้

### 3.2.2 Non-functional Requirements

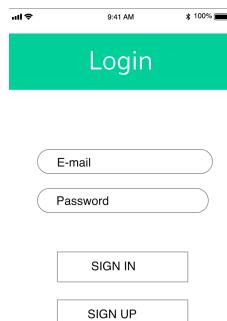
#### 1. แอปพลิเคชัน

- แอปพลิเคชันสามารถดึงข้อมูลจาก Database ได้ภายในระยะเวลา 3 วินาที
- รองรับอาหารที่ผู้ใช้เพิ่มเข้ามาอย่างน้อย 500 ข้อมูล
- รองรับการใช้งานของผู้ใช้พร้อมกันอย่างน้อย 100 คน

### 3.3 User Interface Design

ในการออกแบบ User Interface Design ของแอปพลิเคชันสแกนอาหาร ออกแบบมาให้มีลักษณะเสมือนจริงมากที่สุด โดยสามารถโต้ตอบกับผู้ใช้ได้

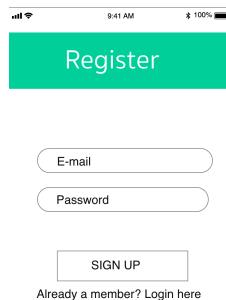
- การออกแบบหน้าจอเข้าสู่ระบบ



รูปที่ 3.2: หน้าจอ เข้าสู่ระบบ

จากการที่ 3.2 แสดงหน้าจอ เข้าสู่ระบบ เมื่อผู้ใช้ทำการติดตั้งแอปพลิเคชันครั้งแรก ระบบจะแสดงหน้านี้ขึ้นมาเพื่อให้ผู้ใช้ทำการสมัครสมาชิกและเข้าสู่ระบบ หน้าจอเข้าสู่ระบบจะแสดงเพียงครั้งเดียวหากผู้ใช้เข้าสู่ระบบแล้วจะไม่มีการแสดงหน้าจออีก จนกว่าผู้ใช้จะทำการออกจากระบบ

- การออกแบบหน้าจอสมัครสมาชิก



รูปที่ 3.3: หน้าจอ สมัครสมาชิก

จากภาพที่ 3.3 เมื่อผู้ใช้ทำการกดที่ปุ่ม SIGN UP ระบบจะทำการแสดงหน้าจอสมัครสมาชิกขึ้นมา ผู้ใช้จะต้องทำการใส่ Email address และ Password สำหรับการสมัครสมาชิกเมื่อใส่ข้อมูลครบแล้วทำการกดปุ่ม SIGN UP ในหน้าสมัครสมาชิก ก็จะเป็นการสมัครสมาชิก และเข้าสู่ระบบ

- การออกแบบหน้าจอหลัก



รูปที่ 3.4: หน้าจอ แดชบอร์ด

จากภาพที่ 3.4 หน้าแดชบอร์ด จะเป็นหน้าจอแรกที่แสดงหลังทำการเข้าสู่ระบบผู้ใช้สามารถดูข้อมูลสรุประยุวัน สปดาห์และเดือน ได้และผู้ใช้สามารถไปยังหน้าจออื่นๆได้ผ่านหน้าจอแดชบอร์ดนี้

– การออกรูปแบบหน้าจอสแกนอาหาร



รูปที่ 3.5: หน้าจอ สแกนอาหาร

จากภาพที่ 3.5 ในหน้าจอสแกนอาหารผู้ใช้สามารถทำการกดที่ปุ่มสแกน โดยให้วัตถุที่ต้องการสแกนอยู่ในกรอบของกล้อง หลังจากทำการกดปุ่มสแกนแล้ว จะมีชื่อวัตถุที่สแกนได้ปรากฏขึ้นมาแล้วระบบจะทำการดึงหน้าจอแสดง ข้อมูลอาหารขึ้นมาโดยอัตโนมัติ

– การออกรูปแบบหน้าจอแสดงข้อมูลอาหาร



รูปที่ 3.6: หน้าจอ แสดงข้อมูลอาหาร

จากภาพที่ 3.6 หน้าจอแสดงข้อมูลอาหารจะแสดงขึ้นมาหลังจากที่ผู้ใช้ทำการกดที่ปุ่มสแกนอาหาร ในหน้าจอนี้หากผู้ใช้ทำการกดปุ่ม SAVE จะเป็นการบันทึกข้อมูลอาหารลงในระบบ

- การออกรูปแบบหน้าจอแสดงอาหารที่เพิ่มโดยผู้ใช้



รูปที่ 3.7: หน้าจอ แสดงอาหารที่เพิ่มโดยผู้ใช้

จากภาพที่ 3.7 ในหน้าจอนี้จะแสดงอาหารที่ผู้ใช้เพิ่มเข้ามาด้วยผู้ใช้เอง หากกดที่ปุ่ม ADD Food จะทำการแสดงหน้าจอสำหรับเพิ่มอาหาร

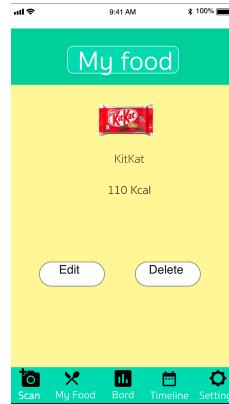
- การออกรูปแบบหน้าจอสำหรับเพิ่มอาหาร



รูปที่ 3.8: หน้าจอ สำหรับเพิ่มอาหาร

จากภาพที่ 3.8 หน้าจอเพิ่มอาหารนี้จะมีช่องให้ใส่ชื่อและข้อมูลของอาหาร หากผู้ใช้ทำการกดปุ่ม ADD จะทำการเพิ่มอาหารเข้าไปในระบบ

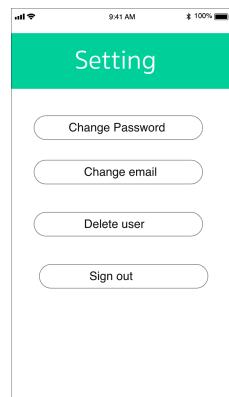
- การออกรูปแบบหน้าจอแก้ไขอาหาร



รูปที่ 3.9: หน้าจอแก้ไขอาหาร

จากภาพที่ 3.9 หน้าจอแก้ไขอาหาร ผู้ใช้ต้องทำการกดที่ชื่ออาหาร หน้าจอแก้ไขอาหารจะง่ายจะแสดงขึ้นมา ในหน้าจอแก้ไขอาหารผู้ใช้สามารถแก้ไขและลบข้อมูลอาหารได้

- การออกรูปแบบหน้าจอจัดการข้อมูลผู้ใช้



รูปที่ 3.10: หน้าจอจัดการข้อมูลผู้ใช้

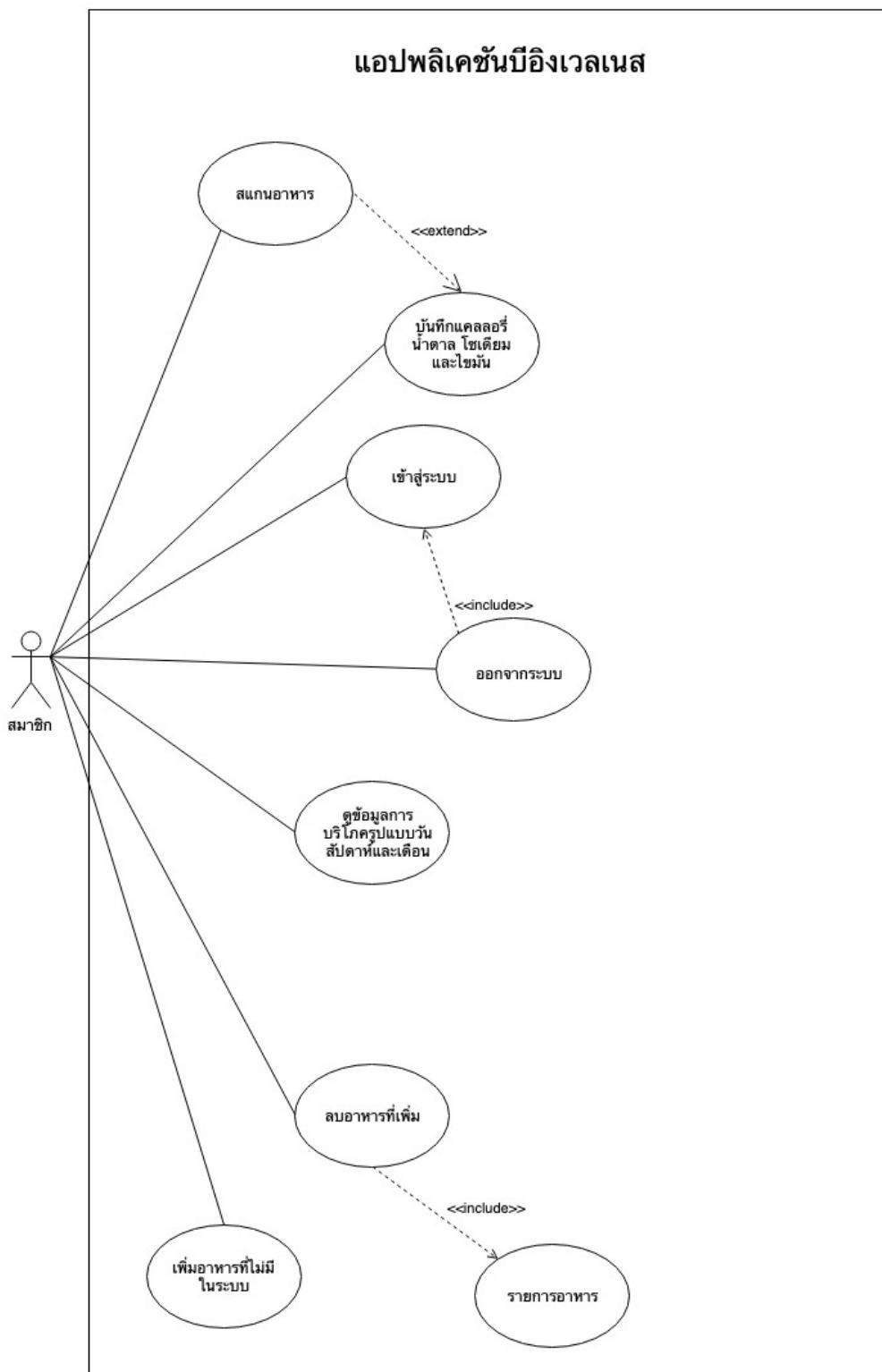
จากภาพที่ 3.10 ในหน้าจัดการข้อมูลผู้ใช้ ผู้ใช้สามารถแก้ไขอีเมล รหัสผ่าน และออกจากระบบได้

### 3.4 Use Case Diagram

Use Case Diagram เป็นแผนผังเพื่อแสดงฟังก์ชันและถูกการทำงานของระบบโดยรวม แสดงส่วนประกอบในระบบและกิจกรรมที่เกิดขึ้นในระบบซึ่งในระบบของทุนเงินให้กู้ยืมเพื่อการศึกษา คณะวิทยาศาสตร์ มหาวิทยาลัยอุบลราชธานี ผู้ใช้จำเป็นต้องเข้าสู่ระบบเพื่อใช้งานระบบ สัญลักษณ์ที่ใช้ในการเขียน Use Case Diagram แสดงในตารางที่ 3.1

ตารางที่ 3.1: สัญลักษณ์ของ Use case Diagram

สัญลักษณ์	การใช้งาน
Use case	Use case คือส่วนย่อยของระบบงาน แทนด้วยวงรีและชื่อของ Use case ภายในวงรี
	Actor คือบุคคลหรือระบบงานอื่นที่ใช้งานระบบหรือได้รับประโยชน์จากการซึ่งอยู่ภายนอกระบบ แทนด้วยรูปคนและมีชื่อบทบาทการใช้งานระบบ
	เส้นตรงที่แสดงถึงการใช้งาน Use case ของผู้กระทำ
	กรอบ สีเหลี่ยม แสดงถึงขอบเขตของระบบ โดยแสดงชื่อระบบภายในหรือด้านบนกรอบสีเหลี่ยม Use case อยู่ภายในกรอบสีเหลี่ยม และ actor อยู่ภายนอกกรอบสีเหลี่ยม
	ความสัมพันธ์แบบ «includes» แสดงว่า Use case หนึ่งดำเนินการตามขั้นตอนของ Use case อื่น โดยแทนด้วยสัญลักษณ์ลูกศรเส้นประซึ่ง Use case ที่ทางลูกศรเรียกใช้งาน Use case ที่หัวลูกศรทุกครั้งที่มีการทำงาน
	ความสัมพันธ์แบบ «extend» แสดงว่า Use case หนึ่งดำเนินการตามขั้นตอนของ Use case อื่น โดยแทนด้วยสัญลักษณ์ลูกศรเส้นประซึ่ง Use case ที่หัวลูกศรเรียกใช้งาน Use case ที่ทางลูกศรแต่การใช้งานไม่จำเป็นต้องเกิดขึ้นทุกครั้งที่มีการเชื่อมโยงกับเจือนี้ระหว่างการทำงาน



รูปที่ 3.11: Use Case Diagram ของแอปพลิเคชันบีอิงเวลเนส

ตารางที่ 3.2: อธิบาย Use Case หน้าที่ของระบบโดยทุกการทำงานจะต้องอาศัยการเข้าสู่ระบบทุกครั้ง ในภาพที่ 3.11

Use Case	คำอธิบาย
สแกนอาหาร	สมาชิกสามารถสแกนอาหารแต่ไม่จำเป็นต้องบันทึกข้อมูลการบริโภคทุกครั้ง
บันทึกแคลลอรี น้ำตาล โซเดียมและไขมัน	การบันทึกข้อมูลการบริโภคสมาชิกไม่จำต้องทำการสแกนทุกครั้ง
เข้าสู่ระบบ	สมาชิกสามารถทำการเข้าสู่ระบบได้
ออกจากระบบ	สมาชิกจะต้องทำการเข้าสู่ระบบก่อนจึงจะสามารถออกจากระบบได้
ดูข้อมูลการบริโภครูปแบบวัน สัปดาห์และเดือน	หลังจากสมาชิกทำการเข้าสู่ระบบแล้ว สมาชิกสามารถดูข้อมูลการบริโภคในรูปแบบวัน สัปดาห์และเดือนได้
ลบอาหารที่เพิ่ม	สมาชิกสามารถลบอาหารที่เพิ่มเข้ามาได้ ด้วยการเลือกที่รายการอาหาร
เพิ่มอาหารที่ไม่มีในระบบ	สมาชิกสามารถเพิ่มอาหารที่ไม่มีในระบบลงได้

ตารางที่ 3.3: Use Case สแกนอาหาร

<b>Use Case Title :</b> สแกนอาหาร	<b>Use case Id :</b> 1
Primary Actor : สมาชิก	
Main Flow : สมาชิกจะต้องทำการเข้าสู่ระบบก่อนจึงจะสามารถทำการสแกนได้	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ต จะไม่สามารถทำการสแกนได้	

ตารางที่ 3.4: Use Case บันทึกแคลลอรี่ น้ำตาล โซเดียมและไขมัน

<b>Use Case Title :</b> บันทึกแคลลอรี่ น้ำตาล โซเดียมและไขมัน	<b>Use case Id :</b> 2
Primary Actor : สมาชิก	
Main Flow : หากสมาชิกต้องการบันทึกสามารถทำการสแกนหรือบันทึกจากอาหารที่เพิ่ม ลงได้	
Exceptional Flow ที่ 1 : หากสมาชิกไม่เชื่อมต่ออินเทอร์เน็ต จะไม่สามารถทำการบันทึกข้อมูล อาหารได้	

ตารางที่ 3.5: Use Case เข้าสู่ระบบ

<b>Use Case Title :</b> ดาวน์โหลดเอกสาร	<b>Use case Id :</b> 3
Primary Actor : สมาชิก	
Main Flow : สมาชิกต้องทำการสมัครสมาชิกก่อนจึงจะสามารถทำการเข้าสู่ระบบได้	
Exceptional Flow ที่ 1 : หากสมาชิกไม่เชื่อมต่ออินเทอร์เน็ต จะไม่สามารถทำการเข้าสู่ระบบได้	
Exceptional Flow ที่ 2 : หากสมาชิกไม่ทำการสมัครสมาชิกจะไม่สามารถทำการเข้าสู่ระบบได้	

ตารางที่ 3.6: Use Case ออกจากระบบ

<b>Use Case Title :</b> ออกจากระบบ	<b>Use case Id :</b> 4
Primary Actor : สมาชิก	
Main Flow : สมาชิกจะต้องทำการเข้าสู่ระบบก่อนจึงจะสามารถออกจากระบบได้	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่ทำการเข้าสู่ระบบ จะไม่สามารถออกจากระบบได้	

ตารางที่ 3.7: Use Case ดูข้อมูลการบริโภครูปแบบวัน สัปดาห์และเดือน

<b>Use Case Title :</b> ดูข้อมูลการบริโภครูปแบบวัน สัปดาห์และเดือน	<b>Use case Id :</b> 5
Primary Actor : สมาชิก	
Main Flow : เมื่อสมาชิกทำการเข้าสู่ระบบแล้ว สมาชิกสามารถดูข้อมูลการบริโภคในรูปแบบวัน สัปดาห์และเดือนได้	
Exceptional Flow ที่ 1 : หากผู้ใช้ไม่เชื่อมต่ออินเทอร์เน็ต จะไม่สามารถดูข้อมูลการบริโภคในรูปแบบวัน สัปดาห์และเดือนได้	

ตารางที่ 3.8: Use Case ลบอาหารที่เพิ่ม

<b>Use Case Title :</b> ลบอาหารที่เพิ่ม	<b>Use case Id :</b> 6
Primary Actor : สมาชิก	
Main Flow : สมาชิกสามารถทำการลบอาหารได้ด้วยการเลือกที่เมนูอาหาร	
Exceptional Flow ที่ 1 : หากสมาชิกไม่เชื่อมต่ออินเทอร์เน็ต จะไม่สามารถทำการลบอาหารได้	

ตารางที่ 3.9: Use Case เพิ่มอาหารที่ไม่มีในระบบ

<b>Use Case Title :</b> เพิ่มอาหารที่ไม่มีในระบบ	<b>Use case Id :</b> 7
Primary Actor : สมาชิก	
Main Flow : สมาชิกสามารถเพิ่มอาหารที่ไม่มีในระบบได้ด้วยการกดปุ่ม ADD	
Exceptional Flow ที่ 1 : หากสมาชิกไม่เชื่อมต่ออินเทอร์เน็ต จะไม่สามารถเพิ่มอาหารได้	

