ระบบนำทางในอาคารโดยใช้เทคโนโลยีเอ็นเอฟซีบนสมาร์ตโฟน

INDOOR NAVIGATION SYSTEM USING NFC ON SMARTPHONE

เปี่ยมรัก ภัทรมุกขานนท์
PIAMRAK PATTARAMOOKANON

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2557

ระบบนำทางในอาคารโดยใช้เทคโนโลยีเอ็นเอฟซีบนสมาร์ตโฟน

INDOOR NAVIGATION SYSTEM USING NFC ON SMARTPHONE

เปี่ยมรัก ภัทรมุกขานนท์
PIAMRAK PATTARAMOOKANON

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2557

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INDOOR NAVIGATION SYSTEM USING NFC ON SMARTPHONE

PIAMRAK PATTARAMOOKANON

THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INFORMATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2014

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

หัวข้อปริญญานิพนธ์	ระบบนำทางในอาคารโดยใช้เทคโนโลยีเอ็นเอฟซีบนสมาร์ตโฟน
Thesis Title	INDOOR NAVIGATION SYSTEM USING NFC ON SMARTPHONE
ชื่อนักศึกษา	นางสาวเปี่ยมรัก ภัทรมุกขานนท์
ระดับปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมสารสนเทศ
ปริญญานิพนธ์ปีการศึกษา	2557

(.....)

ผศ.บุณย์ชนะ ภู่ระหงษ์

อาจารย์ที่ปรึกษาปริญญานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ ระบบนำทางในอาคารโดยใช้เทคโนโลยีเอ็นเอฟซีบนสมาร์ตโฟน
Thesis Title INDOOR NAVIGATION SYSTEM USING NFC ON SMARTPHONE
ชื่อนักศึกษา นางสาวเปี่ยมรัก ภัทรมุกขานนท์ รหัสนักศึกษา 54010836
ระดับปริญญา วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมสารสนเทศ
ปีการศึกษา 2557
อาจารย์ที่ปรึกษาปริญญานิพนธ์ ผศ.บุณย์ชนะ ภู่ระหงษ์

บทคัดย่อ

ในอนาคตเทคโนโลยีการสื่อสารไร้สายระยะสั้นที่เรียกว่า NFC (Near Field Communication) มีแนวโน้มที่จะใช้งานอย่างแพร่หลาย เนื่องจากสามารถเชื่อมต่อและแลกเปลี่ยนข้อมูลระหว่างอุปกรณ์ได้อย่างรวดเร็วและปลอดภัยเพียงแค่นำอุปกรณ์มาสัมผัสกัน ปัจจุบันมีสมาร์ตโฟน (Smartphone) ที่นำเทคโนโลยี NFC นี้มาใช้งานมากขึ้น ตัวอย่างการใช้งานเช่น การใช้โทรศัพท์มือถือเพื่อชำระค่าสินค้าและบริการ การชำระค่ารถโดยสาร การยืนยันตัวตน เป็นต้น

ในโครงงานนี้จึงนำเสนอระบบแผนที่ภายในอาคาร เช่น ศูนย์ประชุมหรือห้างสรรพสินค้า โดยใช้เทคโนโลยี NFC เพื่อใช้ในการระบุตำแหน่งปัจจุบันของผู้ใช้งาน และยังสะดวกในการบอกรายละเอียดของร้านค้า แอปพลิเคชั่นสามารถแสดงแผนที่ภายในอาคารและเส้นทางที่สั้นที่สุดตั้งแต่ต้นทางไปปลายทางซึ่งคำนวณด้วยเทคนิคกราฟ แอปพลิเคชั่นนี้ช่วยอำนวยความสะดวกให้ผู้ใช้งานในการนำทางสำหรับผู้ที่ไม่เคยไปสถานที่นั้นๆให้ถึงเป้าหมายได้อย่างถูกต้องและรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	INDOOR NAVIGATION SYSTEM USING NFC ON SMARTPHONE
Student	Miss.Piamrak Pattaramookanon Student ID. 54010836
Degree	Bachelor of Engineering
Program	Information Engineering
Academic Year	2557
Thesis Advisor	Asst.Prof. Boonchana Purahong

ABSTRACT

In the future, technology of short-range wireless communication called NFC (Near Field Communication) tends to use the widespread. Currently NFC technology integrated with smartphone for more capability such as payment and services, payment of the bus, identity verification, etc.

This project presents indoor navigation system, such as convention center or shopping malls. NFC technology using for show the current position of the user. It's easy to show the details of the store. An application can display a map of the building and the shortest path from source to destination, which is calculated by using the graph. The application is able to facilitate the user to navigate for those who have never been to that place to the target accurately and quickly.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สามารถสำเร็จลุล่วงไปได้ด้วยดี ข้าพเจ้าขอกราบขอบพระคุณอาจารย์ที่ปรึกษา ผศ.บุณย์ชนะ ภู่ระหงษ์ ที่ให้คำปรึกษา คำชี้แนะ ช่วยแก้ไขปัญหา ตลอดจนให้ความรู้และความเข้าใจในด้านต่างๆ ข้าพเจ้ารู้สึกซาบซึ้งในความอนุเคราะห์เป็นอย่างสูง

ขอกราบขอบพระคุณ บิดา มารดา และครอบครัวที่เป็นกำลังใจที่ดีเสมอและให้แรงผลักดันที่ทำให้ข้าพเจ้ามุ่งมั่นในการทำโครงงานนี้สำเร็จลุล่วงด้วยดี

ขอกราบขอบพระคุณคณาจารย์ในสาขาวิชาวิศวกรรมสารสนเทศทุกๆท่านที่ให้ความเมตตาประสิทธิ์ประสาทวิชาแก่ข้าพเจ้า

ขอบคุณพี่ๆและเพื่อนๆในห้องโปรเจคที่ให้คำปรึกษาและช่วยเหลือในด้านต่างๆ

ขอบคุณนายนิพัทธ์ กานต์กัมพล และนายปรัชญ์ เจียรสถาวงศ์ รุ่นพี่ชุมนุม Computer Club ที่ให้คำปรึกษาและช่วยแก้ไขปัญหาของโครงงานนี้

ข้าพเจ้าขอกราบขอบพระคุณมา ณ โอกาสนี้และหวังเป็นอย่างยิ่งว่าปริญญานิพนธ์นี้จะสามารถนำไปเป็นแนวทางในการประยุกต์ใช้เพื่อให้เกิดประโยชน์ต่อไปได้

เปี่ยมรัก ภัทรมุกขานนท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I
บทคัดย่อภาษาอังกฤษ II
กิตติกรรมประกาศ III
สารบัญ IV
สารบัญตาราง VII
สารบัญรูป VIII
บทที่ 1 บทนำ 1
1.1 ความสำคัญและที่มา 1
1.2 จุดประสงค์ 2
1.3 ขอบเขตของโครงงาน 2
1.4 ผลที่คาดว่าจะได้รับ 2
1.5 อุปกรณ์ที่ต้องใช้ 2
1.6 ขั้นตอนการดำเนินงาน 3
บทที่ 2 ทฤษฎีพื้นฐานที่ใช้ 4
2.1 แอนดรอยด์ (Android) 4
2.1.1 พัฒนาการของแอนดรอยด์ 4
2.1.2 ประเภทของระบบปฏิบัติการแอนดรอยด์ 7
2.1.3 สถาปัตยกรรมแอนดรอยด์ 7
2.1.4 ส่วนประกอบของแอปพลิเคชั่น (Application Component) 10
2.2 NFC (Near Field Communication) 11
2.2.1 หลักการทำงานของ NFC 11
2.2.2 การใช้งานเทคโนโลยี NFC 13
2.2.3 สถาปัตยกรรม NFC 14
2.2.4 ประโยชน์ของ NFC 14
2.3 โปรแกรมอีคลิป (Eclipse) 16
2.3.1 ประวัติ 16
2.3.2 ส่วนประกอบของโปรแกรมอีคลิป 17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.3.3 สถาปัตยกรรมของโปรแกรมอีคลิป 18
2.4 จาวา (Java) 19
2.4.1 ข้อดีของภาษาจาวา 19
2.5 กราฟ (Graph) 20
2.5.1 ประเภทของกราฟ 20
2.5.2 ขั้นตอนวิธีการทำงานของ Dijkstra's algorithm 21
บทที่ 3 การออกแบบ 25
3.1 การออกแบบ 25
3.1.1 สมาร์ตโฟนสัมผัสกับป้าย NFC 26
3.1.2 สมาร์ตโฟนได้รับข้อมูลจากป้าย NFC นั้น 27
3.1.3 การส่งข้อมูลที่ได้รับไปยังส่วนประมวลผล 28
3.1.4 การแสดงตำแหน่งปัจจุบันของผู้ใช้งานให้ตรงกับข้อมูลที่ได้รับจากป้าย NFC 28
3.1.5 การแสดงเส้นทางที่สั้นที่สุดซึ่งคำนวณโดย Dijkstra's algorithm 32
3.2 เครื่องมือที่ใช้ในการทดลอง 34
3.2.1 โทรศัพท์มือถือ 34
3.2.2 ป้าย NFC 34
3.2.3 โปรแกรมอีคลิป (Eclipse) 35
3.2.4 โปรแกรม TagWriter 35
บทที่ 4 ผลการทดลอง 36
4.1 ผลการทดสอบสมาร์ตโฟนสัมผัสกับป้าย NFC 36
4.2 ผลการทดสอบเมื่อสมาร์ตโฟนได้รับข้อมูลจากป้าย NFC 37
4.3 ผลการทดสอบการส่งข้อมูลที่ได้รับไปยังส่วนประมวลผล 38
4.4 ผลการทดสอบการแสดงตำแหน่งปัจจุบันของผู้ใช้งานให้ตรงกับข้อมูลที่ได้รับจากป้าย NFC 39
4.5 ผลการทดสอบการแสดงเส้นทางที่สั้นที่สุดซึ่งคำนวณโดย Dijkstra's algorithm 42
บทที่ 5 สรุปผล 46
5.1 สรุปผล 46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

5.2 ข้อเสนอแนะ 47

เอกสารอ้างอิง 48

ภาคผนวก ก โค้ดของโทรศัพท์แอนดรอยด์ 51

ภาคผนวก ข โค้ดของ Dijkstra's algorithm 66

ภาคผนวก ค Poster 71

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 1.1 ขั้นตอนการดำเนินงานของโครงงาน 3
ตารางที่ 2.1 พัฒนาการของแอนดรอยด์แต่ละรุ่น 4
ตารางที่ 2.2 รายละเอียดของเทคโนโลยี NFC ที่รองรับ RFID 11
ตารางที่ 2.3 ความแตกต่างระหว่าง NFC และ Bluetooth 14

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 สถาปัตยกรรมแอนดรอยด์ 8
รูปที่ 2.2 การทำงานของแอปพลิเคชั่นที่มีมากกว่า 1 Activity 10
รูปที่ 2.3 การใช้ประโยชน์ของ NFC 15
รูปที่ 2.4 สัญลักษณ์ของโปรแกรมอีคลิป (Eclipse) 16
รูปที่ 2.5 ส่วนประกอบของโปรแกรมอีคลิป 17
รูปที่ 2.6 สถาปัตยกรรมของโปรแกรมอีคลิป 18
รูปที่ 2.7 กราฟแสดงทิศทาง 20
รูปที่ 2.8 กราฟไม่แสดงทิศทาง 21
รูปที่ 2.9 กราฟที่มีต้นทางและปลายทางเป็นโหนดเดียวกัน 21
รูปที่ 2.10 ตัวอย่างการใช้งานของ Dijkstra's algorithm 22
รูปที่ 3.1 แผนผังการทำงานของระบบ 25
รูปที่ 3.2 Flowchart ตรวจสอบการใช้งาน NFC 26
รูปที่ 3.3 หน้า User Interface เมื่อเข้าแอปพลิเคชั่น 27
รูปที่ 3.4 โค้ด split ข้อมูลเพื่อเลือกเฉพาะข้อมูลที่ต้องการ 27
รูปที่ 3.5 โค้ดการส่งข้อมูลไปยังส่วนประมวลผล 28
รูปที่ 3.6 User Interface หลังส่งข้อมูล 28
รูปที่ 3.7 Flowchart การแสดงตำแหน่งปัจจุบันของผู้ใช้งาน 29
รูปที่ 3.8 หมุดสีแดงใช้เพื่อแสดงตำแหน่งปัจจุบันของผู้ใช้งาน 29
รูปที่ 3.9 โค้ดแสดงฟังก์ชัน hideallimage 30
รูปที่ 3.10 User Interface หลังจากใช้ฟังก์ชัน hideallimage 30
รูปที่ 3.11 โค้ดแสดงฟังก์ชัน showimage 30
รูปที่ 3.12 User Interface หลังจากใช้ฟังก์ชัน showimage 31
รูปที่ 3.13 โค้ดแสดงฟังก์ชัน showdialog 31
รูปที่ 3.14 Flowchart การแสดงเส้นทางที่สั้นที่สุด 32
รูปที่ 3.15 กราฟโหนดและเส้นเชื่อมของแผนที่อาคารเรียนรวม 12 ชั้น ที่ชั้น 11 33
รูปที่ 3.16 กราฟชื่อของโหนดทั้งหมด 33
รูปที่ 3.17 โทรศัพท์มือถือยี่ห้อ Sony Xperia Z 34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 3.18 บัตร Mifare Classic ความจุ 1 KB 34
รูปที่ 3.19 สัญลักษณ์ของโปรแกรมอีคลิป (Eclipse) 35
รูปที่ 3.20 User Interface ของแอปพลิเคชั่น TagWriter 35
รูปที่ 4.1 User Interface หน้า Wireless & networks 36
รูปที่ 4.2 หน้า User Interface เมื่อเข้าแอปพลิเคชั่น (และมีการเปิดใช้งาน NFC แล้ว) 37
รูปที่ 4.3 ผลลัพธ์ทั้งหมดเมื่อสมาร์ตโฟนสัมผัสป้าย NFC 37
รูปที่ 4.4 ผลลัพธ์สุดท้ายที่ได้เมื่อทำการ split แล้ว 37
รูปที่ 4.5 โค้ดการส่งข้อมูลไปยังส่วนประมวลผล 38
รูปที่ 4.6 โค้ดการเรียกใช้ข้อมูลที่ส่งมาจากส่วนการอ่านข้อมูล 38
รูปที่ 4.7 ผลลัพธ์ของส่วนประมวลผลที่ได้รับข้อมูลจากส่วนการอ่านข้อมูล 38
รูปที่ 4.8 ผลลัพธ์เมื่อสมาร์ตโฟนสัมผัสป้าย NFC หน้าลิฟต์ 39
รูปที่ 4.9 User Interface เมื่อสัมผัสป้าย NFC ที่หน้าลิฟต์ 39
รูปที่ 4.10 ผลลัพธ์เมื่อสมาร์ตโฟนสัมผัสป้าย NFC หน้าห้อง E12-1106 40
รูปที่ 4.11 User Interface เมื่อสัมผัสป้าย NFC ที่หน้าห้อง E12-1106 40
รูปที่ 4.12 User Interface แสดงรายละเอียดเมื่อสัมผัสป้าย NFC ที่หน้าลิฟต์ 41
รูปที่ 4.13 User Interface แสดงรายละเอียดเมื่อสัมผัสป้าย NFC ที่หน้าห้อง E12-1106 41
รูปที่ 4.14 User Interface แสดงการเลือกปลายทางจาก dropdown 42
รูปที่ 4.15 โค้ดการเปลี่ยนข้อมูลรูปแบบ JSON ให้เป็นข้อมูลในรูปของตัวแปร string 42
รูปที่ 4.16 กราฟแสดงโหนดตั้งแต่ E12-1106 ถึง toilet-female 43
รูปที่ 4.17 กราฟแสดงเส้นทางตั้งแต่ E12-1106 ถึง toilet-female 43
รูปที่ 4.18 ผลลัพธ์การเปลี่ยนข้อมูลรูปแบบ JSON ให้เป็นข้อมูลในรูปของตัวแปร string 43
รูปที่ 4.19 กราฟแสดงเส้นทางที่ไปได้ทั้งหมด 3 เส้นทาง 44
รูปที่ 4.20 โค้ดแสดงการทำงานการรับตำแหน่งจาก Array 44
รูปที่ 4.21 User Interface แสดงเส้นทางจาก E12-1106 ถึง toilet-female 45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญ

เทคโนโลยีโทรศัพท์มือถือมีการพัฒนาอย่างรวดเร็ว โดยจะพบว่าโทรศัพท์มือถือที่สามารถอำนวยความสะดวกและตอบสนองความต้องการในชีวิตประจำวันหรือที่เรียกว่าสมาร์ตโฟน (Smartphone) นั้น มีการใช้งานอย่างแพร่หลาย โดยสมาร์ตโฟนสามารถพัฒนาให้ใช้งานได้ตรงความต้องการมากขึ้น หนึ่งในแอปพลิเคชั่น (Application) ที่ผู้คนส่วนใหญ่ใช้เป็นจำนวนมาก คือ แอปพลิเคชั่นที่มีความสามารถในการระบุตำแหน่ง โดยอาศัยเทคโนโลยี GPS (Global Positioning System) ซึ่งสามารถทำงานได้ดีในสภาพแวดล้อมที่เป็นที่โล่ง แต่ถ้าเป็นภายในอาคารจะไม่สามารถทำงานได้อย่างมีประสิทธิภาพ

ระบบแผนที่ภายในอาคารในปัจจุบันยังไม่มีการใช้งานกันอย่างกว้างขวาง โดยระบบนี้จะใช้เทคโนโลยีที่แตกต่างกันออกไปแล้วแต่สถานที่และองค์ประกอบต่างๆ ตัวอย่างเช่น อาจจะใช้เทคโนโลยีสัญญาณวายฟาย (Wi-Fi) ภายในอาคารที่มีการติดตั้งเซ็นเซอร์ (Sensor) สำหรับตรวจวัดพิกัดข้อมูล ใช้เทคโนโลยีสนามแม่เหล็กเป็นตัวนำทางโดยถือสมาร์ตโฟนที่จะทำการบันทึกค่าสนามแม่เหล็กไปตลอดเส้นทางเดิน หรือเทคโนโลยี NFC (Near Field Communication) เป็นต้น และเนื่องจาก สมาร์ตโฟนในปัจจุบันได้มีการติดตั้ง NFC บนโทรศัพท์มือถือ ซึ่งเป็นระบบสื่อสารไร้สายระยะใกล้ที่กำลังได้รับความนิยมมาก เพราะสามารถเชื่อมต่อและแลกเปลี่ยนข้อมูลระหว่างอุปกรณ์ได้อย่างรวดเร็วและปลอดภัยเพียงแค่นำอุปกรณ์มาสัมผัสกัน อีกทั้งยังไม่ได้รับผลกระทบจากสัญญาณรบกวนจากระบบอื่น โดยการทำงานที่รองรับการสัมผัสกันแบบนี้ ทำให้ NFC ถูกนำไปประยุกต์ใช้งานได้หลากหลาย

ในโครงงานนี้จึงนำเสนอระบบแผนที่ภายในอาคาร เช่น ศูนย์ประชุมหรือห้างสรรพสินค้า โดยใช้เทคโนโลยี NFC เพื่อใช้ในการระบุตำแหน่งปัจจุบันของผู้ใช้งาน และยังสะดวกในการบอกรายละเอียดของร้านค้า ทำให้ลดภาระของพนักงานอีกด้วย โดยแอปพลิเคชั่นสามารถแสดงแผนที่ภายในอาคารและผู้ใช้งานสามารถเลือกตำแหน่งปลายทางที่ต้องการไปได้ จากนั้นแอปพลิเคชั่นจะแสดงเส้นทางที่สั้นที่สุดตั้งแต่ต้นทางไปปลายทางให้ผู้ใช้งานซึ่งคำนวณด้วยเทคนิคกราฟ ดังนั้นแอปพลิเคชั่นนี้ช่วยอำนวยความสะดวกให้ผู้ใช้งานในการนำทางสำหรับผู้ที่ไม่เคยไปสถานที่นั้นๆให้ถึงเป้าหมายได้อย่างถูกต้องและรวดเร็ว โดยแอปพลิเคชั่นสามารถนำไปประยุกต์ให้ใช้งานได้อย่างหลากหลายยิ่งขึ้นไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 จุดประสงค์

- เพื่อศึกษาการเขียนแอปพลิเคชั่นบนแอนดรอยด์ (Android) ให้สามารถแสดงแผนที่ภายในอาคาร และทราบตำแหน่งปัจจุบันของผู้ใช้งานจากการสัมผัสป้าย NFC ได้
- เพื่อศึกษาการหาเส้นทางที่ดีที่สุดของเป้าหมายที่ผู้ใช้งานต้องการโดยการใช้กราฟ
- เพื่อพัฒนาระบบแผนที่ภายในอาคารโดยใช้เทคโนโลยี NFC ทำให้มีการใช้งานที่ง่ายขึ้น ประหยัดค่าใช้จ่าย และสามารถบอกตำแหน่งแต่ละจุดของผู้ใช้งานได้อย่างแม่นยำ ซึ่งเป็นการทดแทนข้อจำกัดของเทคโนโลยี GPS ที่ไม่แม่นยำเมื่อใช้งานภายในอาคาร

1.3 ขอบเขตของโครงงาน

- สร้างแอปพลิเคชั่นบนแอนดรอยด์ ที่สามารถแสดงแผนที่ภายในอาคารได้โดยสัมผัสป้าย NFC
- สามารถแสดงตำแหน่งปัจจุบันของผู้ใช้งานได้
- สามารถแสดงเส้นทางที่สั้นที่สุดเพื่อบอกเป้าหมายที่ผู้ใช้งานต้องการไป
- สามารถแสดงรายละเอียดของร้านค้าต่างๆที่อยู่ภายในอาคารได้โดยการสัมผัสป้าย NFC
- สมาร์ตโฟนสามารถอ่านป้าย NFC ได้

1.4 ผลที่คาดว่าจะได้รับ

- ระบบที่พัฒนาขึ้นสามารถนำไปใช้งานในระบบแผนที่ภายในอาคารได้ง่ายขึ้น สะดวกรวดเร็วต่อการค้นหาจุดหมายปลายทางที่ต้องการไป และเป็นระบบที่สามารถบำรุงรักษาได้ง่าย ประหยัดค่าใช้จ่าย
- เป็นระบบแผนที่ภายในอาคารที่บอกตำแหน่งปัจจุบันของผู้ใช้งานได้แม่นยำและถูกต้อง

1.5 อุปกรณ์ที่ต้องใช้

ฮาร์ดแวร์ (Hardware)

- สมาร์ตโฟนระบบปฏิบัติการแอนดรอยด์
- ป้าย NFC

ซอฟต์แวร์ (Software)

