



โปรแกรมคำนวณค่าโภชนาการอาหาร
 เพื่อเป็นแนวทางการควบคุมน้ำหนัก บนสมาร์ตโฟน ระบบปฏิบัติการแอนดรอยด์
 NUTRITION CALCULATOR PROGRAM FOR WEIGHT CONTROL GUIDE LINE
 ON ANDROID SMART PHONE



นายนิติวุฒิ หงษ์สุวรรณ รหัส 50370608
 นายสุบิน สุทธิ รหัส 50371292

ปริญญาบัตรนี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิทยาศาสตรบัณฑิต
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
 คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2555

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ..... 20.06.2558
เลขทะเบียน..... 16526820
เลขเรียกหนังสือ..... 1/6
มหาวิทยาลัยนเรศวร ๑๕ ๕๘ ๒

2555



ใบรับรองโครงการ

หัวข้อโครงการ โปรแกรมคำนวณค่าโภชนาการอาหารเพื่อเป็นแนวทางการควบคุมน้ำหนัก
บนสมาร์ตโฟน ระบบปฏิบัติการแอนดรอยด์

ผู้ดำเนินโครงการ นายนิติวุฒิ หงษ์สุวรรณ รหัส 50370608
นายสุบิน สุทธิ รหัส 50371292

ที่ปรึกษาโครงการ อาจารย์เสรมฐา ตั้งคำวานิช

สาขาวิชา วิศวกรรมคอมพิวเตอร์

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2555

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญาบัตรฉบับนี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรวิศวกรรมศาสตรบัณฑิตสาขาวิชาวิศวกรรมคอมพิวเตอร์
คณะกรรมการสอบ โครงการวิศวกรรม

.....
ได้.ม.ต. ศรี ดักน ที่ปรึกษาโครงการ
(อาจารย์เสรมฐา ตั้งคำวานิช)

.....กรรมการ
(อาจารย์กาญจพงศ์ สอนคม)

.....กรรมการ
(อาจารย์รัฐภูมิ วรรณสาสน์)

.....กรรมการ
(ดร.วรลักษณ์ คงเด่นฟ้า)

หัวข้อโครงการ โปรแกรมคำนวณค่าโภชนาการอาหารเพื่อเป็นแนวทางการควบคุมน้ำหนัก บนสมาร์ตโฟน ระบบปฏิบัติการแอนดรอยด์
ผู้ดำเนินโครงการ นายนิติวุฒิ หงษ์สุวรรณ รหัส 50370608
นายสุบิน สุทธิ รหัส 50371292
ที่ปรึกษาโครงการ อาจารย์เศรษฐา ตั้งคำวานิช
สาขาวิชา วิศวกรรมคอมพิวเตอร์
ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา 2555

บทคัดย่อ

โครงการนี้เป็นการพัฒนาโปรแกรมคำนวณค่าโภชนาการอาหารเพื่อเป็นแนวทางการควบคุมน้ำหนัก บนสมาร์ตโฟน ระบบปฏิบัติการแอนดรอยด์ เนื่องจากในปัจจุบัน ผู้คนหันมาควบคุมน้ำหนักมากขึ้นแต่ยังขาดวินัยในการควบคุมน้ำหนัก หรือความสม่ำเสมอตนเอง ด้วยปัญหานี้เราจึงคิดที่จะทำให้การควบคุมน้ำหนัก เป็นเรื่องที่เข้าถึงง่ายให้อยู่ในรูปแบบที่ใกล้ตัวมากที่สุด เพื่อความสะดวกในการควบคุมน้ำหนัก จึงได้ข้อสรุปว่าต้องเป็นการพัฒนาโปรแกรมบนโทรศัพท์มือถือ เพราะปฏิเสธไม่ได้ว่าในปัจจุบันโทรศัพท์มือถือ นับว่าเป็นสิ่งจำเป็นสำหรับการดำเนินชีวิตของคนในยุคปัจจุบัน การควบคุมน้ำหนักด้วยโทรศัพท์มือถือ จึงเป็นสิ่งที่น่าจะทำให้การควบคุมน้ำหนักมีประสิทธิภาพมากขึ้น

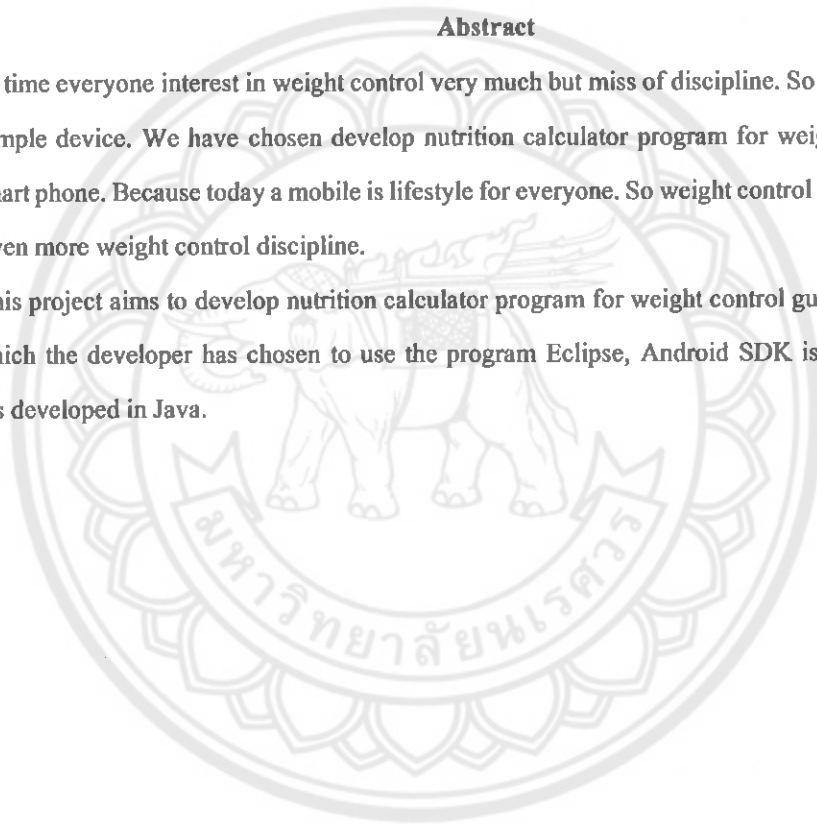
ในโครงการนี้ผู้พัฒนาได้เลือกใช้ โปรแกรม Eclipse และ Android SDK เป็นเครื่องมือในการพัฒนา และใช้ภาษาในการพัฒนาเป็นภาษาจาวา

Project Title Nutrition calculator program for weight control guide line on android smart phone
Name Mr. Nitiwut Hongsuwan ID. 50370608
Mr. Subin Sutti ID. 50371292
Project Advisor Mr. Settha Thangkawanit
Major Computer Engineering
Department Electrical and Computer Engineering
Academic Year 2013

.....
Abstract

In time everyone interest in weight control very much but miss of discipline. So form problem we have to think in simple device. We have chosen develop nutrition calculator program for weight control guide line on android smart phone. Because today a mobile is lifestyle for everyone. So weight control with smart phone is likely to cause even more weight control discipline.

This project aims to develop nutrition calculator program for weight control guide line on android smart phone. Which the developer has chosen to use the program Eclipse, Android SDK is a development tool and language is developed in Java.



กิตติกรรมประกาศ

การจัดทำโครงการ โปรแกรมคำนวณค่าโภชนาการอาหารเพื่อเป็นแนวทางการควบคุมน้ำหนัก บนสมาร์ทโฟน ระบบปฏิบัติการแอนดรอยด์ สามารถพัฒนาขึ้นจนสำเร็จลุล่วงไปได้ด้วยดี เพราะได้รับการสนับสนุนจากภาควิชาชีพวิศวกรรมไฟฟ้าและคอมพิวเตอร์ที่ได้ให้เงินสนับสนุนการทำโครงการนี้ และขอขอบพระคุณอาจารย์ เศรษฐา ตั้งคำวานิช ซึ่งเป็นที่ปรึกษาของโครงการนี้เป็นอย่างมาก ที่ให้คำปรึกษา คำแนะนำและแนวทางการพัฒนาโปรแกรมในด้านต่างๆ ที่ทำให้ผู้จัดทำโครงการนี้ทำงานผ่านไปได้ด้วยดี

ขอขอบพระคุณคณะกรรมการที่ได้ติชมและเสนอแนะแนวทางการพัฒนาโปรแกรมให้มีความสมบูรณ์ รวมถึงให้คำปรึกษาในการจัดทำโครงการนี้

นายนิติวุฒิ หงษ์สุวรรณ

นายสุบิน สุทธิ



สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	ก
บทคัดย่ออังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญรูป.....	ช
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบเขตของโครงการ.....	1
1.4 ขั้นตอนการดำเนินการ.....	2
1.5 ผลที่คาดว่าจะได้รับ.....	3
1.6 งบประมาณของโครงการ.....	3
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง.....	4
2.1 เทคโนโลยีภาษาจาวา.....	4
2.1.1 จุดมุ่งหมายของภาษาจาวา.....	4
2.1.2 การเขียนโปรแกรมเชิงวัตถุ (OOP = Object-Oriented Programming).....	4
2.1.3 แนวคิดของการเขียนโปรแกรมเชิงวัตถุ (OOP Concepts).....	4
2.2 แอนดรอยด์ (Android)	5
2.2.1 เวอร์ชันของแอนดรอยด์ (Android Version)	5
2.2.2 ภาพรวมของแอนดรอยด์ 4.0.....	5
2.2.3 Android SDK (Android Software Development Kit).....	6
2.3 SQLite.....	7
2.4 XML.....	7
2.4.1 จุดเด่นของ XML.....	8
2.5 BMI และ BMR.....	9

สารบัญ (ต่อ)

หน้า

บทที่ 3 วิธีการดำเนินงาน.....	10
3.1 Use Case Diagram.....	10
3.2 Flow Chart.....	11
3.3 รายละเอียดของการรับข้อมูลจากผู้ใช้.....	12
1. ข้อมูลพื้นฐานส่วนตัว (Profile)	12
2. ค่าแคลอรีอาหาร (Food)	12
3. ค่าการเบิร์นแคลอรี (Activity)	12
3.4 การทำงานของโปรแกรม.....	13
3.5 การออกแบบหน้าตาโปรแกรม.....	17
3.6 การออกแบบระบบการทำงานของโปรแกรม.....	19
บทที่ 4 ผลการทดลอง.....	29
4.1 หน้าหลัก (Main Page)	29
4.2 การใช้งานปุ่ม การนำเข้าพลังงาน (Food)	32
1. ทดสอบการนำเข้าพลังงานทั้งแบบรายวัน.....	32
2. ทดสอบการนำเข้าพลังงานทั้งแบบย้อนหลัง.....	33
3. ทดสอบการใช้งานปุ่มการจำลองการเผาผลาญ.....	34
4. ทดสอบปุ่ม View Eat เพื่อดูรายการอาหารที่นำเข้าในแต่ละวัน.....	35
4.3 การใช้งานปุ่ม กิจกรรมการเผาผลาญพลังงาน (Activity)	36
1. ทดสอบการใส่เวลาหาค่าแคลอรี.....	36
2. ทดสอบการจับเวลาหาค่าแคลอรี.....	37
3. ทดสอบการ Share ผ่าน Google Plus.....	38
4.4 การใช้งานปุ่ม ข้อมูลส่วนตัว (Profile)	39
4.5 การใช้งานปุ่มรายงานผล (Result)	41
4.6 ทดสอบการจัดการข้อมูล Import และ Export.....	47
บทที่ 5 สรุปและผลดำเนินงาน.....	49
5.1 สรุปและอภิปรายผล.....	49

สารบัญ (ต่อ)

	หน้า
5.2 ปัญหาที่พบในการพัฒนาระบบ.....	50
5.3 ข้อเสนอแนะ.....	50
5.4 ความสามารถของโปรแกรม.....	50
เอกสารอ้างอิง.....	51
ภาคผนวก ก คู่มือการติดตั้ง Plug in Android บน eclipse.....	52
ก.1 ติดตั้ง ADT (android developer tools)	52
ก.2 ติดตั้ง Platform Android เพิ่ม/ลบ.....	55
ก.3 สร้าง Emulator สำหรับ ทดสอบ ไลค์แอนดรอยด์.....	57
ภาคผนวก ข ตัวอย่างโค้ดที่ใช้พัฒนา.....	61
ประวัติผู้ดำเนินโครงการ.....	88



สารบัญรูป

	หน้า
รูปที่ 2.1 แสดงภาพ Android SDK เมื่อทำการ Run.....	6
รูปที่ 2.2 แสดงตัวอย่าง เอกสาร XML.....	8
รูปที่ 3.1 แสดง Use case Diagram ของโปรแกรม.....	10
รูปที่ 3.2 แสดง Flow Chart ของโปรแกรม.....	11
รูปที่ 3.3 System Diagram รับข้อมูลพื้นฐาน.....	13
รูปที่ 3.4 System Diagram หน้าหลักของโปรแกรม.....	14
รูปที่ 3.5 System Diagram รับค่าอาหาร.....	14
รูปที่ 3.6 System Diagram รับค่ากิจกรรม.....	15
รูปที่ 3.7 System Diagram หน้ารายงานผล.....	15
รูปที่ 3.8 แสดงขั้นตอนการทำงานของโปรแกรม.....	16
รูปที่ 3.9 ออกแบบหน้าตาโปรแกรม.....	17
รูปที่ 3.10 ออกแบบหน้าตาโปรแกรม.....	18
รูปที่ 3.11 ออกแบบระบบการทำงานของโปรแกรม.....	19
รูปที่ 3.12 ออกแบบระบบการทำงานของโปรแกรม.....	19
รูปที่ 3.13 ออกแบบระบบการทำงานของโปรแกรม.....	20
รูปที่ 3.14 ออกแบบระบบการทำงานของโปรแกรม.....	21
รูปที่ 3.15 ออกแบบระบบการทำงานของโปรแกรม.....	22
รูปที่ 3.16 ออกแบบระบบการทำงานของโปรแกรม.....	23
รูปที่ 3.17 ออกแบบระบบการทำงานของโปรแกรม.....	24
รูปที่ 3.18 ออกแบบระบบการทำงานของโปรแกรม.....	25
รูปที่ 3.19 ออกแบบระบบการทำงานของโปรแกรม.....	26
รูปที่ 3.20 ออกแบบระบบการทำงานของโปรแกรม.....	27
รูปที่ 3.21 ออกแบบระบบการทำงานของโปรแกรม.....	28
รูปที่ 4.1 แสดงผลหน้าหลัก.....	29
รูปที่ 4.2 แสดงรายการน้ำหนั.....	30
รูปที่ 4.3 การอัพเดทน้ำหนั.....	31
รูปที่ 4.4 ทดสอบการนำเข้าพลังงาน	32

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 4.5 ทดสอบการนำเข้าพลังงาน.....	33
รูปที่ 4.6 ทดสอบการจำลองการเผาผลาญ.....	34
รูปที่ 4.7 ทดสอบปุ่ม View Eat.....	35
รูปที่ 4.8 ทดสอบปุ่ม View Eat.....	35
รูปที่ 4.9 ทดสอบการใส่เวลา.....	36
รูป 4.10 ทดสอบการจับเวลาหาแคลอรี.....	37
รูป 4.11 ทดสอบการ Share ผ่าน Google Plus.....	38
รูปที่ 4.12 ผลการแสดงผลข้อมูลส่วนตัว.....	39
รูปที่ 4.13 ทดสอบการแก้ไขข้อมูล.....	40
รูปที่ 4.14 ทดสอบการแสดงผลทุกวัน.....	41
รูปที่ 4.15 ทดสอบการแสดงผลแบบเลือกวัน.....	42
รูปที่ 4.16 การแสดงผลของกราฟ.....	43
รูปที่ 4.17 กราฟน้ำหนัก.....	44
รูปที่ 4.18 กราฟการรักษาค่าได้.....	45
รูปที่ 4.19 กราฟแสดงผลการกินต่อวัน.....	46
รูปที่ 4.20 Export ข้อมูล.....	47
รูปที่ 4.21 Import ข้อมูล.....	48
รูปที่ ก.1 แสดงขั้นตอนการติดตั้ง ADT.....	49
รูปที่ ก.2 แสดงขั้นตอนการติดตั้ง ADT.....	49
รูปที่ ก.3 แสดงขั้นตอนการติดตั้ง ADT.....	50
รูปที่ ก.4 แสดงขั้นตอนการติดตั้ง ADT.....	50
รูปที่ ก.5 แสดงขั้นตอนการติดตั้ง ADT.....	50
รูปที่ ก.6 แสดงขั้นตอนการติดตั้ง ADT.....	50
รูปที่ ก.7 แสดงขั้นตอนการติดตั้ง ADT.....	51
รูปที่ ก.8 แสดงขั้นตอนการติดตั้ง ADT.....	51
รูปที่ ก.9 แสดงขั้นตอนการติดตั้ง ADT.....	51
รูปที่ ก.10 แสดงขั้นตอนการติดตั้ง ADT.....	52
รูปที่ ก.11 ติดตั้ง Platform Android เพิ่ม/ลด.....	52

สารบัญรูป (ต่อ)

	หน้า
รูปที่ ก.12 ติดตั้ง Platform Android เพิ่ม/ลบ.....	53
รูปที่ ก.13 ติดตั้ง Platform Android เพิ่ม/ลบ.....	53
รูปที่ ก.14 ติดตั้ง Platform Android เพิ่ม/ลบ.....	53
รูปที่ ก.15 ติดตั้ง Platform Android เพิ่ม/ลบ.....	54
รูปที่ ก.16 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์.....	54
รูปที่ ก.17 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์.....	55
รูปที่ ก.18 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์.....	55
รูปที่ ก.19 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์.....	55
รูปที่ ก.20 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์.....	56
รูปที่ ก.21 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์.....	56
รูปที่ ก.22 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์.....	57
รูปที่ ก.23 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์.....	57



บทที่ 1

บทนำ

1.1 แนวคิดและที่มาของปริญญาณิพนธ์

ปัจจุบัน ผู้คนหันมาควบคุมน้ำหนักกันมากขึ้นแต่ยังขาดวินัยในการควบคุม หรือความสม่ำเสมอ นั่นเอง ด้วยปัญหานี้เราจึงคิดที่จะทำให้การควบคุมน้ำหนักเป็นเรื่องที่เข้าถึงง่ายให้อยู่ในรูปแบบที่ใกล้ตัวมากที่สุด เพื่อความสะดวกในการควบคุมน้ำหนัก จึงได้ข้อสรุปว่าต้องเป็นการพัฒนาโปรแกรมบนโทรศัพท์มือถือ จึงเป็นสิ่งที่น่าจะทำให้การควบคุมน้ำหนักมีประสิทธิภาพมากขึ้น รูปแบบการใช้งานโปรแกรม จะทำการรับข้อมูลจากผู้ใช้แล้วนำไปคำนวณหาค่าทางสุขภาพ เช่น ค่าดัชนีมวลกาย (BMI: Body Mass Index) และ พลังงานที่ใช้ต่อวันในภาวะร่างกายปกติ (BMR: Basal Metabolism Rate) เพื่อใช้ในการวางแผนควบคุมน้ำหนัก โดยสามารถตรวจสอบผลของการควบคุมน้ำหนัก ในรูปแบบของ กราฟน้ำหนัก,กราฟการรักษา ค่าพลังงาน ได้ และ กราฟแสดงการกินเปรียบเทียบกับค่าพลังงานที่ร่างกายควรได้รับต่อวัน ถ้าปฏิบัติอยู่ในกรอบของโปรแกรม น่าจะสามารถควบคุมน้ำหนักได้อย่างมีประสิทธิภาพ

1.2 วัตถุประสงค์

1. พัฒนาโปรแกรมการควบคุมน้ำหนักที่มีประสิทธิภาพสะดวกแก่การใช้งาน
2. เพื่อความสะดวกในการวางแผนการควบคุมน้ำหนักอย่างเป็นระบบ สร้างนิสัยการดูแลตนเองเพื่อคุณภาพชีวิตที่ดีขึ้น
3. เพื่อให้การบันทึกข้อมูลสามารถบันทึกได้ทุกสถานที่และทันต่อเหตุการณ์

1.3 ขอบเขตของโครงการ

1. พัฒนาโปรแกรมที่ใช้บนโทรศัพท์มือถือ สามารถบันทึกข้อมูล ปริมาณค่าโภชนาการของอาหาร ค่าพลังงานที่ใช้ในแต่ละวัน และ คำนวณค่าที่ได้จากค่าพลังงานที่ได้จากอาหารและค่าพลังงานที่ใช้ในแต่ละวัน เพื่อสรุปว่าในแต่ละวันสามารถลดการนำเข้าพลังงานไปได้มากน้อยเท่าไร
2. ตัวโปรแกรมสามารถวางแผนการควบคุมน้ำหนักให้กับผู้ใช้ล่วงหน้าได้ โดยกำหนดการลดการนำเข้าพลังงานต่อวันจากผู้ใช้
3. ภาษาที่ใช้กับโปรแกรมคือภาษาไทย เมนูอาหารสำหรับคนไทย

1.5 ผลที่คาดว่าจะได้รับ

1. ได้โปรแกรมควบคุมน้ำหนักที่ใช้บนโทรศัพท์มือถือ โดยสามารถบันทึกข้อมูล ปริมาณค่าโภชนาการของอาหาร ค่าพลังงานที่ใช้ในแต่ละวัน และ คำนวณค่าที่ได้จากค่าพลังงานที่ได้จากอาหาร และค่าพลังงานที่ใช้ในแต่ละวัน เพื่อสรุปว่าในแต่ละวันสามารถลดการนำเข้าพลังงานไปได้มากน้อยเท่าไร

2. ผู้ใช้สามารถใช้โปรแกรมบนโทรศัพท์มือถือในการดูและแก้ไขข้อมูลการควบคุมน้ำหนัก และรายละเอียดต่างๆ ของผู้ใช้ได้

3. นักศึกษาได้เรียนรู้การพัฒนาโปรแกรมด้วยภาษาคอมพิวเตอร์

5. นักศึกษาได้เรียนรู้ระบบการส่งข้อมูลระหว่างโทรศัพท์มือถือกับคอมพิวเตอร์

6. นักศึกษาได้เรียนรู้การทำงานเป็นกลุ่ม

7. สามารถนำโปรแกรมที่ได้ไปใช้กับการวางแผนในการควบคุมน้ำหนักได้

8. โปรแกรมสามารถนำไปพัฒนาต่อเพื่อประยุกต์ใช้กับการดูแลสุขภาพประเภทอื่นได้

1.6 งบประมาณที่ใช้

1. อุปกรณ์เชื่อมต่อต่างๆ	1000 บาท
2. ค่าเอกสารและวัสดุอุปกรณ์อื่นๆ	1000 บาท
รวมเป็นเงินทั้งสิ้น	2000 บาท

บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

ในการศึกษาข้อมูลเพื่อนำมาเขียนโปรแกรมบนมือถือระบบปฏิบัติการแอนดรอยด์ นั้นต้องมีความรู้ความเข้าใจในหลายๆส่วนด้วยกัน เพื่อให้โปรแกรมที่พัฒนาได้มีประสิทธิภาพ ตรงตามวัตถุประสงค์ที่ตั้งไว้ในบทนี้จะพูดถึง ภาษาจาวา, แอนดรอยด์, SQLite , XML และค่า BMI กับ BMR

2.1 เทคโนโลยีภาษาจาวา

ภาษาจาวานั้นเป็นภาษาโปรแกรมเชิงวัตถุที่ถูกพัฒนาขึ้น โดยบริษัท Sun Microsystems ซึ่งมีจุดประสงค์หลักที่ทำการพัฒนาก็เพื่อนำมาใช้แทนภาษา C++

2.1.1 จุดมุ่งหมายของภาษาจาวา

1. ใช้ภาษาโปรแกรมเชิงวัตถุ
2. ไม่ขึ้นกับแพลตฟอร์ม (platform) สถาปัตยกรรม และระบบปฏิบัติการ
3. เหมาะสำหรับการใช้ในระบบเครือข่าย เพราะมีไลบรารี (library) ให้เรียกใช้
4. เรียกใช้งานจากระยะไกลได้อย่างปลอดภัย