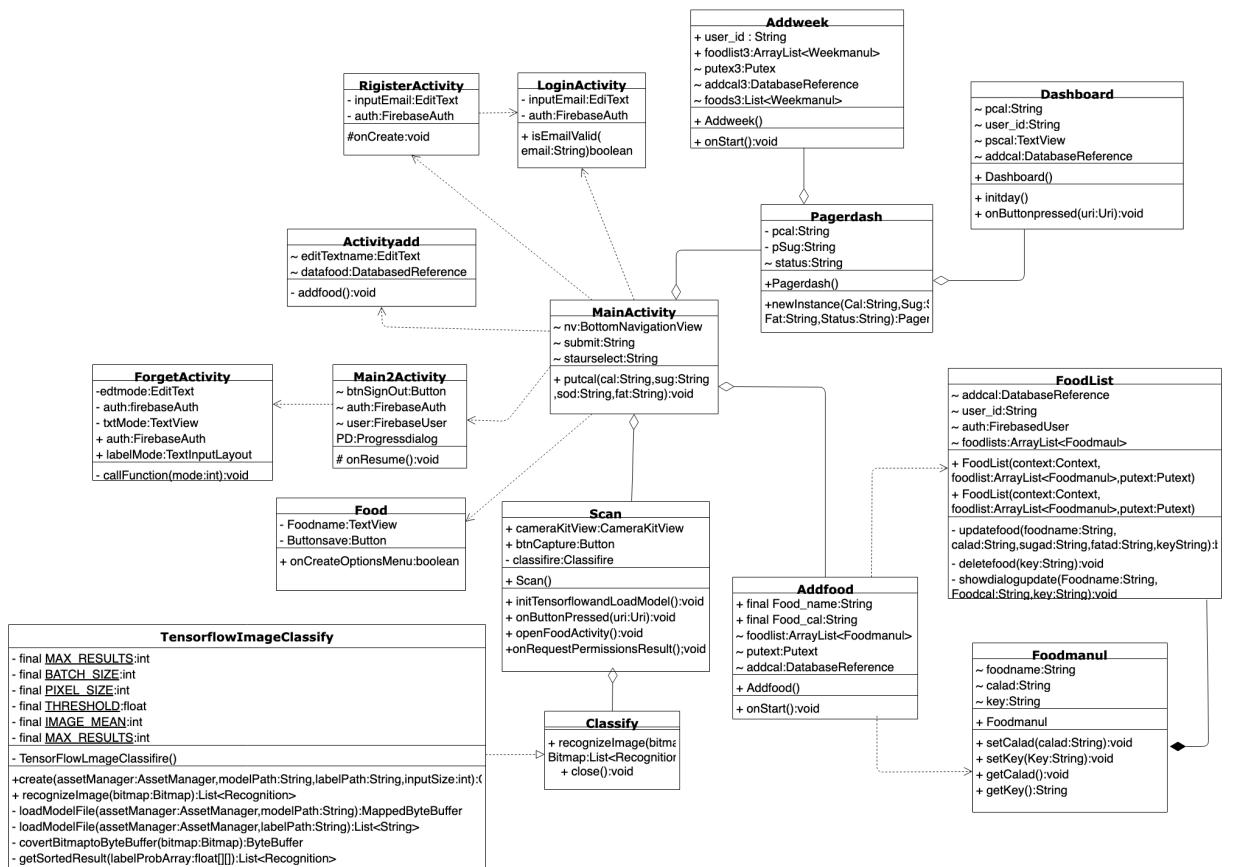
### 3.5 Class Diagram

Class Diagram คือแผนภาพที่ใช้แสดงคลาสและความสัมพันธ์ในแบบต่างๆ ระหว่างคลาส สัญลักษณ์ที่ใช้ในการเขียน Class Diagram แสดงในตารางที่ 3.10

ตารางที่ 3.10: สัญลักษณ์ของ Class Diagram

สัญลักษณ์	การใช้งาน
	<p>คลาส สัญลักษณ์แทนด้วยสี่เหลี่ยมแบ่งเป็น 3 ส่วน ส่วนบน เป็นชื่อของ class ส่วนกลาง เป็นชื่อ Attribute และ ส่วนล่าง เป็น Operation Name หรือ Method ใช้สำหรับเขียนฟังก์ชันในการทำงานของคลาสนั้น ๆ ชนิดของ Visibility ของ Method และ Attribute แบ่งเป็น 3 ชนิด ได้แก่</p> <ol style="list-style-type: none"> <li>1. Public แทนสัญลักษณ์ด้วยเครื่องหมายบวก (+)</li> <li>2. Private แทนสัญลักษณ์ด้วยเครื่องหมายลบ (-)</li> <li>3. Protected แทนสัญลักษณ์ด้วยเครื่องหมายซาร์ป ()</li> </ol>
	Dependency Relationship หมายความว่า คลาสที่อยู่ผู้ตั้นลูกศร สามารถเรียกใช้คลาสที่อยู่ผู้หัวลูกศร
	Composition Relationship เป็นความสัมพันธ์ระหว่างคลาสเจ้า หรือคลาสแบบขึ้นต่อ กันและมีความเกี่ยวข้องกันเสมอ
	Realization Relationship เป็นความสัมพันธ์ระหว่าง Object หรือ Class ในลักษณะของการสืบทอดคุณสมบัติจาก Class หนึ่ง (Super class) ไปยังอีก Class หนึ่ง (Subclass)
	Connector เป็นสัญลักษณ์แทนด้วยรูปห้าเหลี่ยมและมีชื่ออู่ตรากลาง จะสร้างสัญลักษณ์นี้ไว้เมื่อต้องการเชื่อมต่อคลาสที่อยู่คนละหน้า

Class Diagram แสดงความสัมพันธ์ในรูปแบบต่างๆ ระหว่างคลาสของแอปพลิเคชันระบบกองทุนเงินให้กู้ยืมเพื่อการศึกษา คณะวิทยาศาสตร์ มหาวิทยาลัยอุบลราชธานี อธิบายได้ตามภาพที่ 3.12 ดังต่อไปนี้



รูปที่ 3.12: Class Diagram ของแอปพลิเคชันบีอิงเวลเนส

จากรูปภาพที่ 3.12 สามารถอธิบายแผนภาพ Class Diagram ได้ดังนี้

ตารางที่ 3.11: อธิบาย Class Diagram ของแอปพลิเคชันบีอิงเวลเนส

Class Diagram	คำอธิบาย
LoginActivity	คลาส LoginActivity จะถูกเรียกใช้เมื่อสมาชิกเปิดแอปพลิเคชัน โดยวัตถุประสงค์การทำงานของคลาสนี้คือ เพื่อให้สมาชิกทำการเข้าสู่ระบบและสมัครสมาชิก
RigisterActivity	คลาส RigisterActivity จะถูกใช้งานเมื่อผู้ใช้กดปุ่ม SiGN UP โดย วัตถุประสงค์การทำงานของคลาสนี้คือ เพื่อให้สมาชิกทำการสมัครสมาชิก
MainActivity	คลาส MainActivity เป็นคลาสหลักที่ใช้ในการทำงานของแอป-พลิเคชันโดยการทำงานของคลาสนี้เน้นไปที่การเรียกใช้ Fragment โดยองค์ประกอบของคลาสนี้ประกอบไปด้วยคลาสของ Fragment ได้แก่ Pagerdash Myfood Scan
Main2Activiy	คลาส Main2Activiy เป็นคลาสที่ให้สมาชิกได้เลือกการจัดการบัญชีสมาชิก ได้แก่ เปลี่ยนอีเมล เปลี่ยนรหัสผ่าน ลบผู้ใช้ และออกจากระบบ
ForgetActivity	คลาส ForgetActivity เป็นคลาสที่ทำการจัดการบัญชีสมาชิก ได้แก่ เปลี่ยนอีเมล เปลี่ยนรหัสผ่าน ลบผู้ใช้
Activityadd	คลาส Activityadd เป็นคลาสที่มีไว้สำหรับให้สมาชิกเพิ่มอาหารที่ไม่มีในระบบ
Food	คลาส Food จะถูกใช้งานเมื่อสมาชิกทำการสแกนอาหาร โดยวัตถุประสงค์การทำงานของคลาสนี้เพื่อแสดงข้อมูลอาหารที่สแกนได้

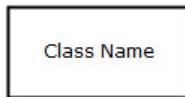
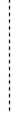
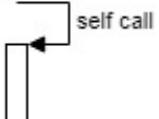
ตารางที่ 3.12: อธิบาย Class Diagram ของแอปพลิเคชันบอิงเวลเนส(ต่อ)

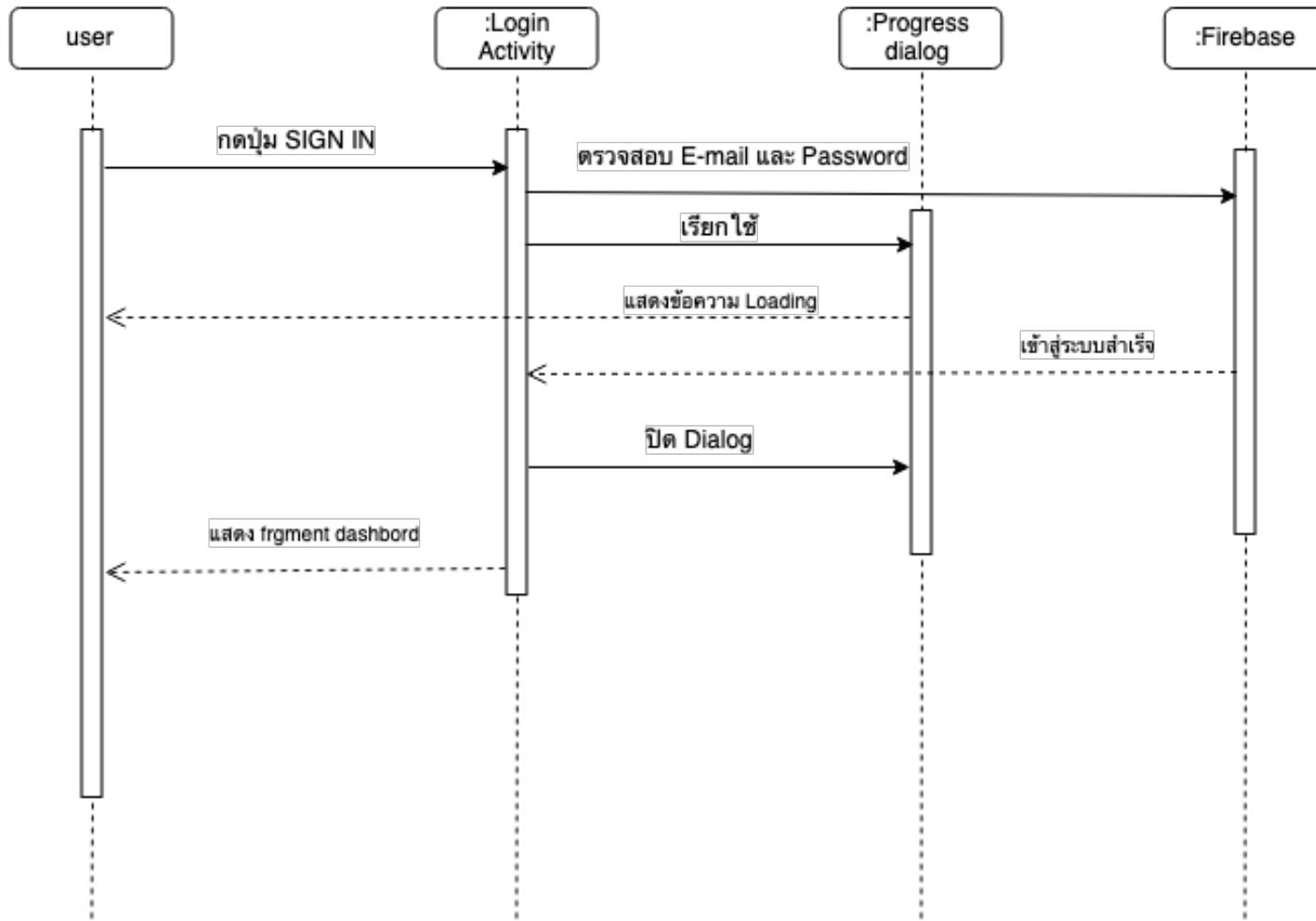
<b>Class Diagram</b>	<b>คำอธิบาย</b>
Pagerdash	คลาส Pagerdash เป็นคลาสหลักที่ใช้ในการจัดการ Tab เมนูในการเลื่อนเพื่อดูรูปแบบของข้อมูลการบริโภค
Addweek	คลาส Addweek เป็นคลาสที่จะแสดงเมื่อสมาชิกทำการเลื่อนมาที่ Tab Week เป็นคลาสที่แสดงข้อมูลการบริโภคในรูปแบบสัปดาห์
Dashboard	คลาส Dashboard เป็นคลาสหลักที่จะถูกเรียกใช้เมื่อผู้ใช้ทำการเปิดแอปพลิเคชันหลังจากทำการเข้าสู่ระบบแล้ว โดยวัตถุประสงค์การทำงานของคลาสนี้คือ แสดงข้อมูลการบริโภคในรูปแบบวัน
Addfood	คลาส Addfood เป็นคลาสที่จะแสดงเมื่อผู้ใช้เลือกเมนู My food โดยวัตถุประสงค์การทำงานของคลาสคือ แสดงรายการอาหารที่เพิ่มเข้ามา
FoodList	คลาส FoodList เป็นคลาสที่จะถูกใช้งานเมื่อคลาส Addfood ถูกใช้ โดยวัตถุประสงค์การทำงานของคลาสนี้คือ จัดรูปแบบการแสดงรายการอาหาร
Foodmanul	คลาส Foodmanul เป็นคลาสที่กำหนดค่าต่างๆ ที่จำเป็นในการสร้างรายการอาหาร
Scan	คลาส Scan เป็นคลาสที่จะแสดงเมื่อผู้ใช้เลือกเมนู Scan โดยวัตถุประสงค์การทำงานของคลาสนี้คือสแกนอาหาร
Classify	คลาส Classify เป็นคลาสที่ทำงานหลังจากสมาชิกทำการสแกน โดยวัตถุประสงค์การทำงานของคลาสคือ แยกประเภทรูปภาพที่สมาชิกทำการถ่าย
TensorflowImageClassify	คลาส TensorflowImageClassify เป็นคลาสที่ทำงานร่วมกับคลาส Classify โดยจะทำหน้าที่นำโมเดลเข้ามาเพื่อใช้ในการระบุภาพอาหารที่สมาชิกทำการถ่าย

### 3.6 Sequence Diagram

Sequence Diagram เป็น Diagram ที่แสดงขั้นตอนการทำงานของแต่ละ Use Case ระหว่าง Object ต่างๆ โดย Sequence Diagram จะช่วยให้มองเห็นการทำงานของภาพรวมของระบบ ส่วนประกอบสัญลักษณ์ที่ใช้ในการเขียน Sequence Diagram แสดงดังตารางที่ 3.13

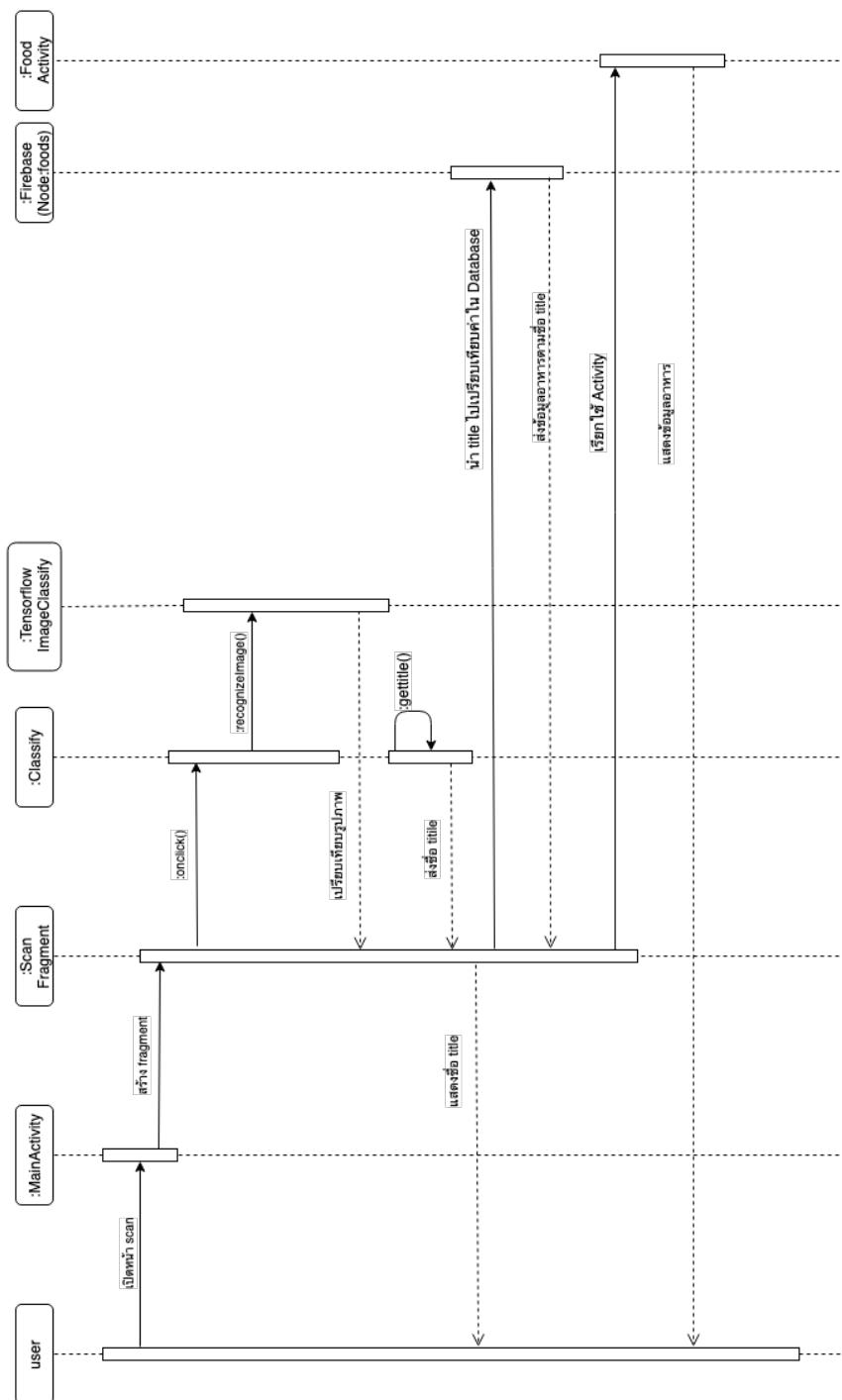
ตารางที่ 3.13: สัญลักษณ์ของ Sequence Diagram

สัญลักษณ์	การใช้งาน
	Class แสดงถึงการทำงานของ Use Case ใน การส่งหรือรับข้อมูลความแทนด้วยสัญลักษณ์สีเหลืองมีชื่อคลาสอยู่ภายใน
	Lifeline หรือเส้นอายุขัย แสดงช่วงเวลาตั้งแต่เริ่มสร้าง object ในคลาสนั้น จนกระทั่ง object นั้นถูกทำลาย สัญลักษณ์แทนด้วยเส้นประ
	Focus of control หรือจุดควบคุม เป็นจุดควบคุมที่ object ใช้ทำการส่งหรือรับข้อมูล สัญลักษณ์แทนด้วยสีเหลือง
	Message คือ ข้อมูลที่รับส่งระหว่าง Object สัญลักษณ์แทนด้วยลูกศร และประกอบด้วย 2 ส่วน คือ ข้อมูล (Data) และฟังก์ชัน (Function)
	Return Message เป็นข้อมูลที่ส่งกลับหลังจากทำงานเสร็จ
	Self call เป็นการเรียกฟังก์กการทำงานภายในตัวเอง
	สร้างกรอบการทำงานของโปรแกรม เพื่อให้รู้ขอบเขตของการทำงาน เช่น ลูป(loop)



