ระบบติดตามรถไฟสายตะวันออกอัจฉริยะ
INTELLIGENT TRAIN TRACKING SYSTEM FOR EASTERN RAIL ROUTE

โดย

นางสาวอภิชญา สิริวณิชพงศ์

นายอภิเชษฐ์ พรมเกตุ

นายอารีฟ ดาแตสาตู

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

ระบบติดตามรถไฟสายตะวันออกอัจฉริยะ
INTELLIGENT TRAIN TRACKING SYSTEM FOR EASTERN RAIL ROUTE

โดย

นางสาวอภิชญา สิริวณิชพงศ์

นายอภิเชษฐ์ พรมเกตุ

นายอารีฟ ดาแตสาตู

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

ระบบติดตามรถไฟสายตะวันออกอัจฉริยะ

INTELLIGENT TRAIN TRACKING SYSTEM FOR EASTERN RAIL ROUTE

โดย

นางสาวอภิชญา	สิริวณิชพงศ์	56011406
นายอภิเชษฐ์	พรมเกตุ	56011408
นายอารีฟ	ดาแตสาตู	56011461

อาจารย์ที่ปรึกษา

ศ.ดร.พรชัย ทรัพย์นิธิ

ผศ.ดร.ณัฐกานต์ พุทธรักษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

ผ่านการตรวจรูปเล่มแล้ว

(.....)

อาจารย์ที่ปรึกษา

...../...../.....

วิศวกรรมโทรคมนาคม Telecommunications Engineering

ผ่านการตรวจชิ้นงานแล้ว

(.....)

กรรมการผู้ตรวจชิ้นงาน

...../...../.....

วิศวกรรมโทรคมนาคม Telecommunications Engineering

ปริญญานิพนธ์ปีการศึกษา 2559
ภาควิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง ระบบติดตามรถไฟสายตะวันออกอัจฉริยะ
INTELLIGENT TRAIN TRACKING SYSTEM FOR EASTERN RAIL ROUTE

ผู้จัดทำ
1. นางสาวอภิชญา สิริวณิชพงศ์ 56011406
2. นายอภิเชษฐ์ พรมเกตุ 56011408
3. นายอารีฟ ดาแตสาตู 56011461

.....
(ศ.ดร.พรชัย ทรัพย์นิธิ) **อาจารย์ที่ปรึกษา**

.....
(ผศ.ดร.ณัฐกานต์ พุทธรักษ์) **อาจารย์ที่ปรึกษาร่วม**

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จลุล่วงไปด้วยดี ด้วยความช่วยเหลือของ ศ.ดร.พรชัย ทรัพย์นิธิ ผศ.ดร.ณัฐกานต์ พุทธรักษ์ และดร.เวธิต ภาคย์พิสุทธิ์ เป็นอาจารย์ที่ปรึกษา ซึ่งท่านได้ให้คำแนะนำและข้อคิดเห็นต่างๆ อันเป็นประโยชน์อย่างยิ่ง อีกทั้งยังช่วยแก้ปัญหาต่างๆ ที่เกิดขึ้นระหว่างการดำเนินงานอีกด้วย และขอขอบคุณนักศึกษาปริญญาเอกและโท คุณศิรวิชญ์ กิตติวิชญกุล คุณสุทัศน์ จงสินทวี และคุณจิรภูมิ บุตรโท และรุ่นพี่สำหรับข้อแนะนำและความช่วยเหลือในทุกๆด้าน และขอขอบคุณการรถไฟแห่งประเทศไทยร่วมถึงพนักงานรักษารถไฟในการสนับสนุนในการทำปริญญานิพนธ์

สุดท้ายนี้ ผู้วิจัยขอขอบพระคุณบิดามารดา และครอบครัว ซึ่งเปิดโอกาสให้ได้รับการศึกษาเล่าเรียน ตลอดจนคอยช่วยเหลือและให้กำลังใจเสมอมา

นางสาวอภิชญา สิริวณิชพงศ์
นายอภิเชษฐ์ พรมเกตุ
นายอารีฟ ดาแตสาตู
ผู้จัดทำ

ระบบติดตามรถไฟสายตะวันออกอัจฉริยะ
INTELLIGENT TRAIN TRACKING SYSTEM FOR EASTERN RAIL ROUTE

โดย นางสาวอภิชญา สิริวณิชพงศ์ 56011406
นายอภิเชษฐ์ พรมเกตุ 56011408
นายอารีฟ ดาแตสาตู 56011461

อาจารย์ที่ปรึกษา ศ.ดร.พรชัย ทรัพย์นิธิ
ผศ.ดร.ณัฐกานต์ พุทธรักษ์

บทคัดย่อ

ในปัจจุบันมีประชาชนจำนวนมากนิยมเดินทางด้วยรถไฟโดยเฉพาะรถไฟในสายตะวันออก ซึ่งพบว่ามีนักเรียน นักศึกษาและคนทำงานใช้งานกันอยู่เป็นประจำ อย่างไรก็ตามบ่อยครั้งขบวนรถไฟมักจะไม่ตรงตามเวลาตารางรถไฟ ส่งผลให้เสียเวลาและพลาดในการใช้บริการขบวนรถไฟที่ต้องการเดินทาง ดังนั้น งานวิจัยนี้จึงได้มีการออกแบบและพัฒนาระบบติดตามขบวนรถไฟสายตะวันออก ซึ่งผู้วิจัยได้ออกแบบและพัฒนา เว็บไซต์ แอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ และจอมอนิเตอร์ เพื่อแสดงตำแหน่งรถไฟแก่ผู้ใช้งาน

ABSTRACT

Currently, many people use the train transportation, especially who live in eastern suburban zone. Through our observation, there are many students and workers who always use the train transportation. Frequently, the trains are not on time bring about the waste of time and the missed of train service. Therefore, the train tracking systems have been invited. In this research, the researchers designed and developed the website, android application and monitor to display the train positions for the user.

สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญรูป	V
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของปริญญานิพนธ์	1
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง	2
2.1 จีพีเอส (GPS)	2
2.2 ระบบพิกัดในแผนที่	4
2.3 JAVA	7
2.4 การโปรแกรมเชิงวัตถุ (Object Oriented Programming)	9
2.5 ระบบปฏิบัติการแอนดรอยด์ (Android)	10
2.6 Web Server	10
2.7 ฐานข้อมูล	11
2.8 PHP(Personal Home Page)	12
2.9 JavaScript	12
2.10 บอร์ด Arduino	13
2.11 GPS Module Ublox NEO 6M	16
2.12 Voltage Sensor	17
2.13 3G Module	18
บทที่ 3 การออกแบบและการจัดทำปริญญานิพนธ์	22
3.1 การออกแบบ	22
3.2 เครื่องมือที่ใช้ในการทดลอง	46
3.3 การจัดเก็บผลการทดลอง	48

สารบัญ (ต่อ)

บทที่ 4 ผลการทดลอง 49
4.1 การทดสอบแอปพลิเคชันบนโทรศัพท์มือถือระบบแอนดรอยด์ 49
4.2 การทดสอบการติดตั้งแอปพลิเคชันบน Google Play 56
4.3 การทดสอบหน้าจอมอนิเตอร์ 57
4.4 การทดสอบการทำงานของเว็บไซต์ 59
4.5 การวัดความต่างศักย์ของแบตเตอรี่ 62
4.6 ทดสอบเปลี่ยนค่าสถานะของระบบติดตามรถไฟ (S) อัตโนมัติ 67

บทที่ 5 สรุปผลและข้อเสนอแนะ 69
5.1 สรุปผล 69
5.2 ข้อเสนอแนะ 69

บรรณานุกรม 70

ภาคผนวก โค้ดคำสั่ง 71

สารบัญรูป

รูปที่	หน้า
2.1 ดาวเทียมในการกำหนดตำแหน่งบนพื้นโลก [1]	2
2.2 การเชื่อมต่อ arduino กับ computer [7]	14
2.3 Layout & Pin out Arduino Board (Model: Arduino UNO R3) [7]	14
2.4 GPS Module Ublox NEO 6M [8]	17
2.5 Voltage Sensor 0-24 โวลต์ [9]	17
2.6 3G Module [10]	18
2.7 ส่วนประกอบของ 3G Module [10]	19
2.8 ส่วนประกอบของ 3G Module [10]	19
3.1 บล็อกไดอะแกรมของระบบติดตามรถไฟสายตะวันออกอัจฉริยะ	22
3.2 หลักการทำงานของอุปกรณ์ติดตามรถไฟในส่วนของการรับค่าพิกัด	23
3.3 หลักการทำงานของอุปกรณ์ติดตามรถไฟในส่วนของการวัดเปอร์เซ็นต์แบตเตอรี่	23
3.4 โปรแกรม Arduino IDE	24
3.5 วิธีการวัดความต่างศักย์ของแบตเตอรี่	24
3.6 หลักการทำงานโหมดประหยัดพลังงาน	25
3.7 การส่งข้อมูลเข้าฐานข้อมูล	25
3.8 แผนผัง User Interface ของแอปพลิเคชัน	26
3.10 การติดตั้งโปรแกรม Android Studio	29
3.11 การติดตั้งโปรแกรม Java SE Development Kit	29
3.12 โปรแกรม Genymotion	30
3.13 การสร้าง New Project	30
3.14 การกำหนดเวอร์ชัน API ขั้นต่ำของแอปพลิเคชัน	31
3.15 การกำหนด Activity ของแอปพลิเคชัน	31
3.16 การกำหนดชื่อไฟล์	32
3.17 โปรแกรม Android Studio พร้อมที่จะเขียนแอปพลิเคชัน	32
3.18 บล็อกไดอะแกรมการดึงข้อมูลจากฐานข้อมูลของแอปลิเคชัน	33
3.19 กำหนด AndroidManifest.xml	34
3.20 กำหนด class และ object	35
3.21 ออกแบบ layout	36
3.22 การออกแบบ PRODUCT DETAILS ใน Google Play	37
3.23 การออกแบบ GRAPHIC ASSETS ใน Google Play	38
3.24 ส่วนประกอบของหน้าจอมอนิเตอร์	38

3.25 หน้าจอมอนิเตอร์ภาษาไทย 39
3.26 หน้าจอมอนิเตอร์ภาษาอังกฤษ 39
3.27 แผนผังการทำงานหน้าจอมอนิเตอร์ 40
3.28 หน้าเว็บไซต์ใหม่ภาษาไทย 41
3.29 หน้าเว็บไซต์ใหม่ภาษาอังกฤษ 41
3.30 เมนูเลือกขบวนรถไฟในหน้าหลักของหน้าเว็บไซต์แสดงผล 42
3.31 หน้าแสดงตำแหน่งขบวน 275-276 กรุงเทพฯ-อรัญประเทศ 43
3.32 หน้าแสดงตำแหน่งขบวน 283-284 กรุงเทพฯ-บ้านพลูตาหลวง 43
3.33 หน้าแสดงตำแหน่งขบวน 281-282 กรุงเทพฯ-กบินทร์บุรี 44
3.34 หน้าแสดงตำแหน่งขบวน ขบวน 367-368 กรุงเทพ-ฉะเชิงเทรา 44
3.35 ข้อมูลที่ส่งจากติดตามรถไฟ 45
3.36 การออกแบบและเขียนโปรแกรม เพื่อเปลี่ยนสถานะ (s) อัตโนมัติ 46
3.37 GPS Module Ublox NEO 6M [9] 46
3.38 อุปกรณ์ติดตามรถไฟ 47
4.1 ไอคอนของแอปพลิเคชัน 49
4.2 หน้าหลักของแอปพลิเคชัน 50
4.3 ระบบแจ้งเตือนผู้ใช้งาน เมื่ออินเตอร์เน็ตไม่ได้ถูกเปิด 50
4.4 หน้าเมนูขบวนรถไฟ 51
4.5 หน้าแสดงผลตำแหน่งของรถไฟ 51
4.6 หน้าเมนูสถานีรถไฟที่ใกล้ตัวผู้ใช้งานที่สุดและเมนูรายการสถานีอื่นๆ 52
4.7 แจ้งเตือนเมื่อผู้ใช้งานไม่ได้เปิด GPS Service 52
4.8 หน้าเมนูรายการสถานีรถไฟที่ใกล้ตัวที่สุด 53
4.9 หน้าแสดงขบวนรถไฟที่กำลังเข้าสถานีรถไฟ 53
4.10 หน้าแสดงไม่มีขบวนรถไฟที่กำลังเข้าสถานีรถไฟ 54
4.11 หน้าแสดงตำแหน่งของรถไฟ 54
4.12 รายการสถานีที่ต้องการให้แอปพลิเคชันแจ้งเตือน 55
4.13 หน้าแสดงการแจ้งเตือนของแอปพลิเคชัน 55
4.14 แอปพลิเคชันบน Google Play Store 56
4.15 Google Play Developer Console 56
4.16 ตารางในฐานข้อมูล 57
4.17 ส่วนต่างๆของหน้าจอมอนิเตอร์ 57
4.18 หน้าจอมอนิเตอร์เมื่อค่า s มีค่าเป็น 0 58
4.19 หน้าจอมอนิเตอร์แสดงภาพเคลื่อนไหวแจ้งเตือน 59
4.20 หน้าเว็บไซต์ 59

4.21 หน้าเว็บไซต์แสดงตำแหน่งรถไฟ 60
4.22 หน้าเว็บไซต์เมื่อค่า s มีค่าเป็น 0 61
4.23 แอปพลิเคชันจาก Google Play Store 62
4.24 ตารางเวลาตามกำหนดการของการรถไฟแห่งประเทศไทย 62
4.25 การทดสอบวัดความต่างศักย์ของแบตเตอรี่ครั้งที่ 1 63
4.26 การทดสอบวัดความต่างศักย์ของแบตเตอรี่ครั้งที่ 2 63
4.27 การทดสอบวัดความต่างศักย์ของแบตเตอรี่ครั้งที่ 3 63
4.28 ค่าเฉลี่ยจากการทดสอบจากการวัดความต่างศักย์ทั้ง 3 ครั้ง 64
4.29 ค่าเฉลี่ยจากการทดสอบจากการวัดความต่างศักย์และเวลา 64
4.30 การทำ curve fitting ในส่วนของช่วงที่เป็นเส้นตรง 65
4.31 การทำ curve fitting ในส่วนของ Exponential 65
4.32 การส่งข้อมูลทุกๆ 20 วินาที 66
4.33 การส่งข้อมูลทุกๆ 40 วินาที 67
4.34 การส่งข้อมูลทุกๆ 60 วินาที 67
4.35 การเปลี่ยนสถานะของระบบติดตามรถไฟอัตโนมัติ 67
4.36 ค่าพิกัดและเปอร์เซ็นต์ของแบตเตอรี่ในฐานข้อมูล 68

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากปัจจุบันมีประชาชนจำนวนมากเดินทางด้วยรถโดยสารสาธารณะมากกว่ารถโดยสารส่วนตัว รวมไปถึงการเดินทางด้วยรถไฟโดยเฉพาะเขตชานเมืองสายตะวันออก ซึ่งมีนักเรียนนักศึกษาและคนทำงานใช้งานอยู่ประจำ แต่ขบวนรถไฟมักจะความล่าช้าและความไม่แน่นอนของเวลาตามตารางรถไฟ เช่น รถไฟจอดรออีกขบวนสวนไปก่อนเนื่องจากมีรางเดี่ยว หรือจอดรอช่วงตัดผ่านถนนเป็นต้น ส่งผลให้เสียเวลาและพลาดในการใช้บริการขบวนรถไฟที่ต้องการเดินทาง จึงเกิดปริญญานิพนธ์ระบบติดตามรถไฟสายตะวันออกอัจฉริยะขึ้นมา

ทางกลุ่มจึงได้นำความสำคัญดังกล่าวมาพัฒนาปรับปรุงต่อเพื่อสะดวกในการใช้งานมากขึ้น โดยพัฒนาส่วนของอุปกรณ์ติดตามรถไฟ ส่วนของเว็บไซต์ แอปพลิเคชันของสมาร์ตโฟนในระบบปฏิบัติการแอนดรอยด์และจอแสดงผลอัจฉริยะ

1.2 วัตถุประสงค์

1) ออกแบบและพัฒนาอุปกรณ์ติดตามรถไฟสายตะวันออกขบวน 275-276 และ 283-284

2) พัฒนาเว็บไซต์ แอปพลิเคชันของสมาร์ตโฟนในระบบปฏิบัติการแอนดรอยด์ เพื่อแสดงตำแหน่งรถไฟและให้การใช้งานของผู้ใช้บริการรถไฟสะดวกและง่ายขึ้น

3) ออกแบบและพัฒนาจอแสดงผลอัจฉริยะเพื่อแสดงตำแหน่งรถไฟ, ข่าวสารประชาสัมพันธ์ และติดตั้งใช้งานภายในสถาบัน

1.3 ขอบเขตของปริญญานิพนธ์

ผู้วิจัยออกแบบและพัฒนาอุปกรณ์ติดตามรถไฟขบวน 275-276 และ 283-284 โดยประมวลผลข้อมูลพิกัดที่ได้จากโมดูล Global Navigation Satellite System (GNSS) และวัดค่าเปอร์เซ็นต์แบตเตอรี่โดยใช้โวลต์เตจเซ็นเซอร์ (Voltage sensor) เพื่อพัฒนาอัลกอริทึมไม่ให้แบตเตอรี่หมดระหว่างการใช้งาน จากนั้นส่งข้อมูลไปยังระบบฐานข้อมูลโดยใช้โมดูลสามจี (3G Module) จากนั้นแสดงผลตำแหน่งของรถไฟบนหน้าเว็บไซต์ แอปพลิเคชันของสมาร์ตโฟน และหน้าจอแสดงผลอัจฉริยะเพื่อให้ระบบติดตามรถไฟให้บริการกับผู้ใช้งานได้จริง

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 จีพีเอส (GPS)

GPS : Global Positioning System หมายถึง "ระบบกำหนดตำแหน่งบนพื้นโลก" ระบบนี้ได้พัฒนาขึ้นโดยกระทรวงกลาโหม ประเทศสหรัฐอเมริกา ซึ่งจัดทำโครงการมาตั้งแต่ปี พ.ศ. 2521 โดยอาศัยดาวเทียมและระบบคลื่นวิทยุนำร่องและรหัสที่ส่งมาจากดาวเทียม NAVSTAR จำนวน 24 ดวง ดังรูปที่ 2.1 โดยแบ่งเป็นชุด ชุดละ 4 ดวงโดยทำการโคจรอยู่รอบโลกวันละ 2 รอบ และมีตำแหน่งอยู่เหนือพื้นโลกที่ความสูง 20,200 กิโลเมตร และองค์ประกอบหลักของ GPS ระบบกำหนดตำแหน่งบนโลก ประกอบด้วย 3 ส่วนหลัก คือ ส่วนอวกาศ (Space segment) ส่วนสถานีควบคุม (Control segment) และส่วนผู้ใช้ (User segment)

1) ส่วนอวกาศ (Space segment) เป็นส่วนที่อยู่บนอวกาศ ประกอบด้วยดาวเทียมอย่างน้อย 24 ดวง โดยมี 21 ดวง แบ่งเป็น 6 วงโคจร วงโคจรละ 4 ดวง อยู่สูงจากพื้นดินประมาณ 20,200 กิโลเมตร ทำหน้าที่ส่งสัญญาณคลื่นวิทยุจากอวกาศ

2) ส่วนสถานีควบคุม (Control segment) ประกอบไปด้วยสถานีภาคพื้นดินที่ควบคุมระบบ ที่กระจายอยู่ตามส่วนต่าง ๆ ของโลก โดยแบ่งออกเป็นสถานีควบคุมหลัก ตั้งอยู่ที่ฐานทัพอากาศในเมืองโคโลราโดปสริงส์ (Colorado Springs) มลรัฐโคโรลาโดของสหรัฐอเมริกา สถานีติดตามดาวเทียม 5 แห่ง

3) ส่วนผู้ใช้ (User segment) ประกอบด้วยเครื่องรับสัญญาณหรือเครื่องรับจีพีเอส GPS ซึ่งมีหลายขนาด สามารถพกพาติดตัวหรือ จะติดไว้ในรถ เรือ เครื่องบินได้

การทำงานของจีพีเอสใช้หลักการตรวจวัดสัญญาณที่ส่งจากดาวเทียมที่ทราบวงโคจรที่แน่นอน โดยสัญญาณนี้เป็นสัญญาณที่ส่งออกมาจากดาวเทียมทางเดียวและจะถูกรับโดยเครื่องรับ และเครื่องรับจีพีเอสจะทำหน้าที่ในการเปลี่ยนสัญญาณจากดาวเทียมเป็นตำแหน่ง (X,Y,Z) ความเร็ว (Speed) และเวลา (Time) หากต้องการทราบค่าพิกัด X, Y, Z และเวลาต้องใช้ดาวเทียมอย่างน้อย 4 ดวง ความถูกต้องของตำแหน่งขึ้นอยู่กับนาฬิกาและเครื่องรับจีพีเอส

รูปที่ 2.1 ดาวเทียมในการกำหนดตำแหน่งบนพื้นโลก [1]

ที่มาของค่าพิกัดจากเครื่องรับจีพีเอส จะได้ค่ามาจากการคำนวณสมการ Pseudorange ซึ่งเป็นสมการที่ใช้หาค่าพิกัดโดยในการคำนวณต้องอ้างอิงค่าจากดาวเทียม ดังสมการที่ 1

$$p = \rho + c(dt - dT) + d_\rho + d_{ion} + d_{trop} + \varepsilon_{mp} + \varepsilon_p \qquad (2.1)$$

โดย

p คือระยะทางเทียม (เมตร)

ρ คือระยะทางจริง (เมตร)

c คือความเร็วแสง (เมตรต่อวินาที)

dt คือค่าความผิดพลาดสัญญาณนาฬิกาของดาวเทียม (วินาที)

dT คือค่าความผิดพลาดสัญญาณนาฬิกาของเครื่องรับสัญญาณ (วินาที)

d_ρ คือค่าผิดพลาดจากดาวเทียม (เมตร)

d_{ion} คือค่าผิดพลาดในชั้น ionosphere (เมตร)

d_{trop} คือค่าผิดพลาดในชั้น troposphere (เมตร)

ε_{mp} คือค่าผิดพลาดหลายวิถี (เมตร)

ε_p คือค่าสัญญาณรบกวนจากเครื่องรับ (เมตร)

จากสมการ เมื่อเขียนค่าระยะทางจริงในระบบพิกัดคาร์ทีเซียนสามมิติ โดยคำนึงถึงค่าความผิดพลาดสัญญาณนาฬิกาเท่านั้น แล้วแทนค่าที่ทราบจะเหลือตัวแปรที่ไม่ทราบค่าเพียง 4 ตัว คือค่าพิกัดของเครื่องรับและค่าความผิดพลาดของสัญญาณนาฬิกาของดาวเทียม จึงต้องใช้ดาวเทียมอย่างน้อย 4 ดวงได้ 4 สมการในการหาค่าดังสมการที่ 2, 3, 4 และ 5

$$p1 = \sqrt{(X_{s1} - X_r)^2 + (Y_{s1} - Y_r)^2 + (Z_{s1} - Z_r)^2} + c(dt - dT_1) \qquad (2.2)$$

$$p2 = \sqrt{(X_{s2} - X_r)^2 + (Y_{s2} - Y_r)^2 + (Z_{s2} - Z_r)^2} + c(dt - dT_2) \qquad (2.3)$$

$$p3 = \sqrt{(X_{s3} - X_r)^2 + (Y_{s3} - Y_r)^2 + (Z_{s3} - Z_r)^2} + c(dt - dT_3) \qquad (2.4)$$

$$p4 = \sqrt{(X_{s4} - X_r)^2 + (Y_{s4} - Y_r)^2 + (Z_{s4} - Z_r)^2} + c(dt - dT_4) \qquad (2.5)$$

โดย

X_{sk} คือค่าพิกัดในแกน X ของดาวเทียมดวงที่ k โดยที่ k = 1, 2, 3, 4, ...

X_r คือค่าพิกัดในแกน X ของเครื่องรับ

Y_{sk} คือค่าพิกัดในแกน Y ของดาวเทียมดวงที่ k โดยที่ k = 1, 2, 3, 4, ...

Y_r คือค่าพิกัดในแกน Y ของเครื่องรับ

Z_{sk} คือค่าพิกัดในแกน Z ของดาวเทียมดวงที่ k โดยที่ k = 1, 2, 3, 4, ...

Z_r คือค่าพิกัดในแกน Z ของเครื่องรับ

การคำนวณระยะทางระหว่างจุดสองจุดบนพื้นผิวโค้งไม่สามารถใช้สมการการคำนวณระยะห่างระหว่างจุดแบบทั่วไปได้ โดยสามารถคำนวณได้จากสมการ haversine แสดงดังสมการที่ 6, 7 และ 8 [2]

$$a = \sin^2(\Delta\varphi / 2) + \cos\varphi_1 \cdot \cos\varphi_2 \cdot \sin^2(\Delta\lambda / 2) \quad (2.6)$$

$$c = 2 \cdot \tan^{-1} 2(\sqrt{a}, \sqrt{(1-a)}) \quad (2.7)$$

$$d = R \cdot c \quad (2.8)$$

โดย

φ_1 คือลองจิจูดตำแหน่งที่ 1

φ_2 คือลองจิจูดตำแหน่งที่ 2

$\Delta\varphi$ คือผลต่างลองจิจูด (เรเดียน)

$\Delta\lambda$ คือผลต่างละติจูด (เรเดียน)

d คือระยะทางระหว่างจุดสองจุดบนพื้นผิวโลก (เมตร)

R คือรัศมีโลกเท่า 6,371 (กิโลเมตร)

2.2 ระบบพิกัดในแผนที่

เนื่องจากโลกเป็นทรงกลมเมื่อมีการกำหนดตำแหน่งต่างๆ บนโลก จึงต้องถ่ายทอดตำแหน่งจากพื้นที่จริงลงมาสู่แผนที่ด้วยระบบพิกัด โดยระบบพิกัดแผนที่ คือ การอ้างอิงตำแหน่งของโลกที่ถ่ายทอดลงมาสู่แผนที่ซึ่งมีลักษณะแบนราบ โดยกำหนดให้มีจุดกำเนิดของพิกัดอยู่บนผิวโลก และมีลักษณะเป็นระบบพิกัดฉาก อันเกิดจากการตัดกันของแกนสมมติ ตั้งแต่ 2 แกนขึ้นไป ระบบพิกัดแผนที่มีอยู่ด้วยกันสองชนิด คือ ระบบพิกัด 2 มิติ และระบบพิกัด 3 มิติ ซึ่งพิกัดเหล่านี้ได้อ้างอิงกับตำแหน่งบนโลกด้วยระบบพิกัดทางภูมิศาสตร์

2.2.1 ระบบพิกัดภูมิศาสตร์ (Geographic coordinate systems)

เป็นระบบพิกัดที่กำหนดตำแหน่งต่างๆ บนพื้นโลก ด้วยวิธีการอ้างอิงบอกตำแหน่งเป็นค่าระยะเชิงมุมของละติจูด (Latitude) และลองจิจูด (Longitude) ตามระยะเชิงมุมที่ห่างจากศูนย์กำเนิดของละติจูดและลองจิจูดที่กำหนดขึ้นสำหรับศูนย์กำเนิดของละติจูด (Origin of latitude) นั้นกำหนดขึ้นจากแนวระดับที่ตัดผ่านศูนย์กลางของโลกและตั้งฉากกับแกนหมุน เรียกแนวระนาบศูนย์

กำเนิดนั้นว่า เส้นระนาบศูนย์สูตรซึ่งแบ่งโลกออกเป็นซีกโลกเหนือและซีกโลกใต้ ฉะนั้นค่าระยะเชิงมุมของละติจูด จะเป็นค่าเชิงมุมที่เกิดจากมุมที่ศูนย์กลางของโลก กับแนวระดับฐานกำเนิดมุมที่เส้นระนาบศูนย์สูตร โดยวัดค่าของมุมออกไปทางซีกโลกเหนือและทางซีกโลกใต้ ค่าของมุมจะสิ้นสุดที่ขั้วโลกเหนือและขั้วโลกใต้ มีค่าเชิงมุม 90 องศาพอดี ดังนั้นการใช้ค่าระยะเชิงมุมของละติจูดอ้างอิงบอกตำแหน่งต่างๆ นอกจากจะกำหนดเรียกค่าวัดเป็น องศา ลิปดา และพิลิปดา แล้ว จะกำกับด้วยตัวอักษรบอกทิศทางเหนือหรือใต้เสมอ เช่น ละติจูดที่ 30 องศา 20 ลิปดา 15 พิลิปดาเหนือ

ส่วนศูนย์กำเนิดของลองจิจูด (Origin of longitude) นั้น กำหนดขึ้นจากแนวระนาบทางตั้งที่ผ่านแกนหมุนของโลกตรงบริเวณตำแหน่งบนพื้นโลกที่ผ่านหอสังเกตการณ์ดาราศาสตร์เมืองกรีนิช (Greenwich) ประเทศอังกฤษ เรียกศูนย์กำเนิดนี้ว่า เส้นเมริเดียนแรก (Prime meridian) เป็นเส้นที่แบ่งโลกออกเป็นซีกโลกตะวันตกและซีกโลกตะวันออก

ค่าระยะเชิงมุมของลองจิจูดเป็นค่าที่วัดมุมออกไปทางตะวันตก และตะวันออกของเส้นเมริเดียนแรกวัดจากศูนย์กลางของโลกตามแนวระนาบที่มีเส้นเมริเดียนแรกเป็นฐานกำเนิดมุม ค่าของมุมจะสิ้นสุดที่เส้นเมริเดียนตรงข้ามกับเส้นเมริเดียนแรกซึ่งมีค่าของมุมซีกโลกละ 180 องศา การใช้ค่าอ้างอิงบอกตำแหน่งใช้เรียกกำหนดเช่นเดียวกับละติจูด แต่ต่างกันที่ต้องบอกเป็นซีกโลกตะวันตก หรือซีกโลกตะวันออกแทน เช่น ลองจิจูดที่ 90 องศา 20 ลิปดา 45 พิลิปดาตะวันตก

2.2.2 ระบบพิกัดยูทีเอ็ม (UTM coordinate systems)

ระบบพิกัดยูทีเอ็ม เป็นระบบที่ปรับมาจากระบบเส้นโครงแผนที่แบบทรานสเวิร์สเมอร์เคเตอร์ (Transverse Mercator) เพื่อเป็นการรักษารูปร่างโดยใช้ทรงกระบอกตัดลูกโลกระหว่างละติจูด 84 องศาเหนือ – 80 องศาใต้ โดยมีรัศมีทรงกระบอกสั้นกว่ารัศมีของลูกโลก ผิวทรงกระบอกจะผ่านเข้าไปตามแนวเมริเดียนของโซน 2 แนว คือ ตัดเข้ากับตัดออกเรียกลักษณะนี้ว่า เส้นตัด (Secant) ทำให้ความถูกต้องมีมากขึ้นโดยเฉพาะบริเวณสองข้างเมริเดียนกลาง

ระบบพิกัดชนิดนี้กองทัพของสหรัฐอเมริกาได้นำมาใช้ในปี ค.ศ. 1946 เพื่อให้ได้แผนที่ที่มีความละเอียดถูกต้องมากยิ่งขึ้น ระบบนี้ได้มาจากการฉายแผนที่แบบคงทิศทาง รักษารูปร่าง และมีข้อกำหนดในรายละเอียดต่างๆ ให้ถือเป็นเกณฑ์มาตรฐานเพื่อใช้งานครอบคลุมได้ทั่วโลก กำหนดให้ใช้หน่วยวัดระยะทางเป็นเมตร ระบบพิกัดยูทีเอ็ม ปัจจุบันเป็นที่ยอมรับกันอย่างแพร่หลายทั้งในกิจการทหารและกิจการพลเรือน สำหรับประเทศไทย รัฐบาลไทยกับรัฐบาลสหรัฐอเมริกาได้ทำความตกลงทำแผนที่ภายในประเทศเมื่อปี พ.ศ. 2493 โดยได้ใช้ระบบเส้นโครงแผนที่แบบทรานสเวิร์สเมอร์เคเตอร์ ระบบพิกัดยูทีเอ็ม

พื้นที่ของโลกระหว่างละติจูด 80 องศาใต้ ถึงละติจูด 84 องศาเหนือ ถูกแบ่งออกเป็นเขต (Zone) เขตละ 6 องศา รวมเป็น 60 เขต (Zone) ตามแนวลองจิจูดโดยมีหมายเลขกำกับโซนตั้งแต่ 1 ถึง 60 ตามลำดับ โดยโซนที่ 1 อยู่ระหว่างลองจิจูด 180 องศาตะวันตก ถึง 174 องศาตะวันตก โซนที่ 2 ก็อยู่ถัดไปทางด้านตะวันออกตามลำดับจนถึงโซนที่ 60 ซึ่งอยู่ระหว่างลองจิจูด

174 องศาตะวันออก ถึง 180 องศาตะวันออก และประชิดกับโซนที่ 1 ในแต่ละโซนจะมีเมริเดียนกลาง (Central meridian) เป็นของตนเอง ตัวอย่าง เช่น โซนที่ 1 ลองจิจูด 180-174 ตะวันตก มีลองจิจูด 177 ตะวันตก เป็นเมริเดียนกลาง ซึ่งจะมีแบบนี้จนครบทุกโซน