2.1.2 การเขียนโปรแกรมเชิงวัตถุ (OOP = Object-Oriented Programming)

การเขียนโปรแกรมที่ประกอบด้วยกลุ่มของวัตถุ(Object) แต่ละวัตถุจะจัดเป็นกลุ่มในรูปแบบของคลาส ซึ่งแต่ละคลาสอาจจะมีคุณสมบัติ การปกป้อง (Encapsulation), การสืบทอด (Inheritance) และการพ้องรูป (Polymorphism)

2.1.3 แนวคิดของการเขียนโปรแกรมเชิงวัตถุ (OOP Concepts)

- 1.การปกป้อง (Encapsulation)
 - การรวมกลุ่มของข้อมูลและกลุ่มของโปรแกรมเพื่อการปกป้องและเลือกตอบสนอง
- 2.การสืบทอด (Inheritance)
 - ยอมให้นำไปใช้หรือเขียนขึ้นมาทดแทนของเดิม
- 3.การพ้องรูป (Polymorphism)
 - Overloading มีชื่อโปรแกรมเดียวกันแต่หลายรายการตัวแปร (Parameter) ต่างกัน
 - Overriding มีชื่อโปรแกรมและตัวแปรเหมือนกันเพื่อเขียน behavior ขึ้นมาใหม่

2.2 แอนดรอยด์ (Android)

แอนดรอยด์ในที่นี้หมายถึงระบบปฏิบัติการบน Smart Phone หรือ Tablet ที่ถูกพัฒนาจากระบบปฏิบัติการ Linux การพัฒนาโปรแกรมที่ทำงานบนระบบปฏิบัติการแอนดรอยด์จะใช้ภาษาจาวา (Java) เป็นหลัก โดยผ่าน Google Developed Java Libraries ที่พัฒนาโดย บริษัท กูเกิล (Google)

2.2.1 เวอร์ชันของแอนดรอยด์ (Android Version)

ตารางที่ 2.1 เวอร์ชันของแอนดรอยด์

เวอร์ชัน	ชื่อ	เปิดตัว	API level
1.5	Cupcake	30 เมษายน 2552	3
1.6	Donut	15 กันยายน 2552	4
2.0-2.1	Eclair	26 ตุลาคม 2552	7
2.2	Froyo	20 พฤษภาคม 2553	8
2.3-2.3.2	Gingerbread	6 ธันวาคม 2553	9
2.3.3-2.3.7	Gingerbread	9 กุมภาพันธ์ 2554	10
3.1	Honeycomb	10 พฤษภาคม 2554	12
3.2	Honeycomb	15 กรกฎาคม 2554	13
4.0.x	Ice Cream Sandwich	16 ธันวาคม 2554	14 - 15
4.1.x	Jelly Bean	9 กรกฎาคม 2555	16
4.2.x	Jelly Bean	13 พฤศจิกายน 2555	17

2.2.2 ภาพรวมของแอนดรอยด์ 4.0

1. USER INTERFACE: Ice Cream Sandwich

UI บนระบบปฏิบัติการแอนดรอยด์ 4.0 Ice Cream Sandwich เป็นการผสมระหว่างระบบปฏิบัติการแอนดรอยด์ 2.3 Gingerbread และระบบปฏิบัติการแอนดรอยด์ 3.0 Honeycomb แต่กลิ่นอายของระบบปฏิบัติการ Android 4.0 Ice Cream Sandwich หน้าจอตอนปลดคือรวมไปถึงปุ่ม 3 ปุ่มอย่าง ปุ่ม Back, Home และ Multitasking นอกจากนี้ หน้า Home screen นั้น ยังสามารถสร้างโฟลเดอร์ รวมแอปพลิเคชันให้เป็นหมวดหมู่ได้

2. NOTIFICATION: Ice Cream Sandwich

การใช้งานเป็นแบบ Dropdown เหมือนเดิม แต่ภาพรวมของ Notification นั้น ถูกปรับปรุงให้ใช้งานได้มีประสิทธิภาพมากขึ้นกว่าเดิม (ใช้งานง่ายขึ้น) และสามารถดูในส่วนของ Notification ได้จากหน้า Lock Screen ได้ทันที เช่นเดียวกับหมวดกล้องถ่ายรูป โดยไม่ต้องทำการปลดล็อกเหมือนแต่ก่อน

มาในส่วนของ Virtual Keyboard ได้มีการแก้ไขในเรื่องของการสะกดคำ และการแก้ไขคำให้ถูกต้องได้เป็นอย่างดี และหน้าของ Visual Multitasking สามารถเรียกใช้ Application ที่ถูกเปิดใช้งานก่อนหน้านี้ ได้จากคณุ่มนี้เพียงปุ่มเดียว

3. DATA USAGE & FACE UNLOCK: Ice Cream Sandwich

Data Usage เป็นตัวช่วยควบคุมการใช้งานของเครื่อง ไม่ให้มีการใช้งานเกินขีดจำกัด และ Face Unlock โดยทำการสแกนใบหน้าเจ้าของเครื่องเพื่อนำมาทำการ Unlock เครื่อง

4. ANDROID BEAM: Ice Cream Sandwich

เป็นการเชื่อมต่อกับเทคโนโลยี NFC ซึ่งเป็นแชร์ข้อมูลระหว่างเครื่องที่มีระบบ NFC ด้วยกัน โดยการนำเครื่องมาวางใกล้ๆ กัน แล้ว Beam นั่นเอง

2.2.3 Android SDK (Android Software Development Kit)

เป็นโปรแกรมสำหรับนักพัฒนาที่เปิดให้เข้ามาพัฒนาแอปพลิเคชัน ซึ่งทาง Google ได้เปิดดาวน์โหลดไฟล์ Android SDK ฟรีเพื่อให้เกิดแอปพลิเคชันใหม่ๆ ใน Android SDK มี Emulator ทำให้ในระหว่างการพัฒนาโค้ดแอปพลิเคชัน สามารถจำลองการทำงานของมือถือแอนดรอยด์และทดลองใช้งานแอปพลิเคชันที่ได้ทำการพัฒนาโดยไม่จำเป็นต้องมีโทรศัพท์จริง



รูปที่ 2.1 แสดงภาพ Android SDK เมื่อทำการ Run

2.3 SQLite

ฐานข้อมูล SQLite เป็น Database ขนาดเล็กที่ได้รับความนิยมอย่างมากกับ Application ที่ทำงานบน Smart Phone ประเภทต่าง ๆ รูปแบบการทำงานของ SQLite เป็นแบบ Standalone ทำงานอยู่ใน Application นั้นๆ SQLite มีโครงสร้างง่ายต่อการจัดเก็บและนำไปใช้ และไฟล์ที่จัดเก็บนั้นก็มีความเล็กมากเกือบเท่ากับการเก็บข้อมูลจริง เพราะฉะนั้น SQLite Database จึงเหมาะสมกับ Application ที่ทำงานบน Smartphone อย่างยิ่ง โดยเฉพาะ อันเนื่องมาจากข้อจำกัดทางด้าน Hardware และ Memory รวมทั้งความสามารถในการ Process ข้อมูลต่าง ๆ ใน Smartphone ย่อมน้อยกว่า PC Desktop เป็นธรรมดา สำหรับ SQLite ถูกนำไปใช้กับ OS ที่ทำงานอยู่ใน Smartphone หลายตัว เช่น Windows Phone , iOS ของ Apple , Symbian หรือแม้กระทั่ง Android ก็สามารถนำ SQLite Database มาใช้ร่วมกับการจัดเก็บข้อมูลได้เช่นเดียวกัน

2.4 XML

XML ย่อมาจาก Extensible Markup Language คือภาษาหนึ่งที่ใช้ในการแสดงผลข้อมูล ซึ่งภาษาที่ใช้กำหนดรูปแบบของคำสั่งภาษา HTML หรือที่เรียกว่า Meta Data ซึ่งจะใช้สำหรับกำหนดรูปแบบของคำสั่ง Markup ต่าง ๆ ถ้าเปรียบเทียบกับภาษา HTML จะแตกต่างกันที่ HTML ถูกออกแบบมาเพื่อแสดงผลอย่างเดียวนั้น เช่น ให้แสดงผลตัวเล็ก ตัวหนา ตัวเอียง เหมือนที่คุณเคยเห็นในเว็บเพจทั่วไป แต่ภาษา XML นั้นถูกออกแบบมาเพื่อเก็บข้อมูล โดยทั้งข้อมูลและโครงสร้างของข้อมูลนั้นๆ ไปด้วยกัน ส่วนการแสดงผลก็จะใช้ภาษาเฉพาะ ซึ่งก็คือ XSL (Extensible Style sheet Language)

ภาษา XML มีโครงสร้างที่ประกอบด้วยแท็กเปิด และแท็กปิด เช่นเดียวกับภาษา HTML แต่ภาษา XML คุณสามารถสร้างแท็กรวมทั้งกำหนดโครงสร้างของข้อมูลได้เอง ซึ่งความสามารถครั้งนี้ตัวภาษา ทำไม่ได้ เพราะภาษา HTML ถูกกำหนดแท็กตายตัวโดย W3C หรือ World Wide Web Consortium อาจกล่าวได้ว่า XML เป็นส่วนเสริมของ HTML เพราะตัว XML ไม่สามารถแสดงผลได้ในตัวของมันเอง หากต้องการแสดงผลที่ถูกต้อง จะต้องมีการใช้ร่วมกับภาษาอื่น เช่น HTML, JSP, PHP , ASP หรือภาษาอื่น ๆ ที่สนับสนุน XML จะมีนามสกุลเป็น .XML สามารถสร้างขึ้นจากโปรแกรมประเภท Text Editor ใดก็ได้ เช่น Notepad, Editplus , Dreamweaver, MS Word เป็นต้น

สิ่งที่ถือได้ว่าเป็นเสน่ห์ของ XML นั้นจะเป็นความสะดวกในการจัดการด้านระบบการติดต่อกับผู้ใช้ จากโครงสร้างของข้อมูล เราสามารถนำข้อมูลจากหลายแหล่งมาแสดงผลและประมวลผลร่วมกันได้ ไม่ว่าจะ เป็นข้อมูลลูกค้า รายการสั่งซื้อ ผลการวิจัย รายการรับชำระเงินข้อมูลเวชระเบียน รายการสินค้าหรือข้อมูลสารสนเทศอื่นๆ ก็สามารถแปลงให้เป็น XML ได้ และในส่วนของข้อมูลสามารถปรับให้เป็น HTML ได้ สำหรับประโยชน์ในการใช้งานนั้น เราจะสามารถนำมาใช้สำหรับการเข้าถึงระบบข้อมูลขนาดใหญ่ใช้กับระบบเครือข่ายในองค์กร หรืออินเทอร์เน็ตเพื่อดูข้อมูลหรือเรียกใช้ข้อมูลที่ทำให้การแสดงผลทางหน้าจอที่รวดเร็ว

2.4.1 จุดเด่นของ XML

1. ดูเอกสาร ได้ง่าย สะดวก และได้ผลดีเหมือน HTML
2. สนับสนุนการประยุกต์ใช้กับงานต่าง ๆ และสนับสนุนโปรแกรมประยุกต์ต่าง ๆ
3. เขียนง่าย
4. อ่านได้ด้วยมนุษย์ โดยไม่ต้องอาศัย โปรแกรมหรือเครื่องมือช่วยแปล
5. การเขียน XML ทำได้ด้วยการใช้ Text editor ทั่วไป และไม่ต้องการเครื่องมือที่ซับซ้อน
6. ใช้เป็นตัวแทนข้อมูล (Meta data) จึงเป็นแนวทางในการขนส่งข้อมูล และสร้างการเชื่อมโยงระหว่างแอปพลิเคชัน ได้ง่าย
7. สนับสนุน UNICODE ทำให้ใช้ได้กับหลากหลายภาษา และผสมกันได้หลากหลายภาษา
8. ดึงเอกสาร XML มาใช้งานได้ง่าย และใช้ร่วมกับโปรแกรมประยุกต์อื่นได้ง่าย เช่น โปรแกรม DB2, Oracle, SAP เป็นต้น
9. นำไปประยุกต์ใช้ในการดำเนินกิจกรรมบนเครือข่ายได้มาก เช่น E-Business, EDI, E-Commerce, การจัดการ Supply chain / Demand chain management, การดำเนินการแบบ intranet และ Web Base Application

```

<NAME>Izzy</NAME>
<BREED>Siamese</BREED>
<AGE>6</AGE>
<ALTERED>yes</ALTERED>
<DECLAWED>no</DECLAWED>
<LICENSE>Izz1380od</LICENSE>
<OWNER>Colin Wilcox</OWNER>

```

รูปที่ 2.2 แสดงตัวอย่าง เอกสาร XML

2.5 BMI และ BMR

BMI (Body Mass Index)

การหาค่าดัชนีมวลกาย คือเป็นมาตรการที่ใช้ประเมินภาวะอ้วนและผอมในผู้ใหญ่ตั้งแต่อายุ 20 ปีขึ้นไป สามารถทำได้โดยการชั่งน้ำหนักตัวเป็นกิโลกรัม และวัดส่วนสูงเป็นเซนติเมตร แล้วนำมาหาค่าดัชนีมวลกายโดยใช้โปรแกรมวัดค่าความอ้วนข้างต้น

$$\text{BMI} = \text{น้ำหนักตัว เป็นกิโลกรัม} / \text{ความสูงเป็นเมตรยกกำลัง 2}$$

ตัวอย่างเช่น น้ำหนักตัว 60 กก. ส่วนสูง 170 ซม. เมื่อคำนวณผ่านโปรแกรมวัดค่าความอ้วน จะได้ค่า BMI ประมาณ 20.8 ดังนั้น ค่าดัชนีมวลกาย (BMI) โดยเกณฑ์ของ BMI ควรอยู่ระหว่าง 18.5 - 22.9 ผลที่ได้คือน้ำหนักปกติ

BMR (Basal Metabolism Rate)

Basal Metabolism Rate (BMR) คือ สูตรคำนวณพลังงานที่ใช้ต่อวันในภาวะร่างกายปกติเฉพาะคน ซึ่งออกแบบมาให้มีความแม่นยำ ช่วยกำหนดและจำกัดจำนวนพลังงานที่ร่างกายต้องการ โดยคิดจากน้ำหนัก ส่วนสูง อายุ เพศ และ โครงสร้างของร่างกาย

$$\text{สำหรับผู้ชาย: BMR} = 66 + (13.7 \times \text{น้ำหนักตัวเป็น กก.}) + (5 \times \text{ส่วนสูงเป็น ซม.}) - (6.8 \times \text{อายุ})$$

$$\text{สำหรับผู้หญิง: BMR} = 665 + (9.6 \times \text{น้ำหนักตัวเป็น กก.}) + (1.8 \times \text{ส่วนสูงเป็น ซม.}) - (4.7 \times \text{อายุ})$$

ตัวอย่างเช่น

$$\text{ผู้หญิง อายุ 35 ปี ส่วนสูง 165 ซม. น้ำหนัก 60 กก. BMR จะเท่ากับ } 665 + (9.6 \times 60) + (1.8 \times 165) - (4.7 \times 35) = 1373.5 \text{ แคลอรี}$$

หมายความว่าสามารถรับประทานอาหารได้ไม่เกิน 1373 แคลอรีต่อวัน โดยสามารถนำมาบวกจากค่าแคลอรีที่เบิร์นในรูปของการออกกำลังกายได้ ตัวอย่างเช่นว่ายน้ำท่าฟรีสไตล์ต่อการเผาผลาญต่อครึ่งชั่วโมงเท่ากับ 206 แคลอรี สามารถนำมาบวกกับค่าแคลอรีที่เราสามารถรับประทานได้จะได้เป็น 1509 แคลอรีต่อวัน หากมีกิจกรรมที่ต้องใช้พลังงาน หรือออกกำลังกายเพิ่มเติม ร่างกายก็จะมีการเผาผลาญพลังงานเพื่อนำมาใช้มากขึ้น จึงสามารถรับประทานอาหารได้มากขึ้นโดยไม่กระทบกับน้ำหนัก

จากสูตรหาค่า BMI สามารถแปลงหาน้ำหนักส่วนเกินได้ดังนี้

$$\text{หาค่าน้ำหนักส่วนเกิน} = (\text{BMI} - 22.9) \times \text{ส่วนสูงเป็นเมตรยกกำลัง 2}$$

ตัวอย่างเช่น น้ำหนักตัว 90 กก. ส่วนสูง 180 จะได้ค่า BMI ประมาณ 27.77 แทนค่าในสูตร

$$\text{หาค่าน้ำหนักส่วนเกิน} = (27.77 - 22.9) \times 1.8^2$$

เพราะฉะนั้นน้ำหนักไขมันส่วนเกินอยู่ที่ 15.8 กิโลกรัม

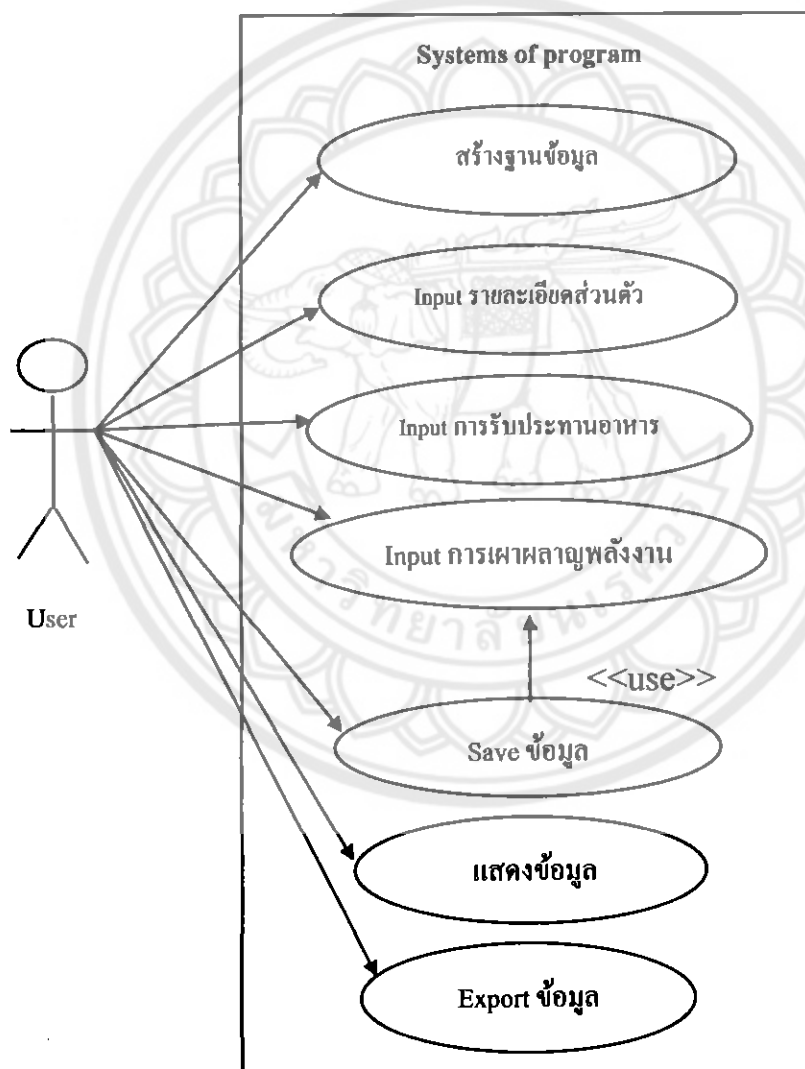
$$\text{คิดเป็นแคลอรีได้ } 15.8 \text{ (Kg)} \times 7700 \text{ (Kcal)} = 121688.86 \text{ กิโลแคลอรี}$$

บทที่ 3

วิธีการดำเนินงาน

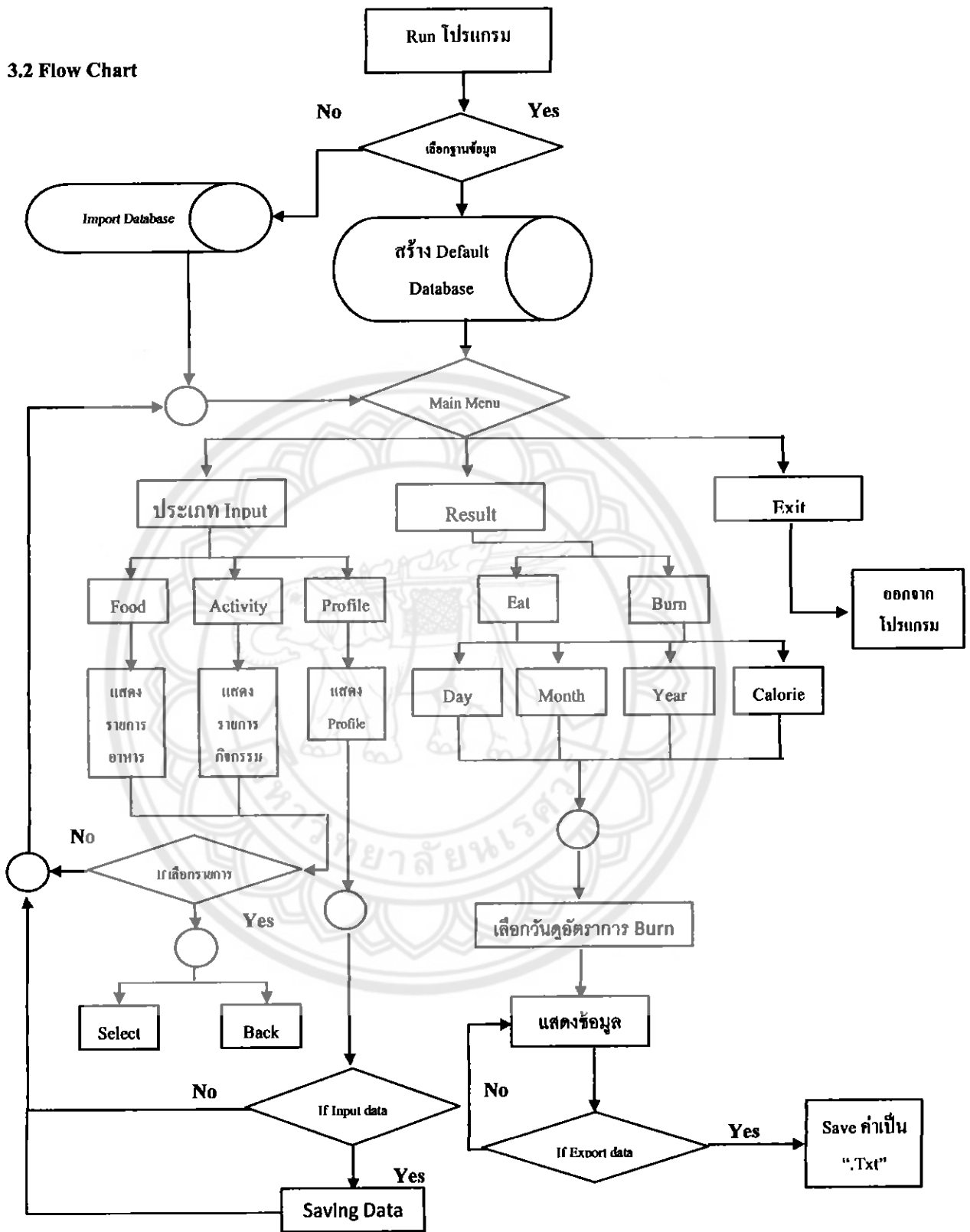
ในการสร้างแอปพลิเคชันคำนวณค่าโภชนาการของอาหารบนมือถือระบบปฏิบัติการแอนดรอยด์ จะต้องพิจารณาใน 2 ส่วนใหญ่ๆของโปรแกรมคือ ส่วนในการรับ Input ข้อมูลจากผู้ใช้ และส่วนที่เป็น Output หรือส่วนที่ใช้แสดงผลของข้อมูล ซึ่งภาพรวมของการทำงานจะเป็นดังนี้

