อภินันทนาการ

สำนักหอสมุด

โปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉิน บนระบบปฏิบัติการแอนดรอยด์

APPLICATON FOR EMERGENCY RESCUE ON ANDROID OPERATING SYSTEM

นายอัฐพล แตงชัยภูมิ รหัส 52363172
นายกำภู สุจริตจันทร์ รหัส 52363447

สำนักหอสมุด มหาวิทยาลัยนเรศวร
26 ก.ย. 2560
เลขทะเบียน 1718642
เลขเรียกหนังสือ ว
0534ป
2559

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
ปีการศึกษา 2558

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน	โปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉินบนระบบปฏิบัติการแอนดรอยด์
ผู้ดำเนินโครงงาน	นายอัฐพล แตงชัยภูมิ รหัส 52363172
	นายกำภู สุจริตจันทร์ รหัส 52362441
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. พรพิศุทธิ์ วรจิรันตน์
อาจารย์ที่ปรึกษาร่วม	ผู้ช่วยศาสตราจารย์ ดร.สุวรรณา ถาวรรุ่งโรจน์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2558

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

.....ประธานกรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. พรพิศุทธิ์ วรจิรันตน์)

.....กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. พนัส นัถฤทธิ์)

.....กรรมการ

(อาจารย์ภาณุพงศ์ สอนคม)

ชื่อหัวข้อโครงงาน	โปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉิน บนระบบปฏิบัติการแอนดรอยด์
ผู้ดำเนินโครงงาน	นายอัฐพล แตงชัยภูมิ รหัส 52363172
	นายกำภู สุจริตจันทร์ รหัส 52362441
ที่ปรึกษาโครงงาน	ผู้ช่วยศาสตราจารย์ ดร.พรพิศุทธิ์ วรจิรันตน์
	ผู้ช่วยศาสตราจารย์ ดร.สุวรรณา ถาวรรุ่งโรจน์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2558

.....

บทคัดย่อ

การเจ็บป่วยฉุกเฉินเป็นสาเหตุของการเสียชีวิตสูงสุดในเกือบทุกประเทศ เกิดได้จากหลายโรค เช่น โรคหัวใจ อุบัติเหตุ อุบัติภัย ภัยพิบัติ และการได้รับพิษ จากผลการศึกษาพบว่า ผู้ป่วยจากอุบัติเหตุที่รุนแรง ร้อยละ 60 – 80 ของการเสียชีวิต เกิดจากการบาดเจ็บที่เกิดขึ้นก่อนถึงโรงพยาบาล การเกิดเหตุนั้นผู้คนส่วนใหญ่ไม่สามารถขอความช่วยเหลือด้วยการพูดโทรศัพท์ได้

จากที่กล่าวมาข้างต้น ทางคณะผู้วิจัยได้เล็งเห็นความสำคัญของการช่วยเหลือผู้ประสบเหตุที่รวดเร็วจะทำให้ผู้ป่วยได้รับการช่วยเหลือเบื้องต้นและถูกนำส่งโรงพยาบาลได้รวดเร็วยิ่งขึ้น จึงได้สร้างโปรแกรมประยุกต์ขอความช่วยเหลือในภาวะฉุกเฉินขึ้น การพัฒนาแอพพลิเคชั่นบนมือถือ ได้มีการนำเอาระบบระบุตำแหน่งมาใช้ประโยชน์เพื่อระบุตำแหน่งของผู้ขอความช่วยเหลือ เมื่อมีการตั้งค่าในโปรแกรมประยุกต์และมีการตอบรับการยืนยันหมายเลขโทรศัพท์ปลายทาง 5 หมายเลขแล้ว ระบบจะทำการเก็บค่าตำแหน่งปัจจุบันหรือตำแหน่งสุดท้ายที่สามารถระบุตำแหน่งได้ เมื่อต้องการขอความช่วยเหลือเพียงเข้าโปรแกรมประยุกต์แล้วกดส่งหรือใช้การพลิก-หงายเครื่อง จะทำให้ผู้รับ 5 หมายเลข หรือผู้ที่อยู่ในบริเวณใกล้เคียงสามารถเข้าไปช่วยเหลือได้รวดเร็วขึ้นและลดความเสี่ยงของการพิการและนำสู่การเสียชีวิตได้

Project Title	Application for emergency rescue On android operating system
Name	Mr.Auttapon Tangchaiyaphoom ID. 52363172
	Mr.Kampoo Sutjaritjan ID. 52362441
Project Advisor	Asst. Prof. Dr. Ponpisut Worrajiran
	Asst. Prof. Dr. Suwanna Tawonrungrojn
Major	Computer Engineering
Department	Electrical and Computer Engineering
Academic Year	2558

ABSTRACT

Emergency is a major cause of death in almost every country. It can cause from many diseases such as a heart disease, accident, disaster, and intoxication. Many studies have found that 60-80 percent of all deaths are from accidents and severe injuries before arriving to the hospital. Most of people were not in a state of calling for help by talking on phones.

As a result, it is important to gain an access to an emergency treatment and to be delivered to a hospital, as soon as possible. Accordingly, an objective of this project is to create a mobile phone application for people who may need help in emergency time. With the purpose of guiding the helpers to the application user, a GPS service is incorporated into the application, to assure fast access and decreasing a probability of death. After setting up the application, and the receivers accept the request for adding them in a list of receivers, the system will store the current or last position can be located. Then the user can ask for help by just clicking the application or use the Flip - Turn the mobile phone.

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ก

บทคัดย่อภาษาอังกฤษ ..... ข

สารบัญ ..... ค

สารบัญตาราง ..... จ

สารบัญรูป ..... ฉ

บทที่ 1 บทนำ 1

1.1 ที่มาและความสำคัญของโครงงาน 1

1.2 จุดประสงค์ของโครงงาน 4

1.3 ขอบเขตของโครงงาน 4

1.4 ขั้นตอนการดำเนินงาน 4

1.5 ขั้นตอนการดำเนินงาน 5

1.6 ผลที่คาดว่าจะได้รับ 6

1.7 งบประมาณของโครงงาน 6

บทที่ 2 หลักการและทฤษฎี 7

2.1 ระบบปฏิบัติการแอนดรอยด์ [2] 8

2.2 GPS และการระบุพิกัด [3] 10

2.3 โปรแกรมประยุกต์ "EMS1669" ช่วยเหลือผู้ป่วยฉุกเฉินของสถาบันการแพทย์ฉุกเฉิน 13

บทที่ 3 วิธีการดำเนินงาน 16

3.1 ข้อมูลที่ใช้ในการพัฒนาระบบ (Requirement Specifition) 17

3.2 การออกแบบระบบ (System Design) 18

สารบัญ (ต่อ)

หน้า

3.3 การออกแบบระบบฐานข้อมูล (Database Design) 23
3.4 การออกแบบหน้าตาแอพพลิเคชั่นบนมือถือ (User Interface) 26

บทที่ 4 ผลการทดลอง 55
4.1 กระบวนการทำงานของโปรแกรมประยุกต์ 55
4.2 การทดสอบการทำงานของโปรแกรมประยุกต์ 69

บทที่ 5 บทสรุปและข้อเสนอแนะ 87
5.1 สรุปผลการดำเนินงาน 87
5.2 ปัญหาและอุปสรรคที่พบในการดำเนินงาน 89
5.3 ข้อเสนอแนะและแนวทางในการพัฒนา 90

เอกสารอ้างอิง 91

ประวัติผู้ดำเนินโครงงาน 93

สารบัญตาราง

ตารางที่ หน้า

1.1 ตารางแสดงขั้นตอนการดำเนินงาน 5

3.1 Data Dictionary ของตารางข้อมูลส่วนตัวของสมาชิก (DatabaseUser) 24

3.2 Data Dictionary ของตารางตรวจสอบความพร้อมของระบบ (DatabaseValue) 25

3.3 Data Dictionary ของตารางของพิกัดตำแหน่ง (Location) 25

3.4 Data Dictionary ของตารางของเบอร์โทรศัพท์ที่ได้รับการยืนยันแล้ว (DatabaseAccept) 25

4.1 ตารางทดสอบการส่งข้อความขอความช่วยเหลือโดยการกดปุ่มส่งข้อความบริเวณภายในมหาวิทยาลัยนเรศวร 76

4.2 ตารางทดสอบการส่งข้อความขอความช่วยเหลือโดยการพลิก – หงาย บริเวณภายในมหาวิทยาลัยนเรศวร 80

4.3 ตารางทดสอบการใช้งานโปรแกรมประยุกต์โดยการดำเนินชีวิตประจำวันเป็นเวลา 1 สัปดาห์ 84

4.4 ตารางการทำกิจวัตรปกติ 8 ชั่วโมงทดสอบโดยการใช้ชีวิตประจำวันในช่วงเวลากลางวัน 85

4.5 ตารางการทำกิจวัตรปกติ 4 ชั่วโมงทดสอบโดยการใช้ชีวิตประจำวันในช่วงเวลากลางคืน 86

สารบัญรูป

รูปที่ หน้า

1.1 แสดงจำนวนผู้ป่วยฉุกเฉินที่จำแนกตามระดับความฉุกเฉิน ในปี พ.ศ. 2553 และ พ.ศ. 2554 1

1.2 แสดงแนวโน้มการปฏิบัติการฉุกเฉินทางบก ระหว่างปี 2553-2556 จำแนกตามระดับความรุนแรง 2

2.1 แสดงรุ่นพัฒนาของแอนดรอยด์ 9

2.2 ภาพแสดงการแบ่งองศาของละติจูด 10

2.3 ภาพแสดงการแบ่งองศาของลองจิจูด 11

2.4 แสดงองค์ประกอบของระบบดาวเทียม GPS 13

2.5 หน้าจอแสดงโปรแกรมประยุกต์ "EMS1669" ในโทรศัพท์ 13

2.6 หน้าจอแสดงหน้าแรกของโปรแกรมประยุกต์ "EMS1669" 14

2.7 หน้าจอแสดงหน้าเมนูของโปรแกรมประยุกต์ "EMS1669" 14

2.8 หน้าจอแสดงหน้าข้อมูลส่วนตัวเมื่อกดปุ่มบันทึกข้อมูลส่วนตัว 15

2.9 หน้าจอแสดงหน้า EMS 1669 เมื่อกดปุ่มเรียกรถพยาบาล 15

3.1 แสดงการวางแผนระบบการทำงานของระบบการส่งข้อมูลด้วยข้อความบนระบบปฏิบัติการ แอนดรอยด์ 16

3.2 แสดงถึง Conceptual Design ของระบบ 18

3.3 แสดง Use Case Diagram ของระบบ 19

3.4 แสดง Sequence Diagram ของการส่งข้อความแบบธรรมดา 20

3.5 แสดง Sequence Diagram ของระบบในการส่งข้อความด้วย Sensor (Send SMS By Accelerometer Sensor) 21

3.6 แสดง Scquence Diagram ของระบบในส่วนของผู้รับ (Receiver) 22

3.7 แสดงฐานข้อมูลในระบบ 23

3.8 แสดงการออกแบบโดยใช้โปรแกรม Eclipse SDK Version:4.2.1 26

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.9 แสดงการออกแบบหน้าเมนู (Home Page) 27

3.10 สถานะของระบบระบุตำแหน่ง (GPS) ที่สถานะเปิดอยู่ 28

3.11 สถานะของระบบระบุตำแหน่ง (GPS) ที่สถานะปิดอยู่ 28

3.12 สถานะความพร้อมในการใช้งานที่สถานะพร้อมใช้ 29

3.13 สถานะความพร้อมในการใช้งานที่สถานะไม่พร้อมใช้ 30

3.14 แสดงการทำงานเมื่อกดปุ่มหยุดการทำงาน 30

3.15 แสดงการทำงานเมื่อกดปุ่มตั้งค่า 31

3.16 แสดงการทำงานเมื่อกดปุ่มผู้รับ 32

3.17 ปุ่มสำหรับส่งข้อความโดยผู้ใช้เอง 33

3.18 แสดงการออกแบบหน้าข้อมูลส่วน (Profile Page) 34

3.19 ตัวอย่างข้อความที่ผู้รับได้รับเมื่อกดปุ่มส่งข้อความแก้ 36

3.20 แสดงการออกแบบหน้ารายชื่อของข้อความที่ถูกส่งเข้ามา (Received Listview Page) 37

3.21 แสดงการออกแบบหน้ารายละเอียดของข้อความ (Message Page) 38

3.22 แสดงการออกแบบหน้าแสดงแผนที่ (Map Page) 39

3.23 แสดงการออกแบบหน้าคู่มือการใช้ (User Manual Page) 40

3.24 แสดงการการออกแบบหน้าการยืนยันเบอร์โทรศัพท์ในฝั่งของผู้รับ 41

3.25 แสดงการออกแบบในการเขียนคำสั่งหยุดการทำงานของโปรแกรม 42

3.26 แสดงการออกแบบในการเขียนคำสั่งตรวจสอบ Checkbox 43

3.27 แสดงการออกแบบในการเขียนคำสั่งการทำงานของ Accelcrometer Sensor 44

3.28 แสดงการออกแบบในการเขียนคำสั่งเพื่อส่งค่าละติจูดและลองจิจูดจากถาดเข้าข้อความ 45

3.29 แสดงการออกแบบในการเขียนคำสั่งในการแก้ข้อความที่ส่งผิด 46

3.30 แสดงการออกแบบในการเขียนคำสั่งปุ่มบันทึก 47

3.31 แสดงการออกแบบในการเขียนคำสั่งเพื่อตรวจสอบช่องว่างในช่องกรอกเบอร์โทรศัพท์ 48

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.32 แสดงการออกแบบในการเขียนคำสั่งเพื่อทำการจัดเก็บข้อมูลลงบนฐานข้อมูลและเริ่มการทำงานบนพื้นหลัง 49
3.33 แสดงการออกแบบในการเขียนคำสั่งเมื่อต้องการส่งข้อมูลด้วยตัวเอง 50
3.34 แสดงการออกแบบในการเขียนคำสั่งเพื่อตรวจสอบสถานะของ ระบบพิกัดบอกตำแหน่ง (GPS) 51
3.35 แสดงการออกแบบในการเขียนคำสั่งเพื่อตรวจสอบข้อความจากถาดเข้าข้อความ 52
3.36 แสดงการออกแบบในการเขียนคำสั่งเพื่อเปิดแผนที่จาก Google map 53
3.37 แสดงการออกแบบในการเขียนคำสั่งตั้งค่าเสียงไซเรนเป็นระดับสูงสุดของโทรศัพท์ 54
4.1 หน้าจอเมนูบนโทรศัพท์ 55
4.2 หน้าการตั้งค่า 56
4.3 หน้าหน้าจอล็อค 56
4.4 หน้าเลือกการล็อคหน้าจอ 57
4.5 หน้าเมนูบนหน้าจอโทรศัพท์ 57
4.6 หน้าจอหลัก 58
4.7 หน้าแรก (Homepage) 59
4.8 หน้าตั้งค่า (Setting) เมื่อเริ่มการทำงานของโปรแกรมประยุกต์ครั้งแรก 59
4.9 หน้าตั้งค่า (Setting) เมื่อกรอกข้อมูลครบทุกช่อง 60
4.10 หน้าตอบรับการยืนยันในฝั่งผู้รับ 60
4.11 ข้อความการร้องขอการยืนยันที่ผู้รับได้รับ 61
4.12 ข้อความตอบรับการยืนยันที่ผู้รับได้ส่งกลับ 61
4.13 ปุ่มส่งข้อความโดยผู้ใช้ 62
4.14 เลือกส่งแบบพลิก – หงายในหน้าตั้งค่า 62
4.15 ถาดเข้าข้อความของผู้รับ 63

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.16 ถาดเข้าข้อความของผู้รับ 64
4.17 หน้าแรก (Homepage) 65
4.18 หน้าแสดงรายการข้อความของผู้รับ 65
4.19 หน้าแสดงตัวเลขตำแหน่งของผู้ขอความช่วยเหลือ (ละติจูดและลองจิจูด) 66
4.20 หน้าแสดงแผนที่จากตัวเลขตำแหน่งจากรูปที่ 4.19 66
4.21 ปุ่มตั้งค่า 67
4.22 ปุ่มส่งข้อความแก้ 68
4.23 ตัวอย่างข้อความที่ผู้รับได้รับเมื่อกดปุ่มส่งข้อความแก้ 68
4.24 ตัวอย่างข้อความที่ผู้รับได้รับจากผู้ต้องการขอความช่วยเหลือ 69
4.25 หน้าผู้รับ (Receiver) แสดงรายการข้อความของผู้ขอความช่วยเหลือ 70
4.26 หน้าผู้รับ (Receiver) แสดงตัวเลขตำแหน่งของผู้ขอความช่วยเหลือ (ละติจูดและลองจิจูด) 70
4.27 หน้าผู้รับ (Receiver) แสดงแผนที่จากตัวเลขตำแหน่งจากรูปที่ 4.26 71
4.28 แสดงแผนที่จาก Google map ก่อนทำการขอความช่วยเหลืออยู่บริเวณหน้าอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร 72
4.29 แสดงสถานะพร้อมก่อนทำการขอความช่วยเหลือ อยู่บริเวณหน้าอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร 72
4.30 แสดงข้อความการขอความช่วยเหลือ และแผนที่ที่ได้จากโปรแกรม อยู่บริเวณหน้าอาคาร วิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร 73
4.31 แสดงแผนที่จาก Google map ก่อนทำการขอความช่วยเหลืออยู่บริเวณด้านในอาคาร วิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร 74
4.32 แสดงสถานะพร้อมก่อนทำการขอความช่วยเหลือ อยู่บริเวณภายในอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร เมื่อไม่มีสัญญาณระบบระบุตำแหน่ง 74

สารบัญรูป (ต่อ)

รูปที่ หน้า

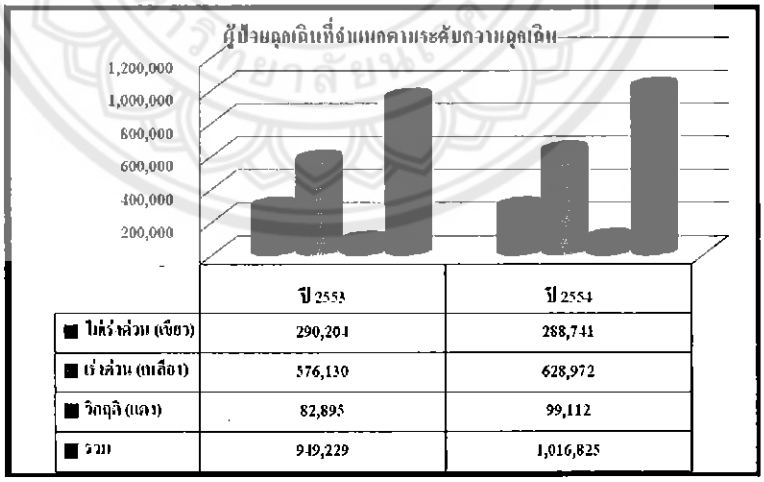
4.33 แสดงแผนที่เมื่อทำการขอความช่วยเหลืออยู่บริเวณภายในอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร เมื่อไม่มีสัญญาณระบบระบุตำแหน่ง.....75

บทที่ 1

บทนำ

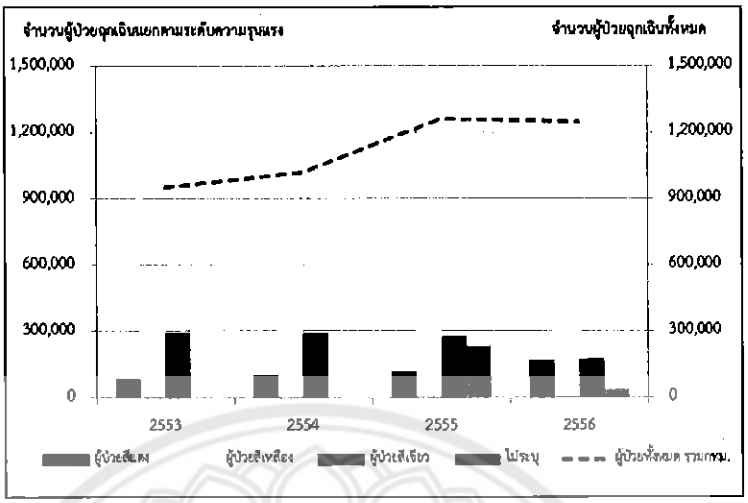
1.1 ที่มาและความสำคัญของโครงงาน

ภาวะฉุกเฉิน คือ เหตุการณ์ที่เกิดขึ้นอย่างกะทันหันและอาจก่อให้เกิดการสูญเสียชีวิต หรือทรัพย์สิน [1] ซึ่งการเจ็บป่วยฉุกเฉินเป็นสาเหตุของการเสียชีวิตสูงสุดในเกือบทุกประเทศ เกิดได้จากหลายโรค เช่น โรคหัวใจ อุบัติเหตุ อุบัติภัย ภัยพิบัติ และการได้รับพิษ จากผลการศึกษาพบว่า ผู้ป่วยจากอุบัติเหตุที่รุนแรง ร้อยละ 60 – 80 ของการเสียชีวิต เกิดจากการบาดเจ็บที่เกิดขึ้นก่อนถึงโรงพยาบาล [2] โดยพบว่าการเกิดภาวะฉุกเฉินมีแนวโน้มที่เพิ่มสูงขึ้น อ้างอิงจากข้อมูลของสถาบันการแพทย์ฉุกเฉินแห่งชาติ ในปี พ.ศ.2553 มีจำนวนผู้ประสบภาวะฉุกเฉินจำนวน 949,229 คน และในปี พ.ศ.2554 มีจำนวน 1,016,825 คน [3] แสดงให้เห็นว่าผู้ประสบภาวะฉุกเฉิน มีจำนวนที่เพิ่มสูงขึ้น และมีแนวโน้มที่จะเพิ่มขึ้นในปีถัดไป ในกรณีที่ผู้ประสบภาวะฉุกเฉินได้รับการช่วยเหลือที่ล่าช้าส่งผลให้อัตราการเสียชีวิตจากภาวะฉุกเฉินเพิ่มขึ้น



	ปี 2553	ปี 2554
ไม่รีบด่วน (เขียว)	290,204	288,741
เร่งด่วน (เหลือง)	576,130	628,972
วิกฤติ (แดง)	82,895	99,112
รวม	949,229	1,016,825

รูปที่ 1.1 แสดงจำนวนผู้ป่วยฉุกเฉินที่จำแนกตามระดับความฉุกเฉิน ในปี พ.ศ. 2553 และ พ.ศ. 2554



รูปที่ 1.2 แสดงแนวโน้มการปฏิบัติการฉุกเฉินทางบก ระหว่างปี 2553-2556 จำแนกตามระดับความรุนแรง

จากอุบัติการณ์ข้างต้นแสดงให้เห็นว่า ในแต่ละปีมีผู้ประสบภาวะฉุกเฉินจำนวนมากและมีแนวโน้มสูงขึ้นเรื่อย ๆ ในปีถัดไป จึงเป็นสถานการณ์ที่ควรติดตามและตระหนักถึงวิธีการลดอัตราความเสี่ยงของการเสียชีวิตจากภาวะฉุกเฉิน อย่างเช่น ภาวะกล้ามเนื้อหัวใจขาดเลือด (Myocardial Ischemia) ซึ่งเป็นภาวะที่เกิดขึ้น เมื่อปริมาณออกซิเจนในเลือดไม่เพียงพอต่อความต้องการของกล้ามเนื้อหัวใจ เมื่อกล้ามเนื้อหัวใจขาดเลือดจะเกิดอาการแสดงที่สำคัญ คือ อาการเจ็บแน่นหน้าอกที่รุนแรงมาก และหากขาดเลือดเป็นเวลานานจะทำให้เกิดการตายของกล้ามเนื้อหัวใจซึ่งไม่สามารถกลับสู่สภาพเดิมได้ แม้จะให้การรักษาอย่างเต็มที่แล้วก็ตาม [4] สถานการณ์การเข้าถึงการรักษาภาวะกล้ามเนื้อหัวใจขาดเลือดเฉียบพลันใช้เวลานาน ทำให้มีอัตราการเสียชีวิตสูงมาก จึงเป็นเหตุผลที่ผู้ประสบเหตุฉุกเฉินเหล่านี้จำเป็นต้องได้รับการช่วยเหลืออย่างรวดเร็ว นอกจากนี้ยังมีภาวะฉุกเฉินอื่น ๆ ได้แก่ อุบัติเหตุทางถนน คนหลงป่า เป็นต้นดังนั้น ในกรณีที่เกิดอุบัติเหตุหรือเกิดการบาดเจ็บ หากได้รับการช่วยเหลือล่าช้า อาจทำให้เกิดการเสียชีวิตและทุพพลภาพได้ จากเหตุการณ์ดังกล่าวทำให้มีผู้พยายามพัฒนาอุปกรณ์เพื่อนำมาใช้กับผู้ป่วยที่มีโอกาสตกอยู่ในภาวะฉุกเฉิน ในปี พ.ศ.2555 งานวิจัยของกมล บุยศ์เพชรและคณะ ได้พัฒนาระบบแจ้งเตือนและระบบเซ็นเซอร์วัดความเร่ง 3 แกน [5] ตรวจจับความเร่งและวัดแรงสั่นสะเทือนของอุปกรณ์ที่ติดอยู่กับตัวผู้สูงอายุ โดยสามารถตรวจจับความเร่งที่ผิดปกติในขณะเกิดอุบัติเหตุ ซึ่งจะเกิดความเร่งมากขึ้นในช่วงขณะหนึ่งนอกจากนั้นยัง

ตรวจจับการสั่นสะเทือนของอุปกรณ์เมื่อเกิดการกระแทกจากอุบัติเหตุ แล้วนำไปประมวลผลโดยใช้คอนโทรลเลอร์ (Controller) เป็นตัวตัดสินใจ ถ้าผู้สูงอายุล้มก็จะมีการแจ้งเตือนโดยใช้เสียงเตือนเพื่อให้คนที่อยู่ใกล้ ๆ มาช่วยพร้อมแจ้งตำแหน่งด้วยพิกัดจาก GPS ส่งข้อความ SMS ไปยังมือถือของผู้ดูแล ซึ่งยังมีข้อจำกัดเรื่องตัวอุปกรณ์มีขนาดใหญ่ทำให้พกพาไม่สะดวก ในปี พ.ศ.2554 งานวิจัยของนายรัฐภูมิ ตู้จินดาและคณะ ได้พัฒนาระบบแผนที่แจ้งเตือนเหตุฉุกเฉินส่วนตัว โดยการแจ้งทางโทรศัพท์ของบุคคลทั่วไปผ่านทางสถานีวิทยุ [6] เพื่อแจ้งเตือนผู้ใช้งานในกรณีที่เกิดเหตุร้ายในบริเวณที่ผู้ใช้งานสนใจ ซึ่งเมื่อเกิดมีเหตุการณ์ฉุกเฉินเกิดขึ้นในบริเวณที่ผู้ใช้กำหนด ระบบจะแจ้งเตือนไปยังผู้ใช้ผ่านอีเมล์และ โทรศัพท์มือถือ แต่งานวิจัยนี้ยังมีขอบเขตพื้นที่ในการแจ้งเตือนในวงแคบ

จากที่กล่าวมาข้างต้น ทางคณะผู้วิจัยได้เล็งเห็นความสำคัญของการช่วยเหลือผู้ประสบเหตุที่รวดเร็ว จะทำให้ผู้ป่วยได้รับการช่วยฟื้นคืนชีพเบื้องต้นและถูกนำส่งโรงพยาบาลได้รวดเร็วยิ่งขึ้น จึงได้สร้างโปรแกรมประยุกต์ขอความช่วยเหลือในภาวะฉุกเฉินขึ้น เพื่อระบุตำแหน่งของผู้ประสบภาวะฉุกเฉิน ทำให้สามารถเข้าไปช่วยเหลือได้รวดเร็วขึ้น ลดความเสี่ยงของการพิการและนำสู่การเสียชีวิตได้

1.2 จุดประสงค์ของโครงงาน

1.2.1 เพื่อสร้างโปรแกรมประยุกต์ขอความช่วยเหลือฉุกเฉินบนโทรศัพท์มือถือแบบสมาร์ทโฟน