พื้นที่ในแต่ละโซนถูกแบ่งย่อยให้เป็นขอบเขตสี่เหลี่ยม โดยแนวเส้นขนานละติจูดช่วงละ 8 องศา เริ่มจากเส้นขนานละติจูด 80 องศาใต้ แบ่งทีละ 8 องศา ผ่านเส้นระนาบศูนย์สูตรไปจนถึงเส้นขนานละติจูด 72 องศาเหนือ และจากเส้นขนานละติจูด 72-84 องศาเหนือ แบ่งออกเป็นช่องละ 12 องศา รวมทั้งหมดแบ่งได้ 20 ช่องพื้นที่สี่เหลี่ยมเหล่านี้เรียกว่า เขตกริด (Grid zone) ซึ่งมีทั้งหมด 1,200 โซน การแบ่งวิธีนี้ทำให้เกิดสี่เหลี่ยมผืนผ้าเขตกริดขนาด 6 องศา x 8 องศา ยกเว้นช่วงระหว่างเส้นขนานละติจูด 72-84 องศาเหนือ มีขนาดเขตกริดเท่ากับ 6 องศา x 12 องศา เมื่อแบ่งเสร็จแล้วได้กำหนดอักษรโรมันกำกับไว้ตั้งแต่ C ถึง X (ยกเว้น I กับ O) โดยเริ่มกำหนดอักษร C ตั้งแต่โซนของละติจูด 80 องศาใต้

การแบ่งตารางเขตกริดเหล่านี้ จะมีเลขอักษรประจำโซนของกริด (UTM Grid zone destination) โดยการอ่านหมายเลขไปทางขวาแล้วอ่านขึ้น เช่น “47 Q” หมายถึง เลขกำกับโซนในแนวตั้งที่ 47 และอักษรกำกับโซนในแนวนอนที่ Q สำหรับอักษร A, B และ Y, Z ใช้สำหรับกำกับในยูนิเวอร์ซัลโพลาร์สเตริโอกราฟิก (Universal Polar Stereographic : UPS) บริเวณขั้วโลกทั้งสองข้าง

ตามระบบพิกัดยูทีเอ็มใช้หน่วยระยะทางเป็นเมตร โดยในแต่ละโซนเส้นเมริเดียนกลางตัดกับเส้นระนาบศูนย์สูตรเป็นมุมฉาก ณ จุดตัดนี้เรียกว่า จุดกำเนิดโซน ของระบบพิกัดยูทีเอ็ม ทิศทางที่ขนานกับแนวเมริเดียนกลาง และชี้ขึ้นไปทางเหนือ เรียกว่า ทิศเหนือกริด มีการกำหนดค่าพิกัดตะวันออกให้เส้นเมริเดียนกลางเป็น 500,000 เมตร (Easting 500,000 m.) ห่างจากจุดกำเนิดสมมติ (False origin) และกำหนดให้พิกัดเหนือสำหรับเส้นระนาบศูนย์สูตรไว้เป็น 2 กรณี สำหรับซีกโลกเหนือให้มีค่าเป็น 0 เมตร (Northing 0 m.) ห่างจากเส้นระนาบศูนย์สูตร ส่วนบริเวณใต้เส้นระนาบศูนย์สูตรมีค่าเป็น 10,000,000 เมตร (Northing 10,000,000 m.) ห่างจากจุดกำเนิดสมมติ ดังนั้นจุดศูนย์กำเนิดโซนของระบบพิกัดยูทีเอ็ม จึงมีค่าพิกัดเป็น E 500,000 m ; N 0 m สำหรับการใช้งานในซีกโลกเหนือและ E 500,000 m.;N 10,000,000 m. สำหรับซีกโลกใต้ นอกจากนี้ขอบเขตการใช้ค่าพิกัดยูทีเอ็มสามารถเหลื่อมเข้าไปในโซนข้างเคียงได้เป็นพื้นที่กว้าง 40 กิโลเมตร เพื่อความสะดวกในการใช้งานบริเวณขอบโซน

2.2.3 ระบบพิกัดแผนที่ GLO (General Land Office grid system)

เป็นระบบพิกัดแผนที่อีกชนิดหนึ่งที่ใช้ในการแบ่งพื้นที่สำรวจเพื่อจัดทำแผนที่ภูมิประเทศ มักใช้ในการอ่านและการทำแผนที่ธรณีวิทยา ระบบพิกัดนี้มีการแบ่งพื้นที่ออกเป็นส่วนๆ และให้ความหมายในแต่ละส่วนดังนี้

เส้นฐานและเส้นเขตเมือง (Base line and Township line) ในบริเวณที่สำรวจเส้นละติจูดที่ใช้ในการอ้างอิง (จะเป็นเส้นใดก็ได้) เรียกว่าเส้นฐาน เส้นขนานเหนือและใต้เส้นฐานในระยะห่างกันทุก 6 ไมล์ คือ เส้นเขตเมือง

เส้นเมริเดียนหลักและเส้นพิสัย (Principal meridian and Range line) เส้นลองจิจูดที่ใช้อ้างอิงในการสำรวจ เรียกว่า เส้นเมริเดียนหลัก จุดที่ตัดกับเส้นฐานเรียกว่า จุดเริ่มต้น (Initial point) เส้นที่ลากขนานกับเส้นเมริเดียนหลัก ไปทางตะวันออกและตะวันตกในระยะห่างทุก 6 ไมล์ คือ เส้นพิสัย

เขตเมือง คือ พื้นที่จัตุรัสกว้างด้านละ 6 ไมล์ ซึ่งล้อมรอบด้วยเส้นเขตเมือง และเส้นพิสัย พื้นที่ 36 ตารางไมล์นี้ กำหนดได้โดยใช้ตำแหน่งซึ่งห่างจากเส้นฐาน และเส้นเมริเดียนหลัก เช่น 2N., R.1W. อยู่ในเส้นเขตเมือง ที่ 2 เหนือจาก เส้นฐาน และเส้นพิสัย ที่ 1 ตะวันตกของเส้นเมริเดียนหลัก

ส่วนย่อย (Section) พื้นที่ 36 ตารางไมล์ ของเส้นเขตเมืองแบ่งออกเป็นรูปจัตุรัส 36 รูป มีพื้นที่รูปละ 1 ตารางไมล์ พื้นที่ 1 ตารางไมล์นี้เรียกว่า ส่วนย่อย

แผนที่รูปสี่เหลี่ยม (Quadrangle) แผนที่ภูมิประเทศซึ่งแบ่งตามระบบนี้ โดยปกติเป็นรูปสี่เหลี่ยมผืนผ้า เรียกว่า แผนที่รูปสี่เหลี่ยม พื้นที่ของแผนที่รูปสี่เหลี่ยมล้อมรอบด้วยลองจิจูดทางทิศตะวันออกและตะวันตก และละติจูดทางทิศเหนือและใต้ ชื่อของแผนที่รูปสี่เหลี่ยมเรียกตามชื่อเมืองสำคัญ หรือลักษณะภูมิประเทศที่เด่นในแผนที่ฉบับนั้น แผนที่รูปสี่เหลี่ยมที่ใช้ในสหรัฐฯ แบ่งออกตามระยะห่างระหว่างลองจิจูดและละติจูดล้อมรอบอยู่เป็น 4 ชนิด คือ

แผนที่ชุด	1	องศา	(1 Degree series)	ใช้มาตราส่วน	1: 250,000
แผนที่ชุด	30	ลิปดา	(30 Minute series)	ใช้มาตราส่วน	1: 125,000
แผนที่ชุด	15	ลิปดา	(15 Minute series)	ใช้มาตราส่วน	1: 62,500
แผนที่ชุด	7.5	ลิปดา	(7.5 Minute series)	ใช้มาตราส่วน	1: 24,000

2.3 JAVA

Java หรือ Java programming language คือภาษาโปรแกรมเชิงวัตถุ พัฒนาโดย เจมส์ กอสลิง และวิศวกรคนอื่นๆ ที่บริษัท ซัน ไมโครซิสเต็มส์ ภาษานี้มีจุดประสงค์เพื่อใช้แทนภาษาซีพลัสพลัส C++ โดยรูปแบบที่เพิ่มเติมขึ้นคล้ายกับภาษาอ็อบเจกต์ทีฟซี (Objective-C) แต่เดิมภาษานี้เรียกว่า ภาษาโอ๊ก (Oak) ซึ่งตั้งชื่อตามต้นโอ๊กใกล้ที่ทำงานของ เจมส์ กอสลิง แล้วภายหลังจึงเปลี่ยนไปใช้ชื่อ "จาวา" ซึ่งเป็นชื่อกาแฟแทน จุดเด่นของภาษาจาวาอยู่ที่ผู้เขียนโปรแกรมสามารถใช้หลักการของ Object-Oriented Programming มาพัฒนาโปรแกรมของตนด้วย Java ได้

ภาษาจาวาเป็นภาษาสำหรับเขียนโปรแกรมที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (OOP : Object-Oriented Programming) โปรแกรมที่เขียนขึ้นถูกสร้างภายในคลาส ดังนั้นคลาส

คือที่เก็บเมทอด (Method) หรือพฤติกรรม (Behavior) ซึ่งมีสถานะ (State) และรูปพรรณ (Identity) ประจำพฤติกรรม (Behavior)

ข้อดีของภาษาจาวา

1) ภาษาจาวาเป็นภาษาที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุแบบสมบูรณ์ ซึ่งเหมาะสำหรับพัฒนาระบบที่มีความซับซ้อน การพัฒนาโปรแกรมแบบวัตถุจะช่วยให้เราสามารถใช้คำหรือชื่อ ต่าง ๆ ที่มีอยู่ในระบบงานนั้นมาใช้ในการออกแบบโปรแกรมได้ ทำให้เข้าใจได้ง่ายขึ้น

2) โปรแกรมที่เขียนขึ้นโดยใช้ภาษาจาวา จะมีความสามารถทำงานได้ในระบบปฏิบัติการที่แตกต่างกัน ไม่จำเป็นต้องดัดแปลงแก้ไขโปรแกรม เช่น หากเขียนโปรแกรมบนเครื่อง Sun โปรแกรมนั้นก็สามารถถูก compile และ run บนเครื่องพีซีธรรมดาได้

3) ภาษาจาวามีการตรวจสอบข้อผิดพลาดทั้งตอน compile time และ runtime ทำให้ลดข้อผิดพลาดที่อาจเกิดขึ้นในโปรแกรม และช่วยให้ debug โปรแกรมได้ง่าย

4) ภาษาจาวามีความซับซ้อนน้อยกว่าภาษา C++ เมื่อเปรียบเทียบ code ของโปรแกรมที่เขียนขึ้นโดยภาษาจาวากับ C++ พบว่า โปรแกรมที่เขียนโดยภาษาจาวา จะมีจำนวน code น้อยกว่าโปรแกรมที่เขียนโดยภาษา C++ ทำให้ใช้งานได้ง่ายกว่าและลดความผิดพลาดได้มากขึ้น

5) ภาษาจาวาถูกออกแบบมาให้มีความปลอดภัยสูงตั้งแต่แรก ทำให้โปรแกรมที่เขียนขึ้นด้วยจาวามีความปลอดภัยมากกว่าโปรแกรมที่เขียนขึ้น ด้วยภาษาอื่น เพราะจาวามี security ทั้ง low level และ high level ได้แก่ electronic signature, public and private key management, access control และ certificates

6) มี IDE, application server, และ library ต่างๆ มากมายสำหรับจาวาที่เราสามารถใช้งานได้โดยไม่ต้องเสียค่าใช้จ่าย ทำให้เราสามารถลดค่าใช้จ่ายที่ต้องเสียไปกับการซื้อ tool และ s/w ต่าง ๆ

ข้อเสียของภาษาจาวา

1) ทำงานได้ช้ากว่า native code (โปรแกรมที่ compile ให้อยู่ในรูปของภาษาเครื่อง) หรือโปรแกรมที่เขียนขึ้นด้วยภาษาอื่น อย่างเช่น C หรือ C++ ทั้งนี้ก็เพราะว่าโปรแกรมที่เขียนขึ้นด้วยภาษาจาวาจะถูกแปลงเป็นภาษากลาง ก่อน แล้วเมื่อโปรแกรมทำงานคำสั่งของภาษากลางนี้จะถูกเปลี่ยนเป็นภาษาเครื่องอีก ทีหนึ่ง ทีล่ะคำสั่ง (หรือกลุ่มของคำสั่ง) ณ runtime ทำให้ทำงานช้ากว่า native code ซึ่งอยู่ในรูปของภาษาเครื่องแล้วตั้งแต่ compileโปรแกรมที่ต้องการความเร็วในการทำงานจึงไม่นิยมเขียนด้วยจาวา

2) Tool ที่มีในการใช้พัฒนาโปรแกรมจาวามักไม่ค่อยเก่ง ทำให้หลายอย่างโปรแกรมเมอร์จะต้องเป็นคนทำเอง ทำให้ต้องเสียเวลาทำงานในส่วนที่ tool ทำไม่ได้ ถ้าเราดู tool ของ MS จะใช้งานได้ง่ายกว่าและพัฒนาได้เร็วกว่า

2.4 การโปรแกรมเชิงวัตถุ (Object Oriented Programming)

เมื่อโปรแกรมมีความซับซ้อนมากขึ้นนักพัฒนาก็จำเป็นต้องหาวิธีการจัดการโครงสร้างของโปรแกรมเพื่อให้มีประสิทธิภาพและสามารถจัดการบำรุงรักษาได้โดยง่าย จึงเป็นที่มาของการใช้โปรแกรมเชิงวัตถุที่อาศัยแนวคิดในการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming) ซึ่งการโปรแกรมด้วยภาษาต่างๆ ในอดีตจะอาศัยลำดับการทำงานของชุดคำสั่งต่างๆ มาประกอบเป็นโปรแกรม ในขณะที่ภาษาเชิงวัตถุนอกจากจะรวบรวมคุณสมบัติที่ดีของการโปรแกรมเชิงกระบวนการไว้ ยังเพิ่มความสามารถในการจัดการกับอัลกอริทึมและโครงสร้างข้อมูลขึ้นใหม่ โดยเน้นไปที่การรวบรวมการทำงานและข้อมูลไว้ในหน่วยเดียวกันในรูปของออปเจค ขณะที่การโปรแกรมเชิงกระบวนการจะแยกทั้งสองส่วนนี้ออกจากกันอย่างชัดเจน

2.4.1 แนวคิดเชิงวัตถุ (Object Oriented Concept)

ถ้าดูพจนานุกรม Object หรือวัตถุ หมายถึง “สิ่งที่จับต้องได้ สิ่งที่สัมผัสได้ หรือสิ่งที่รู้สึกได้” ในชีวิตประจำวันของเรา วัตถุมีมากมาย เช่น หนังสือ สมุด ปากกา ฯลฯ ส่วนวิธีการแบบ Object Oriented นั้นเป็นการแปลงความต้องการของระบบให้เป็นรายการของวัตถุหรือออปเจคบูช (Grady Booch) ได้ให้นิยามของออปเจคไว้ดังนี้ “ออปเจคจะประกอบไปด้วยคุณสมบัติ 3 ประการคือ สถานะ (state), พฤติกรรมการทำงาน (Behavior) และรูปพรรณ (Identify) โครงสร้างและการกระทำของออปเจคที่คล้ายกันจะถูกกำหนดขึ้นจากคลาสเดียวกัน” ความหมายของ Object ตามนิยามที่กำหนดไว้จาก Java คือ “ซอฟต์แวร์ที่ประกอบไปด้วย ตัวแปรและ method”

2.4.1.1 ออปเจค (Object)

เกิดมาจาก class นั้นเราเรียกกันว่า instance ของ class และ object นี้จะมีส่วนประกอบที่class มีให้คือ ตัวแปรต่าง ๆ ที่อยู่ใน class ตัวแปรเหล่านี้จะมีข้อแตกต่างกัน คืออาจเป็นตัวแปรที่เป็น class variable และอาจเป็นตัวแปรที่เป็น instance variable

2.4.1.2 คลาส (Class)

คลาส (Class) เป็นกลุ่มของออปเจคที่ใช้สถานะและพฤติกรรมการทำงานร่วมกัน คลาสนำเสนอสิ่งที่เป็นนามธรรม ในขณะที่ออปเจคใช้สำหรับการนำเสนอสิ่งที่เป็นรูปธรรม ในมุมมองของการโปรแกรมคลาสจะมีลักษณะเช่นเดียวกับชนิดของข้อมูล (Data Types) ที่ถูกกำหนดไว้ในโปรแกรม ส่วนออปเจคจะเป็นเสมือนตัวแปร (Variables) ที่ใช้นั่นเอง

2.4.2 วิธีการพัฒนาเชิงวัตถุ (Object Oriented Approach)

การใช้โปรแกรมเชิงวัตถุเป็นการอาศัยแนวคิดของคลาสมาใช้เป็นส่วนประกอบหลักในการพัฒนาระบบ และเป็นจุดเริ่มต้นของวิธีการพัฒนาระบบแนวใหม่ ที่เรียกว่า การวิเคราะห์และออกแบบเชิงวัตถุ (Object Oriented Analysis and Design) ซึ่งมีแนวโน้มที่จะได้รับความนิยม

มากขึ้นเรื่อยๆ ในปัจจุบัน วิธีการเชิงวัตถุเป็นการนำเสนอแบบจำลองที่แตกต่างไปจากการพัฒนาแบบวิธีการแบบโครงสร้าง ซึ่งอาศัยการทำงานของ Function และ Procedure เป็นหลัก โดยวิธีการเชิงวัตถุจะเน้นไปที่ออปเจคภายในระบบและความสัมพันธ์ที่เกิดขึ้นระหว่างกัน แนวคิดของออปเจคเป็นการรวมการทำงานและข้อมูลไว้ในหน่วยเดียวกัน ในขณะที่วิธีการเชิงโครงสร้างจะแยกทั้งสองส่วนนี้ออกจากกันอย่างชัดเจน ดังนั้นภายในโปรแกรมแทนที่จะทำการเขียนโค้ดจำนวนมากเพื่อรองรับการทำงานทุกๆ อย่างภายในระบบ จะอาศัยการทำงานของออปเจคต่างๆ ร่วมกันแทน นั่นคือทุกสิ่งทุกอย่างภายในระบบจะถูกมองในรูปแบบของออปเจคที่มีความสัมพันธ์กัน แต่ละออปเจคสามารถติดต่อกันได้โดยอาศัยกลไกในการรับและข้อความระหว่างกัน

2.5 ระบบปฏิบัติการแอนดรอยด์ (Android)

ระบบปฏิบัติการแอนดรอยด์ (Android) คือระบบปฏิบัติการแบบเปิดเผยซอฟต์แวร์ต้นฉบับ (Open Source) โดยบริษัทกูเกิล (Google Inc.) ที่ได้รับความนิยมเป็นอย่างสูง กูเกิลแอนดรอยด์เป็นชื่อเรียกอย่างเป็นทางการของเจ้าแอนดรอยด์ เนื่องจากบริษัทกูเกิลเป็นผู้ที่ถือสิทธิบัตรในตราสัญญาลักษณ์ชื่อและรหัสต้นฉบับ (Source Code) ของแอนดรอยด์ภายใต้เงื่อนไขการพัฒนาแบบ GNL โดยเปิดให้นักพัฒนา (Developer) สามารถนำรหัสต้นฉบับไปพัฒนาปรับแต่งได้อย่างเปิดเผย (Open source) ทำให้แอนดรอยด์มีผู้เข้าร่วมพัฒนาเป็นจำนวนมากและพัฒนาไปได้อย่างรวดเร็ว เนื่องจากอุปกรณ์ที่ใช้ระบบปฏิบัติการแอนดรอยด์มีจำนวนมาก อุปกรณ์มีหลากหลายระดับหลายราคา รวมทั้งสามารถทำงานบนอุปกรณ์ที่มีขนาดหน้าจอและความละเอียดแตกต่างกันได้ ทำให้ผู้บริโภคสามารถเลือกได้ตามต้องการ

สำหรับนักพัฒนาโปรแกรม (Programmer) แล้วนั้น การพัฒนาโปรแกรมเพื่อใช้งานบนระบบปฏิบัติการแอนดรอยด์ไม่ใช่เรื่องที่ยาก เพราะมีข้อมูลในการพัฒนารวมทั้ง Android SDK (Software Development Kit) เตรียมไว้ให้กับนักพัฒนาได้เรียนรู้ และเมื่อนักพัฒนาต้องการจะเผยแพร่หรือจำหน่ายโปรแกรมที่พัฒนาแล้วเสร็จ แอนดรอยด์ก็ยังมีตลาดในการเผยแพร่โปรแกรมผ่าน Android Market แต่หากจะกล่าวถึงโครงสร้างภาษาที่ใช้ในการพัฒนานั้น สำหรับ Android SDK จะยึดโครงสร้างของภาษาจาวา (Java language) ในการเขียนโปรแกรม เพราะโปรแกรมที่พัฒนามาได้จะต้องทำงานอยู่ภายใต้ Dalvik Virtual Machine เช่นเดียวกับโปรแกรมจาวา ที่ต้องทำงานอยู่ภายใต้ Java Virtual Machine (Virtual Machine เปรียบได้กับสภาพแวดล้อมที่โปรแกรมทำงานอยู่)

2.6 Web Server

ในการติดต่อสื่อสารในรูปแบบของการร้องขอ (Request) และการตอบสนอง (Response) ระหว่าง Browser ทำงานอยู่บนคอมพิวเตอร์ผู้ใช้ (หรือเรียกว่า Client) กับ Web Server (หรือเรียกสั้นๆว่า server) ที่ทำหน้าที่ให้บริการหรือตอบสนองคำร้องขอต่างๆที่ส่งมาจาก

คอมพิวเตอร์ที่ร้องขอ ซึ่งโดยทั่วไปจะเป็นการร้องขอเว็บเพจ โดยระหว่างการติดต่อสื่อสารของ server กับ client อาจจะต้องผ่านอุปกรณ์เป็นจำนวนมาก เช่น Router หรือ Gateway เป็นต้น ส่วนการรับส่งข้อมูลจะเป็นหน้าที่ของเครือข่าย Internet โดย server จะรองรับการเชื่อมต่อจาก client ได้พร้อมกันมากกว่า 1 connection เมื่อ Web Server ให้บริการแก่ทุกคำร้องขอครบแล้ว จะหยุดรอและตรวจสอบว่ามีการร้องของบริการเข้ามาใหม่หรือไม่ หากมีการร้องขอเข้ามา จึงจะทำการตอบกลับเพื่อแจ้งให้ client ทราบว่าได้รับการร้องขอแล้ว จากนั้นจึงให้บริการตามที่ client ต้องการ

Web Server ซึ่งทำหน้าที่เป็นศูนย์กลางของเว็บไซต์ โดยในที่นี้จะใช้ Apache เป็น Web Server เนื่องจากมีความน่าเชื่อถือและมีประสิทธิภาพทำงานได้หลาย Platform รองรับการทำงานได้หลายภาษาและมีเครื่องมือจัดการเว็บไซต์หลากหลาย นอกจากนี้ยังสามารถดาวน์โหลดมาใช้โดยไม่เสียค่าใช้จ่าย

2.6.1 ประเภทของ Web Server

โดยทั่วไปโปรแกรมที่ใช้จำลอง Web Server บน Local Computer สามารถแบ่งได้เป็น 2 ประเภท

1) Stand Alone Web Server เป็นโปรแกรมที่ทำหน้าที่เป็น Web Server อย่างเดียวโดยผู้พัฒนาจะต้องหาภาษา Script โปรแกรมฐานข้อมูลและเครื่องมือที่ใช้จัดการกับฐานข้อมูลมาติดตั้งเพิ่มเอง ตัวอย่างของ Web Server ประเภทนี้ได้แก่ Apache, IIS (Internet Information Server) เป็นต้น

2) Web Server Utility เป็น Web Server ที่รวมชุดของซอฟต์แวร์ต่างๆ ที่จำเป็นในการพัฒนา Web Application ไว้โดยภายในอาจมีทั้ง Apache, PHP, PERL, MySQL และ phpMyAdmin รวมไว้ด้วยกัน ตัวอย่าง ได้แก่ AppServ เป็นต้น [5]

2.7 ฐานข้อมูล

เป็นการเก็บข้อมูลที่มีความสัมพันธ์กันให้อยู่ในหัวเรื่องหรือจุดประสงค์อย่างใดอย่างหนึ่ง เช่น ชื่อ-นามสกุล ที่อยู่ หมายเลขโทรศัพท์เพื่อน สินค้าที่มีอยู่ในคลังสินค้า เป็นต้นซึ่งเมื่อได้รับการจัดหมวดหมู่แล้วจะทำให้สามารถนำเอาส่วนประกอบนั้นๆ เป็นตัวตั้งในการนำออกมาใช้ประโยชน์ได้

ระบบฐานข้อมูล (Database System) หมายถึงระบบที่รวบรวมข้อมูลต่าง ๆ ที่เกี่ยวข้องกันเข้าไว้ด้วยกันอย่างมีระบบมีความสัมพันธ์ระหว่างข้อมูลต่าง ๆ ที่ชัดเจน ในระบบฐานข้อมูลจะประกอบด้วยแฟ้มข้อมูลหลายแฟ้มที่มีข้อมูล เกี่ยวข้องสัมพันธ์กันเข้าไว้ด้วยกันอย่างเป็นระบบและเปิดโอกาสให้ผู้ใช้สามารถใช้งานและดูแลรักษา ป้องกันข้อมูลเหล่านี้ ได้อย่างมีประสิทธิภาพ ในการนำข้อมูลมาจัดเก็บไว้ในคอมพิวเตอร์ก็เช่นเดียวกันคือ จะต้องสร้างแฟ้มข้อมูล

ขึ้นมาสำหรับจัดเก็บข้อมูลที่เกี่ยวข้องกันเป็นเรื่องๆ โดยแต่ละแฟ้มข้อมูลจะประกอบด้วยข้อมูลส่วนย่อย ๆ หรือฟิลด์ (Field) หลายชิ้นที่มีความสัมพันธ์เกี่ยวข้องกันและข้อมูลแต่ละชุดที่จัดเก็บในแฟ้มหรือระเบียน (Record) การจัดเก็บในลักษณะนี้จะแยกออกเป็นแฟ้มๆคล้ายกับการจัดเก็บในแฟ้มเอกสารที่เป็นกระดาษปกติ แต่มีประสิทธิภาพการใช้งานและการบำรุงรักษาที่ดีกว่า

เนื่องจากขอบเขตการจัดการฐานข้อมูลนั้นกว้างมาก ดังนั้นจึงจะรู้จักกับองค์ประกอบต่าง ๆ ของฐานข้อมูล ดังนี้

User คือ ผู้ใช้งานฐานข้อมูลโดยคนเหล่านั้นไม่จำเป็นต้องมีความรู้เกี่ยวกับฐานข้อมูลก็ได้ แต่รู้ว่าต้องการข้อมูลอะไรบ้างในการทำงาน

Data คือ ข้อมูลในฐานข้อมูล เป็นส่วนที่ถูกนำมาใช้งาน ถูกเก็บอยู่ภายในอุปกรณ์อิเล็กทรอนิกส์ โดยในมุมมองของผู้ใช้งานนั้นข้อมูลจะถูกเก็บไว้ในตารางต่าง ๆ ของฐานข้อมูล

DBMS (Database Management System) คือซอฟต์แวร์ที่ทำหน้าที่คอยจัดการดูแลฐานข้อมูลให้สามารถใช้งานได้ง่าย มีประสิทธิภาพ และรักษาข้อมูลที่เก็บอยู่ภายในให้เชื่อถือได้เสมอ

Database Server คือระบบคอมพิวเตอร์ที่เก็บข้อมูลในฐานข้อมูล ซึ่งมักจะติดตั้ง DBMS ไว้ภายในคอยทำหน้าที่ จัดการฐานข้อมูลโดยปกติมักจะเป็นคอมพิวเตอร์ที่มีประสิทธิภาพ การทำงานในระดับสูงมากเพราะต้องคอยรับการใช้งานพร้อม ๆ กันจาก User

DBA (Database Administrator) คือเจ้าหน้าที่ที่รับผิดชอบดูแลรักษาฐานข้อมูล โดยจะใช้ DBMS เป็นเครื่องมือ และคอยจัดการแก้ไขปัญหาที่เกิดขึ้นกับฐานข้อมูล

2.8 PHP(Personal Home Page)

ภาษาพีเอชพี (PHP : Personal Home Page) [5] ซึ่งเป็นภาษาสคริปต์ที่ถูกฝังไว้ในเว็บเพจที่สร้างด้วยภาษา HTML (Hyper Text Markup Language) โดยเว็บเพจที่มีสคริปต์ PHP แทรกอยู่นั้นจะทำงานที่ Web Server

ความสามารถพื้นฐานที่ภาษาสคริปต์ทั่วๆไปนั้นต้องมี PHP ก็มีความสามารถทำงานได้ทัดเทียมกับภาษาสคริปต์ CGI อื่นๆ ความสามารถทั่วไป เช่น การรับส่งข้อมูลจากแบบฟอร์ม, การสร้างหน้าจอที่ไม่หยุดอยู่กับที่, รับส่งคุกกี้เพื่อแลกเปลี่ยนข้อมูลระหว่างผู้ใช้งานกับ Web Server ซึ่งภาษา PHP มีความง่ายในการใช้งานโดยทำการแทรกส่วนที่เป็นเครื่องหมายพิเศษเข้าไประหว่างส่วนที่เป็นภาษา HTML ได้ทันที

2.9 JavaScript

JavaScript [6] เป็นภาษาที่ทำให้เว็บเพจสามารถโต้ตอบกับผู้ใช้งานได้ดียิ่งขึ้น เช่น ใช้สร้างปฏิทิน ภาพเคลื่อนไหวเพิ่มลูกเล่น หรือตรวจสอบความถูกต้องของข้อมูลในแบบฟอร์มที่ผู้ใช้กรอกบนเว็บ เป็นต้น การใช้งาน JavaScript บน Browser แต่ละชนิด อาจมีความแตกต่างกัน

เนื่องจากผู้ผลิตพยายามเพิ่มลูกเล่น หรือฟังก์ชันการทำงานต่างๆ ให้กับ Browser ของตน ดั้งนั้น ฟังก์ชันบางอย่างไม่สามารถนำไปใช้กับ Browser บางชนิดได้

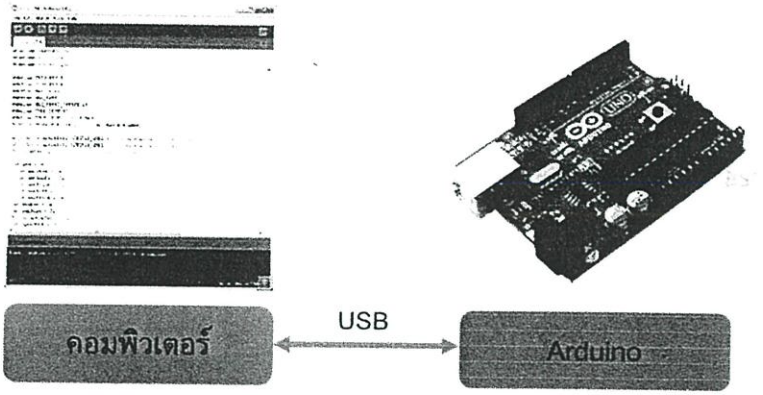
JavaScript เป็นเครื่องมือที่ใช้ควบคุม Element ต่างๆภายในเอกสาร HTML นิยมนำมาใช้รวมกับ CSS (Cascading Style Sheets) เพื่อสร้างชุดคำสั่งสำหรับจัดรูปแบบการแสดงผลของเว็บไซต์ โดย JavaScript ยังเป็นส่วนประกอบสำคัญของเทคโนโลยีที่เรียกว่า "AJAX" หรือ Asynchronous JavaScript And XML ซึ่งเป็นเทคโนโลยีที่ทำให้ผู้ใช้สามารถอัพเดตเฉพาะข้อมูลบางส่วนบนเว็บเพจที่ผู้ใช้ต้องการได้ นอกจากนี้การติดต่อสื่อสารกับ Web Server แบบ Asynchronous ยังช่วยให้ผู้ใช้ไม่ต้องหยุดรอการประมาณผลข้อมูลจาก Server ให้เสร็จทั้งหมดโดย Web Server จะส่งข้อมูลที่ประมวลผลเสร็จแล้วกลับไปก่อน แล้วส่งข้อมูลส่วนที่เหลือตามไปทีหลัง หรืออาจส่งผลลัพธ์เฉพาะข้อมูลที่ผู้ใช้ต้องการเท่านั้น AJAX เป็นเทคโนโลยีที่สำคัญที่ใช้ในการสร้าง Web 2.0 ซึ่งช่วยให้การแสดงผลเว็บเพจทำได้รวดเร็ว และมีประสิทธิภาพมากขึ้น โดยจะอัพเดตเฉพาะ Element ที่จำเป็นบนเว็บเพจ ลักษณะดังกล่าวช่วยลดปริมาณข้อมูลที่ใช้ในการติดต่อสื่อสารระหว่าง Web Server กับ Client

2.9.1 ลักษณะการทำงานของ JavaScript

จากที่เราได้ทราบแล้วว่าการเขียน JavaScript เราสามารถเขียนรวมกับแท็กคำสั่งของภาษา HTML ได้โดยการทำงานที่ฝังอยู่ในเอกสาร HTML โดย JavaScript จะทำการแปลโค้ดคำสั่งภาษาต่างๆ ที่ได้เขียนในรูปแบบ Interpleter ซึ่งจัดเก็บไว้ใน Text File ร่วมกับเอกสาร HTML ไฟล์คำสั่งภาษา JavaScript นั้นจะมีนามสกุลเป็น .HTM หรือ .HTML การเขียน JavaScript เราสามารถเขียนอยู่รวมในไฟล์เดียวกับ HTML ดังที่กล่าวเอาไว้แล้วข้างต้นหรือเขียนแยกออกเป็นไฟล์ต่างหากเองก็ได้

