

แอปพลิเคชันสำหรับการจัดการข้อมูลการเงินส่วนบุคคล
APPLICATION FOR MANAGE PERSONAL FINANCIAL

นายฐากร ลีนะกุล รหัส 50370271
นายอริการ กอบก่อง รหัส 50371384

ปริญญานิพนธ์นี้ เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยมหิดล
ปีการศึกษา 2554

ห้องสมุดคณะวิศวกรรมศาสตร์
รับที่วัน..... 2 ก.ค. 2556
เลขทะเบียน..... 16281473
เลขเรียกหนังสือ..... ปร.
มหาวิทยาลัยมหิดล 162 @

๗

2554

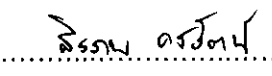


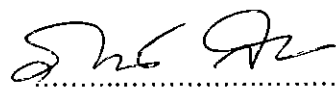
ใบรับรองปริญญาโท

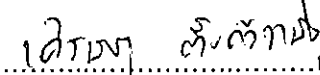
ชื่อหัวข้อโครงการ แอปพลิเคชันสำหรับการจัดการข้อมูลการเงินส่วนบุคคล
ผู้ดำเนินโครงการ นายฐากร ลีนะกุล รหัส 50370271
นายอชิการ กอบกอง รหัส 50371384
ที่ปรึกษาโครงการ อาจารย์เศรษฐา ตั้งคำวานิช
สาขาวิชา วิศวกรรมคอมพิวเตอร์
ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา 2554

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญาโทฉบับนี้เป็นส่วนหนึ่ง
ของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์


.....กรรมการ
(อาจารย์สุรเดช จิตประไพกุลศาล)


.....กรรมการ
(อาจารย์สิรภพ คชรรัตน์)


.....กรรมการ
(อาจารย์ภาณุพงศ์ สอนคม)


.....กรรมการ
(อาจารย์เศรษฐา ตั้งคำวานิช)

Project title Application for Manage Personal Financial

Name Mr. Tharkun Lccnakun ID. 50370271

 Mr. Athigarn Gobgong ID. 50371384

Project advisor Mr. Settha Thangkawanit

Major Computer Engineering

Department Electrical and Computer Engineering

Academic year 2011

.....

Abstract

In the present, Spending is routine for people and many families do an accounting income system for each a day for planning money how we spending ourselves but record must be the last time each a day because it is not convenient to write on a book or a computer so we set the accounting income system for the man who has an android system for recording accounting income system easily just has a telephone.

The project is the creating android program system for recording accounting income system and then process information for a user how to record for accounting income , planning how to use money easily and we can check statistics accounting income system our spending.

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จได้ด้วยดีเนื่องจาก อาจารย์เศรษฐา ตั้งค้ำวานิช ที่ได้ให้ความรู้ ให้คำปรึกษาและคำแนะนำในส่วนวิธีการและรูปแบบของโครงการที่ควรจะเป็น รวมทั้งช่วยตรวจสอบข้อผิดพลาดของโครงการ จึงขอขอบพระคุณมา ณ โอกาสนี้

ขอขอบพระคุณ อ.ภาณุพงศ์ สอนคม และ อ.สิริภพ กษรัตน์ และ อ.สุรเดช จิตประไพกุล ศาสตราจารย์ที่ให้คำแนะนำและคำปรึกษาในการปรับปรุงและแก้ไขโครงการให้ดียิ่งขึ้น จนทำให้โครงการนี้สำเร็จลุล่วงไปได้ด้วยดี

สุดท้ายขอขอบคุณเพื่อนๆทุกคนที่ให้คำแนะนำและคำปรึกษา รวมถึงให้กำลังใจ ผู้จัดทำโครงการนี้ขอขอบพระคุณทุกท่านไว้ ณ โอกาสนี้ด้วย



คณะผู้จัดทำ

นายฐาณกร ถิ่นะกุล

นายอริการ กอบก่อง

สารบัญ

หน้า

ใบรับรองโครงการวิจัย.....	ก
บทคัดย่อภาษาไทย.....	ข
บทคัดย่อภาษาอังกฤษ.....	ค
กิตติกรรมประกาศ.....	ง
สารบัญ.....	จ
สารบัญตาราง.....	ช
สารบัญรูป.....	ฉ
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบเขตของโครงการ.....	1
1.4 แผนการดำเนินงาน.....	2
1.5 ขั้นตอนการดำเนินงาน.....	2
1.6 ผลที่คาดว่าจะได้รับ.....	3
1.7 งบประมาณที่ใช้.....	3
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง.....	4
2.1 ระบบปฏิบัติการแอนดรอยด์.....	4
2.1.1 โครงสร้างโปรเจกแอนดรอยด์.....	6
2.1.2 ส่วนประกอบของแอนดรอยด์.....	7
2.1.3 โครงสร้างทางซอฟต์แวร์แอนดรอยด์.....	7
2.2 ระบบฐานข้อมูล.....	9

สารบัญ(ต่อ)

	หน้า
2.2.1 ชนิดของข้อมูลที่ใช้ในฐานข้อมูล.....	9
2.2.2 การตั้งชื่อตาราง.....	10
2.2.3 ฟังก์ชัน.....	13
2.2.4 การใช้งานกลุ่มคำสั่ง SQL.....	16
2.3 การพยากรณ์รายจ่าย.....	17
2.3.1 ค่าเฉลี่ยเคลื่อนที่.....	17
2.3.2 การปรับให้เรียบแบบเอกซ์โปเนนเชียลครึ่งเดียว.....	19
บทที่ 3 วิเคราะห์ ออกแบบ และพัฒนาระบบ.....	21
3.1 การวางแผนการทำงาน.....	21
3.2 การออกแบบโครงสร้างแอปพลิเคชัน.....	22
3.2.1 การออกแบบเมนูรายรับ.....	22
3.2.2 การออกแบบเมนูรายจ่าย.....	23
3.2.3 การออกแบบเมนูเรียกดูข้อมูล.....	23
3.2.4 การออกแบบเมนูลบข้อมูล.....	24
3.2.5 การออกแบบเมนูเรียกดูกราฟแท่งรายปี.....	24
3.2.6 การออกแบบเมนูพยากรณ์รายจ่าย.....	25
3.3 Use Case Diagram ของแอปพลิเคชัน.....	26
3.4 Class Diagram ของแอปพลิเคชัน.....	26
3.5 การออกแบบฐานข้อมูลที่ Input เข้ามา.....	27
3.5.1 การออกแบบฐานข้อมูลของแอปพลิเคชัน.....	27
3.6 ตัวอย่างการจำลองตารางการใช้จ่ายภายใน 1 สัปดาห์ที่จะนำไปเก็บที่ฐานข้อมูล.....	28
3.6.1 ตัวอย่างการจำลองตารางการใช้จ่ายภายใน 1 วันที่จะนำไปเก็บที่ฐานข้อมูล.....	28
บทที่ 4 การทดสอบระบบ.....	30
4.1 การเพิ่มข้อมูลในฐานข้อมูล.....	30
4.1.1 การเพิ่มข้อมูลรายรับ.....	31
4.1.2 การเพิ่มข้อมูลรายจ่าย.....	32

สารบัญ(ต่อ)

	หน้า
4.2 การลบข้อมูลในฐานข้อมูล.....	33
4.3 การเรียกดูข้อมูลรายรับ-รายจ่าย.....	34
4.3.1 การดูข้อมูลรายรับ รายจ่าย ในรูปแบบกราฟวงกลม.....	35
4.3.2 การนำข้อมูลรายรับ รายจ่าย บันทึกลงไฟล์ .txt	36
4.4 การดูข้อมูลรายรับ รายจ่าย ต่อปีในรูปแบบกราฟแท่ง.....	37
4.4.1 การดูข้อมูลรายจ่ายแยกประเภท ต่อปีรูปแบบกราฟแท่ง.....	38
4.5 การใช้เมนูพยากรณ์รายจ่าย.....	44
บทที่ 5 สรุปผลการทดลอง.....	50
5.1 สรุปผลโครงการ.....	51
5.2 ปัญหาที่พบ.....	51
5.3 แนวทางในการพัฒนาเพิ่มเติมและข้อเสนอแนะ.....	52
5.3.1 แนวทางแก้ไขปัญหาเพื่อพัฒนาเพิ่มเติม.....	52
5.3.2 ข้อเสนอแนะและคุณสมบัติของซอฟต์แวร์ในอนาคต.....	52
5.3.3 ทักษะและความรู้ที่สำคัญต่อการต่อยอดโครงการ.....	53
เอกสารอ้างอิง.....	54
ภาคผนวก.....	55
ก. การตั้งค่าเครื่องมือที่ใช้ในการพัฒนาโปรแกรม.....	55
ประวัติผู้เขียน.....	62

สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางแสดงขั้นตอนการดำเนินการ.....	2
2.1 ตารางคุณสมบัติของ Android	5
2.2 ตารางชื่อเฉพาะของภาษา SQL ที่ไม่สามารถนำไปสร้างเป็นชื่อตัวแปรได้.....	10
2.3 ตารางตัวอย่างการหาค่าเฉลี่ยด้วยวิธีหาค่าเฉลี่ยเคลื่อนที่ (Moving averages).....	18
3.1 ตัวอย่างการจำลองตารางการใช้จ่ายภายใน 1 สัปดาห์ที่จะนำไปเก็บที่ฐานข้อมูล.....	28
3.2 ตัวอย่างการจำลองตารางการใช้จ่ายภายใน 1 วันที่จะนำไปเก็บที่ฐานข้อมูล.....	28
4.1 จำลองค่าใช้จ่ายในแต่ละเดือน (1).....	44
4.2 จำลองค่าใช้จ่ายในแต่ละเดือน (2).....	46
4.3 จำลองค่าใช้จ่ายในแต่ละเดือน (3).....	47
5.1 ตารางแสดงปัญหาที่พบและแนวทางการแก้ไขปัญหา.....	51

สารบัญรูป

รูปที่	หน้า
2.1 แสดงโครงสร้างพื้นฐานแอนครอยด์.....	6
2.2 โครงสร้างทางซอฟต์แวร์แอนครอยด์.....	7
3.1 แสดง Flow chart ขั้นตอนการทำงาน.....	21
3.2 ออกแบบเมนูต่างๆ ในแอปพลิเคชัน.....	22
3.3 ออกแบบเมนูรายรับ.....	22
3.4 การออกแบบเมนูรายจ่าย.....	23
3.5 การออกแบบเมนูเรียกดูข้อมูล.....	23
3.6 การออกแบบเมนูลบข้อมูล.....	24
3.7 การออกแบบเมนูเรียกดูกราฟแท่งรายปี.....	24
3.8 การออกแบบเมนูพยากรณ์รายจ่าย.....	25
3.9 Use Case Diagram ของแอปพลิเคชัน.....	26
3.10 Class Diagram ของแอปพลิเคชัน.....	26
3.11 การออกแบบฐานข้อมูลที่ Input เข้ามา.....	27
3.12 การออกแบบฐานข้อมูลของแอปพลิเคชัน.....	27
4.1 รูปแสดงเมนูต่างๆของแอปพลิเคชัน.....	30
4.2 แสดงการเพิ่มข้อมูลรายรับ.....	31
4.3 แสดงการบันทึกรายรับเรียบร้อยแล้ว.....	31
4.4 แสดงการแจ้งเตือนให้ใส่เงินเดือน.....	31
4.5 แสดงการจำแนกประเภทของรายจ่าย.....	32
4.6 แสดงข้อมูลรายรับ รายจ่ายที่มีก่อนทำการลบข้อมูล.....	33
4.7 แสดงการลบข้อมูล.....	33
4.8 รูปหลังจากลบข้อมูลเรียบร้อยแล้ว.....	33
4.9 แสดง Report ที่ user เรียกดู.....	34
4.10 แสดงค่าใช้จ่ายของแต่ละประเภทรูปแบบกราฟวงกลม.....	35

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.11 เรียกดูข้อมูลก่อนการบันทึก.....	36
4.12 ใส่ชื่อ Text File ที่ต้องการบันทึก.....	36
4.13 Text File บันทึกเรียบร้อยแล้ว.....	36
4.14 รูปแบบ Text File ที่ได้รับ.....	36
4.15 กราฟแท่ง รายรับ รายจ่ายต่อปี.....	37
4.16 กราฟแท่ง รายรับ รายจ่ายต่อปีแบบชุม.....	37
4.17 กราฟแท่งค่าอาหาร ต่อปี.....	38
4.18 กราฟแท่งค่าอาหาร ต่อปีแบบชุม.....	38
4.19 กราฟแท่งค่ารักษาพยาบาล ต่อปี.....	39
4.20 กราฟแท่งค่ารักษาพยาบาลต่อปีแบบชุม.....	39
4.21 กราฟแท่งค่าพาหนะ ต่อปี.....	40
4.22 กราฟแท่งค่าพาหนะต่อปีแบบชุม.....	40
4.23 กราฟแท่งค่าเสื้อผ้า ต่อปี.....	41
4.24 กราฟแท่งค่าเสื้อผ้า ต่อปีแบบชุม.....	41
4.25 กราฟแท่งค่าใช้จ่ายภายในบ้าน ต่อปี.....	42
4.26 กราฟแท่งค่าใช้จ่ายภายในบ้านต่อปีแบบชุม.....	42
4.27 กราฟแท่งค่าใช้จ่ายอื่นๆ ต่อปี.....	43
4.28 กราฟแท่งค่าใช้จ่ายอื่นๆต่อปีแบบชุม.....	43
4.29 เมนูพยากรณ์รายจ่าย.....	44
4.30 การใช้เมนูพยากรณ์รายจ่ายด้วยวิธี Moving Averaging (1).....	45
4.31 การใช้เมนูพยากรณ์รายจ่ายด้วยวิธี Exponential (1).....	45
4.32 กราฟแสดงการเปรียบเทียบระหว่างค่าใช้จ่ายจริงกับการพยากรณ์ค่าใช้จ่าย (1).....	45
4.33 การใช้เมนูพยากรณ์รายจ่ายด้วยวิธี Moving Averaging (2).....	46
4.34 การใช้เมนูพยากรณ์รายจ่ายด้วยวิธี Exponential (2).....	46
4.35 กราฟแสดงการเปรียบเทียบระหว่างค่าใช้จ่ายจริงกับการพยากรณ์ค่าใช้จ่าย (2).....	47

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.36 การใช้เมนูพยากรณ์รายจ่ายด้วยวิธี Moving Averaging (3).....	48
4.37 การใช้เมนูพยากรณ์รายจ่ายด้วยวิธี Exponential (3).....	48
4.38 กราฟแสดงการเปรียบเทียบระหว่างค่าใช้จ่ายจริงกับการพยากรณ์ค่าใช้จ่าย (3).....	48
รูปที่ภาคผนวก1 การติดตั้ง Eclipse 1	53
รูปที่ภาคผนวก2 การติดตั้ง Eclipse 2	54
รูปที่ภาคผนวก3 การติดตั้ง Eclipse 3	54
รูปที่ภาคผนวก4 การติดตั้ง Eclipse 4	55
รูปที่ภาคผนวก5 การติดตั้ง Eclipse 5	55
รูปที่ภาคผนวก6 การติดตั้ง Eclipse 6	56
รูปที่ภาคผนวก7 การติดตั้ง Eclipse 7	56
รูปที่ภาคผนวก8 การติดตั้ง Eclipse 8	57
รูปที่ภาคผนวก9 การติดตั้ง Eclipse 9	57
รูปที่ภาคผนวก10 การติดตั้ง Eclipse 10	58
รูปที่ภาคผนวก11 การติดตั้ง Eclipse 11	58
รูปที่ภาคผนวก12 การติดตั้ง Eclipse 12	59
รูปที่ภาคผนวก13 การติดตั้ง Eclipse 13	59

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

การบริหารการเงินส่วนบุคคลมีความสำคัญต่อชีวิตของคนทุกคนเพื่อให้สามารถจัดการรายรับ รายจ่ายให้มีความเหมาะสมต่อการดำรงชีวิตได้ การบันทึกการเงินส่วนบุคคล เพื่อเก็บไว้วิเคราะห์นั้นย่อมเป็นประโยชน์ต่อผู้บันทึกทั้งทางตรงและทางอ้อมแต่ในแต่ละวันการใช้จ่ายจะเกิดขึ้นได้ทุกที่ทุกเวลาและทุกสถานที่ แล้วเราจะบันทึกข้อมูลการเงินส่วนบุคคลได้อย่างไรในยุคนี้สิ่งที่เราพาคิดตัวอยู่เสมอก็คือ โทรศัพท์มือถืออีกทั้งวงการมือถือในปัจจุบันมีโทรศัพท์กลุ่มที่เรียกว่า สมาร์ทโฟน ซึ่งมือถือทำอะไรได้มากกว่า การโทรเข้าโทรออกโดยสามารถเข้าถึงบริการต่างๆบนอินเทอร์เน็ตผ่านแอปพลิเคชันหรือโปรแกรมบนสมาร์ตโฟนทำให้โทรศัพท์มือถือในกลุ่ม สมาร์ตโฟนเป็นอะไรที่ดึงดูดผู้ใช้งานมือถือที่ต้องการอะไรที่ใหม่ๆ สำหรับจัดการข้อมูลการเงินส่วนบุคคล บน โทรศัพท์มือถือที่มีระบบปฏิบัติการแอนดรอยด์เพื่อสร้างความสะดวกสบายและรวดเร็วในทุกๆที่ที่มีการใช้จ่ายของผู้ใช้

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อเรียนรู้การเขียน โปรแกรมบนระบบปฏิบัติการแอนดรอยด์
- 1.2.2 เพื่อเรียนรู้พัฒนา SQLite
- 1.2.3 เพื่อให้ผู้ใช้สามารถดูสถิติของระบบรายรับ-รายจ่ายได้
- 1.2.4 เพื่อความสะดวกต่อการบันทึกรายรับ-รายจ่าย บนมือถือระบบปฏิบัติการแอนดรอยด์

1.3 ขอบเขตของโครงการ

- 1.3.1 โปรแกรมสามารถบันทึกข้อมูลรายรับ-รายจ่ายได้
- 1.3.2 โปรแกรมสามารถเรียกดูรายละเอียดของข้อมูลรายรับ-รายจ่ายได้ ซึ่งประกอบด้วย
 - วัน เดือน ปี ที่มีรายรับและรายจ่าย
 - ประเภทของรายจ่าย
 - ราคาและรายละเอียดของรายจ่าย
 - ยอดเงินคงเหลือ
- 1.3.3 โปรแกรมสามารถแสดงสถิติข้อมูลรายรับ-รายจ่ายในรูปแบบกราฟวงกลมและกราฟแท่งได้
- 1.3.4 โปรแกรมสามารถส่งออก (Export) ข้อมูลเป็นไฟล์ .txt ได้
- 1.3.5 โปรแกรมสามารถพยากรณ์ค่าใช้จ่ายในอนาคตได้โดยใช้ฐานข้อมูลเก่าเป็นตัวบ่งชี้

1.4 แผนการดำเนินงาน

- 1.4.1 ศึกษาการทำบัญชีระบบรายรับ-รายจ่าย และความต้องการของผู้ใช้
- 1.4.2 ศึกษาการเขียน โปรแกรมบนระบบปฏิบัติการแอนดรอยด์และ SQLite
- 1.4.3 นำข้อมูลที่ได้มาใช้ในการออกแบบโปรแกรม
- 1.4.4 พัฒนาโปรแกรมที่ได้ออกแบบไว้
- 1.4.5 ทดสอบโปรแกรมเพื่อหาข้อผิดพลาดและแก้ไขโปรแกรม
- 1.4.6 เขียนคู่มือการใช้งานและสรุปผลของโครงการ

1.5 ขั้นตอนการดำเนินงาน

ตารางที่ 1.1 ตารางแสดงขั้นตอนการดำเนินการ

ขั้นตอนการดำเนินงาน	ปีการศึกษา 2554									
	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	
ศึกษาการทำบัญชีระบบรายรับ-รายจ่าย และความต้องการของผู้ใช้										
ศึกษาค้นคว้า ระบบปฏิบัติการแอนดรอยด์ และ SQLite										
นำข้อมูลที่ได้มาใช้ในการออกแบบ โปรแกรม										
พัฒนาโปรแกรมที่ได้ออกแบบไว้										
ทดสอบโปรแกรมเพื่อหาข้อผิดพลาดและแก้ไขโปรแกรม										
เขียนคู่มือการใช้งานและสรุปผลของโครงการ										

1.6 ผลที่คาดว่าจะได้รับ

1.6.1 ได้รับความรู้ในเรื่องระบบปฏิบัติการแอนดรอยด์

1.6.2 ได้รับความรู้ในส่วนขอ SQLite Database

1.6.3 ได้เรียนรู้เทคนิคการประยุกต์ใช้ โลกบริโอเพนซอร์สต่างๆ

1.6.4 ได้รับโปรแกรมโทรศัพท์มือถือที่ทำงานบนระบบปฏิบัติการแอนดรอยด์ ซึ่งมีฟังก์ชันต่างๆตามที่ได้กำหนดไว้

1.7 งบประมาณที่ใช้

1.7.1 ค่าหนังสือ	1,000 บาท
1.7.2 ค่าถ่ายเอกสารและจัดทำรูปเล่ม	500 บาท
1.7.3 ค่าอุปกรณ์	300 บาท
1.7.4 ค่าหมึกพิมพ์, แผ่น DVD	200 บาท
รวม	2,000 บาท

หมายเหตุ ถัวเฉลี่ยทุกรายการ



บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

การพัฒนาแอปพลิเคชันสำหรับการจัดการข้อมูลการเงินส่วนบุคคลจำเป็นต้องศึกษาว่าแต่ละส่วนต้องใช้อะไรมาเป็นตัวดำเนินการ เช่น การสร้างฐานข้อมูลเราจะใช้ SQL เป็นภาษาในการสร้างฐานข้อมูล ดังนั้นเราจึงต้องศึกษาข้อมูลและเรียนรู้เกี่ยวกับ ตัวดำเนินการเหล่านี้

2.1 ระบบปฏิบัติการแอนดรอยด์ [1]

ระบบปฏิบัติการแอนดรอยด์ คือ ระบบปฏิบัติการ (OS) หรือแพลตฟอร์ม ที่จะใช้ควบคุมการทำงานบนอุปกรณ์อิเล็กทรอนิกส์ต่างๆ สำหรับโทรศัพท์มือถือและอุปกรณ์พกพา โดยมี กูเกิล อิงก์, ที-โมบาย, เอชทีซี, ควอลคอมม์, โมโตโรลา และบริษัทชั้นนำอีกมากมายร่วมพัฒนา โปรเจกต์แอนดรอยด์ ผ่านกลุ่มพันธมิตรเครื่องมือถือระบบเปิด (Open Handset Alliance) ซึ่งเป็นกลุ่มพันธมิตรชั้นนำระดับนานาชาติด้านเทคโนโลยีและเครื่องมือถือ สารเคลื่อนที่ ซึ่งแอนดรอยด์ประกอบด้วยระบบปฏิบัติการ ไลบรารี เฟรมเวิร์ค และซอฟต์แวร์อื่นๆ ที่จำเป็นในการพัฒนา ซึ่งเทียบเท่ากับ Windows Mobile, Palm OS, Symbian, OpenMoko และ Maemo ของโนเกีย โดยใช้อองค์ประกอบที่เป็นโอเพนซอร์สหลายอย่าง เช่น Linux Kernel, SSL, OpenGL, FreeType, SQLite, WebKit