1.2.2 โปรแกรมประยุกต์ออกแบบเพื่อใช้ในกรณีฉุกเฉินเท่านั้น

1.3 ขอบเขตของโครงงาน

1.3.1 สามารถใส่หมายเลขโทรศัพท์สูงสุดของผู้รับข้อความได้ 5 หมายเลข

1.3.2 โปรแกรมประยุกต์นี้จะมีการส่งเสียงเพื่อให้ผู้ที่อยู่ในบริเวณใกล้เคียงมาช่วยเหลือได้พร้อมๆกับมีการส่ง

1.3.3 ข้อความในลักษณะของ SMS ไปยังหมายเลขปลายทางโดยระบุตำแหน่งละติจูดและลองจิจูดของผู้ใช้งานในสถานที่ที่มีการใช้งานจะต้องเป็นบริเวณที่มีสัญญาณโทรศัพท์เท่านั้น

1.3.4 แสดงตำแหน่งของผู้ส่งข้อความบนแผนที่จากละติจูดและลองจิจูดบนเครื่องของผู้รับส่งตำแหน่งสุดท้ายที่เครื่องสามารถบันทึกสัญญาณ GPS ได้

1.3.5 เลขหมายปลายทางต้องใส่รหัสประเทศ เช่น ประเทศไทยรหัสประเทศคือ +66 ยกตัวอย่างหมายเลขปลายทางเบอร์ 08 1234 5678 จะเปลี่ยนเป็น +668 1234 5678

1.4 ขั้นตอนการดำเนินงาน

1.4.1 วางแผนการดำเนินโครงงานและแบ่งหน้าที่รับผิดชอบ

1.4.2 ศึกษาค้นคว้าข้อมูลเกี่ยวกับระบบปฏิบัติการแอนดรอยด์

1.4.3 ออกแบบและสร้างโปรแกรมขอความช่วยเหลือฉุกเฉิน

1.4.4 ดำเนินการเขียนโปรแกรม

1.4.5 ทดสอบการทำงานของโปรแกรม

1.4.6 แก้ไขข้อผิดพลาดและเพิ่มเติมรายละเอียดปลีกย่อย

1.4.7 สรุปผลการทดลองและจัดทำรูปเล่มโครงงาน

1.5 ขั้นตอนการดำเนินงาน

ตารางที่ 1.1 ตารางแสดงขั้นตอนการดำเนินงาน

รายการ	พ.ศ. 2557								
	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.
1. วางแผนการดำเนินโครงงาน									
2. รวบรวมศึกษาค้นคว้าข้อมูล									
3. ออกแบบโครงสร้าง									
4. ดำเนินการเขียนโปรแกรม									
5. ทดสอบการทำงาน									
6. แก้ไขข้อผิดพลาดและเพิ่มเติมส่วนปลีกย่อย									
7. สรุปผลการทดลองและจัดทำรูปเล่มโครงงาน									

1.6 ผลที่คาดว่าจะได้รับ

1.6.1 สามารถนำโปรแกรมขอความช่วยเหลือฉุกเฉินไปใช้งานได้จริง

1.6.2 สามารถสร้างกระบวนการคิด วิเคราะห์ระบบได้อย่างเป็นระเบียบ

1.6.3 สามารถนำผลงานไปประยุกต์ใช้งานกับระบบปฏิบัติการอื่น ๆ ได้

1.7 งบประมาณของโครงงาน

1.7.1	ค่าเอกสารและการเข้าเล่ม	1000 บาท
1.7.2	ค่าอุปกรณ์ไฟฟ้าและอิเลคทรอนิกส์	1000 บาท
	รวมทั้งสิ้น	2000 บาท

มหาวิทยาลัยนเรศวร

บทที่ 2
หลักการและทฤษฎี

ในภาวะที่ผู้ประสบเหตุได้รับบาดเจ็บจากอุบัติเหตุหรือมีอาการเจ็บป่วยจากโรคประจำตัว จำเป็นต้องได้รับการช่วยเหลือและนำส่งโรงพยาบาลอย่างเร่งด่วน เพื่อลดอัตราการเสียชีวิตจากการบาดเจ็บที่รุนแรง หรือจากภาวะแทรกซ้อนที่อันตราย จากการศึกษาของสมาคมแพทย์โรคหัวใจประเทศสหรัฐอเมริกา พบว่าสมองของคนไข้ที่หัวใจหยุดเต้นจะตายภายใน 4–6 นาที ถ้าไม่ได้รับการช่วยเหลือ รวมทั้งอัตราการรอดชีวิตจะลดลงร้อยละ 7-10 ทุก ๆ 1 นาทีที่ไม่ได้รับการช่วยเหลือขั้นสูง ซึ่งอัตราการรอดชีวิตจะลดลงอย่างมากถ้าการช่วยเหลือช้ามากกว่า 10 นาที [7] ดังนั้น ระยะเวลาในการเข้าถึงตัวผู้ป่วยเป็นปัจจัยหนึ่งที่ทำให้ผู้ป่วยมีอัตราการรอดชีวิตสูงขึ้น ตัวอย่างเช่น การเกิดอุบัติเหตุทางถนน ทำให้ผู้ป่วยมีการเสียเลือด ถ้าผู้ป่วยมีการเสียเลือดมากถึงร้อยละ 15 หรือประมาณ 600 มิลลิลิตร ใน 4,000 มิลลิลิตร จะนำไปสู่ภาวะ Shock และถ้าปล่อยให้เสียเลือดต่อไปจะถึงแก่ความตายได้ [8] ในกรณีของกล้ามเนื้อหัวใจขาดเลือด หากเกิดการตายของกล้ามเนื้อหัวใจอย่างถาวร (Necrosis) จะไม่สามารถรักษาให้กล้ามเนื้อ หัวใจกลับสู่สภาพเดิมได้ แม้ว่าจะทำการรักษาอย่างเต็มที่ก็ตาม นอกจากนี้ยังมีโอกาสเกิดภาวะหัวใจห้องล่างเต้นแบบสั่นพลิ้ว (Ventricular Fibrillation) ซึ่งเกิดขึ้น ได้ภายใน 60 นาที หลังเกิดการตายของกล้ามเนื้อหัวใจและเป็นสาเหตุการเสียชีวิตที่สำคัญของภาวะกล้ามเนื้อหัวใจขาดเลือดเฉียบพลัน [9] ดังนั้น การช่วยเหลือผู้ป่วยภาวะดังกล่าวควรเป็นไปด้วยความรวดเร็ว โดยระยะเวลาตั้งแต่เริ่มมีอาการเจ็บแน่นหน้าอกจนกระทั่งได้รับยาละลายลิ่มเลือดหรือได้รับการขยายหลอดเลือดหัวใจ (Total Ischemic Time) ไม่ควรเกิน 120 นาที [10]

ระยะเวลาตั้งแต่ประสบเหตุจนเสียชีวิต ขึ้นอยู่กับผลกระทบที่เกิดจากการบาดเจ็บในภาวะฉุกเฉินนั้น หากการบาดเจ็บทำให้เกิดการฉีกขาดของสมอง ไขสันหลัง หัวใจหรือหลอดเลือดใหญ่ พบว่าร้อยละ 50 ของผู้ประสบเหตุมักเสียชีวิตทันทีหรือภายใน 2-3 นาที กรณีที่การบาดเจ็บหรือการเจ็บป่วย ส่งผลกระทบต่อการทำงานของระบบหายใจและระบบไหลเวียนโลหิต เช่น ภาวะพร่องออกซิเจน ภาวะเลือดไหลไม่หยุด พบว่าร้อยละ 30 ของผู้ประสบเหตุมักจะเสียชีวิตภายใน 2-3 ชั่วโมง

แรก ดังนั้นระยะเวลาในการเข้าถึงผู้ประสบเหตุก็มีความสำคัญต่ออัตราการรอดชีวิตของผู้ประสบเหตุด้วยเช่นกัน [11]

2.1 ระบบปฏิบัติการแอนดรอยด์ [2]

Android OS คือระบบปฏิบัติการสำหรับอุปกรณ์พกพา เช่น โทรศัพท์มือถือ, แท็บเล็ต, คอมพิวเตอร์และเน็ตบุ๊ก ที่ทำงานบนลินุกซ์ เคอร์เนล [12] เริ่มพัฒนาโดยบริษัทแอนดรอยด์ จากนั้นบริษัทแอนดรอยด์ถูกซื้อโดยกูเกิล และทางกูเกิลได้นำแอนดรอยด์ไปพัฒนาต่อ ส่วนด้านลิขสิทธิ์ของโค้ดแอนดรอยด์จะใช้ในลักษณะของซอฟต์แวร์เสรีหรือโอเพ่นซอร์ส (Open Source) ทำให้นักพัฒนาสามารถแก้ไข ดัดแปลงโค้ดแอนดรอยด์ได้อย่างอิสระ และที่สำคัญคือแจกฟรี สำหรับระบบปฏิบัติการแอนดรอยด์เป็นที่รู้จักต่อสาธารณชนเมื่อวันที่ 5 พฤศจิกายน 2550 และแอนดรอย์เวอร์ชั่น 1.0 ถูกปล่อยออกมาใช้งานอย่างเป็นทางการครั้งแรกเมื่อวันที่ 28 กันยายน 2551

ปัจจุบันระบบปฏิบัติการแอนดรอยด์ถูกนำไปใช้งานกันอย่างแพร่หลาย ผู้ผลิตมือถือยักษ์ใหญ่หลายเจ้าต่างพัฒนาและผลิตสมาร์ทโฟนที่ใช้ระบบปฏิบัติการแอนดรอยด์ออกมาวางจำหน่ายมากมาย ไม่ว่าจะเป็น Samsung, HTC, Motorola และ Sony เป็นต้น และทางกูเกิลซึ่งเป็นผู้พัฒนาแอนดรอยด์ก็ได้ผลิตสมาร์ทโฟนของตัวเองออกมาเช่นกัน เป็นสมาร์ทโฟนตระกูล Galaxy Nexus นั่นเอง

รุ่นพัฒนาของแอนดรอยด์จะใช้รหัสชื่อเป็นชื่อขนมหวาน โดยมีตัวอักษรขึ้นต้นเรียงลำดับกัน

รุ่น	ชื่อเล่น	ระดับเอพีไอ	ลินุกซ์ เคอร์เนล	เปิดตัว
1.0	-	1		23 กันยายน 2551
1.1	-	2		9 กุมภาพันธ์ 2552
1.5	Cupcake (คัพเค้ก)	3	2.6.27	30 เมษายน 2552[58]
1.6	Donut (โดนัท)	4	2.6.29	15 สิงหาคม 2552 (SDK)
2.0	Eclair (เอแคลร์)	5	2.6.29	26 ตุลาคม 2552[59]
2.0.1	Eclair (เอแคลร์)	6	2.6.29	3 ธันวาคม 2552
2.1	Eclair (เอแคลร์)	7	2.6.29	12 มกราคม 2553 (SDK) [60]
2.2	Froyo (โฟรซเซนโยเกิร์ต)	8	2.6.32[61]	20 พฤษภาคม 2553 (SDK)
2.3	Gingerbread (ขนมปังขิง)	9	2.6.35[62]	6 ธันวาคม 2553 (SDK)
2.3.3	Gingerbread (ขนมปังขิง)	10	2.6.35	9 กุมภาพันธ์ 2554 (SDK)
3.0	Honeycomb (รวงผึ้ง)	11	2.6.36[63]	22 กุมภาพันธ์ 2554 (SDK)
3.1	Honeycomb (รวงผึ้ง)	12	2.6.36	10 พฤษภาคม 2554 (SDK)
3.2	Honeycomb (รวงผึ้ง)	13	2.6.36	15 กรกฎาคม 2554 (SDK)
4.0	Ice Cream Sandwich (แซนด์วิชไอศกรีม)	14	3.0.1[64]	19 ตุลาคม 2554 (SDK)
4.0.3	Ice Cream Sandwich (แซนด์วิชไอศกรีม)	15		16 ธันวาคม 2554 (SDK)
4.1	Jelly Bean (เจลลีบีน)	16	3.0.31	28 มิถุนายน 2555
4.2	Jelly Bean (เจลลีบีน)	17	3.4.0	29 ตุลาคม 2555
4.3	Jelly Bean (เจลลีบีน)	18	3.4.0	24 กรกฎาคม 2556
4.4	KitKat (คิทแคท)	19	3.10	31 ตุลาคม 2556
4.4.1	KitKat (คิทแคท)	20	3.10	05 ธันวาคม 2556
4.4.2	KitKat (คิทแคท)	21	3.10	09 ธันวาคม 2556

รูปที่ 2.1 แสดงรุ่นพัฒนาของแอนดรอยด์

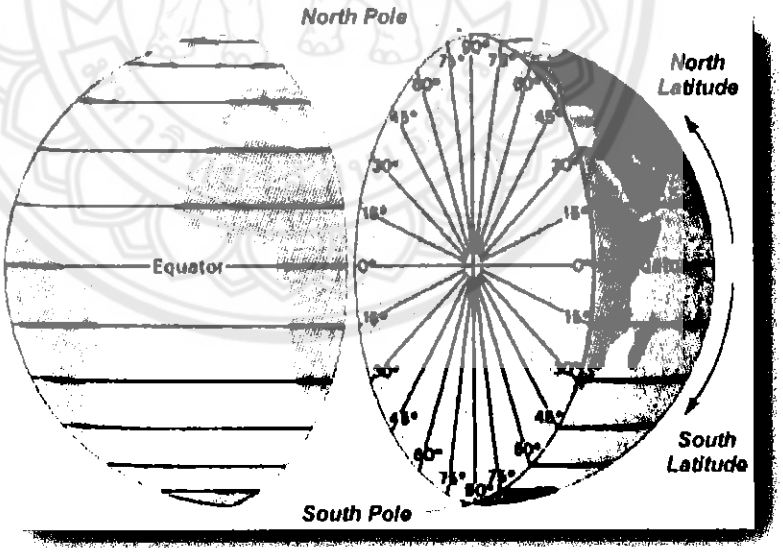
การสร้างโปรแกรมประยุกต์นี้ได้สร้างบนระบบปฏิบัติการแอนดรอยด์ (Android) ซึ่งเป็นระบบปฏิบัติการสำหรับอุปกรณ์พกพา เช่น โทรศัพท์มือถือแบบสมาร์ทโฟน แท็บเล็ต (Tablet) เป็นต้น โดยระบบปฏิบัติการแอนดรอยด์ถูกพัฒนามาจากระบบปฏิบัติการลินุกซ์ (Linux) และเนื่องด้วยระบบปฏิบัติการแอนดรอยด์เป็น Open Source นักพัฒนาจึงสามารถเข้าไปปรับแต่ง เพิ่มเติมและพัฒนาการทำงานรวมถึงคิดค้น Application ใหม่ ๆ ได้อย่างอิสระ โดยตัวพัฒนาโปรแกรมในแอนดรอยด์นั้นสามารถโหลดมาใช้ได้ฟรีและไม่ได้มีข้อจำกัดเหมือน iPhone ที่เวลาโอนถ่ายข้อมูลระหว่าง

โทรศัพท์กับคอมพิวเตอร์ต้องต่อสายและโอนข้อมูลผ่าน iTunes เท่านั้น และการสร้างโปรแกรมประยุกต์ขอความช่วยเหลือนี้จะสร้างโดยใช้โปรแกรม Eclipse เนื่องจากตัวโปรแกรม Eclipse สามารถติดตั้งได้ง่าย ใช้กับตัวพัฒนาโปรแกรมในแอนดรอยด์ได้ทุกเวอร์ชั่น รองรับภาษาต่างประเทศได้หลายภาษา และสามารถทำงานได้กับไฟล์หลายชนิด

การส่ง SMS ขอความช่วยเหลือในโปรแกรมประยุกต์นี้ จะส่งข้อความเป็นพิกัดภูมิศาสตร์ที่ระบุตำแหน่งของจุดใด ๆ บนพื้นผิวโลก สามารถกำหนดได้ด้วยค่าละติจูด (Latitude) และลองจิจูด (Longitude)

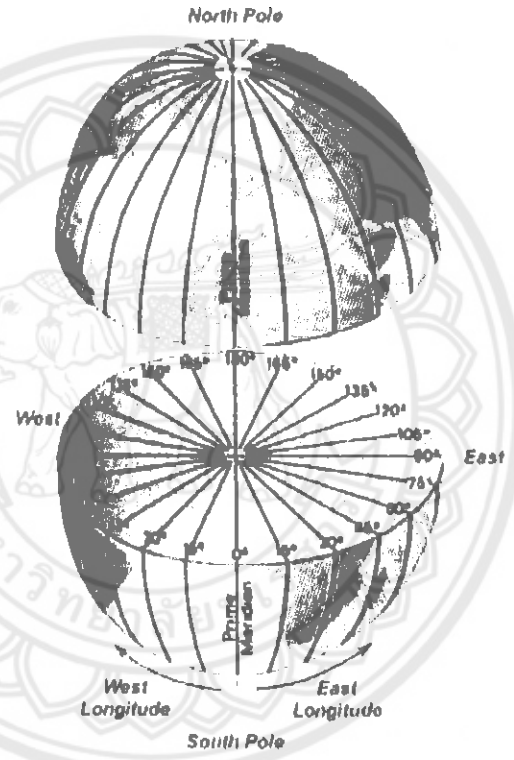
2.2 GPS และการระบุพิกัด [3]

ละติจูด เป็นระยะทางเชิงมุมที่วัดไปตามขอบเมริเดียน โดยนับศูนย์องศาจากเส้นศูนย์สูตรไปทางเหนือหรือใต้จนถึง 90 องศาที่ขั้วโลกทั้งสอง หรือเป็นมุมแนวตั้งที่ศูนย์กลางโลกระหว่างเส้นรัศมีของโลกที่ผ่านจุดซึ่งเส้นเมริเดียนตัดเส้นศูนย์สูตรกับเส้นรัศมี



รูปที่ 2.2 ภาพแสดงการแบ่งองศาของละติจูด

ลองจิจูด เป็นระยะทางเชิงมุมระหว่างเมริเดียนกรีนิชกับเมริเดียน ซึ่งวัดไปตามขอบของเส้นศูนย์สูตร หรือขอบของเส้นขนานละติจูด หรือเป็นมุมแนวระดับที่แกนโลกในระหว่างพื้นของเมริเดียนกรีนิชกับพื้นของเมริเดียน ตามปกติวัดเป็นองศาลิปดา และฟิลิปดา โดยนับศูนย์องศาจากเมริเดียนกรีนิชจนถึง 180 องศาไปทางตะวันออกหรือตะวันตกของเมริเดียนกรีนิช



รูปที่ 2.3 ภาพแสดงการแบ่งองศาของลองจิจูด

ในการระบุพิกัดตำแหน่งจากโปรแกรมประยุกต์นี้ ต้องอาศัยข้อมูลที่ได้จาก Global Positioning System (GPS) ซึ่ง GPS เป็นระบบบอกตำแหน่งบนพื้นผิวโลก โดยอาศัยการคำนวณพิกัดจากดาวเทียมระบุตำแหน่งจำนวน 24 ดวง ที่โคจรอยู่รอบโลกในระดับสูงประมาณ 20,000 กิโลเมตร ทำให้สามารถชี้บอกตำแหน่งได้ทุกแห่งบนโลกตลอดเวลา 24 ชั่วโมง โดยอุปกรณ์ที่ใช้ร่วมกับการนำทางด้วย GPS ประกอบด้วย ตัวรับสัญญาณ หน่วยประมวลผล โปรแกรมการนำทาง และข้อมูลแผนที่นำทาง

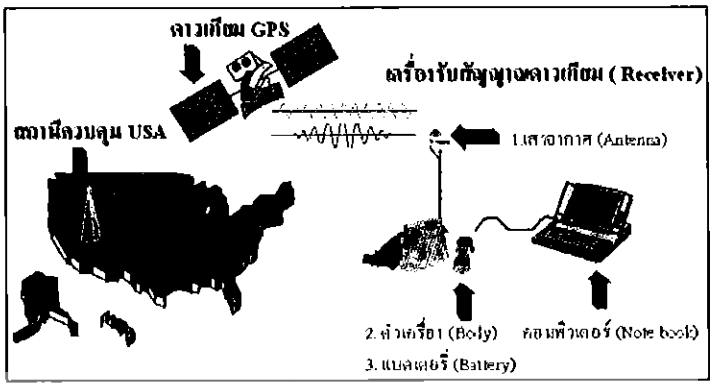
โปรแกรม GPS บน Android Smart Phone โดยทั่วไปมี 2 แบบ คือ GPS และ A-GPS ซึ่ง GPS นั้นจะรับสัญญาณตำแหน่ง (ข้อมูลตำแหน่งและข้อมูลเวลา ณ ปัจจุบัน) จากดาวเทียม GPS โดยตรง จึงมีข้อดีคือไม่ว่าจะอยู่ตำแหน่งไหนของโลกที่เป็นที่โล่งแจ้งไม่มีอะไรมาบดบังท้องฟ้า ก็สามารถรับสัญญาณจากดาวเทียมได้ แต่อาจจะมีข้อเสียในเรื่องของการทำงานได้ช้าในครั้งแรกและสิ้นเปลืองพลังงานของเครื่อง รวมถึงบางตำแหน่งอาจจะรับสัญญาณได้อ่อน สำหรับ A-GPS (Assisted GPS) จะรับสัญญาณตำแหน่งจากสถานีฐานของเครือข่ายโทรศัพท์มือถือ (GPS Basc Station) ซึ่งทำหน้าที่รับข้อมูลต่าง ๆ จากดาวเทียม GPS โดยตรง แล้วค่อยส่งต่อให้กับเครื่องที่เป็น A-GPS อีกที จึงทำให้มีข้อดีในเรื่องของความเร็วในการทำงาน แต่จะมีข้อเสียคือไม่สามารถใช้งานได้ในพื้นที่ที่ไม่สามารถรับสัญญาโทรศัพท์ได้ และ GPS Application Software บางตัวจะไม่สามารถทำงานได้ถ้าไม่มีการเปิดใช้งานอินเตอร์เน็ตด้วย

GPS บน Android Smart Phone ส่วนใหญ่จะเป็น A-GPS แต่มีบางรุ่นที่เป็น GPS ซึ่งสามารถรับสัญญาณจากดาวเทียมโดยตรงได้ เช่น HTC เป็นต้น

ข้อมูล GPS ในปัจจุบัน ให้บริการโดย GPS ของรัฐบาลสหรัฐอเมริกา ซึ่งมีนโยบายการให้บริการ 2 ด้าน คือ Precise Positioning Services (PPS) และ Standard Positioning Services (SPS)

Precise Positioning Services (PPS) ใช้ในการทางทหารเป็นหลัก ซึ่งข้อมูลจะมีการเข้ารหัสเฉพาะผู้ที่มีเครื่องถอดรหัสจึงจะสามารถใช้งานได้ โดยมีความถูกต้องของพิกัด คือ 22 เมตรในแนวราบ 27.7 เมตรในแนวดิ่ง และ 200 Nanosecond (UTC)

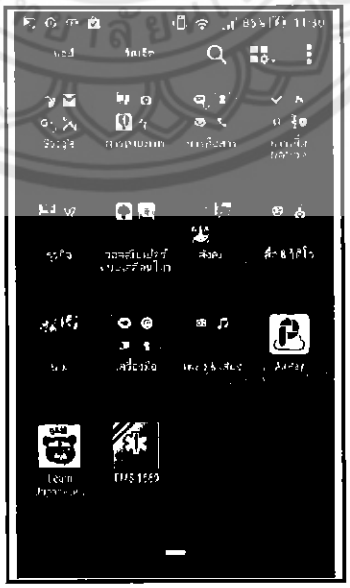
Standard Positioning Services (SPS) ใช้ในกิจการพลเรือนเป็นหลัก โดยจะมีความถูกต้องน้อยกว่า PPS ซึ่ง SPS มีความถูกต้องของพิกัด คือ 100 เมตรในแนวราบ 156 เมตรในแนวดิ่งและ 340 Nanosecond (UTC) [13, 14, 15, 16]



รูปที่ 2.4 แสดงองค์ประกอบของระบบดาวเทียม GPS

2.3 โปรแกรมประยุกต์ “EMS1669” ช่วยเหลือผู้ป่วยฉุกเฉินของสถาบันการแพทย์ฉุกเฉิน

สำหรับแอพลิเคชั่น “EMS1669” นั้น พัฒนาให้มีการใช้งานคือหากพบเห็นผู้บาดเจ็บหรือผู้ป่วยฉุกเฉิน เพียงแค่กดเรียกรถพยาบาล จากนั้นระบบจะให้ผู้แจ้งเหตุบันทึกประวัติส่วนตัว อาทิ ชื่อผู้แจ้ง เบอร์ติดต่อกลับ และระบบแจ้งพิกัดที่เกิดเหตุฉุกเฉินไปยังสถาบันการแพทย์ฉุกเฉินแห่งชาติอัตโนมัติ ซึ่งจะทำให้ศูนย์สั่งการรู้พิกัดจุดเกิดเหตุได้อย่างแม่นยำเพื่อส่งทีมกู้ชีพออกปฏิบัติการช่วยเหลือผู้ป่วยฉุกเฉินได้อย่างทันท่วงที นอกจากนี้ผู้แจ้งเหตุยังสามารถแนบไฟล์ภาพเหตุการณ์ เพื่อแจ้งสถานการณ์เพิ่มเติมได้ด้วย รวมทั้งสามารถแจ้งเหตุเพิ่มเติมผ่านการสนทนากับศูนย์สั่งการได้อีกด้วย

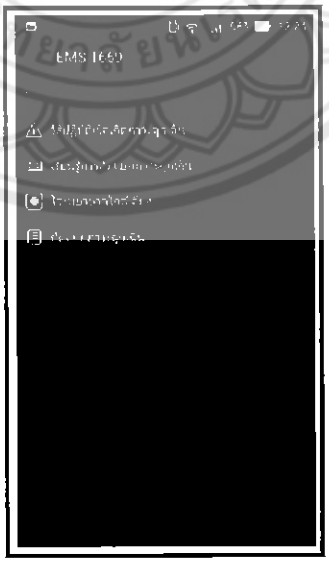


รูปที่ 2.5 หน้าจอแสดงโปรแกรมประยุกต์ “EMS1669” ในโทรศัพท์

การใช้งานโปรแกรมประยุกต์ “EMS1669” เมื่อเข้าสู่โปรแกรมประยุกต์ “EMS1669” จะพบกับหน้าแรกจะประกอบด้วย 3 ส่วน คือ เมนู บันทึกข้อมูลส่วนตัว และปุ่มกดเพื่อเรียกรถพยาบาล

รูปที่ 2.6 หน้าจอแสดงหน้าแรกของโปรแกรมประยุกต์ “EMS1669”

ส่วนเมนูจะมีการแนะนำการวิธีการปฏิบัติเมื่อเกิดภาวะฉุกเฉิน การคัดแยกภาวะฉุกเฉินและค้นหาโรงพยาบาลใกล้เคียง



รูปที่ 2.7 หน้าจอแสดงหน้าเมนูของโปรแกรมประยุกต์ “EMS1669”

ส่วนบันทึกข้อมูลส่วนตัว เมื่อกดปุ่มบันทึกข้อมูลส่วนตัวจะขึ้นหน้าต่างข้อมูลส่วนตัว เพื่อให้กรอกข้อมูลของผู้ใช้ ประกอบด้วย ชื่อ หมายเลขโทรศัพท์ เลขประจำตัวประชาชน โรคประจำตัว การแพ้ยา

รูปที่ 2.8 หน้าจอแสดงหน้าข้อมูลส่วนตัวเมื่อกดปุ่มบันทึกข้อมูลส่วนตัว

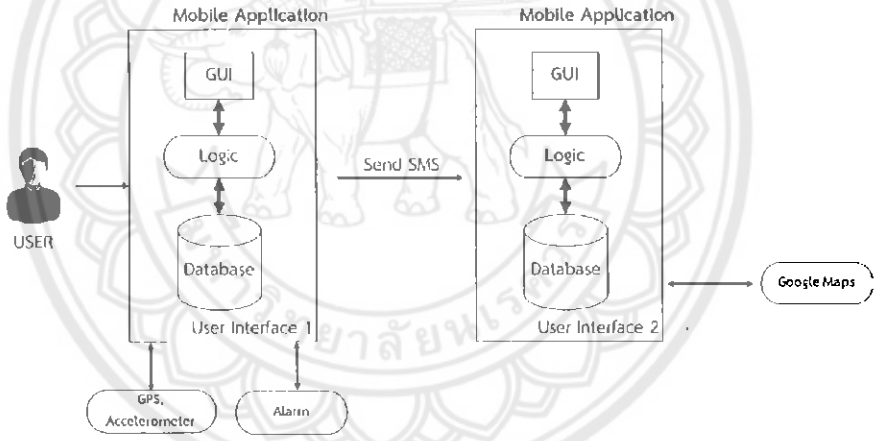
ส่วนกดปุ่มเรียกรถพยาบาล เมื่อกดปุ่มเรียกรถพยาบาลจะแสดงหน้า EMS 1669 ให้ผู้ใช้งานระบุเบอร์โทรศัพท์ติดต่อกลับและสามารถแนบรูปถ่ายไปได้อีกด้วย แต่เมื่อไม่มีสัญญาณระบุตำแหน่งโปรแกรมประยุกต์ EMS 1669 จะไม่สามารถบันทึกและแสดงตำแหน่งของผู้ขอความช่วยเหลือได้