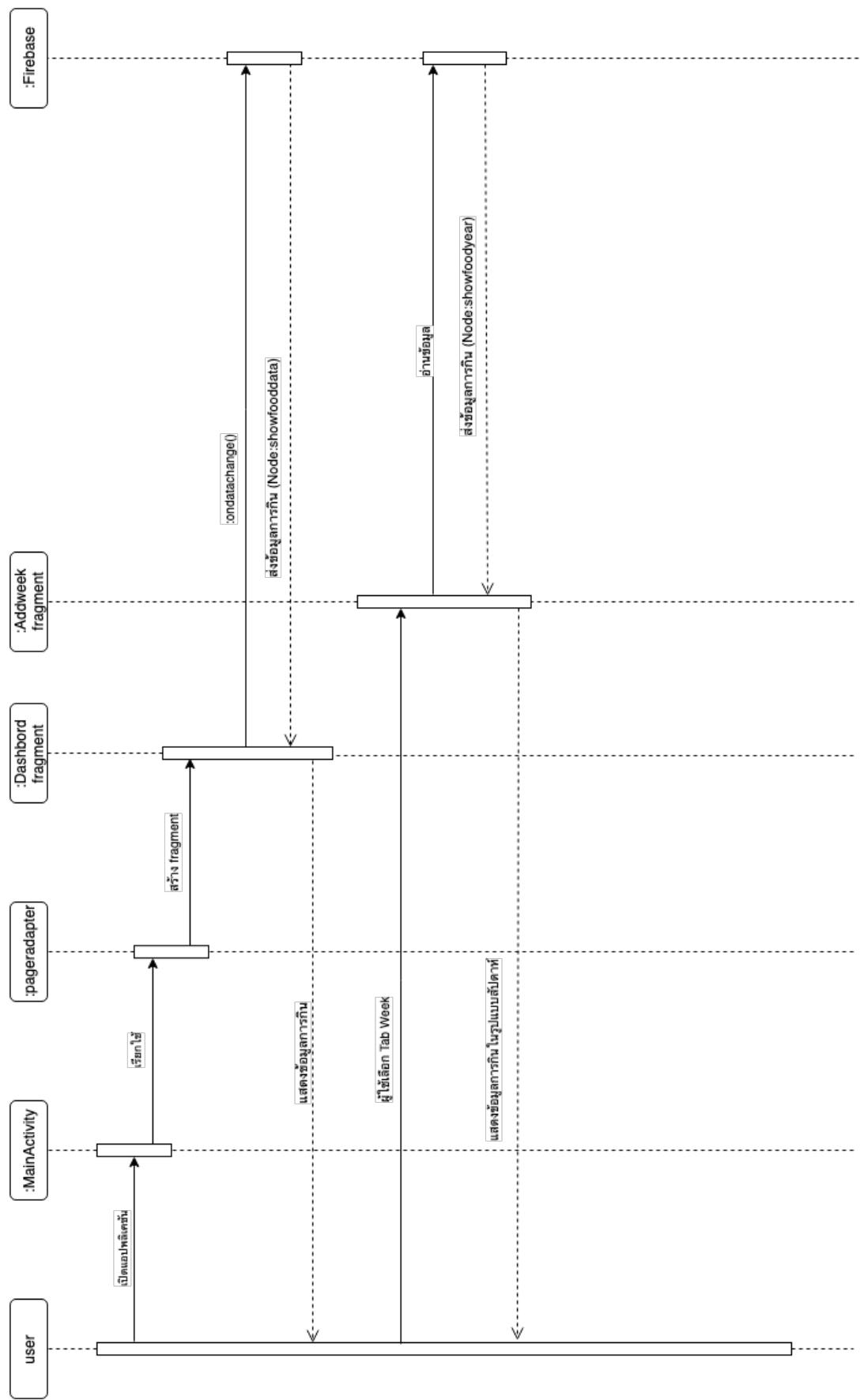
รูปที่ 3.13: Sequence Diagram การเข้าสู่ระบบ

จากภาพที่ 3.6 สามารถอธิบายแผนภาพ Sequence Diagram การเข้าสู่ระบบ ได้ดังนี้ เมื่อสมาชิกกดปุ่ม SIGN IN ที่คลาส LoginActivity จะทำการตรวจสอบ E-mail และ Password โดยส่งไปที่ Firebase ในขณะเดียวกันก็จะเรียก Progress dialog แสดงข้อความไปยังสมาชิกว่า Loading เมื่อเข้าสู่ระบบสำเร็จทำการปิด Dialog และแสดง fragment dashbord ที่สมาชิก



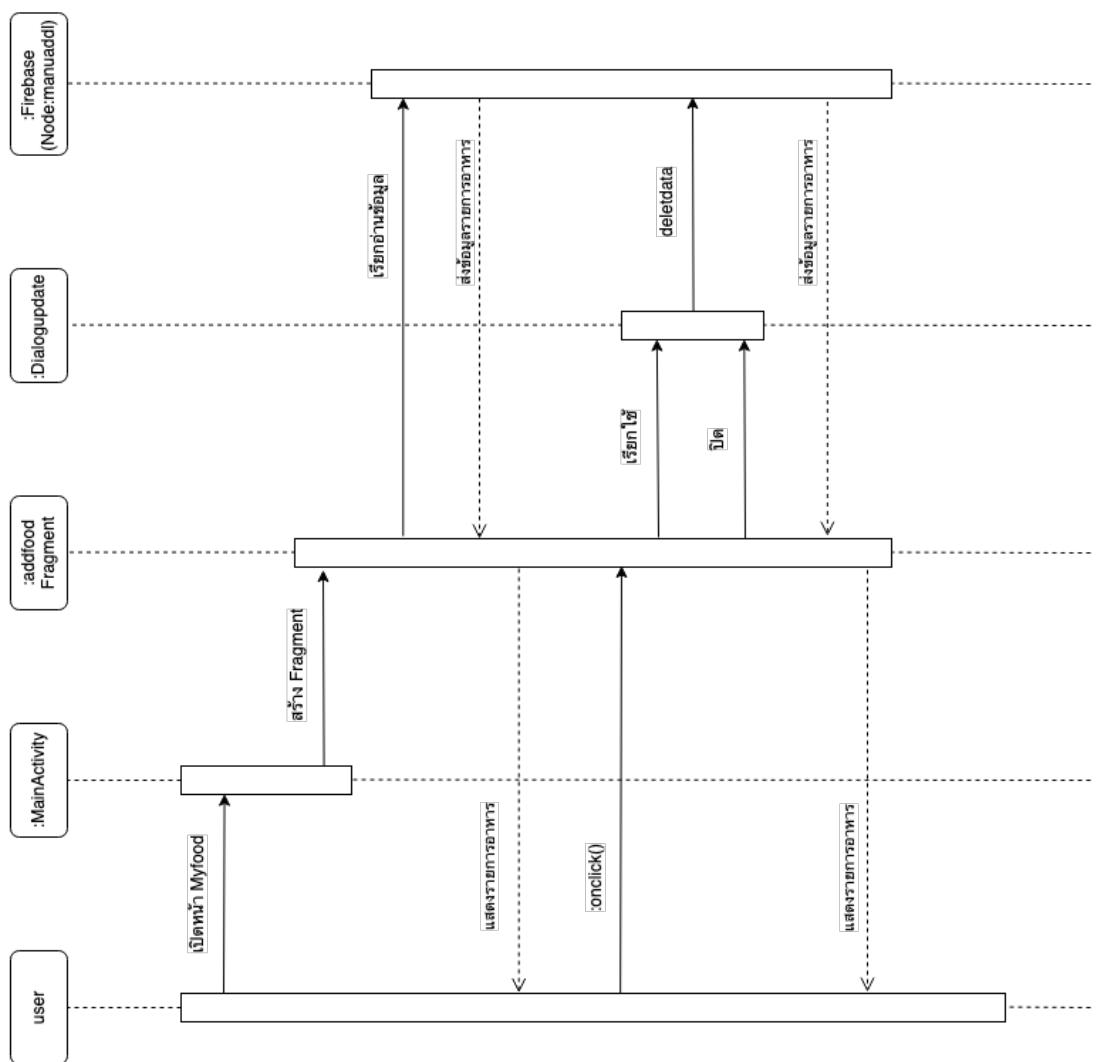
ຮູບທີ 3.14: Sequence Diagram ກາງແຕກໄມ

จากภาพที่ 3.14 สามารถอธิบายแผนภาพ Sequence Diagram การสแกน ได้ดังนี้ เมื่อスマชิกทำการเปิดหน้าสแกน คลาส MainActivity จะทำการสร้าง Fragment Scan ขึ้นมาและเมื่อスマชิกกดปุ่ม Scan พังก์ชัน Onclick() จะทำการส่งข้อมูลไปยังคลาส Classify จากนั้นคลาส Classify จะทำการ :recognizeImage() แล้วให้คลาส TensorflowImageClassify ทำการเปรียบเทียบรูปภาพ หลังจากการเปรียบเทียบภาพแล้วที่ Fragment Scan จะใช้เมธอด gettitle() แล้วส่งชื่อ title ที่ได้มาแสดงให้หัวสมาชิก และนำ title ที่แสดงไปเปรียบกับค่าใน Firebased ที่โหนด foods แล้วส่งข้อมูลหารที่ตรงกันกับ title จากนั้นเรียกใช้ FoodActivity ที่แสดงข้อมูลอาหารให้กับสมาชิก



รูปที่ 3.15: Sequence Diagram ดูข้อมูลการปรีกครุประบบวัน สำหรับเดือนต่อไป

จากภาพที่ 3.15 สามารถอธิบายแผนภาพ Sequence Diagram ดูข้อมูลการบริโภครูปแบบ  
วัน สัปดาห์ และเดือน ได้ดังนี้ เมื่อสมาชิกเปิดแอปพลิเคชันคลาส MainActivity จะเรียกใช้ Pagerdash  
ซึ่งประกอบไปด้วย Fragment ดังนี้ Dashboard Addweek และ Addmonth จากนั้นแต่ละ Fragment  
จะทำเมธอด ondatachange() ด้วยการอ่านข้อมูลที่ Firebase โดย Showfooddata Showfoodyear  
และ ShowfoodMonth ตาม Tab ที่ผู้ใช้เลือกดู



รูปที่ 3.16: Sequence Diagram ลิบโอหารที่เพิ่ม

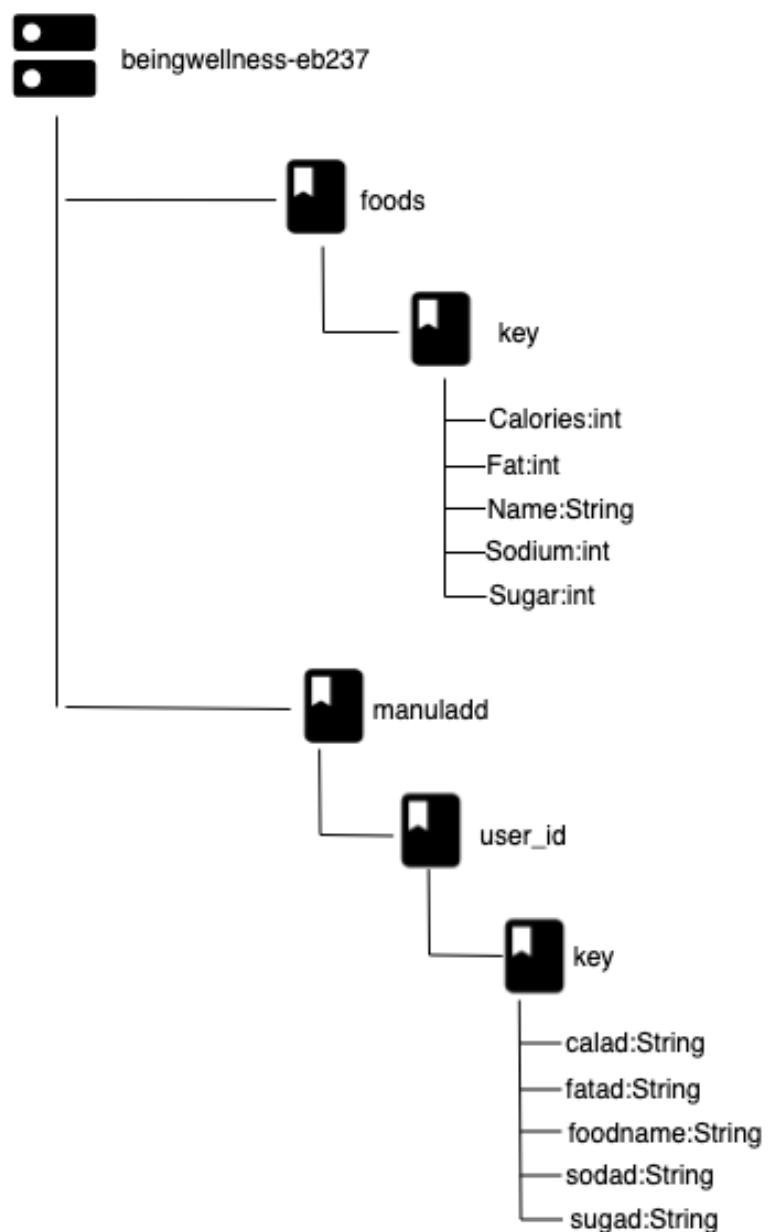
จากภาพที่ 3.16 สามารถอธิบายแผนภาพ Sequence Diagram ลบอาหารที่เพิ่ม ได้ดังนี้ เมื่อผู้ใช้เปิดหน้า Myfood คลาส MainActivity จะทำการสร้าง Fragment addfood จากนั้นจะทำการอ่านข้อมูลที่ Firebase มาแสดงรายการอาหารที่สมาชิก และเมื่อผู้ใช้กดที่รายการอาหาร Fragment addfood จะเรียกใช้ Dialogupdate และลบข้อมูลออก หลังจากที่ลบข้อมูลแล้วก็จะทำการปิด Dialogupdate จากนั้น Firebase จะส่งข้อมูลรายการอาหารที่มีอยู่มา แสดงที่สมาชิก อีกครั้ง

### 3.7 โครงสร้างฐานข้อมูลไฟร์เบส(Firebase Database Structure)

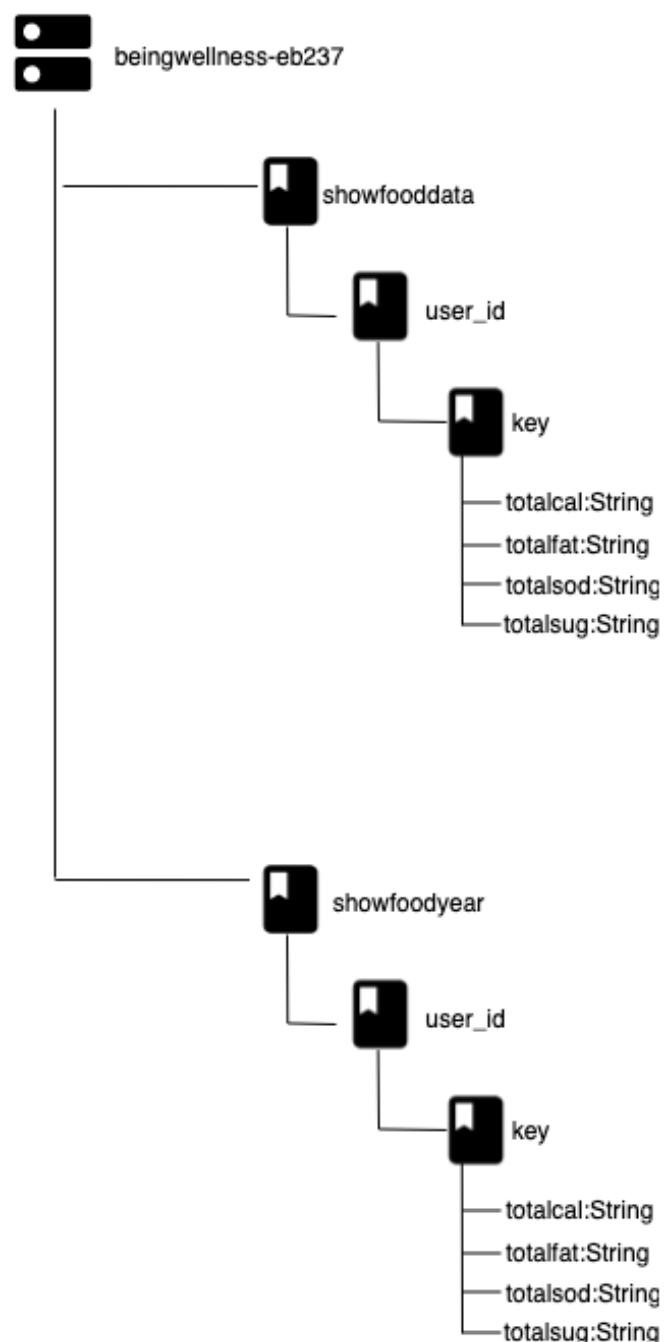
Firebase Database นั้นเป็น Database แบบ NoSQL และเป็น JSON database ที่มีโครงสร้างที่เป็น Key และ Value จัดเก็บข้อมูลในลักษณะโหนด หากต้องการเรียกงานจะเรียกใช้โดยการท่องไปยังโหนดที่ต้องการ ส่วนประกอบสัญลักษณ์ที่ใช้ในการเขียนโครงสร้างฐานข้อมูลแบบ Firebase แสดงดังตารางที่ 3.14

ตารางที่ 3.14: สัญลักษณ์ของโครงสร้างฐานข้อมูลแบบ Firebase

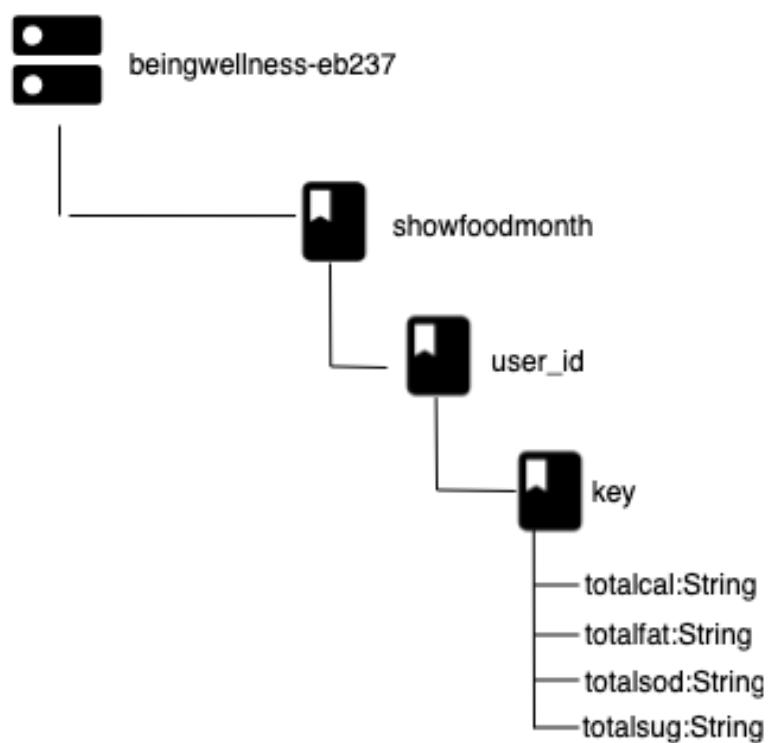
สัญลักษณ์	คำอธิบาย
	Database เป็นการเรียกชื่อแทนรากโหนด(Root Node)บนสุดที่ใช้ในการเก็บข้อมูล
	Key เป็นโหนด(Node) ที่รองลงมาจากรากโหนด ซึ่งประกอบไปด้วย KeyและValue