3.1 Use Case Diagram



รูปที่ 3.1 แสดง Use case Diagram ของโปรแกรม

3.2 Flow Chart



รูปที่ 3.2 แสดง Flow Chart ของโปรแกรม

3.3 รายละเอียดของการรับข้อมูลจากผู้ใช้

ในการรับข้อมูลในจากผู้ใช้ในโปรแกรม A GOOD HEALTH นั้นจะมีรูปแบบในการรับข้อมูลดังต่อไปนี้

1. ข้อมูลพื้นฐานส่วนตัว (Profile)

- ส่วนสูง (Height)
- น้ำหนัก (Weight)
- อายุ (Age)
- น้ำหนักเป้าหมาย (Target weight)
- เพศ (Gender)
- ปริมาณน้ำหนักที่ต้องการจะลดต่อสัปดาห์ (Goal Lose Kg / Week)

2. ค่าแคลอรีอาหาร (Food)

- อาหารจานด่วน (Food)
- เครื่องดื่ม (Drink)
- ผลไม้ (Fruit)
- ของหวาน (Candy)
- ซูชิ (Sushi)
- เคเอฟซี (KFC)
- แมคโดนัลด์ (McDonald)
- เซเว่นอีเลฟเว่น (7-Eleven)
- พิซซ่าฮัท (Pizza Hut)

3. ค่าการเบิร์นแคลอรี (Activity)

- กิจกรรมการออกกำลังกาย (Activity)

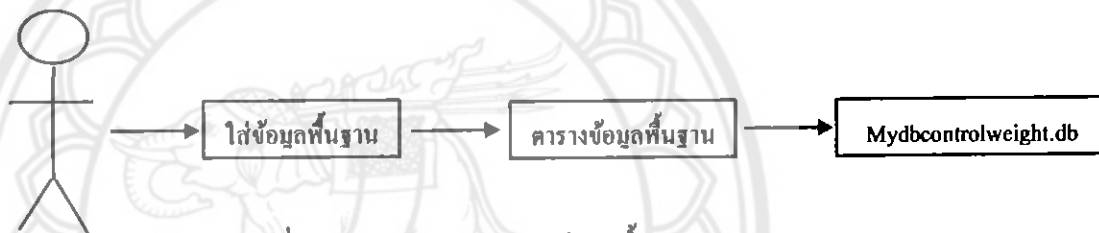
3.4 การทำงานของโปรแกรม

1. หน้ารับข้อมูลพื้นฐาน (Input)

ใส่ข้อมูลพื้นฐาน

- น้ำหนัก
- ส่วนสูง
- อายุ
- น้ำหนักเป้าหมาย
- เพศ
- เป้าหมายลดน้ำหนัก

System Diagram



รูปที่ 3.3 System Diagram รับข้อมูลพื้นฐาน

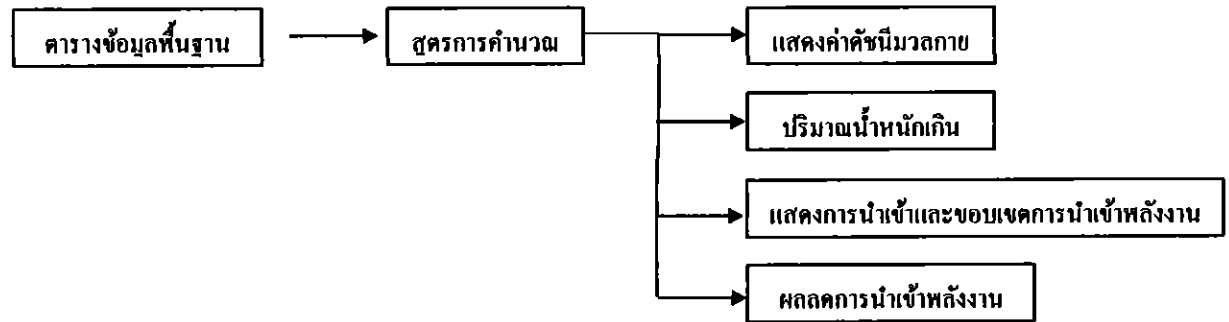
จากรูปที่ 3.3 จะแสดงให้เห็นถึงการนำเข้าสู่ข้อมูลจากผู้ใช้โปรแกรม โดยรับค่าจะผู้ใช้ผ่านหน้ารับข้อมูล จากนั้นนำข้อมูลที่ได้ไปบันทึกไว้ในตารางข้อมูลพื้นฐาน แล้วบันทึกเข้าฐานข้อมูลเพื่อนำไปใช้งานต่อไป

2. หน้าหลักของโปรแกรม (Main Page)

หน้าหลักโปรแกรม

- ค่าดัชนีมวลกาย
- ปริมาณน้ำหนักที่เกิน
- แสดงการนำเข้าและขอบเขตการนำเข้าพลังงาน
- ผลลดการนำเข้าพลังงาน

System Diagram



รูปที่ 3.4 System Diagram หน้าหลักของโปรแกรม

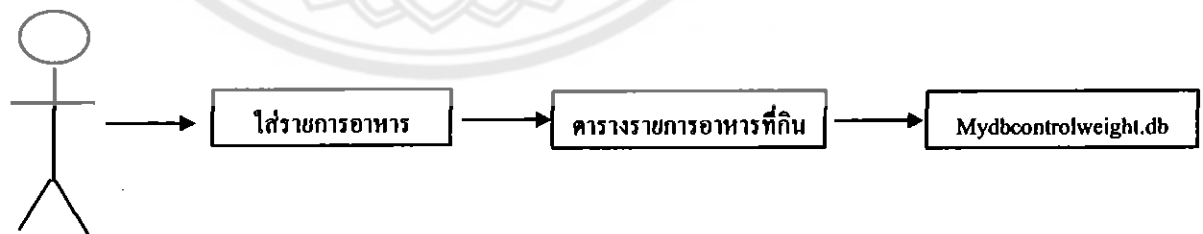
จากรูปที่ 3.4 จะแสดงให้เห็นถึงการนำเข้าสู่ข้อมูลจากตารางข้อมูลพื้นฐานมาใช้ในการคำนวณหาค่าต่างๆภายในโปรแกรม เช่น ค่าดัชนีมวลกาย, ปริมาณน้ำหนักที่เกิน, แสดงการนำเข้าและขอบเขตการนำเข้าพลังงาน

3. การทำงานของโปรแกรม รายการอาหาร (Food)

หน้ารายการอาหาร

- รายการอาหาร
- ตารางแคลอรี
- การเปรียบเทียบพลังงาน
- การสุ่มรายการอาหาร

System Diagram



รูปที่ 3.5 System Diagram รับค่าอาหาร

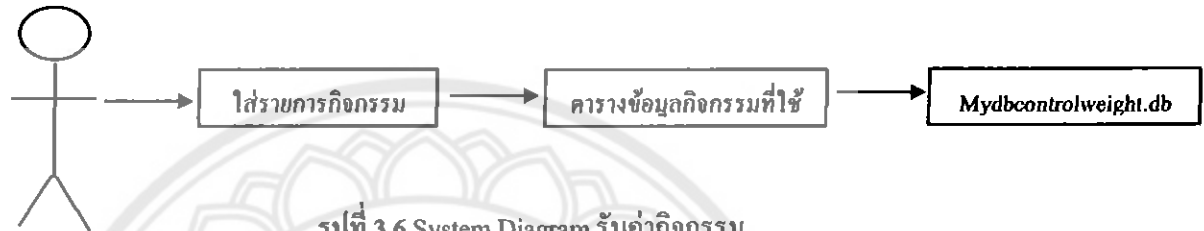
จากรูปที่ 3.5 จะแสดงให้เห็นถึงการเลือกข้อมูลการนำเข้าพลังงานของผู้ใช้ หลังจากเลือกเสร็จ จะทำการบันทึกไว้ในตารางรายการอาหารที่กิน แล้วบันทึกลงฐานข้อมูลเพื่อนำไปใช้งานต่อไป

4. การทำงานของปุ่ม กิจกรรม (Activity)

หน้ารายการกิจกรรม

- รายการกิจกรรม
- ตารางแคลอรี

System Diagram



รูปที่ 3.6 System Diagram รับค่ากิจกรรม

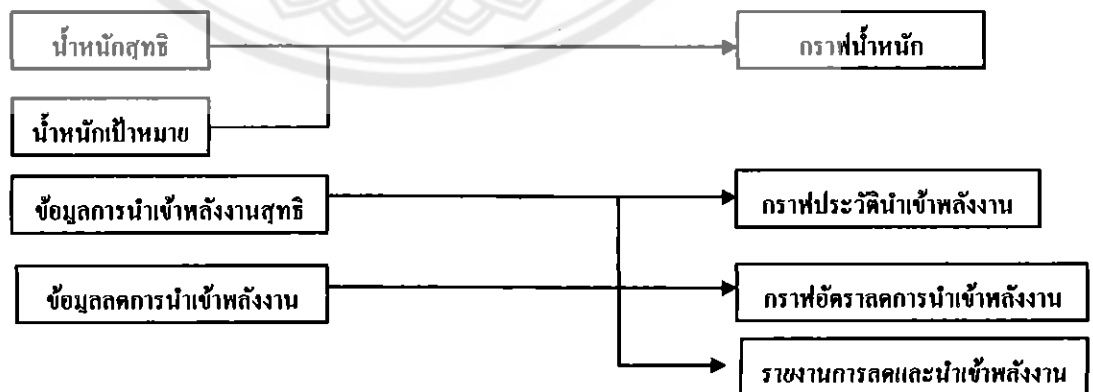
จากรูปที่ 3.6 จะแสดงให้เห็นถึงการเลือกข้อมูลการกิจกรรมการเผาผลาญพลังของผู้ใช้ หลังจากเลือกเสร็จจะทำการบันทึกไว้ในตารางกิจกรรมที่ใช้ แล้วบันทึกลงฐานข้อมูลเพื่อนำไปใช้งานต่อไป

5. การทำงานของปุ่ม รายงานผล (Result)

หน้ารายงานผล

- กราฟน้ำหนัก
- กราฟอัตราการนำเข้าพลังงาน
- กราฟประวัตินำเข้าพลังงาน
- แสดงรายละเอียด

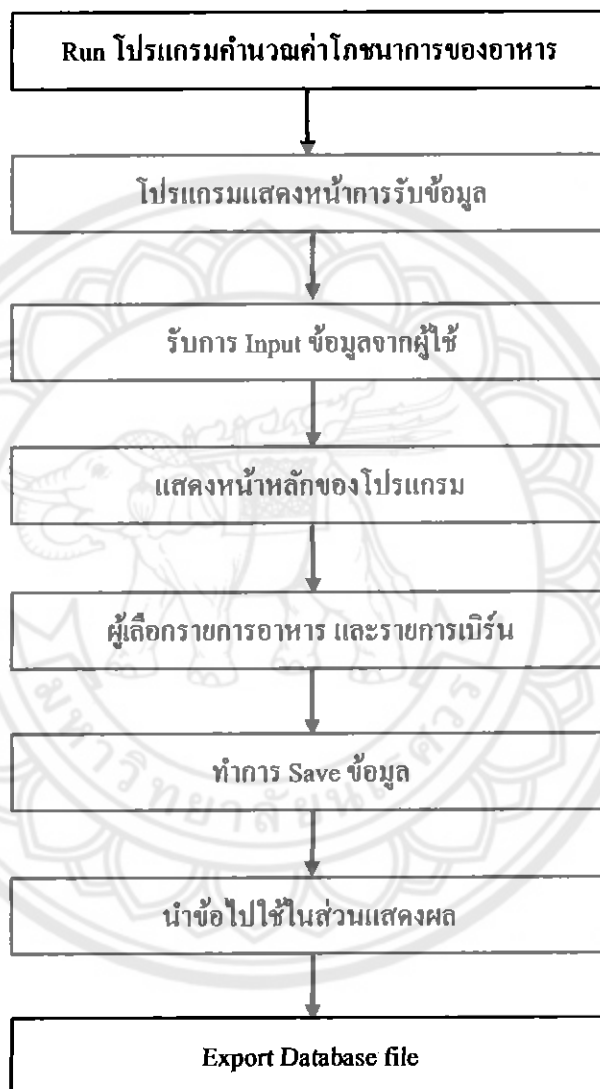
System Diagram



รูปที่ 3.7 System Diagram หน้ารายงานผล

จากรูปที่ 3.7 จะแสดงผลจากการรับค่าข้อมูลและค่าที่คำนวณภายในโปรแกรมเพื่อนำมาแสดงผลทางสถิติ ออกมาในรูปแบบของกราฟเส้นแสดงการเปรียบเทียบในรูปแบบต่างๆ

6. แสดงการทำงานของโปรแกรม



รูปที่ 3.8 แสดงขั้นตอนการทำงานของโปรแกรม

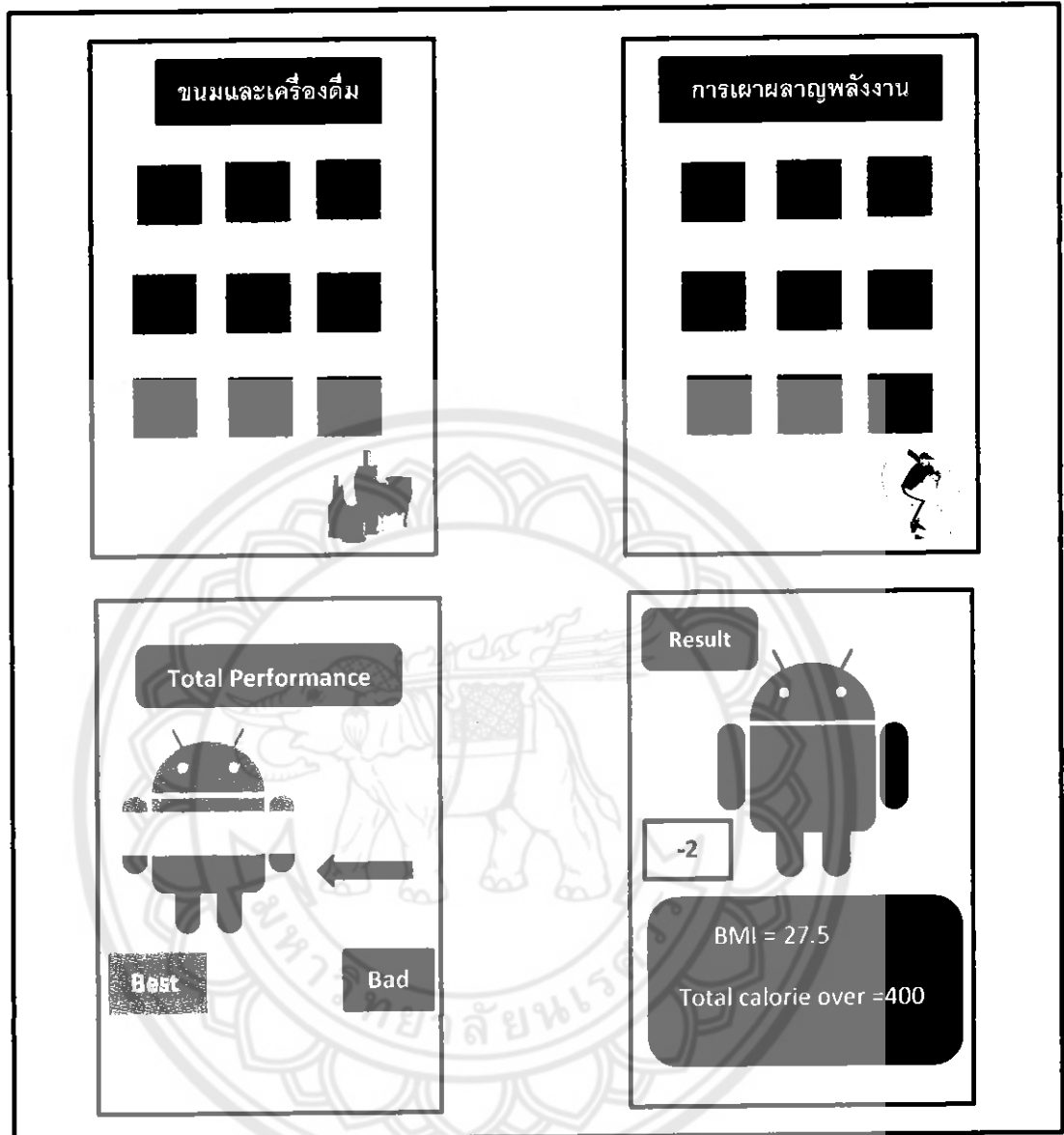
3.5 การออกแบบหน้าตาโปรแกรม

ตัวอย่าง การออกแบบระบบของโปรแกรมเบื้องต้น

The figure displays four distinct user interface screens for a program, arranged in a 2x2 grid. Each screen is enclosed in a rectangular frame and features a title bar at the top and an 'OK' button at the bottom right.

- Top-left screen:** Titled 'ใส่ค่าเพื่อคำนวณ' (Enter value for calculation). It contains two input fields: 'น้ำหนัก' (Weight) and 'ส่วนสูง' (Height), each with a small square icon below it.
- Top-right screen:** Titled 'ใส่ค่าเพื่อคำนวณ' (Enter value for calculation). It contains two input fields: 'เพศ' (Gender) and 'อายุ' (Age), each with a small square icon below it.
- Bottom-left screen:** Titled 'อาหารจานเดียว' (One-dish food). It features a 3x3 grid of nine square buttons and a small icon of a plate with a slice of pizza at the bottom center.
- Bottom-right screen:** Titled 'อาหารชุด' (Set meal). It features a 3x3 grid of nine square buttons and a small icon of a plate with a fork and knife at the bottom center.

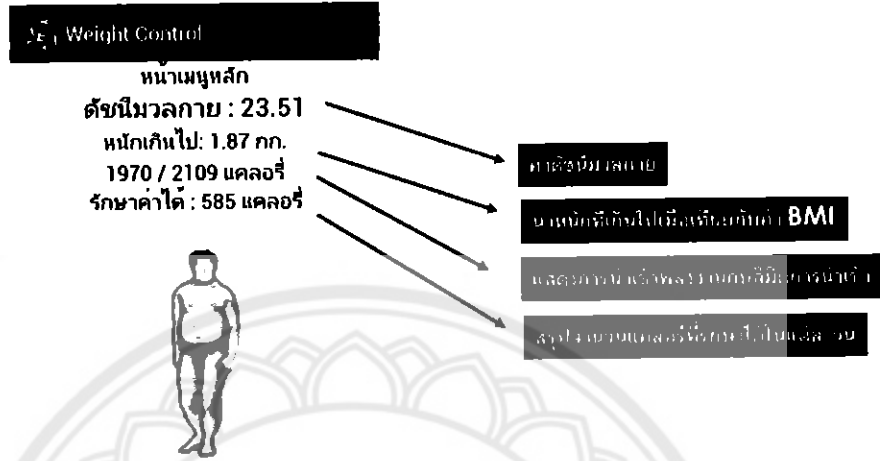
รูปที่ 3.9 ออกแบบหน้าตาโปรแกรม



รูปที่ 3.10 ออกแบบหน้าตาโปรแกรม

3.6 ออกแบบระบบการทำงานของโปรแกรม

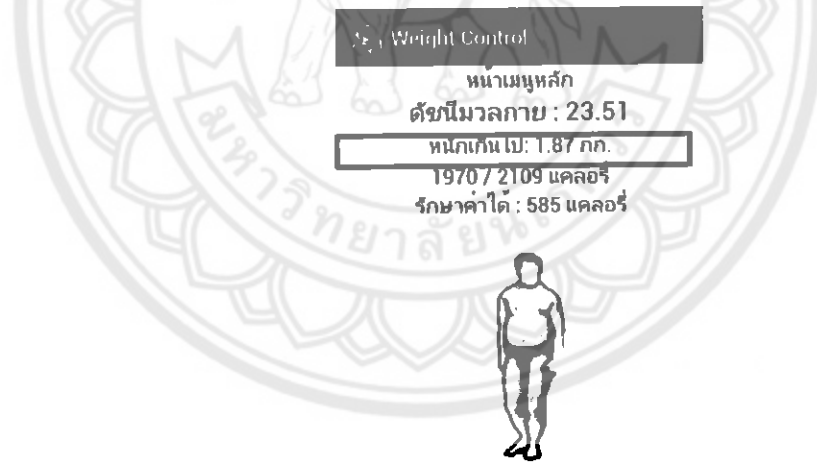
1. หน้าหลัก (Main Page)



เพิ่มน้ำหนัก

รูปที่ 3.11 ออกแบบระบบการทำงานของ โปรแกรม

2. แสดงผล จำนวนน้ำหนักที่กิน



เพิ่มน้ำหนัก

รูปที่ 3.12 ออกแบบระบบการทำงานของ โปรแกรม

Pseudo code

Double น้ำหนัก, ส่วนสูง, BMI, BMI ปกติ, Over BMI;

BMI = น้ำหนัก / ส่วนสูงเป็นเมตรยกกำลัง 2

BMI ปกติ = 22.9

Over BMI = (BMI - BMI ปกติ) x (ส่วนสูงเป็นเมตรยกกำลัง 2)

Print out Over BMI, "Kg";

3. แสดงผลค่าแคลอรีที่ควรได้รับต่อวัน เปรียบเทียบกับค่าความเป็นจริง



รูปที่ 3.13 ออกแบบระบบการทำงานของ โปรแกรม

Pseudo code

Double A,B,Save;

//A = ค่าแคลอรีที่ควรได้รับต่อวัน

//B = ค่าแคลอรีที่นำเข้า

A = 66 + (13.7 x น้ำหนัก) + (5 x ส่วนสูง) - (6.8 x อายุ) x 1.2;

Save = A - B;

Print out Save, "Cal", "/",A;

4. การแสดงรูปภาพสถานะน้ำหนัก

Weight Control

หน้าเมนูหลัก

ดัชนีมวลกาย : 23.51

หนักเกินไป: 1.87 กก.