ประเภทของชุดซอฟต์แวร์ เนื่องจากแอนดรอยด์นั้นเปิดให้นักพัฒนาเข้าไปชมรหัสต้นฉบับได้ ทำให้มีผู้พัฒนาจากหลายฝ่ายนำเอารหัสต้นฉบับมาปรับแต่ง และสร้างแอนดรอยด์ในแบบฉบับของตนเองขึ้น เราจึงแบ่งประเภทของแอนดรอยด์ออกได้เป็น 3 ประเภท ดังต่อไปนี้

1. Android Open Source Project (AOSP) เป็น แอนดรอยด์ประเภทแรกที่ถูกเปิดให้สามารถนำ “ต้นฉบับแบบเปิด” ไปติดตั้งและใช้งานในอุปกรณ์ต่างๆ ได้โดยไม่ต้องเสียค่าใช้จ่าย

2. Open Handset Mobile (OHM) เป็น แอนดรอยด์ที่ได้รับการพัฒนาร่วมกับกลุ่มบริษัทผู้ผลิตอุปกรณ์พกพา ที่เข้าร่วมกับกูเกิลในนาม Open Handset Alliances (OHA) ซึ่งบริษัทเหล่านี้จะพัฒนาแอนดรอยด์ในแบบฉบับของตนเองออกมา โดยรูปร่างหน้าตาการแสดงผล และฟังก์ชันการใช้งาน จะมีความเป็นเอกลักษณ์ และมีลิขสิทธิ์เป็นของตนเอง พร้อมได้รับสิทธิ์ในการมีบริการเสริมต่าง ๆ จากกูเกิล ที่เรียกว่า Google Mobile Service (GMS) ซึ่งเป็นบริการเสริมที่ทำให้แอนดรอยด์มีประสิทธิภาพ เป็นไปตามจุดประสงค์ของแอนดรอยด์ แต่การจะได้มาซึ่ง GMS นั้น ผู้ผลิตจะต้องทำการทดสอบระบบ และขออนุญาตกับทางกูเกิลก่อน จึงจะนำเครื่องออกสู่ตลาดได้

3. Cooking หรือ Customize เป็นแอนดรอยด์ที่นักพัฒนานำเอารหัสต้นฉบับจากแหล่งต่างๆ มาปรับแต่ง ในแบบฉบับของตนเอง โดยจะต้องทำการปลดล็อคสิทธิ์การใช้งานอุปกรณ์ หรือ Unlock เครื่องก่อน จึงจะสามารถติดตั้งได้ โดยแอนดรอยด์ประเภทนี้ถือเป็นประเภทที่มีความสามารถมากที่สุด เท่าที่อุปกรณ์เครื่องนั้น ๆ จะรองรับได้ เนื่องจากได้รับการปรับแต่งให้เข้ากับอุปกรณ์นั้นๆจากผู้ใช้งานจริง

ตารางที่ 2.1 ตารางคุณสมบัติของแอนดรอยด์ [2]

Handset layouts	การปรับขนาด VGA, 2D graphic library, 3D graphic ไลบรารีพื้นฐานของบน OpenGL ES. 2.0 คุณสมบัติทั่วไปของสมาร์ตโฟน
Storage	Database software ที่ใช้ SQLite เป็นตัวเก็บข้อมูล
การเชื่อมต่อ	แอนดรอยด์ การเชื่อมต่อ GSM/EDGE, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi,LTE, WIMAX
ข้อความ	MMS และ SMS และ Android Cloud to Device Messaging Framework (C2DM)
Web browser	Web browser มีใช้ใน แอนดรอยด์มีพื้นฐานการพัฒนา มาจาก open-source Web Kit application framework ร่วมกับ V8 JavaScript engine ของ chrome
Java support	Software ที่เขียนด้วย Java สามารถ compile และexecute ใน Dalvik virtual machine ซึ่ง เป็น การนำ VM implementation แบบพิเศษ ที่ ออกแบบ มา สำหรับ การ ใช้ โทรศัพท์ เคลื่อนที่ ถึงแม้ว่ามันจะไม่ได้เทคนิค มาตรฐาน ของ Java Virtual Machine ก็ตาม สนับสนุน J2ME ในรูปแบบของ third-party-applications.
Media support	แอนดรอยด์สนับสนุน media video audio จากพวกH.263, H.264 (in 3GP or MP4 container), MPEG-4 SP, AMR, AMR-WB (in 3GP container), AAC, HE-AAC (in MP4 or 3GP container), MP3, MIDI, OGG Vorbis, WAV, JPEG, PNG, GIF, BMP.
Additional hardware support	แอนดรอยด์สามารถใช้ วิดีโอ / กล้อง ยัง คง touch screen, GPS, accelerometers, magnetometers, accelerate 2D (มีการปรับฮาร์ดแวร์เปลี่ยนแปลงรูปแบบ pixel) และ เร่ง กราฟิก 3D.
Development environment	รวม Device emulator, เครื่องมือ สำหรับ debugging, memory และประวัติการทำงาน plug-in สำหรับ Eclipse IDE.

2.1.1 โครงสร้างโปรเจกแอนดรอยด์ [3]

โครงสร้างโปรเจกแอนดรอยด์จะมีโครงสร้างพื้นฐานของโปรเจกดังนี้



รูปที่ 2.1 แสดงโครงสร้างพื้นฐานแอนดรอยด์

1. src (Source) เป็นส่วนของ Source code ที่เราสร้างขึ้นและเขียนเป็น ไฟล์ .java ซึ่งจะเห็นชื่อคลาส แอดทริบิวต์ และเมธอด
2. gen (generated Java Files) เป็นส่วนที่ถูกสร้างขึ้นอัตโนมัติ เช่น ไฟล์ R.java ซึ่งภายในประกอบด้วย Text และ UI Element โดยถูกนำมาโปรเจกผ่าน Android Plug-in ซึ่งไฟล์นี้จะเหมือน pointer ไปยัง drawable, layout, value directory
3. res (Resource) จะเป็นส่วนของการแสดงผล ของไฟล์อื่นๆ ที่นำมาเป็นส่วนประกอบร่วมกับโค้ดที่เราเขียนขึ้น โดยแอนดรอยด์จะมีการแบ่งไฟล์เคอร์ไว้ตามชนิดของ resource นั้น โดยตอนที่เราสร้างโปรเจกขึ้นมาใหม่จะ directory ย่อยๆอีก 5 directory คือ
 - drawable-hdpi สำหรับเก็บภาพ
 - drawable-ldpi สำหรับเก็บภาพ
 - drawable-mdpi สำหรับเก็บภาพ
 - layout จัดวาง view ต่างๆ
 - values จัดการค่าต่างๆที่นำไปใช้ในแอปพลิเคชัน
4. AndroidManifest.xml เป็น โครงสร้างของ xml ไฟล์ ซึ่งใน xml นั้นจะเป็นการกำหนดคุณสมบัติและการตั้งค่าต่างๆของแอปพลิเคชัน เช่น ชื่อของแอปพลิเคชัน, เวอร์ชันของโค้ด, การกำหนดสิทธิ์ต่างๆ ในการเข้าถึงของแอปพลิเคชันและอื่นๆ

2.1.2 ส่วนประกอบของแอนดรอยด์

ส่วนประกอบของแอนดรอยด์แอปพลิเคชันมีอยู่ 4 ประเภทดังนี้

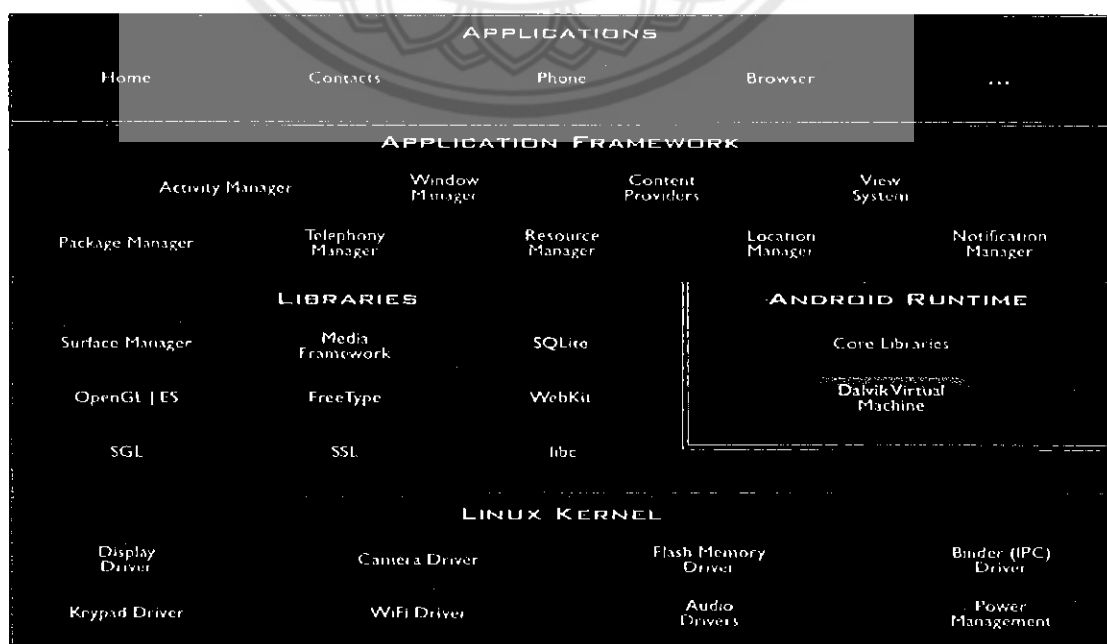
1. Activity (User Interface) คือ สิ่งที่ใช้ในการแสดงผล ออกมาเพื่อให้ผู้ใช้งานได้เห็น และ ได้ใช้งาน โดยในแต่ละ แอปพลิเคชัน นั้นอาจจะมี activity เดียวหรือหลายๆactivity ก็ได้หรืออาจจะ ไม่มีactivity ก็ได้ และสิ่งที่อยู่ในactivity นั้นจะเรียกว่า view ซึ่งก็มีอยู่ในหลายรูปแบบ เช่น buttons , text fields , scroll bars , menu items ,check boxes และอื่นๆ

2. Service (Service Provider) คือ สิ่งที่ไม่มีส่วนของการแสดงผล แต่ถูกเรียกว่ารันอยู่ใน ลักษณะของ background process โดย service นั้นอาจจะมีกรกระทำบางอย่าง เช่น คิดต่อรับส่ง ข้อมูลผ่านเครือข่าย หรือ คำนวณค่าต่างๆแล้วก็ทำการส่งผลลัพธ์นั้นไปแสดงยัง activity ได้

3. Broadcast receiver (Data Provider) คือ ตัวที่ใช้สำหรับคอยรับ และตอบสนองต่อ เหตุการณ์ต่างๆที่เกิดขึ้น เช่น เมื่อแบตเตอรี่ต่ำ, ผู้ใช้ทำการเปลี่ยนภาษา, มีการโทรออก, มีข้อความ เข้า และอื่นๆถึงแม้ broadcast receiver จะ ไม่มีส่วนของการแสดงผล แต่มันก็สามารถเรียกที่จะเรียก activity ขึ้นมาแสดงผลให้ผู้ใช้งานได้ หรืออาจจะเรียกว่า Notification Manager ซึ่งจะเป็นตัวที่แจ้งเตือน ในรูปแบบของ การสั่น, การแสดงไฟกระพริบที่หน้าจอ หรือการส่งเสียงออกมา โดยจะมี icon แสดงอยู่บน status bar เพื่อให้ผู้ใช้กดเข้าไปเปิดดูเหตุการณ์ที่เกิดขึ้น

4. Content provider (System Event Listener) คือกลุ่มของข้อมูลที่สร้างขึ้นจาก แอปพลิเคชันอื่นๆ ได้นำไปใช้โดยการจับเก็บ ข้อมูลของ content provider นั้นจะอยู่ในลักษณะของ ไฟล์, ฐานข้อมูล SQLite และอื่นๆ ตัวอย่างแอปพลิเคชันที่ใช้งาน content provider ที่เห็นชัดเจน ที่สุดคือ โปรแกรม Contacts ที่แสดงรายชื่อใน Contacts ของเรานั้นเอง

2.1.3 โครงสร้างทางซอฟต์แวร์แอนดรอยด์ [4]



รูปที่ 2.2 โครงสร้างทางซอฟต์แวร์แอนดรอยด์ [4]

จากรูปที่ 2.2 โครงสร้างทางซอฟต์แวร์แอนดรอยด์มีดังนี้

1. Java applications ที่รันบน Java object oriented application framework ที่ประกอบไปด้วย Java core libraries ที่รันบน Dalvik virtual machine กับ JIT compilation (โปรแกรมที่เปลี่ยนกลับ bytecode ของ Java ให้เป็นคำสั่งที่สามารถส่งตรงไปที่โพรเซสเซอร์)

2. ไลเบรารีต่างๆที่ถูกเขียนขึ้นด้วยภาษา C เป็นระบบจัดการขั้นพื้นฐาน

3. ใช้ OpenCore เป็น media framework (ตัวจัดการด้านมัลติมีเดีย)

4. ใช้ SQLite เป็นระบบจัดการฐานข้อมูล

5. ใช้ OpenGL ES 2.0 API เป็นตัวจัดการกราฟิกแบบ 3 มิติ 3D

6. WebKit เลย์เอาต์เอนจิน

7. SGL กราฟิกเอนจิน Skia Graphics Engine

8. SSL Transport Layer Security (TLS) หรือชื่อเดิม Secure Sockets Layer (SSL) เป็นโปรโตคอลที่ใช้เข้ารหัสข้อมูลที่ส่งในอินเทอร์เน็ต เช่น เว็บเพจ จดหมายอิเล็กทรอนิกส์ โปรแกรมสนทนา และอื่นๆ เพื่อความปลอดภัยในการส่งข้อมูล มีข้อแตกต่างในรายละเอียดทางเทคนิคระหว่าง SSL 3.0 และ TLS 1.0 เพียงเล็กน้อย ดังนั้นตัวย่อ SSL จะหมายถึงโปรโตคอลทั้งคู่ในกรณีที่ไม่ระบุว่าตัวใดตัวหนึ่งเป็นพิเศษ

9. Bionic C Library ไลเบรารีภาษา C สำหรับอุปกรณ์เคลื่อนที่

โดยรวมแล้ว Source Code สำหรับระบบปฏิบัติการแอนดรอยด์ ที่มีมากถึง 12 ล้านบรรทัด นั้นประกอบไปด้วย

ซอร์สโค้ดที่เป็น XML (Extensible Markup Language) Layout ต่างๆ ประมาณ 3 ล้านบรรทัด

ซอร์สโค้ดที่เขียนจากภาษา C 2.8 ล้านบรรทัด

ซอร์สโค้ดที่เขียนจากภาษา JAVA 2.1 ล้านบรรทัด

ซอร์สโค้ดที่เขียนจากภาษา C++ 1.75 ล้านบรรทัด

2.2 ระบบฐานข้อมูล

ในการจัดการกับฐานข้อมูลเราเลือกใช้ SQL เป็นตัวดำเนินการในการจัดการกับฐานข้อมูล ซึ่ง SQL เป็นผลิตภัณฑ์ทางด้านฐานข้อมูลที่มีโครงสร้างข้อมูลแบบ Relation จำเป็นอย่างยิ่งที่จะต้องมีความรู้ทางด้านฐานข้อมูล เช่น ภาษา SQL (Structure Query Language) ภาษา QBE (Query by Example) และภาษา Quel ฯลฯ เป็นต้น ภาษาเหล่านี้ได้ถูกพัฒนาขึ้น จากแนวคิดที่ต่างกัน เช่น ภาษา QBE ซึ่งถูกพัฒนาขึ้นจากแนวคิด ของ Relational Calculus ส่วนภาษา Quel ถูกพัฒนาขึ้นจากแนวคิด ของ Tuple Relational Calculus และ Relational Algebra เป็นหลัก แต่อย่างไรก็ตาม ภาษาที่ได้รับความนิยมมากที่สุดคือภาษา "SQL" หลักการของภาษา SQL ที่ใช้พื้นฐานจากระบบจัดการฐานข้อมูลแบบเชิงสัมพันธ์ โดยที่ใช้รูปแบบของตาราง (Table) ที่ใช้แทนความสัมพันธ์ระหว่างข้อมูล และในแต่ละตารางจะมีเขตข้อมูล (Field) ต่าง ๆ ที่มีความสัมพันธ์ซึ่งกันและกัน ประกอบตัวเป็นตารางต่าง ๆ ทุกตารางของ SQL จะเป็นตารางแบบสองมิติ คือ แถว และสดมภ์ (Rows and Columns) โดยที่มีคำว่าแอททริบิวต์ (Attributes) หมายถึงข้อมูลตามแนว Columns และ ทัพเพิล (Tuple) หมายถึงข้อมูลตาม Row และสามารถเรียกได้อีกอย่างหนึ่งว่า เรคอร์ด (Record)

2.2.1 ชนิดของข้อมูลที่ใช้ในฐานข้อมูล [5]

1. ตัวหนังสือแบบความยาวคงที่ (fixed-length character) จะใช้ char(n) หรือ character(n) แทนประเภทของข้อมูลที่เป็นตัวหนังสือใดๆที่มีความยาวของข้อมูลคงที่โดยมีความยาว n ตัวหนังสือประเภทนี้จะมีการจองเนื้อที่ตามความยาวที่คงที่ตามที่กำหนดไว้ ชนิดของข้อมูลประเภทนี้จะเก็บความยาวของข้อมูลได้มากที่สุดได้ 255 ตัวอักษร

ตัวหนังสือแบบความยาวไม่คงที่ (variable-length character) จะใช้ varchar(n) แทนประเภทของข้อมูลที่เป็นตัวหนังสือใดๆที่มีความยาวของข้อมูลไม่คงที่ โดยมีความยาว n ตัวหนังสือประเภทนี้จะมีการจองเนื้อที่ตามความยาวของข้อมูล ชนิดของข้อมูลประเภทนี้จะเก็บความยาวของข้อมูลได้มากที่สุดได้ 4000 ตัวอักษร

2. จำนวนเลขที่มีจุดทศนิยม (decimal) ในภาษา SQL จะใช้ dec(m,n) หรือ decimal(m,n) เป็นประเภทข้อมูลที่เป็นจำนวนเลขที่มีจุดทศนิยมโดย m คือจำนวนตัวเลขทั้งหมด (รวมจุดทศนิยม) และ n คือจำนวนตัวเลขหลังจุดทศนิยม

จำนวนเลขที่ไม่มีจุดทศนิยมในภาษา SQL จะใช้ int หรือ integer เป็นเลขจำนวนเต็มบวกหรือลบขนาดใหญ่ เป็นตัวเลข 10 หลัก ที่มีค่าตั้งแต่ -2,147,483,648 ถึง +2,147,483,647 และในภาษา SQL จะใช้ smallint เป็นประเภทข้อมูลที่เป็นเลขจำนวนเต็มบวกหรือลบขนาดเล็ก เป็นตัวเลข 5 หลัก ที่มีค่าตั้งแต่ - 32,768 ถึง + 32,767 ตัวเลขจำนวนเต็มประเภทนี้จะมีการจองเนื้อที่น้อยกว่าแบบ integer เลขจำนวนจริง ในภาษา SQL อาจใช้ number(n) แทนจำนวนเลขที่ไม่มีจุดทศนิยมและจำนวนเลขที่มีจุดทศนิยม

2.2.2 การตั้งชื่อตาราง

การตั้งชื่อตาราง American National Standard Institute (ANSI) ได้อนุมัติให้ภาษา SQL เป็นภาษาที่ได้มาตรฐาน โดยมีชื่อ SQL-92 แต่ภาษามาตรฐานก็มีข้อจำกัด จึงได้มีการขยายขยายในภาษาที่ใช้จริงในระบบจัดการฐานข้อมูลทั่วไป ให้มีการเพิ่มสัญลักษณ์และกฎให้มากขึ้น ซึ่งทำให้ระบบจัดการฐานข้อมูลต่าง ๆ มีภาษา SQL ที่ต่างกันไป ขึ้นอยู่กับการพัฒนาของผู้ผลิตแต่ละราย การตั้งชื่อของตารางในภาษา SQL นั้น มีหลักการง่าย ๆ กล่าวคือ จะต้องนำหน้าชื่อตารางด้วยพยัญชนะใด ๆ หรือ \$, # หรือ @ ก็ได้ และไม่ควรถังชื่อยาวกว่า 18 อักขร และภายในชื่อสามารถผสมคำจากอักขรใด ๆ และอักขรโรมันตัวเล็ก เลข 0..9, \$, #, @ และ _ (Underscores) อย่างไรก็ตามชื่อไม่ควรสั้นเกินไป เช่น m หรือ nn หรือไม่ควรยาวเกินไป เช่น my_little_brother_money ข้อควรระวังอีกอย่างหนึ่งคือ SQL จะมี คำสงวน (Reserved Words) หรือ Keyword ที่ต้องระวังไม่ให้ใช้ในการประกาศเป็นตัวแปร หรือ ชื่อของตารางใด ๆ ซึ่งคำสงวนจะมีคำดังนี้

ตารางที่ 2.2 ตารางชื่อเฉพาะของภาษา SQL ที่ไม่สามารถนำไปสร้างเป็นชื่อตัวแปรได้

ADD	EXCEPT	PERCENT
ALL	EXEC	PLAN
ALTER	EXECUTE	PRECISION
AND	EXISTS	PRIMARY
ANY	EXIT	PRINT
AS	FETCH	PROC
ASC	FILE	PROCEDURE
AUTHORIZATION	FILLFACTOR	PUBLIC
BACKUP	FOR	RAISERROR
BEGIN	FOREIGN	READ
BETWEEN	FREETEXT	READTEXT
BREAK	FREETEXTTABLE	RECONFIGURE
BROWSE	FROM	REFERENCES

ตารางที่ 2.2 (ต่อ) ตารางชื่อเฉพาะของภาษา SQL ที่ไม่สามารถนำไปสร้างเป็นชื่อตัวแปรได้

BULK	FULL	REPLICATION
BY	FUNCTION	RESTORE
CASCADE	GOTO	RESTRICT
CASE	GRANT	RETURN
CHECK	GROUP	REVOKE
CHECKPOINT	HAVING	RIGHT
CLOSE	HOLDLOCK	ROLLBACK
CLUSTERED	IDENTITY	ROWCOUNT
COALESCE	IDENTITY_INSERT	ROWGUIDCOL
COLLATE	IDENTITYCOL	RULE
COLUMN	IF	SAVE
COMMIT	IN	SCHEMA
COMPUTE	INDEX	SELECT
CONSTRAINT	INNER	SESSION_USER
CONTAINS	INSERT	SET
CONTAINSTABLE	INTERSECT	SETUSER
CONTINUE	INTO	SHUTDOWN
CONVERT	IS	SOME
CREATE	JOIN	STATISTICS
CROSS	KEY	SYSTEM_USER
CURRENT	KILL	TABLE
CURRENT_DATE	LEFT	TEXTSIZE
CURRENT_TIME	LIKE	THEN
CURRENT_TIMESTAMP	LINENO	TO
CURRENT_USER	LOAD	TOP
CURSOR	NATIONAL	TRAN