รูปที่ 3.17: โครงสร้างฐานข้อมูลแบบ Firebase

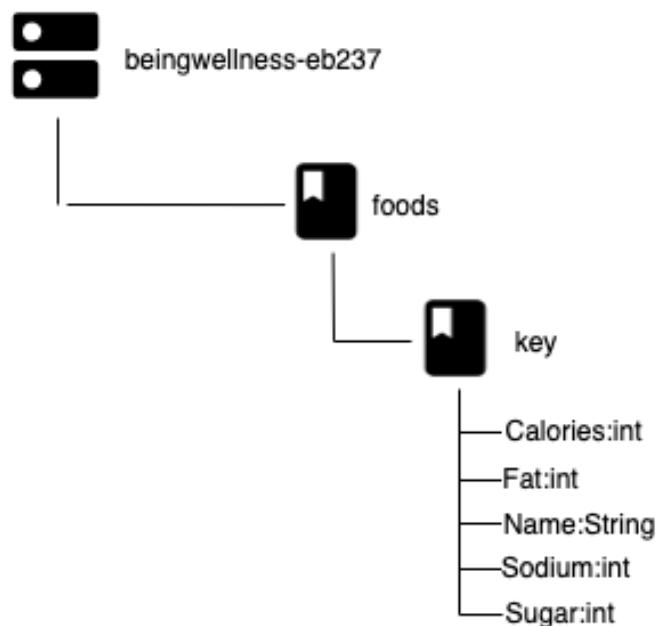


รูปที่ 3.18: โครงสร้างฐานข้อมูลแบบ Firebase(ต่อ)



รูปที่ 3.19: โครงสร้างฐานข้อมูลแบบ Firebase(ต่อ)

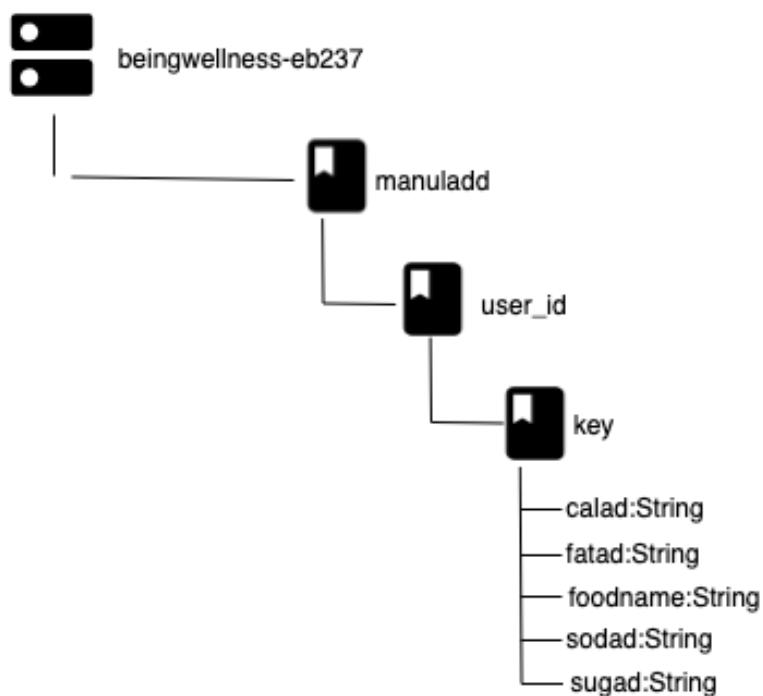
จากรูปที่ 3.17-3.19 สามารถอธิบายโครงสร้างของข้อมูลได้ดังนี้



รูปที่ 3.20: โหนดเก็บข้อมูลประภาก

ตารางที่ 3.15: อธิบายโหนดที่ใช้เก็บข้อมูลอาหาร

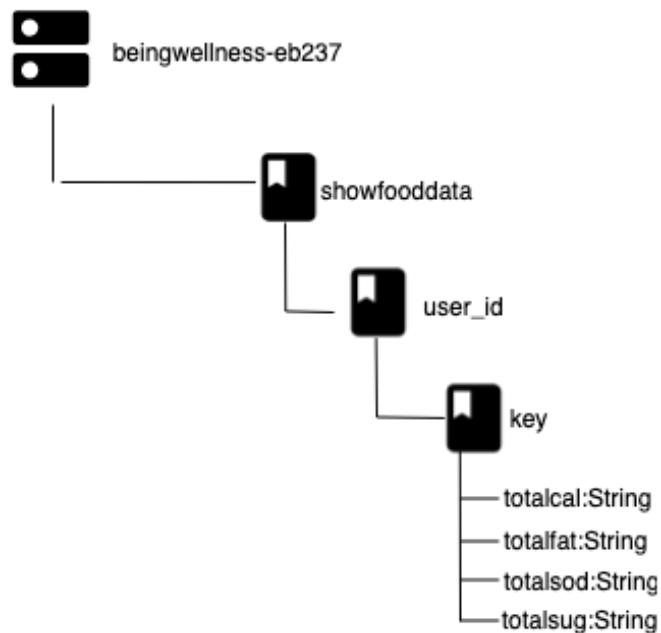
Key	คำอธิบาย
foods	โหนดสำหรับเก็บข้อมูลอาหารในระบบทั้งหมด
Key	โหนดสำหรับเก็บคีย์ของอาหาร
Calories	สำหรับเก็บแคลลอรีของอาหาร
Fat	สำหรับเก็บค่าไขมันของอาหาร
Name	สำหรับเก็บชื่อของอาหาร
Sodium	สำหรับเก็บค่าโซเดียมของอาหาร
Sugar	สำหรับเก็บค่าน้ำตาลของอาหาร



รูปที่ 3.21: โหนดเก็บข้อมูลอาหารที่สมาชิกเพิ่มเข้ามา

ตารางที่ 3.16: อธิบายโหนดเก็บข้อมูลอาหารที่สมาชิกเพิ่มเข้ามา

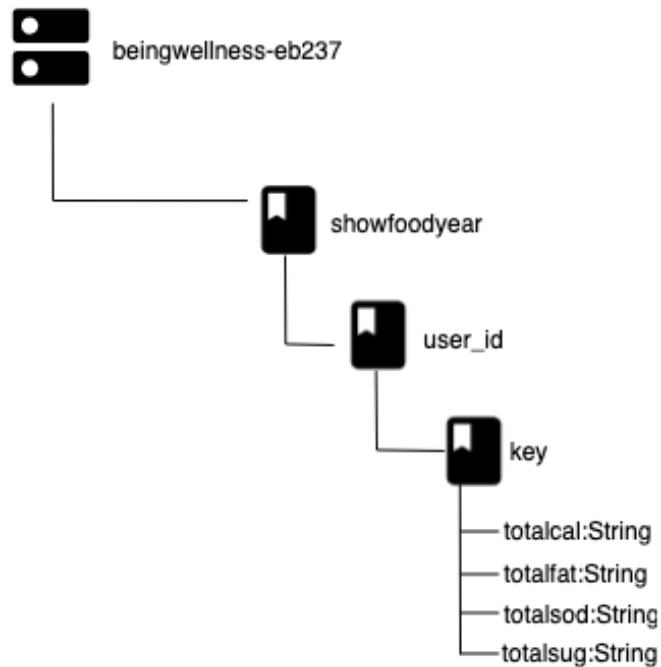
Key	คำอธิบาย
manuladd	โหนดสำหรับเก็บข้อมูลอาหารที่สมาชิกเพิ่มเข้ามาทั้งหมด
user_id	โหนดสำหรับเก็บข้อมูลอาหารที่เพิ่มเข้าของสมาชิกแต่ละคน
key	โหนดสำหรับเก็บ ID ของอาหารที่สมาชิกเพิ่มเข้ามา
calad	สำหรับเก็บแคลลอรี่ของอาหารที่ผู้ใช้เพิ่มเข้ามา
fataad	สำหรับเก็บแคลลอรี่ของอาหารที่ผู้ใช้เพิ่มเข้ามา
foodname	สำหรับเก็บแคลลอรี่ของอาหารที่ผู้ใช้เพิ่มเข้ามา
sodad	สำหรับเก็บแคลลอรี่ของอาหารที่ผู้ใช้เพิ่มเข้ามา
sugad	สำหรับเก็บแคลลอรี่ของอาหารที่ผู้ใช้เพิ่มเข้ามา



รูปที่ 3.22: โหนดเก็บข้อมูลการบริโภคในรูปแบบวัน

ตารางที่ 3.17: อธิบายโหนดเก็บข้อมูลการบริโภคในรูปแบบวัน

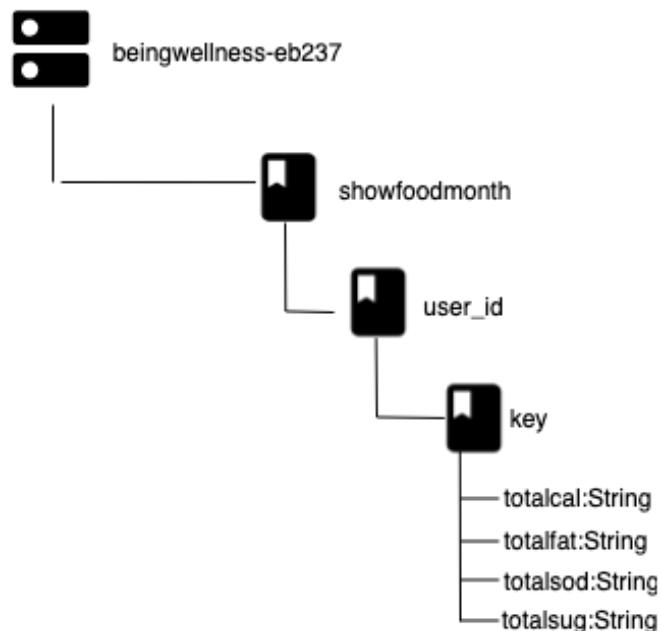
Key	คำอธิบาย
showfooddata	โหนดสำหรับเก็บข้อมูลการบริโภคในรูปแบบวันทั้งหมด
User_id	สำหรับเก็บข้อมูลการบริโภคในรูปแบบวันของสมาชิกแต่ละคน
Key	สำหรับเก็บข้อมูลวันที่
totalcal	สำหรับเก็บข้อมูลแคลอรี่ในรูปแบบวัน
totalfat	สำหรับเก็บข้อมูลไขมันรีในรูปแบบวัน
totalsod	สำหรับเก็บข้อมูลโซเดียมในรูปแบบวัน
totalsug	สำหรับเก็บข้อมูลน้ำตาลในรูปแบบวัน



รูปที่ 3.23: โหนดเก็บข้อมูลการบริโภคในรูปแบบสัปดาห์

ตารางที่ 3.18: อธิบายโหนดที่เก็บข้อมูลการบริโภคในรูปแบบสัปดาห์

Key	คำอธิบาย
showfoodyear	โหนดสำหรับเก็บข้อมูลการบริโภคในรูปแบบสัปดาห์ทั้งหมด
User_id	สำหรับเก็บข้อมูลการบริโภคในรูปแบบสัปดาห์ของสมาชิกแต่ละคน
Key	สำหรับเก็บข้อมูลสัปดาห์
totalcal	สำหรับเก็บข้อมูลแคลลอรี่ในรูปแบบสัปดาห์
totalfat	สำหรับเก็บข้อมูลไขมันรีในรูปแบบสัปดาห์
totalsod	สำหรับเก็บข้อมูลโซเดียมในรูปแบบสัปดาห์
totalsug	สำหรับเก็บข้อมูลน้ำตาลในรูปแบบสัปดาห์



รูปที่ 3.24: โหนดเก็บข้อมูลการบริโภคในรูปแบบเดือน

ตารางที่ 3.19: อธิบายโหนดที่เก็บข้อมูลการบริโภคในรูปแบบเดือน

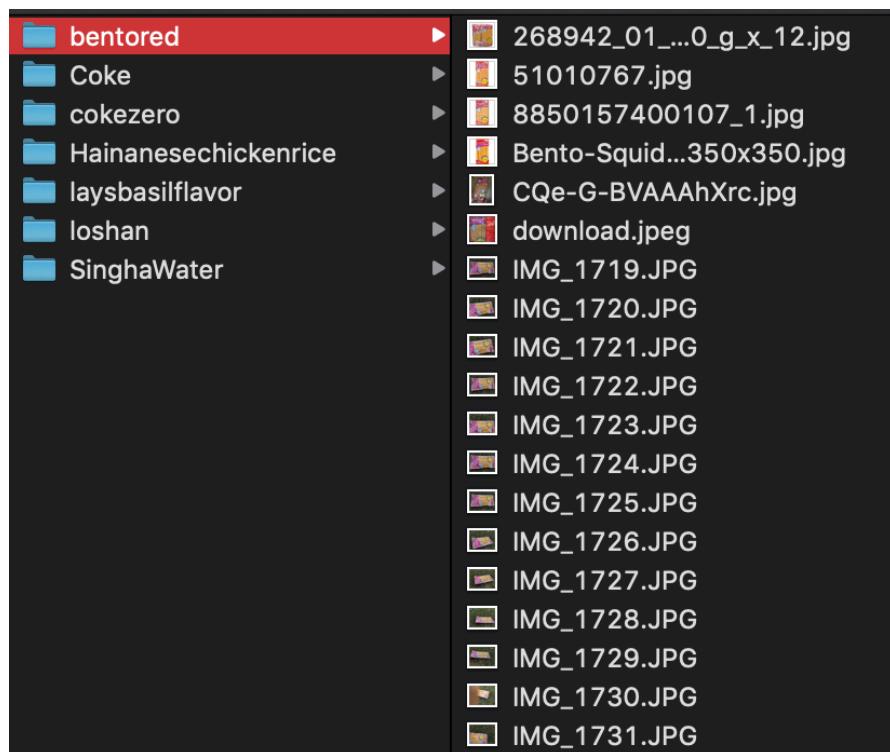
Key	คำอธิบาย
showfoodyear	โหนดสำหรับเก็บข้อมูลการบริโภคในรูปแบบเดือนทั้งหมด
User_id	สำหรับเก็บข้อมูลการบริโภคในรูปแบบเดือนของสมาชิกแต่ละคน
Key	สำหรับเก็บข้อมูลเดือน
totalcal	สำหรับเก็บข้อมูลแคลลอรี่ในรูปแบบเดือน
totalfat	สำหรับเก็บข้อมูลไขมันริ่วในรูปแบบเดือน
totalsod	สำหรับเก็บข้อมูลโซเดียมในรูปแบบเดือน
totalsug	สำหรับเก็บข้อมูลน้ำตาลในรูปแบบเดือน

### 3.8 ขั้นตอนการเตรียมข้อมูลและการสร้างโมเดล

ในขั้นตอนการเตรียมข้อมูลและการสร้างโมเดลเป็นหนึ่งในขั้นตอนที่สำคัญของแอปพลิเคชัน เนื่องจากการเตรียมข้อมูลที่ดีจะส่งผลถึงความถูกต้องและแม่นยำในการ สแกนอาหาร ใน การ ฝึกฝนได้เลือกใช้เทคโนโลยี Tensorflow Lite เป็นเครื่องมือหลัก เพราะ Tensorflow Lite ได้ใช้ เทคโนโลยีโครงข่ายประสาทเทียมเพื่อเพิ่มความ ถูกต้องและความแม่นยำในการสแกนวิภาต์ที่ยัง สนับสนุนการนำไปใช้ร่วมกับระบบปฏิบัติการแอนดรอยด์ ซึ่งขั้นตอนการเตรียมข้อมูลมีดังต่อไปนี้

#### 3.8.1 การเก็บข้อมูลรูปอาหาร

การสร้างโมเดลของระบบ ได้ใช้ภาพสินค้าที่แตกต่างกันจำนวน 50 รูป ซึ่งเป็นสินค้าที่ขายดี ที่สุดใน เชเว่น-อีเลฟเว่น



รูปที่ 3.25: การเก็บข้อมูลภาพอาหาร

จากภาพที่ 3.25 สามารถอธิบายได้ว่า ตั้งแต่ 1 ถึง 50 ภาพ แต่ละชนิดจะใช้ภาพจำนวน 100 ภาพในการ ฝึกฝน

### 3.8.2 การตั้งค่าขนาดภาพและการใช้รูปแบบการจำแนก

ในขั้นตอนนี้จะทำการตั้งค่าขนาดภาพและการเลือกใช้รูปแบบการจำแนก MobileNet\_v1 โดยใช้คำสั่งเพื่อรันดังภาพที่ 3.26

```
1 IMAGE_SIZE=224
2 ARCHITECTURE="mobilenet_v1_1.0_${IMAGE_SIZE}"
```

รูปที่ 3.26: คำสั่งการตั้งค่าขนาดภาพและการใช้รูปแบบการจำแนก

จากภาพที่ 3.26 คำสั่งการตั้งค่าขนาดภาพและการใช้รูปแบบการจำแนก สามารถอธิบาย การทำงานได้ดังนี้

- บรรทัดที่ 1 เป็นการตั้งค่าขนาดของภาพเพื่อให้ภาพมีขนาดเท่ากัน
- บรรทัดที่ 2 เป็นการเลือกใช้รูปแบบการจำแนกเพื่อทำการฝึกฝน

### 3.8.3 การสร้างโมเดลด้วย Tensorflow Lite

ในขั้นตอนนี้จะทำการตั้งค่าขนาดภาพและการเลือกใช้รูปแบบการจำแนก MobileNet\_v1 โดยใช้คำสั่งเพื่อรันดังภาพที่ 3.27

```
1 sudo python -m scripts.retrain \
2 --bottleneck_dir=tf_files/bottlenecks \
3 --how_many_training_steps=40000 \
4 --model_dir=tf_files/models/ \
5 --summaries_dir=tf_files/training_summaries/"${ \
6     ARCHITECTURE}" \
7 --output_graph=tf_files/retrained_graph.pb \
8 --output_labels=tf_files/retrained_labels.txt \
9 --architecture="${ARCHITECTURE}" \
9 --image_dir=tf_files/test
```

รูปที่ 3.27: คำสั่งการตั้งค่าขนาดภาพและการใช้รูปแบบการจำแนก

จากภาพที่ 3.27 โครงสร้างขคำสั่งการตั้งค่าขนาดภาพและการใช้รูปแบบการจำแนกสามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เป็นคำสั่งรันสคริปท์ทาง Google ได้ทำไว้
- บรรทัดที่ 2 เป็นคำสั่งสำหรับการเข้ารหัสรูปภาพ
- บรรทัดที่ 3 เป็นคำสั่งที่ระบุจำนวนของการฝึกฝน
- บรรทัดที่ 4 เป็นคำสั่งระบุที่อยู่ของโมเดล
- บรรทัดที่ 5 เป็นคำสั่งระบุที่อยู่ในการจัดเก็บผลลัพธ์จากการฝึกฝน
- บรรทัดที่ 6 เป็นคำสั่งระบุที่อยู่ในการ Export ไฟล์ retrained\_graph.PB
- บรรทัดที่ 7 เป็นคำสั่งระบุที่อยู่ในการ Export ไฟล์ retrained\_labels.text
- บรรทัดที่ 8 เป็นคำสั่งใช้งานรูปแบบการจำแนก
- บรรทัดที่ 9 เป็นคำสั่งระบุที่อยู่ของรูปภาพ

#### 3.8.4 การแปลงไฟล์ PB ไปเป็น lite

ในขั้นตอนนี้จะแปลงไฟล์ PB ไปเป็น lite เพื่อนำไปใช้กับแอนดรอยด์แอปพลิเคชัน โดยใช้คำสั่งเพื่อรันดังภาพที่ 3.28

```

1 IMAGE_SIZE=224
2 tflite_convert \
3 --graph_def_file=tf_files/retrained_graph.pb \
4 --output_file=tf_files/optimized_graph.lite \