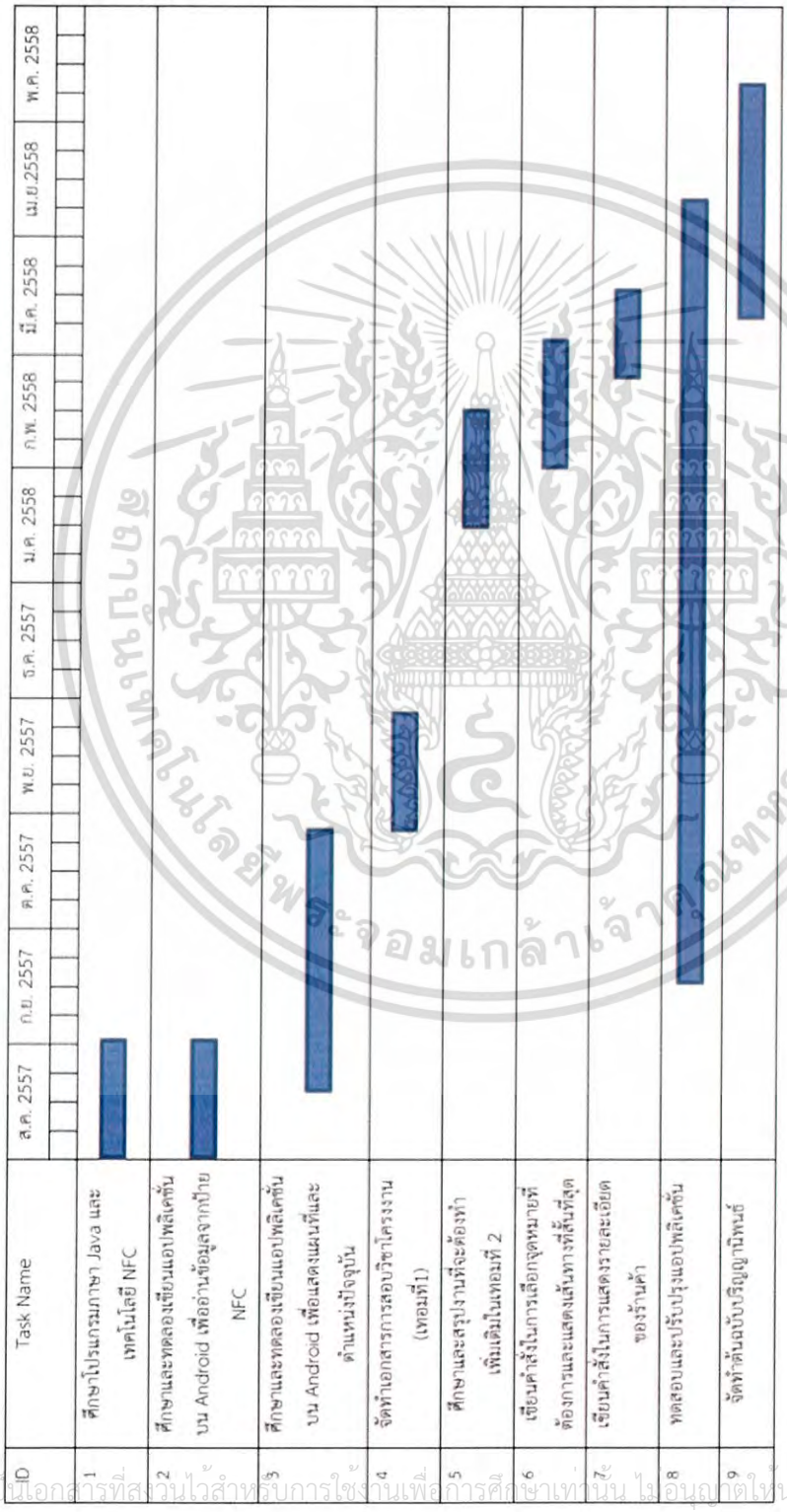
- โปรแกรมอีคลิป (Eclipse)
- โปรแกรม TagWriter
- เซิร์ฟเวอร์ (Server)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 ขั้นตอนการดำเนินงาน

ตารางเวลาในการดำเนินโครงงาน ตั้งแต่วันที่ 14 สิงหาคม 2557 ถึงวันที่ 30 พฤษภาคม 2558 ดังแสดงในตารางที่ 1.1

ตารางที่ 1.1 ขั้นตอนการดำเนินงานของโครงงาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีพื้นฐานที่ใช้

ในหัวข้อนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ ที่เกี่ยวข้องในการทำโครงงาน ซึ่งจะอธิบายดังต่อไป

2.1 แอนดรอยด์ (Android)

แอนดรอยด์ เป็นระบบปฏิบัติการที่มีพื้นฐานอยู่บนระบบปฏิบัติการลินุกซ์ (Linux) คิดค้นและพัฒนาโดยบริษัท แอนดรอยด์ (Android, Inc.) ซึ่งต่อมา กูเกิล (Google) ได้ทำการซื้อต่อบริษัทในปี พ.ศ. 2548 แอนดรอยด์ถูกเปิดตัวเมื่อ ปี พ.ศ. 2550 แอนดรอยด์เป็นระบบปฏิบัติการโอเพนซอร์ส (Open Source) และกูเกิลได้เผยแพร่ภายใต้ลิขสิทธิ์อาปาเช (Apache) ซึ่งอนุญาตให้ผู้ผลิตปรับแต่งและวางจำหน่ายได้ รวมไปถึงนักพัฒนาและผู้ให้บริการเครือข่ายด้วย อีกทั้งแอนดรอยด์ยังเป็นระบบปฏิบัติการที่รวมนักพัฒนาที่เขียนโปรแกรมประยุกต์ ภายใต้ภาษาจาวา (Java)

2.1.1 พัฒนาการของแอนดรอยด์

ประวัติรุ่นของแอนดรอยด์ เริ่มตั้งแต่การเผยแพร่รุ่นทดลองในเดือนพฤศจิกายน พ.ศ.2550 และได้มีการเปิดตัวแอนดรอยด์ 1.0 ในเดือนกันยายน พ.ศ.2551 ภายใต้การพัฒนาอย่างต่อเนื่องของกูเกิลและโอเพนแฮนด์เซตอัลไลแอนซ์ (Open Handset Alliances) นับตั้งแต่เดือนเมษายน พ.ศ. 2552 รุ่นของแอนดรอยด์พัฒนาภายใต้โค้ดเนม (Code Name) และเผยแพร่โดยเรียงตามตัวอักษรคือ Cupcake (1.5), Donut (1.6), Éclair (2.0–2.1), Froyo (2.2–2.2.3), Gingerbread (2.3–2.3.7), Honeycomb (3.0–3.2.6), Ice Cream Sandwich (4.0–4.0.4), Jelly Bean (4.1–4.3) และ KitKat (4.4) ในวันที่ 3 กันยายน 2556 กูเกิลได้ยืนยันว่ามีอุปกรณ์ที่ใช้ระบบปฏิบัติการแอนดรอยด์เปิดใช้งาน 1 พันล้านเครื่องทั่วโลก รุ่นล่าสุดที่ได้รับการอัปเดตคือ 5.0 Lollipop API (Application Programming Interface) 24 ซึ่งได้ปล่อยการอัปเดตในวันที่ 15 ตุลาคม 2557 ในตารางที่ 2.1 แสดงพัฒนาการของแอนดรอยด์ตั้งแต่รุ่น 4.0-4.4

ตารางที่ 2.1 พัฒนาการของแอนดรอยด์แต่ละรุ่น [2]

Platform	Code Name	API Level	Features
Android 4.0-4.0.2	ไอศกรีมแซนวิช (Ice Cream Sandwich)	14	1. ปลดล็อกหน้าจอด้วยใบหน้า ซึ่งเป็นคุณลักษณะของซอฟต์แวร์ที่จะจดจำใบหน้าของผู้ใช้และผู้ใช้สามารถปลดล็อกซอฟต์แวร์ได้ด้วยใบหน้าของตนเองจากกล้องหน้าของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 พัฒนาการของแอนดรอยด์แต่ละรุ่น [2] (ต่อ)

Platform	Code name	API Level	Features
			2. ส่วนของการใช้งานข้อมูลในการตั้งค่า สามารถให้ผู้ใช้ตั้งการแจ้งเตือนเมื่อใกล้ถึงขีดจำกัดของการใช้ข้อมูล และสามารถปิดการใช้ข้อมูล เมื่อมีการใช้ข้อมูลถึงขีดจำกัด 3. แอนดรอยด์ บีม ซึ่งเป็นคุณสมบัติการใช้ NFC สำหรับการแบ่งปันข้อมูลในระยะใกล้ 4. รองรับ WebP 5. ไว-ไฟ ไดเรกต์ (Wi-Fi Direct)
Android 4.1	เจลลี่บีน (Jelly Bean)	16	1. ส่วนต่อกราฟิก (Graphic) กับผู้ใช้ที่ลื่นไหลขึ้น ประกอบด้วย : - V-sync เป็นการกำหนดภาพเคลื่อนไหวโดยแอนดรอยด์ เฟรมเวิร์ก (Framework) รวมไปด้วยการใช้งานองค์ประกอบของหน้าจอ - ทริเปิลบัฟเฟอริง (Triple buffering) ในกราฟิก 2. ความสามารถในการปิดการแจ้งเตือนของแอปพลิเคชั่นแบบเฉพาะเจาะจง 3. ทางลัดและวิดเจ็ต (Widget) มีความสามารถในการจัดวางและปรับขนาดเพื่อให้รายการใหม่พอดีกับหน้าจอหลัก 4. การแบ่งปันข้อมูลผ่านบลูทูธ (Bluetooth) สำหรับแอนดรอยด์ บีม 5. การเชื่อมต่อเสียงจากยูเอสบี (USB) 6. เบราว์เซอร์ (Browser) ในตัวเครื่องถูกแทนที่ด้วยกูเกิล โครม ในอุปกรณ์เจลลีบีน
Android 4.2	เจลลี่บีน (Jelly Bean)	17	1. คีย์บอร์ด (Keyboard) ที่มาพร้อมกับการเดาท่าทางการพิมพ์ 2. การปรับปรุงหน้าจอล็อก รวมไปด้วยการรองรับวิดเจ็ตและความสามารถในการเลื่อนเพื่อเปิดกล้องอย่างรวดเร็ว 3. รองรับ Wireless Display

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 พัฒนาการของแอนดรอยด์แต่ละรุ่น [2] (ต่อ)

Platform	Code name	API Level	Features
			4. การปรับปรุงการเข้าถึง รวมไปด้วยการแตะ 3 ครั้ง เพื่อขยายหน้าจอ, การกวาดมือเพื่อขยายด้วยการใช้ 2 นิ้ว, การโต้ตอบด้วยเสียงและคุณสมบัติสำหรับบุคคลที่พิการทางสายตา 5. เพิ่มการขยายการแจ้งเตือน และ การแจ้งเตือนสำหรับแอปพลิเคชั่นเพิ่มเติม เพื่อให้ผู้ใช้สามารถตอบสนองได้จากการแจ้งเตือน โดยไม่ต้องเข้าแอปพลิเคชั่นโดยตรง 6. การส่งข้อความแบบกลุ่ม
Android 4.3	เจลลี่บีน (Jelly Bean)	18	1. รองรับบลูทูธแบบประหยัดพลังงาน 2. รองรับ AVRCP 1.3 3. คุณสมบัติการจำกัดการเข้าถึงสำหรับผู้ใช้ใหม่ 4. การปรับปรุงการรักษาความปลอดภัยจำนวนมาก, การปรับปรุงประสิทธิภาพการทำงาน และการแก้ไขข้อผิดพลาดต่างๆ 5. รองรับจีโอเฟนซ์ (Geo-fence) และ การค้นหาด้วยวายฟาย 6. การเข้าสู่ระบบของนักพัฒนา และ การวิเคราะห์การปรับปรุง 7. รองรับอีก 5 ภาษา
Android 4.4	คิทแคท (Kit Kat)	19	1. ปรับปรุงอินเตอร์เฟส (Interface) ใหม่พร้อมกับแถบสถานะที่มีไอคอนและตัวอักษรสีขาว, ความสามารถสำหรับแอปพลิเคชั่นที่จะใช้งานและสถานะในแบบโปร่งใส 2. ความสามารถในการสั่งพิมพ์แบบไร้สาย 3. การใช้ NFC เป็นเสมือนบัตรสมาร์ตการ์ด (Smart Card) 4. เฟรมเวิร์กใหม่สำหรับการเปลี่ยนแปลงอินเตอร์เฟส 5. เอพีไอใหม่สำหรับเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 ประเภทของระบบปฏิบัติการแอนดรอยด์

ระบบปฏิบัติการแอนดรอยด์ เป็นซอฟแวร์ระบบเปิด จึงอนุญาตให้นักพัฒนาหรือผู้ที่สนใจสามารถดาวน์โหลด (Download) ซอร์สโค้ด (Source Code) ได้ จึงทำให้ผู้พัฒนาหลายๆฝ่าย นำซอร์สโค้ดมาปรับแต่งและพัฒนาสร้างแอปพลิเคชั่นบนระบบแอนดรอยด์ในฉบับของตนเองมากขึ้น โดยสามารถแบ่งระบบปฏิบัติการของแอนดรอยด์ ออกเป็น 3 กลุ่มประเภทใหญ่ๆ ดังต่อไปนี้

2.1.2.1 Android Opensource Project (AOSP)

เป็นระบบปฏิบัติการแรกที่กูเกิล เปิดให้สามารถดาวน์โหลดซอร์สโค้ดไปติดตั้งและใช้งานในอุปกรณ์ต่างๆได้โดยไม่เสียค่าใช้จ่าย

2.1.2.2 Open Handset Mobile (OHM)

เป็นแอนดรอยด์ที่ได้รับการพัฒนาร่วมกับ Open Handset Alliances (OHA) ซึ่งบริษัทเหล่านี้จะพัฒนาระบบแอนดรอยด์ในแบบฉบับของตนเอง โดยมีรูปร่างหน้าตาการแสดงผลที่แตกต่างกันรวมไปถึงอาจจะมีเอกลักษณ์และรูปแบบการใช้งานเป็นของแต่ละบริษัท และโปรแกรมแอนดรอยด์ประเภทนี้ก็จะได้รับสิทธิบริการเสริมต่างๆจากกูเกิล ที่เรียกว่า GMS (Google Mobile Service) ซึ่งเป็นบริการเสริมที่ทำให้แอนดรอยด์มีประสิทธิภาพมากขึ้น

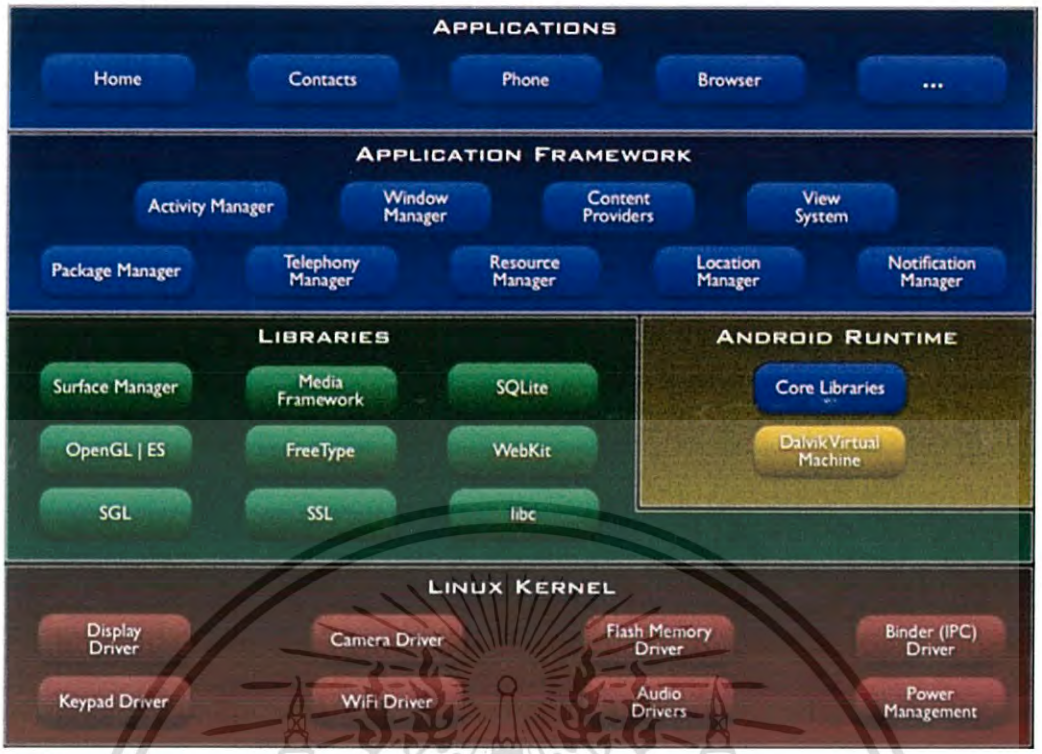
2.1.2.3 Cooking or Customize

เป็นระบบแอนดรอยด์ที่นักพัฒนานำเอาซอร์สโค้ด จากแหล่งต่างๆมาปรับแต่งให้อยู่ในแบบฉบับของตนเอง ซึ่งการพัฒนาต้องปลดล็อคสิทธิในการใช้งานอุปกรณ์เสียก่อนจึงจะสามารถติดตั้งได้ ทั้งนี้ระบบแอนดรอยด์ประเภทนี้ถือเป็นประเภทที่มีความสามารถสูงสุด เนื่องจากได้รับการปรับแต่งขีดความสามารถต่างๆ ให้มีความเข้ากันได้กับอุปกรณ์นั้นๆ จากผู้ใช้งานจริง

2.1.3 สถาปัตยกรรมแอนดรอยด์

แอนดรอยด์เป็นซอฟต์แวร์ที่มีโครงสร้างแบบเรียงทับซ้อนหรือแบบสแต็ก (Stack) ซึ่งรวมเอาระบบปฏิบัติการ (Operating System), มิดเดิลแวร์ (Middleware) และแอปพลิเคชั่นที่สำคัญเข้าไว้ด้วยกัน เพื่อใช้สำหรับทำงานบนอุปกรณ์พกพาเคลื่อนที่ เช่น โทรศัพท์มือถือ เป็นต้น การทำงานของแอนดรอยด์มีพื้นฐานอยู่บนระบบลีนุกซ์ เคอร์เนล (Linux Kernel) ซึ่งใช้ Android SDK (Software Development Kit) เป็นเครื่องมือสำหรับการพัฒนาแอปพลิเคชั่นบนระบบปฏิบัติการแอนดรอยด์ และใช้ภาษาจาวาในการพัฒนา สถาปัตยกรรมของแอนดรอยด์ (Android Architecture) นั้นถูกแบ่งออกเป็นลำดับชั้น ออกเป็น 4 ชั้นหลักดังแสดงในรูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 สถาปัตยกรรมแอนดรอยด์ [4]

2.1.3.1 ชั้นแอปพลิเคชั่น

ชั้นนี้จะเป็นชั้นที่อยู่บนสุดของโครงสร้างสถาปัตยกรรมแอนดรอยด์ ซึ่งเป็นส่วนของแอปพลิเคชั่นที่พัฒนาขึ้นมาใช้งาน เช่น แอปพลิเคชั่นรับ/ส่งอีเมล์, ปฏิทิน, แผนที่, เว็บเบราเซอร์ (Web Browser), รายชื่อผู้ติดต่อ เป็นต้น ซึ่งแอปพลิเคชั่นจะอยู่ในรูปแบบของไฟล์ ".apk" โดยทั่วไปแล้วจะอยู่ในไดเร็คทอรี (Directory) data/app (รูปตัวอย่างของแอปพลิเคชั่น)

2.1.3.2 ชั้นแอปพลิเคชั่นเฟรมเวิร์ค (Application Framework)

ในชั้นนี้จะอนุญาตให้นักพัฒนาสามารถเข้าเรียกใช้งาน โดยผ่าน API ซึ่งแอนดรอยด์ได้ออกแบบไว้เพื่อลดความซ้ำซ้อนในการใช้งานส่วนประกอบของแอปพลิเคชั่น

โดยในชั้นนี้ประกอบด้วยแอปพลิเคชั่นเฟรมเวิร์คดังนี้

- View System : เป็นส่วนที่ใช้ในการควบคุมการทำงานสำหรับการสร้างแอปพลิเคชั่น
- Location Manager : เป็นส่วนที่จัดการเกี่ยวกับค่าตำแหน่งของเครื่องอุปกรณ์พกพาเคลื่อนที่
- Content Provider : เป็นส่วนที่ใช้ควบคุมการเข้าถึงข้อมูลที่มีการใช้งานร่วมกันระหว่างแอปพลิเคชั่นที่แตกต่างกัน เช่น ข้อมูลผู้ติดต่อ
- Resource Manager : เป็นส่วนที่จัดการข้อมูลต่างๆ ที่ไม่ใช่ส่วนของโค้ด (Code) โปรแกรม เช่น รูปภาพ, Localized Strings, Layout ซึ่งจะอยู่ในไดเร็คทอรี res

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Notification Manager : เป็นส่วนที่ควบคุมอีเวนต์ (Event) ต่างๆ ที่แสดงบนแถบสถานะเช่น ในกรณีที่ได้รับข้อความหรือสายที่ไม่ได้รับและการแจ้งเตือนอื่นๆ เป็นต้น
- Activity Manager : เป็นส่วนควบคุม Life Cycle ของแอปพลิเคชั่น

2.1.3.3 ชั้นไลบรารี (Library)

แอนดรอยด์ได้รวบรวมกลุ่มของไลบรารีต่างๆ ที่สำคัญและมีความจำเป็นเอาไว้มากมาย เพื่ออำนวยความสะดวกให้กับนักพัฒนาและง่ายต่อการพัฒนาโปรแกรม โดยตัวอย่างของไลบรารีที่สำคัญเช่น

- System C library : เป็นกลุ่มของไลบรารีมาตรฐานที่อยู่บนพื้นฐานของภาษา C ไลบรารี สำหรับ Embedded System ที่มีพื้นฐานมาจากลินุกซ์
- Media Libraries : เป็นกลุ่มการทำงานมัลติมีเดีย เช่น MPEG4, H.264, MP3, AAC, AMR, JPG, และ PNG
- Surface Manager : เป็นกลุ่มการจัดการรูปแบบหน้าจอ การวาดหน้าจอ
- 2D/3D library : เป็นกลุ่มของกราฟิกแบบ 2 มิติ หรือ SGL (Scalable Graphics Library) และแบบ 3 มิติ หรือ OpenGL
- FreeType : เป็นกลุ่มของบิตแมป (Bitmap) และเวคเตอร์ (Vector) สำหรับการเรนเดอร์ (Render) ภาพ
- SQLite : เป็นกลุ่มของฐานข้อมูล โดยนักพัฒนาสามารถใช้ฐานข้อมูลนี้เก็บข้อมูลแอปพลิเคชั่นต่างๆ ได้
- Browser Engine : เป็นกลุ่มของการแสดงผลบนเว็บเบราเซอร์โดยอยู่บนพื้นฐานของ Webkit ซึ่งจะมีลักษณะคล้ายกับกูเกิลโครม (Google Chrome)

2.1.3.4 ชั้นลีนุกซ์เคอร์เนล

ระบบแอรดรอยด์ นั้นถูกสร้างบนพื้นฐานของระบบปฏิบัติการลินุกซ์ โดยในชั้นนี้จะมีฟังก์ชัน (Function) การทำงานหลายๆ ส่วน แต่โดยส่วนมากแล้วจะเกี่ยวข้องกับฮาร์ดแวร์โดยตรง เช่น การจัดการหน่วยความจำ (Memory Management) การจัดการโพรเซส (Process Management) การเชื่อมต่อเครือข่าย (Networking) เป็นต้น

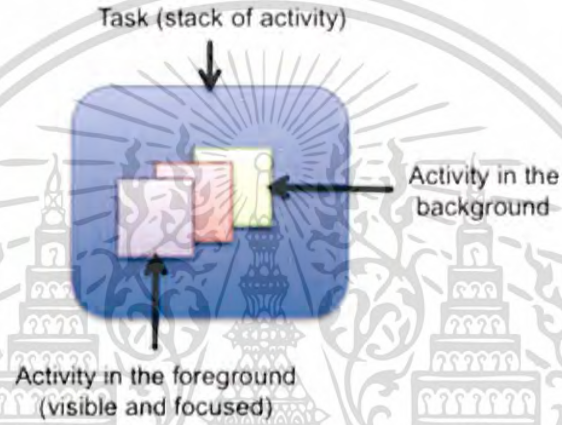
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 ส่วนประกอบของแอปพลิเคชั่น (Application Component)

ส่วนประกอบของแอปพลิเคชั่นของแอนดรอยด์สามารถแบ่งออกเป็น 4 ส่วนคือ Activity, Service, Broadcast Receiver and Intent Receiver และ Content Provider