ตารางที่ 2.2 (ต่อ) ตารางชื่อเฉพาะของภาษา SQL ที่ไม่สามารถนำไปสร้างเป็นชื่อตัวแปรได้

DATABASE	NOCHECK	TRANSACTION
DBCC	NONCLUSTERED	TRIGGER
DEALLOCATE	NOT	TRUNCATE
DECLARE	NULL	TSEQUAL
DEFAULT	NULLIF	UNION
DELETE	OF	UNIQUE
DENY	OFF	UPDATE
DESC	OFFSETS	UPDATETEXT
DISK	ON	USE
DISTINCT	OPEN	USER
DISTRIBUTED	OPENDATASOURCE	VALUES
DOUBLE	OPENQUERY	VARYING
DROP	OPENROWSET	VIEW
DUMMY	OPENXML	WAITFOR
DUMP	OPTION	WHEN
ELSE	OR	WHERE
END	ORDER	WHILE
ERRLVL	OUTER	WITH
ESCAPE	OVER	WRITETEXT

จากตารางที่ 2.2 คือชื่อเฉพาะของภาษา SQL ที่ไม่สามารถนำไปใช้ตั้งเป็นชื่อตัวแปรได้

2.2.3 ฟังก์ชัน [6]

ฟังก์ชันของ SQL จะมีหลายฟังก์ชันซึ่งเราสามารถเรียกใช้ฟังก์ชันต่างๆ นี้เขียนโปรแกรมเพื่อลดระยะเวลาในการเขียนโปรแกรมได้ซึ่งฟังก์ชันต่างๆ มีดังต่อไปนี้

1. AGV เป็นคำสั่งที่ใช้ในการหาค่าเฉลี่ยของตัวเลขหนึ่งชุด โดยตำแหน่งที่ไม่มีค่าใดบรรจุอยู่ (Null value) จะไม่รวมอยู่ในชุดการคำนวณ หากทุกค่าในชุดตัวเลขนั้นเป็น Null value ค่าที่ได้ก็จะเป็น Null เช่นกัน หากใช้ AGV (Column-name) จะคำนวณค่าเฉลี่ยของค่าทุกตัวใน Column นั้น รวมทั้งตัวที่มีค่าซ้ำกันด้วย เราอาจใช้คำสั่ง AGV (DISTINCT column-name) เพื่อหาค่าเฉลี่ยโดยไม่ต้องนำค่าที่ซ้ำกันมาคำนวณ

ตัวอย่างการหาค่าเฉลี่ยของเงินเดือนจากตาราง

```
EMPLOYEE
SELECT AVG(SALARY)
FROM EMPLOYEE;
```

2. SUM เป็นคำสั่งที่ใช้ในการรวมหรือบวกตัวเลขหนึ่งชุด โดย Null value จะไม่ถูกนำมาคำนวณ และหากค่าทั้งหมดเป็น Null ผลลัพธ์ที่ได้จะเป็น Null เช่นกัน ตัวอย่างการใช้ SUM ในการรวมเงินเดือนทั้งหมดในตาราง

```
EMPLOYEE
SELECT SUM(SALARY)
FROM EMPLOYEE;
```

ตัวอย่าง การรวมเงินเดือนผู้ที่ทำงานเฉพาะแผนก MIS จากตาราง

```
EMPLOYEE
SELECT SUM(SALARY)
FROM EMPLOYEE
WHERE DEPT = 'MIS';
```

3. Min เป็นคำสั่งที่ใช้ในการรวมหรือบวกตัวเลขหนึ่งชุด โดย MIN จะทำการประเมินหาค่าที่น้อยที่สุดจากชุดตัวเลขนั้น

ตัวอย่างการหาค่าเงินเดือนที่น้อยที่สุดในตาราง

```
EMPLOYEE
SELECT MIN(SALARY)
FROM EMPLOYEE;
```

4. Max เป็นคำสั่งที่สามารถใช้กับชุดตัวเลขชนิดใดก็ได้ โดยที่ MAX จะทำการประเมินหาค่าที่มากที่สุดจากชุดตัวเลขนั้น

ตัวอย่างการหาค่าเงินเดือนที่มากที่สุดในตาราง

```
EMPLOYEE
SELECT MAX(SALARY)
FROM EMPLOYEE;
```

5. Count เป็นคำสั่งที่สามารถใช้กับตารางหรือ Column ใด ๆ เพื่อบันทึกจำนวนของ Table หรือ Field ใน Column ซึ่งจะมีการใช้อยู่ 2 แบบดังนี้

COUNT (*) เป็นคำสั่งที่ใช้บันทึกจำนวนแถว (Rows) ใน Table โดยรวมถึงค่าที่ซ้ำ และตำแหน่งที่ไม่มี (Null)

ตัวอย่างการนับจำนวนว่ามีข้อมูลจำนวนรายการในตาราง

```
EMPLOYEE
SELECT COUNT(*)
FROM EMPLOYEE;
```

ตัวอย่างการนับจำนวนว่ามีข้อมูลจำนวนรายการในตาราง EMPLOYEE ที่มีเงินเดือนมากกว่าหรือเท่ากับ 20000

```
SELECT COUNT(*)
FROM EMPLOYEE
WHERE SALARY >=20000;
```

ตัวอย่างการนับจำนวนพนักงานตามแผนกที่ตั้งกักอยู่ และให้นับเฉพาะแผนกที่มีพนักงานตั้งแต่ 2 คนขึ้นไปเท่านั้น จากตาราง EMPLOYEE

```
SELECT DEPT, COUNT(*)
FROM EMPLOYEE
GROUP BY DEPT
HAVING COUNT(*) >=2;
```

COUNT (DISTINCT Column-name) เป็นคำสั่งที่ใช้บันทึกจำนวนแถว (Rows) ใน Table โดยไม่รวมถึงค่าที่ซ้ำ และตำแหน่งที่ไม่มี (Null)

ตัวอย่างการนับว่ามีกี่แผนกจากข้อมูลในตาราง ซึ่งเป็นการใช้ Function ผสมผสาน ซึ่งคำตอบที่ได้จะเป็นคำตอบที่นับจำนวน DEPT ที่ไม่ซ้ำกัน

```
SELECT COUNT (DISTINCT DEPT)
FROM EMPLOYEE;
```

2.2.4 การใช้งานกลุ่มคำสั่ง SQL

การใช้งานกลุ่มคำสั่ง SQL ในระบบฐานข้อมูล แบ่งออกเป็น 3 ประเภท คือ การเรียกดูข้อมูล การเพิ่มข้อมูลใหม่เข้าไป และการลบข้อมูล ซึ่งในระดับของผู้ใช้งาน จะไม่ทราบและไม่ต้องสนใจว่าวิธีการจัดเก็บข้อมูลจริง ๆ นั้นว่าเป็นอย่างไร ดังนั้น การใช้งานข้อมูลในระดับของผู้ใช้นั้นก็สามารถกระทำได้โดยผ่านทาง DBMS โดยการใช้อำนาจสำหรับการจัดดำเนินการข้อมูล (Data Manipulation Language) หรือที่เรียกว่าย่อ ๆ ว่า DML นั่นเอง คำสั่งต่าง ๆ ในภาษาประเภท DML มีดังนี้

1. คำสั่ง SELECT

```
SELECT [predicate] { * I table.* I[ table.]field1 [AS alias1] [, [table.]field2[ AS alias2][,...]]
FROM tableexpression [,...] [IN externaldatabase]
```

WHERE คือ เงื่อนไขในการค้นหา

GROUP BY คือ เงื่อนไขในการรวมกลุ่ม

HAVING คือ เงื่อนไขของกลุ่ม

ORDER BY คือ การเรียงลำดับ

ข้อความในเครื่องหมายวงเล็บใหญ่ [...] หมายความว่าไม่จำเป็นที่จะใช้ก็ได้ คำสั่งในการค้นคืน (Retrieval) ข้อมูลจากตารางต่าง ๆ จะใช้คำสั่ง SELECT ในการค้นข้อมูลจากตาราง SELECT เป็นคำสั่งที่ใช้ในการเลือกฟิลด์ข้อมูลที่ต้องการจากตารางที่กำหนดให้ หรือถ้าต้องการดูทั้งหมด จะใส่สัญลักษณ์ * (SELECT*) โดยต้องใส่ชื่อของตารางลงในช่อง FROM ซึ่งจะเป็นชื่อตาราง หรือ View ก็ได้ นอกจากนั้น หลังจากคำสั่ง WHERE สามารถตามด้วยเงื่อนไขที่ต้องการเพื่อเลือกข้อมูลที่เหมาะสม ได้อีกด้วย

1.1 การเรียกดูข้อมูลทั้งหมดหรือบางส่วนจากตารางการกำหนดค่า Predicate เป็น All หมายถึงการเลือกทุกเรคอร์ดที่อยู่ในเงื่อนไข โดยต้องใช้เครื่องหมาย * หรือชื่อฟิลด์ ร่วมด้วย ซึ่งถ้าใช้เครื่องหมาย * จะหมายถึงเลือกทุกฟิลด์ ในกรณีที่ไม่มี Predicate เป็น All ก็จะหมายถึงเลือกทุกเรคอร์ด

คำสั่ง SELECT ในการเรียกดูข้อมูลทั้งหมดจากตาราง EMPLOYEE

```
SELECT ALL*
```

```
FROM EMPLOYEE; หรือ
```

```
SELECT *
```

```
FROM EMPLOYEE;
```

จะเป็นการเลือกข้อมูลทั้งหมดจากตาราง EMPLOYEE

1.2 การเรียกดูข้อมูลโดยใช้คำสั่ง Distinct คำสั่ง DISTINCT เป็นคำสั่งที่ใช้ในการตัดฟิลด์ที่มีข้อมูลซ้ำกันออกและจัดเรียงข้อมูลใหม่จากน้อยไปหามาก

```
SELECT DISTINCT DEPT
FROM EMPLOYEE
```

1.3 การเปรียบเทียบตามเงื่อนไข WHERE การเปรียบเทียบตามเงื่อนไขของภาษา SQL จะต้องอยู่หลังคำสั่ง WHERE ซึ่งสามารถเปรียบเทียบตามสัญลักษณ์ LOGICAL BOOLEAN ดังนี้

= เท่ากับ
< > ไม่เท่ากับ
> มากกว่า
>= มากกว่าหรือเท่ากับ
< น้อยกว่า
<= น้อยกว่าหรือเท่ากับ

นอกจากนั้นยังสามารถใช้เงื่อนไขพิเศษเช่น NOT, OR และ AND ได้อีกด้วย

2. คำสั่ง insert คำสั่ง SQL ที่ใช้ในการเพิ่มแถวข้อมูลลงในตาราง จะใช้คำสั่ง INSERT ซึ่งแบ่งออกเป็น 2 รูปแบบคือแบบที่ใช้เพิ่มข้อมูลที่ละแถว โดยทำการระบุค่าข้อมูลของแต่ละฟิลด์ลงไปในการคำสั่ง INSERT โดยตรง และแบบที่ใช้คำสั่งค้นหาข้อมูล (SELECT) เพื่อดึงกลุ่มข้อมูลส่งให้กับคำสั่ง INSERT ทำการเพิ่มข้อมูลลงในตาราง รูปแบบทั้งสองแบบมีดังนี้คือ คำสั่งการเพิ่มข้อมูลที่ละแถว

```
INSERT INTO TABLE [(FIELD 1 [,FIELD2[,...]]) [ ชื่อตารางที่จะเพิ่มข้อมูล ]
VALUES (VALUE1[VALUE2[,...]]) [ ค่าข้อมูลของแต่ละคอลัมน์ ]
```

คำสั่งการเพิ่มข้อมูลมากกว่าหนึ่งแถว

```
INSERT INTO TABLE[IN EXTERNALDATABASE] [(FIELD 1 [,FIELD2[,...]])
SELECT [SOURCE.] FIELD 1 [,FIELD2[,...]]
VALUES (VALUE1[VALUE2[,...]]) [ ค่าข้อมูลของแต่ละคอลัมน์ ]
FROM TABLEEXPRESSION
```

โดยที่

TABLE คือ ตารางที่ต้องการเพิ่มข้อมูล

EXTERNALDATABASE คือ ที่อยู่ของตารางหรือคิวรีของฐานข้อมูลอื่น

SOURCE คือ ชื่อของตารางหรือคิวรีที่ต้องการคัดลอกข้อมูล

TABLEEXPRESSION คือ ชื่อของตารางข้อมูลที่ต้องการนำไปเพิ่ม

3. คำสั่ง UPDATE การแก้ไขข้อมูล หรือการ UPDATE จะใช้คำสั่ง UPDATE ซึ่งมีรูปแบบดังนี้

```
UPDATE <table name>
SET <attribute name> = '<name value>'
<WHERE <condition>>;
```

4. คำสั่ง delete การลบข้อมูลในภาษา SQL จะใช้คำสั่ง DELETE ซึ่งคำสั่งนี้จะลบข้อมูลทั้งหมด หรือทั้งหมด ตามเงื่อนไขประกอบ ซึ่งมีรูปแบบดังนี้

```
DELETE FROM <table name>
<WHERE <condition>
```

2.3 การพยากรณ์รายจ่าย [7]

การพยากรณ์รายจ่ายจะมีด้วยกันหลายวิธีซึ่งในที่นี้เราจะใช้วิธีการพยากรณ์แบบหาค่าเฉลี่ยเคลื่อนที่ (Moving averages) และ การปรับให้เรียบแบบเอกซ์โปเนนเชียลครั้งเดียว (Single Exponential Smoothing Method: SES) โดยจะใช้ข้อมูลก่อนเดือนที่จะพยากรณ์เป็นตัวชี้การพยากรณ์ ซึ่งทั้งสองวิธีนี้จะเหมาะกับข้อมูลที่ไม่มีแนวโน้มและเหมาะกับการพยากรณ์ในระยะสั้น

2.3.1 ค่าเฉลี่ยเคลื่อนที่ (Moving averages)

การค่าเฉลี่ยเคลื่อนที่จะเป็นการประยุกต์ใช้เทคนิคการปรับให้เรียบแบบค่าเฉลี่ยเคลื่อนที่ ซึ่งถ้าข้อมูลที่ใช้ในการคำนวณค่าเฉลี่ยเคลื่อนที่มากเท่าใดอิทธิพลของความเรียบของค่าประมาณก็จะมีค่ามากยิ่งขึ้น ดังนั้นการพยากรณ์จึงต้องเลือกจำนวนหน่วยเวลาที่ใช้คำนวณ โดยที่ค่าดังกล่าวจะมีค่าอยู่ระหว่าง 1 ถึง n การหาค่าเฉลี่ยเคลื่อนที่หาได้จากสมการที่ 1

$$Y_{(t)} = \frac{\sum_{i=1}^N y_{(t-i)}}{N} = \frac{y_{(t-1)} + y_{(t-2)} + y_{(t-3)} + \dots + y_{(t-N)}}{N} \quad \leftarrow \text{สมการที่ 1}$$

Y = ค่าเฉลี่ยที่เป็นค่าพยากรณ์

y = จำนวนรายจ่าย

$t-i$ = ลำดับคาบเวลาที่ i ใดๆ

\sum = ผลรวม

N = จำนวนคาบเวลาที่ต้องการเคลื่อนที่

จากสูตรจำเป็นต้องใช้ข้อมูลในอดีตรวมทั้งข้อมูลในปัจจุบันมาพยากรณ์ในหน่วยเวลาล่วงหน้าหนึ่งหน่วย

ตารางที่ 2.3 ตารางตัวอย่างการหาค่าเฉลี่ยด้วยวิธีหาค่าเฉลี่ยเคลื่อนที่ (Moving averages)

เดือนที่	เดือน	ค่าใช้จ่าย	N=3	N=5
1	มกราคม	200.0	-	-
2	กุมภาพันธ์	135.0	-	-
3	มีนาคม	195.0	-	-
4	เมษายน	197.5	176.7	-
5	พฤษภาคม	310.0	175.8	-
6	มิถุนายน	175.0	234.2	207.5
7	กรกฎาคม	155.0	227.5	202.5
8	สิงหาคม	130.0	213.3	206.5
9	กันยายน	220.0	153.3	193.5
10	ตุลาคม	277.0	168.3	198.0
11	พฤศจิกายน	235.0	209.2	191.4
12	ธันวาคม	-	244.2	203.5

จากตารางที่ 2.3 เป็นตัวอย่างการหาค่าเฉลี่ยด้วยวิธีหาค่าเฉลี่ยเคลื่อนที่ (Moving averages) จะเห็นได้ว่าการพยากรณ์ที่ $Y(t)$ เป็นการปรับปรุงจากค่าก่อนหน้าที่ $N=3$ จะนำข้อมูลย้อนหลัง 3 เดือนมาพยากรณ์ และ $N=5$ จะนำข้อมูลย้อนหลัง 5 เดือนมาพยากรณ์ แต่ในที่นี้เราได้เลือกหาค่าเฉลี่ยเคลื่อนที่อันดับที่ 5 ของอนุกรมเวลา

ในกรณีที่รูปแบบพยากรณ์ที่ใช้ไม่ประสบผลสำเร็จอาจจะเป็นเพราะข้อมูลที่ใช้ไม่สอดคล้องหรือไม่เป็นไปตามรูปแบบของการพยากรณ์เบื้องต้น ซึ่งการพยากรณ์ด้วยค่าเฉลี่ยเคลื่อนที่ (Moving averages) จะมีความเหมาะสมกับข้อมูลอนุกรมเวลาที่มีค่ากระจายรอบค่าเฉลี่ยโดยไม่มีแนวโน้มหรือฤดูกาลเข้ามาเกี่ยวข้อง

2.3.2 การปรับให้เรียบแบบเอกซ์โปเนนเชียลครั้งเดียว (Single Exponential Smoothing

Method: SES)

เป็นวิธีที่ใช้หลักการหาค่าเฉลี่ยแบบถ่วงน้ำหนัก ด้วย α หรือเรียกว่า Exponential Smoothing Constant กล่าวคือ ข้อมูลแต่ละช่วงเวลามีความสำคัญแตกต่างกัน ดังนั้นน้ำหนักถ่วงจึงไม่ควรเท่ากัน โดยให้ความสำคัญกับข้อมูลเวลาล่าสุดมากที่สุด และเวลาห่างออกไปจะลดหลั่นในลักษณะเอกซ์โปเนนเชียล โดยที่ $0 \leq \alpha \leq 1$

เช่น ข้อมูลที่เป็นปัจจุบัน ให้น้ำหนักถ่วงเป็น α

ส่วนน้ำหนักถ่วงก่อนหน้านี้นี้ 1 ช่วงเวลาเป็น $\alpha(1-\alpha)$

และน้ำหนักถ่วงก่อนหน้านี้นี้ 2 ช่วงเวลาเป็น $\alpha(1-\alpha)^2$ เช่นนี้ไปเรื่อยๆ

วิธีนี้เหมาะสำหรับข้อมูลที่มีลักษณะการเคลื่อนไหวอยู่ในระดับคงที่ (Horizontal Data) หรือ ข้อมูลที่ไม่มีแนวโน้มและไม่มีฤดูกาล และ เหมาะสมสำหรับการพยากรณ์ในระยะสั้น รูปแบบของข้อมูลคือ $Y_t = \text{ค่าคงที่} + \text{ความคลาดเคลื่อน}$ ณ เวลา t กำหนดให้ข้อมูล Y_t ; $t = 1, 2, \dots, n$ โดยขั้นตอนการหาค่าพยากรณ์สามารถหาได้ดังนี้

1. การกำหนดค่าถ่วงน้ำหนัก α

วิธีการกำหนดค่า α โดย ค่า α จะอยู่ระหว่าง $0 \leq \alpha \leq 1$ พิจารณาจากค่าสัมประสิทธิ์สหสัมพันธ์ โดยใช้ค่า α จะขึ้นอยู่กับดุลพินิจ ในทางปฏิบัติจะลองเปลี่ยนค่าไปเรื่อยๆ หากค่า α ค่าใดทำให้ข้อมูลพยากรณ์มีค่าใกล้เคียงกับค่าจริงในอดีตมากที่สุดก็ให้เลือกมาใช้ในสมการ โดยใช้ค่า α ที่ต่ำกับอนุกรมเวลาที่มีค่าสัมประสิทธิ์สหสัมพันธ์ต่อเนื่องที่ต่ำ และใช้ค่า α ที่สูงกับอนุกรมเวลาที่มีค่าสัมประสิทธิ์สหสัมพันธ์ต่อเนื่องที่สูง โดยค่าสัมประสิทธิ์สหสัมพันธ์ (r)

$$r = \frac{\sum(Y_t - \bar{Y})(\hat{Y}_t - \bar{\hat{Y}}_t)}{\sqrt{\sum(Y_t - \bar{Y})^2 \sum(\hat{Y}_t - \bar{\hat{Y}}_t)^2}} \quad \leftarrow \text{สมการที่ 2}$$

สุ่มค่า α มา 2-3 ค่า แล้วทำการคำนวณค่า SSE, MSE, RMSE, MAPE ค่าใดค่าหนึ่ง แล้วนำมาเปรียบเทียบ ถ้าค่าใดให้ค่าที่ต่ำกว่าจึงเลือก α นั้น
ถ้าคำนวณจาก Minitab ในที่นี้เราจะใช้ค่า $\alpha = 0.1$

2. คำนวณหาค่าพยากรณ์จากสมการ

สมการที่ใช้คำนวณ มี 2 รูปแบบคือ

รูปแบบที่ 1 แบบปรับให้เรียบ (Smoothing form)

$$\text{สมการพยากรณ์ } \hat{Y}_{t+1} = \alpha Y_t + (1 - \alpha) \hat{Y}_t \longleftarrow \text{สมการที่ 3}$$

รูปแบบที่ 2 แบบปรับด้วยค่าคลาดเคลื่อน (Error correction form)

$$\text{สมการพยากรณ์ } \hat{Y}_{t+1} = \hat{Y}_t + \alpha e_t \longleftarrow \text{สมการที่ 4}$$

$$e_t \text{ คือ ค่าความคลาดเคลื่อนที่เวลา } t \text{ ซึ่ง } e_t = Y_t - \hat{Y}_{t-1} \quad (1)$$

ถ้าพยากรณ์ต่อไปค่าพยากรณ์ที่ได้จะคงที่เพราะเทคนิคการปรับให้เรียบแบบเอกซ์โปเนนเชียลอย่างง่ายจะพยากรณ์ค่าในอนาคตเป็นค่าคงที่ ด้วยเหตุนี้ถ้าอนุกรมเวลามีลักษณะคงที่ไม่มีส่วนประกอบแนวโน้ม หรือส่วนประกอบอนุกรมเวลาอื่นๆ การปรับแบบเอกซ์โปเนนเชียลอย่างง่ายเป็นวิธีที่เหมาะสมวิธีหนึ่งในการพยากรณ์อนุกรมเวลาที่จะเกิดในอนาคต