รูปที่ 2.9 หน้าจอแสดงหน้า EMS 1669 เมื่อกดปุ่มเรียกรถพยาบาล

บทที่ 3

วิธีการดำเนินงาน

ในบทนี้จะกล่าวถึงการออกแบบการทำงานของระบบการส่งข้อความขอความช่วยเหลือฉุกเฉินด้วยระบบปฏิบัติการแอนดรอยด์ เพื่อให้ผู้ใช้งานสามารถได้รับการช่วยเหลือจากการส่งข้อมูลที่สำคัญออกไป จากข้อมูลในระบบ และวิธีการออกแบบหน้าตาของแอพพลิเคชั่นบนมือถือ เพื่อให้ผู้ใช้สามารถใช้งานได้ตามความต้องการ ดังแสดงในรูปที่ 3.1



รูปที่ 3.1 แสดงการวางแผนระบบการทำงานของระบบการส่งข้อมูลด้วยข้อความบนระบบปฏิบัติการแอนดรอยด์

จากรูปที่ 3.1 เป็นการออกแบบระบบโดยรวมของโครงงานนี้ โดยแบ่งออกเป็น 2 ส่วน ส่วนแรกเป็นส่วนแสดงผลตามความต้องการของผู้ส่ง ซึ่งแสดงผลในรูปแบบแอพพลิเคชั่นบนมือถือ (Mobile Application) โดยจะมีส่วนประกอบภายใน 2 ส่วน คือ ส่วนแสดงผล (Graphic User Interface) ส่วนการทำงานภายใน (Logic) ส่วนเก็บข้อมูลเพื่อแสดงผล (Database) และมีการเปิดใช้คำสั่งเสียงที่มี

ระดับเสียงที่ดังที่สุดของโทรศัพท์ ส่วนที่สองเป็นส่วนการทำงาน เป็นส่วนแสดงผลตามความต้องการของผู้รับ และมีการเรียกใช้ Google API ในการแสดงแผนที่ในระบบ

3.1 ข้อมูลที่ใช้ในการพัฒนาระบบ (Requirement Specifition)

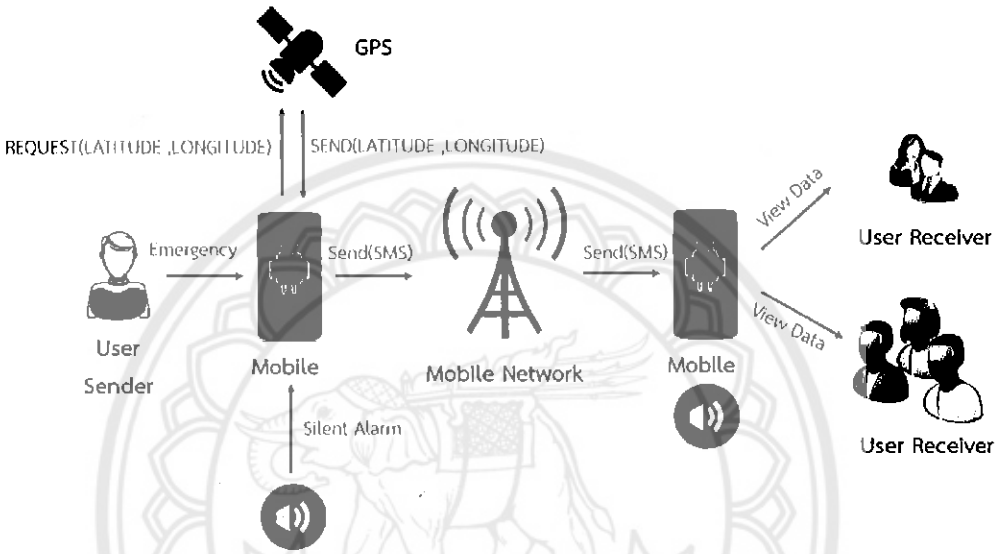
แอพพลิเคชั่นบนมือถือ (Mobile Application) สำหรับการขอความช่วยเหลือฉุกเฉินบนแพลตฟอร์มแอนดรอยด์ (Android) สามารถรับส่งและแลกเปลี่ยนข้อมูลระหว่างฝั่ง User ผู้เป็นคนส่งและ User ผู้เป็นคนรับด้วยข้อความช่วยเหลือ (Short Message Service: SMS) นอกจากนั้นยังนำเอาระบบกำหนดตำแหน่งบนโลก (Global Positioning System: GPS) มาใช้ในการระบุตำแหน่งที่อยู่ปัจจุบันของผู้ใช้งานที่ต้องการขอความช่วยเหลือนี้ โดยผู้ใช้งานระบบสามารถเลือกใช้งาน Accelerometer เพื่อวัดความเร่งในการเอียงเครื่องทั้ง 3 ทิศ โดยการพลิกหรือการหงายโทรศัพท์เพื่อส่งข้อความฉุกเฉิน เมื่อรับข้อความผู้รับสามารถดูพิกัดการขอความช่วยเหลือของผู้ใช้งานที่ต้องการขอความช่วยเหลือ โดยการดึงเลขพิกัดจากข้อความ เพื่อแสดงแผนที่ (Google Map) ได้อีกด้วย ขั้นตอนการใช้งานคือ ผู้ใช้งานระบบต้องกรอกข้อมูลส่วนตัวและบันทึกลงในฐานข้อมูลในโทรศัพท์ของผู้ใช้งานเอง ต่อมาผู้ใช้งานสามารถเลือกที่จะส่งข้อมูลส่วนตัวในบางส่วนและเลือกรูปแบบการส่งข้อความตามที่ผู้ใช้ต้องการ ผู้ใช้งานจะต้องรอการยืนยันจากเลขหมายที่ระบุไว้ เพื่อที่จะมีการส่งข้อความมาขอความช่วยเหลือ ส่วนการแก้ไขหรือการลบข้อมูลนั้น ผู้ที่ใช้งานโทรศัพท์เท่านั้นที่สามารถกระทำได้

ความต้องการทางด้าน Hardware และ Software

- Eclipse SDK Version: 4.2.1
- SQLite Version 3.8.8.2
- Android Development Tools 23.0.6
- Genymotion Version 4.4.4
- Android Debug Bridge

3.2 การออกแบบระบบ (System Design)

3.2.1 Conceptual Design



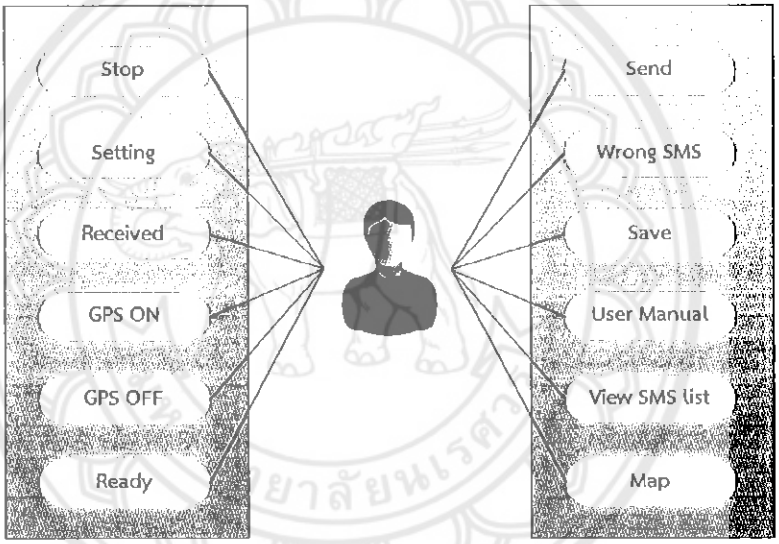
รูปที่ 3.2 แสดงถึง Conceptual Design ของระบบ

จากรูปที่ 3.2 แสดงถึงรูปแบบการทำงานหลักของการส่งข้อความขอความช่วยเหลือ (Short Message Service: SMS) แสดงผลในรูปแบบของแอพพลิเคชั่นบนมือถือ (Mobile Application)

การใช้งานโปรแกรมต้องทำการกรอกรายละเอียดประวัติส่วนตัวเพื่อเก็บข้อมูลในระบบฐานข้อมูล และทำการส่งข้อความการขอตอบรับการยืนยัน เพื่อจะมีการส่งข้อความมาขอความช่วยเหลือ จากนั้น ถ้าหมายเลขปลายทางยอมรับแล้ว ระบบถึงจะทำการเตรียมส่งขอความช่วยเหลือ เมื่อกรอกเสร็จแล้ว ระบบจะทำงานตลอดเวลาแม้ปิดโปรแกรมไปแล้วก็ตามและที่สำคัญต้องเปิดระบบบอกตำแหน่งบนพื้นผิวโลก (Global Positioning System: GPS) เอาไว้ตลอดเวลาเพื่อเก็บค่าพิกัดลงฐานข้อมูลเช่นเดียวกัน ระบบจะทำการตรวจสอบสถานะความพร้อมใช้งานทำงานบน background เก็บพิกัด GPS ทุกๆ 2 วินาที เมื่อเกิดเหตุฉุกเฉินผู้ใช้จะส่งข้อมูล เป็นพิกัดสุดท้ายก่อนที่จะรับสัญญาณ GPS ไม่ได้ซึ่งได้มาจากระบบบอกตำแหน่งบนพื้นผิวโลก (Global Positioning System: GPS) ของโทรศัพท์เคลื่อนที่

(Mobile Phone) พร้อมส่งเสียงเตือนที่จะเป็นเสียงระดับเสียงที่ดังที่สุดของโทรศัพท์ แล้วทำการส่งข้อมูลออกไปในรูปแบบข้อความ (SMS) ผ่าน Mobile Network จนไปถึงเครื่องของผู้รับ เมื่อผู้รับได้ข้อความแล้วโทรศัพท์เคลื่อนที่ (Mobile Phone) จะส่งเสียงเตือน แล้วทำการเปิดข้อมูลผ่านโปรแกรมประยุกต์บนมือถือ (Mobile Application) เพื่อที่จะได้ดูรายละเอียดของพิกัดที่อยู่ของผู้ขอความช่วยเหลือและทำการเปิดแผนที่ต่อไป

3.2.2 Use Case Diagram



รูปที่ 3.3 แสดง Use Case Diagram ของระบบ

จากรูปที่ 3.3 ผู้ใช้งานสามารถเข้าถึงข้อมูล และฟังก์ชันอื่นๆในระบบได้ดังนี้

1. Stop ผู้ใช้งานหยุดการทำงานของโปรแกรมประยุกต์ ปุ่มหยุดการทำงานจะทำการหยุดการทำงานบนพื้นหลัง (Stop Service) และหยุดเสียง พร้อมทั้งทำการออกจากโปรแกรมประยุกต์

2. Setting ผู้ใช้งานใช้สำหรับกรอกข้อมูล แก้ไขข้อมูล เลือกใช้งานบนพื้นหลังและเลือกภาษา

3. Received ผู้ใช้งานสามารถข้อมูลที่ถูกส่งเข้ามาเป็นรูปแบบของข้อความ

4. GPS ON ผู้ใช้งานกำลังเปิดการใช้งานของระบบระบุตำแหน่ง

5. GPS OFF ผู้ใช้งานกำลังปิดการใช้งานของระบบระบุตำแหน่ง

6. Ready ผู้ใช้งานสามารถดูความพร้อมของการเก็บข้อมูลตำแหน่ง ละติจูด, ลองจิจูดของผู้ใช้

7. Send ผู้ใช้งานสามารถส่งข้อมูลด้วยตนเองหลังจากกรอกข้อมูลและระบบพร้อมใช้งาน

8. Wrong SMS เมื่อกดส่งขอความช่วยเหลือผิดพลาด โดยที่ผู้ส่งไม่ได้ตั้งใจจะสามารถกดปุ่มส่งข้อความแก้ (Wrong SMS) เพื่อให้ผู้รับได้รับทราบ

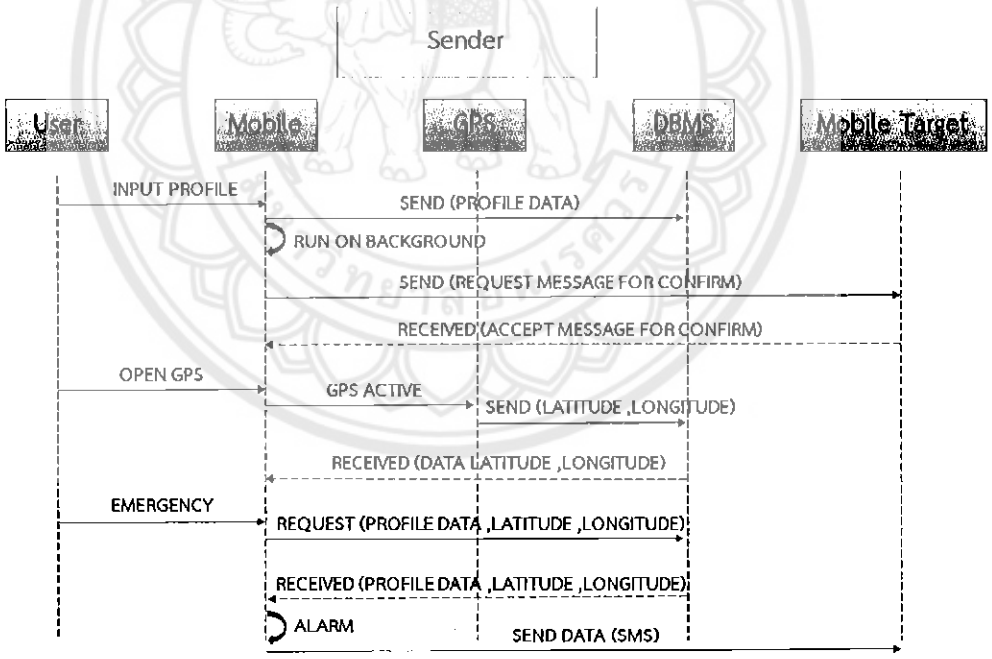
9. Save สำหรับบันทึกข้อมูลที่กรอกลงในฐานข้อมูลผู้ใช้

10. User Manual ผู้ใช้งานสามารถดูวิธีการใช้งานโปรแกรมประยุกต์

11. View List SMS ผู้ใช้งานสามารถเข้าดูข้อมูลที่ถูกส่งมาจะแสดงเป็นลำดับข้อความ

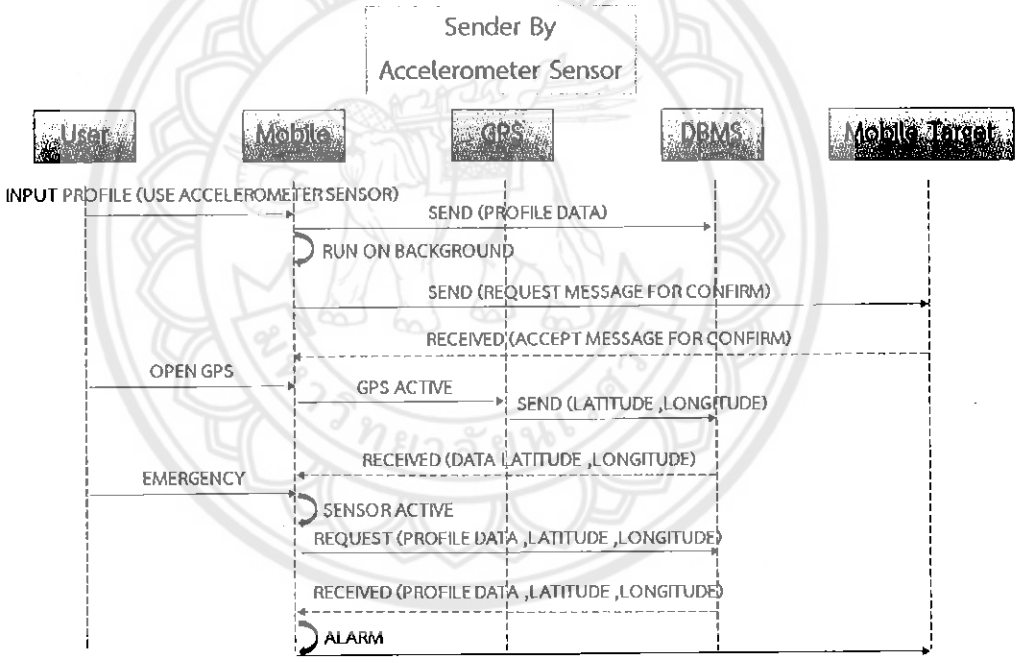
12. Map ผู้ใช้งานสามารถเข้าถึงสถานที่มีผู้ขอความช่วยเหลือจากแผนที่

3.2.3 Sequence Diagram



รูปที่ 3.4 แสดง Sequence Diagram ของการส่งข้อความแบบธรรมดา

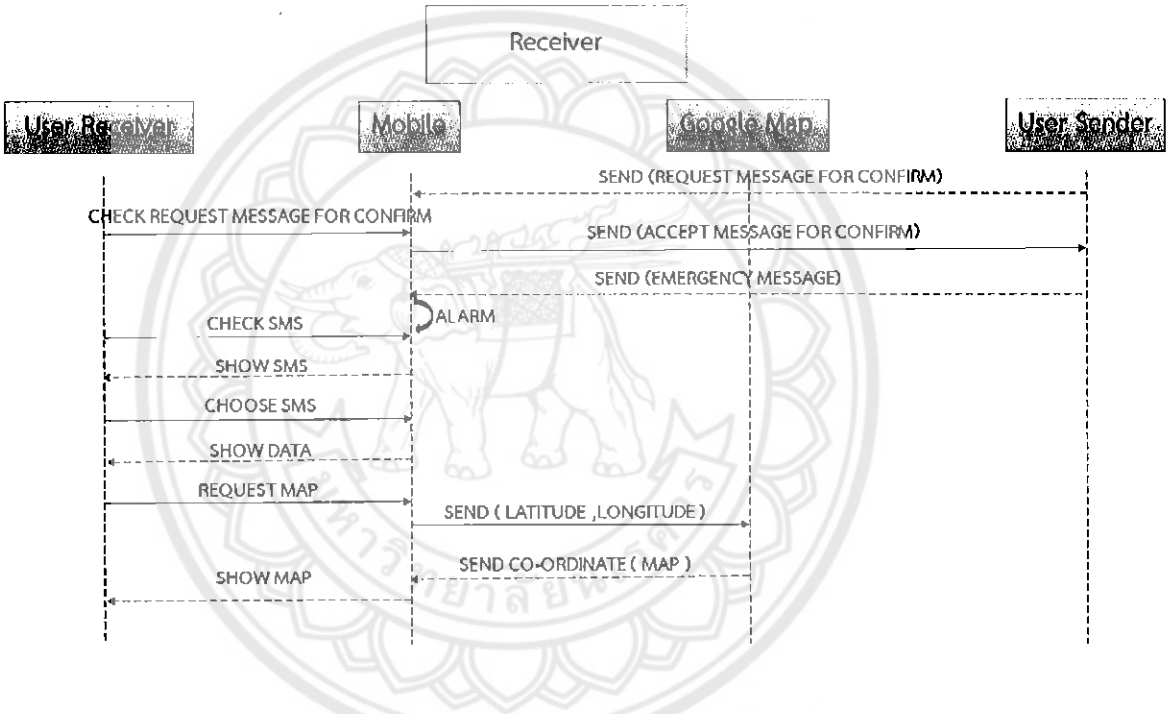
จากรูปที่ 3.4 ผู้ใช้งานระบบ (User) เมื่อเข้าสู่โปรแกรมครั้งแรก จะต้องทำการเปิดระบบกำหนดตำแหน่งบนโลก (GPS Sensor) แล้วป้อนข้อมูลส่วนตัว (Profile) และหมายเลขปลายทางที่ต้องการส่งไปเพื่อขอความช่วยเหลือ เมื่อกดปุ่มบันทึกระบบจะส่งค่าไปเก็บไว้ที่ฐานข้อูล (Database) และผู้ใช้งานต้องรอการยืนยันจากหมายลเขปลายทางที่ระบุไว้จึงจะสามารถเริ่มการใช้งานของโปรแกรมประยุกต์ได้อย่างสมบูรณ์ เมื่อได้รับการตอบรับการยืนยันแล้วโปรแกรมประยุกต์ก็จะเริ่มทำงานบนพื้นหลังของระบบตลอดเวลาเพื่อจะมีการส่งค่าพิกัดตำแหน่งบนโลกไปเก็บไว้ในฐานข้อมูล (Database) ทุกๆ 2 วินาที เมื่อมีการกดปุ่มส่งข้อความก็จะมีการไปเรียกขอข้อมูลส่วนตัว และ พิกัดตำแหน่งบนโลกในจุดที่ผู้ใช้อยู่จากฐานข้อมูล แล้วเครื่องก็จะส่งข้อความช่วยเหลือออกไป



รูปที่ 3.5 แสดง Sequence Diagram ของระบบในการส่งข้อความด้วย Sensor (Send SMS By Accelerometer Sensor)

จากรูปที่ 3.5 ผู้ใช้งานระบบ (User) เมื่อเข้าสู่โปรแกรมครั้งแรก จะต้องทำการเปิดระบบกำหนดตำแหน่งบนโลก (GPS Sensor) แล้วป้อนข้อมูลส่วนตัว (Profile) และหมายเลขปลายทางที่ต้องการส่งไปเพื่อขอความช่วยเหลือ เมื่อกดปุ่มบันทึกระบบจะส่งค่าไปเก็บไว้ที่ฐานข้อูล (Database) และผู้ใช้งานต้องรอการยืนยันจากหมายลเขปลายทางที่ระบุไว้จึงจะสามารถเริ่มการใช้งานของโปรแกรม

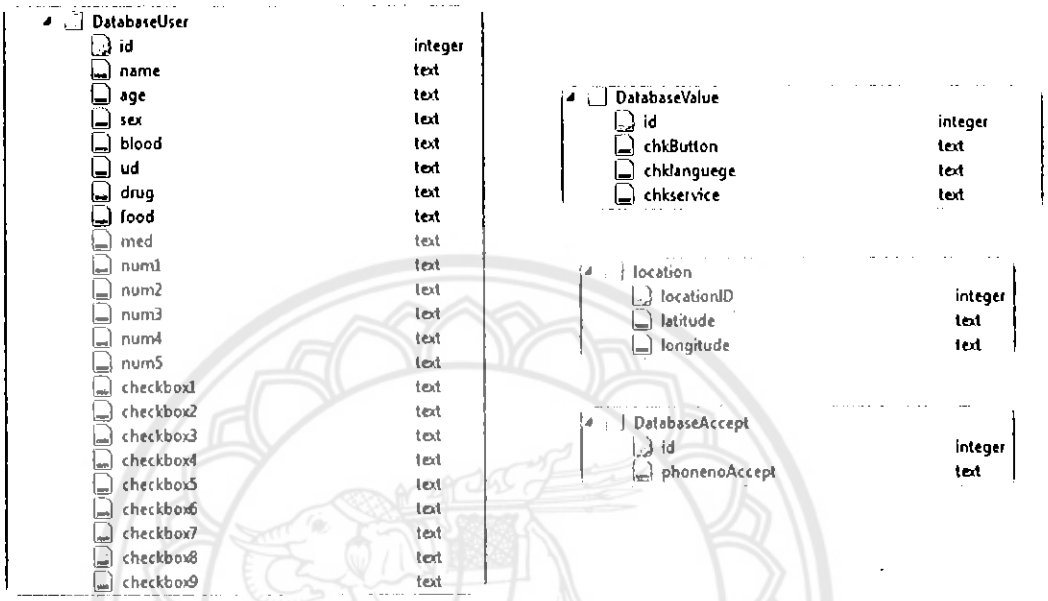
ประยุกต์ได้อย่างสมบูรณ์ เมื่อได้รับการตอบรับการยืนยันแล้วโปรแกรมประยุกต์ก็จะเริ่มทำงานบนพื้นหลังของระบบตลอดเวลาเพื่อจะมีการส่งค่าพิกัดตำแหน่งบนโลกไปเก็บไว้ในฐานข้อมูล (Database) ทุกๆ 2 วินาที เมื่อมีการพลิกคว่ำ-หงาย ของโทรศัพท์ ก็จะมีการไปเรียกขอ ข้อมูลส่วนตัว และ พิกัดตำแหน่งบนโลกในจุดที่ผู้ใช้อยู่ จากฐานข้อมูล แล้วเครื่องก็จะทำการส่งข้อความช่วยเหลือออกไป



รูปที่ 3.6 แสดง Sequence Diagram ของระบบในส่วนของผู้รับ (Receiver)

จากรูปที่ 3.6 ในส่วนของผู้รับจะมีข้อความมาขอตอบรับการยืนยัน เพื่อจะมีการส่งข้อความมาขอความช่วยเหลือหากตกลงระบบจะส่งข้อความยืนยันกลับไป เมื่อมีการส่งข้อความขอความช่วยเหลือเข้ามาจะมีเสียงดังขึ้น ให้เปิดโปรแกรมประยุกต์แล้วกดปุ่มผู้รับ (Receiver) จะแสดงรายการข้อความขอความช่วยเหลือ ให้เลือกรายการข้อความที่ต้องการ จะแสดงตัวเลขตำแหน่งของผู้ขอความช่วยเหลือ (ละติจูดและลองจิจูด) ถ้าต้องการให้แสดงแผนที่ให้กดปุ่มแผนที่ (Map) จะแสดงหน้าแผนที่และจุดที่ผู้ขอความช่วยเหลืออยู่ เพื่อจะสามารถขอความช่วยเหลือจากบริเวณใกล้เคียงหรือสามารถเข้าไปช่วยเหลือเองได้อย่างรวดเร็ว

3.3 การออกแบบระบบฐานข้อมูล (Database Design)



รูปที่ 3.7 แสดงฐานข้อมูลในระบบ

จากรูปที่ 3.7 แสดงฐานข้อมูลในระบบมีทั้งหมด 4 ตารางดังนี้

1. ตาราง DataUser จะเก็บข้อมูลส่วนตัวของผู้ใช้งาน

2. ตาราง DataValue จะเก็บข้อมูลของความพร้อมของระบบ

3. ตาราง Location จะใช้เก็บข้อมูลของพิกัดตำแหน่งบนโลก

4. ตาราง DatabaseAccept จะใช้เก็บข้อมูลของเบอร์โทรศัพท์ที่ได้รับการยืนยันแล้ว

3.3.1 พจนานุกรมข้อมูลของฐานข้อมูล (Data Dictionary)

หลังจากการได้แบบโครงสร้างความสัมพันธ์ต่างๆ ของข้อมูลแล้ว เราสามารถมากำหนดชื่อตาราง ชื่อฟิลด์ ชนิดข้อมูล รูปแบบข้อมูล ตามตารางต่อไปนี้

ตารางที่ 3.1 Data Dictionary ของตารางข้อมูลส่วนตัวของสมาชิก (DatabaseUser)

Field	Key	Data Type	Description
id	Primary key	Integer	รหัสอ้างอิง
name		Text	ชื่อจริง
age		Text	อายุ
sex		Text	เพศ
blood		Text	หมู่เลือด
ud		Text	โรคประจำตัว
drug		Text	ยาที่แพ้
food		Text	อาหารที่แพ้
med		Text	ยาที่ใช้อยู่
num1		Text	หมายเลขโทรศัพท์ 1
num2		Text	หมายเลขโทรศัพท์ 2
num3		Text	หมายเลขโทรศัพท์ 3
num4		Text	หมายเลขโทรศัพท์ 4
num5		Text	หมายเลขโทรศัพท์ 5
Checkbox1		Checkbox	เลือกส่งชื่อจริง
Checkbox2		Checkbox	เลือกส่งอายุและเพศ
Checkbox3		Checkbox	เลือกส่งหมู่เลือด
Checkbox4		Checkbox	เลือกส่งโรคประจำตัว
Checkbox5		Checkbox	เลือกส่งยาที่แพ้
Checkbox6		Checkbox	เลือกส่งอาหารที่แพ้
Checkbox7		Checkbox	เลือกส่งยาที่ใช้อยู่
Checkbox8		Checkbox	เลือกใช้งานเซนเซอร์
Checkbox9		Checkbox	เลือกใช้งานภาษาไทย