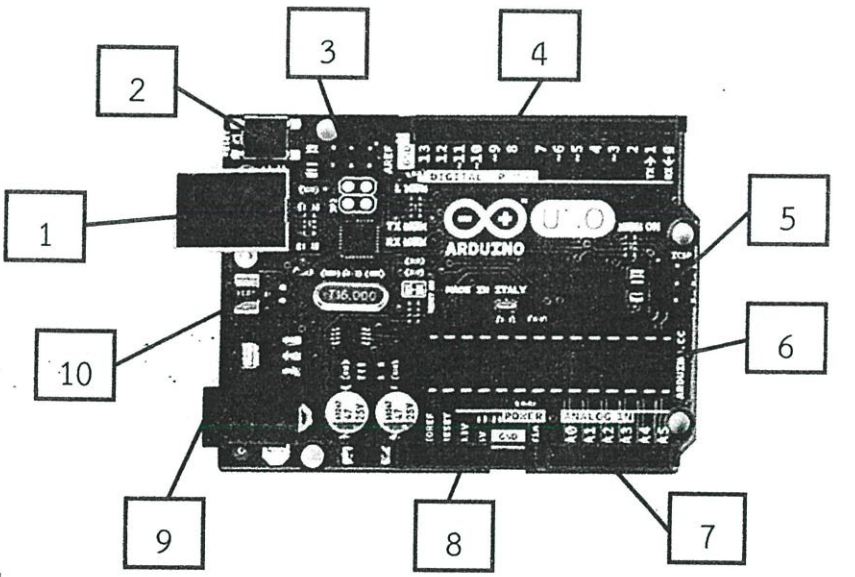
2.10 บอร์ด Arduino

Arduino เป็นบอร์ดไมโครคอนโทรเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัวบอร์ด Arduino ถูกออกแบบมาให้ใช้งานได้ง่าย สามารถเชื่อมต่อกับคอมพิวเตอร์เพื่ออัพโหลดโปรแกรมผ่านทางโปรแกรม ArduinoIDE และจ่ายไฟให้กับบอร์ด ได้ด้วยสาย USB ดังรูปที่ 2.2 ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา โดยผู้ใช้งานยังสามารถดัดแปลง พัฒนาต่อยอดได้ และบอร์ด Arduino ยังสามารถต่อกับอุปกรณ์และวงจรอิเล็กทรอนิกส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ขา I/O ของบอร์ด แสดงดังรูปที่ 2.3



รูปที่ 2.2 การเชื่อมต่อ arduino กับ computer [7]

2.10.1 ส่วนประกอบของ Arduino UNO R3



รูปที่ 2.3 Layout & Pin out Arduino Board (Model: Arduino UNO R3) [7]

1) USB port: ใช้ต่อกับ Computer เพื่ออัพโหลดโปรแกรมที่ใช้งานเข้า บอร์ด Arduino และเป็นจ่ายไฟให้กับบอร์ด

2) Reset Button: เป็นปุ่ม Reset บอร์ด Arduino

3) ICSP Port: ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2

4) I/OPort: Digital I/O ตั้งแต่ขา D0 ถึง D13

5) ICSP Port: Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader

6) MCU Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino

7) I/O Port: Analog I/O ตั้งแต่ขา A0-A5

8) Power Port: ไฟเลี้ยงของบอร์ด ประกอบด้วยขาไฟเลี้ยง +3.3 โวลต์, +5 โวลต์, GND, V_{IN}

9) Power Jack: เป็นแหล่งจ่ายไฟให้ บอร์ด Arduino

10) MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่านAtmega16U2

2.10.2 บอร์ด Arduino UNO R3

การเขียนโปรแกรม Arduino จะใช้รูปแบบการเขียนของภาษาซี ซึ่งสามารถแบ่งส่วนประกอบได้ 5 ส่วนคือ

1) ปรีโปรเซสเซอร์ไดเร็กทีฟ (Preprocessor directives) โดยส่วนนี้จะเป็นส่วนที่คอมไพลเลอร์จะมีการประมวลผลและทำตามคำสั่งก่อนที่จะมีการคอมไพล์โปรแกรม ซึ่งจะเริ่มต้นด้วยเครื่องหมายไดเร็กทีฟ (directive) หรือเครื่องหมายสี่เหลี่ยม # แล้วจึงตามด้วยชื่อคำสั่งที่ต้องการเรียกใช้ โดยส่วนนี้จะอยู่ในส่วนบนสุด หรือส่วนหัวของโปรแกรม และต้องอยู่นอกฟังก์ชันหลักใดๆ

2) ส่วนของการกำหนดค่า (Global declarations) ส่วนนี้จะเป็นส่วนที่ใช้ในการกำหนดชนิดตัวแปรแบบนอกฟังก์ชัน หรือประกาศฟังก์ชัน เพื่อให้ฟังก์ชันที่ประกาศสามารถกำหนดหรือเรียกใช้ได้จากทุกส่วนของโปรแกรม

3) ฟังก์ชัน setup() และฟังก์ชัน loop() ฟังก์ชัน setup() และฟังก์ชัน loop() เป็นคำสั่งที่ถูกบังคับให้ต้องมีในทุกโปรแกรม โดยฟังก์ชัน setup() จะเป็นฟังก์ชันแรกที่ถูกเรียกใช้ นิยมใช้กำหนดค่า หรือเริ่มต้นใช้งานไลบารี่ต่างๆ เช่น ในฟังก์ชัน setup() จะมีคำสั่ง pinMode() เพื่อกำหนดให้ขาใดๆก็ตามเป็นดิจิตอลอินพุต หรือเอาต์พุต ส่วนฟังก์ชัน loop() จะเป็นฟังก์ชันที่ทำงานหลังจากฟังก์ชัน setup() ได้ทำงานเสร็จสิ้นไปแล้ว และมีการวนรอบแบบไม่รู้จบ เมื่อฟังก์ชัน loop() งานครบตามคำสั่งแล้ว ฟังก์ชัน loop() ก็จะถูกเรียกขึ้นมาใช้อีก

4) การสร้างฟังก์ชัน และการใช้งานฟังก์ชัน (Users-defined function) ในการสร้างฟังก์ชันขึ้นมา คำสั่งต่างๆที่อยู่ภายในฟังก์ชัน ต้องอยู่ภายใต้เครื่องหมายปีกกาเปิด { และปีกกาปิด } เท่านั้น ภายใต้เครื่องหมาย {} เราสามารถนำฟังก์ชันหรือคำสั่งใดๆก็ได้มาใส่ไว้ แต่จะต้องคั่นแต่ละคำสั่งด้วยเครื่องหมายเซมิโคล่อน ; โดยจะนำคำสั่งทั้งหมดไว้บรรทัดเดียวกันเลย หรือแยกบรรทัดกันก็ได้

5) ส่วนอธิบายโปรแกรม (Progarm comments) ส่วนอธิบายโปรแกรม หรือการคอมเม้นโปรแกรมเป็นส่วนที่สำคัญอย่างมากที่จะช่วยให้ผู้ที่ไม่ได้เขียนโปรแกรม หรือเป็นผู้เขียนโปรแกรมเข้าใจโปรแกรมได้ง่ายขึ้นโดยอ่านจากคอมเม้น แทนการทำความเข้าใจโปรแกรมโดยอ่าน

แต่ละฟังก์ชัน ส่วนอธิบายโปรแกรม หรือส่วนคอมเม้นนี้ จะไม่มีผลใดๆกับขนาดของโปรแกรมหลังคอมไพล์ เนื่องจากส่วนนี้จะถูกตัดทิ้งทั้งหมดเนื่องจากไม่ได้ถูกนำไปใช้งาน มีผลเพียงแค่ว่าไฟล์โค้ดโปรแกรมจะใหญ่ขึ้นมา หากมีการคอมเม้นโค้ดเยอะๆ แต่ขนาดก็จะเพิ่มขึ้นตามตัวอักษร ดังนั้นการคอมเม้นโค้ดจึงไม่คิดพื้นที่มากนัก แต่ผู้เขียนแนะนำให้คอมเม้นโค้ดให้สั้น และกระชับ เพื่อให้เกิดความรวดเร็วในการทำความเข้าใจ การคอมเม้นโค้ดมีอยู่ 2 รูปแบบ คือ

รูปแบบที่1 เป็นเปิดด้วย /* และปิดด้วย */ เป็นการคอมเม้นโค้ดแบบข้ามบรรทัด คือ ตราบใดที่ยังไม่มี */ตรงส่วนนั้นจะเป็นคอมเม้นทั้งหมด

รูปแบบที่ 2 เป็นการคอมเม้นบรรทัดเดียว คือเปิดด้วยเครื่องหมาย // และปิดด้วยการขึ้นบรรทัดใหม่

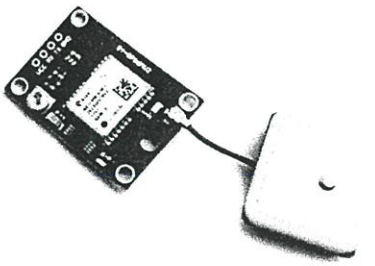
โดยบอร์ด Arduino Uno R3 มีข้อมูลจำเพาะดังนี้

1) ชิปไอซีไมโครคอนโทรเลอร์	: ATmega328
2) ใช้แรงดันไฟฟ้า	: 5V
3) รองรับการจ่ายแรงดันไฟฟ้า	: 6 – 20V
4) พอร์ต Digital I/O	: 14 พอร์ต
5) พอร์ต Analog Input	: 6 พอร์ต
6) กระแสไฟที่จ่ายได้ในแต่ละพอร์ต	: 40mA
7) กระแสไฟที่จ่ายได้ในพอร์ต 3.3V	: 50mA
8) พื้นที่โปรแกรมภายใน	: 32KB
9) พื้นที่หน่วยความจำถาวร (EEPROM)	: 1KB
10) ความถี่คริสตัล	: 16MHz
11) ขนาด	: 68.6x53.4 mm
12) น้ำหนัก	: 25 กรัม

2.11 GPS Module Ublox NEO 6M

Ublox NEO 6M เป็น GPS โมดูลขนาดเล็ก ที่มาพร้อมกับเสาอากาศแบบเซรามิกในการรับสัญญาณดังรูปที่ 2.4 โดยใช้พอร์ต 4 พอร์ต ในการต่อเข้ากับ บอร์ด Arduino คือ

1) Vcc : มีไว้สำหรับจ่ายกระแสไฟฟ้าให้กับ GPS Module
2) Rx : ใช้สำหรับรับคำสั่งจาก บอร์ด Arduino
3) Tx : ใช้สำหรับส่งข้อมูลมาที่ บอร์ด Arduino
4) GND : เป็นกราวด์ของ GPS Module



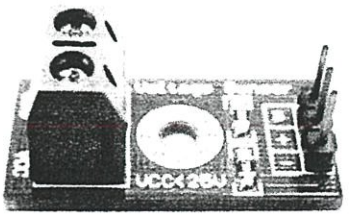
รูปที่ 2.4 GPS Module Ublox NEO 6M [8]

2.11.1 ข้อมูลจำเพาะของ GPS Module Ublox NEO 6M

1) แหล่งจ่ายไฟ 3-5 โวลต์
2) เวลาในการแก้ไขครั้งแรกจะใช้เวลา 1 วินาที
3) ความเร็วในการรับสัญญาณในที่ร่ม 162 dBm
4) มีเทคโนโลยีการป้องกันการรบกวนของสัญญาณ
5) รองรับ SBAS (WAAS, EGNOS, MSAS, GAGAN)
6) อัตราการอัพเดตตำแหน่งเป็น 5Hz
7) การทำงานในช่วงอุณหภูมิ -40 ถึง 85 ° C
8) มี EEPROM บันทึกการตั้งค่าข้อมูลพารามิเตอร์ เมื่ออุปกรณ์ไม่มีพลังงาน
9) มี LED ระบุสถานะ
10) มีแบตเตอรี่ช่วยสำรองข้อมูล
11) อัตราการส่งข้อมูลเริ่มต้น 9600

2.12 Voltage Sensor

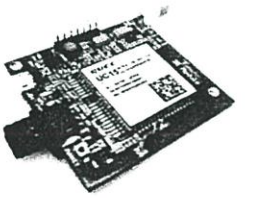
โมดูลวัดค่าแรงดันช่วง 0-24 โวลต์ ดังรูปที่ 2.5 จะใช้งานร่วมกันกับ Arduino โดยใช้หลักการของแรงดันไฟฟ้าจากขา Analog ของ Arduino ที่สามารถอ่านแรงดันไฟฟ้าได้ 0-5 โวลต์ วงจรนี้เป็นวงจรแบ่งแรงดัน ลดไฟจาก 0-24 โวลต์ ให้อยู่ในช่วง 5 โวลต์ และมีค่าสัมประสิทธิ์ทางอุณหภูมิต่ำ ค่าความต้านทานมีการเปลี่ยนเนื่องจากอุณหภูมิน้อยมาก ทำให้มีความแม่นยำในการวัด สามารถวัดได้สูงสุด 24.9 โวลต์ โดยใช้ไฟเลี้ยง 5 โวลต์ หรือ 16.5 โวลต์ ถ้าใช้ไฟเลี้ยง 3.3 โวลต์



รูปที่ 2.5 Voltage Sensor 0-24 โวลต์ [9]

2.13 3G Module

3G Module ดังรูปที่ 2.6 บอร์ดโมดูลสื่อสาร ใช้ชิพสื่อสารผ่านเครือข่ายโทรศัพท์เคลื่อนที่จาก Quectel รุ่น UC15-T รองรับคลื่นความถี่ 850/2100 MHz ในระบบ 3G UMTS และรองรับคลื่นความถี่ 850/900/1800/1900 ในระบบ 2G GSM ความเร็วในการสื่อสารข้อมูลสูงสุดแบบ HSPDA ที่ 3.6 Mbps (downlink) และ 384 Kbps (uplink)

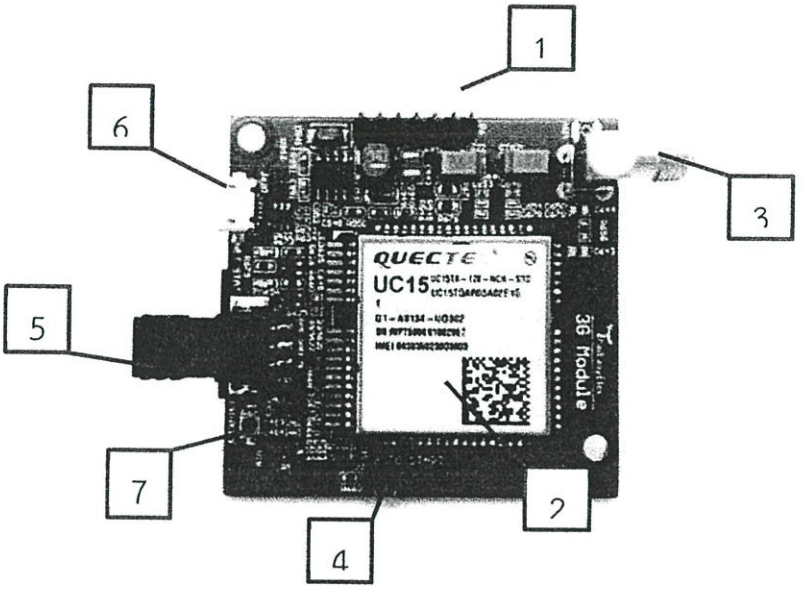


รูปที่ 2.6 3G Module [10]

2.13.1 ข้อมูลจำเพาะของบอร์ด 3G Module

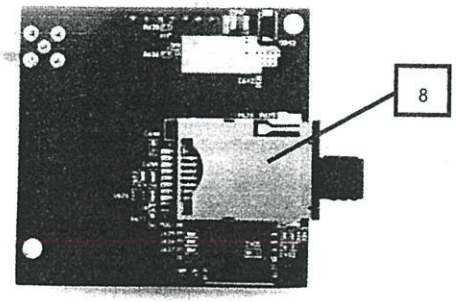
1) ใช้โมดูลสื่อสาร UC15-T จาก Quectel Wireless Solutions Co., Ltd.
2) รองรับความถี่ 850/2100 MHz ในระบบ UMTS 3G
3) รองรับความถี่ 850/900/1800/1900 MHz ในระบบ GSM 2G
4) รองรับความเร็วในการสื่อสาร HSPDA สูงสุดที่ 3.6 Mbps Downlink และ 384 Kbps Uplink
5) รองรับ HSPA Release 5 (Category 6)
6) รองรับ EDGE Downlink only
7) รองรับ GPRS Multi-slot class 12
8) รองรับ UMTS Release 99/5
9) รองรับ GSM Release 99/4
10) รองรับคำสั่ง AT Command 3GPP TS27.007 กับ 3GPP TS27.005 และ enhanced AT command ของ Quectel
11) รองรับโปรโตคอลต่างๆ ได้แก่ PPP / TCP / UDP / FTP / HTTP / FILE / MMS / SMTP / SSL

2.13.2 ส่วนประกอบของบอร์ด 3G Module



รูปที่ 2.7 ส่วนประกอบของ 3G Module [10]

1) ขาพินเฮดเดอร์ P6 สำหรับเชื่อมต่อขาสัญญาณต่างๆ
2) UC15-T Module
3) Connector แบบ SMA สำหรับเสียบเสาอากาศสื่อสารหลัก (Main Antenna)
4) Auto Start Jumper J3 สำหรับให้โมดูลเริ่มต้นทำงานอัตโนมัติ
5) Connector เชื่อมต่อสัญญาณ Audio out และ Mic In
6) Micro USB สำหรับพอร์ต USB Modem / USB AT / USB DM
7) Power Key Switch S3 สำหรับเริ่มการทางานของโมดูล
8) Sim Card Socket สำหรับใส่ sim card



รูปที่ 2.8 ส่วนประกอบของ 3G Module [10]

2.13.3 การเปิด-ปิดการทำงานโมดูล

ผู้ใช้งานสามารถเลือกเปิด-ปิดการทำงานของโมดูลได้ 3 วิธี ดังนี้

1) เปิดอัตโนมัติ ผู้ใช้สามารถให้โมดูลเปิดอัตโนมัติได้โดยเชื่อมจัมพ์เปอร์ J3 (3G Module มี Jumper มาให้) เมื่อจ่ายไฟให้กับบอร์ด 3G Module โมดูล UC15 จะเริ่มทำงานทันที

2) เปิด-ปิดด้วยโปรแกรม ผู้ใช้สามารถสั่งเปิด-ปิดโมดูลโดยต่อขา PWRKEY บนคอนเน็คเตอร์ P6 เข้ากับไมโครคอนโทรลเลอร์ตามต้องการ ตัวอย่างเช่น เชื่อมต่อกับขา D4 ของ Arduino (อ้างอิงจาก Arduino 1.0 pinout บน Arduino Uno R3) แล้วเขียนโปรแกรมควบคุมให้ขา D4 เป็น Digital Output ดังนี้

- การเปิดโมดูล หลังจากจ่ายไฟให้กับบอร์ด ในโปรแกรมให้สั่งลอจิก Low ค้างไว้ 1 – 2 วินาทีจากนั้นสั่งลอจิก High ไฟแสดงสถานะ STAT จะติด

- การปิดโมดูล ในขณะที่โมดูลทำงาน ในโปรแกรมให้สั่งลอจิก Low ค้างไว้ 1 – 2 วินาทีจากนั้นสั่งลอจิก High ไฟแสดงสถานะ STAT จะดับ

3) เปิด-ปิดด้วยตัวเอง ผู้ใช้สามารถสั่งเปิด-ปิดโมดูลโดยกดสวิตช์ S3 ดังนี้

- การเปิดโมดูล หลังจากจ่ายไฟให้กับบอร์ด ในโปรแกรมให้สั่งลอจิก Low ค้างไว้ 1 – 2 วินาทีจากนั้นสั่งลอจิก High ไฟแสดงสถานะ STAT จะติด

- การปิดโมดูล ในขณะที่โมดูลทางาน ในโปรแกรมให้สั่งลอจิก Low ค้างไว้ 1 – 2 วินาทีจากนั้นสั่งลอจิก High ไฟแสดงสถานะ STAT จะดับ

2.13.4 การใช้งาน AT Command

ชุดคำสั่งแบบ AT Command จะมีรูปแบบการใช้งานเหมือนกับ Modem มาตรฐานทั่วๆไป เพียงแต่จะมีการเพิ่มเติม Option และคำสั่งพิเศษอื่นๆเพิ่มเติมขึ้นมาอีก เพื่อให้เหมาะสมและสอดคล้องกับความสามารถในการทำงานของโมดูลได้อย่างครบถ้วน โดย รูปแบบของคำสั่งต่างๆที่เป็น AT Command นั้น จะเริ่มต้นคำสั่งด้วยรหัส ASCII ของตัวอักษร 2 ตัว คือ “A” และ “T” จากนั้นก็จะตาม ด้วยรหัสคำสั่ง และ Option ต่างๆของคำสั่ง(ถ้ามี) โดยทุกๆคำสั่งจะต้องจบด้วยรหัส Enter เสมอ เช่นคำสั่ง รีเซ็ต จะใช้รูปแบบคำสั่งเป็น “ATZ” โดย รูปแบบคำสั่งทั้งหมดจะแบ่งออกเป็น 4 แบบด้วยกัน ดังนี้

1) ทดสอบคำสั่ง มีรูปแบบคือ AT+<x>=? โดยรูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งอ่านค่ารูปแบบและพารามิเตอร์ต่างๆของคำสั่ง โดยถ้าคำสั่งนั้นมีอยู่จริง โมดูลจะตอบรับด้วยการพิมพ์ค่าของพารามิเตอร์ต่างๆของคำสั่งที่มีอยู่ทั้งหมดให้ทราบ

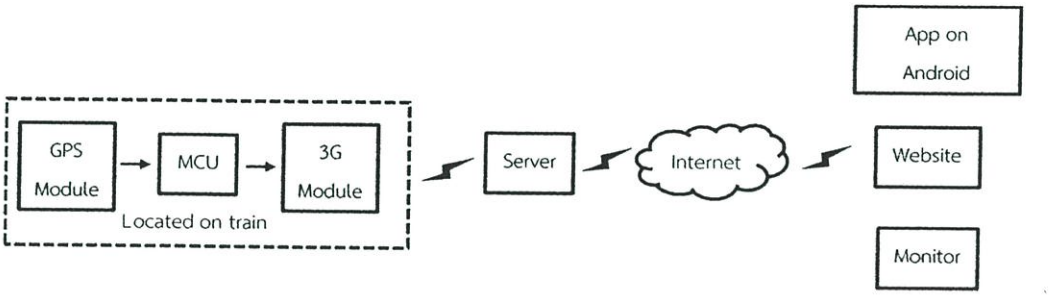
2) อ่านค่าพารามิเตอร์ มีรูปแบบคือ AT+<x>? โดยรูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งอ่านค่าพารามิเตอร์ที่กำหนดไว้แล้วของคำสั่งนั้นๆ โดยโมดูลจะตอบรับด้วยการพิมพ์ค่าพารามิเตอร์ปัจจุบันที่กำหนดไว้แล้วให้ทราบ

3) กำหนดค่าการทำงาน มีรูปแบบคือ AT+<x>=<...> โดยรูปแบบการใช้คำสั่งแบบนี้จะใช้สำหรับสั่งเขียนหรือกำหนดค่าพารามิเตอร์ให้กับคำสั่ง เช่น การกำหนดค่า Baudrate

4) สั่งให้ทำงาน มีรูปแบบคือ AT+<x> โดยรูปแบบการใช้คำสั่งแบบนี้ จะใช้สำหรับสั่งงานให้โมดูลปฏิบัติตามคำสั่งที่ต้องการ เช่น การสั่งรีเซ็ต (ATZ)

บทที่ 3
การออกแบบและการจัดทำปริญญานิพนธ์

3.1 การออกแบบ

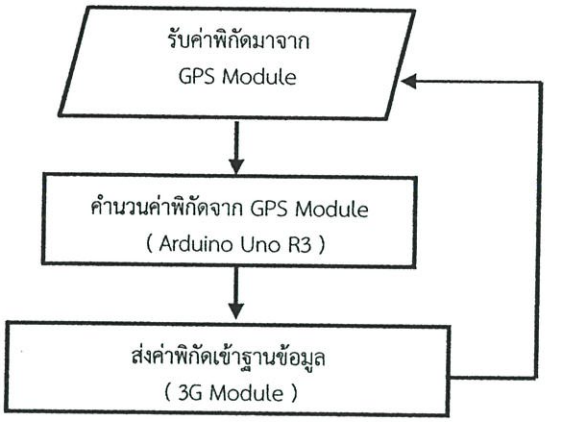


รูปที่ 3.1 บล็อกไดอะแกรมของระบบติดตามรถไฟสายตะวันออกอัจฉริยะ

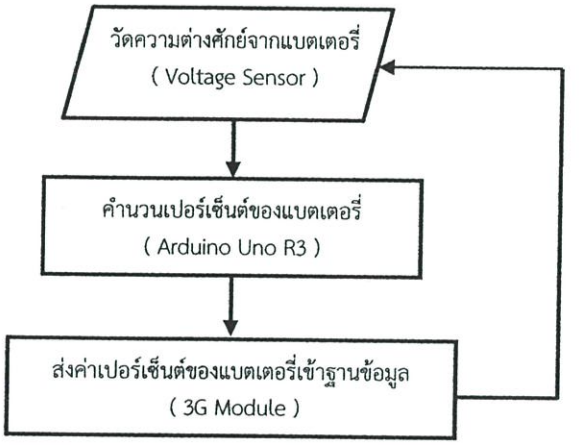
บล็อกไดอะแกรมภาพรวมของระบบติดตามรถไฟแสดงดังรูปที่ 3.1 ในส่วนแรก (กรอบเส้นประ) เป็นอุปกรณ์ที่ถูกติดตั้งบนรถไฟประกอบไปด้วย โมดูล Global Navigation Satellite System (GNSS) ที่ใช้สำหรับรับพิกัดตำแหน่ง และความเร็วของรถไฟ จากนั้น ไมโครคอนโทรลเลอร์ (MCU) จะนำข้อมูลที่ได้มาประมวลผล และส่งไปยังฐานข้อมูลผ่านโมดูลสามจี (3G Module) ในส่วนการแสดงผลนั้น ผู้วิจัยได้มีการออกแบบและพัฒนา ส่วนการแสดงตำแหน่งรถไฟแก่ผู้ใช้บริการผ่านช่องทางต่างๆ ได้แก่ เว็บไซต์ แอปพลิเคชันบนระบบปฏิบัติการณ์แอนดรอยด์ และหน้าจอมอนิเตอร์

3.1.1 การออกแบบอุปกรณ์ติดตามรถไฟ

สำหรับการออกแบบอุปกรณ์ติดตามรถไฟนั้นจะเป็นการทำงานร่วมกันของอุปกรณ์ทั้งหมด 4 ส่วน คือ GPS Module Ublox Neo-6M, Voltage Sensor, บอร์ดไมโครคอลโทรลเลอร์ Arduino Uno R3 และ 3G Module โดยมีแบตเตอรี่สำรองเป็นแหล่งจ่ายกระแสไฟฟ้า ผู้วิจัยเขียนโปรแกรมสั่งการอุปกรณ์ติดตามรถไฟ โดยใช้ภาษา C ผ่าน บอร์ด Arduino ซึ่งหลักการทำงานของอุปกรณ์ติดตามรถไฟเป็นดังรูปที่ 3.2 และรูปที่ 3.3



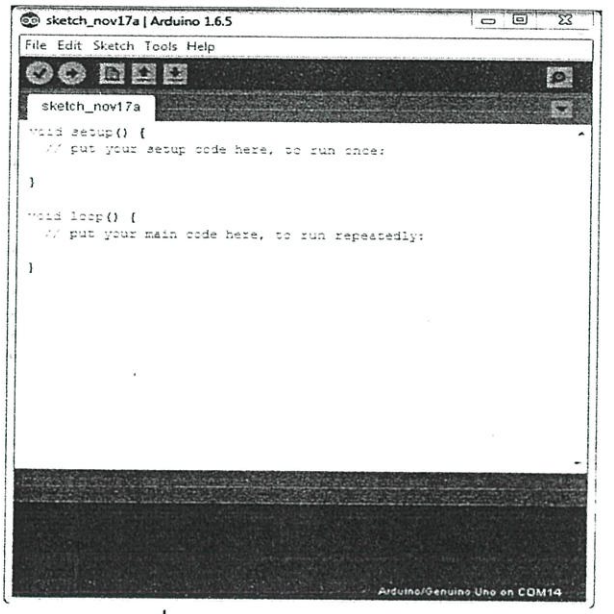
รูปที่ 3.2 หลักการทำงานของอุปกรณ์ติดตามรถไฟในส่วนของการรับค่าพิกัด



รูปที่ 3.3 หลักการทำงานของอุปกรณ์ติดตามรถไฟในส่วนของการวัดเปอร์เซ็นต์แบตเตอรี่

3.1.4.1 ติดตั้งโปรแกรม Arduino IDE

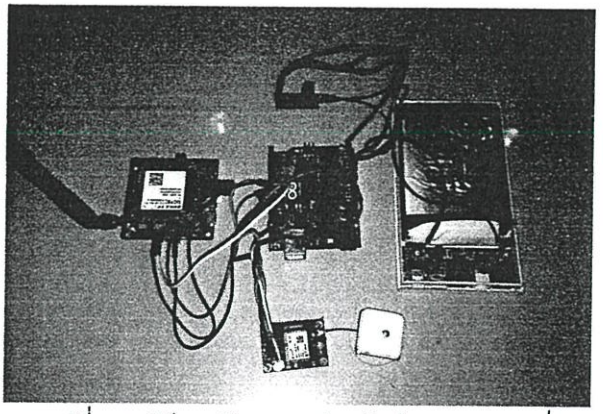
ทำการติดตั้งโปรแกรม Arduino IDE เพื่อเขียนโปรแกรมการคำนวนเปอร์เซ็นแบตเตอรี่และส่งค่าดังกล่าวเข้าสู่ฐานข้อมูล ดังรูปที่ 3.4



รูปที่ 3.4 โปรแกรม Arduino IDE

3.1.4.2 การออกแบบการวัดความต่างศักย์ของแบตเตอรี่

ทำการออกแบบการวัดความต่างศักย์ของแบตเตอรี่ด้วยโวลต์เตจเซ็นเซอร์ (Voltage-Sensor) เพื่อหาสมการอัตราการลดลงของแบตเตอรี่โดยวัดจากการใช้อุปกรณ์ติดตามรถไฟจริง เพื่อให้สมการอัตราการลดลงของแบตเตอรี่มีความแม่นยำ จากนั้นจะนำข้อมูลที่ได้ไปคำนวณเพื่อหาเปอร์เซ็นต์ของแบตเตอรี่คงเหลือ ดังรูปที่ 3.5

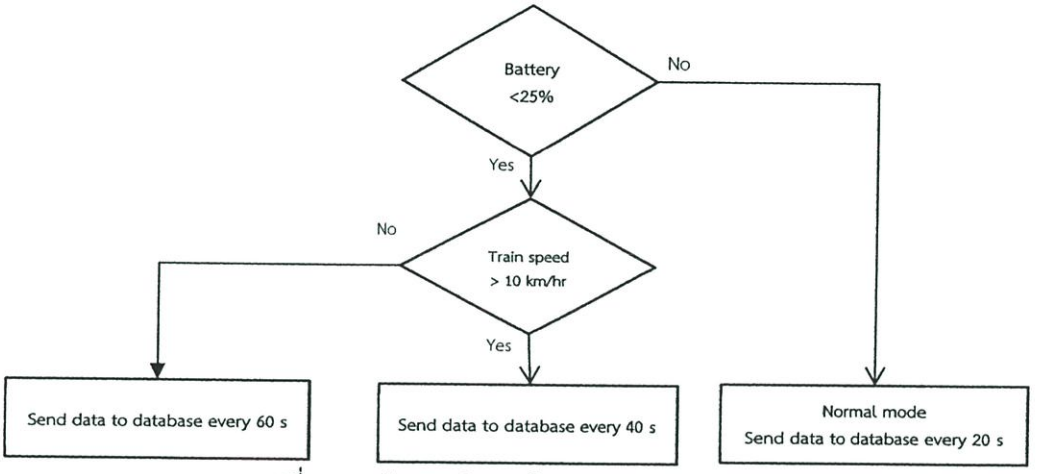


รูปที่ 3.5 วิธีการวัดความต่างศักย์ของแบตเตอรี่

3.1.4.3 การออกแบบโหมดประหยัดพลังงาน

ทำการออกแบบการใช้โหมดประหยัดพลังงานเพื่อให้อุปกรณติดตามรถไฟสามารถทำงานได้นานขึ้น เนื่องจากความถี่ในการส่งข้อมูลเข้าฐานข้อมูลมีผลต่อการใช้พลังงานจากแบตเตอรี่