2.1.4.1 Activity

คือหน้าจอที่ติดต่อกับผู้ใช้ ทั้งนี้ในแต่ละแอปพลิเคชั่นอาจจะมีมากกว่า 1 หน้าจอ หรือ 1 Activity ซึ่งแต่ละ Activity จะทำหน้าที่เก็บสถานะ การใช้งานในส่วนต่างๆ ดังแสดงในรูปที่ 2.2 ตัวอย่างเช่น สำหรับแอปพลิเคชั่นส่งข้อความอาจจะมี Activity หนึ่งที่แสดงรายการของส่วนติดต่อในการส่งข้อความ และอีก Activity หนึ่งจะเป็นส่วนของการเลือกการติดต่อ เป็นต้น



รูปที่ 2.2 การทำงานของแอปพลิเคชั่นที่มีมากกว่า 1 Activity [5]

2.1.4.2 Service

คือส่วนการทำงานที่ไม่มีหน้าจอที่ติดต่อกับผู้ใช้ service นั้นจะทำงานอยู่ในส่วนของ Background เช่น โปรแกรมเล่นเพลงต่างๆ ก็จะมีหน้าจอที่ติดต่อกับผู้ใช้ (นั้นคือส่วนของ Activity) ไฟล์เพลงก็จะถูกเล่นโดยมีการทำงานแบบ service หลักจากผู้ใช้กด Back หรือ Home หน้าจอของเครื่องเล่นเพลง ก็จะถูกเก็บไป แต่ในส่วนของ service ที่เล่นเพลงนั้นก็ยังคงเล่นเพลงต่อไป

2.1.4.3 Broadcast Receiver and Intent Receiver

คือ การตอบสนองซึ่งโดยปกติแล้ว Broadcast Receiver จะเป็นการตอบสนองต่อการเกิดอีเวนต์ของระบบในวงกว้าง นอกจากนี้ Intent Receiver เป็นส่วนทำให้แอปพลิเคชั่นอื่นๆ เข้าถึงการทำงาของ Activity และ Service

2.1.4.4 Content Provider

คือส่วนของการให้บริการข้อมูลสำหรับแต่ละแอปพลิเคชั่น ทั้งนี้ข้อมูลสามารถเก็บอยู่ในรูปแบบของระบบไฟล์ หรือฐานข้อมูลก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 NFC (Near Field Communication)

NFC เป็นเทคโนโลยีสื่อสารไร้สายระยะสั้นระยะประมาณ 4 เซนติเมตร ที่ใช้ได้ดีกับโครงสร้างพื้นฐานแบบไร้สัมผัส ช่วยสนับสนุนรองรับการสื่อสารระหว่างเครื่องมืออิเล็กทรอนิกส์ในระยะใกล้ๆ NFC ถูกพัฒนาขึ้นโดย Sony และ NXP โดยใช้คลื่นความถี่ 13.56 MHz บนพื้นฐานมาตรฐาน ISO 14443 (Philips MIFARE and Sony's FeliCa) ปัจจุบันบริษัททั้งสองได้ร่วมมือกับบริษัทผู้ผลิตและพัฒนาโทรศัพท์เคลื่อนที่ จัดตั้งเป็น NFC Forum เพื่อให้เกิดการใช้งานในรูปแบบต่างๆมากขึ้น ในระยะเริ่มแรกมีบริษัทโทรศัพท์มือถือชั้นนำของโลกประกาศนำเทคโนโลยีนี้มา ใช้กับโทรศัพท์มือถือแล้ว เช่น Nokia, Samsung, Motorola เป็นต้น ซึ่งเทคโนโลยี NFC จะรองรับกับเทคโนโลยีอาร์เอฟไอดี (Radio-frequency identification) ตามรายละเอียดดังตารางที่ 2.2

ตารางที่ 2.2 รายละเอียดของเทคโนโลยี NFC ที่รองรับอาร์เอฟไอดี [6]

NFC Forum Platform	RFID Compatible
NFC Forum Type 1 tag	Innovision Topaz
NFC Forum Type 2 tag	Mifare Ultralight Mifare Ultralight C
NFC Forum Type 3 tag	Sony Felica
NFC Forum Type 4 tag	DESfire SmartMX

การประยุกต์ใช้งานส่วนใหญ่มักนำ NFC มาใช้กับการชำระเงินที่ต้องการความรวดเร็วและมีมูลค่าไม่สูง ซึ่งจะทำให้ โทรศัพท์เคลื่อนที่ สามารถใช้เพื่อการชำระเงิน โดยวิธีการแตะบนเครื่องอ่านหรือเครื่องชำระเงิน เช่น การให้บริการในร้านอาหารจานด่วน ร้านขายสินค้า ระบบการซื้อขายตั๋วและระบบการแลกเปลี่ยนข้อมูลแบบ peer-to-peer เช่น เพลง เกม และรูปภาพ การชำระเงินค่าโดยสารในระบบขนส่งมวลชน เป็นต้น การชำระเงินแบบไร้สัมผัสนี้ก่อให้เกิดการชำระเงินที่ง่ายและรวดเร็ว ลดการเข้าคิวชำระเงินในร้านค้า ห้างสรรพสินค้า และร้านสะดวกซื้อต่างๆ

2.2.1 หลักการทำงานของ NFC

หลักการทำงานของ NFC นั้น มีด้วยกันทั้งหมด 3 โหมด คือ

2.2.1.1 NFC Card Emulation Mode

ในโหมดนี้จะทำงานเสมือนเป็นบัตร Contactless ซึ่งนั่นหมายความว่าอุปกรณ์มือถือตามมาตรฐาน NFC จะทำตัวเป็นบัตรในรูปแบบใดก็ได้ตามมาตรฐาน ISO 14443 และ FeliCa ที่พบมากก็คือเป็น Contactless Smart Card เพื่อใช้ในการทำธุรกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนั้นยังสามารถขยายการใช้งานอุปกรณ์ NFC ชิ้นเดียวเป็นบัตรหลายใบได้ เช่นเป็นบัตรเครดิต บัตรโดยสารรถ BTS MRT บัตรเงินสด บัตรสะสมแต้ม เป็นต้น ทำให้เมื่อ NFC เป็นที่นิยมและผู้ให้บริการบัตรต่างๆ ทำแอปพลิเคชั่นสำหรับบริการของตนลงบนอุปกรณ์ ทำให้ไม่ต้องพกบัตรมากมาย เช่น Visa อาจจะออกแอปพลิเคชั่น ที่สามารถแทน Visa Wave หลายบัญชี และให้เลือกบัญชีที่ต้องการใช้ได้จากโทรศัพท์มือถือ หรือ Sony อาจจะออกแอปพลิเคชั่น ที่ใช้จำลองบัตร FeliCa สามารถเอาโทรศัพท์มือถือไปลงทะเบียนกับ MRT เพื่อใช้แทนบัตรโดยสารได้

2.2.1.2 Peer-to-Peer Mode

ในโหมดนี้จะทำการแลกเปลี่ยนข้อมูลระหว่างอุปกรณ์ NFC ด้วยกัน คล้ายกับการที่มือถือมีบลูทูธ แล้วทำการแลกเปลี่ยนข้อมูลโดยการจับคู่เครื่อง เข้าด้วยกันแล้วแลกเปลี่ยนข้อมูลเช่น นามบัตร รูปถ่าย แฟ้มข้อมูลอื่นๆ แต่ว่าสำหรับ NFC แล้ว ไม่ต้องมีการจับคู่เหมือนบลูทูธ เพียงแค่เลือกข้อมูลที่ต้องการแลกเปลี่ยนแล้วนำอุปกรณ์ NFC ที่รองรับโหมดนี้มาแตะกัน ข้อมูลก็จะทำการถูกถ่ายกันระหว่างเครื่อง เพราะรัศมีของ NFC อยู่ในระดับน้อยกว่า 10 เซนติเมตร ซึ่งต่างจากบลูทูธซึ่งออกแบบไว้ให้สื่อสารข้อมูลในระยะหลายเมตร

การแลกเปลี่ยนข้อมูลทำได้ผ่านโปรโตคอล TCP/IP หรือ OBEX (เหมือนกับการแลกเปลี่ยนข้อมูลผ่านบลูทูธหรือ IrDA) นอกจากแลกเปลี่ยนข้อมูลแล้วยังสามารถใช้ทำการ Synchronize ข้อมูลกับอุปกรณ์อื่นๆได้ด้วย

2.2.1.3 Reader/Writer Mode

ในโหมดนี้อุปกรณ์ NFC สามารถทำตัวเสมือนเป็นเครื่องอ่านเขียน Contactless Smart Card หรือบางครั้งเรียกว่า Tag โดยจะสามารถอ่านข้อมูลจาก Tag ที่ติดอยู่ใน Smartposter หรือจุดให้บริการข้อมูล การประยุกต์ใช้งานเช่น ทำการส่งเสริมการขายโดยแจกคูปองส่วนลดสำหรับ 50 คนแรกที่มาอ่านโฆษณาที่จุดให้บริการ ซึ่งการกำหนดจำนวนแบบนี้ไม่สามารถทำได้โดยการใช้ 2D Bar Code

Tag ยังสามารถทำ One-Touch Setup สำหรับวายฟายและบลูทูธ คือช่วยในการจับคู่อุปกรณ์ NFC ที่มีบลูทูธ หรือวายฟายในโหมด Ad-hoc เพียงแค่เอาอุปกรณ์มาแตะกันก็จะทำการจับคู่ให้อัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 การใช้งานเทคโนโลยี NFC

เทคโนโลยี NFC นี้ช่วยให้การสื่อสารระหว่างอุปกรณ์สามารถทำได้ง่ายขึ้น ต่างจากเทคโนโลยีไร้สายประเภทอื่น ได้แก่วายฟายหรือบลูทูธที่ต้องมีตั้งค่าต่างๆ ก่อนการใช้งาน แต่เทคโนโลยี NFC เพียงแค่นำอุปกรณ์มือถือ เช่น โทรศัพท์ไปใกล้กับเครื่องอ่านหรืออาร์เอฟไอดีการ์ด (RFID tag) ก็สามารถที่จะทำการส่งข้อมูลระหว่างกันได้ โดยไม่จำเป็นต้องมีการตั้งค่าใดๆก่อนการใช้งาน การใช้งานเทคโนโลยี NFC มีได้ 3 ลักษณะ ได้แก่

2.2.2.1 ทำงานเป็นอาร์เอฟไอดีการ์ด

อุปกรณ์มือถือ เช่นโทรศัพท์มือถือที่มีเครื่องอ่าน NFC ฝังอยู่สามารถทำงานเป็นอาร์เอฟไอดีการ์ดได้ ซึ่งต่างจากเครื่องอ่านอาร์เอฟไอดีในปัจจุบันที่ทำหน้าที่เป็นเครื่องอ่านเพียงอย่างเดียว การทำงานในลักษณะนี้ส่วนใหญ่จะใช้ในแอปพลิเคชั่นในเรื่องการเงิน เช่น การจ่ายเงินชำระค่าผ่านทาง เพียงแค่นำโทรศัพท์มือถือไปใกล้กับเครื่องอ่านอาร์เอฟไอดี ที่ติดตั้งไว้ที่จุดชำระเงินก็สามารถทำการชำระเงินได้ แทนการชำระเงินด้วยบัตรอาร์เอฟไอดีหรือเงินสด

2.2.2.2 ทำงานเป็นเครื่องอ่านอาร์เอฟไอดี

อุปกรณ์มือถือ เช่นโทรศัพท์มือถือที่มีเครื่องอ่าน NFC ฝังอยู่ สามารถทำงานเป็นเครื่องอ่านอาร์เอฟไอดี เมื่อต้องการอ่านข้อมูลจากอาร์เอฟไอดีการ์ดเช่น การใช้งานในลักษณะ Smart Poster เป็นต้น โดยที่โปสเตอร์จะมีอาร์เอฟไอดีสติกเกอร์ (RFID Sticker) เมื่อต้องการอ่านข้อมูลจากอาร์เอฟไอดีสติกเกอร์ เพียงแค่นำโทรศัพท์มือถือที่มีเครื่องอ่านไปอ่านอาร์เอฟไอดีสติกเกอร์บนโปสเตอร์ ข้อมูลในโปสเตอร์ก็จะปรากฏขึ้นมาบนโทรศัพท์มือถือ

2.2.2.3 การสื่อสารในลักษณะ Peer to Peer (P2P)

เครื่องอ่าน NFC สองเครื่องสามารถที่จะติดต่อสื่อสารกันโดยตรงได้ เมื่อต้องการส่งข้อมูล ตัวอย่างเช่น โทรศัพท์มือถือสองเครื่องที่มีฟังก์ชัน NFC สามารถที่จะส่งข้อมูลให้แก่กันได้โดยตรง เพียงนำโทรศัพท์ทั้งสองเครื่อง เข้ามาใกล้กันในระยะที่เครื่องอ่านที่อยู่ในโทรศัพท์ทั้งสองสามารถอ่านกันได้ ก็สามารถที่จะส่งข้อมูลถึงกันได้ โดยไม่ต้องผ่านเครือข่ายของโทรศัพท์มือถือ ไม่ว่าจะเป็น GPRS หรือ EDGE เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 สถาปัตยกรรม NFC

ในระดับฮาร์ดแวร์ จะมีชิปหนึ่งที่เรียกว่า NFC Radio เพื่อไว้ทำงานธุรกรรมต่างๆที่ต้องการความปลอดภัย เช่น การจ่ายเงิน การคมนาคม และ การระบุตัวตนเพื่อเข้าไปยังอาคารต่างๆ เป็นต้น ทั้งนี้การทำงานของ NFC Radio จำเป็นต้องใช้อีกชิปหนึ่งเรียก SE (Secure Element) ตัว NFC Radio จะมีปฏิกิริยาเมื่อนำไปใกล้กับ Tags, Reader หรืออุปกรณ์ NFC อื่นๆ โดย NFC Radio ตัวนี้จะเชื่อมต่อไปยังโฮส ซึ่งสามารถจะเป็น Baseband, Application Processes บนโทรศัพท์ หรือหน่วยประมวลผลกลางบน PC ส่วน SE คือชิปที่แยกออกมาต่างหากซึ่งจะมี Secure Processor, Temper Proof และ Execution Memory

และการทำงานจะแตกต่างจาก Host Processor หรือ PC Processor คือ มันถูกสร้างขึ้นมาเพื่อมีจุดมุ่งหมายเดียวคือการทำให้ธุรกรรมต่างๆมีความปลอดภัย โดย SE จะมีการทำงานที่พึ่งพา Secure Key ข้างใน Secure Processor เสมอ

2.2.4 ประโยชน์ของ NFC

เทคโนโลยี NFC นอกจากจะมีไว้สำหรับแลกเปลี่ยนข้อมูลระหว่างอุปกรณ์ที่รองรับแล้ว ยังพัฒนาให้สามารถอำนวยความสะดวกด้านอื่นด้วย เช่น การใช้โทรศัพท์มือถือเพื่อชำระค่าสินค้าและบริการ การชำระค่าอาหารและเครื่องดื่ม การชำระค่ารถโดยสาร ซื้อตั๋วดูหนัง การยืนยันตัวตน เป็นต้น หลักการทำงานก็เพียงแค่นำโทรศัพท์มือถือหรืออุปกรณ์พกพาที่รองรับมาแตะเข้าที่เครื่องอ่าน NFC หรือเครื่องอ่านประเภทอาร์เอฟไอดี เท่านั้น ซึ่งลักษณะการทำงานจะคล้ายๆกับบลูทูธมาก เพียงแต่ไม่ต้องทำการจับคู่อุปกรณ์ทั้งสองก่อนการใช้งาน เปรียบเทียบความแตกต่างระหว่าง NFC และบลูทูธดังตารางที่ 2.3

ตารางที่ 2.3 ความแตกต่างระหว่าง NFC และ Bluetooth [6]

	NFC	Bluetooth	Bluetooth พลังงานต่ำ
การเข้ากันได้กับอาร์เอฟไอดี	ISO 18000-3	active	Active
มาตรฐาน	ISO/IEC	Bluetooth SIG	Bluetooth SIG
มาตรฐานเครือข่าย	ISO 13157 etc.	IEEE 802.15.1	IEEE 802.15.1
ชนิดของเครือข่าย	Point-to-Point	WPAN	WPAN
การเข้ารหัส	Not with RFID	Available	Available
ระยะ	น้อยกว่า 0.2 เมตร	ประมาณ 10 เมตร	ประมาณ 1 เมตร
ความถี่	13.56 MHz	2.4-2.5 GHz	2.4-2.5 GHz
อัตราบิต	424 kbit/s	2.1 Mbit/s	1.0 Mbit/s

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NFC เป็นเทคโนโลยีที่พัฒนาขึ้นมาบนพื้นฐานของอาร์เอฟไอดี ซึ่งเป็นระบบสื่อสารไร้สายที่กำลังได้รับความนิยมมากในปัจจุบัน เนื่องจากสามารถต่อเชื่อมและแลกเปลี่ยนข้อมูลระหว่างอุปกรณ์ได้อย่างรวดเร็วและปลอดภัย ไม่มีการรบกวนสัญญาณจากระบบอื่น ใช้ประโยชน์ได้หลากหลาย โดยเฉพาะอย่างยิ่งการชำระค่าสินค้าและบริการที่มียอดเงินไม่มาก ตอนนี้ มีการติดตั้งชิป NFC บนโทรศัพท์มือถือหลายรุ่น โดยเฉพาะ Nokia และ Samsung Galaxy ที่สามารถใช้ NFC ที่ติดกับมือถือนี้มาใช้แทนกระเป๋าเงินสด ใช้ชำระค่าโดยสาร ค่าอาหาร ซึ่ง ในไทยก็มีการนำมาใช้ 3-4 ปีแล้ว แต่ยังไม่ค่อยเป็นที่นิยมนัก แต่มีการใช้กันอย่างแพร่หลายในญี่ปุ่น เกาหลี และฮ่องกง การใช้ NFC ยังมีอีกมากมาย ดังแสดงในรูปที่ 2.3 เช่น เวลาออกจากโรงหนัง สามารถเอามือถือแตะที่เครื่องเพื่อตอบแบบสอบถาม จะดูตัวอย่างหนังที่กำลังจะเข้าโรงก็ได้ หรือถ้าไปคอนเสิร์ต ก็หาตำแหน่งที่นั่งของคุณได้จากมือถือทันทีที่เดินผ่านประตู ถ้าขึ้นรถเมล์พอแตะมือถือจ่ายค่าตั๋วรถเมล์ ก็จะเปิดแอปพลิเคชั่นให้คุณทราบตำแหน่งเส้นทางของรถ และสามารถเตือนเมื่อใกล้ถึงจุดหมายได้ด้วย บางครั้งนำชิป NFC มาติดไว้บนนามบัตร ก็จะทำให้คู่ค้าของคุณรู้จักคุณมากขึ้น เวลาใช้ เพียงแค่เอานามบัตรที่มี NFC ติดอยู่ นำไปแตะมือถือของผู้รับที่มีชิป NFC เท่านั้น โทรศัพท์ก็จะอ่านค่าต่างๆออกมา เข้าไปที่เว็บไซต์ (Website) ได้ หรือให้ ดู Resume ของคุณแบบออนไลน์ (Online) เป็นต้น

รูปที่ 2.3 การใช้ประโยชน์ของ NFC [8]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 โปรแกรมอีคลิป (Eclipse)

2.3.1 ประวัติ

อีคลิปคือโปรแกรมที่ใช้สำหรับพัฒนาภาษาจาวา มีสัญลักษณ์ดังแสดงในรูปที่ 2.4 ซึ่งโปรแกรมอีคลิป เป็นโปรแกรมหนึ่งที่ใช้ในการพัฒนาแอปพลิเคชั่นเซิร์ฟเวอร์ ได้อย่างมีประสิทธิภาพ และเนื่องจากอีคลิปเป็นซอฟต์แวร์โอเพนซอร์ซ (Open Source) ที่พัฒนาขึ้นเพื่อใช้โดยนักพัฒนาเอง ทำให้ความก้าวหน้าในการพัฒนาของอีคลิปเป็นไปอย่างต่อเนื่องและรวดเร็ว อีคลิปมีองค์ประกอบหลักที่เรียกว่าอีคลิปแพลตฟอร์ม (Eclipse Platform) ซึ่งให้บริการพื้นฐานหลักสำหรับรวบรวมเครื่องมือต่างๆจากภายนอกให้สามารถเข้ามาทำงานร่วมกันในสภาพแวดล้อมเดียวกัน และมีองค์ประกอบที่เรียกว่า Plug-in Development Environment (PDE) ซึ่งใช้ในการเพิ่มความสามารถในการพัฒนาซอฟต์แวร์มากขึ้น เครื่องมือภายนอกจะถูกพัฒนาในรูปแบบที่เรียกว่าอีคลิปปลั๊กอิน (Plug-in) ดังนั้นหากต้องการให้อีคลิปทำงานใดเพิ่มเติม ก็เพียงแต่พัฒนาปลั๊กอินสำหรับงานนั้นขึ้นมาและนำปลั๊กอินนั้นมาติดตั้งเพิ่มเติมให้กับอีคลิปที่มีอยู่เท่านั้นอีคลิปปลั๊กอินที่มีมาพร้อมกับอีคลิปเมื่อดาวน์โหลดมาครั้งแรกก็คือองค์ประกอบที่เรียกว่า Java Development Toolkit (JDT) ซึ่งเป็นเครื่องมือในการเขียนและ Debug โปรแกรมภาษาจาวา

ข้อดีของโปรแกรมอีคลิป คือ ติดตั้งง่าย สามารถใช้ได้กับ J2SDK ได้ทุกเวอร์ชั่น (Version) รองรับภาษาต่างประเทศอีกหลายภาษา มีปลั๊กอินที่ใช้เสริมประสิทธิภาพของโปรแกรม สามารถทำงานได้กับไฟล์หลายชนิด เช่น HTML, Java, C, JSP, EJB, XML และ GIF ใช้งานได้กับระบบปฏิบัติการ วินโดวส์ (Windows), ลินุกซ์และแมคโอเอส (Mac OS)

รูปที่ 2.4 สัญลักษณ์ของโปรแกรมอีคลิป [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 ส่วนประกอบของโปรแกรมอีคลิป

ส่วนประกอบต่าง ๆ ของอีคลิปแสดงในรูปที่ 2.5 สามารถแบ่งได้ ดังนี้

(1) เป็นส่วนของแถบเมนู และเครื่องมือต่างๆ

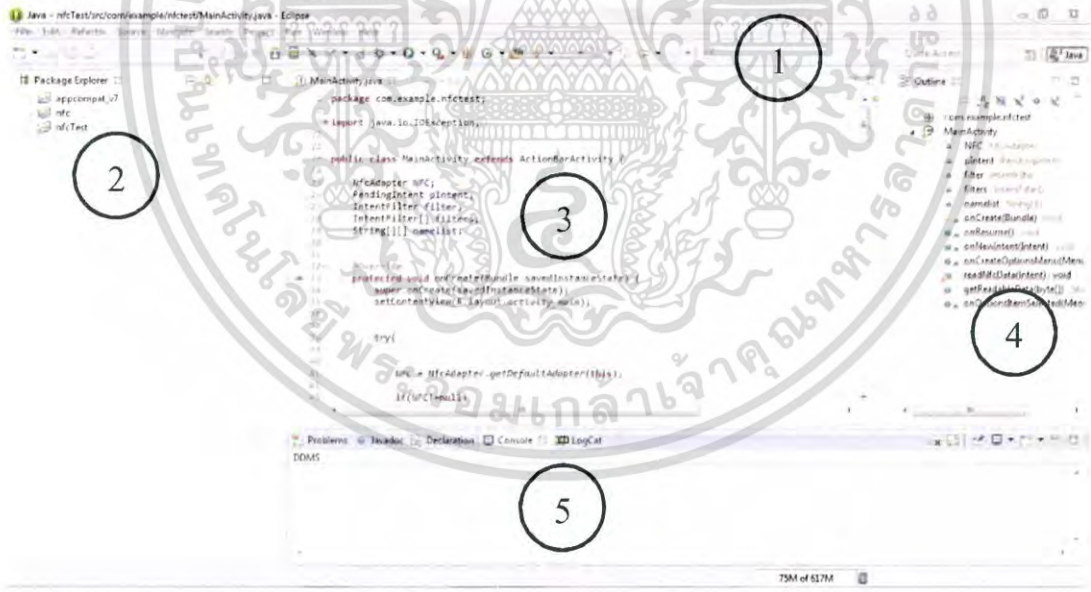
(2) เป็นส่วนที่แสดงและจัดการโปรเจ็กต์ (Project) ต่างๆ เหมือนเป็นการเรียกดูโปรเจ็กต์หรือไฟล์ต่างๆ

