การพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถ โดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์

THE IOS APPLICATION DEVELOPMENT OF BUS ROUTING MANAGEMENT USING APPLE SWIFT BASED DYNAMIC A* ALGORITHM

กมลวรรณ พงษ์ศิลป์

KAMONWAN PONGSIN

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
มหาวิทยาลัยศรีปทุม
พ.ศ. 2561

ลิขสิทธิ์ของมหาวิทยาลัยศรีปทุม

การพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถ โดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์

กมลวรรณ พงษ์ศิลป์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
มหาวิทยาลัยศรีปทุม
พ.ศ. 2561

ลิขสิทธิ์ของมหาวิทยาลัยศรีปทุม

THE IOS APPLICATION DEVELOPMENT OF BUS ROUTING MANAGEMENT USING APPLE SWIFT BASED DYNAMIC A* ALGORITHM

KAMONWAN PONGSIN

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN INFORMATION TECHNOLOGY

SCHOOL OF INFORMATION TECHNOLOGY

SRIPATUM UNIVERSITY

2018

COPYRIGHT OF SRIPATUM UNIVERSITY

หัวข้อวิทยานิพนธ์	การพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ THE IOS APPLICATION DEVELOPMENT OF BUS ROUTING MANAGEMENT USING APPLE SWIFT BASED DYNAMIC A* ALGORITHM
นักศึกษา	กมลวรรณ พงษ์ศิลป์ รหัสประจาตัว 60500272
หลักสูตร	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะ	เทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ดร.สุขสวัสดี ณัฏฐวุฒิสิทธิ์

คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลกัสูตรปริญญามหาบัณฑิต

..... คณบดี คณะเทคโนโลยีสารสนเทศ
(ผู้ช่วยศาสตราจารย์ ดร.ธนา สุขวารี)

วันที่……. เดือน……..……. พ.ศ. …….

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ผู้ช่วยศาสตราจาย์ ดร.นิเวศ จิระวิชิตชัย)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.เทพฤทธิ์ บัณฑิตวัฒนาวงศ์)

..... กรรมการ
(ดร.สุขสวัสดี ณัฏฐวุฒิสิทธิ์

วิทยานิพนธ์เรื่อง การพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์

คำสำคัญ การคัดเลือก รถประจำทาง เส้นทาง อัลกอริทึม เอ สตาร์

นักศึกษา กมลวรรณ พงษ์ศิลป์

อาจารย์ที่ปรึกษาวิทยานิพนธ์ ดร.สุขสวัสดี ณัฏฐวุฒิสิทธิ์

หลักสูตร วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะ เทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม

พ.ศ. 2561

บทคัดย่อ

งานวิจัยนี้เป็นงานวิจัยเชิงทดลอง มีวัตถุประสงค์เพื่อศึกษาปัญหาและแนวทางการพัฒนากลไกการเลือกเส้นทางโดยสารรถประจำทาง โดยผู้วิจัยได้นำแนวคิดการคัดเลือกเส้นทางด้วยอัลกอริทึมพลวัต เอสตาร์มาใช้เพื่อค้นหาและคัดเลือกเส้นทางที่เหมาะสมแก่ผู้ใช้งาน ทำให้ช่วยคำนวณเวลา และวางแผนการเดินทางได้อย่างมีประสิทธิภาพ เครื่องมือที่ใช้ในงานวิจัย ได้แก่ แบบจำลองสารสนเทศเพื่อแนะนำเส้นทางโดยสารรถประจำทางที่ถูกพัฒนาขึ้นด้วยภาษาสวิฟท์บนพื้นฐานอัลกอริทึมพลวัต เอ สตาร์ กลุ่มตัวอย่าง ได้แก่ ข้อมูลการโดยสารรถประจำทางองค์การขนส่งมวลชนกรุงเทพ (ขสมก.) เส้นทางสาย 207, 51ร, 522 ประกอบด้วย (1) ค่าความเสี่ยงของเส้นทาง (2) ค่าความเหมาะสมของจุดขึ้น-ลงรถโดยสารประจำทาง และ (3) ค่านัยสำคัญของเส้นทางการโดยสารรถประจำทาง สำหรับใช้คำนวณหาเส้นทางที่เหมาะสม ผลการวิจัย พบว่า ผลลัพธ์จากการประเมินโดยผู้เชี่ยวชาญมีค่าเฉลี่ยในระดับดี (ค่าเฉลี่ย ($\bar{X}$) = 4.30 และค่าเบี่ยงเบนมาตรฐาน S.D. = 0.273) ประโยชน์ที่ได้รับจากงานวิจัยนี้สามารถนำไปประยุกต์ใช้กับการคัดเลือกเส้นทางโดยสารรถประจำทางอื่น ๆ รวมทั้งองค์ความรู้นี้ยังสามารถนำไปขยายผลทางวิชาการได้ต่อไปในอนาคต

THESIS TITLE	THE DEVELOPMENT IOS APPLICATION FOR BUS ROUTE MANAGEMENT, DYNAMIC MODELING OF THE ASTAR ALGORITHM WITH SWIFT ON CLOUD
KEYWORDS	SELECTION, BUS, ROUTE, ALGORITHM, A* STAR
STUDENT	KAMONWAN PONSING
THESIS ADVISOR	Dr. SOOKSAWATDEE NATTAWUTTISIT
LEVEL OF STUDY	MASTER OF SCIENCE IN INFORMATION TECHNOLOGY
INSTITUTE	SCHOOL OF INFORMATION TECHNOLOGY SRIPATUM UNIVERSITY
YEAR	2018

ABSTRACT

This research is experimental research. The purpose of this study was to investigate the problems and ways to develop a bus route selection mechanism. The researcher has adopted the concept of route selection with dynamic algorithms. Aster Star is used to find and select the appropriate route for the user. Help to calculate time And plan to travel efficiently. The research tool was a traffic information model developed by Swift, based on the Astartes dynamic algorithm. The samples were bus traffic data, mass transit (Routes 207, 51๖, 522 consist of: (1) Route risk; (2) Approach to take-off bus; and (3) Significance of route. Routes The results showed that the results from the expert evaluation were good (mean ($\bar{X}$) = 4.30 and standard deviation S.D. = 0.273). This research can be applied to the selection of other bus routes as well as this knowledge can be extended to further academic results in the future.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ สำเร็จลุล่วงไปด้วยดี ด้วยความกรุณาจากดร.สุขสวัสดี ณัฏฐวุฒิสิทธิ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ได้เสียสละเวลาอันมีค่าให้คำปรึกษาและให้คำแนะนำ แก้ไขปัญหา รวมถึงข้อบกพร่องต่าง ๆ ที่เกิดขึ้นอย่างดีที่เป็นประโยชน์เพื่อให้ งานวิจัยฉบับนี้สมบูรณ์และถูกต้องมากยิ่งขึ้น

ขอขอบพระคุณคณาจารย์ทุกท่านที่ประสิทธิ์ประสาทวิชาความรู้จนสามารถสำเร็จการศึกษาในระดับมหาบัณฑิต และขอขอบคุณผู้ที่เกี่ยวข้องกับงานวิจัยนี้ที่ให้ความช่วยเหลือในครั้งนี้เป็นอย่างดี

ขอขอบคุณแหล่งข้อมูลต่าง ๆ ที่เป็นประโยชน์โดยเปิดโอกาสให้ผู้วิจัยสามารถเข้าถึงข้อมูลต่าง ๆ ที่เกี่ยวข้องเป็นอย่างดี

สุดท้ายขอขอบพระคุณ คุณพ่อ คุณแม่ ที่สนับสนุนเป็นกำลังใจในการทำวิทยานิพนธ์ จนสำเร็จลุล่วงไปด้วยดี

กมลวรรณ พงษ์ศิลป์

สารบัญ

หน้า

บทคัดย่อภาษาไทย…….. I

บทคัดย่อภาษาอังกฤษ…….. II

กิตติกรรมประกาศ…… III

สารบัญ…… IV

สารบัญตาราง…….. VI

สารบัญรูป……... VII

บทที่

1 บทนำ……...…….. 1

1.1 ความเป็นมาและความสำคัญของปัญหา……..……. 1

1.2 คำถามของการวิจัย……..…..…….. 2

1.3 สมมติฐานการวิจัย……..…….. 2

1.4 วัตถุประสงค์การวิจัย……..…….. 2

1.5 กรอบแนวคิดในการวิจัย…….…… 2

1.6 ขอบเขตของการวิจัย……..……... 3

1.7 ประโยชน์ที่คาดว่าจะได้รับ...…….…… 3

1.8 นิยามศัพท์...……..……... 3

2 แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง……..……..... 5

2.1 วงจรการพัฒนาซอฟต์แวร์ (System development Life Cycle : SDLC)…...…..….. 5

2.2 โมบายแอพพลิเคชั่น (Mobile Application)…….. 9

2.3 ระบบปฏิบัติการไอโอเอส (iOS) / สถาปัตยกรรมระบบไอโอเอส (iOS)……. 14

2.4 ภาษาสวิฟท์ (Swift)……..... 16

2.5 โปรแกรมเอกซ์โค้ด (Xcode)…… 22

2.6 จีพีเอส (GPS)……... 24

2.7 Google Maps SDK……..... 26

2.8 อัลกอริทึม (Algorithm)…….....…….. 28

2.9 สัญกรณ์บิ๊กโอ (Big O Notation)……... 35

สารบัญ (ต่อ)

หน้า

2.10 คลาวด์คอมพิวติ้ง (Cloud Computing)…….. 39
2.11 งานวิจัยที่เกี่ยวข้อง……..…….. 41

3 วิธีดำเนินการวิจัย..... 46
3.1 ศึกษาปัญหาและแนวทางการพัฒนาแอปพลิเคชัน..... 46
3.2 การวิเคราะห์ระบบ..... 47
3.3 การออกแบบระบบ..... 49
3.5 การพัฒนาระบบ..... 55
3.6 การทดสอบระบบ..... 55
3.7 การประเมินผลการใช้งาน..... 57
3.8 แผนการดำเนินงานวิจัย..... 57

4 ผลงานวิจัย..... 58
4.1 ผลการวิจัยตามวัตถุประสงค์ข้อที่ 1..... 58
4.2 ผลการวิจัยตามวัตถุประสงค์ข้อที่ 2..... 60
4.3 ผลการทดลองระบบ..... 63

5 สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ..... 65
5.1 สรุปผลการวิจัย..... 65
5.2 อภิปรายผลการวิจัย..... 65
5.3 ข้อเสนอแนะงานวิจัย..... 66
5.4 ปัญหา และอุปสรรค..... 66

บรรณานุกรม..... 67
ภาคผนวก ก คู่มือการใช้งาน..... 71
ภาคผนวก ข ใบตอบรับการตีพิมพ์วารสารซายน์เทค มรภ.ภูเก็ต..... 75
ภาคผนวก ค ใบตอบรับการตีพิมพ์วารสาร..... 86
ประวัติผู้วิจัย..... 97

สารบัญตาราง

ตารางที่ **หน้า**

2.1 สรุปวงจรการพัฒนาระบบ SDLC..... 8

2.2 รายละเอียดของแอพพลิเคชั่น ในกลุ่มต่าง ๆ (Application Categories)..... 12

3.1 แผนการดำเนินงาน..... 57

4.1 สรุปการวิเคราะห์และออกแบบระบบโดยการเ
ปรียบเทียบกับงานวิจัยที่เกี่ยงข้อง. 59

4.2 ตารางเวลาการเดินรถประจำทาง..... 61

4.3 ค่าตัวแปรที่ถูกกำหนดขึ้นเพื่อใช้ทดสอบเส้นทางด้วยสถานการณ์จำลอง..... 62

4.4 แสดงการค้นหาครั้งที่ 1 ในสถานการณ์ปกติ..... 63

4.5 แสดงการค้นหาครั้งที่ 2 ในสถานการณ์ปกติ..... 64

สารบัญรูป

ภาพประกอบที่ **หน้า**

1.1 กรอบทฤษฎีหรือกรอบแนวความคิด..... 2
2.1 วงจรชีวิตของการพัฒนาซอฟต์แวร์ SDLC……...….. 7
2.2 Native Application Interaction with Mobile Device……... 13
2.3 แบบฟอร์มการพัฒนาโปรแกรมประยุกต์บนมือถือ (MADPS)..... 14
2.4 สถาปัตยกรรมระบบไอโอเอส (iOS)..... 15
2.5 ความนิยมของภาษาสวิฟท์..... 16
2.6 ภาษาสวิฟท์ (Swift)…… 18
2.7 โปรแกรมเอกซ์โค้ด (Xcode)……..... 23
2.8 โครงสร้างของ Xcode…….. 23
2.9 รูปแบบการทำงานของระบบ GPS…...……... 24
2.10 การทำงานของ GPS..... 26
2.11 แสดง Pseudo Code อัลกอริทึม แบบ เอ สตาร์..... 29
2.12 กำหนดจุดเริ่มต้น..... 30
2.13 เส้นทางที่ออกจากจุดเริ่มต้นไปจุดถัดไป..... 30
2.14 เส้นทางที่ออกจากตำแหน่งปัจจุบัน (จุด 1) ไปจุดถัดไป..... 30
2.15 เส้นทางที่ออกจากตำแหน่งปัจจุบัน (จุด 3) ไปจุดถัดไป..... 31
2.16 เส้นทางที่ออกจากตำแหน่งปัจจุบัน(จุด 3 และ 4 ตามลำดับ) ไปจุดถัดไป..... 31
2.17 Map ที่ใช้ในการศึกษา A* อัลกอริทึม……. 32
2.18 ตัวอย่างการหาเส้นทางโดยใช้ A* อัลกอริทึม ขั้นที่ 1…… 33
2.19 ตัวอย่างการหาเส้นทางโดยใช้ A* อัลกอริทึม ขั้นที่ 2…….. 34
2.20 ตัวอย่างการหาเส้นทางโดยใช้ A* อัลกอริทึม ขั้นที่ 3……. 35
2.21 ตัวอย่างของสัญกรณ์บิ๊ก O: f 36
2.22 กราฟของฟังก์ชันที่ใช้กันโดยทั่วไปในการวิเคราะห์อัลกอริทึม..... 38
2.23 ระบบการประมวลผลแบบกลุ่มเมฆ…….. 39
2.24 คลาวด์คอมพิวติ้ง (Cloud Service Models)…… 41
3.1 ขั้นตอนการดำเนินการวิจัย..... 47
3.2 แนวคิดโดยรวมของระบบ..... 48

สารบัญรูป (ต่อ)

ภาพประกอบที่ **หน้า**

3.3 ออกแบบระบบ..... 49
3.4 ขั้นตอนการทำงานของอุปกรณ์ฮาร์ดแวร์..... 50
3.5 ตัวอย่างการเตรียมข้อมูลที่ถูกนำไปใช้งาน..... 51
3.6 ขั้นตอนการทำงานของระบบ..... 52
3.7 แสดง Use Case Diagram ของระบบ..... 53
3.8 แสดงผังลำดับการทำงานในส่วน เครื่องแม่ข่าย (Server)..... 54
3.9 แสดงผังลำดับการทำงานในส่วนเครื่องลูกข่าย (client)..... 54
3.10 แสดงส่วนของผู้ใช้ระบบ..... 55
3.11 การทดสอบระบบด้วยการจำลองการทำงานของอัลกอริทึมของเอสตาร์..... 56
4.1 ตัวอย่างเส้นทางรถโดยสารขสมกสาย 207, 51ร, 522……..... 60
4.2 ตัวอย่างแบบจำลองเส้นทางรูปแบบกราฟที่ได้จากตารางเส้นทางการการเดินรถประจำทาง……..... 62
4.3 ผลการทดสอบด้วยเส้นทางบนภาษา Swift..... 64

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การเดินทางด้วยรถโดยสารประจำทางมีความสำคัญอย่างยิ่งสำหรับประชาชนที่เดินทางไปยังสถานที่ต่าง ๆ ตัวอย่างเช่น ที่ทำงาน สถานศึกษา หรือเดินทางเพื่อติดต่อธุรกิจ ดังนั้นการคมนาคมนับได้ว่ามีบทบาทสำคัญต่อการดำเนินชีวิตของมนุษย์ยุคนี้ และเนื่องจากปัจจุบันจำนวนประชากรมีเพิ่มขึ้น จึงทำให้การใช้เส้นทางเดินทางเกิดปัญหาการจราจรติดขัด โดยทั่วไปสาเหตุของปัญหาเกิดจากการใช้งานยานพาหนะส่วนบุคคลกันมากกว่ารถโดยสารประจำทาง เมื่อมีการเปรียบเทียบจากอัตราการใช้พื้นที่ถนนต่อคน พบว่า การใช้งานรถยนต์ส่วนบุคคลจะทำให้เปลืองพื้นที่ถนนมากกว่าการใช้รถโดยสารประจำทางมาก จึงทำให้เกิดความล่าช้าในการเดินทางขึ้น และยังเป็นการสิ้นเปลืองพลังงาน รวมทั้งส่งผลกระทบต่อต้นทุนค่าใช้จ่ายในการดำเนินชีวิตและธุรกิจจากสาเหตุปัญหาการจราจรติดขัดเหล่านี้

ปัจจุบันเทคโนโลยีสารสนเทศและการสื่อสารมีการพัฒนาอย่างต่อเนื่อง จึงทำให้ประสิทธิภาพการทำงานของเครื่องโทรศัพท์มีความเร็ว และมีความจุของการจัดเก็บข้อมูลเทียบเท่ากับเครื่องคอมพิวเตอร์ นอกจากนี้ยังสามารถนำระบบปฏิบัติการใส่โปรแกรมลงในเครื่องโทรศัพท์ได้ จึงทำให้เกิดการพัฒนาสมาร์ทโฟนและแอปพลิเคชันบนสมาร์ทโฟนขึ้นเพื่อช่วยอำนวยความสะดวกให้แก่ผู้ใช้งาน และจากปัญหาดังกล่าวข้างต้น ผู้วิจัยจึงมีแนวคิดการพัฒนาระบบโมบายแอปพลิเคชันด้วยภาษาสวิฟท์ (Swift) บนระบบไอโอเอส (iOS) บนคลาวด์ขึ้น เพื่อมุ่งเน้นการแก้ไขปัญหาการจราจรติดขัด โดยนำขั้นตอนวิธี หรือ อัลกอริทึมการจัดการเส้นทางที่สั้นที่สุดรูปแบบ เอ สตาร์ (A Star หรือ A*) มาประยุกต์ใช้กับงานวิจัย เพื่อให้ผู้ใช้งานสามารถวางแผนและบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางผ่านทางแอปพลิเคชันด้วยสมาร์ทโฟน

ประโยชน์ที่จะได้รับจากงานวิจัยนี้สามารถช่วยยกระดับงานบริการที่ดีแก่ผู้ให้บริการและผู้ใช้บริการสามารถวางแผนการเดินทางได้อย่างมีประสิทธิภาพ และนอกจากนี้ยังสามารถนำไปขยายผลองค์ความรู้เพื่อพัฒนาระบบการขนส่งมวลชนกรุงเทพให้มีประสิทธิภาพมากยิ่งขึ้น

1.2 คำถามในการวิจัย

1.2.1 ไอ โอ เอสแอปพลิเคชันสามารถนำมาประยุกต์ใช้ในการติดตามรถประจำทางได้หรือไม่

1.2.2 ไอ โอ เอสแอปพลิเคชันสามารถแก้ไขปัญหาการจราจรติดขัดได้หรือไม่

1.3 สมมติฐานการวิจัย

1.3.1 ไอ โอ เอสแอปพลิเคชันสามารถนำมาประยุกต์ใช้ในการติดตามรถประจำทางได้

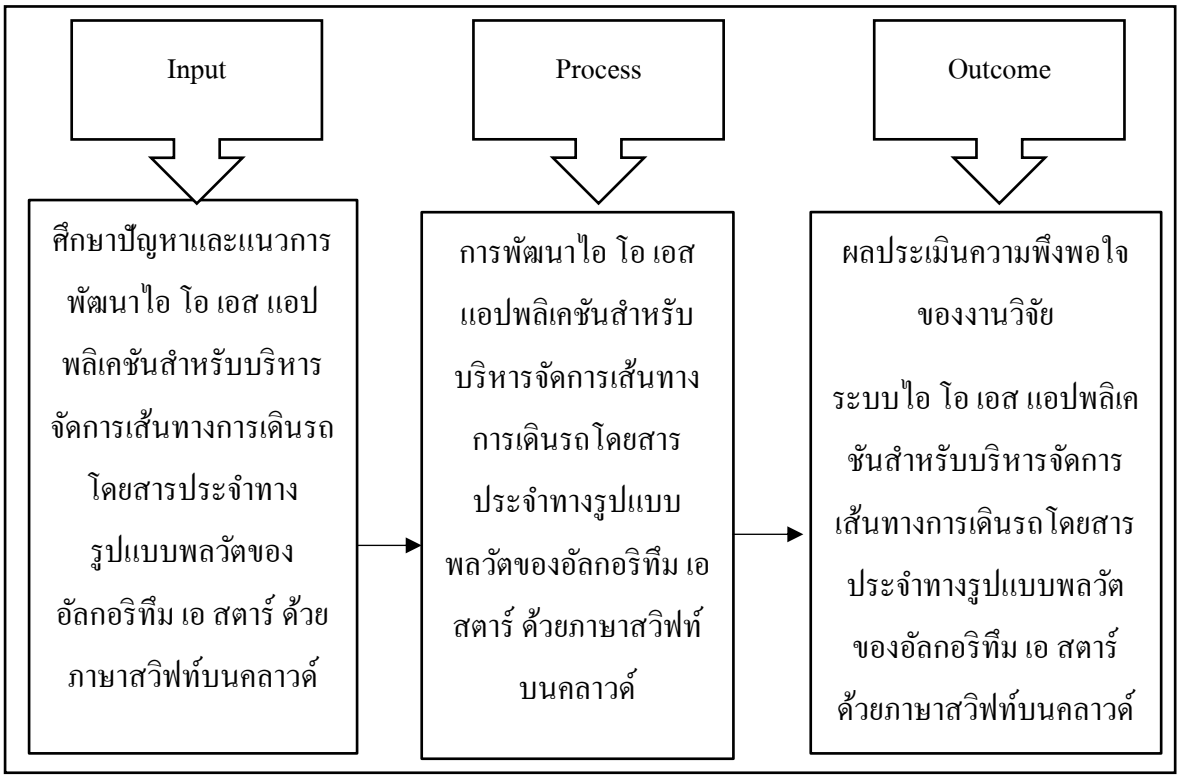
1.3.2 ไอ โอ เอสแอปพลิเคชันสามารถแก้ไขปัญหาการจราจรติดขัดได้

1.4 วัตถุประสงค์การวิจัย

1.4.1 เพื่อศึกษาปัญหาและแนวการพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์

1.4.2 เพื่อการพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์

1.5 กรอบแนวความคิดงานวิจัย



ภาพประกอบที่ 1.1 กรอบแนวความคิดงานวิจัย

1.6 ขอบเขตของการวิจัย

1.6.1 โมบายแอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ของ iOS

1.6.2 การทำงานของแอปพลิเคชันจะใช้เทคนิคขั้นตอนวิธีการเลือกเส้นทางที่สั้นและเหมาะสมที่สุดรูปแบบเอ สตาร์

1.6.3 ขอบเขตด้านเส้นทางการเดินรถโดยสารประจำทางและเวลาการการเดินรถโดยสารประจำทาง จะใช้กรณีศึกษาจากตารางการเดินรถโดยสารประจำทาง (ขสมก.) เขตพหลโยธินกรุงเทพมหานคร

1.6.4 ระยะเวลาในการดำเนินงานวิจัย 12 เดือน ตั้งแต่เดือน สิงหาคม พ.ศ. 2560 – สิงหาคม พ.ศ. 2561

1.7 ประโยชน์ที่คาดว่าจะได้รับ

1.7.1 แอปพลิเคชันเพื่อบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางแต่ละสายด้วยระบบ iOS สามารถช่วยในการแก้ไขปัญหาทางเศรษฐกิจและสังคมด้านการคมนาคมขนส่ง โดยลดปัญหารการจราจรติดขัดด้วยการวางแผนและบริหารจัดการเส้นทางการเดินทางที่ดี

1.7.2 ช่วยลดค่าใช้จ่ายและต้นทุนการดำเนินธุรกิจ โดยผู้ให้บริการและผู้ใช้บริการรถโดยสารประจำทาง สามารถทราบถึงเวลาของการมาของรถโดยสารประจำทางแต่ละคันได้อย่างถูกต้อง และสามารถบริหารจัดการเวลาของตนเองในการใช้บริการของรถโดยสารประจำทางได้อย่างมีประสิทธิภาพต่อไป

1.7.3 เกิดองค์ความรู้ใหม่ด้านการพัฒนาประสิทธิภาพของแอปพลิเคชันเพื่อบริหารจัดการเส้นทางการเดินรถโดยสารประจำทาง โดยมีการเผยแพร่ผลงานทางวิชาการและการนำไปทดสอบใช้งานเพื่อขยายผลองค์ความรู้แก่ผู้สนใจในอนาคตได้อย่างเหมาะสม

1.8 นิยามศัพท์

1.8.1 สวิฟท์ (Swift) เป็นภาษาคอมพิวเตอร์สำหรับพัฒนาแอปพลิเคชันของแมคโอเอส (MacOS) ไอโอเอส (iOS) วอตช์โอเอส (WatchOS) และ ทีวีโอเอส (tvOS) ที่ทั้งทรงพลังและใช้งานง่าย จนทำให้การเขียนสวิฟท์โค้ด(Swift code) นั้นสนุกกว่าการเขียนโปรแกรมภาษาอื่น ๆ รูปแบบการเขียนของ สวิฟท์ (Swift) นั้นไม่ยาก ทำให้โปรแกรมดูกระชับและเข้าใจง่ายเหมือนการเขียนประโยคทั่วไปในชีวิตประจำวัน

1.8.2 ระบบไอโอเอส (iOS) ระบบปฏิบัติการไอโอเอส (iOS) มีชื่อเดิมว่าไอโฟนโอเอส (iPhone OS) เริ่มต้นด้วยการเปิดตัวของไอโฟน(iPhone) เมื่อวันที่ 29 มิถุนายน 2550

ระบบปฏิบัติการไอโอเอส (iOS) เป็นระบบปฏิบัติการสำหรับสมาร์ทโฟน (Smartphone) ของแอปเปิล (Apple) โดยเริ่มต้นพัฒนาสำหรับใช้ในโทรศัพท์ไอโฟน (iPhone) และได้พัฒนาต่อใช้สำหรับไอพอด (iPotd) และไอแพด (iPad)

1.8.3 เอ สตาร์ อัลกอริทึม (A Star หรือ A* Algorithm) คือ ขั้นตอนวิธีที่ใช้ในการหาเส้นทางและการท่องในกราฟที่สั้นและเหมาะสมที่สุด ซึ่งเป็นกระบวนการในการหาเส้นทางระหว่างจุด (เรียกจุดดังกล่าวว่า "โนด" (node)) ดังนั้นขั้นตอนวิธีนี้จึงมีประสิทธิภาพและความแม่นยำสูง จึงมีการนำไปใช้งานอย่างแพร่หลาย

1.8.4 บิ๊ก โอ (Big O notation) คือ ระยะเวลาที่แย่ที่สุด หรือพื้นที่ในหน่วยประมวลผลที่มากที่สุด ที่คอมพิวเตอร์ต้องจ่ายให้กับการ run อัลกอริทึม หรือเรียกว่า ความซับซ้อนของอัลกอริทึม (Algorithm Complexity)

บทที่ 2

แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง

บทนี้กล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้องต่าง ๆ ที่ใช้ในการจัดทำงานวิจัย โดยเป็นการศึกษาข้อมูลเกี่ยวกับ อุปกรณ์เพื่อนำมาพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ ประกอบไปด้วย สมาร์ทโฟน ระบบปฏิบัติการไอโอเอส (iOS) การทำงานของสวิฟท์ (Swift) ทั้งหมดนี้เป็นการศึกษาเพื่อให้เกิดความรู้ความเข้าใจและนำมาประยุกต์ใช้งาน

2.1 วงจรการพัฒนาซอฟต์แวร์ (System development Life Cycle : SDLC)

2.2 โมบายแอพพลิเคชั่น (Mobile Application)

2.2.1 เนทีฟ แอปพลิเคชัน (Native Application)

2.3 ระบบปฏิบัติการไอโอเอส (iOS) / สถาปัตยกรรมระบบไอโอเอส (iOS)

2.4 ภาษาสวิฟท์ (Swift)

2.5 โปรแกรมเอกซ์โค้ด (Xcode)

2.6 จีพีเอส (GPS)

2.7 Google Maps SDK

2.7.1 เอพีไอ (Application programming interface : API)

2.8 อัลกอริทึม (Algorithm)

2.9 สัญกรณ์บิ๊กโอ (Big O notation)

2.10 คลาวด์คอมพิวติ้ง (Cloud Computing)

2.11 งานวิจัยที่เกี่ยวข้อง

2.1 วงจรชีวิตของการพัฒนาซอฟต์แวร์ SDLC

วงจรชีวิตของการพัฒนาซอฟต์แวร์ หรือ Software Development Life Cycle (SDLC) เพื่อให้การวางแผนพัฒนาระบบสารสนเทศดำเนินอย่างมีประสิทธิภาพ นักวิเคราะห์ระบบควรทำความเข้าใจรายละเอียดของวงจรการพัฒนาระบบสารสนเทศ ซึ่งบางตำราอาจจะกล่าวถึงเนื้อหารายละเอียดของขั้นตอนวงจรการพัฒนาระบบแตกต่างกันไปบ้าง แต่อย่างไรก็ดีตำราส่วนใหญ่จะเริ่มต้นจากจุดเดียวกันคือ ปัญหา (Problem) และความต้องการของผู้ใช้งาน (Requirement) และ