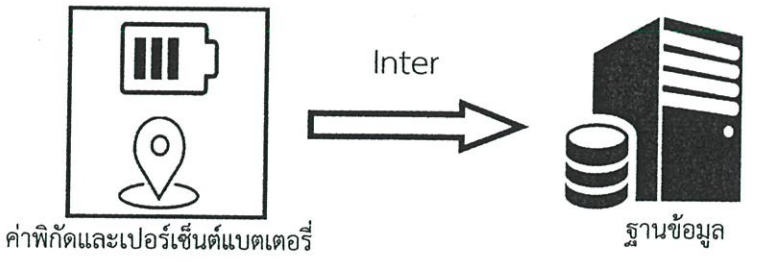
โดยถ้าส่งข้อมูลด้วยความถี่ในการส่งมากก็จะใช้พลังงานในแบตเตอรี่มาก จึงทำการออกแบบคือ ถ้าเปอร์เซ็นต์ของแบตเตอรี่มากกว่า 25 เปอร์เซ็นต์จะเป็นการส่งข้อมูลในโหมดปกติ คือจะส่งข้อมูลในทุกๆ 20 วินาที แต่ถ้าเปอร์เซ็นต์ของแบตเตอรี่น้อยกว่าหรือเท่ากับ 25 เปอร์เซ็นต์ จะเข้าสู่โหมดประหยัดพลังงาน ซึ่งจะใช้ความเร็วเป็นตัวกำหนดความถี่ในการส่งข้อมูลคือ ถ้ารถไฟมีความเร็วมากกว่า 10 กิโลเมตรต่อชั่วโมง จะส่งข้อมูลในทุกๆ 40 วินาที แต่ถ้าถ้ารถไฟมีความเร็วน้อยกว่า 10 กิโลเมตรต่อชั่วโมง จะส่งข้อมูลในทุกๆ 60 วินาที แสดงได้ดังรูปที่ 3.6



รูปที่ 3.6 หลักการทำงานโหมดประหยัดพลังงาน

3.1.4.4 การออกแบบการส่งข้อมูล

ทำการออกแบบการส่งข้อมูลซึ่งข้อมูลดังกล่าวประกอบไปด้วย ค่าพิกัดของรถไฟ เปอร์เซ็นต์แบตเตอรี่ และความเร็วของรถไฟผ่านเครือข่าย 3G แสดงได้ดังรูปที่ 3.7



รูปที่ 3.7 การส่งข้อมูลเข้าฐานข้อมูล

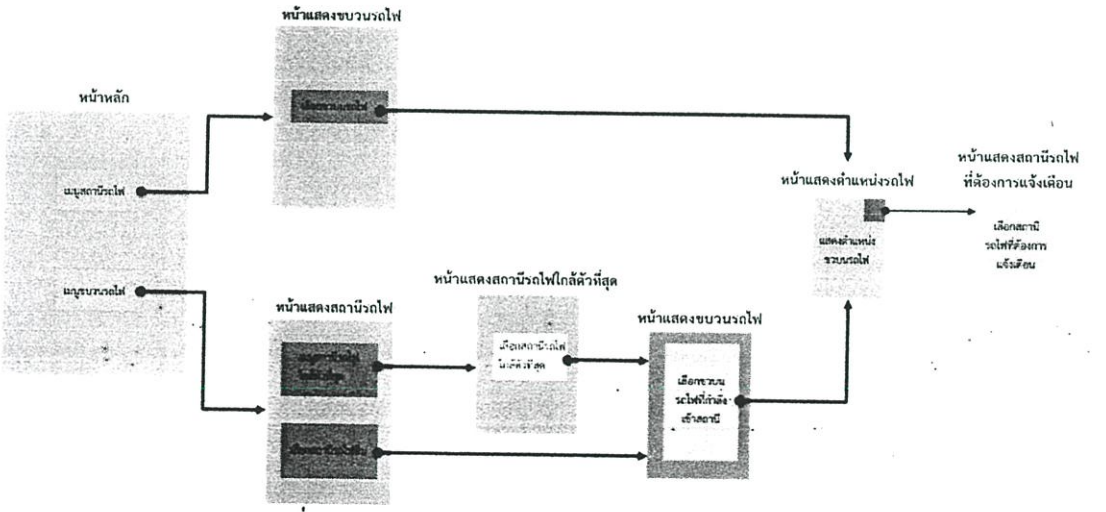
3.1.2 การออกแบบแอปพลิเคชันบนสมาร์ทโฟน

3.1.2.1 การออกแบบ User Interface และการทำงานแอปพลิเคชัน

การออกแบบ User Interface ของแอปพลิเคชันนั้น ได้ออกแบบหน้าหลักซึ่งจะแสดง 2 เมนู คือ เมนูขบวนรถไฟและเมนูสถานีรถไฟ โดยเมนูขบวนรถไฟจะแสดงขบวนรถไฟต่างๆที่กำลังเดินทางพร้อมทั้งแสดงตำแหน่งขบวนรถไฟ

ส่วนเมนูสถานีรถไฟจะแสดง 2 เมนู คือเมนูสถานีรถไฟที่ใกล้ตัวผู้ใช้งานที่สุด (โดยอ้างอิงจากตำแหน่งพิกัดของมือถือผู้ใช้)และเมนูรายการสถานีรถไฟอื่นๆ โดยจะแสดงขบวนรถไฟที่กำลังเข้าสถานีรถไฟต่างๆเมื่อผู้ใช้ทำการกดเลือกสถานี

นอกจากนั้น โครงการนี้ผู้วิจัยได้มีการออกแบบและพัฒนาระบบแจ้งเตือน ซึ่งจะมีปุ่มกดการแจ้งเตือนในหน้าแสดงตำแหน่งรถไฟ เมื่อผู้ใช้งานกดปุ่มการแจ้งเตือนแอปพลิเคชัน จะแสดงสถานีรถไฟต่างๆ เพื่อให้ผู้ใช้งานสามารถเลือกสถานีใดสถานีหนึ่งที่ผู้ใช้งานต้องการให้แจ้งเตือน ซึ่งผู้พัฒนาได้ทำการออกแบบให้ระบบแจ้งเตือน 1 สถานีก่อนเข้าสถานีที่ผู้ใช้งานเลือก โดยแอปพลิเคชันที่ทำการออกแบบจะมีการแจ้งเตือนผ่านระบบเสียงและสั่น โดยแผนผังช่วยในการทำความเข้าใจในลำดับของแต่ละหน้าดังรูปที่ 3.8



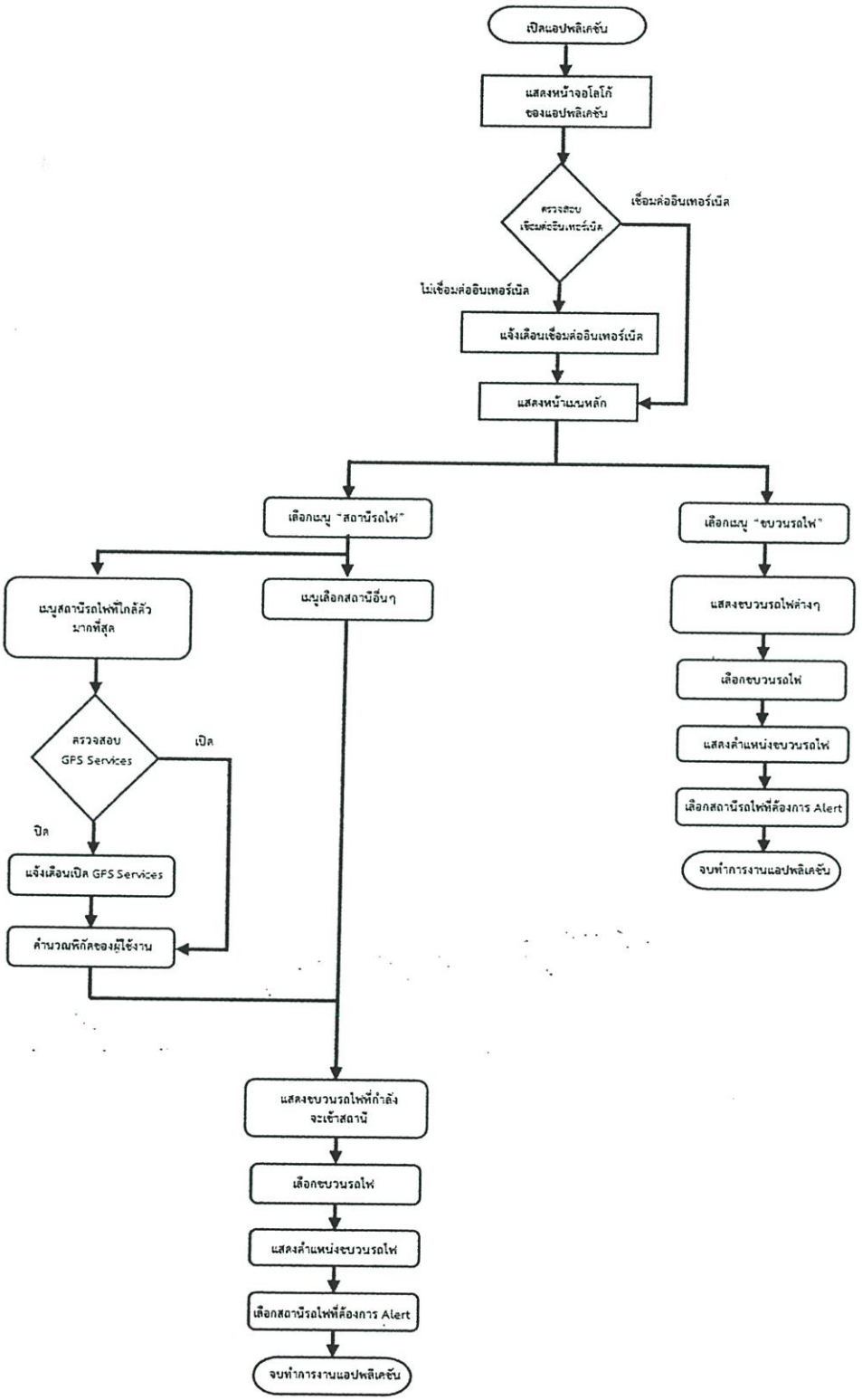
รูปที่ 3.8 แผนผัง User Interface ของแอปพลิเคชัน

แผนผังการทำงานของแอปพลิเคชันฝั่งผู้ใช้งาน ดังรูปที่ 3.9 เมื่อผู้ใช้งานเปิดแอปพลิเคชันซึ่งก่อนเข้าถึงหน้าเมนูหลักจะแอปพลิเคชันแสดงหน้าโลโก้และตรวจสอบการเชื่อมต่ออินเทอร์เน็ต เนื่องจากแอปพลิเคชันต้องใช้อินเทอร์เน็ตในการดึงข้อมูลจากฐานข้อมูล เมื่อผู้ใช้งานไม่ได้เชื่อมต่ออินเทอร์เน็ตจะทำการแจ้งเตือนให้เชื่อมต่ออินเทอร์เน็ตจึงสามารถเข้าถึงหน้าเมนูหลักได้

เมื่อเข้าหน้าเมนูหลักแล้ว หากผู้ใช้งานเลือกเมนูขบวนรถไฟ แอปพลิเคชันจะแสดงขบวนรถไฟต่างๆ และเมื่อเลือกขบวนรถไฟ แอปพลิเคชันจะแสดงตำแหน่งของขบวนรถไฟขบวนนั้นๆ ในทางตรงกันข้ามหากผู้ใช้งานเลือกเมนูสถานีรถไฟแอปพลิเคชันจะแสดงเมนูสถานีรถไฟที่ใกล้ตัวผู้ใช้งานที่สุดและเมนูรายการสถานีรถไฟอื่นๆ ก่อนเข้าถึงเมนูสถานีรถไฟที่ใกล้ตัวผู้ใช้งานที่สุด แอปพลิเคชันจะทำการตรวจสอบ GPS Services เนื่องจากแอปพลิเคชันต้องรับค่าละติจูดและลองจิจูดของผู้ใช้งานมาคำนวณเพื่อหาสถานีรถไฟใกล้ตัวที่สุด เมื่อผู้ใช้งานไม่ได้เปิด GPS Services แอปพลิเคชันจะทำการแจ้งเตือนให้เปิด GPS Services จึงสามารถเข้าถึงหน้าเมนูสถานีรถไฟที่ใกล้ตัวผู้ใช้งานที่สุดได้ พร้อมกันนั้นแอปพลิเคชันยังมีการแสดงขบวนรถไฟที่กำลังเข้าสถานีรถไฟอื่นๆ โดยเมื่อเลือกขบวนรถไฟนั้นๆ แสดงตำแหน่งขบวนรถไฟ มากกว่านั้นผู้วิจัยยังมีการสร้างระบบแจ้งเตือนซึ่งจะมีเมนู Alert เป็นเมนูสำหรับการแจ้งเตือนรถไฟกำลังเข้าสถานีนั้นๆที่ผู้ใช้งานเลือก

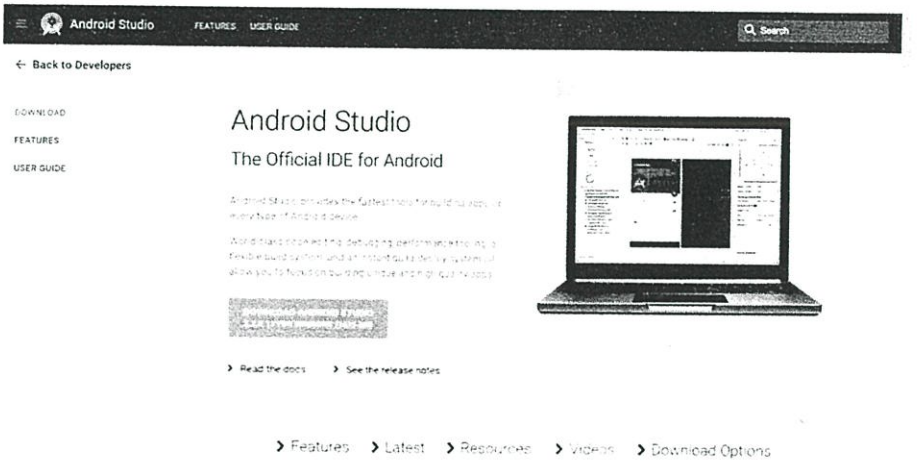
3.1.2.2 การติดตั้งโปรแกรมเพื่อออกแบบแอปพลิเคชันสำหรับสมาร์ทโฟน

สำหรับการออกแบบแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ จะใช้โครงสร้างทางภาษาจาวา (JAVA) เป็นหลักโดยมีรูปแบบการเขียนเชิงวัตถุ (Object Oriented Programming) โดยการเขียนบนแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์บน Android Studio จะต้องติดตั้งโปรแกรมดังนี้



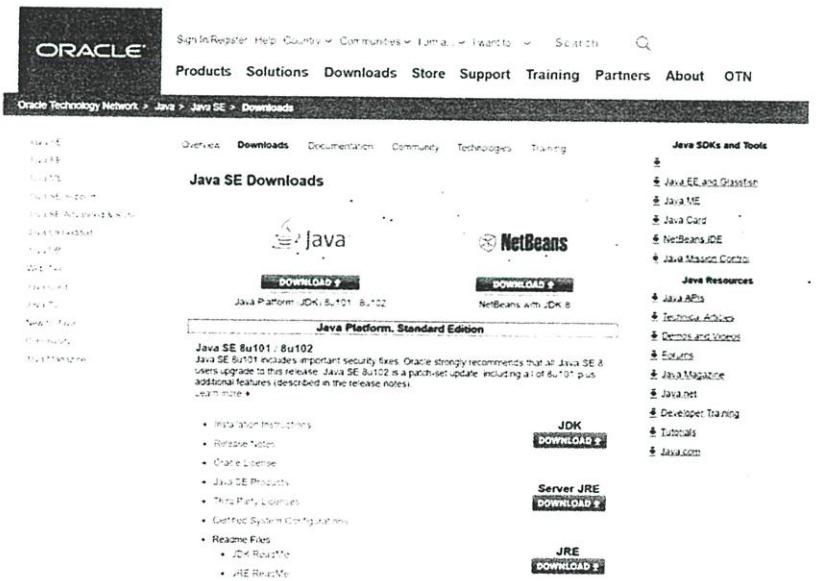
รูปที่ 3.9 แผนผังการทำงานของแอปพลิเคชัน

1) ติดตั้ง Android Studio ดังรูปที่ 3.10 [3]



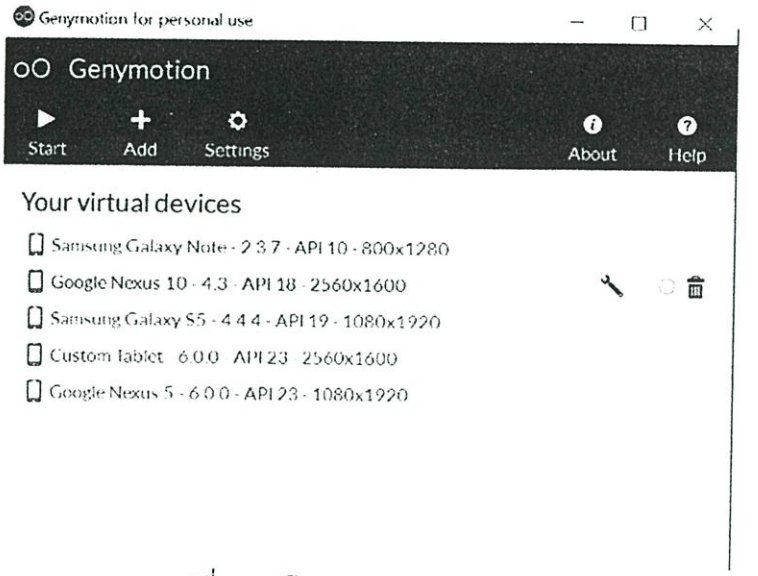
รูปที่ 3.10 การติดตั้งโปรแกรม Android Studio

2) ติดตั้งโปรแกรม Java SE Development Kit สำหรับการพัฒนาแอปพลิเคชันบนแอนดรอยด์ ดังรูปที่ 3.11 [4]



รูปที่ 3.11 การติดตั้งโปรแกรม Java SE Development Kit

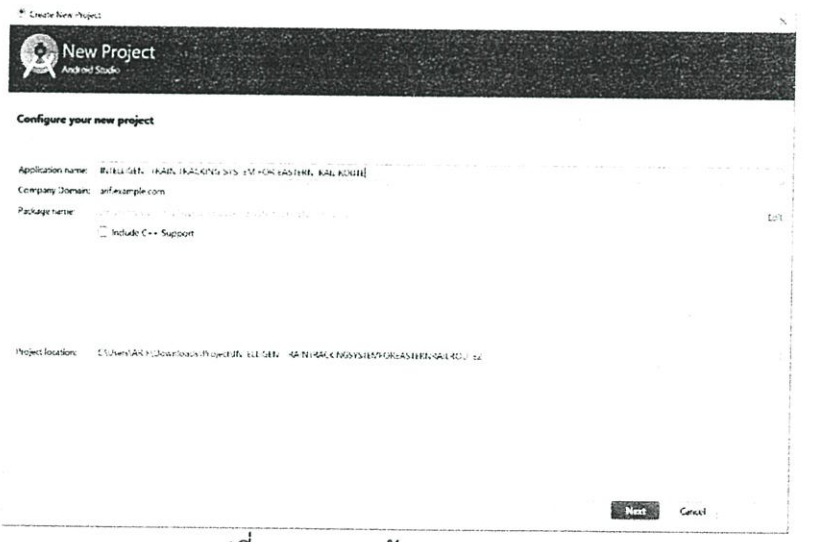
3) ติดตั้ง Genymotion Simulator เป็น Android Virtual Devices หรือที่เราเรียกย่อๆว่า AVD เป็นการจำลองหรือ emulator สำหรับทดสอบโปรแกรม android ดังรูปที่ 3.12



รูปที่ 3.12 โปรแกรม Genymotion

3.1.2.3 การออกแบบแอปพลิเคชันแสดงตำแหน่งรถไฟ

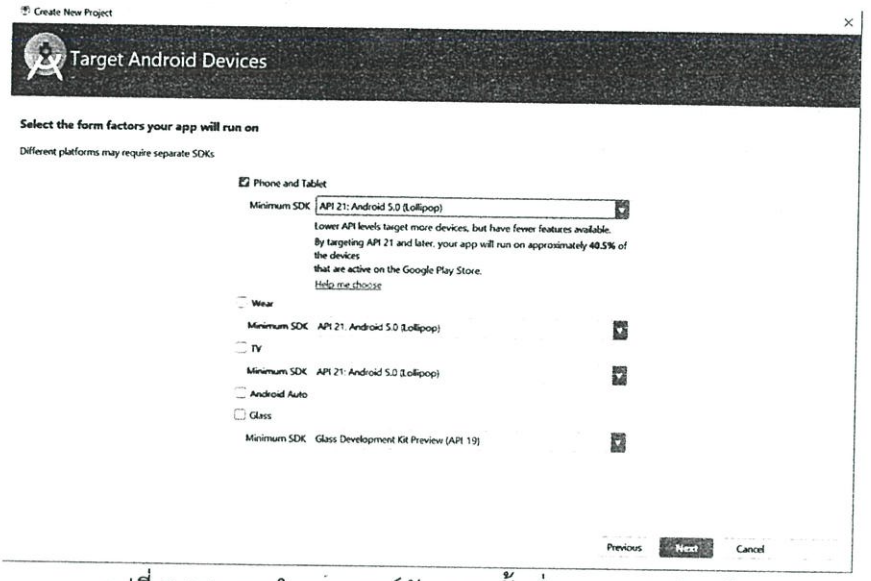
1) เปิด Android Studio เลือก Start a new Android Studio project เพื่อทำการสร้างโปรเจคใหม่เปิด โดยตั้งชื่อว่า “INTELLIGENT TRAIN TRACKING SYSTEM FOR EASTERN RAIL ROUTE” ดังรูปที่ 3.13



รูปที่ 3.13 การสร้าง New Project

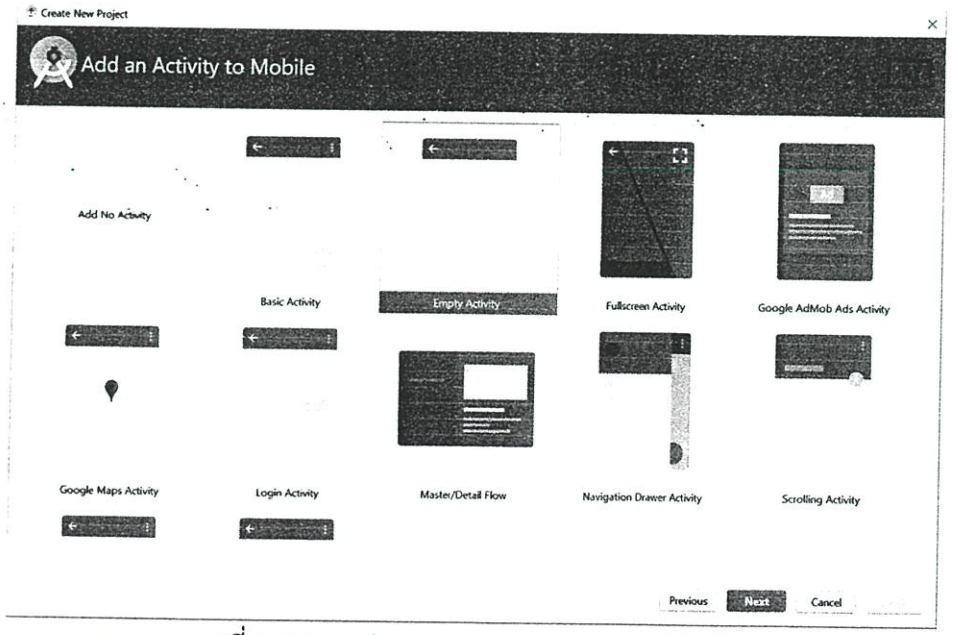
2) การกำหนดว่าจะสร้างโปรเจคสำหรับอุปกรณ์ต่างๆ โดยเลือกเป็น Phone and Tablet โดยจะมีการกำหนดเวอร์ชันขั้นต่ำที่รองรับ (อุปกรณ์แอนดรอยด์ที่เวอร์ชันต่ำกว่าจะไม่

สามารถติดตั้งได้ แต่เวอร์ชันที่สูงกว่าจะสามารถติดตั้งใช้งานได้) ซึ่งเลือก API 21: Android 5.0 (Lollipop) ดังรูปที่ 3.14



รูปที่ 3.14 การกำหนดเวอร์ชัน API ขั้นต่ำของแอปพลิเคชัน

3) การกำหนด Activity ของแอปพลิเคชันเบื้องต้น โดยเลือก “Empty Activity” จะเป็นหน้า Activity ว่างเปล่า ดังรูปที่ 3.15

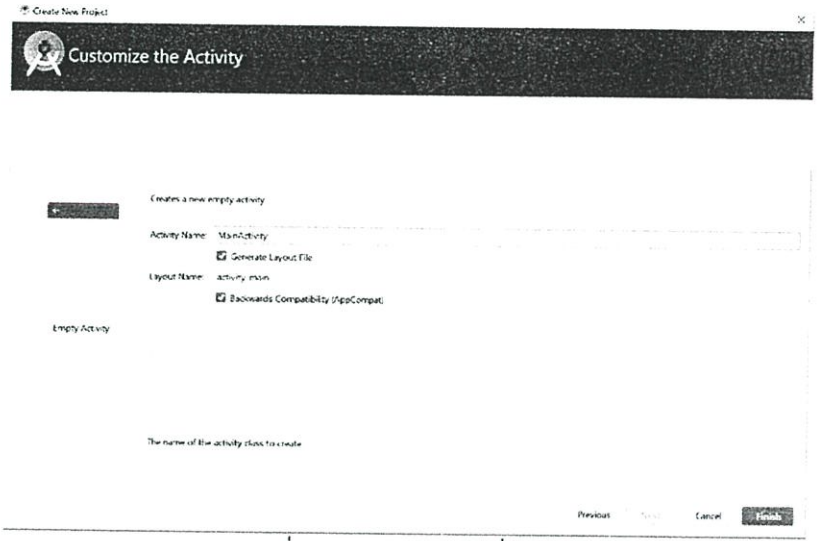


รูปที่ 3.15 การกำหนด Activity ของแอปพลิเคชัน

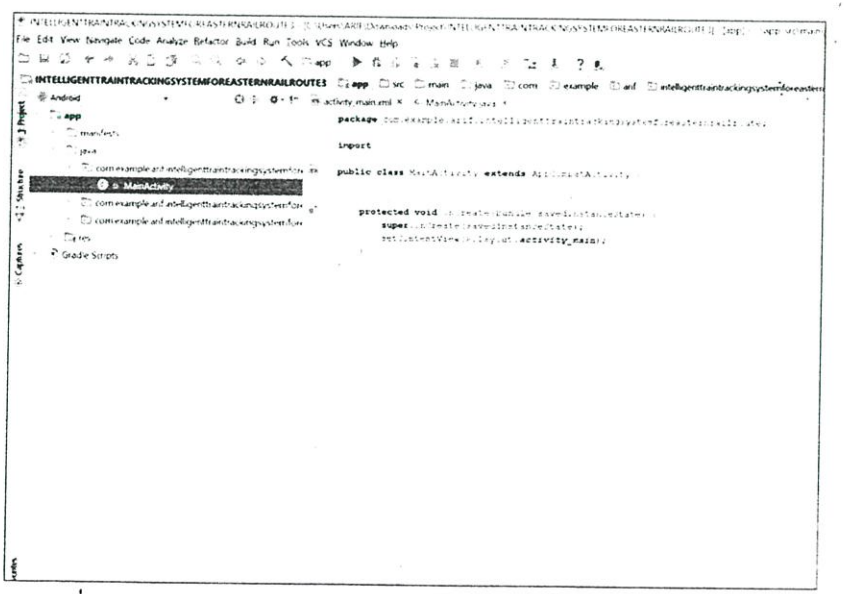
4) การกำหนดชื่อไฟล์ ดังรูปที่ 3.16 โดยจะมีดังนี้

- Activity Name ชื่อไฟล์จาวาสำหรับเขียนโค้ดควบคุมการทำงาน Activity
- Layout Name ชื่อไฟล์ .xml สำหรับออกแบบหน้าตาของแอปพลิเคชัน
- Title ชื่อหัวข้อที่จะแสดงในหน้าแรก
- Manu Resource Name ชื่อไฟล์ .xml ที่ใช้สร้างเมนูย่อยใน Activity นั้นๆ

และสุดท้ายจะได้หน้าตาของโปรแกรม Android Studio พร้อมที่จะเขียนแอปพลิเคชันดังรูปที่ 3.17



รูปที่ 3.16 การกำหนดชื่อไฟล์



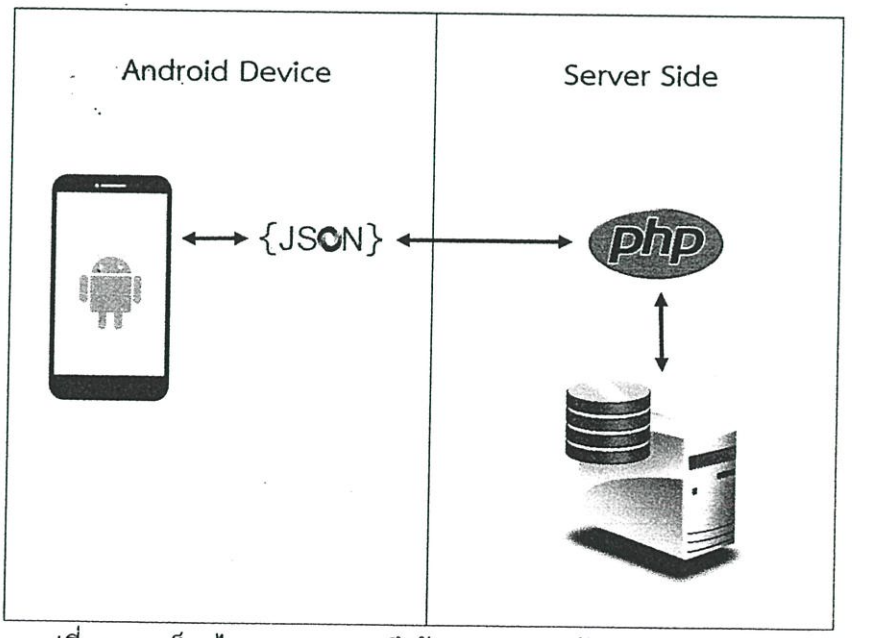
รูปที่ 3.17 โปรแกรม Android Studio พร้อมที่จะเขียนแอปพลิเคชัน

5) การรับข้อมูลจากฐานข้อมูล (เช่น พิกัด ความเร็วของรถไฟ เปอร์เซ็นต์แบตเตอรี่ เป็นต้น) ดังรูปที่ 3.18 แสดงเป็นบล็อกไดอะแกรมในการแบ่งการทำงานเป็น 2 ส่วน คือส่วนของโทรศัพท์แอนดรอย์รับส่งข้อมูลโดยใช้ JSON ซึ่งเป็นมาตรฐานในการรับส่งข้อมูลผ่านอินเตอร์เฟส (Interface) จะรับส่งข้อมูลภายนอกแอปพลิเคชัน และส่วนของเซิร์ฟเวอร์จะใช้ HTTP ผ่านไฟล์ PHP

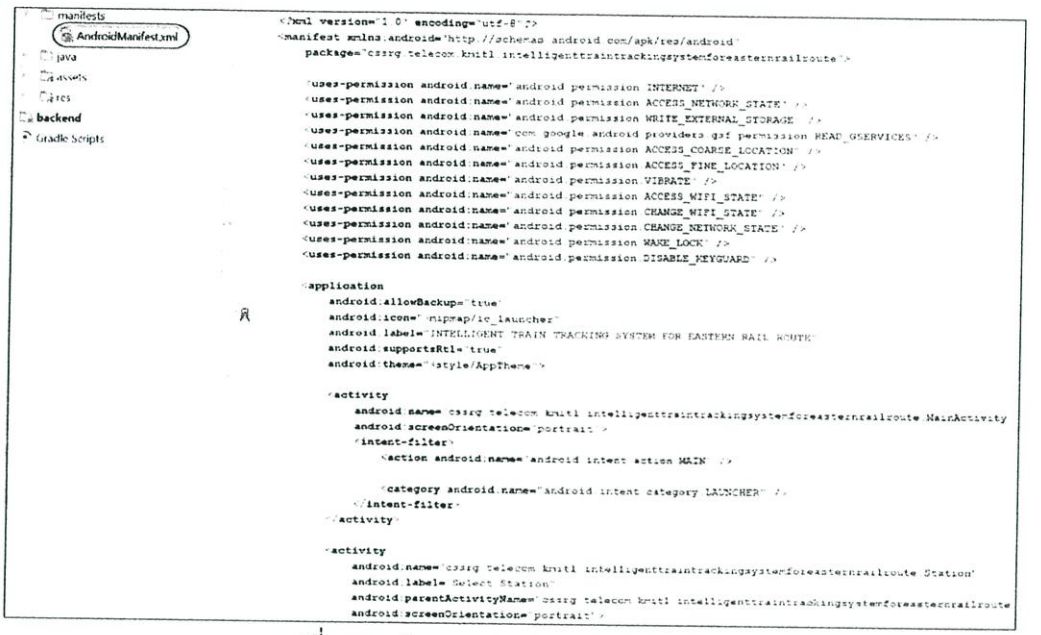
6) กำหนด AndroidManifest.xml เป็นไฟล์จัดการการทำงานโดยรวมของแอปพลิเคชัน ซึ่งจะกำหนดคุณสมบัติต่างๆของแอปพลิเคชันไว้ในไฟล์นี้เป็นหลัก อย่างเช่น ชื่อแอปพลิเคชัน, เวอร์ชันของแอปพลิเคชัน, API ที่รองรับ Activity ที่จะให้ทำงาน และการขออนุญาติใช้งานบางอย่าง (Permission) ซึ่งจะเป็นการทำงานโดยรวมของแอปพลิเคชันเท่านั้น ซึ่งจะไม่เกี่ยวกับการเขียนโค้ดหรือรูปแบบหน้าตาแอปพลิเคชัน โดยกำหนดดังนี้