(3) ส่วนที่เขียนโปรแกรม

(4) เป็นส่วนที่แสดงถึงโครงสร้างหรือส่วนประกอบของคลาส (Class) เช่น Attribute Method และด้านบนสุดเป็นส่วนที่ใช้จัดการกับปลั๊กอินต่างๆ

(5) เป็นส่วนที่ใช้ในการแสดงผลการทำงานต่างๆ เมื่อทำการดำเนินงานโปรแกรม และยังมีส่วนของการดีบั๊ก (Debug) โปรแกรมด้วย

Perspective ในอีคลิปเป็นชื่อสำหรับการดู การจัดเก็บ และแก้ไขไฟล์ในพื้นที่ซึ่ง Perspective เริ่มต้นคือจาวาซึ่งอีคลิปสามารถเปิด Perspective หลายๆหน้าต่างพร้อมกันได้ แต่เวลาทำงาน จะทำงานใน Perspective ใด Perspective หนึ่งเท่านั้น ผู้ใช้สามารถที่จะสลับ Perspective ต่างๆ หรือเปิด Perspective ในหน้าต่างใหม่ได้ นอกจากนี้ Perspective ยังควบคุมเมนูที่แสดงอยู่ได้อีกด้วย



รูปที่ 2.5 ส่วนประกอบของโปรแกรมอีคลิป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 สถาปัตยกรรมของโปรแกรมอีคลิป

อีคลิปเป็นแพลตฟอร์มที่เพิ่มความสามารถและเชื่อถือได้ ซึ่งได้ถูกนำมาใช้โดยนักพัฒนาหลายคน การเรียนรู้แพลตฟอร์มอีคลิปต้องใช้เวลานานมากเนื่องจากมีคุณสมบัติมากมาย วิธีหนึ่งที่จะทำให้มันง่ายขึ้นก็คือการทำความเข้าใจการออกแบบสถาปัตยกรรมของโปรแกรม การทำงานของอีคลิป และรูปแบบการออกแบบที่ใช้ในแต่ละชั้นของสถาปัตยกรรมอีคลิปดังแสดงในรูปที่ 2.6 สถาปัตยกรรมอีคลิปมีโครงสร้างชั้นที่ชัดเจน แต่ละชั้นมีความเป็นอิสระต่อกัน แต่สามารถโต้ตอบกับในแต่ละชั้นอื่นๆ

รูปที่ 2.6 สถาปัตยกรรมของโปรแกรมอีคลิป [11]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 จาวา (Java)

ภาษาจาวา เป็นภาษาโปรแกรมเชิงวัตถุ (Object Oriented Programming) พัฒนาโดย เจมส์ กอสลิง และวิศวกรคนอื่นๆ ที่ซันไมโครซิสเต็มส์ (Sun Microsystems) ภาษาจาวาถูกพัฒนาขึ้นในปี พ.ศ. 2534 โดยเป็นส่วนหนึ่งของ โครงการกรีน (The Green Project) และสำเร็จออกสู่สาธารณะในปี พ.ศ. 2538 ซึ่งภาษานี้มีจุดประสงค์เพื่อใช้แทนภาษาซีพลัสพลัส (C++) โดยรูปแบบที่เพิ่มเติมขึ้นคล้ายกับภาษาอ็อบเจกต์ทีฟซี (Objective-C) แต่เดิมภาษานี้เรียกว่า ภาษาโอ๊ก (Oak) ซึ่งตั้งชื่อตามต้นโอ๊กใกล้ที่ทำงานของ เจมส์ กอสลิง แต่ว่ามีปัญหาทางลิขสิทธิ์ จึงเปลี่ยนไปใช้ชื่อ "จาวา" ซึ่งเป็นชื่อกาแฟแทน และแม้ว่าจะมีชื่อคล้ายกัน แต่ภาษาจาวาไม่มีความเกี่ยวข้องใดๆ กับภาษาจาวาสคริปต์ (JavaScript) ปัจจุบันมาตรฐานของภาษาจาวาดูแลโดย Java Community Process ซึ่งเป็นกระบวนการอย่างเป็นทางการ ที่อนุญาตให้ผู้ที่สนใจเข้าร่วมกำหนดความสามารถในจาวาแพลตฟอร์มได้

2.4.1 ข้อดีของภาษาจาวา

2.4.1.1 ภาษาจาวา เป็นภาษาโปรแกรมที่ง่ายในการเรียนรู้

ภาษาจาวา มีคุณลักษณะต่างๆ ดังนี้เช่น เชื่อมต่อข้ามแพล็ตฟอร์ม (Platforms) ต่างๆ ได้ สามารถเขียนโปรแกรมแบบ OOP (Object-Oriented Programming) ได้ง่ายมาก โปรแกรมมีขนาดเล็ก และมีวิธีการเขียนไม่ยุ่งยากซับซ้อน ดังนั้นโปรแกรมที่เขียนด้วยภาษาจาวา จึงคอมไพล์ได้ง่ายตลอดจนตรวจสอบหาข้อผิดพลาดโปรแกรมได้ง่ายด้วย มีประสิทธิภาพในการทำงานและมีความยืดหยุ่นสูง

2.4.1.2 ภาษาจาวาเป็นการเขียนโปรแกรมเชิงวัตถุ OOP

การเขียนโปรแกรมเชิงวัตถุ เป็นเทคนิคการเขียนโปรแกรมให้มีลักษณะเป็นโมดูล (Module) แบ่งโปรแกรมเป็นส่วนๆ ตามสภาวะแวดล้อมการทำงานของโปรแกรมซึ่งเรียกว่า Method โดยทุก Method ก็คือ ระเบียบวิธี หรือการทำงานอย่างใดอย่างหนึ่ง โดยจะถูกรวบรวมอยู่ในคลาส ซึ่งหลักการเขียนโปรแกรมเชิงวัตถุจะมองค์ประกอบของโปรแกรมต่างๆเป็นคลาสหรือวัตถุเรียกว่า Object

2.4.1.3 ภาษาจาวาเป็นอิสระต่อแพล็ตฟอร์ม (Java is Platform-Independent)

ทั้งระดับซอร์สโค้ดและไบนารีโค้ด (Binary Code) ช่วยให้สามารถเคลื่อนย้ายโปรแกรมจากระบบคอมพิวเตอร์หนึ่งไปยังระบบคอมพิวเตอร์อื่นได้อย่างง่ายดาย เพราะว่าโปรแกรมที่เขียนด้วยภาษาจาวา ได้รวบรวมคำสั่งต่างๆไว้ในไลบรารีคลาสพื้นฐานต่างๆ เป็น Java Packages ช่วยอำนวยความสะดวกในการเขียนคำสั่ง เมื่อย้ายโปรแกรมไปยังแพล็ตฟอร์มอื่น โดยไม่ต้องเขียนซอร์สโค้ด ขึ้นใหม่ทำให้ประหยัดเวลามาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 กราฟ (Graph)

กราฟเป็นโครงสร้างข้อมูลไม่เป็นเชิงเส้น (Nonlinear Data Structure) กราฟเป็นโครงสร้างข้อมูลประเภทหนึ่งที่แสดงความสัมพันธ์ระหว่างโหนด (Vertex) และเส้นเชื่อม (Edge) กราฟจะประกอบด้วยกลุ่มของโหนดซึ่งแสดงในกราฟด้วยสัญลักษณ์รูปวงกลม และ กลุ่มของเส้นเชื่อมระหว่างโหนด ใช้แสดงถึงความสัมพันธ์ระหว่างโหนด หากมีโหนดตั้งแต่ 2 โหนดขึ้นไปมีความสัมพันธ์กัน ใช้สัญลักษณ์เส้นตรงซึ่งอาจมีหัวลูกศร หรือไม่มีก็ได้ กราฟสามารถเขียนแทนด้วยสัญลักษณ์ ดังนี้

G = (V, E)

G คือ กราฟ

V คือ กลุ่มของ Vertex

E คือ กลุ่มของ Edge

ศัพท์ที่เกี่ยวข้อง

1.เวอร์เทก (Vertex) หมายถึง โหนด

2.เอดจ (Edge) หมายถึง เส้นเชื่อมของโหนด

3.ดีกรี (Degree) หมายถึง จำนวนเส้นเข้าและออก ของโหนดแต่ละโหนด

4.แอดจาเซนท์โหนด (Adjacent Node) หมายถึง โหนดที่มีการเชื่อมโยงกัน

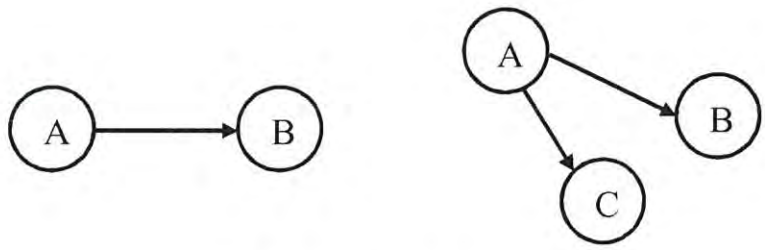
ตัวอย่างของกราฟในชีวิตประจำวัน เช่น กราฟของการเดินทางระหว่างเมือง ซึ่งโหนดคือกลุ่มของเมืองต่างๆ และเส้นเชื่อมคือ เส้นทางเดินระหว่างเมือง หรือ ในเครือข่ายคอมพิวเตอร์ (Computer Network) โหนดก็คือ กลุ่มของเครื่องคอมพิวเตอร์และเส้นเชื่อมก็คือ เส้นทางการติดต่อสื่อสารระหว่างโหนดต่างๆ เป็นต้น

2.5.1 ประเภทของกราฟ

แบ่งเป็น 3 ประเภทโดยแบ่งตามประเภทของเส้นเชื่อมได้ดังนี้

2.5.1.1 กราฟแสดงทิศทาง (Direct Graph)

เป็นกราฟที่แสดงเส้นเชื่อมระหว่างโหนดโดยแสดงทิศทางของการเชื่อมต่อด้วย ตัวอย่างแสดงดังรูปที่ 2.7

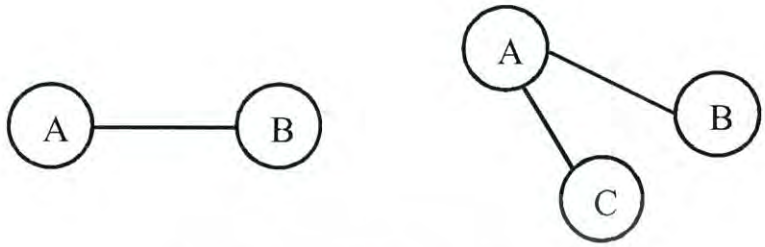


รูปที่ 2.7 กราฟแสดงทิศทาง [13]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1.2 กราฟไม่แสดงทิศทาง (Undirected Graph)

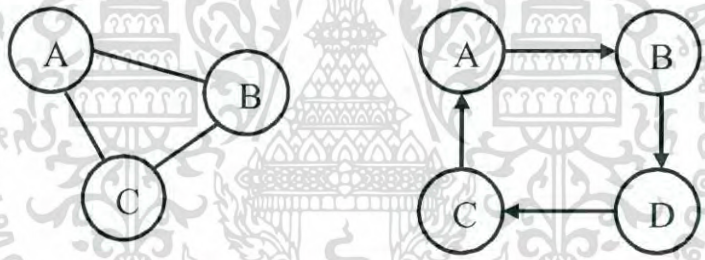
เป็นกราฟที่แสดงเส้นเชื่อมต่อระหว่างโหนดแต่ไม่แสดงทิศทางของการเชื่อมต่อ ตัวอย่างแสดงดังรูปที่ 2.8



รูปที่ 2.8 กราฟไม่แสดงทิศทาง [13]

2.5.1.3 กราฟที่มีต้นทางและปลายทางเป็นโหนดเดียวกัน (Cyclic Graph)

เป็นกราฟที่มีเส้นเชื่อมต่อระหว่างโหนดที่ทำให้โหนดมีลักษณะเป็นวงจรปิด (Cycle) เส้นเชื่อมต่อระหว่างโหนด อาจจะแสดงทิศทางหรือไม่แสดงทิศทางการเชื่อมต่อก็ได้ ดังรูปที่ 2.9



รูปที่ 2.9 กราฟที่มีโหนดต้นทางและโหนดปลายทางเป็นโหนดเดียวกัน [13]

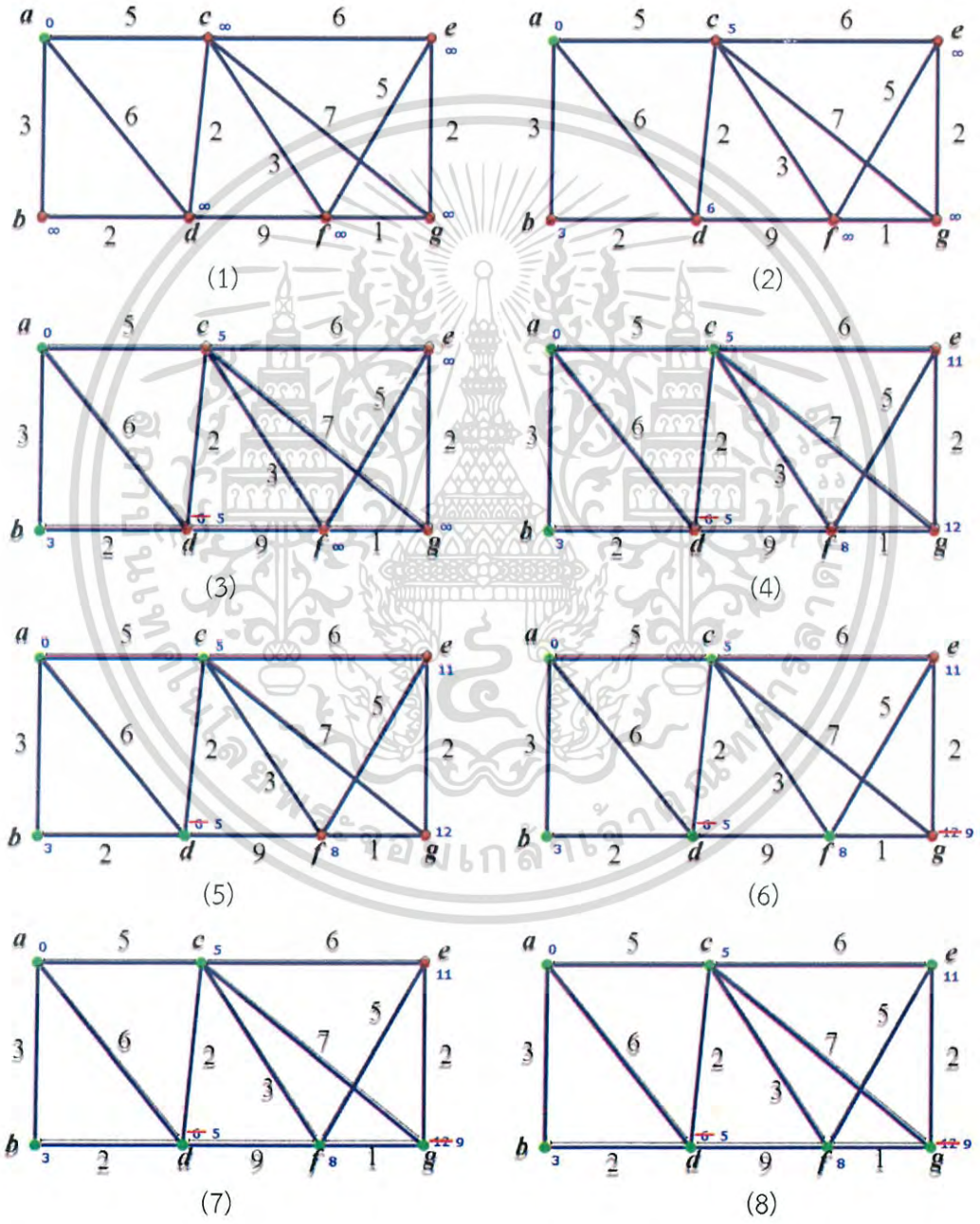
2.5.2 ขั้นตอนวิธีการทำงานของ Dijkstra's algorithm

กำหนดให้จุดหนึ่งภายในกราฟเป็นจุดเริ่มต้น (Initial node) และกำหนดให้ "ระยะทางของจุด X" หมายถึงระยะทางจากจุดเริ่มต้นไปยังจุด X โดย Dijkstra's algorithm จะกำหนดค่าระยะทางเริ่มต้นไว้บางจุดและจะเพิ่มค่าไปทีละขั้นตอน

1. กำหนดให้จุดเริ่มต้นมีค่าเป็นศูนย์ และจุดอื่นๆมีค่าเป็นอนันต์
2. ทำเครื่องหมายทุกจุด ยกเว้นจุดเริ่มต้นว่ายังไม่ไปเยือน (Unvisited) จุดเริ่มต้นเป็นจุดปัจจุบัน สร้างเซตของจุดที่ยังไม่ไปเยือนขึ้นมาซึ่งประกอบด้วยทุกจุด ยกเว้นจุดเริ่มต้น
3. จากจุดปัจจุบัน พิจารณาจุดข้างเคียงตามเส้นเชื่อมทุกจุดที่ยังไม่ไปเยือน และคำนวณระยะทางต่อเนื่องของเส้นเชื่อม ถ้าระยะทางที่คำนวณได้มีค่าน้อยกว่าค่าระยะทางที่บันทึกอยู่ของจุดนั้น ให้เขียนทับค่าระยะทางของจุดดังกล่าว แม้ว่าจุดข้างเคียงได้ถูกพิจารณาแล้ว แต่ก็ยังไม่ทำเครื่องหมายว่าไปเยือนแล้ว (Visited) ในขั้นตอนนี้ จุดข้างเคียงจะยังคงอยู่ในเซตของจุดที่ยังไม่ไปเยือนเช่นเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อพิจารณาจุดข้างเคียงจากจุดปัจจุบันครบทุกจุดแล้ว ทำเครื่องหมายที่จุดปัจจุบันว่าไปเยือนแล้ว และนำออกจากเซตของจุดที่ยังไม่ไปเยือน โดยจุดที่ไปเยือนแล้วนี้จะไม่ถูกนำมาตรวจสอบอีก ค่าระยะทางที่บันทึกอยู่จะสิ้นสุดและมีค่าน้อยที่สุด
5. จุดปัจจุบันตัวถัดไปที่ถูกเลือกจะมีค่าระยะทางน้อยสุดในเซตของจุดที่ยังไม่ไปเยือน
6. ถ้าเซตของจุดที่ยังไม่ไปเยือนว่างแล้วให้หยุดการทำงาน แต่ถ้าหากไม่ใช่ให้เลือกจุดที่ยังไม่ไปเยือนที่มีค่าระยะทางน้อยสุดเป็นจุดปัจจุบัน แล้ววนไปทำขั้นตอนที่ 3



รูปที่ 2.10 ตัวอย่างการใช้งานของ Dijkstra's algorithm [14]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หาเส้นทางที่สั้นที่สุดจากจุด a ไปยังจุด g

(1) กำหนดให้จุดเริ่มต้น ซึ่งในที่นี้คือจุด a มีค่าเป็น 0 และกำหนดให้จุดอื่นๆที่เหลือมีค่าเป็นอนันต์ (Infinity) และเพิ่มจุด a เข้าไปในเซตของจุดที่เคยมาเยือนแล้ว และเพิ่มจุดที่เหลือไปยังจุดที่ไม่เคยไปเยือน

Visieted Set = {a}

Unvisited Set = {b,c,d,e,f,g}

(2) จากจุด a ดูว่ามีเส้นเชื่อมไปยังจุดใดบ้างที่ยังไม่ได้ไปเยือน ซึ่งก็คือจุด b,c,d ทำการคำนวณระยะทางไปยังจุดที่มีเส้นเชื่อม ซึ่งค่าที่คำนวณได้น้อยกว่าค่าอนันต์จึงทำการเขียนค่าใหม่ทับลงไป ซึ่งในขั้นตอนนี้ Visited set และ Unvisited set ยังไม่เปลี่ยนแปลง

Visieted Set = {a}

Unvisited Set = {b,c,d,e,f,g}

(3) อัลกอริทึมจะเลือกพิจารณาจุด b เป็นจุดต่อไปเพราะว่า b เป็นจุดที่มีระยะทางสั้นที่สุดใน Unvisited set โดยดูเส้นเชื่อมจากจุด b ว่าไปยังจุดใดบ้างที่อยู่ใน Unvisited set ถ้าหากระยะทางจากจุด b ไปยังจุดนั้นน้อยกว่าค่าเดิม ก็จะทำการแก้ไขค่าระยะทางที่จุดนั้น โดยในตัวอย่างก็คือจุด d โดยมีระทางเท่ากับ 3+2 = 5 ซึ่งน้อยกว่าค่าเก่าคือ 6 อัลกอริทึมจึงทำการแก้ไขค่าระยะทางไปจุด d เป็น 5 และเมื่อพิจารณาเส้นเชื่อมทั้งหมดที่มาจากจุด b ครบแล้ว ก็จะเพิ่ม b เข้าไปใน Visited set

Visieted Set = {a,b}

Unvisited Set = {c,d,e,f,g}

(4) อัลกอริทึมจะเลือกพิจารณาจุด c เป็นจุดต่อไปเพราะว่า c เป็นจุดที่มีระยะทางสั้นที่สุดใน Unvisited set ถึงแม้ว่า c จะมีระยะทาง 5 หน่วยเท่ากับ d แต่ว่า c มาก่อน อัลกอริทึมจึงเลือกจุด c โดยดูเส้นเชื่อมจากจุด c ว่าไปยังจุดใดบ้างที่อยู่ใน Unvisited set ถ้าหากระยะทางจากจุด c ไปยังจุดนั้นน้อยกว่าค่าเดิม ก็จะทำการแก้ไขค่าระยะทางที่จุดนั้น ซึ่งจุดที่เชื่อมกับ c คือ d,e,f,g แต่ระยะทางจาก c ไป d โดยรวมค่าเก่าด้วยคือ 5+2 = 7 ซึ่งมากกว่าระยะทางที่จุด d เก็บไว้ ดังนั้นระยะทางที่สั้นที่สั้นที่สุดในการไปจุด d จึงไม่เปลี่ยนแปลง และเมื่อพิจารณาเส้นเชื่อมทั้งหมดที่มาจากจุด c ครบแล้ว ก็จะเพิ่ม c เข้าไปใน Visited set

Visieted Set = {a,b,c}

Unvisited Set = {d,e,f,g}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(5) อัลกอริทึมจะเลือกพิจารณาจุด d เป็นจุดต่อไปเพราะว่า d เป็นจุดที่มีระยะทางสั้นที่สุดใน Unvisited set อัลกอริทึมจึงเลือกจุด d โดยดูเส้นเชื่อมจากจุด d ว่าไปยังจุดใดบ้างที่อยู่ใน Unvisited set ซึ่งตอนนี้มีจุดเดียวคือ f แต่ระยะทางจาก d ไป f โดยรวมค่าเก่าด้วยคือ 5+9 = 14 ซึ่งมากกว่าระยะทางที่จุด f เก็บไว้ ดังนั้นระยะทางที่สั้นที่สั้นที่สุดในการไปจุด f จึงไม่เปลี่ยนแปลง และเมื่อพิจารณาเส้นเชื่อมทั้งหมดที่มาจากจุด d ครบแล้ว ก็จะเพิ่ม d เข้าไปใน Visited set

