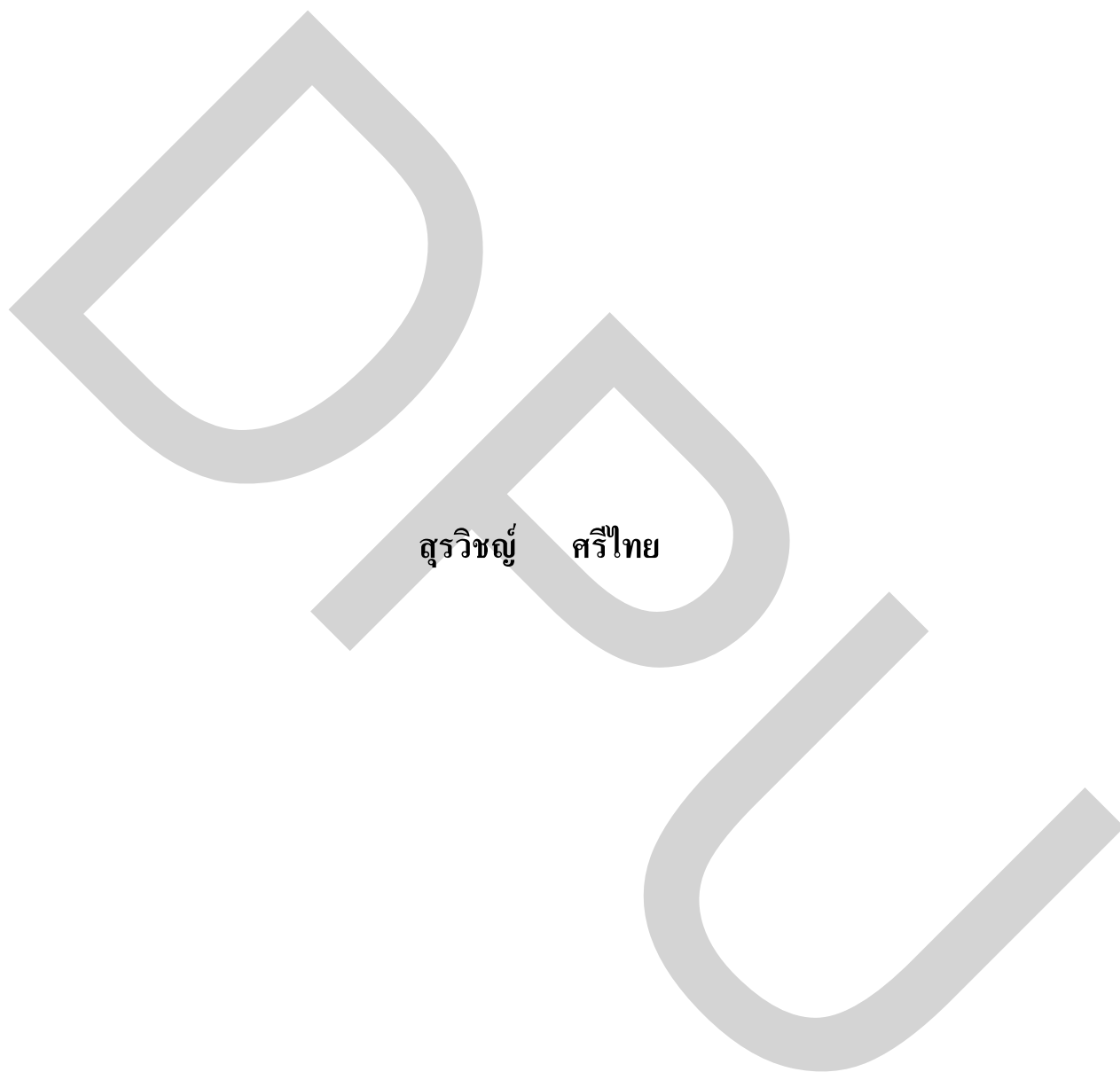


โปรแกรมกรองข้อความสั้นบนโทรศัพท์เคลื่อนที่แอนดรอยด์
โดยอาศัยข้อมูลในสมุดโทรศัพท์



สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์และโทรคมนาคม คณะวิศวกรรมศาสตร์
มหาวิทยาลัยธุรกิจบัณฑิตย์

พ.ศ. 2555

**Software Filtering Short Message Service on Android Mobile Phone
based on Information in Phone book**



SURAWIT SRITHAI

**A Thematic Paper Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering
Department of Computer and Telecommunication Engineering
Faculty of Engineering, Dhurakij Pundit University**

2012

กิตติกรรมประกาศ

สารนิพนธ์นี้สำเร็จลุล่วงไปได้ด้วยดี โดยได้รับความกรุณาเป็นอย่างยิ่งจาก ท่านอาจารย์ ดร.ชัยพร เขมะภาคะพันธ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่คอยให้คำแนะนำ ตลอดจนในห้องักความรู้ ในการค้นคว้าข้อมูลให้แก่ผู้วิจัย ขอขอบพระคุณ ท่านอาจารย์ ดร.ประศาสน์ จันทราทิพย์ และท่าน อาจารย์ ดร.เจนจบ วีระพานิชเจริญกิจ กรรมการสอบสารนิพนธ์ ซึ่งสละเวลามาเป็นกรรมการสอบ สารนิพนธ์ และได้ให้ข้อคิดเห็นที่เป็นประโยชน์ต่องานวิจัย รวมทั้งให้ความรู้อันเป็นประโยชน์แก่ ผู้วิจัยมาโดยตลอด นอกจากนี้ ผู้วิจัยขอขอบพระคุณคณาจารย์ทุก ๆ ท่านในภาควิชา วิศวกรรมศาสตร์ ที่ได้ถ่ายทอดความรู้แก่ผู้วิจัยตลอดระยะเวลาการศึกษา

ผู้วิจัยขอขอบพระคุณ เจ้าหน้าที่ที่เกี่ยวข้องทุกท่าน ในภาควิชาวิศวกรรมศาสตร์ ที่คอย ให้ความช่วยเหลือ ตลอดจนแนะนำกระบวนการในการทำงานให้แก่ผู้วิจัยด้วยดีเสมอมา

ผู้วิจัยขอขอบพระคุณ คุณศิริชัย เอ่งฉ้วน ผู้จัดการส่วนผลิตภัณฑ์มัลติมีเดีย บริษัท กสท โทรคมนาคม จำกัด (มหาชน) ที่ให้คำแนะนำ ช่วยเหลือและส่งเสริมในการทำวิจัยครั้งนี้จนสำเร็จ ลุล่วงได้ด้วยดี

ผู้วิจัยขอขอบคุณเพื่อน ๆ ร่วมสาขาวิชาทุกคนที่ช่วยเหลือและให้กำลังใจกันเสมอมา ตลอดระยะเวลาการศึกษา

ท้ายสุดนี้ ผู้วิจัยขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ที่ได้เลี้ยงดูและครอบครัวที่คอย เป็นกำลังใจและให้การสนับสนุนผู้วิจัยในทุก ๆ ด้านเสมอมาจนสำเร็จการศึกษา

สุรวิษณุ ศรีไทย

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ฉ
บทคัดย่อภาษาอังกฤษ.....	ง
กิตติกรรมประกาศ.....	จ
สารบัญตาราง.....	ฉ
สารบัญภาพ.....	ฉ
บทที่	
1. บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	3
1.3 สมมติฐานของการวิจัย.....	3
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	3
2. แนวคิด ทฤษฎี และผลงานวิจัยที่เกี่ยวข้อง.....	4
2.1 แนวคิดและทฤษฎีที่เกี่ยวข้อง.....	4
2.2 ประวัติระบบปฏิบัติการแอนดรอยด์.....	4
2.3 สถาปัตยกรรมของแอนดรอยด์.....	6
2.4 Short Message Service (SMS).....	10
2.5 นิยามของ Spam.....	11
2.6 แอปพลิเคชันที่เกี่ยวข้อง.....	14
3. ระเบียบวิธีวิจัย.....	18
3.1 แนวทางการวิจัยและพัฒนา.....	18
3.2 เครื่องมือที่ใช้ในงานวิจัย.....	18
3.3 แผนการดำเนินงาน.....	19
3.4 ขั้นตอนการดำเนินงานวิจัย.....	20
4. โปรแกรมกรองข้อความสแปมบนโทรศัพท์เคลื่อนที่แอนดรอยด์โดยอาศัยข้อมูลใน สมุดโทรศัพท์.....	27
4.1 ภาพรวมการทำงานของระบบ.....	27

สารบัญ (ต่อ)

บทที่	หน้า
4.2 ข้อกำหนดของโปรแกรมกรองข้อความสั้น.....	27
4.3 ขั้นตอนการทำงานของโปรแกรมกรองข้อความสั้น.....	28
5. ผลการวิจัย.....	41
5.1 การติดตั้งแอปพลิเคชัน.....	41
5.2 การกรองข้อความสั้น.....	43
5.3 การจัดการแอคทีวิตีบนแอปพลิเคชัน.....	45
5.4 เปรียบเทียบแอปพลิเคชัน.....	52
5.5 ผลการทดลอง.....	53
6. สรุปผลการศึกษาและวิจัย.....	54
6.1 สรุปผลการศึกษาและวิจัย.....	54
6.2 ข้อเสนอแนะและงานวิจัยในอนาคต.....	54
บรรณานุกรม.....	55
ภาคผนวก.....	59
ก ชุดรายชื่อและเลขหมายตัวอย่างบนสมุดโทรศัพท์.....	60
ข ผลการกรองข้อความสั้นที่ไม่มีรายชื่ออยู่ในสมุดโทรศัพท์.....	63
ประวัติผู้เขียน.....	66

สารบัญตาราง

ตารางที่	หน้า
2.1 แอปพลิเคชันของ Anti SMS Bomber	14
2.2 แอปพลิเคชันของ Blacklist	15
2.3 แอปพลิเคชันของ Blacklist Free	15
2.4 แอปพลิเคชันของ Anti Spam SMS	16
2.5 แอปพลิเคชันของ SMS Filter	16
2.6 แอปพลิเคชันของ Anti SMS Spam & Text Filter	17
3.1 แผนการดำเนินงาน.....	20
3.2 การอนุญาตให้เข้าถึงข้อมูลของข้อความสั้น (SMS).....	21
4.1 การอนุญาตให้เข้าถึงข้อมูลของข้อความสั้น (SMS).....	28
5.1 เปรียบเทียบแอปพลิเคชันในตลาด.....	52
5.2 ชุดข้อมูลทดสอบที่ได้บันทึกลงบนสมุดโทรศัพท์.....	53
5.3 ชุดข้อมูลที่ไม่ได้บันทึกบนสมุดโทรศัพท์.....	53

สารบัญภาพ

ภาพที่	หน้า
1.1 จำนวนส่วนแบ่งทางการตลาดของแต่ละระบบปฏิบัติการ.....	2
2.1 หน้าเพจของเว็บไซต์ Android Developer	5
2.2 สถาปัตยกรรมของระบบแอนดรอยด์.....	6
2.3 สถาปัตยกรรมชั้นแอปพลิเคชัน.....	6
2.4 สถาปัตยกรรมชั้นแอปพลิเคชันเฟรมเวิร์ค.....	7
2.5 สถาปัตยกรรมชั้นลินุกซ์เคอร์เนล.....	9
2.6 สถาปัตยกรรมของแอนดรอยด์ทั้งระบบ.....	9
2.7 การส่ง SMS จาก mobile Phone A ไปยัง Mobile Phone B	11
3.1 โครงสร้างพื้นฐานของโปรแกรมที่ใช้พัฒนา.....	23
3.2 สถานะของ Activity	25
4.1 โครงสร้างของโปรแกรมรองรับข้อความสั้นบนโทรศัพท์เคลื่อนที่แอนดรอยด์..	27
4.2 วงจรชีวิตของ Broadcast Receiver.....	29
4.3 ฟังก์ชันการรับและตรวจสอบข้อความสั้น.....	30
4.4 ฟังก์ชันการตรวจสอบบนสมุดโทรศัพท์ของระบบ.....	32
4.5 ฟังก์ชันการลบข้อความสั้นบนกล่องข้อความของระบบ.....	33
4.6 คำสั่ง OpenOrCreateDatabase	33
4.7 การสร้างตารางฐานข้อมูลและการบันทึกข้อมูล.....	34
4.8 การเพิ่มข้อมูลลงในฐานข้อมูล.....	34
4.9 การทำงานของเมธอด onCreate().....	35
4.10 การทำงานของเมธอด onItemClick().....	36
4.11 การทำงานของเมธอด onItemLongClick().....	36
4.12 การทำงานของเมธอด onCreateContextMenu().....	36
4.13 การทำงานของเมธอด onContextItemSelected().....	37
4.14 การทำงานของเมธอด onActivityResult().....	38
4.15 ขั้นตอนการทำงานของโปรแกรมภายในเมื่อเปิดโปรแกรม.....	39
4.16 ความสัมพันธ์ระหว่างคลาส.....	40

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
5.1 ภาพแอปพลิเคชันเมื่อติดตั้งผ่านสาย USB	41
5.2 ภาพแอปพลิเคชันเมื่อติดตั้งผ่านหน่วยความจำภายนอก (SD-Card).....	42
5.3 รายละเอียดแอปพลิเคชันบนเครื่องโทรศัพท์เคลื่อนที่.....	43
5.4 Log การทำงานเมื่อมีข้อความสั้นถูกส่งเข้ามา.....	44
5.5 รายการเลขหมายเมื่อดึงออกมาจากฐานข้อมูลของแอปพลิเคชัน.....	45
5.6 Log การทำงานเมื่อมีการเปิดแอปพลิเคชัน.....	45
5.7 การอ่านข้อความสั้น.....	46
5.8 Log การทำงานเมื่อเปิดอ่านรายละเอียดข้อความ.....	46
5.9 การเพิ่มรายชื่อ.....	47
5.10 Log การทำงานเมื่อเพิ่มรายชื่อ.....	48
5.11 การลบรายชื่อ.....	49
5.12 Log การทำงานเมื่อลบรายชื่อหรือเลขหมาย.....	49
5.13 การลบข้อความสั้นย่อย.....	50
5.14 Log การทำงานเมื่อลบข้อความสั้นบนหมายเลขนั้น ๆ.....	51

หัวข้อสารนิพนธ์	โปรแกรมกรองข้อความสั้นบนโทรศัพท์เคลื่อนที่แอนดรอยด์ โดยอาศัยข้อมูลในสมุดโทรศัพท์
ผู้เขียน	ศุภวิชญ์ ศรีไทย
อาจารย์ที่ปรึกษา	อาจารย์ ดร. ชัยพร เขมะภาคะพันธ์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์และโทรคมนาคม
ปีการศึกษา	2555

บทคัดย่อ

งานศึกษานี้มีจุดประสงค์เพื่อสร้างโปรแกรมกรองข้อความสั้นบนโทรศัพท์เคลื่อนที่ เพื่อลดและป้องกันข้อความสแปมต่าง ๆ เช่น ข้อความขยะ หรือข้อความโฆษณาได้ ด้วยเพราะข้อความสั้นที่อนุญาตให้รับเข้ามาในกล่องข้อความเข้าบนตัวเครื่องนั้น จะเป็นข้อความสั้นที่มีเลขหมายหรือชื่อผู้ติดต่อที่รู้จักในสมุดโทรศัพท์เท่านั้น และเมื่อมีข้อความสั้นที่ไม่มีเลขหมายอยู่ในสมุดโทรศัพท์ โปรแกรมจะทำการกรองและเก็บไว้ในฐานข้อมูลของโปรแกรมต่อไป โดยโปรแกรมที่สร้างขึ้นทำงานบนโทรศัพท์เคลื่อนที่ระบบปฏิบัติการแอนดรอยด์ จากการทดสอบและใช้งานโปรแกรม พบว่าสามารถกรองข้อความสั้นได้อย่างสมบูรณ์ทั้งจากการจำลองการทำงานและบนโทรศัพท์เคลื่อนที่จริง

Thematic Paper Title	Software Filtering Short Message Service on Android Mobile Phone based on Information in Phone book
Author	Surawit Srithai
Thematic Paper Advisor	Chaiyaporn Khemapatapan, Ph.D.
Department	Computer and Telecommunication Engineering
Academic Year	2012

ABSTRACT

This study is to implement an SMS-filtering program run on a mobile phone. The benefit of this program is to reduce spam SMS messages such as junk message or advertisement message. SMS messages sent by the senders who are in contact list of mobile phone will be allowed and passed to be in an SMS inbox. However, SMS messages sent by others not in the contact list will be blocked and stored in the program in database. The program will operate on an Android phone. The testing results from simulation and real environment found that the program can work correctly.

บทที่ 1

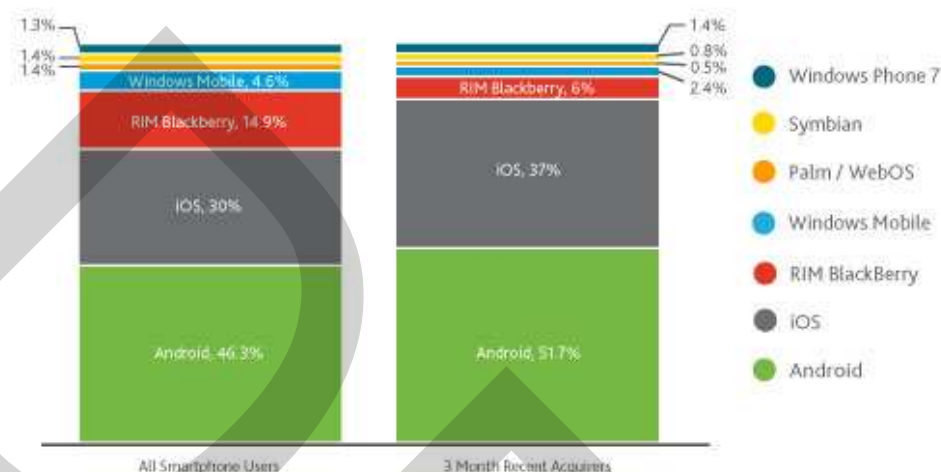
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

โลกของเทคโนโลยีในปัจจุบันนี้ ได้เจริญเติบโตก้าวหน้าอย่างรวดเร็ว โดยเฉพาะอย่างยิ่งโลกของเทคโนโลยีสื่อสารไร้สาย ซึ่งมีอยู่หลากหลายรูปแบบ หนึ่งในนั้น มีเทคโนโลยีโทรศัพท์เคลื่อนที่ รวมอยู่ด้วย หากมองย้อนไปในอดีตจะเห็นได้ว่า โทรศัพท์เคลื่อนที่ที่มีการพัฒนาอยู่ตลอดเวลา จากเดิมโทรศัพท์เคลื่อนที่ สามารถโทรเข้าและโทรออกได้อย่างเดียว จอขาวดำ หน้าจอเล็ก บันทึกข้อมูล เช่นเลขหมายโทรศัพท์ที่ได้น้อย จนกลายมาเป็นโทรศัพท์เคลื่อนที่ ที่รองรับการใช้งานได้หลากหลาย โดยนอกจากจะสามารถโทรเข้าหรือโทรออกแล้ว ยังสามารถรับส่งข้อความสั้น ข้อความมัลติมีเดีย และยังสามารถใช้งานการรับส่งข้อมูลความเร็วสูงได้บนโทรศัพท์เคลื่อนที่ เครื่องนั้น อีกทั้งยังรองรับระบบปฏิบัติการ ทำให้โทรศัพท์เคลื่อนที่ สามารถใช้งานหรือทำงาน เทียบเท่า คอมพิวเตอร์ขนาดเล็กได้ ซึ่งในปัจจุบันก็มีระบบปฏิบัติการต่าง ๆ ทำงานบนโทรศัพท์เคลื่อนที่ต่าง ๆ อาทิเช่น ระบบปฏิบัติการซิมเบียน (Symbian) วินโดว์ โมบาย (Windows Mobile) ปาล์ม โอเอส (Palm OS) ไอ โอเอส (i-OS) แบล็คเบอรี่ (BlackBerry) บาด้า (Bada) และ แอนดรอยด์ (Android) ในที่นี้จะขอกล่าวถึงระบบปฏิบัติการแอนดรอยด์ ซึ่งเป็นระบบปฏิบัติการที่กำลังเป็นที่นิยมในปัจจุบันนี้ อีกทั้งยังสามารถพัฒนาแอปพลิเคชัน (Application) ได้ด้วย ซึ่งจากรายงานวิจัย พบว่า ระบบปฏิบัติการแอนดรอยด์มีส่วนแบ่งการตลาดเป็นจำนวน 51.7% ตามมาด้วย ไอโอเอส 37% (Nielsen, 2012)

Operating System Share – All Smartphone Consumers vs. Recent Smartphone Acquirers (3Mo).

Q4 2011, Nielsen Mobile Insights



ภาพที่ 1.1 จำนวนส่วนแบ่งทางการตลาดของแต่ละระบบปฏิบัติการ

ที่มา: Operating System Share-All Smartphone Consumers vs. Recent Smartphone Acquirers (3Mo) [online] : เข้าถึง 4 ก.พ. 2555. จาก

<http://blog.nielsen.com/nielsenwire/consumer/more-us-consumers-choosing-smartphones-as-apple-closes-the-gap-on-android/>

เมื่อเทคโนโลยีโทรศัพท์เคลื่อนที่ที่มีการพัฒนามากขึ้น ทำให้เกิดช่องทางใหม่ ๆ ของการทำธุรกิจ คงปฏิเสธไม่ได้ว่า บริการรับส่งข้อความสั้น (Short Message Service) มีการตอบรับและใช้งานอย่างต่อเนื่อง สิ่งที่หลีกเลี่ยงไม่ได้กับการใช้งานดังกล่าวคือ ข้อความขยะต่าง ๆ ที่เจ้าของโทรศัพท์ที่ไม่ต้องการ ทำให้เกิดการรบกวน ราคาและเสียเวลาในการเข้าไปลบข้อความขยะเหล่านี้ อีกทั้งยังเป็นข้อความที่ไม่ประสงค์ดี ข้อความล่อลวง หรือข้อความที่อาจก่อให้เกิดความเสียหายในภายหลังได้ หากสามารถเลือกรับข้อความสั้น จากบุคคลที่รู้จัก ก็จะสามารถลดความเสี่ยงจากหลาย ๆ เหตุผลข้างต้น ได้มากพอสมควร ซึ่งจะสามารถตั้งค่าหรือกรองข้อความสั้นได้ โดยจะต้องอาศัยแอปพลิเคชันเข้ามาช่วยจัดการหรือกรองข้อความเข้าเหล่านี้ก่อน ดังนั้นในการศึกษานี้จึงมุ่งสร้างแอปพลิเคชันที่สามารถกรองข้อความสั้น โดยอาศัยข้อมูลจากรายชื่อผู้ติดต่อในสมุดโทรศัพท์ช่วยในการตัดสินใจ

1.2 วัตถุประสงค์ของการวิจัย

1. เพื่อต้องการออกแบบและพัฒนาแอปพลิเคชันให้สามารถรองรับข้อความสั้น บนโทรศัพท์เคลื่อนที่ที่รองรับระบบปฏิบัติการแอนดรอยด์ได้
2. เพื่อให้สามารถรองรับข้อความสั้นเฉพาะหมายเลขที่อยู่ในสมุดโทรศัพท์ ผ่านเข้ามาในกล่องข้อความเข้าได้
3. เพื่อลดจำนวนข้อความสั้นที่ไม่ต้องการหรือข้อความขยะจากบุคคลอื่นที่ไม่มีอยู่ในสมุดโทรศัพท์ได้

1.3 สมมติฐานของการวิจัย

สามารถรองรับข้อความสั้น โดยให้แสดงข้อความในกล่องข้อความเข้าเฉพาะเลขหมายที่ได้บันทึกในสมุดโทรศัพท์ได้เท่านั้น ส่วนเลขหมายอื่น ๆ หรือข้อความอื่น ๆ จะต้องถูกเก็บไว้ในแอปพลิเคชัน รวมถึงสามารถจัดการกับข้อความสั้นเหล่านั้นได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

สามารถพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ รวมถึงการรองรับข้อความเพื่อลดข้อความขยะและยังสามารถป้องกันและลดความเสี่ยงของข้อความอันตรายหรือข้อความคุกคามได้

บทที่ 2

แนวคิด ทฤษฎี และผลงานวิจัยที่เกี่ยวข้อง

2.1 แนวคิดและทฤษฎีที่เกี่ยวข้อง

ทุกวันนี้โลกของเทคโนโลยีมีความเจริญก้าวหน้าแบบก้าวกระโดด ยิ่งเทคโนโลยีที่เกี่ยวข้องกับการสื่อสารเคลื่อนที่โดยเฉพาะอย่างยิ่งระบบโทรศัพท์เคลื่อนที่ที่มีการเจริญก้าวหน้าเป็นอย่างมาก รวมถึงอุปกรณ์โทรศัพท์เคลื่อนที่ก็ยิ่งต้องพัฒนาเพื่อให้สามารถรองรับกับเทคโนโลยีนั้นได้ เมื่อมีการพัฒนาทางด้านฮาร์ดแวร์แล้ว ก็ต้องพัฒนาทางด้านซอฟต์แวร์ด้วย จะเห็นได้จากในปัจจุบันนี้ จะมีโทรศัพท์เคลื่อนที่จากหลายยี่ห้อ ต่างใช้ความสามารถทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์เพื่อช่วงชิงส่วนแบ่งทางการตลาดและมั่งคั่งของกลุ่มลูกค้าเป้าหมาย ยกตัวอย่างเช่น บริษัทโนเกีย ผลิตโทรศัพท์ยี่ห้อโนเกีย ใช้ระบบปฏิบัติการซิมเบียน บริษัทแอปเปิล ผลิตโทรศัพท์ไอโฟนหรือไอแพด เป็นต้น ใช้ระบบปฏิบัติการไอ (i-OS) บริษัทรีเซอส อิน โมชัน (RIM) ผลิตโทรศัพท์แบล็คเบอร์รี่ ใช้ระบบปฏิบัติการแบล็คเบอร์รี่ (Blackberry OS) บริษัทไมโครซอฟต์ คิดค้นและพัฒนา ระบบปฏิบัติการวินโดวส์โมบาย (Windows Mobile OS) โดยทำงานบนวินโดวส์โฟนและสิ่งที่สำคัญคือ ระบบปฏิบัติการแอนดรอยด์ คิดค้นและพัฒนาโดยบริษัทกูเกิล ซึ่งเป็นซอฟต์แวร์เปิด (Open-source OS) สามารถทำงานได้บนโทรศัพท์เคลื่อนที่หลาย ๆ ยี่ห้อและหลาย ๆ รุ่น