- android:icon กำหนดไอคอนที่จะใช้ในแอปพลิเคชัน
- android:label กำหนดชื่อหัว (Title) ของแอปพลิเคชัน
- android:theme กำหนดรูปแบบของธีมที่จะใช้กับแอปพลิเคชัน
- android:name กำหนดชื่อ Activity ที่จะใช้งาน
- android:label กำหนดชื่อหัว (Title) เมื่อเข้าสู่หน้า Activity นั้นๆ
- android:screenOrientation กำหนดการล็อคทิศทางของหน้าจอ

และกำหนด Permission เพื่อประกาศขออนุญาตบางอย่างที่ต้องเข้าถึงข้อมูลหรือการทำงานบางส่วนที่ระบบป้องกันไว้ เช่น การเชื่อมต่ออินเทอร์เน็ตและ GPS Services เป็นต้น ดังรูปที่ 3.19



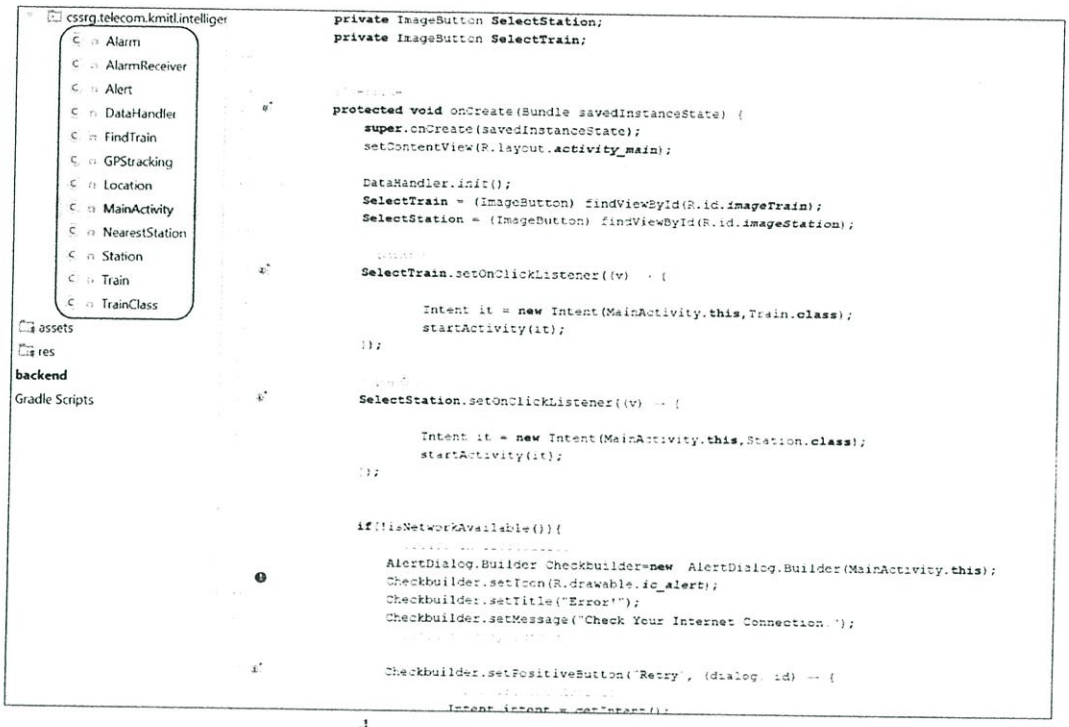
รูปที่ 3.18 บล็อกไดอะแกรมการดึงข้อมูลจากฐานข้อมูลของแอปลิเคชัน



รูปที่ 3.19 กำหนด AndroidManifest.xml

7) กำหนด class และ object ด้วยภาษาจาวาเพื่อออกแบบการใช้งานแอปพลิเคชัน โดยในส่วนของการทำงานของแอปพลิเคชัน ดังรูปที่ 3.20 เพื่อให้แสดงในส่วนการแสดงผล โดยกำหนด class ดังนี้

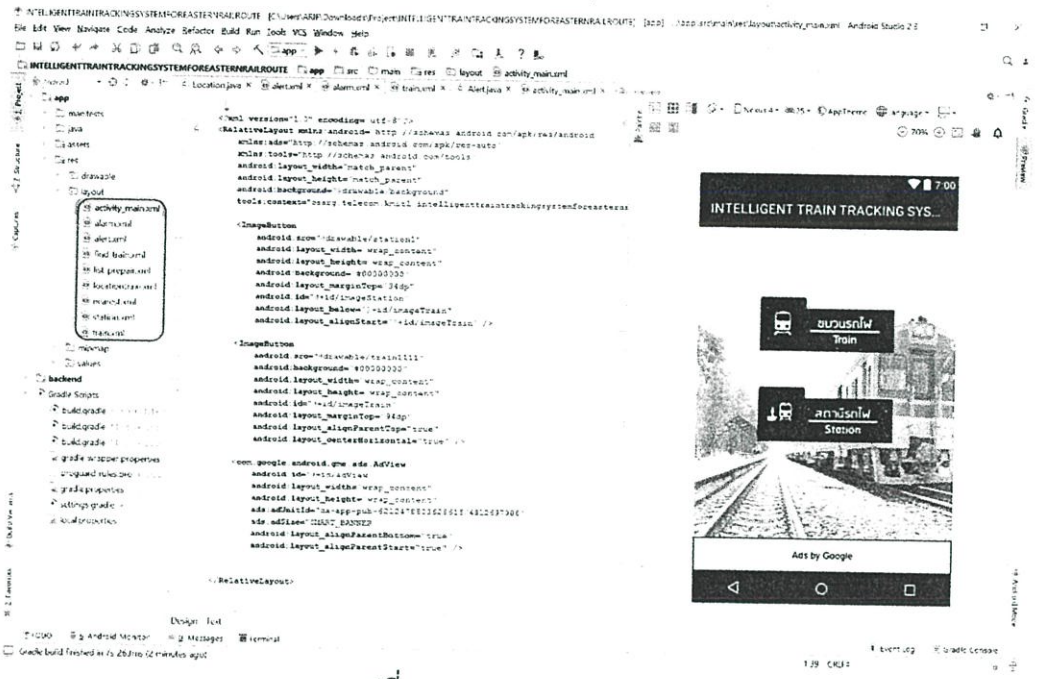
- MainActivity กำหนดการทำงานของหน้าหลักของแอปพลิเคชัน
- Train กำหนดการทำงานของเมนูขบวนรถไฟ
- Station กำหนดการทำงานของเมนูสถานีรถไฟ
- DataHandler กำหนดการทำงานเพื่อรับข้อมูลจากฐานข้อมูลและรายละเอียดข้อมูลของขบวนรถไฟในแต่ละขบวน
- TrainClass กำหนดรับส่งข้อมูลของขบวนรถไฟในแต่ละขบวนเพื่อใช้ class อื่นๆ
- Location กำหนดการทำงานเพื่อหาตำแหน่งของขบวนรถไฟ
- GPStracking กำหนดการทำงานเพื่อดึงพิกัดของผู้ใช้งาน
- NearestStation กำหนดการทำงานเพื่อหาสถานีรถไฟใกล้ตัวที่สุด
- FindTrain กำหนดการทำงานเพื่อหาขบวนรถไฟที่กำลังเข้าสถานีต่างๆ
- Alarm กำหนดการทำงานในการแสดงไดอะล็อคเตือนพร้อมเสียง
- Alert กำหนดการทำงานของเมนู Alert
- AlarmReceiver กำหนดการทำงานรับการแจ้งเตือน และเรียก Class Alarm ให้แสดง



รูปที่ 3.20 กำหนด class และ object

8) กำหนด layout เพื่อออกแบบ User Interface ของแอปพลิเคชัน จะใช้ภาษา XML ดังรูปที่ 3.21 โดยกำหนด layout ดังนี้

- activity_main.xml แสดงหน้าหลักของแอปพลิคชัน
- train.xml แสดงหน้าเมนูขบวนรถไฟ
- station.xml แสดงหน้าเมนูสถานีรถไฟ
- location.xml แสดงตำแหน่งของขบวนรถไฟ
- nearest.xml แสดงสถานีรถไฟใกล้ตัวที่สุด
- find_train.xml และ list_prepair.xml แสดงขบวนรถไฟที่กำลังเข้าสถานีต่างๆ
- alarm.xml และ alert.xml แสดงการแจ้งเตือนเมื่อขบวนรถไฟกำลังเข้าสถานี



รูปที่ 3.21 ออกแบบ layout

3.1.3 การออกแบบการติดตั้งแอปพลิเคชันบน Google Play

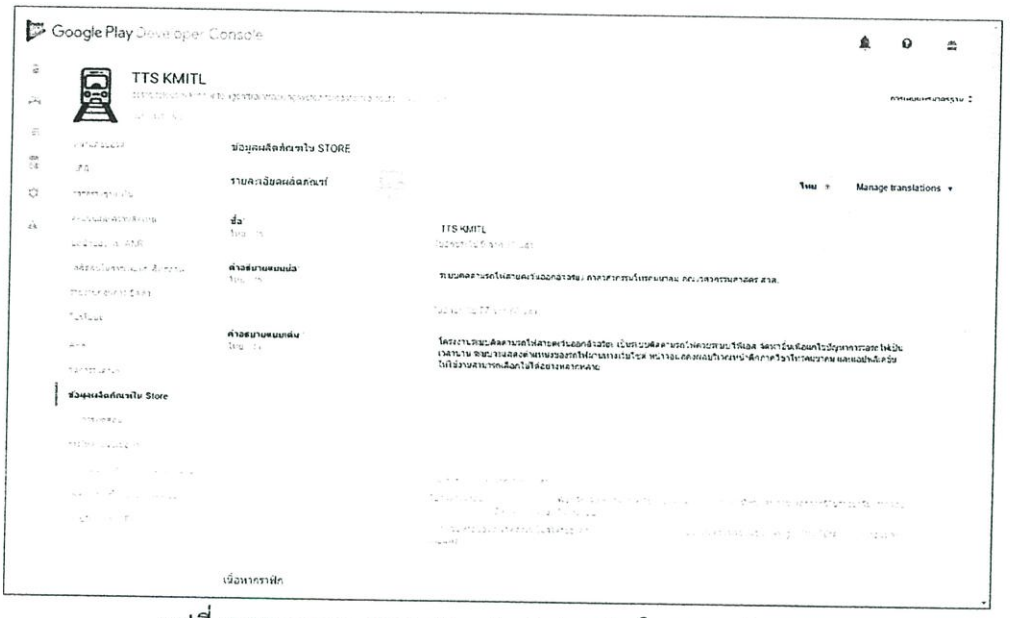
การติดตั้งแอปพลิเคชันบน Google Play จำเป็นต้องมีการสมัคร Developer Account และเมื่อสมัครเสร็จแล้ว จะสามารถอัพไฟล์ APK ขึ้น Google Play ได้

ข้อมูลรายละเอียดเกี่ยวกับแอปพลิเคชันจะแบ่งเป็นส่วนๆ ส่วนแรกสุดคือ PRODUCT DETAILS ดังรูปที่ 3.22

Title: ชื่อแอปพลิเคชันที่จะแสดงใน Google Play

Short description: คำอธิบายเกี่ยวกับแอปพลิเคชันแบบสั้นๆไม่เกิน 80 ตัวอักษร

Full description: คำอธิบายเกี่ยวกับแอปพลิเคชันแบบเต็มๆไม่เกิน 4000 ตัวอักษร



รูปที่ 3.22 การออกแบบ Product details ใน Google Play

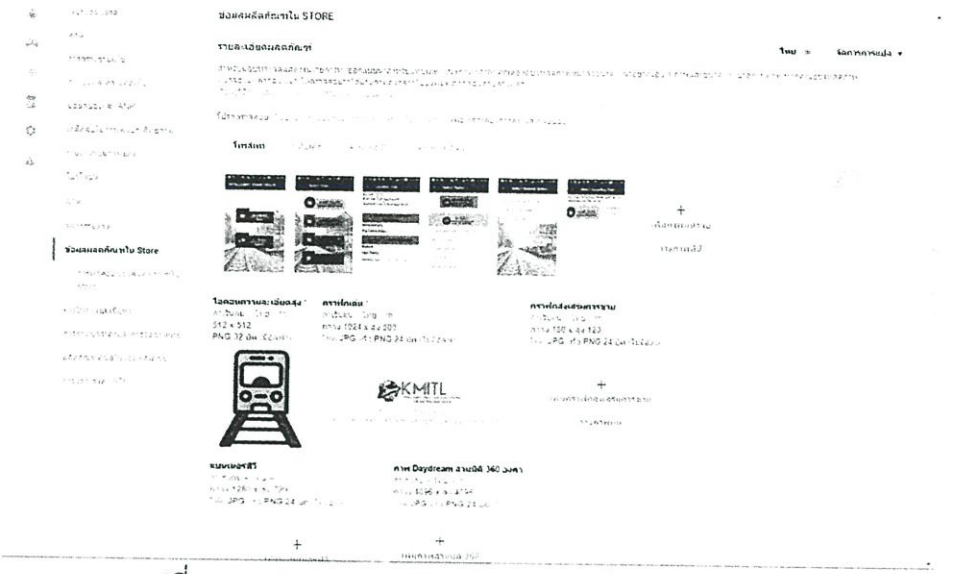
ต่อมาเป็นส่วนของ Graphic assets สำหรับออกแบบรายละเอียดเกี่ยวกับภาพ และสามารถใส่ภาพที่พื้นหลังเป็น Transparent ได้ด้วย ดังรูปที่ 3.23

Screenshots: ภาพตัวอย่างการใช้งานแอปพลิเคชัน โดยต้องใส่ภาพหน้าจอหรือสามารถทำภาพขึ้นมาเพื่อใส่ในส่วนนี้ได้ เช่น ภาพประกอบที่มีข้อความอธิบายวิธีการใช้งานแอปพลิเคชันบนหน้าจอเพื่อให้ผู้ใช้งานเข้าใจได้ง่ายขึ้น

Hi-res icon: ภาพ Icon ของแอปพลิเคชันขนาด 512 x 512 PX เพื่อใช้แสดงใน Google Play

Feature graphic: ภาพหลักของหน้าดาวน์โหลดที่จะอยู่ข้างบนสุดของหน้าดาวน์โหลดแอปพลิเคชัน เป็นเหมือนภาพหลักที่ใช้แสดงให้ผู้ใช้งานเห็นเมื่อเข้ามาดูรายละเอียดของแอปพลิเคชันก่อนจะกดดาวน์โหลด โดยจะต้องใช้เป็นภาพขนาด 1024 x 500 PX

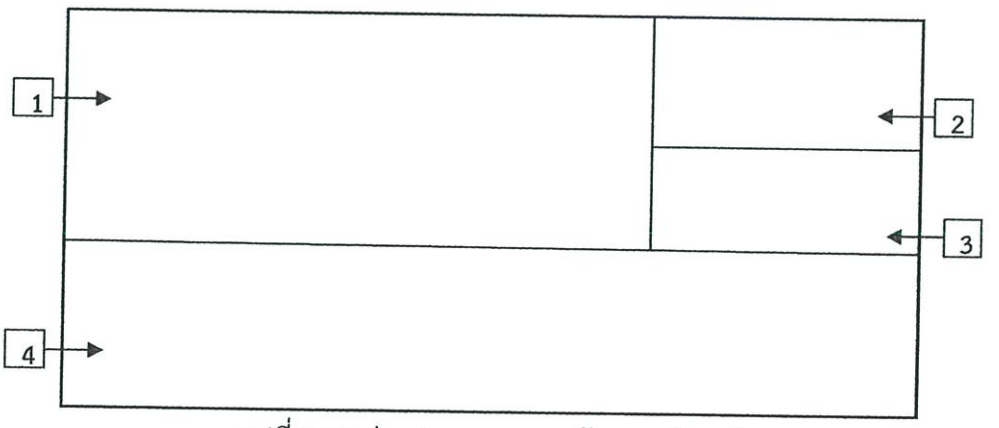
Promo Video: วีดีโอประกอบแอปพลิเคชัน เพื่อให้ผู้ใช้งานเห็นภาพการทำงานของแอปพลิเคชันได้ง่าย



รูปที่ 3.23 การออกแบบ Graphic assets ใน Google Play

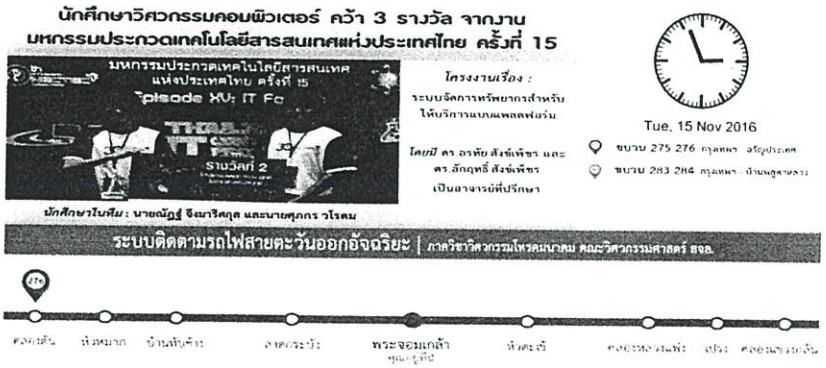
3.1.3 การออกแบบหน้าจอมอนิเตอร์

หน้าจอทีวีมอนิเตอร์ใช้สำหรับแสดงตำแหน่งของรถไฟซึ่งถูกติดตั้งภายในสถาบันโดย หน้าจอมอนิเตอร์ประกอบไปด้วย 4 ส่วนหลักๆ ดังรูปที่ 3.24 ซึ่งทีวีมอนิเตอร์นั้นจะแสดงข้อมูลเป็น ภาษาไทยและภาษาอังกฤษซึ่งสลับกันทุกๆ 10 วินาที ดังรูปที่ 3.25 และรูปที่ 3.26 ตามลำดับ

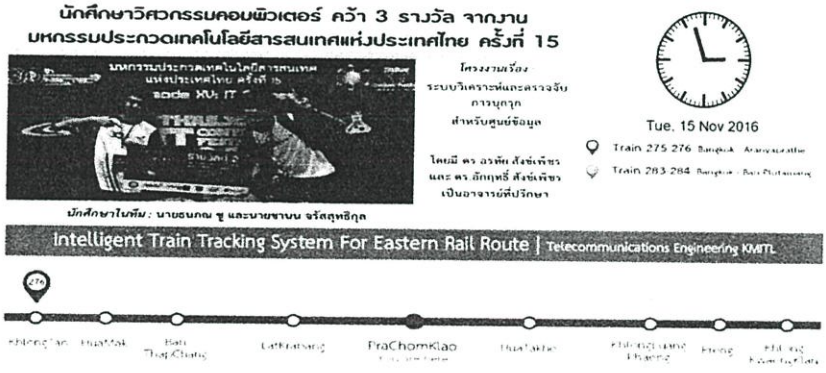


รูปที่ 3.24 ส่วนประกอบของหน้าจอมอนิเตอร์

หมายเลข 1 เป็นส่วนแสดงข่าวสารภายในสถาบัน
หมายเลข 2 เป็นส่วนแสดงเวลาและวันที่ปัจจุบัน
หมายเลข 3 เป็นส่วนแสดงข้อมูลของขบวนรถไฟที่ติดตั้งระบบติดตามรถไฟ
หมายเลข 4 เป็นส่วนแสดงตำแหน่งรถไฟในสถานีใกล้เคียงสถานีพระจอมเกล้า

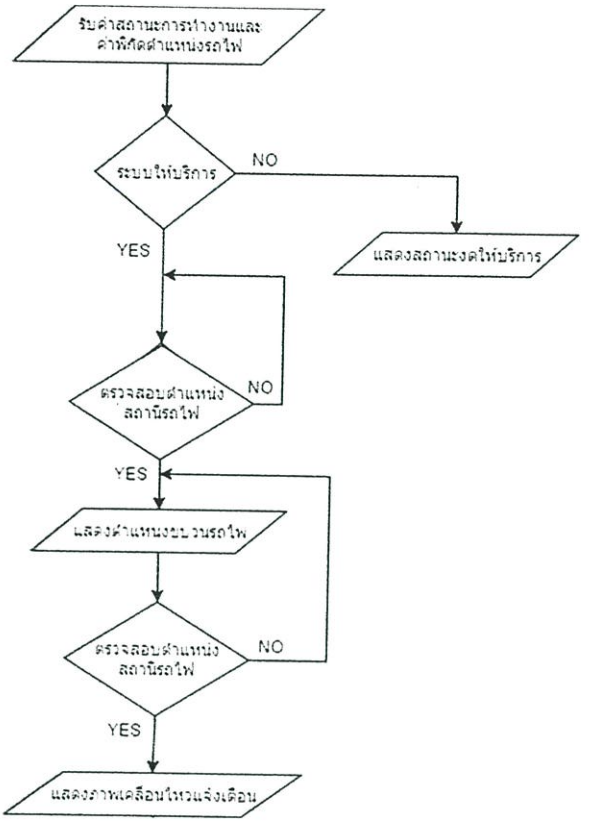


รูปที่ 3.25 หน้าจอมอนิเตอร์ภาษาไทย



รูปที่ 3.26 หน้าจอมอนิเตอร์ภาษาอังกฤษ

หน้าแสดงผลของหน้าจอมอนิเตอร์เขียนโดยใช้ภาษา HTML (Hypertext Markup Language) และใช้ภาษา JavaScript ในการเชื่อมต่อกับ PHP เพื่อนำค่าพิกัดตำแหน่งปัจจุบันของขบวนรถไฟมาประมวลผลตามเงื่อนไขที่กำหนด โดยรับค่าพิกัดขบวนรถไฟและค่าสถานะการทำงาน ถ้าระบบติดตามรถไฟให้บริการจะเข้าเงื่อนไขการแสดงตำแหน่งขบวนรถไฟแต่ถ้าระบบติดตามรถไฟไม่ให้บริการจะแสดงข้อความงดให้บริการ ในส่วนของเงื่อนไขการแสดงตำแหน่งขบวนรถไฟถ้าตำแหน่งขบวนรถไฟอยู่ในสถานที่ใกล้ที่สุดจากสถานีพระจอมเกล้าจะแสดงภาพเคลื่อนไหวเพื่อแจ้งเตือนว่ารถไฟกำลังมาถึงสถานี ซึ่งมีแผนผังการทำงานดังรูปที่ 3.27



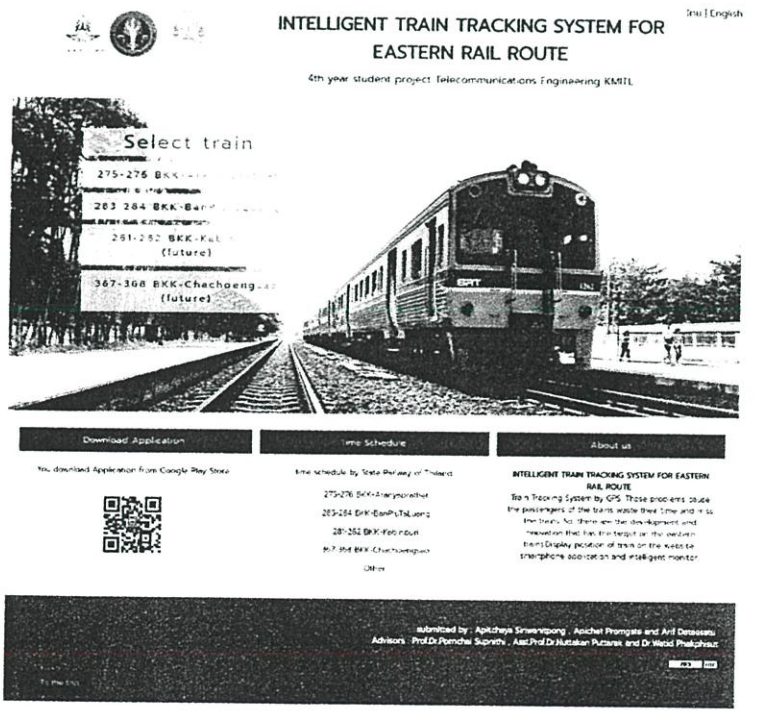
รูปที่ 3.27 แผนผังการทำงานหน้าจอมอนิเตอร์

3.1.3 การออกแบบเว็บไซต์

ในโครงการนี้มีการออกแบบหน้าเว็บไซต์แสดงผลเพื่อให้ใช้งานง่ายขึ้นและมีความสวยงามมากขึ้นจากเดิมโดยหน้าเว็บไซต์แสดงผลที่ได้ออกแบบใหม่แสดงผลทั้งภาษาไทยและภาษาอังกฤษ ดังรูปที่ 3.28 และ รูปที่ 3.29 โดยมีส่วนประกอบต่างๆ บนเว็บไซต์ดังนี้

1) เมนูเลือกขบวนรถไฟ ซึ่งเป็นส่วนหลักเพื่อแสดงตำแหน่งของรถไฟ
2) ส่วนดาวน์โหลดแอปพลิเคชัน
3) ส่วนตารางเวลาของรถไฟของการรถไฟแห่งประเทศไทย
4) ส่วนเกี่ยวกับเรา แสดงข้อมูลที่มาและความสำคัญของโครงงาน

รูปที่ 3.28 หน้าเว็บไซต์ใหม่ภาษาไทย

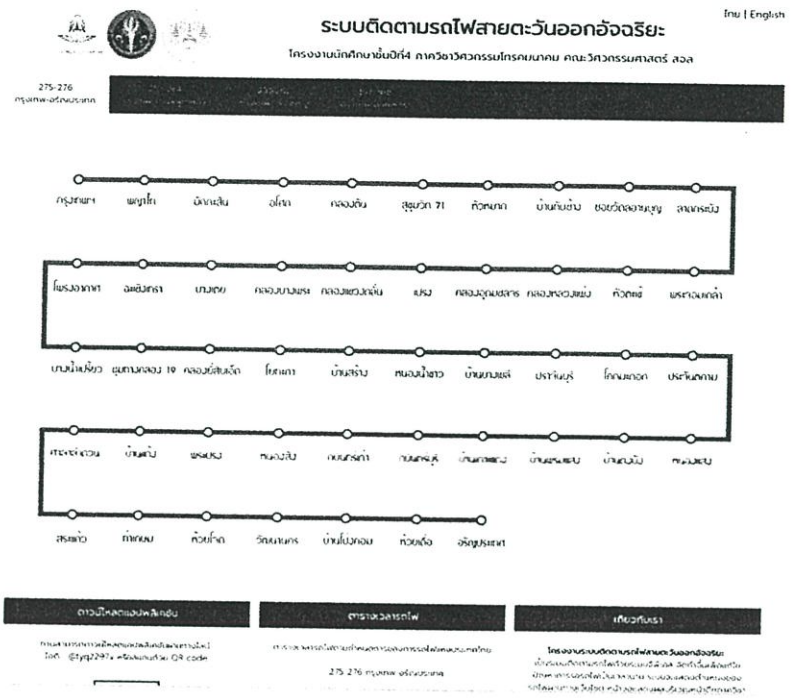


รูปที่ 3.29 หน้าเว็บไซต์ใหม่ภาษาอังกฤษ

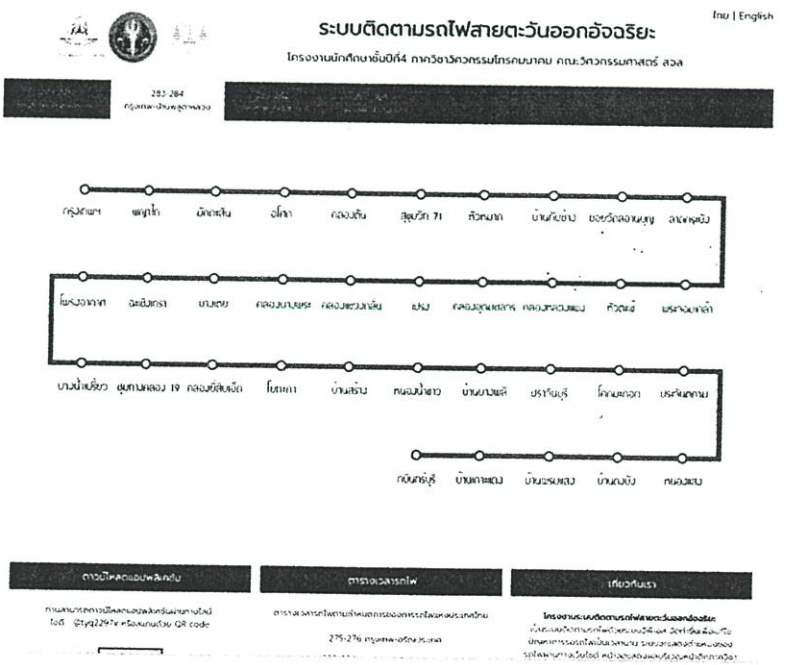
การออกแบบส่วนของการแสดงตำแหน่งรถไฟทั้ง 4 ขบวน ได้แก่ ซึ่งโดยจะมีการบอกตำแหน่งปัจจุบันของรถไฟขบวนนั้นๆ โดยเข้าไปที่เมนูเลือกขบวนรถไฟจากส่วนหน้าหลักของหน้าเว็บไซต์แสดงผล ดังรูปที่ 3.30

รูปที่ 3.30 เมนูเลือกขบวนรถไฟในหน้าหลักของหน้าเว็บไซต์แสดงผล

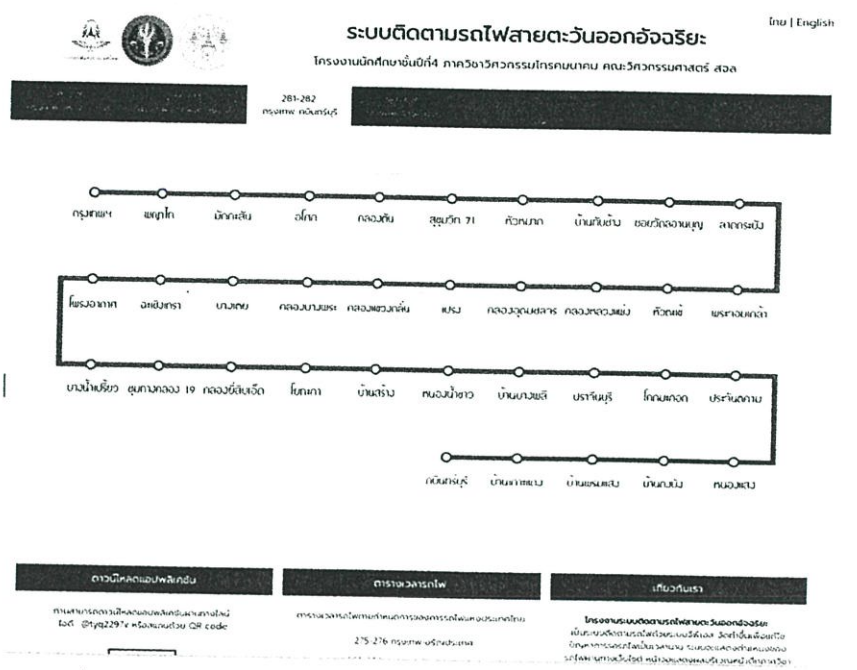
เมื่อผู้ใช้งานเลือกขบวนรถไฟจากเมนูเลือกขบวนรถไฟเว็บไซต์จะเข้าสู่หน้าแสดงตำแหน่งของขบวนรถไฟนั้นๆ ได้แก่ หน้าแสดงตำแหน่งขบวน 275-276 กรุงเทพฯ-อรัญประเทศ หน้าแสดงตำแหน่งขบวน 283-284 กรุงเทพฯ-บ้านพลูตาหลวง หน้าแสดงตำแหน่งขบวน 281-282 กรุงเทพฯ-กบินทร์บุรีและหน้าแสดงตำแหน่งขบวน 367-368 กรุงเทพฯ-ฉะเชิงเทรา ดังรูปที่ 3.31 ถึง 3.34 ตามลำดับ



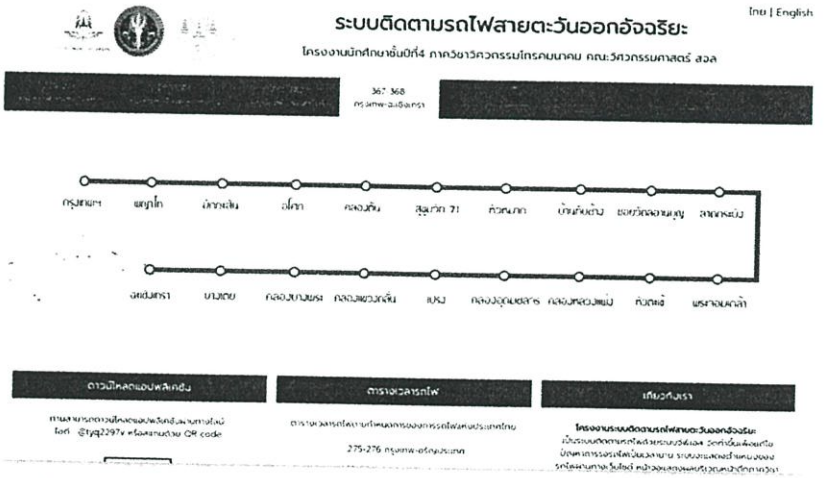
รูปที่ 3.31 หน้าแสดงตำแหน่งขบวน 275-276 กรุงเทพฯ-อรัญประเทศ



รูปที่ 3.32 หน้าแสดงตำแหน่งขบวน 283-284 กรุงเทพฯ-บ้านพลูตาหลวง



รูปที่ 3.33 หน้าแสดงตำแหน่งขบวน 281-282 กรุงเทพฯ-กบินทร์บุรี



รูปที่ 3.34 หน้าแสดงตำแหน่งขบวน ขบวน 367-368 กรุงเทพ-ฉะเชิงเทรา

ซึ่งในแต่ละหน้านั้น สามารถเลือกดูรถไฟสายอื่นๆ ได้โดยตรง โดยไม่จำเป็นต้องกดกลับสู่หน้าหลักดังเช่นเว็บไซต์เดิมโดยการทำงานจะเหมือนกับหน้าจอแสดงผลมอนิเตอร์