ป
053ป
2558

สำนักหอสมุด
26 ก.ย. 2560

ตารางที่ 3.2 Data Dictionary ของตารางตรวจสอบความพร้อมของระบบ (DatabaseValue)

Field	Key	Data Type	Description
ID	Primary Key	Integer	รหัสอ้างอิง
chkButton		Text	การคว่ำ-หงายเครื่อง
chklanguage		Text	สลับภาษา
chkservice		Text	ใช้งานบนพื้นหลัง

ตารางที่ 3.3 Data Dictionary ของตารางของพิกัดตำแหน่ง (Location)

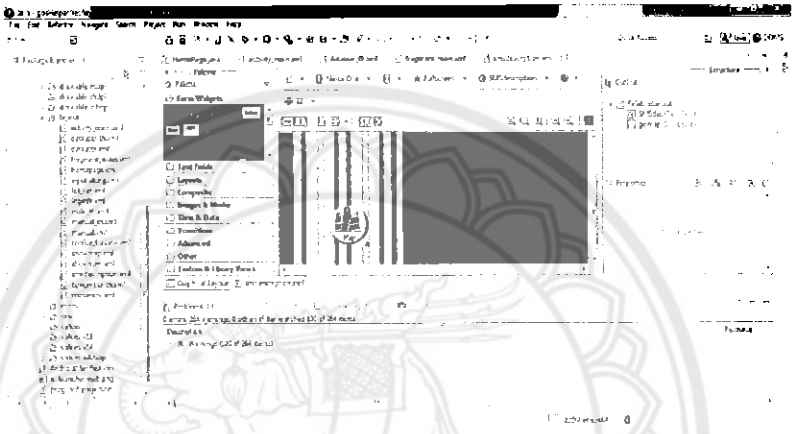
Field	Key	Data Type	Description
LocationID	Primary Key	Integer	รหัสอ้างอิง
latitude		Text	ค่าละติจูด
longitude		Text	ค่าลองจิจูด

ตารางที่ 3.4 Data Dictionary ของตารางของเบอร์โทรศัพท์ที่ได้รับการยืนยันแล้ว (DatabaseAccept)

Field	Key	Data Type	Description
ID	Primary Key	Integer	รหัสอ้างอิง
phonenoAccept		Text	หมายเลขโทรศัพท์ที่ได้รับการยืนยันแล้ว

3.4 การออกแบบหน้าตาแอพพลิเคชั่นบนมือถือ (User Interface)

การออกแบบหน้าตาแอพพลิเคชั่นบนมือถือ จะต้องคำนึงถึงความเหมาะสม ง่ายต่อการเรียกใช้งาน และสามารถตอบสนองความต้องการผู้ใช้งานได้เป็นอย่างดี ซึ่งในการออกแบบหน้าแสดงผลจะใช้โปรแกรม Eclipse SDK Version: 4.2.1 เขียนโดยใช้ภาษา Java และ XML ดังแสดงในรูปที่ 3.8

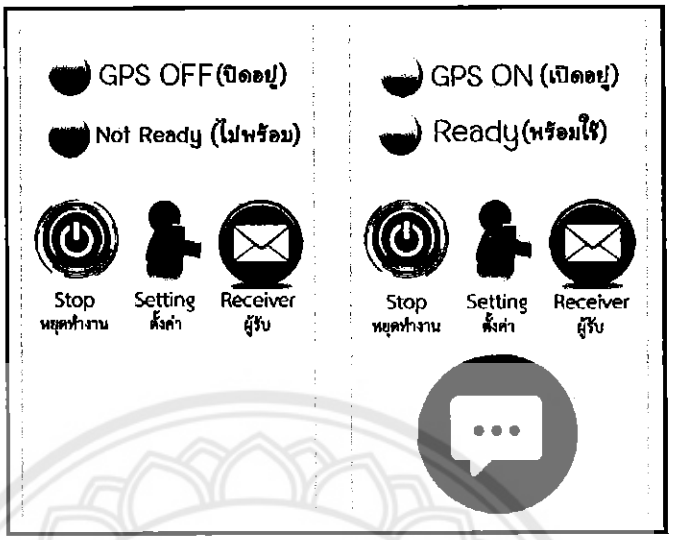


รูปที่ 3.8 แสดงการออกแบบโดยใช้โปรแกรม Eclipse SDK Version: 4.2.1

ส่วนต่อไปจะเป็นการออกแบบหน้าตาของแอพพลิเคชั่นบนมือถือ (Mobile Application)

3.4.1 การออกแบบหน้าเมนู (Home Page)

การออกแบบหน้าเมนู จะมีการแสดงผลในรูปแบบของ Relative Layout [13] เพราะว่า Layout นี้ง่ายต่อการจัดวางฟอร์มเนื่องจากการวางวัตถุจะมีความสัมพันธ์กัน ทำให้ GUI ในส่วนนี้ดูสวยงามขึ้น ซึ่งประกอบไปด้วยรูปภาพ (Image View) ดังแสดงในรูปที่ 3.9

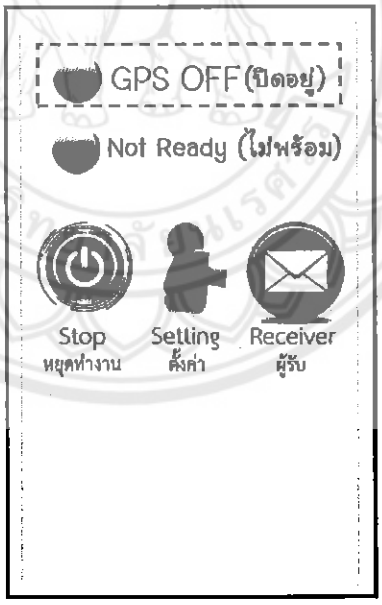


รูปที่ 3.9 แสดงการออกแบบหน้าเมนู (Home Page)

หน้าแรกจะเป็นหน้าหลักในการรวบรวมคำสั่งในการทำงานต่าง ๆ เช่น การส่งข้อความขอความช่วยเหลือด้วยตัวเอง เช็คสถานะระบบระบุตำแหน่งและสถานะพร้อมทำการส่งของโปรแกรมประยุกต์เป็นต้น หน้าแรกประกอบด้วย

- รูปแสดงสถานะของระบบระบุตำแหน่ง (GPS) จะแสดงสถานะของระบบระบุตำแหน่งว่าเปิดอยู่หรือไม่ หากเปิดอยู่จะแสดงรูปวงกลมสีเขียวพร้อมข้อความ "GPS ON (เปิดอยู่)" ดังแสดงในรูปที่ 3.10 แต่หากไม่ถูกเปิดอยู่จะแสดงรูปวงกลมสีแดงพร้อมข้อความ "GPS OFF (ปิดอยู่)" ดังแสดงในรูปที่ 3.11

รูปที่ 3.10 สถานะของระบบระบุตำแหน่ง (GPS) ที่สถานะเปิดอยู่

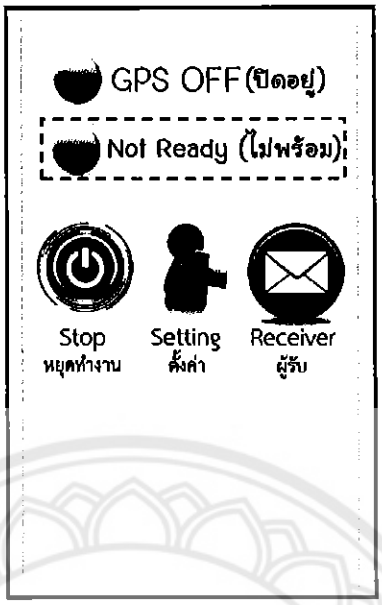


รูปที่ 3.11 สถานะของระบบระบุตำแหน่ง (GPS) ที่สถานะปิดอยู่

- รูปแสดงสถานะความพร้อมในการใช้งาน จะแสดงสถานะของฐานข้อมูล (Database) ว่าสามารถใช้ฐานข้อมูลในการส่งข้อความออกไปได้หรือไม่ โดยจะตรวจสอบจากฐานข้อมูลตำแหน่ง

(DatabaseLocation) ซึ่งเรียกจากฐานข้อมูลในโปรแกรมประยุกต์ (SQLite) หากระบบระบุตำแหน่งทำงานแล้วมีค่าพิกัดตำแหน่งที่อยู่บันทึกลงในฐานข้อมูลที่อยู่ (DatabaseLocation) รูปแสดงสถานะความพร้อมในการใช้งานจะแสดงรูปวงกลมสีเขียวพร้อมข้อความ "Ready (พร้อมใช้)" ดังแสดงในรูปที่ 3.12 แต่หากไม่มีค่าพิกัดตำแหน่งที่อยู่ถูกบันทึกลงในฐานข้อมูลที่อยู่ (DatabaseLocation) รูปสถานะความพร้อมในการใช้งานจะแสดงรูปวงกลมสีแดงพร้อมข้อความ "Not Ready (ไม่พร้อมใช้) ดังแสดงในรูปที่ 3.13

รูปที่ 3.12 สถานะความพร้อมในการใช้งานที่สถานะพร้อมใช้

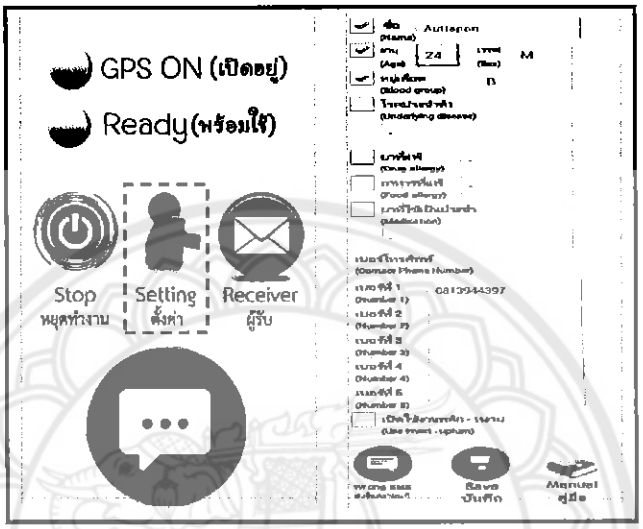


รูปที่ 3.13 สถานะความพร้อมในการใช้งานที่สถานะไม่พร้อมใช้

- ปุ่มหยุดการทำงาน (Stop) ใช้สำหรับเมื่อขอความช่วยเหลือ แล้วต้องการหยุดการทำงานของโปรแกรมประยุกต์ ปุ่มหยุดการทำงานจะทำการหยุดการทำงานบนพื้นหลัง (Stop Service) และหยุดเสียง พร้อมทั้งทำการออกจากโปรแกรมประยุกต์ ดังแสดงในรูปที่ 3.14

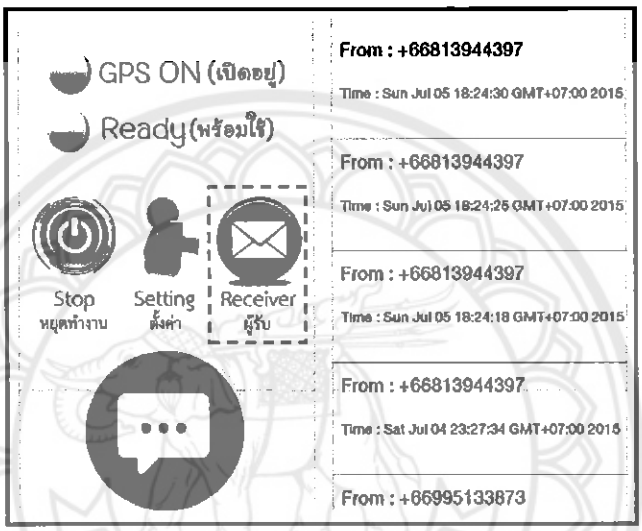
รูปที่ 3.14 แสดงการทำงานเมื่อกดปุ่มหยุดการทำงาน

- ปุ่มตั้งค่า (Setting) ใช้สำหรับกรอกข้อมูล แก้ไขข้อมูล ใช้งานบนพื้นหลังและเลือกเปิดการใช้งานของ Accelerometer Sensor ดังแสดงในรูปที่ 3.15



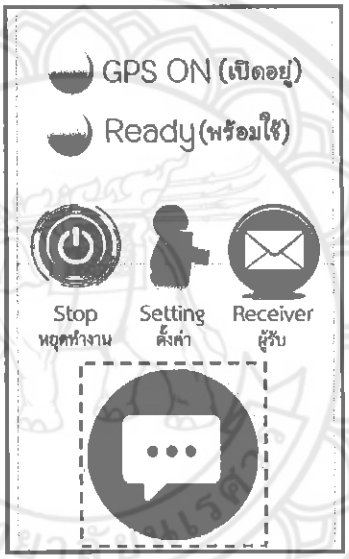
รูปที่ 3.15 แสดงการทำงานเมื่อกดปุ่มตั้งค่า

- ปุ่มผู้รับ (Receiver) ใช้สำหรับเมื่อผู้ใช้ส่งข้อความมายังหมายเลขของผู้รับ ผู้รับสามารถเข้าโปรแกรมประยุกต์แล้วใช้งานในส่วนปุ่มผู้รับ ในส่วนของผู้รับเมื่อกดปุ่มจะแสดงข้อความที่ผู้ใช้ส่งมายังผู้รับและเมื่อกดไปที่ข้อความจะแสดงพิกัดที่ผู้ใช้อยู่ และปุ่มแสดงแผนที่เพื่อแสดงแผนที่จากพิกัดที่ผู้ใช้ส่งมา ดังแสดงในรูปที่ 3.16



รูปที่ 3.16 แสดงการทำงานเมื่อกดปุ่มผู้รับ

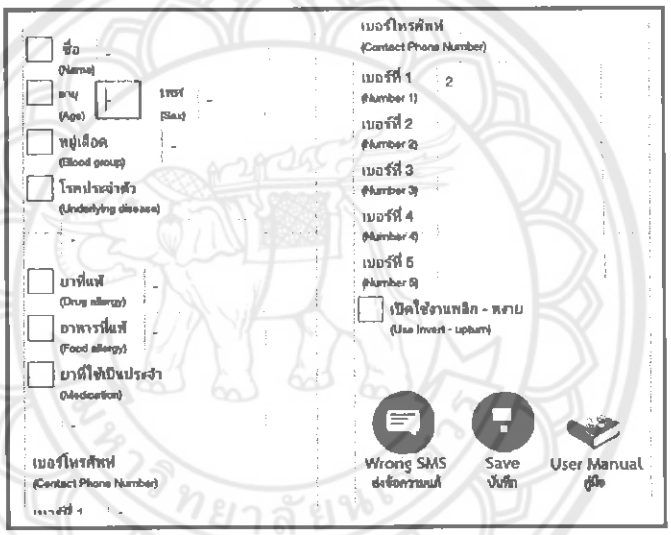
- ปุ่มสำหรับส่งข้อความโดยผู้ใช้เอง ใช้สำหรับผู้ใช้งานที่ไม่ต้องการใช้งานการทำงานบนพื้นหลังโดยการพลิกหงาย แต่ต้องการกดปุ่มขอความช่วยเหลือด้วยตัวเอง เมื่อเข้าโปรแกรมประยุกต์ในครั้งแรก ปุ่มสำหรับส่งข้อความโดยผู้ใช้เองจะถูกซ่อนไว้เพื่อกันความผิดพลาดในการกดส่งข้อความโดยไม่ได้ตั้งใจ ปุ่มสำหรับส่งข้อความโดยผู้ใช้เองจะแสดงก็ต่อเมื่อมีค่าในฐานข้อมูลตำแหน่ง (DatabaseLocation) และฐานข้อมูลผู้ใช้ (DatabaseUser) ดังแสดงในรูปที่ 3.17



รูปที่ 3.17 ปุ่มสำหรับส่งข้อความโดยผู้ใช้เอง

3.4.2 การออกแบบหน้าข้อมูลส่วนตัว (Profile Page)

การออกแบบหน้าข้อมูลส่วนตัว การออกแบบหน้าข้อมูลส่วนตัว จะใช้ Android Layout ชนิด Relative Layout [13] เพราะว่า Layout นี้ง่ายต่อการจัดวางฟอร์มเนื่องจากการวางวัตถุจะมีความสัมพันธ์กัน และสร้าง Scroll View ซึ่งสามารถเพิ่มฟิลด์ในการกรอกรายละเอียดต่างๆ ของข้อมูลส่วนตัวนั้นได้มากขึ้น ที่สำคัญยังต้องสร้าง Check Box เพื่อใช้ในการเลือกค่าที่ต้องการส่งด้วย และข้อมูลที่ได้จากผู้ใช้งานจะถูกนำไปเก็บไว้ในฐานข้อมูล (Database) และส่วนที่แสดงค่าคือข้อความ (Text View) ดังแสดงในรูปที่ 3.18



รูปที่ 3.18 แสดงการออกแบบหน้าข้อมูลส่วนตัว (Profile Page)

หน้าตั้งค่า (Setting) เมื่อเข้ามายังหน้าตั้งค่า (Setting) หรือใช้งานเป็นครั้งแรกโปรแกรมประยุกต์จะทำการแสดงหน้าตั้งค่า (Setting) เพื่อให้ผู้ใช้ได้กรอกข้อมูลก่อน ผู้ใช้ควรกรอกข้อมูลให้ครบถ้วนทุกช่องยกเว้นช่องหมายเลขโทรศัพท์ต้องกรอกอย่างน้อย 1 หมายเลข หากผู้ใช้ต้องการให้ส่งข้อมูลส่วนใดก็ให้เลือกที่หน้าช่องนั้น ข้อมูลจะถูกส่งไปพร้อมกับข้อความที่ส่งไปยังหมายเลขปลายทางด้วย ผู้ใช้สามารถเลือกการทำงานเซนเซอร์เพื่อใช้การพลิกหงายในการส่งข้อความหรือไม่เลือกก็ได้ เมื่อกรอกข้อมูลเสร็จสิ้นผู้ใช้สามารถกดปุ่มบันทึก (Save) เพื่อบันทึกข้อมูลลงในฐานข้อมูลผู้ใช้ (DatabaseUser) และส่งข้อความเพื่อขอตอบรับการยืนยันจากหมายเลขปลายทาง หรือหากผู้ใช้งานไม่ทราบวิธีการใช้งานสามารถเข้าไปดูได้ที่ปุ่มคู่มือ (Manual) เมื่อผู้ใช้บันทึกข้อมูลลงในฐานข้อมูลโปรแกรมประยุกต์จะ

กลับมายังหน้าแรก (Home page) เพื่อให้ผู้ใช้เริ่มใช้งานโปรแกรมประยุกต์ หน้าตั้งค่า (Setting) ประกอบด้วย

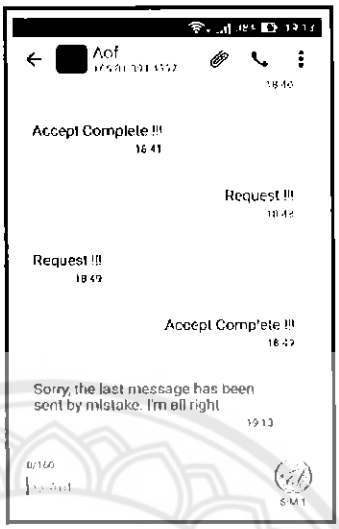
- ชื่อ (Name)
- อายุ (Age)
- เพศ (Sex)
- หมู่เลือด (Blood group)
- โรคประจำตัว (Underlying disease)
- ยาที่แพ้ (Drug allergy)
- อาหารที่แพ้ (Food allergy)
- ยาที่ใช้เป็นประจำ (Medication)
- หมายเลขโทรศัพท์ปลายทาง 5 หมายเลข (Contact telephone number)
- เลือกการทำงานพลิกหงายเครื่อง (Use service)

เงื่อนไขในการกรอกข้อมูลมีดังนี้

- เมื่อเข้าใช้งานโปรแกรมประยุกต์เป็นครั้งแรก โปรแกรมประยุกต์จะดำเนินการใส่เครื่องหมาย "-" หากต้องการกรอกข้อมูลให้ทำการลบเครื่องหมาย "-" แล้วกรอกข้อมูลลงในช่อง ข้อมูลที่กรอกจะต้องไม่มีช่องว่าง ยกเว้นหมายเลขโทรศัพท์ต้องกรอกอย่างน้อย 1 หมายเลข
- อายุ (Age) กรอกได้เฉพาะตัวเลขเท่านั้นและสามารถกรอกข้อมูลได้สูงสุด 3 ตัวเลข
- หมู่เลือด (Blood group) กรอกได้เฉพาะตัวอักษร "a", "A", "b", "B", "o" , "O", สัญลักษณ์ "+" และสามารถกรอกข้อมูลได้สูงสุด 2 ตัวอักษร
- หมายเลขปลายทาง 5 หมายเลขจะกรอกได้เฉพาะตัวเลขและสัญลักษณ์เกี่ยวกับการโทร สามารถกรอกได้สูงสุด 20 ตัวเลข

นอกจากนั้นหน้าตั้งค่ามีปุ่มซึ่งประกอบด้วย

- ปุ่มข้อความแก้ (Wrong SMS) เมื่อกดส่งขอความช่วยเหลือผิดพลาด โดยที่ผู้ส่งไม่ได้ตั้งใจจะสามารถกดปุ่มข้อความแก้ (Wrong SMS) เพื่อให้ผู้รับได้รับทราบ เมื่อเข้าโปรแกรมประยุกต์ในครั้งแรกปุ่มข้อความแก้จะถูกซ่อนไว้เพื่อกันความผิดพลาดในการกดส่งข้อความโดยไม่ได้ตั้งใจ ดังแสดงในรูปที่ 3.19

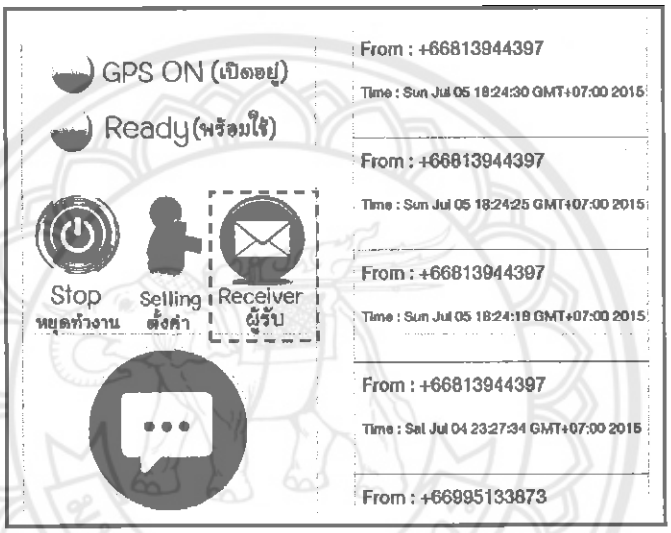


รูปที่ 3.19 ตัวอย่างข้อความที่ผู้รับได้รับเมื่อกดปุ่มส่งข้อความแก้

- ปุ่มบันทึก (Save) ใช้สำหรับบันทึกข้อมูลที่กรอกลงในฐานข้อมูลผู้ใช้ (DatabaseUser) และส่งข้อความเพื่อขอการตอบรับการยืนยันจากหมายเลขปลายทางที่ระบุไว้
- ปุ่มคู่มือ (Manual) จะบอกวิธีการใช้งานโปรแกรมประยุกต์

3.4.3 การออกแบบหน้ารายชื่อของข้อความที่ถูกส่งเข้ามา (Received Listview Page)

การออกแบบหน้ารายชื่อของความที่ถูกส่งเข้ามา จะใช้ Android Layout ชนิด Relative Layout [13] และสร้าง Scroll View ซึ่งสามารถเลื่อนดูข้อมูลของข้อความที่ถูกส่งเข้ามาได้มากขึ้น และใน Listview นี้จะดึงเฉพาะข้อมูลที่มีการส่งมาจากผู้ที่ขอความช่วยเหลือเท่านั้น เพื่อง่ายต่อการเรียกดู ดังแสดงในรูปที่ 3.20

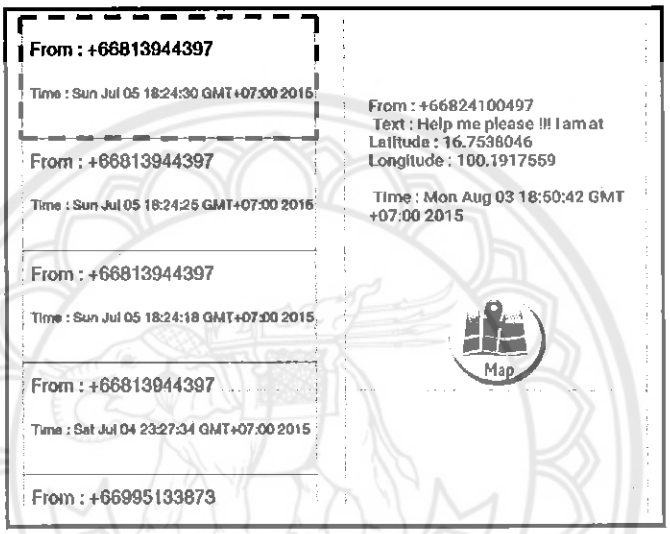


รูปที่ 3.20 แสดงการออกแบบหน้ารายชื่อของข้อความที่ถูกส่งเข้ามา (Received Listview Page)

เมื่อกดปุ่มผู้รับ (Receiver) จะเข้ามายังรายการแสดงข้อความที่ผู้ใช้ส่งข้อความขอความช่วยเหลือมายังผู้รับ โดยเมื่อกดปุ่มผู้รับ (Receive) จะแสดงรายการข้อความที่คัดเลือกเฉพาะข้อความที่มีคำว่า "Help !!!" มาแสดง

3.4.4 การออกแบบหน้ารายละเอียดของข้อความ (Message Page)

การออกแบบหน้ารายละเอียดของข้อความ จะใช้ Android Layout ชนิด Relative Layout เพราะว่า Layout นี้ง่ายต่อการจัดวางฟอร์มเนื่องจากการวางวัตถุจะมีความสัมพันธ์กัน Layout นี้ง่ายต่อการจัดวางฟอร์ม ทำให้ GUI ในส่วนนี้ดูสวยงามขึ้น ดังแสดงในรูปที่ 3.21

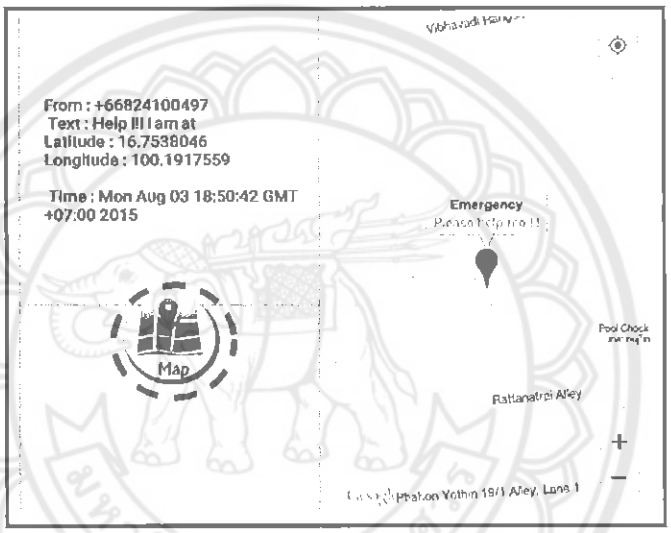


รูปที่ 3.21 แสดงการออกแบบหน้ารายละเอียดของข้อความ (Message Page)

เมื่อกดเลือกข้อความจะแสดงตัวเลขระบุตำแหน่ง (ละติจูดและลองจิจูด) ที่ผู้ใช้ส่งมาเพื่อขอความช่วยเหลือ สามารถนำตัวเลขพิกัดไปค้นหาเองหรือกดปุ่มแผนที่ (Map) เพื่อให้แสดงแผนที่จากตัวเลขระบุตำแหน่งที่แสดง