จากบทนี้ได้ทำการศึกษาส่วน โครงสร้างของแอนดรอยด์ทำให้รู้ว่าจะสร้างไฟล์จาวาไว้ตรงไหนและการเขียนโปรแกรมแอนดรอยด์ต้องทำอะไร ศึกษาในส่วนของการนำเก็บไฟล์ภาพตามระดับความสำคัญ ความละเอียด การดูเวอร์ชันที่ใช้พัฒนาว่าโปรแกรมที่ต้องการพัฒนาเหมาะกับเวอร์ชันใด ระบบการทำงานของแอนดรอยด์เป็นอย่างไรจากนั้นก็ทำการศึกษาภาษา SQL ที่ต้องนำมาใช้ทำฐานข้อมูลว่ามีโครงสร้าง การลบ และคำสั่งการเรียกดูข้อมูลอย่างไร อีกทั้งได้ศึกษาสูตรในการคำนวณที่จำเป็นต้องใช้ในแอปพลิเคชัน รวมไปถึงสูตรและวิธีที่ใช้ในการพยากรณ์รายจ่ายล่วงหน้า นั่นคือการหาค่าเฉลี่ยเคลื่อนที่ (Moving averages) และการปรับให้เรียบแบบเอกซ์โปเนนเชียลครั้งเดียว (Single Exponential Smoothing Method: SES) เพื่อใช้ในการออกแบบและสร้างแอปพลิเคชันต่อไป

บทที่ 3

วิเคราะห์ ออกแบบ และพัฒนาระบบ

3.1 การวางแผนการทำงาน

1. ศึกษาหาข้อมูลเกี่ยวกับการทำบัญชีระบบรายรับ-รายจ่าย ทัวไป
2. ศึกษาหาข้อมูลเครื่องมือที่ใช้พัฒนา ในที่นี้ใช้ Eclipse IDE 3.7 (64bit), Android SDK, Sqlite พัฒนาที่เวอร์ชัน 2.2
3. ศึกษาภาษา Java และ Sqlite เพื่อการพัฒนา โปรแกรมระบบรายรับ รายจ่าย
4. ศึกษาโปรแกรมรายรับ รายจ่ายที่มีมาก่อน
5. ออกแบบฐานข้อมูล ว่ามีรายละเอียดข้อมูลอย่างไร ในการพัฒนา
6. ทำการสร้าง โปรแกรมตามวัตถุประสงค์ที่วางไว้
7. ทำการแก้ไขโปรแกรม ปรับปรุง เพิ่มเติม ให้มีความเสถียรมากขึ้น

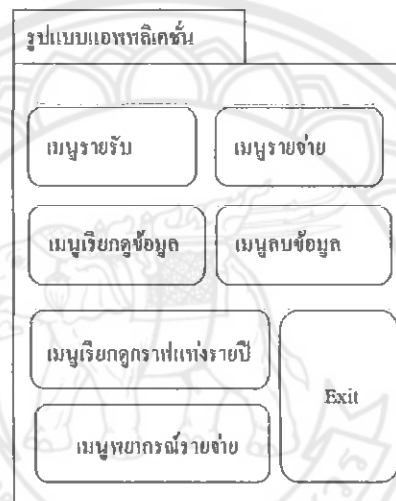


รูปที่ 3.1 แสดง Flow chart ขั้นตอนการทำงาน

3.2 การออกแบบโครงสร้างแอปพลิเคชัน

การออกแบบแอปพลิเคชัน ภายในแอปพลิเคชันจะมีหลายเมนูประกอบด้วย

- เมนูรายรับ
- เมนูรายจ่าย
- เมนูเรียกดูข้อมูล
- เมื่อกลับข้อมูล
- เมนูเรียกดูกราฟแท่งรายปี
- เมนูพยากรณ์รายจ่าย



รูปที่ 3.2 ออกแบบเมนูต่างๆในแอปพลิเคชัน

3.2.1 การออกแบบเมนูรายรับ

รูปที่ 3.3 ออกแบบเมนูรายรับ

จากรูปที่ 3.3 การออกแบบเมนูรายรับ เมนูรายรับจะประกอบด้วย การรับข้อมูล วัน/เดือน/ปี และจำนวนเงินของรายรับเข้ามา แล้วนำไปเก็บไว้ในฐานข้อมูล

3.2.2 การออกแบบเมนูรายจ่าย

รูปที่ 3.4 การออกแบบเมนูรายจ่าย

จากรูปที่ 3.4 เป็นการออกแบบเมนูรายจ่ายซึ่งประกอบด้วย วัน/เดือน/ปี ,จำนวนเงินของรายจ่าย,รายละเอียดของรายจ่าย และประเภทของรายจ่าย เข้ามาแล้วนำไปเก็บไว้ในฐานข้อมูล

3.2.3 การออกแบบเมนูเรียกดูข้อมูล

รูปที่ 3.5 การออกแบบเมนูเรียกดูข้อมูล

จากรูปที่ 3.5 เป็นการออกแบบเมนูเรียกดูข้อมูลซึ่งประกอบด้วย วัน/เดือน/ปี เริ่มและสิ้นสุด ที่ต้องการดู เมื่อได้ช่วงข้อมูลที่ต้องการดูออกมาแล้วจะสามารถดูข้อมูลนั้นในรูปแบบกราฟวงกลมได้และสามารถบันทึกข้อมูลนั้นๆออกมาเป็นไฟล์ .txt ได้ ซึ่งข้อมูลที่ได้นั้นมาจากฐานข้อมูล

3.2.4 การออกแบบเมนูลบข้อมูล

รูปที่ 3.6 การออกแบบเมนูลบข้อมูล

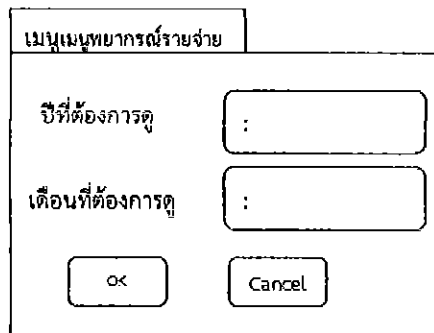
จากรูปที่ 3.6 เป็นการออกแบบเมนูลบข้อมูล ซึ่งประกอบด้วยการเรียกดูข้อมูลเมื่อเรียกดูข้อมูลขึ้นมาจะทำการเลือกข้อมูลที่ต้องการจะลบ

3.2.5 การออกแบบเมนูเรียกดูกราฟแท่งรายปี

รูปที่ 3.7 การออกแบบเมนูเรียกดูกราฟแท่งรายปี

จากรูปที่ 3.7 เป็นการออกแบบเมนูเรียกดูกราฟแท่งรายปีซึ่งประกอบด้วย ปีที่ต้องการดูกราฟและประเภทของรายจ่ายที่ต้องการดูกราฟ ผู้ใช้จะต้องใส่ปี(ค.ศ.)ลงในปีที่ต้องการดูแล้วเลือกประเภทที่ต้องการจะดู

3.2.6 การออกแบบเมนูพยากรณ์รายจ่าย



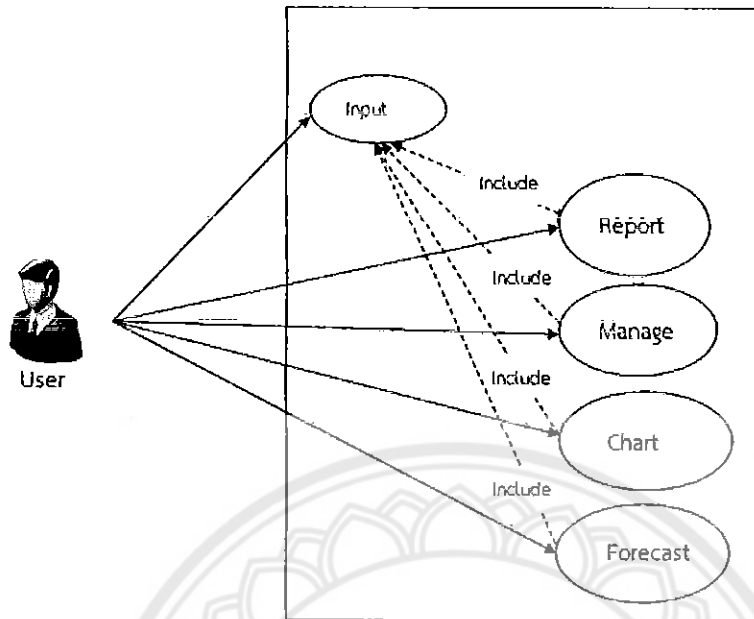
The image shows a dialog box titled "เมนูเมนูพยากรณ์รายจ่าย" (Payment Prediction Menu). It contains two input fields for "ปีที่ต้องการดู" (Year to view) and "เดือนที่ต้องการดู" (Month to view), each followed by a colon and a text box. At the bottom, there are two buttons: "OK" and "Cancel".

รูปที่ 3.8 การออกแบบเมนูพยากรณ์รายจ่าย

จากรูปที่ 3.8 เป็นการออกแบบเมนูการพยากรณ์รายจ่ายซึ่งจะมีช่องให้กรอกปีและเดือนที่ต้องการดูทั้งนี้การใช้เมนูพยากรณ์จำเป็นต้องมีข้อมูลย้อนหลังเก็บไว้ในฐานข้อมูลอย่างน้อย 3 เดือน



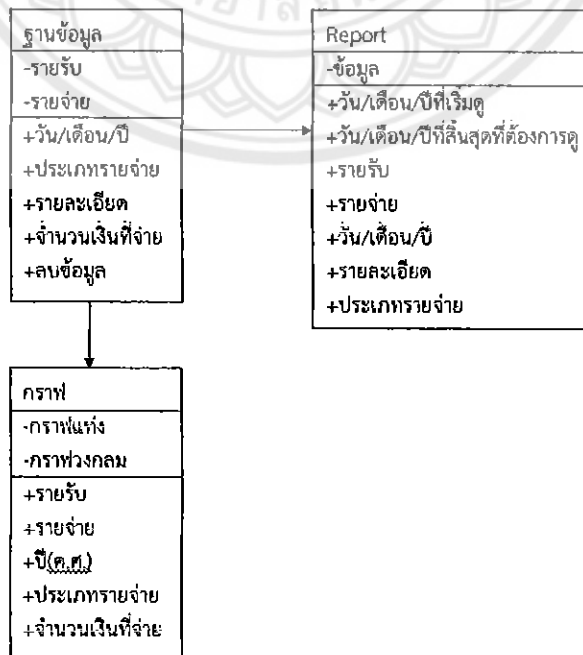
3.3 Use Case Diagram ของแอปพลิเคชัน



รูปที่ 3.9 Use Case Diagram ของแอปพลิเคชัน

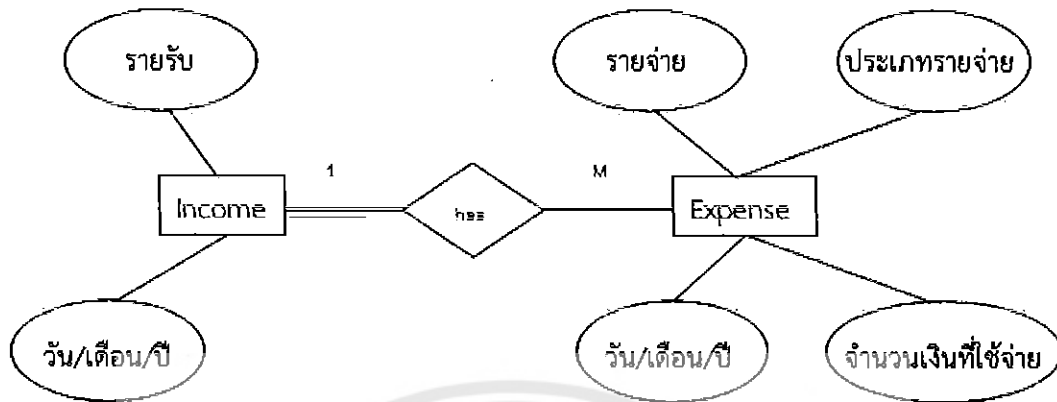
จากรูปที่ 3.9 Use Case Diagram ของแอปพลิเคชัน User จะป้อน Input นั่นคือ รายรับและรายจ่าย เข้ามาจากนั้นข้อมูลจะถูกนำไปเก็บไว้ในฐานข้อมูล จากนั้น เมื่อ User ต้องการเรียกใช้เมนู Report ,Forecast หรือ Chart นั้น โปรแกรมก็จะนำข้อมูลของผู้ใช้ที่ถูกเก็บไว้ในฐานข้อมูลข้างต้นมาแสดงตามรูปแบบของเมนู ส่วน เมนู Manage จะใช้สำหรับการลบข้อมูลเมื่อไม่ต้องการข้อมูล

3.4 Class Diagram ของแอปพลิเคชัน



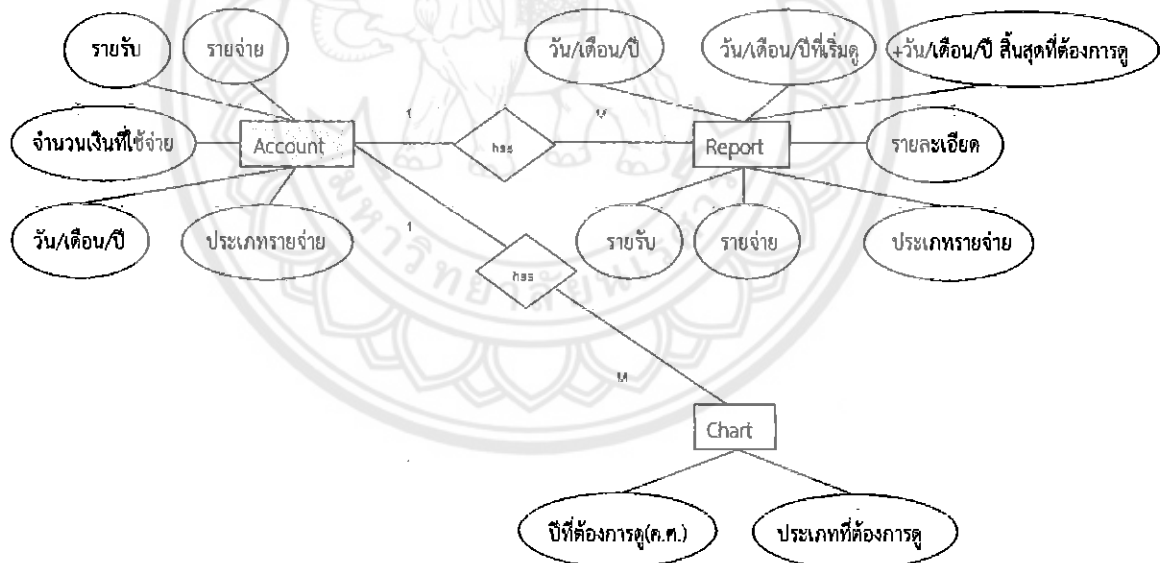
รูปที่ 3.10 Class Diagram ของแอปพลิเคชัน

3.5 การออกแบบฐานข้อมูลที่ Input เข้ามา



รูปที่ 3.11 การออกแบบฐานข้อมูลที่ Input เข้ามา

3.5.1 การออกแบบฐานข้อมูลของแอปพลิเคชัน



รูปที่ 3.12 การออกแบบฐานข้อมูลของแอปพลิเคชัน

จากรูปที่ 3.12 เป็นการออกแบบฐานข้อมูลของแอปพลิเคชันเมื่อมีการ Input ข้อมูลเข้ามา ข้อมูลนั้นก็จะถูกนำไปเก็บไว้ในฐานข้อมูลเพื่อรอการนำไปใช้ จาก Report และ Chart ใน 1 บัญชีของผู้ใช้จะสามารถมีหลาย Report และ Chart แต่ Report และ Chart ของบัญชีนั้นจะใช้ได้กับบัญชีนั้นบัญชีเดียว

3.6 ตัวอย่างการจำลองตารางการใช้จ่ายภายใน 1 สัปดาห์ที่จะนำไปเก็บที่ฐานข้อมูล ตารางที่ 3.1 ตัวอย่างการจำลองตารางการใช้จ่ายภายใน 1 สัปดาห์ที่จะนำไปเก็บที่ฐานข้อมูล

วัน/เดือน/ปี	รายรับ	รายจ่าย						จำนวนรายจ่าย
		ประเภทรายจ่าย						
		ค่าอาหาร	ค่ารักษาพยาบาล	ค่าพาหนะ	ค่าเสื้อผ้า	ค่าใช้จ่ายภายในบ้าน	อื่นๆ	
1/1/2012	15,000	✓		✓		✓	✓	750
2/1/2012		✓		✓				500
3/1/2012		✓		✓	✓	✓	✓	1,500
4/1/2012		✓	✓	✓	✓		✓	1,200
5/1/2012		✓		✓				550
6/1/2012		✓		✓		✓	✓	870
7/1/2012		✓		✓		✓		600

จากตารางที่ 3.1 เป็นการจำลองการใช้จ่ายภายใน 1 สัปดาห์ค่าใช้จ่ายเหล่านี้จะถูกนำไปเก็บในฐานข้อมูลเพื่อรอการนำไปใช้จากเมนูอื่น

3.6.1 ตัวอย่างการจำลองตารางการใช้จ่ายภายใน 1 วันที่จะนำไปเก็บที่ฐานข้อมูล ตารางที่ 3.2 ตัวอย่างการจำลองตารางการใช้จ่ายภายใน 1 วันที่จะนำไปเก็บที่ฐานข้อมูล

วันที่ 1/2/2012		
ประเภทรายจ่าย	จำนวนที่ใช้จ่าย	รายละเอียดของรายจ่าย
ค่าอาหาร	250	อาหารเช้า,อาหารกลางวัน,อาหารเย็น,น้ำเปล่า,ขนม
ค่ารักษาพยาบาล	290	ค่าวิตามิน
ค่าพาหนะ	150	ค่ารถเมล์,ค่าแท็กซี่
ค่าเสื้อผ้า	500	ค่าเสื้อ
ค่าใช้จ่ายภายในบ้าน	1,300	ค่าปากกา,ค่าเก้าอี้,ค่าไฟฟ้า
อื่นๆ	1,200	ค่าน้ำมัน
	รวม=3,690	

จากตารางที่ 3.1 จะเห็นว่ารายรับเข้ามาที่ 15,000 บาทจากนั้นก็จะมีค่าใช้จ่ายต่างๆ ซึ่งข้อมูลเหล่านี้จะถูกเก็บไว้ในฐานข้อมูลเพื่อรอการนำไปใช้จากเมนู Report และ Chart และ จากตารางที่ 3.2 เป็นการจำลองการใช้จ่ายใน 1 วัน แล้วนำข้อมูลนั้นมาเก็บไว้ในฐานข้อมูลเพื่อรอการนำไปใช้จากเมนู Report และ Chart

หลังจากศึกษาวิธีการต่างๆ ได้เริ่มออกแบบและพัฒนาฐานข้อมูล รูปร่างและ โครงสร้างของโปรแกรมในเมนูต่างๆ ว่าจะมีในลักษณะไหน โดยการออกแบบนี้จะต้องนำมาประยุกต์เข้ากับโปรแกรม Eclipse IDE 3.7 , Android SDK และ Sqlite บนเวอร์ชัน 2.2 โดยการออกแบบในขั้นต้นก็จะกำหนดให้โปรแกรมสามารถรับและลบข้อมูล รายรับ รายจ่ายได้ และเก็บข้อมูลในรูปแบบสถิติและกราฟได้เพื่อที่ในอนาคตต่อไปของเรามาจะสามารถนำฐานข้อมูลเข้ามาในโปรแกรมได้ อีกทั้งโปรแกรมต้องสามารถพยากรณ์รายจ่ายจากวิธีการที่ได้กล่าวมาได้โดยในอนาคตต่อไปจะพูดถึง การทดสอบระบบซึ่งเป็นการนำฐานข้อมูลเข้ามาในโปรแกรม การเรียกดูกราฟวงกลมและกราฟแท่ง การบันทึกข้อมูล การเรียกดูข้อมูลย้อนหลัง การพยากรณ์รายจ่าย และ จัดสรรรายละเอียดต่างๆ ในโปรแกรม



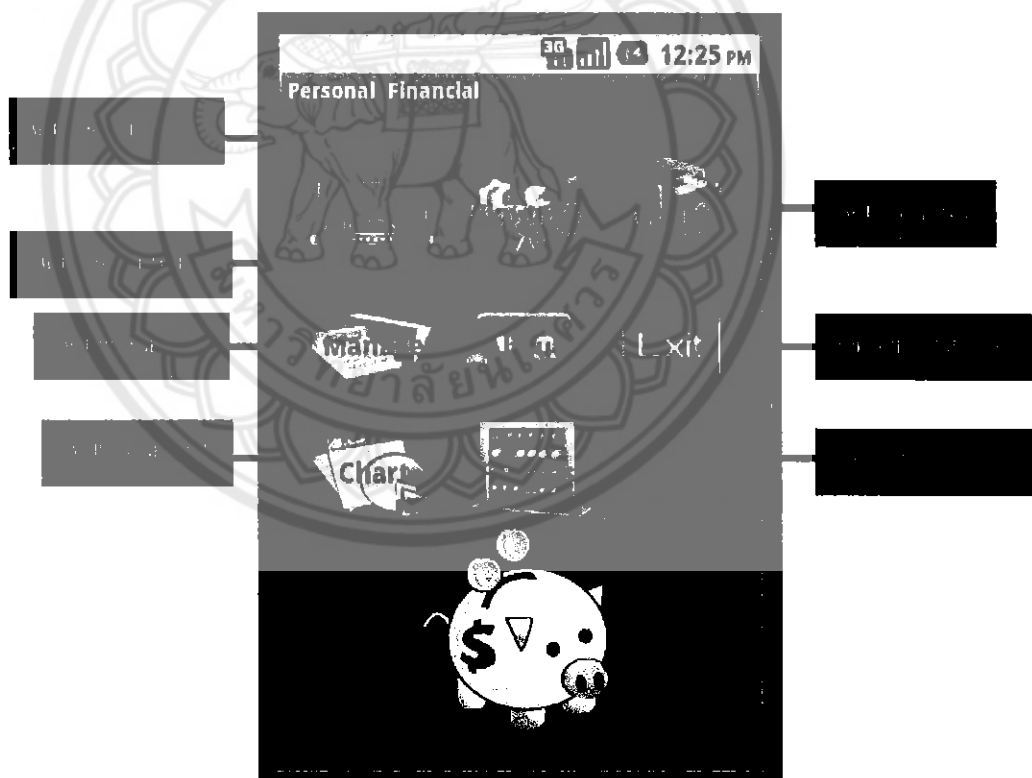
บทที่ 4

การทดสอบระบบ

จากบทที่ผ่านมาได้กล่าวถึงระบบในส่วนต่างๆ รวมไปถึงการพัฒนาโปรแกรมเพื่อนำมาใช้งานและในบทนี้จะเป็นการทดสอบการทำงานของโปรแกรม โดยการทดสอบนี้จะแบ่งเป็นการทดสอบในส่วนของการรับข้อมูล การลบข้อมูล การดูรายรับ รายจ่ายผ่านทางกราฟ ไฟล์เอกสาร เพื่อดูว่าโปรแกรมทำตามที่ออกแบบได้หรือไม่