สิ้นสุดในขั้นตอนการบำรุงรักษา (Maintenance) โดยทั่วไปวงจรการพัฒนาระบบสารสนเทศจะแบ่งออกเป็น 7 ขั้นตอน

1. การระบุปัญหา (Problem Identification) นักวิเคราะห์ระบบจะต้องศึกษาเพื่อค้นหาปัญหา ข้อเท็จจริงที่แท้จริง ซึ่งหากปัญหาที่ค้นพบ มิใช่ปัญหาที่แท้จริง ระบบงานที่พัฒนาขึ้นมาก็จะตอบสนองการใช้งานไม่ครบถ้วน ปัญหาหนึ่งของระบบงานที่ใช้ในปัจจุบันคือ โปรแกรมที่ใช้งานในระบบงานเดิมเหล่านั้นถูกนำมาใช้งานในระยะเวลาที่นานอาจเป็นโปรแกรมที่เขียนขึ้นมาเพื่อติดตามผลงานใดงานหนึ่งโดยเฉพาะเท่านั้น ไม่ได้เชื่อมโยงถึงกันเป็นระบบ ดังนั้นนักวิเคราะห์ระบบจึงต้องมองเห็นปัญหาที่เกิดขึ้นในทุกหน่วยงานที่เกี่ยวข้องกับระบบงานที่จะพัฒนา แล้วดำเนินการแก้ไขปัญหา

2. ศึกษาความเป็นไปได้ (Feasibility Study) การวิเคราะห์ จะต้องรวบรวมข้อมูลความต้องการ (Requirements) ต่าง ๆ มาให้มากที่สุด ซึ่งการสืบค้นความต้องการของผู้ใช้สามารถดำเนินการได้จากการรวบรวมเอกสารการสัมภาษณ์ การออกแบบสอบถาม และการสังเกตการณ์บนสภาพแวดล้อมการทำงานจริง เมื่อได้นำความต้องการ มาผ่านการวิเคราะห์เพื่อสรุปเป็นข้อกำหนดที่ชัดเจนแล้ว ขั้นตอนต่อไปของนักวิเคราะห์ระบบก็คือ การนำข้อกำหนดเหล่านั้นไปพัฒนาเป็นความต้องการของระบบใหม่ด้วยการพัฒนาเป็นแบบจำลองขึ้นมา ซึ่งได้แก่ แบบจำลองกระบวนการ (Data Flow Diagram) และแบบจำลองข้อมูล (Data Model) เป็นต้น

3. การวิเคราะห์ระบบ (System Analysis) เป็นระยะที่นำผลลัพธ์ที่ได้จากการวิเคราะห์ที่เป็นแบบจำลองเชิงตรรกะมาพัฒนาเป็นแบบจำลองเชิงกายภาพ โดยแบบจำลองเชิงตรรกะที่ได้จากขั้นตอนการวิเคราะห์ มุ่งเน้นว่ามีอะไรที่ต้องทำในระบบในขณะที่แบบจำลองเชิงกายภาพจะนำแบบจำลองเชิงตรรกะมาพัฒนา ต่อด้วยการมุ่งเน้นว่าระบบดำเนินการอย่างไรเพื่อให้เกิดผลตามต้องการ งานออกแบบระบบประกอบด้วยงานออกแบบสถาปัตยกรรมระบบที่เกี่ยวข้องกับฮาร์ดแวร์ ซอฟต์แวร์ และระบบ เครือข่าย การออกแบบรายงาน การออกแบบหน้าจออินพุตข้อมูล การออกแบบผังงานระบบ การออกแบบฐานข้อมูล และการออกแบบโปรแกรม เป็นต้น

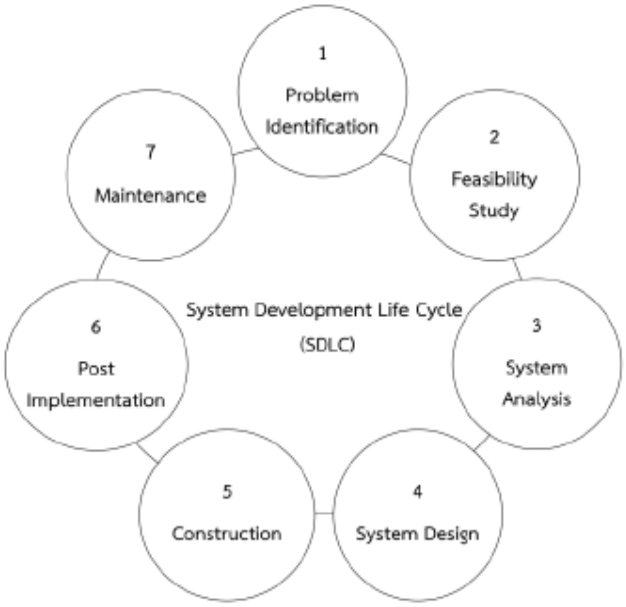
4. การออกแบบระบบ (System Analysis) เป็นระยะที่เกี่ยวข้องกับการพัฒนาโปรแกรม โดยทีมงานโปรแกรมเมอร์จะต้องพัฒนาโปรแกรมตามที่นักวิเคราะห์ระบบได้ออกแบบไว้ การเขียนชุดคำสั่งเพื่อสร้างเป็นระบบงานทางคอมพิวเตอร์ขึ้นมา โดยโปรแกรมเมอร์สามารถนำเครื่องมือเข้ามาช่วยในการพัฒนาโปรแกรมได้เพื่อช่วยให้ระบบงานพัฒนาได้เร็วขึ้นและมีคุณภาพ

5. การพัฒนาและทดสอบระบบเพื่อนำไปติดตั้งใช้งานจริงในองค์กร (Construction) เมื่อโปรแกรมได้พัฒนาขึ้นมาแล้ว ยังไม่สามารถนำระบบไปใช้งานได้ทันทีจำเป็นต้องดำเนินการทดสอบระบบก่อนที่จะนำไปใช้งานจริงเสมอ ควรมีการทดสอบข้อมูลเบื้องต้นก่อน ด้วยการสร้าง

ข้อมูลจำลองขึ้นมาเพื่อใช้ตรวจสอบการทำงานของระบบงาน หากพบข้อผิดพลาดก็ปรับปรุงแก้ไขให้ถูกต้อง การทดสอบระบบจะมีการตรวจสอบไวยากรณ์ของภาษาเขียน และตรวจสอบว่าระบบตรงกับความต้องการของผู้ใช้หรือไม่

6. ขั้นตอนหลังการติดตั้งระบบ (Post Implementation) เมื่อดำเนินการทดสอบระบบจนมั่นใจว่าระบบที่ได้รับการทดสอบนั้นพร้อมที่จะนำไปติดตั้งเพื่อใช้งานบนสถานการณ์จริง ขั้นตอนการนำระบบไปใช้งานอาจเกิดปัญหา จากการที่ระบบที่พัฒนาใหม่ไม่สามารถนำไปใช้งานแทนระบบงานเดิมได้ทันที จึงมีความจำเป็นต้องแปลงข้อมูลระบบเดิมให้อยู่ในรูปแบบที่ระบบใหม่สามารถนำไปใช้งานได้เสียก่อน หรืออาจพบข้อผิดพลาดที่ไม่คาดคิดเมื่อนำไปใช้ในสถานการณ์จริง ครั้นเมื่อระบบสามารถรันได้จนเป็นที่น่าพอใจทั้งสองฝ่าย ก็จะต้องจัดทำเอกสารคู่มือระบบรวมถึงการฝึกอบรมผู้ใช้

7. การบำรุงรักษาระบบ (System Maintenance) เป็นขั้นตอนสุดท้ายที่ใช้ระยะเวลานานที่สุด เนื่องจากเป็นขั้นตอนที่ดูแลรักษาเพื่อให้สามารถใช้งานระบบได้อย่างมีประสิทธิภาพและมีอายุการใช้งานที่ยาวนานที่สุด จนกระทั่งระบบดังกล่าวไม่สามารถตอบสนองความต้องการใช้งานได้อีกต่อไป วงจรการพัฒนาระบบจึงจะเริ่มต้นใหม่อีกครั้ง



ภาพประกอบที่ 2.1 วงจรชีวิตของการพัฒนาซอฟต์แวร์ SDLC
(ที่มา : สุขสวัสดี ณัฏฐวุฒิสิทธิ์, 2559)

ตารางที่ 2.1 สรุปวงจรการพัฒนาระบบ SDLC

หน้าที่	ทำอะไร
1. เข้าใจปัญหา	1. ตระหนักว่ามีปัญหาในระบบ
2. ศึกษาความเป็นไปได้	1. รวบรวมข้อมูล 2. คาดคะเนค่าใช้จ่าย ผลประโยชน์และอื่น 3. ตัดสินใจว่าจะเปลี่ยนแปลงระบบหรือไม่
3. วิเคราะห์	1. ศึกษาระบบเดิม 2. กำหนดความต้องการของระบบ 3. แผนภาพระบบเก่าและระบบใหม่ 4. สร้างระบบทดลองของระบบใหม่
4. ออกแบบ	1. เลือกซื้อคอมพิวเตอร์ฮาร์ดแวร์และซอฟต์แวร์ 2. เปลี่ยนแผนภาพจากการวิเคราะห์เป็นแผนภาพลำดับขั้น 3. คำนึงถึงความปลอดภัยของระบบ 4. ออกแบบ Input และ Output 5. ออกแบบไฟล์ฐานข้อมูล
5. พัฒนา	1. เตรียมสถานที่ 2. เขียนโปรแกรม 3. ทดสอบโปรแกรม 4. เตรียมคู่มือการใช้และฝึกอบรม
6. นำมาใช้งานจริง	1. ป้อนข้อมูล 2. เริ่มใช้งานระบบใหม่
7. บำรุงรักษา	1. เข้าใจปัญหา 2. ศึกษาสิ่งที่จะต้องแก้ไข 3. ตัดสินใจว่าจะแก้ไขหรือไม่ 4. แก้ไขเอกสาร คู่มือ 5. แก้ไขโปรแกรม 6. ทดสอบโปรแกรม 7. ใช้งานระบบที่แก้ไขแล้ว

2.2 โมบายแอปพลิเคชัน (Mobile Application)

Mobile Application เป็นการพัฒนาโปรแกรมประยุกต์สำหรับอุปกรณ์เคลื่อนที่ เช่น โทรศัพท์มือถือแท็บเล็ตโดยโปรแกรมจะช่วยตอบสนองความต้องการของผู้บริโภค อีกทั้งยังสนับสนุน ให้ผู้ใช้โทรศัพท์ได้ใช้ง่ายยิ่งขึ้น ในปัจจุบันโทรศัพท์มือ หรือ สมาร์ทโฟน มีหลายระบบปฏิบัติการที่พัฒนาออกมาให้ผู้บริโภคใช้ ส่วนที่มีคนใช้และเป็นที่นิยมมากก็คือ iOS และ Android จึงทำให้เกิดการเขียนหรือพัฒนา Application ลงบนสมาร์ทโฟนเป็นอย่างมาก อย่างเช่น แผนที่, เกมส์, โปรแกรมคุยต่าง ๆ และหลายธุรกิจก็เข้าไปเน้นในการพัฒนา Mobile Application เพื่อเพิ่มช่องทางในการสื่อสารกับลูกค้ามากขึ้น ตัวอย่าง Application ที่ติดมากับโทรศัพท์ อย่างแอปพลิเคชันเกมส์ชื่อดังที่ชื่อว่า Angry Birds หรือ Facebook ที่สามารถแชร์เรื่องราวต่าง ๆ ไม่ว่าจะเป็น ความรู้สึก สถานที่ รูปภาพ ผ่านทางแอปพลิเคชันได้โดยตรงไม่ต้องเข้าเว็บบราวเซอร์

ประโยชน์ของ โมบายแอปพลิเคชัน (Mobile Application)

ปัจจุบันนี้ผู้ที่ประกอบธุรกิจต่าง ๆไม่ว่าจะขนาดใหญ่หรือขนาดเล็กได้ให้ความสนใจในการที่จะพัฒนา Mobile Application เพื่อให้เป็นอีกช่องทางหนึ่งในการโฆษณา ประชาสัมพันธ์ รวมไปถึงติดต่อกลุ่มลูกค้าของตน อาทิเช่น

- กลุ่มธุรกิจการการเท่องเที่ยวและการเดินทาง มีการจัด Mobile App ระบบแนะนำสถานที่ท่องเที่ยว แนะนำโรงแรมและการจองโรงแรมที่พัก มีระบบการจองตั๋วรถทัวร์ จองตั๋วเครื่องบิน รวมไปถึงการเช็คอินได้ด้วย

- กลุ่มธุรกิจเพื่อการศึกษา มีการโฆษณาประชาสัมพันธ์ การซื้อขายสื่อการเรียนการสอน การแลกเปลี่ยนความรู้ต่าง ๆ กลุ่มธุรกิจการค้าและแฟชั่น หรือมีระบบการซื้อขายแลกเปลี่ยนสินค้าเป็นต้น

ประโยชน์ด้านผู้ให้บริการ

- กลุ่มธุรกิจต่าง ๆ ทั้งขนาดใหญ่และขนาดเล็ก มีเครื่องมือในการประชาสัมพันธ์ ธุระกิจของตนเอง ทำให้เป็นที่รู้จักในสังคมมากยิ่งขึ้น

- ลดค่าใช้จ่ายในการผลิตสื่อเพื่อโฆษณาประชาสัมพันธ์

- เพิ่มภาพพจน์ร้านให้ดูดี มีจุดเด่น มีจุดขายที่ชัดเจน ส่งผลให้ ธุรกิจมียอดขายที่เพิ่มขึ้น

- มีการบริการลูกค้าสัมพันธ์ที่ดีขึ้น

- ลดขั้นตอนการทำงาน ทำให้การติดต่องานต่าง ๆสะดวกมากยิ่งขึ้น

ประโยชน์ด้านผู้บริโภค

- ลดขั้นตอน และประหยัดเวลา ในการติดต่อธุรกรรมต่าง ๆ เช่น ต้องการโอนเงินค่าสินค้าก็สามารถใช้ Mobile App ทำธุรกรรมทางการเงินได้เลย เป็นต้น

- มีตัวเลือกที่หลากหลาย สามารถเปรียบเทียบ คุณภาพ และราคาได้ โดยที่ไม่ต้องออกไปสำรวจราคาสินค้าที่เราต้องการให้เสียเวลา

- ประหยัดค่าใช้จ่ายในการออกไปเลือกซื้อสินค้าและบริการต่าง ๆ

- ช่วยลดเวลาในการเดินทาง คำนวณเวลาได้ ค้นหาเส้นทางหรือแนะนำเลี่ยงเส้นทางที่มีการจราจรติดขัดได้

ระบบแอปพลิเคชัน

เป็นส่วนซอฟท์แวร์ระบบทีรองรับการใช้งานของแอปพลิเคชัน หรือ โปรแกรมต่าง ๆ ได้ ในปัจจุบันระบบปฏิบัติการที่นิยมจากค่ายอุปกรณ์เคลื่อนที่ต่าง ๆ มีดังนี้

1) Symbian OS จุดเด่นอยู่ที่รูปแบบส่วนติดต่อผู้ใช้งาน (UI) ที่ดูเรียบง่าย มีฟังก์ชัน การใช้งานพื้นฐานอย่างครบครัน อีกทั้งยังติดตั้งแอปพลิเคชัน รวมทั้งไฟล์สื่อต่าง ๆ ไม่ว่าจะเป็นรูปภาพหนัง หรือเพลงได้อย่างสะดวก เพราะมีทรัพยากรหน่วยความจำในเครื่องที่มีประสิทธิภาพ จุดเด่นของ Symbian OS คือ เหมาะสำหรับผู้ที่ชอบความง่ายในการติดตั้งโปรแกรม และลงเพลงต่าง ๆ และรองรับการใช้งานทีหลากหลาย

2) Windows Mobile พัฒนาโดยบริษัท ไมโครซอฟท์ ที่ผลิตระบบปฏิบัติการที่รองรับการทำงานของคอมพิวเตอร์มากมาย ได้แก่ Windows XP, Windows Vista หรือ Windows 7 เป็นต้น ลักษณะการใช้งานของ Windows Mobile คล้ายคลึงกับ Windows ในเครื่องคอมพิวเตอร์ ตัวอย่างสมาร์ทโฟนที่ใช้ Windows Mobile ได้แก่ HTC, Acer เป็นต้น

3) BlackBerry OS พัฒนาโดยบริษัท RIM เพื่อรองรับการทำงานของแอปพลิเคชัน ต่าง ๆ ของ BlackBerry โดยตรง จะเน้นการใช้งานทางด้านอีเมล์เป็นหลัก ซึ่งเมื่ออีเมล์เข้ามาสู่ระบบเซิร์ฟเวอร์จะทำการส่งต่อมายัง BlackBerry โดยตรง จะมีการเตือนสถานะที่หน้าจอเพื่อให้ผู้ใช้ได้รับข้อมูลอย่างทันนท่วงที่ ซึงระบบอีเมล์ของ BlackBerry จะมีความปลอดภัยสูงด้วยการเข้ารหัสข้อมูล ส่วนจุดเด่นสำคัญอีกอย่างหนึ่ง คือ ระบบการสนทนาผ่าน BlackBerry Messenger ซึ่งจะทำให้สามารถพิมพ์ข้อความสนทนากับเพื่อน ๆ ทีมี BlackBerry เช่นกันเป็นเรียลไทม์ ด้วยความสามารถในการเชื่อมต่ออินเตอร์เน็ต และมีการเปิดให้รับ – ส่งข้อมูลกับเครือข่ายมือถืออยู่ตลอดเวลา เหมาะสำหรับผู้ที่ต้องติดต่องานต่าง ๆ ผ่านอีเมล์ และกลุ่มวัยรุ่นที่รักการสนทนาผ่านคอมพิวเตอร์

4) iPhone OS พัฒนาโดยบริษัท Apple เพื่อรองรับการทำงานของแอปพลิเคชัน ต่าง ๆ ของ iPhone โดยตรง โดยกลุ่มที่นิยมใช้ iPhone มักจะเป็นผู้ที่ชอบด้านมัลติมีเดีย เช่น การฟังเพลง ดูหนัง หรือการเล่นเกม เป็นต้น บริษัทเกมหลายแห่งจึงผลิตเกมขึ้นมาเพื่อรองรับการทำงานบนiPhone โดยเฉพาะ ซึ่งผู้ที่ใช้สามารถซื้อขายแอปพลิเคชัน ต่าง ๆ บนอินเตอร์เน็ต แล้วชำระเงินผ่านทางบัตรเครดิต ซึ่งเป็นธุรกิจอีกหนึ่งประเภทที่กำลังเติบโตไปพร้อมกับธุรกิจในกลุ่มสมาร์ทโฟน

5) Android OS พัฒนาโดยบริษัท Google เป็นระบบปฏิบัติการล่าสุดที่กำลังเป็นที่นิยม รองรับการเชื่อมต่ออินเตอร์เน็ตแบบเรียลไทม์เพื่อใช้บริการจาก Google ได้อย่างเต็มที่ทั้ง Search Engine, Gmail, Google Calendar, Google Docs และ Google Maps มีจุดเด่น คือ เป็นระบบปฏิบัติการแบบ Open Source ซึ่งทำให้มีการพัฒนาไปอย่างรวดเร็ว ซึ่งตอนนี้มีโปรแกรมให้เลือกใช้งานอยู่มากมาย จึงเหมาะสำหรับผู้ที่ต้องการใช้งานบริการต่าง ๆ จากทาง Google รวมทั้งต้องการเชื่อมต่ออินเตอร์เน็ตอยู่ตลอดเวลา

แอปพลิเคชันที่สนองความต้องการของกลุ่มผู้ใช้

เนื่องจากผู้ใช้มีความต้องการใช้แอปพลิเคชันแตกต่างกัน จึงมีผู้ผลิต และพัฒนาแอปพลิเคชัน ใหม่ ๆ ขึ้นเป็นจำนวนมาก ได้แก่

1. แอปพลิเคชันในกลุ่มเกม เนื่องจากมีผู้นิยมเล่นเกมบนโทรศัพท์เป็นจำนวนมาก ผู้ผลิตเกมจึงคิดค้นเกมใหม่ ๆ ออกสู่ตลาดมากขึ้น ซึ่งผู้เล่นมักนิยมเล่นเกมออนไลน์ รวมทั้งมีการเชื่อมโยงกันในกลุ่มเครือข่ายสังคมออนไลน์ (Social Networking) เช่น เกมที่อยู่ใน Twitter หรือFacebook เป็นต้น

2. แอปพลิเคชันในกลุ่มเครือข่ายสังคมออนไลน์ เพื่อให้ผู้ใช้สามารถปรับข้อมูลให้ทันสมัยตลอดเวลา ทั้งข้อมูลของตนเอง หรือของกลุ่มเพื่อน ซึ่งกำลังเป็นที่นิยมในกลุ่มวัยรุ่นอย่างสูง เช่นใน Facebook, Tweetter เป็นต้น และแม้แต่ LINE ก็มีช่องทางเพื่อให้ลูกค้าได้สนทนากันผ่านทาง LINE โดยการแลก PIN กับเพื่อน ๆ ในกลุ่ม

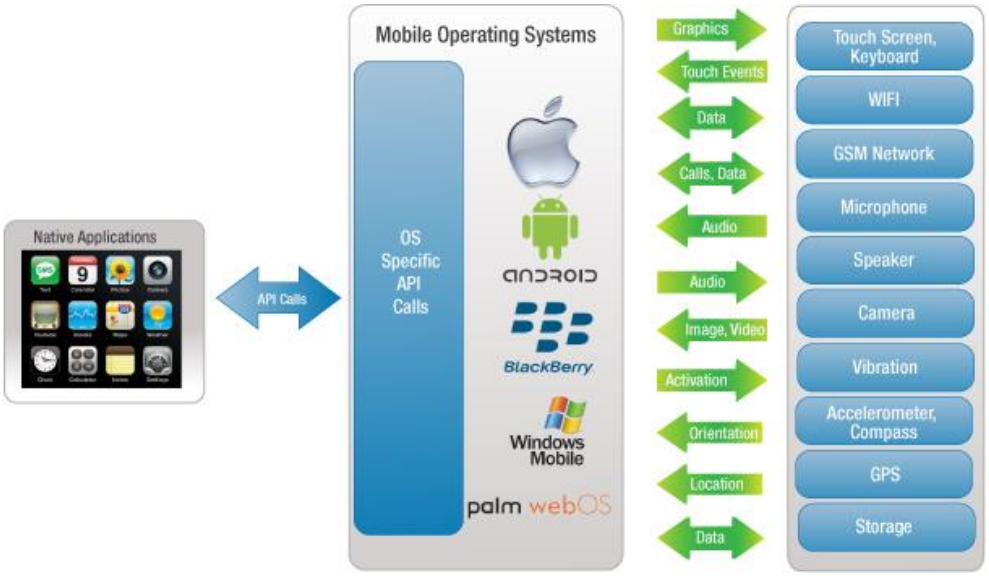
3. แอปพลิเคชันในกลุ่มมัลติมีเดีย เพื่อให้ผู้ใช้สามารถเรียกใช้ไฟล์ข้อมูลในรูปแบบต่าง ๆ ได้แก่ เสียงที่เป็นไฟล์ในแบบ mp3, wav หรือ midi เป็นต้น ภาพนิ่งในรูปแบบ gif, jpg หรือ bmp เป็นต้น หรือภาพเคลื่อนไหวคลิปวิดีโอในรูปแบบ mp4 หรือ avi เป็นต้น

ตารางที่ 2.2 รายละเอียดของแอปพลิเคชัน ในกลุ่มต่าง ๆ (Application Categories)

Categories	Description
Games	- กลุ่มเกมมีหลายชนิด เช่น action, arcade, puzzle, card, casual เป็นต้น
Lifestyle & Healthcare	- เป็นโปรแกรมดูแลสุขภาพ เช่น calorie trackers หรือ pedometers และโปรแกรมที่เกี่ยวกับวิถีการดำเนินชีวิต หรือ lifestyle เช่น location-based search, navigation, news & infotainment, photography, travel
Education & Reference	- เป็นโปรแกรมในกลุ่มการศึกษา แลหลักฐานอ้างอิง ได้แก่ Ebooks, language courses, encyclopedias, IQtests,atlases, othereducational aids เป็นต้น
Multimedia & Entertainment	- โปรแกรมสำหรับความบันเทิง เช่น ฟังเพลง ดูหนัง ดูทีวีหรือ - โปรแกรมตกแต่งพื้นหลังของโทรศัพท์ เป็นต้น
Finance & Productivity	- โปรแกรมในด้านการเงิน ได้แก่ Currency converters, taxcalculators, budget management, mobile banking เป็นต้น - Personal management, typing tutorials, document readers, spreadsheets, spell checkers, etc.
Social Networking	- โปรแกรมที่อำนวยความสะดวกในการเข้าใช้เครือข่ายสังคมออนไลน์ เช่น facebook, Tweetter เป็นต้น

2.2.1 เนทีฟ แอปพลิเคชัน (Native Application)

Native Application คือ แอปพลิเคชันที่ถูกพัฒนาขึ้นโดยอาศัย Library หรือ SDK ของแพลตฟอร์ม (Platform) นั้น ๆ และจะต้องพัฒนาด้วยภาษาของแต่ละแพลตฟอร์ม เช่น แอนดรอยด์ (Android) ใช้ภาษาจาวา (Java) วินโดวส์โฟน (Windows Phone) ใช้ภาษาซีชาร์ป (C#) และไอโอเอส (iOS) ใช้ภาษาอ็อปเจ็คซี (Object-C) เป็นต้น ทั้งนี้ข้อดีของการพัฒนาแอปพลิเคชันแบบ Native คือสามารถดึงทรัพยากรของระบบมาใช้งานได้เต็มที่และมีประสิทธิภาพสูงสุด แต่ก็ยังมีข้อเสียก็คือ เมื่อต้องการพัฒนาแอปพลิเคชันให้สามารถใช้งานกับแพลตฟอร์มอื่นได้ จะต้องเริ่มพัฒนาแอปพลิเคชันใหม่ ซึ่งทำให้ต้นทุนในการพัฒนาสูงและใช้เวลานาน



ภาพประกอบที่ 2.2 Native Application Interaction with Mobile Device (ที่มา : XRG, 2015)

ข้อดีของการพัฒนาแอปพลิเคชันแบบเนทีฟ

- สามารถเข้าถึงฟังก์ชันการทำงานของ Platform นั้น ๆ ได้อย่างครบถ้วนและมีประสิทธิภาพ

- ประสิทธิภาพการทำงานสูงสุด มีความยืดหยุ่นเอื้อประโยชน์ต่อนักพัฒนาสูงสุด

- ด้าน UX (User Experience) แอปพลิเคชัน แบบเนทีฟจะถูกพัฒนาอยูบน่ พื้นฐาน System interface components ทำให้ End User คุ้นเคย และเข้าใจการใช้ งานได้ดีกว่า

เครื่องมือในการพัฒนา Native Application

• ระบบ Android พัฒนาโดยใช้ Android Studio ภาษา Java

• ระบบ iOS พัฒนาโดยใช้ Xcode ภาษา C++ หรือ Swift

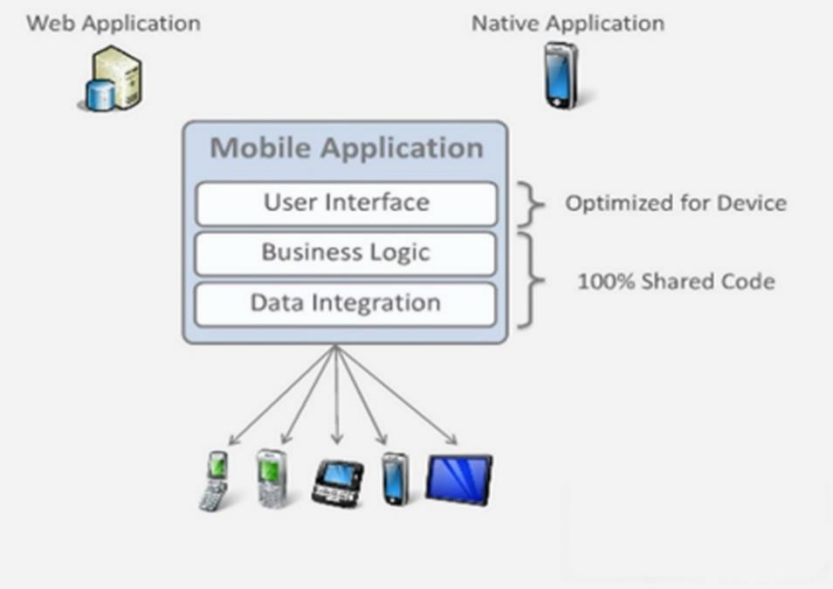
• ระบบ Windows พัฒนาโดยใช้ Visual Studio ภาษา C++

ภาพประกอบที่ 2.3 แบบฟอร์มการพัฒนาโปรแกรมประยุกต์บนมือถือ (MADPS)

(ที่มา : Charles McLellan, 2014)