1970 / 2109 แคลอรี

รักษาค่าได้ : 585 แคลอรี



เพิ่มน้ำหนัก

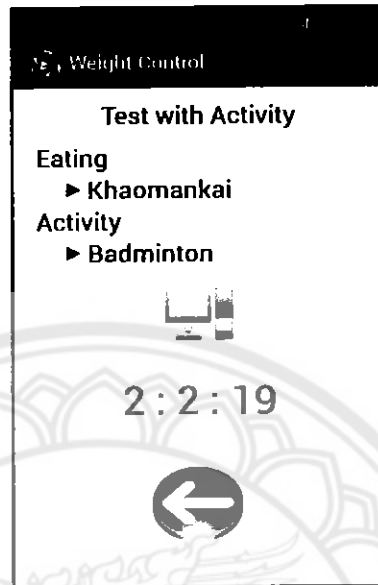
รูปที่ 3.14 ออกแบบระบบการทำงานของ โปรแกรม

Pseudo code

```

if(BMI<18.5){
    โชว์รูปภาพที่ 1;
}else if(BMI>=18.5 && BMI<=24.9){
    โชว์รูปภาพที่ 2;
}else if(BMI>=25 && BMI<=29.9){
    โชว์รูปภาพที่ 3;
}else if(BMI>=30 && BMI<=39.9){
    โชว์รูปภาพที่ 4;
}else if(BMI>=40 && BMI<=60){
    โชว์รูปภาพที่ 5;
}
  
```


5. การใช้งานปุ่ม เปรียบเทียบการเผาผลาญพลังงาน (Compare)



รูปที่ 3.15 ออกแบบระบบการทำงานของโปรแกรม

pseudo code

```
double food, Activity; int cod; int timeH = 0,timeM = 0,timeS=0;
cod = นำค่าแคลอรีของfood x ค่าแคลอรีของ Activity เป็นวินาที
timeH= (int)cod/60/60;
timeM= ((int)cod/60)%60;
timeS= (int)cod%60;
print out timeH,timeM,timeS;
```

6. การใช้งานปุ่ม ใ้ค่าการใช้พลังงานแบบกรอก (The system time)



รูปที่ 3.16 ออกแบบระบบการทำงานของโปรแกรม

pseudo code

A=เลือกกิจกรรมการเผาผลาญ

B=ใส่เวลาที่ใช้นาฬิกา

C=เวลาที่ใช้นาที

Double A,B,C;Cal

If (กรูปรูปภาพที่ 1)

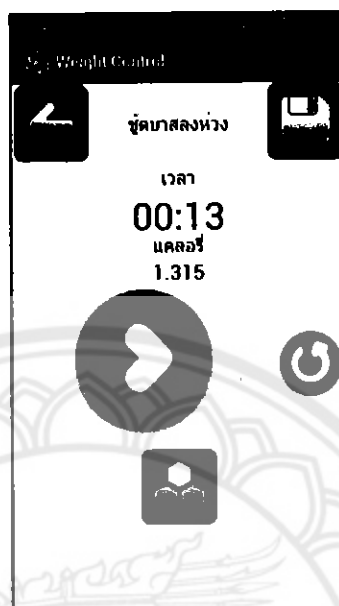
{ Cal= A x ((B/60)+C)

Print out Cal "แคลอรี";

Else (กรูปรูปภาพที่ 2){

End }}

7. การใช้งานปุ่ม ใส่ค่าการใช้พลังงานแบบจับเวลา (Timer)



รูปที่ 3.17 ออกแบบระบบการทำงานของโปรแกรม

pseudo code

double A,B,C,StopAndCal;

A=เลือกกิจกรรมการเผาผลาญ

B=ใส่เวลาที่ใช้น้อยนาที

C=เวลาที่ใช้น้อยวินาที

If (จากรูปภาพที่ 1) {

 เพื่อใช้จับเวลา

Else (จากรูปภาพที่ 2) {

 StopAndCal = A x ((B/60)+C)

}}

8. การใช้งานปุ่ม ข้อมูลส่วนตัว (Profile)

Weight Control

รายละเอียด



ความสูง : 175 น้ำหนัก : 72
 อายุ : 25 เพศ : ชาย
 จะลดน้ำหนัก : 0.6 กก.
 สถานะสุขภาพ : หนักเกินปกติ
 เมทบอลรายสัปดาห์ : 2114 แคลอรี
 ควรจะลดค่าแคลอรีต่อวัน
 : 660.0 Calorie



รูปที่ 3.18 ออกแบบระบบการทำงานของโปรแกรม

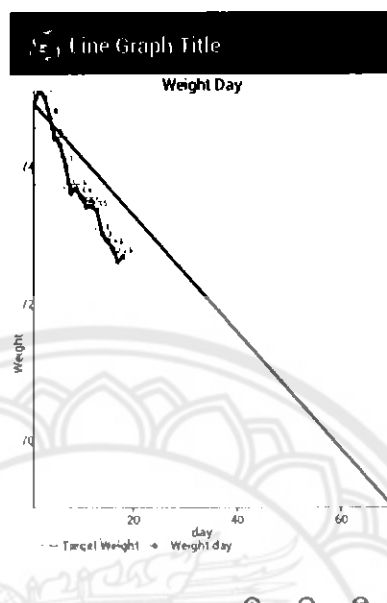
pseudo code

```

If (กตรูปภาพที่ 1) {
  เพื่อแก้ไขข้อมูลส่วนตัว
Else (กตรูปภาพที่ 2) {
  Share ข้อความ
}}

```

9. คู่อัตราการลดน้ำหนักเมื่อเทียบกับค่าเป้าหมาย



รูปที่ 3.19 ออกแบบระบบการทำงานของโปรแกรม

กราฟเส้นสีน้ำเงิน = น้ำหนักที่กรอกแต่ละวัน

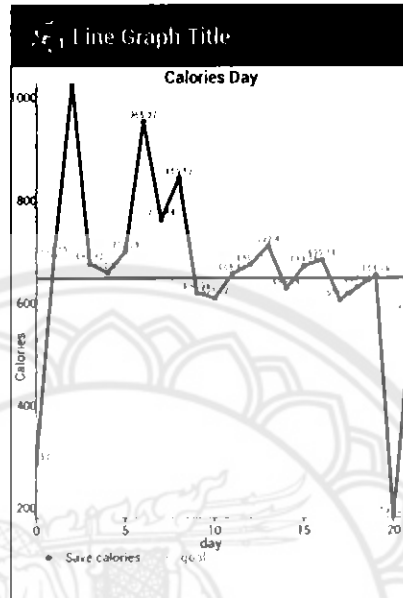
กราฟเส้นสีเขียว = ค่าน้ำหนักขั้นต่ำในการลด

ถ้าเส้นสีน้ำเงิน อยู่เหนือสีเขียว คือ อัตราการลดน้ำหนักน้อยลง

ถ้าเส้นสีน้ำเงิน อยู่ต่ำกว่าสีเขียว คือ อัตราการลดน้ำหนักเพิ่มขึ้น

ผลที่ได้จะแสดงให้เห็นว่าการจำกัดค่าแคลลอรี่ในแต่ละวันได้ผลกับการควบคุมน้ำหนักของผู้ใช้โปรแกรมหรือไม่

10. การเปรียบเทียบระหว่างค่าเป้าหมายกับค่าที่ทำได้ในแต่ละวัน



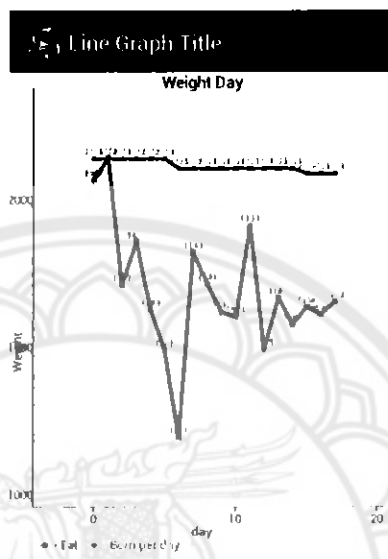
รูปที่ 3.20 ออกแบบระบบการทำงานของโปรแกรม

กราฟเส้นสีแดง = จำนวนเป้าหมายการลดการนำเข้าพลังงานในแต่ละวัน

กราฟเส้นสีน้ำเงิน = จำนวนการลดการนำเข้าพลังงานที่ทำได้จริง

ผลที่ได้จะใช้ทำการเปรียบเทียบดูว่าเรากินอยู่ในขอบเขตหรือไม่จะได้ปรับเมนูให้เข้ากับการควบคุมน้ำหนักมากยิ่งขึ้น

11.แนวโน้มการนำเข้าพลังงานในแต่ละวัน



รูปที่ 3.21 ออกแบบระบบการทำงานของโปรแกรม

กราฟเส้นสีแดง = จำนวนจำกัดการนำเข้าพลังงานในแต่ละวัน

กราฟเส้นสีเขียว = จำนวนการนำเข้าพลังงานที่ทำได้จริง

ถ้ากราฟเส้นสีเขียวอยู่เหนือเส้นแดง แสดงถึง ควบคุมเกินจำนวนจำกัด

ถ้ากราฟเส้นสีเขียวอยู่เสมอสันแดง ควบคุมพอดี

ถ้ากราฟเส้นสีเขียวอยู่ต่ำเส้นแดง ควบคุมการนำเข้าพลังงานได้

จากผลที่ได้ผู้ใช้จะสามารถดูแนวโน้มการนำเข้าพลังงานในแต่ละวันได้และจำกัดขอบเขตการนำเข้าพลังงานได้

บทที่ 4

ผลการทดลอง

4.1 หน้าหลัก (Main Page)

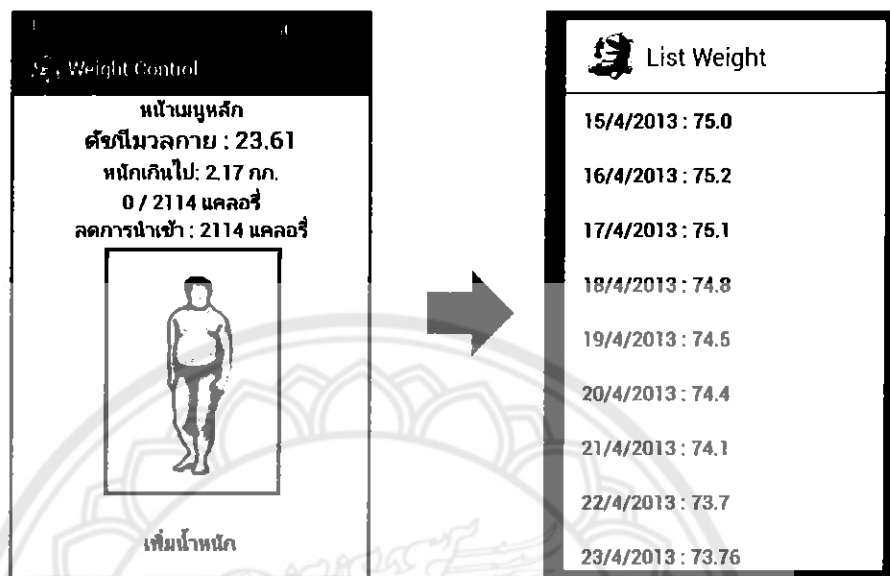
1. ทดลองการกรอกรายละเอียดข้อมูลเริ่มต้น และการแสดงผลของหน้าหลัก

The screenshot shows the 'Weight Control' app interface. On the left, there are input fields for 'น้ำหนัก' (Weight) set to 75, 'ความสูง' (Height) set to 175, 'อายุ' (Age) set to 25, and 'น้ำหนักเป้าหมาย' (Target Weight) set to 69. Below these are checkboxes for 'เพศ' (Gender) with 'ชาย' (Male) selected, and 'เป้าหมายที่ลดกิโลกรัมต่อสัปดาห์' (Target weight reduction per week) set to 0.5. There are also dropdown menus for 'สถานะการออกกำลังกาย' (Exercise status) set to 'ออกกำลังกายปานกลาง' (Moderate exercise) and 'ขกเล็ก' (Small portion). On the right, a summary box displays 'น้ำหนักเฉลี่ย' (Average weight) as 24.49, 'หนักเกินไป: 4.87 กก.' (Overweight: 4.87 kg), '0 / 2518 แคลอรี' (0 / 2518 calories), and 'ลดการนำเข้า: 2518 แคลอรี' (Reduce intake: 2518 calories). A silhouette of a person is shown with the text 'เพิ่มน้ำหนัก' (Gain weight). At the bottom right, there are icons for 'ช้อนและส้อม' (Fork and spoon), 'วิ่ง' (Run), 'รถเข็น' (Wheelchair), 'รายละเอียด' (Details), 'ไอ' (I), and 'วี' (V).

รูปที่ 4.1 แสดงผลหน้าหลัก

จากการทดสอบ โดยกรอกข้อมูล น้ำหนัก 72 กิโลกรัม ความสูง 175 เซนติเมตร อายุ 25 ปี น้ำหนักเป้าหมาย 69 กิโลกรัม เพศ ชาย เป้าหมายที่ลดกิโลกรัมต่อสัปดาห์ 0.5 กิโลกรัม สถานะการออกกำลังกายคือ ออกกำลังกายปานกลาง จากรูป 4.1 ด้านซ้าย แล้ว โปรแกรมก็จะทำการคำนวณผลที่ได้คือ คำนีวมวลเท่ากับ 24.49 , น้ำหนักที่เกินไป 4.87 กิโลกรัม ,แคลอรีที่เผาผลาญได้ในแต่ละวัน เท่ากับ 0 ต่อ 2518 แคลอรี , ลดการนำเข้าแคลอรีได้ 2518 แคลอรี หมายเหตุ ที่เป็น 0 เพราะยังไม่มีกรนำเข้าพลังงานจากอาหาร แสดงในรูปที่ 4.1 ด้านขวาในกรอบสีแดง ผลที่ได้ตรวจสอบแล้วมีค่าถูกต้อง

2. ทดลองการดูรายการน้ำหนักในแต่ละวันจากการคลิกที่ภาพ



รูปที่ 4.2 แสดงรายการน้ำหนัก

จากการทดลอง เมื่อคลิกที่ภาพในกรอบสีแดง จากรูป 4.2 ด้านซ้าย โปรแกรมก็จะทำการดึงข้อมูลน้ำหนักออกมา แสดงเป็นรายการน้ำหนักในแต่ละวัน ว่า วันที่/เดือนที่/ปีที่/ : น้ำหนัก เช่น

15/4/2013: 75.0 กิโลกรัม

16/4/2013: 75.2 กิโลกรัม

17/4/2013: 75.1 กิโลกรัม

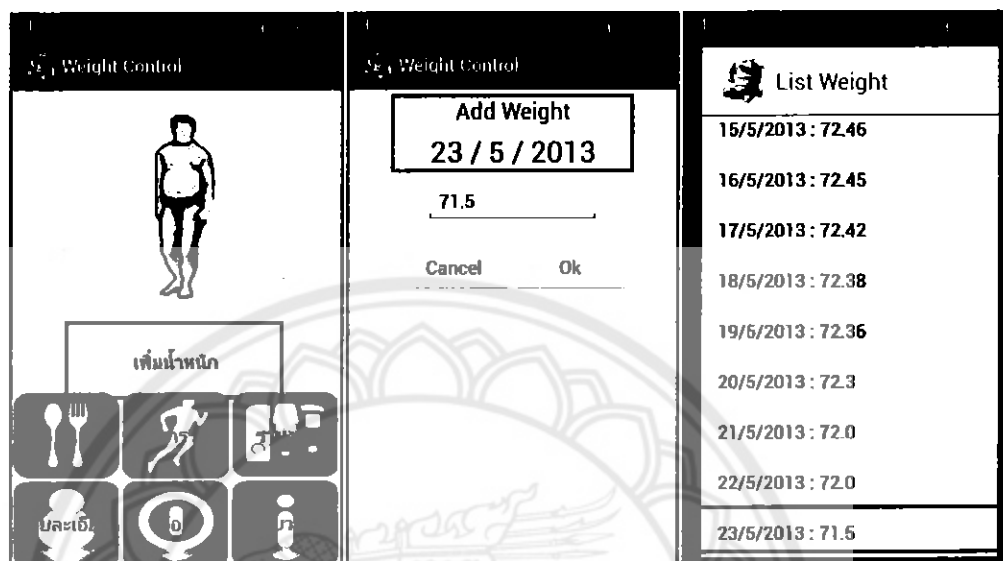
18/4/2013: 74.8 กิโลกรัม.

19/4/2013: 74.5 กิโลกรัม

20/4/2013: 74.4 กิโลกรัม

ผลที่ได้จากการทดลอง จะเป็นไปตามรูป 4.2 ด้านขวา เพื่อดูการเปลี่ยนแปลงของน้ำหนักในแต่ละวันตั้งแต่เริ่มต้นใช้โปรแกรมจนถึงข้อมูลวันล่าสุด โดยสามารถแสดงข้อมูลออกมาได้อย่างถูกต้อง

3.ทดสอบการอัปเดตน้ำหนักโดยคลิกที่ปุ่ม เพิ่มน้ำหนัก

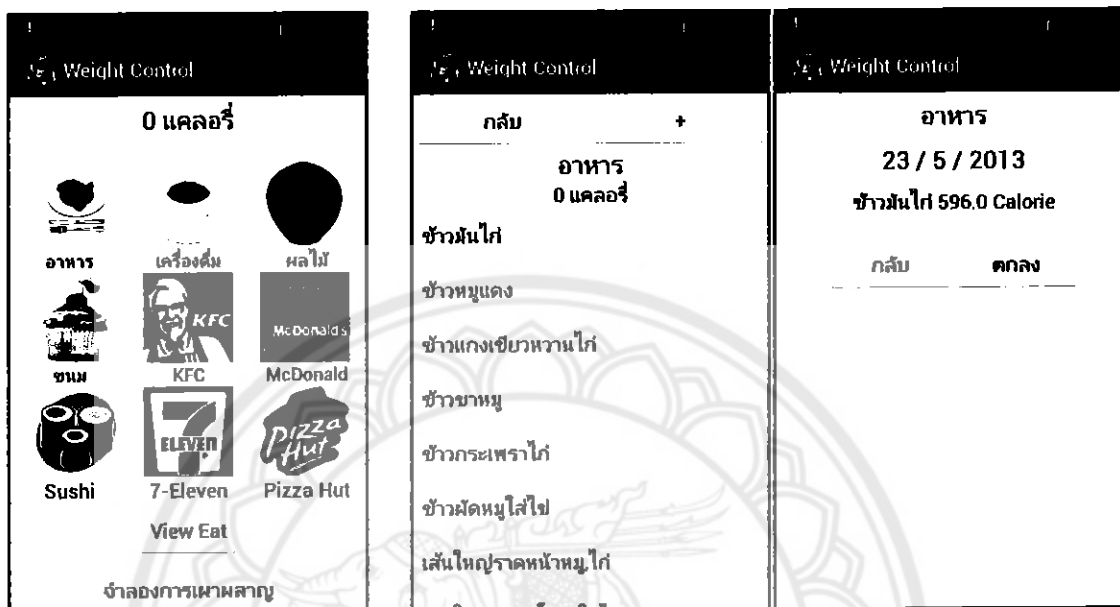


รูปที่ 4.3 การอัปเดตน้ำหนัก

จากทดสอบการอัปเดตน้ำหนักตามรูปที่ 4.3 ด้านซ้าย ให้คลิกที่ปุ่มเพิ่มน้ำหนัก แล้วรอกน้ำหนักลงไป 71.5 กิโลกรัม และใส่วันที่เป็น 23/5/2013 ตามรูปที่ 4.3 ตรงกลาง แล้วคลิกที่ปุ่ม Ok เพื่อทำการบันทึกข้อมูล ก็จะกลับสู่หน้าหลักและคลิกที่ภาพ ตามรูป 4.2 ด้านซ้าย เพื่อดูข้อมูลที่เรอัปเดตน้ำหนัก ไปจะได้ค่า 23/5/2013 : 71.5 กิโลกรัม ตามรูปที่ 4.3 ด้านขวาในกรอบสีแดง ผลที่ได้มีค่าถูกต้อง

4.2 การใช้งานปุ่ม การนำเข้าพลังงาน (Food)

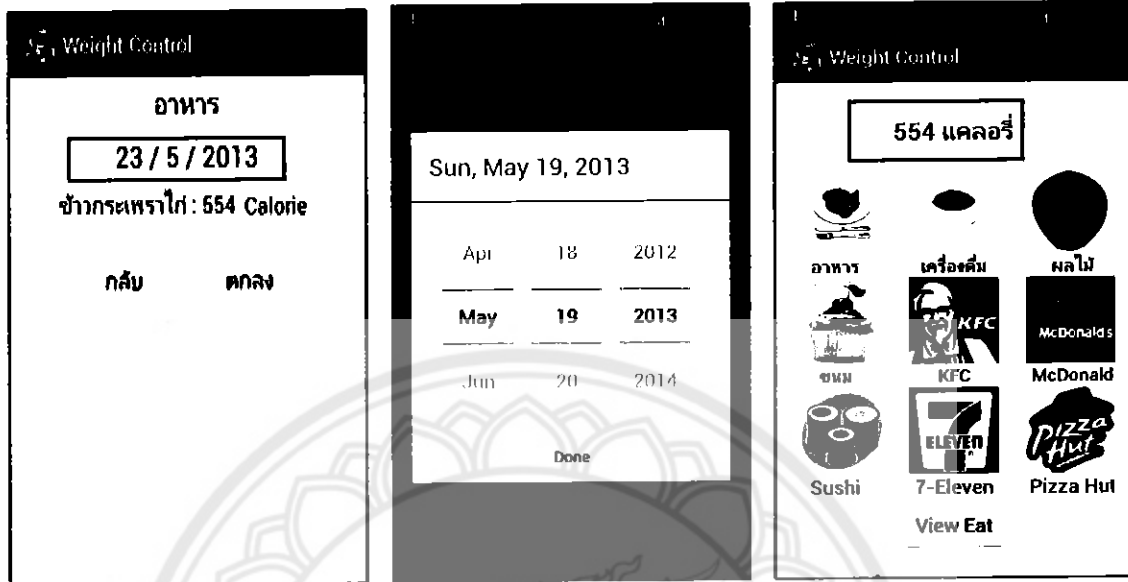
1. ทดสอบการนำเข้าพลังงานทั้งแบบรายวัน



รูปที่ 4.4 ทดสอบการนำเข้าพลังงาน

จากทดสอบการการนำเข้าพลังงานแบบรายวัน โดยทำการเลือกหมวดอาหารจากรูปที่ 4.4 ด้านซ้าย โปรแกรมก็จะทำการดึงข้อมูลอาหารออกมาแสดงในรูปที่ 4.4 ตรงกลาง ให้เลือกรายการอาหาร จากการทดสอบเลือกอาหารเป็น ข้าวมันไก่ ก็จะไปหน้าขึ้นชั้นข้อมูลซึ่งจะแสดงรายละเอียดดังนี้ หมวด "อาหาร", วันที่ "23/5/2013", ชื่ออาหาร "ข้าวมันไก่ 596.0 Calorie" ตามรูปที่ 4.4 ด้านขวา ผลที่ได้โปรแกรมสามารถแสดงผลได้อย่างถูกต้อง

2.ทดสอบการนำเข้าพลังงานแบบย้อนหลัง



รูปที่ 4.5 ทดสอบการนำเข้าพลังงาน

จากการทดสอบการนำเข้าพลังงานย้อนหลัง โดยทำการเลือกหมวดอาหารแล้วเลือกอาหารเป็น "ข้าวกระเพราไก่" จากรูปที่ 4.4 ตรงกลาง จากนั้นจะถูกนำมามาหน้าขึ้นขึ้น วัน เดือน ปี และรายการอาหารที่นำเข้าเป็น "ข้าวกระเพราไก่ : 554 Calorie" โดยสามารถแก้ไข วัน เดือน ปี ที่นำเข้าพลังงานได้โดยทำการคลิกที่ ตัวเลขวันที่ "23/5/2013" ในกรอบสีแดงของรูปที่ 4.5 ทางด้านซ้าย จากนั้นทำการเลือก วัน เดือน ปี เป็นวันที่ "May 19 2013" ในรูปที่ 4.5 กลาง ผลที่แสดงออกมาจะเป็นค่าแคลอรีอาหารที่นำเข้าไป คำนวณที่ 4.5 ด้านขวาในกรอบสีแดง ซึ่งผลที่ได้มีค่าถูกต้อง

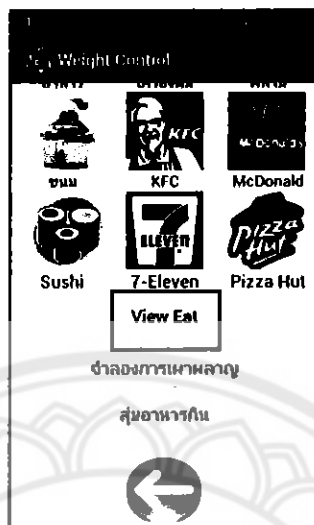
3.ทดสอบการใช้งานปุ่มการจำลองการเผาผลาญ



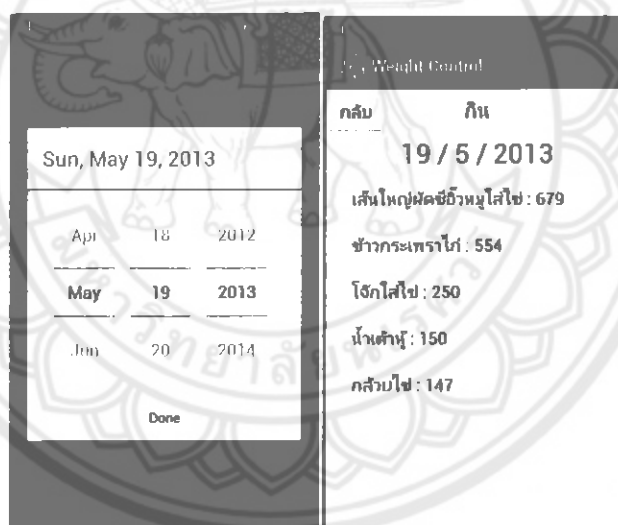
รูปที่ 4.6 ทดสอบการจำลองการเผาผลาญ

จากการทดสอบคลิกที่ปุ่ม จำลองการเผาผลาญกรอบสีแดงในรูปที่ 4.6 ด้านซ้าย จากเชื่อมต่อมายังหน้าโปรแกรม ในรูปที่ 4.6 ด้านขวาโดยทำการทดลองเลือกรายการอาหารเป็น ข้าวกระเพราไก่ เปรียบเทียบกับ ค่ากิจกรรมการเผาผลาญเป็น ชุดบาสลงห่วง ได้ผลการคำนวณออกมาเป็น 1:41:58 แสดงผลเป็น ชั่วโมง:นาที:วินาที ในกรอบสีแดง ซึ่งมีค่าถูกต้อง