Visieted Set = {a,b,c,d}

Unvisited Set = {e,f,g}

(6) อัลกอริทึมจะเลือกพิจารณาจุด f เป็นจุดต่อไปเพราะว่า f เป็นจุดที่มีระยะทางสั้นที่สุดใน Unvisited set อัลกอริทึมจึงเลือกจุด f โดยดูเส้นเชื่อมจากจุด f ว่าไปยังจุดใดบ้างที่อยู่ใน Unvisited set ซึ่งก็คือจุด e กับ g โดยในขั้นตอนนี้ g จะได้ระยะทางที่สั้นที่สุดค่าใหม่ก็คือ 8+1 = 9 และเมื่อพิจารณาเส้นเชื่อมทั้งหมดที่มาจากจุด f ครบแล้ว ก็จะเพิ่ม f เข้าไปใน Visited set

Visieted Set = {a,b,c,d,f}

Unvisited Set = {e,g}

(7) อัลกอริทึมจะเลือกพิจารณาจุด g เป็นจุดต่อไปเพราะว่า g เป็นจุดที่มีระยะทางสั้นที่สุดใน Unvisited set อัลกอริทึมจึงเลือกจุด g โดยดูเส้นเชื่อมจากจุด g ว่าไปยังจุดใดบ้างที่อยู่ใน Unvisited set ซึ่งก็คือจุด e โดยในขั้นตอนนี้ e ไป g จะมีระยะทางที่สั้นที่สุดเท่ากับค่าเก่าคือ 9+2 = 11 ดังนั้นระยะทางจึงไม่เปลี่ยนแปลง และเมื่อพิจารณาเส้นเชื่อมทั้งหมดที่มาจากจุด g ครบแล้ว ก็จะเพิ่ม g เข้าไปใน Visited set

Visieted Set = {a,b,c,d,f,g}

Unvisited Set = {e}

(8) อัลกอริทึมจะเลือกพิจารณาจุด e เป็นจุดสุดท้ายเพราะว่า e เป็นจุดที่มีระยะทางสั้นที่สุดใน Unvisited set ที่เหลืออยู่ อัลกอริทึมจึงเลือกจุด e แต่ว่า e ไม่มีเส้นเชื่อมไปยังจุดอื่นๆที่เหลืออยู่ใน Unvisited set อีกแล้ว ดังนั้นจึงเพิ่มจุด e เข้าไปใน Visited set ได้เลย และเป็นการจบการทำงานของ Dijkstra's algorithm เพราะจุดทุกจุดถูกเพิ่มไปยัง Visited set ทั้งหมดแล้ว

Visieted Set = {a,b,c,d,f,g,e}

Unvisited Set = {}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบ

3.1 การออกแบบ

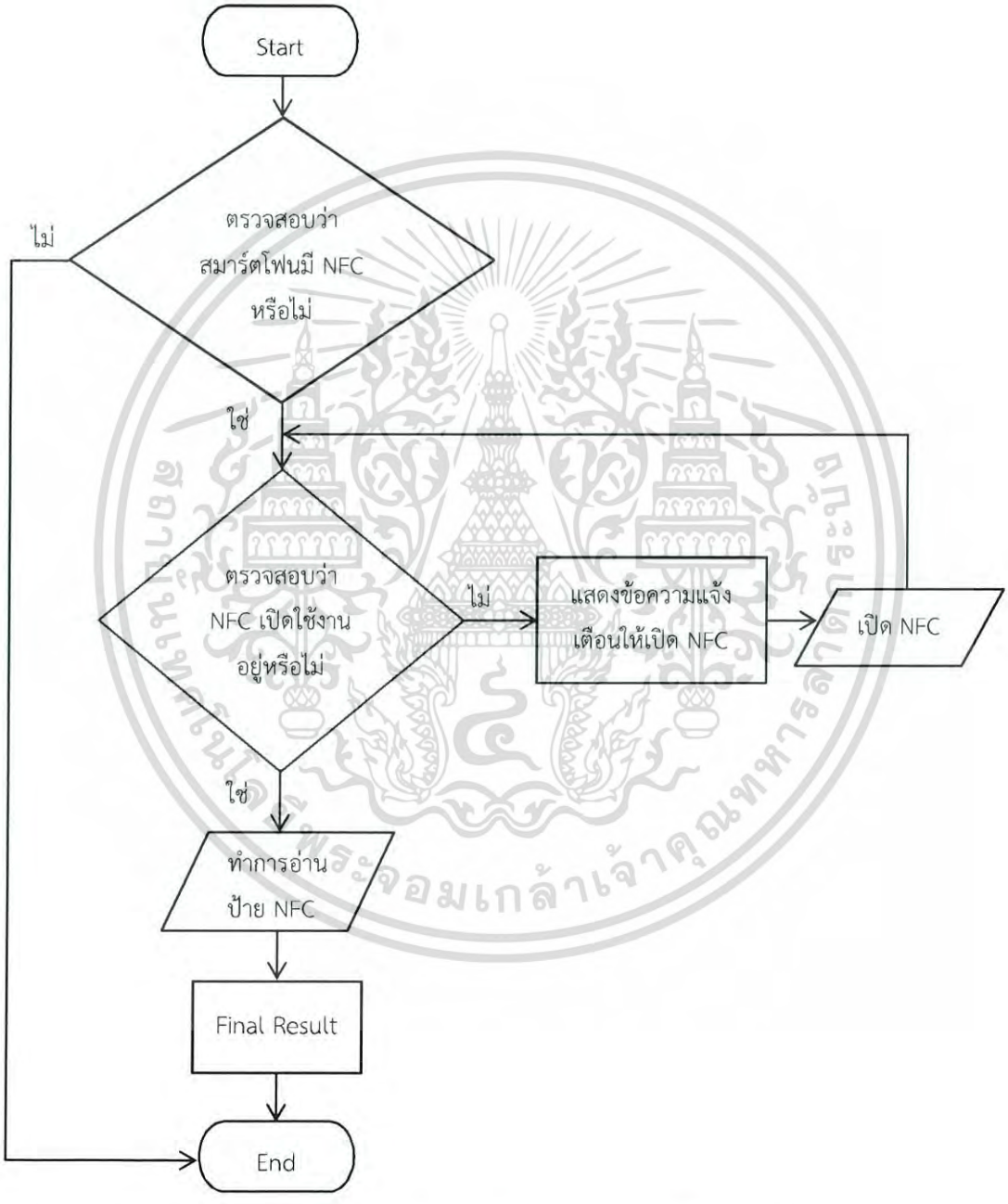
การทำงานของระบบเริ่มจากนำโทรศัพท์มือถือระบบแอนดรอยด์ โครงงานนี้ใช้งานเป็นรุ่น Sony Xperia Z ไปสัมผัสกับป้าย NFC ตามจุดต่างๆภายในอาคาร เพื่อให้แอปพลิเคชั่นแสดงแผนที่ตำแหน่งปัจจุบันของผู้ใช้งาน และรายละเอียดของร้านค้า โดยสมาร์ตโฟนสัมผัสกับป้าย NFC จะได้รับข้อมูลจากป้าย NFC จากนั้นจะทำการส่งข้อมูลไปยังส่วนประมวลผล หรือหน้าแสดงแผนที่ เมื่อได้รับข้อมูลที่ถูกส่งมาจากส่วนการอ่านข้อมูลแล้ว จะทำการแสดงแผนที่ ตำแหน่งปัจจุบันของผู้ใช้งานและรายละเอียดของร้านค้า ให้ตรงกับข้อมูลที่ได้รับมา และเมื่อผู้ใช้งานเลือกจุดหมายปลายทาง ระบบจะแสดงเส้นทางที่สั้นที่สุด โดยใช้ทฤษฎีกราฟ คำนวณโดย Dijkstra's algorithm ซึ่งแผนการทำงานแสดงดังรูปที่ 3.1

รูปที่ 3.1 แผนผังการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

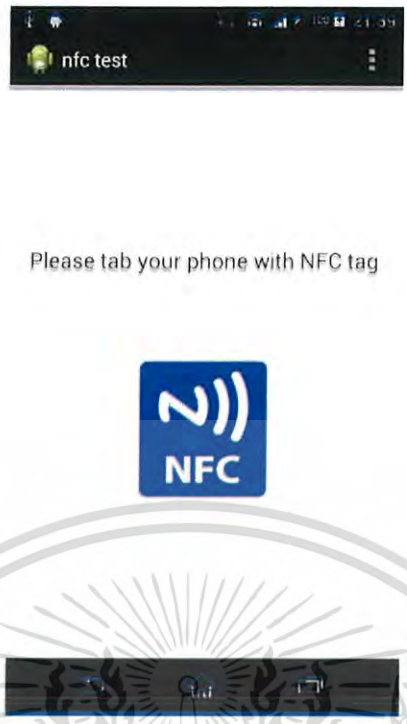
3.1.1 สมาร์ตโฟนสัมผัสกับป้าย NFC

ผู้ใช้งานนำสมาร์ตโฟนที่มีแอปพลิเคชั่นแล้ว ไปสัมผัสกับป้าย NFC ถ้าสมาร์ตโฟนของผู้ใช้งานไม่ได้เปิดการใช้งาน NFC แอปพลิเคชั่นจะทำการแสดงข้อความแจ้งเตือนและสลับหน้า User Interface ไปยังหน้าตั้งค่าการใช้งาน เพื่อให้ผู้ใช้งานเปิด NFC ซึ่งมีขั้นตอนการทำงานดังแสดงในรูปที่ 3.2 และเมื่อเปิดใช้งาน NFC แล้ว แอปพลิเคชั่นจะแสดงหน้า User Interface ดังรูปที่ 3.3



รูปที่ 3.2 Flowchart ตรวจสอบการใช้งาน NFC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 หน้า User Interface เมื่อเข้าแอปพลิเคชั่น

3.1.2 สมาร์ตโฟนได้รับข้อมูลจากป้าย NFC นั้น

เมื่อสมาร์ตโฟนสัมผัสป้าย NFC แล้ว สมาร์ตโฟนจะได้รับข้อมูลจากป้าย NFC นั้น โดยที่ข้อมูลที่ได้รับมานั้นจะมีข้อมูลที่ไม่ต้องการอยู่ด้วย เพราะฉะนั้นจึงต้องทำการตัดส่วนที่ไม่ต้องการออก เลือกเฉพาะข้อมูลที่ต้องการใช้เท่านั้น โดยใช้คำสั่ง split ข้อมูล ดังแสดงในรูปที่ 3.4 จากโค้ดในรูปที่ 3.4 ได้ทำการใส่ “//==” และ “==//” ในคำสั่ง เพื่อตัดเฉพาะส่วนที่ต้องการ ซึ่งข้อมูลในป้าย NFC จะต้องใส่ “//==” และ “==//” ไว้ที่ส่วนหน้าและส่วนหลังของข้อมูล ตัวอย่างเช่น “//==E12-Lift==//”, “//==E12-1106==//” เป็นต้น

```
String[] read_data = result.split("//==");
if(read_data.length == 2){
      Log.v("msg", "Found //==");
      String[] read_data2 = read_data[1].split("==//");
      if(read_data2.length == 2){
            Log.v("msg", "Found ==//");
            TextView tv = (TextView)findViewById(R.id.textView1);
            String text2 = read_data2[0];
```

รูปที่ 3.4 โค้ด split ข้อมูลเพื่อเลือกเฉพาะข้อมูลที่ต้องการ

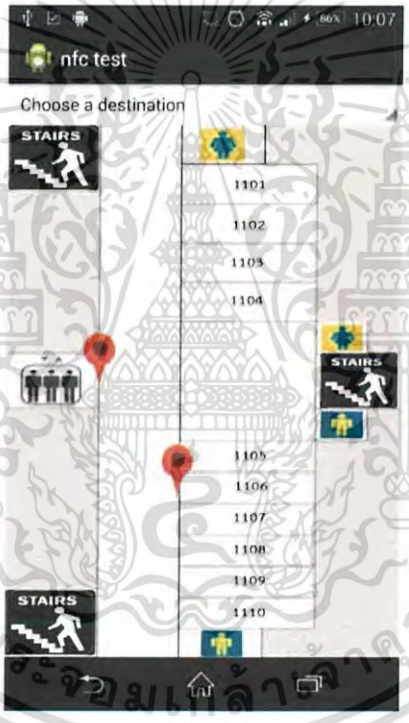
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 การส่งข้อมูลที่ได้รับไปยังส่วนประมวลผล

การส่งข้อมูลที่ได้รับไปยังส่วนประมวลผลนั้น ทำได้โดยการใส่โค้ดเข้าไปในส่วนการอ่านข้อมูล ดังแสดงในรูปที่ 3.5 จากนั้นเมื่อส่งข้อมูลไปแล้ว หน้า User Interface ของแอปพลิเคชั่นจะแสดงแผนที่ และตำแหน่งที่ได้บันทึกไว้ทุกจุดขึ้นมา ในที่นี้ได้จำลองแผนที่อาคารเรียนรวม 12 ชั้นเฉพาะชั้น 11 ที่ตำแหน่งหน้าลิฟต์ และหน้าห้อง 1106 ดังแสดงในรูปที่ 3.6

```
Intent i = new Intent(getApplicationContext(), Second.class);
i.putExtra("fromnfc",text2);
startActivity(i);
```

รูปที่ 3.5 โค้ดการส่งข้อมูลไปยังส่วนประมวลผล

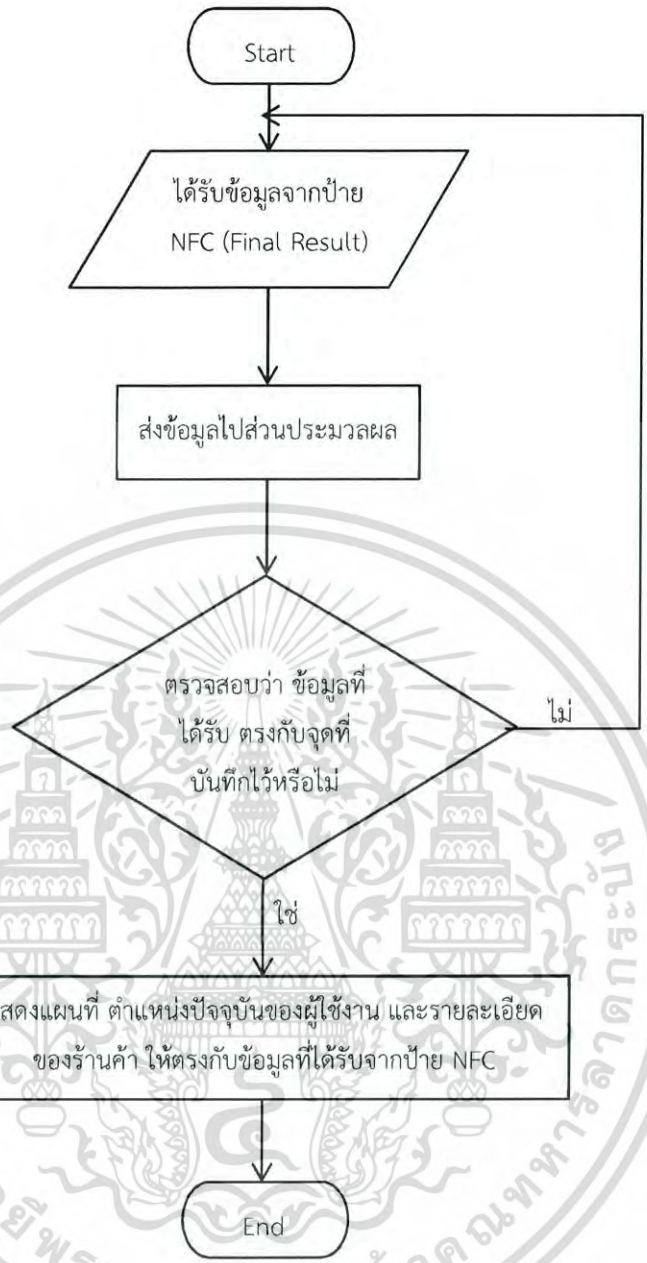


รูปที่ 3.6 User Interface หลังส่งข้อมูล

3.1.4 การแสดงตำแหน่งปัจจุบันของผู้ใช้งานและรายละเอียดของร้านค้า ให้ตรงกับข้อมูลที่ได้รับจากป้าย NFC

เมื่อสมาร์ตโฟนได้รับข้อมูลจากป้าย NFC แล้ว จะทำการส่งข้อมูลไปยังส่วนประมวลผล จากนั้นส่วนประมวลผลจะตรวจสอบว่าข้อมูลที่ได้รับมานั้นตรงกับข้อมูลที่บันทึกไว้หรือไม่ ถ้าตรงกับข้อมูลที่บันทึกไว้ แอปพลิเคชั่นจะแสดงตำแหน่งปัจจุบันของผู้ใช้งานและรายละเอียดของร้านค้า ให้ตรงกับข้อมูลที่ได้รับมา ซึ่งมีขั้นตอนการทำงานดังแสดงในรูปที่ 3.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 Flowchart การแสดงตำแหน่งปัจจุบันของผู้ใช้งาน

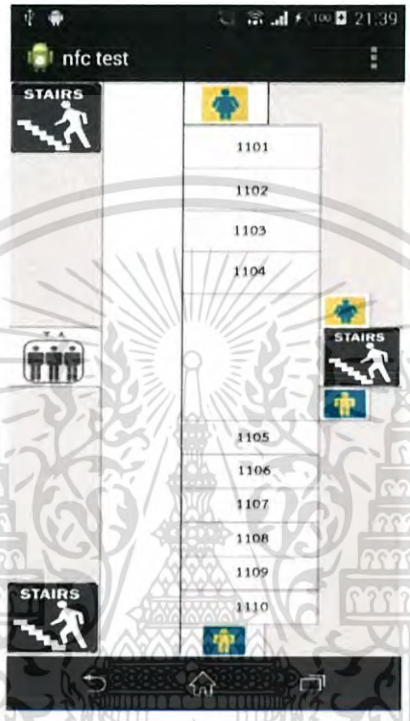
โดยทำการสร้างฟังก์ชัน hideallimage ขึ้นมา เพื่อทำการซ่อนตำแหน่งทั้งหมดในแผนที่ (รูปหมุดสีแดง ดังแสดงในรูปที่ 3.8) ฟังก์ชัน hideallimage ดังแสดงในรูปที่ 3.9 จะทำให้ได้ User Interface ดังแสดงในรูปที่ 3.10

รูปที่ 3.8 หมุดสีแดงใช้เพื่อแสดงตำแหน่งปัจจุบันของผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
public void hideallimage()
{
        here.setVisibility(View.INVISIBLE);
        here2.setVisibility(View.INVISIBLE);
}
```

รูปที่ 3.9 โค้ดแสดงฟังก์ชัน hideallimage



รูปที่ 3.10 User Interface หลังจากใช้ฟังก์ชัน hideallimage

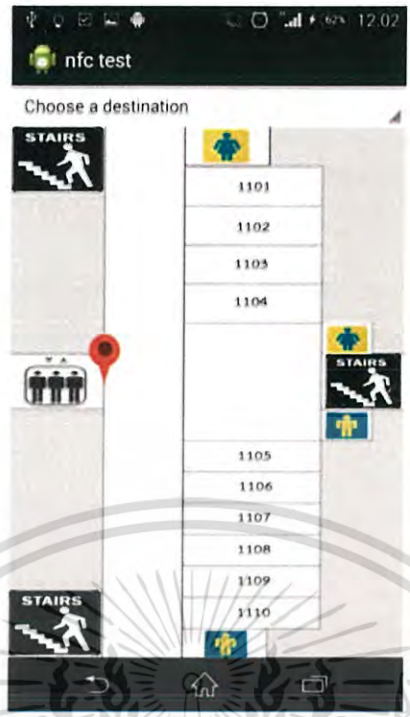
จากนั้นทำการสร้างฟังก์ชัน showimage ขึ้นมา โดยกำหนดเงื่อนไขว่าถ้าข้อมูลที่ได้รับมานั้นตรงกับตำแหน่งใดที่ได้บันทึกไว้ จะทำการแสดงตำแหน่งนั้นขึ้นมา ซึ่งเป็นตำแหน่งปัจจุบันของผู้ใช้งานนั่นเอง ดังแสดงในรูปที่ 3.11 จะทำให้ได้ User Interface ดังแสดงในรูปที่ 3.12