2.3 ระบบปฏิบัติการ iOS

ระบบปฏิบัติการไอโอเอส (iOS) มีชื่อเดิมว่า iPhone OS เริ่มต้นด้วยการเปิดตัวของ iPhone เมื่อวันที่ 29 มิถุนายน 2550 ระบบปฏิบัติการไอโอเอส (iOS) เป็นระบบปฏิบัติการสำหรับสมาร์ทโฟน (Smartphone) ของแอปเปิล โดยเริ่มต้นพัฒนาสำหรับใช้ในโทรศัพท์ iPhone และได้พัฒนาต่อใช้สำหรับ iPot Touch และiPad ปัจจุบันระบบปฏิบัติการไอโอเอส (iOS) ได้พัฒนาเป็นเวอร์ชั่น 8.x หรือที่เรียกว่า iOS 8 (Apple, 2014) ซึ่งมีอินเทอร์ เฟส เหมือน iOS 7 แต่ปรับเปลี่ยนฟีเจอร์บางอย่างเล็กน้อย โดยบน iOS 8 นั้น รองรับการติดตั้งคีย์บอร์ดจากผู้พัฒนารายอื่นแล้ว หลังจากที่ผูกขาดเฉพาะคีย์บอร์ดที่แอปเปิลพัฒนามาโดยตลอด นอกจากนี้ ในส่วนของ Notification จะเป็นการแจ้งเตือนที่สามารถตอบกลับได้เลย โดยที่ไม่ต้องเปิดแอปพลิเคชันนั้น ๆ, ส่วนของ Multi-tasking เพิ่มตัว Favourite และผู้ติดต่อที่ติดต่อกันบ่อยที่สุด ไม่เพียงเท่านั้น ยังเน้นการใช้งานด้านสุขภาพมากขึ้นกับแอปฯ Health และสุดท้าย รองรับการใช้งานกับ Apple Watch



ภาพประกอบที่ 2.4 สถาปัตยกรรมระบบไอโอเอส (iOS) (ที่มา : ABT, 2012)

คุณสมบัติของระบบปฏิบัติการ iOS

หน้าจอหลัก (Home screen)

เป็นส่วนแสดงข้อมูลที่ประกอบด้วย application ที่มีบนเครื่อง ซึ่ง Application เหล่านี้รองรับการทำงานในระบบปฏิบัติการiOS สามารถดาวน์โหลดหรือซื้อได้ที่ Apple's store หรือผ่านเว็บไซต์ของแอปเปิลได้โดยตรง การใช้งานทำได้ง่ายและสะดวกเพียงนำนิ้วจิ้มหรือแตะไปที่รูปไอคอนที่ต้องการใช้งาน

แฟ้มข้อมูล (Folders)

iOS 4 การแนะนำของระบบโฟลเดอร์ที่เรียบง่ายมา เมื่อใช้งานอยู่ในโหมด "กระตุก" ใด ๆ สองสามารถลากด้านบนของแต่ละอื่น ๆ เพื่อสร้างโฟลเดอร์และจากนั้นมาปรับแอปพลิเคชันเพิ่มเติมสามารถเพิ่มลงในโฟลเดอร์โดยใช้ขั้นตอนเดียวกันได้ถึง 12 บน iPhone และ iPod touch และ 20 ใน iPad ชื่อสำหรับโฟลเดอร์ที่ถูกเลือกโดยอัตโนมัติตามประเภทของการใช้งานภายใน แต่ชื่อยังสามารถแก้ไขได้โดยผู้ใช้

การแจ้งเตือน (Notification)

ใน iOS 5 ปรับปรุงคุณลักษณะการแจ้งเตือนที่ถูกออกแบบใหม่อย่างสมบูรณ์ เพื่อให้การทำงานรวดเร็วและตอบสนองความต้องการของผู้ใช้ได้ง่ายขึ้น

2.4 ภาษาสวิฟท์ (Swift)

Swift ประกาศในปี 2014 ภาษา Swift ได้กลายเป็นหนึ่งในภาษาที่เติบโตเร็วที่สุด ซึ่งภาษา Swift เป็นโปรแกรมภาษาใหม่สำหรับการพัฒนาแอปพลิเคชันบน iOS และ OSX โดยมีพื้นฐานมาจากภาษา C และ Objective C แต่จะพัฒนาให้มีการใช้รูปแบบการเขียนโปรแกรมที่ปลอดภัย และเพิ่มลักษณะที่ทันสมัย เพื่อใช้สร้าง และปรับปรุงแก้ไขโปรแกรมได้ง่ายขึ้น และทำให้การทำงานของซอฟแวร์เป็นไปอย่างมีประสิทธิภาพและเปิดให้เป็นโอเพ่นซอร์ส ความนิยมเพิ่มมากขึ้นจนมาติด 1 - 20 ภาษาเขียนโปรแกรมที่นิยมมากที่สุด

Jan 2016	Jan 2015	Change	Programming Language	Ratings	Change
1	2	^	Java	21.465%	+5.94%
2	1	v	C	16.036%	-0.67%
3	4	^	C++	6.914%	+0.21%
4	5	^	C#	4.707%	-0.34%
5	8	^	Python	3.854%	+1.24%
6	6		PHP	2.706%	-1.08%
7	16	^^	Visual Basic .NET	2.582%	+1.51%
8	7	v	JavaScript	2.565%	-0.71%
9	14	^^	Assembly language	2.095%	+0.92%
10	15	^^	Ruby	2.047%	+0.92%
11	9	v	Perl	1.841%	-0.42%
12	20	^^	Delphi/Object Pascal	1.786%	+0.95%
13	17	^^	Visual Basic	1.684%	+0.61%
14	25	^^	Swift	1.363%	+0.62%
15	11	vv	MATLAB	1.228%	-0.16%
16	30	^^	Pascal	1.194%	+0.52%
17	82	^^	Groovy	1.182%	+1.07%
18	3	vv	Objective-C	1.074%	-5.88%
19	18	v	R	1.054%	+0.01%
20	10	vv	PL/SQL	1.016%	-1.00%

ภาพประกอบที่ 2.5 ความนิยมของภาษาสวิฟท์ (ที่มา : บริษัท TIOBE Software BV, 2018)

Swift ช่วยให้เขียนซอฟต์แวร์ที่รวดเร็วและปลอดภัยอย่างไม่น่าเชื่อโดยการออกแบบเป้าหมายของ Swift มีความทะเยอทะยาน ทำให้การเขียนโปรแกรมเป็นเรื่องง่ายและเป็นไปได้ยาก Swift เป็นภาษาคอมพิวเตอร์จากบริษัท Apple ใช้เพื่อสร้างและพัฒนาแอปพลิเคชันบนอุปกรณ์ iOS, Apple TV, Apple Watch และ macOS ซึ่งแต่เดิมการพัฒนาแอปพลิเคชันบนเครื่องแมคอินทอชหรืออุปกรณ์ต่าง ๆ ของ Apple จะใช้ภาษา Objective-C แต่ด้วยการพัฒนาด้านต่าง ๆ ทั้งซอฟต์แวร์และฮาร์ดแวร์ ภาษา Objective-C ที่ใช้มายาวนานจึงไม่อาจตอบรับเทคโนโลยีใหม่ๆ ได้ทั้งหมด ดังนั้นภาษา Swift จึงถูกพัฒนาขึ้นมาแทนที่ ซึ่งมีคุณสมบัติที่มากมาย อาทิ Type Inference, Optional,

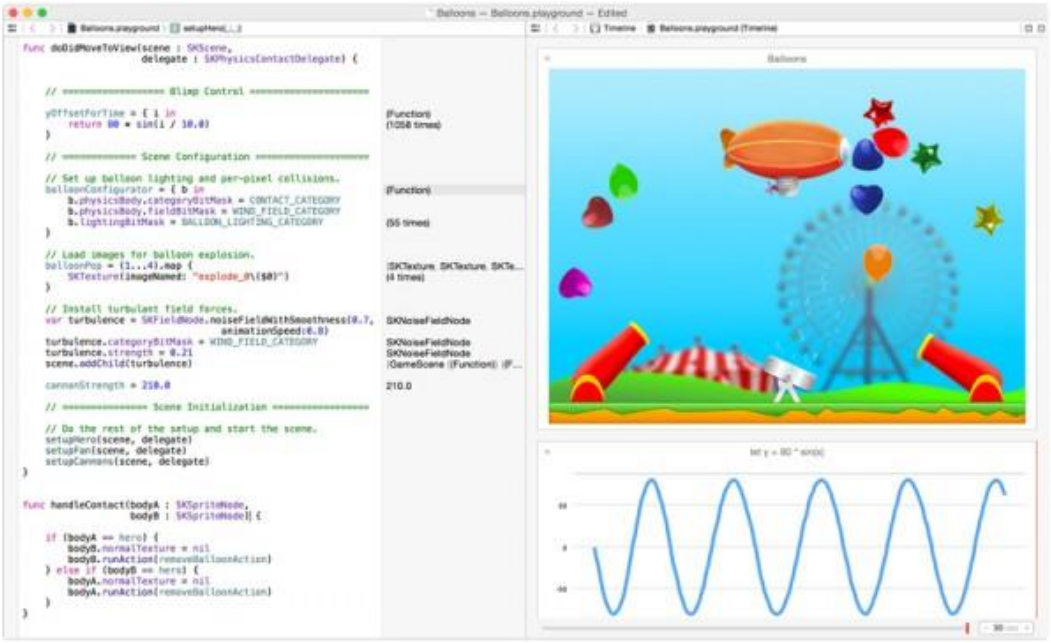
Closures, Tuples, Generics, Extension และอื่น ๆ แต่แม้ว่า Swift จะถูกพัฒนาขึ้นใหม่ทั้งหมด แต่ภาษา Swift ยังสามารถใช้งานร่วมกับภาษา C หรือ Objective-C ได้อย่างไม่มีปัญหา ซึ่งหมายความว่าในแอปพลิเคชันสามารถใช้ภาษา Swift ผสมกับภาษา Objective-C หรือภาษา C ก็ได้ ภาษา Swift ยังถูกออกแบบให้มีความปลอดภัยในการเขียนโปรแกรมมากขึ้น เป็นความปลอดภัยในเชิงของการพัฒนา ทำให้ลดข้อผิดพลาดของโปรแกรมที่พัฒนาขึ้น ยกตัวอย่างเช่น

- ไม่อนุญาตให้มีตัวแปรที่ไม่ได้ถูกกำหนดค่าในโปรแกรม
- ไม่ต้องเขียนสัญลักษณ์ * (Asterisk) ขณะประกาศตัวแปร Pointer
- ตรวจสอบการใช้งานค่าต่ำสุดและสูงสุดของตัวเลขจำนวนเต็ม
- จะต้องเขียนวงเล็บปีกกาครอบส่วนของโปรแกรมที่อยู่ภายใต้เงื่อนไขใด ๆ

สำหรับสวิฟท์ในปีที่ผ่านมา Apple ได้วางรากฐานสวิฟท์โดยยกระดับ compiler ที่มีอยู่ของพวกเรา แก้ไขจุดบกพร่อง และวางกรอบโครงสร้างพื้นฐาน เราสามารถจัดการหน่วยความจำได้ง่ายขึ้นด้วย Automatic Reference counting (ARC) กรอบโครงสร้างของเราสร้างขึ้นจากฐานที่มั่นคงของ Cocoa ที่ทันสมัยและมีมาตรฐานทั้งหมด Objective-C ตัวมันเองมีการพัฒนาเพื่อสนับสนุน blocks ตัวเก็บรวบรวม และ modules งานนี้ทำให้เกิดเทคโนโลยีภาษาที่ทันสมัยมาใช้โดยไม่ทำให้เกิดปัญหา ขอบคุณรากฐานนี้ ที่ทำให้ตอนนี้พวกเราสามารถเข้าสู่ภาษาใหม่สำหรับการพัฒนาซอฟแวร์ Apple ในอนาคต Swift เหมือนนักพัฒนา Object - C มันนำเอาความสามารถในการอ่านค่าพารามิเตอร์ของ Object - C และรูปแบบวัตถุไดนามิกของ Object - C มาใช้ มันสามารถเข้าถึงการทำงานของ CoCoa ที่มีอยู่ได้อย่างราบรื่น และผสมผสานกับความสามารถในการทำงานร่วมกับรหัส Object - C การสร้างจากพื้นฐานธรรมดานี้ ทำให้สวิฟท์ได้เปิดตัวลักษณะเด่น (คุณสมบัติ) ใหม่ๆ หลายอย่างและรวมขั้นตอนบางส่วนของ Object-oriented

สวิฟท์ง่ายต่อโปรแกรมเมอร์มือใหม่ มันคือ ระบบการเขียนโปรแกรมภาษาต้นๆ ระดับคุณภาพ นั่นหมายถึงภาษาที่แสดงออกและมันมีลูกเล่นที่หลากหลาย คุณสมบัติใหม่ๆ ที่ช่วยให้โปรแกรมเมอร์ทดลองใช้กับ swift code และรู้ผลทันที ไม่มีค่าใช้จ่ายและไม่มีการเรียกใช้แอพในการสร้าง

สวิฟท์ เป็นการผสมผสานที่ดีที่สุดในการคิดภาษาสมัยใหม่ด้วยความฉลาดของ Apple ที่กว้างขึ้น The compiler ได้ปรับให้มีประสิทธิภาพและภาษาเหมาะสำหรับการพัฒนา โดยปราศจากข้อผูกมัด มันออกแบบมาเพื่อปรับขนาด “hello, world” ไปสู่ระบบปฏิบัติการทั้งหมด ทั้งหมดนี้ทำให้ swift เป็นหนึ่งในการลงทุนที่ดีในอนาคตสำหรับนักพัฒนาและสำหรับ Apple



ภาพประกอบที่ 2.6 ภาษาสวิฟท์ (Swift) (ที่มา : Apple, 2014)

การกำหนดค่าตัวแปรและการคำนวณ รวมถึงการแปลงค่า

```
var str = "Hello, playground"
let name = "game"
```

การกำหนดค่าแบบ var และ let

ในส่วนการกำหนดค่าโดย var กับ let แตกต่างกัน โดยการกำหนดแบบ let จะไม่สามารถเปลี่ยนแปลงค่าได้ เหมือนกับการล็อคค่าไว้ไม่ให้มีเปลี่ยนแปลง ถ้าเราพยายามจะเปลี่ยนค่าก็จะขึ้น error ซึ่งจะตรงข้ามกับ var ซึ่ง var สามารถเปลี่ยนแปลงค่าได้

```
var str = "Hello, playground"
str = "hello game" // ค่าที่ได้จะเป็น hello game

let name = "game"
name = "GG" // แบบนี้จะerror
```

การตั้งชื่อตัวแปรก็เหมือนกับภาษาอื่นที่ไม่สามารถใช้ชื่อเฉพาะได้ โดยการนำ “ตัวแปร” สามารถใช้ชื่อนั้นได้

```
let `let` = 10
print(`let`)
```

แบบนี้จะไม่เกิด error

การคำนวณ การนับตัวอักษรในภาษา Swift ซึ่งการนับตัวอักษรต้องใช้ String ซึ่งคำสั่งก็ไม่ได้ยากอะไรมาก เพียงแค่มี characters.count

```
let str = "hello"
var num = str.characters.count
print(num)
```

ผลลัพธ์ คือ 5

เมื่อกำหนดค่าเป็นแล้วเราก็เริ่มเข้าสู้ขั้นตอนต่อไป คือ การคำนวณ ซึ่งส่วนนี้ไม่ได้ต่างจากภาษาCมาก

```
let x = 2 , y = 4 , z = 5
var ans = 0
ans = x + y + z

print(ans) //ผลลัพธ์ คือ 11
```

การกำหนด var กับ let จะใช้ var ไปทั้งหมดได้ คิดที่จะมีการเปลี่ยนค่าตัวแปรที่นำมาใช้งาน เช่น

```
var x = 2 , y = 4 , z = 5
var ans = 0

x=10
y=y-1
z=z*z

ans = x + y + z
ans = ans/2

print(ans)
```

ผลลัพธ์ คือ 19

เครื่องหมายที่ใช้ไม่ได้แตกต่างจากภาษาC ซึ่งในกรณีนี้ ค่า ans แรก ที่ได้จากการบวกรวมกันนั้นเท่ากับ 38 เมื่อหารด้วย 2 ซึ่งเป็นการหารลงตัวนั้นได้จำนวนเต็มพอดี เลยผลลัพธ์ที่ได้ออกมาจึงมีผลลัพธ์ที่ออกมาตรง ถ้าลองเปลี่ยนเป็นเลขอื่นที่หารไม่ลงตัว คำตอบที่ได้ก็จะออกมาไม่ตรง อันนี้ไม่ต่างจากภาษาอื่นที่ว่า Int/Int=Int ดังนั้นถ้าได้ค่าที่เป็นเศษจากการหารไม่ลงตัวเราสามารถทำได้โดยวิธีการดังนี้

```
var x = 2 , y = 4 , z = 5
var ans : Float = 0

x=10
y=y-1
z=z*z

ans = Float(x + y + z)
ans = ans/3

print(ans)
```

ผลลัพธ์ คือ **12.6667**

ถ้าเลขที่เป็นทศนิยม การใช้ Float กับ Double นั้นมีความแตกต่างกันอย่างไร คือ Double จะมีความละเอียดของตัวเลขมากขึ้น ซึ่งจากตัวอย่างข้างบนที่ใช้ Float จะได้ ans = 12.6667 แต่ถ้าลองใส่ Double ไปแทนที่ Float ค่าที่ได้ออกมาจะเท่ากับ 12.6666666666667

การใช้ Optional ซึ่ง Optional จะช่วยบอกว่าตัวแปรนั้นมีค่าอยู่หรือไม่ ซึ่งการใช้ก็ไม่ยาก

```
var num : Int? = 0
print(num!)
```

ผลลัพธ์ คือ 0 ถ้าไม่ใส่ '!' หลังตัวแปร ผลลัพธ์จะได้ Optional(0)

ซึ่งจากตัวอย่างการคำนวณ ไม่จำเป็นที่ต้องตั้ง ans = 0 ไว้ สามารถใช้ Optional โดยการกำหนดค่าเป็น nil ได้ เพื่อให้รู้ว่าค่านั้นยังไม่มีค่า แต่ไม่สามารถนำมาคำนวณได้ทันทีเหมือนการแทนค่าเป็นตัวเลขได้ ถ้าอยากนำตัวแปรนั้นมาใช้งานต่อ จำเป็นต้องแทนค่าให้ตัวแปรตัวนั้นไปก่อน

```
var num : Int? = nil
num = 20
print(num!)

var str : String? = nil
str = "hello"
print(str!)
```

ซึ่งการคำนวณ หรือ การใช้ Operater ที่นำมาข้างต้นเป็นแค่แบบ Binary เพียงเท่านั้น ยังมีแบบ Unary และ Ternary

```
var result : Bool? = nil
result = 5<4 ? true : false
print(result!)
```

ผลลัพธ์ คือ **false**

```
var result : String? = nil
result = 5<4 ? "hello" : "hi"
print(result!)
```

ผลลัพธ์ คือ **hi**

2.5 โปรแกรมเอกซ์โค้ด (Xcode)

เอกซ์โค้ด (Xcode) คือ code เป็นเครื่องมือของนักพัฒนา Apple เอาไว้สร้างแอปพลิเคชันสำหรับเครื่อง Mac, iPhone และ iPad ถ้าเทียบกับเครื่องมือพัฒนาของฝั่งไมโครซอฟท์ก็คือ Visual Studio นั่นเอง Xcode เวอร์ชั่นล่าสุด ณ ที่เขียนบทความนี้ คือเวอร์ชั่น 4 ต้องติดตั้งลงบนเครื่องที่มี OS เป็น Mac OS X เวอร์ชั่น 10.6 ขึ้นไป สามารถหาซื้อได้จาก Mac App Store ในราคา $4.99 เหรียญสหรัฐ แต่ถ้าคุณสมัครเป็นสมาชิกของ iOS หรือ Mac Developer Program เสียค่าสมาชิกปีละ $99 เหรียญสหรัฐ สามารถโหลดได้ฟรี Xcode 4 ประกอบด้วย Xcode IDE, Instruments และ SDK สำหรับ Mac OSX และ iOS

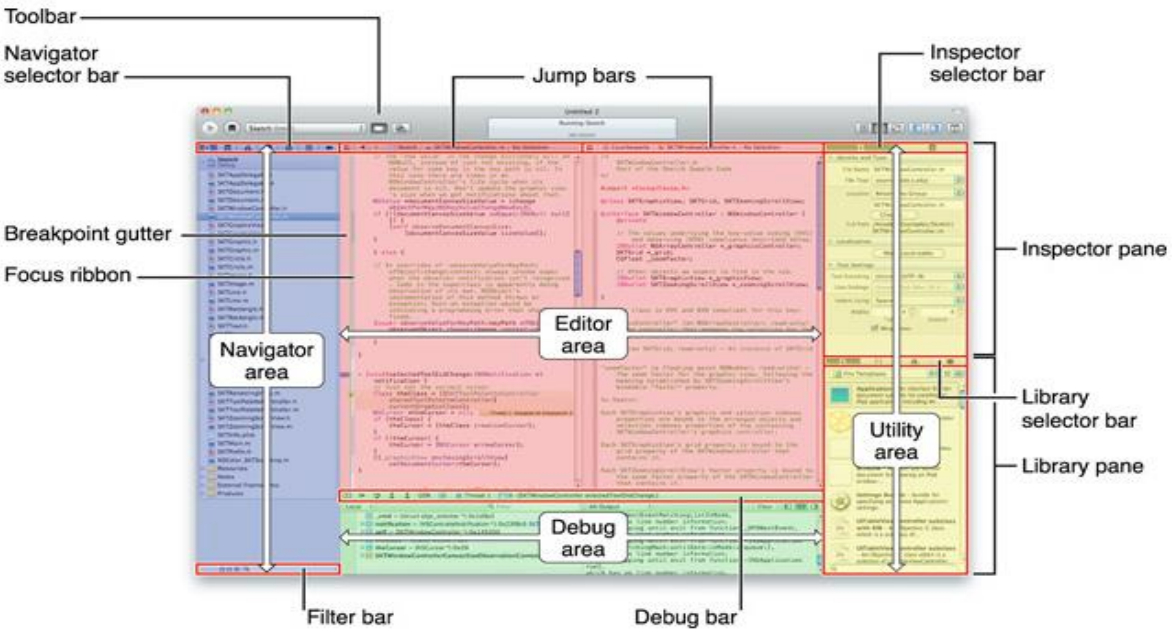
- Xcode IDE คือ สภาพแวดล้อมในการพัฒนาแอปพลิเคชัน ประกอบด้วยพื้นที่ทำงานสำหรับเขียน code , พื้นที่สำหรับออกแบบหน้าจอ user interface เรียกว่า Interface builder, มีคอมไพล์เลอร์ที่มีประสิทธิภาพสูง ชื่อว่า LLVM Compiler 2.0 สามารถคอมไพล์โค้ดได้เร็วกว่า GCC ถึง 2 เท่า สร้างแอปพลิเคชันให้ทำงานได้เร็วขึ้น มีระบบตรวจสอบโค้ดที่มีประสิทธิภาพ แก้ไขโค้ดที่ผิดให้อัตโนมัติ, มี Version Editor ที่ตรวจสอบ Source Code ทั้ง 2 เวอร์ชั่น แบบเทียบกันหน้าต่อหน้าให้เห็นบรรทัดที่แตกต่างกันได้อย่างมีประสิทธิภาพ, มี Debugger Engine ที่ชื่อว่า LLDB สามารถ track code ได้ขณะที่โปรแกรมกำลัง run ทดสอบอยู่, มี Source Control ในการควบคุมเวอร์ชั่นของ Project ได้

- Instruments เป็นเครื่องมือที่ใช้ตรวจสอบประสิทธิภาพของแอปพลิเคชันที่พัฒนาเสร็จแล้ว โดยวัดการใช้งาน memory, CPU Time, Overhead ต่าง ๆ

- SDK สำหรับ Mac OS X และ iOS คือ Class Library สำหรับใช้พัฒนาแอปพลิเคชันแบ่งเป็น Mac OS X และ iOS

ภาพประกอบที่ 2.7 โปรแกรมเอกซ์โค้ด (Xcode) (ที่มา : Apple, 2018)

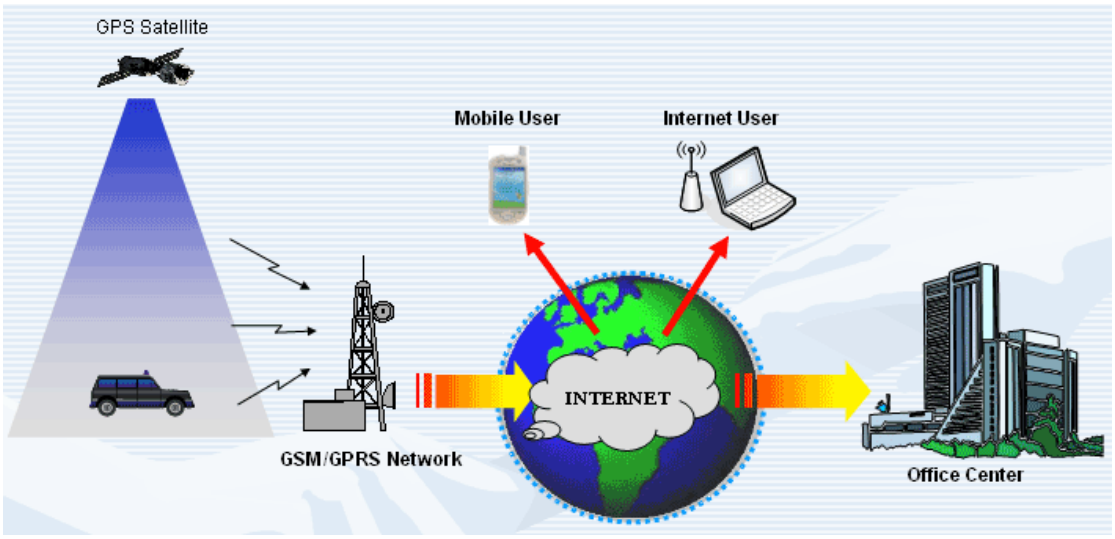
โครงสร้างของ Xcode IDE และ ไฟล์ที่ได้จากการ Create Project บน Xcode (iOS, iPhone, iPad) การเริ่มเขียนโปรแกรมบน iPhone และ iPad ด้วย Tools ของ Xcode IDE จะประกอบด้วยหลาย ๆ ส่วน และ Item ไว้สำหรับปรับแต่ง Project ใน Project มีไฟล์หรือโฟเดอร์ ที่ถูกสร้างมาให้อัตโนมัติ



ภาพประกอบที่ 2.8 โครงสร้างของ Xcode (ที่มา : Apple, 2015)

2.6 จีพีเอส (GPS)

GPS (Global Positioning System) คือ ระบบระบุตำแหน่งบนพื้นโลก ซึ่งทำงานร่วมกับดาวเทียมบอกตำแหน่งทั้งหมด 24 ดวง ดาวเทียมGPS เป็นดาวเทียมที่มีวงโคจรระดับกลาง (Medium Earth Orbit) ที่ระดับความสูงประมาณ 20,200 กิโลเมตร จากพื้นผิวโลก โดยที่แนวคิดในการพัฒนาระบบGPS เริ่มต้นตั้งแต่ปี ค.ศ. 1957 เมื่อนักวิทยาศาสตร์ของสหรัฐอเมริกา แล้วในปี ค.ศ. 1960 ก็เริ่มทดสอบใช้งานกันจริง ๆในกองทัพเรือสหรัฐอเมริกา แต่สาเหตุที่ทำให้เรามี GPS ใช้กันอย่างแพร่หลายจนถึงทุกวันนี้เกิดจากเหตุการณ์ในปี ค.ศ. 1983 ที่เครื่องบินโคเรียนแอร์ไลน์เที่ยวบินที่ 007 ของเกาหลีใต้ บินพลัดหลงเข้าไปในน่านฟ้าของสหภาพโซเวียต และถูกยิงตก ผู้โดยสาร 269 คนเสียชีวิตทั้งหมด ประธานาธิบดีโรนัลด์ เรแกนได้ประกาศว่า เมื่อพัฒนาระบบจีพีเอสแล้วเสร็จ จะอนุญาตให้ประชาชนทั่วไปใช้งานได้ ทำให้ GPS ได้ถูกพัฒนาในเชิงพาณิชย์นอกจากที่จะใช้ในการทหารเพียงอย่างเดียวเท่านั้นซึ่งระบบ GPS จะทำงานควบคู่กับดาวเทียม GPS เพื่อระบุตำแหน่งบนพื้นผิวโลก โดยตัวเครื่องรับสัญญาณ GPS จะต้องประมวลผลความแตกต่างของเวลาในการรับสัญญาณเทียบกับเวลาจริง โดยสามารถที่จะระบุตำแหน่งบนผิวโลกได้ชัดเจน



ภาพประกอบที่ 2.9 รูปแบบการทำงานของระบบ GPS (ที่มา : บริษัท โทเทิล ดิจิตอล จำกัด, 2015)

GPS มีกี่ประเภท อะไรบ้าง

– GPS ที่เราใช้งานกันอยู่ในปัจจุบันจะมีอยู่ 2 ประเภทด้วยกันคือ GPS Navigator (อุปกรณ์และระบบนำทาง) และ GPS Tracking System (อุปกรณ์และระบบติดตามรถ ยาพาหะนะหรือสัตว์เลี้ยง)

– GPS Navigator (อุปกรณ์และระบบนำทาง) เป็น GPS ที่เราใช้งานในรถยนต์ทั่วไปที่บอกแผนที่การเดินทางด้วยการป้อนข้อมูลของเป้าหมายลงไปเครื่องนำทาง GPS

– GPS Tracking System (อุปกรณ์และระบบติดตามรถ ยาพาหะนะหรือสัตว์เลี้ยง) ซึ่งเป็น GPS ที่สามารถติดตามการเดินทาง และบอกพิกัดและตำแหน่งของ เครื่อง GPS ได้ด้วย โดยเราสามารถแบ่งเป็นออกได้อีก 2 แบบด้วยกันคือ อุปกรณ์ติดตามรถแบบ Offline สามารถตรวจสอบประวัติการเดินทางได้ แต่ไม่สามารถตรวจสอบตำแหน่งที่อยู่ของเครื่อง GPS ได้ และแบบที่สองอุปกรณ์ติดตามรถแบบกึ่ง Offline ซึ่งจะทำงานร่วมกับมือถือเราสามารถที่จะดูประวัติการเดินทางพร้อมทั้งตำแหน่งปัจจุบันของอุปกรณ์ GPS นั้น ๆได้อีกด้วย