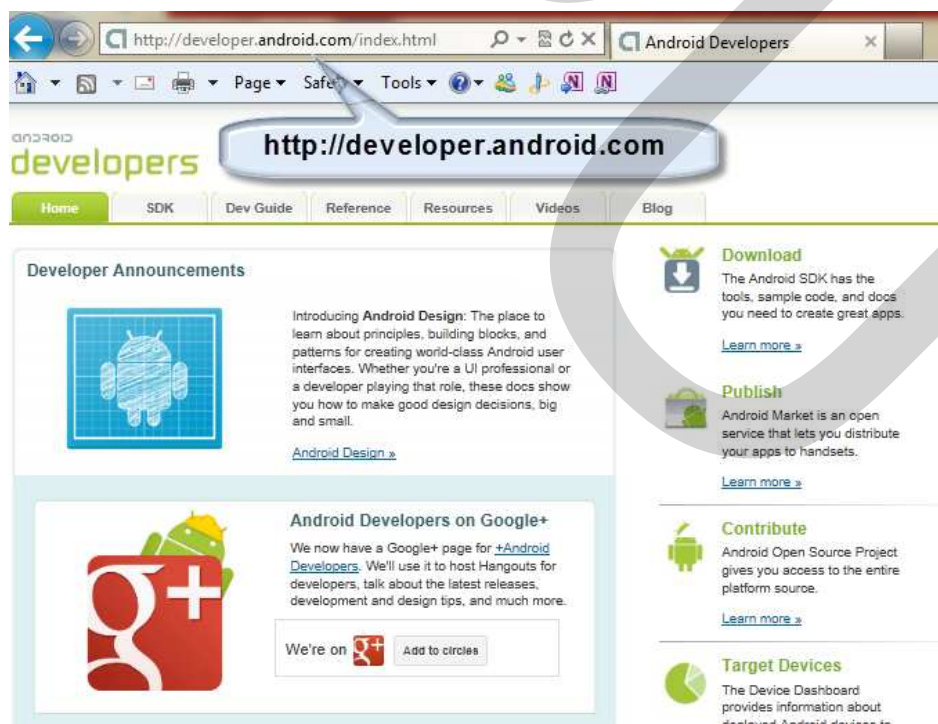
เมื่อมีการเจริญเติบโตทางด้านฮาร์ดแวร์และซอฟต์แวร์แล้ว สิ่งก็ตามมาคือโอกาสทางธุรกิจของผู้ให้บริการข้อมูลเนื้อหา (Content) ต่าง ๆ รวมถึงการใช้งานโปรแกรมขั้นพื้นฐาน ที่มีอยู่บนโทรศัพท์เคลื่อนที่ทุกยี่ห้อ นั่นคือ การใช้งานข้อความสั้น หรือ SMS นั่นเอง ยังมีข้อมูลเนื้อหา มากขึ้นเท่าใด การโฆษณาหรือการเชิญชวนผ่านบริการข้อความสั้น ก็จะสามารถทำได้ง่ายขึ้น หรือแม้กระทั่งข้อความสั้นที่เป็นอันตรายต่อผู้ใช้แล้ว อาจจะมีผลกระทบ คุณภาพ สิทธิและเสรีภาพ ได้

2.2 ประวัติระบบปฏิบัติการแอนดรอยด์

แอนดรอยด์ (Android) เป็นซอฟต์แวร์ที่มีโครงสร้างแบบเรียงทับซ้อนหรือแบบสแต็ก ซึ่งรวมเอาระบบปฏิบัติการ มิดเดิลแวร์ และแอปพลิเคชันที่สำคัญเข้าไว้ด้วยกันเพื่อใช้สำหรับทำงานบนอุปกรณ์พกพาโดยเฉพาะ เช่น โทรศัพท์มือถือ เป็นต้น การทำงานของแอนดรอยด์นี้มี

พื้นฐานอยู่บนระบบลินุกซ์เคอร์เนล ซึ่งใช้ Android SDK เป็นเครื่องมือสำหรับการพัฒนาแอปพลิเคชัน บนระบบปฏิบัติการแอนดรอยด์ โดยใช้ภาษาจาวา (Java) ในการพัฒนา

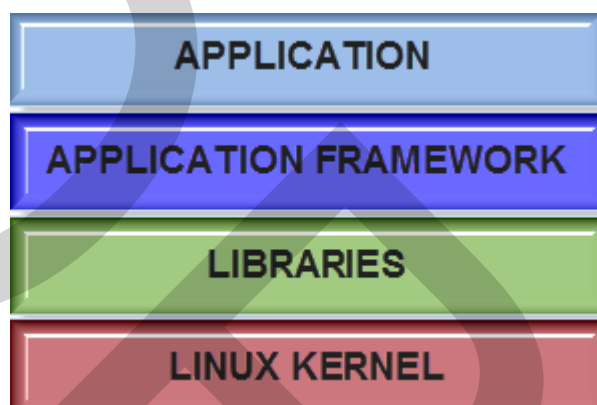
แอนดรอยด์เริ่มพัฒนาโดยบริษัทแอนดรอยด์และต่อมาได้ผนวกเข้ากับบริษัท กูเกิล (Google) ซึ่งในเดือนพฤศจิกายน พ.ศ. 2550 ได้มีการร่วมมือกันระหว่างบริษัทชั้นนำมากกว่า 33 บริษัทเพื่อพัฒนาระบบแอนดรอยด์ ทั้งบริษัทผู้ให้บริการโทรศัพท์เคลื่อนที่ บริษัทผู้ผลิตอุปกรณ์โทรศัพท์เคลื่อนที่ บริษัทซอฟต์แวร์ และบริษัทเอกชน ต่าง ๆ เช่น HTC, LG, Motorola, Samsung, China Mobile Communications, KDDI, DoCoMo, Sprint/Nextel, T-Mobile, Telecom Italia, Telefonica, Audience, Broadcom, Intel, Marvel, NVidia, eBay, Packet Video, TAT และ Wind River เป็นต้น โดยใช้ชื่อกลุ่มว่า OHA (Open Handset Alliance) ทั้งนี้ OHA ได้ร่วมมือกันพัฒนามาตรฐานสำหรับการพัฒนาซอฟต์แวร์ระบบเปิด (Open System) หรือโอเพ่นซอร์ส (Open Source) โดยมีลิขสิทธิ์ตาม Apache Version 2 license ซึ่งหลักลิขสิทธิ์ของ Apache จะอนุญาตให้ผู้พัฒนาสามารถนำโค้ดที่มีอยู่ไปพัฒนาต่อได้ ทั้งในส่วนของแบบการค้า (Commercial) หรือซอฟต์แวร์กรรมสิทธิ์ (Proprietary) และแบบใช้ฟรีหรือฟรีแวร์ (Freeware) ก็ได้ โดยสามารถศึกษาหรือดาวน์โหลดซอฟต์แวร์ได้จากเว็บไซต์ <http://developer.android.com> (จักรชัย โสอินทร์ และ พงษ์ศธร จันทร์ช้อย, 2554)



ภาพที่ 2.1 หน้าเพจของเว็บไซต์ Android Developer

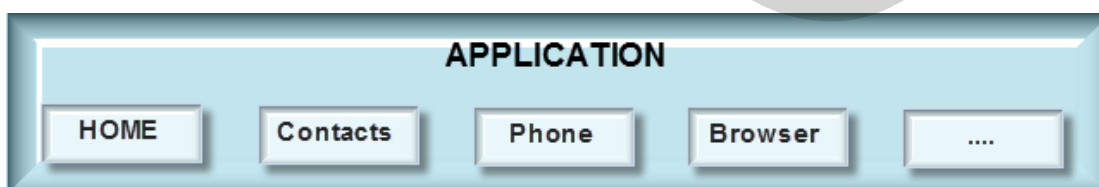
2.3 สถาปัตยกรรมของแอนดรอยด์

สถาปัตยกรรมของการออกแบบแอนดรอยด์ (พริ่อมเลิศ หล่อวิจิตร, 2555) ถูกแบ่งออกเป็นลำดับชั้นหรือจะเรียกว่าเลเยอร์ (Layer) โดยที่แต่ละชั้นจะเรียกใช้บริการจากระดับชั้นที่อยู่ด้านล่างของตัวเอง ซึ่งสามารถแบ่งออกเป็น 4 ชั้นหลัก คือ ชั้นลินุกซ์เคอร์เนล (Linux Kernel) ชั้นไลบรารี (Library) ชั้นแอปพลิเคชันเฟรมเวิร์ก (Application Framework) และชั้นแอปพลิเคชัน ตามภาพที่ 2.2



ภาพที่ 2.2 สถาปัตยกรรมของระบบแอนดรอยด์

2.3.1 ชั้นแอปพลิเคชัน (Application) เป็นชั้นบนสุดของโครงสร้างแอนดรอยด์ ซึ่งเป็นส่วนของแอปพลิเคชันที่พัฒนาขึ้นมาใช้งาน เช่น แอปพลิเคชันรับ-ส่ง อีเมลล์ แอปพลิเคชันโทรศัพท์ (Phone Dial) แอปพลิเคชันเว็บเบราว์เซอร์ (Web Browser) เป็นต้น ทั้งนี้โปรแกรมในชั้นแอปพลิเคชันจะอยู่ในรูปแบบของไฟล์ .apk ซึ่งโดยทั่วไปแล้วจะอยู่ในไดเรกทอรี data/app ของโทรศัพท์เคลื่อนที่ ตามภาพที่ 2.3



ภาพที่ 2.3 สถาปัตยกรรมชั้นแอปพลิเคชัน

2.3.2 ชั้นแอปพลิเคชันเฟรมเวิร์ก (Application Framework) โดยปกติแล้วนักพัฒนาสามารถเรียกใช้งานแอนดรอยด์ผ่าน API (Application Programming Interface) ได้ ซึ่งแอนดรอยด์ได้ออกแบบไว้เพื่อลดความซ้ำซ้อนในการใช้งานซ้ำของ Application Component ซึ่งมีตัวอย่างแอปพลิเคชันเฟรมเวิร์ก ดังนี้

2.3.2.1 Activity Manager เป็นส่วนควบคุมวงจรชีวิตของแอปพลิเคชัน

2.3.2.2 Window Manager จัดการโปรแกรมของ Windows ทั้งหมด

2.3.2.3 Content Providers เป็นส่วนควบคุมการเข้าถึงของข้อมูลที่มีการใช้งานได้ร่วมกันระหว่างแอปพลิเคชันที่แตกต่างกัน เช่น ข้อมูลผู้ใช้ (Contact)

2.3.2.4 View System เป็นส่วนควบคุมการทำงานสำหรับการสร้างแอปพลิเคชัน เช่น lists, grids, text boxes, buttons และ embeddable web browser

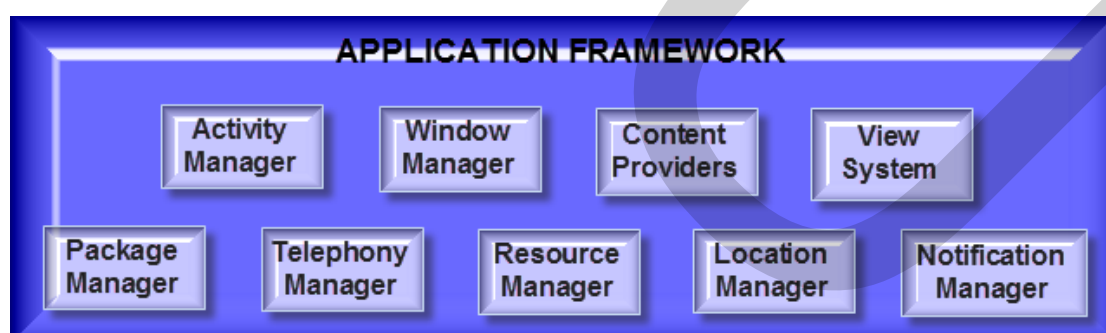
2.3.2.5 Package Manager เป็นการจัดการโปรแกรมของแอนดรอยด์ที่อยู่ในระบบทั้งหมด

2.3.2.6 Telephony Manager จัดการฟังก์ชันทั้งหมดที่เกี่ยวข้องกับโทรศัพท์

2.3.2.7 Resource Manager เป็นส่วนดูแลการเข้าใช้ข้อมูลต่าง ๆ ที่ไม่ใช่โค้ด เช่น localized string, graphics และ layout ซึ่งจะอยู่ในไดเรกทอรี res/

2.3.2.8 Location Manager เป็นส่วนดูแลตำแหน่งของอุปกรณ์พกพาเคลื่อนที่

2.3.2.9 Notification Manager เป็นส่วนควบคุมเหตุการณ์ต่าง ๆ ที่แสดงบนแถบสถานะ เช่น กรณีที่ได้รับข้อความและการแจ้งเตือนต่าง ๆ



ภาพที่ 2.4 สถาปัตยกรรมชั้นแอปพลิเคชันเฟรมเวิร์ก

2.3.3 ชั้นไลบรารี (Library) แอนดรอยด์ได้รวบรวมกลุ่มของไลบรารีต่าง ๆ ที่สำคัญและมีความจำเป็นต่อการพัฒนาโปรแกรม ซึ่งถูกเขียนไว้ด้วยภาษาซี (C) และ ซีพลัสพลัส (C++) โดยตัวอย่างของไลบรารีที่สำคัญมีดังนี้

2.3.3.1 System C library เป็นกลุ่มของไลบรารีมาตรฐานที่อยู่บนพื้นฐานของภาษาซี ไลบรารี (libc)

2.3.3.2 Media Libraries เป็นกลุ่มการทำงานมัลติมีเดีย เช่น ออดิโอ วิดีโอ รวมถึงรูปภาพต่าง ๆ เช่น MPEG4, H.264, MP3, AAC, AMR, JPG และ PNG

2.3.3.3 Surface Manager เป็นกลุ่มการจัดการรูปแบบของหน้าจอและการวาดหน้าจอ

2.3.3.4 2D/3D Libraries เป็นกลุ่มของกราฟิกแบบ 2 มิติ หรือ SGL (Scalable Graphics Library) และแบบ 3 มิติ หรือ OpenGL

2.3.3.5 Free Type เป็นกลุ่มของบิตแมป (Bitmap) และเวกเตอร์ (Vector) สำหรับการเรนเดอร์ภาพ

2.3.3.6 SQLite เป็นกลุ่มของฐานข้อมูล ซึ่งเป็นระบบฐานข้อมูลที่ใช้เช่นเดียวกับซอฟต์แวร์ Firefox และ Apple iPhone ซึ่งสามารถใช้ฐานข้อมูลนี้เพื่อเก็บข้อมูลของแอปพลิเคชันต่าง ๆ ได้

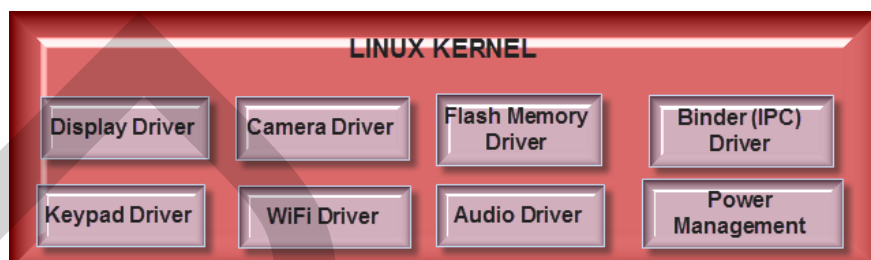
2.3.3.7 Browser Engine เป็นกลุ่มของการแสดงผลบนเว็บเบราว์เซอร์โดยอยู่บนพื้นฐานของ Webkit สำหรับการเรียกใช้แอปพลิเคชันต่าง ๆ ในชั้นไลบรารี จะไม่สามารถเรียกใช้ไลบรารีในระดับเดียวกับตัวเองได้ โดยจะต้องเรียกใช้แอปพลิเคชัน ในชั้นที่สูงกว่าเท่านั้นจึงจะสามารถเรียกใช้ได้ นอกจากนี้ในชั้นไลบรารีนี้ แอนดรอยด์ยังแบ่งเป็นชั้นย่อยที่เรียกว่า Android Runtime ซึ่งประกอบไปด้วย 2 ส่วนหลักดังนี้

2.3.3.8 Dalvik VM (Virtual Machine) ส่วนนี้ถูกเขียนด้วยภาษาจาวา เพื่อใช้เฉพาะการใช้งานอุปกรณ์พกพาเคลื่อนที่ โดยจะมีเครื่องมือที่ชื่อว่า dx ทำหน้าที่ในการบีบอัดคลาสจาวา ทั้งนี้ไฟล์ .dex จะมีขนาดกะทัดรัดและเหมาะสมกับอุปกรณ์พกพามากกว่า .class โดยเฉพาะอย่างยิ่งประสิทธิภาพในการใช้พลังงานจากแบตเตอรี่

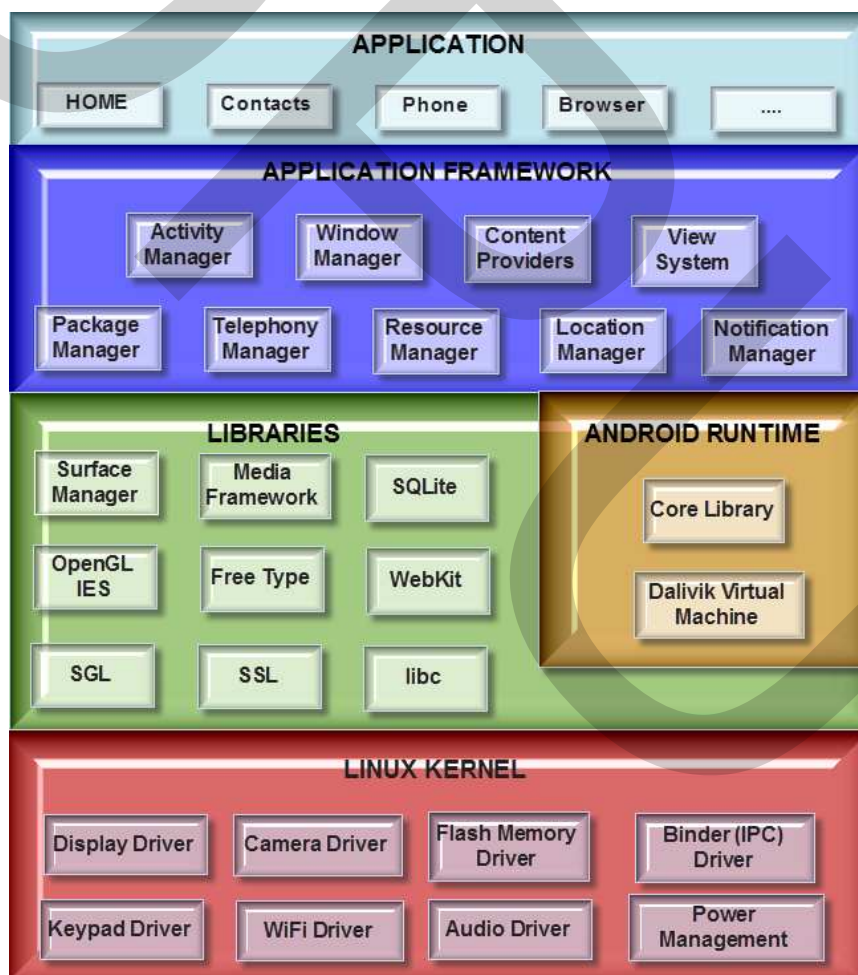
2.3.3.9 Core Java Library ส่วนนี้เป็นไลบรารีมาตรฐาน แต่ก็จะมีความแตกต่างจากไลบรารีของ Java SE และ Java ME

2.3.4 ชั้นลินุกซ์เคอร์เนล (Linux Kernel) ระบบแอนดรอยด์อยู่บนพื้นฐานของระบบปฏิบัติการลินุกซ์ โดยชั้นลินุกซ์เคอร์เนล นั้นมีฟังก์ชันการทำงานหลาย ๆ ส่วน ซึ่งแต่ละส่วนถูกพัฒนาขึ้นด้วยภาษาซี เช่น การจัดการหน่วยความจำ การจัดการโปรเซส การเชื่อมต่อเครือข่าย

และฟังก์ชันการทำงานส่วนอื่นที่เกี่ยวข้องกับระบบปฏิบัติการและสถาปัตยกรรมของแอนดรอยด์
ทั้งระบบตามภาพที่ 2.6



ภาพที่ 2.5 สถาปัตยกรรมชั้นลินุกซ์เคอร์เนล



ภาพที่ 2.6 สถาปัตยกรรมของแอนดรอยด์ทั้งระบบ

2.4 Short Message Service (SMS)

ในประเทศไทยมีการเริ่มใช้งานในเครือข่ายโทรศัพท์เคลื่อนที่ระบบ GSM ตั้งแต่ปี 1992 ต่อมามีการใช้งานในเครือข่ายโทรศัพท์เคลื่อนที่ระบบ CDMA ในเอกสารนี้ได้กล่าวถึงหลักการส่งข้อความ SMS ของระบบเครือข่ายโทรศัพท์เคลื่อนที่ซึ่งประกอบด้วย ข้อความ (Message) ผู้ส่ง (Sender) เครือข่ายการสื่อสาร (Network) และ ผู้รับ (Receiver)

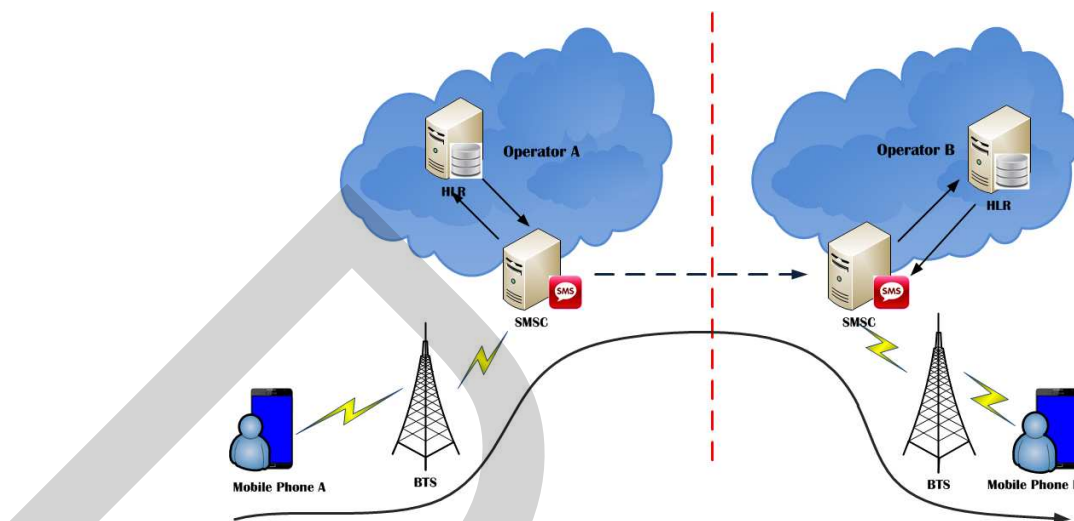
2.4.1 Message โครงสร้างของข้อความในระบบ SMS ประกอบด้วยตัวอักษรแบบ ASCII ขนาด 7 บิต สามารถบรรจุตัวอักษรภาษาอังกฤษได้จำนวน 160 ตัวอักษรต่อข้อความ โดยอักษรแต่ละตัวมีความแตกต่างกัน 126 แบบ ซึ่งมีทั้งตัวอักษรภาษาอังกฤษตัวพิมพ์เล็ก ตัวพิมพ์ใหญ่ ตัวเลข และสัญลักษณ์พิเศษต่าง ๆ และสามารถบรรจุได้จำนวน 70 ตัวอักษรต่อข้อความในแบบ Unicode ขนาด 16 บิต เช่น ภาษาไทย เป็นต้น

2.4.2 Sender สามารถแบ่งได้ 2 ประเภท ได้แก่

2.4.2.1 ผู้ส่งที่ส่งข้อความจากโทรศัพท์เคลื่อนที่หรืออุปกรณ์เคลื่อนที่อื่น ๆ

2.4.2.2 ผู้ส่งที่ส่งข้อความผ่านอินเทอร์เน็ต โดยผู้ให้บริการโทรศัพท์เคลื่อนที่ (Operator) อาจเปิดให้บริการส่งข้อความผ่านโปรโตคอล TCP/IP โดยการให้ดาวน์โหลดโปรแกรมสำหรับส่ง SMS โดยเฉพาะผู้ส่งกลุ่มนี้มีความยืดหยุ่นสูงในการส่งข้อความ และส่งได้ครั้งละหลายข้อความ เช่น การส่ง SMS จากโรงพยาบาล เพื่อเตือนให้คนไข้มาตามนัด หรือการรายงานธุรกรรมทางการเงิน เป็นต้น