4. ทดสอบปุ่ม View Eat เพื่อดูรายการอาหารที่นำเข้ามาในแต่ละวัน โดยคลิกที่ตัวเลขวันที่



รูปที่ 4.7 ทดสอบปุ่ม View Eat



รูปที่ 4.8 ทดสอบปุ่ม View Eat

ทำการทดสอบคลิกที่ปุ่ม View Eat ในกรอบสี่เหลี่ยมรูปที่ 4.7 ทำการคลิกที่ตัวเลขวัน เดือน ปี ที่แสดงเพื่อเลือกวันที่ "May 19 2013" เป็นวันที่ต้องการทราบข้อมูล ดังในรูปที่ 4.8 ซ้าย จากนั้น โปรแกรมจะไปยังหน้าแสดงผลรายการอาหารที่นำเข้ามาในวันที่ "19/5/2013" และแสดงข้อมูลต่อไปนี้เส้นใหญ่ผัดซีอิ้วหมูใส่ไข่ : 679, ข้าวกระเพราไก่ : 554, โจ๊กใส่ไข่ : 250, น้ำเต้าหู้ : 150, กล้วยไข่ : 147 ดังรูปที่ 4.8 ด้านขวา ผลที่ได้มีข้อมูลถูกต้อง

4.3 การใช้งานปุ่ม กิจกรรมการเผาผลาญพลังงาน (Activity)

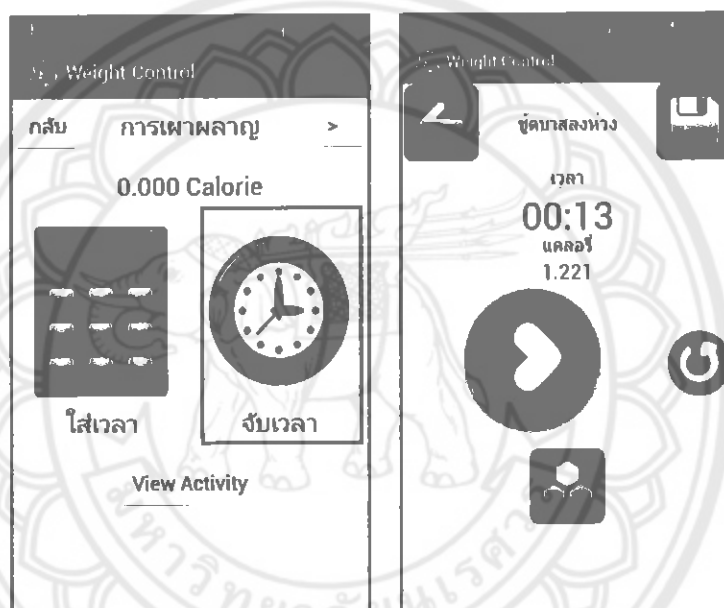
1. ทดสอบการใส่เวลาหาค่าแคลอรี



รูปที่ 4.9 ทดสอบการใส่เวลา

จากการทดสอบใส่เวลาหาค่าแคลอรีให้เลือกรุ่นใส่เวลา ตามรูปที่ 4.9 ด้านบนซ้าย แล้วก็เลือกกิจกรรมตามรูปที่ 4.9 ด้านบนขวา จากการทดสอบเลือกชุดบาสลงห่วง เลือกแล้วก็จะเข้าหน้าใส่เวลาตามรูปที่ 4.9 ล่างซ้าย ให้ใส่เวลาเป็น นาที และ วินาที จากการทดสอบในไป 120 นาที 10 วินาที แล้วเลือกรุ่นคำนวณ แล้วก็จะมีแสดงค่าแคลอรีที่เผาผลาญได้ออกมาตามรูปที่ 4.9 ล่างซ้าย แล้วเลือกรุ่น Save ก็จะไปหน้ารายงานเผาผลาญที่จะบันทึก ตามรูปที่ 4.9 ล่างขวา แล้วก็เลือกรุ่น “ตกลง” เพื่อทำการบันทึกค่า

2. ทดสอบการจับเวลาหาค่าแคลอรี



รูป 4.10 ทดสอบการจับเวลาหาค่าแคลอรี

จากการทดสอบจับเวลาหาค่าแคลอรีให้เลือกรุ่นจับเวลา ตามรูปที่ 4.10 ด้านบนซ้าย แล้วก็เลือกกิจกรรมตามรูปที่ 4.9 ด้านบนขวา จากการทดสอบเลือกชุดบาสลงห่วง เลือกแล้วก็จะเข้าหน้าจับเวลาตามรูปที่ 4.10 ด้านขวา ให้กดเริ่มจับเวลาไป ถ้ากดหยุด โปรแกรมก็จะทำการคำนวณค่าแคลอรีที่เผาผลาญได้ จากการทดสอบกดเริ่มไป 13 วินาที แล้วถึงกดหยุด ค่าแคลอรีที่เผาผลาญได้ คือ 1.221 แคลอรี แล้วเลือกรุ่น Save ก็จะไปหน้ารายงานเผาผลาญที่จะบันทึก ตามรูปที่ 4.9 ล่างขวา แล้วก็เลือกรุ่น “ตกลง” เพื่อทำการบันทึกค่า

3. ทดสอบการ Share ผ่าน Google Plus



รูป 4.11 ทดสอบการ Share ผ่าน Google Plus

จากการทดสอบการ Share ผ่าน Google Plus ให้เลือกปุ่มตามรูปที่ 4.11 ด้านซ้าย แล้วก็เลือกที่จะ Share ไปที่ไหนตามรูปที่ 4.11 ตรงกลาง จากการทดสอบเลือก Share ผ่าน Google Plus เมื่อเลือกแล้วก็จะไปหน้าที่ Share ข้อมูล Google Plus จะปรากฏข้อความ “แบ่งปัน กิจกรรม ชั่งน้ำหนักห้วง เวลา 13.0 เหน้าถุ 1.221 จาก App Weight Control”ตามรูปที่ 4.11 ด้านขวา

4.4 การใช้งานปุ่ม ข้อมูลส่วนตัว (Profile)

1. ผลการแสดงผลข้อมูลส่วนตัว




รูปที่ 4.12 ผลการแสดงผลข้อมูลส่วนตัว

จากการทดสอบการใช้งานปุ่ม ข้อมูลส่วนตัว หลังจากทำการคลิกที่ปุ่มแล้วโปรแกรมจะนำมามาหน้าแสดงรายละเอียดของมุลส่วนตัวดังรูปที่ 4.12 ซึ่งผลที่ได้จะแสดงเป็น ความสูง 175 เซนติเมตร , น้ำหนัก 72 กิโลกรัม, อายุ 25 ปี, เพศชาย, จะลดน้ำหนัก 0.6 กิโลกรัมต่อสัปดาห์, สถานะสุขภาพ หนักเกินปกติ, เผาผลาญต่อวัน 2114 แคลอรี, ควรจะลดค่าแคลอรีต่อวัน 660 แคลอรีต่อวัน จากค่าที่แสดงออกมามีผลของข้อมูลที่ถูกต้อง

2. ทดสอบการแก้ไขข้อมูลส่วนตัว



Weight Control		Weight Control	
น้ำหนัก	ความสูง	รายละเอียด	
90	180		
อายุ			
23			
น้ำหนักเป้าหมาย			
75			
เพศ		ความสูง : 180 น้ำหนัก : 90 อายุ : 23 เพศ : ชาย จะลดน้ำหนัก : 0.5 กก. สถานะสุขภาพ : ย้วน เผาผลาญ ต่อวัน : 2860 แคลอรี ควรจะลดค่าแคลอรีต่อวัน : 550.0 Calorie	
<input checked="" type="checkbox"/> ชาย	<input type="checkbox"/> หญิง		
สถานะการออกกำลังกาย			
0.5			
Stats Exercise			
ออกกำลังกายปานกลาง			
ยกเลิก	ตกลง		

รูปที่ 4.13 ทดสอบการแก้ไขข้อมูล

จากการทำการทดสอบปุ่มแก้ไขข้อมูลส่วนตัวดังรูปที่ 4.13 ด้านซ้าย สามารถเรียกขึ้นมาใช้เพื่อรับค่าข้อมูลได้เป็นปกติ จากนั้นทำการกรอกข้อมูลเพื่อทดสอบการใช้งานโดยกำหนดให้ ความสูง 180 เซนติเมตร , น้ำหนัก 90 กิโลกรัม, อายุ 23 ปี, เพศ ชาย, จะลดน้ำหนัก 0.5 กิโลกรัมต่อสัปดาห์, สถานะสุขภาพ ย้วน, เผาผลาญต่อวัน 2860 แคลอรี, ควรจะลดค่าแคลอรีต่อวัน 550 แคลอรีต่อวัน ผลปรากฏว่ามีการแก้ไขข้อมูลได้อย่างถูกต้องดังรูปที่ 4.3 ด้านขวา

4.5 การใช้งานปุ่ม รายงานผล (Result)

1. ทดลองการแสดงผลเลือกดูทุกวัน

ประวัติและรายงานผล			กลับ	
กราฟ	นำออก	กลับ	กลับ	ทุกวัน
เลือกวัน		ดูทุกวัน		
23 / 5 / 2013				
กิน	เผาผลาญ			
<div style="border: 1px solid black; padding: 2px;"> วันที่เลือก : 23/5/2013 เวลาเลือก : 12:00:00 เวลาปิดโปรแกรม : 12:00:00 </div>				
			15/4/2013	
			-> การกิน	:2086
			-> เผาผลาญ	:227
			-> ลดการนำเข้า	:300
			16/4/2013	
			-> การกิน	:2161
			-> เผาผลาญ	:707
			-> ลดการนำเข้า	:705
			17/4/2013	
			-> การกิน	:1732
			-> เผาผลาญ	:614
			-> ลดการนำเข้า	:1040
			18/4/2013	
			-> การกิน	:1877
			-> เผาผลาญ	:407
			-> ลดการนำเข้า	:688
			19/4/2013	
			-> การกิน	:1649
			-> เผาผลาญ	:162

รูปที่ 4.14 ทดสอบการแสดงผลทุกวัน

จากผลการทดสอบปุ่มการรายงานผลรายละเอียดการกิน การเผาผลาญ การลดการนำเข้า แบบดูทุกวันจากการคลิกที่ปุ่ม ดูทุกวันเพื่อตรวจสอบในรูปที่ 4.14 ด้านซ้าย จากนั้น โปรแกรมจะทำการแสดงข้อมูล รายละเอียดการกิน การเผาผลาญ การลดการนำเข้าพลังในแต่ละวัน เช่น

15/4/2013

->การกิน :2086

->เผาผลาญ :227

->ลดการนำเข้า :300

16/4/2013

->การกิน :2161

->เผาผลาญ :707

->ลดการนำเข้า :705

17/4/2013

->การกิน :1732

->เผาผลาญ :614

->ลดการนำเข้า :1040

18/4/2013

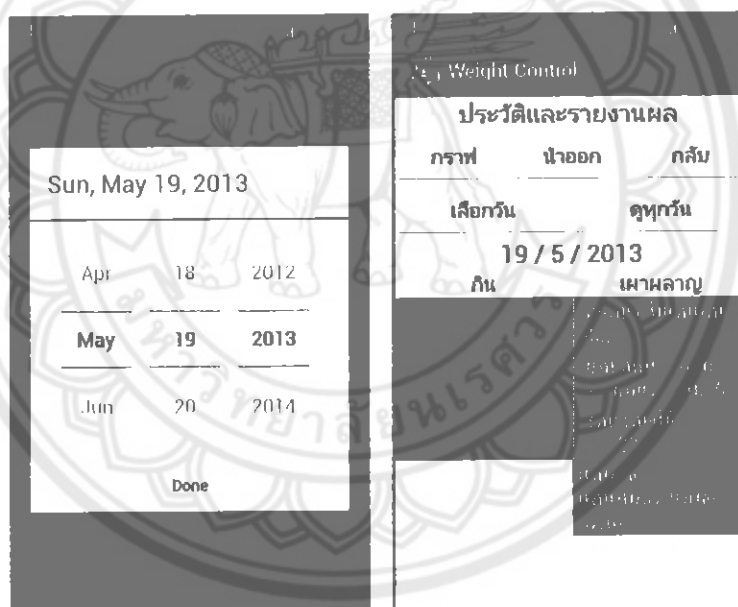
->การกิน :1877

->เผาผลาญ :407

->ลดการนำเข้า :688

ผังรูปที่ 4.14 ด้านขวา ผลการทดสอบข้อมูลที่แสดงผลออกมามีความถูกต้อง ครบถ้วน

2. ทดลองการแสดงผลเลือกวัน



รูปที่ 4.15 ทดสอบการแสดงผลแบบเลือกวัน

ทดลองการแสดงผลเลือกวันให้เลือกปุ่มเลือกวันตามรูปที่ 4.15 ด้านขวา แล้วก็จะมึหน้าให้เลือก เดือน/วัน/ปี ตามรูปที่ 4.15 ด้านซ้าย จากการทดลองการแสดงผลเลือกวันเลือก วันที่ 19 เดือน 5 ปี 2013 แล้วเลือกปุ่ม Done ก็จึแสดงผลตามรูปที่ 4.15 ด้านขวา เป็นการแสดงค่าแคลอรีการกิน และ เผาผลาญ เช่น

วันที่ 19/5/2013

กิน

ผลไม้ 147 กิโลแคลอรี

อาหาร 1483 กิโลแคลอรี

เครื่องดื่ม 150 กิโลแคลอรี

เผาผลาญ

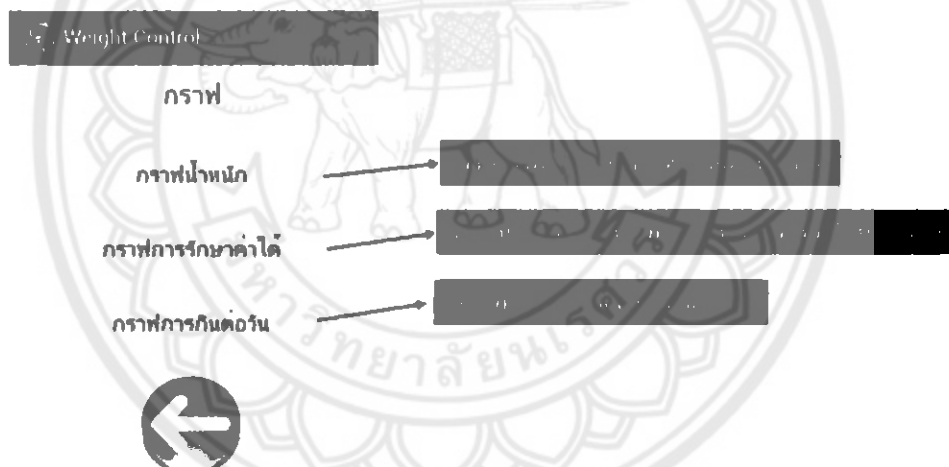
พาสุนัขไปเดินเล่น 43.2 กิโลแคลอรี

ขกน้ำหนัก 16 to 24 ปอนด์ 110.75 กิโลแคลอรี

รดน้ำต้นไม้ 21.722 กิโลแคลอรี

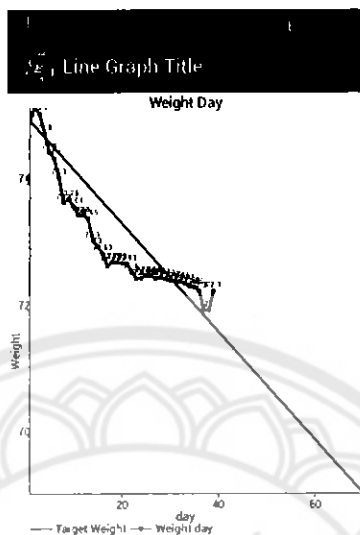
เดินวิ่งกับสัตว์เลี้ยง 57.6 กิโลแคลอรี

3. ทดสอบการแสดงผลของกราฟ



รูปที่ 4.16 รูปแบบการแสดงผลของกราฟ

1.ทดสอบการแสดงผลกราฟน้ำหนัก



รูปที่ 4.17 กราฟน้ำหนัก

กราฟเส้นสีน้ำเงิน = น้ำหนักที่กรอกแต่ละวัน

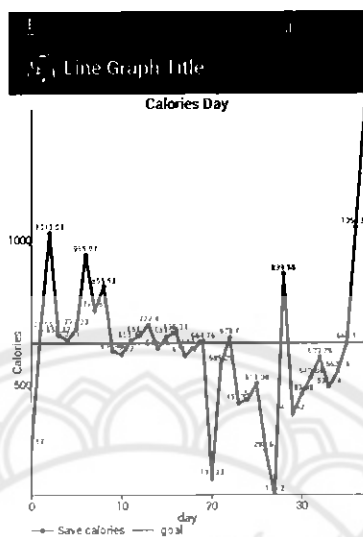
กราฟเส้นสีเขียว = คำนวณน้ำหนักขึ้นต่ำในการลด

ถ้าเส้นสีน้ำเงิน อยู่เหนือสีเขียว คือ อัตราการลดน้ำหนักน้อยลง

ถ้าเส้นสีน้ำเงิน อยู่ต่ำกว่าสีเขียว คือ อัตราการลดน้ำหนักเพิ่มขึ้น

ผลที่ได้จะแสดงให้เห็นว่าการจำกัดค่าแคลลอรี่ในแต่ละวันได้ผลกับการควบคุมน้ำหนักของผู้ใช้โปรแกรมหรือไม่

2.ทดสอบการแสดงผลกราฟการรักษาค่าได้



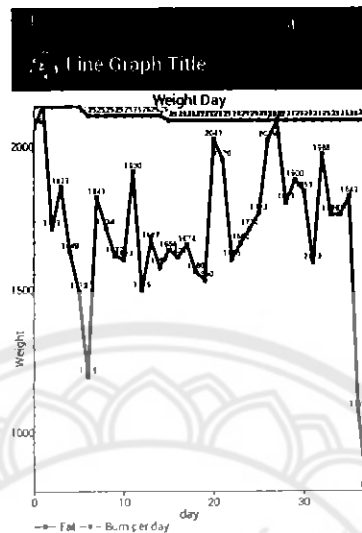
รูปที่ 4.18 กราฟการรักษาค่าได้

กราฟเส้นสีแดง = จำนวนเป้าหมายการลดการนำเข้าพลังงานในแต่ละวัน

กราฟเส้นสีน้ำเงิน = จำนวนการลดการนำเข้าพลังงานที่ทำได้จริง

ผลที่ได้จะใช้ทำการเปรียบเทียบดูว่าเรากินอยู่ในขอบเขตหรือไม่จะได้ปรับเมนูให้เข้ากับการควบคุมน้ำหนักมากยิ่งขึ้น

3.ทดสอบการแสดงผลกราฟการกินต่อวัน



รูปที่ 4.19 กราฟแสดงผลการกินต่อวัน

กราฟเส้นสีแดง = จำนวนจำกัดการนำเข้าพลังงานในแต่ละวัน

กราฟเส้นสีเขียว = จำนวนการนำเข้าพลังงานที่ทำได้จริง

ถ้ากราฟเส้นสีเขียวอยู่เหนือเส้นแดง แสดงถึง คุณกินเกินจำนวนจำกัด

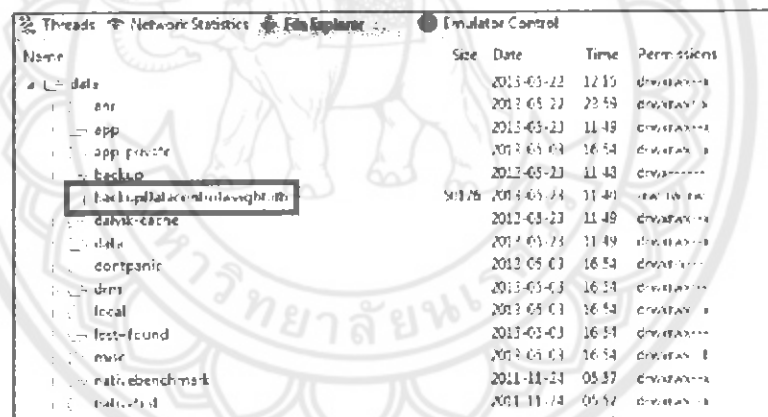
ถ้ากราฟเส้นสีเขียวอยู่เสมอสันแดง คุณกินพอดี

ถ้ากราฟเส้นสีเขียวอยู่ต่ำเส้นแดง คุณลดการนำเข้าพลังงานได้

จากผลที่ได้ผู้ใช้จะสามารถควบคุมแนวโน้มการนำเข้าพลังงานในแต่ละวันได้และจำกัดขอบเขตการนำเข้าพลังงานได้

4.6 ทดสอบการจัดการข้อมูล Import และ Export

1. ทดสอบการ Export ข้อมูล

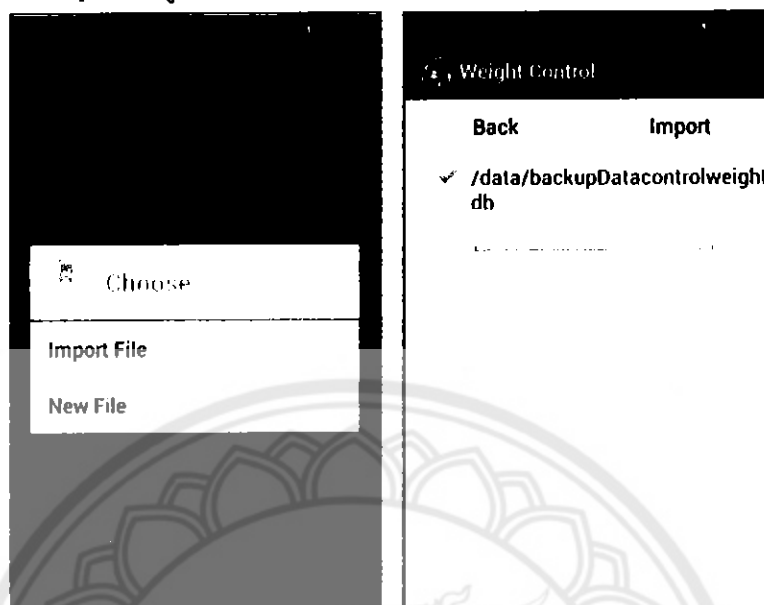


รูปที่ 4.20 Export ข้อมูล

จากทดสอบการ Export ข้อมูลจากรูปที่ 4.20 บนด้านซ้าย ให้คลิกที่ปุ่มนำออก ก็จะไปเป็นตามรูปที่ 4.20 บนด้านขวา ให้เลือกที่อยู่ของไฟล์ที่จะ Export ไป หรือพิมพ์ที่อยู่ของไฟล์ก็ได้ จากการทดสอบ Export เลือกที่อยู่ไฟล์คือ

“/data/data/nucpe.com.controlweights/databases/mydbcontrolweight.db” ตามรูปที่ 4.20 บนด้านขวา ส่วนรูปที่ 4.20 ด้านล่างจะเป็นตำแหน่งที่อยู่ไฟล์ที่ Export ออกมา

2.ทดสอบการ Import ข้อมูล



รูปที่ 4.21 Import ข้อมูล

จากทดสอบการ Import ข้อมูล เมื่อเปิด โปรแกรมครั้งแรกจะมีให้เลือกว่าจะ “Import File” หรือ “New File” ตามรูปที่ 4.21 ด้านซ้าย ให้เลือก “Import File” แล้วก็จะไปเป็นหน้าตามรูปที่ 4.21 ด้านขวา ให้เลือกที่อยู่ไฟล์ที่เราต้องการ Import เข้ามา หรือจะพิมพ์ที่อยู่ไฟล์ก็ได้ จากการทดสอบเลือกที่อยู่ไฟล์ เป็นตาม รูปที่ 4.21 ด้านขวา คือ “/data/backupDatacontrolweight.db” แล้วกด Import