ประโยชน์ของ GPS

ประโยชน์ของ GPS นั้นมีมากมาย แต่ที่เห็นได้ชัดเจนก็คือการประหยัดค่าใช้จ่ายในการเดินทาง เราสามารถวางแผนการเดินทางหลีกเลี่ยงเส้นทางที่ไม่สะดวก คำนวณค่าใช้จ่ายในการเดินทางได้อย่างสะดวกสบาย

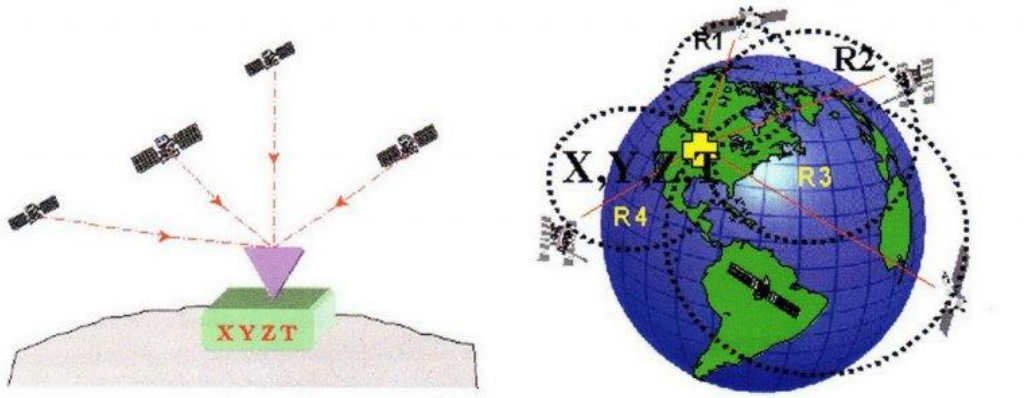
ในด้านธุรกิจขนส่ง ระบบนำทาง GPS สามารถทำให้ผู้ประกอบการตรวจสอบการเดินทางของรถขนส่งสินค้าได้อย่างทันท่วงทีเมื่อเกิดปัญหาขึ้น โดยส่วนมากแล้วเทคโนโลยี GPS เป็นเทคโนโลยีที่มีอยู่ในโทรศัพท์มือถือแล้ว ทำให้ทุกคนสามารถเข้าถึงระบบนำทาง GPS ได้ง่ายมากขึ้น แต่การนำไปใช้ก็ขึ้นอยู่กับความสามารถของซอฟแวร์และระบบ GPS ในอุปกรณ์อิเล็กทรอนิกส์นั้น ๆด้วยว่าจะมีประสิทธิภาพมากน้อยขนาดไหน

การทำงานของ GPS

เครื่องรับสัญญาณ GPS ทำงานโดยรับข้อมูล 2 อย่าง คือ เวลาและตำแหน่ง ณ ขณะที่ดาวเทียมส่งสัญญาณมายังเครื่องรับสัญญาณ แล้วนำมาประมวลผลเป็นระยะทางระหว่างเครื่องรับสัญญาณกับดาวเทียมดวงที่ส่งข้อมูลมา สามารถคำนวณได้ด้วยสูตรฟิสิกส์พื้นฐาน คือ ระยะทาง = เวลา × ความเร็ว ดังนี้

ระยะห่างระหว่างเครื่องรับสัญญาณกับดาวเทียม = (เวลาที่ได้รับสัญญาณ – เวลาที่สัญญาณถูกส่ง) × ความเร็วของคลื่นวิทยุ

หากพื้นโลกแบนราบอย่างที่คนทั้งโลกกล่าวกับโคลัมบัสเราคงใช้ดาวเทียมเพียง 3 ดวง ในการระบุตำแหน่งได้แม่นยำ แต่โลกเรากลมอย่างโคลัมบัสเชื่อ เราจึงต้องใช้ดาวเทียมอย่างน้อย 4 ดวง โดยดาวเทียมดวงที่เพิ่มเข้ามาจะใช้คำนวณความสูง เพื่อให้ได้ตำแหน่งที่แม่นยำมากขึ้น



ภาพประกอบที่ 2.10 การทำงานของ GPS (ที่มา : SCIREN, 2016)

2.7 Google Maps SDK

การทำงานกับแผนที่ใน iOS ประกอบด้วยบทการเขียนโปรแกรมทั้งเล่มเนื่องจากมีสิ่งต่าง ๆ มากมายที่นักพัฒนาซอฟต์แวร์สามารถทำกับพวกเขาได้ จากการนำเสนอตำแหน่งบนแผนที่เพื่อวาดเส้นทางการเดินทางด้วยตำแหน่งระดับกลางหรือแม้แต่ใช้ประโยชน์จากความเป็นไปได้ของแผนที่ในรูปแบบที่แตกต่างไปจากเดิมอย่างสิ้นเชิงการจัดการกับสิ่งเหล่านี้คือประสบการณ์ที่ยอดเยี่ยมที่นำไปสู่ผลลัพธ์ที่น่าอัศจรรย์ iOS ใช้ บริการ Google Mobile Maps เพื่อให้เข้าถึงแผนที่และบริการที่เกี่ยวข้องทั้งหมด ตั้งแต่นั้นเป็นต้นมาสิ่งต่าง ๆ เปลี่ยนไปและแอปเปิ้ลแนะนำ Map Kit ซึ่งเป็นกรอบการทำงานใหม่เอี่ยมที่สร้างขึ้นมาใหม่ภายในบ้านซึ่งใช้มาจนถึงปัจจุบัน เมื่อ Google หยุดให้บริการแผนที่ของ Google Google ตัดสินใจที่จะสร้าง SDK Maps สำหรับทุกแพลตฟอร์มรวมถึง iOS และวิธีการแข่งขัน Map kit หรือ SDK แผนที่อื่น ๆ ที่ใช้แพลตฟอร์มอื่น ๆ ขณะนี้ Google ประกอบด้วยผู้เล่นที่แข็งแกร่งในฟิลด์นี้เนื่องจากนักพัฒนาซอฟต์แวร์หลายคนใช้ SDK นั้น ดังนั้นการเขียน Google แผนที่ SDK สำหรับ iOS เป็นสิ่งที่คุ้มค่ามากที่จะทำในช่วงเขียนของบทแนะนำนี้ Google Maps SDK สำหรับ iOS จะอยู่ในเวอร์ชั่น 1.9.2 คุณลักษณะนี้มีคุณลักษณะมากมายซึ่งเป็นสิ่งที่รวมอยู่ในแผนที่เวอร์ชั่นเว็บมากที่สุด แต่ก็มีคุณลักษณะที่ขาดหายไปซึ่งไม่สามารถทำงานบนแพลตฟอร์มมือถือได้ จุดเด่นคือในเวอร์ชั่นนี้ SDK มีขนาดค่อนข้างใหญ่ (MB) และแน่นอนว่านี่เป็นสิ่งที่คุณต้องพิจารณาหากคุณต้องการคัดลอกไฟล์ต้นฉบับของกรอบงานในโครงการของคุณ อย่างไรก็ตามคุณลักษณะที่นำเสนอมีความน่าสนใจและมีความสำคัญอย่างยิ่งเพื่อที่จะได้รับการปฏิเสธโดยไม่ได้คิดเป็นครั้งที่สอง นักพัฒนาซอฟต์แวร์มักทำเมื่อใช้กับแผนที่ ในระยะสั้นคือสิ่งที่จะเห็น

2.7.1 เอพีไอ (Application programming interface : API)

เอพีไอ (API) ย่อมาจาก Application Programming Interface คือช่องทางการเชื่อมต่อเพื่อแลกเปลี่ยนข้อมูล จากระบบหนึ่งไปสู่ระบบอื่น ๆ โดยผ่าน library (Function/module/utility) ของผู้สร้างหรือให้บริการ API เพื่อให้ผู้พัฒนาระบบอื่น ๆ สามารถเชื่อมต่อกับ API ของผู้ที่เปิดให้บริการได้ API ช่วยให้เราสามารถพัฒนาระบบได้ง่ายและรวดเร็วขึ้น เป็นระบบมากขึ้น โดยที่เราไม่ต้องเข้าใจหรือไปแตะต้องโค้ดของ API แค่รู้ว่าต้องทำงานยังไง เรียกใช้ยังไง ส่งค่าอะไร และได้รับค่าอะไรกลับมาก็พอสำหรับ Web Developer หรือนักพัฒนาเว็บไซต์ API ก็คือรูปแบบคำสั่งที่นักพัฒนาจะต้องเรียกใช้เมื่อต้องการเข้าถึงข้อมูลบนเว็บไซต์ที่มีการเปิด API ไว้ให้ การเข้าถึงข้อมูลอาจเป็นทั้งการนำข้อมูลออกมา หรือเป็นการส่งข้อมูลเข้าไปก็ได้ API จึงเป็นเสมือนภาษาที่คอมพิวเตอร์ใช้คุยกับคอมพิวเตอร์เพื่อแลกเปลี่ยนข้อมูลกัน ระหว่างเครือข่าย (Server) กับผู้ใช้ (Client) หรือ Server กับ Server ด้วย

ประโยชน์ของ API

-สามารถรับส่งหรือเชื่อมต่อเพื่อแลกเปลี่ยนข้อมูลข้าม Server ได้
-ไม่จาเป็นต้องเข้าหน้าเว็บหลัก ก็มีข้อมูลของเว็บหลัก จากเว็บที่ดึง API
-ทำให้ผู้พัฒนาระบบสามารถพัฒนาได้ง่ายและรวดเร็วขึ้น

ประเภทของ API

-เอพีไอที่ขึ้นกับภาษา (language-dependent API) คือ เอพีไอ ที่สามารถการเรียกใช้จากโปรแกรมที่เขียนขึ้นด้วยภาษาเพียงภาษาใดภาษาหนึ่ง
-เอพีไอไม่ขึ้นกับภาษา (language-independent API) คือ เอพีไอ ที่สามารถเรียกได้จากโปรแกรมหลายๆภาษา

Google Maps API

Google Maps API คือบริการของ Google อีกรูปแบบหนึ่ง เราสามารถนำข้อมูลของ Google Maps มาแสดงในหน้าเว็บเพจของเราตามที่เราต้องการ เช่น สามารถกำหนดตำแหน่งที่ตั้ง, สถานที่, ที่นัดหมาย เป็นต้น โดยเราสามารเรียกใช้ข้อมูลและ Method ต่าง ๆที่ Google ได้จัดเตรียมไว้ในแล้วหรือที่เราเรียกว่า API (Application Programming Interface)

Google Maps API เป็นชุด API ของ Google สำหรับพัฒนา web application และ mobile application (Android, iOS) ไว้สำหรับเรียกใช้แผนที่และชุด service ต่าง ๆ ของ Google เพื่อพัฒนา Application ได้เหมือนกับที่ Google โดยแผนที่ยัง features ต่าง ๆ มากมายให้เรียกใช้

– การปรับแต่งแผนที่ (Styled Map)

– ชุดควบคุมแผนที่ (Map Control)

– ชุดเครื่องมือวาดภาพบนแผนที่ (Drawing)

– การนำทางจากจุดหนึ่งไปยังอีกจุดหนึ่ง (Directions Service)

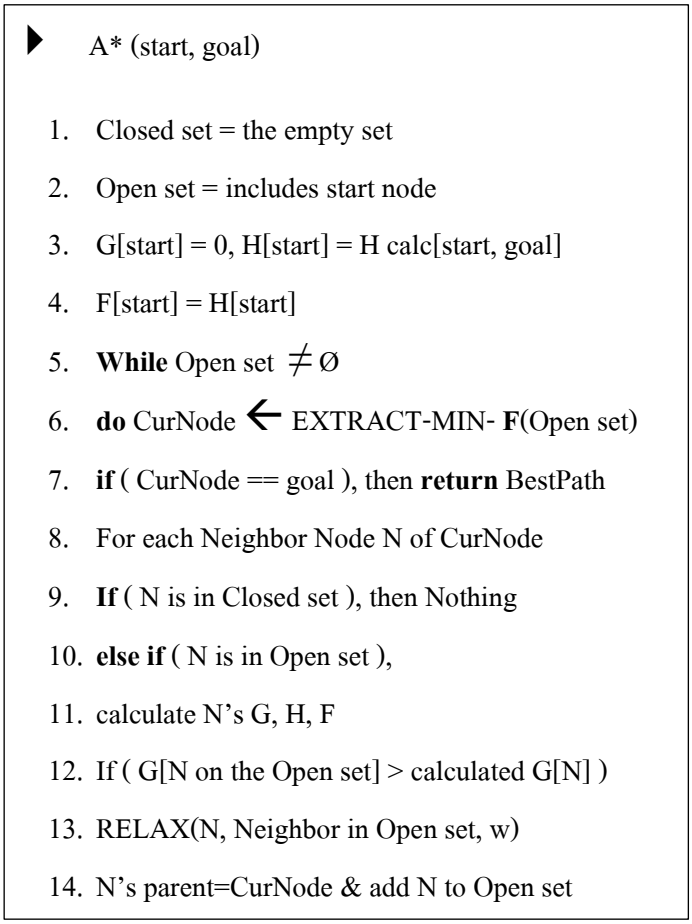
– การคำนวณความสูงของจุดพิกัด (Elevation Service)

– การแปลงที่อยู่เป็นพิกัด Lattitude และ Longtitude (GeoCoding Service)

– การดึงข้อมูล POI (Point of Interest) คือข้อมูลสถานที่ต่าง ๆ ที่ Google รวบรวมไว้ให้ เช่น โรงแรม ห้างสรรพสินค้า โรงเรียน -สถานที่ราชการต่าง ๆ และอื่น ๆ อีกมากมาย (Places API) มาใช้งานใน application

2.8 อัลกอริทึม (Algorithm)

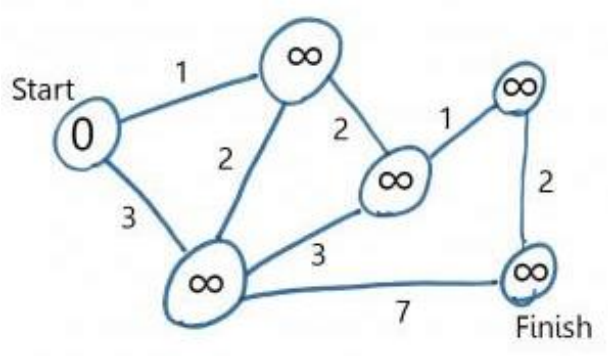
กระบวนการแก้ปัญหาที่สามารถเข้าใจได้ มีลำดับหรือวิธีการในการแก้ไขปัญหาใดปัญหาหนึ่งอย่างเป็นขั้นเป็นตอนและชัดเจน เมื่อนำเข้าอะไร แล้วจะต้องได้ผลลัพธ์เช่นไร ซึ่งแตกต่างจากการแก้ปัญหาแบบสามัญสำนึก หรือฮิวริสติก (heuristic)โดยทั่วไป ขั้นตอนวิธี จะประกอบด้วยวิธีการเป็นขั้นๆ และมีส่วนที่ต้องทำแบบวนซ้ำ (iterate) หรือ เวียนเกิด (recursive)โดยใช้ตรรกะ (logic) และ/หรือ ในการเปรียบเทียบ (comparison) ในขั้นตอนต่าง ๆ จนกระทั่งเสร็จสิ้นการทำงาน ในการทำงานอย่างเดียวกัน เราอาจจะเลือกขั้นตอนวิธีที่ต่างกันเพื่อแก้ปัญหาได้ โดยที่ผลลัพธ์ที่ได้ในขั้นสุดท้ายจะออกมาเหมือนกันหรือไม่ก็ได้ และจะมีความแตกต่าง ที่จำนวนและชุดคำสั่งที่ใช้ต่างกันซึ่งส่งผลให้ เวลา (time) , และขนาดหน่วยความจำ (space) ที่ต้องการต่างกัน หรือเรียกได้อีกอย่างว่ามีความซับซ้อน (complexity) ต่างกันการนำขั้นตอนวิธีไปใช้ ไม่จำกัดเฉพาะการเขียนโปรแกรมคอมพิวเตอร์ แต่สามารถใช้กับปัญหาอื่น ๆ ได้เช่น การออกแบบวงจรไฟฟ้า, การทำงานเครื่องจักรกล, หรือแม้กระทั่งปัญหาในธรรมชาติ เช่น วิธีของสมองมนุษย์ในการคิดเลข หรือวิธีการขนอาหารของแมลง



ภาพประกอบที่ 2.11 แสดง Pseudo Code อัลกอริทึม แบบ เอ สตาร์
(ที่มา : สุขสวัสดี ณัฏฐวุฒิสิทธิ์, ประสงค์ ปราณีตพลกรัง, 2011)

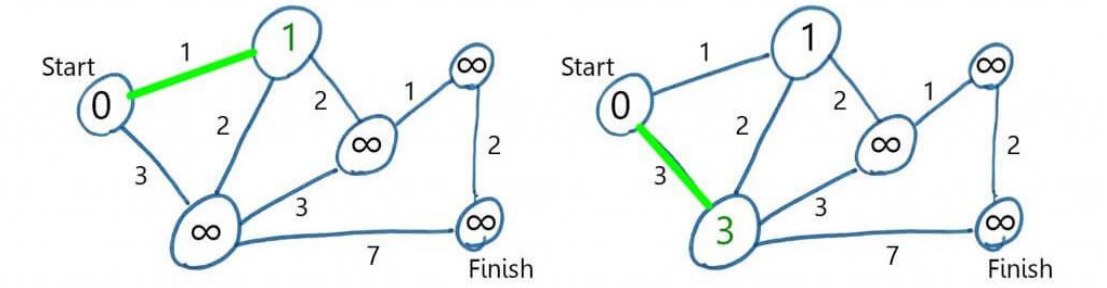
จากภาพประกอบที่ 2.10 อัลกอริทึมแบบ เอ สตาร์ มีความเหมาะสมสำหรับนำมาใช้พัฒนาเป็นระบบค้นหาเส้นทาง เนื่องจากการทำงานในขั้นตอนของอัลกอริทึมมีส่วนช่วยลดระยะเวลาในการประมวลผลและขั้นตอนการค้นหาเส้นทางของอัลกอริทึมไม่มีการวนลูปที่ซ้ำซ้อน จึงเป็นที่นิยมนำมาใช้ประยุกต์กับระบบค้นหาเส้นทางของการคมนาคมขนส่ง ตัวอย่างขั้นตอนวิธีการหาทางเดินสั้นที่สุด

1. กำหนดจุดเริ่มต้น (ตำแหน่งปัจจุบัน) เป็น 0 และจุดอื่น ๆ เป็น infinity หรือไม่กำหนด



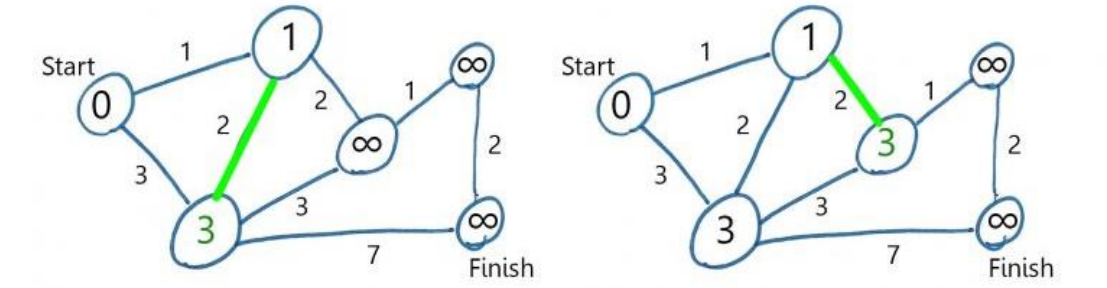
ภาพประกอบที่ 2.12 กำหนดจุดเริ่มต้น (ที่มา : SCIREN, 2016)

2. เลือกเส้นทางที่ออกจากจุดเริ่มต้นไปจุดถัดไป แล้วบันทึกระยะทางที่เดินบนจุดนั้น



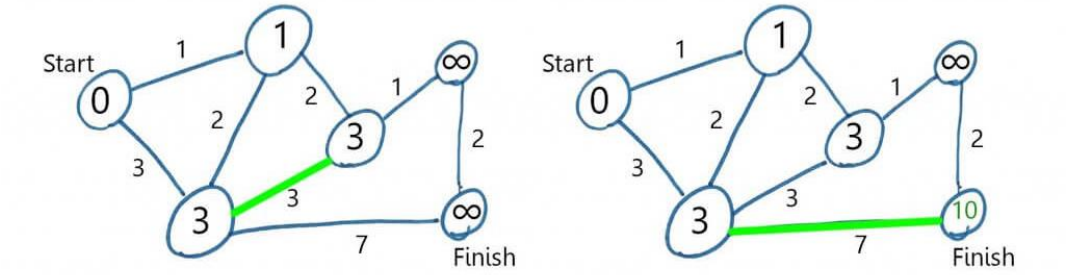
ภาพประกอบที่ 2.13 เส้นทางที่ออกจากจุดเริ่มต้นไปจุดถัดไป (ที่มา : SCIREN, 2016)

3. เลือกเส้นทางที่ออกจากตำแหน่งปัจจุบัน (จุด 1) ไปจุดถัดไป แล้วบันทึกระยะทางที่เดินรวมกับค่าบนจุดที่เดินออกมาไว้บนจุดถัดไป



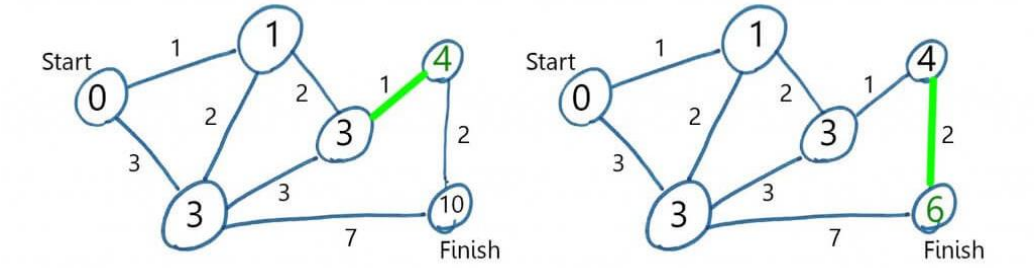
ภาพประกอบที่ 2.14 เส้นทางที่ออกจากตำแหน่งปัจจุบัน (จุด 1) ไปจุดถัดไป
(ที่มา : SCIREN, 2016)

4. เลือกเส้นทางที่ออกจากตำแหน่งปัจจุบัน (จุด 3) ไปจุดถัดไป แล้วบันทึกระยะทางที่เดินรวมกับค่าบนจุดที่เดินออกมาไว้บนจุดถัดไป



ภาพประกอบที่ 2.15 เส้นทางที่ออกจากตำแหน่งปัจจุบัน (จุด 3) ไปจุดถัดไป
(ที่มา : SCIREN, 2016)

5. เลือกเส้นทางที่ออกจากตำแหน่งปัจจุบัน(จุด 3 และ 4 ตามลำดับ) ไปจุดถัดไป แล้วบันทึกระยะทางที่เดินรวมกับค่าบนจุดที่เดินออกมาไว้บนจุดถัดไป



ภาพประกอบที่ 2.16 เส้นทางที่ออกจากตำแหน่งปัจจุบัน(จุด 3 และ 4 ตามลำดับ) ไปจุดถัดไป
(ที่มา : SCIREN, 2016)

6. เมื่อถึงจุดปลายทางครบทุกรูปแบบ จะได้ระยะทางสั้นที่สุดจากการใช้เส้นเชื่อมที่ทำให้ค่าบนจุดปลายทางมีค่าน้อยที่สุด ซึ่งกรณีนี้มี 2 เส้นทางที่สั้นที่สุดในการเดินจากจุดเริ่มต้น ไปยังจุดปลายทาง ด้วยระยะทาง 6 หน่วย

เอ สตาร์ A* (A Star) เป็นอัลกอริทึมที่ใช้ในการหาเส้นทางระหว่างตำแหนง่ สองตำแหน่งบนแผนที่ ซึ่งก็มีอัลกอริทึมอื่น ๆ ที่ใช้ในการหาเส้นทางได้เหมือนกัน แต่ A* จะหาเส้นทางที่สั้นที่สุดให้ (ในกรณีที่ มีเส้นทางที่สั้นที่สุด) ในเวลาที่รวดเร็ว A* เป็น อัลกอริทึมแบบ Directed นั่นหมายความว่าจะไม่มีการค้นหาแบบไม่รู้ทิศทาง แต่จะมีการกำหนดค่าในการค้นหาเส้นทางที่ดีที่สุดแทน และบางครั้งก็มีการย้อนกลับมาหาเส้นทางเก่าที่หามาแล้ว ซึ่งทำให้การค้นหาเส้นทางโดยใช้ A* มีความยืดหยุ่นสูง

ในการศึกษาเกี่ยวกับ A* นั้นมีสิ่งจำเป็นที่จะต้องทำความเข้าใจดัง นี้

Map (หรือ Graph) ซึ่งเป็นพื้นที่ที่ A* ใช้สำหรับการค้นหาเส้นทางระหว่างสองตำแหน่ง อาจจะประกอบด้วยรูปสี่เหลี่ยมหรือ หกเหลี่ยม หรืออาจจะเป็นพื้นที่ในระบบพิกัด 3 มิติแต่อย่างไร ก็ตามก็คือพื้นที่ที่ A* ใช้ในการทำงานนั่นเอง

Nodes คือโครงสร้างที่แทนตำแหน่งใน Map ซึ่งโนดนี้จะเก็บข้อมูลต่าง ๆ ที่สำคัญสำหรับ A* เช่น ข้อมูลตำแหน่ง ดังนั้นโนดจะทำหน้าที่เหมือนเป็นที่ที่เก็บความก้าวหน้าของการค้นหา

Distance (หรือ Heuristic) เป็นส่วนที่จะตัดสินใจว่าจะทำการค้นหาไปที่โนดใด

Cost เป็นค่าปัจจัยอื่น ๆ ที่จะต้องนำมาพิจารณา เช่น เวลาค่าใช้จ่าย หรือ ค่าความสิ้นเปลือง อื่น ๆ

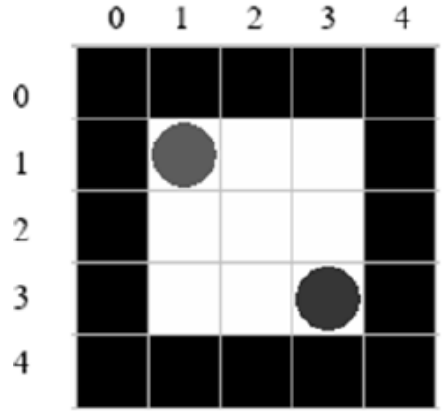
นอกจากนี้ในการค้นหาด้วย A * ยังจะต้องมีค่าคุณลักษณะ 3 อย่างคือ ค่า g(n) h(n) และ f(n) ซึ่งมีรายละเอียดดังนี้

g(n) เป็นค่าที่คิดจากโนดเริ่มต้น (Start state) ถึงโนดปัจจุบัน โดยอาจจะมีหลายเส้นทางที่เริ่มจากโนดเริ่มต้นจนถึงตำแหน่งใน Map ปัจจุบัน

h(n) คือค่าประมาณจากโนดปัจจุบันถึงโนดเป้าหมาย (Goal state)

f(n) เป็นค่าผลรวมของ g และ h ซึ่งจะเป็นตัวบอกว่าจะไปที่โนดใดต่อไป โดยจะเลือกเอาโนดที่มีค่า f(n) น้อยที่สุด

นอกจากนี้ A* ยังเก็บสถิติข้อมูลสถานะของโนดไว้ 2 ประเภท คือ Open list และ Close listโดย Open list ประกอบด้วยโนดที่ยังไม่ได้ทำการค้นหา ส่วน Close list ประกอบด้วยโนดที่เคยค้นหาไปแล้ว ดังตัวอย่างต่อไปนี้



ภาพประกอบที่ 2.17 Map ที่ใช้ในการศึกษา A* อัลกอริทึม

(ที่มา :สัญญา เครือหงษ์ และ สุรางค์รัตน์ เชาว์โคกสูง, 2551)

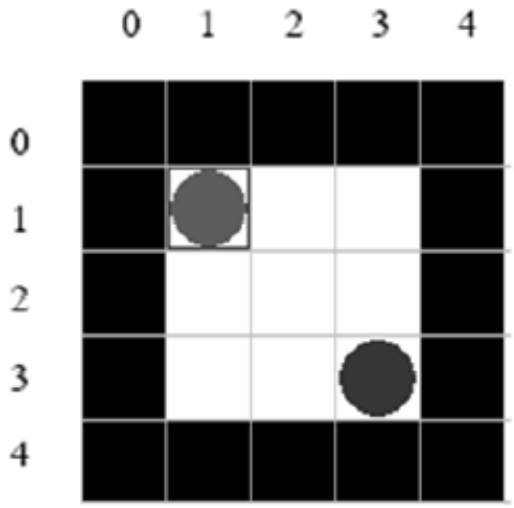
กำหนดให้ โนดเริ่มต้น พิกัด (1,1) โนดเป้าหมาย พิกัด (3,3) g(n) หาได้จากระยะที่โนดปัจจุบันห่างจากโนด เริ่มต้น

h(n) หาได้จาก h(n) = | dx - sx | + | dy - sy | เมื่อ

(dx,dy) เป็นตำแหน่งเป้าหมาย และ

(sx,sy) เป็นตำแหน่งเริ่มต้น

f(n) หาได้จาก f(n) = g(n) + h(n)