3.4.5 การออกแบบหน้าแสดงแผนที่ (Map)

การออกแบบหน้าแสดงแผนที่ จะต้องมีการเรียกใช้ Google Maps API จึงต้องมีการใช้ Android Layout ชนิด Fragment Layout เพื่อสามารถนักพัฒนาที่นำไปใช้งานร่วมกับ Fragment ตัวอื่นๆได้ ดังนั้นถ้าใช้กับ Activity โดยตรงก็ทำได้ และถ้าใช้ร่วมกับ Fragment ตัวอื่นๆก็ทำได้เช่นกัน ดังแสดงในรูปที่ 3.22

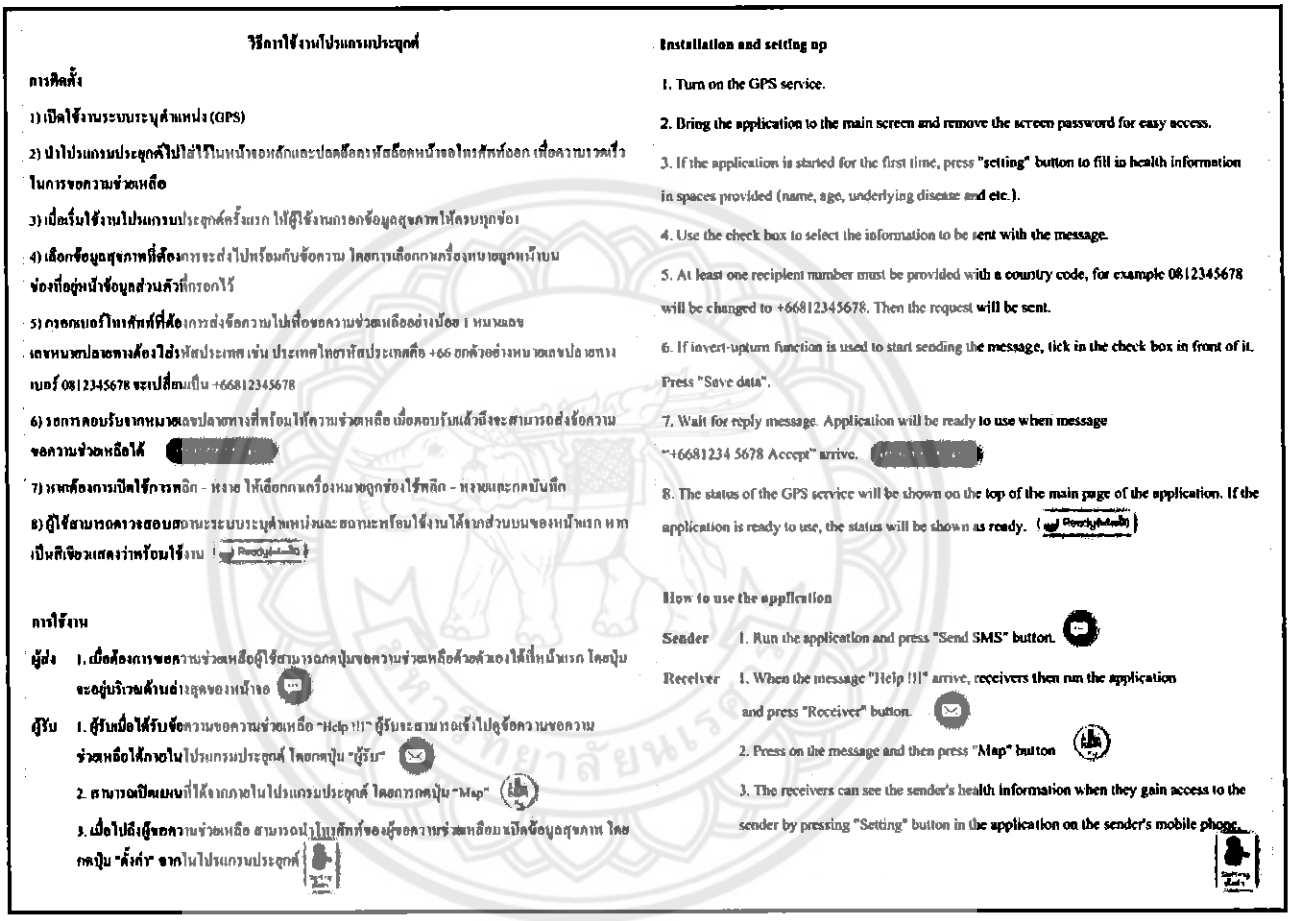


รูปที่ 3.22 แสดงการออกแบบหน้าแสดงแผนที่ (Map Page)

เมื่อกดปุ่ม Map จะแสดง การระบุตำแหน่ง (ละติจูดและลองจิจูด) ที่ผู้ใช้ส่งมาเพื่อขอความช่วยเหลือในรูปแบบแผนที่

3.4.6 การออกแบบหน้าคู่มือการใช้ (User Manual Page)

การออกแบบหน้าคู่มือการใช้ จะใช้ Android Layout ชนิด Relative Layout ซึ่งเป็น Layout เริ่มต้นในการสร้าง GUI และจะแสดงข้อมูลการใช้โดยใช้ รูปภาพ (Image View) ดังแสดงในรูปที่ 3.23



วิธีการใช้งานโปรแกรมประยุกต์

การติดตั้ง

1) เปิดใช้งานระบบระบุตำแหน่ง (GPS)

2) นำโปรแกรมประยุกต์ไปใส่ไว้ในหน้าจอหลักและปลดล็อคการัสล็อคหน้าจอโทรศัพท์ออก เพื่อความรวดเร็วในการขอความช่วยเหลือ

3) เมื่อเริ่มใช้งานโปรแกรมประยุกต์ครั้งแรก ให้ผู้ใช้งานกรอกข้อมูลสุขภาพให้ครบทุกช่อง

4) เลือกข้อมูลสุขภาพที่ต้องการจะส่งไปพร้อมกับข้อความ โดยการเลือกกาเครื่องหมายถูกหน้าบนช่องที่อยู่หน้าข้อมูลส่วนตัวที่กรอกไว้

5) การกดเบอร์โทรศัพท์ที่ต้องการส่งข้อความไปเพื่อขอความช่วยเหลืออย่างน้อย 1 หมายเลข เลขหมายปลายทางต้องใส่รหัสประเทศ เช่น ประเทศไทยรหัสประเทศคือ +66 ยกตัวอย่างหมายเลขปลายทางเบอร์ 0812345678 จะเปลี่ยนเป็น +66812345678

6) รอการตอบรับจากหมายเลขปลายทางที่พร้อมให้ความช่วยเหลือ เมื่อตอบรับแล้วนี้จะสามารถส่งข้อความขอความช่วยเหลือได้ [redacted]

7) หากต้องการปิดใช้การหลิก - หรอย ให้เลือกกดเครื่องหมายถูกช่องไร้หลิก - หรอยและกดบันทึก

8) ผู้ใช้สามารถตรวจสอบสถานะระบบระบุตำแหน่งและสถานะพร้อมใช้งานได้จากส่วนบนของหน้าแรก หากเป็นสีเขียวแสดงว่าพร้อมใช้งาน [Ready to use]

การใช้งาน

ผู้ส่ง 1. เมื่อต้องการขอความช่วยเหลือผู้ใช้สามารถกดปุ่มขอความช่วยเหลือด้วยตัวเองได้ที่หน้าแรก โดยปุ่มจะอยู่บริเวณด้านล่างสุดของหน้าจอ

ผู้รับ 1. ผู้รับเมื่อได้รับข้อความขอความช่วยเหลือ "Help !!!" ผู้รับจะสามารถเข้าไปดูข้อความขอความช่วยเหลือได้ภายในโปรแกรมประยุกต์ โดยกดปุ่ม "ผู้รับ"

2. สามารถเปิดแผนที่ได้จากภายในโปรแกรมประยุกต์ โดยการกดปุ่ม "Map"

3. เมื่อไปถึงผู้ขอความช่วยเหลือ สามารถนำโทรศัพท์ของผู้ขอความช่วยเหลือมาเปิดข้อมูลสุขภาพ โดยกดปุ่ม "ตั้งค่า" จากในโปรแกรมประยุกต์

Installation and setting up

1. Turn on the GPS service.

2. Bring the application to the main screen and remove the screen password for easy access.

3. If the application is started for the first time, press "setting" button to fill in health information in spaces provided (name, age, underlying disease and etc.).

4. Use the check box to select the information to be sent with the message.

5. At least one recipient number must be provided with a country code, for example 0812345678 will be changed to +66812345678. Then the request will be sent.

6. If invert-upturn function is used to start sending the message, tick in the check box in front of it. Press "Save data".

7. Wait for reply message. Application will be ready to use when message "+66681234 5678 Accept" arrive. [redacted]

8. The status of the GPS service will be shown on the top of the main page of the application. If the application is ready to use, the status will be shown as ready. [Ready to use]

How to use the application

Sender 1. Run the application and press "Send SMS" button.

Receiver 1. When the message "Help !!!" arrive, receivers then run the application and press "Receiver" button.

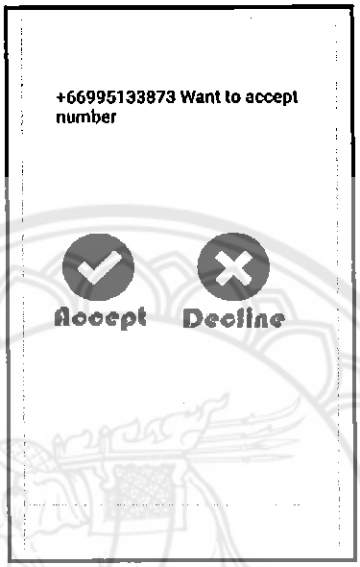
2. Press on the message and then press "Map" button

3. The receivers can see the sender's health information when they gain access to the sender by pressing "Setting" button in the application on the sender's mobile phone.

รูปที่ **3.23** แสดงการออกแบบหน้าคู่มือการใช้ (User Manual Page)

3.4.7 การออกแบบหน้าการยืนยันเบอร์โทรศัพท์ในฝั่งของผู้รับ

การออกแบบหหน้าการยืนยันเบอร์โทรศัพท์ในฝั่งของผู้รับ จะใช้ Android Layout ชนิด Relative Layout ซึ่งเป็น Layout เริ่มต้นในการสร้าง GUI ดังแสดงในรูปที่ 3.24



รูปที่ 3.24 แสดงการการออกแบบหน้าการยืนยันเบอร์โทรศัพท์ในฝั่งของผู้รับ

คำสั่งที่สำคัญของโปรแกรม

คำสั่งหยุดการทำงานของ โปรแกรม ดังแสดงในรูปที่ 3.25

```
1    stop.setOnClickListener (new View.OnClickListener()
2        {
3        public void onClick ( View v )
4        {
5            Toast.makeText(getApplicationContext(), "Stop Service",
6            Toast.LENGTH_LONG).show();
7
8            valueDb.UpdateData("1","chkButton","0");
9
10           play.pause();
11           stopService(new Intent(HomePage.this, MyService.class));
12           Intent intent = new Intent(Intent.ACTION_MAIN);
13           intent.addCategory(Intent.CATEGORY_HOME);
14           startActivity(intent);
15           int pid = android.os.Process.myPid();
16           android.os.Process.killProcess(pid);
17           System.exit(0);
18           //play.reset();
19
20       }
21       });
```

รูปที่ 3.25 แสดงการออกแบบในการเขียนคำสั่งหยุดการทำงานของโปรแกรม

คำอธิบาย Code

ในบรรทัดที่ 10 → Play.pause คือ การหยุดเล่นเสียง

ในบรรทัดที่ 11 → stopService คือ การหยุดทำงานบนพื้นหลัง

ในบรรทัดที่ 13 → CATEGORY_HOME คือ การเรียกใช้ฟังก์ชั่นของปุ่ม HOME

ในบรรทัดที่ 16 → killProcess คือ การปิดการทำงานของโปรแกรมประยุกต์ทั้งหมด

คำสั่งนี้จะใช้ในการหยุดทำงานของโปรแกรมประยุกต์และหยุดเสียงไซเรน คำสั่งนี้อยู่ใน HomePage.class

คำสั่งตรวจสอบ Checkbox ดังแสดงในรูปที่ 3.26

```
1   if(chkBox9.isChecked() == true)
2   {
3       Log.d("checkbox9", "true");
4       userDb.UpdateData("1","checkbox9","1");
5       valueDb.UpdateData("1","chklanguage","1");
6   }
7   else{
8       Log.d("checkbox9", "false");
9       userDb.UpdateData("1","checkbox9","0");
10      valueDb.UpdateData("1","chklanguage","0");
11  }
```

รูปที่ 3.26 แสดงการออกแบบในการเขียนคำสั่งตรวจสอบ Checkbox

คำอธิบาย Code

คำสั่งนี้จะทำการด้วยการตรวจสอบ Checkbox ว่าถูกกาเครื่องหมายหรือไม่ หากถูกกาเครื่องหมายอยู่ จะจัดเก็บค่า "1" ลงในฐานข้อมูล หากไม่ถูกกาเครื่องหมายอยู่ จะจัดเก็บค่า "0" ลงในฐานข้อมูล คำสั่งนี้อยู่ใน Confix_EN.class (Setting)

คำสั่งใช้งานการทำงานของ Accelerometer Sensor ดังแสดงในรูปที่ 3.27

```
1   @Override
2   public void onSensorChanged(SensorEvent event) {
3       // TODO Auto-generated method stub
4       // TODO Auto-generated method stub
5       //Context context = null;
6       float x = event.values[0];
7       float y = event.values[1];
8       float z = event.values[2];
9
10          if(z >= 9.0 && chkss[1] == 1 && (-3 < y && y < 3)){
11              Log.d("loop", "send message");
12              sendsms();
13
14
15              chkss[0] = 0;
16              chkss[1] = 0;
17          }
18
19          else if(z <= -9.0 && chkss[0] == 1 && (-3 < y && y < 3)){
20              //Log.d("loop", "before send");
21              chkss[1] = 1;
22          }
23
24          else if(z >= 9.0 && (-3 < y && y < 3)){
25              //Log.d("loop", "check 1 send");
26              chkss[0] = 1;
27          }
28
29          else if(-3 > y || y > 3){
30              //Log.d("loop", "cancel send");
31              chkss[0] = 0;
32              chkss[1] = 0;
33          }
34
35      }
```

รูปที่ 3.27 แสดงการออกแบบในการเขียนคำสั่งการทำงานของ Acceleromcter Sensor

<u>คำอธิบาย</u>โค้ด

คำสั่งนี้จะใช้ if else เข้ามาช่วยในการตรวจสอบค่าของแกนในแนว z และ y ของ Accelcrometer ว่าอยู่ในแนวใด โดยจะตรวจสอบการคว่ำเครื่องเป็นอย่างแรก หากมีการคว่ำเครื่องจะให้เก็บค่าใน array chkss[1] เป็น "1" เมื่อเก็บค่าแล้วหากมีการหงายเครื่องจะทำการส่งข้อความและรีเซ็ตค่าใน array ทั้งหมด แต่หากมีการกระทำอื่น ๆ ที่ไม่ใช่การหงายเครื่อง array ก็จะถูกรีเซ็ตเช่นกันและต้องมาตรวจสอบการคว่ำเครื่องอีกรอบหนึ่ง ดังนั้นคำสั่งนี้จะเป็นการตรวจสอบทำงานของ Accelerometer Sensor เพื่อวัดความเร่งในการเอียงเครื่องทั้ง 3 ทิศ คำสั่งนี้อยู่ใน Service.class

คำสั่งส่งค่าละติจูดและลองจิจูดจากถาดเข้าข้อความ ดังแสดงในรูปที่ 3.28

```
1       ImageView mapBut = (ImageView) this.findViewById (R.id.gomap);
2       mapBut.setOnClickListener(new OnClickListener() {
3               @Override
4               public void onClick(View v) {
5                   // TODO Auto-generated method stub
6
7                   String lat = "";
8                   String lon = "";
9                   Scanner scanner = new Scanner(text);
10                  Log.d("Text",text);
11                    while (scanner.hasNextLine()) {
12                      String line = scanner.nextLine();
13                      // process the line
14                      if(line.indexOf("Latitude") != -1){
15                          String[] splits = line.split(":");
16                          lat = splits[splits.length-1];
17                      }
18                      else if(line.indexOf("Longitude") != -1){
19                          String[] splits = line.split(":");
20                          lon = splits[splits.length-1];
21                      }
22                    }
23                    scanner.close();
24
25                    Log.d("Latitude",lat);
26                    Log.d("Longitude",lon);
27
28                    double latitude = Double.parseDouble(lat);
29                    double longitude = Double.parseDouble(lon);
30
31                  Intent map = new Intent(SMSdescription.this,MapViewer.class);
32                  map.putExtra("Latitude", latitude);
33                  map.putExtra("Longitude",longitude);
34                  startActivity(map);
35              }
36          });
```

รูปที่ 3.28 แสดงการออกแบบในการเขียนคำสั่งเพื่อส่งค่าละติจูดและลองจิจูดจากถาดเข้าข้อความ

คำอธิบายโค้ด

ในบรรทัดที่ 7 → สร้างตัวแปร lat และ lon ขึ้นมาเพื่อทำการเก็บค่าละติจูดและลองจิจูด

ในบรรทัดที่ 9 → Scanner ในที่นี้คือการตรวจสอบค่าจากข้อความที่ผู้รับเลือกและค้นหาจากคำว่า Latitude และ Longitude เพื่อนำตัวเลขหลังคำไปใช้ในการส่งค่าเพื่อเปิดแผนที่

ในบรรทัดที่ 32 → ส่งค่าจาก class นี้ไปยัง class ถัดไปโดยกำหนดชื่อตัวแปรให้ class ถัดไปได้เรียกใช้ คำสั่งนี้จะเป็นการส่งค่าละติจูดและลองจิจูดจากถาดเข้าข้อความ เพื่อเปิดแผน คำสั่งนี้อยู่ใน SMSDescription.class

คำสั่งแก้ข้อความที่ส่งผิด ดังแสดงในรูปที่ 3.29

```
1   String Message = "Sorry, the last message has been sent by mistake. I'm all right.";
2
3               SmsManager smsManager = SmsManager.getDefault();
4               if(swit1)
5               {
6                   smsManager.sendTextMessage(PhoneNo1, null, Message, null, null);
7               }
8               if(swit2)
9               {
10                  smsManager.sendTextMessage(PhoneNo2, null, Message, null, null);
11              }
12              if(swit3)
13              {
14                  smsManager.sendTextMessage(PhoneNo3, null, Message, null, null);
15              }
16              if(swit4)
17              {
18                  smsManager.sendTextMessage(PhoneNo4, null, Message, null, null);
19              }
20              if(swit5)
21              {
22                  smsManager.sendTextMessage(PhoneNo5, null, Message, null, null);
23              }
```

รูปที่ 3.29 แสดงการออกแบบในการเขียนคำสั่งในการแก้ข้อความที่ส่งผิด

คำอธิบายโค้ด

ในบรรทัดที่ 1 → สร้างตัวแปร String เพื่อที่จะนำตัวแปรนี้ไปใช้ในการส่งข้อความ

ในบรรทัดที่ 3 → เรียกใช้ฟังก์ชั่นในการส่งข้อความ โดยตรวจสอบหมายเลขโทรศัพท์ 5 หมายเลขจากฐานข้อมูลและตรวจสอบข้อความจากตัวแปร String ที่สร้างไว้

เมื่อมีการส่งข้อมูลออกไปโดยไม่ตั้งใจเราสามารถใช้คำสั่งนี้ในการส่งข้อความไปเพื่อบอกว่าเป็นการส่งผิดไม่ใช่การร้องขอความช่วยเหลือ คำสั่งนี้อยู่ใน Confix_EN.class

คำสั่งใช้งานปุ่มบันทึก ดังแสดงในรูปที่ 3.30

```
1  startservice.setOnClickListener(new OnClickListener() {
2          @Override
3          public void onClick(View v) {
4              // TODO Auto-generated method stub
5
6              if(chkBox1.isChecked() == true)
7              {
8                  Log.d("checkbox1", "true");
9                  userDb.UpdateData("1","checkbox1","1");
10             }
11             else{
12                 Log.d("checkbox1", "false");
13                 userDb.UpdateData("1","checkbox1","0");
14             }
15
16             if(chkBox2.isChecked() == true)
17             {
18                 Log.d("checkbox2", "true");
19                 userDb.UpdateData("1","checkbox2","1");
20             }
21             else{
22                 Log.d("checkbox2", "false");
23                 userDb.UpdateData("1","checkbox2","0");
24             }
25
26             if(chkBox3.isChecked() == true)
27             {
28                 Log.d("checkbox3", "true");
29                 userDb.UpdateData("1","checkbox3","1");
30             }
31             else{
32                 Log.d("checkbox3", "false");
33                 userDb.UpdateData("1","checkbox3","0");
34             }
```

รูปที่ 3.30 แสดงการออกแบบในการเขียนคำสั่งปุ่มบันทึก

คำอธิบายโค้ด

ในบรรทัดที่ 1 → สร้างฟังก์ชั่นการทำงานของปุ่ม

ในบรรทัดที่ 6 → ตรวจสอบการกาเครื่องหมายในช่องกาเครื่องหมาย หากมีการกาเครื่องหมายให้บันทึกค่า "1" ลงในฐานข้อมูล แต่หากไม่มีการกาเครื่องหมายให้บันทึกค่า "0" ลงในฐานข้อมูล

- เมื่อกดปุ่ม save จะทำการตรวจสอบ Checkbox ด้วยคำสั่งนี้ คำสั่งนี้อยู่ใน Confix_EN.class

คำสั่งตรวจสอบช่องว่างในช่องกรอกเบอร์โทรศัพท์ 5 หมายเลข ด้วยคำสั่งนี้ ดังแสดงในรูปที่ 3.31

```
1   if(temp1.matches("") || temp2.matches("") || temp6.matches("") ){
2
3       if(temp1.matches("")){
4           //userDb.UpdateData("1","name","-");
5       }else if(temp2.matches("")){
6           //userDb.UpdateData("1","age","-");
7       }else if(temp6.matches("")){
8           //userDb.UpdateData("1","sex","-");
9       }
10
11      Toast.makeText(getApplicationContext(), "Please fill in all the required information (*)",
12      Toast.LENGTH_LONG).show();
13
14  }else if(temp9.matches("") && temp10.matches("") && temp11.matches("") && temp12.matches("") &
15      Toast.makeText(getApplicationContext(), "Insert phone number",
16      Toast.LENGTH_LONG).show();
17      }
```

รูปที่ 3.31 แสดงการออกแบบในการเขียนคำสั่งเพื่อตรวจสอบช่องว่างในช่องกรอกเบอร์โทรศัพท์

คำอธิบายโค้ด

ในบรรทัดที่ 1 → ตรวจสอบช่องว่างในช่องกรอกข้อมูลส่วนตัวของผู้ใช้

ในบรรทัดที่ 3 → ทำการตรวจสอบช่องว่างในช่องข้อมูลที่สำคัญ (*)

ในบรรทัดที่ 11 → เมื่อกดบันทึก หากผู้ใช้กรอกข้อมูลลงในช่องที่มีเครื่องหมาย (*)ไ ม่ครบ จะมีการแจ้งเตือน “Please fill in all the required information (*)” ขึ้นมา

ในบรรทัดที่ 14 → ตรวจสอบช่องว่างในช่องกรอกเบอร์โทรศัพท์ 5 หมายเลข

ในบรรทัดที่ 15 → เมื่อกดบันทึก หากผู้ใช้ไม่กรอกเบอร์โทรศัพท์อย่างน้อย 1 ช่องจะทำการโชว์คำว่า “Insert phone number” ขึ้นมา

คำสั่งนี้จะเป็นคำสั่งที่ใช้ในการตรวจสอบช่องว่างของข้อมูลส่วนตัวและเบอร์โทรศัพท์ เพื่อที่จะไม่ต้องการให้มีค่า NULL ในฐานข้อมูล คำสั่งนี้อยู่ใน Confix_EN.class

คำสั่งจัดเก็บข้อมูลลงบนฐานข้อมูลและเริ่มการทำงานบนพื้นหลัง ดังแสดงในรูปที่ 3.32

```
1           }else{
2
3               userDb.UpdateData("1","name",temp1);
4               userDb.UpdateData("1","age",temp2);
5               userDb.UpdateData("1","sex",temp3);
6               userDb.UpdateData("1","blood",temp4);
7               userDb.UpdateData("1","[illegible]",temp5);
8               userDb.UpdateData("1","drug",temp6);
9               userDb.UpdateData("1","food",temp7);
10              userDb.UpdateData("1","[illegible]",temp8);
11              userDb.UpdateData("1","num1",temp9);
12              userDb.UpdateData("1","num2",temp10);
13              userDb.UpdateData("1","num3",temp11);
14              userDb.UpdateData("1","num4",temp12);
15              userDb.UpdateData("1","num5",temp13);
16
17              //valueDb.UpdateData("1","chkButton","1");
18
19              Toast.makeText(getApplicationContext(), "Data saved",
20              Toast.LENGTH_LONG).show();
21
22              startService(new Intent(Confix_EN.this, MyService.class));
23
24              Intent start = new Intent(Confix_EN.this,HomePage.class);
25              startActivity(start);
26              finish();
27
28              }
```

รูปที่ 3.32 แสดงการออกแบบในการเขียนคำสั่งเพื่อทำการจัดเก็บข้อมูลลงบนฐานข้อมูลและเริ่มการทำงานบนพื้นหลัง

<u>คำอธิบายโค้ด</u>

ในบรรทัดที่ 3 → บันทึกข้อมูลลงในฐานข้อมูล

ในบรรทัดที่ 19 → แสดงข้อความ “Data saved” เมื่อมีการกดปุ่มบันทึก

ในบรรทัดที่ 22 → คำสั่ง startService จะเป็นคำสั่งในการเริ่มทำงานบนพื้นหลัง

ในบรรทัดที่ 24 → แสดงหน้า HomePage เมื่อกดปุ่มบันทึก

เมื่อผู้ใช้กรอกข้อมูลครบถ้วนและกดบันทึก ข้อมูลส่วนตัวและเบอร์โทรศัพท์จะถูกบันทึกลงในฐานข้อมูลและเริ่มการทำงานบนพื้นหลัง คำสั่งนี้อยู่ใน Confix_EN.class

คำสั่งส่งข้อความด้วยตัวเอง ดังแสดงในรูปที่ 3.33

```
1    ImageView sendsms = (ImageView) this.findViewById (R.id.callen);
2           sendsms.setOnClickListener (new View.OnClickListener()
3           {
4           public void onClick ( View v )
5           {
6
7               Log.d("loop", "send");
8               Toast.makeText(getApplication(), "Send Complete " , Toast.LENGTH_LONG).show();
9               play = MediaPlayer.create(getBaseContext(), R.raw.siren2);
10              play.setLooping(true);
11              play.start();
12              sendsms();
13          }
14          });
```

รูปที่ 3.33 แสดงการออกแบบในการเขียนคำสั่งเมื่อต้องการส่งข้อมูลด้วย**ตัวเอง**

คำอธิบายโค้ด

เมื่อผู้ใช้กดปุ่มส่งข้อความจะมีเสียงไซเรนดังขึ้นและข้อความถูก**ส่งออกไปโดยเรียก**การทำงานของเมดธอด sendsms() คำสั่งนี้จะใช้งานเมื่อต้องการส่งข้อมูล**ด้วยตัวเอง** เพียงแค่สัมผัสคำสั่งนี้ข้อมูลก็จะถูกส่งออกไปทันที คำสั่งนี้อยู่ใน Homepage.class

หมายเหตุ คำสั่งนี้จะสามารถใช้งานได้ก็ต่อเมื่อสถานะระบบแสดงพร้อมใช้**งานเท่านั้น**

คำสั่งตรวจสอบสถานะของ ระบบพิกัดบอกตำแหน่ง (GPS) ดังแสดงในรูปที่ 3.34

```
1   lm.requestLocationUpdates(LocationManager.GPS_PROVIDER, 2000, 5, this);
2         // lm.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 3000, 5, this);
3          // 5 is minimum distance for update
4
5         if(!gpsEnabled && loc_db.DBsize() == 0) {
6             Log.d("1", "in");
7             gpson.setVisibility(View.INVISIBLE);
8             gpsoff.setVisibility(View.VISIBLE);
9             ready.setVisibility(View.INVISIBLE);
10            notready.setVisibility(View.VISIBLE);
11
12            sendsms.setVisibility(View.INVISIBLE);
13        }
14        else if(!gpsEnabled && loc_db.DBsize() != 0){
15            Log.d("2", "in");
16            gpson.setVisibility(View.INVISIBLE);
17            gpsoff.setVisibility(View.VISIBLE);
18            ready.setVisibility(View.VISIBLE);
19            notready.setVisibility(View.INVISIBLE);
20            sendsms.setVisibility(View.VISIBLE);
21        }
22        else if(gpsEnabled && loc_db.DBsize() == 0){
23            Log.d("3", "in");
24            gpson.setVisibility(View.VISIBLE);
25            gpsoff.setVisibility(View.INVISIBLE);
26            ready.setVisibility(View.INVISIBLE);
27            notready.setVisibility(View.VISIBLE);
28            sendsms.setVisibility(View.INVISIBLE);
29        }
30        else {
31            Log.d("4", "in");
32            gpson.setVisibility(View.VISIBLE);
33            gpsoff.setVisibility(View.INVISIBLE);
34            ready.setVisibility(View.VISIBLE);
35            notready.setVisibility(View.INVISIBLE);
36            sendsms.setVisibility(View.VISIBLE);
37        }
38    }
```

รูปที่ 3.34 แสดงการออกแบบในการเขียนคำสั่งเพื่อตรวจสอบสถานะของ ระบบพิกัดบอกตำแหน่ง (GPS)

คำอธิบายโค้ด

ในบรรทัดที่ 1 → เปิดใช้งานฟังก์ชั่นระบบระบุตำแหน่งและเก็บค่าทุกๆ 2 วินาที

ในบรรทัดที่ 5 → ตรวจสอบว่าระบบระบุตำแหน่งถูกเปิดอยู่หรือไม่และค่าถูกบันทึกลงในฐานข้อมูลแล้วหรือยัง หากระบบระบุตำแหน่งเปิดอยู่จะแสดงสีเขียว แต่หากปิดอยู่จะแสดงสีแดง และหากมีข้อมูลตำแหน่งถูกบันทึกลงในฐานข้อมูลจะแสดงสีเขียว แต่หากไม่มีข้อมูลตำแหน่งถูกบันทึกลงในฐานข้อมูลจะแสดงสีแดง

คำสั่งนี้ใช้เพื่อตรวจสอบสถานะของ ระบบพิกัดบอกตำแหน่ง (GPS) คำสั่งนี้อยู่ใน Service.class

คำสั่งตรวจสอบข้อความจากถาดเข้าข้อความ ดังแสดงในรูปที่ 3.35

```
1    Uri uriSMSURI = Uri.parse("content://sms/inbox");
2    Cursor cur = getContentResolver().query(uriSMSURI, null, null, null,null);
3
4    Scanner scanner = new Scanner(line);
5    -                  while (scanner.hasNextLine()) {
6    |                    String lineTemp = scanner.nextLine();
7    -                    if(lineTemp.indexOf("Help!!!") != -1){
8    |                        switchs = true;
9    |                        break;
10   |                    }
11   L                  }
12                      scanner.close();
```

รูปที่ 3.35 แสดงการออกแบบในการเขียนคำสั่งเพื่อตรวจสอบข้อความจากถาด**เข้าข้อความ**

คำอธิบายโค้ด

ในบรรทัดที่ 1 → ตรวจสอบถาดข้อความจากถาดเข้าข้อความ โดยส่วนใหญ่ข้อความจะอยู่ที่ "content://sms/inbox"

ในบรรทัดที่ 4 → ตรวจสอบข้อความทั้งหมดจากถาดเข้าข้อความและคัดเลือกเฉพาะข้อความที่มีคำว่า "IHelp !!!"