```

รูปที่ 3.28: คำสั่งในการแปลงไฟล์ PB ไปเป็น lite

จากภาพที่ 3.28 คำสั่งการแปลงไฟล์ PB ไปเป็น lite สามารถอธิบายการทำงานได้ดังนี้

- บรรทัดที่ 1 เป็นการเรียกใช้สคริปสำหรับการแปลงไฟล์
- บรรทัดที่ 2 เป็นคำสั่งระบุที่อยู่ของไฟล์ PB
- บรรทัดที่ 3 เป็นคำสั่งระบุที่จัดเก็บไฟล์ lite

## บทที่ 4

### การพัฒนาระบบ

หลังจากที่ได้มีการเตรียมความพร้อมสำหรับการพัฒนาในด้านต่าง ไม่ว่าจะเป็นที่มาและความสำคัญของปัญหา เทคโนโลยีที่มีความเหมาะสมกับระบบ และการออกแบบระบบการทำงานรวมไปถึงโครงสร้างของข้อมูล ในบทนี้จะเป็นการพูดถึงการสร้างระบบที่ได้มีการออกแบบไว้ในบทที่แล้วจะถูกนำเสนอในบทนี้ ซึ่งจะอธิบายถึงตัวอย่างการทำงานของแอปพลิเคชันดังนี้

#### 4.1 โครงสร้างของการสร้างหน้า MainActivity

```
1 Scan scan;
2 Addfood addfood;
3 Dashboard dash;
4 Pagerdash pagerdash;
5 BottomNavigationView nv;
```

รูปที่ 4.1: ตัวแปรในคลาส MainActivity

จากภาพที่ 4.1 ตัวแปรที่ประกาศขึ้นเพื่อใช้ในการทำงานของคลาส MainActivity สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 ตัวแปร scan ใช้แสดงผลหน้าสแกน
- บรรทัดที่ 2 ตัวแปร addfood ใช้แสดงผลหน้าจอเพิ่มอาหารโดยสมาชิก
- บรรทัดที่ 3 ตัวแปร dash ใช้แสดงผลหน้าจอแดชบอร์ด
- บรรทัดที่ 4 ตัวแปร pagerdash เป็นตัวแปลงหน้าจอเมื่อผู้ใช้เลือก Tab Week หรือ Tab Month
- บรรทัดที่ 5 ตัวแปร nv เป็นตัวเก็บเมนู Scan และ Myfood มีไว้ให้สมาชิกเลือก

```

1     @Override
2     public boolean onNavigationItemSelected(@NonNull
3         MenuItem menuItem) {
4         switch (menuItem.getItemId()) {
5             case R.id.cam:
6                 if (staurselect != "scan") {
7                     ft = getSupportFragmentManager().
8                         beginTransaction().replace(R.id.
9                             Framefag, scan);
10                ft.commit();
11                staurselect = "scan";
12            }
13            break;
14        case R.id.dash:
15            if (staurselect != "dashbord" ) {
16                Calories = Sugar = Sodium = Fat =
17                    submit = null;
18                ft = getSupportFragmentManager().
19                    beginTransaction().replace(R.id.
20                        Framefag, new Pagerdash().
21                            newInstance(Calories, Sugar, Sodium,
22                            Fat, submit));
23                ft.commit();
24                staurselect = "dashbord";
25            }
26            break;
27        case R.id.cart:
28            if (staurselect != "Addfood") {
29                ft = getSupportFragmentManager().
30                    beginTransaction().replace(R.id.
31                        Framefag, addfood );
32                ft.commit();
33                staurselect = "Addfood";
34            }
35            break;
36        }
37        return true;
38    }

```

รูปที่ 4.2: โค้ดส่วนที่ใช้ในการสร้างเมนูนำทางหลักภายในคลาส MainActivity

จากภาพที่ 4.2 สามารถอธิบายการทำงานโดยส่วนที่ใช้เมนูนำทางหลักภายในคลาส MainActivity ได้ดังนี้

- บรรทัดที่ 1 เป็นการเพิ่มการตั้งจับอีเวนต์ (Event) เพื่อสลับหน้าจอการแสดงผลที่เกิดขึ้น เมื่อผู้ใช้กดที่เมนูนำทาง
- บรรทัดที่ 2-3 เป็นการเรียกใช้เมธอดชั้นจากคลาสที่ extends มา
- บรรทัดที่ 4 เป็นการรับ ID
- บรรทัดที่ 5 เป็นการระบุ ID ไปยังไอคอน cam ที่ไฟล์ xml
- บรรทัดที่ 6-9 เป็นการแสดงหน้าจอสแกนและป้องกันการกดหน้าสแกนซ้ำ
- บรรทัดที่ 10 เป็นการหยุดการทำงานของ case เพื่อไม่ให้ทำงานต่อเนื่อง
- บรรทัดที่ 11 เป็นการระบุไปยังไอคอน dash ที่ไฟล์ xml
- บรรทัดที่ 12-17 เป็นการแสดงหน้าจอแดชบอร์ดและป้องกันการกดหน้าสแกนซ้ำ
- บรรทัดที่ 18 เป็นการหยุดการทำงานของ case เพื่อไม่ให้ทำงานต่อเนื่อง
- บรรทัดที่ 19 เป็นการระบุ ID ไปยังไอคอน cart ที่ไฟล์ xml
- บรรทัดที่ 20-24 เป็นการแสดงหน้าจอเพิ่มอาหารและป้องกันการกดหน้าสแกนซ้ำ
- บรรทัดที่ 25-29 เป็นการหยุดการทำงานของ case เพื่อไม่ให้ทำงานต่อเนื่อง และจบการทำงานของเมนูนำทาง

## 4.2 โครงสร้างของการสร้างหน้า Main2Activity

```

1 Button btnSignOut;
2 FirebaseAuth auth;
3 FirebaseAuth user;
4 ProgressDialog PD;
```

รูปที่ 4.3: ตัวแปรในคลาส Main2Activity

จากภาพที่ 4.3 ตัวแปรที่ประกาศขึ้นเพื่อใช้ในการทำงานของคลาส FeedFragment สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 ตัวแปร btnSignOut ใช้ในการแสดงข้อมูลลิสต์รายการข่าวสาร
- บรรทัดที่ 2 ตัวแปร auth ใช้จัดเก็บและยืนตัวตนของสมาชิก
- บรรทัดที่ 3 ตัวแปร user ใช้จัดการข้อมูลของสมาชิก
- บรรทัดที่ 4 ตัวแปร pd ใช้สำหรับเป็นแจ้งเตือน

```

1 btnSignOut.setOnClickListener(new View.
2     OnClickListener() {
3         @Override
4         public void onClick(View view) {
5             auth.signOut();
6             Intent intent = new Intent(Main2Activity.this,
7                 LoginActivity.class);
8             startActivity(intent);
9             finish();
10            MainActivity.activityMain.finish();
11        }
12    });
13
14 findViewById(R.id.change_password_button).
15     setOnClickListener(new View.OnClickListener() {
16         @Override
17         public void onClick(View view) {
18             startActivity(new Intent(getApplicationContext(),
19                 ForgetActivity.class).putExtra("Mode",
20                 1));
21     });
22
23 findViewById(R.id.change_email_button).
24     setOnClickListener(new View.OnClickListener() {
25         @Override
26         public void onClick(View view) {
27             startActivity(new Intent(
28                 getApplicationContext(),
29                 ForgetActivity.class).putExtra("Mode",
30                 2));
31     });
32
33 findViewById(R.id.delete_user_button).
34     setOnClickListener(new View.OnClickListener() {
35         @Override
36         public void onClick(View view) {
37             startActivity(new Intent(
38                 getApplicationContext(),
39                 ForgetActivity.class).putExtra("Mode",
40                 3));
41         }
42     });
43 }

```

รูปที่ 4.4: โค้ดส่วนที่ใช้ในการสร้างเมนูนำทางหลักภายในคลาส Main2Activity

จากภาพที่ 4.4 โค้ดส่วนที่ใช้ในการสร้างเมนูนำทางหลักภายในคลาส Main2Activity สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 ทำหน้าที่ดักจับอีเวนต์ตักจักษ์เมื่อผู้ใช้ทำการกดปุ่ม SIGN OUT
- บรรทัดที่ 2-10 เป็นการทำงานเมื่อผู้ใช้กดปุ่ม SIGN OUT จะทำการออกจากระบบและนำทาง Activity ทั้งหมด
- บรรทัดที่ 12-17 เป็นการทำงานเมื่อผู้ใช้กดปุ่ม CHANGE PASSWORD ระบบจะทำการส่งค่า Mode 1
- บรรทัดที่ 19-22 เป็นการทำงานเมื่อผู้ใช้กดปุ่ม CHANGE Email ระบบจะทำการส่งค่า Mode 2
- บรรทัดที่ 23-29 เป็นการทำงานเมื่อผู้ใช้กดปุ่ม Delete User ระบบจะทำการส่งค่า Mode 3

### 4.3 โครงสร้างของการสร้างหน้า DashboardFragment

```

1 List<PieEntry> entries;
2 PieChart pieChart;
3 public static ObjectAnimator progresanimator3,
   progresanimator2, progresanimator1;
4 public static ProgressBar progressBar3, progressBar2
   , progressBar1;
```

รูปที่ 4.5: ตัวแปรในคลาส DashboardFragment

จากภาพที่ 4.5 ตัวแปรที่ประกาศขึ้นเพื่อใช้ในการทำงานของคลาส DashboardFragment สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 ตัวแปร entries ใช้สำหรับเก็บข้อมูลลงใน chart
- บรรทัดที่ 2 ตัวแปร ObjectAnimator progresanimator3, progresanimator2, progresanimator1 ใช้ตั้งค่าทำให้ chart เคลื่อนไหว
- บรรทัดที่ 3 ตัวแปร ProgressBar progressBar3, progressBar2, progressBar1 ใช้ตั้งค่า ProgressBar

```

1 Date currentTime = Calendar.getInstance().getTime();
2 SimpleDateFormat df = new SimpleDateFormat("dd-MM-
   YYYY");
3 SimpleDateFormat simplemonth = new SimpleDateFormat
   ("MMM-YYYY", Locale.ENGLISH);
4 SimpleDateFormat simpleyear = new SimpleDateFormat(
   "ww-YYYY");
5 final String month2 = simplemonth.format(currentTime)
   ;
6 final String year = simpleyear.format(currentTime);
7 final String date = df.format(currentTime);

```

รูปที่ 4.6: โค้ดส่วนที่ใช้ในการสร้างโหนดวัน สัปดาห์และเดือนภายในคลาส Dashbord

จากภาพที่ ค.3 โค้ดส่วนที่ใช้ในการสร้างเมนูนำทางหลักภายในคลาส Main2Activity สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 เป็นการตึงวันที่จากสมาร์ทโฟนมาใช้งาน
- บรรทัดที่ 2 เป็นการเลือกรูปแบบวันที่ให้อยู่ในรูปแบบ วัน
- บรรทัดที่ 3 เป็นการเลือกรูปแบบวันที่ให้อยู่ในรูปแบบ เดือน
- บรรทัดที่ 4 เป็นการเลือกรูปแบบวันที่ให้อยู่ในรูปแบบ สัปดาห์
- บรรทัดที่ 5-7 เป็นการแปลงวันที่ให้อยู่ในรูปแบบ String

```

1 if (status == null) {
2     addcal.child("showfooddata").child(user_id).child(
3         date).addListenerForSingleValueEvent (new
4             ValueEventListener() {
5                 @Override
6                     public void onDataChange(@NonNull
7                         DataSnapshot dataSnapshot) {
8                             if (dataSnapshot.getValue() != null) {
9                                 int gscal = Integer.parseInt(
10                                     dataSnapshot.child("totalcal") .
11                                         getValue().toString());
12                                 if (gsto < 0) {
13                                     gsto = 0;
14                                 }
15                                 pscal.setText(String.valueOf(gscal));
16                                 chart(gsto, gscal);
17                                 initday2(gsfat, gssod, gssug);
18                             } else {
19                                 DatabaseReference zerostatus = addcal.
20                                     child("showfooddata").child(user_id)
21                                     .child(date);
22                                 zerostatus.child("totalcal").setValue
23                                     ("0");
24                                 pscal.setText(pcal);
25                                 int gscal = Integer.parseInt(pcal);
26                             }
27                         }
28                     );
29             });
30 }
```

รูปที่ 4.7: โค้ดส่วนที่ใช้การตั้งค่า Database

จากราพที่ 4.7 ໂຄດສ່ວນທີ່ໃຊ້ການຕັ້ງຄ່າ Database ສາມາຮອບຮືບຍໍໄດ້ດັ່ງນີ້

- ບຣທັດທີ 1 ເປັນການຕຽບສອບວ່າຄ້າ Database ມີຄ່າວ່າງ
- ບຣທັດທີ 2 ເປັນການບັນທຶກຂໍ້ມູນລົງໃນ Database
- ບຣທັດທີ 3 ເປັນການເຮັດໃຫ້ເມືອດໜັນຈາກຄລາສທີ່ extends ມາ
- ບຣທັດທີ 4-11 ເປັນການອ່ານຂໍ້ມູນຈາກ Database ກາຍໃຫ້ເຈື່ອນໄຂວ່າ ຄ້າDatabase ໄມມີຄ່າວ່າງ  
ໃຫ້ກຳກັນການອ່ານຂໍ້ມູນແລະແສດງຂໍ້ມູນ
- ບຣທັດທີ 12-13 ໃຫ້ສໍາຮັບເຮັດໃຫ້ເມືອດສ້າງ Chart ແລະ Progress Bar
- ບຣທັດທີ 14-24 ເປັນເຈື່ອນໄຂທີ່ Database ມີຄ່າວ່າງໃຫ້ກຳກັນການຕັ້ງຄ່າ Database ເປັນ 0 ແລ້ວໃຫ້  
ທຳການແສດງຂໍ້ມູນ

```

1 if (status == null) {
2     addcal.child("showfooddata").child(user_id).child(
3         date).addListenerForSingleValueEvent (new
4             ValueEventListener() {
5                 @Override
6                     public void onDataChange(@NonNull
7                         DataSnapshot dataSnapshot) {
8                             if (dataSnapshot.getValue() != null) {
9                                 int gscal = Integer.parseInt(
10                                     dataSnapshot.child("totalcal") .
11                                         getValue().toString());
12                                 if (gsto < 0) {
13                                     gsto = 0;
14                                 }
15                                 pscal.setText(String.valueOf(gscal));
16                                 chart(gsto, gscal);
17                                 initday2(gsfat, gssod, gssug);
18                             } else {
19                                 DatabaseReference zerostatus = addcal.
20                                     child("showfooddata").child(user_id)
21                                     .child(date);
22                                 zerostatus.child("totalcal").setValue
23                                     ("0");
24                                 pscal.setText(pcal);
25                                 int gscal = Integer.parseInt(pcal);
26                             }
27                         }
28                     );
29             });
30 }
```

รูปที่ 4.8: โค้ดส่วนที่ใช้ในการแสดงผลหน้าจอ Dashboard

จากภาพที่ 4.8 โค้ดส่วนที่ใช้ในการสร้างเมนูนำทางหลักภายในคลาส Main2Activity สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 เป็นการตรวจสอบว่าถ้า Database มีค่าว่าง
- บรรทัดที่ 2 เป็นการบันทึกข้อมูลลงใน Database
- บรรทัดที่ 3 เป็นการเรียกใช้เมธอดชั้นจากคลาสที่ extends มา
- บรรทัดที่ 4-11 เป็นการอ่านข้อมูลจาก Database ภายใต้เงื่อนไขว่า ถ้าDatabaseไม่มีค่าว่างให้ทำการอ่านข้อมูลและแสดงข้อมูล
- บรรทัดที่ 12-13 ใช้สำหรับเรียกใช้เมธอดสร้าง Chart และ ProgressBar
- บรรทัดที่ 14-24 เป็นเงื่อนไขที่ Database มีค่าว่างให้ทำการตั้งค่า Database เป็น 0 แล้วให้ทำการแสดงข้อมูล

#### 4.4 โครงสร้างของการสร้างหน้า ScanFragment

```

1 public CameraKitView cameraKitView;
2 public static final String MODEL_PATH = "
    optimized_graph-lite";
3 public static final String LABEL_PATH = "
    retrained_labels.txt";
4 private Classifier classifier;
```

รูปที่ 4.9: ตัวแปรในคลาส ScanFragment

จากภาพที่ 4.9 ตัวแปรที่ประกาศขึ้นเพื่อใช้ในการทำงานของคลาส ScanFragment สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 ตัวแปร cameraKitView; ใช้สำหรับสร้างกล้องเพื่อทำการถ่ายภาพ
- บรรทัดที่ 2 ตัวแปร String MODEL\_PATH = optimized\_graph-lite; ใช้สำหรับระบุที่อยู่และชื่อของไฟล์ lite
- บรรทัดที่ 3 ตัวแปร String LABEL\_PATH = retrained\_labels.txt; ใช้สำหรับระบุที่อยู่และชื่อของไฟล์ txt
- บรรทัดที่ 4 ตัวแปร classifier ใช้สำหรับเรียกใช้คลาส classifier

```

1 cameraKitView.captureImage(new CameraKitView.
2     ImageCallback() {
3         @Override
4         public void onImage(CameraKitView
5             cameraKitView, final byte[] photo) {
6             Bitmap bitmap = BitmapFactory.decodeByteArray(
7                 photo, 0, photo.length);
8             bitmap = Bitmap.createScaledBitmap(bitmap,
9                 INPUT_SIZE, INPUT_SIZE, false);
10            final List<Classifier.Recognition> results =
11                classifier.recognizeImage(bitmap);
12            tvLabel.setText(results.get(0).getTitle());
13            final String ss = tvLabel.getText().toString().
14                trim();
15            DatabaseReference dbfood = FirebaseDatabase.
16                getInstance().getReference();
17            dbfood.child("foods").orderByKey().equalTo(ss).
18                addListenerForSingleValueEvent(new
19                    ValueEventListener() {
11    @Override
12        public void onDataChange(@NonNull DataSnapshot
13            dataSaps
14            Intent intent = new Intent(getActivity(), Food
15                .class);
16            for (DataSnapshot ds : dataSnapshot.
17                getChildren()) {
18                String ds1 = ds.getKey();
19                intent.putExtra("Name", dataSnapshot.child(ds1)
20                    .child("Name").getValue().toString());
21            }
22            getContext().startActivity(intent)
23        }

```

รูปที่ 4.10: โค้ดส่วนที่ใช้ในการสแกน

จากภาพที่ 4.10 โค้ดส่วนที่ใช้ในการสแกนสามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 เป็นการใช้ฟังก์ชันถ่ายรูป
- บรรทัดที่ 2 เป็นการเรียกใช้เมธอดชั้นจากคลาสที่ extends มา
- บรรทัดที่ 3-6 เป็นการทำRecognition เพื่อนำภาพที่ถ่ายไปจำแนก
- บรรทัดที่ 7 เป็นการนำค่าที่ได้จากการจำแนกมาแสดง
- บรรทัดที่ 9-14 เป็นการอ่านข้อมูลไปยัง Database โดยการนำค่าที่ได้จากการจำแนกไปเทียบกับค่า Key ใน Database
- บรรทัดที่ 15-19 เมื่อค่าจากการจำแนกตรงกับค่า Key ให้ทำการส่งข้อมูลไปยัง คลาส Food และทำการเปิดคลาส Food