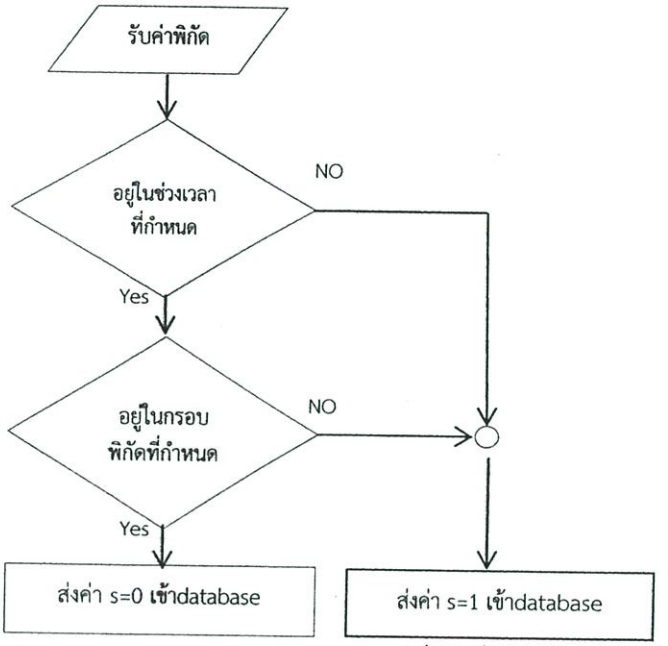
3.1.4 การออกแบบการเปลี่ยนสถานะของการติดตามรถไฟ (s) อัตโนมัติ

เนื่องจากในระบบติดตามรถไฟนั้น ในการใช้งานจริงยังเกิดปัญหาในการตรวจสอบตำแหน่งของรถไฟ อาจมีสาเหตุมาจากปัจจัยหลายๆอย่าง เช่น แบตเตอรี่ของอุปกรณ์ติดตามรถไฟหมด พนักงานประจำรถไฟลืมอุปกรณ์ติดตามรถไฟไว้ที่สถานี ซึ่งทำให้ผู้ใช้งานที่เข้ามาดูตำแหน่งของรถไฟนั้นเกิดความเข้าใจผิด จึงได้มีการกำหนดสถานะ s ขึ้นมาแสดงดังรูปที่ 3.35 โดยถ้าสถานะของ s = 1 หมายถึงระบบติดตามรถไฟทำงานตามปกติ แต่ถ้าสถานะของ s = 0 หมายถึงหยุดให้บริการระบบติดตามรถไฟ

	day	month	year	hour	minute	sec	id	L	n	bat	spd	s	No
	9	12	2016	7	16	36	2	13.723275	100.782546	100	0.61	1	437
	9	12	2016	7	17	1	2	13.723291	100.782623	82	0.93	1	438
	9	12	2016	7	17	24	2	13.723295	100.782783	88	0.31	1	439
	9	12	2016	7	17	49	2	13.723288	100.782798	97	0.41	1	440
	9	12	2016	7	18	13	2	13.723296	100.782829	97	0.17	1	441
	9	12	2016	7	20	14	2	13.723450	100.782875	91	0.96	1	443
	9	12	2016	7	21	1	2	13.723358	100.782722	94	2.33	1	445
	9	12	2016	7	21	52	2	13.723422	100.782814	91	0.74	1	447
	9	12	2016	7	22	23	2	13.723381	100.782768	91	2.02	1	448

รูปที่ 3.35 ข้อมูลที่ส่งจากติดตามรถไฟ

จากนั้นจึงทำการออกแบบและเขียนโปรแกรมภาษา php เพื่อเปลี่ยนสถานะ (s) อัตโนมัติที่ฐานข้อมูลในตอนที่รถไฟเข้าใกล้สถานีสุดท้าย(สถานีหัวลำโพง) เพื่อไม่ให้ผู้ใช้งานเข้าใจผิดในกรณีที่ถ้าวันถัดไป หากรถไฟหยุดให้บริการก็จะมีแสดงสถานะ การงดให้บริการระบบติดตามรถไฟที่ หน้าเว็บไซต์ มอนิเตอร์ และแอปพลิเคชัน เพื่อให้ผู้ใช้บริการได้รู้ แสดงขั้นตอนดังรูปที่ 3.36



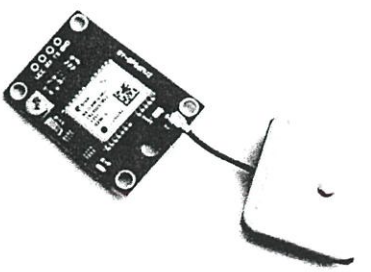
รูปที่ 3.36 การออกแบบและเขียนโปรแกรม เพื่อเปลี่ยนสถานะ (s) อัตโนมัติ

3.2 เครื่องมือที่ใช้ในการทดลอง

3.2.1 GPS Module Ublox NEO 6M

ในส่วนของอุปกรณ์ติดตามรถไฟ ผู้วิจัยได้ใช้อุปกรณ์ในการทำงาน 4 ส่วนด้วยกัน คือ GPS Module Ublox Neo 6M สำหรับรับพิกัดของรถไฟ และใช้ Voltage Sensor ในการวัดค่าความต่างศักย์ของแบตเตอรี่ จากนั้นจะนำข้อมูลที่ได้ไปประมวลผลที่ บอร์ดไมโครคอลโทรลเลอร์ Arduino Uno R3 แล้วส่งข้อมูลดังกล่าวไปที่ฐานข้อมูลโดยใช้ 3G Module ผ่านเครือข่ายโทรศัพท์ 3 จี

GPS Module ดังแสดงในรูปที่ 3.37 จะถูกต่อเข้ากับ บอร์ด Arduino Uno R3 โดยต่อขา Tx เข้ากับพอร์ตดิจิตอลขา 10 ของบอร์ด Arduino Uno R3 และต่อขา Rx เข้ากับพอร์ตดิจิตอลขา 11 ของบอร์ด Arduino Uno R3 และจ่ายกระแสไฟฟ้าให้กับ GPS Module +5 โวลต์



รูปที่ 3.37 GPS Module Ublox NEO 6M [9]

3.2.2 Voltage Sensor

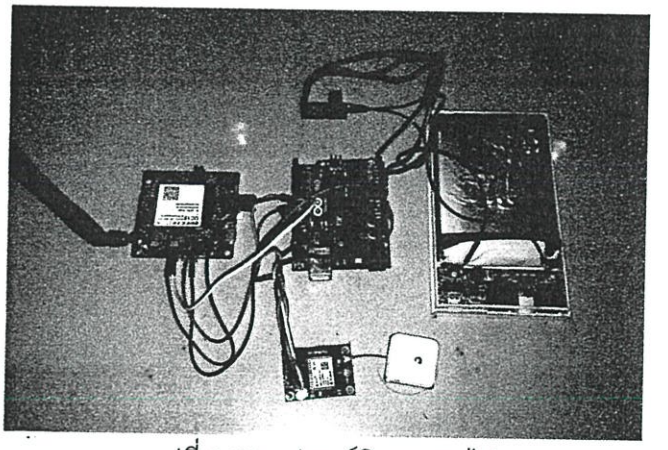
Voltage Sensor จะถูกต่อเข้ากับบอร์ด Arduino Uno R3 โดยให้ขาที่ใช้ส่งข้อมูลของ Voltage Sensor ต่อเข้ากับพอร์ตอนาลอกขา 1 ของบอร์ด Arduino Uno R3 เนื่องจากข้อมูลที่ได้จาก Voltage Sensor เป็นข้อมูลอนาลอก แล้วจ่ายกระแสไฟฟ้าให้กับ Voltage Sensor +5 โวลต์ดังรูปที่ 3.38

3.2.3 บอร์ด Arduino Uno R3

เป็นตัวรับข้อมูลของพิกัดของรถไฟ ความเร็ว และค่าความต่างศักย์ของแบตเตอรี่ เพื่อนำมาประมวลผล และใช้ควบคุมการทำงานของอุปกรณ์ติดตามรถไฟเพื่อที่จะส่งข้อมูลเข้าฐานข้อมูล โดยบอร์ด Arduino Uno R3 รองรับการจ่ายกระแสไฟฟ้าได้ 6-20 โวลต์ ดังรูปที่ 3.38

3.2.4 3G Module

3G Module จะถูกต่อเข้ากับ บอร์ด Arduino Uno R3 เพื่อที่จะส่งข้อมูลเข้าฐานข้อมูล ผ่านเครือข่าย 3 จี โดยต่อขา Tx เข้ากับ พอร์ตดิจิตอลขา 8 ของบอร์ด Arduino Uno R3 และต่อขา Rx เข้ากับพอร์ตดิจิตอลขา 10 ของบอร์ด Arduino Uno R3 เพื่อรับและส่งข้อมูล จากนั้นจ่ายกระแสไฟฟ้าให้กับ 3G Module 5-12 โวลต์ ดังรูปที่ 3.38



รูปที่ 3.38 อุปกรณ์ติดตามรถไฟ

3.3 การจัดเก็บผลการทดลอง

การจัดเก็บผลการทดลองแบ่งออกเป็น 4 ส่วน คือ

1) การจัดเก็บผลการทดลองของระบบติดตามรถไฟจะจัดเก็บพิกัดรถไฟ เมื่อเครื่องส่งพิกัดส่งพิกัดของรถไฟ ณ ขณะนั้นเข้ามายังฐานข้อมูลดังนี้

1.1) เมื่อนำเครื่องส่งพิกัดอยู่บนขบวนรถไฟแล้วทำการเปิดระบบ ระบบจะทำงานโดยการส่งพิกัด ณ ตำแหน่งนั้นๆ ของเครื่องส่งมายังฐานข้อมูล แยกตามหมายเลขขบวน

1.2) ส่วนของการแสดงผล คือ จอแสดงผล และแอปพลิเคชัน แต่ละหน่วยจะทำการดึงค่าตัวแปรที่ได้จากฐานข้อมูล เพื่อนำไปแสดงผลต่อผู้ใช้ได้รับทราบตำแหน่งของรถไฟ

2) การเก็บผลการทดลองจากการวัดความต่างศักย์ของแบตเตอรี่ เพื่อที่จะนำค่าที่ได้ไปคำนวณหาสมการอัตราการลดลงเปอร์เซ็นต์ของแบตเตอรี่

3) ตรวจสอบการส่งข้อมูลโดยใช้โหมดประหยัดพลังงาน

4) ตรวจสอบการเปลี่ยนสถานะของระบบติดตามรถไฟ (s) ว่าเปลี่ยนสถานะอัตโนมัติหรือไม่เมื่อถึงสถานีสุดท้าย

บทที่ 4
ผลการทดลอง

4.1 การทดสอบแอปพลิเคชันบนโทรศัพท์มือถือระบบแอนดรอยด์

การทดสอบการทำงานแอปพลิเคชันบนโทรศัพท์มือถือระบบแอนดรอยด์ โดยขั้นตอนแรก ผู้วิจัยจะติดตั้งแอปพลิเคชัน ที่ถูกพัฒนาจากโปรแกรมแอนดรอยด์สตูดิโอ (Android Studio) ลงบนโทรศัพท์มือถือสมาร์ตโฟน ซึ่งมีไอคอนของแอปพลิเคชันดังรูปที่ 4.1

รูปที่ 4.1 ไอคอนของแอปพลิเคชัน

เมื่อกดเข้าสู่แอปพลิเคชันจะแสดงหน้าหลักของแอปพลิเคชัน โดยหน้าหลักจะประกอบไปด้วย 2 เมนู คือเมนูเลือกขบวนรถไฟ และเลือกสถานีรถไฟ ตามลำดับ ดังรูปที่ 4.2

รูปที่ 4.2 หน้าหลักของแอปพลิเคชัน

ในกรณีที่ผู้ใช้งานเปิดใช้งานแอปพลิเคชัน ในขณะที่ไม่ได้ทำการเชื่อมต่ออินเตอร์เน็ต แอปพลิเคชันจะทำการแจ้งเตือนเป็นข้อความ เพื่อให้ผู้ใช้งานทำการเปิดอินเตอร์เน็ต ดังรูปที่ 4.3

รูปที่ 4.3 ระบบแจ้งเตือนผู้ใช้งาน เมื่ออินเตอร์เน็ตไม่ได้ถูกเปิด

4.1.1 การทดสอบแอปพลิเคชันเมื่อทำกดเลือกขบวนรถไฟในหน้าหลัก

เมื่อผู้ใช้งานทำการต่ออินเตอร์เน็ตเรียบร้อยแล้ว หากผู้ใช้งานกดเลือกเมนูขบวนรถไฟในหน้าหลัก แอปพลิเคชันจะแสดงขบวนรถไฟในสายต่างๆ ดังรูปที่ 4.4

รูปที่ 4.4 หน้าเมนูขบวนรถไฟ

เมื่อผู้ใช้งานกดเลือกขบวนรถไฟในสายใดๆ ก็ตาม แอปพลิเคชันจะแสดง ตำแหน่งสถานีปัจจุบัน และสถานีถัดไปของรถไฟขบวนนั้นๆ ดังรูปที่ 4.5

รูปที่ 4.5 หน้าแสดงผลตำแหน่งของรถไฟ

4.1.2 การทดสอบแอปพลิเคชันเมื่อทำกดเลือกสถานีไฟในหน้าหลัก

อย่างไรก็ตาม หากผู้ใช้งานกดเลือกเมนูสถานีรถไฟในหน้าหลัก แอปพลิเคชันจะแสดงเมนูสถานีรถไฟที่ใกล้ตัวผู้ใช้งานมากที่สุด และเมนูรายการสถานีรถไฟอื่นๆ ดังรูปที่ 4.6

รูปที่ 4.6 หน้าเมนูสถานีรถไฟที่ใกล้ตัวผู้ใช้งานที่สุดและเมนูรายการสถานีอื่นๆ

เมื่อผู้ใช้งานเลือกเมนูสถานีรถไฟที่ใกล้ตัวผู้ใช้งานมากที่สุด แอปพลิเคชันจะมีการตรวจสอบ GPS Service หากผู้ใช้งานไม่ได้เปิด GPS Services ระบบจะทำการแจ้งเตือนให้เปิด GPS Services ดังแสดงในรูปที่ 4.7

รูปที่ 4.7 แจ้งเตือนเมื่อผู้ใช้งานไม่ได้เปิด GPS Service

รูปที่ 4.8 หน้าเมนูรายการสถานีรถไฟที่ใกล้ตัวที่สุด

เมื่อผู้ใช้งานดำเนินการเชื่อมต่อ GPS เรียบร้อย แอปพลิเคชันจะแสดงสถานีที่ใกล้ผู้ใช้งานมากที่สุด 3 สถานี ดังรูปที่ 4.8 จากนั้น เมื่อผู้ใช้งานทำการเลือกสถานีใดๆก็ตาม แอปพลิเคชันจะแสดงขบวนรถไฟที่กำลังเข้าสถานีรถไฟที่ผู้ใช้งานเลือก ดังแสดงในรูปที่ 4.9

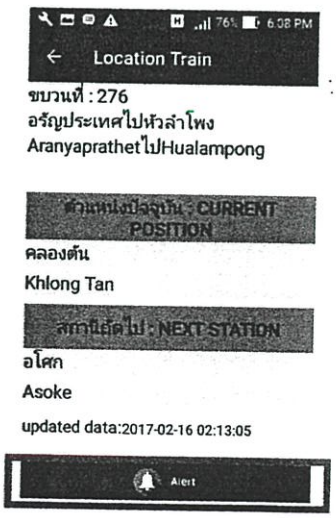
รูปที่ 4.9 หน้าแสดงขบวนรถไฟที่กำลังเข้าสถานีรถไฟ

กรณีที่ไม่มีขบวนรถไฟเข้าสู่สถานีรถไฟที่ผู้ใช้งานเลือก แอปพลิเคชันจะแสดงดังรูปที่ 4.10

รูปที่ 4.10 หน้าแสดงไม่มีขบวนรถไฟที่กำลังเข้าสถานีรถไฟ

4.1.3 การทดสอบแอปพลิเคชันเมื่อทำเรียกใช้ระบบการแจ้งเตือน

สำหรับการทดสอบระบบการแจ้งตือนของแอปพลิเคชันมีหลักการทำงานคือ ระบบจะทำการแจ้งเตือนเมื่อขบวนรถไฟกำลังเข้าสู่สถานีที่ผู้ใช้งานต้องการ โดยผู้ใช้งานสามารถเข้าไปยังหน้าแสดงตำแหน่งของรถไฟ ซึ่งจะมีเมนู Alert ปรากฏอยู่ ดังรูปที่ 4.11



รูปที่ 4.11 หน้าแสดงตำแหน่งของรถไฟ

หลังจากผู้ใช้งานกดปุ่ม Alert แอปพลิเคชันจะแสดงรายการสถานีรถไฟต่างๆ เพื่อให้ผู้ใช้งานสามารถเลือกสถานี ที่ต้องการให้แอปพลิเคชันทำการแจ้งเตือน ดังรูปที่ 4.12

รูปที่ 4.12 รายการสถานีที่ต้องการให้แอปพลิเคชันแจ้งเตือน

เมื่อรถไฟกำลังเข้าสู่สถานีที่ผู้ใช้งานเลือก (ผู้พัฒนาได้ทำการออกแบบแอปพลิชันให้แจ้งเตือน 2 สถานีก่อนเข้าสถานีที่ผู้ใช้งานเลือก) แอปพลิเคชันจะแสดงการแจ้งเตือนเป็นข้อความและเสียงเตือน ดังรูปที่ 4.13

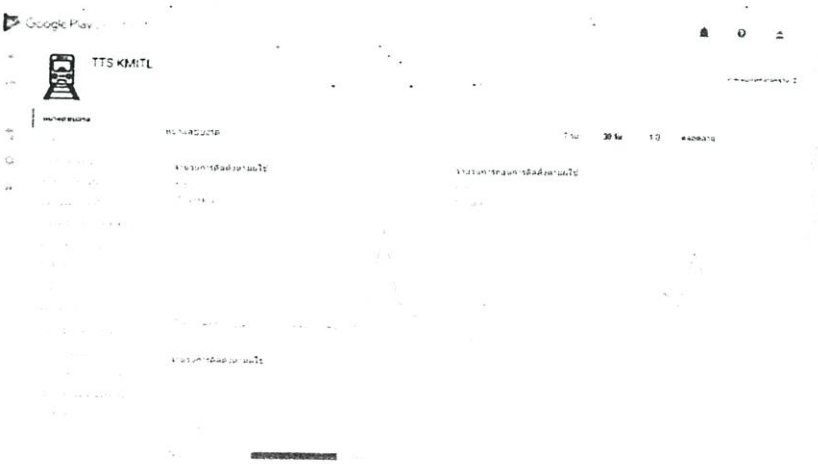
รูปที่ 4.13 หน้าแสดงการแจ้งเตือนของแอปพลิเคชัน

4.2 การทดสอบการติดตั้งแอปพลิเคชันบน Google Play

สำหรับการติดตั้งแอปพลิเคชันบน Google Play นั้น เมื่อผู้ใช้งานค้นหาแอปพลิเคชัน “ระบบติดตามรถไฟสายตะวันออกอัจฉริยะ” ผ่าน Google Play Store โดยค้นหาคำว่า “TTS KMITL” ซึ่งจะแสดงแอปพลิเคชัน TTS KMITL จะปรากฏขึ้นมา ดังรูปที่ 4.14

รูปที่ 4.14 แอปพลิเคชันบน Google Play Store

โดยที่ผู้พัฒนาสามารถติดตามความเคลื่อนไหวของแอปพลิเคชันบนเว็บไซต์ Google Play Developer Console โดยจะมีเมนูหน้าแดชบอร์ด แสดงจำนวนการติดตั้ง และจำนวนการถอนการติดตั้ง นอกจากนั้น ยังมี สถิต, การสร้างฐานผู้ใช้, คะแนน และความคิดเห็น, ข้อขัดข้องและ ANR เป็นต้น ดังรูปที่ 4.15



รูปที่ 4.15 Google Play Developer Console

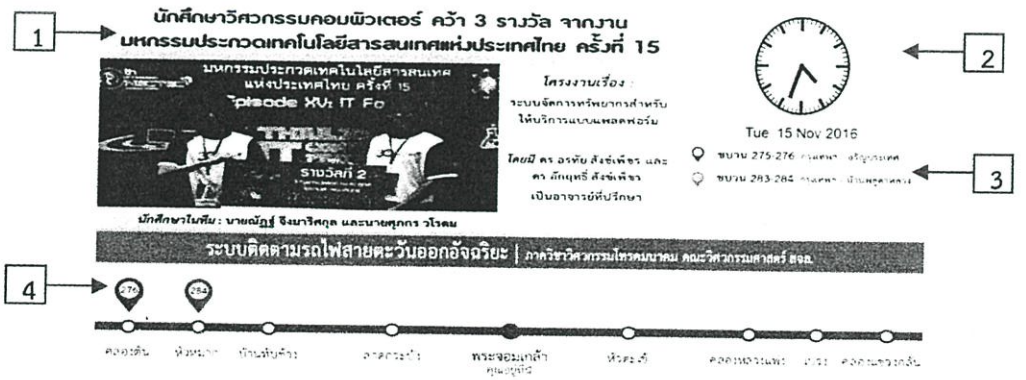
4.3 การทดสอบหน้าจอมอนิเตอร์

หน้าจอมอนิเตอร์จะรับค่าพิกัดจากฐานข้อมูล โดยผู้พัฒนาได้เขียนคำสั่ง PHP เพื่อนำข้อมูลที่สำคัญ เช่น ค่า day id L n bat spd No และ s ซึ่งหมายถึง วันเวลา ลำดับข้อมูล ค่าละติจูด ลองจิจูด ระดับแบตเตอรี่ ความเร็วขบวนรถไฟ เลขระบุขบวน และสถานะของระบบติดตามรถไฟตามลำดับ ดังรูปที่ 4.16

	day	id	L	n	bat	spd	No	s
	2016-11-15 18:36:06	2	13.741593	100.517410	100	1.09	32792	1
	2016-11-15 18:36:24	2	13.741469	100.517349	100	2.83	32793	1
	2016-11-15 18:36:42	2	13.741259	100.517311	100	3.46	32794	1
	2016-11-15 18:37:00	2	13.741073	100.517280	100	3.50	32795	1
	2016-11-15 18:37:18	2	13.740938	100.517280	100	1.72	32796	1
	2016-11-15 18:37:36	2	13.740814	100.517311	100	3.20	32797	1
	2016-11-15 18:37:55	2	13.740674	100.517311	100	2.17	32798	1
	2016-11-15 18:38:13	2	13.740406	100.517158	100	3.17	32799	1
	2016-11-15 18:38:30	2	13.740184	100.517028	100	1.26	32800	1
	2016-11-15 18:38:48	2	13.740329	100.517127	100	7.00	32801	1
	2016-11-15 18:41:24	2	13.739312	100.516807	100	12.20	32802	1

รูปที่ 4.16 ตารางในฐานข้อมูล

โดยหน้าจอมอนิเตอร์ประกอบไปด้วย 4 ส่วนต่างๆ ดังรูปที่ 4.17



รูปที่ 4.17 ส่วนต่างๆของหน้าจอมอนิเตอร์

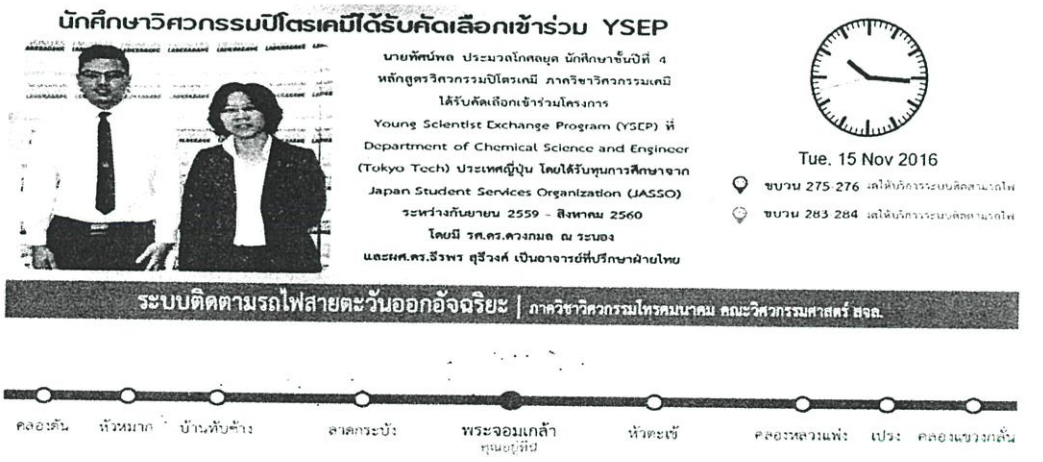
1) ส่วนที่ 1 แสดงข่าวสารภายในสถาบัน
2) ส่วนที่ 2 แสดงวันที่ และเวลาปัจจุบัน
3) ส่วนที่ 3 แสดง เส้นทางรถที่ได้ทำการติดตาม ประกอบไปด้วย
   - ขบวน 275-276 กรุงเทพฯ-อรัญประเทศ
   - ขบวน 283-284 กรุงเทพฯ-บ้านพลูตาหลวง

ในอนาคตอีก 2 เส้นทาง (แสดงเป็นสีเทา) ดังนี้

- ขบวน 281-282 กรุงเทพฯ-กบินทร์บุรี
- ขบวน 367-368 กรุงเทพฯ-ชุมทางฉะเชิงเทรา

4) ส่วนที่ 4 แสดงตำแหน่งของรถไฟ ซึ่งระบบจะแสดงเฉพาะสถานีใกล้เคียงสถานีพระจอมเกล้า (สถานีหน้าสถาบัน) มากที่สุดจำนวน 9 สถานี ซึ่งประกอบไปด้วย สถานีคลองตัน สถานีหัวหมาก สถานีบ้านทับช้าง สถานีลาดกระบัง สถานีพระจอมเกล้า สถานีหัวตะเข้ สถานีคลองหลวงแพ่ง สถานีเปรง และสถานีคลองแขวงกลั่น โดยตำแหน่งรถไฟนั้น จะแสดงเป็นรูปหมุด ที่อยู่บนแผนที่เส้นทาง ซึ่งจะปรากฏเลขขบวนรถไฟ และทิศทางของรถไฟด้วย อย่างไรก็ตาม รูปหมุดบนแผนที่จะแสดงถี่ขึ้น เมื่อรถไฟอยู่ระหว่างสถานีบ้านทับช้างกับสถานีคลองหลวงแพ่ง

เงื่อนไขในการทำงานของหน้าจอมอนิเตอร์คือ เมื่อหน้าจอมอนิเตอร์รับค่า สถานะของระบบติดตามรถไฟ (s) ที่มีค่าเป็น 1 มอนิเตอร์จะแสดงตำแหน่งรถไฟปกติ ในทางตรงกันข้าม หากมอนิเตอร์รับค่า s ที่มีค่าเป็น 0 มอนิตอร์จะแสดงข้อความแจ้งเตือนว่า “ระบบติดตามรถไฟงดให้บริการ” ดังแสเงในรูปที่ 4.18



รูปที่ 4.18 หน้าจอมอนิเตอร์เมื่อค่า s มีค่าเป็น 0

นอกจากนั้น ผู้พัฒนาได้ออกแบบหน้าจอมอนิเตอร์ ให้ส่วนของโฆษณา (ส่วนที่1) ทำการเปลี่ยนเป็นภาพเคลื่อนไหวว่า “TRAIN IS COMING” เมื่อขบวนรถไฟกำลังเข้าสู่สถานีพระจอมเกล้า (รถไฟอยู่ระหว่างสถานีลาดกระบังและหัวตะเข้) ดังแสดงในรูปที่ 4.19

รูปที่ 4.19 หน้าจอมอนิเตอร์แสดงภาพเคลื่อนไหวแจ้งเตือน

4.4 การทดสอบการทำงานของเว็บไซต์

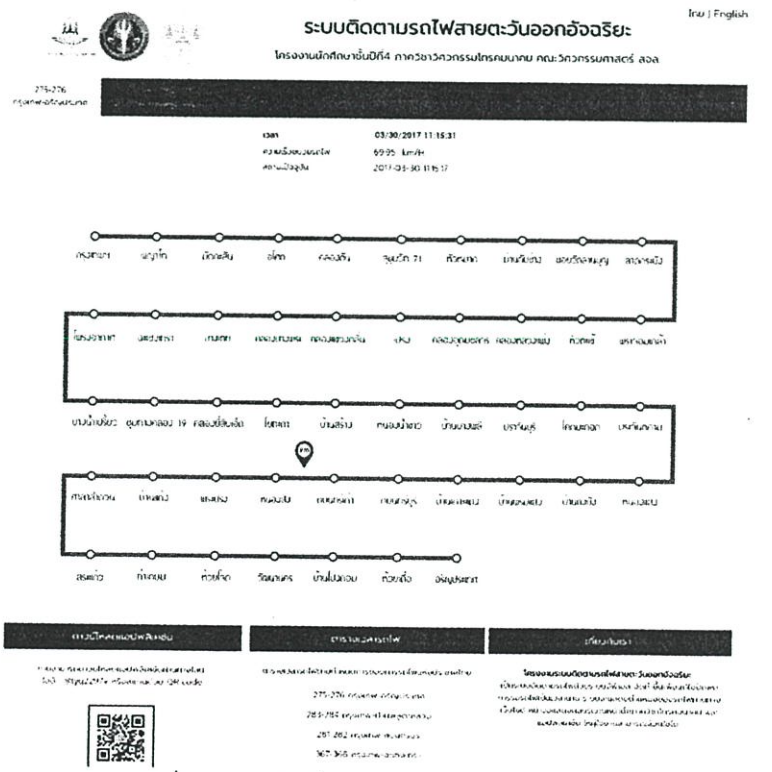
การทำงานของเว็บไซต์จะใช้วิธีการรับค่าต่างๆ คล้ายกับหน้าจอมอนิเตอร์ เพียงแต่จะแสดงผลต่างกัน โดยหน้าเว็บไซต์จะแบ่งเป็น 4 ส่วน ดังรูปที่ 4.20

รูปที่ 4.20 หน้าเว็บไซต์

1) เมนูเลือกขบวนรถไฟ สำหรับแสดงตำแหน่งของรถไฟแต่ละขบวน ซึ่งประกอบไปด้วย 4 ขบวนหลักๆ ได้แก่
- ขบวน 275-276 กรุงเทพ-อรัญประเทศ
- ขบวน 283-284 กรุงเทพ-บ้านพลูตาหลวง
  และ 2 ขบวนในอนาคต
- ขบวน 281-282 กรุงเทพ-กบินทร์บุรี
- ขบวน 367-368 กรุงเทพ-ฉะเชิงเทรา

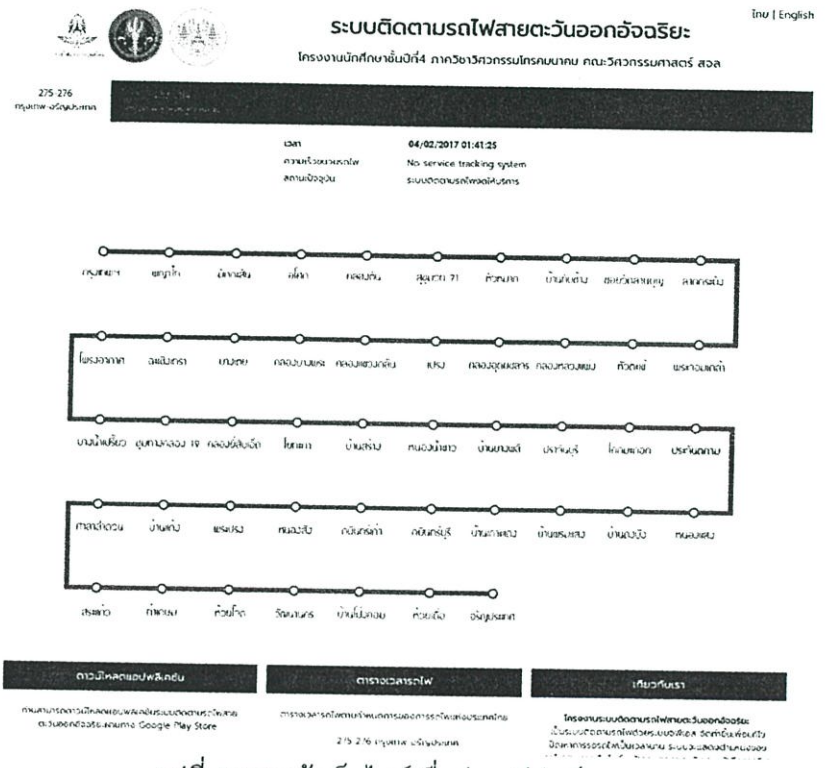
2) สำหรับดาวน์โหลดแอปพลิเคชันที่ออกแบบในหัวข้อ 4.1 (เฉพาะมือถือระบบปฏิบัติการแอนดรอยด์)
3) ส่วนตารางเวลาของรถไฟขบวนต่างๆ ของการรถไฟแห่งประเทศไทย
4) ส่วนแสดงข้อมูลที่มา และความสำคัญของโครงงาน