บทที่ 5

สรุปและผลดำเนินงาน

5.1 สรุปและอภิปรายผล

ปัจจุบัน ผู้คนหันมาดูแลสุขภาพกันมากขึ้นแต่ยังขาดวินัยในการดูแลสุขภาพ หรือความสม่ำเสมอตนเอง ด้วยปัญหานี้เราจึงคิดที่จะทำให้การดูแลสุขภาพเป็นเรื่องที่เข้าถึงง่ายให้อยู่ในรูปแบบที่ใกล้ตัวมากที่สุด เพื่อความสะดวกในการดูแลสุขภาพ จึงได้ข้อสรุปว่าต้องเป็นการพัฒนาแอปพลิเคชันบนโทรศัพท์มือถือ เพราะปฏิเสธไม่ได้ว่าในปัจจุบันโทรศัพท์มือถือ นับว่าเป็นสิ่งจำเป็นสำหรับการดำเนินชีวิตของคนในยุคปัจจุบันไปแล้ว การดูแลสุขภาพด้วยโทรศัพท์มือถือ จึงเป็นสิ่งที่น่าจะทำให้เกิดวินัยในการดูแลสุขภาพมากยิ่งขึ้น

วัตถุประสงค์ เพื่อพัฒนาแอปพลิเคชันที่ใช้งานโทรศัพท์มือถือ สามารถบันทึกข้อมูล ปริมาณค่าโภชนาการของอาหาร ค่าพลังงานที่ใช้ในแต่ละวัน และ คำนวณค่าที่ได้จากค่าพลังงานที่ได้จากอาหาร และค่าพลังงานที่ใช้ในแต่ละวัน เพื่อสรุปว่าในแต่ละวันสามารถลดไปได้มากน้อยเท่าไร เพื่อความสะดวกในการวางแผนการลดน้ำหนักอย่างเป็นระบบ สร้างนิสัยการดูแลตนเองเพื่อคุณภาพชีวิตที่ดีขึ้น

รูปแบบระบบการทำงานของโปรแกรม สามารถบันทึกข้อมูล ปริมาณค่าโภชนาการของอาหาร ค่าพลังงานที่ใช้ในแต่ละวัน และ คำนวณค่าที่ได้จากค่าพลังงานที่ได้จากอาหารและค่าพลังงานที่ใช้ในแต่ละวัน เพื่อสรุปว่าในแต่ละวันสามารถลดไปได้มากน้อยเท่าไร ภาษาที่ใช้กับโปรแกรมสามารถใช้ได้ทั้งสองภาษาคือภาษาไทยและภาษาอังกฤษ

ในส่วนตัวทางด้านเทคนิคการเขียนโปรแกรมด้วย Java บนระบบ Android ถือว่าเป็นความรู้ใหม่ที่เรารเริ่มศึกษาระบบเพื่อนำมาพัฒนา การพัฒนามีความซับซ้อนต้องใช้ความรู้ในทุกๆด้าน ไม่ว่าจะเป็นด้านการเขียนโปรแกรมภาษา JAVA การสร้างฐานข้อมูล โดย SQLite ซึ่งจำเป็นที่จะต้องศึกษาโดยละเอียด เพราะตัวโปรแกรมที่พัฒนานั้น มีความยุ่งยากและซับซ้อน เพื่อให้ตอบสนองกับผลที่คาดหวังไว้ตั้งแต่แรก แต่การมุ่งศึกษาในเรื่องนี้ถือว่าเป็นสิ่งที่มีประโยชน์มาก เพราะอัตราการใช้งานของผู้ใช้โทรศัพท์มือถือ ระบบ Android มีมากขึ้นทุกวันนับว่าเป็นพื้นที่ทางการตลาดที่น่าสนใจเป็นอย่างมาก

5.2 ปัญหาที่พบในการพัฒนาระบบ

1. ผู้พัฒนาไม่เคยเขียนโปรแกรมบนมือถือมาก่อน จึงต้องใช้เวลาในการศึกษาพอสมควร
2. ผู้พัฒนาไม่เคยศึกษาโครงสร้างของ xml มาก่อนจึงจำเป็นต้องใช้เวลาศึกษา
3. มีปัญหาในการออกแบบระบบฐานข้อมูล มีการแก้ไขฐานข้อมูลหลายครั้ง
4. มีปัญหาในการดึงข้อมูลจากฐานข้อมูลมาแสดงในหน้า xml
5. มีปัญหาการใช้ฟังก์ชันต่างๆภายในการพัฒนาเนื่องจากไม่เคยศึกษาจึงต้องใช้เวลาในศึกษา
6. มีปัญหาการในการเขียนโค้ดจาวาเชื่อมกัน xml และ ฐานข้อมูล
7. ขาดประสบการณ์ในการออกแบบกราฟฟิคอินเตอร์เฟซ ทำให้ยากต่อการออกแบบให้ตอบสนองกับผู้ใช้งาน

5.3 ข้อเสนอแนะ

การพัฒนาแอปพลิเคชันในโทรศัพท์มือถือระบบ แอนดรอยด์ นั้นต้องอาศัยความรู้ตั้งแต่พื้นฐานไปจนถึงระดับสูง มีความจำเป็นอย่างมากที่จะต้องเข้าใจการเขียนโปรแกรมในภาษา JAVA เพราะการพัฒนาโปรแกรมโดยส่วนใหญ่เราจะใช้ภาษานี้เป็นหลัก ต้องรู้จักแก้ไขปัญหาที่เกิดขึ้นระหว่างการพัฒนาโปรแกรม เพื่อให้เกิดองค์ความรู้คู่ไปกับประสบการณ์ในการพัฒนาตัวโปรแกรม การวางแผนการทำงานเป็นสิ่งที่มีความจำเป็นอย่างมากเพราะเนื่องจากระยะเวลาที่พัฒนานั้นผู้วิจัยจะต้องเรียนรู้ไปด้วยและพัฒนาโปรแกรมไปด้วยจึงเป็นงานที่ต้องแข่งกับเวลาที่มีจำกัด และการสำรวจความต้องการของผู้ใช้นั้น ควรจำกัดวงของกลุ่มผู้ใช้ให้ชัดเจนเพื่อความง่ายในการพัฒนา

การแก้ไขปัญหาความรู้ทางการเขียนโปรแกรม ซึ่งบางครั้งอาจจะเกิดปัญหาที่เราไม่เคยเจอมาก่อน วิธีแก้ก็คือเราควรศึกษาเพิ่มเติมจากนักพัฒนาโปรแกรมคนอื่น ๆ ที่เคยมีประสบการณ์จึงจะทำให้การแก้ไขปัญหาและการพัฒนาโปรแกรมเป็นไปได้อย่างราบรื่น

5.4 ความสามารถของโปรแกรม

โปรแกรมที่พัฒนา	โปรแกรมอื่นๆ
1. มีเมนูอาหารสำหรับคนไทย ใช้งานง่าย	1. เมนูอาหารไม่เหมาะสำหรับคนไทย ใช้งานยาก
2. มีการรวมหลายๆ โปรแกรมด้านสุขภาพเข้าด้วยกัน	2. แยกประเภทโปรแกรม แยกเป็น 3-4 โปรแกรม
3. สามารถทำการ สำรองฐานข้อมูลออกมาได้	3. ไม่สามารถสำรองข้อมูลได้
4. มีการแสดงผลของข้อมูลสถานะสุขภาพชัดเจน	4. การแสดงผลบางรายการเข้าใจยาก
5. สามารถแก้ไข หรือ เพิ่มข้อมูลย้อนหลังได้	5. แก้ไขข้อมูล หรือ เพิ่มข้อมูลย้อนหลังไม่ได้
6. รายงานผลในส่วนของกราฟหลากหลายประเภท	6. การรายงานผลของกราฟมีแบบเดียว

ตารางที่ 5.1 ตารางเปรียบเทียบความสามารถ

เอกสารอ้างอิง

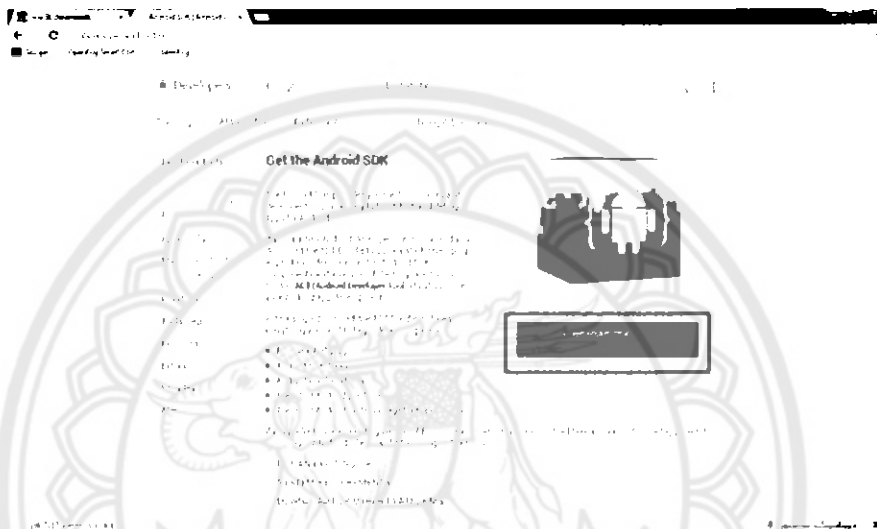
- [1] Jane Steele, Nelson To (2554) รวมโค้ด Android App กรุงเทพฯ: สำนักพิมพ์ ทรู ดิจิตอล คอนเท้นท์ แอนด์ มีเดีย
- [2] รศ.ธีรวัฒน์ ประกอบผล (2555) Java OOP กรุงเทพฯ: สำนักพิมพ์ ชิมพลีฟาย
- [3] โอภาส เอี่ยมสิริวงศ์ (2551) ระบบฐานข้อมูล กรุงเทพฯ: สำนักพิมพ์ ซีเอ็ดดูเคชั่น
- [4] Google (2556) Android Getting Started จาก
<http://developer.android.com/training/index.html>
- [5] Google (2556) Android Sharing Content จาก
<http://developer.android.com/training/sharing/index.html>
- [6] McDonald's USA (2556) Nutrition Facts for Popular Menu Items จาก
nutrition.mcdonalds.com/getnutrition/nutritionfacts.pdf
- [7] KFC (2556) NUTRITION GUIDE จาก
www.kfc.com/nutrition/pdf/kfc_nutrition.pdf
- [8] พร้อมเลิศ หล่อวิจิตร (2556) คู่มือเขียนแอป Android ฉบับสมบูรณ์ กรุงเทพฯ: สำนักพิมพ์ โปรวิชั่น
- [9] กอบเกียรติ สระอุบล (เมษายน 2556) พัฒนา App Android กรุงเทพฯ: สำนักพิมพ์ มีเดีย เนทเวิร์ก
- [10] กองโภชนาการ กรมอนามัย (2555) การดำเนินการวิเคราะห์คุณค่าทางโภชนาการของอาหารไทย จาก <http://nutrition.anamai.moph.go.th/FoodTable/Html/frame.html>
- [11] Pizza Hut (2556) NUTRITIONAL INFORMATION จาก
<http://www.pizzahut.com/nutritioninformation.html>
- [12] Love fit (2556) ผลงานในอาหารญี่ปุ่น “ซูชิ” จาก
<http://www.lovefitt.com/calories-monitor/ผลงานในอาหารญี่ปุ่น-ซูชิ>
- [13] Love fit (2556) ปริมาณแคลอรีของเมนูฮอตฮิตจาก 7-11
<http://www.lovefitt.com/calories-monitor/ปริมาณแคลอรีของเมนูฮอตฮิตจาก-7-11>
- [14] Nutristrategy (2555) Calories Burned During Exercise จาก
<http://www.nutristrategy.com/activitylist4.htm>
- [15] Harvard Medical School (2555) Calories burned in 30 minutes for people of three different weights จาก <http://www.health.harvard.edu/newsweek/Calories-burned-in-30-minutes-of-leisure-And-routine-activities.htm>
- [16] เอกสาร การประชุมวิชาการสมาคมนักกำหนดอาหารแห่งประเทศไทย 2556

ภาคผนวก ก

คู่มือการติดตั้ง Plug in Android บน eclipse

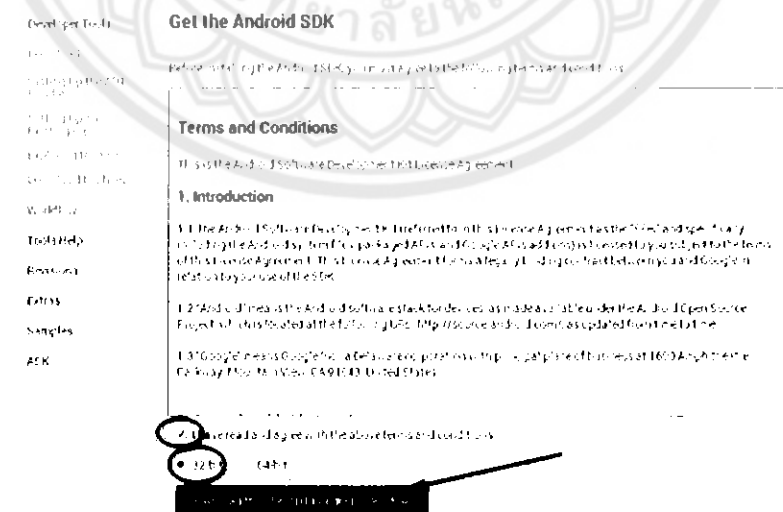
ก.1 ติดตั้ง ADT (android developer tools)

1. เป็นเครื่องมือที่ประกอบไปด้วย eclipse และ SDK โดย eclipse ทำหน้าที่ช่วยให้เราสามารถเขียนโค้ดแอนดรอยด์ได้ง่ายขึ้น ส่วน SDK ทำหน้าที่เก็บรูปแบบคำสั่งต่างๆ, คลาสต่างๆ รอให้เรานำไปใช้ในการจะใช้ให้ไปโหลด ADT มาไว้ในเครื่องก่อน ไปที่ <http://developer.android.com/sdk/index.html>



รูปที่ ก.1 แสดงขั้นตอนการติดตั้ง ADT

2. คลิกที่ Download the SDK ได้ อย่าลืม Read Agree และเลือก โหลดให้เหมาะกับสถาปัตยกรรมของระบบปฏิบัติการของตนเอง



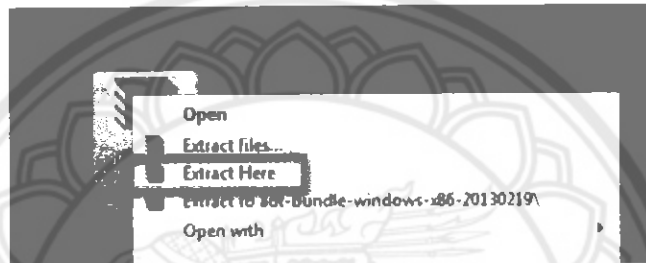
รูปที่ ก.2 แสดงขั้นตอนการติดตั้ง ADT

3. โหลดเสร็จ จะได้ ไฟล์ ADT ที่ zip ไว้แบบนี้



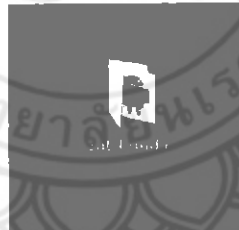
รูปที่ ก.3 แสดงขั้นตอนการติดตั้ง ADT

4. ให้ Extract หรือ แดกออก



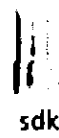
รูปที่ ก.4 แสดงขั้นตอนการติดตั้ง ADT

5. แดกเสร็จจะได้ไฟล์เคอร์ ADT แบบนี้



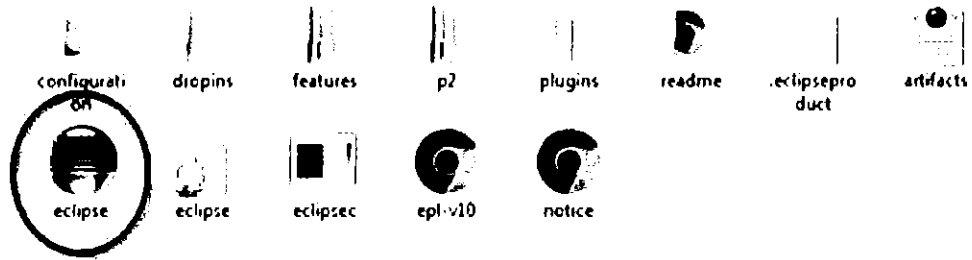
รูปที่ ก.5 แสดงขั้นตอนการติดตั้ง ADT

6. ลองเข้าไปในโฟลเดอร์นี้



รูปที่ ก.6 แสดงขั้นตอนการติดตั้ง ADT

7. จะสังเกตเห็นโฟลเดอร์ eclipse, sdk ให้เข้าไปในโฟลเดอร์ eclipse



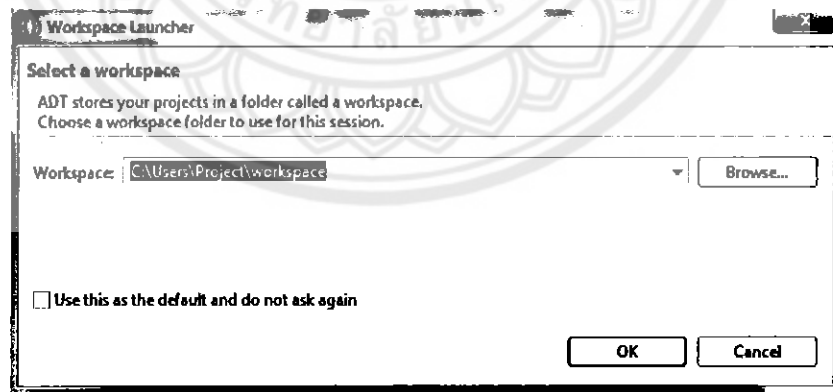
รูปที่ ก.7 แสดงขั้นตอนการติดตั้ง ADT

8. จะเห็น Application File ที่ชื่อว่า eclipse ดับเบิ้ลคลิก เพื่อเปิด eclipse หรือจะ Sent to Desktop เพื่อสร้าง Shot-Cut ในการเปิด eclipse ได้ง่าย ๆ



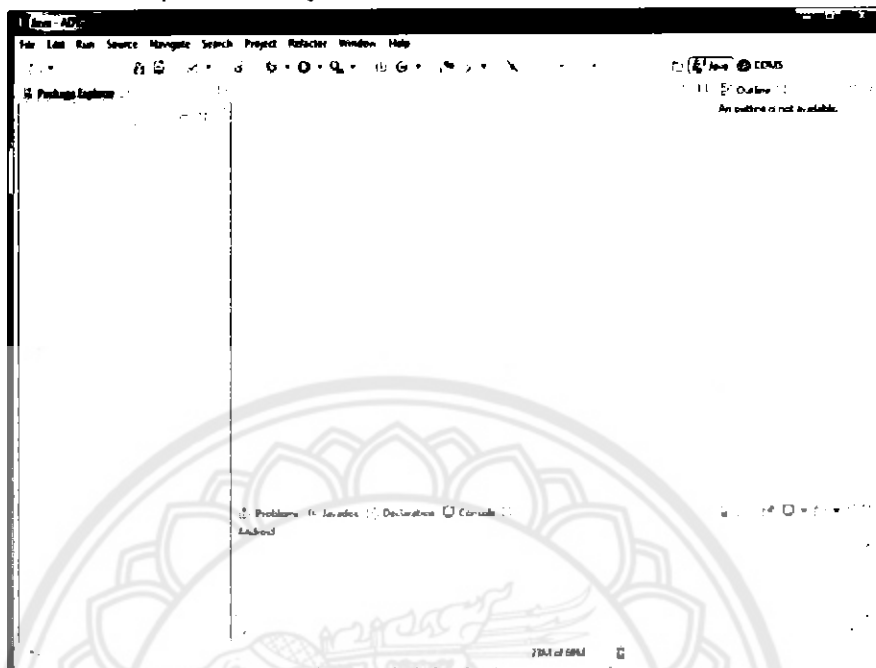
รูปที่ ก.8 แสดงขั้นตอนการติดตั้ง ADT

9. ตัว eclipse จะถามหา ตำแหน่งโฟลเดอร์ ที่ใช้สำหรับ เก็บ Source Code Android ที่จะเขียนขึ้น (Workspace)



รูปที่ ก.9 แสดงขั้นตอนการติดตั้ง ADT

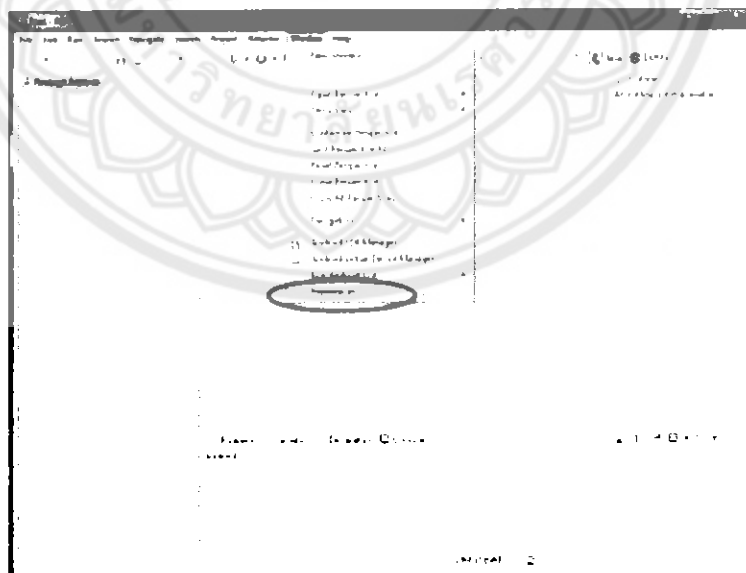
10. หลังจากนั้นก็ปุ่ม OK จะเข้าสู่หน้าพัฒนาโปรแกรม



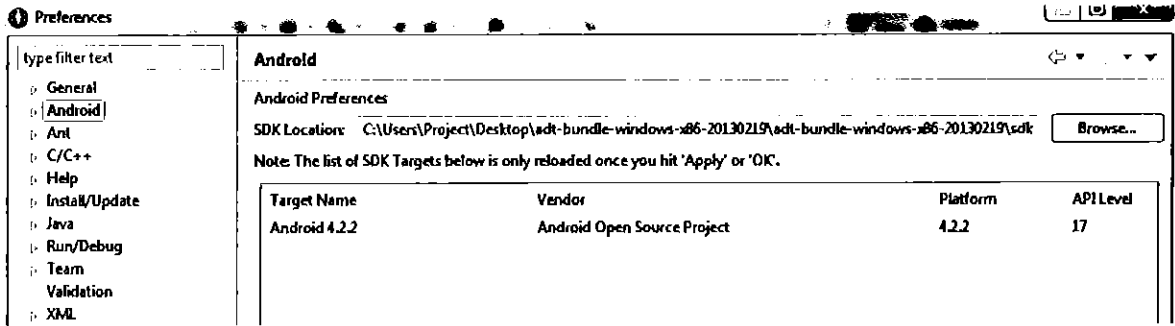
รูปที่ ก.10 แสดงขั้นตอนการติดตั้ง ADT

ก.2 ติดตั้ง Platform Android เพิ่ม/ลด

1. SDK Manager ตัว ADT ที่เราโหลด มาแตก จะได้ eclipse และ SDK โดย SDK ตัวนี้จะมี Platform ตัวล่าสุดอย่างตอนนี้ คือ android 4.2