จากภาพที่ 2.7 เป็นการส่ง SMS จากผู้ส่ง A ไปยังผู้รับ B โดยสามารถแบ่งออกเป็น 2 ส่วน ได้แก่ ส่วนแรก คือ A ฟังผู้ส่ง (Mobile Originating) หรือ MO ประกอบด้วย Base Station (BS) ที่ทำหน้าที่จัดการคลื่นความถี่วิทยุ และส่งเข้าไปยัง Short Messaging Service Center (SMSC) ของผู้ส่ง A และหลังจากนั้นจะถูกส่งต่อไปยังระบบ SMSC ของผู้รับ B (Mobile Terminating) หรือ MT



ภาพที่ 2.7 การส่ง SMS จาก mobile Phone A ไปยัง Mobile Phone B

2.4.3 Short Messaging Service Center (SMSC) ทำหน้าที่เป็นตัวกลางที่ใช้ในการถ่ายทอดเก็บ และส่งต่อ SMS (Store and Forward) ซึ่งจะต้องมีความน่าเชื่อถือสูง รองรับผู้ใช้และ Message Throughput ได้จำนวนมาก นอกจากนี้ระบบควรจะสามารถขยายการตอบสนองต่อความต้องการใช้งาน SMS ที่เพิ่มขึ้นได้ง่าย

2.4.4 Receiver ผู้รับคือเครื่องโทรศัพท์เคลื่อนที่ปลายทาง เมื่อผู้รับเปิดใช้งาน เครื่องโทรศัพท์จะเข้าไปลงทะเบียนที่ Home Location Register (HLR) บริการต่าง ๆ ที่เกิดขึ้นก่อนที่ผู้รับเปิดเครื่อง เช่น SMS หรือ MMS จะทราบตำแหน่งของเครื่องผู้รับ และดำเนินการส่งให้กับผู้รับอย่างถูกต้อง

2.5 นิยามของ Spam

สยามอินเทลลิเจนซ์ยูนิต (SIU., 2553) Spam เป็นชื่อที่ใช้เรียก “ข้อความที่ไม่ได้เรียกร้องให้ส่ง” ที่ถูกส่งผ่านสื่ออิเล็กทรอนิกส์ ในรูปแบบต่าง ๆ เช่น จดหมายอิเล็กทรอนิกส์ (Email) ข้อความสั้น (SMS) ข้อความมัลติมีเดีย (MMS) หรือข้อความด่วน (Instant Messaging) ในบางกรณีอาจครอบคลุมถึงโทรสาร (Fax) ด้วยเช่นกัน โดย SMS Spam ก็จะเป็นตัวบ่งบอกว่าเป็น Spam ที่ส่งผ่านระบบข้อความสั้น (SMS) ของโทรศัพท์เคลื่อนที่

2.5.1 Spam บนโทรศัพท์เคลื่อนที่ (SMS Spam) โดยตัวอย่างของข้อความที่เข้าข่าย spam ด้านโทรศัพท์เคลื่อนที่มีดังนี้

2.5.1.1 ข้อความขายสินค้าหรือบริการ

2.5.1.2 ข้อความทาง SMS ที่ล่อหลอก ให้ผู้รับโทรเข้าไปยังหมายเลขที่คิดราคาค่าโทร ในราคาที่ต่างไปจากปกติ

2.5.1.3 ข้อความลวง ที่ล่อหลอกให้ผู้รับส่งข้อมูลส่วนบุคคลหรือข้อมูลทางการเงิน ที่สามารถใช้ทำธุรกรรมได้

2.5.2 มาตรการป้องกัน SMS Spam ในปัจจุบัน (SIU., 2553)

2.5.2.1 การลงทะเบียนเลขหมายไม่รับข้อความขยะ วิธีนี้นำแนวคิดมาจากนโยบาย Do-not-call Service ที่เคยใช้ประสบความสำเร็จกับการโฆษณาผ่านโทรศัพท์ ขั้นตอนคือ ผู้ใช้โทรศัพท์มือถือต้องลงทะเบียนเลขหมายของตัวเอง เพื่อระบุว่าไม่ต้องการรับข้อความขยะทาง SMS ซึ่งผู้ส่งข้อความจะไม่สามารถส่งข้อความไปยังหมายเลขที่ลงทะเบียน โดยไม่ได้รับอนุญาตล่วงหน้าได้ มิฉะนั้นจะมีความผิดทางกฎหมาย วิธีการนี้มีจุดอ่อนตรงที่ไม่สามารถป้องกันข้อความขยะหรือการโทรศัพท์เพื่อขายสินค้าได้ครบทุกกรณี เช่นการติดต่อจากหมายเลขต่างประเทศ เป็นต้น

2.5.2.2 การสร้างช่องทางรายงาน SMS spam ในประเทศฝรั่งเศส ผู้ให้บริการโทรศัพท์มือถือภายในประเทศ ได้ชักชวนให้ผู้ได้รับ SMS spam ทำการส่งต่อ (forward) ข้อความไปยังศูนย์ stop-spam ที่หมายเลข 33700 เพื่อรายงานข้อความที่ถือว่าเป็น spam ผลที่เกิดขึ้นคือ ได้รับข้อความขยะถึง 500,000 ครั้ง ส่งผลให้สามารถสั่งปิดเลขหมายโทรศัพท์ที่เป็นผู้ส่งข้อความขยะได้ 300 เลขหมาย และแจ้งเตือนเจ้าของเลขหมายอื่น ๆ อีกเป็นจำนวนมากให้หยุดส่งข้อความขยะ

2.5.2.3 การสร้างฐานข้อมูล SMS spam ระดับนานาชาติ โดยที่ GSM Association ได้พัฒนาระบบการแจ้งว่าข้อความใดเป็น SMS spam (GSMA Spam Reporting Service) เพื่อให้ผู้ให้บริการโทรศัพท์เคลื่อนที่ทั่วโลก สามารถรายงานข้อมูลเข้าไปยังฐานข้อมูลกลางระดับโลกได้ที่หมายเลข 7726 การรวบรวมข้อมูลในระดับนานาชาติ จะช่วยให้การวิเคราะห์รูปแบบและพฤติกรรมของผู้ส่งข้อความขยะทำได้แม่นยำมากขึ้น ขณะนี้ระบบ GSMA Spam Reporting Service กำลังอยู่ในช่วงทดลองใช้โดยผู้ให้บริการโทรศัพท์มือถือรายใหญ่บางแห่ง เช่น AT&T ของสหรัฐอเมริกา Korea Telecom ของเกาหลีใต้ และ SFR ของฝรั่งเศส เป็นต้น

2.5.2.4 การสร้างข้อตกลงเบื้องต้นระหว่างผู้ให้บริการโทรศัพท์มือถือ โดยหน่วยงาน GSM Association ซึ่งเป็นสมาคมของผู้ให้บริการโทรศัพท์มือถือในระดับนานาชาติ ได้ออกคำแนะนำที่เรียกว่า “Mobile Spam Code of Practice” เพื่อเป็นจุดเริ่มต้นให้บริษัทผู้ให้บริการโทรศัพท์มือถือที่เป็นสมาชิกของสมาคม ได้นำไปปฏิบัติเพื่อลดปัญหา SMS spam แก่ผู้ใช้คำแนะนำนี้ไม่ใช่ข้อบังคับของ GSM Association ผู้ให้บริการสามารถปฏิบัติตามได้ตามความ

สมัครใจ โดยครอบคลุม SMS spam 3 รูปแบบ ได้แก่ ข้อความเชิงการค้าที่ส่งมายังผู้รับ โดยไม่ได้ อนุญาต ข้อความเชิงการค้าที่เชิญชวนให้ผู้รับทำการโทรหรือส่งข้อความเข้าไปยังหมายเลขที่คิด ราคาพิเศษ และข้อความผิดกฎหมายที่ส่งไปยังผู้รับจำนวนมาก เช่น ปลอมแปลงชื่อผู้ส่งให้เข้าใจผิด หรือล่อลวงข้อมูลส่วนบุคคล

2.5.2.5 การใช้ซอฟต์แวร์กรอง SMS spam ที่ฝั่งเซิร์ฟเวอร์ โดยผู้ให้บริการ โทรศัพท์เคลื่อนที่บางราย ได้เริ่มติดตั้งซอฟต์แวร์กรอง SMS spam ที่ฝั่งเซิร์ฟเวอร์ ของผู้ให้บริการ เอง ตัวอย่างเช่น บริษัท Pakistan Mobile Communications Limited เจ้าของบริษัท Mobilink GSM ได้ติดตั้งซอฟต์แวร์ TekMedia Anti-Spam Firewall เพื่อกรองข้อความขยะ ในเดือนกรกฎาคม ค.ศ. 2008 และ China Mobile ได้ประกาศว่าจะสแกนข้อความใน SMS ทุกฉบับที่ส่งผ่านเครือข่าย เพื่อ ตรวจสอบว่ามีข้อความที่ไม่ได้รับอนุญาตจากรัฐบาลหรือไม่

2.5.2.6 การใช้ซอฟต์แวร์กรอง SMS spam ที่ฝั่งเครื่องโทรศัพท์ โดยที่โทรศัพท์มือถือ บางรุ่นที่อนุญาตให้ผู้ผู้ใช้ติดตั้งโปรแกรมเพิ่มเติมเองได้ สามารถติดตั้งโปรแกรมช่วยกรอง SMS spam ที่ฝั่งผู้ใช้ได้ ลักษณะการทำงานจะคล้ายคลึงกับตัวกรองอีเมลล์

2.5.2.7 การปิดบริการ SMS เป็นบางประเภท ในบางประเทศ เช่น สหรัฐอเมริกา ผู้ใช้งานโทรศัพท์มือถือสามารถปิดกั้นไม่รับ SMS เป็นบางประเภทได้ เช่น เลิกไม่รับ SMS ที่ส่ง จากอีเมลล์หรือส่งจากหน้าเว็บ ซึ่งมีโอกาสเป็น SMS spam สูงกว่า โดยสามารถเข้าไปเลือกปิดที่ เว็บไซค์ของผู้ให้บริการมือถือที่ใช้อยู่

2.5.2.8 การปิดบริการไม่ใช้งาน SMS ทุกกรณี คือปิดการใช้งาน SMS ทั้งหมด โดยไม่ สนใจว่าเป็นข้อความชนิดใด ซึ่งผู้ให้บริการโทรศัพท์มือถือส่วนมากอนุญาตให้ทำได้

2.5.3 SMS spam ในประเทศไทย จากงานวิจัยมีการสำรวจข้อมูลของ SMS spam จากผู้ใช้ โทรศัพท์มือถือในประเทศไทย จากเว็บบอร์ด Pantip.com ห้อง MBK (SIU., 2553) พบว่าผู้ใช้ โทรศัพท์มือถือรายงานปัญหา SMS รบกวนเข้ามาเป็นจำนวนมาก ดังนี้

2.5.3.1 มีผู้รายงานปัญหา SMS spam เข้ามาทั้ง 3 เครือข่ายหลัก คือ AIS DTAC และ TrueMove จำนวนข้อความเฉลี่ยที่ผู้ใช้งานโทรศัพท์รายหนึ่งระบุอยู่ประมาณ 4-7 ข้อความต่อวัน

2.5.3.2 ผู้ใช้โทรศัพท์รายหนึ่งได้รวบรวมจดหมายขยะที่ได้รับตลอดเดือนสิงหาคม 2553 ได้ทั้งสิ้น 41 ข้อความหรือคิดเฉลี่ยวันละ 1.32 ข้อความ

2.5.3.3 ผู้ใช้โทรศัพท์รายหนึ่งรายงานว่าได้รับข้อความจาก True Sport วันละ 4-5 ข้อความ คิดบริการครั้งละ 5.35 บาทโดยระบุว่าไม่ได้สมัครใช้บริการใด ๆ

2.5.3.4 โทรศัพท์เกือบทั้งหมดรายงานว่าจะไม่เคยสมัครบริการเพื่อรับ SMS เหล่านี้เลย

2.5.4 ชนิดของข้อความ SMS Spam ในประเทศไทย สามารถจำแนกข้อความได้เป็นหลายกลุ่มดังนี้

2.5.4.1 ข้อความจากผู้ให้บริการโดยตรง มักเป็นข้อความประชาสัมพันธ์บริการของเครือข่าย โดยอาจมีหมายเลขโทรศัพท์สำหรับสอบถามข้อมูลเพิ่มเติม ซึ่งไม่คิดค่าบริการ

2.5.4.2 เป็นข้อความจากผู้ให้บริการโดยตรง แต่เป็นข้อความประชาสัมพันธ์บริการของเครือข่ายที่คิดค่าใช้บริการ ในบางกรณีจะเข้าข่ายการชิงโชค โดยประชาสัมพันธ์ว่าส่งข้อความกลับไปยังหมายเลขที่กำหนดมากเท่าใด ก็จะมีสิทธิ์ลุ้นรับโชคมากขึ้น

2.5.4.3 เป็นข้อความประชาสัมพันธ์จากห้างร้าน บริษัท ร้านค้า ที่มีความสัมพันธ์ทางธุรกิจกับผู้ให้บริการเครือข่ายโดยตรง

2.5.4.4 เป็นข้อความประชาสัมพันธ์จากห้างร้าน บริษัท ร้านค้าต่าง ๆ ที่ไม่ระบุแน่ชัดว่ามีความสัมพันธ์เชิงธุรกิจกับผู้ให้บริการเครือข่ายหรือไม่ แต่เป็นบริษัท ร้านค้า ที่มีชื่อเสียงในสังคม และเป็นที่รู้จักกันทั่วไป ตัวอย่างเช่น เครือบริษัท GMM และ RS ซึ่งเน้นให้บริการโหลดเพลงผ่านโทรศัพท์มือถือ หรือร้านอาหาร Sizzler ที่ให้ส่งข้อความเพื่อชิงโชค

2.5.4.5 ข้อความที่มาจาก บริษัท ห้างร้าน ที่ไม่เป็นที่รู้จักกันมากนัก และในบางกรณีไม่ได้ลงชื่อผู้ส่ง แต่ลงเป็นหมายเลขโทรศัพท์แทน ข้อความเกือบทุกชิ้นเป็นการประชาสัมพันธ์บริการผ่านโทรศัพท์มือถือ เชิญชวนให้ส่ง SMS เพื่อชิงโชค บางข้อความมีลักษณะส่อไปถึงเรื่องเพศ ส่วนหัวข้อการประชาสัมพันธ์มักเป็นเรื่องดวงชะตา และเรื่องข่าวซุบซิบในวงการบันเทิง

2.6 แอปพลิเคชันที่เกี่ยวข้อง

2.6.1 Anti SMS Bomber (provost, 2012)

ตารางที่ 2.1 แอปพลิเคชันของ Anti SMS Bomber

ความสามารถ	บล็อกข้อความสั้นที่ซ้ำกันจากผู้ส่งหลาย ๆ คนได้ เรียกดูประวัติข้อความได้ ตัดข้อความสั้นที่ว่างเปล่าได้
ความต้องการของระบบ	Android: 2.0 หรือสูงกว่า
รุ่นปัจจุบัน	1.1
ขนาดไฟล์	132 k

ตารางที่ 2.1 (ต่อ) แอปพลิเคชันของ Anti SMS Bomber

นักพัฒนา	Sebastian pruvost
----------	-------------------

2.6.2 Blacklist (Anttek, 2012)

ตารางที่ 2.2 แอปพลิเคชันของ Blacklist

ความสามารถ	บล็อก การโทร/ข้อความสั้น จากรายการเลขหมายได้ บล็อก การโทร/ข้อความสั้น สำหรับเลขหมายที่ไม่รู้จักได้ อิมพอร์ต สมุดโทรศัพท์/ประวัติการโทร/ ข้อความสั้นที่อยู่ใน บล็อก ลิส ได้ สามารถจัดการประวัติการบล็อกได้
ความต้องการของระบบ	Android: 2.0 -2.2
รุ่นปัจจุบัน	0.97
ขนาดไฟล์	167 k
นักพัฒนา	Anttek

2.6.3 BlackList free (Ming, 2012)

ตารางที่ 2.3 แอปพลิเคชันของ Blacklist Free

ความสามารถ	บล็อก การโทร/ข้อความสั้น บล็อก การโทร สำหรับเลขหมายที่ไม่รู้จัก กรองข้อความสั้นโดยใช้ คำเฉพาะ ป้องกันด้วยรหัสผ่าน
ความต้องการของระบบ	Android: 1.6 หรือสูงกว่า
รุ่นปัจจุบัน	1.8.1
ขนาดไฟล์	269 k

ตารางที่ 2.3 (ต่อ) แอปพลิเคชันของ Blacklist Free

นักพัฒนา	Ming Software
----------	---------------

2.6.4 AntiSpamSMS (Airplanez, 2012)

ตารางที่ 2.4 แอปพลิเคชันของ Anti Spam SMS

ความสามารถ	บล็อก Spam ข้อความสั้น บล็อกเลขหมายที่ไม่มีการแสดง
ความต้องการของระบบ	Android: 1.6 หรือสูงกว่า
รุ่นปัจจุบัน	1.02
ขนาดไฟล์	322 k
นักพัฒนา	airplanez ประเทศเกาหลี

2.6.5 SMS Filter (Nachev, 2012)

ตารางที่ 2.5 แอปพลิเคชันของ SMS Filter

ความสามารถ	เพิ่ม/ลบ เลขหมายจากรายการแบล็กลิส ได้ บันทึกประวัติการบล็อกเป็นไฟล์ได้ แสดง/ล้าง ประวัติของข้อความเข้าได้
ความต้องการของระบบ	Android 1.5 หรือสูงกว่า
รุ่นปัจจุบัน	1.0.1
ขนาดไฟล์	35 k
นักพัฒนา	Tsvetan Nachev

2.6.6 Anti SMS Spam & Text Filter (Droid Mate, 2012)

ตารางที่ 2.6 แอปพลิเคชันของ Anti SMS Spam & Text Filter

ความสามารถ	คำเฉพาะแบล็กลิส รายการเลขหมายที่บล็อก บล็อกเลขหมายที่ไม่รู้จัก
ความต้องการของระบบ	Android: 2.0 หรือสูงกว่า
รุ่นปัจจุบัน	0.96
ขนาดไฟล์	162 k
นักพัฒนา	Droid Mate

บทที่ 3

ระเบียบวิธีวิจัย

3.1 แนวทางการวิจัยและพัฒนา

สารนิพนธ์นี้มีวัตถุประสงค์เพื่อต้องการออกแบบและพัฒนาแอปพลิเคชัน การกรองข้อความสั้น บนโทรศัพท์เคลื่อนที่ ที่รองรับระบบปฏิบัติการแอนดรอยด์ ให้สามารถกรองข้อความสั้นเฉพาะหมายเลขที่มีอยู่ในสมุดโทรศัพท์ได้เพื่อลดจำนวนข้อความสั้นที่ไม่ต้องการ โดยแบ่งขั้นตอนการทำงานออกเป็น 3 ส่วน ดังนี้

3.1.1 ศึกษาข้อมูลทั่วไป ประกอบด้วย หลักการทำงานของระบบปฏิบัติการแอนดรอยด์ วิธีการพัฒนาแอปพลิเคชันบนแอนดรอยด์ เครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชัน และวิธีการเชื่อมต่อและตรวจสอบระหว่างการทำงานของแอปพลิเคชันการกรองข้อความสั้นกับระบบข้อความสั้นบนแอนดรอยด์

3.1.2 ออกแบบแอปพลิเคชันสำหรับการกรองข้อความสั้นบนโทรศัพท์เคลื่อนที่แอนดรอยด์ ในการออกแบบแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์นั้นจะต้องใช้เครื่องมือเพื่อช่วยในการพัฒนา ทั้งด้านของการเชื่อมต่อกันระหว่าง แอปพลิเคชันกับโทรศัพท์เคลื่อนที่แอนดรอยด์ ระบบข้อความสั้นกับแอปพลิเคชันการกรองข้อความ และการจัดการกับข้อความสั้นที่ได้กรองไว้

3.1.3 พัฒนาแอปพลิเคชันการกรองข้อความสั้น ตามที่ได้ออกแบบไว้ และแก้ไขปรับปรุงแอปพลิเคชันให้สามารถทำงานได้ตามวัตถุประสงค์ที่ได้วางไว้ โดยทดสอบกับอิมูเลเตอร์และบนโทรศัพท์เคลื่อนที่จริง

3.2 เครื่องมือที่ใช้ในงานวิจัย

3.2.1 โทรศัพท์เคลื่อนที่ที่รองรับ ระบบปฏิบัติการแอนดรอยด์ จำนวน 1 เครื่อง ได้แก่ Samsung Galaxy S IIระบบ WCDMA/HSPA 850/900/1900/2100 MHz.

โครงข่าย 3G WCDMA/HSPA+

ความเร็วอินเทอร์เน็ต ดาว์นโหลด 21 Mbps อัปโหลด 5.76 Mbps

หน้าจอ 4.3 นิ้ว WVGA Super AMOLED Plus

ความละเอียดหน้าจอ 480x800 พิกเซล 16 ล้านสี

หน่วยประมวลผล 1.2 กิกะเฮิร์ตซ์ ดูอัลคอร์

หน่วยความจำภายใน 16 กิกะไบต์

กล้องถ่ายรูป 8 ล้านพิกเซล พร้อมระบบออโต้โฟกัส

บลูทูธ V3.0 + A2DP

Wi-Fi 802.11 b/g/n

ระบบปฏิบัติการ Android 2.3 Gingerbread

ข้อความ SMS และ MMS

อีเมล POP3/IMAP/SMTP

จาวา MIDP emulator

แบตเตอรี่ ลิเทียมไอออน 1650 มิลลิแอมป์

3.2.2 เครื่องคอมพิวเตอร์ส่วนบุคคล จำนวน 1 เครื่อง มีคุณสมบัติดังนี้

CPU Intel Core i5-2410M (2.30 GHz, 3 MB L3 Cache)

RAM 4 GB DDR3

HARD DISK 640 GB 5400 RPM

Graphic Chip nVidia GeForce GT 520M (1 GB GDDR3)

DVD +/-RW Drive

USB 2.0

Display 14.1 inch WXGA (1366x768) LED

MS Windows 7

3.2.3 ซอฟต์แวร์สำหรับการพัฒนา

Java Development Kit (JDK)

โปรแกรม Eclipse

Android SDK

Android Development Tools Plugin

3.3 แผนการดำเนินงาน

ในการออกแบบและพัฒนาระบบกรองข้อความสั้นเฉพาะเลขหมายนั้น มีแผนการดำเนินงานดังตารางที่ 3.1

ตารางที่ 3.1 แผนการดำเนินงาน

ลักษณะงาน	ตารางเวลาของโครงการ (เดือนที่)							
	1	2	3	4	5	6	7	8
1. ศึกษาการทำงานของระบบปฏิบัติการแอนดรอยด์								
2. ศึกษาวิธีการและเครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์								
3. ออกแบบแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์								
4. พัฒนาวิธีการกรองข้อความสั้นที่ใช้พัฒนาแอปพลิเคชัน								
5. ทดสอบแอปพลิเคชัน								
6. แก้ไขข้อผิดพลาดและปรับปรุงให้สมบูรณ์								

3.4 ขั้นตอนการดำเนินงานวิจัย

3.4.1 รวบรวมข้อมูลและศึกษาการทำงานของระบบปฏิบัติการแอนดรอยด์

1. ประเภทของชุดซอฟต์แวร์ เนื่องจากแอนดรอยด์นั้นเปิดให้นักพัฒนาเข้าไปดูรหัสต้นฉบับได้ ทำให้มีผู้พัฒนาจากหลายฝ่ายนำเอารหัสต้นฉบับมาปรับแต่ง และสร้างแอนดรอยด์ในแบบฉบับของตนเองขึ้น จึงแบ่งประเภทของแอนดรอยด์ออกได้เป็น 3 ประเภท (จักรชัย โสอินทร์ และ พงษ์ศธร จันทร์ช้อย, 2554) ดังต่อไปนี้คือ

1.1) Android Open Source Project (AOSP) เป็นแอนดรอยด์ประเภทแรกที่ถูกเปิดให้สามารถนำ “ต้นฉบับแบบเปิด” ไปติดตั้งและใช้งานในอุปกรณ์ต่าง ๆ ได้โดยไม่ต้องเสียค่าใช้จ่าย

1.2) Open Handset Mobile (OHM) เป็นแอนดรอยด์ที่ได้รับการพัฒนาร่วมกับกลุ่มบริษัทผู้ผลิตอุปกรณ์พกพา ที่เข้าร่วมกับกูเกิลในนาม Open Handset Alliances (OHA) ซึ่งบริษัทเหล่านี้จะพัฒนาแอนดรอยด์ในแบบฉบับของตนเองออกมา โดยรูปร่างหน้าตาการแสดงผล และฟังก์ชันการใช้งาน จะมีความเป็นเอกลักษณ์ และมีลิขสิทธิ์เป็นของตน พร้อมได้รับสิทธิ์ในการมีบริการเสริมต่าง ๆ จากกูเกิล ที่เรียกว่า Google Mobile Service (GMS) ซึ่งเป็นบริการเสริมที่ทำให้แอนดรอยด์มีประสิทธิภาพ เป็นไปตามจุดประสงค์ของแอนดรอยด์ แต่การจะได้มาซึ่ง GMS นั้น ผู้ผลิตจะต้องทำการทดสอบระบบ และขออนุญาตกับทางกูเกิลก่อน จึงจะนำเครื่องออกสู่ตลาดได้

1.3) Cooking หรือ Customize เป็นแอนดรอยด์ที่นักพัฒนานำเอารหัสต้นฉบับจากแหล่งต่าง ๆ มาปรับแต่ง ในแบบฉบับของตนเอง โดยจะต้องทำการปลดล็อคสิทธิ์การใช้งานอุปกรณ์ หรือ Unlock เครื่องก่อน จึงจะสามารถติดตั้งได้ โดยแอนดรอยด์ประเภทนี้ถือเป็นประเภทที่มีความสามารถมากที่สุด เท่าที่อุปกรณ์เครื่องนั้น ๆ จะรองรับได้ เนื่องจากการปรับแต่งให้เข้ากับอุปกรณ์นั้น ๆ จากผู้ใช้งานจริง