คำสั่งนี้ใช้เพื่อตรวจสอบตรวจสอบข้อความจากถาดเข้าข้อความ คำสั่งนี้อยู่ใน GetSMS.class

คำสั่งเปิดแผนที่จาก Google map ดังแสดงในรูปที่ 3.36

```
1  private void setUpMapIfNeeded() {
2      if (mMap != null) {
3
4          Bundle bundle = getIntent().getExtras();
5
6          double Latitude = bundle.getDouble("Latitude");
7          double Longitude = bundle.getDouble("Longtitude");
8         //double Latitude = 13.772;
9         //double Longitude = 100.572;
10
11         LatLng mylocation = new LatLng(Latitude, Longitude);
12
13        mMap.setMyLocationEnabled(true);
14        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(mylocation, 16));
15
16        mMap.addMarker(new MarkerOptions()
17                .title("Emergency")
18                .snippet("Help!!!")
19                .position(mylocation));
20    }
21    mMap = ((SupportMapFragment) getSupportFragmentManager().findFragmentById(R.id.map)).getMap();
22    if (mMap == null) {
23        return;
24    }
```

รูปที่ 3.36 แสดงการออกแบบในการเขียนคำสั่งเพื่อเปิดแผนที่จาก Google map

<u>คำอธิบาย</u>โค้ด

คำสั่งนี้จะเป็นการเรียกแผนที่จาก Google map โดยจะใช้พิกัดจากค่า Latitude และ Longitude จากการคัดเลือกข้อความที่มีคำว่า "Help !!!" มาใช้ในการเปิดแผนที่ โดยที่เมื่อผู้ใช้เลือกข้อความที่ต้องการแล้วจะทำการส่งค่า Latitude และ Longitude มายังคลาสนี้เพื่อนำค่าไปใส่ในตัวแปร (Latitude,Longitude) แผนที่จะแสดงพร้อมกับค่าการขยายที่ระดับ 18 ซึ่งจะเป็นระดับขยายที่ทาง Google ตั้งให้เป็นพื้นฐาน และผู้ใช้สามารถย่อหรือขยายแผนที่ได้ตามต้องการ โดยในบรรทัดที่ 16 จะทำให้ผู้ใช้เห็นจุดที่ผู้ต้องการขอความช่วยเหลืออยู่จากการแสดงจุดสังเกต ดังนั้นคำสั่งนี้ใช้เพื่อตรวจสอบข้อความจากถาดเข้าข้อความ คำสั่งนี้อยู่ใน MapViewer.class

คำสั่งตั้งค่าเสียงไซเรนเป็นระดับสูงสุดของโทรศัพท์ ดังแสดงในรูปที่ 3.37

```
1    int origionalVolume = mAudioManager.getStreamVolume(AudioManager.STREAM_MUSIC);
2    mAudioManager.setStreamVolume(AudioManager.STREAM_MUSIC,
3    mAudioManager.getStreamMaxVolume(AudioManager.STREAM_MUSIC), 15);//range(0-15)
```

รูปที่ 3.37 แสดงการออกแบบในการเขียนคำสั่งตั้งค่าเสียงไซเรนเป็นระดับสูงสุดของโทรศัพท์

<u>คำอธิบายโค้ด</u>

คำสั่งนี้จะเป็นการตั้งค่าเสียงให้อยู่ในระดับสูงที่สุด โดยที่ระดับ "0" จะเป็นระดับปกติ ระดับ "15" คือระดับที่สูงที่สุด คำสั่งนี้จะอยู่ใน HomePage.class

บทที่ 4

ผลการทดลอง

ในบทนี้กล่าวถึงกระบวนการทำงาน การทดสอบการทำงาน และวิเคราะห์ผลการทำงาน ของโปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉิน (Application for security rescue)

4.1 กระบวนการทำงานของโปรแกรมประยุกต์

กระบวนการทำงานของโปรแกรมประยุกต์จะแบ่งการทำงานเป็น 3 ส่วนหลัก คือ

4.1.1 การตั้งค่า

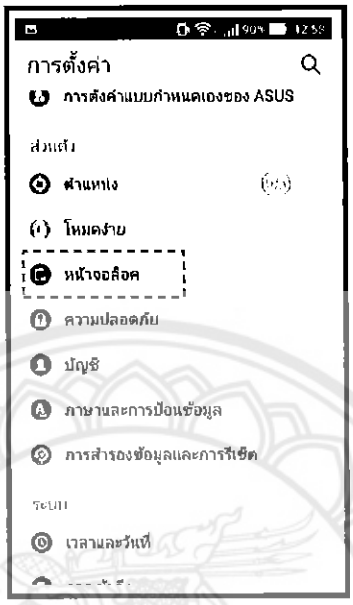
การตั้งค่าภายนอกโปรแกรมประยุกต์ เมื่อติดตั้งโปรแกรมประยุกต์เสร็จเรียบร้อยแล้ว ผู้ใช้ควรจะต้องสร้างทางลัดเข้าสู่โปรแกรมประยุกต์บนหน้าจอหลักและไม่ควรมีการใส่รหัสเพื่อเข้าใช้งานโทรศัพท์ เพื่อให้ผู้ใช้งานที่ต้องการขอความช่วยเหลือสามารถเข้าสู่โปรแกรมประยุกต์ได้อย่างสะดวกและรวดเร็ว

วิธีการตั้งค่าไม่ให้มีการใส่รหัสบนเครื่องโทรศัพท์

- กดปุ่มการตั้งค่า ในหน้าจอเมนูของโทรศัพท์ ดังแสดงในรูปที่ 4.1

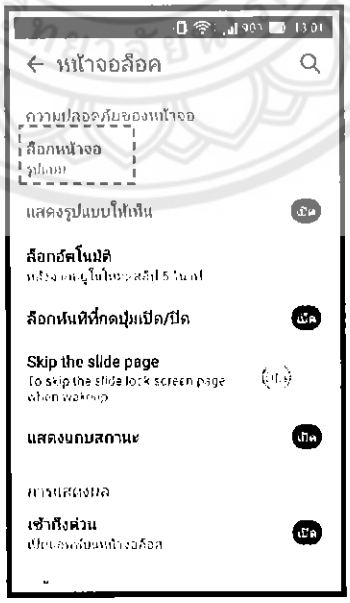
รูปที่ 4.1 หน้าจอเมนูบนโทรศัพท์

- เข้าสู่หน้าการตั้งค่า เลือกหน้าจอล็อค ดังแสดงในรูปที่ 4.2



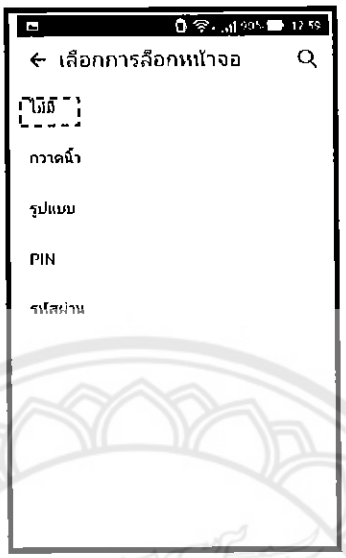
รูปที่ 4.2 หน้าการตั้งค่า

- เข้าสู่หน้าหน้าจอล็อค เลือกรูปแบบล็อคหน้าจอ ดังแสดงในรูปที่ 4.3



รูปที่ 4.3 หน้าหน้าจอล็อค

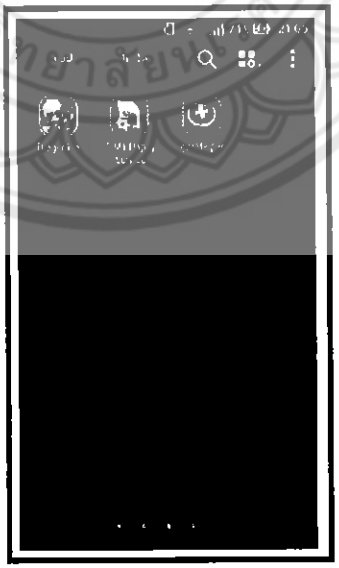
- เข้าสู่หน้าเลือกการล็อคหน้าจอ เลือกไม่มี ดังแสดงในรูปที่ 4.4



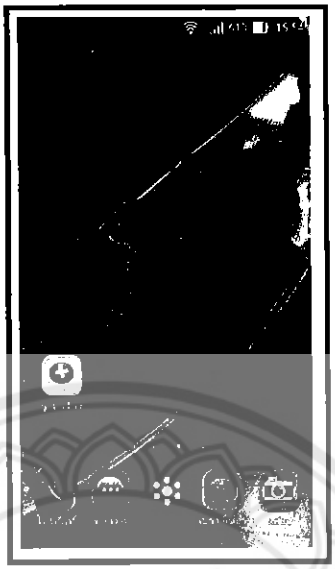
รูปที่ 4.4 หน้าเลือกการล็อคหน้าจอ

วิธีการสร้างทางลัดเข้าสู่โปรแกรมประยุกต์บนหน้าจอหลัก

- กดปุ่มโปรแกรมประยุกต์ในหน้าจอเมนูของโทรศัพท์ค้างไว้ จนกว่าจะแสดงหน้าจอหลัก แล้วปล่อยปุ่มโปรแกรมประยุกต์บนหน้าจอหลัก ดังแสดงในรูปที่ 4.5 และ 4.6



รูปที่ 4.5 หน้าเมนูบนหน้าจอโทรศัพท์



รูปที่ 4.6 หน้าจอหลัก

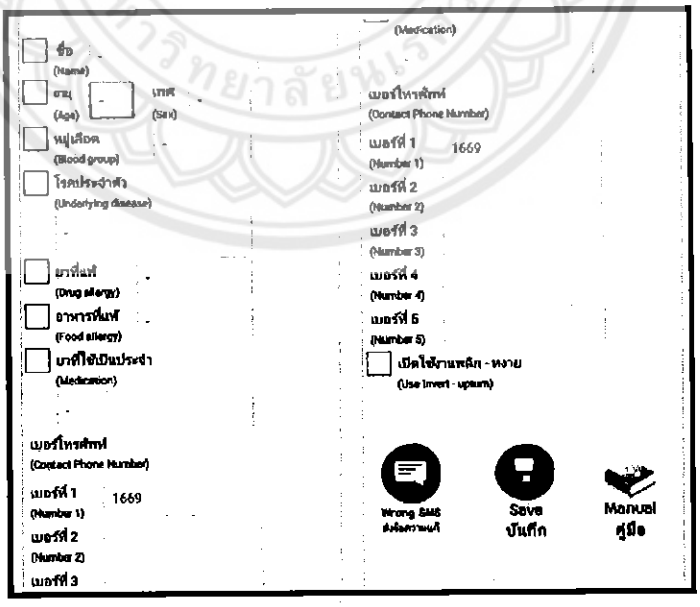
การตั้งค่าภายในโปรแกรมประยุกต์ เมื่อผู้ใช้เข้าสู่โปรแกรมประยุกต์ ผู้ใช้จะต้องกรอกข้อมูลส่วนตัวของผู้ใช้ให้ครบถ้วนทุกช่อง เลือกช่องสำหรับกาเครื่องหมายในส่วนที่ต้องการส่งข้อมูลส่วนตัวไปพร้อมกับข้อความและเลือกวิธีการส่งตามที่ผู้ใช้เห็นว่าสะดวกต่อผู้ใช้มากที่สุด การตั้งค่าจะทำเพียงครั้งเดียวหลังจากติดตั้งโปรแกรมประยุกต์ ซึ่งหากผู้ใช้ไม่สะดวกในการตั้งค่าเองผู้ใช้สามารถให้ผู้อื่นช่วยตั้งค่าได้ เมื่อตั้งค่าเสร็จแล้วผู้ใช้สามารถเริ่มใช้งานได้ทันที

วิธีการตั้งค่าภายในโปรแกรมประยุกต์

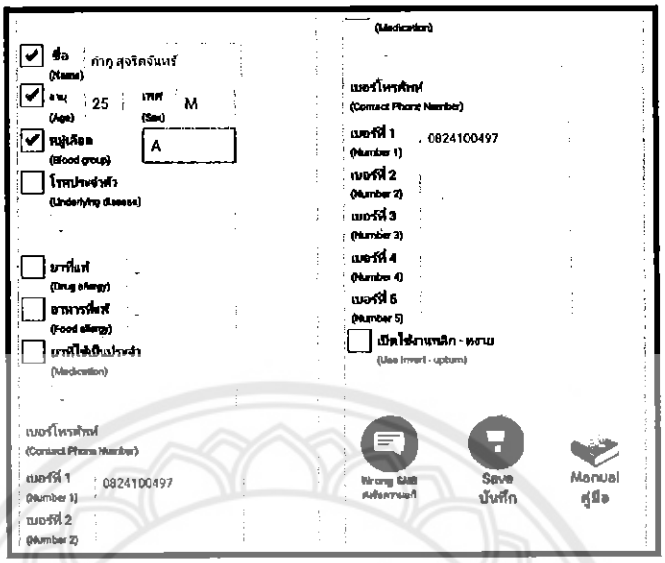
- เข้าสู่โปรแกรมประยุกต์ เลือกตั้งค่าในหน้าแรก ดังแสดงในรูปที่ 4.7

รูปที่ 4.7 หน้าแรก (Homepage)

- เมื่อเข้าสู่หน้าตั้งค่าให้ผู้ใช้กรอกข้อมูลให้ครบทุกช่องและกาเครื่องหมายในช่องของข้อมูลที่ต้องการส่งไปพร้อมกับข้อความ ดังแสดงในรูปที่ 4.8 และ 4.9

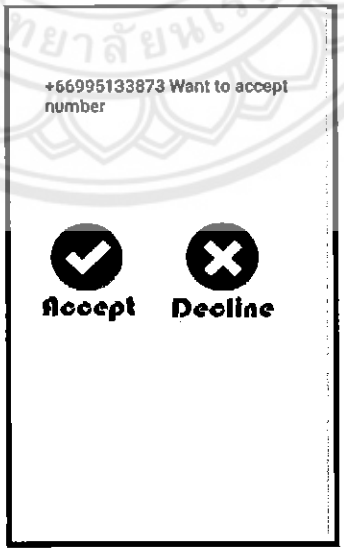


รูปที่ 4.8 หน้าตั้งค่า (Setting) เมื่อเริ่มการทำงานของโปรแกรมประยุกต์ครั้งแรก

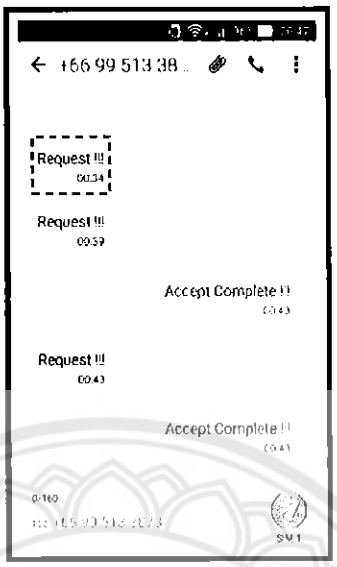


รูปที่ 4.9 หน้าตั้งค่า (Setting) เมื่อกรอกข้อมูลครบทุกช่อง

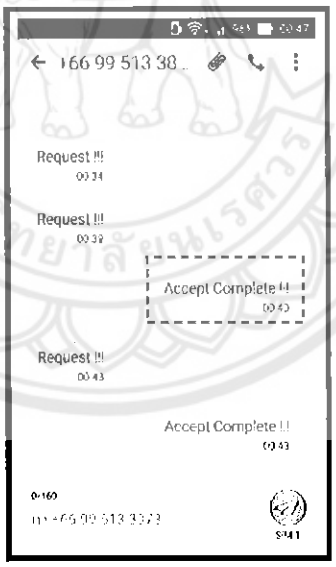
- เมื่อกรอกข้อมูลครบทุกช่องแล้วให้กดปุ่มบันทึก โปรแกรมประยุกต์จะทำการส่งข้อความไปยังหมายเลขปลายทางที่ระบุไว้เพื่อขอการยืนยัน หมายเลขปลายทางที่ระบุไว้ต้องมีโปรแกรมประยุกต์ติดตั้งไว้ จึงจะสามารถตอบรับการยืนยันเบอร์โทรศัพท์ได้ ดังรูปที่ 4.10 เมื่อหมายเลขปลายทางยืนยันแล้วจึงจะสามารถใช้งานการขอความช่วยเหลือได้



รูปที่ 4.10 หน้าตอบรับการยืนยันในฝั่งผู้รับ



รูปที่ 4.11 ข้อความการร้องขอการยืนยันที่ผู้รับได้รับ

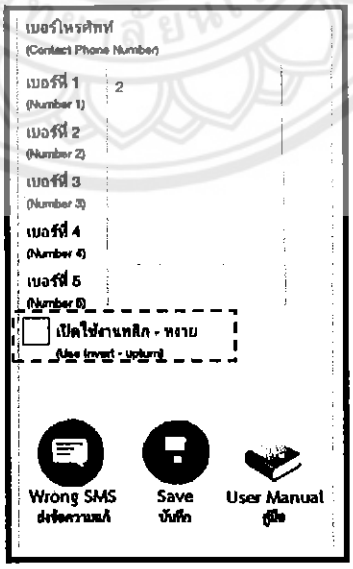


รูปที่ 4.12 ข้อความตอบรับการยืนยันที่ผู้รับได้ส่งกลับ

4.1.2 **การส่ง**

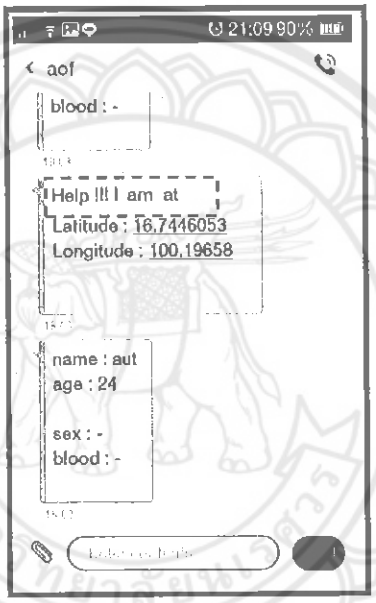
การส่งผู้ใช้จะต้องเลือกวิธีการส่งจากการตั้งค่า โดยผู้ใช้สามารถเลือกวิธีการส่งแบบกดปุ่มด้วยตัวผู้ใช้เองภายในโปรแกรมประยุกต์ ดังแสดงในรูปที่ 4.13 หรือวิธีการส่งแบบพลิก – หงาย ซึ่งวิธีการส่งแบบพลิก – หงาย ดังแสดงในรูปที่ 4.14 ผู้ใช้จะต้องเข้าไปตั้งค่าในส่วนของการเลือกส่งแบบพลิก – หงายจึงจะสามารถใช้งานได้

รูปที่ 4.13 ปุ่มส่งข้อความโดยผู้ใช้



รูปที่ 4.14 เลือกส่งแบบพลิก – หงายในหน้าตั้งค่า

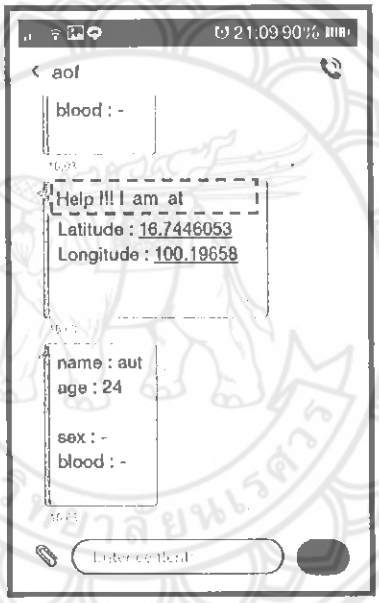
เมื่อผู้ใช้ส่งข้อความขอความช่วยเหลือ ข้อความจะถูกส่งไปยังหมายเลขปลายทางที่ผู้ใช้ระบุไว้ ภายในข้อความที่ส่งออกจะระบุถึงข้อมูลส่วนตัวที่ผู้ใช้ได้เลือกและกรอกข้อมูลไว้ในส่วนของการตั้งค่า และตำแหน่งพิกัดของผู้ใช้ในลักษณะตัวเลขละติจูดและลองจิจูด โดยผู้รับข้อความจะสามารถสังเกตได้จากคำว่า "Help !!!" ทำให้รู้ว่าผู้ใช้ต้องการขอความช่วยเหลืออยู่ นอกจากนี้เมื่อกดส่งข้อความแล้วจะมีเสียงไซเรนดังขึ้นมาจากโทรศัพท์ ทำให้ผู้ต้องการความช่วยเหลือได้รับการช่วยเหลือได้เร็วขึ้น ดังแสดงในรูปที่ 4.15



รูปที่ 4.15 ภาคเข้าข้อความของผู้รับ

4.1.3 การรับและดูแผนที่

ในส่วนของผู้รับหากได้รับข้อความขอความช่วยเหลือ ผู้รับสามารถอ่านข้อความในถาดเข้าของข้อความหรือเข้าสู่โปรแกรมประยุกต์และกดปุ่มผู้รับ (Receive) โปรแกรมประยุกต์จะทำการเลือกข้อความจากถาดเข้าของข้อความภายในโทรศัพท์ที่มีคำว่า "Help !!!" และจะแสดงตารางข้อความที่เข้าข่ายให้ผู้รับเลือก เมื่อผู้รับเลือกข้อความแล้วจะแสดงหน้ารายละเอียดข้อความในแบบตัวเลขละติจูดและลองจิจูดเพื่อให้ผู้รับสามารถนำไปค้นหาได้หรือผู้รับสามารถกดปุ่มแสดงแผนที่เพื่อเปิดแผนที่จากภายในโปรแกรมประยุกต์ก็ได้ ดังแสดงในรูปที่ 4.16



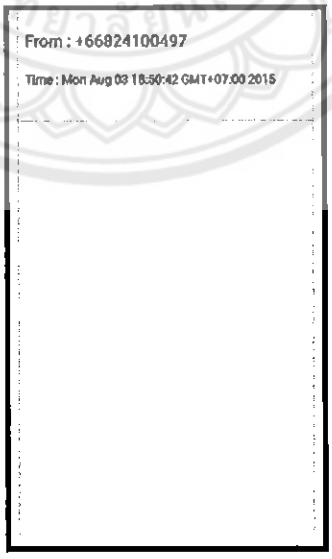
รูปที่ 4.16 ถาดเข้าข้อความของผู้รับ

วิธีการใช้งานในส่วนผู้รับของโปรแกรมประยุกต์

- เมื่อได้รับข้อความขอความช่วยเหลือ ให้ผู้รับเข้าสู่โปรแกรมประยุกต์และกดปุ่มผู้รับ (Receiver) ดังแสดงในรูปที่ 4.17

รูปที่ 4.17 หน้าแรก (Homepage)

- เมื่อกดปุ่มผู้รับ (Receiver) แล้วจะพบกับหน้ารายการข้อความ ให้ผู้รับเลือกข้อความจากรายการข้อความที่ต้องการเปิดแผนที่ ดังแสดงในรูปที่ 4.18

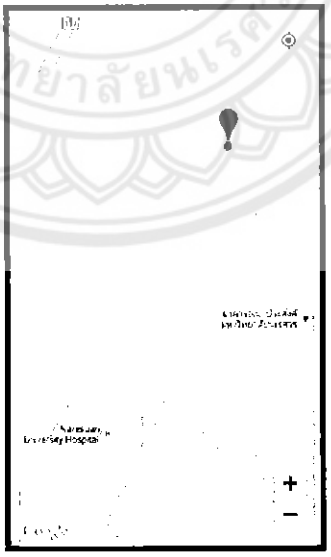


รูปที่ 4.18 หน้าแสดงรายการข้อความของผู้รับ

- เมื่อเลือกข้อความจากรายการข้อความแล้ว ผู้รับจะพบกับหน้าแสดงตัวเลขตำแหน่งของผู้ขอความช่วยเหลือ ดังแสดงในรูปที่ 4.19

รูปที่ 4.19 หน้าแสดงตัวเลขตำแหน่งของผู้ขอความช่วยเหลือ (ละติจูดและลองจิจูด)

- ให้ผู้รับกดปุ่มแผนที่ (Map) เพื่อแสดงตำแหน่งที่ผู้ส่งข้อความอยู่



รูปที่ 4.20 หน้าแสดงแผนที่จากตัวเลขตำแหน่งจากรูปที่ 4.19

4.1.4 **การส่งข้อความแก้เมื่อมีการส่งข้อความโดยไม่ได้ตั้งใจ**

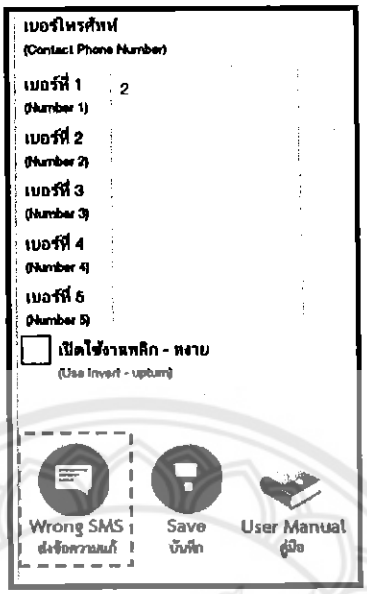
ในกรณีที่ผู้ใช้ได้ส่งข้อความขอความช่วยเหลือออกไปโดยที่ไม่ได้ตั้งใจ ผู้ใช้สามารถส่งข้อความแก้ เพื่อบอกให้ผู้รับรับรู้ว่าข้อความที่ผู้ใช้ส่งไปนั้นเป็นการส่งโดยไม่ได้ตั้งใจ