4.1 การเพิ่มข้อมูลในฐานข้อมูล

การทดสอบการรับข้อมูล ทั้งรายรับและรายจ่าย ว่าสามารถบันทึกลงสู่ฐานข้อมูลได้หรือไม่

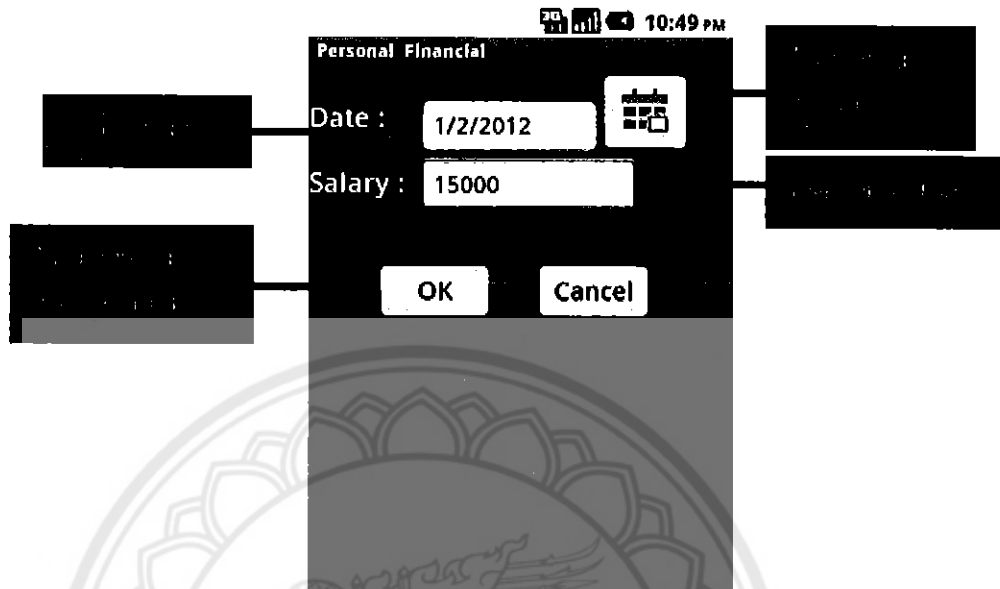


รูปที่ 4.1 รูปแสดงเมนูต่างๆของแอปพลิเคชัน

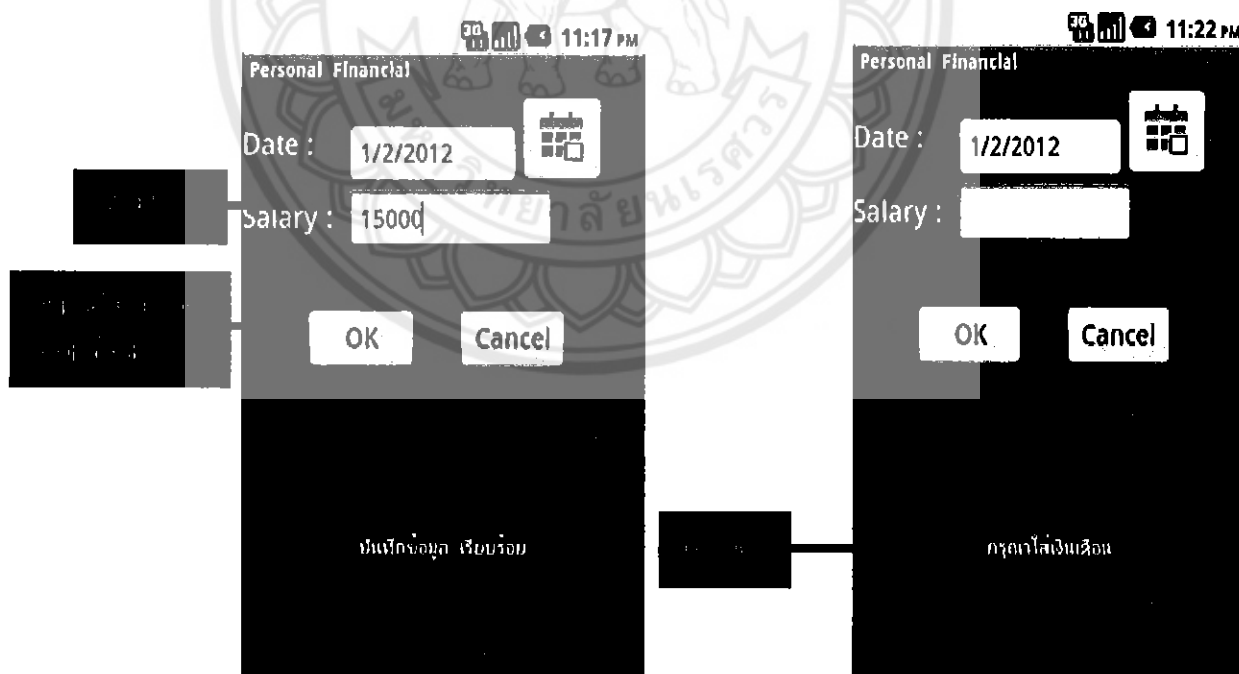
จากรูปที่ 4.1 จะแสดงเมนูต่างๆของแอปพลิเคชันคือ เมนู Income ใช้เพื่อกรอกรายรับ เมนู Expense ใช้เพื่อกรอกรายจ่าย เมนู Report ใช้เพื่อเรียกดูสถิติของรายรับ รายจ่าย เมนู Manage ใช้สำหรับ ลบข้อมูลที่ไม่ต้องการ และเมนู Chart ใช้สำหรับเรียกดูกราฟแท่งของรายรับ รายจ่าย

4.1.1 การเพิ่มข้อมูลรายรับ

รูปที่ 4.1 แสดงการเพิ่มข้อมูลรายรับเข้าสู่ระบบซึ่งข้อมูลที่ใส่จะถูกจัดเก็บไว้ในฐานข้อมูลเพื่อรอการนำข้อมูลดังกล่าวมาใช้ต่อไป



รูปที่ 4.2 แสดงการเพิ่มข้อมูลรายรับ



รูปที่ 4.3 แสดงการบันทึกการรายรับเรียบร้อย

รูปที่ 4.4 แสดงการแจ้งเตือนให้ใส่เงินเดือน

รูปที่ 4.3 แสดงการบันทึกข้อมูลเรียบร้อยส่วนรูปที่ 4.4 แสดงการแจ้งเตือนกรณีที่ user ไม่ได้ใส่เงินเดือนแล้วกด OK

4.1.2 การเพิ่มข้อมูลรายจ่าย

Personal Financial

Date : 1/2/2012

Type : อาหาร

Detail :

Price : 50

OK Cancel

บันทึกข้อมูล เก็บประวัติ

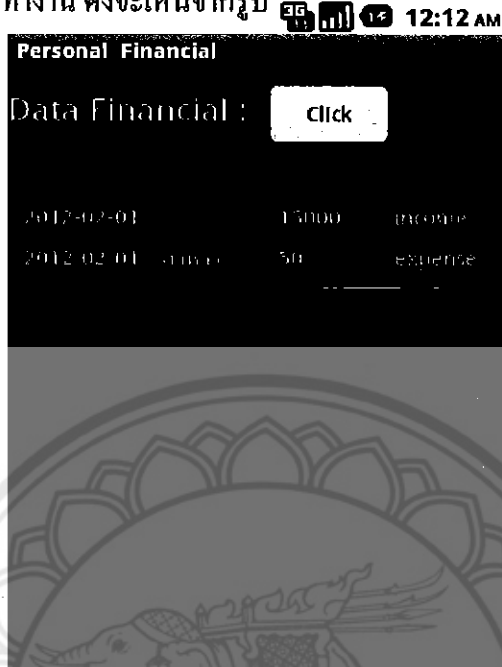
รูปที่ 4.5 แสดงการจำแนกประเภทของรายจ่าย

จากรูปที่ 4.5 เป็นการเพิ่มข้อมูลรายจ่ายเข้าสู่ระบบ ซึ่งจะประกอบด้วยมีช่องกรอก วัน/เดือน/ปี ที่ทำการบันทึก, ช่องเลือกประเภทของรายจ่าย ,ช่องใส่รายละเอียดของรายจ่าย ,และ ช่องกรอกราคาของรายจ่าย ข้อมูลทั้งหมดจะถูกนำไปเก็บที่ฐานข้อมูลเพื่อรอในการนำไปใช้ต่อไป ซึ่งชนิดของข้อมูลรายจ่ายที่ทำการจัดเก็บจำแนกออกเป็น 6 ประเภทดังนี้

1. ค่าอาหาร
2. ค่ารักษาพยาบาล
3. ค่าพาหนะ
4. ค่าเสื้อผ้า
5. ค่าใช้จ่ายภายในบ้าน
6. อื่นๆ

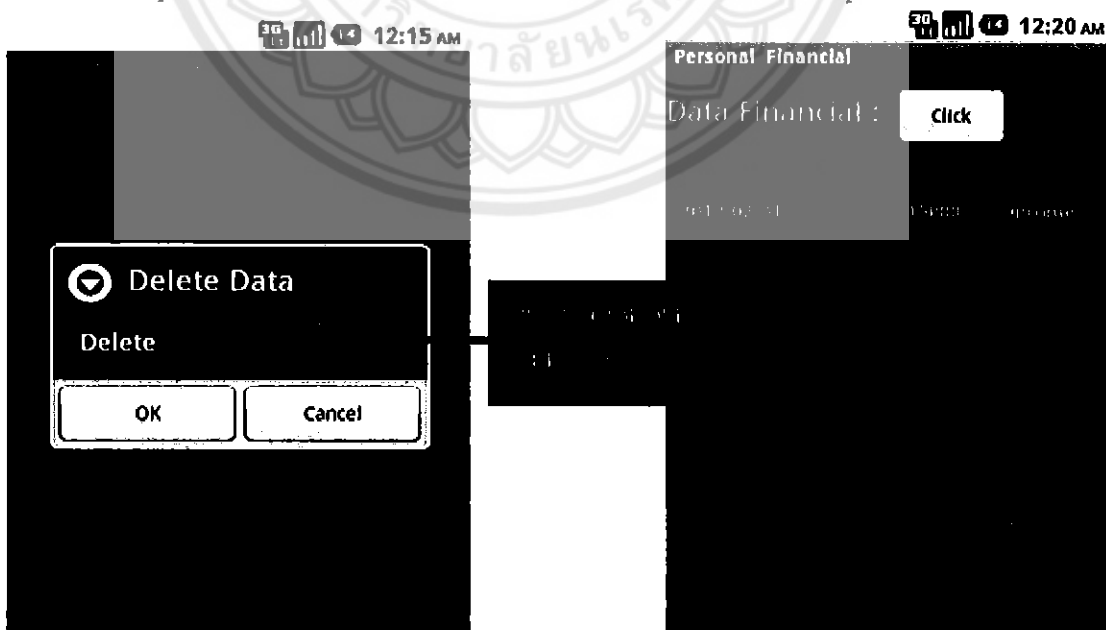
4.2 การลบข้อมูลในฐานข้อมูล

เมื่อมีการนำข้อมูลเข้ามาแน่นอนจำเป็นจะต้องมีระบบกำจัดข้อมูลที่ไม่ต้องการหรือไม่จำเป็นเพื่อลดพื้นที่การทำงาน ดังจะเห็นจากรูป



รูปที่ 4.6 แสดงข้อมูลรายรับ รายจ่ายที่มีก่อนทำการลบข้อมูล

เลือกข้อมูลที่ไม่ต้องการจากรายการบันทึกแล้วกด OK เพื่อขึ้นชั้นการลบข้อมูล



รูปที่ 4.7 แสดงการลบข้อมูล

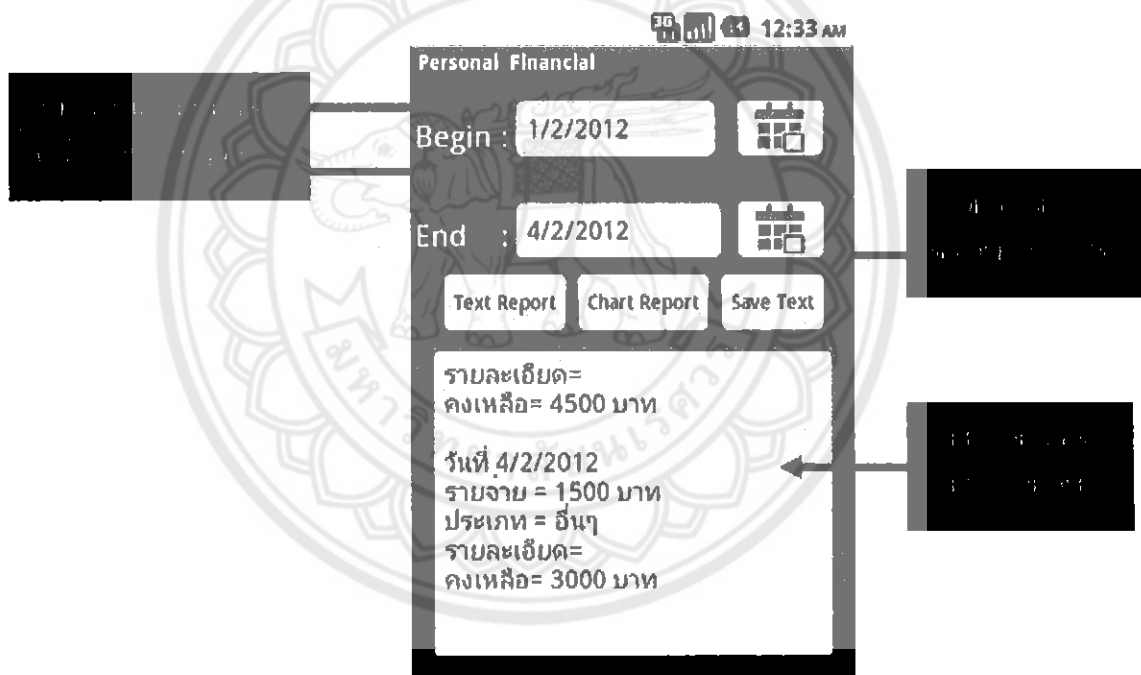
รูปที่ 4.8 รูปหลังจากลบข้อมูลเรียบร้อยแล้ว

4.3 การเรียกดูข้อมูลรายรับ-รายจ่าย

เป็นการดูข้อมูล โดยเข้าที่เมนู Report ของโปรแกรม เพื่อดูว่าข้อมูลของระบบรายรับรายจ่าย ต่อเดือนว่าภายในเดือนเราใช้จ่ายอะไรไปบ้าง โดยได้ทำการทดสอบข้อมูลดังนี้

1. เงินเดือน 15000 บาท
2. ค่าอาหาร 3000 บาท
3. ค่ารักษาพยาบาล 500 บาท
4. ค่าพาหนะ 2000 บาท
5. ค่าเสื้อผ้า 1000 บาท
6. ค่าใช้จ่ายภายในบ้าน 4000 บาท
7. อื่นๆ 1500 บาท

ซึ่งรวมจ่ายทั้งหมดจะเท่ากับ 12000 บาท



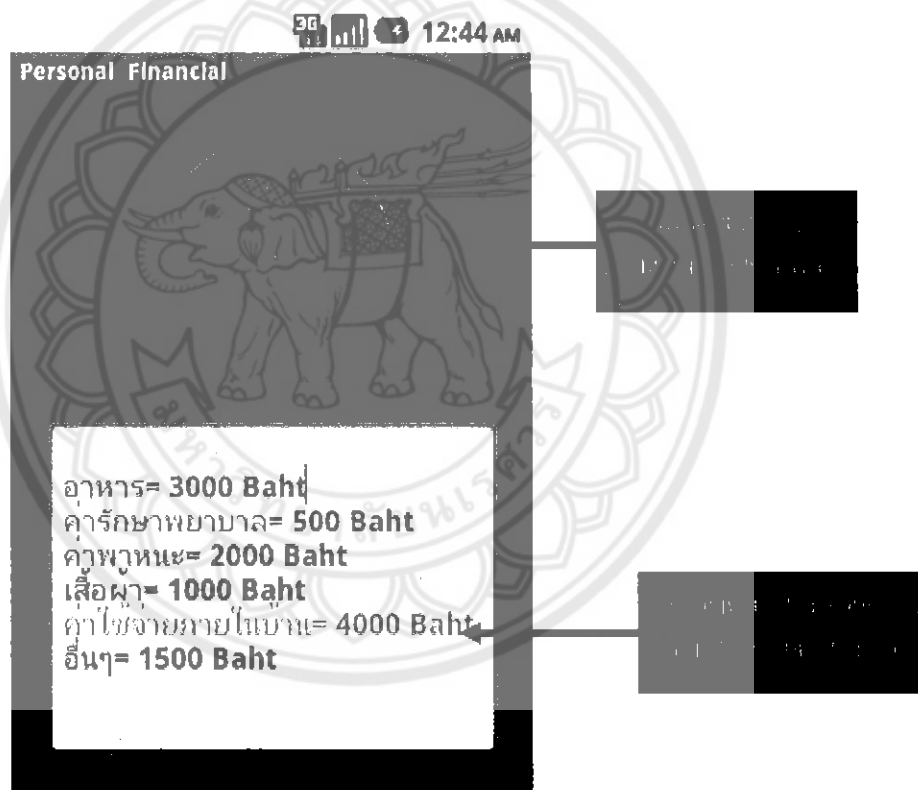
รูปที่ 4.9 แสดง Report ที่ user เรียกดู

จากรูปที่ 4.9 จะแสดงรายงานที่ user ต้องการดูตั้งแต่ วันที่ 1/2/2012 จนถึงวันที่ 4/2/2012 จากรายรับ 15,000 บาท และรายจ่ายโดยรวมคือ 12,000 บาท โปรแกรมก็จะนำข้อมูลข้างต้นมาประมวลผลและแสดงยอดคงเหลือจะเท่ากับ 3,000 บาท และสามารถดูข้อมูลในรูปแบบกราฟวงกลมได้โดยการเข้าเมนู Chart Report

4.3.1 การดูข้อมูลรายรับ รายจ่าย ในรูปแบบกราฟวงกลม

เป็นการนำข้อมูลของระบบรายรับ รายจ่ายต่อเดือนมาแสดงผลในรูปแบบกราฟวงกลมเพื่อ มุมมองที่ง่ายขึ้น และจะเห็นได้ว่าภายในเดือนเราใช้จ่ายในส่วนไหนเยอะและน้อย โดยจะนำ ประเภทของการใช้จ่ายมาจำแนกเป็นแต่ละสี ดังรูปที่ 4.10 โดยใช้ข้อมูลดังนี้

1. ค่าอาหาร 3000 บาท
2. ค่ารักษาพยาบาล 500 บาท
3. ค่าพาหนะ 2000 บาท
4. ค่าเสื้อผ้า 1000 บาท
5. ค่าใช้จ่ายภายในบ้าน 4000 บาท
6. อื่นๆ 1500 บาท



รูปที่ 4.10 แสดงค่าใช้จ่ายของแต่ละประเภทรูปแบบกราฟวงกลม

4.3.2 การนำข้อมูลรายรับ รายจ่าย บันทึกลงไฟล์ .txt

เป็นการนำข้อมูลรายรับ รายจ่ายต่อเดือนที่เลือกมาลงบันทึกเป็นไฟล์เอกสารเพื่อนำเก็บไว้ในกรณีที่ต้องการสำรองข้อมูล หรือนำไปใช้อีกอย่างอื่น



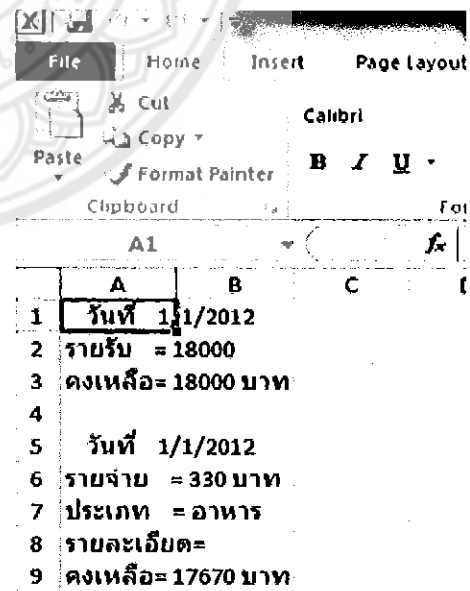
รูปที่ 4.11 เรียกดูข้อมูลก่อนการบันทึก



รูปที่ 4.12 ใส่ชื่อ Text File ที่ต้องการบันทึก



รูปที่ 4.13 Text File บันทึกเรียบร้อยแล้ว



รูปที่ 4.14 รูปแบบ Text File ที่ได้รับ

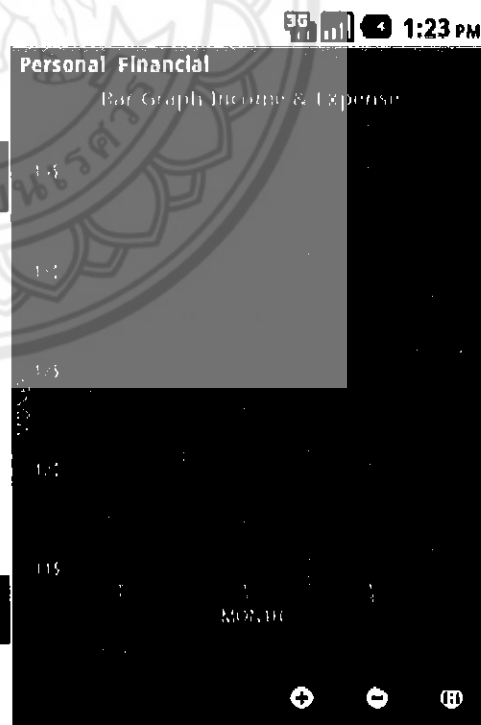
4.4 การดูข้อมูลรายรับ รายจ่าย ต่อปีในรูปแบบกราฟแท่ง

เป็นการนำข้อมูลรายรับ รายจ่ายทั้งปีจากฐานข้อมูล มาเปรียบเทียบกัน จะเห็นได้ว่าในแต่ละเดือนภายในปี เรามีรายรับและรายจ่ายเป็นอย่างไร โดยทดสอบข้อมูลดังนี้