2. สิทธิ์ในการใช้งานระบบ เช่นเดียวกับระบบปฏิบัติการทั่วไป ที่มีการจำกัดการใช้งานและการเข้าถึงส่วนต่าง ๆ ภายในระบบ เพื่อความปลอดภัยของระบบ และ ผู้ใช้งาน อุปกรณ์ที่ติดตั้งระบบแอนดรอยด์จึงมีการจำกัดสิทธิ์ไว้ (เว้นแต่ได้ทำการปลดล็อคสิทธิ์ หรือ root เครื่องแล้ว) สามารถแบ่งสิทธิ์ของผู้ใช้ในการเข้าถึงระบบได้ดังต่อไปนี้

2.1) สิทธิ์ root สิทธิ์การใช้ใช้งานระดับราก ซึ่งถือว่าเป็นรากฐานของระบบ จึงมีความสามารถในการเข้าถึงทุก ๆ ส่วนของระบบ

2.2) สิทธิ์ ADB (Android Develop Bridge) นักพัฒนาสามารถเข้าถึงส่วนต่าง ๆ ของระบบได้ผ่านสิทธิ์นี้

2.3) Application & System สิทธิ์ของโปรแกรมในการเข้าถึงระบบ และสิทธิ์ของระบบในการเข้าถึงอุปกรณ์โดยสิทธิ์เหล่านี้ ตัวระบบจะเป็นตัวจัดการมอบและถอนสิทธิ์ ตามเงื่อนไขที่กำหนดซึ่งจะถูกแบ่งย่อยออกเป็นหลายหัวข้อ

ตารางที่ 3.2 การอนุญาตให้เข้าถึงข้อมูลของข้อความสั้น (SMS)

การอนุญาตที่เกี่ยวกับโทรศัพท์	วัตถุประสงค์
android.permission.RECEIVE_SMS	อนุญาตให้แอปพลิเคชันสามารถตรวจสอบข้อความสั้นที่เข้ามาได้
android.permission.READ_SMS	อนุญาตให้แอปพลิเคชันสามารถอ่านข้อความสั้นที่เข้ามาได้
android.permission.SEND_SMS	อนุญาตให้แอปพลิเคชันสามารถส่งข้อความสั้นได้
android.permission.WRITE_SMS	อนุญาตให้สามารถเขียนข้อความสั้นลงบนแอปพลิเคชันที่สร้างขึ้นได้

ที่มา: Android Developer [online] : เข้าถึง 15 ต.ค. 2554. จาก

<http://developer.android.com/reference/android/Manifest.permission.html>

2.4) End-user ผู้ใช้งานขั้นสุดท้าย ที่ใช้การเข้าถึงส่วนต่าง ๆ ของระบบผ่านช่องทางสิทธิ์ที่โปรแกรมได้รับอีกที โดยจะถูกจำกัดไม่ให้เข้าถึงในส่วนที่เป็นอันตรายต่อแกนระบบและอุปกรณ์

3.4.2 ศึกษาเครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ (จักรชัย โสอินทร์ และ พงษ์ศธร จันทร์ยอย, 2554)

1. Java Development Kit (JDK) เป็นชุดคำสั่งในการพัฒนาโปรแกรมด้วยภาษาจาวา โดยชุดพัฒนาโปรแกรม JDK ประกอบด้วย 3 รุ่นดังนี้

1.1) Java SE (Standard Edition) สำหรับพัฒนาโปรแกรมบนเครื่องคอมพิวเตอร์ทั่วไป

1.2) Java ME (Micro Edition) สำหรับพัฒนาโปรแกรมบนอุปกรณ์พกพา เช่น โทรศัพท์มือถือ หรือ พีดีเอ ส่วนมากใช้สำหรับเขียนโปรแกรมเกมส์

1.3) Java EE (Enterprise Edition) สำหรับพัฒนาโปรแกรมในองค์กรใหญ่ ๆ หรือมีขอบเขตของโครงการที่กว้างมาก ชุดพัฒนาโปรแกรม JDK นั้น เป็นมาตรฐานที่ถูกกำหนดโดยบริษัท ซัน ไมโครซิสเต็มส์ และมีผู้พัฒนาชุดโปรแกรม JDK ออกมามากมาย เช่น ซัน ไมโครซิสเต็มส์ ไอบีเอ็ม และ บีอีเอ ซิสเต็มส์

2. โปรแกรมที่ใช้สำหรับเขียนโปรแกรมภาษาจาวา (Java)(Eclipse) เป็นเครื่องมือที่ช่วยสนับสนุนในการพัฒนาซอฟต์แวร์ โดยเฉพาะภาษาจาวา และที่สำคัญ eclipse เป็นซอฟต์แวร์แบบเปิด (open source) ที่พัฒนาขึ้นเพื่อใช้โดยนักพัฒนาเอง ทำให้ความก้าวหน้าในการพัฒนาของ eclipse เป็นไปอย่างต่อเนื่องและรวดเร็ว Eclipse มีองค์ประกอบหลักที่เรียกว่า Eclipse Platform ซึ่งให้บริการขั้นพื้นฐานหลักสำหรับรวบรวมเครื่องมือต่าง ๆ จากภายนอกให้สามารถเข้ามาทำงานร่วมกันในสภาพแวดล้อมเดียวกัน และมีส่วนประกอบที่เรียกว่า plug-in development (PDE) ซึ่งใช้ในการเพิ่มความสามารถในการพัฒนาซอฟต์แวร์มากขึ้น เครื่องมือภายนอกจะถูกพัฒนาในรูปแบบที่เรียกว่า Eclipse plug-ins ดังนั้นหากต้องการให้ Eclipse ทำงานใดเพิ่มเติม ก็เพียงแค่พัฒนา plug-in สำหรับงานนั้นขึ้นมา และนำ plug-in นั้นมาติดตั้งเพิ่มเติมให้กับ Eclipse ที่มีอยู่เท่านั้น Eclipse plug-in ที่มาพร้อมกัน เมื่อมีการดาวน์โหลดมาครั้งแรกก็คือองค์ประกอบที่เรียกว่า Java Development Toolkit (JDT) ซึ่งเป็นเครื่องมือในการเขียนและ debug โปรแกรมภาษาจาวา

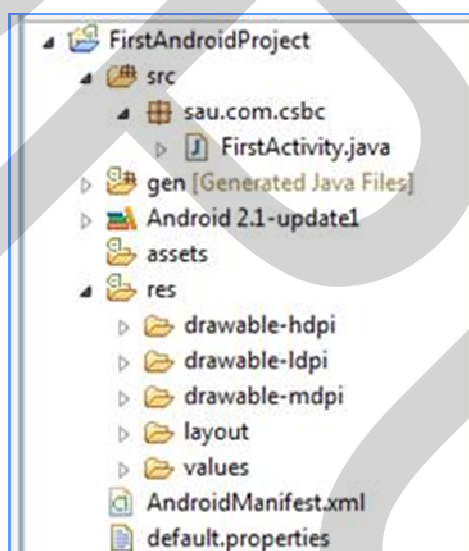
3. Android Development Tool (ADT) โปรแกรมเสริมที่ทำให้พัฒนาแอนดรอยด์ บน Eclipse ได้ ทำหน้าที่เป็นตัวเชื่อมระหว่าง Android SDK และ Eclipse ให้สามารถทำงานเสมือนเป็นเครื่องมือขึ้นเดียวกัน หากใช้ Eclipse แล้วไม่เอา ADT ไปด้วยก็เหมือนกับไม่ได้ใช้ความสามารถ

ของ Eclipse เลย โดยสามารถดาวน์โหลดได้จาก http://developer.android.com/sdk/adt_download.html

4. Android SDK ย่อมาจาก Android Software Development Kit ซึ่งเป็นชุดโปรแกรมที่ ถูกพัฒนาออกมาเพื่อแจกจ่ายให้นักพัฒนาแอปพลิเคชัน หรือผู้สนใจไปดาวน์โหลดเพื่อนำมาใช้ กัน โดยไม่มีค่าใช้จ่าย ซึ่งในชุด SDK นั้นจะมีโปรแกรมและไลบรารีต่าง ๆ ที่จำเป็นต่อการพัฒนา แอปพลิเคชันบนแอนดรอยด์ อย่างเช่น Emulator ซึ่งทำให้ผู้ใช้สามารถสร้างแอปพลิเคชันและนำมา ทดลองรันบนอิมูเลเตอร์ก่อน โดยมีสภาพแวดล้อมเหมือนมือถือที่รันอยู่บนระบบปฏิบัติการแอนดรอยด์จริง ๆ

3.4.3 ออกแบบและพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์

1. โครงสร้างพื้นฐานของแอนดรอยด์ (android project structure)



ภาพที่ 3.1 โครงสร้างพื้นฐานของโปรแกรมนักพัฒนา

1.1) src (Source Code) เป็นส่วนของซอร์สโค้ดที่สร้างขึ้นและเขียนขึ้นเป็นไฟล์ .java โดยจะมีทั้งชื่อคลาส แอตทริบิวต์ และเมธอด

1.2) gen (Generated Java Files) เป็นส่วนที่ถูกสร้างขึ้นโดยอัตโนมัติ เช่นไฟล์ R.java ซึ่งภายในประกอบด้วย text และ UI Element โดยถูกนำไปยังโปรเจกต์ผ่าน android plug-in ซึ่งไฟล์นี้จะเป็นเหมือนตัวชี้ (pointer) ไปยัง drawable, layout, values directories

1.3) res (Resource) จะเป็นส่วนของการแสดงผลของไฟล์อื่น ๆ ที่นำมาเป็นส่วนประกอบ ร่วมกับโค้ดที่เขียนขึ้น โดยแอนดรอยด์จะมีการแบ่งไฟล์ตามชนิดของ resource นั้น โดยที่เมื่อมีการสร้าง โปรเจกต์ขึ้นมาใหม่จะมี directory ย่อย ๆ อีก 5 directory คือ

1.4) drawable-hdpi/drawable-ldpi/drawable-mdpi ซึ่งใช้สำหรับเก็บภาพ

1.5) layout ใช้สำหรับจัดวาง view ต่าง ๆ

1.6) values ใช้สำหรับจัดการค่าต่าง ๆ ที่นำไปใช้ในแอปพลิเคชัน ทั้งนี้ยังสามารถสร้าง directory ต่าง ๆ ขึ้นเองเพื่อเก็บข้อมูลอื่น ๆ ที่จะใช้ในแอปพลิเคชันได้ด้วย

1.7) AndroidManifest.xml เป็นโครงสร้างของ xml ไฟล์ ซึ่งใน xml นั้นจะเป็นการกำหนดคุณสมบัติ และการตั้งค่าต่าง ๆ ของแอปพลิเคชัน เช่น ชื่อของแอปพลิเคชัน เวอร์ชันของโค้ดการกำหนดสิทธิ์ ต่าง ๆ ในการเข้าถึงแอปพลิเคชัน

2. องค์ประกอบของแอนดรอยด์แอปพลิเคชัน (Android Application Components) (พร้อมเลิส หล่อวิจิตร, 2555)

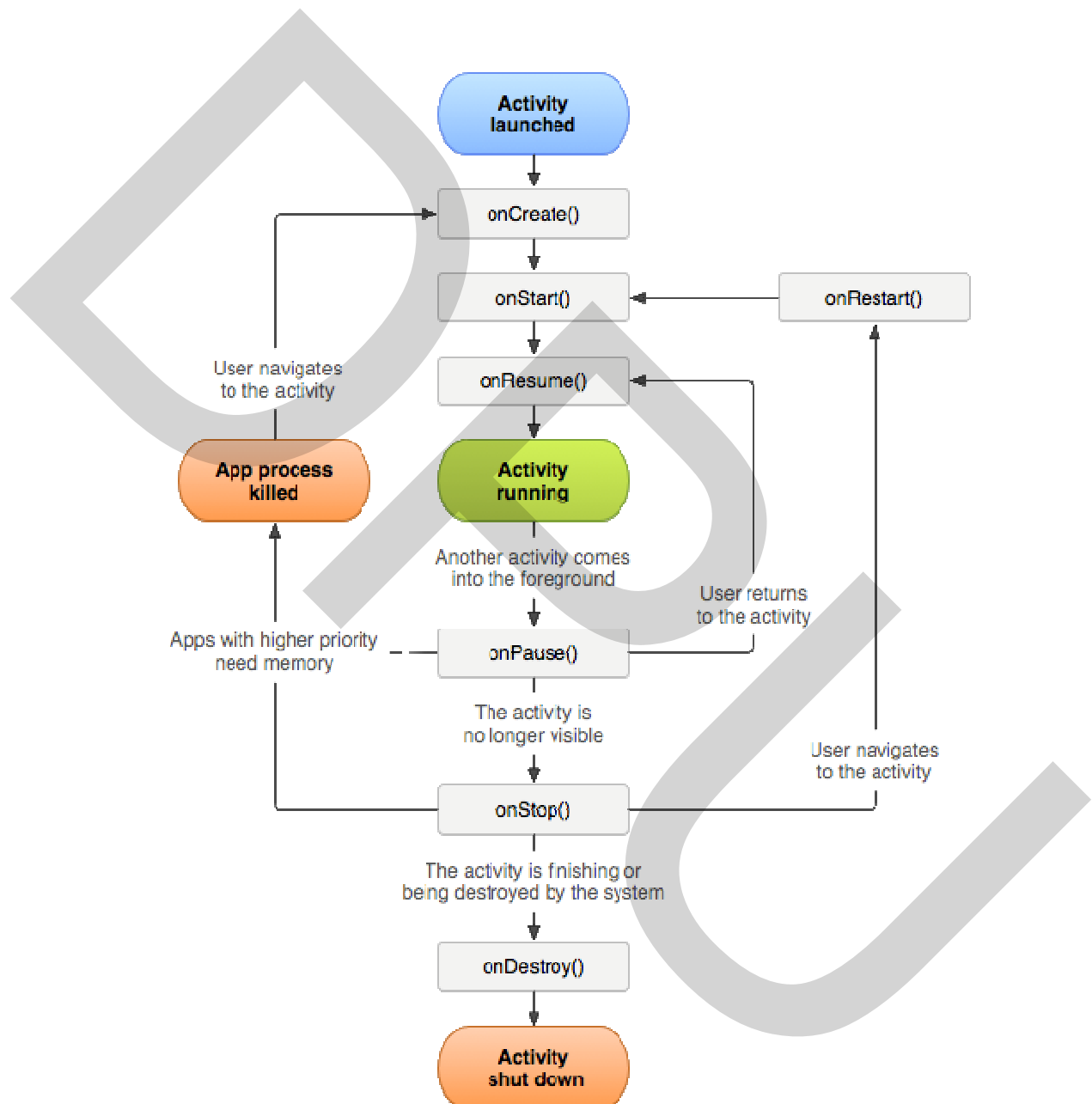
2.1) Activity (User Interface) คือสิ่งที่ใช้ในการแสดงผล เพื่อให้ผู้ใช้งานได้เห็นและได้ใช้งาน โดยในแต่ละแอปพลิเคชันนั้น อาจมี activity เดียว หรือหลาย activity ก็ได้ และสิ่งที่อยู่ใน activity นั้นจะเรียกว่า view ซึ่งก็มีอยู่ในหลาย ๆ รูปแบบ เช่น buttons, text, fields, scroll bars, menu items, check boxes และอื่น ๆ อีกมากมาย

2.2) Service (Service Provider) คือสิ่งที่ไม่มีส่วนของการแสดงผล แต่ถูกเรียกให้รันในลักษณะของ background process โดย service นั้น อาจมีการกระทำอะไรบางอย่าง เช่น ติดต่อรับส่งข้อมูลผ่านเครือข่าย หรือคำนวณค่าต่าง ๆ แล้วก็ส่งผลลัพธ์นั้นไปแสดงยัง activity ก็ได้

2.3) Broadcast receiver (Data Provider) คือตัวที่ใช้สำหรับคอยรับ และตอบสนองต่อเหตุการณ์ต่าง ๆ ที่เกิดขึ้น เช่น เมื่อแบตเตอรี่ต่ำ ผู้ใช้ทำการเปลี่ยนภาษา มีการโทรออก มีข้อความเข้า และอื่น ๆ ถึงแม้ broadcast receiver จะไม่มีส่วนของการแสดงผล แต่ก็สามารถเรียก activity ขึ้นมาแสดงผลให้ผู้ใช้งานได้ หรืออาจจะใช้สิ่งที่เรียกว่า Notification Manager ซึ่งจะเป็นตัวที่แจ้งเตือนในรูปแบบของการสั่น การแสดงไฟกระพริบที่หน้าจอ หรือการส่งเสียงออกมาจากลำโพง โดยจะมีไอคอน แสดงอยู่บน status bar เพื่อให้ผู้ใช้เข้าไปกดเพื่อดูเหตุการณ์ที่เกิดขึ้น

2.4) Content Provider (System Event Listener) คือกลุ่มของข้อมูลที่สร้างขึ้นจากแอปพลิเคชัน เพื่อให้แอปพลิเคชันอื่น ๆ ได้นำไปใช้ โดยการจัดเก็บข้อมูลของ content provider นั้นจะอยู่ในลักษณะของไฟล์ ฐานข้อมูล SQLite และอื่น ๆ ตัวอย่างแอปพลิเคชันที่ใช้งาน Content Provider ที่เห็นเด่นชัดที่สุดคือ โปรแกรม Contacts ที่แสดงรายชื่อใน contact นั้นเอง

3. Activity Life Cycle มีหน้าที่ในการสร้างและควบคุมการทำงานของหน้าจอ เพื่อโต้ตอบกับผู้ใช้งาน ซึ่งจะมีแค่ activity เดียวเท่านั้นที่โต้ตอบกับผู้ใช้งานในขณะใดขณะหนึ่ง โดยแต่ละ activity แต่ละตัวจะมีวงจรชีวิตเป็นของตนเอง โดยแบ่งเป็นสถานะดังนี้



ภาพที่ 3.2 สถานะของ Activity

ที่มา: Activity Lifecycle [online] : เข้าถึง 5 ต.ค. 2554. จาก

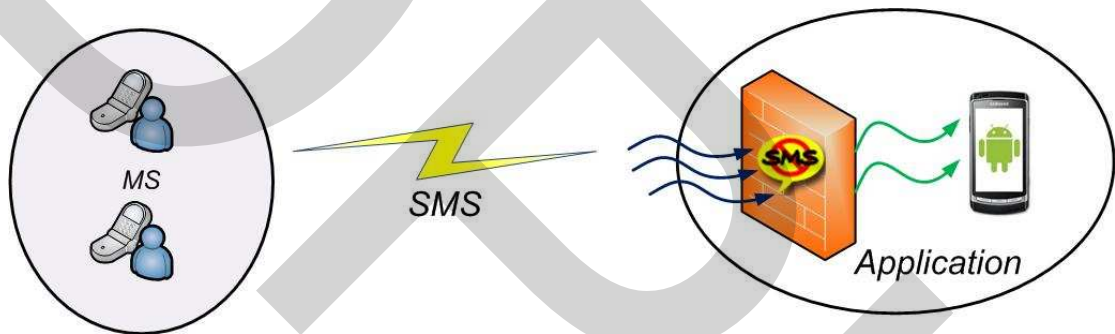
<http://developer.android.com/reference/android/app/Activity.html>

- 3.1) Running เป็นสถานะที่ activity กำลังทำงานในขณะนั้น
- 3.2) Pause เป็นสถานะที่ activity ปรากฏอยู่บนหน้าจอแต่ไม่ได้ทำงานในขณะนั้น
- 3.3) เป็นสถานะที่ activity ไม่ปรากฏบนหน้าจอ
- 3.4) Kill เป็นสถานะที่ activity ถูกทำลายไปแล้วเมื่อ activity อยู่ในสถานะ pause หรือ stop activity สามารถ ถูกทำลายได้โดยระบบ
- 3.5) onCreate(Bundle) จะถูกเรียกเมื่อ activity มีการสร้างตัวเองครั้งแรก เป็นเมธอดสำคัญในการสร้างหน้าจอการทำงานต่าง ๆ
- 3.6) onStart() จะถูกเรียกหลังจากที่ activity กลับมาจากสถานะ stop เพื่อเริ่มทำงานใหม่อีกครั้ง
- 3.7) onResume() จะถูกเรียกเมื่อ activity เริ่มแสดงให้ผู้ใช้งานเห็น
- 3.8) onPause() จะถูกเรียกเมื่อ activity สามารถโต้ตอบกับผู้ใช้งานได้
- 3.9) onStop() จะถูกเรียกเมื่อมี activity อื่นกำลังทำงานเป็นเมธอด ในการเก็บข้อมูลที่สำคัญหรือทำการปล่อยหน่วยความจำเพื่อให้ activity อื่น ได้ใช้งาน
- 3.10) onDestroy() จะถูกเรียกเมื่อ activity ไม่ปรากฏบนหน้าจอ แล้วจะเข้าสู่สถานะ stop
- 3.11) onSaveInstanceState(Bundle) จะถูกเรียกก่อน onPause() เพื่อเก็บสถานะต่าง ๆ ของ activity ไว้ใช้เมื่อ activity นี้ได้กลับมาทำงานอีกครั้ง
- 3.12) onRestoreInstanceState(Bundle) จะถูกเรียกก่อน onResume() เพื่อนำสถานะต่าง ๆ ของ activity ที่เก็บไว้มาใช้งาน

บทที่ 4

โปรแกรมกรองข้อความสั้นบนโทรศัพท์เคลื่อนที่แอนดรอยด์ โดยอาศัยข้อมูลในสมุดโทรศัพท์

4.1 ภาพรวมการทำงานของระบบ



ภาพที่ 4.1 โครงสร้างของโปรแกรมกรองข้อความสั้นบนโทรศัพท์เคลื่อนที่แอนดรอยด์

การทำงานของระบบนี้ได้แบ่งออกเป็น 2 ส่วน ดังภาพที่ 4.1 ซึ่งแต่ละส่วนมีรายละเอียดดังต่อไปนี้

4.1.1 โทรศัพท์เคลื่อนที่ทั่วไป (MS) ทำหน้าที่ส่งข้อความสั้น เพื่อเข้ามายังโทรศัพท์เคลื่อนที่แอนดรอยด์ที่ได้ติดตั้งแอปพลิเคชันการกรองข้อความสั้น โดยอาศัยข้อมูลในสมุดโทรศัพท์

4.1.2 แอปพลิเคชัน ทำหน้าที่กรองข้อความสั้นที่เข้ามา โดยอาศัยข้อมูลในสมุดโทรศัพท์เป็นตัวกรอง

4.2 ข้อกำหนดของโปรแกรมกรองข้อความสั้น

4.2.1 เมื่อมีข้อความสั้นที่ถูกส่งมาจากโทรศัพท์เคลื่อนที่ทั่วไปเข้ามา แอปพลิเคชันจะทำการตรวจสอบว่าเลขหมายที่ส่งข้อความสั้นเข้ามานั้น มีการบันทึกชื่ออยู่ในสมุดโทรศัพท์หรือไม่ หากมีการบันทึก ชื่อข้อความสั้นนั้นก็จะเข้ากล่องข้อความเข้าของตัวเครื่องปกติ แต่หากไม่มีชื่อบันทึกอยู่ในสมุดโทรศัพท์ ข้อความสั้นนั้นก็จะถูกเก็บไว้ในแอปพลิเคชัน

4.2.2 การทำงานภายในแอปพลิเคชัน จะเริ่มตั้งแต่เมื่อมีการเปิดแอปพลิเคชัน โดยทำการอ่านข้อมูลจากฐานข้อมูลทั้งหมด แล้วนำมาแสดงในรายการ (list) ซึ่งจะเป็นข้อความสั้นทั้งหมดที่แอปพลิเคชันได้กรองไว้ โดยสามารถเลือกเข้าไปยังแต่ละรายการ เพื่อดูข้อความสั้นที่กรองไว้ของแต่ละเลขหมาย โดยในแต่ละรายการจะสามารถเพิ่มเลขหมายและวันที่กรายชื่อลงในฐานข้อมูลของเครื่องโทรศัพท์ได้ ซึ่งจะเป็นการปรับปรุงสมุดโทรศัพท์ในตัวเครื่อง ทั้งนี้ยังสามารถลบข้อความสั้นที่อยู่ในแอปพลิเคชันได้ด้วย หากไม่ต้องการเก็บข้อความสั้นนั้นไว้

4.3 ขั้นตอนการทำงานของโปรแกรมกรองข้อความสั้น

การทำงานจะเริ่มต้นขึ้น เมื่อโปรแกรมได้ถูกติดตั้ง (Installed) ลงบนโทรศัพท์เคลื่อนที่แอนดรอยด์ และจะทำงานอัตโนมัติเมื่อมีข้อความใหม่เข้ามาที่เครื่องโทรศัพท์ โดยอาศัยกระบวนการและขั้นตอนการทำงานดังนี้

4.3.1 ทำการลงทะเบียนเพื่อเปิดบริการกรองข้อความสั้น สำหรับการพัฒนาแอปพลิเคชันนั้น ๆ จะต้องมีการตั้งค่าการอนุญาต (Permission) เป็นการให้สิทธิต่าง ๆ กับแอปพลิเคชันตามตารางที่ 4.1 ซึ่งจะทำบนไฟล์ AndroidManifest.xml