ภาพประกอบที่ 2.18 ตัวอย่างการหาเส้นทางโดยใช้ A* อัลกอริทึม ขั้นที่ 1

(ที่มา : สัญญา เครือหงษ์ และ สุรางค์รัตน์ เชาว์โคกสูง, 2551)

เริ่มแรกเก็บโนด (1,1) เข้า Open list แล้วทำการคำนวณหาค่า f(1,1) จะได้

g(1,1) = 0

h(1,1) = | 3 – 1 | + | 3 – 1 | = 2 + 2 = 4

f(1,1) = 0+4 = 4

เลือกโนด (1,1) เป็นโนดที่ดีที่สุด (มีโนดเดียว)แล้วดูโนดต่อไปที่ติดกับโนด (1,1) นั้นคือ โนด (2,1) (2,2) และ (1,2) จากนั้นนำเข้า Open list

g(2,1) = 1

h(2,1) = | 3 - 2 | + | 3 – 1 | = 3

f(2,1) = 1 + 3 = 4

g(2,2) = 1

h(2,2) = | 3 - 2 | + | 3 – 2 | = 2

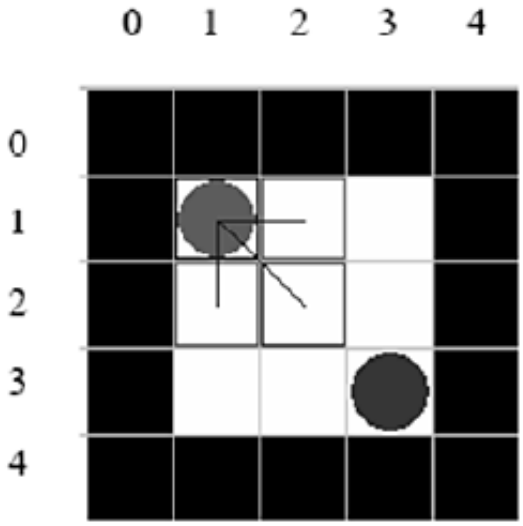
f(2,2) = 1 + 2 = 3

g(1,2) = 1

h(1,2) = | 3 - 1 | + | 3 – 2 | = 3

f(1,2) = 1 + 3 = 4

จากค่า f(n) ใน Open list จะเลือกโนดที่มีค่า f(n) น้อยสุดมาพิจารณา ในตอนนี้จะต้องเลือกโนด (2,2) เพราะมีค่า f(n) น้อยที่สุด จากนั้นย้ายโนด (1,1) จาก Openlist เข้า Close list



ภาพประกอบที่ 2.19 ตัวอย่างการหาเส้นทางโดยใช้ A* อัลกอริทึม ขั้นที่ 2

(ที่มา : สัญญา เครือหงษ์ และ สุรางค์รัตน์ เชาว์โคกสูง, 2551)

แล้วดูโนดต่อไปที่ติดกับโนด (2,2) นั่นคือโนด (3,1) (3,2) (3,3) (2,3) (1,3) และทำการคำนวณหาค่า f(n) แล้วจึงเก็บเข้า Open list

g(3,1) = 2

h(3,1) = | 3 - 3 | + | 3 – 1 | = 2

f(3,1) = 2 + 2 = 4

g(3,2) = 2

h(3,2) = | 3 - 3 | + | 3 – 2 | = 1

f(3,2) = 2 + 1 = 3

g(3,2) = 2

h(3,2) = | 3 - 3 | + | 3 – 2 | = 1

f(3,2) = 2 + 1 = 3

g(3,3) = 2

h(3,3) = | 3 - 3 | + | 3 – 3 | = 0

f(3,3) = 2 + 0 = 2

g(2,3) = 2

h(2,3) = | 3 - 2 | + | 3 – 3 | = 1

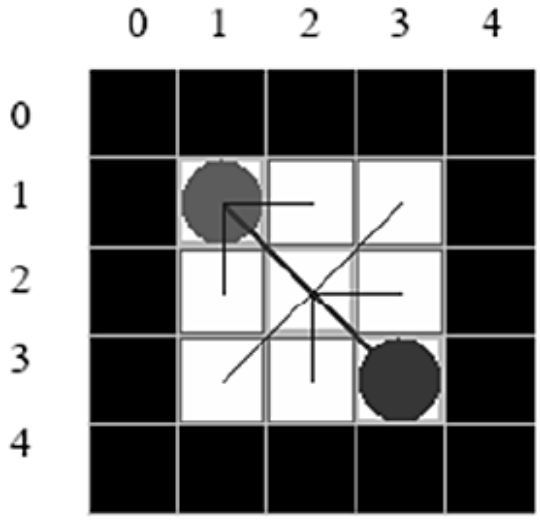
f(2,3) = 2 + 1 = 3

g(1,3) = 2

h(1,3) = | 3 - 1 | + | 3 – 3 | = 2

f(1,3) = 2 + 2 = 4

จากค่า f(n) ที่อยู่ใน Open list จะได้ว่าโนด (3,3) จะถูกเลือกเพราะมีค่า f(n) น้อยที่สุด และโนด (3,3) ก็เป็นโนดเป้าหมายด้วย จะได้เส้นทางจากตำแหน่งเริ่มต้นถึงตำแหน่งเป้าหมายตามโนดดังนี้คือ (1,1) (2,2) และ (3,3)



ภาพประกอบที่ 2.20 ตัวอย่างการหาเส้นทางโดยใช้ A* อัลกอริทึม ขั้นที่ 3

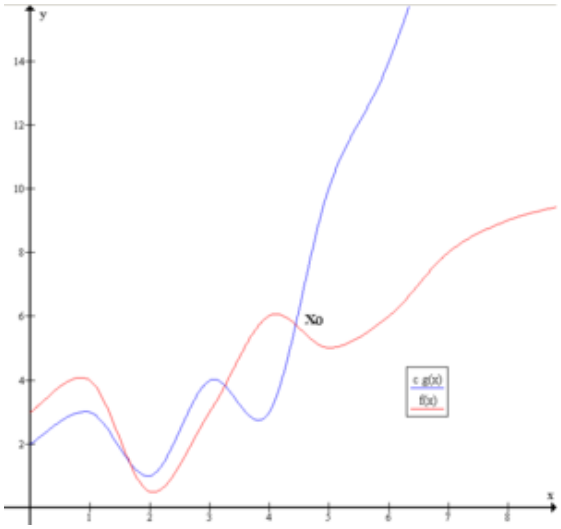
(ที่มา : สัญญา เครือหงษ์ และ สุรางค์รัตน์ เชาว์โคกสูง, 2551)

2.9 สัญกรณ์บิ๊กโอ (Big O notation)

สัญกรณ์ O Big เป็น สัญกรณ์ ทางคณิตศาสตร์ที่อธิบาย พฤติกรรมการ จำกัด การ ทำงาน เมื่ออาร์กิวเมนต์มีแนวโน้มที่จะมีต่อค่าเฉพาะหรืออินฟินิตี้ มันเป็นสมาชิกคนหนึ่งของครอบครัวของสัญกรณ์ประดิษฐ์โดย พอลแบชแมนน์ เอ๊ดมันด์รถม้า และอื่น ๆ รวมเรียกว่า แบชแมนน์ - กุ๊บสัญกรณ์ หรือ เครื่องหมายสัณฐานใน วิทยาการคอมพิวเตอร์ โน้ตขนาดใหญ่ O ถูกใช้ในการจำแนกอัลกอริทึม ตามระยะเวลาในการทำงานหรือความต้องการพื้นที่ของพวกเขาเมื่อขนาดของ

อินพุตเติบโตขึ้น ใน ทฤษฎีจำนวนวิเคราะห์ สัญกรณ์ใหญ่ O มักจะใช้ในการแสดงผูกพันเกี่ยวกับความแตกต่างระหว่าง ฟังก์ชั่นเลขคณิต และเข้าใจได้ดีขึ้น; ตัวอย่างที่มีชื่อเสียงของความแตกต่างดังกล่าวคือส่วนที่เหลือใน ทฤษฎีบทตัวเลขที่สำคัญ สัญกรณ์บิ๊ก O ทำหน้าที่ characterizes ตามอัตราการเติบโตของพวกเขาฟังก์ชั่นที่แตกต่างกันที่มีอัตราการเติบโตเดียวกันอาจจะแสดงโดยใช้สัญกรณ์ O เดียวกัน

ตัวอักษร O ถูกใช้เนื่องจากอัตราการเติบโตของฟังก์ชันเรียกว่าเป็น ลำดับของฟังก์ชัน คำอธิบายของฟังก์ชันในรูปแบบของสัญกรณ์โอ้โฮจะให้ ขอบเขต บนอัตราการเติบโตของฟังก์ชันเท่านั้น ที่เกี่ยวข้องกับสัญกรณ์ O ใหญ่เป็นสัญกรณ์ที่เกี่ยวข้องหลายอย่างโดยใช้สัญลักษณ์ o , Ω, ωและΘ เพื่ออธิบายขอบเขตอื่น ๆ ที่มีต่ออัตราการเติบโตของ asymptotic



ภาพประกอบที่ 2.21 ตัวอย่างของสัญกรณ์บิ๊ก (ที่มา : JSO Software)

สัญกรณ์ Big O มีสองส่วนหลักของโปรแกรม:

ใน คณิตศาสตร์ มันเป็นเรื่องธรรมดาที่ใช้เพื่ออธิบาย ว่าใกล้เคียงกับชุด จำกัด approximates กำหนดฟังก์ชัน โดยเฉพาะอย่างยิ่งในกรณีของ เทย์เลอร์ ตัด ชุด หรือ การขยายตัวของอาณานิคมใน วิทยาการคอมพิวเตอร์ จะมีประโยชน์

ในการ วิเคราะห์อัลกอริทึมในทั้งสองโปรแกรมฟังก์ชัน g (x) ที่ปรากฏอยู่ภายใน O (...) จะถูกเลือกให้เรียบง่ายที่สุดเท่าที่จะเป็นไปได้โดยละเว้นปัจจัยคงที่และเงื่อนไขการสั่งซื้อที่ต่ำกว่ามีสองอย่างเป็นทางการปิด แต่แตกต่างกันอย่างเห็นได้ชัดประเพณีของสัญกรณ์นี้:

asymptotics อนันต์

asymptotics ไม่มีที่สิ้นสุด

ความแตกต่างนี้มีเฉพาะในแอพพลิเคชั่นและไม่ได้อยู่ในหลักการอย่างไรก็ตามคำจำกัดความที่เป็นทางการสำหรับ "โอ้โอ" ก็เหมือนกันสำหรับทั้งสองกรณีเท่านั้นโดยมีข้อ จำกัด ที่แตกต่างกันสำหรับอาร์กิวเมนต์ของฟังก์ชัน

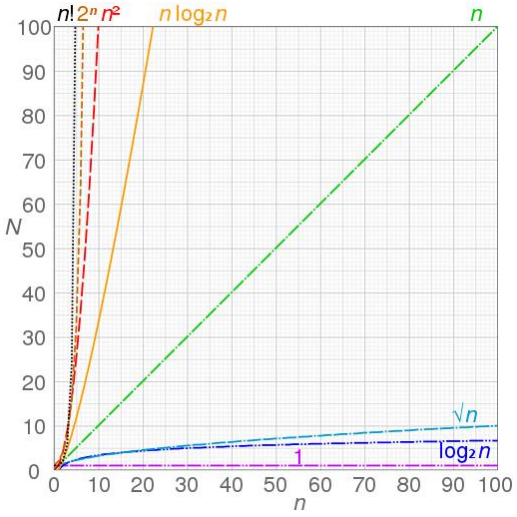
สัญกรณ์ Big O มีประโยชน์เมื่อ วิเคราะห์อัลกอริทึม อย่างมีประสิทธิภาพ ตัวอย่างเช่นเวลา (หรือจำนวนขั้นตอน) ที่ใช้ในการแก้ไขปัญหาของขนาด n อาจพบได้เท่ากับ T (n) = 4 n 2 - 2 n + 2. เนื่องจาก n เติบโตขึ้นขนาด n 2 term will มาครอบงำเพื่อให้ทุกคำอื่น ๆ สามารถละเลยเช่นเมื่อ n = 500, คำ 4 n 2 เป็น 1000 ครั้งใหญ่เป็นระยะ 2 n ละเว้นหลังจะมีผลเล็กน้อยต่อค่าของนิพจน์สำหรับวัตถุประสงค์มากที่สุด นอกจากนี้ ค่าสัมประสิทธิ์ จะไม่เกี่ยวข้องถ้าเราเปรียบเทียบกับลำดับ การแสดงออกอื่น ๆ เช่นนิพจน์ที่ประกอบด้วยคำ n 3 หรือ 4 ถึงแม้ว่า T (n) = 1,000,000 n 2 ถ้า U (n) = n3 ค่าที่ได้จะมากกว่าอดีตเมื่อ n เติบโตขึ้นกว่า 1,000,000 (T (1,000,000) = 1,000,000 3 = U (1,000,000)) นอกจากนี้จำนวนขั้นตอนขึ้นอยู่กับรายละเอียดของรูปแบบเครื่องที่อัลกอริทึมทำงาน แต่เครื่องประเภทต่าง ๆ มักจะแตกต่างกันไปตามปัจจัยคงที่เพียงอย่างเดียวในจำนวนขั้นตอนที่จำเป็นในการประมวลผลอัลกอริทึม ดังนั้นสัญกรณ์ O ใหญ่จับสิ่งที่เหลืออยู่: เราเขียนอย่างใดอย่างหนึ่ง

{\ displaystyle \ T (n) = O (n ^ {2})} {\ displaystyle \ T (n) = O (n ^ {2})}

หรือ

{\ displaystyle T (n) \ in O (n ^ {2})} {\ displaystyle T (n) \ in O (n ^ {2})}

และบอกว่าอัลกอริทึมมี ลำดับของ ความซับซ้อน ของ n 2 ครั้ง โปรดทราบว่า "=" ไม่ได้หมายถึงการแสดงออก "เท่ากับ" ในความหมายทางคณิตศาสตร์ตามปกติ แต่จะใช้คำว่า "เป็น" มากขึ้นดังนั้นการแสดงออกที่สองถือเป็นความถูกต้องมากยิ่งขึ้น (ดูการอภิปราย " Equals sign " ด้านล่าง) ในขณะที่ คนแรกถือว่าเป็นการ ละเมิดสัญญา



ภาพประกอบที่ 2.22 กราฟของฟังก์ชันที่ใช้กันโดยทั่วไปในการวิเคราะห์อัลกอริทึม
(ที่มา : Wikipedia)

ใช้ในวิทยาการคอมพิวเตอร์

ข้อมูลเพิ่มเติม: การ วิเคราะห์อัลกอริทึม อย่างไม่เป็นทางการโดยเฉพาะอย่างยิ่งในด้านวิทยาการคอมพิวเตอร์สัญกรณ์บิ๊ก O มักได้รับอนุญาตให้ทำร้ายค่อนข้างเพื่ออธิบายขอบเขตแบบจำกัด ที่ใช้ Big Theta สัญกรณ์อาจมีความเหมาะสมมากขึ้นตามความเป็นจริงในบริบทที่กำหนด ตัวอย่างเช่นเมื่อพิจารณาถึงฟังก์ชัน T (n) = 73 n 3 + 22 n 2 + 58 ทั้งหมดต่อไปนี้เป็นที่ยอมรับโดยทั่วไป แต่ขอบเขตที่เข้มงวดมากขึ้น (เช่นตัวเลข 2 และ 3 ด้านล่าง) เป็นที่ต้องการอย่างมากเหนือขอบเขตที่คลาย (กล่าวคือหมายเลข 1 ด้านล่าง)

T (n) = O (n 100)

T (n) = O (n 3)

T (n) = Θ (n 3)

งบภาษาอังกฤษที่เท่ากันมีดังนี้:

T (n) เติบโตได้ไม่เร็วกว่า n 100

T (n) เติบโตขึ้นอย่างรวดเร็วไม่เกิน n 3

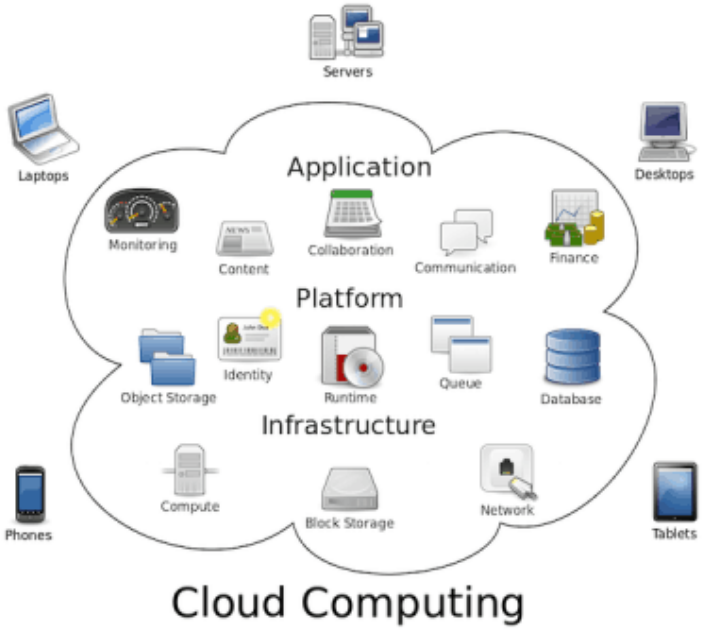
T (n) เติบโต asymptotically เร็วที่สุดเท่าที่ n 3

ดังนั้นในขณะที่ทั้งสามคำนี้เป็นความจริงความก้าวหน้าข้อมูลเพิ่มเติมมีอยู่ในแต่ละ ในบางสาขาอย่างไรก็ดีสัญกรณ์ O ใหญ่ (หมายเลข 2 ในรายการด้านบน) จะถูกนำมาใช้โดยทั่วไปมากกว่าสัญกรณ์ Big Theta (สัญลักษณ์แสดงหัวข้อย่อยที่ 3 ในรายการด้านบน) ตัวอย่างเช่นถ้า T (n) หมายถึงเวลาทำงานของอัลกอริทึมที่พัฒนาขึ้นใหม่สำหรับขนาดอินพุท n นักประดิษฐ์และผู้ใช้

อัลกอริทึมอาจเอนเอียงที่จะใส่ค่า asymptotic ขึ้นไปว่าจะใช้เวลานานเท่าใด คำจำกัดความที่ชัดเจนเกี่ยวกับขอบเขตที่น้อยลง

2.10 คลาวด์คอมพิวติ้ง (Cloud Computing)

Cloud computing หมายถึง ระบบการประมวลผลแบบกลุ่มเมฆ ซึ่งเป็นลักษณะของการให้บริการรูปแบบหนึ่งแก่ผู้ใช้บริการคอมพิวเตอร์ผ่านระบบเครื่องข่ายอินเตอร์เน็ตโดยผู้ให้บริการจะทาการจัดสรรทรัพยากรที่มีอยู่ให้ผู้ที่ต้องการใช้งาน ซึ่งระบบประมวลผลแบบกลุ่มเมฆนั้น เป็นลักษณะการให้บริการที่พัฒนาขึ้นมาจากแนวคิดการให้บริการการทางานแบบเสมือน (Virtualization) โดยผู้ใช้บริการไม่จาเป็นต้องรับรู้หรือเข้าใจการทางานเชิงเทคนิคของโครงสร้างพื้นฐาน และ หลักการทางานของระบบประมวลผลแบบกลุ่มเมฆ



ภาพประกอบที่ 2.23 ระบบการประมวลผลแบบกลุ่มเมฆ
(ที่มา : กลุ่มภาระงานด้านเทคโนโลยีสารสนเทศ คณะวิทยาศาสตร์ มหาวิทยาลัยขอนแก่น)

จากภาพด้านบนนี้ จะเห็นว่าด้านในของกรอบที่เป็นก้อนเมฆก็คือทรัพยากรของผู้ให้บริการที่มีทั้ง Hardware และ Software (ซึ่งก็ทำงานบน Hardware ของผู้ให้บริการเช่นกัน) ผู้ใช้บริการเพียงแค่ต่อเชื่อมเข้าไปใช้ผ่าน Network ด้วยเว็บบราวเซอร์ หรือ Client แอพพลิเคชั่น บนอุปกรณ์ต่าง ๆ ของตน เช่น มือถือ, Tablet, Notebook, หรือ Chromebook เป็นต้น

ประเภทของบริการ คลาวด์คอมพิวติ้ง (Cloud Service Models) บริการ Cloud Computing **มีหลากหลายรูปแบบ 3 แบบได้แก่**

Software as a Service (SaaS)

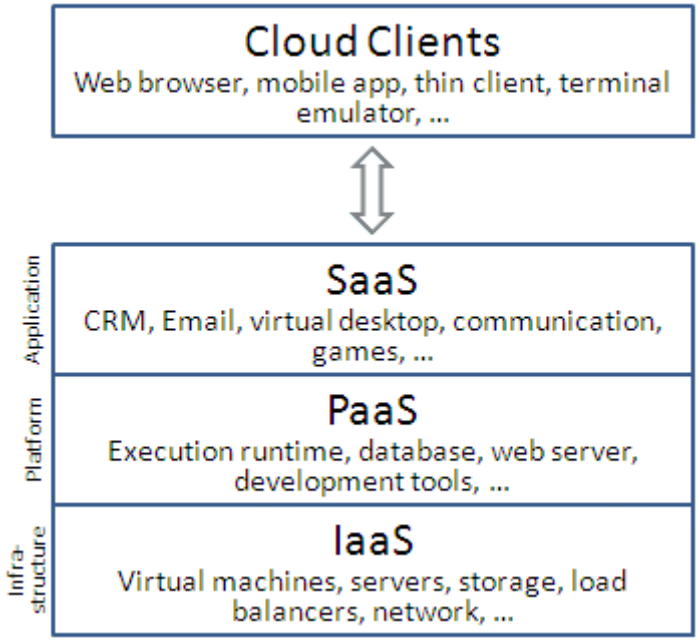
เป็นการที่ใช้หรือเช่าใช้บริการซอฟต์แวร์หรือแอพพลิเคชั่น ผ่านอินเทอร์เน็ต โดยประมวลผลบนระบบของผู้ให้บริการ ทำให้ไม่ต้องลงทุนในการสร้างระบบคอมพิวเตอร์ ฮาร์ดแวร์ ซอฟต์แวร์เอง ไม่ต้องพะวงเรื่องค่าใช้จ่ายในการดูแลระบบ เพราะซอฟต์แวร์จะถูกเรียกใช้งานผ่าน **Cloud** จากที่ไหนก็ได้ ซึ่งบริการ **Software as a Service** ที่ใกล้ตัวเรามากที่สุดก็คือ GMail นั่นเอง นอกจากนั้นก็เช่น Google Docs หรือ Google Apps ที่เป็นรูปแบบของการใช้งานซอฟต์แวร์ผ่านเว็บบราวเซอร์ สามารถใช้งานเอกสาร คำนวณ และสร้าง Presentation โดยไม่ต้องติดตั้งซอฟต์แวร์บนเครื่องเลย แถมใช้งานบนเครื่องไหนก็ได้ ที่ไหนก็ได้ แชร์งานร่วมกันกับผู้อื่นก็สะดวก ซึ่งการประมวลผลจะทำบน Server ของ Google ทำให้เราไม่ต้องการเครื่องที่มีกำลังประมวลผลสูงหรือพื้นที่เก็บข้อมูลมาก ๆ ในการทำงาน Chromebook ราคาประหยัดซักเครื่องก็ทำงานได้แล้ว มหาวิทยาลัยทั้งในไทยและต่างประเทศหลายแห่งในปัจจุบัน ก็ยกเลิกการตั้ง Mail Server สำหรับใช้งาน e-mail ของบุคลากร และนักศึกษาในมหาวิทยาลัยกันแล้ว แต่หันมาใช้บริการอย่าง Google Apps แทน เป็นการลดต้นทุน, ภาระในการดูแล, และความยุ่งยากไปได้มาก

Platform as a Service (PaaS)

สำหรับการพัฒนาแอพพลิเคชั่นนั้น หากเราต้องการพัฒนาเว็บแอพพลิเคชั่นที่ค่อนข้างซับซ้อน ซึ่งรันบนเซิร์ฟเวอร์ หรือ Mobile application ที่มีการประมวลผลทำงานอยู่บนเซิร์ฟเวอร์ เราก็ต้องตั้งเซิร์ฟเวอร์ เชื่อมต่อระบบเครือข่าย และสร้างสภาพแวดล้อม เพื่อทดสอบและรันซอฟต์แวร์และแอพพลิเคชั่น เช่น ติดตั้งระบบฐานข้อมูล, Web server, Runtime, Software Library, Frameworks ต่าง ๆ เป็นต้น จากนั้นก็อาจยังต้องเขียนโค้ดอีกจำนวนมาก แต่ถ้าใช้บริการ **PaaS** ผู้ให้บริการจะเตรียมพื้นฐานต่าง ๆ เหล่านี้ไว้ให้เราต่อยอดได้เลย พื้นฐานทั้ง Hardware, Software, และชุดคำสั่ง ที่ผู้ให้บริการเตรียมไว้ให้เราต่อยอดนี้เรียกว่า Platform ซึ่งก็จะทำให้ลดต้นทุนและเวลาที่ใช้ในการพัฒนาซอฟท์แวร์อย่างมาก ตัวอย่าง เช่น Google App Engine, Microsoft Azure ที่หลายๆบริษัทนำมาใช้เพื่อลดต้นทุนและเป็นตัวช่วยในการทำงาน Application ดังๆหลายตัว เช่น **Snapchat** ก็เลือกเช่าใช้บริการ **PaaS** อย่าง Google App Engine ทำให้สามารถพัฒนาแอพที่ให้บริการคนจำนวนมหาศาลได้ โดยใช้เวลาพัฒนาไม่นาน

Infrastructure as a Service (IaaS)

เป็นบริการให้ใช้โครงสร้างพื้นฐานทางคอมพิวเตอร์อย่าง หน่วยประมวลผล ระบบจัดเก็บข้อมูล ระบบเครือข่าย ในรูปแบบระบบเสมือน (Virtualization) ข้อดีคือองค์กรไม่ต้องลงทุนสิ่งเหล่านี้เอง, ยืดหยุ่นในการปรับเปลี่ยนโครงสร้างระบบไอทีขององค์กรในทุกรูปแบบ, สามารถขยายได้ง่าย ขยายได้ทีละนิดตามความเติบโตขององค์กรก็ได้ และที่สำคัญ ลดความยุ่งยากในการดูแล เพราะหน้าที่ในการดูแล จะอยู่ที่ผู้ให้บริการ



ภาพประกอบที่ 2.24 คลาวด์คอมพิวติ้ง (Cloud Service Models)
(ที่มา : กลุ่มภาระงานด้านเทคโนโลยีสารสนเทศ คณะวิทยาศาสตร์ มหาวิทยาลัยขอนแก่น)

2.11 งานวิจัยที่เกี่ยวข้อง

การพัฒนาโมบายแอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางด้วยสวิฟท์บนคลาวด์ งานวิจัยที่เกี่ยวข้องมีรายละเอียดงานวิจัยในประเทศ 5 เรื่องดังนี้

นายพีระเดช สำรวมรัมย์ (2015) วัตถุประสงค์พัฒนาโปรแกรมแสดงพิกัดและค้นหาผู้ป่วยฉุกเฉินเพื่อใช้ในสมาร์ทโฟน (Smartphone) ที่ใช้ระบบปฏิบัติการแอนดรอยด์ Android Operating System ซึ่งพัฒนางานวิจัยนี้เป็นการพัฒนาระบบสำหรับรองรับการใช้งานโทรศัพท์แบบสมาร์ทโฟน (Smartphone) ที่ใช้ระบบปฏิบัติการ แอนดรอยด์ (Android Operating System) และอุปกรณ์รับสัญญาณดาวเทียมจีพีเอส (GPS Satellite) ในโทรศัพท์ เพื่อระบุพิกัดตำแหน่งของอุปกรณ์ ณ

เวลาปัจจุบัน ซึ่งการใช้โทรศัพท์แบบสมาร์ทโฟนรับสัญญาณจากดาวเทียมจีพีเอสและส่งสัญญาณผ่านโครงข่ายโทรศัพท์เคลื่อนที่ระบบ 3G ในงานวิจัยนี้ จะมุ่งเน้นไปที่การเขียนโปรแกรมเพื่อนำมาใช้งานร่วมกับระบบดังกล่าวให้สามารถใช้งานได้แบบกึ่งอัตโนมัติ และทำการทดสอบโปรแกรมที่เขียนขึ้น โดยมุ่นเน้นไปที่ผู้ป่วยกลุ่มเสี่ยง STROKE and STEMI ซึ่งอาศัยอยู่ในพื้นที่อำเภอสร้างคอม จังหวัดอุดรธานี เพื่อการค้นหาผู้ป่วยเมื่อทราบพิกัด ทาได้โดยการใช้โปรแกรมที่ผู้วิจัยพัฒนาขึ้น โดยใช้พิกัดที่ได้จากการส่งสัญญาณขอความช่วยเหลือมาสร้างเส้นทางจากจุดเริ่มต้นที่โรงพยาบาลสร้างคอม ไปยังพิกัดที่ผู้ป่วยอาสาสมัครส่งสัญญาณมาแสดงบนแผนที่กูเกิ้ล โปรแกรมที่ผู้วิจัยพัฒนาจะคานวณระยะทาง สร้างเส้นทางพร้อมคานวณเวลาเดินทางไปถึงจุดที่ผู้ป่วยอยู่ให้โดยอัตโนมัติ