#### 4.5 โครงสร้างของการสร้างหน้า AddfoodFragment

```

1 ArrayList<Foodmanul> foodlist;
2 FirebaseDatabase addfood = FirebaseDatabase.
    getInstance();
```

รูปที่ 4.11: ตัวแปรในคลาส AddfoodFragment

จากภาพที่ 4.11 ตัวแปรที่ประกาศขึ้นเพื่อใช้ในการทำงานของคลาส AddfoodFragment สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 ตัวแปร cameraKitView; ใช้สำหรับสร้างกล้องเพื่อทำการถ่ายภาพ
- บรรทัดที่ 2 ตัวแปร foodlist; จัดเก็บข้อมูลอาหาร

```
1 addcal.addValueEventListener(new ValueEventListener
2     () {
3     @Override
4     public void onDataChange(@NonNull DataSnapshot
5         dataSnapshot) {
6         foodlist.clear();
7         int i = 0;
8         for (DataSnapshot foodSnapshot :
9             dataSnapshot.getChildren()) {
10            foodlist.add(foodSnapshot.
11                getValue(Foodmanul.class)
12            );
13            foodlist.get(i).setKey(
14                foodSnapshot.getKey());
15            i++;
16        }
17        listViewfood.setLayoutManager(new
18            LinearLayoutManager(getContext()));
19        listViewfood.setHasFixedSize(true);
20        FoodList adapter = new FoodList(getContext(),
21            foodlist, putex);
22        listViewfood.setAdapter(adapter);
23    }
24 }
```

รูปที่ 4.12: โค้ดส่วนที่ใช้แสดงผลหน้าจอเพิ่มอาหาร

จากภาพที่ 4.12 โค้ดส่วนที่ใช้แสดงผลหน้าจอเพิ่มอาหาร สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 เป็นเมธอดการอ่านค่าจาก Database
- บรรทัดที่ 2 เป็นการเรียกใช้เมธอดชั้นจากคลาสที่ extends มา
- บรรทัดที่ 3-10 อ่านข้อมูลจากคลาส Foodmanul
- บรรทัดที่ 11-14 เป็นการนำข้อมูลที่ได้มาแปลงเป็นลิสต์และแสดงผล

#### 4.6 โครงสร้างของคลาส FoodList

```

1 CardView cardViewfood;
2 ArrayList<Foodmanul> foodlists;
3 String user_id;
```

รูปที่ 4.13: ตัวแปรในคลาส FoodList

จากภาพที่ 4.13 ตัวแปรที่ประกาศขึ้นเพื่อใช้ในการทำงานของคลาส FoodList สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 ตัวแปร cardViewfood; เป็นตัวแสดง CardView
- บรรทัดที่ 2 ตัวแปร foodlist; ใช้จัดเก็บข้อมูลอาหารจากคลาส Foodmanul
- บรรทัดที่ 3 ตัวแปร user\_id; ใช้เก็บค่า ID ของสมาชิก

```
1 Buuttonupdaet.setOnClickListener(new View.  
2     OnClickListener() {  
3         @Override  
4         public void onClick(View v) {  
5             String name = editTextName.getText()  
6                 .toString().trim();  
7             String cal = editTextCal.getText()  
8                 .toString().trim();  
9             if(TextUtils.isEmpty(name)) {  
10                 editTextName.setError("enter name pls");  
11                 return;  
12             }  
13             updatefood(name,cal,Sug,Sod,Fat,key);  
14             alertDialog.dismiss();  
15         }  
16     } );  
17     Buttondelete.setOnClickListener(new View.  
18         OnClickListener() {  
19             @Override  
20             public void onClick(View v) {  
21                 deletefood(key);  
22                 alertDialog.dismiss();  
23             }  
24         } );
```

รูปที่ 4.14: โค้ดการทำงานของคลาส Foodlist

จากภาพที่ 4.14 โค้ดการทำงานของคลาส Foodlist สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1-5 เป็นการตั้งจับอีเวนต์เมื่อผู้ใช้ทำการกดปุ่ม update
- บรรทัดที่ 6-9 เป็นการตั้งจับค่า่ว่างในกรณีที่สมาชิกไม่ทำการใส่ข้อความแล้วให้แสดงแจ้งเตือนกลับมา
- บรรทัดที่ 10-13 เรียกใช้เมธอด updatefood และทำการปิด Dialog
- บรรทัดที่ 14-17 เป็นการตั้งจับอีเวนต์เมื่อผู้ใช้ทำการกดปุ่ม delete โดยเลือกลบที่ Key
- บรรทัดที่ 18-20 ทำการปิด Dialog

#### 4.7 โครงสร้างของคลาส LoginActivity

```
1 private FirebaseAuth auth;
2 private ProgressDialog PD;
```

รูปที่ 4.15: ตัวแปรในคลาส LoginActivity

จากภาพที่ 4.15 ตัวแปรที่ประกาศขึ้นเพื่อใช้ในการทำงานของคลาส LoginActivity สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 ตัวแปร auth; ใช้จัดเก็บและยืนยันตัวตนของสมาชิก
- บรรทัดที่ 2 ตัวแปร PD; ใช้แสดงข้อความให้กับสมาชิก

```

1 auth.signInWithEmailAndPassword(email, password)
2     .addOnCompleteListener(LoginActivity.this, new
3         OnCompleteListener<AuthResult>() {
4             @Override
5                 public void onComplete(@NonNull
6                     Task<AuthResult> task) {
7                 if (task.isSuccessful()) {
8                     Intent intent = new Intent(LoginActivity
9                         .this, MainActivity.class);
10                    startActivity(intent);
11                    finish();
12                } else {
13                    Toast.makeText(
14                        LoginActivity.this,
15                        "Authentication Failed",
16                        Toast.LENGTH_LONG).show();
17                    Log.d("error", task.getException().toString());
18                }
19            PD.dismiss();
20        }
21    );
22 });

```

รูปที่ 4.16: โค้ดส่วนที่ใช้ในการเข้าสู่ระบบในคลาส LoginActivity

จากภาพที่ 4.16 โค้ดส่วนที่ใช้ในการเข้าสู่ระบบในคลาส LoginActivity สามารถอธิบายได้ดังนี้

- บรรทัดที่ 1-3 เป็นการตรวจสอบ Email และ Password ของผู้ใช้
- บรรทัดที่ 4-8 เป็นการทำงานมั�กรณีที่เข้าสู่ระบบสำเร็จจะทำการเรียกใช้คลาส MainActivity
- บรรทัดที่ 9-18 เป็นการทำงานเมื่อเข้าสู่ระบบไม่สำเร็จจะทำการแสดงข้อความว่า Authentication Failed

## บทที่ 5

### การทดสอบระบบ

การทดสอบการทำงานของแอปพลิเคชันระบบกองทุนเงินให้กู้ยืมเพื่อการศึกษา คณะวิทยาศาสตร์ มหาวิทยาลัยอุบลราชธานี และทดสอบการทำงานในส่วนของเว็บไซต์ โดยทำการทดสอบในลักษณะ Black-box Testing [12] หรือ Data-Driven testing ซึ่งเป็นการทดสอบแบบที่ไม่สนใจโปรเซส (Process) การทำงานภายในของโปรแกรมว่าทำงานอย่างไร แต่จะเน้นไปที่ Input และ Result ที่ได้มากกว่าการทำงานต่าง ๆ ถูกต้องตามความต้องการ (Requirement) หรือไม่ ซึ่งการทดสอบการใช้งานของแอปพลิเคชัน และ ความแม่นยำของ Tensorflow Lite ได้ผลดังนี้

#### 5.1 การทดสอบการใช้งานของแอปพลิเคชัน

- การทดสอบการใช้งานเมนูนำทางของแอปพลิเคชัน การทดสอบเมนูนำทางของแอปพลิเคชันในการนำทางสมาชิก ซึ่งเมนูหลักประกอบด้วย เมนูเข้าสู่ระบบ เมนูแดชบอร์ด เมนูสแกน เมนูเพิ่มอาหาร หน้าจອกรดูข้อมูลการบริโภคแบบสัปดาห์ หน้าจອกรดูข้อมูลการบริโภคแบบเดือน เมนูตั้งค่า ผลทดสอบดังตารางที่ 5.1-5.5

ตารางที่ 5.1: ผลการทดสอบเมนูนำทาง

การทำงาน	เงื่อนไขการทดสอบ	ผลการทดสอบ
เมนูเดชบอร์ด	กดปุ่มเมนูเดชบอร์ด	ระบบแสดงผลหน้าจอเดชบอร์ดพร้อมແນບສໍາຮັບເລືອນດູ ຂໍ້ມູນ ການ ບຣິໂກຄ ຮູບ ແບບ ສັບຕາໜໍ ແລະ ເດືອນ
เมนูສแกນ	กดปุ่มเมนูສแกນ	ຮະແບບແສດງຜລໝາຈອສແກນ ພຣ້ອມກັບເປີດກລັອງ
	กดปຸ່ມຍັນກລັບ	ຮະບບທຳການປິດແອປພລິເຄັນ
เมนูເພີ່ມອາຫາຣ	กดປຸ່ມເມນູເພີ່ມອາຫາຣ	ຮະແບບ ແສດງ ຜລ ແນ້າ ຈອ ເພີ່ມ ອາຫາຣ ແລະ ມີຮາຍກາຣອາຫາຣທີ່ ເພີ່ມແລ້ວມາແສດງ
	กดປຸ່ມຍັນກລັບ	ຮະບບທຳການປິດແອປພລິເຄັນ
ໜ້າ ຈອ ກາຣ ດູ ຂໍ້ມູນ ກາຣ ບຣິໂກຄ ແບບ ສັບຕາໜໍ	ເລືອນໄປແນບຂໍ້ມູນກາຣບຣິໂກຄ ແບບ ສັບຕາໜໍ	ຮະບບແສດງຮາຍກາຣຂໍ້ມູນກາຣບຣິໂກຄຂອງແຕ່ລະສັບຕາໜໍ
	กดປຸ່ມຍັນກລັບ	ຮະບບທຳການປິດແອປພລິເຄັນ
ໜ້າ ຈອ ກາຣ ດູ ຂໍ້ມູນ ກາຣ ບຣິໂກຄ ແບບ ເດືອນ	ເລືອນໄປແນບຂໍ້ມູນກາຣບຣິໂກຄ ແບບ ເດືອນ	ຮະບບແສດງຮາຍກາຣຂໍ້ມູນກາຣບຣິໂກຄຂອງແຕ່ລະເດືອນ
	กดປຸ່ມຍັນກລັບ	ຮະບບທຳການປິດແອປພລິເຄັນ
ເມນູຕັ້ງຄ່າ	กดເມນູຕັ້ງຄ່າ	ຮະບບແສດງເມນູຕັ້ງຄ່າ
	กดປຸ່ມຍັນກລັບ	ຮະບບ ແສດງ ຜລ ແນ້າ ຈອ ແດ ທ ບອຮົດພຣ້ອມແນບສໍາຮັບເລືອນ ດູ ຂໍ້ມູນ ກາຣ ບຣິໂກຄ ຮູບ ແບບ ສັບຕາໜໍ ແລະ ເດືອນ

- การทดสอบหน้าแดชบอร์ด ในการแสดงผลหน้าจอรายละเอียดประกาศนั้นจะประกอบไปด้วย Chart สำหรับการดูรายละเอียดการบริโภคพร้อมกับแบบสำรวจสำหรับดูข้อมูลการบริโภครูปแบบสัปดาห์และเดือน ผลการทดสอบดังตารางที่ 5.2

ตารางที่ 5.2: ผลการทดสอบหน้าแดชบอร์ด

การทำงาน	เงื่อนไขการทดสอบ	ผลการทดสอบ
หน้าแดชบอร์ด	เลื่อนไปยังแบบ Week	ระบบแสดงรายการข้อมูลการบริโภคของแต่ละสัปดาห์
	เลื่อนไปยังแบบ Month	ระบบแสดงรายการข้อมูลการบริโภคของแต่ละเดือน
	เลื่อนกลับมายังแบบ Week	ระบบแสดงรายการข้อมูลการบริโภคของแต่ละสัปดาห์
	เลื่อนกลับมายังแบบ Day	ระบบแสดง Chart และรายละเอียดการบริโภคของวัน

- การทดสอบหน้าสแกนในการแสดงผลหน้าจอสันทนาณ์จะประกอบไปด้วยกล้องสำหรับทำการถ่ายภาพ ปุ่ม Scan ผลการทดสอบดังตารางที่ 5.3

ตารางที่ 5.3: การทดสอบหน้าสแกน

การทำงาน	เงื่อนไขการทดสอบ	ผลการทดสอบ
หน้าสแกน	กดปุ่มสแกน	ระบบทำการสแกนเมื่อสแกนพบรำคำการแสดงชื่อที่สแกน และ แสดง ข้อมูล อาหาร ในหน้าแสดงข้อมูลอาหาร
	กดปุ่ม SUBMIT	ระบบนำข้อมูลอาหารที่ได้ไปเพิ่มในหน้าเดชบอร์ดแล้วเปิดหน้าเดชบอร์ด
	กดปุ่มย้อนกลับ	ระบบแสดงหน้าจอสแกน

- การทดสอบหน้าเพิ่มอาหาร ในการแสดงผลหน้าเพิ่มอาหารนั้นจะประกอบไปด้วยรายการอาหาร ปุ่มสำหรับเพิ่มอาหาร ผลการทดสอบดังตารางที่ 5.4

ตารางที่ 5.4: ผลการทดสอบหน้าเพิ่มอาหาร

การทำงาน	เงื่อนไขการทดสอบ	ผลการทดสอบ
หน้าเพิ่มอาหาร	กดปุ่ม ADD FOOD	ระบบ จะ ทำการ แสดง หน้า เพิ่มอาหารขึ้นมาโดยมีพื้นที่กรอกข้อความ กับช่องกรอกข้อความ
	พิมพ์อักษร ครบ ทุก ช่อง และ กดปุ่ม Add	ระบบ ย้อน กลับ ไป หน้า เพิ่ม อาหาร แล้ว มี รายการ อาหารเพิ่มขึ้นมา
	พิมพ์ อักษร ไม่ ครบ ทุก ช่อง และ กดปุ่ม Add	ระบบ ทำการ เจ้ง เตือน ให้ ส่อ อักษร ใน ช่อง ที่ ว่าง
	กดเลือกที่รายการอาหาร	ระบบ ทำการแสดง Dialog ที่ ประกอบ ไป ด้วย ช่อง กรอก ข้อความ ปุ่ม Update และ ปุ่ม Delete
	พิมพ์ อักษร ครบ ทุก ช่อง และ กดปุ่ม Update	มี การ แจ้ง เตือน และ ข้อมูล อาหารเปลี่ยนตามที่พิมพ์
	พิมพ์ อักษร ไม่ ครบ ทุก ช่อง และ กดปุ่ม Update	มี การ แจ้ง เตือน ให้ ส่อ อักษร
	กดปุ่ม Delete	มี การ แจ้ง เตือน และ ไม่มี อาหาร ที่ ลบ แสดงอยู่ ใน รายการ อาหาร

- การทดสอบหน้าเมนูตั้งค่า ในการแสดงผลหน้าเมนูตั้งค่าเน้นจะประกอบไปด้วยปุ่ม Change E-mail ปุ่ม Change Password ปุ่ม Delete user และปุ่ม SIGN OUT ผลการทดสอบดังตารางที่ 5.5

ตารางที่ 5.5: ผลการทดสอบหน้าเมนูตั้งค่า

การทำงาน	เงื่อนไขการทดสอบ	ผลการทดสอบ
หน้าเมนูตั้งค่า	กดปุ่ม Change E-mail	ระบบ ทำการ เปิด หน้า Change email
	กดปุ่ม Change Password	ระบบ ทำการ เปิด หน้า Change Password
	กดปุ่ม Delete user	ระบบทำการเปิดหน้า Delete user
	กดปุ่ม Delete user	ระบบ ทำการ กลับ มา ที่ หน้า เข้าสู่ระบบ

## 5.2 การทดสอบความแม่นยำของ Tensorflow Lite

การทดสอบความแม่นยำในการทำนาย เป็นการทดสอบเพื่อหาความแม่นยำที่ใช้ในการทำนายภาพอาหาร โดยทำการถ่ายภาพอาหาร 5 รูป ซึ่งเป็นรูปภาพอาหารไม่ซ้ำกัน โดยผลการทดสอบมีผลลัพธ์ดังนี้

- การถ่ายภาพอาหารชื่อ Cokezero



รูปที่ 5.1: การทดสอบความแม่นยำอาหารชื่อ Cokezero

จากรูปที่ 5.1 เป็นรูปผลลัพธ์ของการทำนายอาหารชื่อ Cokezero พบว่ามีค่าความแม่นยำที่ร้อยละ 78

– การถ่ายภาพอาหารชื่อ Coke



รูปที่ 5.2: การทดสอบความแม่นยำอาหารชื่อ Coke

จากรูปที่ 5.3 เป็นรูปผลลัพธ์ของการทำนายอาหารชื่อ Coke พบร่วมค่าความแม่นยำที่ร้อยละ 78

- การถ่ายภาพอาหารชื่อ laysbasilflavor



รูปที่ 5.3: การทดสอบความแม่นยำอาหารชื่อ laysbasilflavor

จากรูปที่ 5.3 เป็นรูปผลลัพธ์ของการทำนายอาหารชื่อ laysbasilflavor พบว่ามีค่าความแม่นยำที่ร้อยละ 76

- การถ่ายภาพอาหารชิ้น bentored



รูปที่ 5.4: การทดสอบความแม่นยำอาหารชิ้น bentored

จากรูปที่ 5.4 เป็นรูปผลลัพธ์ของการทำนายอาหารชิ้น bentored พบว่ามีค่าความแม่นยำที่ร้อยละ 75

– การถ่ายภาพอาหาร hainanesechickenrice



[[1] hainanesechickenrice (78.7%)]

รูปที่ 5.5: การทดสอบความแม่นยำอาหารชื่อ hainanesechickenrice พบ  
ว่ามีค่าความแม่นยำ ร้อยละ 78

จากรูปที่ 5.5 เป็นรูปผลลัพธ์ของการทำนายอาหารชื่อ hainanesechickenrice พบ  
ว่ามีค่าความแม่นยำ ร้อยละ 78

ตารางที่ 5.6: ผลการทดสอบความถูกต้องในการทำนาย

ลำดับ	ชื่ออาหารที่ถ่าย	ผลลัพธ์การทำนาย	ค่าความถูกต้อง
รูปภาพที่ 1	Cokezero	ถูกต้อง	78%
รูปภาพที่ 2	Coke	ถูกต้อง	78%
รูปภาพที่ 3	laysbasilflavor	ถูกต้อง	76%
รูปภาพที่ 4	ben	ถูกต้อง	75%
รูปภาพที่ 5	hainanese chickenrice	ถูกต้อง	78%

จากตารางที่ 5.6 เป็นค่าความแม่นยำจากการถ่ายภาพอาหารที่แตกต่างกัน 5 ชนิด โดยมีค่าความแม่นยำมากกว่าร้อยละ 70