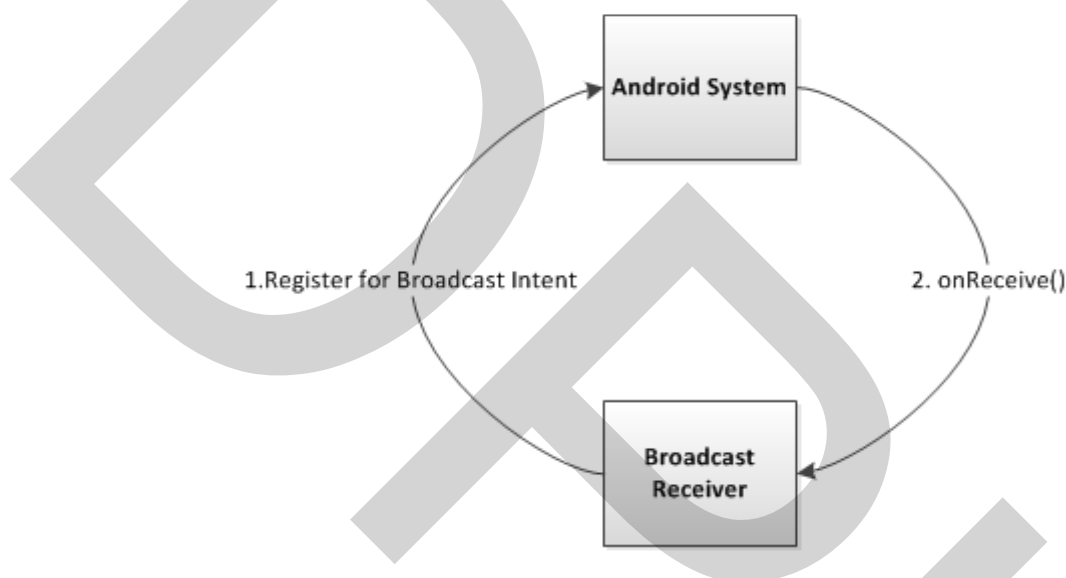
ตารางที่ 4.1 การอนุญาตให้เข้าถึงข้อมูลของข้อความสั้น (SMS)

การอนุญาตที่เกี่ยวกับโทรศัพท์	วัตถุประสงค์
android.permission.RECEIVE_SMS	อนุญาตให้แอปพลิเคชันสามารถตรวจสอบข้อความสั้นที่เข้ามาได้
android.permission.READ_SMS	อนุญาตให้แอปพลิเคชันสามารถอ่านข้อความสั้นที่เข้ามาได้
android.permission.SEND_SMS	อนุญาตให้แอปพลิเคชันสามารถส่งข้อความสั้นได้
android.permission.WRITE_SMS	อนุญาตให้สามารถเขียนข้อความสั้นลงบนแอปพลิเคชันที่สร้างขึ้นได้

ที่มา: Android Developer [online] : เข้าถึง 15 ต.ค. 2554. จาก

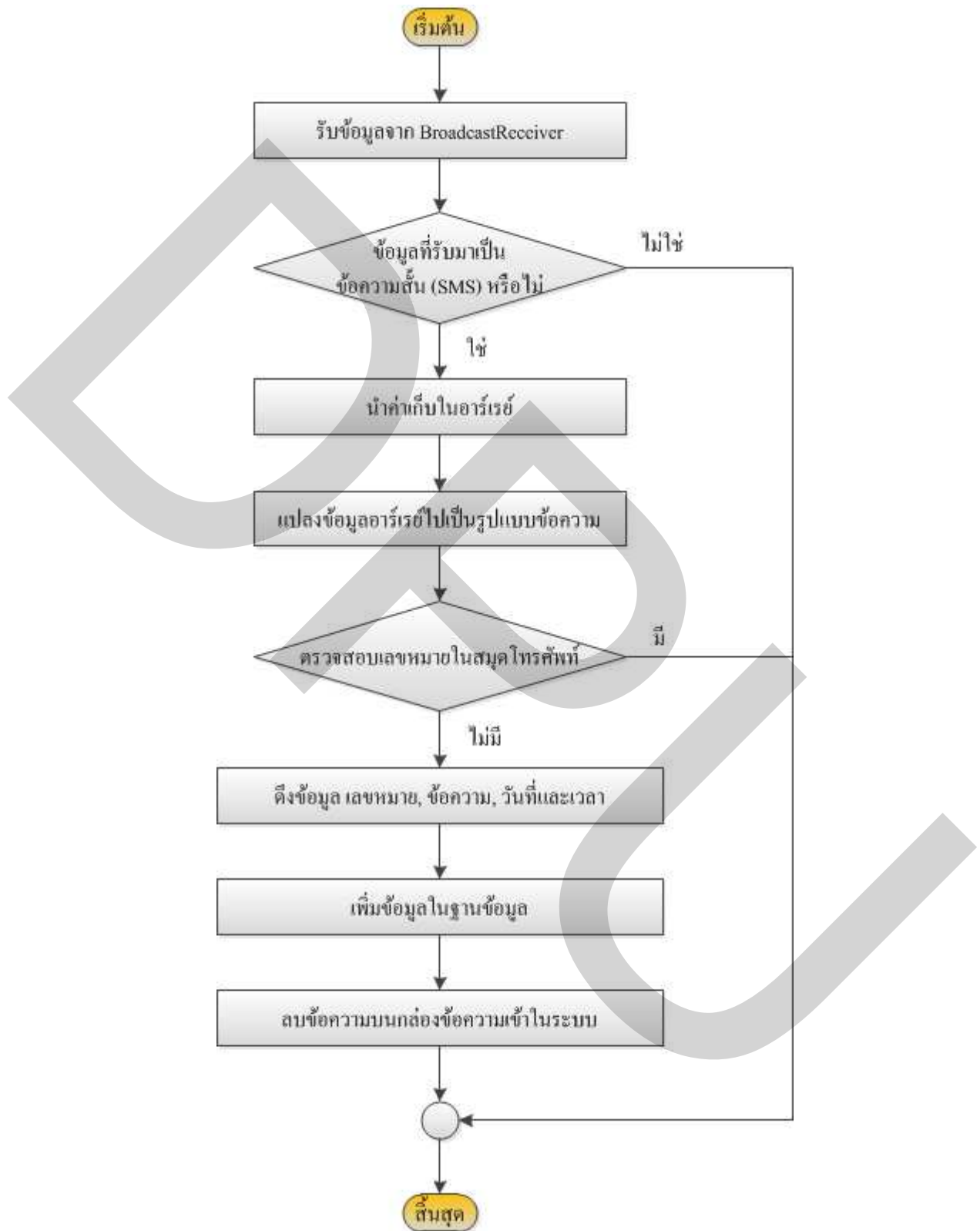
<http://developer.android.com/reference/android/Manifest.permission.html>

4.3.2 การใช้งานข้อความสั้น (SMS) ในเฟรมเวิร์คของระบบปฏิบัติการแอนดรอยด์จะมีไลบรารีที่เกี่ยวข้องกับการใช้บริการข้อความสั้น คือคลาส SmsManager จะอยู่ในแพ็คเกจ android.telephony โดยจะรองรับทั้งระบบ GSM และระบบ CDMA รวมถึงการประกาศ Broadcast Receiver เพื่อตรวจจับข้อความสั้นที่ส่งเข้ามา โดยมีวงจรชีวิตของ Broadcast Receiver ตามภาพที่ 4.2



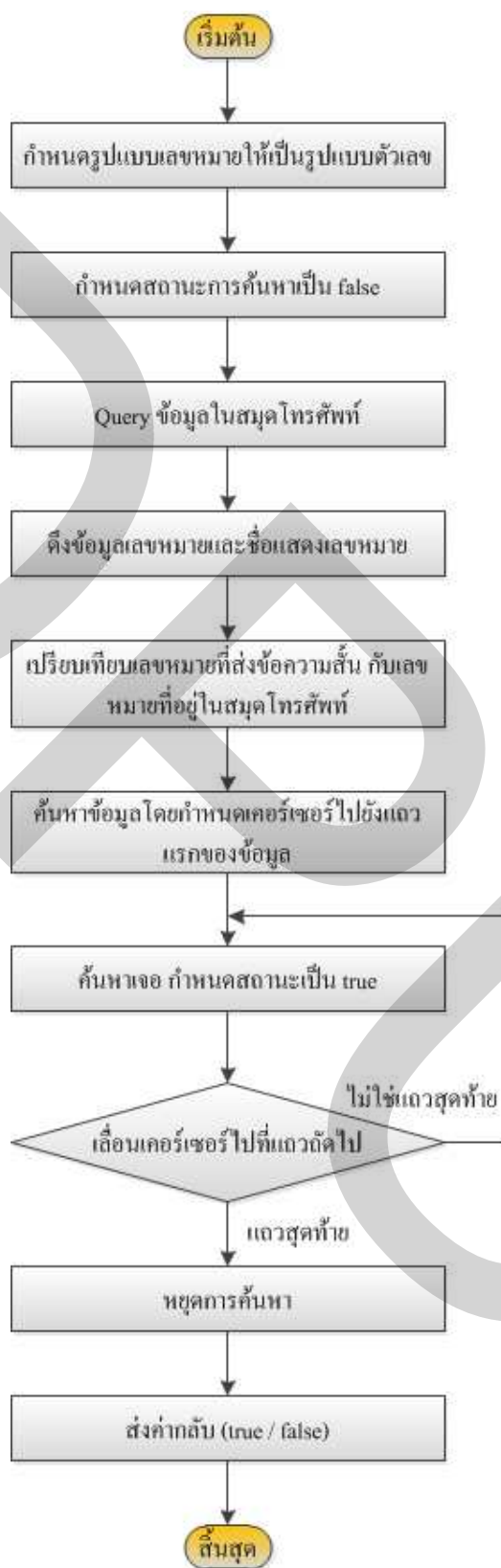
ภาพที่ 4.2 วงจรชีวิตของ Broadcast Receiver

4.3.3 ฟังก์ชัน onReceive() จะถูกเรียกใช้งานเมื่อมีการรับข้อมูลเข้ามา โดยทำการตรวจสอบว่าข้อมูลที่รับมานั้นเป็นข้อมูลประเภทข้อความสั้นหรือไม่ ในการตรวจจับข้อความสั้นที่ถูกส่งเข้ามานั้นจะใช้ค่าของ PDU (Protocol Description Unit) ซึ่งประกอบด้วย ข้อความสั้น ค่าต่าง ๆ ที่ใช้ในการส่งโดยเก็บไว้ในอาร์เรย์ออบเจกต์ (object) และเมธอด createFromPdu() ซึ่งจะใช้แปลงข้อมูลอาร์เรย์ออบเจกต์ ไปเป็น SmsMessage และใช้คำสั่ง getOriginatingAddress() เพื่ออ่านค่าเลขหมายโทรศัพท์ของผู้ส่ง และใช้คำสั่ง getMessageBody() เพื่ออ่านค่าข้อความ หลังจากนั้นจะนำค่าเลขหมายโทรศัพท์ ไปตรวจสอบกับเมธอด isHaveContact() บนคลาส ContactPhone เพื่อเปรียบเทียบเลขหมายระหว่างเลขหมายของข้อความสั้นใหม่ และเลขหมายบนสมุดโทรศัพท์ของระบบ ซึ่งหากตรวจสอบแล้วไม่มี ก็จะนำค่าทั้งเลขหมาย ข้อความ วันที่และเวลา ไปเพิ่มในฐานข้อมูลของแอปพลิเคชัน และเข้าไปลบข้อความสั้นนั้นบนกล่องข้อความเข้าของระบบ โดยใช้เมธอด deleteSMS() บนคลาส SMSReceive ตามภาพที่ 4.3

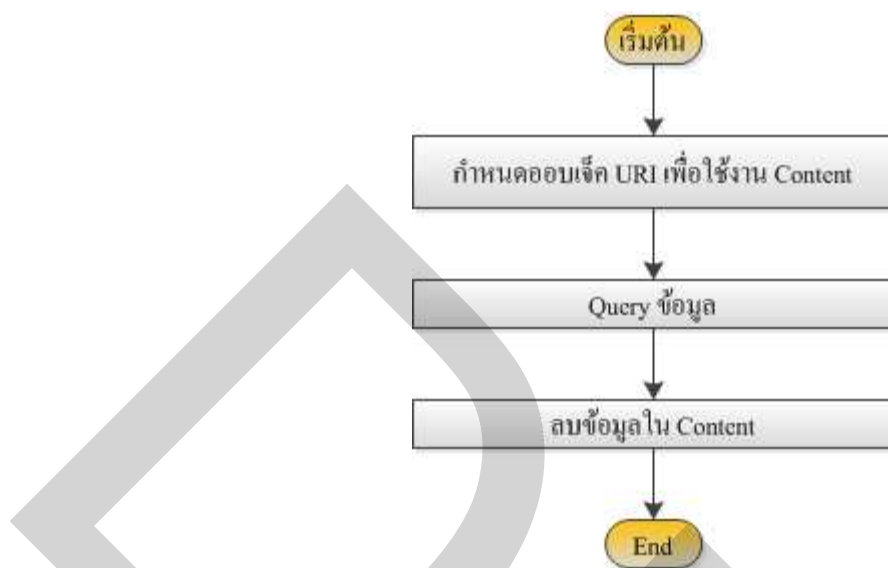


ภาพที่ 4.3 ฟังก์ชันการรับและตรวจสอบข้อความสั้น

สำหรับการตรวจสอบเลขหมายในสมุดโทรศัพท์ ตามภาพที่ 4.4 ได้แยกย่อยการทำงานออกมา โดยสร้างคลาส `isHaveContact()` เป็นตัวตรวจสอบเลขหมายของข้อความสั้นนั้น ๆ กับเลขหมายหรือรายชื่อบนสมุดโทรศัพท์ของระบบ โดยใช้รูปแบบ `PhoneNumberUtils.formatNumber` เพื่อเรียกเครื่องมือสำหรับจัดการสมุดโทรศัพท์และกำหนดรูปแบบของตัวเลข โดยกำหนดสถานะเริ่มต้นเป็น `false` หลังจากนั้นจะใช้เคอร์เซอร์ ในการสืบค้น (query) ข้อมูล โดยกำหนด `Phone.NUMBER` เป็นหมายเลขโทรศัพท์ และ `Phone.DISPLAY_NAME` เป็นชื่อที่แสดงแทนเลขหมายนั้น ๆ โดยในการสืบค้นนั้นจะย้ายเคอร์เซอร์ไปยังแถวแรกของข้อมูลแล้วเปรียบเทียบข้อมูล หากตรงกันจะส่งค่ากลับเป็น `true` และย้ายเคอร์เซอร์ไปยังตำแหน่งถัดไป ซึ่งหากแถวนั้นเป็นแถวสุดท้าย ก็จะหยุดการค้นหา และจะส่งค่ากลับของสถานะการค้นหา และเมื่อได้ทำการเพิ่มข้อมูลลงในฐานข้อมูลแล้ว จะต้องให้แอปพลิเคชันเข้าไปลบข้อความสั้นนั้นบนกล่องข้อความเข้าของระบบ โดยใช้เมธอด `deleteSMS()` บนคลาส `SMSReceive` โดยจะเข้าไปทำงานอัตโนมัติ ผ่านการเรียกใช้ URI สำหรับการเข้าถึงข้อมูลใน Content Provider ("`content://sms/inbox`") และจะใช้คำสั่ง query ใน URI โดยมีค่า `ID`, `thread_id`, `date`, `body` และจะเข้าไปลบในส่วนของ `conversations` โดยอ้างอิงจากเลขหมายนั้น ๆ ตามภาพที่ 4.5



ภาพที่ 4.4 ฟังก์ชันการตรวจสอบบนสมุดโทรศัพท์ของระบบ



ภาพที่ 4.5 ฟังงานการลบข้อความต้นบนกล่องข้อความของระบบ

2.3.4 การใช้งานฐานข้อมูล SQLite (เจมส์ สตีล และ เนลสัน โต, 2554) โดยในระบบปฏิบัติการแอนดรอยด์นั้น มีการติดตั้งระบบฐานข้อมูล SQLite มาด้วยแล้ว ซึ่ง SQLite มีลักษณะเป็นฐานข้อมูลเชิงสัมพันธ์ โดยสามารถใช้คำสั่ง SQL ในการสืบค้นข้อมูลได้ แอปพลิเคชันที่ใช้งาน SQLite นั้นจะมีอินสแตนซ์ของฐานข้อมูลเป็นของตัวเอง และสามารถเข้าถึงได้จากแอปพลิเคชันเท่านั้น ในส่วนของแอปพลิเคชันนี้ มีการใช้งาน SQLite ทางด้าน สร้างฐานข้อมูล เพิ่มข้อมูล เรียกดูข้อมูล ลบข้อมูล โดยมีรายละเอียดดังนี้

4.3.4.1 การสร้างฐานข้อมูล มีการประกาศตัวแปร เพื่อใช้เก็บตารางข้อมูล โดยระบุชื่อตารางฐานข้อมูลและประกาศตัวแปรชื่อของฐานข้อมูล โดยระบุชื่อไฟล์ของฐานข้อมูล โดยทำการสร้างฐานข้อมูลภายใต้ชื่อฐานข้อมูลที่ได้ประกาศไว้ โดยใช้รูปแบบคำสั่งตามภาพที่ 4.6

```
public abstract SQLiteDatabase openOrCreateDatabase (String name, int mode, SQLiteDatabase.CursorFactory factory)
```

ภาพที่ 4.6 คำสั่ง openOrCreateDatabase

ในการสร้างฐานข้อมูลจะใช้คำสั่ง execSQL เพื่อสร้างตารางโดยกำหนดชื่อตาราง และฟิลด์ของข้อมูลตามภาพที่ 4.7

Table: table_data

	MSG ID	PHONE NUMBER	MESSAGE
1			
2			
3			

ภาพที่ 4.7 การสร้างตารางฐานข้อมูลและการบันทึกข้อมูล

4.3.4.2 การเพิ่มข้อความสั้นลงในฐานข้อมูล จะใช้คำสั่ง ContentValues.put เพื่อนำค่าเข้าไปยังฟิลด์ของข้อมูล MSG_ID, PHONE_NUMBER, MESSAGE ที่สร้างไว้ และใช้คำสั่ง insert() โดยการอ้างอิงตารางข้อมูล และนำค่าข้อมูลที่อยู่ใน ContentValues ไปเพิ่มข้อมูลลงในฐานข้อมูล ตามภาพที่ 4.8

Table: table_data

	MSG ID	PHONE NUMBER	MESSAGE
1	1337872145909	12345678	Test Send 3
2	1337873525138	0864201039	test from not save 3
3	1337912009223	+66864201035	Test SimpleSpamSMS
4	1337912097996	9999	Activate Data Suces

ภาพที่ 4.8 การเพิ่มข้อมูลลงในฐานข้อมูล

4.3.4.3 การเรียกดูข้อมูล (getDataSMS()) เพื่อแสดงข้อมูลเลขหมาย จำนวนข้อความสั้นต่อเลขหมาย และข้อความสั้น โดยใช้คำสั่ง openOrCreateDatabase() เพื่อเข้าถึงฐานข้อมูล และใช้คำสั่ง SQLiteDatabase.rawQuery() ด้วย MSG_ID, PHONE_NUMBER, MESSAGE บนตารางข้อมูล ด้วยกลุ่มของ PHONE_NUMBER และเรียกใช้คลาส getSizeDatailSMS เพื่อตรวจสอบและนับจำนวนข้อความสั้นต่อเลขหมายนั้น ๆ

4.3.4.4 การเรียกดูรายละเอียดของข้อมูล (getDataDetailSMS()) เพื่อแสดงข้อมูลข้อความสั้น โดยใช้คำสั่ง openOrCreateDatabase() เพื่อเข้าถึงฐานข้อมูล และใช้คำสั่ง SQLiteDatabase.rawQuery() ด้วย MSG_ID, PHONE_NUMBER, MESSAGE บนตารางข้อมูล ด้วย PHONE_NUMBER

4.3.4.5 การลบข้อความ จะเป็นการลบข้อความ 1 ข้อความ บนเลขหมายนั้น ๆ โดยใช้คำสั่ง SQLiteDatabase.delete() ที่ตารางของฐานข้อมูล ด้วยรหัส MSG_ID ในการอ้างอิงเพื่อลบ

4.3.4.6 การลบข้อความทั้งหมดต่อเลขหมายนั้น ๆ โดยใช้คำสั่ง SQLiteDatabase.delete() ที่ตารางของฐานข้อมูล ด้วย PHONE_NUMBER ในการอ้างอิงเพื่อลบข้อความทั้งหมดบนเลขหมายนั้น ๆ

4.3.5 การกำหนดการทำงานของแอกทิวิตี้ต่าง ๆ บนคลาส MainActivity โดยมีการกำหนดแอกทิวิตี้ที่เกี่ยวข้องกับคลาส และมีการทำงานของเมธอดต่าง ๆ ดังนี้

4.3.5.1 เมธอด onCreate() จะมีการเรียกใช้เมื่อแอกทิวิตี้เริ่มทำงานและจะมีการเรียกใช้เลเอาท์โดยการนำไฟล์ main.xml มาวาดเป็นหน้าจอของแอปพลิเคชัน และมีการสร้างออบเจกต์ชื่อ smsReceive บนคลาส SMSReceive(), ออบเจกต์ชื่อ db บนคลาส DataBase(), ออบเจกต์ชื่อ conP บนคลาส ContactPhone() และมีการเรียกใช้เมธอด isPhoneContact() บนคลาส ContactPhone() และทำการหารหัส (id) จากไฟล์ list_sms.xml ที่ผ่านกระบวนการ (process) บนเมธอด onCreate() รวมถึงการเรียกค่าจากเมธอด getDataSMS() บนคลาส DataBase มาเก็บบน vData ตามภาพที่ 4.9

```
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);
    smsReceive = new SMSReceive();
    db = new DataBase(this);
    ContactPhone conP = new ContactPhone(this);
    conP.isPhoneContact();

    list_sms = (List)findViewById(R.id.list_sms);
    list_sms.setOnItemClickListener(this);
    list_sms.setOnItemLongClickListener(this);
    vData = db.getDataSMS();
    if(vData.size() != 0) list_sms.setAdapter(vData);
}
```

ภาพที่ 4.9 การทำงานของเมธอด onCreate()

4.3.5.2 เมธอด onItemClick() จะเป็นการเรียกกลับเมธอดเมื่อมีการเลือกรายการใน AdapterView โดยจะเชื่อมแอกทิวิตี้ต่าง ๆ และส่งผ่านข้อมูลกันระหว่างแอกทิวิตี้โดยใช้ Intent โดยทำหลังจากมีการเลือก แล้วจึงไปทำที่คลาส DetailActivity และสั่งเริ่มการทำงาน ตามภาพที่ 4.10

```
public void onItemClick(AdapterView<?> AV, View v, final int position, long arg3) {
    DataElement dataE = vData.elementAt(position);
    Intent intent = new Intent(this,DetailActivity.class);
    intent.putExtra("PHONE_NUMBER",dataE.getPhoneNumber());
    startActivity(intent);
}
```

ภาพที่ 4.10 การทำงานของเมธอด onItemClick()

4.3.5.3 เมธอด onItemClick() จะเป็นการเรียกกลับเมธอดเมื่อมีการเลือกโดยการกดค้างไว้บนรายการใน AdapterView และจะกำหนด view ที่ใช้งาน Context menu ด้วย

```
public boolean onItemClick(AdapterView<?> AV, View v, final int position, long arg3) {
    registerForContextMenu(List_sms);
    return false;
}
```

ภาพที่ 4.11 การทำงานของเมธอด onItemClick()

4.3.5.4 เมธอด onCreateContextMenu() จะเป็นการสร้างเมนูหลัก เมื่อมีการสัมผัสหน้าจอ โดยมีการสร้างเมนูขึ้นมา 2 เมนูคือ เมนู “Add Contact” เพื่อเพิ่มรายชื่อลงบนสมุดโทรศัพท์บนโทรศัพท์เคลื่อนที่ โดยมีรหัส (id) = 0 และเมนู “Delete Contact” เพื่อลบเลขหมายนี้ออกจากฐานข้อมูลบนแอปพลิเคชัน โดยมีรหัส (id) = 1 ตามภาพที่ 4.12

```
public void onCreateContextMenu(ContextMenu menu, View v, ContextMenuInfo menuInfo)
{
    super.onCreateContextMenu(menu, v, menuInfo);
    AdapterView.AdapterContextMenuInfo info = (AdapterView.AdapterContextMenuInfo)menuInfo;
    DataElement dataE = vData.elementAt(info.position);
    menu.setHeaderTitle(dataE.getPhoneNumber());
    menu.add(0, 0, 0, "Add Contact");
    menu.add(0, 1, 0, "Delete Contact");
}
```

ภาพที่ 4.12 การทำงานของเมธอด onCreateContextMenu()

4.3.5.5 เมธอด onContextItemSelected() จะทำงานเมื่อมีการเลือกรายการเมนู โดยหากเลือกรหัส (id) = 0 ซึ่งก็คือเมนู “Add Contact” จะใช้ Intent เพื่อเชื่อมโยงไปยังแอกทिवิตี ACTION เพื่อ