1. เดือนมกราคม เงินเดือน 15000 บาท ค่าใช้จ่ายทั้งหมด 12000 บาท
2. เดือนกุมภาพันธ์ เงินเดือน 17000 บาท ค่าใช้จ่ายทั้งหมด 13000 บาท
3. เดือนมีนาคม เงินเดือน 15000 บาท ค่าใช้จ่ายทั้งหมด 20000 บาท
4. เดือนเมษายน เงินเดือน 18000 บาท ค่าใช้จ่ายทั้งหมด 14000 บาท
5. เดือนพฤษภาคม เงินเดือน 15000 บาท ค่าใช้จ่ายทั้งหมด 16000 บาท
6. เดือนมิถุนายน เงินเดือน 15000 บาท ค่าใช้จ่ายทั้งหมด 17000 บาท
7. เดือนกรกฎาคม เงินเดือน 19000 บาท ค่าใช้จ่ายทั้งหมด 12000 บาท
8. เดือนสิงหาคม เงินเดือน 16000 บาท ค่าใช้จ่ายทั้งหมด 13000 บาท
9. เดือนกันยายน เงินเดือน 18000 บาท ค่าใช้จ่ายทั้งหมด 15000 บาท
10. เดือนตุลาคม เงินเดือน 16000 บาท ค่าใช้จ่ายทั้งหมด 12000 บาท
11. เดือนพฤศจิกายน เงินเดือน 15000 บาท ค่าใช้จ่ายทั้งหมด 12000 บาท
12. เดือนธันวาคม เงินเดือน 15000 บาท ค่าใช้จ่ายทั้งหมด 12000 บาท



รูปที่ 4.15 กราฟแท่ง รายรับ รายจ่ายต่อปี



รูปที่ 4.16 กราฟแท่ง รายรับ รายจ่ายต่อปีแบบวงกลม

4.4.1 การดูข้อมูลรายจ่ายแยกประเภท ต่อปีรูปแบบกราฟแท่ง

เป็นการเปรียบเทียบข้อมูลรายจ่าย ของแต่ละเดือนภายใน1ปี ว่าแต่ละเดือนเรามีรายจ่ายแต่ละประเภทเป็นอย่างไร

4.4.1.1 การดูข้อมูลรายจ่าย ค่าอาหาร ต่อปี

เป็นการทดสอบดูรายจ่ายประเภท ค่าอาหาร แต่ละเดือนเปรียบเทียบกัน โดยมีการทดสอบข้อมูลดังนี้

- | | |
|--------------------------------------|--------------------------------------|
| 1. เดือนมกราคม ค่าอาหาร 3300 บาท | 2. เดือนกุมภาพันธ์ ค่าอาหาร 4000 บาท |
| 3. เดือนมีนาคม ค่าอาหาร 3500 บาท | 4. เดือนเมษายน ค่าอาหาร 4200 บาท |
| 5. เดือนพฤษภาคม ค่าอาหาร 3900 บาท | 6. เดือนมิถุนายน ค่าอาหาร 3100 บาท |
| 7. เดือนกรกฎาคม ค่าอาหาร 3400 บาท | 8. เดือนสิงหาคม ค่าอาหาร 3500 บาท |
| 9. เดือนกันยายน ค่าอาหาร 3800 บาท | 10. เดือนตุลาคม ค่าอาหาร 3900 บาท |
| 11. เดือนพฤศจิกายน ค่าอาหาร 3700 บาท | 12. เดือนธันวาคม ค่าอาหาร 3800 บาท |



รูปที่ 4.17 กราฟแท่งค่าอาหาร ต่อปี



รูปที่ 4.18 กราฟแท่งค่าอาหาร ต่อปีแบบชุม

4.4.1.2 การดูข้อมูลรายจ่าย ค่ารักษาพยาบาล ต่อปี

เป็นการทดสอบดูรายจ่ายประเภท ค่ารักษาพยาบาล แต่ละเดือนเปรียบเทียบกัน โดยมีการทดสอบข้อมูลดังนี้

1. เดือนมกราคม ค่ารักษาพยาบาล 500 บาท
2. เดือนกุมภาพันธ์ ค่ารักษาพยาบาล 600 บาท
3. เดือนมีนาคม ค่ารักษาพยาบาล 700 บาท
4. เดือนเมษายน ค่ารักษาพยาบาล 550 บาท
5. เดือนพฤษภาคม ค่ารักษาพยาบาล 580 บาท
6. เดือนมิถุนายน ค่ารักษาพยาบาล 800 บาท
7. เดือนกรกฎาคม ค่ารักษาพยาบาล 520 บาท
8. เดือนสิงหาคม ค่ารักษาพยาบาล 650 บาท
9. เดือนกันยายน ค่ารักษาพยาบาล 750 บาท
10. เดือนตุลาคม ค่ารักษาพยาบาล 740 บาท
11. เดือนพฤศจิกายน ค่ารักษาพยาบาล 540 บาท
12. เดือนธันวาคม ค่ารักษาพยาบาล 580 บาท



รูปที่ 4.19 กราฟแท่งค่ารักษาพยาบาล ต่อปี



รูปที่ 4.20 กราฟแท่งค่ารักษาพยาบาลต่อปีแบบซุม

4.4.1.3 การดูข้อมูลรายจ่าย ค่าพาหนะ ต่อปี

เป็นการทดสอบดูรายจ่ายประเภท ค่าพาหนะ แต่ละเดือนเปรียบเทียบกับ โดยมีการทดสอบข้อมูลดังนี้

1. เดือนมกราคม ค่าพาหนะ 2100 บาท
2. เดือนกุมภาพันธ์ ค่าพาหนะ 2300 บาท
3. เดือนมีนาคม ค่าพาหนะ 2200 บาท
4. เดือนเมษายน ค่าพาหนะ 2500 บาท
5. เดือนพฤษภาคม ค่าพาหนะ 2400 บาท
6. เดือนมิถุนายน ค่าพาหนะ 2600 บาท
7. เดือนกรกฎาคม ค่าพาหนะ 2800 บาท
8. เดือนสิงหาคม ค่าพาหนะ 2200 บาท
9. เดือนกันยายน ค่าพาหนะ 2500 บาท
10. เดือนตุลาคม ค่าพาหนะ 2600 บาท
11. เดือนพฤศจิกายน ค่าพาหนะ 2400 บาท
12. เดือนธันวาคม ค่าพาหนะ 2200 บาท



รูปที่ 4.21 กราฟแท่งค่าพาหนะ ต่อปี

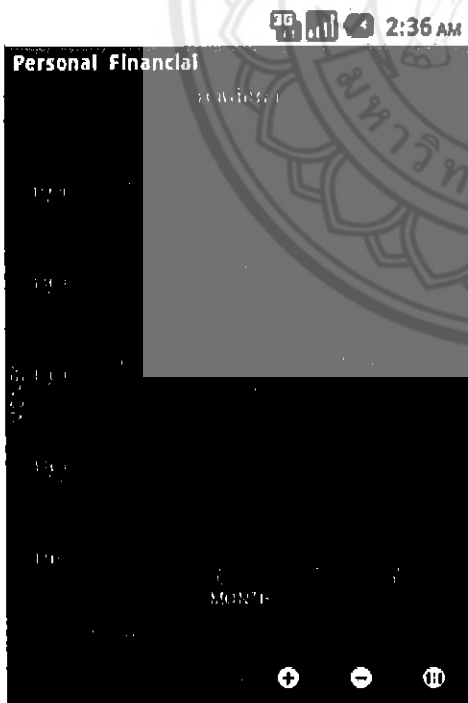


รูปที่ 4.22 กราฟแท่งค่าพาหนะต่อปีแบบชุม

4.4.1.4 การดูข้อมูลรายจ่าย ค่าเสื้อผ้า ต่อปี

เป็นการทดสอบดูรายจ่ายประเภท ค่าพาหนะ แต่ละเดือนเปรียบเทียบกับกัน โดยมีกรทดสอบข้อมูลดังนี้

1. เดือนมกราคม ค่าเสื้อผ้า 1200 บาท
2. เดือนกุมภาพันธ์ ค่าเสื้อผ้า 1300 บาท
3. เดือนมีนาคม ค่าเสื้อผ้า 1500 บาท
4. เดือนเมษายน ค่าเสื้อผ้า 1400 บาท
5. เดือนพฤษภาคม ค่าเสื้อผ้า 1300 บาท
6. เดือนมิถุนายน ค่าเสื้อผ้า 1400 บาท
7. เดือนกรกฎาคม ค่าเสื้อผ้า 1600 บาท
8. เดือนสิงหาคม ค่าเสื้อผ้า 1700 บาท
9. เดือนกันยายน ค่าเสื้อผ้า 1300 บาท
10. เดือนตุลาคม ค่าเสื้อผ้า 1500 บาท
11. เดือนพฤศจิกายน ค่าเสื้อผ้า 1200 บาท
12. เดือนธันวาคม ค่าเสื้อผ้า 1300 บาท



รูปที่ 4.23 กราฟแท่งค่าเสื้อผ้า ต่อปี



รูปที่ 4.24 กราฟแท่งค่าเสื้อผ้าต่อปี
แบบวงกลม

4.4.1.5 การดูข้อมูลรายจ่าย ค่าใช้จ่ายภายในบ้าน ต่อปี

เป็นการทดสอบดูรายจ่ายประเภท ค่าใช้จ่ายภายในบ้าน แต่ละเดือนเปรียบเทียบกับกัน โดยมี การทดสอบข้อมูลดังนี้

1. เดือนมกราคม ค่าใช้จ่ายภายในบ้าน 5500 บาท
2. เดือนกุมภาพันธ์ ค่าใช้จ่ายภายในบ้าน 5400 บาท
3. เดือนมีนาคม ค่าใช้จ่ายภายในบ้าน 5600 บาท
4. เดือนเมษายน ค่าใช้จ่ายภายในบ้าน 5200 บาท
5. เดือนพฤษภาคม ค่าใช้จ่ายภายในบ้าน 5900 บาท
6. เดือนมิถุนายน ค่าใช้จ่ายภายในบ้าน 6000 บาท
7. เดือนกรกฎาคม ค่าใช้จ่ายภายในบ้าน 5000 บาท
8. เดือนสิงหาคม ค่าใช้จ่ายภายในบ้าน 5200 บาท
9. เดือนกันยายน ค่าใช้จ่ายภายในบ้าน 5500 บาท
10. เดือนตุลาคม ค่าใช้จ่ายภายในบ้าน 5600 บาท
11. เดือนพฤศจิกายน ค่าใช้จ่ายภายในบ้าน 5400 บาท
12. เดือนธันวาคม ค่าใช้จ่ายภายในบ้าน 5500 บาท



รูปที่ 4.25 กราฟแท่งค่าใช้จ่ายภายในบ้าน ต่อปี



รูปที่ 4.26 กราฟแท่งค่าใช้จ่ายภายในบ้านต่อปีแบบวงกลม

4.4.1.6 การดูข้อมูลรายจ่าย ค่าใช้จ่ายอื่นๆ ต่อปี

เป็นการทดสอบดูรายจ่ายประเภท ค่าใช้จ่ายอื่นๆ แต่ละเดือนเปรียบเทียบกับ โดยมีการทดสอบข้อมูลดังนี้

1. เดือนมกราคม ค่าใช้จ่ายอื่นๆ 2000 บาท
2. เดือนกุมภาพันธ์ ค่าใช้จ่ายอื่นๆ 2200 บาท
3. เดือนมีนาคม ค่าใช้จ่ายอื่นๆ 2500 บาท
4. เดือนเมษายน ค่าใช้จ่ายอื่นๆ 2400 บาท
5. เดือนพฤษภาคม ค่าใช้จ่ายอื่นๆ 2200 บาท
6. เดือนมิถุนายน ค่าใช้จ่ายอื่นๆ 2400 บาท
7. เดือนกรกฎาคม ค่าใช้จ่ายอื่นๆ 2100 บาท
8. เดือนสิงหาคม ค่าใช้จ่ายอื่นๆ 2200 บาท
9. เดือนกันยายน ค่าใช้จ่ายอื่นๆ 2300 บาท
10. เดือนตุลาคม ค่าใช้จ่ายอื่นๆ 2400 บาท
11. เดือนพฤศจิกายน ค่าใช้จ่ายอื่นๆ 2200 บาท
12. เดือนธันวาคม ค่าใช้จ่ายอื่นๆ 2300 บาท



รูปที่ 4.27 กราฟแท่งค่าใช้จ่ายอื่นๆ ต่อปี



รูปที่ 4.28 กราฟแท่งค่าใช้จ่ายอื่นๆต่อปีแบบซุม

4.5 การใช้เมนูพยากรณ์รายจ่าย

การพยากรณ์รายจ่ายจำเป็นต้องมีข้อมูลอยู่ก่อนปีที่ต้องการพยากรณ์โดยโปรแกรมสามารถคำนวณได้ 2 วิธีคือ

1. Moving Averaging
2. Exponential

The screenshot shows a mobile application interface titled 'Personal Financial'. It features two menu options for forecasting: 'Moving Averaging' and 'Exponential'. Each option includes input fields for 'Month' and 'Year' and an 'OK' button. The interface is displayed on a smartphone screen with a status bar at the top showing the time as 12:34 PM.

รูปที่ 4.29 เมนูพยากรณ์รายจ่าย

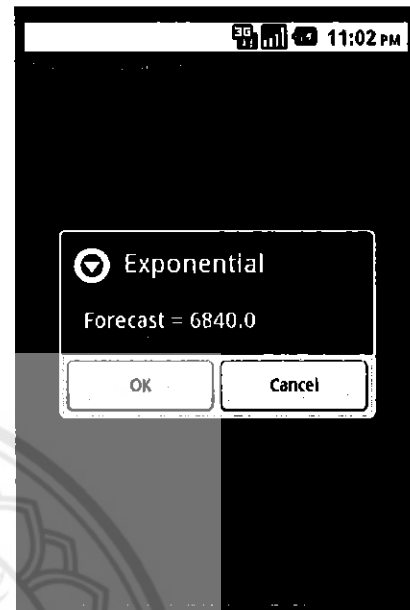
จากรูปที่ 4.28 เป็นเมนูการพยากรณ์รายจ่ายจะเห็นว่ามี 2 วิธีคือแบบ Moving Averaging และแบบ Exponential ซึ่งช่อง Month และ Year มีไว้สำหรับใส่เดือนและปีที่ต้องการจะพยากรณ์ตามลำดับ จากนั้นทำการทดสอบโปรแกรม โดยการจำลองรายจ่ายมาเพื่อทำการพยากรณ์ทั้ง 2 วิธี

ตารางที่ 4.1 จำลองค่าใช้จ่ายในแต่ละเดือน (1)

เดือน	ค่าใช้จ่าย	การพยากรณ์รายจ่าย	
		แบบค่าเฉลี่ยเคลื่อนที่	แบบเอกซ์โปเนนเชียล
มกราคม	6,550		
กุมภาพันธ์	7,200		
มีนาคม	6,400		
เมษายน	6,900		
พฤษภาคม	6,300		
มิถุนายน	6,750		
กรกฎาคม	6,850		
สิงหาคม	6,700		

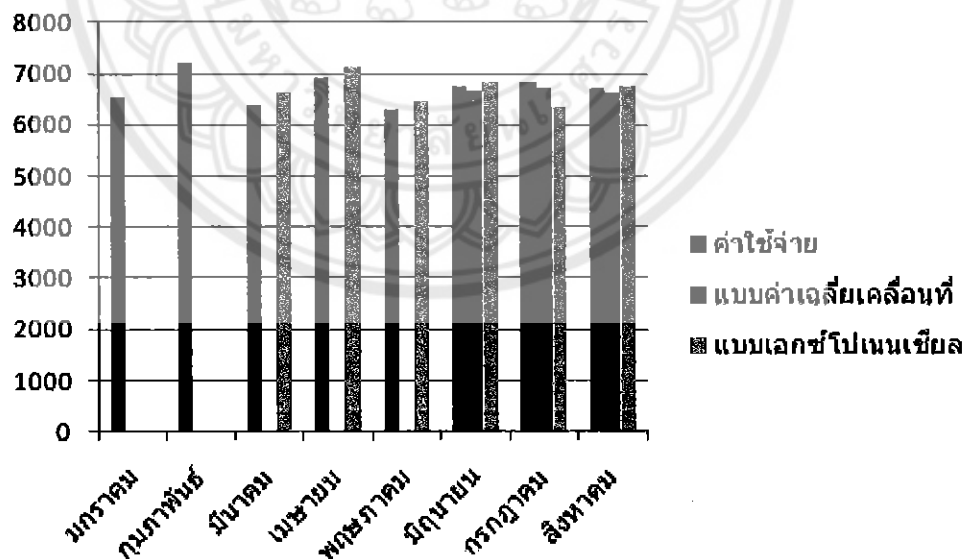
จากตารางที่ 4.1 เป็นจำลองค่าใช้จ่ายในแต่ละเดือนซึ่งเราได้คำนวณค่าพยากรณ์ของแต่ละแบบไว้แล้วซึ่งเราจะทำการทดสอบระบบเพื่อดูว่าค่าที่ได้จะสอดคล้องหรือไม่ซึ่งข้อมูลในตารางที่ 4.1 จะเป็นข้อมูลที่มีรายจ่ายแบบกระจายตัว

เมื่อนำข้อมูลจากตารางที่ 4.1 ป้อนเข้าไปใน โปรแกรมแล้วก็ทำการทดสอบโปรแกรมโดยทำการพยากรณ์รายจ่ายเดือนที่ 6 หรือ มิถุนายน



รูปที่ 4.30 การใช้เมนูพยากรณ์รายจ่าย
ด้วยวิธี Moving Averaging (1)

รูปที่ 4.31 การใช้เมนูพยากรณ์รายจ่าย
ด้วยวิธี Exponential (1)



รูปที่ 4.32 กราฟแสดงการเปรียบเทียบระหว่างค่าใช้จ่ายจริงกับการพยากรณ์ค่าใช้จ่าย (1)

จากรูปที่ 4.30 และรูปที่ 4.31 จะเห็นได้ว่า ข้อมูลที่ได้สอดคล้องกับตารางที่ 4.1 และเมื่อดูข้อมูลในรูปที่ 4.32 แล้วจะเห็นได้ว่าค่าที่ได้จากการพยากรณ์ทั้ง 2 แบบมีค่าใกล้เคียงกับค่าจริง

ตารางที่ 4.2 จําลองค่าใช้จ่ายในแต่ละเดือน (2)

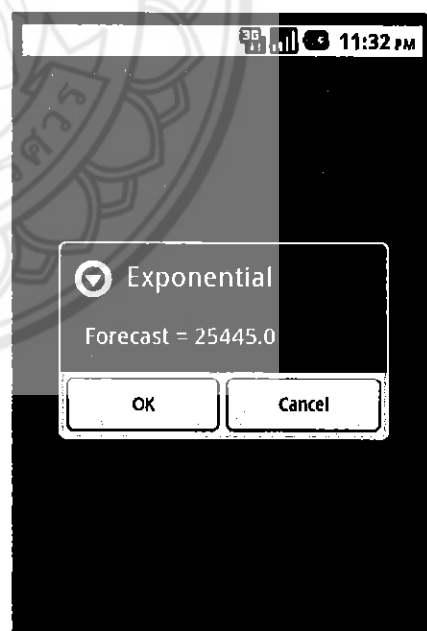
เดือน	ค่าใช้จ่าย	การพยากรณ์รายจ่าย	
			แบบเอกซ์โปเนนเชียล
มกราคม	30,050		
กุมภาพันธ์	22,000		
มีนาคม	28,000		
เมษายน	23,100		
พฤษภาคม	25,550		
มิถุนายน	24,500		
กรกฎาคม	24,390		
สิงหาคม	25,500		

จากตารางที่ 4.1 เป็นจําลองค่าใช้จ่ายในแต่ละเดือนซึ่งเราได้คำนวณค่าพยากรณ์ของแต่ละแบบไว้แล้วซึ่งเราจะทำการทดสอบระบบเพื่อดูว่าค่าที่ได้จะสอดคล้องหรือไม่ซึ่งข้อมูลในตารางที่ 4.1 จะเป็นข้อมูลที่มีรายจ่ายแบบกระจายตัว

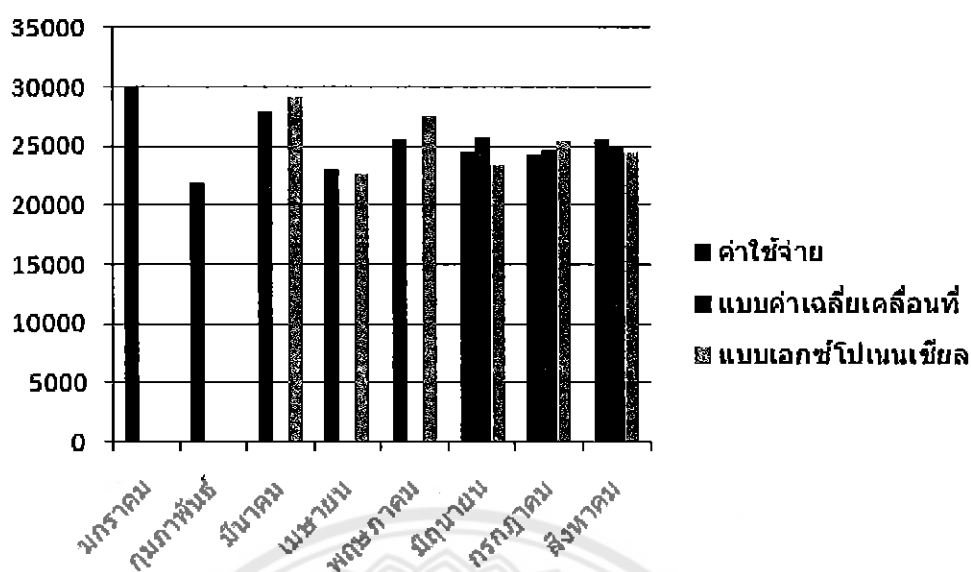
เมื่อนำข้อมูลจากตารางที่ 4.2 ป้อนเข้าไปใน โปรแกรมแล้วก็ทำการทดสอบโปรแกรมโดยทำการพยากรณ์รายจ่ายเดือนที่ 7 หรือ กรกฎาคม



รูปที่ 4.33 การใช้เมนูพยากรณ์รายจ่าย
ด้วยวิธี Moving Averaging (2)



รูปที่ 4.34 การใช้เมนูพยากรณ์รายจ่าย
ด้วยวิธี Exponential (2)

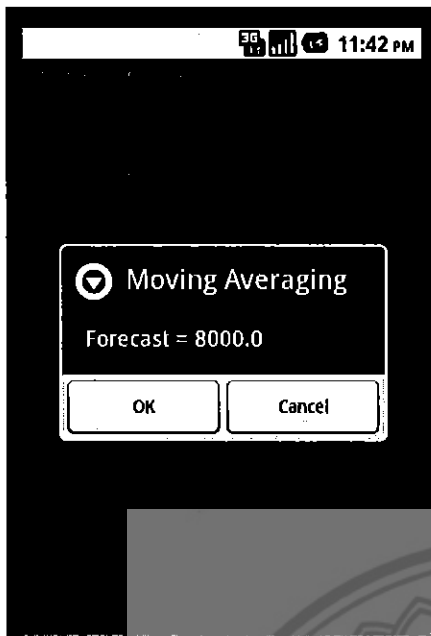


รูปที่ 4.35 กราฟแสดงการเปรียบเทียบระหว่างค่าใช้จ่ายจริงกับการพยากรณ์ค่าใช้จ่าย (2)
จากรูปที่ 4.33 และรูปที่ 4.34 จะเห็นได้ว่า ข้อมูลที่ได้สอดคล้องกับตารางที่ 4.2 และเมื่อดู
ข้อมูลในรูปที่ 4.35 แล้วจะเห็นได้ว่าค่าที่ได้จากการพยากรณ์ทั้ง 2 แบบมีค่าใกล้เคียงกับค่าจริง