## บทที่ 6

### สรุปและข้อเสนอแนะ

การดำเนินโครงการเพื่อพัฒนาแอปพลิเคชันบีอิงเวลเนส นี้ พบร่วมระบบสามารถทำงานได้ตามที่วิเคราะห์และออกแบบไว้ แต่ก็พบปัญหาและอุปสรรคระหว่างการพัฒนา ในบทนี้ผู้พัฒนาจึงขอสรุปความสามารถของระบบ ซึ่งจะแสดงปัญหาและอุปสรรค พร้อมเสนอแนวทางในการพัฒนาแอปพลิเคชัน ต่อ ตามลำดับ

#### 6.1 สรุปความสามารถของระบบ

##### 1. สมาชิก

- สมัครสมาชิกได้
- เข้าสู่ระบบได้
- ดูข้อมูลการบริโภคในรูปแบบวัน สัปดาห์และเดือนได้
- เพิ่มอาหารที่ไม่มีในระบบได้
- แก้ไขอาหารและบันทึกอาหารได้
- เปลี่ยน E-mail Password และลบผู้ใช้ได้
- ออกจากระบบได้

#### 6.2 ปัญหาและอุปสรรคในการพัฒนา

1. การระบุรุคุมความแม่นยำน้อยเนื่องจากต้องมีการนำ อายุ น้ำหนัก ส่วนสูงมารวมใน การคำนวณ  
แนวทางการแก้ไข : เปลี่ยนมาเป็น Progressbar ให้ระบุการบริโภคสูงสุดในแต่ละวัน

#### 6.3 แนวทางการพัฒนาต่อ

1. ปรับแต่ง UI ให้น่าใช้งาน
2. พัฒนาฟังก์ชันระบุรุคให้มีความแม่นยำและนำมาใช้ได้จริง
3. พัฒนาฟังก์ชันการออกกำลังกาย

## บรรณานุกรม

- [1] ทศพล ตันสมบัติ. (ม.ป.ป.). ระบบปฏิบัติการ android [ออนไลน์]. สืบค้นเมื่อ 12 พฤษภาคม 2561. จาก <https://beerkung.wordpress.com/ระบบปฏิบัติการรุ่นล่าส/ระบบปฏิบัติการ-android.html> .
- [2] kissana362. (2559). ระบบปฏิบัติการ android [ออนไลน์]. สืบค้นเมื่อ 12 พฤษภาคม 2561. จาก <http://kissana362.blogspot.com/> .
- [3] ดร.วีระศักดิ์ ชึงถาวร. 2545. JAVA PROGRAMMING Volume I. กรุงเทพฯ: ชีเอ็ดยูเคชั่น.
- [4] marcuscode. (2018). แนะนำภาษา python [ออนไลน์]. สืบค้นเมื่อ 2 พฤษภาคม 2561. จาก <http://marcuscode.com/lang/python/introduction> .
- [5] Janita Jangjaibun (2018). การเขียนภาษาไพธอน (python)เบื้องต้น [ออนไลน์]. สืบค้นเมื่อ 12 พฤษภาคม 2561. จาก <https://prezi.com/p/fpriuacpdxai/python/> .
- [6] nextsoftwares (2014). ความรู้เบื้องต้นเกี่ยวกับการประมวลผลภาพดิจิทัล [ออนไลน์]. สืบค้นเมื่อ 12 พฤษภาคม 2561. จาก <https://nextsoftwares.wordpress.com/2014/05/22/> .
- [7] ร.อ.หญิง กฤติยา รัตแพทย์ (2018). Ai : Artificial intelligence หรือปัญญาประดิษฐ์ [ออนไลน์]. สืบค้นเมื่อ 12 พฤษภาคม 2561. จาก <https://www.gotoknow.org/posts/163433> .
- [8] Athiwat (2561). Deep learning [ออนไลน์]. สืบค้น เมื่อ จาก <https://medium.com/athivvat/deep-leaning-คืออะไร-785e16d01773> .
- [9] DumpDataSci (2561). Machine learning [ออนไลน์]. สืบค้นเมื่อ จาก <https://medium.com/dumpdatasci.th/machine-learning-คืออะไร-พร้อมตัวอย่างในชีวิตจริง-b3f8faf0b4da> .

- [10] Anonymous (2561). Neural network [ออนไลน์]. สืบค้นเมื่อ จาก <https://coladev.com/machine-learning/neural-network/2017/02/22/neural-network-basic>.
- [11] Saengow, J. (2018). Firebase คืออะไร มาดูวิธีสร้าง [ออนไลน์]. สืบค้นเมื่อ จาก <https://medium.com/jed-ng/firebase->.
- [12] Atthaboon S. (2555). Black-box testing strategy [ออนไลน์]. สืบค้นเมื่อ 20 พฤษภาคม 2561. จาก <http://everybitsconsult.com/blog/2015/06/22/black-box-testing.html>.

ภาคผนวก

## ภาคผนวก ก

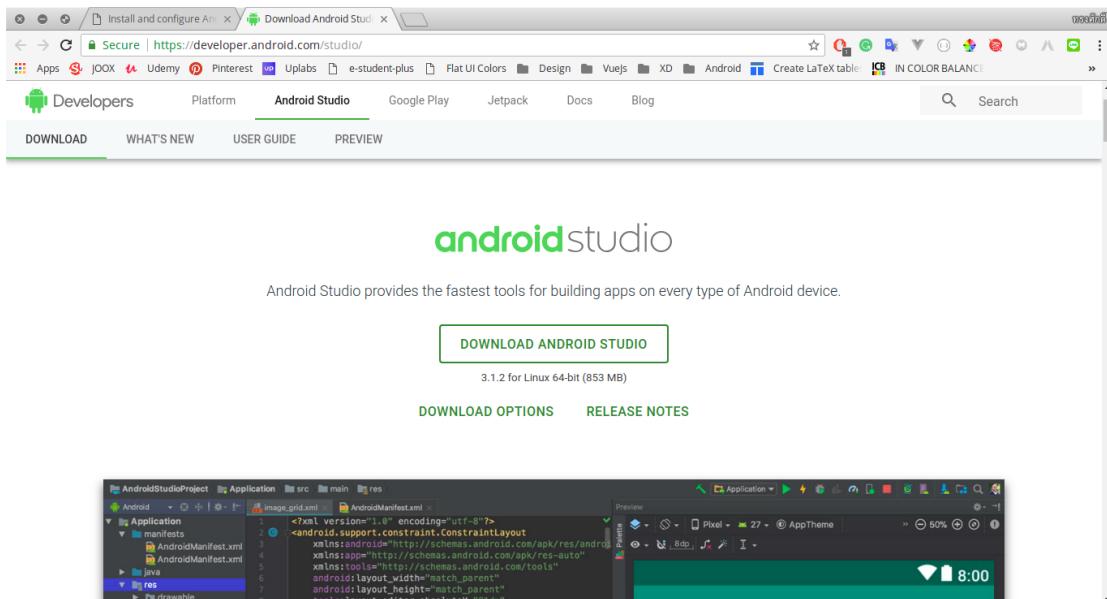
### การติดตั้งเครื่องมือที่ใช้พัฒนาโปรแกรม

การติดตั้งเครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชันระบบกองทุนเงินให้กู้ยืมเพื่อการศึกษา คณะวิทยาศาสตร์ มหาวิทยาลัยอุบลราชธานี มีโปรแกรมที่จำเป็นในการพัฒนาระบบดังต่อไปนี้

- การติดตั้ง Android Studio
- การติดตั้ง Tensorflow Lite

#### ก.1 การติดตั้ง Android Studio

1. สามารถดาวน์โหลด Android Studio installer package ได้ที่ <https://developer.android.com/studio/> ดังแสดงในรูปที่ ก.1



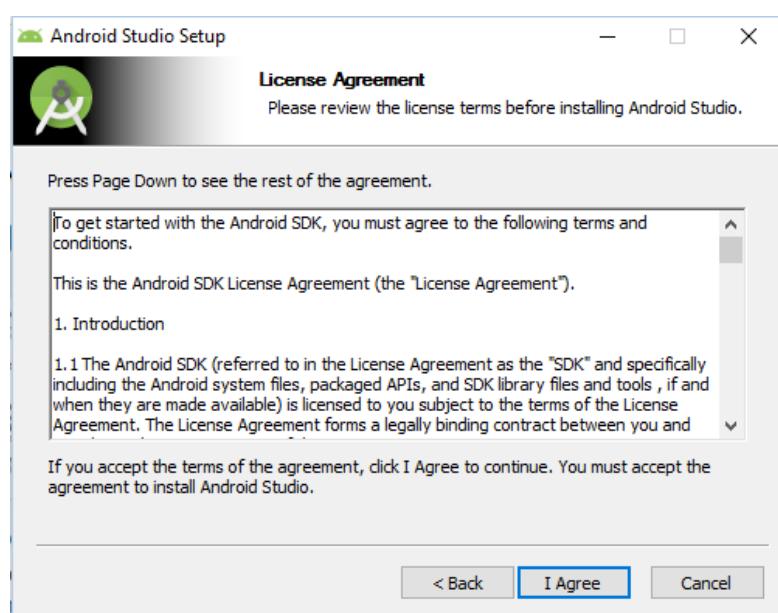
รูปที่ ก.1: หน้าเว็บดาวน์โหลด Android Studio

2. แสดงหน้าต่างต้อนรับของ Android Studio ทำการกด Next เพื่อเริ่มกระบวนการติดตั้ง ดังแสดงในรูปที่ ก.2



รูปที่ ก.2: หน้าต่างต้อนรับของ Android Studio

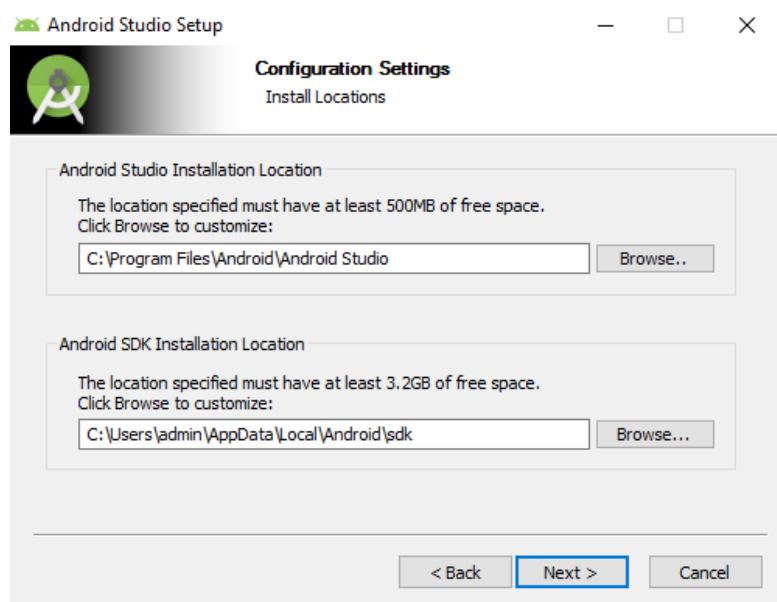
3. แสดงหน้าต่างข้อตกลงการใช้งาน Android Studio ทำการกด I Agree ดังแสดงในรูปที่ ก.3



รูปที่ ก.3: หน้าต่างข้อตกลงการใช้งาน Android Studio

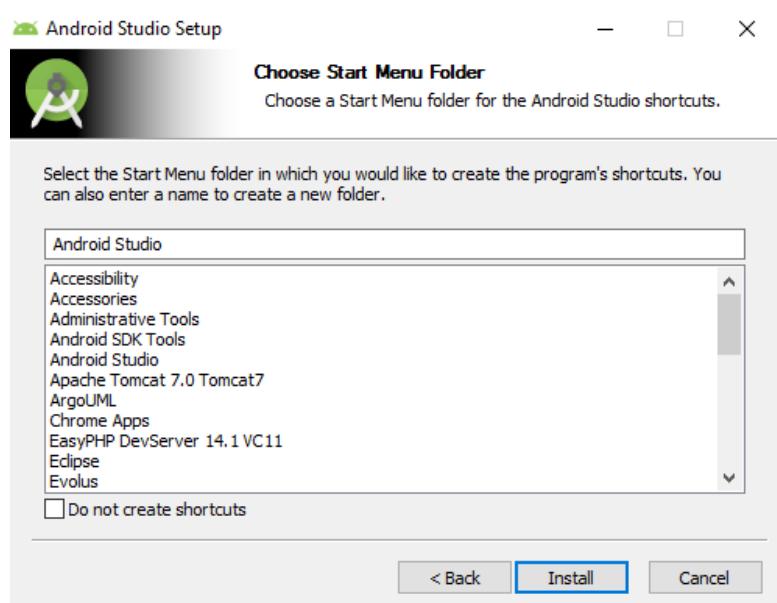
4. แสดงหน้าต่างที่จัดเก็บไฟล์ต่างๆ ของ Android Studio ทำการกด Next ดังแสดงใน

รูปที่ ก.4



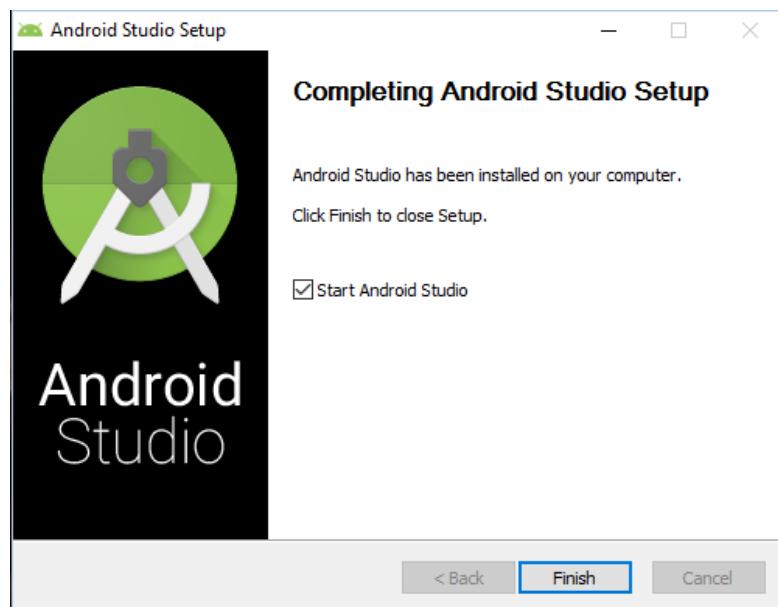
รูปที่ ก.4: หน้าต่างที่จัดเก็บไฟล์ต่างๆ ของ Android Studio

5. หน้าต่างเริ่มทำการติดตั้งทำการกด Install ดังแสดงในรูปที่ ก.5



รูปที่ ก.5: หน้าต่างที่จัดเก็บไฟล์ต่างๆ ของ Android Studio

6. หน้าต่างผลการติดตั้ง Android Studio ดังแสดงในรูปที่ ก.6



รูปที่ ก.6: หน้าต่างผลการติดตั้ง Android Studio

## ก.2 การติดตั้ง Python3.6

1. การติดตั้ง Python3.6 สามารถดาวน์โหลดได้ที่ <https://www.python.org/downloads/release/python-365/> ดังแสดงในภาพที่ ก.7

รูปที่ ก.7: หน้าเว็บดาวน์โหลด Python3.6

2. เปิดไฟล์จะได้หน้าตาดังรูปที่ ก.8 จากนั้นกด Install now รอจนเสร็จ

รูปที่ ก.8: การติดตั้ง Python3.6

### ก.3 การติดตั้ง Tensorflow

โดยในการติดตั้งผ่าน pip สามารถเลือกได้ว่าจะให้ทำงานบน CPU อย่างเดียว หรือทำงานทั้งบน CPU และ GPU โดยมีคำสั่งดังนี้

1. การติดตั้งสำหรับทำงานบน CPU ดังแสดงในภาพที่ ก.9

```
1 pip3 install --upgrade tensorflow
```

รูปที่ ก.9: การติดตั้ง Tensorflow สำหรับทำงานบน CPU

2. การติดตั้งสำหรับทำงานบน CPU และ GPU ดังแสดงในภาพที่ ก.10

```
1 pip3 install --upgrade tensorflow-gpu
```

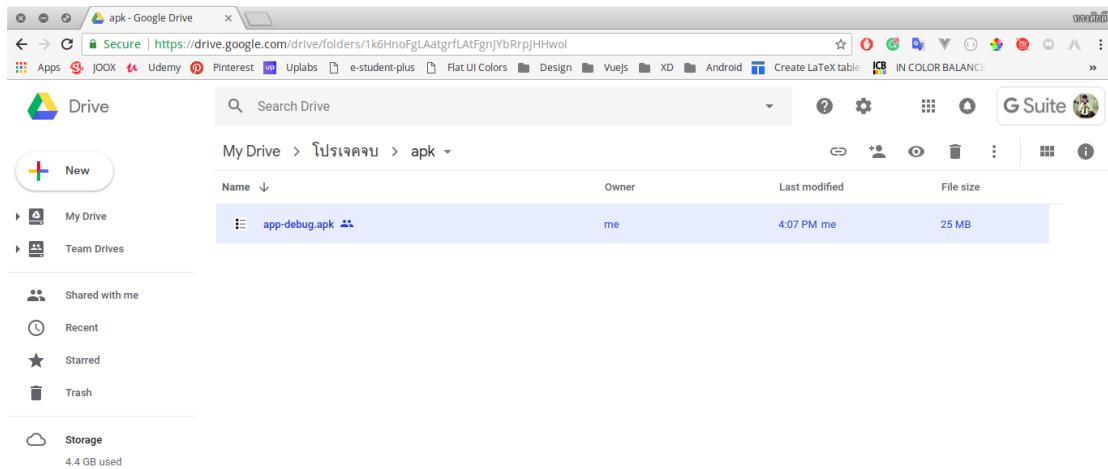
รูปที่ ก.10: การติดตั้ง Tensorflow สำหรับทำงานบน CPU และ GPU

## ภาคผนวก ข

คู่มือการติดตั้งระบบ(ทำการแก้ไขภายหลังจากการสอบเสร็จ)

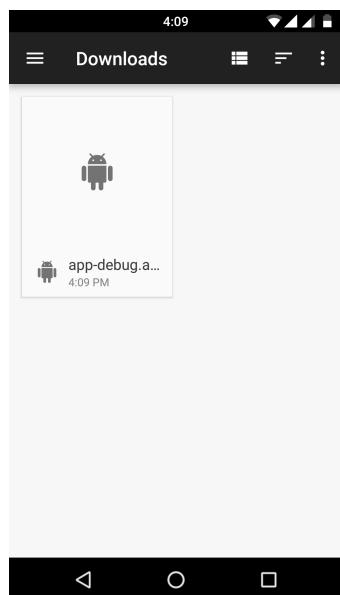
ในการติดตั้งเพื่อใช้งานแอปพลิเคชันระบบกองทุนเงินให้กู้ยืมเพื่อการศึกษา คณะวิทยาศาสตร์ มหาวิทยาลัยอุบลราชธานี หรือ ESP สามารถทำได้โดยมีขั้นตอนดังนี้

- สามารถดาวน์โหลด ESP installer package ได้ที่ <https://drive.google.com/drive/folders/1k6HnoFgLAatgrfLAtFgnJYbRrpJHHwol> ดังแสดงในรูปที่ ข.1



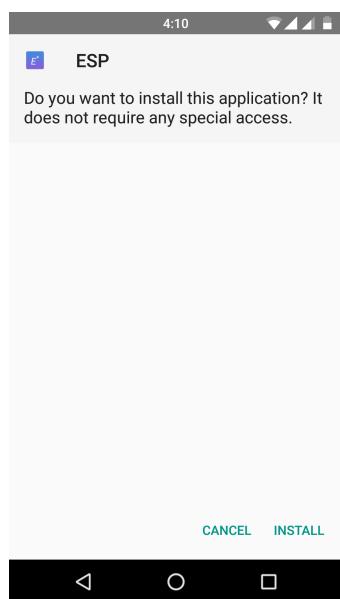
รูปที่ ข.1: หน้าเว็บดาวน์โหลด ESP installer package