เพิ่มข้อมูล และสั่งให้ ActivityForResult ทำงานต่อ หรือหากเลือกรหัส (id) = 1 ซึ่งก็คือเมนู “Delete Contact” จะมีข้อความปรากฏขึ้นบนแถบไตเติ้ลว่า “DELETE SMS MESSAGE” และเนื้อหาข้อความ “Do you want to delete message?” โดยมีปุ่ม “OK” และ “CANCEL” ทั้งนี้ได้ตั้งค่าให้ปุ่ม “OK” เป็น positive ซึ่งจะคอยฟังเหตุการณ์เมื่อมีการคลิก โดยเมื่อมีการคลิกปุ่ม “OK” จะเป็นการเรียกใช้เมธอด deleteDataNumberSMS() บนคลาส DataBase เพื่อลบข้อความสั้นบนเลขหมายนั้นทั้งหมด และมีการปรับปรุงข้อมูลโดยการเรียกใช้เมธอด getDataSMS() บนคลาส DataBase แล้วมาเก็บไว้ที่ตัวแปร vData เพื่อปรับปรุงรายการใหม่ แต่หากเลือกปุ่ม “CANCEL” ซึ่งตั้งค่าเป็น Negative ไว้ ซึ่งเมื่อมีการคลิกก็จะนำรายการทั้งหมดแสดงไว้เหมือนเดิม

```

public boolean onContextItemSelected(MenuItem item)
{
    final AdapterView.AdapterContextMenuInfo info = (AdapterView.AdapterContextMenuInfo)item.getMenuInfo();
    Log.w("", "info.position = "+info.position);
    if(item.getItemId() == 0)
    {
        DataElement dataE = vData.elementAt(info.position);
        Intent intent = new Intent(Intent.ACTION_INSERT);
        intent.setType(ContactsContract.Contacts.CONTENT_TYPE);
        intent.putExtra(ContactsContract.Intents.EXTRA_FORCE_CREATE, true);
        intent.putExtra(ContactsContract.Intents.Insert.PHONE, dataE.getPhoneNumber());
        Phone_numberAddSMS = dataE.getPhoneNumber();
        int PICK_CONTACT = 100;
        startActivityForResult(intent, PICK_CONTACT);
    }
    else if(item.getItemId() == 1) {
        AlertDialog.Builder ad = new AlertDialog.Builder(this);
        ad.setTitle("DELETE SMS MESSAGE");
        ad.setMessage("Do you want to delete message?");
        ad.setPositiveButton("OK", new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int id) {
                DataElement dataE = vData.elementAt(info.position);
                db.deleteDataNumberSMS(dataE.getPhoneNumber());
                vData = db.getDataSMS();
                list_sms.setListAdapter(vData);
            }
        });
        ad.setNegativeButton("CANCEL", new DialogInterface.OnClickListener(){
            public void onClick(DialogInterface dialog, int id) {
            }
        });
        ad.show();
    }
    return true;
}

```

ภาพที่ 4.13 การทำงานของเมธอด onContextItemSelected()

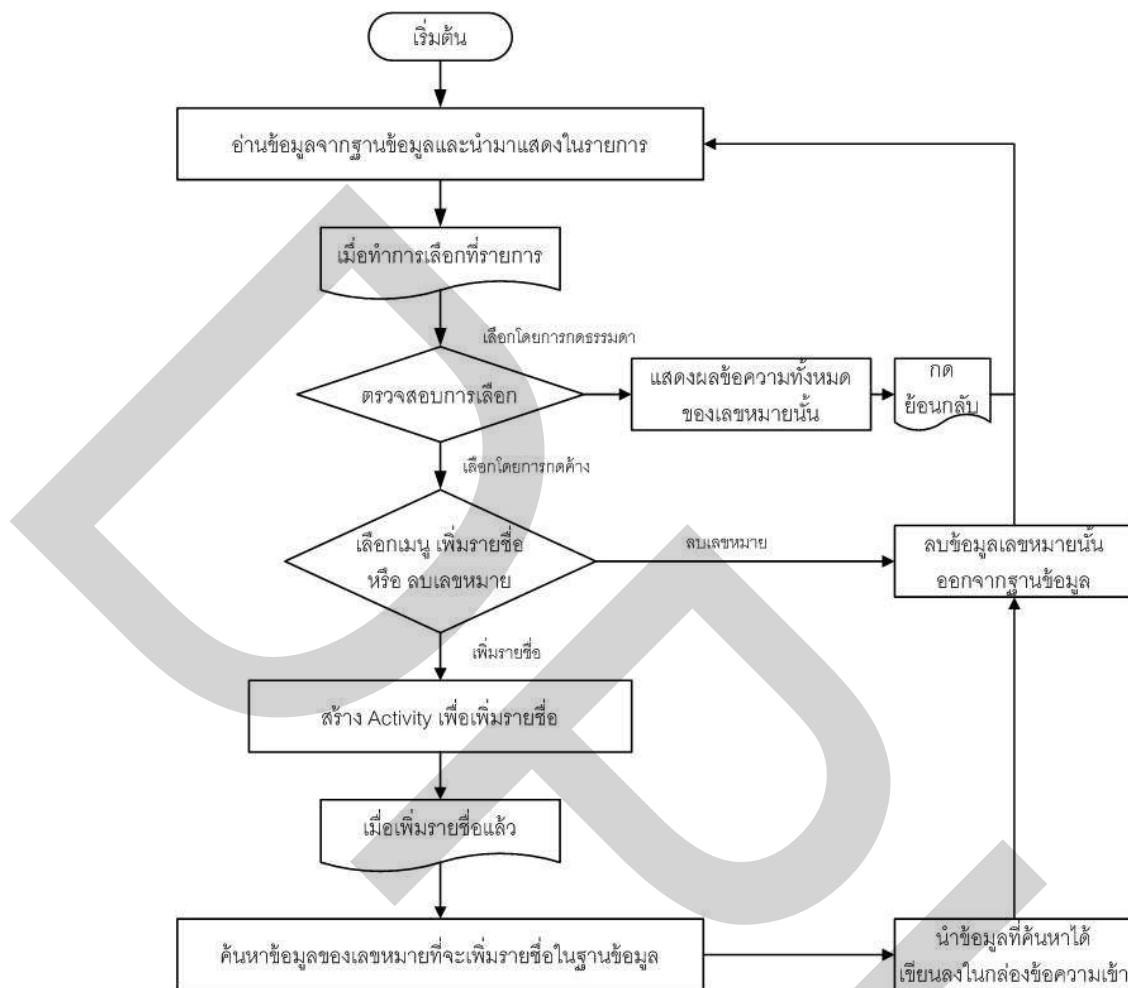
4.3.5.6 เมธอด onActivityResult() จะทำงานต่อจากเมธอด onContextItemSelected() ซึ่งเกิดจากการเลือกเมนู “Add Contact” โดยทำการกำหนดออบเจกต์ cp บนคลาส ContactPhone() และเรียกไปยังเมธอด getDataDetailSMS() เพื่อเรียกข้อมูล บนคลาส DataBase และนำค่าไปเก็บไว้ที่ vdata และตรวจสอบจำนวนข้อความสั้นทั้งหมดบนเลขหมายนั้น แล้วนำไปเขียนบนสมุด

โทรศัพท์ที่โดยผ่านเมธอด writeSMS() บนคลาส ContactPhone() ซึ่งมีข้อมูล เลขหมายโทรศัพท์ข้อความสั้น รหัสข้อความสั้น จากคลาส DataElement() เมื่อเขียนข้อความสั้นลงบนกล่องข้อความเข้าบนเครื่องแล้ว หลังจากนั้นจะเรียกใช้เมธอด deleteDataNumberSMS() บนคลาส DataBase เพื่อลบออกจากแอปพลิเคชัน และปรับปรุงรายการบนฐานข้อมูลแอปพลิเคชัน

```
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    Log.w("", "---ONACTIVITY_RESULT---"+requestCode+": "+resultCode);
    Log.w("", "data1 = "+data);
    if (requestCode == 100) {
        if (resultCode == RESULT_OK)
        {
            ContactPhone cp = new ContactPhone(this);
            Vector<DataElement> vdata = db.getDataDetailSMS(Phone_numberAddSMS);
            for (int i = 0; i < vdata.size(); i++) {
                DataElement dataE = vdata.elementAt(i);
                cp.writeSMS(dataE.getPhoneNumber(), dataE.getMessage(), dataE.getMessageID());
            }
            db.deleteDataNumberSMS(Phone_numberAddSMS);
            vData = db.getDataSMS();
            List_sms.setListAdapter(vData);
        }
    }
}
```

ภาพที่ 4.14 การทำงานของเมธอด onActivityResult()

และขั้นตอนการทำงานทั้งหมดเมื่อเปิดแอปพลิเคชัน สามารถจัดการทำงานได้ตามภาพที่ 4.15



ภาพที่ 4.15 ขั้นตอนการทำงานของโปรแกรมภายในเมื่อเปิดโปรแกรม

4.3.6 ความสัมพันธ์ระหว่างคลาส แสดงตามภาพที่ 4.16

4.3.6.1 คลาส MainActivity ทำหน้าที่ควบคุมและเชื่อมต่อกับส่วนติดต่อผู้ใช้ เช่น จะทำงานเมื่อมีการคลิก คลิกล้าง เลือกไอเทม (item) หรือเมนูย่อย เป็นต้น

4.3.6.2 คลาส DetailActivity ทำหน้าที่รับ Intent มาจาก คลาส MainActivity และเชื่อมต่อกับส่วนติดต่อผู้ใช้ โดยการเลือกไอเทมค้างไว้ และทำงานตามกล่องโต้ตอบที่ได้ตั้งค่าไว้

4.3.6.3 คลาส SMSReceive ทำหน้าที่ดักจับ service ตามที่ได้กำหนด event ไว้ และทำหน้าที่ตรวจสอบกับสมุดโทรศัพท์ และลบข้อความสั้นอัตโนมัติ

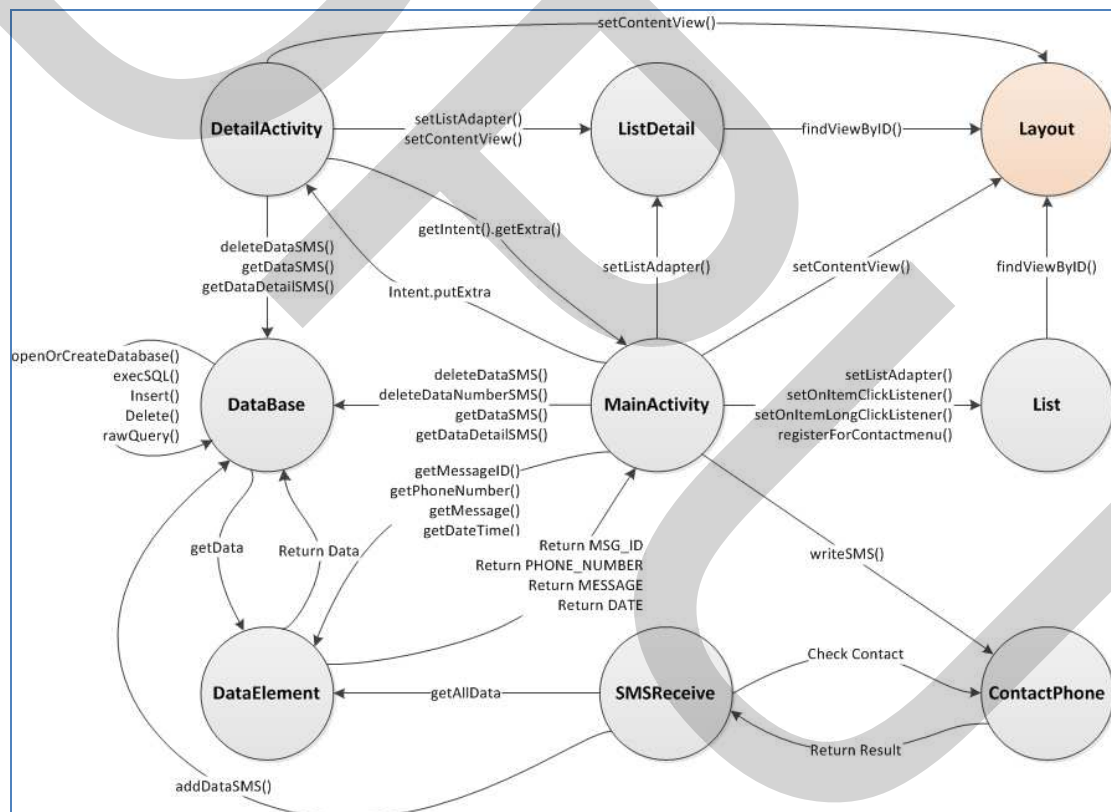
4.3.6.4 คลาส ContactPhone ทำหน้าที่ค้นหาข้อมูลในสมุดโทรศัพท์และนำมาตรวจสอบกับเลขหมายที่เข้ามา ว่ามีอยู่หรือไม่ และทำการเขียนข้อความสั้นลงบนระบบแอนดรอยด์ เมื่อมีการเรียกใช้ activity เพิ่มรายชื่อติดต่อ

4.3.6.5 คลาส Database ทำหน้าที่จัดการกับฐานข้อมูลทั้งหมดบน SQLite เช่น การสร้างฐานข้อมูล การเพิ่มข้อมูล การลบข้อมูล และการค้นหาข้อมูล โดยจะเชื่อมต่อกับคลาส DataElement

4.3.6.6 คลาส DataElement ทำหน้าที่เก็บข้อมูลจากฐานข้อมูล และเชื่อมต่อกับคลาส Database

4.3.6.7 คลาส List ทำหน้าที่แสดงผลข้อมูล โดยการสืบทอดจากคลาส ListView โดยแสดงผลข้อมูล เลขหมาย ข้อความ และ วันที่เวลา

4.3.6.8 คลาส ListDetail ทำหน้าที่แสดงผลข้อมูลเช่นเดียวกับคลาส List โดยการสืบทอดจากคลาส ListView โดยแสดงผลข้อมูล ข้อความ และ วันที่เวลา



ภาพที่ 4.16 ความสัมพันธ์ระหว่างคลาส

บทที่ 5

ผลการวิจัย

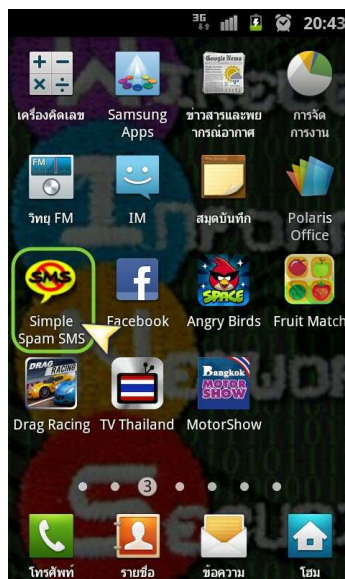
5.1 การติดตั้งแอปพลิเคชัน

เมื่อสร้างแอปพลิเคชันและคอมไพล์เรียบร้อยแล้ว จะต้องส่งออก (Export) ไฟล์ .apk ซึ่งสามารถนำไปรันบนการจำลองโปรแกรม (Emulator) เครื่องอื่น ๆ หรือบนโทรศัพท์เคลื่อนที่จริงได้ดังต่อไปนี้

5.1.1 การนำไฟล์ .apk ไปติดตั้งบนโทรศัพท์เคลื่อนที่จริงโดยผ่านสาย USB

1. เชื่อมต่อโทรศัพท์จริงเข้ากับเครื่องคอมพิวเตอร์ผ่านสาย USB ซึ่งจะต้องติดตั้งไดรเวอร์ของเครื่องโทรศัพท์นั้น ๆ ก่อน
2. พิมพ์คำสั่ง “adb install <.apk file>” ใน Command Prompt
3. ตรวจสอบที่เครื่องโทรศัพท์เคลื่อนที่จริงก็จะปรากฏแอปพลิเคชัน แสดงขึ้น ตาม

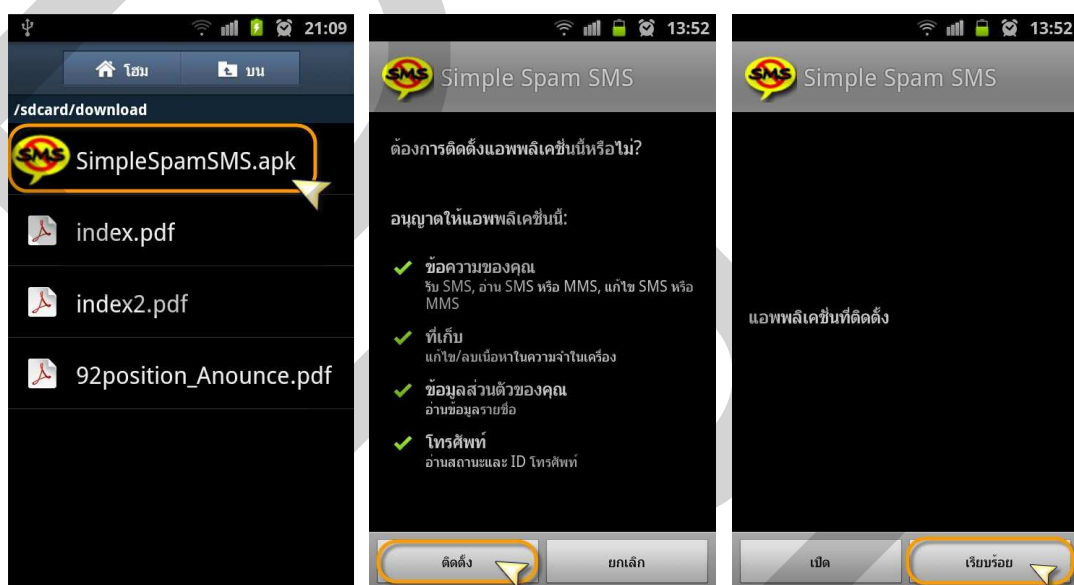
ภาพที่ 5.1



ภาพที่ 5.1 ภาพแอปพลิเคชันเมื่อติดตั้งผ่านสาย USB

5.1.2 การนำไฟล์ .apk ไปติดตั้งบนโทรศัพท์เคลื่อนที่จริงโดยผ่านหน่วยความจำภายนอก (SD-Card) หรือผ่านโพลเดอร์ดาวน์โหลด

1. นำไฟล์ .apk ไปใส่ไว้ที่หน่วยความจำภายนอก (SD-Card)
2. เปิดไฟล์ .apk
3. ระบบจะเข้าสู่หน้าจอการติดตั้งแอปพลิเคชัน โดยคลิกที่ปุ่ม “Install” จากนั้นรอจนกระทั่งระบบจะติดตั้งแอปพลิเคชันเสร็จ ตามภาพที่ 5.2 และภาพที่ 5.3



ภาพที่ 5.2 ภาพแอปพลิเคชันเมื่อติดตั้งผ่านหน่วยความจำภายนอก (SD-Card)



ภาพที่ 5.3 รายละเอียดแอปพลิเคชันบนเครื่องโทรศัพท์เคลื่อนที่

5.1.3 การเผยแพร่แอปพลิเคชันผ่าน กูเกิล เพลย์ สโตร์ (Google Play Store) โดยการใช้งานนั้นจะต้องสร้างบัญชีผู้ใช้งานขึ้นมา ซึ่งมีค่าธรรมเนียมในการใช้งานประมาณ 25 ดอลลาร์ แต่ในงานวิจัยชิ้นนี้ ผู้วิจัยไม่ได้เผยแพร่แอปพลิเคชันผ่าน กูเกิล เพลย์ สโตร์

5.2 การกรองข้อความสั้น

เมื่อติดตั้งแอปพลิเคชันลงบนเครื่องโทรศัพท์เคลื่อนที่แล้ว แอปพลิเคชันจะเริ่มทำงานอัตโนมัติ โดยหากข้อความสั้นที่เข้ามาเป็นหมายเลขที่ไม่มีอยู่ในสมุดโทรศัพท์ แอปพลิเคชันจะทำการดึงข้อความสั้นนั้น เข้ามาเก็บบนฐานข้อมูลบนแอปพลิเคชัน และทำการลบข้อความสั้นนั้นออกจากกล่องข้อความเข้า (Inbox) บนเครื่องโทรศัพท์เคลื่อนที่ ตามภาพ Log File ที่ 5.4

```

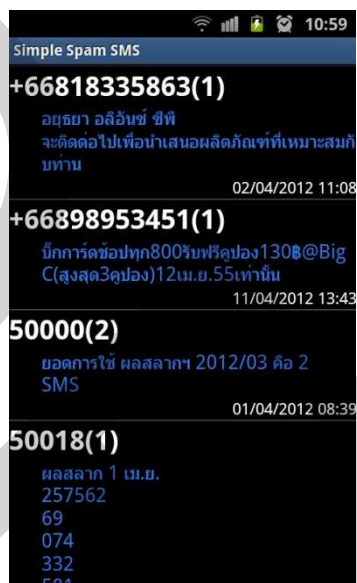
05-02 01:32:15.504: D/Mms/SmsReceiverService(27825): [SMS]Receiver handleMessage :
    Action =android.provider.Telephony.SMS_RECEIVED
05-02 01:32:15.504: D/Mms/SmsReceiverService(27825): handleSmsReceived()
05-02 01:32:15.509: I/MySMSMonitor(27858): onReceive
05-02 01:32:15.509: I/kk(27858): OP_SMS_RECEIVED
05-02 01:32:15.509: D/Gsm/SmsMessage(27825): SMS SC address: +66864220014
05-02 01:32:15.509: D/Gsm/SmsMessage(27825): originatingAddress.address : 50063
05-02 01:32:15.509: D/Gsm/SmsMessage(27825): SMS SC timestamp: 1335892529000
05-02 01:32:15.514: E/Gsm/SmsMessage(27825): hasUserDataHeader : false
05-02 01:32:15.514: D/Gsm/SmsMessage(27825): messageBody : 0864201035;ทดสอบ App:
    Simple Spam SMS ครับ
05-02 01:32:15.519: W/Mms/MTMessageParser(27825): Mobile Tracker - Normal msg received
05-02 01:32:15.519: D/Mms/SmsReceiverService(27825): insertMessage()
05-02 01:32:15.524: V/MmsSmsProvider(2857): uri=content://mms-sms/threadID?recipient=50063
05-02 01:32:15.529: V/Telephony(27825): getOrCreateThreadId cursor cnt: 1
05-02 01:32:15.529: E/Mms/SmsReceiverService(27825): GCF Mode Off
05-02 01:32:15.699: V/MmsSmsProvider(2857): uri=content://mms-sms/conversations/21
05-02 01:32:15.714: D/Mms/ComposeMessage(27825): onQueryComplete()
    MESSAGE_LIST_QUERY_TOKEN
05-02 01:32:15.719: V/MmsSmsProvider(2857): uri=content://mms-sms/conversations/21
05-02 01:32:15.824: D/Mms/MessagingNotification(27825): updateAllHistoryAsRead()
05-02 01:32:15.829: V/MmsProvider(2857): Query uri=content://mms, match=0
05-02 01:32:15.839: D/Mms/MessagingNotification(27825): getSmsNewMessageNotificationInfo:
    count=1, first addr=50063, thread_id=22
05-02 01:32:15.844: D/Mms/MessagingNotification(27825): updateHistoryReadFlag().messageid=109
05-02 01:32:15.879: V/MmsSmsProvider(2857): uri=content://mms-sms/wap-push-messages
05-02 01:32:15.884: I/StatusBarManagerService(2706): REMOVE-50063: 0864201035;
    ทดสอบ App: Simple Spam SMS ครับ , 0x7f020149
05-02 01:32:15.889: D/Mms/MessagingNotification(27825): updateNotification().isNew=true,ticker=
    50063: 0864201035;ทดสอบ App: Simple Spam SMS ครับ
05-02 01:32:16.369: I/LogsProvider(26563): insert data - logtype : 300 number : 50063 contactid : 0
    name : null messageid : 109
05-02 01:32:16.369: I/LogsProvider(26563): delete : _id in (select _id from logs where logtype=300
    OR logtype=200 ORDER BY date DESC limit -1 offset 500)
05-02 01:32:16.369: I/LogsProvider(26563): delete time : 1
05-02 01:32:16.369: D/Mms/MessagingNotification(27825): updateHistoryForSms().address=50063
05-02 01:32:16.519: D/SMSDispatcher(2857): BroadcastReceiver - mResultReceiver
05-02 01:32:16.519: D/SMSDispatcher(2857): GcfMode :
05-02 01:32:16.519: E/SMSDispatcher(2857): intent.getAction() : android.provider.Telephony.SMS_RECEIVED
05-02 01:32:16.519: D/SMSDispatcher(2857): mResultReceiver : success = true
05-02 01:32:16.589: W/(28049): postalCat = <+6644259706:Cat Nakhonratchasima> *
    <+66805217772:Vichai Singhawannakul> *.....
05-02 01:32:16.589: W/(28049): phone_new = 50063
05-02 01:32:16.594: W/(28049): cursor size = 0
05-02 01:32:16.594: W/(28049): isHave = false
05-02 01:32:16.624: W/add Data(28049): 50063
05-02 01:32:16.629: W/getDataSMS size(28049): 3
05-02 01:32:16.634: W/getDataSMS size(28049): 1
05-02 01:32:16.644: W/getDataSMS size(28049): 2
05-02 01:32:16.649: W/getDataSMS size(28049): 1

```

ภาพที่ 5.4 Log การทำงานเมื่อมีข้อความสั้นถูกส่งเข้ามา

5.3 การจัดการแอดทีวีที่บนแอปพลิเคชัน

5.3.1 การเปิดแอปพลิเคชันการกรองข้อความสั้น ให้เลือกหรือสัมผัสไปที่แอปพลิเคชัน “Simple Spam SMS” บนหน้าจอเครื่องโทรศัพท์เคลื่อนที่ แอปพลิเคชันจะทำการดึงข้อมูลจากฐานข้อมูล (SQLite) มาแสดงบนแอปพลิเคชัน ตามภาพที่ 5.5 และภาพ Log File ที่ 5.6



ภาพที่ 5.5 รายการเลขหมายเมื่อดึงออกมาจากฐานข้อมูลของแอปพลิเคชัน

```
05-02 00:40:28.134: I/ActivityManager(2706): Starting: Intent { act=android.intent.action.MAIN
    cat=[android.intent.category.LAUNCHER] flg=0x10200000
    cmp=com.android.smscontact/.MainActivity bnds=[365,494][475,632] } from pid 2881

05-02 00:40:28.379: W/(28049): cursor size = 2676

05-02 00:41:16.694: W/(28049): postalCat = <+6644259706:Cat Nakhonratchasima>* <+66805217772:Vichai Sing
05-02 00:41:16.704: W/getDataSMS size(28049): 3
05-02 00:41:16.709: W/getDataSMS size(28049): 1
05-02 00:41:16.714: W/getDataSMS size(28049): 2
05-02 00:41:16.719: W/getDataSMS size(28049): 3
05-02 00:41:16.739: W/DataElement method date(28049): 1335773131218
05-02 00:41:16.749: W/DataElement method date(28049): 1335808645974
05-02 00:41:16.754: W/DataElement method date(28049): 1335892829925

05-02 00:41:16.789: I/ActivityManager(2706): Displayed com.android.smscontact/.MainActivity: +48s618ms
```