เมื่อผู้ใช้งานต้องการทราบตำแหน่งรถไฟขบวนต่างๆ ผ่านเว็บไซต์ ผู้ใช้งานสามารถใช้งานในส่วนของเมนูเลือกขบวนรถไฟ (ส่วนที่ 1) ซึ่งภายหลังจากเลือกขบวนรถไฟแล้วนั้น เว็บไซต์จะแสดงตำแหน่งขบวนรถไฟ (หมุดสีฟ้า) เวลาปัจจุบัน ความเร็วขบวนรถไฟ และสถานะปัจจุบันของรถไฟ ดังรูปที่ 4.21



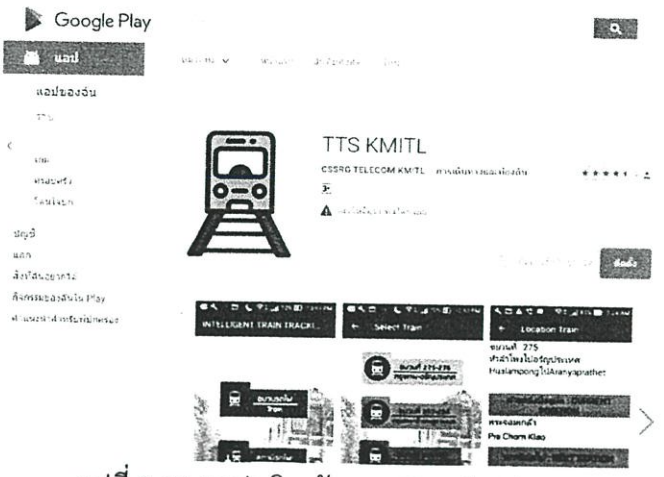
รูปที่ 4.21 หน้าเว็บไซต์แสดงตำแหน่งรถไฟ

เช่นเดียวกันกับมอนิเตอร์ ผู้พัฒนาได้ทำการกำหนดเงื่อนไขในการทำงานของเว็บไซต์ คือ เมื่อเว็บไซต์รับค่า s ที่มีค่าเป็น 1 เว็บไซต์จะแสดงตำแหน่งรถไฟปกติ ในทางตรงกันข้าม หากเว็บไซต์รับค่า s ที่มีค่าเป็น 0 เว็บไซต์จะแสดงข้อความ “ระบบติดตามรถไฟงดให้บริการ” ดังแสดงในรูปที่ 4.22



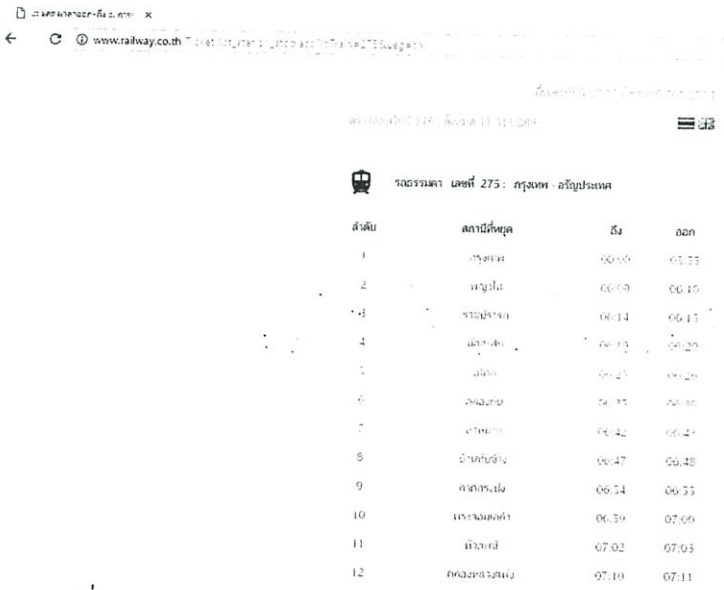
รูปที่ 4.22 หน้าเว็บไซต์เมื่อค่า s มีค่าเป็น 0

ในส่วนดาวน์โหลดแอปพลิเคชัน (ส่วนที่2) ผู้ใช้งานสามารถสแกน QR code หรือเลือกเพื่อดาวน์โหลดแอปพลิเคชันระบบติดตามรถไฟสายตะวันออกอัจฉริยะจาก Google Play Store ดังรูปที่ 4.23



รูปที่ 4.23 แอปพลิเคชันจาก Google Play Store

สำหรับส่วนตารางเวลารถไฟ (ส่วนที่3) ผู้ใช้งานสามารถเลือกขบวนรถไฟที่ต้องการทราบตารางเวลา ตามกำหนดการของการรถไฟแห่งประเทศไทยจากเว็บไซต์การรถไฟแห่งประเทศไทย ดังรูปที่ 4.24

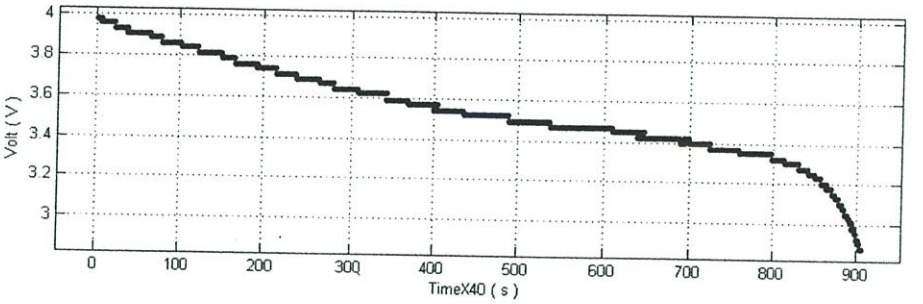


รูปที่ 4.24 ตารางเวลาตามกำหนดการของการรถไฟแห่งประเทศไทย

4.5 การวัดความต่างศักย์ของแบตเตอรี่

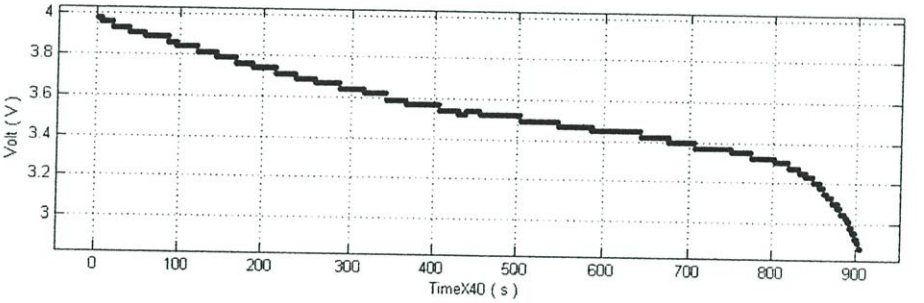
ในส่วนนี้ เป็นการทดลอง และวัดค่าความต่างศักย์ของแบตเตอรี่ เพื่อวิเคราะห์การลดลงของแบตเตอรี่ของอุปกรณ์ที่ซึ่งถูกติดตั้งบนรถไฟ ตั้งแต่แบตเตอรี่เต็มจนแบตเตอรี่หมด โดยทำการทดสอบทั้งหมด 3 ครั้ง และใช้โปรแกรม MATLAB ในการพล็อตกราฟ เพื่อแสดงผลการทดลอง

การทดสอบครั้งที่ 1 แสดงดังรูปที่ 4.25



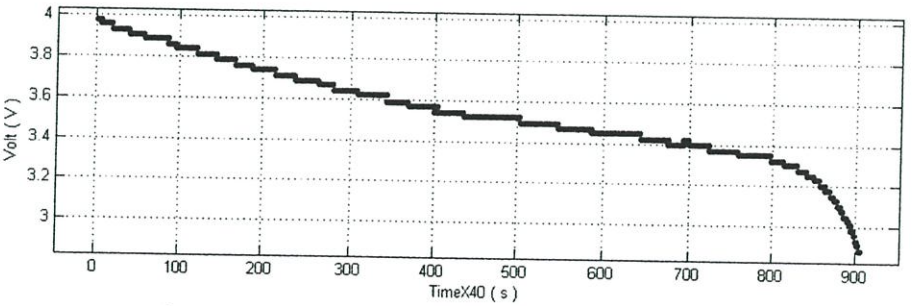
รูปที่ 4.25 การทดสอบวัดความต่างศักย์ของแบตเตอรี่ครั้งที่ 1

การทดสอบครั้งที่ 2 แสดงดังรูปที่ 4.26



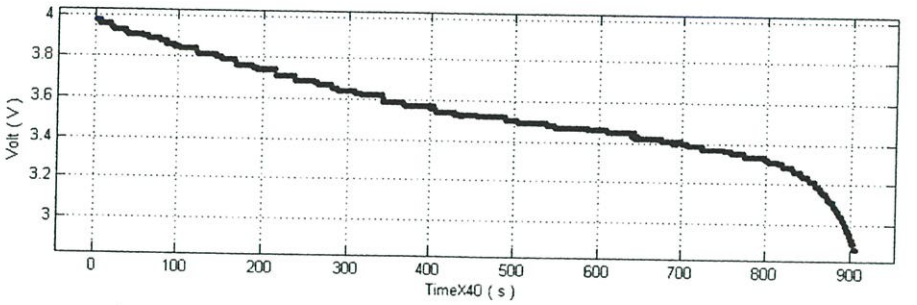
รูปที่ 4.26 การทดสอบวัดความต่างศักย์ของแบตเตอรี่ครั้งที่ 2

การทดสอบครั้งที่ 3 แสดงดังรูปที่ 4.27



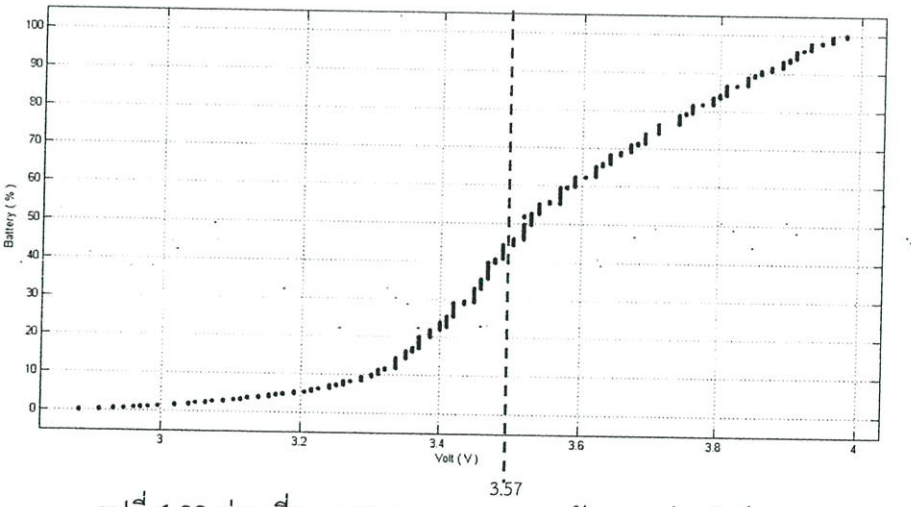
รูปที่ 4.27 การทดสอบวัดความต่างศักย์ของแบตเตอรี่ครั้งที่ 3

จากกราฟในรูปที่ 4.25 – 4.27 แสดงการลดลงของแบตเตอรี่ (แกนy) เทียบกับเวลาที่ใช้ไปในหน่วยวินาทีคูณ 40 (แกนx) จากนั้น นำข้อมูลทั้งหมดที่ทำการทดสอบ มาหาค่าเฉลี่ยดังแสดงดังรูปที่ 4.28



รูปที่ 4.28 ค่าเฉลี่ยจากการทดสอบจากการวัดความต่างศักย์ทั้ง 3 ครั้ง

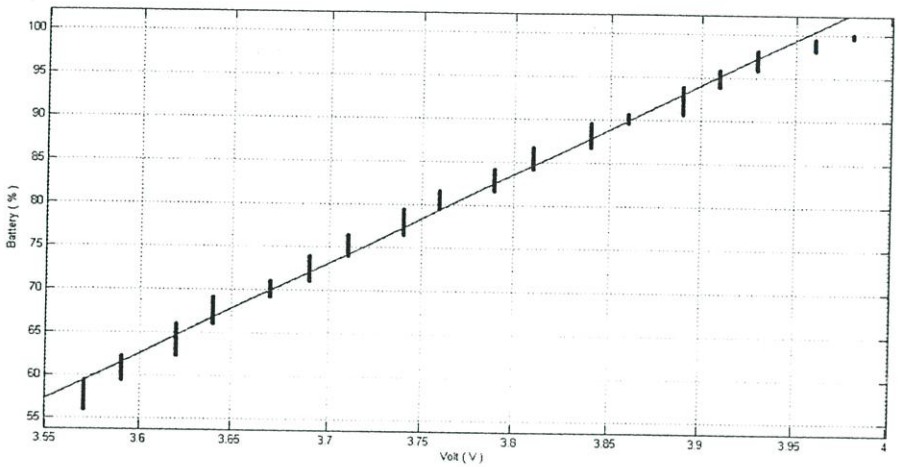
จากรูปที่ 4.28 จะเห็นว่า แบตเตอรี่จะหมดลงที่เวลาประมาณ 900 * 40 วินาที หรือประมาณ 10 ชั่วโมง จากนั้น นำข้อมูลที่ได้ในรูปที่ 4.28 ไปทำ curve fitting ในโปรแกรม MATLAB โดยกำหนดแกนนอนให้เป็นค่าความต่างศักย์ของแบตเตอรี่ภายหลังจากการเฉลี่ย จากนั้น จะทำการแปลงเวลาที่ใช้ทดสอบภายหลังจากการเฉลี่ย ให้เป็นเปอร์เซ็นของแบตเตอรี่คงเหลือ และกำหนดให้เป็นแกนตั้ง ตามลำดับ เพื่อหาสมการอัตราการลดลงของแบตเตอรี่ แสดงได้ดังรูปที่ 4.29



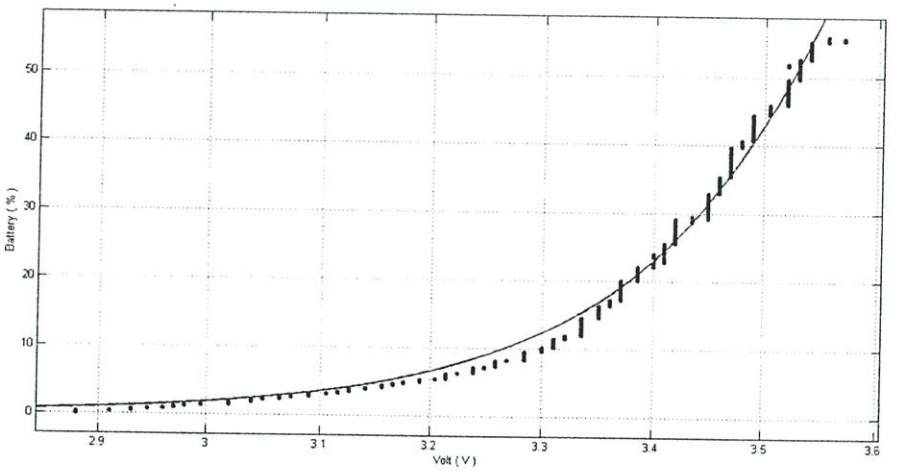
รูปที่ 4.29 ค่าเฉลี่ยจากการทดสอบจากการวัดความต่างศักย์และเวลา

หากเราพิจารณาจากรูปที่ 4.29 จะพบว่าแน้วโน้มของเส้นกราฟในแกนนอน (ค่าความต่างศักย์) ที่มีค่ามากกว่า 3.57 โวลต์ มีลักษณะเป็นเส้นตรง ในทางตรงกันข้ามค่าความต่างศักย์ที่น้อยกว่า 3.57 โวลต์ มีแนวโน้มเส้นกราฟแบบเอกซ์โพเนนเชียล (Exponential) ดังนั้น เราจะ

แบ่งกราฟในรูปที่ 4.29 ออกเป็น 2 ส่วน คือ กราฟที่พิจารณาเฉพาะส่วนของเส้นตรง ดังรูปที่ 4.30 และกราฟที่พิจารณาเฉพาะส่วนของ Exponential ดังรูปที่ 4.31



รูปที่ 4.30 การทำ curve fitting ในส่วนของช่วงที่เป็นเส้นตรง



รูปที่ 4.31 การทำ curve fitting ในส่วนของ Exponential

ภายหลังจากการทำ curve fitting พบว่าหากค่าความต่างศักย์มีค่ามากกว่า 3.57 โวลต์ เราสามารถใช้สมการเส้นตรงในการประมาณค่าเปอร์เซ็นต์ของแบตเตอรี่ได้ดังสมการที่ 4.1

$$Y = 105.2X - 316.1 \tag{4.1}$$

โดยที่ Y คือ ค่าเปอร์เซนต์ที่เหลืออยู่ของแบตเตอรี่
X คือ ค่าความต่างศักย์ที่วัดได้จากแบตเตอรี่

ในทางตรงกันข้าม หากมีค่าความต่างศักย์น้อยกว่า 3.57 โวลต์ สามารถใช้สมการ Exponential ในการประมาณค่าเปอร์เซ็นต์ของแบตเตอรี่ได้ดังสมการการที่ 4.2

$$Y = 1.415 \times 10^{-8} e^{6.237X} \tag{4.2}$$

โดยที่ Y คือ ค่าเปอร์เซนต์ที่เหลืออยู่ของแบตเตอรี่
X คือ ค่าความต่างศักย์ที่วัดได้จากแบตเตอรี่

จากนั้น นำสมการที่ได้ ไปเขียนโปรแกรมสำหรับคำนวณ และส่งค่าเปอร์เซ็นต์แบตเตอรี่เข้าระบบฐานข้อมูลเพื่อนำข้อมูลไปใช้ต่อในส่วนอื่นๆ

4.6 การทดสอบการใช้งานโหมดประหยัดพลังงาน

ความถี่ในการส่งข้อมูลเป็นไปตามเงื่อนไงที่กำหนดคือถ้าเปอร์เซ็นต์ของแบตเตอรี่มากกว่า 25 เปอร์เซ็นต์จะเป็นการส่งข้อมูลในโหมดปกติ คือจะส่งข้อมูลในทุกๆ 20 วินาทีแสดงได้ดังรูปที่ 4.32 แต่ถ้าเปอร์เซ็นต์ของแบตเตอรี่น้อยกว่าหรือเท่ากับ 25 เปอร์เซ็นต์ จะเข้าสู่โหมดประหยัดพลังงาน ซึ่งจะใช้ความเร็วเป็นตัวกำหนดความถี่ในการส่งข้อมูลคือ ถ้ารถไฟมีความเร็วมากกว่า 10 กิโลเมตรต่อชั่วโมง จะส่งข้อมูลในทุกๆ 40 วินาทีแสดงได้ดังรูปที่ 4.33 แต่ถ้ารถไฟมีความเร็วน้อยกว่า 10 กิโลเมตรต่อชั่วโมง จะส่งข้อมูลในทุกๆ 60 วินาที แสดงได้ดังรูปที่ 4.34

	day	id	L	n	bat	spd	s	No
✎ ✕	2017-04-18 17:51:13	2	13.723264	100.782740	100	0.39	1	1
✎ ✕	2017-04-18 17:51:34	2	13.723170	100.782719	100	1.24	1	2
✎ ✕	2017-04-18 17:51:54	2	13.723337	100.782730	100	2.47	1	3
✎ ✕	2017-04-18 17:52:15	2	13.723357	100.782719	100	1.59	1	4
✎ ✕	2017-04-18 17:52:15	2	13.723242	100.782740	100	1.19	1	5

รูปที่ 4.32 การส่งข้อมูลทุกๆ 20 วินาที

2017-04-19 17:08:17	2	13.728073	100.770713	22	50.65	1	20		
2017-04-19 17:09:01	2	13.728017	100.768104	22	61.38	1	21		
2017-04-19 17:09:41	2	13.727927	100.765060	22	65.97	1	22		
2017-04-19 17:10:23	2	13.727837	100.761985	22	66.69	1	23		
2017-04-19 17:11:04	2	13.727767	100.758695	22	64.75	1	24		

รูปที่ 4.33 การส่งข้อมูลทุกๆ 40 วินาที

2017-04-19 18:42:30	2	13.727555	100.775895	20	2.74	1	41		
2017-04-19 18:43:31	2	13.727565	100.775906	20	5.18	1	42		
2017-04-19 18:44:35	2	13.727548	100.775830	19	4.14	1	43		
2017-04-19 18:45:36	2	13.727496	100.775862	19	2.89	1	44		
2017-04-19 18:46:36	2	13.727527	100.775958	20	4.14	1	45		

รูปที่ 4.34 การส่งข้อมูลทุกๆ 60 วินาที

4.7 การทดสอบเปลี่ยนค่าสถานะของระบบติดตามรถไฟ (S) อัตโนมัติ

สถานะของระบบติดตามรถไฟ (s) จะเปลี่ยนเองอัตโนมัติเมื่อรถไฟเข้าสู่สถานีสุดท้าย (หัวลำโพง) เพื่อแจ้งสถานะการงดให้บริการของรถไฟผ่านช่องทางต่างๆ ดังแสดงดังรูปที่ 4.35

day	id	L	n	bat	spd	No	s	w
2017-04-01 19:41:45	1	13.758002	100.528724	100	18.72	368731	1	0
2017-04-01 19:55:00	1	13.742074	100.517616	100	0.44	368732	1	0
2017-04-01 19:55:36	1	13.742014	100.517585	100	0.13	368733	1	0
2017-04-01 19:56:13	1	13.742030	100.517463	100	0.57	368734	1	0
2017-04-01 19:56:48	1	13.742075	100.517440	100	0.81	368735	1	0
2017-04-01 19:57:07	1	13.742024	100.517410	100	1.70	368736	1	0
2017-04-01 19:57:25	1	13.741896	100.517410	100	1.50	368737	1	0
2017-04-01 19:57:44	1	13.741703	100.517356	100	1.43	368738	1	0
2017-04-01 19:58:01	1	13.741557	100.517311	100	0.87	368739	1	0
2017-04-01 19:58:19	1	13.741371	100.517303	100	0.91	368740	1	0
2017-04-01 20:00:25	1	13.740509	100.517051	100	0.81	368741	0	0
2017-04-01 20:01:38	1	13.739792	100.516815	100	4.24	368742	0	0
2017-04-01 20:02:31	1	13.739762	100.516830	100	1.33	368743	0	0
2017-04-01 20:02:50	1	13.739647	100.516609	100	6.30	368744	0	0

รูปที่ 4.35 การเปลี่ยนสถานะของระบบติดตามรถไฟอัตโนมัติ

4.8 3G Module

ส่วนของ 3G Module ผู้พัฒนาได้ทำการเขียนโปรแกรม ให้ระบบมีการส่งพิกัดของรถไฟ และค่าเปอร์เซ็นต์แบตเตอรี่เข้าฐานข้อมูล แสดงดังรูปที่ 4.36 โดยค่าพิกัดของรถไฟจะได้จาก GPS Module ส่วนเปอร์เซ็นต์ของแบตเตอรี่จะได้จากการคำนวณค่าความต่างศักย์ที่อ่านได้จากโวลต์เตจเซ็นเซอร์ (Voltage Sensor)

day	id	L	n	bat	spd	No	s	w
2017-04-01 19:41:45	1	13.758002	100.528724	100	18.72	368731	1	0
2017-04-01 19:55:00	1	13.742074	100.517616	100	0.44	368732	1	0
2017-04-01 19:55:36	1	13.742014	100.517585	100	0.13	368733	1	0
2017-04-01 19:56:13	1	13.742030	100.517463	100	0.57	368734	1	0
2017-04-01 19:56:48	1	13.742075	100.517440	100	0.81	368735	1	0
2017-04-01 19:57:07	1	13.742024	100.517410	100	1.70	368736	1	0
2017-04-01 19:57:25	1	13.741896	100.517410	100	1.50	368737	1	0
2017-04-01 19:57:44	1	13.741703	100.517356	100	1.43	368738	1	0
2017-04-01 19:58:01	1	13.741557	100.517311	100	0.87	368739	1	0
2017-04-01 19:58:19	1	13.741371	100.517303	100	0.91	368740	1	0
2017-04-01 20:00:25	1	13.740509	100.517051	100	0.81	368741	0	0
2017-04-01 20:01:38	1	13.739792	100.516815	100	4.24	368742	0	0
2017-04-01 20:02:31	1	13.739762	100.516830	100	1.33	368743	0	0
2017-04-01 20:02:50	1	13.739647	100.516609	100	6.30	368744	0	0

รูปที่ 4.36 ค่าพิกัดและเปอร์เซ็นต์ของแบตเตอรี่ในฐานข้อมูล

บทที่ 5
สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

ผู้วิจัยได้พัฒนาระบบติดตามรถไฟจากระบบก่อนหน้า โดยส่วนแรก ผู้วิจัยได้เปลี่ยนอุปกรณ์ในการส่งค่าพิกัดของรถไฟจากที่ใช้งานในระบบ 2G ให้เป็นระบบ 3G เพื่อให้รองรับการใช้งานในอนาคต ส่วนที่สอง ผู้วิจัยได้มีการติดตั้งโวลเตจเซนเซอร์เพื่อวัดค่าโวลเตจคงเหลือของโมดูล GNSS เพื่อนำไปออกแบบอัลกอริทึมสำหรับรักษาแบตเตอรี่ให้ใช้งานได้นานยิ่งขึ้น โดยสามารถหลีกเลี่ยงปัญหาแบตเตอรี่หมดกลางทางได้ ส่วนที่สาม ผู้วิจัยได้ทำการออกแบบพัฒนาในส่วนการแสดงผลตำแหน่งรถไฟเพิ่มเติมจากระบบเดิม โดยได้พัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ เว็บไซต์ และจอมอนิเตอร์ ตามลำดับ โดยส่วนของแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์นั้น ผู้วิจัยได้พัฒนาระบบการแสดงผลตำแหน่งของรถไฟที่กำลังเข้าสถานีต่างๆ โดยผู้ใช้สามารถกดเลือกสถานีที่ผู้ใช้งานอยู่ใกล้ที่สุด หรือสถานีอื่นๆได้ นอกจากนั้น แอปพลิเคชันยังมีระบบการแจ้งเตือน เมื่อขบวนรถไฟกำลังเข้าสู่สถานีที่ผู้ใช้งานต้องการ ด้วยระบบการสั่นและเสียง ในส่วนของเว็บไซต์ ผู้วิจัยได้พัฒนาเว็ปไซต์ดั้งเดิมให้ทันสมัย และใช้งานได้ง่ายขึ้น ในส่วนสุดท้าย ส่วนของมอนิเตอร์ ผู้วิจัยได้ออกแบบและพัฒนาจอมอนิเตอร์เพื่อแสดงตำแหน่งรถไฟ โดยจอมอนิเตอร์ได้ถูกติดตั้งไว้ที่หน้าภาควิชาวิศวกรรมโทรคมนาคม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ซึ่งในส่วนเว็ปไซต์ และจอมอนิเตอร์จะมีทั้งเวอร์ชันภาษาไทย และอังกฤษ ซึ่งแอปพลิเคชันทั้งสามส่วนนี้ จะครอบคลุมผู้ใช้งานมากขึ้นจากระบบเดิม จากการสำรวจโดยคร่าว มีผู้ใช้งานทั้ง 3 แอปพลิเคชันดังกล่าวรวมกันประมาณ 300 คนต่อวัน

5.2 ข้อเสนอแนะ

ระบบอยู่ในระหว่างการทดลองใช้งาน ซึ่งในบางเวลาอาจจะไม่สามารถติดตามรถไฟได้ เนื่องจากความขัดข้องของอุปกรณ์ติดตาม เช่นแบตเตอรี่ของอุปกรณ์ติดตามรถไฟหมดขณะใช้งาน หรืออุปกรณ์ไม่สามารถส่งตำแหน่งเข้าระบบฐานข้อมูล

ดังนั้น หากไม่สามารถติดตามรถไฟได้ สามารถตรวจสอบตารางเวลารถไฟได้ที่เว็บไซต์การรถไฟแห่งประเทศไทยดังนี้http://www.railway.co.th/checktime/checktime.asp หรือเบอร์โทรศัพท์ 1690

บรรณานุกรม

[1] GEOG 862 - GPS and GNSS for Geospatial Professionals, "The Pseudorange Equation."
https://www.e-education.psu.edu/geog862/

[2] Movable Type Scripts. "Calculate distance, bearing and more between Latitude/Longitude."
http://www.movable-type.co.uk/scripts/latlong.html

[3] Android Studio. "Download Android Studio"
https://developer.android.com/studio/index.html

[4] Oracle Technology Network. "Downloads"
http://www.oracle.com/technetwork/java/javase/downloads/index.html

[5] สุธี พงศาสกุลชัย. (2010). ทำความรู้จัก Dynamic Web Content: มือใหม่หัดใช้ PHP (หน้า 2-12). สำนักพิมพ์ เคทีพี.

[6] ชาตพล นภาวารี. (2543). การใช้ javascript: JavaScript & Web design (หน้า 30-32). เอส.พี.ซี.บุ๊คส์.