วิธีการส่งข้อความแก้

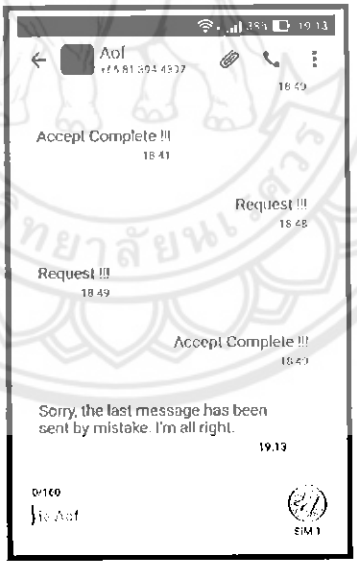
- เมื่อข้อความถูกส่งออกไปโดยที่ผู้ใช้ไม่ได้ตั้งใจ ให้ผู้ใช้เปิดโปรแกรมประยุกต์ แล้วเลือกตั้งค่าดังแสดงในรูปที่ 4.21

- เมื่อผู้ใช้เข้ามายังหน้าตั้งค่า ให้ผู้ใช้กดปุ่มส่งข้อความแก้ (Wrong SMS) ข้อความแก้จะถูกส่งไปยังเลขหมายปลายทางที่ระบุไว้ ดังแสดงในรูปที่ 4.22

รูปที่ 4.21 ปุ่มตั้งค่า



รูปที่ 4.22 ปุ่มส่งข้อความแก้

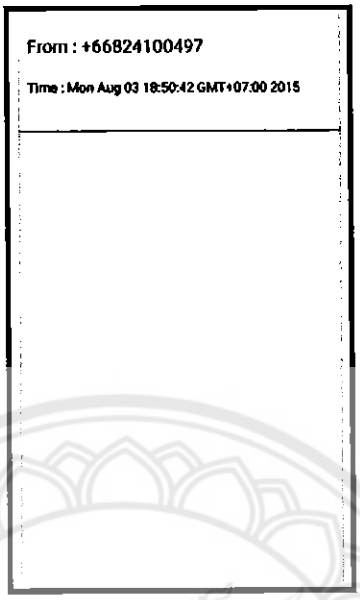


รูปที่ 4.23 ตัวอย่างข้อความที่ผู้รับได้รับเมื่อกดปุ่มส่งข้อความแก้

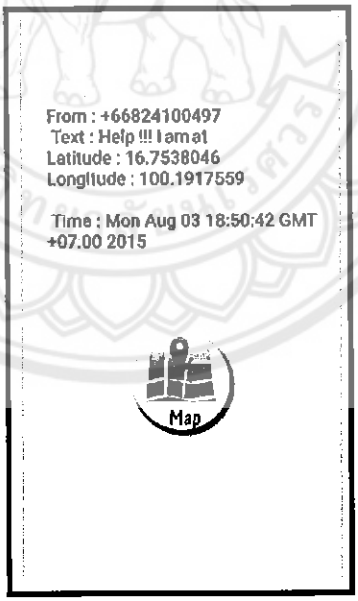
4.2 การทดสอบการทำงานของโปรแกรมประยุกต์

รูปที่ 4.24 ตัวอย่างข้อความที่ผู้รับได้รับจากผู้ต้องการขอความช่วยเหลือ

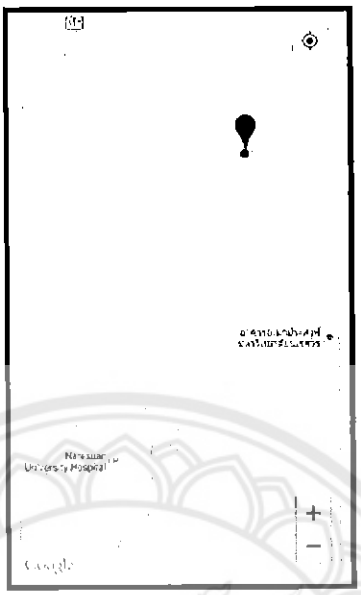
ในส่วนของผู้รับหากได้รับข้อความขอความช่วยเหลือ ให้เปิดโปรแกรมประยุกต์แล้วกดปุ่มผู้รับ (Receiver) จะแสดงรายการข้อความขอความช่วยเหลือ ดังแสดงในรูปที่ 4.25 ให้เลือกรายการข้อความที่ต้องการ จะแสดงตัวเลขตำแหน่งของผู้ขอความช่วยเหลือ (ละติจูดและลองจิจูด) ดังแสดงในรูปที่ 4.26 ถ้าต้องการให้แสดงแผนที่ให้กดปุ่มแผนที่ (Map) จะแสดงหน้าแผนที่และจุดที่ผู้ขอความช่วยเหลืออยู่ เพื่อจะสามารถขอความช่วยเหลือจากบริเวณใกล้เคียงหรือสามารถเข้าไปช่วยเหลือเองได้อย่างรวดเร็ว ดังแสดงในรูปที่ 4.27



รูปที่ 4.25 หน้าผู้รับ (Receiver) แสดงรายการข้อความของผู้ขอความช่วยเหลือ



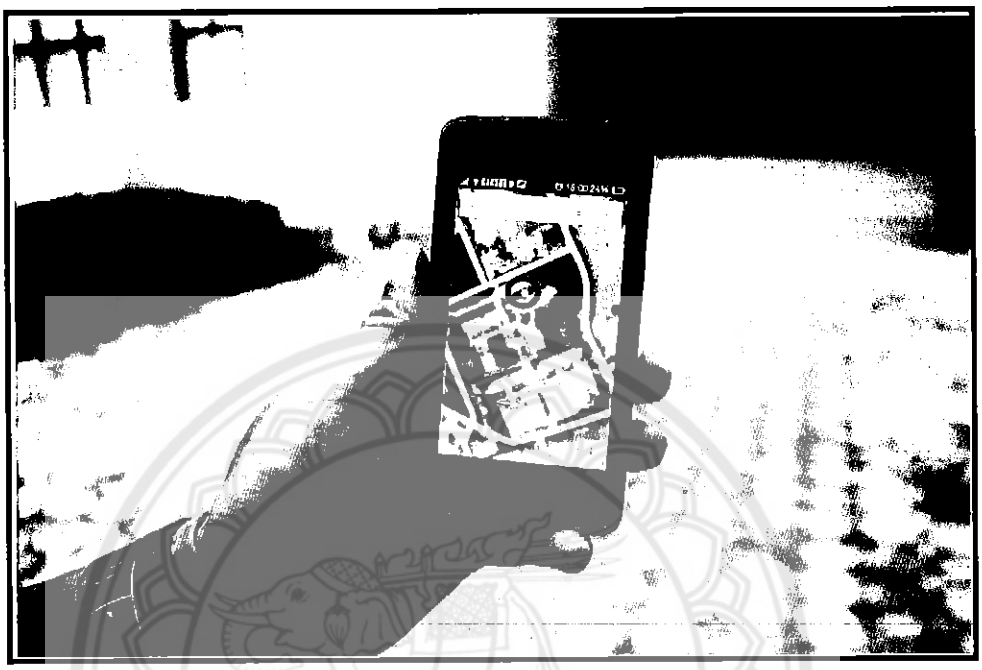
รูปที่ 4.26 หน้าผู้รับ (Receiver) แสดงตัวเลขตำแหน่งของผู้ขอความช่วยเหลือ (ละติจูดและลองจิจูด)



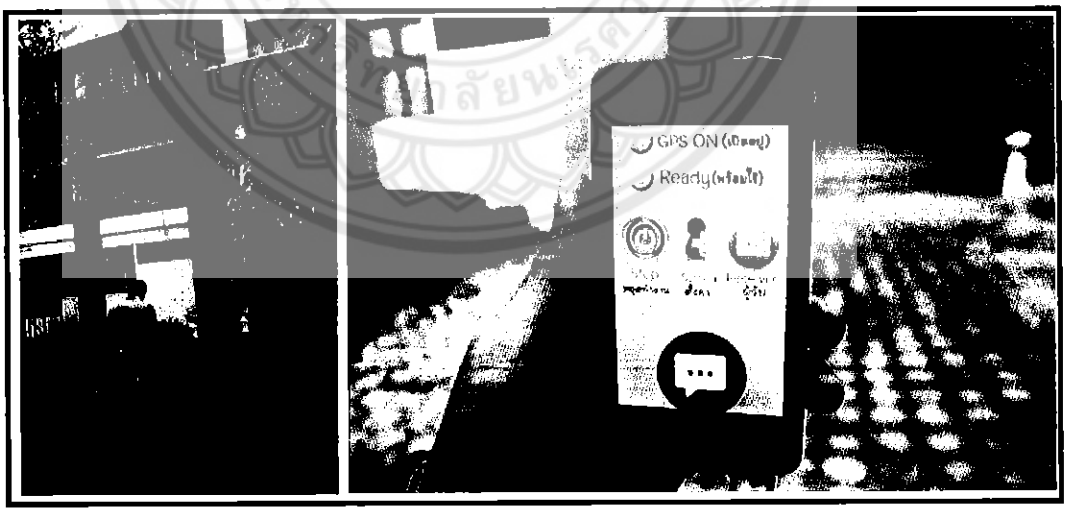
รูปที่ 4.27 หน้าผู้รับ (Receiver) แสดงแผนที่จากตัวเลขตำแหน่งจากรูปที่ 4.26

เมื่อทำการทดสอบสถานที่บริเวณเดียวกัน แต่ทดสอบเมื่อขอความช่วยเหลือจากภายนอกอาคารดังแสดงในรูปที่ 4.28 และภายในอาคารดังแสดงในรูปที่ 4.31 เปรียบเทียบกัน พบว่าตำแหน่งที่ระบุอยู่บนแผนที่จะไม่ได้ระบุอยู่บนตัวอาคาร แต่จะอยู่บริเวณใกล้เคียงกับอาคาร ซึ่งระยะทางจากจุดที่ขอความช่วยเหลือกับจุดที่ระบุอยู่บนแผนที่มีระยะทางห่างกันไม่มากนัก ดังแสดงในรูปที่ 4.34

4.2.1 ทดสอบขอความช่วยเหลือภายนอกอาคาร



รูปที่ 4.28 แสดงแผนที่จาก Google map ก่อนทำการขอความช่วยเหลืออยู่บริเวณหน้าอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร



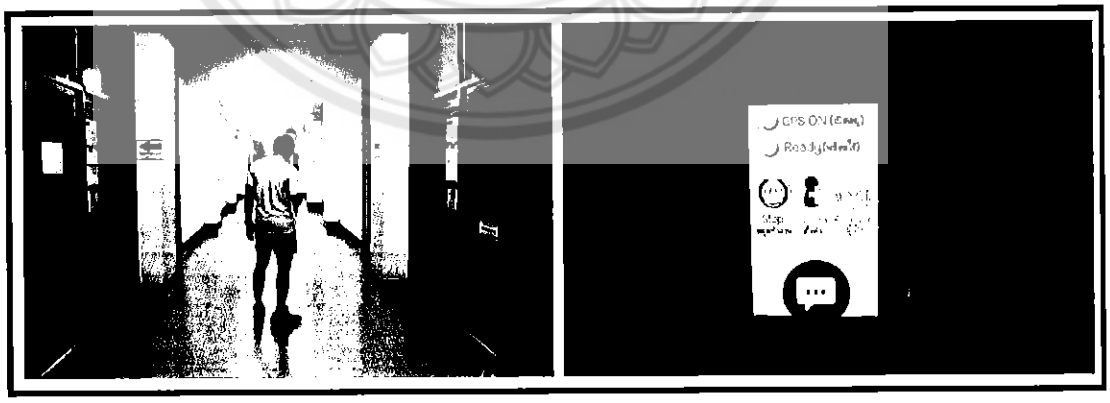
รูปที่ 4.29 แสดงสถานะพร้อมก่อนทำการขอความช่วยเหลือ อยู่บริเวณหน้าอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร

รูปที่ 4.30 แสดงข้อความการขอความช่วยเหลือ และแผนที่ที่ได้จากโปรแกรม อยู่บริเวณหน้าอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร

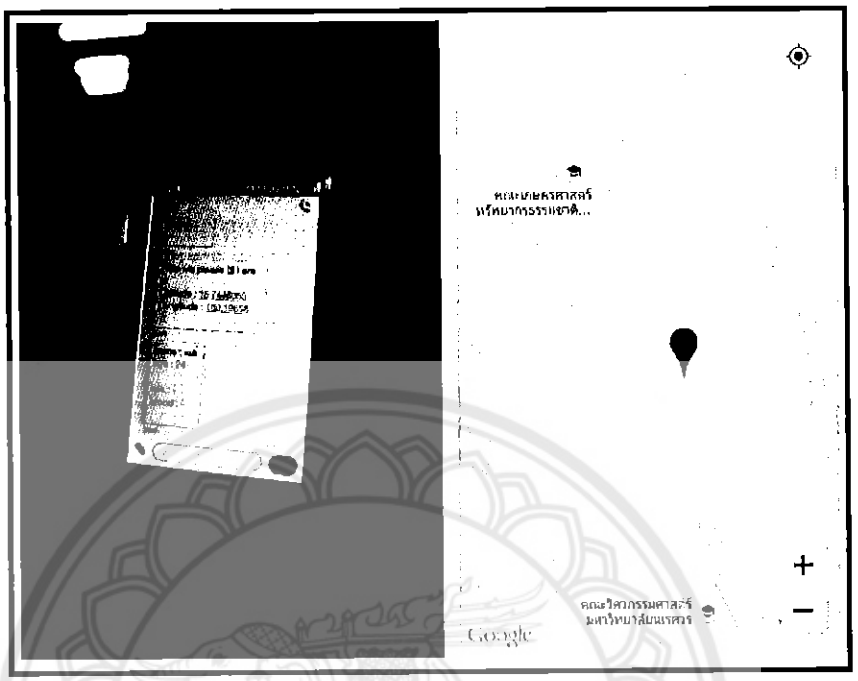
เมื่อทำการขอความช่วยเหลือจากภายนอกอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร บริเวณหน้าอาคารดังรูปที่ 4.28 โดยตรวจสอบตำแหน่งจากการเปิดโปรแกรมประยุกต์ Google map และกดปุ่มส่งข้อความจากโปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉินดังรูปที่ 4.29 พบว่าเมื่อเปิดแผนที่จากภายในโปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉินในหน้าผู้รับดังรูปที่ 4.30 ตำแหน่งของผู้ส่งจากโปรแกรมประยุกต์ Google map จะตรงกับตำแหน่งของโปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉิน

4.2.2 **ทดสอบขอความช่วยเหลือจากภายในอาคาร**

รูปที่ 4.31 แสดงแผนที่จาก Google map ก่อนทำการขอความช่วยเหลืออยู่บริเวณด้านในอาคาร วิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร



รูปที่ 4.32 แสดงสถานะพร้อมก่อนทำการขอความช่วยเหลือ อยู่บริเวณภายในอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร เมื่อไม่มีสัญญาณระบบระบุตำแหน่ง



รูปที่ 4.33 แสดงแผนที่เมื่อทำการขอความช่วยเหลืออยู่บริเวณภายในอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร เมื่อไม่มีสัญญาณระบบระบุตำแหน่ง

เมื่อทำการขอความช่วยเหลือจากภายในอาคารวิศวกรรมไฟฟ้า มหาวิทยาลัยนเรศวร บริเวณชั้นหนึ่งของอาคารดังรูปที่ 4.31 และตรวจสอบตำแหน่งจากการเปิดโปรแกรมประยุกต์ Google map จะพบว่าตำแหน่งปัจจุบันกับตำแหน่งที่แสดงบนโปรแกรมประยุกต์ Google map ไม่ตรงกัน แต่จะมีระยะห่างจากตำแหน่งปัจจุบันพอสมควรและเมื่อกดปุ่มส่งข้อความจากโปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉินดังรูปที่ 4.32 และเปิดแผนที่จากภายในโปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉินในหน้าผู้รับดังรูปที่ 4.33 ตำแหน่งของผู้ส่งปัจจุบันจะคลาดเคลื่นกับตำแหน่งของโปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉินเพียงเล็กน้อย

4.2.3 **ทดสอบการส่งข้อความขอความช่วยเหลือ**

ทดสอบโดยการทดลองส่งข้อความขอความช่วยเหลือภายในบริเวณมหาวิทยาลัยนเรศวร โดยการส่งในรูปแบบส่งโดยการกดปุ่มส่งและส่งโดยพลิก – หงาย และเปรียบเทียบพิกัดที่ผู้รับได้รับเมื่อเปรียบเทียบกับพิกัดจริงที่หาได้จาก Google map เมื่ออยู่ภายในอาคารและจากโปรแกรมประยุกต์แสดงเลขตำแหน่งพิกัด รวมถึงการทดสอบข้อผิดพลาดของการส่งข้อความ โดยการทดลองเปิดโปรแกรมประยุกต์และใช้โทรศัพท์ตามปกติในชีวิตประจำวันเป็นเวลา 1 สัปดาห์

ตารางที่ 4.1 ตารางทดสอบการส่งข้อความขอความช่วยเหลือโดยการกดปุ่มส่งข้อความบริเวณภายในมหาวิทยาลัยนเรศวร

ครั้งที่	บริเวณที่ส่งข้อความ	เสียง Alarm	ผู้รับ	พิกัดที่ผู้รับได้รับ	ความคลาดเคลื่อน
1	ภายในอาคาร วิศวกรรมไฟฟ้า 16.744650, 100.196754	ดัง	มีข้อความเข้า	16.7446053, 100.19658	19.2 เมตร
2	ภายในอาคาร วิศวกรรมไฟฟ้า 16.744650, 100.196754	ดัง	มีข้อความเข้า	16.7446053, 100.19658	19.2 เมตร
3	ภายในอาคาร วิศวกรรมไฟฟ้า 16.744650, 100.196754	ดัง	มีข้อความเข้า	16.7446053, 100.19658	19.2 เมตร
4	ภายในอาคาร วิศวกรรมไฟฟ้า 16.744650, 100.196754	ดัง	มีข้อความเข้า	16.7446053, 100.19658	19.2 เมตร
5	ภายในอาคาร วิศวกรรมไฟฟ้า 16.744650, 100.196754	ดัง	มีข้อความเข้า	16.7446053, 100.19658	19.2 เมตร
6	ภายนอกอาคาร วิศวกรรมไฟฟ้า 16.74478, 100.196661	ดัง	มีข้อความเข้า	16.744781, 100.196660	1.53 เมตร

ครั้งที่	บริเวณที่ส่งข้อความ	เสียง Alarm	ผู้รับ	พิกัดที่ผู้รับได้รับ	ความคลาดเคลื่อน
7	ภายนอกอาคารวิศวกรรมไฟฟ้า 16.74478, 100.196661	ดัง	มีข้อความเข้า	16.744780, 100.196662	0.106 เมตร
8	ภายนอกอาคารวิศวกรรมไฟฟ้า 16.74478, 100.196661	ดัง	มีข้อความเข้า	16.744781, 100.196662	0.153 เมตร
9	ภายนอกอาคารวิศวกรรมไฟฟ้า 16.74478, 100.196661	ดัง	มีข้อความเข้า	16.744780, 100.196660	0.106 เมตร
10	ภายนอกอาคารวิศวกรรมไฟฟ้า 16.74478, 100.196661	ดัง	มีข้อความเข้า	16.744780, 100.196663	0.21 เมตร
11	บริเวณโรงอาหารคณะเกษตรศาสตร์ 16.746029, 100.195401	ดัง	มีข้อความเข้า	16.746035, 100.195405	1.92 เมตร
12	บริเวณโรงอาหารคณะเกษตรศาสตร์ 16.746029, 100.195401	ดัง	มีข้อความเข้า	16.746030, 100.195410	0.96 เมตร
13	บริเวณโรงอาหารคณะเกษตรศาสตร์ 16.746029, 100.195401	ดัง	มีข้อความเข้า	16.746030, 100.195405	0.44 เมตร
14	บริเวณโรงอาหารคณะเกษตรศาสตร์ 16.746029, 100.195401	ดัง	มีข้อความเข้า	16.746032, 100.195407	0.72 เมตร
15	บริเวณโรงอาหารคณะเกษตรศาสตร์ 16.746029, 100.195401	ดัง	มีข้อความเข้า	16.746028, 100.195400	0.15 เมตร

ครั้งที่	บริเวณที่ส่งข้อความ	เสียง Alarm	ผู้รับ	พิกัดที่ผู้รับได้รับ	ความคลาดเคลื่อน
16	ภายในอาคาร คณะสหเวชศาสตร์ 16.745780, 100.189201	ดัง	มีข้อความเข้า	16.745773, 100.189026	18.68 เมตร
17	ภายในอาคาร คณะสหเวชศาสตร์ 16.745780, 100.189201	ดัง	มีข้อความเข้า	16.745773, 100.189026	18.68 เมตร
18	ภายในอาคาร คณะสหเวชศาสตร์ 16.745780, 100.189201	ดัง	มีข้อความเข้า	16.745773, 100.189026	18.68 เมตร
19	ภายในอาคาร คณะสหเวชศาสตร์ 16.745780, 100.189201	ดัง	มีข้อความเข้า	16.745773, 100.189026	18.68 เมตร
20	ภายในอาคาร คณะสหเวชศาสตร์ 16.745780, 100.189201	ดัง	มีข้อความเข้า	16.745773, 100.189026	18.68 เมตร
21	ภายนอกอาคาร คณะสหเวชศาสตร์ 16.745915, 100.189384	ดัง	มีข้อความเข้า	16.745913, 100.189383	0.24 เมตร
22	ภายนอกอาคาร คณะสหเวชศาสตร์ 16.745915, 100.189384	ดัง	มีข้อความเข้า	16.745913, 100.189387	0.38 เมตร
23	ภายนอกอาคาร คณะสหเวชศาสตร์ 16.745915, 100.189384	ดัง	มีข้อความเข้า	16.745920, 100.189380	0.69 เมตร
24	ภายนอกอาคาร คณะสหเวชศาสตร์ 16.745915, 100.189384	ดัง	มีข้อความเข้า	16.745910, 100.189389	0.76 เมตร

ครั้งที่	บริเวณที่ส่งข้อความ	เสียง Alarm	ผู้รับ	พิกัดที่ผู้รับได้รับ	ความคลาดเคลื่อน
25	ภายนอกอาคาร คณะสหเวชศาสตร์ 16.745915, 100.189384	ดัง	มีข้อความเข้า	16.745917, 100.189392	0.88 เมตร
26	ภายในอาคารกีฬาในร่ม 16.747035, 100.197124	ดัง	มีข้อความเข้า	16.747022, 100.197097	3.21 เมตร
27	ภายในอาคารกีฬาในร่ม 16.747035, 100.197124	ดัง	มีข้อความเข้า	16.747030, 100.197119	0.76 เมตร
28	ภายในอาคารกีฬาในร่ม 16.747035, 100.197124	ดัง	มีข้อความเข้า	16.747031, 100.197130	0.77 เมตร
29	ภายในอาคารกีฬาในร่ม 16.747035, 100.197124	ดัง	มีข้อความเข้า	16.747040, 100.197130	0.84 เมตร
30	ภายในอาคารกีฬาในร่ม 16.747035, 100.197124	ดัง	มีข้อความเข้า	16.747037, 100.197128	0.48 เมตร

จากตารางที่ 4.1 เมื่อทำการทดสอบโปรแกรมประยุกต์โดยขอความช่วยจากบริเวณภายในมหาวิทยาลัยนเรศวรทั้งหมด 30 ครั้ง เมื่อทำการกดปุ่มเพื่อขอความช่วยเหลือฉุกเฉินจะมีเสียงสัญญาณเตือนดังขึ้นและฝั่งผู้รับจะได้รับข้อความทุกครั้งที่ทำการทดสอบ จากการทดสอบส่งข้อความภายในอาคารวิศวกรรมไฟฟ้า 5 ครั้งจะสังเกตว่าพิกัดที่ได้จะเป็นเลขเดิมซ้ำกันถึง 5 ครั้ง เกิดจากภายในอาคารวิศวกรรมไฟฟ้าไม่มีสัญญาณระบุตำแหน่ง โปรแกรมประยุกต์จึงจะใช้ตำแหน่งสุดท้ายที่บันทึกได้ในการส่งจึงเป็นสาเหตุให้ผู้รับได้พิกัดซ้ำกันทุกครั้งที่ทดสอบการส่งภายในอาคาร เช่นเดียวกันกับเมื่อขอความช่วยเหลือภายในอาคารคณะสหเวชศาสตร์ 5 ครั้งจะได้พิกัดซ้ำกันทุกครั้ง พิกัดที่ผู้รับได้รับเมื่อเปรียบเทียบกับพิกัดจริงที่หาได้จาก Google map ภายในอาคารวิศวกรรมไฟฟ้าจะมีระยะห่างกันประมาณ 19.2 เมตร และพิกัดที่ผู้รับได้รับเมื่อเปรียบเทียบกับพิกัดจริงที่หาได้จาก Google map ภายในอาคารคณะสหเวชศาสตร์จะมีระยะห่างกันประมาณ 18.68 เมตร ส่วนเมื่อขอความช่วยเหลือบริเวณภายนอกอาคารพิกัดที่ผู้รับได้รับจะแตกต่างกันออกไปตามลักษณะการส่ง ระยะห่างระหว่างพิกัดที่ขอ

ความช่วยเหลือที่ได้จากโปรแกรมประยุกต์แสดงเลขตำแหน่งพิกัดกับพิกัดที่ผู้รับได้รับโดยเฉลี่ยเมื่อทำการทดสอบจะได้ผลดังนี้ ภายนอกอาคารวิศวกรรมไฟฟ้าทดสอบจำนวน 5 ครั้งระยะห่างระหว่างพิกัดที่ขอความช่วยเหลือที่ได้จากโปรแกรมประยุกต์แสดงเลขตำแหน่งพิกัดกับพิกัดที่ผู้รับได้รับประมาณ 0.421 เมตรและภายนอกอาคารคณะสหเวชศาสตร์ทดสอบจำนวน 5 ครั้งระยะห่างระหว่างพิกัดที่ขอความช่วยเหลือที่ได้จากโปรแกรมประยุกต์แสดงเลขตำแหน่งพิกัดกับพิกัดที่ผู้รับได้รับประมาณ 0.59 เมตร ส่วนเมื่อขอความช่วยเหลือจากบริเวณโรงอาหารคณะเกษตรศาสตร์และภายอาคารกีฬาในร่ม ผู้รับจะได้รับพิกัดที่ต่างกัน เนื่องจากถึงแม้ว่าจะอยู่ในอาคารแต่ก็ยังมีสัญญาณระบุตำแหน่งอยู่เพราะรูปแบบอาคารมีลักษณะเปิดโล่งไม่เหมือนกับภายในอาคารวิศวกรรมไฟฟ้าและอาคารคณะสหเวชศาสตร์ซึ่งภายในมีลักษณะปิดทึบ ระยะห่างระหว่างพิกัดที่ขอความช่วยเหลือที่ได้จากโปรแกรมประยุกต์แสดงเลขตำแหน่งพิกัดกับพิกัดที่ผู้รับได้รับ โดยเฉลี่ยเมื่อทำการทดสอบจะได้ดังนี้ ทดสอบจำนวน 5 ครั้งบริเวณโรงอาหารคณะเกษตรศาสตร์ระยะห่างจากจุดที่ขอความช่วยเหลือประมาณ 0.838 เมตรและทดสอบจำนวน 5 ครั้งภายในอาคารกีฬาในร่มระยะห่างจากจุดที่ขอความช่วยเหลือประมาณ 1.212 เมตร