ภาพที่ 5.6 Log การทำงานเมื่อมีการเปิดแอปพลิเคชัน

5.3.2 การอ่านข้อความสั้น เมื่อแอปพลิเคชันดึงข้อความสั้นมาจากฐานข้อมูลแล้ว สามารถอ่านข้อความสั้นจากหน้าแรกนี้ได้ หรือหากมีมากกว่า 1 ข้อความสั้นต่อเลขหมาย สามารถคลิกเข้าไปดูรายละเอียดได้ ตามภาพที่ 5.7 และภาพ Log File ที่ 5.8



ภาพที่ 5.7 การอ่านข้อความสั้น

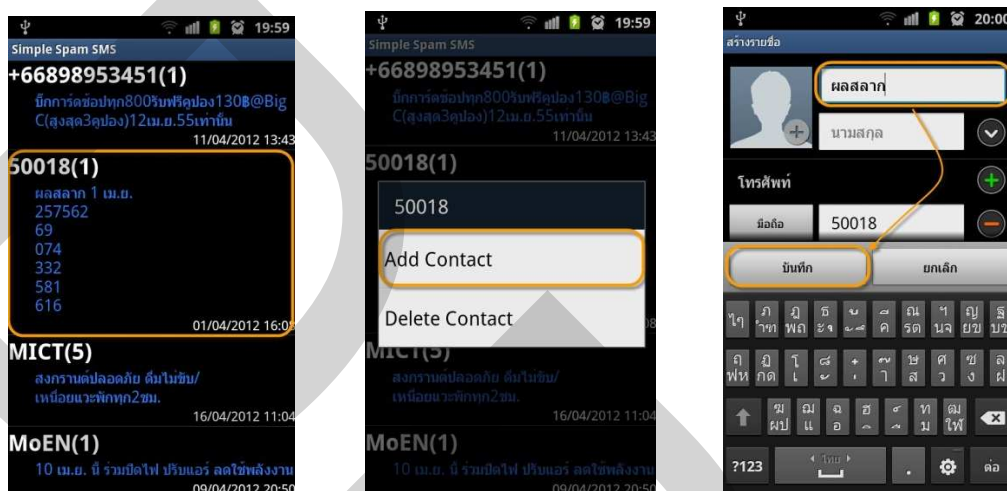
```

05-04 09:32:38.020: I/InputReader(2707): dispatchTouch::touch event's action is 1
05-04 09:32:38.020: I/InputDispatcher(2707): Delivering touch to current input target: action: 1,
channel '409f2120 com.android.smscontact/com.android.smscontact.MainActivity (server)'
05-04 09:32:38.190: I/ActivityManager(2707): Starting: Intent { cmp=com.android.smscontact/
.DetailActivity (has extras) } from pid 32763
05-04 09:32:38.195: W/ActivityManager(2707): Trying to launch com.android.smscontact/.DetailActivity
05-04 09:32:38.315: W/getDataSMS size(32763): 5
05-04 09:32:38.350: W/DataElement method date(32763): 1335773131218
05-04 09:32:38.355: W/DataElement method date(32763): 1335907501554
05-04 09:32:38.360: W/DataElement method date(32763): 1335910239140
05-04 09:32:38.365: W/DataElement method date(32763): 1335911379214
05-04 09:32:38.365: W/DataElement method date(32763): 1335911425146
05-04 09:32:38.570: I/ActivityManager(2707): Displayed com.android.smscontact/.DetailActivity: +377ms
    
```

ภาพที่ 5.8 Log การทำงานเมื่อเปิดอ่านรายละเอียดข้อความ

5.3.3 การเพิ่มรายชื่อ (Add Contact) ในกรณีที่ต้องการเพิ่มเลขหมายลงบนสมุดโทรศัพท์ เพื่อให้เป็นข้อความสั้นปกติ ที่บันทึกไว้บนเครื่องโทรศัพท์เคลื่อนที่ เมื่อบันทึกหรือเพิ่มรายชื่อแล้ว

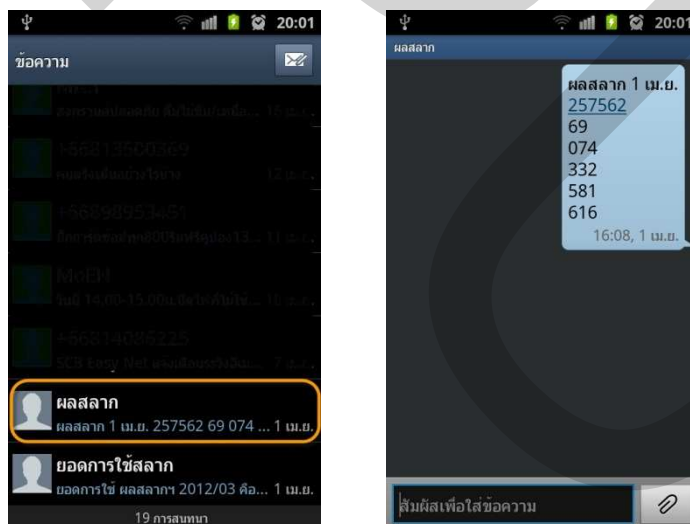
แอปพลิเคชันจะทำการเขียนข้อความสั้นทั้งหมดจากเลขหมายนั้นลงบนกล่องข้อความเข้า (Inbox) บนเครื่องโทรศัพท์เคลื่อนที่ และจะลบเลขหมายรวมถึงข้อความสั้นนั้น ออกจากฐานข้อมูลของแอปพลิเคชัน ตามภาพที่ 5.9 และภาพ Log File ที่ 5.10



(ก)

(ข)

(ค)



(ง)

(จ)

ภาพที่ 5.9 การเพิ่มรายชื่อ

```

05-04 15:38:45.215: I/InputReader(2707): dispatchTouch::touch event's action is 0
05-04 15:38:45.215: I/InputDispatcher(2707): Delivering touch to current input target action: 0, channel
'40c78308 AtchDlg:com.android.smscontact/com.android.smscontact.MainActivity (server)'
05-04 15:38:45.455: I/InputReader(2707): dispatchTouch::touch event's action is 1
05-04 15:38:45.455: I/InputDispatcher(2707): Delivering touch to current input target action: 1, channel
'40c78308 AtchDlg:com.android.smscontact/com.android.smscontact.MainActivity (server)'

05-04 15:38:45.500: I/ActivityManager(2707): Starting: Intent { act=android.intent.action.INSERT typ=vnd.android.
cursor.dir/contact cmp=com.android.contacts/.ui.EditContactActivity (has extras) }
05-04 15:38:45.505: W/ActivityManager(2707): Trying to launch com.android.contacts/.ui.EditContactActivity

05-04 15:38:48.485: D/KindSectionView(2987): [Kind] updateAddEnabled : true
05-04 15:38:48.490: D/[zzang](2987): titleRes2 = 2131427795

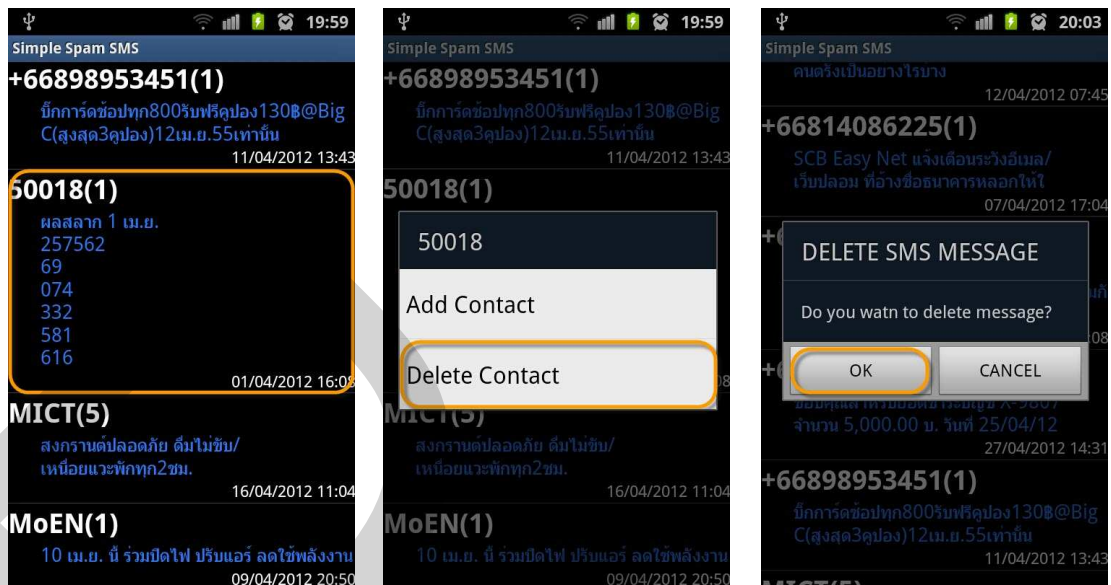
05-04 15:39:04.580: W/(24439): data1 = Intent { dat=content://com.android.contacts/contacts/lookup/
2341r4986-2D31533D572D5335332D532D2A9439/6399 (has extras) }
05-04 15:39:04.580: W/getDataSMS size(24439): 1
05-04 15:39:04.585: V/MmsSmsProvider(2852): uri=content://mms-sms/threadID?recipient=9999
05-04 15:39:04.590: D/MmsSmsProvider(2852): getThreadId: create new thread_id for recipients [9999]
05-04 15:39:04.590: V/MmsSmsProvider(2852): insertThread: recipientIds: 5, numberOfRecipients: 1
05-04 15:39:04.655: V/MmsSmsProvider(2852): insertThread: threadId=21
05-04 15:39:04.660: V/Telephony(2852): getOrCreateThreadId cursor cnt: 1
05-04 15:39:04.770: W/(24439): phone_number = 9999

05-04 15:39:04.810: W/getDataSMS size(24439): 2
05-04 15:39:04.815: W/getDataSMS size(24439): 2
05-04 15:39:04.820: W/getDataSMS size(24439): 20
05-04 15:39:04.840: W/DataElement method date(24439): 1335808645974
05-04 15:39:04.850: W/DataElement method date(24439): 1335907145091

```

ภาพที่ 5.10 Log การทำงานเมื่อเพิ่มรายชื่อ

5.3.4 การลบรายชื่อ (Delete Contact) ในกรณีที่ต้องลบเลขหมายที่ได้กรองไว้ก่อน สามารถทำได้โดยลบทั้งเลขหมาย ซึ่งจะทำให้ข้อความสั้นที่ถูกส่งมาจากเลขหมายนี้ ถูกลบทิ้งทั้งหมด ตามภาพที่ 5.11 และ ภาพ Log File ที่ 5.12



ภาพที่ 5.11 การลบรายชื่อ

```

05-04 16:12:46.795: W/(27739): info.position = 2
05-04 16:12:48.370: W/(27739): phone_number = 9999
05-04 16:12:48.405: W/getDataSMS size(27739): 2
05-04 16:12:48.410: W/getDataSMS size(27739): 2
05-04 16:12:48.420: W/getDataSMS size(27739): 20
05-04 16:12:48.430: W/DataElement method date(27739): 1335808645974
05-04 16:12:48.445: W/DataElement method date(27739): 1335907145091
05-04 16:13:05.160: E/Database(27739): close() was never explicitly called on database
'/data/data/com.android.smscontact/databases/SMS_BLOCK'
05-04 16:13:05.160: E/Database(27739): android.database.sqlite.DatabaseObjectNotClosedException:
Application did not close the cursor or database object that was opened here
05-04 16:13:05.160: E/Database(27739): at android.database.sqlite.SQLiteDatabase.<init>
(SQLiteDatabase.java:1958)
05-04 16:13:05.160: E/Database(27739): at android.database.sqlite.SQLiteDatabase.openDatabase
(SQLiteDatabase.java:887)
05-04 16:13:05.160: E/Database(27739): at android.database.sqlite.SQLiteDatabase.openOrCreateDatabase
(SQLiteDatabase.java:965)
05-04 16:13:05.160: E/Database(27739): at android.database.sqlite.SQLiteDatabase.openOrCreateDatabase
(SQLiteDatabase.java:958)
05-04 16:13:05.160: E/Database(27739): at android.app.ContextImpl.openOrCreateDatabase
(ContextImpl.java:584)
05-04 16:13:05.160: E/Database(27739): at android.content.ContextWrapper.openOrCreateDatabase
(ContextWrapper.java:203)
05-04 16:13:05.160: E/Database(27739): at com.android.smscontact.DataBase.deleteDataNumberSMS
(DataBase.java:198)
05-04 16:13:05.160: E/Database(27739): at com.android.smscontact.MainActivity$1.onClick(MainActivity.java:176)
05-04 16:13:05.160: E/Database(27739): at com.android.internal.app.AlertController$ButtonHandler.
handleMessage(AlertController.java:162)

```

ภาพที่ 5.12 Log การทำงานเมื่อลบรายชื่อหรือเลขหมาย

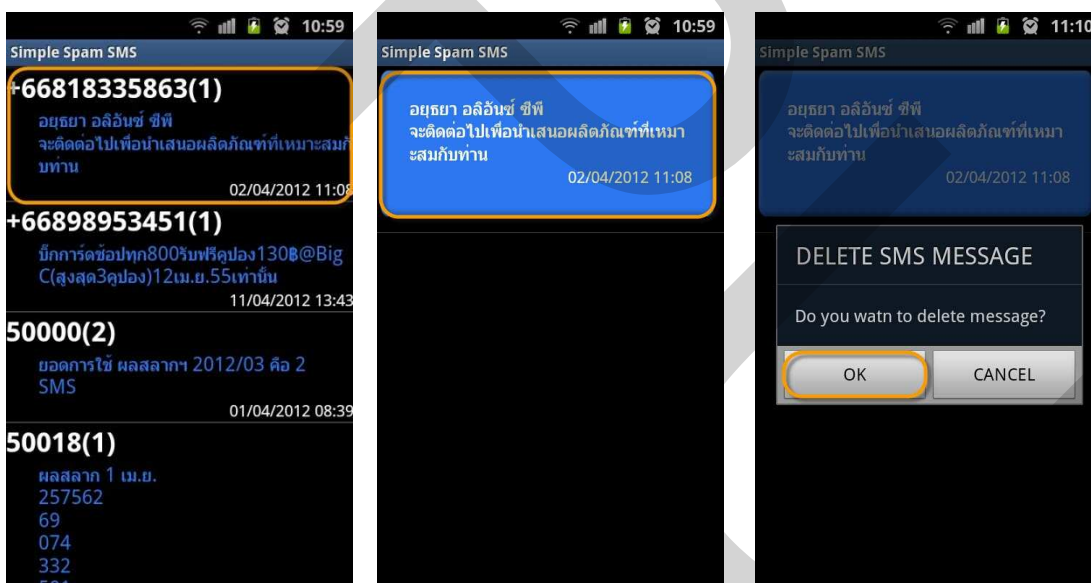
```

05-04 16:13:05.160: E/Database(27739): at android.os.Handler.dispatchMessage(Handler.java:99)
05-04 16:13:05.160: E/Database(27739): at android.os.Looper.loop(Looper.java:123)
05-04 16:13:05.160: E/Database(27739): at android.app.ActivityThread.main(ActivityThread.java:3691)
05-04 16:13:05.160: E/Database(27739): at java.lang.reflect.Method.invokeNative(Native Method)
05-04 16:13:05.160: E/Database(27739): at java.lang.reflect.Method.invoke(Method.java:507)
05-04 16:13:05.160: E/Database(27739): at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:847)
05-04 16:13:05.160: E/Database(27739): at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:605)
05-04 16:13:05.160: E/Database(27739): at dalvik.system.NativeStart.main(Native Method)
05-04 16:13:45.560: I/dalvikvm(27739): threadid=4: reacting to signal 3
05-04 16:13:45.610: I/dalvikvm(27739): Wrote stack traces to '/data/anr/traces.txt'

```

ภาพที่ 5.12 (ต่อ)

5.3.5 การลบข้อความสั้นย่อ ในกรณีที่ต้องลบข้อความสั้นบางข้อความของหมายเลขนั้น ๆ สามารถทำได้โดย เลือกเข้าไปที่เลขหมายนั้น ๆ แล้วเลือกข้อความสั้นที่ต้องการลบ ตามภาพที่ 5.13 และ ภาพที่ 5.14



ภาพที่ 5.13 การลบข้อความสั้นย่อ

```

05-04 21:04:55.235: W/getDataSMS size(18781): 5
05-04 21:04:55.260: W/DataElement method date(18781): 1336139080203
05-04 21:04:55.265: W/DataElement method date(18781): 1336139169549
05-04 21:04:55.270: W/DataElement method date(18781): 1336139258058
05-04 21:04:55.275: W/DataElement method date(18781): 1336139302260
05-04 21:04:55.275: W/DataElement method date(18781): 1336139347683

05-04 21:05:00.715: W/(18781): ID to delete = 1336139080203

05-04 21:05:00.770: W/getDataSMS size(18781): 3
05-04 21:05:00.775: W/getDataSMS size(18781): 2
05-04 21:05:00.780: W/getDataSMS size(18781): 20
05-04 21:05:00.785: W/getDataSMS size(18781): 4
05-04 21:05:00.800: W/DataElement method date(18781): 1335808645974
05-04 21:05:00.805: W/DataElement method date(18781): 1335907145091
05-04 21:05:00.815: W/DataElement method date(18781): 1336139347683

05-04 21:06:22.470: W/getDataSMS size(18781): 4
05-04 21:06:22.485: W/DataElement method date(18781): 1336139169549
05-04 21:06:22.490: W/DataElement method date(18781): 1336139258058
05-04 21:06:22.495: W/DataElement method date(18781): 1336139302260
05-04 21:06:22.495: W/DataElement method date(18781): 1336139347683

05-04 21:06:23.590: E/Database(18781): close() was never explicitly called on database
'/data/data/com.android.smscontact/databases/SMS_BLOCK'
05-04 21:06:23.590: E/Database(18781): android.database.sqlite.DatabaseObjectNotClosedException:
Application did not close the cursor or database object that was opened here
05-04 21:06:23.590: E/Database(18781): at android.database.sqlite.SQLiteDatabase.<init>
(SQLiteDatabase.java:1958)
05-04 21:06:23.590: E/Database(18781): at android.database.sqlite.SQLiteDatabase.openDatabase
(SQLiteDatabase.java:887)
05-04 21:06:23.590: E/Database(18781): at android.database.sqlite.SQLiteDatabase.openOrCreateDatabase
(SQLiteDatabase.java:965)
05-04 21:06:23.590: E/Database(18781): at android.database.sqlite.SQLiteDatabase.openOrCreateDatabase
(SQLiteDatabase.java:958)
05-04 21:06:23.590: E/Database(18781): at android.app.ContextImpl.openOrCreateDatabase
(ContextImpl.java:584)
05-04 21:06:23.590: E/Database(18781): at android.content.ContextWrapper.openOrCreateDatabase
(ContextWrapper.java:203)
05-04 21:06:23.590: E/Database(18781): at com.android.smscontact.DataBase.deleteDataSMS
(DataBase.java:186)
05-04 21:06:23.590: E/Database(18781): at com.android.smscontact.DetailActivity$1.
onClick(DetailActivity.java:45)
05-04 21:06:23.590: E/Database(18781): at com.android.internal.app.AlertController$ButtonHandler.
handleMessage(AlertController.java:162)
05-04 21:06:23.590: E/Database(18781): at android.os.Handler.dispatchMessage(Handler.java:99)
05-04 21:06:23.590: E/Database(18781): at android.os.Looper.loop(Looper.java:123)
05-04 21:06:23.590: E/Database(18781): at android.app.ActivityThread.main(ActivityThread.java:3691)
05-04 21:06:23.590: E/Database(18781): at java.lang.reflect.Method.invokeNative(Native Method)
05-04 21:06:23.590: E/Database(18781): at java.lang.reflect.Method.invoke(Method.java:507)
05-04 21:06:23.590: E/Database(18781): at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.
run(ZygoteInit.java:847)
05-04 21:06:23.590: E/Database(18781): at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:605)
05-04 21:06:23.590: E/Database(18781): at dalvik.system.NativeStart.main(Native Method)

```

ภาพที่ 5.14 Log การทำงานเมื่อลบข้อความสั้นบนหมายเลขนั้น ๆ

5.4 เปรียบเทียบแอปพลิเคชัน

บริการกูเกิลเพลย์ สโตร์ (Google play store) เป็นตลาดดาวน์โหลดแอปพลิเคชัน ให้สามารถค้นหาแอปพลิเคชันต่าง ๆ และเลือกติดตั้งได้จากเพลย์ สโตร์ บนมือถือหรือติดตั้งผ่านเว็บไซต์ โดยในขณะนี้แอปพลิเคชันให้เลือกดาวน์โหลดมากกว่า 675,000 แอปพลิเคชัน ทั้งนี้แอปพลิเคชันที่เกี่ยวข้องกับการกรองข้อความสั้นในกูเกิลเพลย์ สโตร์ เมื่อเทียบกับแอปพลิเคชันในงานวิจัยชิ้นนี้ พบว่าสามารถทำงานได้เหมือนกันและแตกต่างกันในหลายกรณี ตามตารางที่ 5.1

ตารางที่ 5.1 เปรียบเทียบแอปพลิเคชันในตลาด

ความสามารถ / แอปพลิเคชัน	Simple Spam SMS	Anti SMS Bomber	Black list	Black list Free	Anti Spam SMS	SMS Filter	Anti SMS Spam
1. บล็อกข้อความซ้ำกัน		✓					
2. เรียกดูประวัติข้อความ	✓	✓	✓				
3. ดักข้อความสั้นที่ว่าง		✓					
4. บล็อกเลขหมายที่ไม่รู้จัก	✓		✓	✓	✓		✓
5. กรองคำเฉพาะ				✓			✓
6. เพิ่ม/ลบ เลขหมายจากรายการแบล็คลิสต์	✓						
7. แสดง/ล้าง ประวัติข้อความสั้น	✓					✓	
8. แสดงรายการเลขหมายที่บล็อก	✓					✓	✓

ที่มา: Google Play Store [online] : เข้าถึง 5 ก.พ. 2555. จาก

<https://play.google.com/store>

5.5 ผลการทดลอง

5.5.1 ชุดข้อมูลทดสอบสมุดโทรศัพท์ ทำการทดสอบโดยใช้โมเดลเตอร์ ซึ่งสามารถกำหนดรายชื่อลงบนสมุดโทรศัพท์และสามารถทดสอบการส่งข้อความสั้นได้ด้วย โดยใช้ชุดข้อมูลทดสอบจำนวน 107 รายชื่อ ตามตารางที่ 5.2

ตารางที่ 5.2 ชุดข้อมูลทดสอบที่ได้บันทึกลงบนสมุดโทรศัพท์

ประเภทการรายชื่อบนสมุดโทรศัพท์	จำนวน
บันทึกเลขหมาย แต่ไม่บันทึกชื่อ	7
เลขหมายพิเศษ 4-9 หลัก	8
บันทึกชื่อเดียว แต่หลายเลขหมาย	18
ชื่อทั่วไป	74
	107

5.5.2 ชุดข้อมูลทดสอบที่ไม่มีอยู่ในสมุดโทรศัพท์ ทำการทดสอบโดยการสมมติชุดข้อมูลตัวอย่าง ที่มีความหลากหลายของชนิดข้อมูลจำนวน 103 ตัวอย่าง ตามตารางที่ 5.3

ตารางที่ 5.3 ชุดข้อมูลที่ไม่ได้บันทึกบนสมุดโทรศัพท์

ชนิดเลขหมาย	จำนวนครั้งที่ใช้ส่ง	จำนวนครั้งที่กรองได้
เลขหมายพิเศษ 3 หลัก	5	5
เลขหมายพิเศษ 4 หลัก	12	12
เลขหมายพิเศษ 5 หลัก	12	12
เลขหมายพิเศษ 6 หลัก	15	15
เลขหมายพิเศษ 9 หลัก	20	20
เลขหมายโทรศัพท์ทั่วไป 10 หลัก	12	12
เลขหมายต่างประเทศ มากกว่า 10 หลัก	18	18
เลขหมายบริการอดิโอ เท็กซ์	9	9
	103	103