```
public void showimage()
{

        if(tagdata.equals("E12-lift")){
        here.setVisibility(View.VISIBLE);
        }
        else if(tagdata.equals("E12-1106"))
        {
        here2.setVisibility(View.VISIBLE);
        }
}
```

รูปที่ 3.11 โค้ดแสดงฟังก์ชัน showimage

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 User Interface หลังจากใช้ฟังก์ชัน showimage

จากนั้นส่วนประมวลผลจะทำการแสดงรายละเอียดของร้านค้าให้ตรงกับข้อมูลที่ได้รับมา โดยการสร้างฟังก์ชัน showdialog ขึ้นมา โดยกำหนดเงื่อนไขว่าถ้าข้อมูลที่ได้รับมานั้น ตรงกับข้อมูลที่ได้บันทึกไว้ จะทำการแสดงข้อมูลของร้านค้านั้นขึ้นมา ซึ่งเป็นรายละเอียดของร้านค้าในตำแหน่งปัจจุบันของผู้ใช้งานนั่นเอง โค้ดแสดงดังรูปที่ 3.13

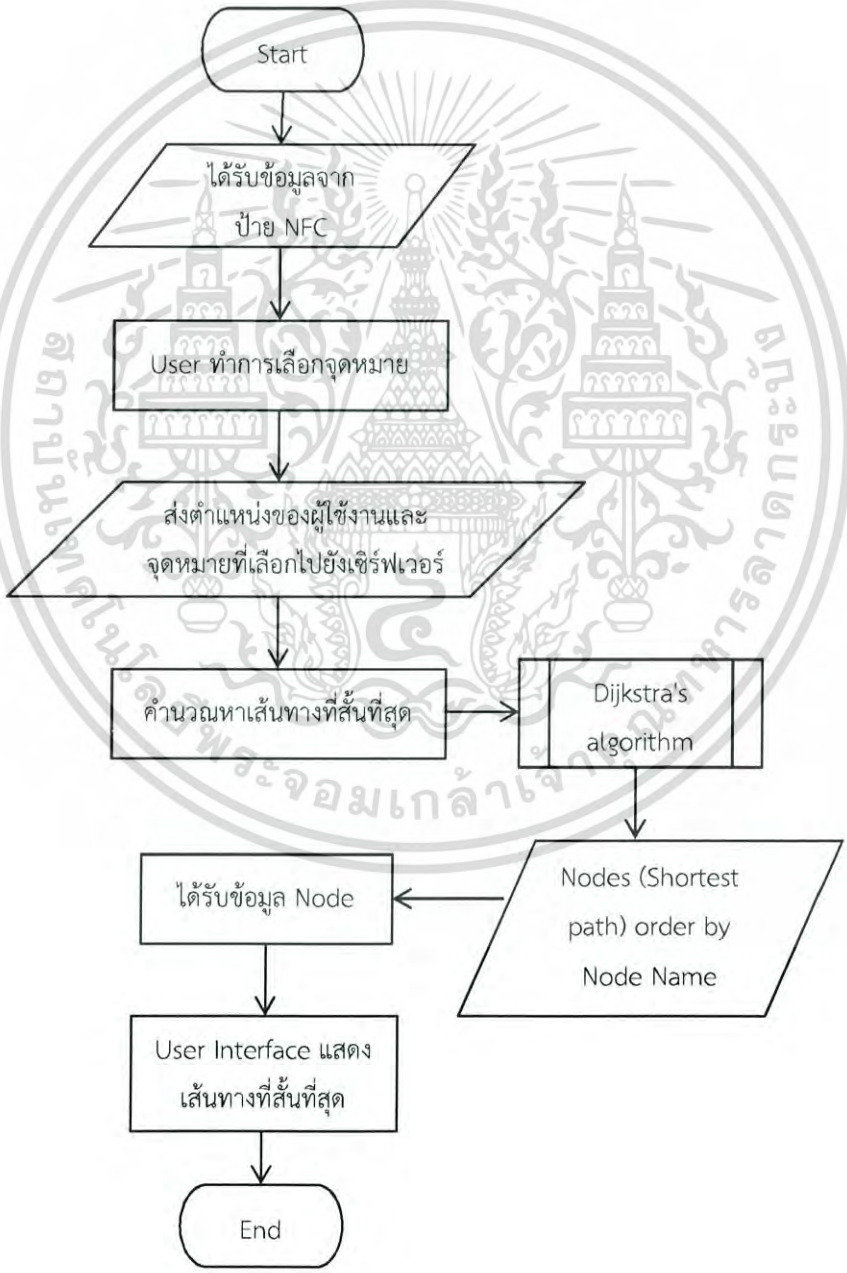
```
public void showdialog()
        {   if(tagdata.equals("E12-Lift")){
AlertDialog.Builder alertDialogBuilder = new
AlertDialog.Builder(context);
            // set title
            alertDialogBuilder.setTitle("Detail");
            // set dialog message
            alertDialogBuilder
            .setMessage("Detail E12-Lift Click back to exit!")
            .setCancelable(false)
            .setNegativeButton("Back",new
            DialogInterface.OnClickListener() {
      public void onClick(DialogInterface dialog,int id) {
                  // if this button is clicked, just close
                  // the dialog box and do nothing
            dialog.cancel();    }});
                  // create alert dialog
      AlertDialog alertDialog = alertDialogBuilder.create();
                  // show it
                  alertDialog.show(); }
```

รูปที่ 3.13 โค้ดแสดงฟังก์ชัน showdialog

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.5 การแสดงเส้นทางที่สั้นที่สุดซึ่งคำนวณโดย Dijkstra's algorithm

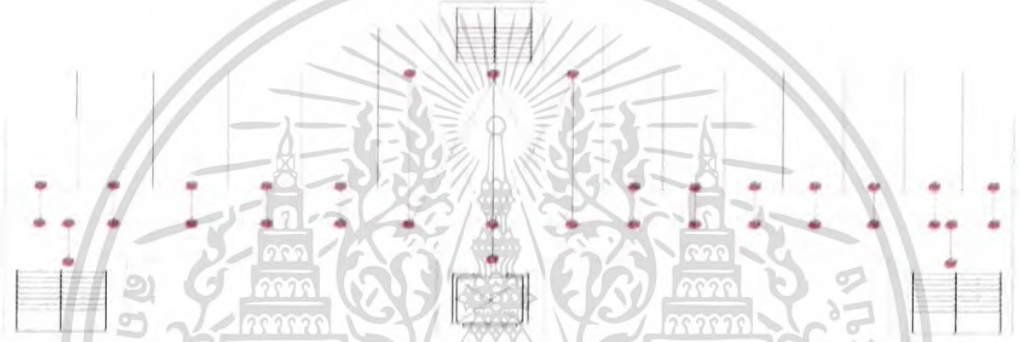
การแสดงเส้นทางที่สั้นที่สุดจากต้นทางไปยังปลายทางนั้น ได้ใช้ทฤษฎีกราฟ โดยสมาร์ตโฟนได้รับข้อมูลจากป้าย NFC ซึ่งก็คือตำแหน่งปัจจุบันของผู้ใช้งาน หลังจากผู้ใช้งานทำการเลือกจุดหมายปลายทางแล้ว แอปพลิเคชั่นจะทำการส่งตำแหน่งต้นทางและปลายทางไปยังเซิร์ฟเวอร์เซิร์ฟเวอร์จากนั้นเซิร์ฟเวอร์จะคำนวณเส้นทางที่สั้นที่สุดโดยใช้ Dijkstra's algorithm ผลลัพธ์สุดท้ายที่ได้จะเป็นชื่อของ Node ตั้งแต่ต้นทางไปถึงปลายทาง แล้วทำการส่งกลับมาที่แอปพลิเคชั่น เมื่อแอปพลิเคชั่นได้รับข้อมูลแล้ว User Interface จะทำการแสดงเส้นทางที่สั้นที่สุดให้ผู้ใช้งาน ซึ่งมีขั้นตอนการทำงานดังแสดงในรูปที่ 3.14



รูปที่ 3.14 Flowchart การแสดงเส้นทางที่สั้นที่สุด

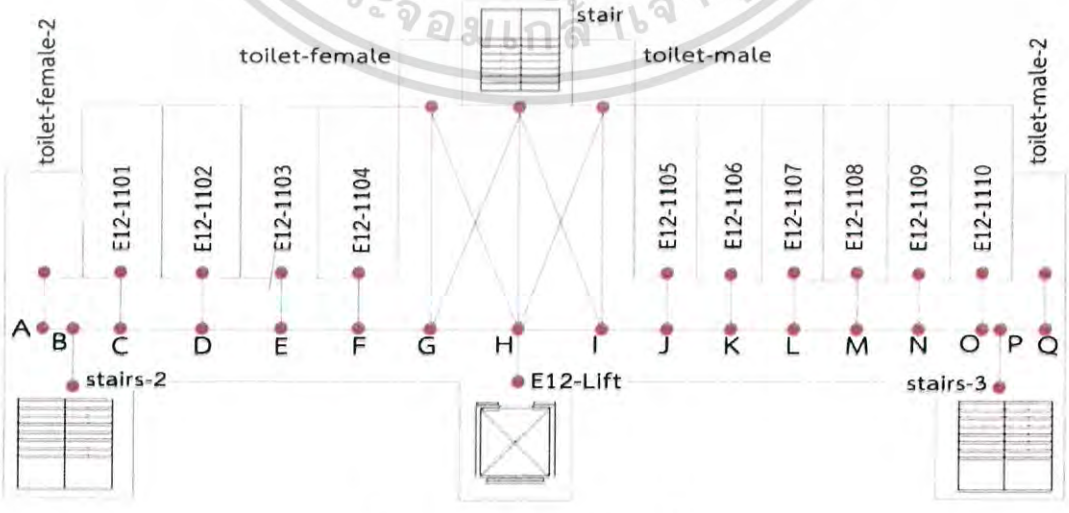
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยการทำงานของทฤษฎีกราฟนั้น ประกอบด้วยโหนด (Vertex) และเส้นเชื่อม (Edge) ที่เชื่อมต่อกันในลักษณะของโครงสร้างข้อมูลต้นไม้ โดยที่โหนดจะแทนตำแหน่งภายในอาคารและบอกข้อมูลที่เกี่ยวข้องกับตำแหน่งนั้น แผนที่ในอาคารจะประกอบไปด้วย ห้องขนาดเล็ก ห้องขนาดใหญ่ ทางเดิน บันได และลิฟต์ ซึ่งทั้งหมดนี้จะต้องถูกแทนที่ด้วยโหนด ส่วนเส้นเชื่อมเป็นเส้นที่ผู้ใช้งานต้องเดินทางระหว่าง 2 โหนด ผู้ใช้งานสามารถเดินทางจากโหนดหนึ่งไปอีกโหนดหนึ่งได้ก็ต่อเมื่อ โหนดนั้นๆมีเส้นเชื่อมถึงกันเท่านั้น ในกราฟของแผนที่ภายในอาคารเรียนรวม 12 ชั้น ที่ชั้น 11 นั้น ได้กำหนดระยะทางขึ้นเองตามความเหมาะสมของสถานที่จริง โดยโหนดห้องจะถูกกำหนดไว้ที่หน้าประตูห้อง ทางเดินจะถูกกำหนดเป็นเส้นเชื่อมแนวตรงหรือแนวนอนเท่านั้น ส่วนพื้นที่ขนาดใหญ่จะมีเส้นเฉียงประกอบด้วย ดังแสดงในรูปที่ 3.15



รูปที่ 3.15 กราฟโหนดและเส้นเชื่อมของแผนที่อาคารเรียนรวม 12 ชั้น ที่ชั้น 11

การกำหนดชื่อของโหนด ถ้าเป็นโหนดที่อยู่หน้าประตู จะเป็นชื่อของห้องนั้นๆ โหนดบริเวณทางเดินเป็นตัวอักษรภาษาอังกฤษตั้งแต่ A-Q บันไดเป็นชื่อ stairs ลิฟต์เป็นชื่อ E12-Lift ห้องน้ำหญิงและชายเป็น toilet-female และ toilet-male ดังแสดงในรูปที่ 3.16



รูปที่ 3.16 กราฟชื่อของโหนดทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 เครื่องมือที่ใช้ในการทดลอง

3.2.1 โทรศัพท์มือถือ

โทรศัพท์มือถือที่สามารถใช้แอปพลิเคชั่นได้ ต้องมีระบบ NFC ในที่นี้ใช้ยี่ห้อ Sony Xperia Z ดังแสดงในรูปที่ 3.17

รูปที่ 3.17 โทรศัพท์มือถือยี่ห้อ Sony Xperia Z [6]

3.2.2 ป้าย NFC

ใช้บัตรไมแฟร์ คลาสสิก (Mifare Classic) ความจุ 1 KB ดังแสดงในรูปที่ 3.18

รูปที่ 3.18 บัตร Mifare Classic ความจุ 1 KB [7]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 โปรแกรมอีคลิป (Eclipse)

เป็น platform เขียนโปรแกรมแอนดรอยด์ โดยมีสัญลักษณ์ดังรูปที่ 3.19

รูปที่ 3.19 สัญลักษณ์ของโปรแกรมอีคลิป (Eclipse) [9]

3.2.4 โปรแกรม TagWriter

การเขียนข้อมูลลงในป้าย NFC ใช้แอปพลิเคชั่น TagWriter ซึ่ง Function ที่ใช้เขียนคือ Plain text โดย User Interface ดังแสดงในรูปที่ 3.20

รูปที่ 3.20 User Interface ของแอปพลิเคชั่น TagWriter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

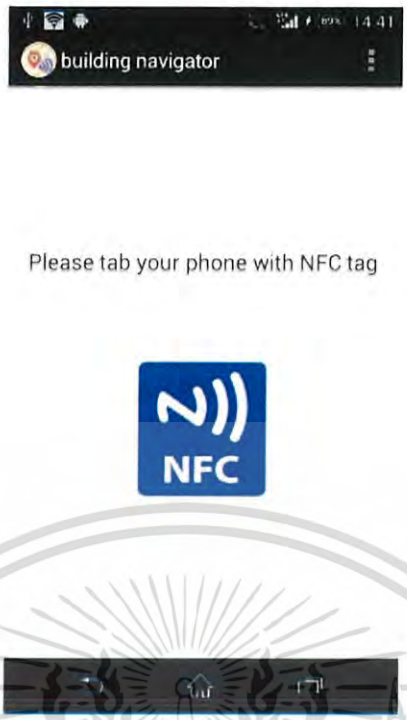
บทที่ 4
ผลการทดลอง

4.1 ผลการทดสอบสมาร์ตโฟนสัมผัสกับป้าย NFC

การทดสอบเมื่อสมาร์ตโฟนสัมผัสกับป้าย NFC นั้น หากสมาร์ตโฟนไม่ได้เปิดการใช้งาน NFC แอปพลิเคชั่นทำการสลับหน้า User Interface ไปยังหน้า Wireless & networks และแสดงข้อความแจ้งเตือนขึ้นมาว่า “Please turn on NFC” เพื่อทำการตั้งค่า NFC ให้มีการเปิดใช้งาน ดังแสดงในรูปที่ 4.1 และเมื่อผู้ใช้งานเปิด NFC ให้มีการใช้งานแล้ว ผลลัพธ์ User Interface ของแอปพลิเคชั่นแสดงดังรูปที่ 4.2

รูปที่ 4.1 User Interface หน้า Wireless & networks

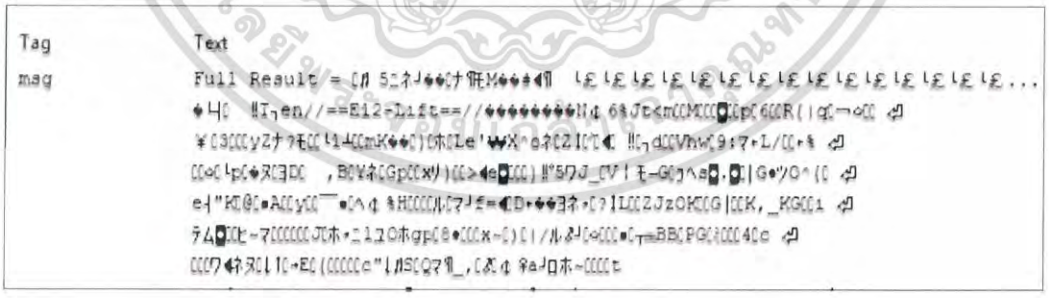
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 หน้า User Interface เมื่อเข้าแอปพลิเคชั่น (และมีการเปิดใช้งาน NFC แล้ว)

4.2 ผลการทดสอบเมื่อสมาร์ตโฟนได้รับข้อมูลจากป้าย NFC

ผลการทดสอบเมื่อสมาร์ตโฟนได้รับข้อมูลจากป้าย NFC นั้น เมื่อสมาร์ตโฟนสัมผัสกับป้าย NFC แล้ว จะได้รับข้อมูลเข้ามา ซึ่งมีข้อมูลที่ไม่ต้องการอยู่ด้วย ผลลัพธ์ที่ได้ทั้งหมดดังแสดงในรูปที่ 4.3 ดังนั้นจึงต้องทำการ split ข้อมูล เพื่อให้ได้ผลลัพธ์สุดท้ายที่ต้องการ ดังแสดงในรูปที่ 4.4



รูปที่ 4.3 ผลลัพธ์ทั้งหมดเมื่อสมาร์ตโฟนสัมผัสป้าย NFC

Tag	Text
result	E12-Lift

รูปที่ 4.4 ผลลัพธ์สุดท้ายที่ได้เมื่อทำการ split แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดสอบการส่งข้อมูลที่ได้รับไปยังส่วนประมวลผล

ผลการทดสอบการส่งข้อมูลที่ได้รับไปยังส่วนประมวลผลนั้น ได้ใส่โค้ดการส่งข้อมูลไปยังส่วนประมวลผลไว้ในส่วนการอ่านข้อมูล ซึ่งส่วนการอ่านข้อมูลจะเรียกว่า MainActivity ดังแสดงในรูปที่ 4.5 และใส่โค้ดการเรียกใช้ข้อมูลที่ส่งมาจากส่วนการอ่านข้อมูลไว้ในส่วนประมวลผล ซึ่งส่วนประมวลผลจะเรียกว่า SecondActivity ดังแสดงในรูปที่ 4.6 ทำให้เมื่อสมาร์ตโฟนได้รับข้อมูลจากป้าย NFC แล้ว ส่วนประมวลผลสามารถเรียกใช้ข้อมูลนั้นได้ ผลลัพธ์ดังแสดงในรูปที่ 4.7

MainActivity.java

```
Intent i = new Intent(getApplicationContext(), Second.class);
i.putExtra("fromnfc",text2);
startActivity(i);
```

รูปที่ 4.5 โค้ดการส่งข้อมูลไปยังส่วนประมวลผล

Second.java

```
Bundle extras = getIntent().getExtras();
tagdata = extras.getString("fromnfc");

Log.v("finalresult", tagdata);
```

รูปที่ 4.6 โค้ดการเรียกใช้ข้อมูลที่ส่งมาจากส่วนการอ่านข้อมูล

LogCat

ots Java regexes. Prefix with pid:, app:, tag: or text: to limit scope.

PID	TID	Application	Tag	Text
23833	23833	com.example.nfc...	finalresult	E12-Lift

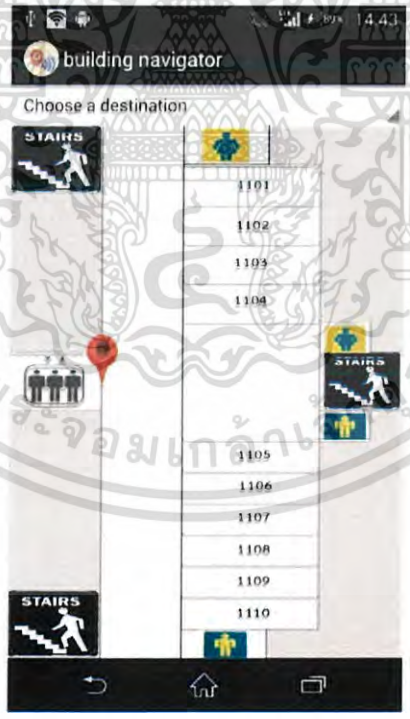
รูปที่ 4.7 ผลลัพธ์ของส่วนประมวลผลที่ได้รับข้อมูลจากส่วนการอ่านข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ผลการทดสอบการแสดงตำแหน่งปัจจุบันของผู้ใช้งานและรายละเอียดของร้านค้า ให้ตรงกับข้อมูลที่ได้รับจากป้าย NFC

ผลการทดสอบการแสดงตำแหน่งปัจจุบันของผู้ใช้งานและรายละเอียดของร้านค้าให้ตรงกับข้อมูลที่ได้รับจากป้าย NFC นั้น จากการออกแบบระบบ ได้ทำการสร้างฟังก์ชัน hideallimage เพื่อซ่อนรูปภาพทั้งหมดก่อน และสร้างฟังก์ชัน showimage เพื่อเลือกเฉพาะรูปภาพที่ต้องการแสดง ทำให้ได้ผลลัพธ์คือ ถ้าสมาร์ตโฟนสัมผัสกับป้าย NFC ที่หน้าลิฟต์ ผลลัพธ์ดังแสดงในรูปที่ 4.8 User Interface ของแอปพลิเคชั่นดังแสดงในรูปที่ 4.9 และถ้าสมาร์ตโฟนสัมผัสกับป้าย NFC ที่หน้าห้อง E12-1106 ผลลัพธ์ดังแสดงในรูปที่ 4.10 User Interface ของแอปพลิเคชั่นดังแสดงในรูปที่ 4.11

รูปที่ 4.8 ผลลัพธ์เมื่อสมาร์ตโฟนสัมผัสป้าย NFC หน้าลิฟต์

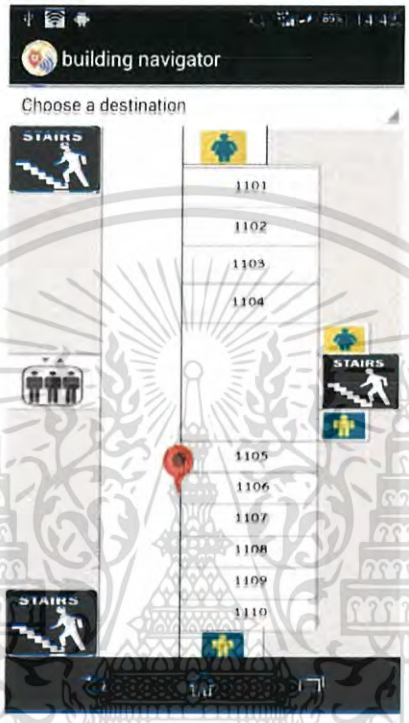


รูปที่ 4.9 User Interface แสดงตำแหน่งปัจจุบันเมื่อสัมผัสป้าย NFC ที่หน้าลิฟต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Tag	Text
result	E12-1106

รูปที่ 4.10 ผลลัพธ์เมื่อสมาร์ตโฟนสัมผัสป้าย NFC หน้าห้อง E12-1106



รูปที่ 4.11 User Interface แสดงตำแหน่งปัจจุบันเมื่อสัมผัสป้าย NFC ที่หน้าห้อง E12-1106

จากการออกแบบระบบได้ทำการสร้างฟังก์ชัน showdialog เพื่อแสดงรายละเอียดของร้านค้า ณ ตำแหน่งของผู้ใช้งาน เช่น เมื่อผู้ใช้งานนำสมาร์ตโฟนสัมผัสกับป้าย NFC ที่หน้าลิฟต์ User Interface ของแอปพลิเคชั่นจะแสดงดังรูปที่ 4.12 และเมื่อนำสมาร์ตโฟนสัมผัสกับป้าย NFC ที่หน้าห้อง E12-1106 User Interface ของแอปพลิเคชั่นจะแสดงดังรูปที่ 4.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

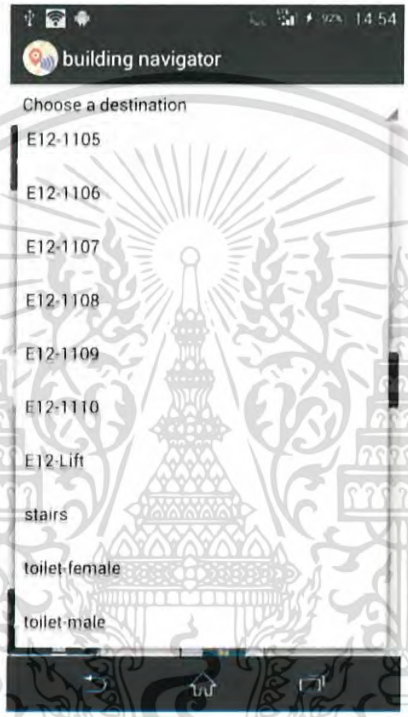
รูปที่ 4.12 User Interface แสดงรายละเอียดเมื่อสัมผัสป้าย NFC ที่หน้าลิฟต์

รูปที่ 4.13 User Interface แสดงรายละเอียดเมื่อสัมผัสป้าย NFC ที่หน้าห้อง E12-1106

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ผลการทดสอบการแสดงเส้นทางที่สั้นที่สุดซึ่งคำนวณโดย Dijkstra's algorithm

ผลการทดสอบการแสดงเส้นทางที่สั้นที่สุดซึ่งคำนวณโดย Dijkstra's algorithm เริ่มจากสมาร์ตโฟนได้รับตำแหน่งปัจจุบันจากป้าย NFC และจุดหมายปลายทางที่เลือกโดยผู้ใช้งานจาก dropdown ดังแสดงในรูปที่ 4.14 ในที่นี้ยกตัวอย่างจุดต้นทางเป็น E12-1106 และจุดปลายทางเป็น toilet-female หลังจากผู้ใช้งานทำการเลือกจุดหมายปลายทางแล้ว แอปพลิเคชั่นจะทำการส่งตำแหน่งต้นทางและปลายทางไปยังเซิร์ฟเวอร์



รูปที่ 4.14 User Interface แสดงการเลือกปลายทางจาก dropdown

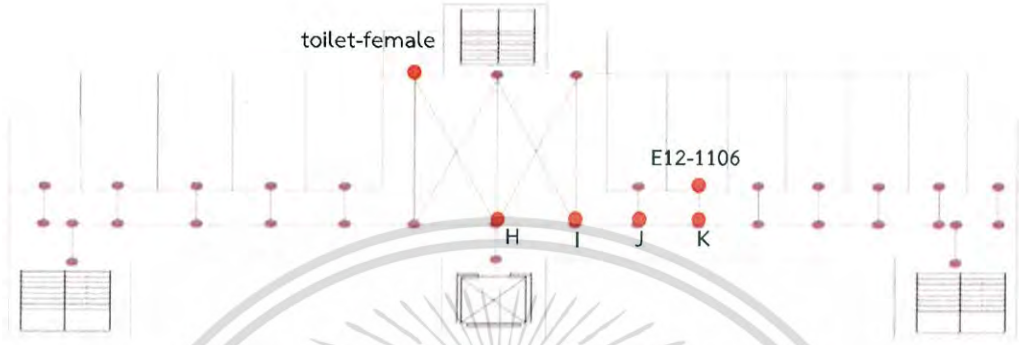
เมื่อเซิร์ฟเวอร์คำนวณเส้นทางแล้ว ผลลัพธ์สุดท้ายที่ได้จะเป็นชื่อของ Node ตั้งแต่ต้นทางไปยังปลายทาง แล้วส่งกลับมาที่แอปพลิเคชั่นซึ่งเป็นข้อมูลที่อยู่ในรูปแบบ JSON เมื่อแอปพลิเคชั่นได้รับข้อมูลแล้วต้องเปลี่ยนข้อมูลจากรูปแบบ JSON ให้อยู่ในรูปของตัวแปร string คำสั่งที่ใช้ ดังรูปที่ 4.15