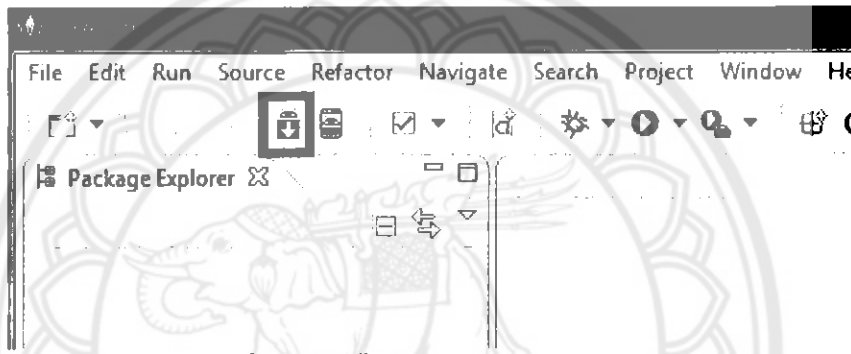


รูปที่ ก.11 ติดตั้ง Platform Android เพิ่ม/ลด



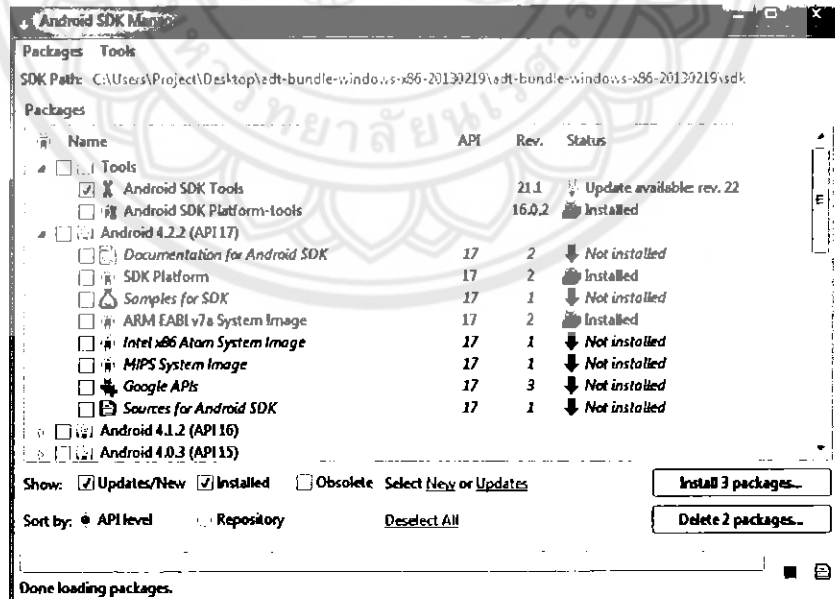
รูปที่ ก.12 ติดตั้ง Platform Android เพิ่ม/ลด

2. สามารถเพิ่มหรือลด ตัว Platform หรือ รายละเอียดภายใน Platform โดยกดปุ่ม Android SDK Manager



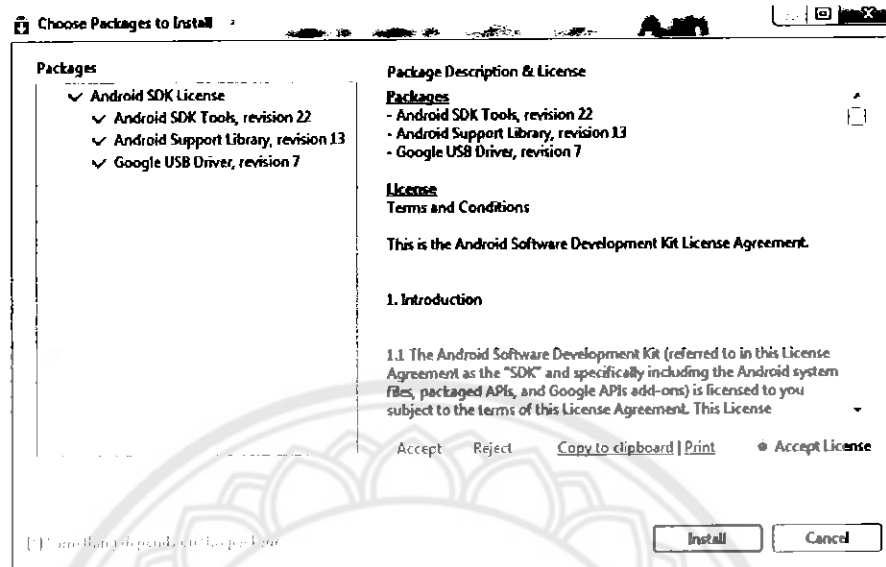
รูปที่ ก.13 ติดตั้ง Platform Android เพิ่ม/ลด

3. หลังจากกดเลือก ไฟล์ที่จะติดตั้งแล้ว ให้ทำการกดปุ่ม Install Packages



รูปที่ ก.14 ติดตั้ง Platform Android เพิ่ม/ลด

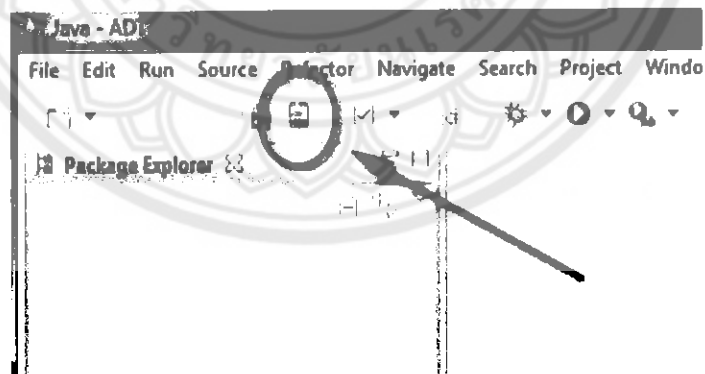
4. หลังจากกดเลือกไฟล์ที่จะติดตั้งแล้วเลือกที่ Accept License ให้ทำการกดปุ่ม Install



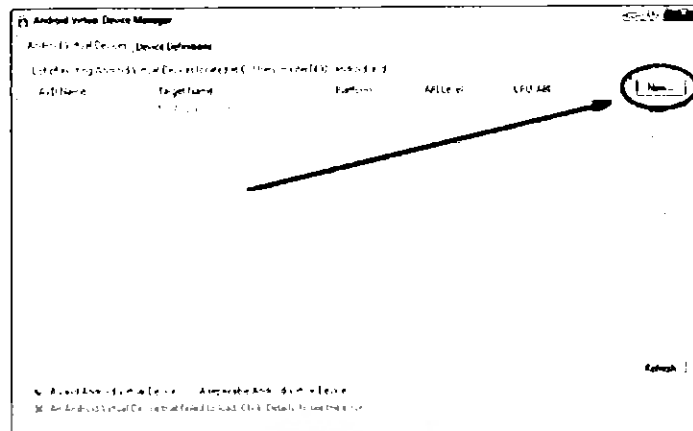
รูปที่ ก.15 ติดตั้ง Platform Android เพิ่ม/ลด

ก.3 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์

1. ในการเขียน โค้ดแอนดรอยด์ จะมีกระบวนการของการทดสอบ โค้ดแอนดรอยด์ ที่เขียนได้ ว่า ทำงานได้อย่างที่ Developer ตั้งใจหรือเปล่า เราเรียก กระบวนการนี้ว่า การทำ Testing เพื่อความ สะดวกจะ สมมุติ มีมือถือที่มีระบบปฏิบัติการแอนดรอยด์ เวอร์ชัน และ หน้าจอ ที่มุ่งหวัง ก่อนที่จะนำโค้ดแอนดรอยด์ ที่เรา เขียนได้ไปทดสอบบนเครื่องจริง วิธีการสร้าง Emulator ให้คนที่ ไอคอน Android Virtual Device

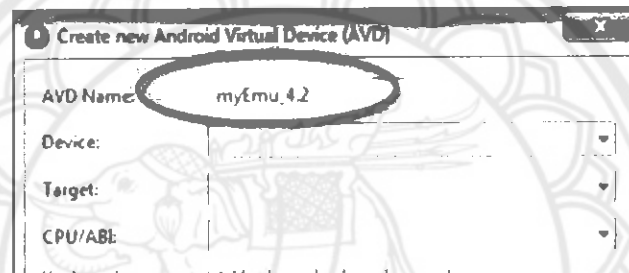


รูปที่ ก.16 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์



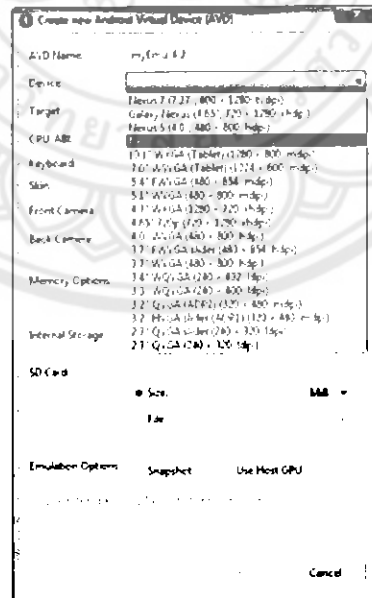
รูปที่ ก.17 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์

2. ที่ AVD Name ให้ พิมพ์ ชื่อของ Emulator ที่ต้องการลงไป



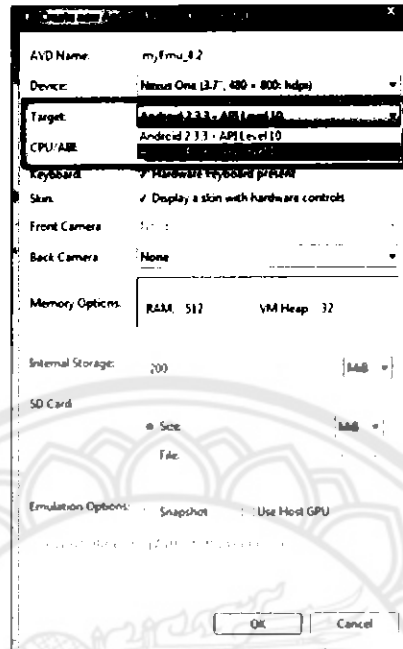
รูปที่ ก.18 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์

3. และ Device เป็นตัวกำหนดขนาด ของหน้าจอ ของ Emulator ที่ต้องการ



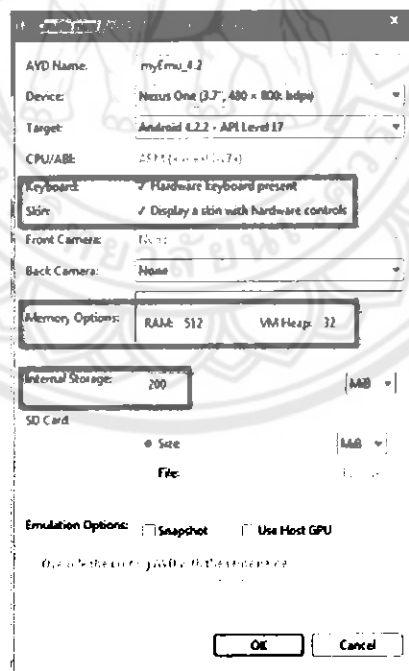
รูปที่ ก.19 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์

4. ต่อไปที่เลือก Platform ของระบบปฏิบัติการแอนดรอยด์ ที่ต้องการให้ติดตั้ง



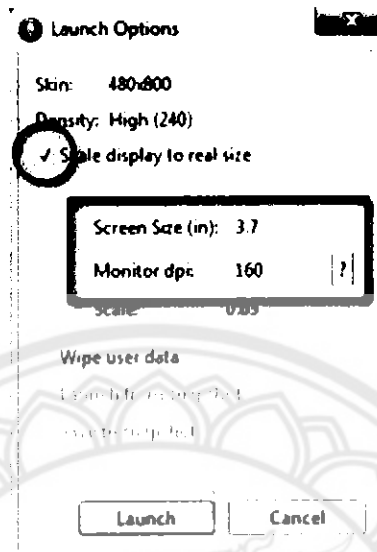
รูปที่ ก.20 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์

5. ต่อมาก็เลือก Option ให้กับ Emulator



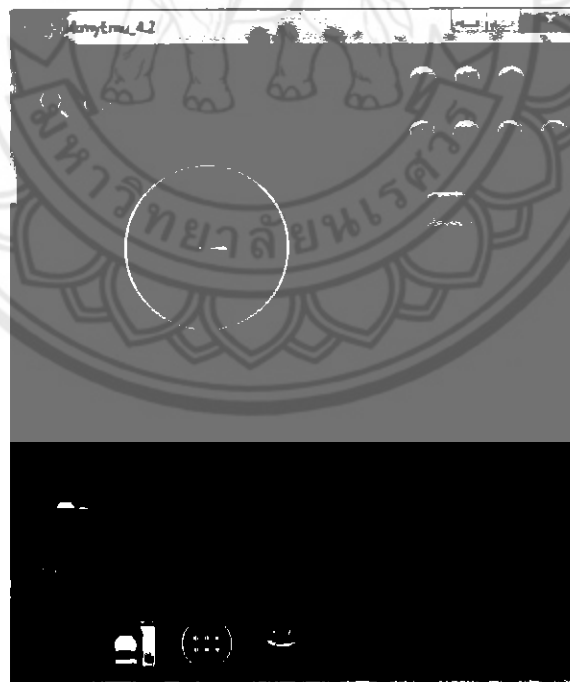
รูปที่ ก.21 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์

6. Start Emulator ให้ทำการ Select Emulator ที่ต้องการ และคลิก Start สังเกตว่าถ้าไม่ Select Emulator ตัวใด ตัวหนึ่ง ปุ่ม Start จะคลิกไม่ได้



รูปที่ ก.22 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์

7. รอจน Emulator บูรจนครบก่อนค่อย ปิด Android Virtual Device Manager



รูปที่ ก.23 สร้าง Emulator สำหรับ ทดสอบ โค้ดแอนดรอยด์

ภาคผนวก ข
ตัวอย่างโค้ดที่ใช้พัฒนา

ตัวอย่าง Code ในการสร้างฐานข้อมูล

```

public class DBcontrolWeights extends Activity {
    //////////////////////////////////////////////////ชื่อ คาด้านเบส//////////////////////////////////////
    public static final String DBNAME = "mydbcontrolweight.db";
    //public static double
    //////////////////////////////////////////////////ตารางข้อมูลพื้นฐาน//////////////////////////////////////
    public static final String DATABASEBASIS ="DATABASEBASIS";
    public static final String KEYID = "KEYID";
    public static final String WEIGHTB = "WEIGHTB";
    public static final String HEIGHTB ="HEIGHTB";
    public static final String AGEB = "AGEB";
    public static final String TARGETB = "TARGETB";
    public static final String GENDEB = "GENDEB";
    public static final String GOALLOSEB = "GOALLOSEB";
    //////////////////////////////////////////////////ตารางน้ำหนัก//////////////////////////////////////
    public static final String DBWEIGHT ="DBWEIGHT";
    public static final String KEYIDW ="KEYIDW";
    public static final String DAYEX = "DAYEX";
    public static final String MONTHEX = "MONTHEX";
    public static final String YEAREX = "YEAREX";
    public static final String WEIGHTD = "WEIGHTD";
    public static final String DataLanguage ="DataLanguage";
    public static final String KEYIDL ="KEYIDL";
    public static final String TLanguage = "TLanguage";

```



```

//////////////////////////////////ตารางอาหาร//////////////////////////////////
public static final String DATAFOOD ="DATAFOOD";
public static final String KEYIDFOOD = "KEYIDFOOD";
public static final String NFOOD = "NFOOD";
public static final String CALORIEFOOD = "CALORIEFOOD";

```

ตัวอย่าง Code กำสั่งสร้างตารางในการเก็บข้อมูลฐานข้อมูล

```

@Override
public void onCreate(SQLiteDatabase db) {
//สร้างตารางซึ่งในตารางข้อมูลพื้นฐาน
db.execSQL( "CREATE TABLE "+DATABASEBASIS +"("+KEYID+" INTEGER PRIMARY KEY
AUTOINCREMENT, "
+WEIGHTB+" DOUBLE, "
+HEIGHTB+" DOUBLE, "
+AGEB+" DOUBLE, "
+TARGETB+" DOUBLE, "
+GENDEB+" String, "
+GOALLOSEB+ " DOUBLE );");
//อาหาร
db.execSQL( "CREATE TABLE "+DATAFOOD+"("+KEYIDFOOD+" INTEGER PRIMARY KEY
AUTOINCREMENT, "
+NFOOD+" String, "
+CALORIEFOOD+ " DOUBLE );");

```

ตัวอย่าง Code การเพิ่มข้อมูลในตารางฐานข้อมูล

```

public void addDATABASEBASIS(double a1, double b1, double c1, double a2, String b2, double
c2){
db = DBHelper.getWritableDatabase();
db.execSQL("INSERT INTO " + DATABASEBASIS+ " (WEIGHTB, HEIGHTB, AGEb, TARGETB,
GENDEB, GOALLOSEB) VALUES( " + a1 + ", " + b1 + ", " + c1 + ", " + a2 + ", " + b2 + ", " +
c2 + " )");
db.close();}

public void addFood(String b2, double c2){
db = DBHelper.getWritableDatabase();
db.execSQL("INSERT INTO " + DATAFOOD+ " (NFOOD, CALORIEFOOD) VALUES ( " + b2 +
", " + c2 + " )");
db.close();}

```

ตัวอย่าง Code การนำข้อมูลเพิ่มในฐานข้อมูลที่จะนำเอาไปใส่ในตาราง

```

private void setFood_En(){
dbfl.addFood("Khaomankai", 596);
dbfl.addFood("KhaoMudaeng", 541);
dbfl.addFood("Khaokaeng Khiaowan Kai", 483);
dbfl.addFood("Khaokhamu", 436);
dbfl.addFood("Khao Kra Phrao Kai", 554);
dbfl.addFood("Khaophat Mu Sai Khai", 557);
dbfl.addFood("Sen Yai Rat Na Mu ,Kai", 397);
dbfl.addFood("Sen Yai Phat Si Io Mu Sai Khai", 679);
dbfl.addFood("Khanomchin Namya", 332);
dbfl.addFood("Chok Sai Khai", 250);
dbfl.addFood("Khaotom Pla", 325);
dbfl.addFood("Kraphopla", 250);

```

ตัวอย่าง Code ในการเลือกภาษาเริ่มต้น

```

dbfl.addWeight(aD, aM, aY, TESTSTRING1);
if(Algorithm.DataLanguage.equals("ไทย")){
    setFood_Th();
}else if(Algorithm.DataLanguage.equals("English")){
    setFood_En();
}
dbfl.addLanguage(Algorithm.DataLanguage);
Intent intent = new Intent(getApplicationContext(),mainmenu.class);
startActivity(intent);}

```

ตัวอย่าง Code ในการรับค่าเริ่มต้น

```

private void setBinding(){
    buttonOKBs = (Button) findViewById(R.id.buttonOKB);
    buttonCancels = (Button) findViewById(R.id.buttonCancel);
    EditInputWeight = (EditText) findViewById(R.id.editTextWeight);
    EditInputHeight = (EditText) findViewById(R.id.editTextHeight);
    EditInputAge = (EditText) findViewById(R.id.editTextAge);
    EditInputTarget = (EditText) findViewById(R.id.editTextTarget);
    EditInputLoseWeight = (EditText) findViewById(R.id.editTextLose);
    chkM = (CheckBox)findViewById(R.id.checkBox1);
    chkF = (CheckBox)findViewById(R.id.checkBox2);
}

```

ตัวอย่าง Code ในการอัปเดตค่าน้ำหนัก

```

buttonUpdateWeight.setOnClickListener(new OnClickListener() {
    public void onClick(View v) {
        AlertDialog.Builder alert = new AlertDialog.Builder(mainmenu.this);
        alert.setTitle("Update Weight");
        alert.setMessage("Input Weight Kg");
        final EditText input = new EditText(mainmenu.this);
        alert.setView(input);
        alert.setPositiveButton("Ok", new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int whichButton) {
                Editable value = input.getText();
                String a = value.toString();
                AlertDialog ab = new AlertDialog.Builder(mainmenu.this).create();
                ab.setTitle("MONTH DAY YEAR");
                String aStringY = Integer.toString(mYear);
                String aStringM = Integer.toString(mMonth);
                String aStringD = Integer.toString(mDay);
                ab.setMessage(aStringM+"-"+aStringD+"-"+aStringY+" : "+a);
                double aD= Double.parseDouble(aStringD);
                double aM= Double.parseDouble(aStringM);
                double aY= Double.parseDouble(aStringY);
                double aW= Double.parseDouble(a);
                dbfl.UpdataaddWeight(aD, aM, aY, aW);
                dbfl.UpdateWeight(aW,Algorithm.Height,Algorithm.Ago,Algorithm.Target,Algorithm
                .Sex,Algorithm.LoseWeight);
                dbfl.open();
                dbfl.OpenDATABASEBASIS();
                ab.setPositiveButton("OK", new DialogInterface.OnClickListener() {

```

```

public void onClick(DialogInterface dialog, int which) {
    });
    ab.show();
    });
    alert.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int whichButton) {
            });
        });
    alert.show();
    });}

```

ตัวอย่าง Code ในการแสดงภาพสอดคล้องกับน้ำหนักตัว

```

private void setImageView(){
    image1 =(ImageView)findViewById(R.id.imageView1);
    String sw = Algorithm.Sex;
    if(sw.equals("Male")){
        if(Algorithm.BMI<18.5){
            image1.setImageResource(R.drawable.mtouw);
        }else if(Algorithm.BMI>=18.5 && Algorithm.BMI<=24.9){
            image1.setImageResource(R.drawable.mtonw);
        }else if(Algorithm.BMI>=25 && Algorithm.BMI<=29.9){
            image1.setImageResource(R.drawable.mtosow);
        }else if(Algorithm.BMI>=30 && Algorithm.BMI<=39.9){
            image1.setImageResource(R.drawable.mtoow);
        }else if(Algorithm.BMI>=40 && Algorithm.BMI<=60){
            image1.setImageResource(R.drawable.mtoeo);
        }
    }
}

```

ตัวอย่าง Code ในการรับค่าพลังงานอาหาร

```

public void onCreate(Bundle savedInstanceState) {
    dbfl = new DBcontrolWeights(this);
    dbfl.open();
    super.onCreate(savedInstanceState);
    setContentView(R.layout.inputcalorie);
    if(Algorithm.DataLanguage.equals("ไทย")){
        setText_Th();}
    txtvPay1 = (TextView)findViewById(R.id.txtView1);
    String payShow = StrNce;
    txtvPay1.setText(payShow);
    DecimalFormat dob = new DecimalFormat("###0");
    txtvPay2 = (TextView)findViewById(R.id.txtView2);
    String payShow1 =String.valueOf(dob.format(Algorithm.Calories));
    txtvPay2.setText(payShow1+" Calorie");
    ListView listView = (ListView) findViewById(R.id.listView1);
    String[] values = dbfl.showNFood();
    ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
    android.R.layout.simple_list_item_1, android.R.id.text1, values);
    listView.setAdapter(adapter);
    listView.setOnItemClickListener(new OnItemClickListener() {
        public void onItemClick(AdapterView<?> parent, View view,int position, long id) {
            String[] values = dbfl.showNFood();
            String[] valuesC = dbfl.inLineshowC("DATAFOOD");
            Algorithm.setpositions(position);
            AlertDialog.Builder alert = new AlertDialog.Builder(inputfood,this);
            alert.setTitle("This is Alert Dialog");
            alert.setMessage(values[position]+" "+valuesC[position]+" Calorie");
            alert.setPositiveButton(StrY,new DialogInterface.OnClickListener() {

```

```

public void onClick(DialogInterface dialog, int id) {
String[] values = dbf1.showNFood();
String[] valuesC = dbf1.inLineshowC("DATAFOOD");
String aStringY = Integer.toString(mYear);
String aStringM = Integer.toString(mMonth);
String aStringD = Integer.toString(mDay);
double aY = Double.parseDouble(aStringY);
double aM = Double.parseDouble(aStringM);
double aD = Double.parseDouble(aStringD);
String nN=values[Algorithm.positions];
double inCto = Double.parseDouble(valuesC[Algorithm.positions]);
dbf1.open();
dbf1.addC(nN, inCto, Algorithm.CalorieToDats, StrNee, aD, aM, aY);
dbf1.open();
dbf1.OpenDatafoodsD(aD,aM,aY);
Intent intent = new Intent(getApplicationContext(),menufood.class);
startActivity(intent);
Toast.makeText(inputfood.this, "Save : " + values[Algorithm.positions],
Toast.LENGTH_SHORT).show();});
alert.setNegativeButton(StrN, new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int id) {
String[] values = dbf1.showNFood();
Toast.makeText(inputfood.this, "No Save : " + values[Algorithm.positions],
Toast.LENGTH_SHORT).show();
});
alert.show();
});