บทที่ 6

สรุปผลการศึกษาและวิจัย

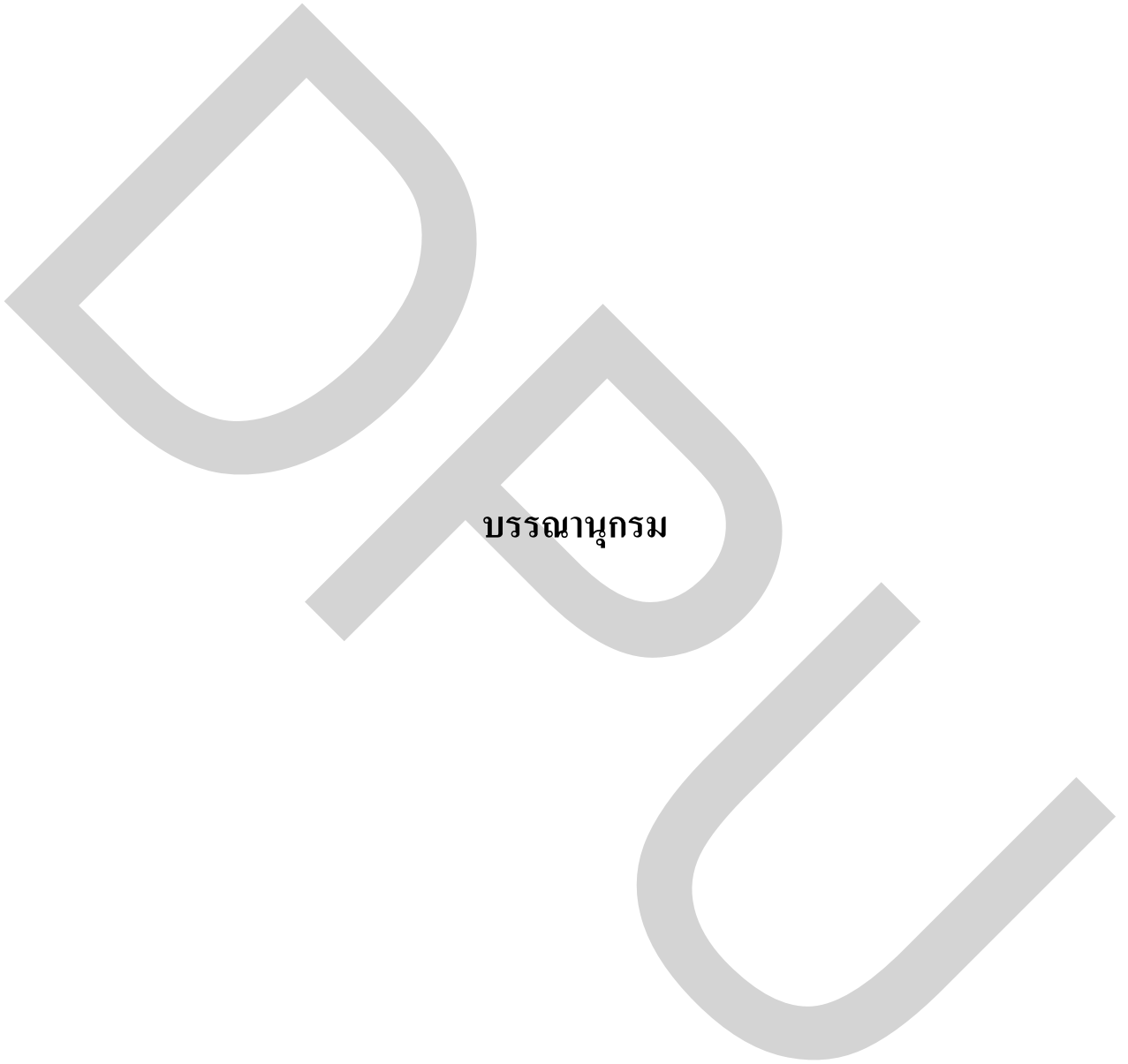
6.1 สรุปผลการศึกษาและวิจัย

จากการสร้างและทดลองแอปพลิเคชันการกรองข้อความสั้นบนโทรศัพท์เคลื่อนที่แอนดรอยด์โดยอาศัยข้อมูลในสมุดโทรศัพท์นั้น แอปพลิเคชันสามารถกรองข้อความสั้นของหมายเลขที่ไม่มีอยู่ในสมุดโทรศัพท์ได้อย่างสมบูรณ์ โดยได้ทำการทดลองทั้งกับโปรแกรมจำลอง (Emulator) และโทรศัพท์เคลื่อนที่จริง ซึ่งสามารถกรองข้อความที่ถูกส่งมาจากเลขหมายได้หลายรูปแบบ เช่น เลขหมายโทรศัพท์ 10 หลักทั่วไป เลขหมายที่มีเครื่องหมายบวก (+) นำหน้า เลขหมายพิเศษ 4 ตัว หรือ 5 ตัว เลขหมายจากต่างประเทศ เลขหมายที่มากกว่า 10 หลัก เลขหมายที่ไม่มีบันทึกรายชื่อ เลขหมายที่บันทึกชื่อและเลขหมายโทรศัพท์มากกว่า 1 เลขหมาย เลขหมายที่บันทึกชื่อเป็นอักขระพิเศษ และเลขหมายที่มีการแสดงเป็นชื่อแทนเลขหมายก็สามารถกรองได้ 100%

6.2 ข้อเสนอแนะและงานวิจัยในอนาคต

6.2.1 จากการทดลองแอปพลิเคชัน สามารถกรองข้อความได้ แต่เป็นการดึงข้อความสั้นจากกล่องข้อความเข้า (Inbox) บนเครื่องโทรศัพท์ แล้วจึงเข้าไปลบข้อความสั้นนั้นออกจากกล่องข้อความเข้า ทั้งนี้เป็นผลอันเนื่องมาจากระบบแอนดรอยด์ ไม่อนุญาตให้เขียนข้อความสั้นลงบนแอปพลิเคชันโดยตรง

6.2.2 แอปพลิเคชันนี้ทำการกรองทุกเลขหมาย ที่ไม่มีอยู่ในสมุดโทรศัพท์ ทำให้ในบางกรณีอาจจะเกิดผลเสียหรือผลกระทบได้ อย่างเช่น ข้อความสั้นการแจ้งเตือนแผ่นดินไหว หรือแจ้งเตือนเมื่อเกิดเหตุภัยพิบัติ ซึ่งข้อความเหล่านี้ได้ถูกส่งจากเลขหมายที่ไม่มีการบันทึกไว้ในสมุดโทรศัพท์ ทำให้ผู้ใช้อาจจะไม่ทราบข้อมูลเหล่านี้ ซึ่งบันทึกไว้บนแอปพลิเคชัน



บรรณานุกรม

บรรณานุกรม

ภาษาไทย

หนังสือ

จักรชัย โสอินทร์, และ พงษ์ศธร จันทรียอย. (2554). *Basic Android App Development*. นนทบุรี: ไอดีซี.

เจมส์ สตีล, และเนลสัน โท. (2554). *รวมโค้ด Android App [The Android Developer's Cookbook]* (ไพบุลย์ สวัสดิ์ปัญญาโชติ, ผู้แปลและเรียบเรียง) กรุงเทพฯ: ทู ดิจิตอล คอนเท้นท์ แอนท์ มีเดีย. (ต้นฉบับพิมพ์ ปี ค.ศ. 2010).

พร้อมเลิศ หล่อวิจิตร. (2555). *คู่มือเขียนแอป Android สำหรับผู้เริ่มต้น*. กรุงเทพฯ: โปรวิชัน.

ยศธร ภูมิสุทธิ. (2554). *การเขียนโปรแกรมโมรคัพที่มีอยู่บนแพลตฟอร์ม Android ขั้นพื้นฐาน*.

กรุงเทพฯ: สำนักฝึกอบรมและพัฒนาทรัพยากรบุคคล มหาวิทยาลัยเทคโนโลยีมหานคร

เอกสารอิเล็กทรอนิกส์

โครงการ My Computer Law กฎหมายคอมพิวเตอร์ฉบับประชาชน. (2554). *ข้อความขยะ...อะไรคือ มาตรการ “เลือกรับ” และ “เลือกไม่รับ”*. สืบค้นเมื่อ 16 กันยายน 2554, จาก <http://mycomputerlaw.in.th/2011/07/spam-sms-opt-in-opt-out/>

ชัยวุฒิ พรหมบุตร. (2554). *ทุก ๆ เรื่อง ที่เกี่ยวกับ android โดย มาสเตอร์ อิ่ง*. สืบค้นเมื่อ 17 กันยายน 2554, จาก <http://androidthai.in.th/>

บริษัท สยามอินเทลลิเจนซ์ยูนิท จำกัด. (2554). *บทเรียนจากต่างประเทศในการแก้ปัญหา SMS Spam ทางกฎหมายและนโยบาย*. สืบค้นเมื่อ 16 กันยายน 2554, จาก <http://www.siamintelligence.com/sms-spam-research-project/>

สุธี สดประเสริฐ. (2554). *Android Development*. สืบค้นเมื่อ 23 มกราคม 2555, จาก <https://bitbucket.org/ssutee>

MyComsci.com. (2554). *Android SQLite: การบันทึกข้อมูลบน SQLite และการดึงข้อมูลมาแสดงบน ListView*. สืบค้นเมื่อ 3 ตุลาคม 2554, จาก <http://www.mycomsci.com/?Mod=view&aid=126>

ภาษาอังกฤษ

BOOK

Reto Meier. (2009). *Professional Android Application Development*, Indiana: Wiley Publishing, Inc.,

ELECTRONIC SOURCES

Airplanez, (2011). *AntiSpamSMS Apps*. Retrieved February 6, 2012, from https://play.google.com/store/apps/details?id=net.airplanez.antispamsms&feature=search_result#?t=W251bGwsMSwxLDEsIm5ldC5haXJwbGFuZXouYW50aXNwYW1zbXMiXQ..

Android developer. (2009). *Manifest permission*. Retrieved October 15, 2011, from <http://developer.android.com/reference/android/Manifest.permission.html>

Anttek company. (2011). *Blacklist Apps*. Retrieved February 5, 2012, from https://play.google.com/store/apps/details?id=org.baole.app.blacklist&feature=search_result#?t=W251bGwsMSwxLDEsIm9yZy5iYW9sZS5hcHAuYmxhY2tsaXN0Ii0..

Droid Mate, (2011). *Anti SMS Spam & Text Filter Apps*. Retrieved February 6, 2012, from https://play.google.com/store/apps/details?id=org.baole.app.antismsspam&feature=search_result#?t=W251bGwsMSwxLDEsIm9yZy5iYW9sZS5hcHAuYW50aXNtc3NwYW0iXQ..

Marakana company. (2012). *Architecting Android Apps*. Retrieved March 3, 2012, from http://marakana.com/s/architecting_android_apps,1178/index.html

Ming software, (2011). *Blacklist free Apps*. Retrieved February 6, 2012, from https://play.google.com/store/apps/details?id=vc.software.blacklist&feature=more_from_developer#?t=W251bGwsMSwxLDEwMiwidmMuc29mdHdhcmUuYmxhY2tsaXN0Ii0..

Nielsen company. (2012). *Operating System Share-All Smartphone Consumers vs.Recent SmartphoneAcquirers (3Mo)*. Retrieved February 4, 2012, from

<http://blog.nielsen.com/nielsenwire/consumer/more-us-consumers-choosing-smartphones-as-apple-closes-the-gap-on-android/>

Sabastien provost. (2011). *Anti SMS Bomber Apps*. Retrieved February 5, 2012, from

https://play.google.com/store/apps/details?id=com.anti.sms.pro&feature=search_result#?t=W251bGwsMSwxLDEsImNvbS5hbnRpLnNtcy5wcm8iXQ..

Tsvetan Nachev, (2011). *SMS Filter Apps*. Retrieved February 6, 2012, from

https://play.google.com/store/apps/details?id=com.nachev.apps.smsfilter&feature=search_result#?t=W251bGwsMSwxLDEsImNvbS5uYWNoZXYuYXBwcy5zbXNmaWx0ZXIiXQ..



ภาคผนวก



ภาคผนวก ก

ชุดรายชื่อและเลขหมายตัวอย่างบนสมุดโทรศัพท์

ตารางที่ ก.1 ชุดข้อมูลรายชื่อและเลขหมายบนสมุดโทรศัพท์ตัวอย่าง

ลำดับ	ชื่อ	เบอร์โทร	ลำดับ	ชื่อ	เบอร์โทร
1		1111	35	Miss.Test10	0865555563
2		0801111111	36	Miss.Test11	0865555564
3		0882222222	37	Miss.Test12	0865555565
4		0803333333	38	Miss.Test13	0865555566
5		0804444444	39	Miss.Test14	0865555567
6		0873333333	40	นายทดสอบ1	0882951628
		0873334444	41	นายทดสอบ2	0882951629
		0873335555	42	นายทดสอบ3	0882951630
7		0873336666	43	นายทดสอบ4	0882951631
		0873337777	44	นายทดสอบ5	0882951632
		0873338888	45	นายทดสอบ6	0882951633
		0873339999	46	นายทดสอบนามสกุล7	0882951634
8	CAT Contact Center	1322	47	##นายทดสอบ8	0882951635
9	DTAC call center	1678	48	_@_นายทดสอบ9	0882951636
10	Qwerty	50069	49	นายทดสอบ10	0882951637
11	Qwerty6	112233	50	นายทดสอบ11	0882951638
12	Qwerty7	4802777	51	นายทดสอบ12	0882951639
13	Qwerty8	08642010	52	นายทดสอบ13	0882951640
14	Klom Kub Kum Rok	025903333	53	นายทดสอบ14	0882951641
15	my Contact Center	024012222	54	นายทดสอบ15	0882951642
16	Mr.wit	0864201035	55	นายทดสอบ16	0882951643
17	Mr.Test1	0819999999	56	นายทดสอบ17	0882951644
18	Mr.Test2	0820000000	57	นายทดสอบ18	0882951645
19	Mr.Test3	0820000001	58	นายทดสอบ19	0882951646
20	Mr.Test4	0820000002	59	นายทดสอบ20	0882951647
21	Mr.Test5	0820000003	60	Khun android 1	0894567890
22	Mr.Test6	0820000004	61	Khun android 2	0894567891
23	Mr.Test7	0820000005	62	Khun android 3	0894567892
24	Mr.Test8	0820000006	63	Khun android 4	0894567893
25	Mr.Test9	0820000007	64	Khun android 5	0894567894
26	Mr.Test10	0820000008	65	Khun android 6	0894567895
27	Miss.Test2	0865555555	66	Khun android 7	0894567896
28	Miss.Test3	0865555556	67	Khun android 8	0894567897
29	Miss.Test4	0865555557	68	Khun android 9	0894567898
30	Miss.Test5	0865555558	69	Khun android 10	0894567899
31	Miss.Test6	0865555559	70	Khun android 11	0894567900
32	Miss.Test7	0865555560	71	Khun android 12	0894567901
33	Miss.Test8	0865555561	72	Khun android 13	0894567902
34	Miss.Test9	0865555562	73	Khun android 14	0894567903

ตารางที่ ก.1 (ต่อ) ชุดข้อมูลรายชื่อและเลขหมายบนสมุดโทรศัพท์ตัวอย่าง

ลำดับ	ชื่อ	เบอร์โทร	ลำดับ	ชื่อ	เบอร์โทร
74	Khun android 15	0894567904			0864444444
75	Khun android 16	0894567905	106	Mix2	0864445555
76	Khun android 17	0894567906			0864446666
77	Khun android 18	0894567907			0864447777
78	Khun android 19	0894567908	107	Mix3	0864448888
79	Khun android 20	0894567909			0864449999
80	Khun android 21	0894567910			
81	Khun android 22	0894567911			
82	Khun android 23	0894567912			
83	Khun android 24	0894567913			
84	Khun android 25	0894567914			
85	Khun android 26	0894567915			
86	Khun android 27	0894567916			
87	Khun android 28	0894567917			
88	Khun android 29	0894567918			
89	Khun android 30	0894567919			
90	mobile	0845557788			
91	mobile2	0845557789			
92	mobile3	0845557790			
93	work1	0845557791			
94	work2	0845557792			
95	work3	0845557793			
96	other1	0845557794			
97	other2	0845557795			
98	other3	0845557796			
99	Home&Mobile1	0845557797			
		0845557798			
100	Home&Mobile2	0845557799			
		0845557800			
101	Mobile&Work1	0845557801			
		0845557802			
102	Mobile&Work2	0845557803			
		0845557804			
103	Work&other1	0845557805			
		0845557806			
104	Work&other2	0845557807			
		0845557808			
105	Mix1	0867777777			
		0867778888			
		0867779999			



ภาคผนวก ข
ผลการกรองข้อความสั้นที่ไม่มีรายชื่ออยู่ในสมุดโทรศัพท์

ตารางที่ ข.1 ผลการกรองข้อความสั้นที่ไม่มีรายชื่ออยู่ในสมุดโทรศัพท์

MSG_ID	PHONE_NUMBER	MESSAGE
1339735146215	5009	Test Sending SMS
1339735505540	2222	Test Sending SMS 4
1339735987827	500699	Test Sending SMS 14
1339736859346	48027778	Test Sending SMS 21
1339750633410	+6680867778888	Test send no have in contact
1339751105831	189	Call center for complaints
1339751141141	189	Call center for complaints 2
1339751148622	189	Call center for complaints 3
1339751203118	189	Call center for complaints 4
1339751208163	189	Call center for complaints 5
1339751265052	1166	The Office of Consumer Protection 1
1339751273259	1166	The Office of Consumer Protection 2
1339751280359	1166	The Office of Consumer Protection 3
1339751286555	1166	The Office of Consumer Protection 4
1339751291957	1166	The Office of Consumer Protection 5
1339751746937	50062	VAS Service of CAT CDMA Service 1
1339751754639	50062	VAS Service of CAT CDMA Service 2
1339751759467	50062	VAS Service of CAT CDMA Service 3
1339751763977	50062	VAS Service of CAT CDMA Service 4
1339751768773	50062	VAS Service of CAT CDMA Service 5
1339751859246	025137070	Department of Export Promotion Center 1
1339751990091	025137070	Department of Export Promotion Center 2
1339751995583	025137070	Department of Export Promotion Center 3
1339752000824	025137070	Department of Export Promotion Center 4
1339752007285	025137070	Department of Export Promotion Center 5
1339752143349	0816678899	Sent from the subscriber. That is not listed in the phone book 1
1339752150120	0816678899	Sent from the subscriber. That is not listed in the phone book 2
1339752156061	0816678899	Sent from the subscriber. That is not listed in the phone book 3
1339752162265	0816678899	Sent from the subscriber. That is not listed in the phone book 4
1339752167874	0816678899	Sent from the subscriber. That is not listed in the phone book 5
1339752498312	0909990009	Sent from the subscriber. That is not listed in the phone book. And for whom more than 10 digits 1
1339752515808	0909990009	Sent from the subscriber. That is not listed in the phone book. And for whom more than 10 digits 2
1339752521113	0909990009	Sent from the subscriber. That is not listed in the phone book. And for whom more than 10 digits 3
1339752656858	0909990009	Sent from the subscriber. That is not listed in the phone book. And for whom more than 10 digits 4
1339752662681	0909990009	Sent from the subscriber. That is not listed in the phone book. And for whom more than 10 digits 5
1339768963707	+66810160171160171	Sent from the subscriber. That is not listed in the phone book. And for whom more than 10 digits 6
1339768971842	+66810160171160171	Sent from the subscriber. That is not listed in the phone book. And for whom more than 10 digits 7
1339768978097	+66810160171160171	Sent from the subscriber. That is not listed in the phone book. And for whom more than 10 digits 8
1339768984591	+66810160171160171	Sent from the subscriber. That is not listed in the phone book. And for whom more than 10 digits 9
1339768991273	+66810160171160171	Sent from the subscriber. That is not listed in the phone book. And for whom more than 10 digits 10
1339769859342	500000	May the distance between us would not change your mind 1
1339769870100	500000	May the distance between us would not change your mind 2
1339769877092	500000	May the distance between us would not change your mind 3
1339769883711	500000	May the distance between us would not change your mind 4
1339769888716	500000	May the distance between us would not change your mind 5
1339769897423	500000	May the distance between us would not change your mind 6
1339769901768	500000	May the distance between us would not change your mind 7
1339769906196	500000	May the distance between us would not change your mind 8
1339769914689	500000	May the distance between us would not change your mind 9
1339769920131	500000	May the distance between us would not change your mind 10
1339770502382	500000	Free! Pure Food dummy call 02-456-2134
1339770525191	500000	Free! Pure Food dummy call 02-456-2135
1339770530878	500000	Free! Pure Food dummy call 02-456-2136
1339770537667	500000	Free! Pure Food dummy call 02-456-2137
1339770769480	+668171234567790	The intensive courses www.pec9.com fresh and Grade 4 - 6 m starting 1 Jun 55

ตารางที่ ข.1 (ต่อ) ผลการกรองข้อความสั้นที่ไม่มีรายชื่ออยู่ในสมุดโทรศัพท์

MSG_ID	PHONE_NUMBER	MESSAGE
1339770831212	+6681712876777	Chance to sell the crabs. Round the world just to get my daily press * 298 * 193 # (4 / day) 022 076 888
1339770850029	022076888	Chance to sell the crabs. Round the world just to get my daily press * 298 * 193 # (4 / day) 022 076 888
1339770864800	022076888	Chance to sell the crabs. Round the world just to get my daily press * 298 * 193 # (4 / day) 022 076 888
1339770870752	022076888	Chance to sell the crabs. Round the world just to get my daily press * 298 * 193 # (4 / day) 022 076 888
1339770875169	022076888	Chance to sell the crabs. Round the world just to get my daily press * 298 * 193 # (4 / day) 022 076 888
1339770882532	022076888	Chance to sell the crabs. Round the world just to get my daily press * 298 * 193 # (4 / day) 022 076 888
1339770954309	027302424	Good!! Free for 15 days plus the Most Popular Mono * Please call 45223111110/027302424
1339770962513	027302424	Good!! Free for 15 days plus the Most Popular Mono * Please call 45223111110_027302424
1339770971831	027302424	Good!! Free for 15 days plus the Most Popular Mono * Please call 45223111110/027302424
1339770976432	027302424	Good!! Free for 15 days plus the Most Popular Mono * Please call 45223111110/027302424
1339770979322	027302424	Good!! Free for 15 days plus the Most Popular Mono * Please call 45223111110/027302424
1339770982856	027302424	Good!! Free for 15 days plus the Most Popular Mono * Please call 45223111110/027302424
1339771079961	50100	Send me a voice VoiceSMS free to call 50100 31 Mar 56 (regional)
1339771121328	50100	ReSend me a voice VoiceSMS free to call 50100 31 Mar 56 (regional)
1339771126867	50100	ReSend me a voice VoiceSMS free to call 50100 31 Mar 56 (regional)
1339771130285	50100	ReSend me a voice VoiceSMS free to call 50100 31 Mar 56 (regional)
1339771139564	50100	ReSend me a voice VoiceSMS free to call 50100 31 Mar 56 (regional)
1339771142230	50100	ReSend me a voice VoiceSMS free to call 50100 31 Mar 56 (regional)
1339771196454	028848692	I race to win the gold weight and cross unlimited number of unit names Pinklao 028848692
1339771261868	028848692	I race to win the gold weight and cross unlimited number of unit names Pinklao 028848692
1339771300950	028848692	I race to win the gold weight and cross unlimited number of unit names Pinklao 028848692
1339771391021	+66845475844030303	Free Bag of 350 B show ticket booth TMBMoneyExpo 8-11 Jul 55_ Ready Cash
1339771397971	+66845475844030303	Free Bag of 350 B show ticket booth TMBMoneyExpo 8-11 Jul 55_ Ready Cash
1339771404054	+66845475844030303	Free Bag of 350 B show ticket booth TMBMoneyExpo 8-11 Jul 55_ Ready Cash
1339771407609	+66845475844030303	Free Bag of 350 B show ticket booth TMBMoneyExpo 8-11 Jul 55_ Ready Cash
1339771409746	+66845475844030303	Free Bag of 350 B show ticket booth TMBMoneyExpo 8-11 Jul 55_ Ready Cash
1339771412423	+66845475844030303	Free Bag of 350 B show ticket booth TMBMoneyExpo 8-11 Jul 55_ Ready Cash
1339771416638	+66845475844030303	Free Bag of 350 B show ticket booth TMBMoneyExpo 8-11 Jul 55_ Ready Cash
1339771499310	1900190065	Inside! Precision: What's your time to. Look decide to call 1900190065
1339771503472	1900190065	Inside! Precision: What's your time to. Look decide to call 1900190065
1339771506969	1900190065	Inside! Precision: What's your time to. Look decide to call 1900190065
1339771513617	1900190065	Inside! !Precision: What's your time to. Look decide to call 1900190065
1339771514996	1900190065	Inside! !Precision: What's your time to. Look decide to call 1900190065
1339771519328	1900190065	Inside! !Precision: What's your time to. Look decide to call 1900190065
1339771567855	1375	More than one year 0% interest free transfer of a mortgage: Call 1375
1339771575086	1375	More than one year 0% interest free transfer of a mortgage: Call 1375
1339771579221	1375	More than one year 0% interest free transfer of a mortgage: Call 1375
1339771842703	1900190088	May I have your monthly fortune or paving round the world have the answers, press * 4988 very accurately
1339771867506	1900190088	May I have your monthly fortune or paving round the world have the answers, press * 4988
1339771890285	1900190088	May I have your monthly fortune or paving round the world have the answers, press * 4988
1339771948824	+6678901122334455	I see from CD 100 Rock Uncensored BKK and get ready for signature
1339771960924	+6678901122334455	I see from Rock Uncensored BKK and get ready for signature
1339771986276	+6678901122334455	I see from Rock Uncensored BKK and get ready for signature
1339772146183	1234	Running
1339772163798	1234	Please!!Running
1339772780748	0882951624	qwertyuioplkjhgfdsazxcvbnm
1339772892066	088773456	DPU Love Love Hero

ประวัติผู้เขียน

ชื่อ-นามสกุล

สุรวิษณุ ศรีไทย

วัน เดือน ปีเกิด

24 มกราคม 2523

ประวัติการศึกษา

สำเร็จการศึกษาระดับปริญญาตรี จากคณะ
วิศวกรรมศาสตร์ สาขาวิชาวิศวกรรมไฟฟ้า
มหาวิทยาลัยเทคโนโลยีมหานคร
ปีการศึกษา 2546

ตำแหน่งและสถานที่ทำงานปัจจุบัน

วิศวกรระดับ 6
ส่วนผลิตภัณฑ์มัลติมีเดีย
ฝ่ายพัฒนาผลิตภัณฑ์สื่อสารไร้สาย
บริษัท กสท โทรคมนาคม จำกัด (มหาชน)