- ตัดลอกไฟล์ app-debug.apk ที่อยู่ในแฟ้มงาน(Folder)ที่อยู่บนคอมพิวเตอร์ไปไว้ในหน่วยความจำบันอุปกรณ์ที่ต้องการ ดังแสดงในรูปที่ ข.2



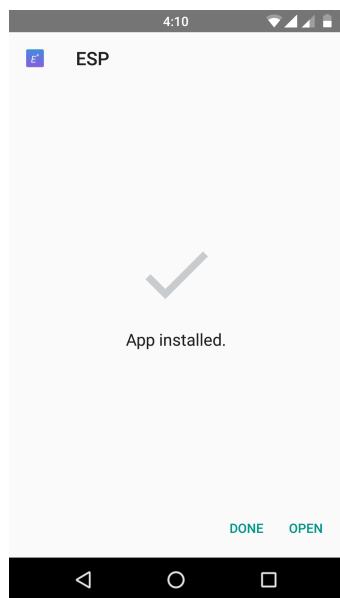
รูปที่ ข.2: ไฟล์ app-debug.apk บนอุปกรณ์

- ทำการเปิดไฟล์ app-debug.apk และกด INSTALL เพื่อทำการติดตั้ง ดังแสดงในรูปที่ ข.3



รูปที่ ข.3: หน้าจอต้อนรับการติดตั้งแอปพลิเคชันบนอุปกรณ์เอนдрอย์

- เมื่อทำการติดตั้งแอปพลิเคชันสำเร็จระบบจะแสดงผล ดังแสดงในรูปที่ ข.4



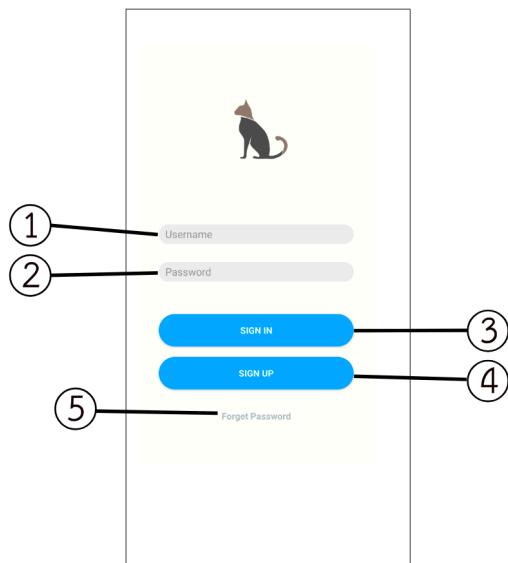
รูปที่ ข.4: หน้าจอติดตั้งแอปพลิเคชันบนอุปกรณ์แอนดรอย์

## ภาคผนวก ค

### คู่มือการใช้งานระบบ

คู่มือการใช้งานทั้งหมดของแอปพลิเคชันบอิงเวลเนส สามารถอธิบายได้ดังนี้

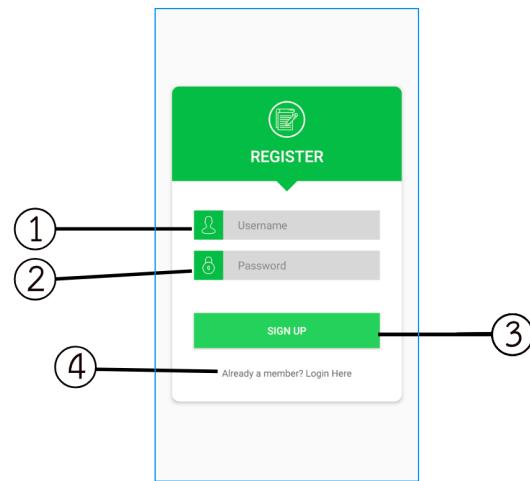
- หน้าจอเข้าสู่ระบบจะแสดงเมื่อสมาชิกเข้าสู่ระบบครั้งแรก ดังแสดงในรูปที่ ค.1



รูปที่ ค.1: หน้าจอแสดงการเข้าสู่ระบบ

จากรูปที่ ค.1 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลข 1 คือ ช่องใส่ Username
  - หมายเลข 2 คือ ช่องใส่ Password
  - หมายเลข 3 คือ ปุ่มสำหรับเข้าสู่ระบบ
  - หมายเลข 4 คือ ปุ่มสำหรับสมัครสมาชิก
  - หมายเลข 5 คือ ปุ่มสำหรับการลืมรหัสผ่าน
- หน้าจอการสมัครสมาชิกจะแสดงเมื่อทำการกดปุ่ม SIG UP ในหน้าเข้าสู่ระบบ ดังแสดงในรูปที่ ค.2

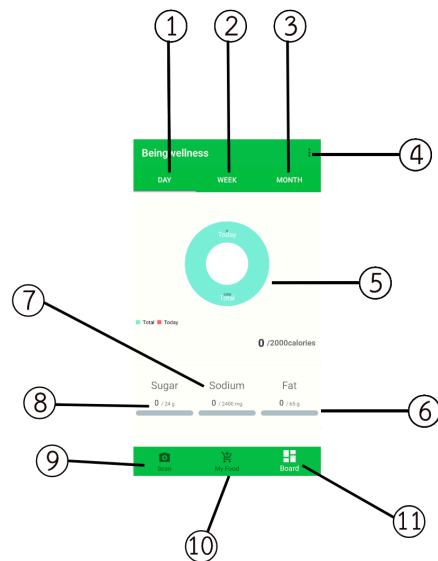


รูปที่ ค.2: หน้าจอแสดงการสมัครสมาชิก

จากรูปที่ ค.2 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลข 1 คือ ช่องใส่ Username
- หมายเลข 2 คือ ช่องใส่ Password
- หมายเลข 3 คือ ปุ่มสำหรับการสมัครสมาชิก
- หมายเลข 4 คือ ปุ่มสำหรับให้กลับไปเข้าสู่ระบบ

3. หน้าจอแสดงการปริโภครูปแบบวัน ดังแสดงในรูปที่ ค.3

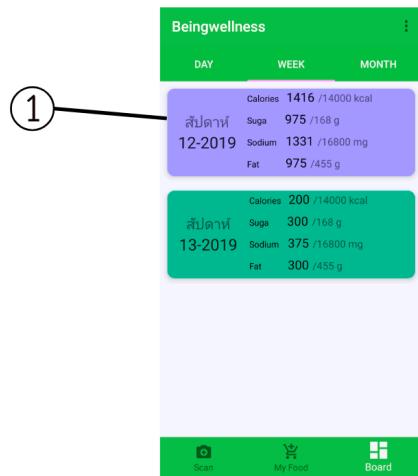


รูปที่ ค.3: หน้าจอแสดงการบริโภครูปแบบวัน

จากรูปที่ ค.3 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลข 1 คือ แถบการแสดงข้อมูลการบริโภครูปแบบวัน
- หมายเลข 2 คือ แถบการแสดงข้อมูลการบริโภครูปแบบสัปดาห์
- หมายเลข 3 คือ แถบการแสดงข้อมูลการบริโภครูปแบบเดือน
- หมายเลข 4 คือ ปุ่มเข้าสู่เมนูตั้งค่า
- หมายเลข 5 คือ Chart และตัวเลข การแสดงข้อมูลการบริโภคแคลลอรี
- หมายเลข 6 คือ Progressbar และตัวเลข การแสดงข้อมูลการบริโภคไขมัน
- หมายเลข 7 คือ Progressbar และตัวเลข การแสดงข้อมูลการบริโภคโซเดียม
- หมายเลข 8 คือ Progressbar และตัวเลข การแสดงข้อมูลการบริโภคน้ำตาล
- หมายเลข 9 คือ เมนูสแกน
- หมายเลข 10 คือ เมนูเพิ่มอาหาร
- หมายเลข 11 คือ เมนูแดชบอร์ด

4. หน้าจอแสดงการบริโภครูปแบบสัปดาห์ ดังแสดงในรูปที่ ค.4

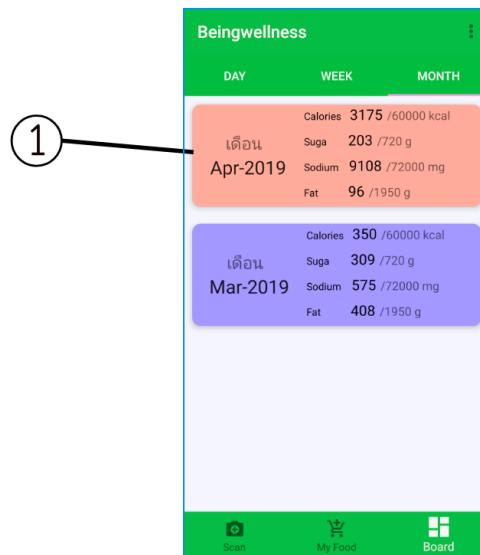


รูปที่ ค.4: หน้าจอแสดงการบริโภครูปแบบสัปดาห์

จากรูปที่ ค.4 สามารถอธิบายการใช้งานได้ดังนี้

– หมายเลข 1 คือ รายการอาหารและข้อมูลการบริโภครูปแบบสัปดาห์

5. หน้าจอแสดงการบริโภครูปแบบเดือน ตั้งแสดงในรูปที่ ค.5



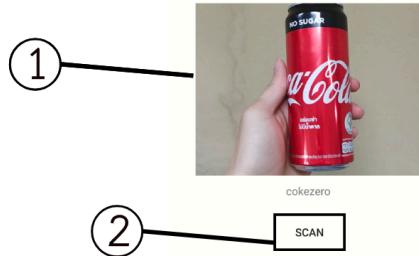
รูปที่ ค.5: หน้าจอแสดงการบริโภครูปแบบเดือน

จากรูปที่ ค.5 สามารถอธิบายการใช้งานได้ดังนี้

– หมายเลข 1 คือ รายการอาหารและข้อมูลการบริโภครูปแบบสัปดาห์

6. หน้าจอแสดงสแกนอาหาร ดังแสดงในรูปที่ ค.6

Beingwellness



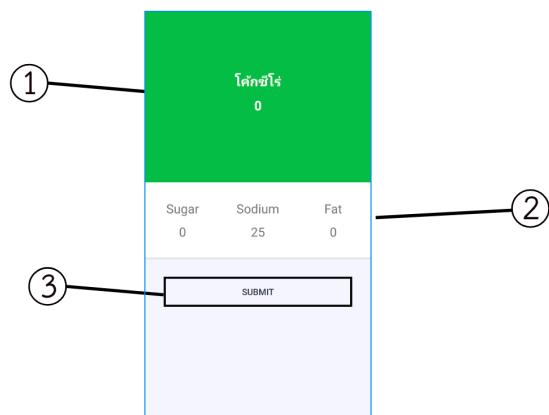
Scan My Food Board

รูปที่ ค.6: หน้าจอแสดงสแกนอาหาร

จากรูปที่ ค.6 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลข 1 คือ กล้องที่จะทำการสแกน
- หมายเลข 2 คือ ปุ่มสแกน

7. หน้าจอแสดงข้อมูลอาหาร ดังแสดงในรูปที่ ค.7

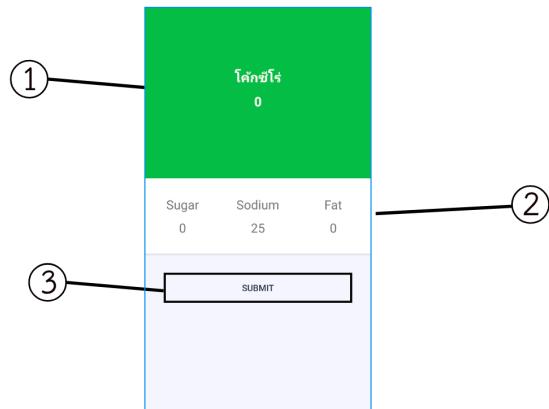


รูปที่ ค.7: หน้าจอแสดงสแกนอาหาร

จากรูปที่ ค.7 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลข 1 คือ ส่วนแสดงข้อมูลอาหารได้แก่ ชื่ออาหารและแคลอรี่
- หมายเลข 2 คือ ส่วนแสดงข้อมูลอาหารได้แก่ น้ำตาล โซเดียม และไขมัน
- หมายเลข 3 คือ ปุ่มทำการบันทึกข้อมูลอาหาร

8. หน้าจอแสดงการเปลี่ยน E-mail และ Password ดังแสดงในรูปที่ ค.8

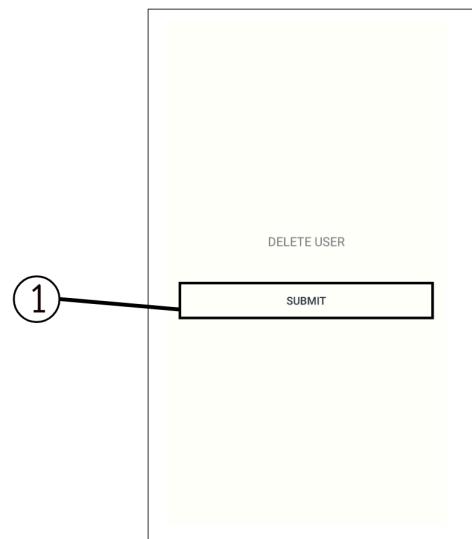


รูปที่ ค.8: หน้าจอแสดงสแกนอาหาร

จากรูปที่ ค.8 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลข 1 คือ ส่วนที่ใส่ E-mail หรือ Password
- หมายเลข 2 คือ ปุ่มทำการยืนยัน

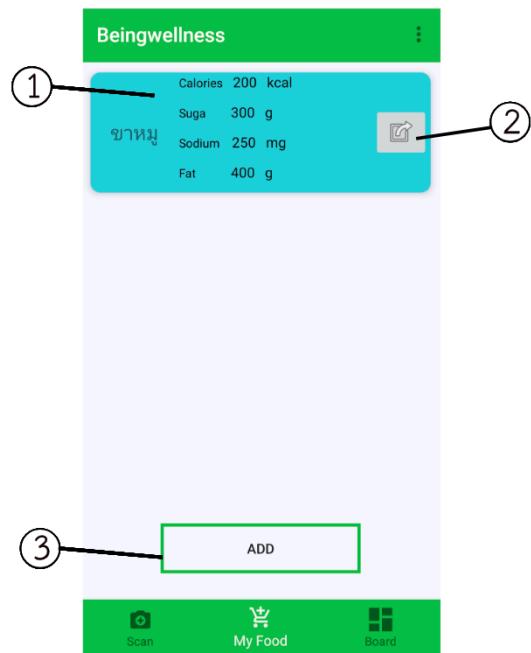
9. หน้าจอแสดงการลบผู้ใช้ ค.9



รูปที่ ค.9: หน้าจอแสดงสแกนอาหาร

จากรูปที่ ค.9 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลข 1 คือ ปุ่มทำการลบสมาชิก
10. หน้าจอรายการอาหารที่เพิ่ม ค.10

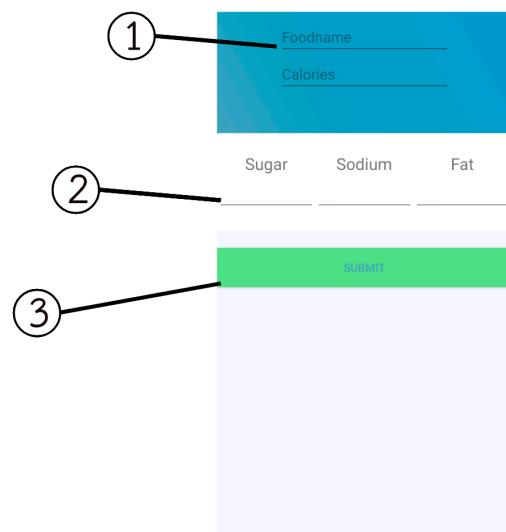


รูปที่ ค.10: หน้าจอรายการอาหารที่เพิ่ม

จากรูปที่ ค.10 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลข 1 คือ รายการอาหารที่เพิ่ม
- หมายเลข 2 คือ ปุ่มสำหรับบันทึกข้อมูลอาหาร
- หมายเลข 3 คือ ปุ่มสำหรับเพิ่มอาหาร

#### 11. หน้าจอเพิ่มอาหาร ค.11

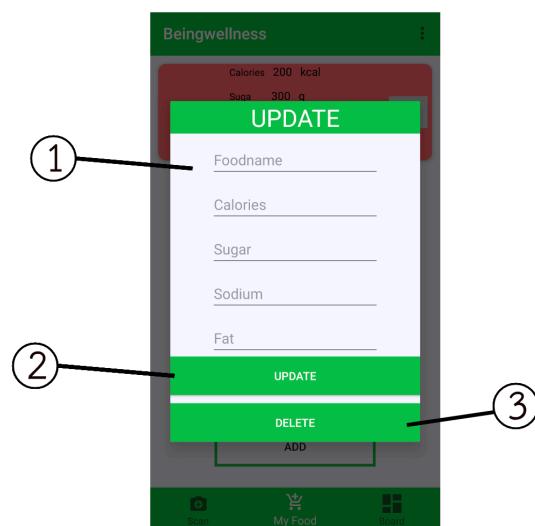


รูปที่ ค.11: หน้าจอเพิ่มอาหาร

จากรูปที่ ค.11 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลข 1 คือ ส่วนของการเพิ่มชื่ออาหารและแคลลอรี่
- หมายเลข 2 คือ ส่วนของการเพิ่มน้ำตาล โดยเดิม ไขมัน
- หมายเลข 3 คือ ปุ่มสำหรับเพิ่มอาหาร

## 12. หน้าจออัพเดตอาหารที่เพิ่ม ค.12



รูปที่ ค.12: หน้าจออัพเดตอาหารที่เพิ่ม

จากรูปที่ ค.12 สามารถอธิบายการใช้งานได้ดังนี้

- หมายเลข 1 คือ ส่วนของการอัพเดท ชื่ออาหาร เคลลอรี น้ำตาล โซเดียมและไขมัน
- หมายเลข 2 คือ ปุ่มอัพเดท
- หมายเลข 3 คือ ปุ่มลบอาหาร

## ประวัติผู้เขียน

ชื่อ-สกุล: นายพัทธดนย์ จัยสิน

รหัสประจำตัวนักศึกษา: 5811403741

วันเกิด: 25 06 2540

ที่อยู่ที่สามารถติดต่อได้: 25 ม.8 ต.ศรีแก้ว อ.ศรีรัตน์ จ.ศรีสะเกษ 33240

เบอร์โทรศัพท์: (+66) 91 827 6378

อีเมลล์: Pattadon.Ja.59@ubu.ac.th

ระดับมัธยมต้น: โรงเรียน กันทรลักษ์วิทยา จังหวัด ศรีสะเกษ

ระดับมัธยมปลาย: โรงเรียน กันทรลักษ์วิทยา จังหวัด ศรีสะเกษ

ระดับอุดมศึกษา: ภาควิชาคณิตศาสตร์ สถิติ และคอมพิวเตอร์ สาขาวิชาการ คอมพิวเตอร์  
คณะวิทยาศาสตร์ มหาวิทยาลัยอุบลราชธานี