[7] Arduino All. "Arduino UNO R3 รำคำ 320 บำท พร้อมสำย USB"
https://www.arduinoall.com/product/16/arduino-uno-r3

[8] Arduino All. "NEO-6M Ublox/u-blox GPS Module"
http://www.arduinoall.com/product/149/neo-6m-ublox-u-blox-gps-module

[9] Arduitronics. "Voltage Sensor Module (0-24 V)"
https://www.arduitronics.com/product/586/arduino-standard-voltage-sensor-module-0-24-v

[10] Thaieasyelec. "3G Module (UC15-T)"
http://www.thaieasyelec.com/products/wireless-modules/gsm-3g/3g-module-uc15-t

ภาคผนวก

โค้ดคำสั่ง

กำหนด AndroidManifest.xml เป็นไฟล์จัดการการทำงานโดยรวมของแอปพลิเคชัน ซึ่งจะกำหนดคุณสมบัติต่างๆของแอปพลิเคชันไว้ในไฟล์นี้เป็นหลัก อย่างเช่น ชื่อแอปพลิเคชัน, เวอร์ชันของแอปพลิเคชัน, API ที่รองรับ Activity ที่จะให้ทำงาน และการขออนุญาติใช้งานบางอย่าง (Permission) ซึ่งจะเป็นการทำงานโดยรวมของแอปพลิเคชันเท่านั้น ดังรูปที่ 1

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="cssrg.telecom.kmitl.intelligenttraintrackingsystemforeasternrailroute">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="com.google.android.providers.gsf.permission.READ_GSERVICES" />
    <uses-permission android:name="android.permission.ACCESS_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.VIBRATE" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
    <uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
    <uses-permission android:name="android.permission.CHANGE_NETWORK_STATE" />
    <uses-permission android:name="android.permission.WAKE_LOCK" />
    <uses-permission android:name="android.permission.DISABLE_KEYGUARD" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="INTELLIGENT TRAIN TRACKING SYSTEM FOR EASTERN RAIL ROUTE"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

        <activity
            android:name="cssrg.telecom.kmitl.intelligenttraintrackingsystemforeasternrailroute.Splash"
            android:screenOrientation="portrait">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity
            android:name="cssrg.telecom.kmitl.intelligenttraintrackingsystemforeasternrailroute.MainActivity"
            android:screenOrientation="portrait">
        </activity>

        <activity
```

รูปที่ 1 กำหนด AndroidManifest.xml

Class DataHandler กำหนดการทำงานเพื่อรับข้อมูลจากฐานข้อมูล เช่น ละติจูด ลองจิจูด เป็นต้น ดังรูปที่ 2

```
static public void Receivelatlng(final int trainNo) {

    new AsyncTask<Void, Void, String>() {
        @Override
        protected String doInBackground(Void... voids) {

            OkHttpClient okHttpClient = new OkHttpClient();
            Request.Builder builder = new Request.Builder();
            Request request = builder.url("http://train.telecom.kmitl.ac.th/downloadlatlng"
                    +Integer.toString(trainNo)+".php").build();

            try {
                Response response = okHttpClient.newCall(request).execute();
                if (response.isSuccessful()) {
                    return response.body().string();
                } else {
                    return "Not Success - code : " + response.code();
                }
            } catch (IOException e) {
                e.printStackTrace();
                return "Error - " + e.getMessage();
            }
        }

        @Override
        protected void onPostExecute(String string) {
            Log.d("debug","onPostExecute");
            super.onPostExecute(string);


            try {
                Log.d("debug","inTry");
                JSONArray data = new JSONArray(string);
                final ArrayList<HashMap<String, String>> MyArrList = new ArrayList<~>();
                HashMap<String, String> map;

                for (int i = 0; i < data.length(); i++) {
                    JSONObject c = data.getJSONObject(i);
                    map = new HashMap<>();
                    map = new HashMap<String, String>();
                    map.put("day", c.getString("day"));
                    map.put("id", c.getString("id"));
```

รูปที่ 2 Class DataHandler

Class TrainClass กำหนดรับส่งข้อมูลของขบวนรถไฟในแต่ละขบวนต่างๆเช่น เลขบวน ต้นทางปลายทาง ตำแหน่งปัจจุบัน ขาไปขากลับ วันเวลา ตารางเวลา เป็นต้น เพื่อใช้ class อื่นๆ ดังรูปที่ 3

```
public class TrainClass {
    private int trainNo; //หมายเลข
    private int currentStation; //หมายเลขสถานีปัจจุบันของรถไฟ (index) array stationTable
    private int service; //หมายเลขวิ่งของ service
    private double lattitude; //หมายเลขละติจูด
    private double longitude; //หมายเลขลองจิจูด
    private int round; //หมายเลขรอบ
    private String day; //เก็บค่าวันที่ปัจจุบันของรถไฟ
    private int image; //หมายเลขรูป drawable
    private String[] timetablego; //หมายเลขตารางไป
    private String[] timetableback; //หมายเลขตารางกลับ


    //Eng = 1, Thai = 0
    public int getImage() { return image; }
    public String getSourceEng() { return source[1]; }

    public String getDestinationEng() { return destination[1]; }

    public String getSourceTh() { return source[0]; }

    public String getDestinationTh() { return destination[0]; }

    private String[] source = new String[2];
    private String[] destination = new String[2];

    public TrainClass(int trainNo, int round, String sourceTh, String sourceEng,
                      String destinationTh, String destinationEng,
                      int image, String[] timetablego , String[] timetableback) {
        this.trainNo = trainNo;
        this.round = round;
        this.source[0] = sourceTh;
        this.destination[0] = destinationTh;
        this.source[1] = sourceEng;
        this.destination[1] = destinationEng;
        this.image = image;
        this.timetablego = timetablego.clone();
        this.timetableback = timetableback.clone();
```

รูปที่ 3 Class TrainClass

Class Location กำหนดการทำงานเพื่อหาตำแหน่งของขบวนรถไฟ แสดงตำแหน่งของรถไฟตามสถานีและระหว่างเส้นทาง ดังรูปที่ 4

```
private void condition(TrainClass train) {
    int currentStation = train.getCurrentStation();
    TextView now = (TextView) findViewById(R.id.textStationThai);
    TextView now2 = (TextView) findViewById(R.id.textStationEng);
    TextView nextstation = (TextView) findViewById(R.id.textNextStationThai);
    TextView nextstation2 = (TextView) findViewById(R.id.textNextStationEng);

    if (train.getService() == 1) {
        if (currentStation == -1) {
            now.setText(" ");
            now2.setText(" ");
            nextstation.setText(" ");
            nextstation2.setText(" ");
        } else if (currentStation % 2 == 0) {
            now.setText(stationName[currentStation / 2]);
            now2.setText(stationName1[currentStation / 2]);

            int index = (currentStation / 2) + train.getRound();

            if ((index == 46 && train.getRound() == 1) || (index == 63 && train.getRound() == 1) ||
                nextstation.setText("-");
                nextstation2.setText("-");
            } else {
                nextstation.setText(stationName[index]);
                nextstation2.setText(stationName1[index]);
            }
        } else {
            now.setText("ระหว่างทางไป" + stationName[(currentStation + train.getRound()) / 2]);
            now2.setText("On the way to " + stationName1[(currentStation + train.getRound()) / 2]);
            nextstation.setText(stationName[(currentStation + train.getRound()) / 2]);
            nextstation2.setText(stationName1[(currentStation + train.getRound()) / 2]);
        }
    } else if (train.getService() == 0) {
        now.setText("ยังไม่มีบริการระบบติดตามรถไฟ");
        now2.setText("สอบถามเพิ่มเติม โทร.1690");
        nextstation.setText("No tracking system service.");
        nextstation2.setText("Please contact 1690.");
    } else if (train.getService() == -1) {
        now.setText("กำลังโหลด...");
        now2.setText("");
        nextstation.setText("downloading...");
        nextstation2.setText("");
```

รูปที่ 4 Class Location

Class NearestStation กำหนดการทำงานคำนวณ haversian หาระยะทางที่สั้นที่สุดระหว่างผู้ใช้งานกับสถานีรถไฟ เพื่อหาสถานีรถไฟใกล้ตัวที่สุด ดังรูปที่ 5

```
private double CalDistance(double latitude, double longitude, double lat2, double lon2) {
    double EARTH_RADIUS = 6367.45;
    double deltalat = Math.toRadians(lat2 - latitude);
    double deltalon = Math.toRadians(lon2 - longitude);

    double a = Math.sin(deltalat / 2) * Math.sin(deltalat / 2) + Math.cos(Math.toRadians(latitude))
            * Math.cos(Math.toRadians(lat2)) * Math.sin(deltalon / 2) * Math.sin(deltalon / 2);
    double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 - a));
    double distance = EARTH_RADIUS * c;
    return distance;
}
```

รูปที่ 5 Class NearestStation

Class FindTrain กำหนดการทำงานเพื่อหาขบวนรถไฟที่กำลังเข้าสถานีต่างๆ ดังรูปที่ 6

```
private ArrayList<TrainClass> findTrain (){
    ArrayList<TrainClass> result = new ArrayList<~>();
    ArrayList<TrainClass> allTrains = DataHandler.getAllTrains();

    TextView title = (TextView) findViewById(R.id.textTitleFindTrain);
    for(int i=0;i<allTrains.size();i++) {

        TrainClass train = allTrains.get(i);
        int round = train.getRound();
        int trainNo = train.getTrainNo();
        int currentStation = train.getCurrentStation();

        if (train.getService() == 1) {

            if (selectedStation >= 0 && selectedStation <= 36) {
                if (findTrainCondition(round, currentStation, selectedStation)) {
                    result.add(train);
                    title.setText("ขบวนรถไฟในระบบคัดตามสถานี" + stationName[selectedStation / 2] + "ในขณะนี้");
                }

            } else if (selectedStation >= 37 && selectedStation <= 92 && trainNo == 275) {
                if (findTrainCondition(round, currentStation, selectedStation)){
                    result.add(train);
                    title.setText("ขบวนรถไฟในระบบคัดตามสถานี" + stationName[selectedStation / 2] + "ในขณะนี้");
                }
            } else if (selectedStation >= 93 && selectedStation <= 124 && trainNo == 283) {
                if (findTrainCondition(round, currentStation, selectedStation)){
                    result.add(train);
                    title.setText("ขบวนรถไฟในระบบคัดตามสถานี" + stationName[selectedStation / 2] + "ในขณะนี้");
                }
            }
        }

        if (result.size()==0 || train.getService()== 0){
            title.setText("ไม่มีขบวนรถไฟในระบบคัดตามสถานี" + stationName[selectedStation / 2] + "ในขณะนี้");
        }

    }

    return result;
}
```

รูปที่ 6 Class FindTrain

Class Alert กำหนดการทำงานของการแจ้งเตือนเมื่อขบวนรถไฟกำลังเข้าสถานี ดังรูปที่ 7

```
new AsyncTask<Void,Void,Void>(){
    @Override
    protected Void doInBackground(Void... voids) {

        TrainClass train = DataHandler.getTrain(trainNo);
        int round = train.getRound();
        int currentStation = train.getCurrentStation();
        int selectStation = position*2;
        if (train.getService() == 1 && !isAlerted) {
            if (trainNo == 275) {
                if ((currentStation - (selectStation) == -2*round)) {
                    ring(context, alarmManager, trainNo);
                }
            } else if (trainNo == 283) {
                if (selectStation >= 0 && selectStation <= 36) {
                    if ((currentStation - (selectStation) == -2*round)) {
                        ring(context, alarmManager, trainNo);
                    }
                } else if (selectStation >= 38) {
                    if ((currentStation - (selectStation + 56) == -2*round)) {
                        ring(context, alarmManager, trainNo);
                    }
                }
            }

        }
        return null;
    }
}.execute();

if(!isAlerted) {
    android.os.Handler h = new android.os.Handler();
    h.postDelayed(() -> {
            loop(trainNo, position, context, alarmManager);
    }, 3000);
}
```

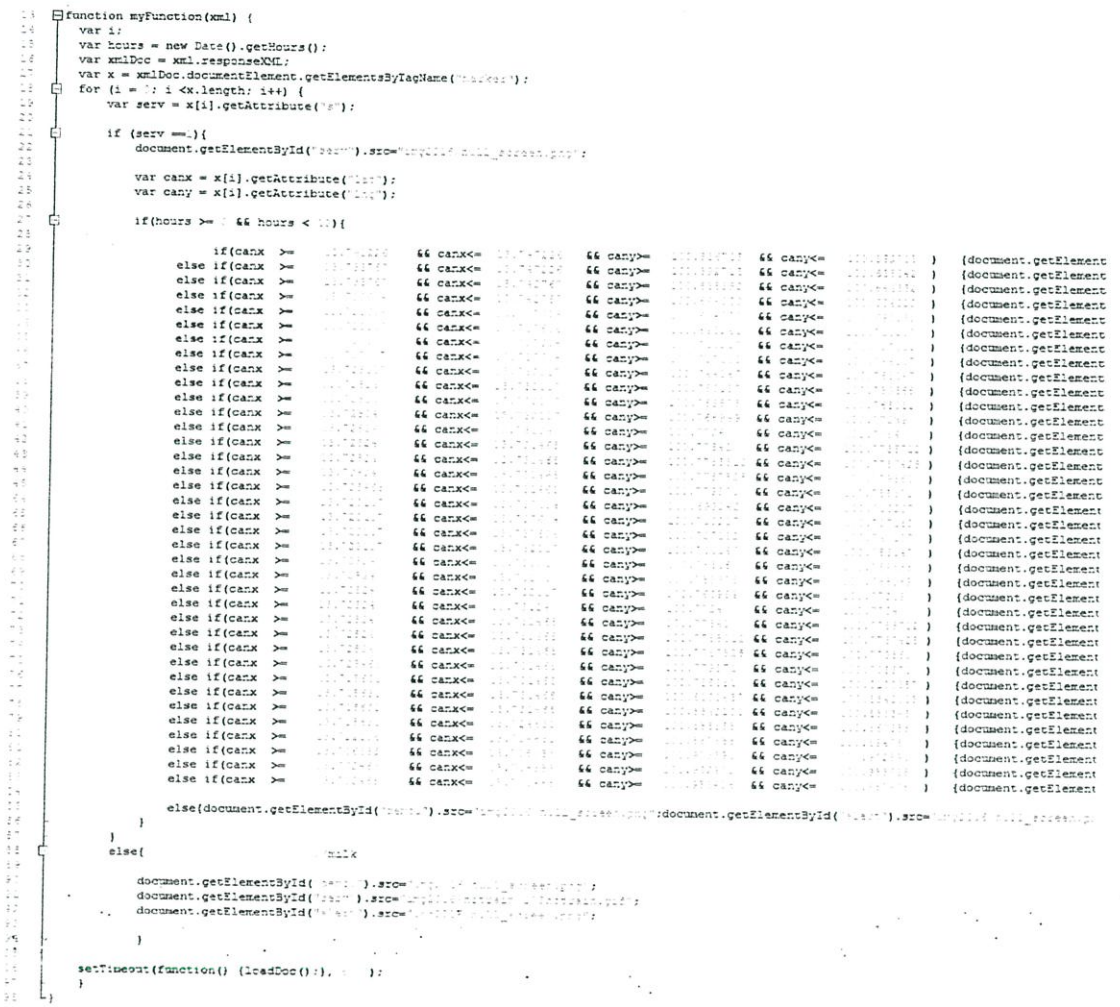
รูปที่ 7 Class Alert

กำหนดไฟล์ testmap2.js เพื่อใช้ในการดึงค่า XML จาก URL แล้วเข้าเงื่อนไขพิกัดสถานีแล้วแสดงข้อมูลบนหน้าจอมอนิเตอร์ โดยฟังก์ชัน loadDoc() ใช้สำหรับติดต่อกับเซิฟเวอร์แล้วตรวจสอบสถานะการทำงาน ดังรูปที่ 8

```
function loadDoc() {
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (xhttp.readyState == 4 && xhttp.status == 200) {
      myFunction(xhttp);
    }
  };
  xhttp.open("GET", "http://localhost/map/", true);
  xhttp.send();
}
```

รูปที่ 8 ฟังก์ชัน loadDoc()

ฟังก์ชัน myFunction(xml) ใช้ในการดึงค่า XML แล้วเข้าเงื่อนไขตรวจสอบตำแหน่งสถานีรถไฟแล้วแสดงผลบนหน้าจอมอนิเตอร์ ดังรูปที่ 9



รูปที่ 9 ฟังก์ชัน myFunction(xml)

กำหนดไฟล์ testindex.html ในการแสดงข้อมูลหน้าจอมอนิเตอร์ที่ผ่านการทำงานจากไฟล์ testmap2.js ดังรูปที่ 10

```
<body>

<div id="welcome">
<div id="contain_logo"><img src="img2016/map.gif" width="100%" height="100%"/>
            <div id="contain_logo1">
            <img id="demo1" width="100%" height="100%"/>    <!-- 275-->
            <script src='testmap2.js'></script>
            </div>
 </div>
 </div>

<div style="margin:0px;" >
<div class="boxA">
 <img src="boxingring.png" name="slide" width="100%" height="100%">
 <script src='testalert.js'></script>
            <div class="boxA2">
            <img id="alert" width="100%" height="100%"/>
            <script src='testmap2.js'></script>
            </div>
 </div>

 <div class="boxB" align="center"></div>

<div class="boxC" align="center"><iframe src="http://free.timeanddate.com/clock/i5fqqvzh/n28/szw160/szh160/hoc000/hbw4/cf100/hgr0/fav0/fiv0/mgc00(
 frameborder="0" width="160" height="160"></iframe>
 </div>

 <div class="boxB" align="center"></div>

<div class="boxD" align="center"><iframe src="http://free.timeanddate.com/clock/i5qqvvsg/n28/tlth/fs24/tt1/tw1/tm1/td1"
 frameborder="0" width="204" height="30"></iframe>
 </div>

    <div class="boxE" align="center"><img src="img2016/table2.gif" align="top" height="100%" width="100%" />
    <div class="boxE2" align="center"><img id="serv" align="top" height="100%" width="100%" />
    <script src='testmap2.js'></script>
    </div>
    </div>

 <div class="header" align="center"><img src="img2016/header.gif" align="top" width="100%" height="100%" /></div>

</div>
</body>
</html>
```

รูปที่ 10 testindex.html

กำหนดไฟล์ index.html ใช้ในการแสดงผลหน้าหลักของเว็บไซต์ โดยแบ่งเป็น 4 ส่วน ดังนี้ ส่วนเลือกขบวนรถไฟ ส่วนดาวน์โหลด ส่วนตารางเวลารถไฟและส่วนเกี่ยวกับเรา ดังรูปที่ 11 12 13 และ 14 ตามลำดับ

```
    <!-- Choise -->
    <header class="w3-display-container w3-wide w3-center" style=""  id="home">
        <img src="img2559\รถไฟ3.jpg" alt="train" height="600px">
        <div class="w3-display-topleft w3-padding-48 w3-center" style="left:10% ;"  >
        <h1 class="w3-xxlarge w3-text-light-black w3-light-gray w3-opacity">เลือกขบวนรถไฟ</h1>
        <h6><a class="w3-btn w3-white w3-padding-large w3-large w3-opacity w3-hover-opacity-off"
        style="width:100%" href="http://train.telecom.kmitl.ac.th/Traintrack/new/select275.html" >
        <span class="w3-hide-small">275-276</span>กรุงเทพ-อรัญประเทศ</a></h6>
        <h6><a class="w3-btn w3-white w3-padding-large w3-large w3-opacity w3-hover-opacity-off"
        style="width:100%" href="http://train.telecom.kmitl.ac.th/Traintrack/new/select283.html" >
        <span class="w3-hide-small">283-284</span>กรุงเทพ-น้ำพุลาหลวง</a></h6>
        <h6><a class="w3-btn w3-white w3-padding-large w3-large w3-opacity w3-hover-opacity-off"
        style="width:100%" href="" ><span class="w3-hide-small">281-282</span>กรุงเทพ-กบินทร์บุรี<br>(ในอนาคต)</a></h6>
        <h6><a class="w3-btn w3-white w3-padding-large w3-large w3-opacity w3-hover-opacity-off"
        style="width:100%" href="" ><span class="w3-hide-small">367-368</span>กรุงเทพ-ฉะเชิงเทรา<br>(ในอนาคต)</a></h6>
        </div>
    </header>
```

รูปที่ 11 index.html ส่วนเลือกขบวนรถไฟ

```
<!-- download-->
<div class="w3-col w3-padding-32  m6 l4 w3-center " >
<div class="w3-container  w3-black" ><h4 class="w3-large">ดาวน์โหลดแอปพลิเคชัน</h4></div>
<div class="w3-container w3-padding w3-white">
<p>ท่านสามารถดาวน์โหลดแอปพลิเคชันระบบติดตามรถไฟสายตะวันออกอัจฉริยะผ่านทาง Google Play Store  <br></p>
<a href="https://goo.gl/V4rVmc"><img class="w3-margin" src="img2559/qrcode.jpg" alt="qrcode" width="30%"></a>
<p>แนะนำติดชมผ่าน line official ID : TTS KMITL(@tyq2297v) <br></p>
</div>
</div>
```

รูปที่ 12 index.html ส่วนดาวน์โหลด

```
<!-- table-->
<div class="w3-col w3-padding-32  m6 l4 w3-center " >
<div class="w3-container  w3-black" ><h4 class="w3-large">ตารางเวลารถไฟ</h4></div>
<div class="w3-container w3-padding w3-white">
<p>ตารางเวลารถไฟตามกำหนดการของการรถไฟแห่งประเทศไทย</p>
<ul class="w3-ul w3-hoverable">
<li><a href="http://www.railway.co.th/Ticket/list_station_stop.asp?IdTrain=275&Leg=th">275-276 กรุงเทพ-อรัญประเทศ</a></li>
<li><a href="http://www.railway.co.th/Ticket/list_station_stop.asp?IdTrain=283&Leg=th">283-284 กรุงเทพ-ปากน้ำโพ</a></li>
<li><a href="http://www.railway.co.th/Ticket/list_station_stop.asp?IdTrain=281&Leg=th">281-282 กรุงเทพ-กบินทร์บุรี</a></li>
<li><a href="http://www.railway.co.th/Ticket/list_station_stop.asp?IdTrain=367&Leg=th">367-368 กรุงเทพ-ฉะเชิงเทรา</a></li>
<li><a href="http://www.railway.co.th/checktime/checktime.asp">ขบวนอื่นๆ</a></li>
</ul></div>
</div>
```

รูปที่ 13 index.html ส่วนตารางเวลารถไฟ

```
<!-- about-->
<div class="w3-col w3-padding-32  m12 l4 w3-center " >
<div class="w3-container  w3-black" ><h4 class="w3-large">เกี่ยวกับเรา</h4></div>
<div class="w3-container w3-padding w3-white">
<p><b>โครงการระบบติดตามรถไฟสายตะวันออกอัจฉริยะ</b> <br>เป็นระบบติดตามรถไฟสายระบบวิทยุเอส  จัดทำขึ้นเพื่อแก้ไขปัญหาการจราจรรถไฟเป็นอย่างมาก ระบบจะแสดงตำแหน่งของรถไฟผ่านทางเว็บไซต์ ท
</div>
</div>

</div>
```

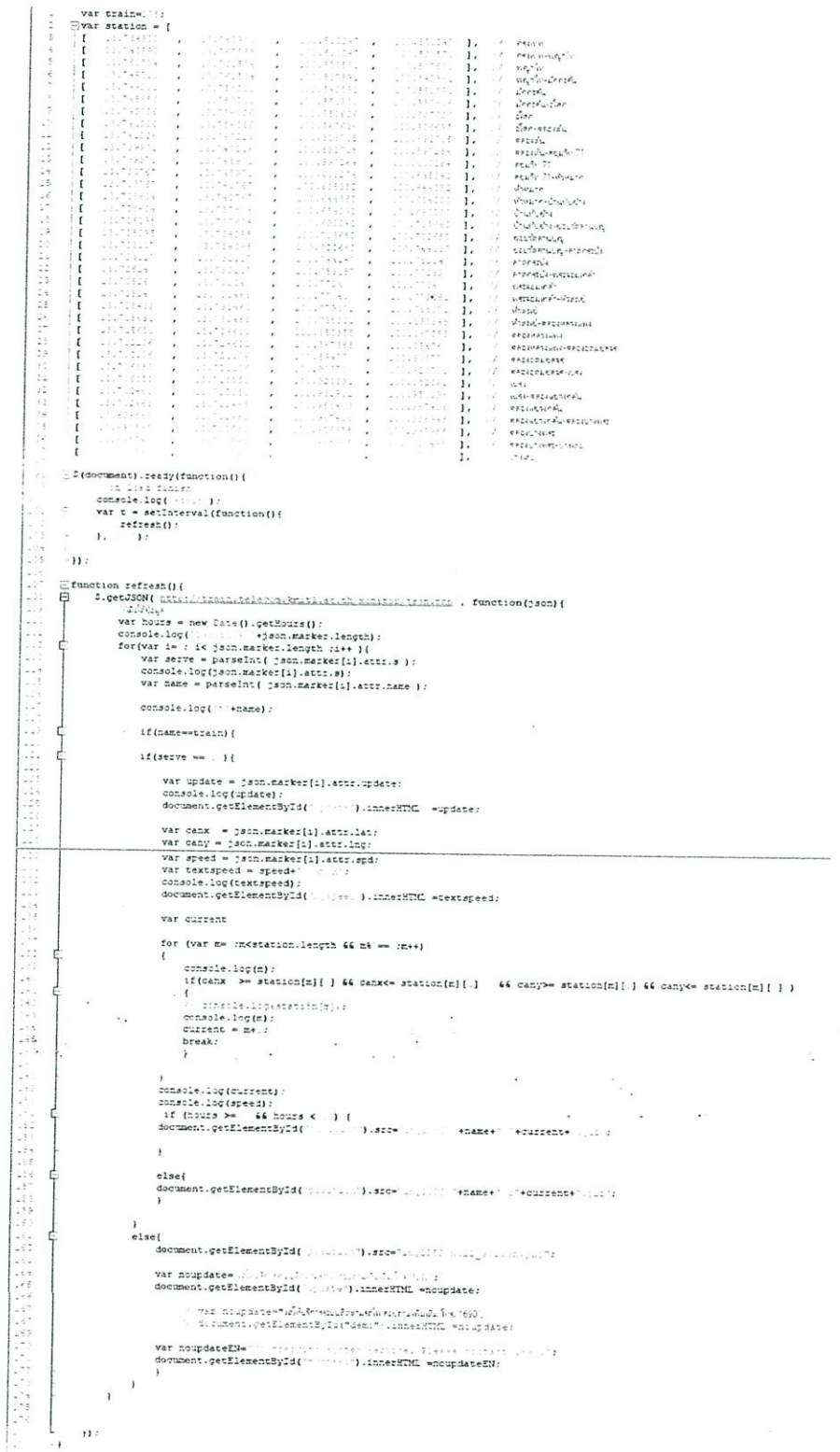
รูปที่ 14 index.html ส่วนเกี่ยวกับเรา

และในหน้าแสดงตำแหน่งขบวนรถไฟ แสดงโดยไฟล์ select275.html ดังรูปที่ 15

```
<div class=" w3-black w3-btn-group">
    <a class="w3-btn w3-light-gray w3-padding-large">275-276<br>กรุงเทพ-อรัญประเทศ</a>
    <a class="w3-btn w3-black w3-padding-large w3-opacity w3-hover-gray" href="http://train.telecom.kmitl.ac.th/Traintrack/new/select283.html" >283-284
</div>

<div class="b w3-light-gray w3-margin">
<table class="" align="center" >
<tr>
  <td style="width:50%;"><b>เวลา</b> </td>
  <td><iframe src="http://free.timeanddate.com/clock/i5n6qxfc/n28/fn6/fs15/tct/pct/fsb/tt0/tw0/tm3/td2/th1/th2" frameborder="0" width="147" height="20"
</td
</tr>

<tr>
<td>ความเร็วขบวนรถไฟ</td>
<td><a id="myspeed"></a>
</tr>

<tr>
  <td>สถานปัจจุบัน </td>
  <td><a id="update"></a>
</tr>
</tr>

</table>
</div>

<div class="c" align="center">
<img src="img2559\select275.jpg" alt="qrcode" width="90%">
</div>
<div class="d" align="center">
<img id="position" width="90%" />
</div>
```

รูปที่ 15 select275.html

ในการดึงข้อมูลจาก database มาแสดงผลบนเว็บไซต์ ใช้ไฟล์ map275.js ดังรูปที่



รูปที่ 16 map275.js

ซึ่งอธิบายได้ดังนี้

บรรทัดที่ 1 กำหนดค่าหมายเลขขบวนรถไฟ

บรรทัดที่ 2-96 กำหนดพิกัดสถานีรถไฟ

บรรทัดที่ 98-103 ให้ทำฟังก์ชัน refresh เมื่อโหลดหน้าเว็บไซต์เสร็จและทำซ้ำทุกๆ 3 วินาที

บรรทัดที่ 107-178 ฟังก์ชัน refresh เงื่อนไขการแสดงพิกัดตำแหน่งรถไฟ

การออกแบบการทำงานของ Arduino เพื่อใช้ในการส่งค่าพิกัดรถไฟ และเปอร์เซ็นของแบตเตอรี่ดังรูปที่ 17

```
#include <SoftwareSerial.h>
#include <TinyGPS.h>
#include <AltSoftSerial.h>

AltSoftSerial mySerial;
TinyGPS gps;
SoftwareSerial ss(..,..);
//GPS

float flat, flon , s;
float l =
float n =
//BATTERY
float val1;
const int numReadings =
int readings[numReadings];      // the readings from the analog input
int readIndex =
int total =
int average =

int inputPin = A0;
float z,y;
```

รูปที่ 17 โค้ดการประกาศตัวแปล และสั่งให้อุปกรณ์ทำงาน

บรรทัดที่ 1-3 เป็นการประกาศใช้ softwareserial ในการสั่งงาน และ ประกาศใช้ library ของ tinyGPS และ altsoftserial

บรรทัดที่ 5-7 เป็นการกำหนดพินของการใช้งานอุปกรณ์

บรรทัดที่ 9-22 เป็นการประกาศ float เก็บค่า พิกัดและความต่างศักย์

```
void setup() {
 Serial.begin(
   for (int thisReading =  ; thisReading < numReadings; thisReading++)
     readings[thisReading] =  ;
  ss.begin(
   delay(
   SendOn();

}
```

รูปที่ 18 โค้ด สั่งให้อุปกรณ์ทำงาน

บรรทัดที่ 24-29 เป็นการประกาศอัตราการเริ่มต้นทำงานของอุปกรณ์
บรรทัดที่ 30 เข้าลูป SendOn ( )

```
34  void loop() |
35  {
36    total = total - readings[readIndex];
37          readings[readIndex] = analogRead(inputPin);
38          total = total + readings[readIndex];
39          readIndex = readIndex + 1;
40
41          if (readIndex >= numReadings)
42            readIndex = 0;
43          average = total / numReadings;
44
45        float temp;
46          val1 = average;
47          temp=val1/
48          z=temp;
49          if(z>=
50            y= (
51            if(y>
52              y=
53             }
54           }
55           if(z<
56              y= (
57              if(y>
58                y=
59              }
61            }
```

รูปที่ 19 โค้ดการหาค่าเฉลี่ยและการคำนวณความต่างศักย์ที่วัดได้จากแบตเตอรี่

บรรทัดที่ 36-43 เป็นการหาค่าเฉลี่ยของค่าที่วัดได้จากแบตเตอรี่เพื่อให้ค่าที่จะนำไปคำนวนความต่างศักย์มีความคลาดเคลื่อนน้อยลง
บรรทัดที่ 47-47 เป็นการคำนวณความต่างศักย์ของแบตเตอรี่
บรรทัดที่ 49-53 การคำนวณเปอร์เซ็นต์ของแบตเตอรี่ถ้าความต่างศักย์มากกว่า 3.57 โวลต์ ให้ใช้สมการเส้นตรงในการพิจารณา
บรรทัดที่ 55-58 การคำนวณเปอร์เซ็นต์ของแบตเตอรี่ถ้าความต่างศักย์น้อยกว่า 3.57 โวลต์ ให้ใช้สมการเอ็กซ์โพเนนเชียลในการพิจารณา

```
61    bool newData = false;
62    for ( unsigned long start = millis(); millis() - start < 1000;)
63    {
64      while (ss.available())
65      {
66        char c = ss.read();
67        if (gps.encode(c))
68        newData = true;
69        }
70      }
71       if (newData)
72      {
73        unsigned long age;
74        gps.f_get_position(&flat, &flon, &age);
75        s = gps.f_speed_kmph();
76        if ((flat != l) || (flon != n))
77        {
78
79          l = flat;
80          n = flon;
81
82          SendSQL();
83
84          }
85      }
86  }
```

รูปที่ 20 โค้ดการถอดรหัสค่าพิกัดโดยใช้ library ของ TinyGPS

บรรทัดที่ 61 เป็นการกำหนดให้ตัวแปร newData เป็น false ก่อนเพื่อนาไปใช้ในลูปต่อไป
บรรทัดที่ 62 กำหนดให้ลูปทางานทุกๆ 1 วินาที และมีข้อมูลเข้ามาถึงจะทำงาน
บรรทัดที่ 64-68 จะเป็นการอ่านค่าจาก จีพีเอสที่ส่งเข้ามาและเก็บในตัวแปร c และถ้าถอดรหัสได้จะกำหนดให้ แปร newData เป็น true
บรรทัดที่ 71-80 เมือ newData เป็น true จะเริ่มทำการ ดึงค่าตำแหน่งและ ความเร็วเป็น float เพื่อเป็นการป้องกันการส่งค่าเป็น 0.000000 จึงมีการเขียนลูปเช็ค ว่า ค่าที่เข้ามาต้องไม่ใช่ค่า 0.000000
บรรทัดที่ 82 เข้าลูปส่ง SendSQL ( )

```
 91   void SendSQL()
 92   {
 93    mySerial.println("AT+[illegible]");
 94    delay(2000);
 95    mySerial.println("AT+[illegible]");
 96    delay(2000);
 97    mySerial.println("AT+[illegible]");
 98    delay(2000);
 99    mySerial.println("AT+[illegible]");
100    delay(5000);
101    mySerial.println("AT+[illegible]");//74
102    delay(2000);
103    mySerial.print("[illegible]");//g
104    mySerial.print("&d=");
105    mySerial.print(l, 6);
106    mySerial.print("&m=");
107    mySerial.print(n, 6);
108    mySerial.print("&s=");
109    mySerial.print(s, 1);
110    mySerial.print("&y=");
111    mySerial.println(y,6);
112    delay(2000);
113    mySerial.println("AT+[illegible]");
114    delay(4000);
115    mySerial.println("AT+[illegible]");
116    delay(2000);
117   }
```

รูปที่ 21 ลูปการส่งค่าพิกัดและเปอร์เซ็นต์แบตเตอรี่ขึ้นเซิร์ฟเวอร์

บรรทัดที่ 93 เป็นการตั้ง PDP context ID = 1

บรรทัดที่ 95 อนุญาตให้ใช้ HTTP header

บรรทัดที่ 97 กำหนด APN (Access Point Name) ของเครือข่ายในที่นี้ใช้ internet

บรรทัดที่ 99 ประกาศพารามิเตอร์เปิดการใช้งาน internet

บรรทัดที่ 101 ตั้งความยาว URL และกำหนด Timeout

บรรทัดที่ 103-112 เป็นการใส่ ที่อยู่ของ php file ตามด้วยค่าตัวแปรต่างๆ

บรรทัดที่ 113 เป็นการสั่งให้เริ่มปฏิบัติการ หลังจากคำสั่งนี้ค่าของพิกัดจะถูกบันทึกลงฐานข้อมูลเป็นที่เรียบร้อย

บรรทัดที่ 115 เป็นการสั่งปิดการเชื่อมต่อ internet

```
172   void SendOn()
173   {
174    mySerial.println("AT+QHTTPCFG=\"contextid\",1");
175    delay(1000);
176    mySerial.println("AT+QHTTPCFG=\"responseheader\",1");
177    delay(1000);
178    mySerial.println("AT+QICSGP=1,1,\"internet\",\"True\",\"true\",1");
179    delay(1000);
180    mySerial.println("AT+QIACT=1");
181    delay(2000);
182    mySerial.println("AT+QHTTPURL=48,80");
183    delay(2000);
184    mySerial.print("http://train.telecom.kmutt.ac.th/");
185    mySerial.print("add=");
186    mySerial.println(1.00,2);
187    delay(2000);
188    mySerial.println("AT+QHTTPPOST=0");
189    delay(8000);
190    mySerial.println("AT+QIDEACT=1");
191    delay(2000);
192    }
```

รูปที่ 22 ลูปการส่งค่าขึ้นเซิร์ฟเวอร์

บรรทัดที่ 174 เป็นการตั้ง PDP context ID = 1
บรรทัดที่ 176 อนุญาตให้ใช้ HTTP header
บรรทัดที่ 178 กำหนด APN (Access Point Name) ของเครือข่ายในที่นี้ใช้ internet
บรรทัดที่ 180 ประกาศพารามิเตอร์เปิดการใช้งาน internet
บรรทัดที่ 182 ตั้งความยาว URL และกำหนด Timeout
บรรทัดที่ 184-187 เป็นการใส่ ที่อยู่ของ php file ตามด้วยค่าตัวแปรต่างๆ
บรรทัดที่ 188 เป็นการสั่งให้เริ่มปฏิบัติการ หลังจากคำสั่งนี้หมายเลขขบวนจะถูกบันทึกลงฐานข้อมูลเป็นที่เรียบร้อย
บรรทัดที่ 190 เป็นการสั่งปิดการเชื่อมต่อ internet

หมายเหตุ การหน่วงเวลานั้นเกิดจากขั้นตอนการทำงานของ 3G Module จะใช้เวลาในการประมวลมล แต่ บอร์ดไมโครคอลโทรลเลอร์นั้น ทำงานจากบรรทัดบนลงล่างไม่สนใจเวลา ทำให้ 3G Module ทำงานตามคำสั่งได้ไม่ทัน จึงใช้การหน่วงเวลาเพื่อที่จะรอให้ 3G Module ทำคำสั่งก่อนหน้าเสร็จก่อน โดยแต่ละคำสั่งใช้เวลาในการทำงานที่ไม่เท่ากัน โดย SendSQL ( ) SendSQL40 ( ) และSendSQL60 ( ) จะต่างกันที่การหน่วงเวลาในการส่ง