```
try {
JSONObject json = new JSONObject (str1);
String des = json.getString("path");
} catch (JSONException e) {
// TODO Auto-generated catch block
e.printStackTrace();
            }
```

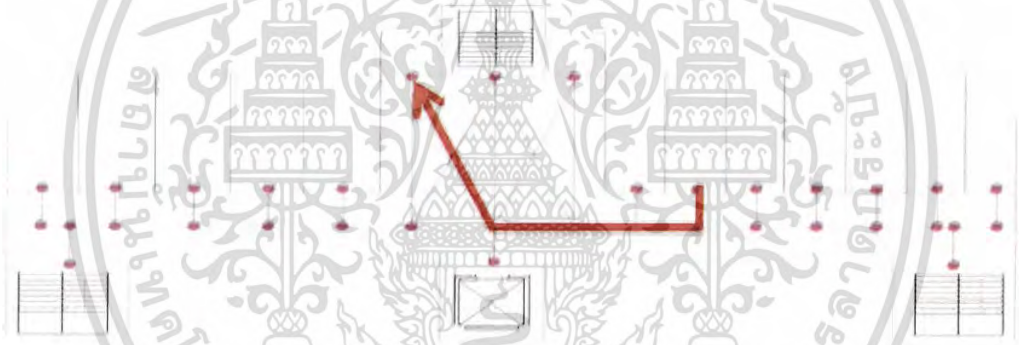
รูปที่ 4.15 โค้ดการเปลี่ยนข้อมูลรูปแบบ JSON ให้เป็นข้อมูลในรูปของตัวแปร string

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

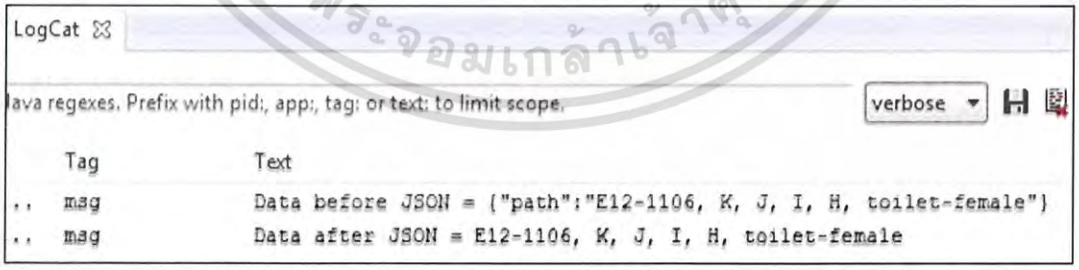
เส้นทางที่สั้นที่สุดที่ได้ เมื่อต้นทางเป็น E12-1106 และปลายทางเป็น toilet-female คือ E12-1106, K, J, I, H, toilet-female โดยโหนดทั้งหมดแสดงดังรูปที่ 4.16 กราฟแสดงเส้นทางแสดงดังรูปที่ 4.17 และผลลัพธ์ที่ได้เมื่อเปลี่ยนข้อมูลรูปแบบ JSON ให้เป็นข้อมูลในรูปของตัวแปร string ดังแสดงในรูปที่ 4.18



รูปที่ 4.16 กราฟแสดงโหนดตั้งแต่ E12-1106 ถึง toilet-female



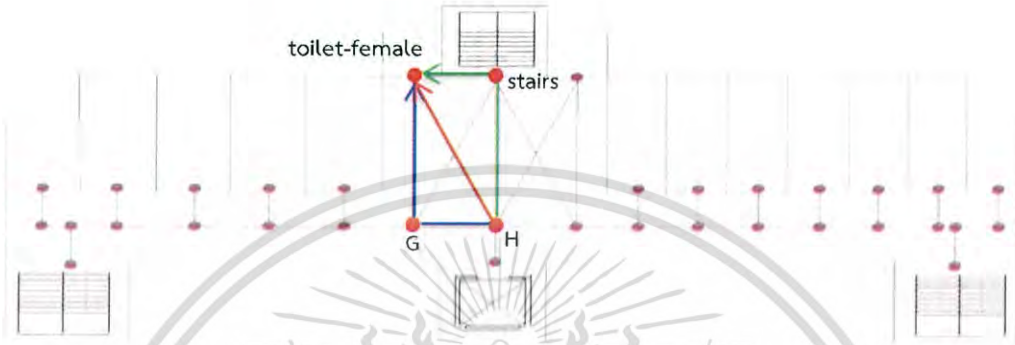
รูปที่ 4.17 กราฟแสดงเส้นทางตั้งแต่ E12-1106 ถึง toilet-female



รูปที่ 4.18 ผลลัพธ์การเปลี่ยนข้อมูลรูปแบบ JSON ให้เป็นข้อมูลในรูปของตัวแปร string

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อต้นทางเป็น E12-1106 และปลายทางเป็น toilet-female จะเห็นว่าเมื่อเลือกใช้ Dijkstra's algorithm แล้ว ทำให้ได้เส้นทางที่สั้นที่สุด โดยเส้นทางที่มาจากโหนด H ไปยังโหนด toilet-female นั้นมีเส้นทางที่ไปได้ทั้งหมด 3 เส้นทาง คือ โหนด H, G, toilet-female โหนด H, stairs, toilet-female และโหนด H, toilet-female ดังแสดงในรูปที่ 4.19 โดย Dijkstra's algorithm คำนวณได้เป็นโหนด H, toilet-female ซึ่งเป็นระยะกระจัด



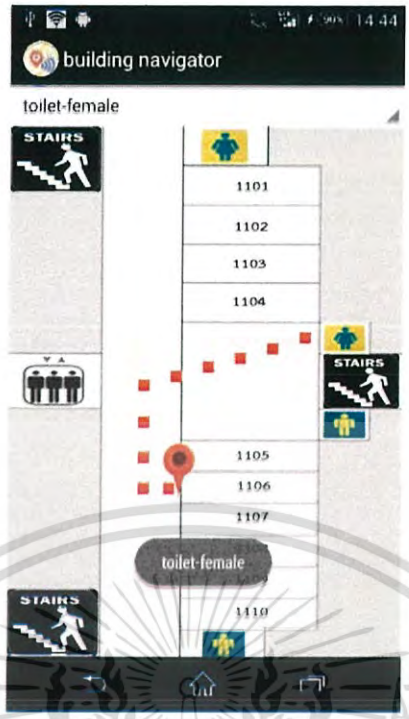
รูปที่ 4.19 กราฟแสดงเส้นทางที่ไปได้ทั้งหมด 3 เส้นทาง

หลังจากแปลงข้อมูลรูปแบบ JSON ให้เป็นข้อมูลในรูปของตัวแปร string แล้ว ต้องทำการ split ข้อมูล โดยใช้คำสั่ง `String[] node = des.split(", ");` ในที่นี้ตัวแปร des คือข้อมูลเส้นทางทั้งหมด จากนั้นเมื่อได้ Node แต่ละตัวแล้ว ทำการเข้าลูป for ดังแสดงในรูปที่ 4.20 เพื่อรับค่าจาก Array ที่ได้เก็บค่าตำแหน่งของแต่ละ Node ไว้ คือ `positionx` และ `positiony` จากนั้นส่งค่า posx และ posy ที่ได้ไปยังฟังก์ชัน showpath เพื่อทำการแสดงเส้นทาง ทำให้ได้ User Interface แสดงเส้นทางดังรูปที่ 4.21

```
int nodecount = node.length;
String nodepath = "";
      for(int i = 0; i < nodecount; i++){
      Integer posx = null;
      Integer posy = null;
            nodepath = node[i];
      posx = myArray.get(node[i]).get("positionx");
      posy = myArray.get(node[i]).get("positiony");
                  showpath(posx,posy);
            }
```

รูปที่ 4.20 โค้ดแสดงการทำงานการรับตำแหน่งจาก Array

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 User Interface แสดงเส้นทางจาก E12-1106 ถึง toilet-female

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

จากโครงงานเรื่องระบบนำทางในอาคารโดยใช้เทคโนโลยีเอ็นเอฟซีบนสมาร์ตโฟน (INDOOR NAVIGATION SYSTEM USING NFC ON SMARTPHONE) ได้นำเสนอเทคโนโลยีและวิธีการสำหรับการสร้างระบบแผนที่ภายในอาคารที่สามารถใช้งานบนสมาร์ตโฟนได้ ในโครงงานนี้สามารถนำไปประยุกต์ใช้กับศูนย์ประชุมหรือห้างสรรพสินค้า ระบบสามารถแบ่งการทำงานออกเป็น 5 ส่วนคือ

1) สมาร์ตโฟนสัมผัสกับป้าย NFC
2) สมาร์ตโฟนได้รับข้อมูลจากป้าย NFC นั้น
3) การส่งข้อมูลที่ได้รับไปยังส่วนประมวลผล
4) การแสดงตำแหน่งปัจจุบันของผู้ใช้งานและรายละเอียดของร้านค้า ให้ตรงกับข้อมูลที่ได้รับจากป้าย NFC
5) การแสดงเส้นทางที่สั้นที่สุดซึ่งคำนวณโดย Dijkstra's algorithm

โดยทำการเขียนแอปพลิเคชั่นให้มีความสามารถในการแสดงแผนที่ภายในอาคาร ตำแหน่งปัจจุบันของผู้ใช้งาน และรายละเอียดของร้านค้าแต่ละร้านได้ ซึ่งใช้วิธีการสื่อสารไร้สายระยะสั้น หรือ NFC ทำให้ระบบมีขนาดเล็ก ราคาถูก ประหยัดพลังงาน มีค่าบำรุงรักษาต่ำ ติดตั้งง่าย และผู้ใช้งานก็สามารถใช้งานได้ง่าย อีกทั้งแอปพลิเคชั่นยังมีความสามารถในการหาเส้นทางที่สั้นที่สุด จากต้นทางไปยังปลายทางให้ผู้ใช้งาน ซึ่งคำนวณด้วยเทคนิคกราฟ โดยใช้ Dijkstra's algorithm ดังนั้นแอปพลิเคชั่นนี้จะช่วยอำนวยความสะดวกให้ผู้ใช้งานในการนำทางสำหรับผู้ที่ไม่เคยไปสถานที่นั้นๆให้ถึงเป้าหมายได้อย่างถูกต้องและรวดเร็ว และเมื่อไปถึงปลายทางแล้วผู้ใช้งานนำสมาร์ตโฟนไปสัมผัสกับป้าย NFC ที่อยู่ปลายทาง แอปพลิเคชั่นจะแสดงรายละเอียดของร้านค้าขึ้นมา ทำให้สะดวกในการบอกรายละเอียดของร้านค้า และลดภาระของพนักงานอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ข้อเสนอแนะ

1) สามารถนำโครงงานนี้ไปประยุกต์ใช้กับพื้นที่ภายในอาคารที่เป็นศูนย์ประชุมหรือห้างสรรพสินค้า เพื่อค้นหาเส้นทางที่ผู้ใช้งานต้องการจะไปร้านค้านั้นๆให้สะดวกมากขึ้น
2) สามารถนำโครงงานนี้ไปพัฒนาในส่วนของ User interface ให้มีความใช้งานง่าย และมีความน่าสนใจมากขึ้นกว่าเดิมได้
3) สามารถนำไปพัฒนาในส่วนของฮาร์ดแวร์ได้ เช่นการทดลองกับสมาร์ตโฟนรุ่นอื่นๆ หรือป้าย NFC ประเภทอื่นๆ เพื่อเพิ่มประสิทธิภาพในการใช้งาน
4) การเพิ่มในส่วนของความถูกต้อง แม่นยำ ยังสามารถพัฒนาในส่วนของการพัฒนาซอฟแวร์ การพัฒนาอัลกอริทึมได้อีก

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] พร้อมเลิศ หล่อวิจิตร. คู่มือเขียนแอพ Android ฉบับสมบูรณ์. พิมพ์ครั้งที่ 1. กรุงเทพฯ : Provision. 2556.

[2] "พัฒนาการของแอนดรอยด์" [ออนไลน์]. เข้าถึงได้จาก: http://th.wikipedia.org/wiki/ประวัติรุ่นของแอนดรอยด์ (สืบค้นวันที่ 25 สิงหาคม 2557)

[3] "ประเภทของระบบปฏิบัติการแอนดรอยด์" [ออนไลน์]. เข้าถึงได้จาก: http://android-basicapp.blogspot.com/2012/12/blog-post.html (สืบค้นวันที่ 27 สิงหาคม 2557)

[4] "สถาปัตยกรรมแอนดรอยด์" [ออนไลน์]. เข้าถึงได้จาก: http://kadroidz.blogspot.com/2012/03/android-architecture.html (สืบค้นวันที่ 27 สิงหาคม 2557)

[5] "ส่วนประกอบของแอปพลิเคชั่น" [ออนไลน์]. เข้าถึงได้จาก: http://lazy4me.blogspot.com/2013/08/droid_509.html (สืบค้นวันที่ 30 สิงหาคม 2557)

[6] "Sony Xperia Z" [ออนไลน์]. เข้าถึงได้จาก: http://www.sony.co.th/product/xperia+z (สืบค้นวันที่ 16 กันยายน 2557)

[7] "NFC (Near Field Communication)" [ออนไลน์]. เข้าถึงได้จาก: https://sites.google.com/site/hukykku/home (สืบค้นวันที่ 3 กันยายน 2557)

[8] "ประโยชน์ของ NFC" [ออนไลน์]. เข้าถึงได้จาก: http://www.tistr.or.th/tistrblog/?p=1480 (สืบค้นวันที่ 3 กันยายน 2557)

[9] "โปรแกรมอีคลิป (Eclipse)" [ออนไลน์]. เข้าถึงได้จาก: http://javaitfun.blogspot.com/2013/09/eclipse.html (สืบค้นวันที่ 10 กันยายน 2557)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[10] "ส่วนประกอบของโปรแกรมอีคลิป" [ออนไลน์]. เข้าถึงได้จาก:
http://www.doesystem.com/ Eclipse-Explorer-ส่วนประกอบต่าง-ๆ-ของ-Eclipse.htm
(สืบค้นวันที่ 10 กันยายน 2557)

[11] "สถาปัตยกรรมของโปรแกรมอีคลิป" [ออนไลน์]. เข้าถึงได้จาก:
http://www.programcreek.com/eclipse-architecture-design/
(สืบค้นวันที่ 11 กันยายน 2557)

[12] "จาวา (Java)" [ออนไลน์]. เข้าถึงได้จาก:
http://th.wikipedia.org/wiki/ภาษาจาวา (สืบค้นวันที่ 14 กันยายน 2557)

[13] "ทฤษฎีกราฟ (Graph)" [ออนไลน์]. เข้าถึงได้จาก: https://www.l3nr.org/posts/251918
(สืบค้นวันที่ 10 มกราคม 2558)

[14] "ขั้นตอนวิธีการทำงานของ Dijkstra's algorithm" [ออนไลน์]. เข้าถึงได้จาก:
http://www.inwprogramming.com/algorithm/shortest-path-dijkstras-algorithm/
(สืบค้นวันที่ 12 มกราคม 2558)

[15] "วิธีการทำให้สมาร์ตโฟนสามารถอ่านป้าย NFC ได้" [ออนไลน์]. เข้าถึงได้จาก:
http://mifareclassicdetectiononandroid.blogspot.com/2011/04/reading-mifare-classic-1k-from-android.html
(สืบค้นวันที่ 1 ตุลาคม 2557)

[16] "การสร้าง Activity ใหม่และการข้ามไปมาระหว่าง Activity" [ออนไลน์]. เข้าถึงได้จาก:
http://www.akexorcist.com/2013/09/android-code-activity-activity.html
(สืบค้นวันที่ 14 ตุลาคม 2557)

[17] "วิธีการตรวจสอบการใช้งาน NFC" [ออนไลน์]. เข้าถึงได้จาก:
http://stackoverflow.com/questions/14957691/how-to-enable-nfc-setting
(สืบค้นวันที่ 16 ตุลาคม 2557)

[18] "การส่งค่าไปยังอีก Activity" [ออนไลน์]. เข้าถึงได้จาก:
http://stackoverflow.com/questions/15245384/android-passing-a-variables-to-another-activity
(สืบค้นวันที่ 29 ตุลาคม 2557)

[19] "Dijkstra's algorithm" [ออนไลน์]. เข้าถึงได้จาก:
http://rosettacode.org/wiki/Dijkstra's_algorithm#PHP
(สืบค้นวันที่ 15 มกราคม 2558)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[20] "การแสดงรายละเอียดของร้านค้า" [ออนไลน์]. เข้าถึงได้จาก:
http://www.mkyong.com/android/android-alert-dialog-example/
(สืบค้นวันที่ 7 กุมภาพันธ์ 2558)

[21] "การส่งค่าและรับค่าระหว่างแอนดรอยด์กับเซิร์ฟเวอร์" [ออนไลน์]. เข้าถึงได้จาก:
http://www.thaicreate.com/mobile/android-httpget-httppost.html
(สืบค้นวันที่ 20 กุมภาพันธ์ 2558)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

โค้ดของโทรศัพท์แอนดรอยด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โค้ดแอปพลิเคชั่น (อ่านข้อมูลจากป้าย NFC)

```
package com.example.nfctest;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.net.URI;

import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.DefaultHttpClient;

import android.app.PendingIntent;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.IntentFilter.MalformedMimeTypeException;
import android.nfc.NfcAdapter;
import android.nfc.Tag;
import android.nfc.tech.MifareClassic;
import android.os.AsyncTask;
import android.os.Bundle;
import android.provider.Settings;
import android.support.v7.app.ActionBarActivity;
import android.util.Log;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.TextView;
import android.widget.Toast;


public class MainActivity extends ActionBarActivity {

      NfcAdapter NFC;
      PendingIntent pIntent;
      IntentFilter filter;
      IntentFilter[] filters;
      String[][] namelist;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);


        try{


            NFC = NfcAdapter.getDefaultAdapter(this);

            if(NFC!=null)
            {
            if(NFC.isEnabled())
            {
            //Nfc settings are enabled
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            }
            else
            {
                  Toast.makeText(getApplicationContext(), "Please turn on
                  NFC",
                  Toast.LENGTH_LONG).show();
                  Intent setnfc = new
Intent(Settings.ACTION_NFC_SETTINGS);
                  startActivity(setnfc);
                  //Nfc Settings are not enabled
            }
            }

                  pIntent = PendingIntent.getActivity(this, 0, new
Intent(this, getClass()).addFlags(Intent.FLAG_ACTIVITY_SINGLE_TOP), 0);

                  filter = new
IntentFilter(NfcAdapter.ACTION_TECH_DISCOVERED);

                  try{   filter.addDataType("*//*");      }
                  catch (MalformedMimeTypeException e) {
                        throw new RuntimeException("fail", e);
                  }

                  filters = new IntentFilter[] {
                              filter,
                  };

                  namelist = new String[][] { new String[] {
MifareClassic.class.getName() } };
            }catch(Exception e){ Log.v("msg", "Err : "+e); }

    }

    @Override
      public void onResume()
            {
            super.onResume();
            try{
                  NFC.enableForegroundDispatch(this, pIntent, filters,
                  namelist);
            }catch(Exception e){ Log.v("msg", "Err : "+e); }
      }

      @Override
      public void onNewIntent(Intent intent){
            readNfcData(intent);
      }



    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is
present.
      MenuInflater inflater = getMenuInflater();
      inflater.inflate(R.menu.activity_main_actions, menu);
       return true;
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    public void readNfcData(Intent intent) {
            String action = intent.getAction();
            Log.v("msg", "Doing...");
            if (NfcAdapter.ACTION_TECH_DISCOVERED.equals(action)) {
                    Log.v("msg", "Search!");
                Tag tagFromIntent =
intent.getParcelableExtra(NfcAdapter.EXTRA_TAG);
                MifareClassic mytag = MifareClassic.get(tagFromIntent);
                byte[] data;
                String result = "";
                try {
                    mytag.connect();
                    boolean auth = false;
                    String cardData = null;
                    int totalSec = mytag.getSectorCount();
                    Log.v("msg", "Sector = "+totalSec);
                    int totalBlock = 0;
                    int currBlock = 0;

                    //read sector
                    for(int j = 0; j < totalSec; j++){
                        auth = mytag.authenticateSectorWithKeyB(j,
MifareClassic.KEY_DEFAULT);
                        if(auth){
                            Log.v("msg", "Authen OK : j="+j);
                            totalBlock =
mytag.getBlockCountInSector(j);
                            Log.v("msg", "totalBlock =
"+totalBlock);
                            int i;
                            currBlock = mytag.sectorToBlock(j);

                            //read block
                            for(i = 0; i < 3 ; i++){

                                data =
mytag.readBlock(currBlock);
                                Log.v("msg", "data = "+data);
                                result +=
getReadableData(data);
                                Log.v("msg", "i = "+i);
                                currBlock++;
                            }
                        }else{
                            Log.v("msg", "Section "+j+" : Authen
Fail.");
                        }
                        Log.v("msg", "j = "+j);
                    }

                    Log.v("msg", "Full Result = "+result);

                    String[] read_data = result.split("//==");
                    if(read_data.length == 2){
                        Log.v("msg", "Found //==");
                        String[] read_data2 =
                        read_data[1].split("==//");
                        if(read_data2.length == 2){
                            Log.v("msg", "Found ==//");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                        TextView tv =
(TextView)findViewById(R.id.textView1);
                                        String text2 = read_data2[0];



                                        //tv.setText(text2);


                            Intent i = new Intent(getApplicationContext(),
Second.class);
                            i.putExtra("fromnfc",text2);
                            startActivity(i);
                                        //send to second page


                                        Log.v("result", text2);


                                    }
                                }
                            }catch (IOException e){
                                Log.e("TAG", e.getLocalizedMessage());
                                Log.v("msg", "Tag Lost");
                            }

                        }
                    }

                public String getReadableData(byte[] _bytes)
                    {
                        String file_string = "";
                        for(int i = 0; i < _bytes.length; i++)
                        {
                            file_string += (char)_bytes[i];
                        }
                        return file_string;
                    }

                @Override
                public boolean onOptionsItemSelected(MenuItem item) {
                    // Handle action bar item clicks here. The action bar will
                    // automatically handle clicks on the Home/Up button, so long
                    // as you specify a parent activity in AndroidManifest.xml.
                    int id = item.getItemId();

                    if (id == R.id.action_settings) {

                        return true;

                    }
                    else if (id == R.id.action_detail) {

                        return true;