ตารางที่ 4.3 จำลองค่าใช้จ่ายในแต่ละเดือน (3)

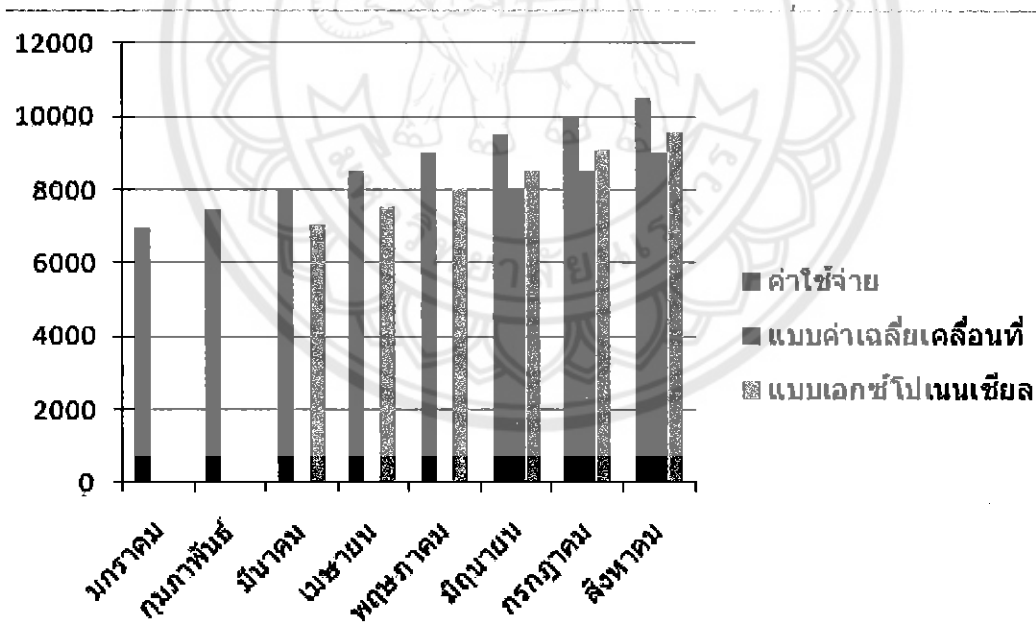
เดือน	ค่าใช้จ่าย	การพยากรณ์รายจ่าย	
		แบบค่าเฉลี่ยเคลื่อนที่	แบบเอกซ์โปเนนเชียล
มกราคม	7,000	22,000	30,000
กุมภาพันธ์	7,500	23,000	28,000
มีนาคม	8,000	24,000	29,000
เมษายน	8,500	25,000	27,000
พฤษภาคม	9,000	26,000	28,000
มิถุนายน	9,500	27,000	25,000
กรกฎาคม	10,000	28,000	26,000
สิงหาคม	10,500	29,000	25,000

จากตารางที่ 4.3 เป็นจำลองค่าใช้จ่ายในแต่ละเดือนซึ่งเราได้คำนวณค่าพยากรณ์ของแต่ละแบบไว้แล้วซึ่งเราจะทำการทดสอบระบบเพื่อดูว่าค่าที่ได้จะสอดคล้องหรือไม่ซึ่งข้อมูลในตารางที่ 4.3 จะเป็นข้อมูลที่มีรายจ่ายแบบมีแนวโน้มกล่าวคือรายจ่ายจะค่อยๆเพิ่มขึ้นในแต่ละเดือน



รูปที่ 4.36 การใช้เมนูพยากรณ์รายจ่าย ด้วยวิธี Moving Averaging (3)

รูปที่ 4.37 การใช้เมนูพยากรณ์รายจ่าย ด้วยวิธี Exponential (3)



รูปที่ 4.38 กราฟแสดงการเปรียบเทียบระหว่างค่าใช้จ่ายจริงกับการพยากรณ์ค่าใช้จ่าย (3)

จากรูปที่ 4.36 และรูปที่ 4.37 จะเห็นได้ว่า ข้อมูลที่ได้สอดคล้องกับตารางที่ 4.3 แต่เมื่อดูข้อมูลในรูปที่ 4.38 แล้วจะเห็นได้ว่าค่าที่ได้จากการพยากรณ์ทั้ง 2 แบบมีค่าไม่ใกล้เคียงค่าจริงและไม่เป็นไปตามแนวโน้มที่กราฟควรจะเป็น

จากการทดสอบเมนูการพยากรณ์รายจ่ายทั้ง 2 แบบนั้นจะเห็นได้ว่าเมื่อทำการทดสอบเมนูจะใช้ข้อมูลก่อนหน้าเดือนที่จะพยากรณ์เป็นตัวชี้การพยากรณ์ ซึ่งวิธี Moving Averaging จะใช้ข้อมูลย้อนหลัง 5 เดือน และวิธี Exponential จะใช้ข้อมูลย้อนหลัง 2 เดือน โดยที่ทั้ง 2 วิธีนี้ จะมีความเหมาะสมกับข้อมูลที่มีค่ากระจายรอบค่าเฉลี่ย โดยที่ไม่มีแนวโน้มหรือฤดูกาลเข้ามาเกี่ยวข้อง และเป็นการพยากรณ์ระยะสั้น

ในบทนี้เราจะพูดถึงการนำข้อมูลเข้ามาในแอปพลิเคชันและบันทึกลงบนฐานข้อมูลเพื่อรอที่จะนำข้อมูลนั้นไปประยุกต์ใช้ในลักษณะต่างๆ เช่น การเรียกดูข้อมูลย้อนหลังในรูปแบบสถิติ กราฟวงกลม และกราฟแท่ง ข้อมูลจะสามารถบันทึกเก็บไว้เพื่อดูย้อนหลังได้ อีกทั้งยังทำการลบข้อมูลที่ไม่ต้องการ หรือข้อความที่มีความผิดพลาดได้ ซึ่งเปรียบเสมือนการใช้งานจริง อีกทั้งโปรแกรมจะสามารถพยากรณ์รายจ่ายได้ 2 วิธีคือ วิธี Moving Averaging ซึ่งวิธีนี้จะต้องมีข้อมูลของรายจ่ายย้อนหลังจากเดือนที่ต้องการพยากรณ์ 5 เดือน และวิธี Exponential จะต้องมีข้อมูลของรายจ่ายย้อนหลังจากเดือนที่ต้องการพยากรณ์ 2 เดือน และการพยากรณ์รายจ่ายทั้ง 2 วิธีนี้จะเหมาะสมสำหรับข้อมูลที่มีค่ากระจายรอบค่าเฉลี่ย โดยที่ไม่มีแนวโน้ม และการพยากรณ์นี้จะเป็นการพยากรณ์ระยะสั้น ซึ่งและในบทต่อไปจะพูดถึงการ สรุป การทำงานของ โปรแกรมว่า โปรแกรมสามารถทำตามอย่างที่เรา ออกแบบ ไว้ตอนแรกหรือไม่ ความพอใจอยู่ในระดับไหนและปัญหาที่เกิดขึ้นมีอะไรบ้าง อย่างไร

บทที่ 5

สรุปผลการทดลอง

การทำแอปพลิเคชันสำหรับการจัดการข้อมูลการเงินส่วนบุคคลเริ่มด้วยการตั้งวัตถุประสงค์ที่จะทำจากการวิเคราะห์ปัญหาที่เกิดขึ้นสำหรับการบันทึกรายรับ รายจ่าย แล้วจึงนำแอปพลิเคชันนี้เข้าไปแก้ไขปัญหาซึ่งการจัดการทำแอปพลิเคชัน จำเป็นต้องมีการกำหนดขอบเขตของแอปพลิเคชันว่าสามารถทำอะไรได้บ้าง แล้วจึงวางแผนการทำงานกำหนดขั้นตอนและระยะเวลาในการจัดทำแอปพลิเคชัน รวมถึงศึกษาเทคโนโลยีต่างๆที่ใช้จัดทำแอปพลิเคชัน

จากนั้นทำการศึกษาระบบปฏิบัติการแอนดรอยด์ว่ามีรูปแบบโครงสร้างอย่างไร และศึกษาระบบฐานข้อมูลที่จะใช้ซึ่งในที่นี้ใช้ SQL เป็นภาษาของฐานข้อมูลอีกทั้งศึกษาสมการที่ใช้ในการคำนวณ การพยากรณ์รายจ่ายซึ่งในที่นี้ใช้สมการของการหาค่าเฉลี่ยเคลื่อนที่และการพยากรณ์แบบเอกซ์โปเนนเชียล

เมื่อได้ข้อมูลจากการศึกษาเกี่ยวกับเทคโนโลยีที่ใช้จัดทำแอปพลิเคชันแล้วจึงทำการออกแบบโครงสร้างของแอปพลิเคชันซึ่งประกอบด้วย เมนูรายรับ เมนูรายจ่าย เมนูเรียกดูข้อมูล เมนูลบข้อมูล เมนูเรียกดูกราฟแท่งรายปี และ เมนูพยากรณ์รายจ่าย จากนั้นจึงออกแบบฐานข้อมูลที่ใช้ในแอปพลิเคชัน และจำลองการใช้จ่ายเพื่อทดสอบแอปพลิเคชัน

การทดสอบแอปพลิเคชันจะจำลองการใช้จ่ายขึ้นมาเพื่อทดสอบระบบต่างๆในแอปพลิเคชัน โดยเริ่มจากการเพิ่มและลบข้อมูลรายรับ-รายจ่ายกับระบบแล้วเรียกดูข้อมูลทั้งในรูปแบบสถิติและกราฟวงกลม จากนั้นทำการเรียกดูข้อมูลรายรับ-รายจ่ายในรูปแบบกราฟแท่งแบบรายเดือนใน 1 ปี และพยากรณ์รายจ่ายในเดือนต่อไปจากข้อมูลรายจ่ายเดิม

5.1 สรุปผลโครงการ

จากการพัฒนาระบบรายรับ รายจ่าย บนระบบแอนครอยด์ เพื่อทำบัญชีเรียกดูข้อมูลแต่ละเดือน ปีได้ และสามารถดูข้อมูลผ่านกราฟต่างๆ ได้ โดยระบบสามารถเพิ่มลบ ข้อมูลเพื่อมาคำนวณ ในบัญชีรายรับ รายจ่ายได้ และยังสามารถนำออกมาเป็นไฟล์เอกสารได้อีกด้วย และสามารถเปิดผ่านโปรแกรมเอกสารอื่นๆ ได้ ซึ่งระบบโปรแกรมสามารถสรุปผลการทดลองได้ ดังนี้

- 1 โปรแกรมสามารถบันทึกข้อมูลรายรับ-รายจ่ายได้
- 2 โปรแกรมสามารถเรียกดูรายละเอียดของข้อมูลรายรับ-รายจ่าย ได้ ซึ่งประกอบด้วย
 - วัน เดือน ปี ที่มีรายรับและรายจ่าย
 - ประเภทของรายจ่าย
 - ราคาและรายละเอียดของรายจ่าย
 - ยอดเงินคงเหลือ
- 3 โปรแกรมสามารถแสดงสถิติข้อมูลรายรับ-รายจ่ายรูปแท่งกราฟวงกลมและกราฟแท่งได้
- 4 โปรแกรมสามารถส่งออก (Export) ข้อมูลเป็นไฟล์ .txt ได้
- 5 โปรแกรมสามารถพยากรณ์ค่าใช้จ่ายในอนาคตได้ โดยใช้ฐานข้อมูลเก่าเป็นตัวบ่งชี้

5.2 ปัญหาที่พบ

ตารางที่ 5.1 ตารางแสดงปัญหาที่พบและแนวทางการแก้ไขปัญหา

ปัญหาที่พบ	แนวทางการแก้ไขปัญหา
1. แอปพลิเคชันที่ทำออกมาในรูปแบบจะคลาดเคลื่อนจากที่เราได้ออกแบบไว้	-
2. เนื่องจากในช่วงแรกของการพัฒนาระบบ ยังไม่ค่อยเข้าใจในส่วนการพัฒนาโปรแกรมด้วยภาษาจาวาจึงทำให้ไม่เป็นไปตามแผนที่วางไว้	ทำการศึกษาภาษาจาวาให้มากขึ้น โดยการอ่านหนังสือ ปรึกษาอาจารย์ เพื่อนและผู้เชี่ยวชาญตามเว็บไซต์ต่างๆ
3. เวลาของการทำโครงการไม่เป็นไปตามเป้าหมาย	ใช้เวลาทำโครงการให้มากขึ้น ลดเวลาของกิจกรรมอย่างอื่นลงและกำหนดการวางแผนใหม่เพื่อให้ได้งานตรงตามเวลา
4. การพัฒนาระบบไม่เป็นไปตามที่วางแผน เนื่องจากประเมิงานในการพัฒนาระบบต่ำไป	ศึกษาเกี่ยวกับการพัฒนาระบบให้มากขึ้นและปรึกษาผู้เชี่ยวชาญ ว่าระบบโครงการนี้ ควรจะทำอย่างไร

5.3 แนวทางในการพัฒนาเพิ่มเติมและข้อเสนอแนะ

5.3.1 แนวทางแก้ไขปัญหาเพื่อพัฒนาเพิ่มเติม

1. ระบบรายรับ-รายจ่ายสามารถแสดงข้อมูลในรูปแบบสถิติได้ในรูปแบบเดียว
2. โปรแกรมเก็บบันทึกข้อมูลที่ไคลงในฐานข้อมูลของเครื่องซึ่งมีความเสี่ยงต่อการสูญหาย
3. โปรแกรมสามารถดูข้อมูลในรูปแบบกราฟได้เพียง 2 กราฟคือกราฟแท่งและกราฟวงกลม
4. โปรแกรมจะสามารถรับค่าได้โดยทางการพิมพ์เพียงช่องทางเดียว
5. โปรแกรมสามารถพยากรณ์ได้โดยวิธีการหาค่าเฉลี่ยเคลื่อนที่และการพยากรณ์แบบเอกซ์

โปเนนเชียล แต่การพยากรณ์ด้วย 2 วิธีข้างต้นมีข้อจำกัดเรื่องรูปแบบของข้อมูลที่มีและเป็นการพยากรณ์ระยะสั้น

5.3.2 ข้อเสนอแนะและคุณสมบัติของซอฟต์แวร์ในอนาคต

1. โปรแกรมควรบันทึกข้อมูลและแสดงข้อมูลในรูปแบบตารางได้
2. โปรแกรมควรบันทึกข้อมูลที่ไคลงบนอินเทอร์เน็ตเพื่อป้องกันการสูญหายของข้อมูล
3. โปรแกรมควรสามารถดูกราฟได้หลายชนิดมากขึ้นเพื่อเปรียบเทียบจุดได้หลายแบบตามลักษณะของกราฟนั้น
4. โปรแกรมควรรับค่าได้หลายช่องทาง เช่น การรับค่าทางเสียง และการพิมพ์
5. โปรแกรมควรมีการพยากรณ์ที่เจาะลึกและแม่นยำมากยิ่งขึ้น และสามารถรองรับกับข้อมูลได้ทุกรูปแบบ

5.3.3 ทักษะและความรู้ที่สำคัญต่อการต่อยอดโครงการ

ซึ่งการพัฒนาแอปพลิเคชันนี้ต่อไปจำเป็นต้องศึกษาระบบปฏิบัติการแอนดรอยด์ว่ามีรูปแบบโครงสร้างอย่างไร และศึกษาระบบฐานข้อมูล SQL ซึ่งเป็นภาษาของฐานข้อมูลของแอปพลิเคชันนี้ และ ศึกษาการเขียน โปรแกรม โดยภาษา java อีกทั้งศึกษาสมการที่ใช้ในการคำนวณการพยากรณ์รายจ่ายซึ่งในที่นี้ใช้สมการของการหาค่าเฉลี่ยเคลื่อนที่และการพยากรณ์แบบเอกซ์โปเนนเชียล

การเขียนโปรแกรมบนระบบแอนดรอยด์ เริ่มต้นจากการเขียนโปรแกรมภาษาจาวา ศึกษาการใช้ฐานข้อมูล SQLite ว่าต้องมีคำสั่งอย่างไรในการสร้าง ลบ เรียกดูข้อมูล ตั้งแต่เริ่มต้นตลอดไปจนถึงการหัดคิดบ้าง เพื่อดูว่าโปรแกรมผิดพลาดตรงไหนในลักษณะต่างๆ เพื่อให้เกิดทักษะและประสบการณ์ในการเขียน ดังนั้นการที่จะพัฒนาระบบออกมาได้สำเร็จนั้นต้องมีความรู้ในหลายๆ ด้าน ทั้งด้านการเขียนโปรแกรม การศึกษาความรู้เฉพาะของโปรแกรมที่จะพัฒนาและที่สำคัญที่สุดคือการวางแผนมีส่วนสำคัญอย่างมาก เพื่อที่จะได้รู้ว่าควรทำขั้นตอนใดก่อนและขั้นตอนต่อไปควรจะทำอย่างไร เมื่อวางแผน ได้ดีและละเอียด ชัดเจน การทำงานก็จะง่ายขึ้น และโอกาส ประสบผลสำเร็จก็มากขึ้นด้วย



เอกสารอ้างอิง

- [1] (30 ตุลาคม 2554). ประวัติแอนดรอยด์. สืบค้นเมื่อ 14 มกราคม 2555, จาก <http://www.gotoknow.org/blogs/posts/466674>
- [2] may[นามแฝง]. (3 มกราคม 2555). คุณสมบัติแอนดรอยด์. สืบค้นเมื่อ 14 มกราคม 2555, จาก <http://bal.buu.ac.th/bal2010/node/172>
- [3] (29 มิถุนายน 2554). การทำงานระบบแอนดรอยด์. สืบค้นเมื่อ 15 มกราคม 2555 จาก <http://www.androidjump.com/วงจรชีวิต-android-activity-life-cycle/>
- [4] (2 กุมภาพันธ์ 2554). โครงสร้างทางซอฟต์แวร์แอนดรอยด์. สืบค้นเมื่อ 15 มกราคม 2555 จาก <http://www.phet.in.th/2011/01/Android-story-3-inside-android/>
- [5] Ibenz[นามแฝง]. (24 สิงหาคม 2554). การสร้างฐานข้อมูล. สืบค้นเมื่อ 15 มกราคม 2555 จาก <http://www.bananacode.net/archives/627>
- [6] (14 มิถุนายน 2553). ตัวอย่างภาษา Sql เว็บไซต์. สืบค้นเมื่อ 29 มกราคม 2555 จาก <http://www.choosak.com/page-29/>
- [7] (จารุณี พิทยานิวัฏฐ์). (18 ธันวาคม 2551). การปรับให้เรียบแบบเอ็กซ์โปเนนเชียล. สืบค้นเมื่อ 13 มีนาคม 2555 จาก http://www.course.ku.ac.th/lms/files/resources_files/32593/99105
- [8] jaboho[นามแฝง]. (30 กุมภาพันธ์ 2554). การลง Android เว็บไซต์. สืบค้นเมื่อ 15 มกราคม 2555 จาก <http://www.codemobiles.com/forum/viewtopic.php?t=904>
- [9] ไพบูลย์ สวัสดิ์ปัญญาโชติ. (2554). The Android Developer's Cookbook. **รวมโค้ด AndroidApp**
- [10] Reto Meier. (2551). Android Application Development. **Inc Indianapolis Indiana**
- [11] Reto Meier. (2552). Android 2 Application Development. **Inc Indianapolis Indiana**
- [12] Mark L. Murphy. (2552). Beginning Android 2. **Springer-Verlag New York**

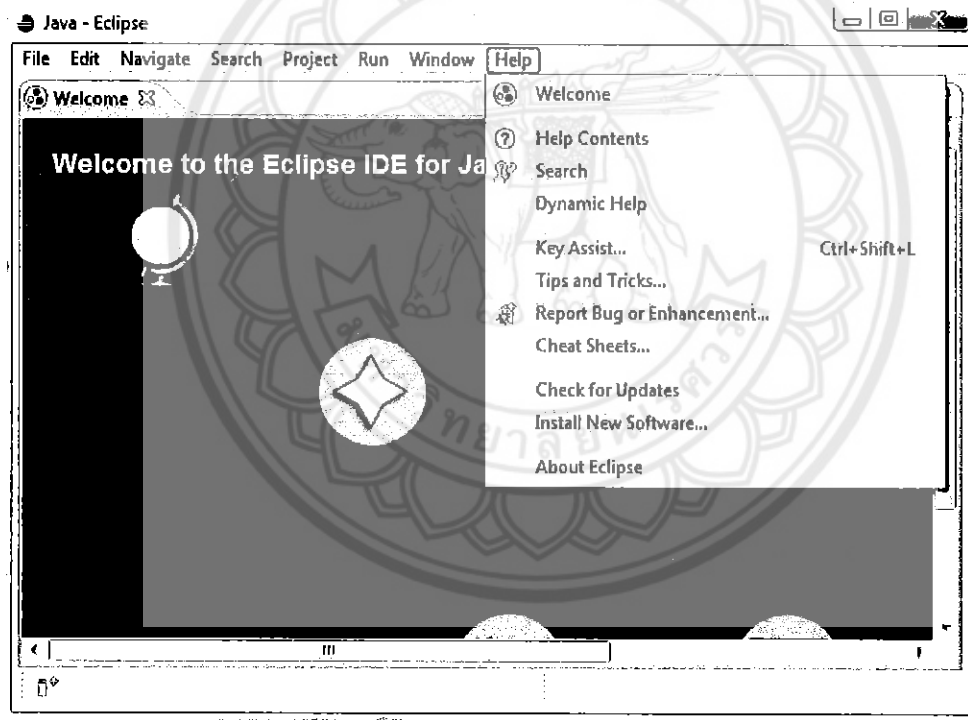
ภาคผนวก

การตั้งค่าเครื่องมือที่ใช้ในการพัฒนาโปรแกรม [8]