ผศ.ดร.สุรเชษฐ์ กานต์ประชา, เศรษฐา ตั้งค้าวานิช (2557) วัตถุประสงค์เพื่อพัฒนาระบบติดตามและประมาณเวลาการเดินรถไฟฟ้าด้วยสมาร์ทโฟนผ่านเครือข่าย 3G ระบบรถไฟฟ้าเป็นระบบขนส่งมวลชนที่ทางมหาวิทยาลัยนเรศวรนำเข้ามาใช้งานเพื่อช่วยให้การเดินทางของนิสิต และบุคลากรมีความสะดวกสบายมากขึ้น และเพื่อเป็นการรณรงค์การใช้พลังงานอย่างมีประสิทธิภาพ เนื่องจากพลังงานที่ใช้ขับเคลื่อนรถไฟฟ้านั้นเป็นพลังงานไฟฟ้านั่นเองอย่างไรก็ตาม เพื่อให้การใช้บริการรถไฟฟ้าของบุคลากรและนิสิตมีความสะดวกมากยิ่งขึ้น ประเด็นหนึ่งที่น่าสนใจคือความสามารถที่จะระบุถึงเวลาการมาถึงของรถไฟฟ้า ณ ป้ายจอดรถ ในงานวิจัยชิ้นนี้ได้อาศัยหลักการของการตรวจจับตำแหน่งของรถไฟฟ้าโดยอาศัยระบบ GPS (Global Positioning System) ซึ่งในปัจจุบันมีอยู่ในอุปกรณ์สื่อสารโดยทั่วไป ยกตัวอย่างเช่น สมาร์ทโฟน และการสมการ ทางคณิตศาสตร์ มาใช้ในการคำนวณและประมาณเวลาในการมาถึงของรถไฟฟ้า ณ ป้ายจอดรถ โดยมีการออกแบบในส่วนของส่วนติดต่อผู้ใช้ผ่านโปรแกรมประยุกต์สำหรับผู้ใช้สมาร์ทโฟน เพื่อให้ผู้ใช้บริการรถไฟฟ้าสามารถประมาณเวลาและตำแหน่งของรถไฟฟ้าที่ต้องการจะใช้บริการอาศัยสมาร์ทโฟน และมีการออกแบบในส่วนของป้าย Seven Segment เพื่อใช้ในการบอกเวลาการมาถึงของรถไฟฟ้า ณ ป้ายจอดรถ ซึ่งสามารถได้ระบบติดตามและประมาณเวลาการเดินรถไฟฟ้าด้วยสมาร์ทโฟนผ่านเครือข่าย 3G ได้

ปราชญา สมชื่อ, ภูมิรพี สืบวงษ์รอด (2557) วัตถุประสงค์เพื่อนำระบบจีพีเอสมาช่วยอำนวยความสะดวกให้กับผู้ใช้ในชีวิตประจำวันซึ่งการสร้างอุปกรณ์ฮาร์ดแวร์และซอฟต์แวร์เพื่อใช้ในการติดตามยานพาหนะ โดยการติดตั้งอุปกรณ์ที่ได้จัดทำไว้ในยานพาหนะ แล้วสามารถเรียกดูประวัติข้อมูล และตำแหน่งของยานพาหนะได้จากซอฟต์แวร์ที่เป็นแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ โดยในส่วนของอุปกรณ์ฮาร์ดแวร์นั้นใช้โมดูล SIM908 ซึ่งในโมดูลตัวนี้ได้รวมโมดูลจีเอสเอ็มและโมดูลจีพีเอส เข้าด้วยกัน และใช้ไมโครคอนโทลเลอร์ของค่าย Arduino รุ่น ATMEGA2560

R3 เพื่อติดต่อรับส่งข้อมูล โดยเริ่มจากการส่งคาสั่งติดต่อกับโมดูลจีพีเอส เพื่อร้องขอตำแหน่งพิกัด แล้วนาข้อมูลทั้งหมดที่ได้มากรองเพื่อให้ได้ข้อมูลของวัน เวลา และตำแหน่งของยานพาหนะแล้วทาการส่งคาสั่งเพื่อให้ โมดูลจีเอสเอ็มทาการส่งข้อมูลไปยังเซอร์วิสโดยผ่านเว็บเซอร์วิส เพื่อที่จะให้แอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์สามารถเรียกดูประวัติข้อมูล และตำแหน่งของยานพาหนะผ่าน แอปพลิเคชันที่ได้ทาการพัฒนาขึ้น ซึ่งสามารถใช้บันทึกข้อมูลในการเดินทางเพื่อให้ผู้ใช้งานสามารถไปวิเคราะห์ต่อได้ เช่น วิเคราะห์ข้อมูลเส้นทางที่ใกล้และปลอดภัยที่สุดได้

วีรชัย สว่างทุกข์ (2557) วัตถุประสงค์เพื่อศึกษาการทำงานของซอฟต์แวร์ติดตาม GPS แบบเปิดรหัสร่วมกับสมาร์ตโฟนสำหรับติดตาม บันทึกเส้นทาง ความเร็ว ตำแหน่งของรถขนส่ง โดยตรวจสอบตำแหน่งพิกัดด้วยการรับสัญญาณจากดาวเทียม GPS แล้วส่งข้อมูลตำแหน่งและข้อมูลอื่น ๆ จากซอฟต์แวร์ไคลเอนต์ที่ติดตั้งในสมาร์ทโฟนด้วยการรับส่งข้อมูลของโทรศัพท์เคลื่อนที่เช่น GPRS มาจัดเก็บที่เซิร์ฟเวอร์ เพื่อแสดงตำแหน่งและเส้นทางของรถขนส่งบนแผนที่ ซึ่งสามารถศึกษาระบบติดตามจีพีเอสแบบเปิดเผยรหัสต้นฉบับควบคู่สมาร์ตโฟน เพื่อใช้ติดตามรถขนส่งกรณีศึกษาน้าดื่มทิพย์เขลางค์ เป็นนาซอฟต์แวร์แบบเปิดเผยรหัส มาใช้ควบคู่กับสมาร์ตโฟนเพื่อทดแทนการซื้อหรือเช่าใช้ซอฟต์แวร์และอุปกรณ์การ สามารถติดตามรถขนส่งของผู้ประกอบการน้าดื่มทิพย์เขลางค์ได้อย่างดี โดยแสดงเส้นทาง ความเร็ว ตำแหน่งของรถขนส่งได้อย่างถูกต้อง

วิภาดา เพชรรัตน์ (2554) วัตถุประสงค์เพื่อสร้างระบบที่สามารถหาเส้นทางในการเดินทางไปพบปะลูกค้าที่มีระยะทางรวมที่สั้นที่สุดหรือใช้เวลาที่น้อยที่สุดสำหรับลดค้าใช้จ่ายในการเดินทางซึ่งได้นำเสนอรูปแบบระบบที่ช่วยวางแผนการเดินทางไปพบลูกค้าของพนักงานในบริษัท โดยระบบจะหาเส้นทางที่เหมาะสมกับเงื่อนไขของการเดินทางซึ่งมรสองอัลกอริทึมเป็นส่วนสำคัญในการวิเคราะห์ ทำให้ได้เส้นทางที่เหมาะสมและรูปแบบเดียวกันระหว่างผู้จัดการกับพนักงานขาย จะส่งผลให้ค่าใช่จ่ายในการเดินทางลดลงและมีประสิทธิภาพมากยิ่งขึ้น โดยประยุกต์อัลกอริทึมดิสตรา สำหรับการหาเส้นทางที่สั้นที่สุดหรือเวลาน้อยที่สุดระหว่างจุด 2 จุด และใช้อัลกอริทึมการแตกกิ่งและจำกัดขอบเขต สำหรับการหาเส้นทางที่สั้นที่สุดและเวลาที่น้อยที่สุดระหว่างจุดหลายจุด เมื่อนำสองอัลกอริทึมมาใช้แก้ปัญหาในการวางแผนการเดินทางของพนักงานขายแล้ว สามารถแนะนำเส้นทางที่เหมาะสมในการเดินทาง เพื่อความรวดเร็วในการเดินทางและลดค่าใช่จ่ายในการเดินทางได้และสามารถเพิ่มประสิทธิภาพในการเดินทางไปพบปะลูกค้าให้ดียิ่งขึ้น

งานวิจัยต่างประเทศ 3 เรื่อง ดังนี้

Ahmed Ahmed, Elshaimaa Nada, Wafaa Al-Mutiri (2017) แอปพลิเคชันสำหรับโทรศัพท์มือถืออัจฉริยะส่วนใหญ่เป็น GPS หลายวิธีถูกเสนอเพื่อลดการสูญเสียเวลาของนักเรียนรอรถบัสมาถึง การติดตามและตรวจสอบระบบติดตามรถจาก GPS โดยใช้การติดตามแผนที่กับ

สถานที่ GPS แอปพลิเคชัน Android ได้รับการออกแบบมาสำหรับนักเรียนที่สามารถทำได้เข้าถึง / ดูตารางเวลาประจำวันของรถบัสเส้นทางรถบัสที่ตั้งของรถประจำทางและรถประจำทางถึงและระยะเวลาการล่าช้าข้อมูล. จุดสำคัญหลักคือการจัดหานักเรียนให้มีระบบดังกล่าวอย่างแน่นอนลดเวลารอคอยและจะให้รายละเอียดที่จำเป็นแก่นักเรียนเวลาที่เดินทางมาถึงของรถบัส, ตำแหน่งที่แน่นอนและเวลาที่แม่นยำ

João Paulo Lima, Rafael Roberto, Francisco Simões, Mozart Almeida, Lucas Figueiredo, João Marcelo Teixeira a, Veronica Teichrieb (2017) นำเสนอระบบการติดตามตามคุณลักษณะตามธรรมชาติที่สมบูรณ์แบบซึ่งสนับสนุนการสร้างแอปพลิเคชันความเป็นจริงที่เน้นไปที่ภาคยานยนต์ ท่อที่นำเสนอประกอบด้วยการสร้างแบบจำลองฉากการปรับเทียบระบบและขั้นตอนการติดตาม แอปพลิเคชันความเป็นจริงที่เพิ่มขึ้นถูกสร้างขึ้นที่ด้านบนของระบบเพื่อระบุตำแหน่งของพิกัด 3 มิติในสภาพแวดล้อมที่กำหนดซึ่งสามารถประยุกต์ใช้กับแอปพลิเคชันต่าง ๆในรถยนต์เช่นผู้ช่วยบำรุงรักษาคู่มืออัจฉริยะและอื่น ๆ อีกมากมาย การวิเคราะห์ระบบได้ดำเนินการระหว่างโฟล์คสวาเก้น / ISMAR Tracking Challenge 2014 ซึ่งมีจุดมุ่งหมายเพื่อประเมินแนวทางการติดตามที่ล้ำสมัยบนพื้นฐานของข้อกำหนดที่พบในการตั้งโรงงานอุตสาหกรรมยานยนต์ สภาพแวดล้อมการแข่งขันที่คล้ายกันนี้ถูกสร้างขึ้นโดยผู้เขียนเพื่อให้สามารถศึกษาเพิ่มเติมได้ ผลการประเมินแสดงให้เห็นว่าระบบอนุญาตให้ผู้ใช้ระบุจุดในงานที่เกี่ยวข้องกับการติดตามยานพาหนะที่หมุนได้อย่างถูกต้องการติดตามข้อมูลบนยานพาหนะที่สมบูรณ์และการติดตามด้วยความแม่นยำสูง การประเมินนี้ยังช่วยให้เข้าใจถึงข้อ จำกัด ของการประยุกต์ใช้เนื้อสัมผัสในสภาพแวดล้อมยานยนต์ที่ไม่มีเนื้อสัมผัสซึ่งเป็นปัญหาที่ไม่ได้กล่าวถึงบ่อยในวรรณคดี ความรู้ที่ดีที่สุดของผู้เขียนคืองานแรกที่ใช้ในการวิเคราะห์ระบบการติดตามที่สมบูรณ์สำหรับความเป็นจริงที่เพิ่มขึ้นซึ่งมุ่งเน้นไปที่ภาคอุตสาหกรรมยานยนต์ซึ่งสามารถทดสอบและรับรองความถูกต้องในมาตรฐานที่สำคัญเช่น Volkswagen / ISMAR Tracking Challenge, การให้ข้อมูลเชิงลึกที่เป็นประโยชน์เกี่ยวกับการพัฒนาผู้เชี่ยวชาญและระบบอัจฉริยะ

Rujuta Kharade, Supriya Kulkarni, Madhura Thorat, Sneha Walhekar (2017) ความสามารถในศักยภาพของ Global Positioning System (GPS) เพื่อช่วยในการนำทางและติดตามการใช้งานช่วยในการกำหนดตำแหน่งที่แม่นยำบนโลก เพื่อประสิทธิภาพในการดำเนินการติดตาม GPS จะขึ้นอยู่กับตัวแปรต่าง ๆ ข้อมูลที่รวบรวมทั้งหมดจะถูกติดตามอย่างถูกต้องในแบบเรียลไทม์ บทความนี้เกี่ยวข้องกับการศึกษาเกี่ยวกับ GPS, LBS และการผสานรวมระบบคลาวด์ บริการตามตำแหน่ง (LBS) คือบริการระดับซอฟต์แวร์ที่ให้ข้อมูลที่แม่นยำเกี่ยวกับตำแหน่งที่จะติดตาม ในระบบที่เสนอฐานข้อมูลถูกเก็บไว้ในระบบคลาวด์ ฐานข้อมูลขนาดใหญ่ได้รับการจัดการอย่างมี

ประสิทธิภาพด้วยเทคนิคการรวมระบบคลาวด์ ระบบใช้บริการ Cloud หลายสำหรับการทำงานที่มีประสิทธิภาพสำหรับฐานข้อมูลขนาดใหญ่ แบบสอบถามต่าง ๆได้รับการประเมินสำหรับการเข้าถึงฐานข้อมูลการใช้งานผู้ใช้ปลายทางของขั้นตอนวิธีการวัดการวัดตำแหน่งจะกล่าวถึง ระบบติดตามตำแหน่งในเวลาจริงสำหรับแอปพลิเคชันแอนดรอยด์ถูกนำมาใช้ นอกจากนี้ระบบจะนำเสนอด้วยผลลัพธ์สุดท้ายในรูปแบบของกราฟข้อความ

บทที่ 3

วิธีดำเนินการวิจัย

บทที่ 3 นี้เป็นรูปแบบวิธีดำเนินการวิจัย โดยนำหลักการและทฤษฎีที่ได้ศึกษาการพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ มาใช้ดำเนินโครงการวิจัย ซึ่งประกอบด้วยการทำงานออกเป็น 5 ส่วน ดังนี้

3.1 ศึกษาปัญหาและแนวทางการพัฒนาแอปพลิเคชัน

3.2 การวิเคราะห์ระบบ

3.3 การออกแบบระบบ

3.4 การพัฒนาระบบ

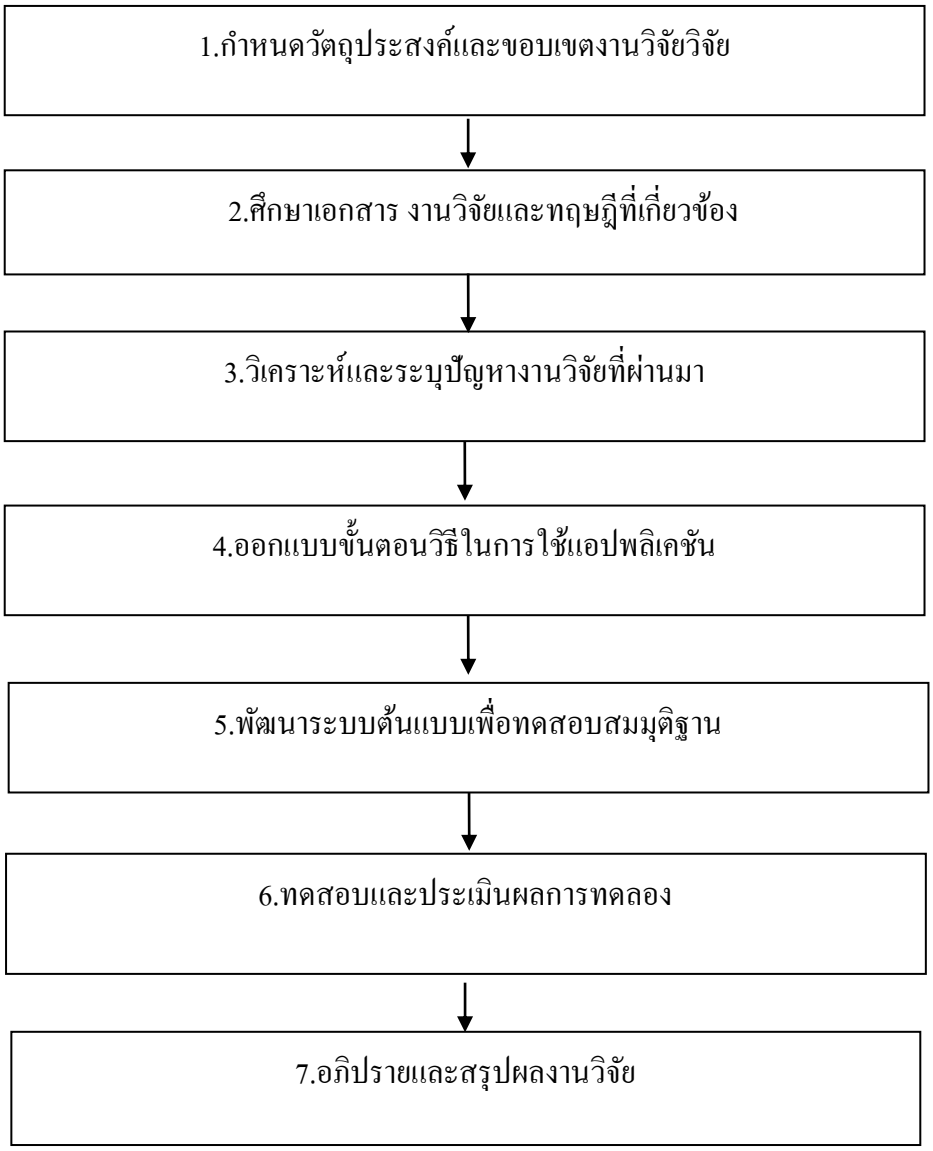
3.5 การทดสอบระบบ

3.6 การประเมินผลการใช้งาน

3.7 แผนการดำเนินงานวิจัย

3.1 ศึกษาปัญหาและแนวทางการพัฒนาแอปพลิเคชัน

ศึกษาปัญหาและแนวทางการพัฒนาแอปพลิเคชัน โดยใช้การพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ ตามวงจรการพัฒนาระบบ (System Development Life Cycle : SDLC) มีขั้นตอนรายละเอียด ดังแสดงในภาพประกอบที่ 3.1 ดังนี้



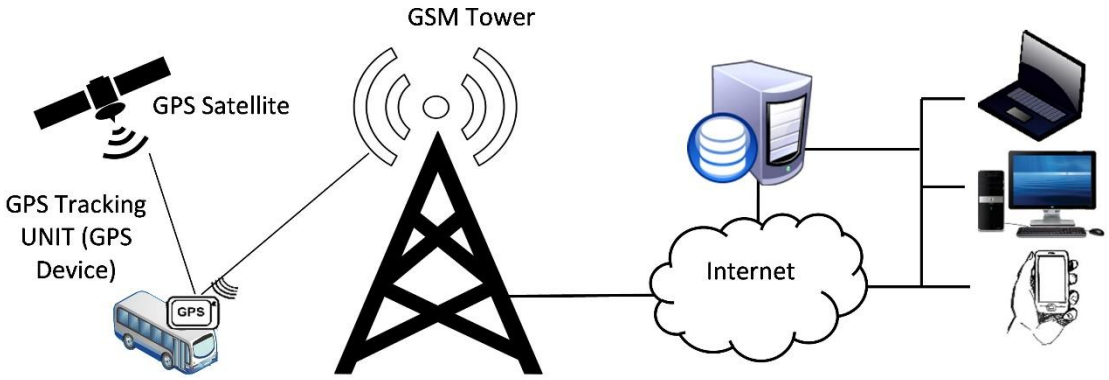
ภาพประกอบที่ 3.1 ขั้นตอนการดำเนินการวิจัย

3.2 การวิเคราะห์ระบบ

การวิเคราะห์และการออกแบบการพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ ได้ทำการศึกษาและรวบรวมข้อมูลทางด้านต่าง ๆ เพื่อนำไปใช้ในการพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ ได้อย่างมีประสิทธิภาพโดยแบ่งการออกแบบออกเป็น 2 ส่วน คือ

3.2.1 การวิเคราะห์ภาพรวมของระบบ

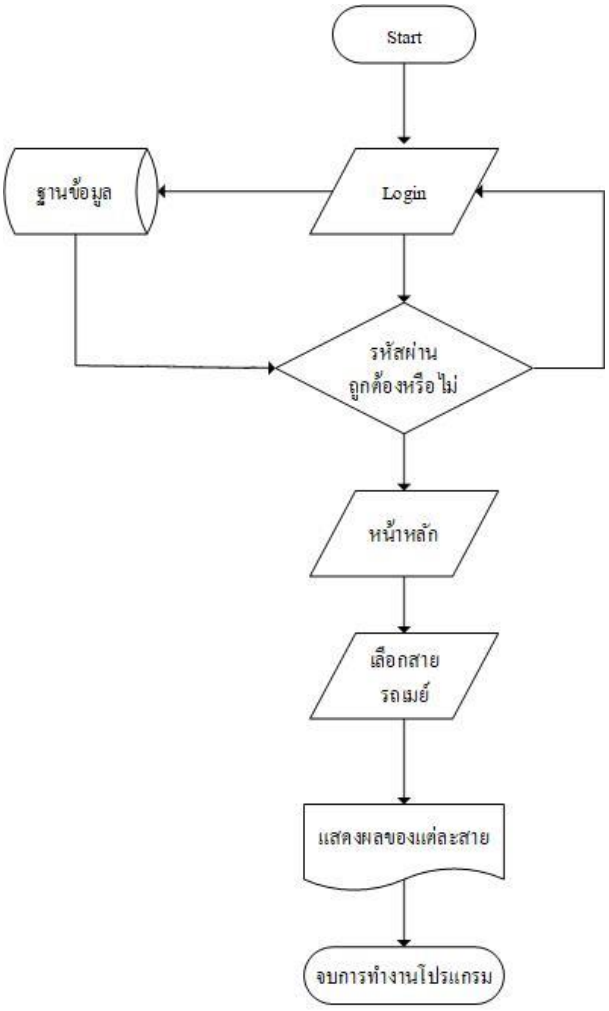
จากการการวิเคราะห์และการออกแบบการพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ การทำงานของระบบจะประกอบไปด้วย 3 ส่วน คือ ส่วนอวกาศ (Space segment) ส่วนควบคุมดาวเทียม (Control Segment) และส่วนผู้ใช้ (User segment) ซึ่งส่วนอวกาศระบบ GPS จะเป็นการทำงานของดาวเทียม GPS จำนวนทั้งหมด 24 ดวง ที่โคจรอยู่รอบโลก ที่ความสูงจากพื้นโลกประมาณ 20,000 กิโลเมตร เป็นตัวส่งสัญญาณบอกพิกัดของจุดที่ต้องการทราบ สัญญาณดังกล่าวจะต้องถูกส่งมาจากดาวเทียมอย่างน้อย 3 ดวงขึ้นไป ในการส่งพิกัดที่ถูกต้องมายังอุปกรณ์บนพื้นโลก ส่วนควบคุมดาวเทียม ซึ่งอยู่บนพื้นโลก ประกอบไปด้วย 1 สถานีหลัก และ 5 สถานีย่อยที่กระจายกันอยู่ตามตำแหน่งต่าง ๆ ศูนย์ควบคุมนี้จะทำหน้าที่ในการควบคุมและติดต่อสื่อสารกับดาวเทียม รวมทั้งคำนวณผลจากดาวเทียมแต่ละดวง และส่งข้อมูลที่ได้โต้ตอบกลับไปยังดาวเทียม ทำให้ข้อมูลที่ได้รับอัพเดทตลอดเวลา ส่วนของผู้ใช้นี้ จะเป็นการดูตำแหน่งหรือพิกัดที่ได้รับจากดาวเทียม ผ่านการประมวลผลจากเครื่องมือรับสัญญาณ เพื่อให้ได้จุดของตำแหน่งที่ต้องการทราบได้



ภาพประกอบที่ 3.2 แนวคิดโดยรวมของระบบ

จากภาพประกอบที่ 3.2 เมื่ออุปกรณ์ที่ติดตั้งบนรถยนต์ทำงาน ไมโครคอนโทรลเลอร์จะเริ่มสั่งให้จีพีเอสโมดูลเก็บพิกัดจีพีเอสและตัดข้อมูลในส่วนที่ไม่ใช่ออก จากนั้นจะส่งข้อมูลไปยังเว็บเซอร์วิสโดยผ่านสัญญาณ จีพีอาเอสเพื่อแปลงข้อมูลเป็นองศาทศนิยม (Degree Decimal) และเก็บลงฐานข้อมูล เมื่อต้องการทราบตำแหน่งหรือประวัติการเดินทางของรถสามารถเรียกดูได้ผ่านแอปพลิเคชันบนระบบปฏิบัติการ iOS

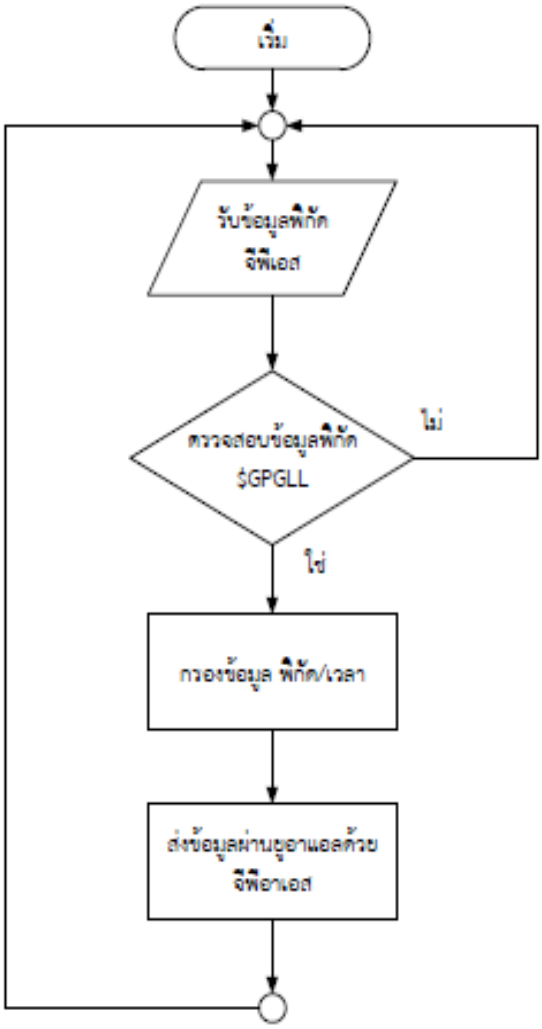
3.3 การออกแบบระบบ



ภาพประกอบที่ 3.3 ออกแบบระบบ

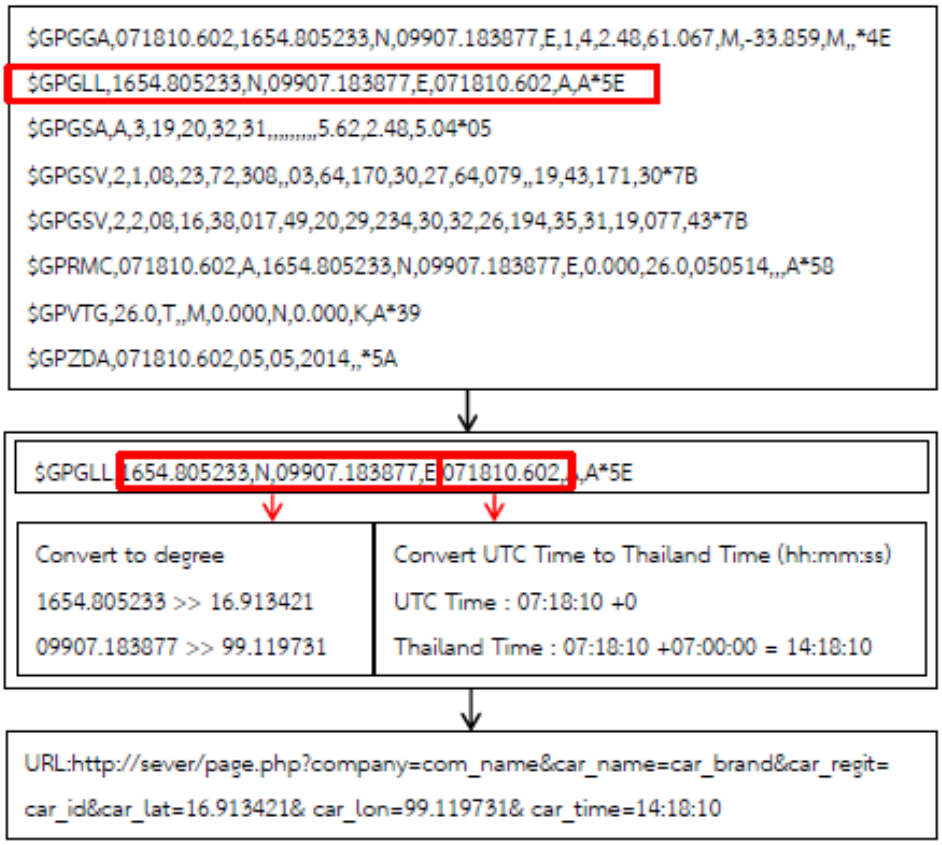
3.3.1 การออกแบบโปรแกรมการทำงานของตัวอุปกรณ์