ตารางที่ 4.2 ตารางทดสอบการส่งข้อความขอความช่วยเหลือโดยการพลิก – หงาย บริเวณภายในมหาวิทยาลัยนเรศวร

ครั้งที่	บริเวณที่ส่งข้อความ	เสียง Alarm	ผู้รับ	พิกัดที่ผู้รับได้รับ	ความคลาดเคลื่อน
1	ภายในอาคารวิศวกรรมไฟฟ้า 16.744650, 100.196754	ไม่ดัง	ไม่มีข้อความเข้า	-	-
2	ภายในอาคารวิศวกรรมไฟฟ้า 16.744650, 100.196754	ไม่ดัง	ไม่มีข้อความเข้า	-	-
3	ภายในอาคารวิศวกรรมไฟฟ้า 16.744650, 100.196754	ดัง	มีข้อความเข้า	16.7446053, 100.19658	19.2 เมตร

ครั้งที่	บริเวณที่ส่งข้อความ	เสียง Alarm	ผู้รับ	พิกัดที่ผู้รับได้รับ	ความคลาดเคลื่อน
4	ภายในอาคารวิศวกรรมไฟฟ้า 16.744650, 100.196754	ดัง	มีข้อความเข้า	16.7446053, 100.19658	19.2 เมตร
5	ภายในอาคารวิศวกรรมไฟฟ้า 16.744650, 100.196754	ดัง	มีข้อความเข้า	16.7446053, 100.19658	19.2 เมตร
6	ภายนอกอาคารวิศวกรรมไฟฟ้า 16.744677, 100.196559	ไม่ดัง	ไม่มีข้อความเข้า	-	-
7	ภายนอกอาคารวิศวกรรมไฟฟ้า 16.744677, 100.196559	ไม่ดัง	ไม่มีข้อความเข้า	-	-
8	ภายนอกอาคารวิศวกรรมไฟฟ้า 16.744677, 100.196559	ดัง	มีข้อความเข้า	16.744670, 100.196553	1.10 เมตร
9	ภายนอกอาคารวิศวกรรมไฟฟ้า 16.744677, 100.196559	ดัง	มีข้อความเข้า	16.744674, 100.196550	1.02 เมตร
10	ภายนอกอาคารวิศวกรรมไฟฟ้า 16.744677, 100.196559	ดัง	มีข้อความเข้า	16.744669, 100.196560	0.89 เมตร
11	บริเวณโรงอาหารคณะเกษตรศาสตร์ 16.746029, 100.195401	ดัง	มีข้อความเข้า	16.746021, 100.195409	1.22
12	บริเวณโรงอาหารคณะเกษตรศาสตร์ 16.746029, 100.195401	ดัง	มีข้อความเข้า	16.746022, 100.195407	1.01

ครั้งที่	บริเวณที่ส่งข้อความ	เสียง Alarm	ผู้รับ	พิกัดที่ผู้รับได้รับ	ความคลาดเคลื่อน
13	บริเวณโรงอาหารคณะเกษตรศาสตร์ 16.746029, 100.195401	ดัง	มีข้อความเข้า	16.746019, 100.19540	1.11
14	บริเวณโรงอาหารคณะเกษตรศาสตร์ 16.746029, 100.195401	ดัง	มีข้อความเข้า	16.746035, 100.195409	1.08
15	บริเวณโรงอาหารคณะเกษตรศาสตร์ 16.746029, 100.195401	ดัง	มีข้อความเข้า	16.746030, 100.195411	1.07
16	ภายในอาคารคณะสหเวชศาสตร์ 16.745780, 100.189201	ดัง	มีข้อความเข้า	16.745773, 100.189026	18.68 เมตร
17	ภายในอาคารคณะสหเวชศาสตร์ 16.745780, 100.189201	ดัง	มีข้อความเข้า	16.745773, 100.189026	18.68 เมตร
18	ภายในอาคารคณะสหเวชศาสตร์ 16.745780, 100.189201	ดัง	มีข้อความเข้า	16.745773, 100.189026	18.68 เมตร
19	ภายในอาคารคณะสหเวชศาสตร์ 16.745780, 100.189201	ดัง	มีข้อความเข้า	16.745773, 100.189026	18.68 เมตร
20	ภายในอาคารคณะสหเวชศาสตร์ 16.745780, 100.189201	ดัง	มีข้อความเข้า	16.745773, 100.189026	18.68 เมตร
21	ภายนอกอาคารคณะสหเวชศาสตร์ 16.745915, 100.189384	ดัง	มีข้อความเข้า	16.745920, 100.189376	1.01 เมตร

ครั้งที่	บริเวณที่ส่งข้อความ	เสียง Alarm	ผู้รับ	พิกัดที่ผู้รับได้รับ	ความคลาดเคลื่อน
22	ภายนอกอาคาร คณะสหเวชศาสตร์ 16.745915, 100.189384	ดัง	มีข้อความเข้า	16.745921, 100.189392	1.08 เมตร
23	ภายนอกอาคาร คณะสหเวชศาสตร์ 16.745915, 100.189384	ดัง	มีข้อความเข้า	16.745924, 100.189377	1.24 เมตร
24	ภายนอกอาคาร คณะสหเวชศาสตร์ 16.745915, 100.189384	ดัง	มีข้อความเข้า	16.745921, 100.189380	0.78 เมตร
25	ภายนอกอาคาร คณะสหเวชศาสตร์ 16.745915, 100.189384	ดัง	มีข้อความเข้า	16.745923, 100.189383	0.89 เมตร
26	ภายในอาคารกีฬาในร่ม 16.747035, 100.197124	ดัง	มีข้อความเข้า	16.747037, 100.197134	1.08 เมตร
27	ภายในอาคารกีฬาในร่ม 16.747035, 100.197124	ดัง	มีข้อความเข้า	16.747029, 100.197125	0.67 เมตร
28	ภายในอาคารกีฬาในร่ม 16.747035, 100.197124	ดัง	มีข้อความเข้า	16.747045, 100.197128	1.18 เมตร
29	ภายในอาคารกีฬาในร่ม 16.747035, 100.197124	ดัง	มีข้อความเข้า	16.747042, 100.197133	1.23 เมตร
30	ภายในอาคารกีฬาในร่ม 16.747035, 100.197124	ดัง	มีข้อความเข้า	16.747047, 100.197123	1.33 เมตร

จากตารางที่ 4.2 เมื่อทำการทดสอบโปรแกรมประยุกต์โดยขอความช่วยจากบริเวณ ภายในมหาวิทยาลัยนเรศวรทั้งหมด 30 ครั้งโดยการพลิก – หงายโทรศัพท์มีความผิดพลาดเกิดขึ้น 4 ครั้ง โดยความผิดพลาดที่เกิดขึ้นคือโทรยัพท์ของผู้ขอความช่วยเหลือไม่มีเสียงดังและฝั่งผู้รับไม่มีข้อความขอ

ความช่วยเหลือเข้า อาจจะเกิดจากความไม่ชินในการใช้โปรแกรมประยุกต์ สังเกตได้จากจะเกิดในครั้งแรก ๆ ของการใช้งานโปรแกรมประยุกต์เมื่อใช้งานจนชำนาญหรือทราบเทคนิคแล้วจะสามารถลดความผิดพลาดให้เกิดขึ้นน้อยลงหรือไม่เกิดขึ้นเลย ดังตารางจะเห็นว่าความผิดพลาดที่เกิดขึ้นจะเกิดขึ้นในครั้งที่ 1 และ 2 ซึ่งเป็นครั้งแรกในการทดสอบและครั้งที่ 6 และ 7 เป็นการย้ายสถานที่ในการทดสอบแต่ในครั้งถัด ๆ มาจนถึงครั้งที่ 30 ไม่มีความผิดพลาดเกิดขึ้นเลย ส่วนพิกัดที่ผู้รับได้รับนั้นจะมีลักษณะเหมือนกันกับเมื่อกดปุ่มส่งขอความช่วยเหลือ

ตารางที่ 4.3 ตารางทดสอบการใช้งานโปรแกรมประยุกต์โดยการดำเนินชีวิตประจำวันเป็นเวลา 1 สัปดาห์

วันที่	รายการ	ข้อความ	เสียง Alarm	หมายเหตุ
1	ทำกิจวัตรในช่วงเวลากลางวัน (8 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
	ทำกิจวัตรในช่วงเวลากลางคืน (4 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
2	ทำกิจวัตรในช่วงเวลากลางวัน (8 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
	ทำกิจวัตรในช่วงเวลากลางคืน (4 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
3	ทำกิจวัตรในช่วงเวลากลางวัน (8 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
	ทำกิจวัตรในช่วงเวลากลางคืน (4 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
4	ทำกิจวัตรในช่วงเวลากลางวัน (8 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
	ทำกิจวัตรในช่วงเวลากลางคืน (4 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
5	ทำกิจวัตรในช่วงเวลากลางวัน (8 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
	ทำกิจวัตรในช่วงเวลากลางคืน (4 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
6	ทำกิจวัตรในช่วงเวลากลางวัน (8 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
	ทำกิจวัตรในช่วงเวลากลางคืน (4 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
7	ทำกิจวัตรในช่วงเวลากลางวัน (8 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	
	ทำกิจวัตรในช่วงเวลากลางคืน (4 ชั่วโมง)	ไม่ถูกส่ง	ไม่มีเสียง	

จากตารางที่ 4.3 ทดสอบโปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉินโดยการทำกิจวัตรประจำวันตามปกติในช่วงเวลากลางวันตั้งแต่เวลา 8.30 น. ถึง 16.30 น. และในช่วงเวลากลางคืนตั้งแต่เวลา 16.30 น. ถึง 20.30 น. เป็นเวลา 1 สัปดาห์พบว่าไม่มีความผิดพลาดของโปรแกรมประยุกต์เกิดขึ้น

ในระหว่างช่วงที่ดำเนินกิจวัตรประจำวันในเวลา 1 สัปดาห์เลย โดยการทดสอบจะวัดความผิดพลาดจากการส่งข้อความโดยไม่ได้ตั้งใจ ซึ่งในขณะที่ทำการทดสอบไม่มีข้อความถูกส่งออกไปยังหมายเลขผู้รับและไม่มีเสียงเตือนดังขึ้นในขณะดำเนินกิจวัตรประจำวัน

ตารางที่ 4.4 ตารางการทำกิจวัตรปกติ 8 ชั่วโมงทดสอบโดยการใช้ชีวิตประจำวันในช่วงเวลากลางวัน

เวลา	รายการ	หมายเหตุ
8.30 – 9.00	เริ่มเปิดโปรแกรมประยุกต์และเดินทางจากมหาวิทยาลัยเรศวรไปยังสนามกีฬากลางจังหวัดพิษณุโลก	เดินทางโดยรถจักรยานยนต์
9.00 – 11.50	นั่งทำงานเอกสารที่โต๊ะทำงาน ณ สนามกีฬากลางจังหวัดพิษณุโลก	
11.50 – 12.30	ช่วงพักกลางวัน เดินไปทานข้าว ณ โรงอาหารสนามกีฬากลางจังหวัดพิษณุโลก	
12.30 – 13.00	นั่งพักผ่อนที่โต๊ะทำงาน ณ สนามกีฬากลางจังหวัดพิษณุโลก	อยู่ในช่วงเวลาพักกลางวัน
13.00 – 14.00	นั่งทำงานเอกสารที่โต๊ะทำงาน ณ สนามกีฬากลางจังหวัดพิษณุโลก	
14.00 – 16.00	ออกพื้นที่สำรวจอุปกรณ์กีฬา บริเวณสนามกีฬากลางจังหวัดพิษณุโลก	
16.00 – 16.45	เดินทางกลับมหาวิทยาลัยนเรศวรจังหวัดพิษณุโลก	

ตารางที่ 4.5 ตารางการทำกิจวัตรปกติ 4 ชั่วโมงทดสอบโดยการใช้ชีวิตประจำวันในช่วงเวลากลางคืน

เวลา	รายการ	หมายเหตุ
16..00 – 16.45	เดินทางจากสนามกีฬากลางจังหวัดพิษณุโลกไปยังมหาวิทยาลัยนเรศวร	เดินทางโดยรถจักรยานยนต์
16.45 – 17.20	แวะทานอาหาร ณ ร้านอาหารโป้นก	
17.20 – 17.30	เดินทางกลับที่พัก	
18.15 – 18.45	เปลี่ยนชุดเพื่อไปออกกำลังกาย ณ สนามกีฬาในร่มมหาวิทยาลัยนเรศวร	
18.45 – 20.00	ออกกำลังกาย ณ สนามกีฬาในร่มมหาวิทยาลัยนเรศวร	
20.00 – 20.30	เดินทางกลับหอพักและปิดโปรแกรมประยุกต์	

จากตาราง 4.4 และตารางที่ 4.5 แสดงกิจวัตรประจำวันที่ทำการทดสอบในแต่ละวันเป็นเวลา 1 สัปดาห์ ซึ่งบางวันอาจจะมีภารกิจมากกว่าที่ระบุไว้ในตาราง แต่ภารกิจหลัก ๆ ในแต่ละวันจะระบุไว้ในตารางที่ 4.4 และตารางที่ 4.5 ซึ่งจะเป็นข้อมูลของตารางที่ 4.3

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการทดสอบโปรแกรมประยุกต์ขอความช่วยเหลือฉุกเฉินพร้อมระบุตำแหน่งบนระบบปฏิบัติการแอนดรอยด์ ผลการทดสอบประสิทธิภาพของโปรแกรมประยุกต์จะแบ่งการทดสอบโดยรูปแบบการส่งข้อความและบริเวณที่ส่งข้อความดังนี้

จากการทดสอบการส่งข้อความในหลายรูปแบบพบว่า วิธีการส่งโดยการกดปุ่มส่งข้อความเป็นวิธีที่มีประสิทธิภาพในการส่งข้อความมากที่สุด โดยการทดสอบทั้ง 30 ครั้ง ข้อความขอความช่วยเหลือจะถูกส่งไปยังผู้รับทุกครั้ง แต่วิธีการส่งข้อความโดยการกดปุ่มส่งข้อความจะมีช่วงเวลาในการเข้าสู่โปรแกรมประยุกต์ก่อนที่จะกดส่งข้อความ ส่วนวิธีการส่งข้อความโดยการพลิก – หงาย จะเกิดความผิดพลาดขึ้นในครั้งแรก ๆ ของการส่งข้อความ ข้อความจะไปถึงผู้รับทั้งหมด 26 ครั้งจากการทดสอบ 30 ครั้ง เนื่องจากต้องมีความคุ้นเคยในการพลิก – หงายโทรศัพท์ เพื่อหามุมพลิก – หงายที่ผู้ใช้ถนัดที่สุด เมื่อผู้ใช้มีความคุ้นเคยจึงจะทำให้ไม่เกิดความผิดพลาดในการพลิก – หงายโทรศัพท์ วิธีการส่งข้อความโดยการพลิก – หงายเป็นวิธีการส่งข้อความที่รวดเร็ว เนื่องจากไม่ต้องเข้าสู่โปรแกรมประยุกต์ เมื่อต้องการความช่วยเหลือสามารถหยิบโทรศัพท์ขึ้นมาเพื่อขอความช่วยเหลือได้ในทันที

ส่วนพิกัดที่ผู้รับได้รับกับจุดที่ขอความช่วยเหลือจากการทดสอบ 60 ครั้งจะมีระยะห่างเฉลี่ย 6.691 เมตร ประกอบไปด้วย เมื่อทดสอบภายในอาคารวิศวกรรมไฟฟ้า 10 ครั้งระยะห่างเฉลี่ย 19.2 เมตร ทดสอบภายนอกอาคารวิศวกรรมไฟฟ้า 10 ครั้งระยะห่างเฉลี่ย 0.639 เมตร ทดสอบบริเวณโรงอาหารคณะเกษตรศาสตร์ 10 ครั้งระยะห่างเฉลี่ย 0.968 เมตร ทดสอบภายในอาคารคณะสหเวชศาสตร์ 10 ครั้งระยะห่างเฉลี่ย 18.68 เมตร ทดสอบภายนอกอาคารคณะสหเวชศาสตร์ 10 ครั้งระยะห่างเฉลี่ย 0.795 เมตร ทดสอบภายในอาคารกีฬาในร่ม 10 ครั้งระยะห่างเฉลี่ย 1.155 เมตร จะเห็นว่าระยะห่างจากจุดที่ขอความช่วยเหลือกับพิกัดที่ผู้รับได้รับจะไม่เกิน 20 เมตร ซึ่งระยะห่างจะถูกทดแทนด้วยเสียงไซเรนทำให้ผู้ที่อยู่บริเวณรอบได้ยินและเข้ามาช่วยเหลือได้

จึงสรุปได้ว่าโปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉินจะมีประสิทธิภาพเมื่อกดปุ่มส่งขอความช่วยเหลือและเมื่อใช้งานจนเกิดความชำนาญการพลิก – หงายโทรศัพท์จะเป็นวิธีที่สามารถขอความช่วยเหลือได้ไวขึ้น พิกัดที่ผู้รับกับจุดที่ขอความช่วยเหลือจะมีความคลาดเคลื่อนกันเล็กน้อย เมื่อส่งขอความช่วยเหลือจากบริเวณภายในอาคาร โปรแกรมประยุกต์จะบันทึกพิกัดสุดท้ายก่อนที่จะเข้าไปภายในอาคาร หากส่งขอความช่วยเหลือจากภายนอกอาคารผู้รับจะได้พิกัดที่แตกต่างกันออกไปในการส่งแต่ละครั้ง ณ จุดส่งจุดเดิม แต่ระยะห่างจะถูกทดแทนเสียงไซเรนเพื่อให้ผู้ที่อยู่ใกล้เคียงหรือผู้ที่เข้ามาช่วยเหลือทราบว่าผู้ส่งขอความช่วยเหลืออยู่ที่ใด

5.2 ปัญหาและอุปสรรคที่พบในการดำเนินงาน

5.2.1 เนื่องจากเมื่อทดลองใช้งานแล้วพบว่า มีปัญหาในเรื่องการส่งข้อความแบบพลิก – หงาย แบบดำเนินชีวิตประจำวันแล้วเกิดการส่งข้อความออกไปบ่อยครั้ง จึงจำเป็นต้องศึกษาและทดลองหาวิธีการส่งโดยใช้แกนของ Accumulator sensor ทั้ง 3 แกน เพื่อลดความผิดพลาดในการส่งให้น้อยที่สุด

5.2.2 เนื่องจากในห้องเรียนไม่มีการเรียนการสอนเกี่ยวกับระบบปฏิบัติการแอนดรอยด์ ซึ่งเป็นแกนหลักสำคัญในการทำโครงงานนี้ จึงทำให้ต้องศึกษาด้วยตัวเองใหม่ทั้งหมดค่อย ๆ รวบรวมแต่ละส่วนเข้าด้วยกันจนมาเป็นโครงงานชิ้นนี้ ทำให้ต้องเสียเวลาในการศึกษาไปมากพอสมควร

5.2.3 เนื่องจากการขอความช่วยเหลือต้องคำนึงถึงความง่ายในการใช้งาน แต่ก็ไม่ควรง่ายเกินไปในการส่งข้อความ จึงต้องใช้เวลาแก้ปัญหาในส่วนนี้ให้สามารถใช้งานได้ดีและเหมาะกับผู้ใช้งานที่สุด โดยการใช้ Accumulator sensor ส่งข้อความโดยการพลิก – หงายหรือส่งข้อความจากภายในโปรแกรมประยุกต์ เพื่อเป็นทางเลือกให้ผู้ใช้งานสามารถใช้งานได้ตามที่ตนเองสะดวกที่สุด

5.2.4 เนื่องจากในฝั่งผู้รับต้องมีการใช้แผนที่ (Google Map) ผู้รับจึงจำเป็นต้องมีอินเตอร์เน็ตเพื่อดาวน์โหลดแผนที่ (Google Map) ในการเปิดแผนที่ในครั้งแรก

5.2.5 โปรแกรมประยุกต์นี้จำเป็นต้องมีการเปิดระบบระบุตำแหน่งบนพื้นโลกตลอดเวลา เนื่องจากต้องการตำแหน่งของผู้ใช้งานที่ต้องการขอความช่วยเหลือที่เป็นปัจจุบันที่สุด การปิดระบบระบุตำแหน่งจะทำให้โปรแกรมประยุกต์ไม่สามารถบันทึกตำแหน่งปัจจุบันของผู้ใช้งานที่ต้องการความช่วยเหลือได้ แต่จะใช้งานตำแหน่งสุดท้ายที่ระบบระบุตำแหน่งบนพื้นโลก (GPS) บันทึกได้แทน

5.3 ข้อเสนอแนะและแนวทางในการพัฒนา

จากโปรแกรมประยุกต์สำหรับขอความช่วยเหลือฉุกเฉินนั้นเมื่อมีการขอความช่วยเหลือฉุกเฉินแล้ว ผู้ขอความช่วยเหลือจะไม่สามารถรู้ได้ว่าปลายทางจะได้รับข้อความขอความช่วยเหลือแล้วหรือยัง หากมีการพัฒนาให้โปรแกรมประยุกต์สามารถบอกได้ว่าผู้รับได้รับข้อความขอความช่วยเหลือแล้วจะทำให้โปรแกรมประยุกต์มีความสมบูรณ์ยิ่งขึ้น ในส่วนภายในโปรแกรมประยุกต์สามารถเพิ่มเติมการใช้งานให้ง่ายยิ่งขึ้นได้โดยการนำระบบการขยายหน้าจอเข้าและออกมาใช้งานหรือการขยายหน้าจอตามรูปบบเครื่องโทรศัพท์มือถือแต่ละเครื่อง โปรแกรมประยุกตสำหรับขอความช่วยเหลือฉุกเฉินนั้นยังสามารถพัฒนาฟังก์ชั่นอื่นๆเพิ่มเติมได้อีกในอนาคต เช่น ระบบนำทาง (Tracking) โดยแผนที่จะแสดงเส้นทางที่สามารถเดินทางไปยังผู้ต้องการความช่วยเหลือที่ใกล้ที่สุด เพื่อจะสามารถเข้าช่วยเหลือเป้าหมายได้อย่างรวดเร็ว การตรวจการกระแทกของรถยนต์จากเซนเซอร์เพื่อให้ตรวจสอบว่ารถยนต์คันดังกล่าวเกิดอุบัติเหตุการชนกันหนักเบาเพียงใด และสามารถเชื่อมต่อกับสังคมออนไลน์อื่น ๆ เช่น เฟสบุ๊ค (Facebook), ทวิตเตอร์ (Twitter) โดยการโพสต์ข้อความลงบนหน้าของตนเอง เพื่อเป็นแนวทางในการขอความช่วยเหลือได้อีกทางหนึ่ง

เอกสารอ้างอิง

[1] กฤษฎา ประเสริฐสุโข. (2548). ภาวะฉุกเฉิน. สืบค้นเมื่อ 6 เมษายน 2557, จาก http://www.diwsafety.org/download/book/รองรับภาวะฉุกเฉิน%20REV%201.pdf

[2] กฤษฎ์ โพธิ์ศรี, อุดมศักดิ์ มหาวีรวัฒน์ และจุฬาภรณ์ โสตะ. (2554). การพัฒนาศักยภาพอาสาสมัครกู้ชีพในการบริการการแพทย์ฉุกเฉินของหน่วยกู้ชีพขั้นพื้นฐาน อำเภอเมือง จังหวัดมหาสารคาม. สืบค้นเมื่อ 6 เมษายน 2557, จาก http://thailand.digitaljournals.org/index.php/RDHSJ/article/viewFile/13620/13065

[3] ดร.สงครามชัย ลีทองดี. (2555). รายงานฉบับสมบูรณ์การประเมินแผนหลักการแพทย์ฉุกเฉินแห่งชาติ ปี 2553 - 2555. สืบค้นเมื่อ 6 เมษายน 2557, จาก http://goo.gl/AOuXjO

[4] ธัชพงศ์ งามอุโฆษ, ชาญ ศรีรัตนสถาวร, บัญชา ศันสนีย์วิทยกุล, ปิยะ เกษมสุวรรณ, กิตติพร อังคสุวพลและสุกิจ แย้มวงษ์. (2550). Cardiac arrhythmias in Thai Acute Coronary Syndrome Registry. สืบค้นเมื่อ 6 เมษายน 2557, จาก http://www.thaiheart.org/images/sub_1291456609/Cardiac%20Arrhythmias.pdf

[5] กมล บุยศ์เพชร, จารุวิชญ์ ศิริรัตนชัยกูล และณัฐ ไวทย์ เนียมรัตน์. (2555). เครื่องแจ้งเตือนผู้สูงอายุเกิดอุบัติเหตุ. วิทยานิพนธ์ วศ.บ., มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี, กรุงเทพฯ.

[6] นายรัฐภูมิ ตู้จินดา. (2555). ระบบแผนที่แจ้งเตือนเหตุฉุกเฉินส่วนตัว. สืบค้นเมื่อ 6 เมษายน 2557, จาก http://www.nectec.or.th/index.php/2011-07-12-08-07-51/2257-2011-07-04-09-11-21.html

[7] กรกฎ อภิรัตน์วรากุล. (2556). The EMS System in Thailand and ASEAN. สืบค้นเมื่อ 5 พฤษภาคม 2557, จาก http://www.md.kku.ac.th/library/main/eproceeding/Lec_64_68.pdf

[8] มหาวิทยาลัยราชภัฏสวนสุนันทา. (2548). การตกเลือดและการห้ามเลือด. สืบค้นเมื่อ 9 พฤษภาคม 2557, จาก http://science.srru.ac.th/org/sci-elearning/courseonline/4072303/pdf_8.pdf

[9] Gheeraert PJ, De Buyzere ML, Taeymans YM, Gillebert TC, Henriques JPS, De Backer G, et al. (2006). Risk factors for primary ventricular fibrillation during acute myocardial infarction: a systematic review and meta-analysis. European Heart Journal. 27(21), 510 - 2499

[10] ประดิษฐ์ ปัญจวีณิน. (2554). Cardiac emergencies. (พิมพ์ครั้งที่ 3). กรุงเทพฯ: ห้างหุ้นส่วนจำกัดภาพพิมพ์.

[11] วีระวัฒน์ เหลืองชนะ. (2548). ตารางพื้นฐานทางศัลยศาสตร์. วารสารอุบัติเหตุ. 24(3), 48 - 137.

[12] พร้อมเลิศ หล่อวิจิตร. (2555). คู่มือเขียนแอพ Android ฉบับสมบูรณ์. สืบค้นเมื่อ 3 พฤษภาคม 2557, จาก http://promlert.com/wp-content/uploads/2012/07/AndroidDev_ch01.pdf

[13] บริษัท โกลบอลไฟว์ จำกัด. (2556). ความรู้เรื่อง GPS. สืบค้นเมื่อ 15 เมษายน 2557, จาก http://www.global5thailand.com/thai/gps.html

[14] สถานวิจัยสารสนเทศภูมิศาสตร์ทรัพยากรธรรมชาติและสิ่งแวดล้อม ศูนย์ภูมิภาคเทคโนโลยีอวกาศและสารสนเทศ (ภาคใต้) คณะการจัดการสิ่งแวดล้อม มหาวิทยาลัยสงขลานครินทร์. (2556). GPS. สืบค้นเมื่อ 23 เมษายน 2557, จาก http://www.rmutphysics.com/charud/oldnews/201/sattlelite/GPS_suphalerk.pdf

[15] สมพงษ์ ดงไม้. (2555). GPS (Global Positioning System). สืบค้นเมื่อ 6 เมษายน 2557, จาก http://www.champsat.com/gps-s.html

[16] อดิศร ขาวสังข์. (2554). โปรแกรม GPS บน Android Smart Phone. สืบค้นเมื่อ 30 เมษายน 2557, จาก http://www.itmanage.info/technology/SmartPhone/gps_on_android/GPS_On_Android.pdf

ประวัติผู้ดำเนินโครงงาน

ชื่อ นายอัฐพล แตงชัยภูมิ

ภูมิลำเนา 92 หมู่ 8 ต.คลองกระจง อ.สวรรคโลก จ.สุโขทัย 64110

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนศรีสำโรงชนูปถัมภ์ จ.สุโขทัย
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 7 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : harumichikung@gmail.com

ชื่อ นายกำภู สุจริตจันทร์

ภูมิลำเนา 1/9 ซ.สองแคว 1 ถ.สายเอเชีย ต.แม่สอด อ.แม่สอด จ.ตาก 63110

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนสรรพวิทยาคม จ.ตาก
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 7 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : pond_na_krub@hotmail.com