                    }
                    return super.onOptionsItemSelected(item);
                    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โค้ดแอปพลิเคชั่น (นำทางในอาคาร)

```
package com.example.nfctest;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.URI;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.Iterator;
import java.util.concurrent.Delayed;
import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.DefaultHttpClient;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import org.json.JSONStringer;

import android.R.string;
import android.app.ActionBar.LayoutParams;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.ContentValues;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.os.AsyncTask;
import android.os.Bundle;
import android.os.Handler;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.View;
import android.widget.AbsoluteLayout;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.RelativeLayout;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;
import android.support.v7.app.ActionBarActivity;


public class Second extends  Activity {

	private Spinner spinner1;
	private ImageView here,here2;
	String tagdata;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
	int currentpos  = 0;
	ImageView
temp0,temp1,temp2,temp3,temp4,temp5,temp6,temp7,temp8,temp9,temp10
	,temp11,temp12,temp13,temp14,temp15,temp16,temp17,temp18,temp19,temp
20,temp21,temp22,temp23,temp24;
	ImageView[] temp =
{temp0,temp1,temp2,temp3,temp4,temp5,temp6,temp7,temp8,temp9,temp10
	,temp11,temp12,temp13,temp14,temp15,temp16,temp17,temp18,temp19,temp
20,temp21,temp22,temp23,temp24} ;
	final Context context = this;
	private Button button;
	//TextView result;
	int nodetotal;


	@Override
	protected void onCreate(Bundle savedInstanceState) {
		// TODO Auto-generated method stub
		super.onCreate(savedInstanceState);
		setContentView(R.layout.second_page);

		Bundle extras = getIntent().getExtras();
		tagdata = extras.getString("fromnfc");


		addListenerOnSpinnerItemSelection(tagdata);

		here = (ImageView) findViewById(R.id.imageView1);
		here2 = (ImageView) findViewById(R.id.imageView2);
				Log.v("finalresult", tagdata);


		hideallimage();
		showimage();
		showdialog();


	}


	public void showpath (int posx, int posy){
		RelativeLayout path = (RelativeLayout)
findViewById(R.id.relativeLayout);
		ImageView iv = new ImageView(this);
		iv.setBackgroundColor(Color.RED);
		RelativeLayout.LayoutParams params = new
RelativeLayout.LayoutParams(30, 30);
		params.leftMargin = posx;
		params.topMargin = posy;
		path.addView(iv, params);

		Log.v("imagev", "currentpos = "+currentpos);
		  temp[currentpos] = iv;
		  currentpos++;
		  nodetotal = currentpos;
	}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        public void showoutput(String str1){

                try {
                        Log.v("msg", "Data before JSON = "+str1);
                        JSONObject json = new JSONObject (str1);
                        String des = json.getString("path");

                        //result = (TextView)findViewById(R.id.textView1);
                        //result.setText(des);
                        Log.v("msg", "Data after JSON = "+des);
                        splitdata(des);

                } catch (JSONException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                }
        }


          public void addListenerOnSpinnerItemSelection(final String
tagdatas) {
                        spinner1 = (Spinner) findViewById(R.id.spinner1);
                        spinner1.setSelection(0, false);
                        spinner1.setOnItemSelectedListener(new
OnItemSelectedListener() {
                            @Override
                            public void onItemSelected(AdapterView<?> parent,
View selectedItemView, int pos, long id) {
                                // your code here
                                Toast.makeText(parent.getContext(),
                                        parent.getItemAtPosition(pos).toString(),
                                        Toast.LENGTH_SHORT).show();

                                        Log.v("msg", "data from tag nfc is
"+tagdatas);

                                new
HttpTask().execute("http://piamrak.hypercube24.com/dijkstra.php?source="+ta
gdatas+"&destination="+parent.getItemAtPosition(pos).toString()+"");

                                //new
HttpTask().execute("http://piamrak.hypercube24.com/outputDATA.php");

                            }

                            @Override
                            public void onNothingSelected(AdapterView<?>
parentView) {
                                // your code here

}
                        });}

          class HttpTaskdelay extends AsyncTask<String, Integer, String>{

                @Override
                protected String doInBackground(String... uri) {
                        // TODO Auto-generated method stub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                BufferedReader in = null;
                try{
                        HttpClient httpclient = new DefaultHttpClient();

                 HttpGet request = new HttpGet();
                 URI website = new URI(uri[0]);
                 request.setURI(website);

                 HttpResponse response = httpclient.execute(request);

                 in = new BufferedReader(new InputStreamReader(
                 response.getEntity().getContent()));
                 String line = in.readLine();

                 Log.e("msgg", "Datadelay = "+line);
                }catch(Exception e){
                        Log.e("msg", "Error");
                }
                return null;
        }
    }

class HttpTask extends AsyncTask<String, Integer, String>{

    @Override
    protected String doInBackground(String... uri) {
            // TODO Auto-generated method stub

            BufferedReader in = null;
            try{
                    HttpClient httpclient = new DefaultHttpClient();

                 HttpGet request = new HttpGet();
                 URI website = new URI(uri[0]);
                 request.setURI(website);

                 HttpResponse response = httpclient.execute(request);

                 in = new BufferedReader(new InputStreamReader(
                 response.getEntity().getContent()));
                 String line = in.readLine();

                 Log.e("msgg", "Data from output = "+line);
                 return line;
                }catch(Exception e){
                        Log.e("msg", "Error");
                }
                return null;
        }

        protected void onPostExecute (String str1){
                showoutput (str1);


        }
    }

public void hideallimage()
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                here.setVisibility(View.INVISIBLE);
                here2.setVisibility(View.INVISIBLE);
        }
        public void showimage()
        {

                if(tagdata.equals("E12-Lift")){
                here.setVisibility(View.VISIBLE);
                }
                else if(tagdata.equals("E12-1106"))
                {
            here2.setVisibility(View.VISIBLE);
                }
        }

        public void showdialog()
        {
                if(tagdata.equals("E12-Lift")){
             AlertDialog.Builder alertDialogBuilder = new
AlertDialog.Builder(context);

                                // set title
                                alertDialogBuilder.setTitle("Detail");
                                //set image
                                LayoutInflater factory =
LayoutInflater.from(context);
                                final View view =
factory.inflate(R.layout.dialog, null);
                                ImageView image=
(ImageView)view.findViewById(R.id.imageViewlift);
                                image.setImageResource(R.drawable.lift11);
                                alertDialogBuilder.setView(view);
                                // set dialog message
                                alertDialogBuilder
                                .setMessage("E12-Elevator\nE12 Building, 11th
floor in front of the elevator\n\nClick back to exit!")
                                .setCancelable(false)
                                .setNegativeButton("Back",new
DialogInterface.OnClickListener() {
                                        public void onClick(DialogInterface
dialog,int id) {
                                                // if this button is clicked, just
close
                                                // the dialog box and do nothing
                                                dialog.cancel();
                                                }
                                        });

                                        // create alert dialog
                                        AlertDialog alertDialog =
alertDialogBuilder.create();
                                        alertDialog.setIcon(R.drawable.info);

                                        // show it
                                        alertDialog.show();
                }


                else if(tagdata.equals("E12-1106")){
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                AlertDialog.Builder alertDialogBuilder = new
AlertDialog.Builder(
                                context);
                    alertDialogBuilder.setTitle("Detail");
                    LayoutInflater factory = LayoutInflater.from(context);
                    final View view = factory.inflate(R.layout.dialog,
null);
                    ImageView image=
(ImageView)view.findViewById(R.id.imageView1106);
                    image.setImageResource(R.drawable.room1106);
                    alertDialogBuilder.setView(view);
                    alertDialogBuilder
                            .setMessage("E12-1106\nผศ.บุญชนะ ภู่ระหงษ์ \n(Asst.Prof.
Boonchana Purahong)\n\nClick back to exit!")
                            .setCancelable(false)
                            .setNegativeButton("Back",new
DialogInterface.OnClickListener() {
                                public void onClick(DialogInterface
dialog,int id) {
                                    dialog.cancel();
                                }
                            });
                            AlertDialog alertDialog =
alertDialogBuilder.create();
                            alertDialog.setIcon(R.drawable.info);
                            alertDialog.show();
        }}

        public void splitdata(String des)
        {
            String[] node = des.split(", ");
            //Log.v("testsplit", ""+node[0]+node[1]+node[2]+node[3]);

            HashMap<String, HashMap<String, Integer>> myArray = new
HashMap<String, HashMap<String, Integer>>();

      myArray.put("toilet-female-2", new HashMap<String, Integer>());
      myArray.get("toilet-female-2").put("positionx", 430);
      myArray.get("toilet-female-2").put("positiony", 150);

      myArray.put("toilet-male-2", new HashMap<String, Integer>());
      myArray.get("toilet-male-2").put("positionx", 430);
      myArray.get("toilet-male-2").put("positiony", 1500);

      myArray.put("toilet-female", new HashMap<String, Integer>());
      myArray.get("toilet-female").put("positionx", 800);
      myArray.get("toilet-female").put("positiony", 670);

      myArray.put("toilet-male", new HashMap<String, Integer>());
      myArray.get("toilet-male").put("positionx", 800);
      myArray.get("toilet-male").put("positiony", 900);

      myArray.put("stairs", new HashMap<String, Integer>());
      myArray.get("stairs").put("positionx", 800);
      myArray.get("stairs").put("positiony", 800);

      myArray.put("stairs-2", new HashMap<String, Integer>());
      myArray.get("stairs-2").put("positionx", 280);
      myArray.get("stairs-2").put("positiony", 200);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
myArray.put("stairs-3", new HashMap<String, Integer>());
myArray.get("stairs-3").put("positionx", 280);
myArray.get("stairs-3").put("positiony", 1460);

myArray.put("A", new HashMap<String, Integer>());
myArray.get("A").put("positionx", 360);
myArray.get("A").put("positiony", 150);

myArray.put("B", new HashMap<String, Integer>());
myArray.get("B").put("positionx", 360);
myArray.get("B").put("positiony", 200);

myArray.put("C", new HashMap<String, Integer>());
myArray.get("C").put("positionx", 360);
myArray.get("C").put("positiony", 260);

myArray.put("D", new HashMap<String, Integer>());
myArray.get("D").put("positionx", 360);
myArray.get("D").put("positiony", 360);

myArray.put("E", new HashMap<String, Integer>());
myArray.get("E").put("positionx", 360);
myArray.get("E").put("positiony", 465);

myArray.put("F", new HashMap<String, Integer>());
myArray.get("F").put("positionx", 360);
myArray.get("F").put("positiony", 580);

myArray.put("G", new HashMap<String, Integer>());
myArray.get("G").put("positionx", 360);
myArray.get("G").put("positiony", 670);

myArray.put("H", new HashMap<String, Integer>());
myArray.get("H").put("positionx", 360);
myArray.get("H").put("positiony", 800);

myArray.put("I", new HashMap<String, Integer>());
myArray.get("I").put("positionx", 360);
myArray.get("I").put("positiony", 900);

myArray.put("J", new HashMap<String, Integer>());
myArray.get("J").put("positionx", 360);
myArray.get("J").put("positiony", 995);

myArray.put("K", new HashMap<String, Integer>());
myArray.get("K").put("positionx", 360);
myArray.get("K").put("positiony", 1080);

myArray.put("L", new HashMap<String, Integer>());
myArray.get("L").put("positionx", 360);
myArray.get("L").put("positiony", 1170);

myArray.put("M", new HashMap<String, Integer>());
myArray.get("M").put("positionx", 360);
myArray.get("M").put("positiony", 1250);

myArray.put("N", new HashMap<String, Integer>());
myArray.get("N").put("positionx", 360);
myArray.get("N").put("positiony", 1340);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
myArray.put("O", new HashMap<String, Integer>());
myArray.get("O").put("positionx", 360);
myArray.get("O").put("positiony", 1425);

myArray.put("P", new HashMap<String, Integer>());
myArray.get("P").put("positionx", 360);
myArray.get("P").put("positiony", 1460);

myArray.put("Q", new HashMap<String, Integer>());
myArray.get("Q").put("positionx", 360);
myArray.get("Q").put("positiony", 1500);

myArray.put("E12-Lift", new HashMap<String, Integer>());
myArray.get("E12-Lift").put("positionx", 280);
myArray.get("E12-Lift").put("positiony", 800);

myArray.put("E12-1101", new HashMap<String, Integer>());
myArray.get("E12-1101").put("positionx", 430);
myArray.get("E12-1101").put("positiony", 260);

myArray.put("E12-1102", new HashMap<String, Integer>());
myArray.get("E12-1102").put("positionx", 430);
myArray.get("E12-1102").put("positiony", 360);

myArray.put("E12-1103", new HashMap<String, Integer>());
myArray.get("E12-1103").put("positionx", 430);
myArray.get("E12-1103").put("positiony", 465);

myArray.put("E12-1104", new HashMap<String, Integer>());
myArray.get("E12-1104").put("positionx", 430);
myArray.get("E12-1104").put("positiony", 580);

myArray.put("E12-1105", new HashMap<String, Integer>());
myArray.get("E12-1105").put("positionx", 430);
myArray.get("E12-1105").put("positiony", 995);

myArray.put("E12-1106", new HashMap<String, Integer>());
myArray.get("E12-1106").put("positionx", 430);
myArray.get("E12-1106").put("positiony", 1080);

myArray.put("E12-1107", new HashMap<String, Integer>());
myArray.get("E12-1107").put("positionx", 430);
myArray.get("E12-1107").put("positiony", 1170);

myArray.put("E12-1108", new HashMap<String, Integer>());
myArray.get("E12-1108").put("positionx", 430);
myArray.get("E12-1108").put("positiony", 1250);

myArray.put("E12-1109", new HashMap<String, Integer>());
myArray.get("E12-1109").put("positionx", 430);
myArray.get("E12-1109").put("positiony", 1340);

myArray.put("E12-1110", new HashMap<String, Integer>());
myArray.get("E12-1110").put("positionx", 430);
myArray.get("E12-1110").put("positiony", 1425);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    for (int i = 0; i<nodetotal; i++){
                          RelativeLayout path = (RelativeLayout)
findViewById(R.id.relativeLayout);
                          path.removeView(temp[i]);
                          Log.v("nodetotal", "nodetotal = "+nodetotal);
                    }

                    currentpos = 0;

                     String nodepath2 = "node";
                int nodecount = node.length;
                Log.v("testnode", "length = "+nodecount);
                for(int i = 0; i < nodecount; i++){
                Integer posx = null;
                      Integer posy = null;
                      String nodepath = node[i];

                      Log.v("testnode", "count = "+nodepath);
                    posx = myArray.get(node[i]).get("positionx");
                      posy = myArray.get(node[i]).get("positiony");
                      showpath(posx,posy);

                      if(nodepath2.equals("G")&&(nodepath.equals("toilet-
female"))){
                            posx = 360;
                            posy = 670;
                            int inc = 113;
                            while (posx < 800) {
                            posx = posx + inc;
                            showpath(posx,posy);}
                              }

                    else
if(nodepath2.equals("H")&&(nodepath.equals("stairs"))){
                            posx = 360;
                            posy = 800;
                            int inc = 113;
                            while (posx < 800) {

                                  posx = posx + inc;
                                  showpath(posx,posy);}
                              }

                    else
if(nodepath2.equals("I")&&(nodepath.equals("toilet-male"))){
                            posx = 360;
                            posy = 900;
                            int inc = 113;
                            while (posx < 800) {

                                  posx = posx + inc;
                                  showpath(posx,posy);}
                            }

                    else
if(nodepath2.equals("H")&&(nodepath.equals("toilet-female"))){
                            posy = 800;
                            int incx = 88;
                            int incy = 25;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        for(posx = 360; posx < 800; posx=posx+incx){
                            if ( posy > 670){
                                showpath(posx,posy);
                                posy=posy-incy;
                            }     }      }

                    else
if(nodepath2.equals("I")&&(nodepath.equals("stairs"))){
                        posy = 900;
                        int incx = 88;
                        int incy = 19;
                        for(posx = 360; posx < 800; posx=posx+incx){
                            if ( posy > 800){
                                showpath(posx,posy);
                                posy=posy-incy;
                            }     }      }

                    else
if(nodepath2.equals("G")&&(nodepath.equals("stairs"))){
                        posy = 670;
                        int incx = 88;
                        int incy = 23;
                        for(posx = 360; posx < 800; posx=posx+incx){
                            if ( posy < 800){
                                showpath(posx,posy);
                                posy=posy+incy;
                            }     }      }

                    else
if(nodepath2.equals("H")&&(nodepath.equals("toilet-male"))){
                        posy = 800;
                        int incx = 85;
                        int incy = 20;
                        for(posx = 360; posx < 800; posx=posx+incx){
                            if ( posy < 900){
                                showpath(posx,posy);
                                posy=posy+incy;
                            }     }      }

                    nodepath2 = nodepath;

                }
        }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

โค้ดของ Dijkstra's algorithm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<?php
$host = 'localhost';
$user = 'webonest_piamrak';
$pass = 'noey23083';
$con = mysql_connect ($host,$user,$pass);
mysql_select_db('webonest_piamrak');
$sourcee = $_GET['source'];
$destination = $_GET['destination'];
function dijkstra($graph_array, $source, $target) {
  $vertices = array();
  $neighbours = array();
  foreach ($graph_array as $edge) {
    array_push($vertices, $edge[0], $edge[1]);
    $neighbours[$edge[0]][] = array("end" => $edge[1], "cost" => $edge[2]);
    $neighbours[$edge[1]][] = array("end" => $edge[0], "cost" => $edge[2]);
  }
  $vertices = array_unique($vertices);
  foreach ($vertices as $vertex) {
    $dist[$vertex] = INF;
    $previous[$vertex] = NULL;
  }
  $dist[$source] = 0;
  $Q = $vertices;
  while (count($Q) > 0) {

    // TODO - Find faster way to get minimum
    $min = INF;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    foreach ($Q as $vertex){
      if ($dist[$vertex] < $min) {
        $min = $dist[$vertex];
        $u = $vertex;
      }
    }
    $Q = array_diff($Q, array($u));
    if ($dist[$u] == INF or $u == $target) {
      break;
    }
    if (isset($neighbours[$u])) {
      foreach ($neighbours[$u] as $arr) {
        $alt = $dist[$u] + $arr["cost"];
        if ($alt < $dist[$arr["end"]]) {
          $dist[$arr["end"]] = $alt;
          $previous[$arr["end"]] = $u;
        }       }
    }   }
$path = array();
$u = $target;
while (isset($previous[$u])) {
   array_unshift($path, $u);
   $u = $previous[$u];
}

array_unshift($path, $u);
return $path;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
$graph_array = array(
        array("toilet-female-2", "A", 8),
        array("A", "B", 4),
        array("B", "stairs-2", 8),
        array("B", "C", 6),
        array("C", "E12-1101", 8),
        array("C", "D", 9),
        array("D", "E12-1102", 8),
        array("D", "E", 9),
        array("E", "E12-1103", 8),
        array("E", "F", 9),
        array("F", "E12-1104", 8),
        array("F", "G", 9),
        array("G", "toilet-female", 17),
        array("G", "stairs", 20),
        array("G", "H", 11),
        array("toilet-female", "stairs", 11),
        array("toilet-female", "H", 20),
        array("stairs", "H", 17),
        array("stairs", "I", 20),
        array("H", "E12-Lift", 20),
        array("H", "I", 11),
        array("H", "toilet-male", 20),
        array("stairs", "toilet-male", 11),
        array("toilet-male", "I", 17),
        array("I", "J", 8),
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        array("J", "E12-1105", 8),
        array("J", "K", 8),
        array("K", "E12-1106", 8),
        array("K", "L", 8),
        array("L", "E12-1107", 8),
        array("L", "M", 8),
        array("M", "E12-1108", 8),
        array("M", "N", 8),
        array("N", "E12-1109", 8),
        array("N", "O", 8),
        array("O", "E12-1110", 8),
        array("O", "P", 8),
        array("P", "stairs-3", 8),
        array("P", "Q", 7),
        array("Q", "toilet-male-2", 8)
        );
$path = dijkstra($graph_array, "$sourcee", "$destination");
$pathh = implode(", ", $path);
$sql = "INSERT INTO `output_nfc` (`path`) VALUES ('{$pathh}');";
mysql_query($sql);
$A = array ("path" => $pathh);
echo json_encode($A);
 ?>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

Poster

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบนำทางในอาคารโดยใช้เทคโนโลยีเอ็นเอฟซีบนสมาร์ตโฟน

INDOOR NAVIGATION SYSTEM USING NFC ON SMARTPHONE

นักศึกษาหลักสูตรวิศวกรรมสารสนเทศ

นางสาวเปี่ยมรัก ภัทรมุกขานนท์

อาจารย์ที่ปรึกษา ผศ.บุณย์ชนะ ภู่ระหงษ์

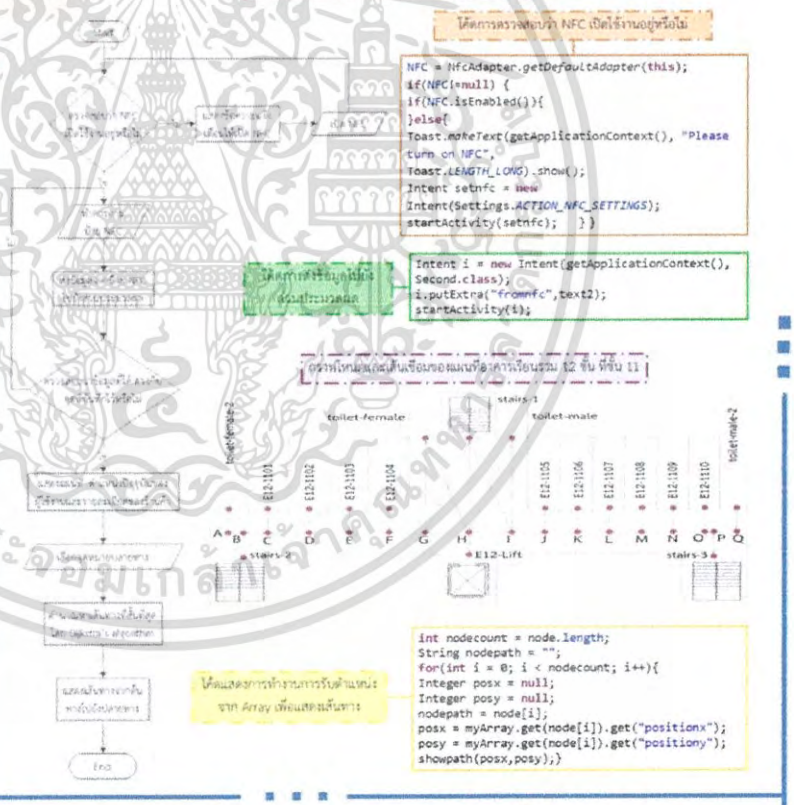
บทคัดย่อ

ในอนาคตเทคโนโลยีการสื่อสารไร้สายระยะสั้นที่เรียกว่า NFC (Near Field Communication) มีแนวโน้มที่จะใช้งานอย่างแพร่หลาย เนื่องจากสามารถเชื่อมต่อและแลกเปลี่ยนข้อมูลระหว่างอุปกรณ์ได้อย่างรวดเร็วและปลอดภัยเพียงแค่นำอุปกรณ์มาสัมผัสกัน ปัจจุบันมีสมาร์ตโฟน (Smart phone) ที่นำเทคโนโลยี NFC นี้มาใช้งานมากขึ้น ตัวอย่างการใช้งานเช่น การใช้โทรศัพท์มือถือเพื่อชำระค่าสินค้าและบริการ การชำระค่ารถโดยสาร การยืนยันตัวตน เป็นต้น

ในโครงงานนี้จึงนำเสนอระบบแผนที่ภายในอาคาร เช่น ศูนย์ประชุมหรือห้างสรรพสินค้า โดยใช้เทคโนโลยี NFC เพื่อใช้ในการระบุตำแหน่งปัจจุบันของผู้ใช้งาน และยังสะดวกในการบอกรายละเอียดของร้านค้า แอปพลิเคชั่นสามารถแสดงแผนที่ภายในอาคารและเส้นทางที่สั้นที่สุดตั้งแต่ต้นทางไปปลายทางซึ่งคำนวณด้วยเทคนิคกราฟ แอปพลิเคชั่นนี้ช่วยอำนวยความสะดวกให้ผู้ใช้งานในการนำทางสำหรับผู้ที่ไม่เคยไปสถานที่นั้นๆให้ถึงเป้าหมายได้อย่างถูกต้องและรวดเร็ว

ภาพรวมของระบบ

การออกแบบ



ผลการทดสอบ

สรุปผลการทดลอง

1. แอปพลิเคชั่นมีความสามารถในการแสดงแผนที่ภายในอาคาร ตำแหน่งปัจจุบันของผู้ใช้งาน และรายละเอียดของร้านค้าแต่ละร้านได้ ซึ่งใช้วิธีการสื่อสารไร้สายระยะสั้น หรือ NFC (Near Field Communication)
2. แอปพลิเคชั่นสามารถแสดงเส้นทางที่สั้นที่สุดจากต้นทางไปยังปลายทางให้ผู้ใช้งาน ซึ่งคำนวณด้วย Dijkstra's algorithm
3. ระบบมีขนาดเล็ก ราคาถูก ประหยัดพลังงาน มีค่าบำรุงรักษาต่ำ ติดตั้งง่าย และผู้ใช้งานสามารถใช้งานได้ง่าย อีกทั้งยังช่วยอำนวยความสะดวกให้ผู้ใช้งานในการนำทางสำหรับผู้ที่ไม่เคยไปสถานที่นั้นๆให้ถึงเป้าหมายได้อย่างถูกต้องและรวดเร็ว

อุปกรณ์ที่ใช้ในการทดลอง

Information Engineering Project Day 2015

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้