ในส่วนการออกแบบโปรแกรมการทำงานนั้น เป็นตัวโปรแกรมที่ถูกเขียนให้กับไมโครคอนโทรลเลอร์เพื่อติดต่อกับโมดูล SIM908 โดยมีขั้นตอนหลัก 3 ขั้นตอน คือ การรับข้อมูลพิกัดจากโมดูลจีพีเอส การนำข้อมูลพิกัดมากรองเพื่อให้ได้ข้อมูลที่ต้องการโดยเลือกใช้รูปแบบพิกัดของ $GPGLL และการส่งข้อมูลพิกัดที่ต้องการไปยังเซิร์ฟเวอร์ผ่าน URL ด้วยโมดูลจีเอสเอ็ม โดยมีผังการทำงานดังภาพประกอบที่ 3.4



ภาพประกอบที่ 3.4 ขั้นตอนการทำงานของอุปกรณ์ฮาร์ดแวร์

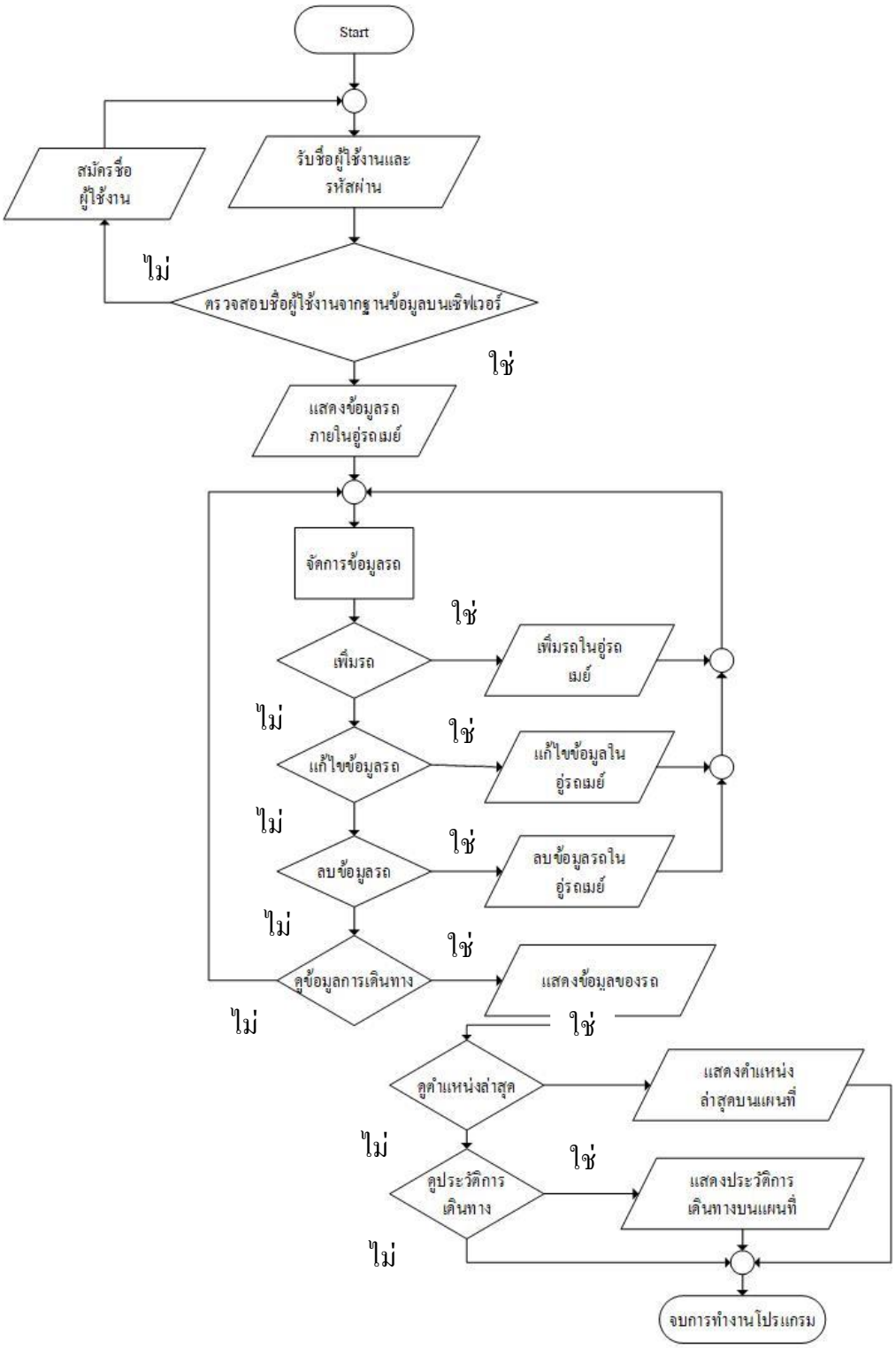
จากภาพประกอบที่ 3.4 สามารถอธิบายได้ คือ เมื่อตัวอุปกรณ์เริ่มทำงานจะทำการรับข้อมูลพิกัด จีพีเอส หลังจากนั้นทำการกรองข้อมูลโดยเฉพาะรูปแบบพิกัดของ $GPGLL แล้วทำการตัดข้อมูล นำเฉพาะข้อมูลพิกัดและเวลามาใช้ หลังจากนั้นนำข้อมูลที่ได้ส่งผ่าน URL ไปยังเซิร์ฟเวอร์



ภาพประกอบที่ 3.5 ตัวอย่างการเตรียมข้อมูลที่ถูกนำไปใช้งาน

3.3.2 การออกแบบโปรแกรมการทำงานไอโอเอสแอปพลิเคชัน

ในส่วนของการออกแบบโปรแกรมการทำงานของแอปพลิเคชันเป็นซอฟต์แวร์ที่ถูกเขียนให้กับอุปกรณ์โทรศัพท์มือที่มีระบบปฏิบัติการเป็นระบบปฏิบัติการไอโอเอส เพื่อรับข้อมูลจากเซิร์ฟเวอร์มาแสดงผลบนแอปพลิเคชัน ในการแสดงผลสามารถแบบเป็น 2 ส่วนหลัก คือ การแสดงข้อมูลรายละเอียดของรถ การแสดงข้อมูลประวัติการใช้รถและตำแหน่งล่าสุดของรถ โดยมีผังการทำงานดังรูปที่ 3.6

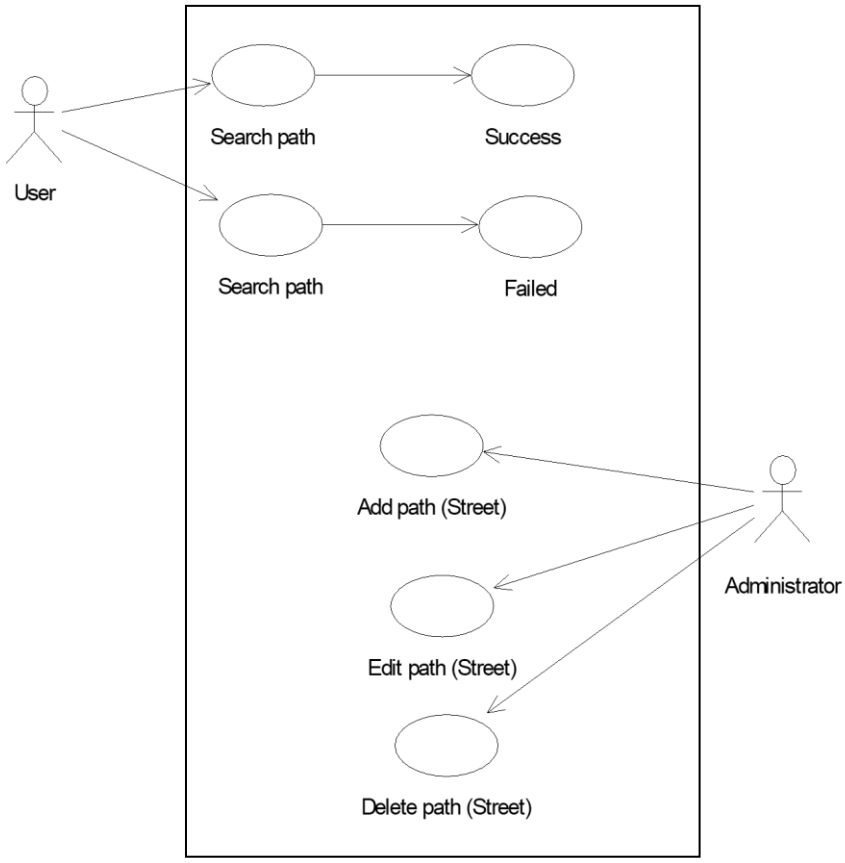


ภาพประกอบที่ 3.6 ขั้นตอนการทำงานของระบบ

จากภาพประกอบที่ 3.6 อัลกอริทึมการทำงานของระบบ โดยเมื่อเริ่มการทำงาน ระบบจะให้ผู้ใช้กรอกชื่อผู้ใช้งานและตรวจสอบว่ามีชื่อผู้ใช้งานอยู่ในระบบหรือไม่ ถ้าไม่มีสามารถสมัครชื่อผู้ใช้งานได้โดยการกดสมัคร เมื่อเข้าระบบแล้วจะมีหน้าเลือกรถเพื่อดูข้อมูลการเดินทาง โดยในส่วนนี้สามารถเพิ่ม ลบและแก้ไข จานวนรถของบริษัทได้ เมื่อเลือกรถที่ต้องการดูแล้ว ระบบจะแสดงรายละเอียดของรถคันนั้นและให้เลือกว่าจะดูตำแหน่งล่าสุดหรือประวัติการเดินทางของรถคันนั้นโดยจะแสดงบนแผนที่จากกูเกิลแมพ

3.3.3 การออกแบบสถาปัตยกรรมระบบ

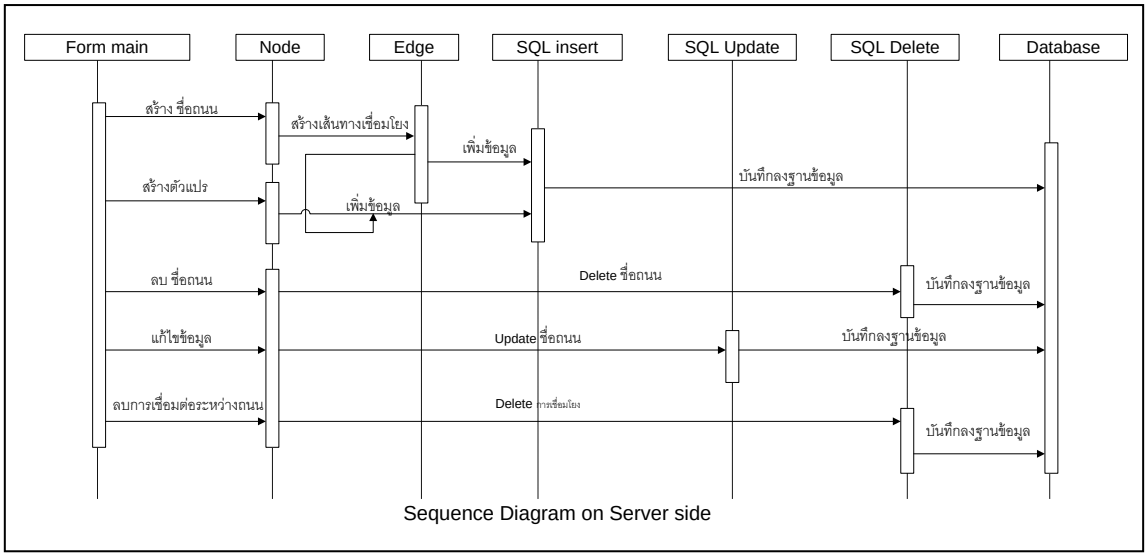
ออกแบบในลักษณะ Client – Server เพื่อผู้ใช้สามารถใช้งานผ่านเว็บไปยังเครื่องแม่ข่าย (Server) ได้โดยมีการออกแบบ Use Case View ที่ประกอบด้วย Use Case Diagram และบุคคลที่เกี่ยวข้องได้แก่ ผู้ใช้งานและผู้ดูแลระบบดังภาพกระกอบที่ 3.7



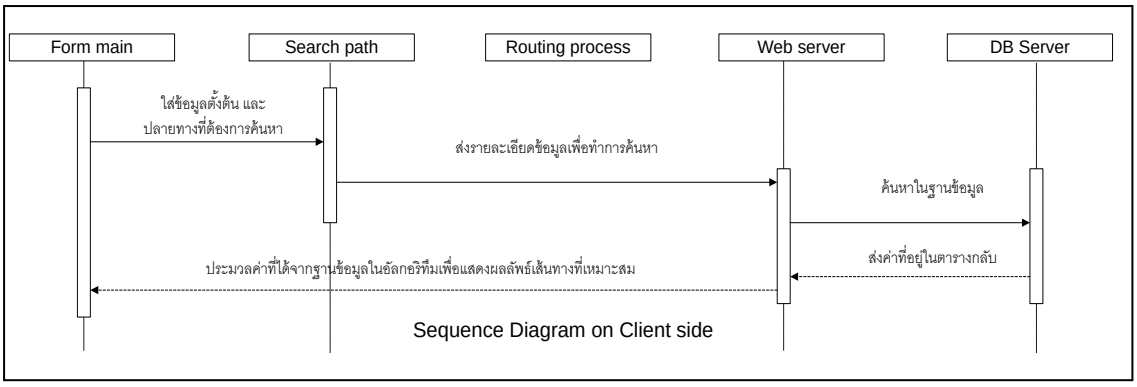
ภาพประกอบที่ 3.7 แสดง Use Case Diagram ของระบบ

(ที่มา : สุขสวัสดี ณัฏฐวุฒิสิทธิ์, ประสงค์ ปราณีตพลกรัง, 2011)

จากนั้นได้ออกแบบผังลำดับการทำงาน (Sequence diagram) ดังมีรายละเอียดตามภาพประกอบที่ 3.8 และ 3.9



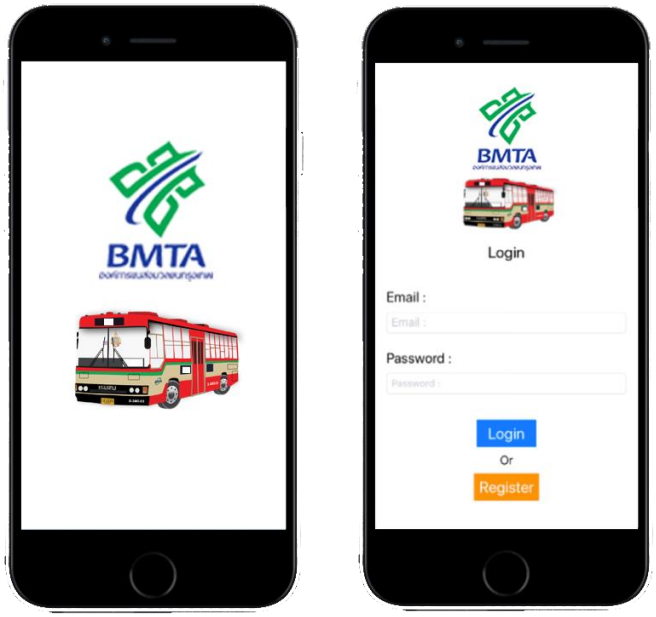
ภาพประกอบที่ 3.8 แสดงผังลำดับการทำงานในส่วน เครื่องแม่ข่าย (Server)
(ที่มา : สุขสวัสดี ณัฏฐวุฒิสิทธิ์, ประสงค์ ปราณีตพลกรัง, 2011)



ภาพประกอบที่ 3.9 แสดงผังลำดับการทำงานในส่วนเครื่องลูกข่าย (client)
(ที่มา : สุขสวัสดี ณัฏฐวุฒิสิทธิ์, ประสงค์ ปราณีตพลกรัง, 2011)

3.5 การพัฒนาระบบต้นแบบ

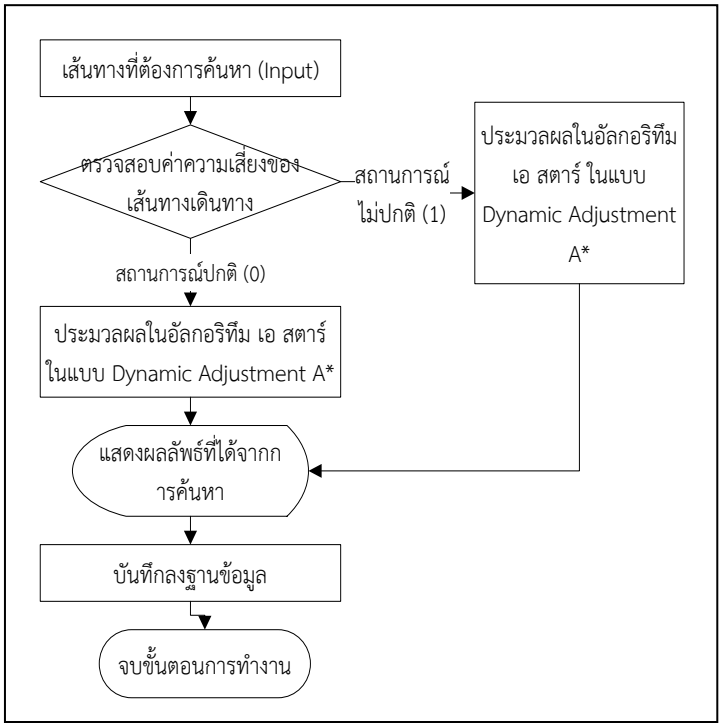
ระบบที่พัฒนาได้แก่ ส่วนของผู้ใช้ระบบและส่วนของผู้ดูแลระบบ ในส่วนของผู้ใช้ระบบจะเป็นการค้นหาเส้นทางรวมถึงการแสดงผลลัพธ์ผ่านหน้าจอระบบ และในส่วนผู้ดูแลระบบจะเป็นการสร้างข้อมูลเส้นทาง และการกำหนดเส้นทางที่จะนำมาใช้ในการค้นหา และมีการใช้งานระบบผ่านทางเบราว์เซอร์ ดังภาพประกอบที่ 3-10



ภาพประกอบที่ 3.10 แสดงส่วนของผู้ใช้ระบบ

3.6 การทดสอบระบบจำลอง

การทดสอบระบบด้วยการจำลองสถานการณ์การทำงานของอัลกอริทึมของเอ สตาร์ โดยใช้ฐานข้อมูลในการจัดเก็บข้อมูลของเส้นทางที่เคยค้นหาเพื่อช่วยลดระยะเวลาของการทำงานในภาพรวมของระบบ ดังแสดงในภาพประกอบที่ 3.11



ภาพประกอบที่ 3.11 การทดสอบระบบด้วยการจำลองการทำงานของอัลกอริทึมของเอสตาร์

1) เริ่มต้นจากอัลกอริทึมตรวจสอบเส้นทางที่ต้องการค้นหาจากฐานข้อมูลเดิมที่เคยถูกเรียกใช้ หากพบว่าเคยมีการเรียกใช้ก่อนหน้า อัลกอริทึมจะแสดงผลลัพธ์ออกมาให้ผู้ใช้ทราบ และบันทึกลงฐานข้อมูล

2) หากพบว่ายังไม่มีการค้นหาของเส้นทางที่ต้องการในฐานข้อมูล อัลกอริทึมจะเริ่มต้นค้นหาตามรูปแบบพลวัต เอ สตาร์

3) จากนั้นอัลกอริทึมทำการเช็คค่าความเสี่ยงของเส้นทางที่เลือกมานั้นว่ามีสถานะเป็นอะไร

$\lambda = 0$ ไม่สามารถเดินทางได้ สถานการณ์ไม่ปกติ (0)

$\lambda = 1$ ไม่สามารถเดินทางได้ สถานการณ์ปกติ (1)

4) หากอัลกอริทึมพบว่ามีเส้นทางใดเส้นทาง หนึ่งไม่สามารถเดินทางได้ ($\lambda = 0$) อัลกอริทึมจะทำการคัดเลือกเส้นทางอื่นที่แทน

5) หากอัลกอริทึมไม่พบเส้นทางที่จะเดินทางได้อัลกอริทึมจะเสนอเส้นทางที่เหมาะสมอื่นให้แทนโดยพิจารณาจากค่า Heuristic (H*) ซึ่งจะแนะนำเส้นทางเลือกจากฐานข้อมูลที่มีอยู่เพื่อให้ผู้ใช้งานได้ทราบและตัดสินใจที่จะเลือกเส้นทางใหม่นี้แทน

3.7 การประเมินผลการใช้งาน

การประเมินผลการใช้งานได้ใช้แบบสอบถามสำหรับการประเมิน ซึ่งมีมาตรวัดอัตราส่วน 5 ระดับ (Rating Scale) ได้แก่ ระดับดีมากเท่ากับ 5, ระดับดีเท่ากับ 4, ระดับปานกลางเท่ากับ 3, ระดับพอใช้เท่ากับ 2 และระดับปรับปรุงเท่ากับ 1 เพื่อประเมินความพึงพอใจของผู้ใช้งานที่มีต่อคลังข้อมูลขนาดใหญ่ โดยแบ่งเป็น 2 แบบ ได้แก่

4.1 แบบประเมินความพึงพอใจผู้ใช้งานคลังข้อมูลขนาดใหญ่โดยสอบถามจากผู้ใช้งานจำนวน 15 คน โดยแบ่งเป็น ผู้บริหาร 2 คน หัวหน้ากลุ่มงาน 3 คน และเจ้าหน้าที่ศูนย์เทคโนโลยีสารสนเทศและการสื่อสาร 10 คน เพื่อหาค่าเฉลี่ย ($\bar{X}$) ดังสมการ (1)

4.2 แบบประเมินความเชื่อมั่นผู้ใช้งานโดยสอบถามจากผู้ใช้งานจำนวน 20 คน โดยแบ่งเป็นเจ้าหน้าที่แต่ละกลุ่มส่วนงานของศูนย์เทคโนโลยีสารสนเทศและการสื่อสารดังสมการ (1)

$$\overline{X} = \frac{\sum X}{N} \qquad (1)$$

3.8 แผนการดำเนินงานวิจัย

ตารางที่ 3.1 แผนการดำเนินงาน

กิจกรรม	พ.ศ. 2561							
	เดือน							
	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.
1. ศึกษาทำความเข้าใจเกี่ยวกับปัญหา	■							
1) รวมรวมข้อมูลและกำหนดแหล่งที่มาของข้อมูล		■						
2) วิเคราะห์ระบบเดิม			■					
3) ออกแบบระบบให้สอดคล้องกับความต้องการ				■				
4) พัฒนาระบบให้เป็นไปตามที่กำหนด					■	■		
5) ทดสอบระบบ							■	
6) สรุปผลและจัดส่งเล่มวิจัย								■

บทที่ 4

ผลงานวิจัย

งานวิจัยครั้งนี้เป็นการพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ เครื่องมือที่ใช้ในงานวิจัย ได้แก่ ทั้งหมด 12 เดือน เริ่มตั้งแต่เดือนสิงหาคม พ.ศ. 2560 – สิงหาคม พ.ศ. 2561 ซึ่งผลการทดลองเป็นไปตามวัตถุประสงค์ที่วางไว้จำนวน 2 ข้อ ดังนี้

1. เพื่อศึกษาปัญหาและแนวทางการพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์
2. เพื่อพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์

4.1 ผลการวิจัยตามวัตถุประสงค์ข้อที่ 1

วัตถุประสงค์งานวิจัยข้อที่ 1 เพื่อศึกษาปัญหาและแนวทางการพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ ซึ่งผลจากการวิจัยพบว่า

1. ระบบที่ได้รับการวิเคราะห์และออกแบบให้ผู้ใช้งานคัดเลือกเส้นทางโดยสารรถประจำทางที่เหมาะสมบนพื้นฐานอัลกอริทึมพลวัต เอ สตาร์ เพื่อให้ผู้ใช้งานสามารถบริหารจัดการเส้นทางและคำนวณเวลา ประโยชน์ที่คาดจะได้รับครั้งนี้จะช่วยลดปัญหาการรอคิวรถโดยสารประจำทางเป็นเวลานานโดยผู้ให้และผู้รับบริการสามารถทราบกำหนดเวลาการเข้าจอดรถได้โดยประมาณ ทำให้สามารถวางแผนการเดินทางของตนเองได้

ตาราง 4.1 สรุปการวิเคราะห์และออกแบบระบบโดยการเปรียบเทียบกับงานวิจัยที่เกี่ยงข้อง

system	โครงการวิจัย เรื่อง ระบบติดตามและประมาณเวลาการเดินรถไฟฟ้าด้วยสมาร์ทโฟนผ่านเครือข่าย 3G	ระบบนำทางอัจฉริยะ : กรณีศึกษา บริษัทวานิชรุ่งเรืองอินเตอร์เทรด จำกัด (สาขาภาคใต้)	ระบบติดตามรถยนต์ด้วยจีพีเอสผ่านแอพพลิเคชันบนระบบปฏิบัติการแอนดรอยด์	พัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์
การเข้าระบบ				✓
การคำนวณเส้นทาง		✓		✓
รายละเอียดของเส้นทาง		✓	✓	✓
การประมาณเวลาในการมาถึง	✓			✓

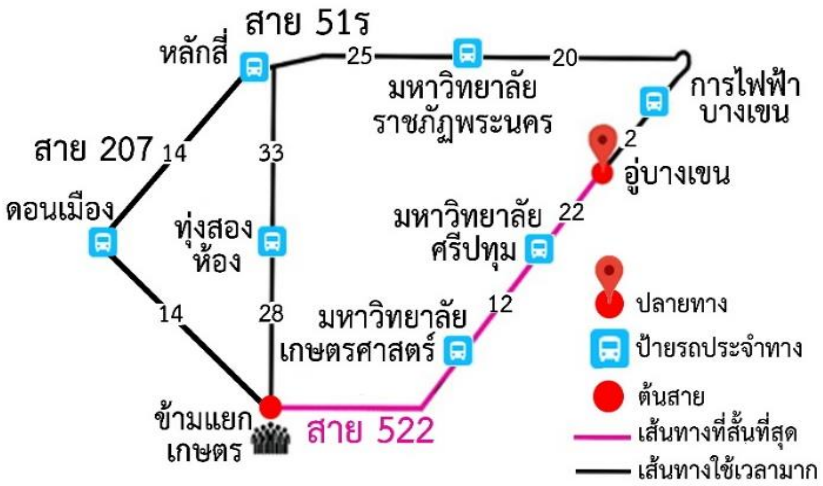
จากตาราง 4.1 มีข้อที่แตกต่างจากงานวิจัยเรื่อง โครงการวิจัย เรื่อง ระบบติดตามและประมาณเวลาการเดินรถไฟฟ้าด้วยสมาร์ทโฟนผ่านเครือข่าย 3G คือ การประมาณเวลาในการมาถึง, ระบบนำทางอัจฉริยะ : กรณีศึกษา บริษัทวานิชรุ่งเรืองอินเตอร์เทรด จำกัด (สาขาภาคใต้) คือ การคำนวณเส้นทาง และรายละเอียดของเส้นทาง, ระบบติดตามรถยนต์ด้วยจีพีเอสผ่านแอพพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ คือ รายละเอียดของเส้นทาง ทำให้มีการทดสอบเป็นไปตามวัตถุประสงค์ที่กำหนด

4.2 ผลการวิจัยตามวัตถุประสงค์ข้อที่ 2

วัตถุประสงค์งานวิจัยข้อที่ 2 เพื่อพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ จากแนวทางวิธีการดำเนินงานที่ได้กล่าวถึงในวิธีดำเนินการวิจัย สามารถแบ่งขั้นตอนต่าง ๆ ตามลำดับได้แก่ ขั้นตอนการศึกษาและรวบรวมข้อมูล การวิเคราะห์ การออกแบบ การพัฒนาระบบ การทดสอบระบบ รวมถึงการแก้ไขปรับปรุงโปรแกรมตามความเหมาะสม โดยในส่วนนี้จะเป็นการกล่าวขยายการดำเนินการในขั้นตอนการเตรียมข้อมูลเพื่อสร้างตัวต้นแบบจำลอง เพื่อ ออกแบบพัฒนาโปรแกรมที่จะนำมาใช้ในการประเมินผลลัพธ์ที่ได้ ซึ่งมีรายละเอียดดังนี้

4.2.1 การศึกษาวิธีการและคัดเลือกอัลกอริทึมการเลือกเส้นทาง

จากการศึกษา และ เปรียบเทียบ อัลกอริทึม ในบทที่ 3 พบว่า อัลกอริทึมพลวัติ เอ สตาร์ เป็นการค้นหาเส้นทางที่สั้นที่สุดที่เหมาะสมที่จะนำมาประยุกต์เพื่อให้สามารถนำไปพัฒนาระบบได้อย่างมีประสิทธิภาพ เนื่องจากสามารถประยุกต์ฟังก์ชันในรูปเงื่อนไขของตัวแปร สำหรับความเสี่ยงของการใช้เส้นทางได้ โดยในงานวิจัยได้สมมุติตัวอย่างเส้นทางที่จะใช้ในการทดสอบระบบดังภาพประกอบที่ 4.1 และรายละเอียดของเส้นทางและความสัมพันธ์ของเส้นทางต่าง ๆที่เชื่อมโยงกันตามตารางที่ 4-1