ก. การตั้งค่าเครื่องมือที่ใช้ในการพัฒนาโปรแกรม

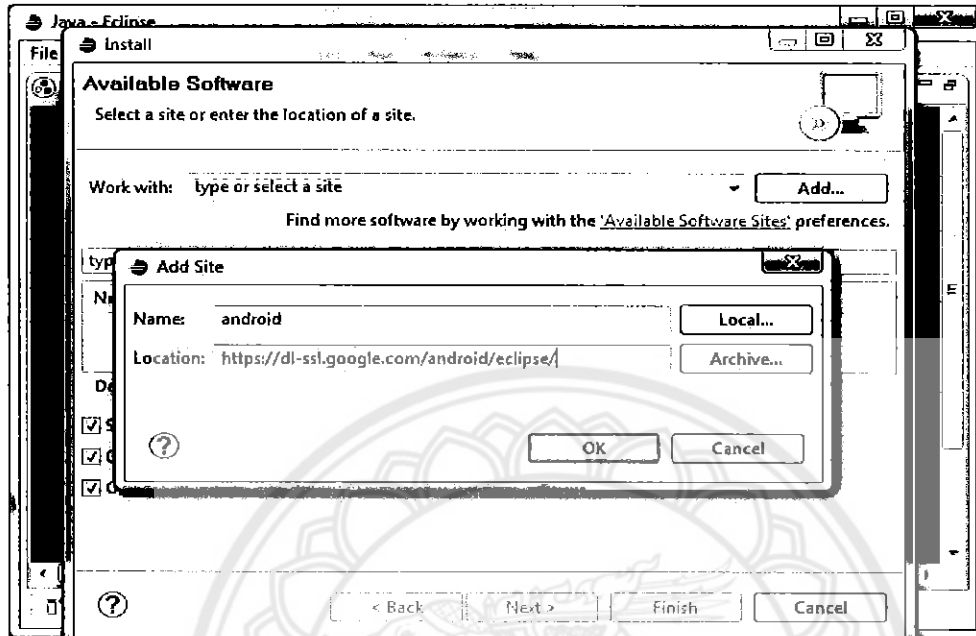
ตั้งค่าเครื่องมือที่ใช้ในการพัฒนาโปรแกรมเป็นการตั้งค่า Eclipse เพื่อใช้ในการพัฒนาโปรแกรมรายรับ
รายจ่าย บนระบบแอนดรอยด์

1. ดาวน์โหลดโปรแกรม Eclipse และ Android SDK
2. แยกไฟล์ โปรแกรม Eclipse และเปิดขึ้นมาแล้วเลือกไปที่เมนู Install New Software



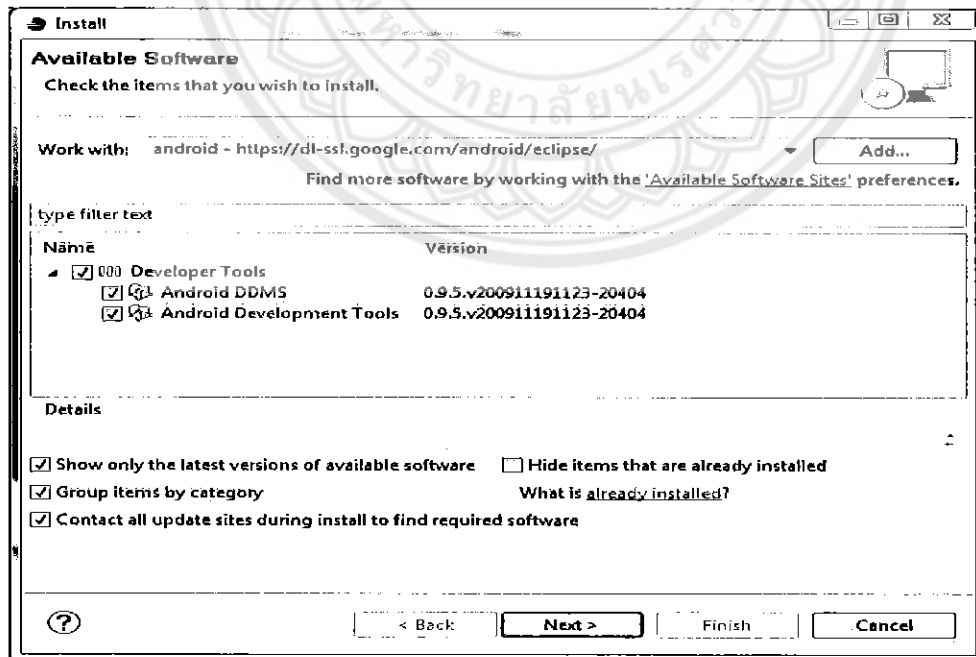
รูปภาคผนวกที่ 1 การติดตั้ง Eclipse 1

3. ให้คอมพิวเตอร์ Add ตามรูป เพื่อทำการติดตั้ง site ที่ดาวน์โหลด ADT plug-in



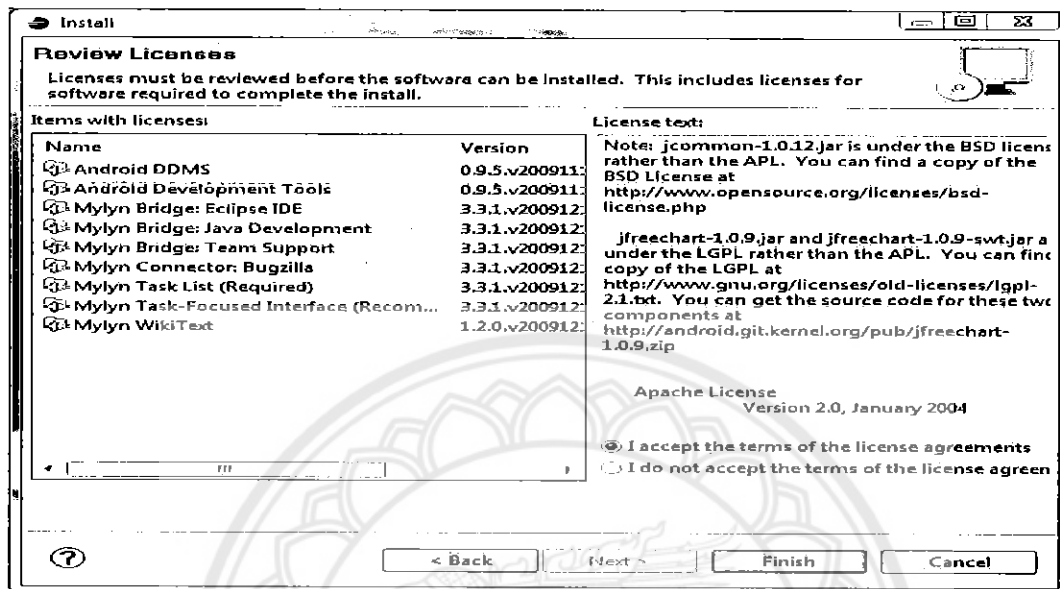
รูปภาคผนวกที่ 2 การติดตั้ง Eclipse 2

4. หลังจากนั้นกด ok และกดเลือกตามรูป



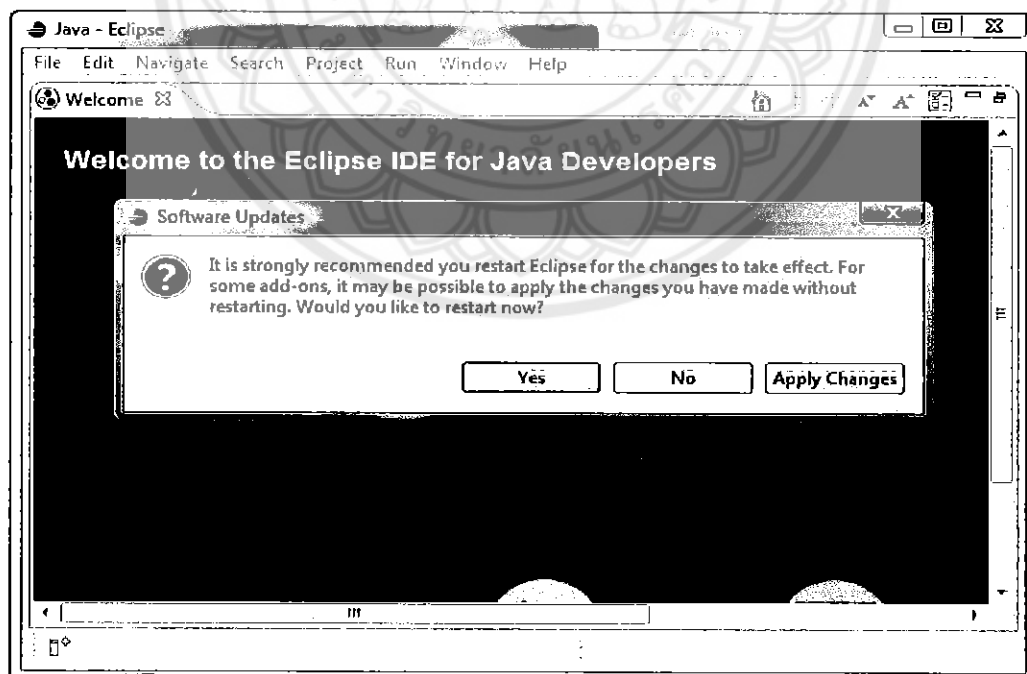
รูปภาคผนวกที่ 3 การติดตั้ง Eclipse 3

5. ให้เลือก I accept the term of the license agreements ตามรูปข้างล่าง แล้วกด Finish



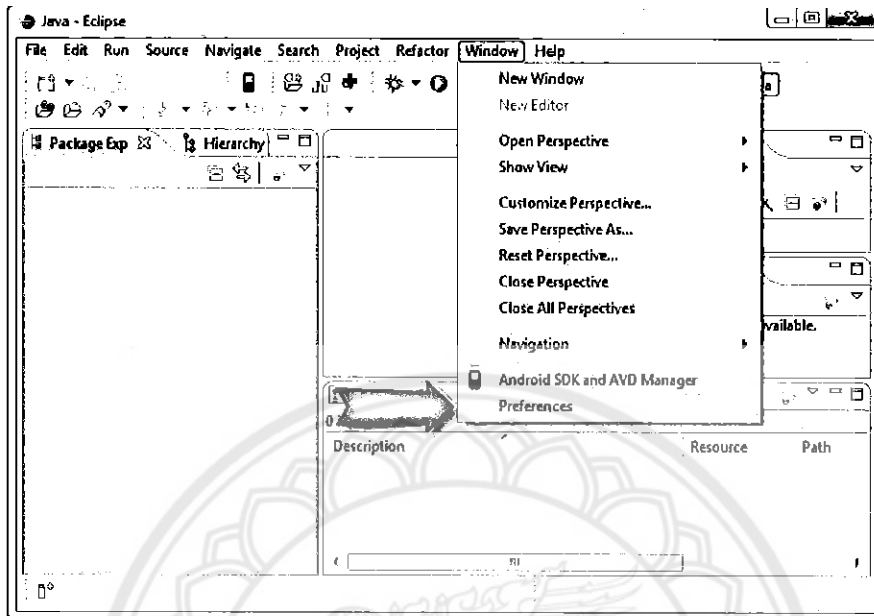
รูปภาคผนวกที่ 4 การติดตั้ง Eclipse 4

6. หลัง install เสร็จให้เรา รีบูต Eclipse ให้กด Yes เพื่อ รีบูต เป็นอันเสร็จขั้นตอน การติดตั้ง ADT



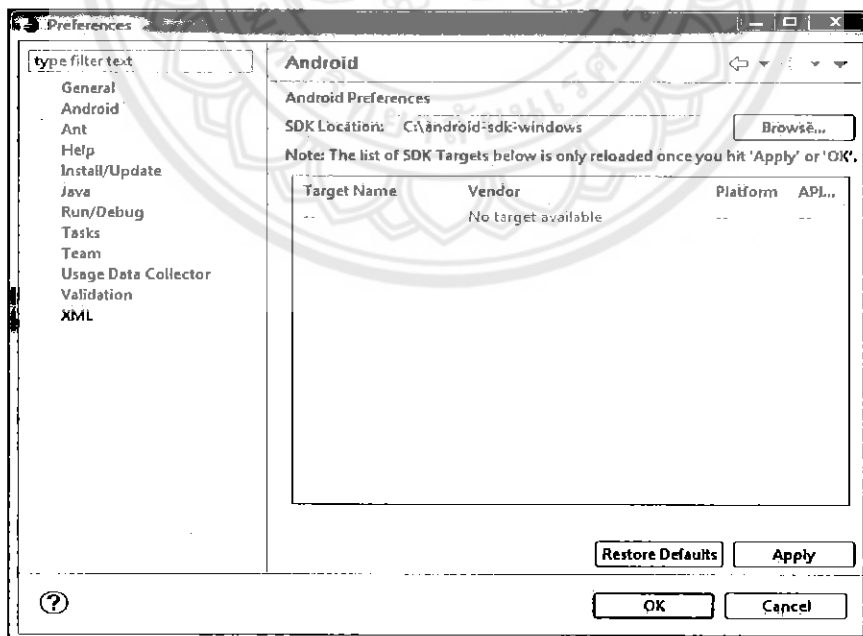
รูปภาคผนวกที่ 5 การติดตั้ง Eclipse 5

7. ไปที่ เมนู Window->Preference



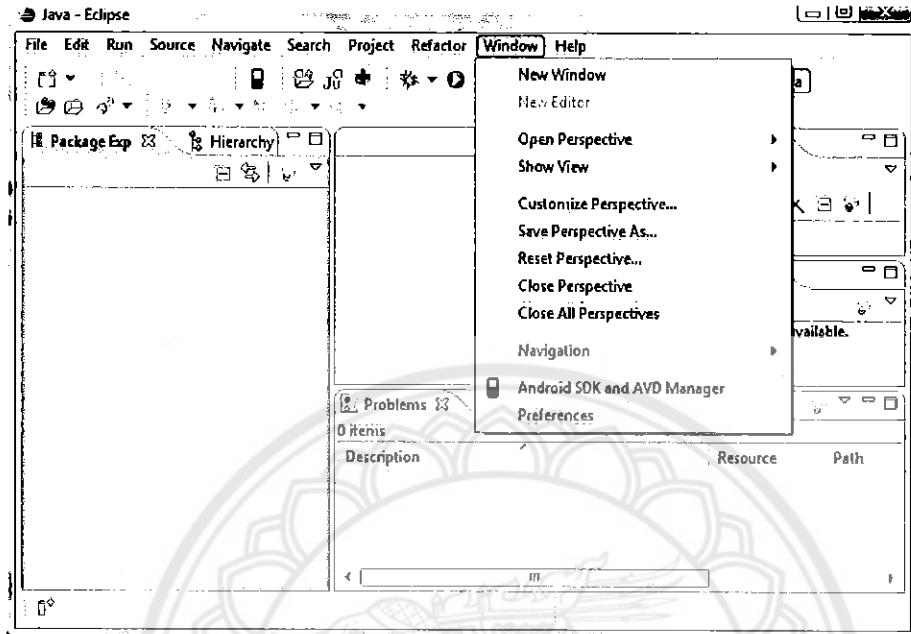
รูปภาพหมวดที่ 6 การติดตั้ง Eclipse 6

8. กดปุ่ม Browse เพื่อระบุ directory path ที่เรา unzip android sdk ก่อนหน้านี้ ในรูปนี้คือ C:\android-sdk-windows หลังจากนั้นกด ok



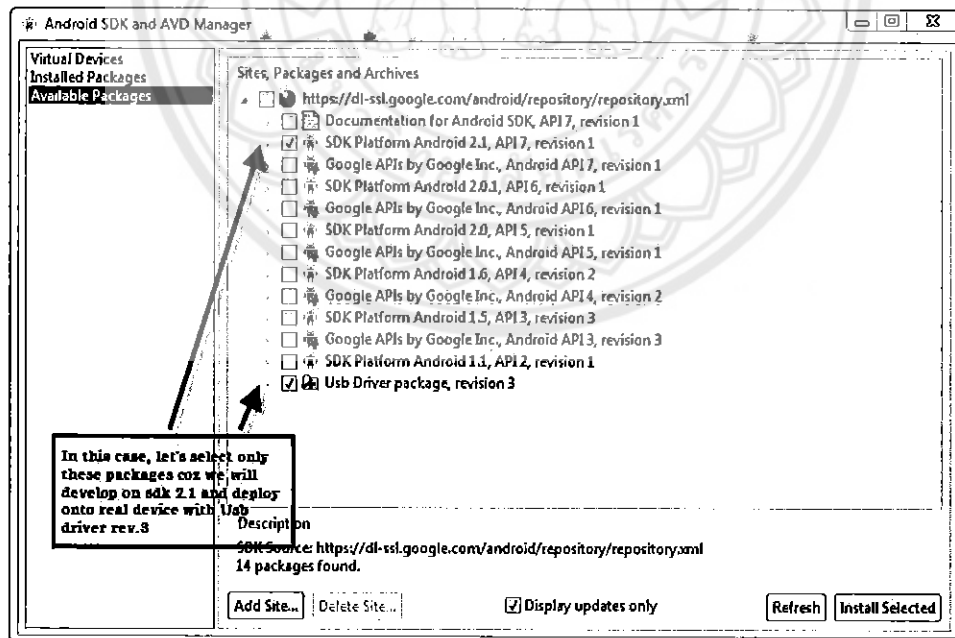
รูปภาพหมวดที่ 7 การติดตั้ง Eclipse 7

9. ไปที่ เมนู Window และ เลือก Android SDK and AVD Manager



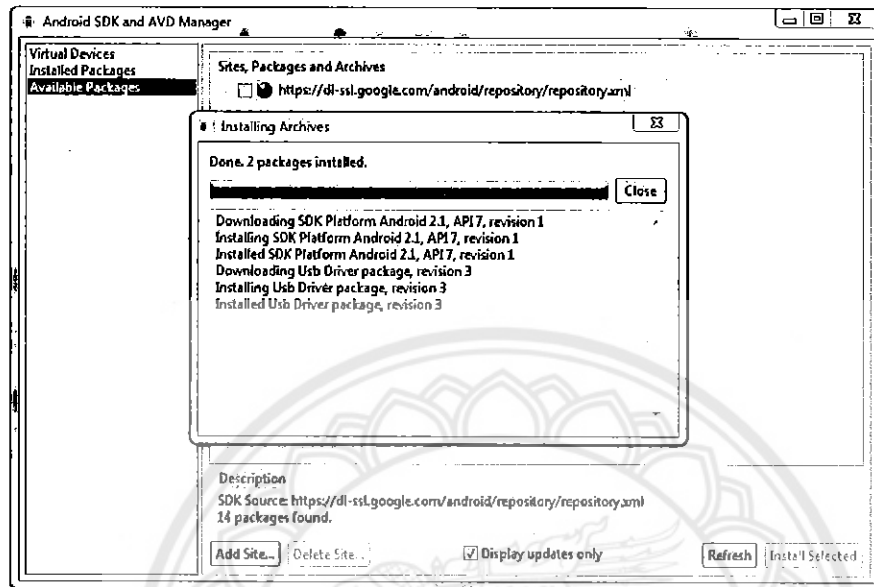
รูปภาคผนวกที่ 8 การติดตั้ง Eclipse 8

10. เลือก Available Packages และเลือก Android SDK Package ที่จะลง



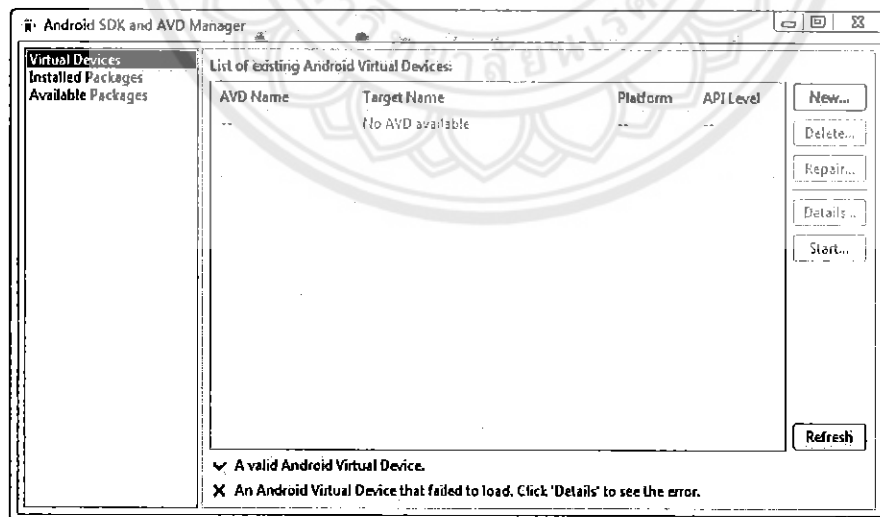
รูปภาคผนวกที่ 9 การติดตั้ง Eclipse 9

11. หลังจากเสร็จจะขึ้นหน้าต่างดังรูป



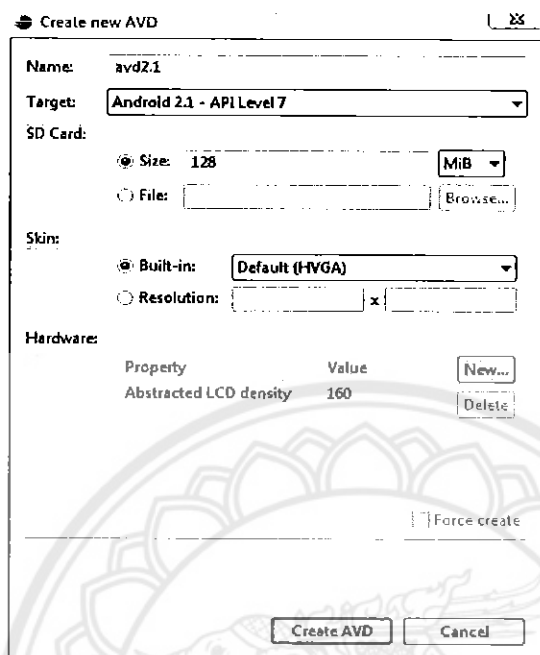
รูปภาพผนวกที่ 10 การติดตั้ง Eclipse 10

12. ไปที่เมนู Window และ เลือก Android SDK and AVD Manager ที่ Virtual Device ให้กดปุ่ม New เพื่อสร้าง Android Emulator Profile หรือ ที่เรียก AVD



รูปภาพผนวกที่ 11 การติดตั้ง Eclipse 11

13. เลือกคุณสมบัติของ Android Emulator ตามที่พัฒนา



รูปภาพหมวดที่ 12 การติดตั้ง Eclipse 12

14. หลังจากนั้นกดเรียกใช้งาน จะได้หน้าต่างดังในรูป



รูปภาพหมวดที่ 13 การติดตั้ง Eclipse 13

ประวัติผู้ดำเนินโครงการ



ชื่อ นายฐากร สีนะกุล
 ภูมิลำเนา 337 หมู่ 8 ต.บัว อ.บัว จ.น่าน 55120
 ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนสตรีนครสวรรค์
- ปัจจุบันกำลังศึกษาอยู่ในระดับปริญญาตรีชั้นปีที่ 5 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: isawaboy@hotmail.com



ชื่อ นายอธิการ กอบกอง
 ภูมิลำเนา 73/1 หมู่ที่ 7 ต.จอมจันทร์ อ.เวียงสา จ.น่าน 55110
 ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนสตรีศรีน่าน
- ปัจจุบันกำลังศึกษาอยู่ในระดับปริญญาตรีชั้นปีที่ 5 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: little_cupid_g@hotmail.com