```

ตัวอย่าง Code ในการเพิ่มรายการอาหาร

```

buttonOKBs = (Button) findViewById(R.id.buttonOKB);
buttonOKBs.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
Intent intent = new Intent(getApplicationContext(),addfood.class);
startActivity(intent);
}});
buttonCancels = (Button) findViewById(R.id.buttonBacks);
buttonCancels.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
Intent intent = new Intent(getApplicationContext(),menufood.class);
startActivity(intent);
}});
final Calendar c = Calendar.getInstance();
mYear = c.get(Calendar.YEAR);
mMonth = c.get(Calendar.MONTH);
mDay = c.get(Calendar.DAY_OF_MONTH);

```

ตัวอย่าง Code ปุ่ม Compare

```

public void onClick(View v) {
final CharSequence[] items = {"Food", "Drink", "Fruit",
"Candy", "KFC", "McDonald", "Sushi", "7-Eleven", "Pizza Hut"};
AlertDialog.Builder builder = new AlertDialog.Builder(Duplication.this);
builder.setTitle("Choose the type of skin");
builder.setIcon(R.drawable.ic_launcher);
builder.setItems(items, new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int item) {
txtvPayVSE.setText(items[item]);
if(item==0){
EatName = "DATAFOOD"; //DATAFOOD

```



```

}else if(item==1){
EatName = "DATADEINK"; //DATADEINK
}else if(item==2){
EatName = "DATADESSERTS"; //DATADESSERTS
}else if(item==3){
EatName = "DATAFRIUT"; //DATAFRIUT
}else if(item==4){
EatName = "DATAKFC"; //1DATAKFC
}else if(item==5){
EatName = "DATAMcDonald"; //2DATAMcDonald
}else if(item==6){
EatName = "DATASushi"; //3DATASushi
}else if(item==7){
EatName = "DATASevenEleven"; //4DATASevenEleven
}else if(item==8){
EatName = "DATAPizzaHut"; //5DATAPizzaHut
}
final String[] itemsEat = dbf1.showNEat(EatName);
AlertDialog.Builder builder = new AlertDialog.Builder(Duplication.this);
builder.setTitle("Choose the type of skin");
builder.setIcon(R.drawable.ic_launcher);
builder.setItems(itemsEat, new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int item) {
txtvPayVSE.setText(itemsEat[item]);
String[] valuesC = dbf1.inLineshowC(EatName);
intCtoE = Double.parseDouble(valuesC[item]);
}});
AlertDialog alert = builder.create();
alert.show();});});

```

```

AlertDialog alert = builder.create();
alert.show();
});

txtvPayVSA = (TextView)findViewById(R.id.textViewSA);
txtvPayVSA.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
final String[] items = dbf1.showActivity();
AlertDialog.Builder builder = new AlertDialog.Builder(Duplication.this);
builder.setTitle("Choose the type of skin");
builder.setIcon(R.drawable.ic_launcher);
builder.setItems(items, new DialogInterface.OnClickListener() {
public void onClick(DialogInterface dialog, int item) {
txtvPayVSA.setText(items[item]);
String[] valuesC = dbf1.inLineshowC("DATAACTIVITY");
inCtoA = Double.parseDouble(valuesC[item]);
});
AlertDialog alert = builder.create();
alert.show();});

buttonCs = (Button) findViewById(R.id.buttonCCO);
buttonCs.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
txtvPayVtime = (TextView)findViewById(R.id.textViewTim);
txtvPayVtime.setText("00 : 00");

double cod ;
cod=inCtoE/(inCtoA*Algorithm.Weight);
int tineH = 0,tineM = 0,tineS=0;
tineH= (int)cod/60/60;
tineM= ((int)cod/60)%60;

```

```

tineS= (int)cod%60;
DecimalFormat dob = new DecimalFormat("##");
String payShowH = String.valueOf(dob.format(tineH));
String payShowM = String.valueOf(tineM);
String payShowS = String.valueOf(tineS);
txtvPayVtime.setText(payShowH + " : "+payShowM + " : "+payShowS);
    });}

```

ตัวอย่าง Code 11 Random

```

private SensorManager sensorManager;
sensorManager=(SensorManager) getSystemService(SENSOR_SERVICE);
sensorManager.registerListener(this,sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER),
SensorManager.SENSOR_DELAY_NORMAL);
String colors[] = {"Food", "Drink", "Fruit", "Candy", "KFC", "McDonald", "Sushi", "7-Eleven", "Pizza Hut"};
final Spinner spinner = (Spinner) findViewById(R.id.spinner1);
ArrayAdapter<CharSequence> adapter = new
ArrayAdapter(this,android.R.layout.simple_spinner_item, colors);
adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
spinner.setAdapter(adapter);
buttonRandoms= (Button) findViewById(R.id.button1);
buttonRandoms.setOnClickListener(new OnClickListener() {
public void onClick(View v) {

int nsa = spinner.getSelectedItemPosition();
if(nsa==0){
values = dbfl.showNFood();
}else if(nsa==1){
values = dbfl.showDeink();

```

```

}else if(nsa==2){
values = dbf1.showFruits();
}else if(nsa==3){
values = dbf1.showDesserts();
}else if(nsa==4){
values = dbf1.showNEat("DATAKFC");
}else if(nsa==5){
values = dbf1.showNEat("DATAMcDonald");
}else if(nsa==6){
values = dbf1.showNEat("DATASushi");
}else if(nsa==7){
values = dbf1.showNEat("DATASevenEleven");
}else{
values = dbf1.showNEat("DATAPizzaHut");}
ssd = values.length;
if(qsw==0){
qsw=1;
intRandoms =randomsB();
}else if(qsw==1){
qsw=0; }
double erte= ((xx+yy+zz)*intRandoms)%ssd;
int ertes=(int) erte;
txtvPayn = (TextView)findViewById(R.id.textView3);
txtvPayn.setText(values[ertes]); }]);}

public static int randomsB(){
int min = 1;
int max = 20;
Random r = new Random();
int ra = r.nextInt(max - min+1)+min;

```

```

return ra;}

public void onAccuracyChanged(Sensor sensor,int accuracy){}

public void seere(){

public void onSensorChanged(SensorEvent event){

if(event.sensor.getType()==Sensor.TYPE_ACCELEROMETER){

float x=event.values[0];

float y=event.values[1];

float z=event.values[2];

xx= x;

yy= y;

zz= z;

if(xx>5){

qsw=1;}

if(yy>5){

qsw=0;}

if(qsw==1){

double erte= ((xx+yy+zz)*intRandoms)%ssd;

int ertes=(int) erte;

txtvPayn = (TextView)findViewById(R.id.textView3);

txtvPayn.setText(values[ertes]); }}}

```

ตัวอย่าง Code ปุ่ม The System time

```

buttonOKBs.setOnClickListener(new OnClickListener() {

public void onClick(View v) {

try{

try{

txtTimeM = Double.parseDouble(EditInputTimeM.getText().toString());

}catch(Exception e){

```

```

textTimeM=0; }

try{
textTimeS = Double.parseDouble(EditInputTimeS.getText().toString());
}catch(Exception e){
textTimeS=0; }

double AMS =(textTimeM*60)+textTimeS;
CAC = AMS*Algorithm.CalorieActivity*Algorithm.Weight;
DecimalFormat dob = new DecimalFormat("###0.000");
txtvPay2 = (TextView)findViewById(R.id.textView5);
String payShowS = String.valueOf(dob.format(CAC));
txtvPay2.setText(payShowS);
String aStringY = Integer.toString(mYear);
String aStringM = Integer.toString(mMonth);
String aStringD = Integer.toString(mDay);
double aY = Double.parseDouble(aStringY);
double aM = Double.parseDouble(aStringM);
double aD = Double.parseDouble(aStringD);
dbf1.addCalorieActivity(Algorithm.NActivity, AMS, CAC, aD, aM, aY);
}catch(Exception e){ });

buttonBack.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
Intent intent = new Intent(getApplicationContext(),mainmenu.class);
startActivity(intent); });

buttonCalculates.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
try{try{
textTimeM = Double.parseDouble(EditInputTimeM.getText().toString());
}catch(Exception e){
textTimeM=0;}

```

```

try{
textTimeS = Double.parseDouble(EditInputTimeS.getText().toString());
}catch(Exception e){
textTimeS=0; }
double AMS =(textTimeM*60)+textTimeS;
double CAC = AMS*Algorithm.CalorieActivity*Algorithm.Weight;
DecimalFormat dob = new DecimalFormat("###0.000");
txtvPay2 = (TextView)findViewById(R.id.textView5);
String payShowS = String.valueOf(dob.format(CAC));
txtvPay2.setText(payShowS); }
catch(Exception e) { });

```

ตัวอย่าง Code ปุ่ม Timer

```

public void onClick(View v) {
myChronometer.start();
if(rCont==0){
myChronometer.start();
image1.setImageResource(R.drawable.action_icons_s);
rCont=1;
}else if(rCont==1){
myChronometer.stop();
image1.setImageResource(R.drawable.action_icons_p);
elapsedMillis = (SystemClock.elapsedRealtime() - myChronometer.getBase())/1000;
double AMS=elapsedMillis;
double CAC = AMS*Algorithm.CalorieActivity*Algorithm.Weight;
DecimalFormat dob = new DecimalFormat("###0.000");
txtvPay2 = (TextView)findViewById(R.id.textView5);
String payShowS = String.valueOf(dob.format(CAC));
txtvPay2.setText(payShowS);

```

```

Toast.makeText(timerc.this, payShowS, Toast.LENGTH_SHORT).show();
rCont=0; } } });
image2.setOnClickListener(new Button.OnClickListener() {
public void onClick(View v) {
SystemClock.elapsedRealtime(); } });
buttonOKBs = (Button) findViewById(R.id.buttonSave);
buttonOKBs.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
AMS=elapsedMillis;
CAC = AMS*Algorithm.CalorieActivity*Algorithm.Weight;
DecimalFormat dob = new DecimalFormat("###0");
txtvPay2 = (TextView)findViewById(R.id.textViews5);
String payShowS = String.valueOf(dob.format(CAC));
txtvPay2.setText(payShowS);
String aStringY = Integer.toString(mYear);
String aStringM = Integer.toString(mMonth);
String aStringD = Integer.toString(mDay);
double aY = Double.parseDouble(aStringY);
double aM = Double.parseDouble(aStringM);
double aD = Double.parseDouble(aStringD);
dbf1.addCalorieActivity(Algorithm.NActivity, AMS, CAC, aD, aM, aY);
Intent intent = new Intent(getApplicationContext(),mainmenu.class);
startActivity(intent); } });
buttonBack = (Button) findViewById(R.id.buttonBack);
buttonBack.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
Intent intent = new Intent(getApplicationContext(),mainmenu.class);
startActivity(intent); } });

```


ตัวอย่าง Code ปุ่ม Share

```

buttonExports.setOnClickListener(new OnClickListener() {
    public void onClick(View v) {
        Intent intent = new Intent(Intent.ACTION_SEND);
        intent.setType("text/plain");
        intent.putExtra(Intent.EXTRA_TEXT, "Share Activity"+Algorithm.NActivity+" Timer
        "+textTimeM+":"+EditInputTimeS+" Burn "+CAC);
        startActivity(Intent.createChooser(intent, "Share "+Algorithm.NActivity+" Timer
        "+textTimeM+":"+EditInputTimeS+" Burn "+CAC));});});

```

ตัวอย่าง Code ปุ่ม Profile

```

public void onCreate(Bundle savedInstanceState) {
    dbfl = new DBcontrolWeights(this);
    dbfl.open();
    super.onCreate(savedInstanceState);
    setContentView(R.layout.inprofile);
    setImageview();
    txtvPay1 = (TextView)findViewById(R.id.textViewHW);
    DecimalFormat df = new DecimalFormat("###0");
    String payShowH = String.valueOf(df.format(Algorithm.Height));
    String payShowW = String.valueOf(df.format(Algorithm.Weight));
    txtvPay1.setText(" Height : "+payShowH+" Weight : "+payShowW);
    DecimalFormat dob = new DecimalFormat("###0");
    txtvPay2 = (TextView)findViewById(R.id.textViewAS);
    String payShowA =String.valueOf(dob.format(Algorithm.Ago));
    String payShowS = Algorithm.Sex;
    txtvPay2.setText(" Age : "+payShowA+" Sex : "+payShowS);
    DecimalFormat dob1 = new DecimalFormat("###0.0");
    txtvPay3 = (TextView)findViewById(R.id.textView4);

```

```

String payShowPay31 = String.valueOf(dob1.format(Algorithm.LoseWeight));
txtvPay3.setText(" Lose Weight : "+payShowPay31+" Kilogram");
DecimalFormat dob2 = new DecimalFormat("###0");
txtvPay4 = (TextView)findViewById(R.id.textView5);
String payShowPay41 =String.valueOf(dob2.format(Algorithm.CalorieToDats));
String payShowPay42 = Algorithm.HealthStatus;
txtvPay4.setText(" Health Status : "+payShowPay42);
txtvPay5 = (TextView)findViewById(R.id.textView6);
txtvPay5.setText(" Burn Calories : "+payShowPay41+" Calorie ");
dbfl.open();
rEat =dbfl.intgetEat();
SEat=String.valueOf(dob2.format(rEat));
txtvPayE = (TextView)findViewById(R.id.textView2);
txtvPayE.setText(" Eat All: "+SEat+" Calorie ");
dbfl.open();
rBurn =dbfl.intgetBurn();
SBurn=String.valueOf(dob2.format(rBurn));
txtvPayA = (TextView)findViewById(R.id.textView3);
txtvPayA.setText(" Burn All: "+SBurn+" Calorie ");
if(Algorithm.DataLanguage.equals("ไทย")){
setText_Th();}
buttonBacks = (Button) findViewById(R.id.buttonBack);
buttonBacks.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
Intent intent = new Intent(getApplicationContext(),mainmenu.class);
startActivity(intent);});}
buttonEds = (Button) findViewById(R.id.buttonEd);
buttonEds.setOnClickListener(new OnClickListener() {
public void onClick(View v) {

```

```

Intent intent = new Intent(getApplicationContext(),editbasis.class);
startActivity(intent));});
buttonExports = (Button) findViewById(R.id.buttonExport);
buttonExports.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
Intent intent = new Intent(Intent.ACTION_SEND);
intent.setType("text/plain");
intent.putExtra(Intent.EXTRA_TEXT, "Share Profile ");
startActivity(Intent.createChooser(intent, "Share Profile")); });}
private void setImageView(){
imageView1 =(ImageView)findViewById(R.id.imageView1);
String sw = Algorithm.Sex;
if(sw.equals("Male")){
if(Algorithm.BMI<18.5){
imageView1.setImageResource(R.drawable.mtouw);
}else if(Algorithm.BMI>=18.5 && Algorithm.BMI<=24.9){
imageView1.setImageResource(R.drawable.mtonw);
}else if(Algorithm.BMI>=25 && Algorithm.BMI<=29.9){
imageView1.setImageResource(R.drawable.mtosow);
}else if(Algorithm.BMI>=30 && Algorithm.BMI<=39.9){
imageView1.setImageResource(R.drawable.mtoow);
}else if(Algorithm.BMI>=40 && Algorithm.BMI<=60){
imageView1.setImageResource(R.drawable.mtoeo); }
else{
imageView1.setImageResource(R.drawable.ic_launcher); }
}else if(sw.equals("Female")){
if(Algorithm.BMI<18.5){
imageView1.setImageResource(R.drawable.ftouw);
}else if(Algorithm.BMI>=18.5 && Algorithm.BMI<=24.9){

```

```

image1.setImageResource(R.drawable.ftonw);
}else if(Algorithm.BMI>=25 && Algorithm.BMI<=29){
image1.setImageResource(R.drawable.ftosow);
}else if(Algorithm.BMI>=30 && Algorithm.BMI<=39){
image1.setImageResource(R.drawable.ftoow);
}else if(Algorithm.BMI>=40 && Algorithm.BMI<=50){
image1.setImageResource(R.drawable.ftoeo);
}else{
image1.setImageResource(R.drawable.ic_launcher);
}}}

```

ตัวอย่าง Code ปุ่ม Result

```

public void onCreate(Bundle savedInstanceState) {
dbfl = new DBcontrolWeights(this);
dbfl.open();
super.onCreate(savedInstanceState);
setContentView(R.layout.historyandresult);
final Calendar c = Calendar.getInstance();
mYear = c.get(Calendar.YEAR);
mMonth = c.get(Calendar.MONTH);
mDay = c.get(Calendar.DAY_OF_MONTH);
String aStringY = Integer.toString(mYear);
String aStringM = Integer.toString(mMonth);
String aStringD = Integer.toString(mDay);
double aY = Double.parseDouble(aStringY);
double aM = Double.parseDouble(aStringM);
double aD = Double.parseDouble(aStringD);
int pD= (int) aD;
int pM= (int) aM;

```

```

int pY= (int) aY;
showLine(pD,pM,pY);
Algorithm.sd = dbfl.OpenDataShowWeight();
buttonEp = (Button) findViewById(R.id.button2);
buttonEp.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
writeToFileE();
writeToFileA();
writeToFileW();}});
buttonGarphs = (Button) findViewById(R.id.buttonChart);
buttonGarphs.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
Intent intent = new Intent(getApplicationContext(),BarChartGraph.class);
startActivity(intent);}});
buttonCancels = (Button) findViewById(R.id.buttonBacks);
buttonCancels.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
Intent intent = new Intent(getApplicationContext(),mainmenu.class);
startActivity(intent);    }});
buttonAlls = (Button) findViewById(R.id.buttonAll);
buttonAlls.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
AllDataShow();}});
buttonDtoD = (Button) findViewById(R.id.buttonDtoD);
buttonDtoD.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
showDatePicker();}});
if(Algorithm.DataLanguage.equals("ไทย")){
setText_Th();}

```

ตัวอย่าง Code 1 ม Chart

```

public class LineGraph{
    public Intent getIntent(Context context) {
        // BMI h
        Algorithm.setBMIWeight18();
        Algorithm.setBMIWeight24();
        Algorithm.setDayLoseWeight(75, 65, 0.7);
        int[] AxisX = new int[Algorithm.DayLoseWeight];
        double[] Axisy1 = new double[Algorithm.DayLoseWeight];
        double[] Axisy2 = new double[Algorithm.DayLoseWeight];
        double[] Axisy3 = new double[Algorithm.DayLoseWeight];
        double[] Axisy4 = Algorithm.sd;
        for(int iic=0;iic<Algorithm.DayLoseWeight;iic++){
            AxisX[iic]=iic; }
        for(int iiy1=0;iiy1<Algorithm.DayLoseWeight;iiy1++){
            Axisy1[iiy1]=Algorithm.BMIWeight24; }
        for(int iiy2=0;iiy2<Algorithm.DayLoseWeight;iiy2++){
            Axisy2[iiy2]=Algorithm.BMIWeight18; }
        for(int iiy3=0;iiy3<Algorithm.DayLoseWeight;iiy3++){
            if(iiy3==0){
                Axisy3[iiy3]=75;
            }else if(iiy3>0){
                Axisy3[iiy3]=Axisy3[iiy3-1]-(0.7/7);
            }
        }
        TimeSeries series1 = new TimeSeries("BMI42");
        for( int i1 = 0; i1 < AxisX.length; i1++)
            {series1.add(AxisX[i1], Axisy1[i1]);}
        // BMI l
        TimeSeries series2 = new TimeSeries("BMI18");
    }
}

```

```

for( int i2 = 0; i2 < AxisX.length; i2++)
{
series2.add(AxisX[i2], Axisy2[i2]);
}
TimeSeries series3 = new TimeSeries("LoseWeight");
for( int i3 = 0; i3 < AxisX.length; i3++)
{
series3.add(AxisX[i3], Axisy3[i3]);
}
TimeSeries series4 = new TimeSeries("LoseWeight");
for( int i4 = 0; i4 < Axisy4.length; i4++)
{
series4.add(AxisX[i4], Axisy4[i4]);
}
XYMultipleSeriesDataset dataset = new XYMultipleSeriesDataset();
dataset.addSeries(series1);
dataset.addSeries(series2);
dataset.addSeries(series3);
XYMultipleSeriesRenderer mRenderer = new XYMultipleSeriesRenderer(); // Holds a collection of
XYSeriesRenderer and customizes the graph
XYSeriesRenderer renderer1 = new XYSeriesRenderer(); // This will be used to customize line 1
XYSeriesRenderer renderer2 = new XYSeriesRenderer(); // This will be used to customize line 2
XYSeriesRenderer renderer3 = new XYSeriesRenderer();
XYSeriesRenderer renderer4 = new XYSeriesRenderer();
mRenderer.setBackgroundColor(Color.DKGRAY);
mRenderer.addSeriesRenderer(renderer1);
mRenderer.addSeriesRenderer(renderer2);
mRenderer.addSeriesRenderer(renderer3);
mRenderer.addSeriesRenderer(renderer4);

```

```

// Customization time for line 1!
renderer1.setColor(Color.RED);
renderer1.setPointStyle(PointStyle.SQUARE);
// Customization time for line 2!
renderer2.setColor(Color.RED);
renderer2.setPointStyle(PointStyle.DIAMOND);
renderer2.setFillPoints(true);
// Customization time for line 3!
renderer3.setColor(Color.BLUE);
renderer3.setPointStyle(PointStyle.POINT);
renderer3.setFillPoints(true);
/// Customization time for line 4!
renderer4.setColor(Color.GRAY);
renderer4.setPointStyle(PointStyle.POINT);
renderer4.setFillPoints(true);

Intent intent = ChartFactory.getLineChartIntent(context, dataset, mRenderer, "Line Graph Title");
return intent; }}

```

ตัวอย่าง Code ปุ่ม Export

```

private void writeToFileW(){
try {
OutputStreamWriter outputStreamWriter = new
OutputStreamWriter(openFileOutput("myWeight.txt", Context.MODE_PRIVATE));
dbfl.OpenDatatoWriterWeight();
String[] writeWeight = dbfl.OpenDatatoWriterWeight();
for(int iiy=0;iiy<writeWeight.length;iiy++){
String towriteing = writeWeight[iiy]+" \n";
outputStreamWriter.write(towriteing);
}
}
}

```



```

}
outputStreamWriter.close();
catch (IOException e) {
Log.e(TAG, "File write failed: " + e.toString());}
private void writeToFileE(){
try {
OutputStreamWriter outputStreamWriter = new OutputStreamWriter(openFileOutput("myEat.txt",
Context.MODE_PRIVATE));
dbfl.OpenDatatoWriterWeight();
String[] writeWeight = dbfl.OpenDatatoWriterEat();
for(int iiy=0;iiy<writeWeight.length;iiy++){
String towriteing = writeWeight[iiy]+" \n";
outputStreamWriter.write(towriteing);}
outputStreamWriter.close();
}
catch (IOException e) {Log.e(TAG, "File write failed: " + e.toString()); } }
private void writeToFileA(){
try {
OutputStreamWriter outputStreamWriter = new OutputStreamWriter(openFileOutput("myBurn.txt",
Context.MODE_PRIVATE));
dbfl.OpenDatatoWriterWeight();
String[] writeWeight = dbfl.OpenDatatoWriterBurn();
for(int iiy=0;iiy<writeWeight.length;iiy++){
String towriteing = writeWeight[iiy]+" \n";
outputStreamWriter.write(towriteing+" \n");}
outputStreamWriter.close();
catch (IOException e) {
Log.e(TAG, "File write failed: " + e.toString());}

```

ตัวอย่าง Code ปุ่ม Date To Date

```
public Dialog onCreateDialog(Bundle savedInstanceState) {  
    final Calendar c = Calendar.getInstance();  
    int year = c.get(Calendar.YEAR);  
    int month = c.get(Calendar.MONTH);  
    int day = c.get(Calendar.DAY_OF_MONTH);  
    return new DatePickerDialog(getActivity(), this, year, month, day); }  
public void onDateSet(DatePicker view, int year, int month, int day) {  
    Algorithm.pYear = year;  
    Algorithm.pDay = day;  
    Algorithm.pMonth = month; }}
```