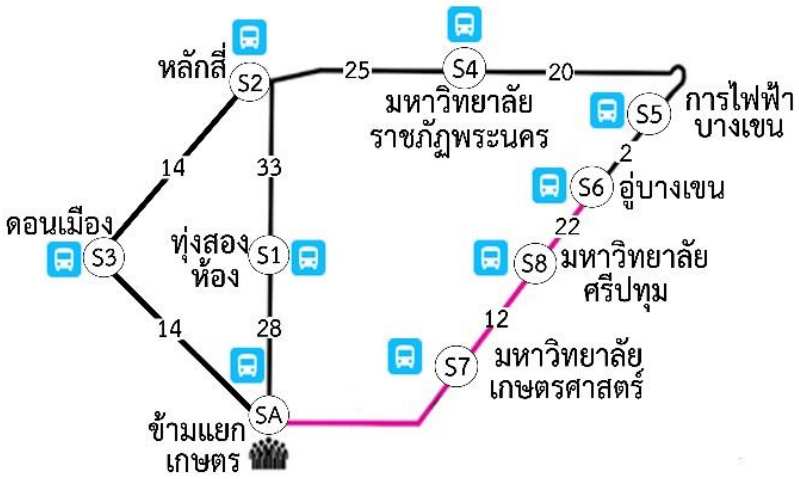
ภาพประกอบที่ 4.1 ตัวอย่างเส้นทางรถโดยสารขสมกสาย 207, 51ร, 522

ผู้วิจัยได้ทดสอบแบบจำลองที่ถูกพัฒนาขึ้น โดยใช้เส้นทางกรณีศึกษารถ ขสมก. รถประจำทางสาย 207, 51ร และ 522 ซึ่งมีตารางเวลาการเดินรถประจำทาง (ตารางที่ 4.2) และสร้างสถานการณ์จำลอง จำนวน 2 สถานการณ์ ได้แก่ จำลองสถานการณ์ปกติ (ใช้ทุกเส้นทางได้ตามปกติ) และ สถานการณ์ที่มีความเสี่ยงที่ทำให้ไม่สามารถใช้เส้นทางได้ สำหรับใช้ทดสอบการทำงานของอัลกอริทึมพลวัต เอ สตาร์ (ตารางที่ 4.2)

ตารางที่ 4.2 ตารางเวลาการเดินรถประจำทาง

รถประจำทางสายที่	**เส้นทางระหว่างโนด**	**Cost (C*)**
207	ข้ามแยกเกษตร (SA) – ทุ่งสองห้อง (S1)	28
207	ทุ่งสองห้อง (S1) – หลักสี่ (S2)	33
207	หลักสี่ (S2) - ดอนเมือง (S3)	14
207	ดอนเมือง (S3) - ข้ามแยกเกษตร (SA)	14
51ร	มหาวิทยาลัยราชภัฏพระนคร (S4) – การไฟฟ้าบางเขน (S5)	20
51ร	การไฟฟ้าบางเขน (S5) – อู่บางเขน (S6)	2
522	ข้ามแยกเกษตร (SA) – มหาวิทยาลัยเกษตรศาสตร์ (S7)	12
522	มหาวิทยาลัยเกษตรศาสตร์ (S7) – มหาวิทยาลัยศรีปทุม (S8)	26
522	มหาวิทยาลัยศรีปทุม (S7) – อู่บางเขน (S6)	22

จากตารางที่ 4.2 เส้นทางรถ ขสมก. รถประจำทางสาย 207 ขึ้นข้ามแยกเกษตร ลงต่อรถที่ป้ายรถเมล์หลักสี่ รถประจำทางสาย 51ร ลงอู่บางเขน ใช้เวลาในการเดินทาง 1 ชั่วโมง 5 นาที และ สายรถประจำทางสาย 522 ขึ้นข้ามแยกเกษตร ถึง อู่บางเขน ใช้เวลา 45 นาที ผู้วิจัยได้นำมาสร้างแบบจำลองเส้นทางลักษณะกราฟ (Graph Model) เพื่อใช้ในการทดสอบการทำงานของอัลกอริทึม ดังแสดงในภาพประกอบที่ 4.2



ภาพประกอบที่ 4.2 ตัวอย่างแบบจำลองเส้นทางรูปแบบกราฟที่ได้จากตารางเส้นทางการเดินรถประจำทาง

จากนั้นผู้วิจัยได้สร้างตารางการทดสอบที่มีการกำหนดค่าHeuristic Function และค่าถ่วงน้ำหนักปรับเชิงพลวัต Dynamic Adjustment Weight (λ) เข้าไปในส่วนตัวแปรค่าความเสี่ยงสำหรับทดสอบสถานการณ์จำลอง 2 รูปแบบ เพื่อใช้วัดและประเมินประสิทธิภาพของแบบจำลองที่ได้พัฒนาขึ้น ดังแสดงในตารางที่ 4.2

ตารางที่ 4.3 ค่าตัวแปรที่ถูกกำหนดขึ้นเพื่อใช้ทดสอบเส้นทางโดยสารรถประจำทางด้วยสถานการณ์จำลอง

ลำดับ	เส้นทาง	รถประจำทางสายที่	Cost (C*)	Heuristic (H*)	λ
1	SA-S1-S2	207	61	3	0
2	SA-S1-S2-S3	207	75	0	1
3	S2-S4-S5-S6	51ร	47	2	1
4	SA-S7-S8-S6	522	60	1	1

จากตารางที่ 4.3 กำหนดให้สถานีต้นทาง (SA) มีรถประจำทางสาย 207 กับ 522 และสถานีปลายทางสาย 522 กับ 51ร โดยเส้นทางที่ใช้ทดสอบได้แก่ เส้นทางที่สั้นที่สุด เริ่มต้น SA ไปยังปลายทาง S6 โดยผู้โดยสารสามารถเลือกเส้นทาง โดยใช้เส้นทางที่เหมาะสม เพื่อให้เดินทางจากต้นทางไปยังปลายทางได้ภายในเวลาที่ต้องการได้อย่างเหมาะสม โดยมีการกำหนดระดับค่าความเสี่ยงของเส้นทาง (H*) เช่น ความเสี่ยงจากจราจรแออัด ความเสี่ยงจากการปิดกันชั่วคราวหรือความเสี่ยง

จากธรรมชาติ เป็นต้น และมีการกำหนดค่าระยะทางจากสถานีจอดรถแต่ละโนด (C*) และกำหนดค่าตัวเลือกที่เหมาะสม (F*) เพื่อให้การคำนวณเส้นทางที่ถูกคัดเลือกด้วย อัลกอริทึม เอ สตาร์ เกิดประสิทธิภาพมากยิ่งขึ้น

4.3 ผลการทดสอบแบบ

ผลการทดสอบด้วยเส้นทางกรณีศึกษารถ ขสมก. ประจำทางสาย 207, 51ร และ 522 ด้วยสถานการณ์จำลอง 2 สถานการณ์ ได้แก่ จำลองสถานการณ์ปกติ (ใช้ทุกเส้นทางได้ตามปกติ) และสถานการณ์ที่มีความเสี่ยงที่ทำให้ไม่สามารถใช้เส้นทางได้ โดยนำเส้นทางที่แสดงในรูปแบบกราฟ (ภาพประกอบที่ 4.4) มาใช้ในการทดสอบการทำงานของอัลกอริทึมพลวัต เอ สตาร์ (ตารางที่ 4.4) พบว่า ผลการทดสอบด้วยสถานการณ์จำลองเป็นสถานการณ์ปกติของการโดยสารรถประจำทาง ซึ่งสามารถใช้ได้กับทุกเส้นทางโดยสารรถประจำทาง ซึ่งผลจากการทดสอบ อัลกอริทึมสามารถแสดงผลลัพธ์ได้ถูกต้องตามลำดับขั้นการทำงาน ดังแสดงในตารางที่ 4.4 ถึง 4.5
การค้นหาครั้งที่ 1 เริ่มจาก S โดยให้ค่า Heuristic ของปลายทาง (G*) มีค่าเท่ากับ 0

ตารางที่ 4.4 แสดงการค้นหาครั้งที่ 1 ในสถานการณ์ปกติ

ลำดับ	เส้นทาง	รถประจำทางสายที่	Cost (C*)	λ	Heuristic (H*)	Priority (F*)
1	SA-S1-S2	207	61	0	3	64
2	SA-S1-S2-S3	207	75	1	0	76
3	S2-S4-S5-S6	51ร	47	1	2	50
4	SA-S7-S8-S6	522	60	1	1	62

จากตารางที่ 4.3 เส้นทาง SA-S1-S2-S3 เส้นทางที่ H* มีค่าเป็น 0 เส้นทางนั้นไม่ได้ผ่านจุดปลายทางระบบจะตัดทิ้ง ดังนั้น ระบบจะทำการคัดเลือก SA S1 S2 ได้แก่ (1) (3) เส้นทางสาย S2-S6 ซึ่งวิ่งโดยรถสาย 51ร (4) SA-S6 ซึ่งโดยรถประจำทางสาย 522 เท่านั้น

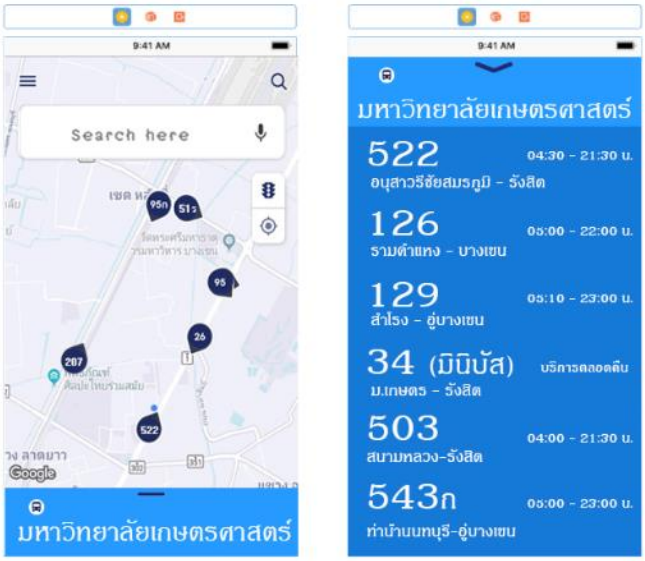
การค้นหาครั้งที่ 2 จะนำเส้นทางจากการค้นหาครั้งแรกที่สั้นที่สุดไปเชื่อมเส้นทางถัดไป โดยจะนำเส้นทางจากการค้นหาก่อนหน้า มาเปรียบเทียบ

ตารางที่ 4.5 แสดงการค้นหาครั้งที่ 2 ในสถานการณ์ปกติ

ลำดับ	เส้นทาง	รถประจำทางสายที่	Cost (C*)	λ	Heuristic (H*)	Priority (F*)
1	SA-S1-S2	207	61	0	3	64
2	S2-S4-S5-S6	51ร	47	1	2	50
3	SA-S7-S8-S6	522	60	1	1	62

จากตารางที่ 4.5 ผลจากการค้นหาตารางที่ 4 พบว่า เส้นทาง SA ปลายทาง S6 ถูกคัดเลือกโดยค่า F* ที่น้อยที่สุด ซึ่งมีค่าเท่ากับ 62 ดังนั้นระบบจะแนะนำให้ผู้โดยสารเลือกเส้นทางโดยสารประจำทางสาย 522 หรือหากเกิดกรณีที่สาย 522 เกิดขัดข้อง ระบบจะแนะนำสายทางเลือกสาย 207 ต่อรถ 51ร SA-S2 และ S2-S6 แทนที่เป็นลำดับถัดไป

ผลการทดสอบด้วยเส้นทางกรณีศึกษารถ ขสมก. ประจำทางสาย 207, 51ร และ 522 ได้ออกมาบนภาษา Swift



ภาพประกอบที่ 4.3 ผลการทดสอบด้วยเส้นทางบนภาษา Swift

บทที่ 5

สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ

5.1 สรุปผลการวิจัย

การศึกษาวิจัยเรื่อง “การพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์” เป็นการวิจัยเพื่อค้นหาและคัดเลือกเส้นทางที่เหมาะสม และพัฒนาระบบสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางในรูปแบบแอปพลิเคชัน ผลการวิจัยเป็นไปตามวัตถุประสงค์ของการพัฒนาแบบจำลองการเลือกเส้นทางโดยสารรถประจำทางรูปแบบพลวัตด้วยอัลกอริทึม เอ สตาร์ โดยผลการประเมินด้วยผู้เชี่ยวชาญ พบว่า มีค่าเฉลี่ยในระดับดี (ค่าเฉลี่ย ($\overline{X}$) = 4.30 และค่าเบี่ยงเบนมาตรฐาน S.D. = 0.273) เนื่องจากระบบได้ถูกออกแบบให้ผู้ใช้งานมีความคล่องตัวในการนำไปใช้และการคำนวณเพื่อระบุตำแหน่งเส้นทางปัจจุบันของรถโดยสารประจำทางมีความถูกต้องและมีการคัดเลือกเส้นทางการเดินทางได้อย่างเหมาะสม

5.2 อภิปรายผลการวิจัย

จากการศึกษาวิจัย การพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ มีความยืดหยุ่นในการคำนวณหาเส้นทาง โดยขั้นตอนจะนำค่าตัวแปรในการตัดสินใจ (Heuristic) และ ค่าความเป็นไปได้ในการใช้เส้นทางนั้น มาประยุกต์ใช้ให้แสดงผลลัพธ์ที่รวดเร็ว นอกจากนี้รูปแบบของ เอ สตาร์ อัลกอริทึมยังใช้ทรัพยากรจากอุปกรณ์ที่ทำงานร่วมกับโปรแกรมน้อยที่สุดเช่น สามารถนำไปประยุกต์ใช้กับเครื่องโทรศัพท์มือถือ เป็นต้น) หรือหากต้องทำการปรับปรุงค่าในการเลือกเส้นทางให้มีความยืดหยุ่นได้มากกว่า ค่า 0 (ศูนย์) กับ 1 ก็ยังสามารถนำไปพัฒนาต่อในรูปแบบของกฎเกณฑ์ (Rules) ของค่า Heuristic ได้ ประโยชน์ที่จะได้รับจากศึกษาวิจัยนี้สามารถนำไปประยุกต์ใช้ในสถานการณ์ที่ต้องการความรวดเร็วในการตัดสินใจ หรือ ไม่ต้องควบคุมการตัดสินใจด้วยมนุษย์ได้ เช่น การเลือกเส้นทางในการอพยพผู้คนในสถานการณ์ฉุกเฉิน หรือการเลือกเส้นทางที่เลี่ยงความเสี่ยงจากอุบัติเหตุ เป็นต้น การพัฒนาต้นแบบของกระบวนการ เป็นการนำเอาอัลกอริทึมในการค้นหาเส้นทางที่สั้นที่สุดมาประยุกต์ใช้กับระบบเทคโนโลยีสารสนเทศเพื่อหาเส้นทางที่เหมาะสมในการเดินทาง โดยในการวิจัยนี้ได้มีการเขียนโปรแกรมในรูปแบบของแอปปลิเคชั่นโดย

ใช้งานผ่านระบบปฏิบัติการ ไอ โอ เอส ส่วนเครื่องมือที่ใช้ในการพัฒนาระบบค้นหาต้นแบบนี้ใช้ภาษาสวิฟท์ ติดต่อกับฐานข้อมูลบนคลาวด์

5.3 ข้อเสนอแนะ

การพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ เป็นเพียงแบบจำลองการคัดเลือกเส้นทางเพื่อสนับสนุนทฤษฎีประยุกต์สำหรับงานวิจัยเท่านั้น ผู้วิจัยมีข้อเสนอแนะแนวทางพัฒนารูปแบบต่าง ๆ ดังนี้

1. งานวิจัยนี้สามารถนำไปประยุกต์ใช้เพื่อรองรับการใช้งานบนระบบปฏิบัติการแอนดรอยด์ (Android) ได้

2 การนำองค์ความรู้ด้านปัญญาประดิษฐ์มาประยุกต์ใช้ร่วมกับการประมวลผลที่ซับซ้อน เช่น การคำนวณเพื่อหาช่วงเวลาเฉลี่ยของการรอรถโดยสารประจำทาง กรณีสภาพอากาศ หรือช่วงเวลาที่มีการจราจรแออัด จะช่วยให้ฟังก์ชั่นการทำงานของระบบมีความแม่นยำเพิ่มขึ้น

5.4 ปัญหา และอุปสรรค

ปัญหา และอุปสรรคจากข้อจำกัดเรื่องระยะเวลาที่ใช้สำหรับงานวิจัย และ ปัญหาการเก็บรวบรวมข้อมูลจากผู้ใช้งาน ซึ่งต้องขออนุมัติหน่วยงานหลายฝ่าย ทำให้เสียเวลาในการค้นคว้าหาข้อมูลและต้องลองผิดลองถูกอยู่เป็นเวลานานกว่าที่ผู้พัฒนาระบบจะได้เริ่มพัฒนาระบบจริง

บรรณานุกรม

Ahmed Ahmed, Elshaimaa Nada, Wafaa Al- Mutiri. (2017). **University buses routing and tracking system. International Journal of Computer Science & InformationTechnology (IJCSIT).** 9(1) : 95-104

Ahmed Ahmed, Elshaimaa Nada, Wafaa Al-Mutiri, **UNIVERSITY BUSES ROUTING AND TRACKING SYSTEM**, International Journal of Computer Science & Information Technology (IJCSIT) Vol 9, No 1, February 2017

AppCoda, **Google Maps SDK**, วันที่สืบค้นวันที่ 24 พฤศจิกายน พ.ศ. 2560, จากเว็บไซต์ https://translate.google.co.th/translate?hl=th&sl=en&u=https://www.appcoda.com/google-maps-api-tutorial/&prev=search

Apple (Thailand), **iphone**, วันที่สืบค้นวันที่ 24 พฤศจิกายน พ.ศ. 2560, จากเว็บไซต์ https://www.apple.com/th/iphone/

Exteen.com, **xcode**, วันที่สืบค้นวันที่ 24 พฤศจิกายน พ.ศ. 2560, จากเว็บไซต์ http://takato.exteen.com/20110826/xcode

Global5 co.,LTD., **ความรู้ทั่วไปเกี่ยวกับ GPS. Retrieved,** วันที่สืบค้นวันที่ 2 มกราคม พ.ศ. 2561, จากเว็บไซต์ http://www.global5thailand.com/thai/gps.htm

João Paulo Lima, Rafael Roberto, Francisco Simões, Mozart Almeida, Lucas Figueiredo, João Marcelo, **Markerless tracking system for augmented reality in the automotive industry**, 2017

JSO Software, **ตัวอย่างของสัญกรณ์บิ๊ก O: f**, วันที่สืบค้นวันที่ 2 มกราคม พ.ศ. 2561, จากเว็บไซต์ https://www.jsosoft.com/dev/computational-complexity-reference/

Lane B. W. (2010). **The relationship between recent gasoline price fluctuations and transit ridership in major US cities.** Journal of Transport Geography 18: 214-225.

Paul J. Burke and Nishitateno, Shuhei. (2011). **Gasoline prices, gasoline consumption, and new-vehicle fuel economy: Evidence for a large sample of countries.** Australian National University Coombs Building 9.

Rujuta Kharade, Supriya Kulkarni, Madhura Thorat, Sneha Walhekar, **GPS Assisted Location Based Service for Smart City with Cloud Integration, JIREEICIJIICE,** Vol. 5, Issue 4, April 2017

Swiftlet, **GOOGLE MAP API**, วันที่สืบค้นวันที่ 24 พฤศจิกายน พ.ศ. 2560, จากเว็บไซต์ https://swiftlet.co.th

Wikipedia, **Big O notation,** วันที่สืบค้นวันที่ 2 มกราคม พ.ศ. 2561, จากเว็บไซต์ https://en.wikipedia.org/wiki/Big_O_notation

เกร็ดความรู้.net. (2559). **GPS คืออะไร ย่อมาจากอะไร มีกี่ประเภท ประโยชน์ของ GPS**, วันที่สืบค้นวันที่ 13 พฤศจิกายน พ.ศ.2560, จากเว็บไซต์ http://www.xn--12cg1cxchd0a2gzc1c5d5a.net/gps/

ชนินทร์ เขียวสนั่น. (2547**). การส่งเสริมระบบขนส่งมวลชนในเขตเมืองชั้นใน กรณีศึกษาพฤติกรรมการเดินทางของผู้ใช้รถยนต์ส่วนบุคคลในย่านธุรกิจถนนสีลม.** วิทยานิพนธ์ปริญญามหาบัณฑิต, สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง.

ไทยครีเอทบริการ, (2557) **โครงสร้างของ Xcode IDE และ ไฟล์ที่ได้จากการ Create Project บน Xcode (iOS, iPhone, iPad),** วันที่สืบค้นวันที่ 24 พฤศจิกายน พ.ศ. 2560, จากเว็บไซต์ http://www.thaicreate.com/mobile/xcode-structure-project-ios-iphone-ipad.html

ธเนศ ขุมทรัพย์. (2549). **ความสัมพันธ์ระหว่างค่าใช้จ่ายด้านที่อยู่อาศัยค่าใช้จ่ายในการเดินทางเข้าสู่แหล่งงาน และที่ตั้ง ที่อยู่อาศัย :กรณีศึกษา ผู้ที่ทางานในอาคารสานักงานย่านถนนสาทร.** วิทยานิพนธ์ปริญญามหาบัณฑิต, จุฬาลงกรณ์มหาวิทยาลัย.

นางสาวจิราพัชร กันหา, **วงจรชีวิตของการพัฒนาซอฟต์แวร์ SDLC**, วันที่สืบค้นวันที่ 25 พฤศจิกายน พ.ศ. 2560, จากเว็บไซต์ https://sites.google.com/site/jjiraphatkanha/wngcr-chiwit-khxng-kar-phathna-sxftwaer-sdlc

นายพีระเดช สำรวมรัมย์. (2015) **การพัฒนาระบบติดตามพิกัดผู้ป่วยกลุ่มเสี่ยง STROKE and STEMI ที่ต้องได้รับบริการจากระบบบริการการแพทย์ฉุกเฉินแบบเคลื่อนที่, พยาบาลวิชาชีพชานาญการ โรงพยาบาลสร้างคอม จังหวัดอุดรธานี**, นักศึกษาหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาการจัดการเทคโนโลยีสารสนเทศ มหาวิทยาลัยรังสิต.

บริษัท อัพบีน จำกัด, **Mobile Application คือ**, วันที่สืบค้นวันที่ 25 พฤศจิกายน พ.ศ. 2560, จาก http://www.admissionpremium.com/it/news/1852

ปราชญา สมชื่อ, ภูมิรพี สืบวงษ์รอด. (2557) **ระบบติดตามรถยนต์ด้วยจีพีเอสผ่านแอพพลิเคชั่นบนระบบปฏิบัติการแอนดรอยด์**, วิศวกรรมศาสตรบัณฑิตสาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา ตาก.

พีระเดช สารวมรัมย์.(2558). **การพัฒนาระบบติดตามพิกัดผู้ป่วยกลุ่มเสี่ยง STROKE and STEMI ที่ต้องได้รับบริการจากระบบบริการการแพทย์ฉุกเฉินแบบเคลื่อนที่.** วิทยาศาสตรมหาบัณฑิต, มหาวิทยาลัยรังสิต.

เพชรไพลิน รัสดีดวง, และกฤติกา สังขวดี, ปัญญา สังขวดี. **แอปพลิเคชันส่งเสริมกีฬาวอลเลย์บอลบนระบบปฏิบัติการแอนดรอยด์.** การประชุมวิชาการระดับชาติราชภัฏเพชรบุรีวิจัยศิลปวัฒนธรรม ครั้งที่ 4. เพชรบุรีฯ: มหาวิทยาลัยราชภัฏเพชรบุรี

ภาษา Swift Tutorial พื้นฐาน 1 (Variable และ Constant). (2014). Retrieved from http://www.swift-tutor.com/project/swift-tutorial-พื้นฐาน-1-variable-constant

วฤชุตา นันตะพานา. (2558). **เครื่องมือตรวจสอบข้อปฏิบัติการตั้งชื่อในโปรแกรมอ็อบเจกทีฟซี**, วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย.

วิภาดา เพชรรัตน์. (2554). **ระบบนาทางอัจฉริยะ : กรณีศึกษา บริษัทวานิชรุ่งเรืองอินเตอร์เทรด จากัด (สาขาภาคใต้).** วิทยาศาสตรมหาบัณฑิต, มหาวิทยาลัยสงขลานครินทร์.

วีรชัย สว่างทุกข์. (2557). **การใช้ระบบติดตามจีพีเอสแบบเปิดเผยรหัสต้นฉบับควบคู่สมาร์ตโฟนเพื่อใช้ติดตามรถขนส่ง กรณีศึกษาน้าดื่มทิพย์เขลางค์.** วารสารวิชาการคณะเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยราชภัฏลาปาง. 7(1): 40-51

วีรพันธ์ รุจิเกียรติกาจร และวีรินทร์ หวังจิรนิรันดร์. (2556**). การศึกษาค่าใช้จ่ายในการเดินทางและค่าใช้จ่ายด้านพลังงานของการเดินทางแต่ละรูปแบบ: กรณีศึกษาเส้นทางสะพานใหม่-สีลม.** วารสารวิจัยพลังงาน, 10(1): 1-11.

สถาบันรับรองมาตรฐานไอเอสโอ, **ISO 29110 (Software Engineering-Lifecycle Profiles For Very Small Enterprises (VSE)**, วันที่สืบค้นวันที่ 24 พฤศจิกายน พ.ศ. 2560, จากเว็บไซต์ http://masci.or.th/iso-29110-software-engineering-lifecycle-profiles-for-very-small-enterprises-vse/

สัญญา เครือหงษ์ และสุรางค์รัตน์ เชาว์โคกสูง. (2551). **การพัฒนาเกมแนวต่อสู้โดยใช้ชุดคาสั่งซีดีเอ็กซ์เอนจินและอัลกอริทึมแบบ เอ-สตาร์.** วารสารเทคโนโลยีสารสนเทศ มหาวิทยาลัยพระจอมเก้าพระนครเหนือ. 4(7): 38-46.

สุขสวัสดี ณัฏฐวุฒิสิทธิ์. (2559). **การวิเคราะห์ออกแบบและการสร้างระบบ**, มหาวิทยาลัยศรีปทุม.

สุขสวัสดี ณัฏฐวุฒิสิทธิ์ และประสงค์ ปราณีตพลกรัง. (2554). **การพัฒนาแบบจาลองการจัดเส้นทางการเดินทางที่เหมาะสมในรูปแบบพลวัตของอัลกอริทึท เอ สตาร์.** ในการประชุมทางวิชาการระดับชาติด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศ ครั้งที่ 7. กรุงเทพฯ: คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.

สุรเชษฐ์ กานต์ประชา และเศรษฐา ตั้งค้าวานิช. (2557). **ระบบติดตามและประมาณเวลาการเดินรถไฟฟ้าด้วยสมาร์ทโฟนผ่านเครือข่าย 3G**. โครงการวิจัยคณะวิศวกรรมศาสตร์, มหาวิทยาลัยนเรศวร.

อ.นภัทร รัตนนาคินทร์, **วงจรการพัฒนาระบบหรือวัฏจักรของการพัฒนาระบบ (System Development Life Cycle : SDLC)**, วันที่สืบค้นวันที่ 12 พฤศจิกายน พ.ศ. 2560, จากเว็บไซต์ http://www.macare.net/analysis/index.php?id=-8

อรอนงค์ กฤตยาเกียรณ์. (2545). **การจัดทำพื้นที่จอดรถยนต์เพื่อสนับสนุนโครงการระบบขนส่งมวลชน.** วิทยานิพนธ์ปริญญามหาบัณฑิต, จุฬาลงกรณ์มหาวิทยาลัย.

ภาคผนวก ก
คู่มือการการใช้งาน

คู่มือการใช้งาน

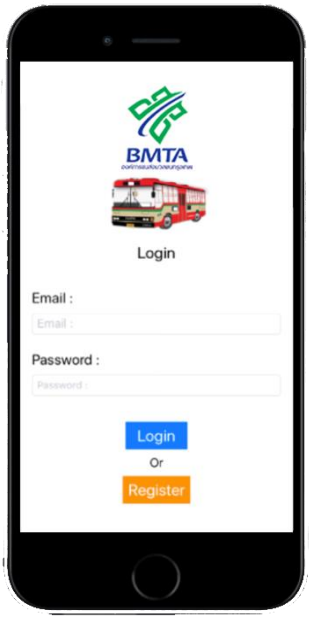
การศึกษาและวิจัยครั้งนี้มีวัตถุประสงค์เพื่อการพัฒนาไอ โอ เอส แอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางรูปแบบพลวัตของอัลกอริทึม เอ สตาร์ ด้วยภาษาสวิฟท์บนคลาวด์ ผู้ศึกษาได้นำองค์ความรู้ต่าง ๆ มาใช้ประกอบพิจารณาเพื่อสนับสนุนแอปพลิเคชัน โดยมีรายละเอียดขั้นตอนการศึกษาดังนี้

การใช้งานแอปพลิเคชัน

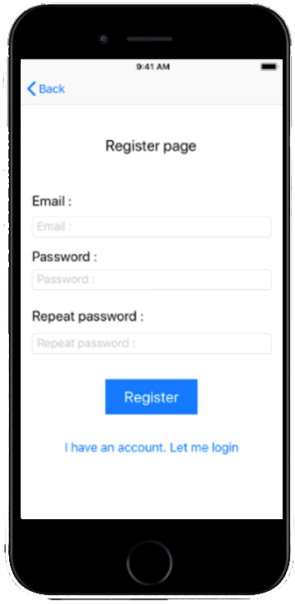
เริ่มต้นใช้งานเริ่มต้นการใช้งานตัวแอปพลิเคชันสำหรับบริหารจัดการเส้นทางการเดินรถโดยสารประจำทางบนระบบปฏิบัติการ iOS iPhone 5 เวอร์ชั่นตั้งแต่ 10.3.3, Apple Watch เวอร์ชั่นตั้งแต่ watchOS 3.2.3, macOS Sierra เวอร์ชั่นตั้งแต่ 10.12.6 ขึ้นไป โดยในบทนี้ จะอธิบายถึงการทางานในเบื้องต้นของการใช้โปรแกรม ดังรายละเอียดต่อไปนี้

การเริ่มเปิดโปรแกรม

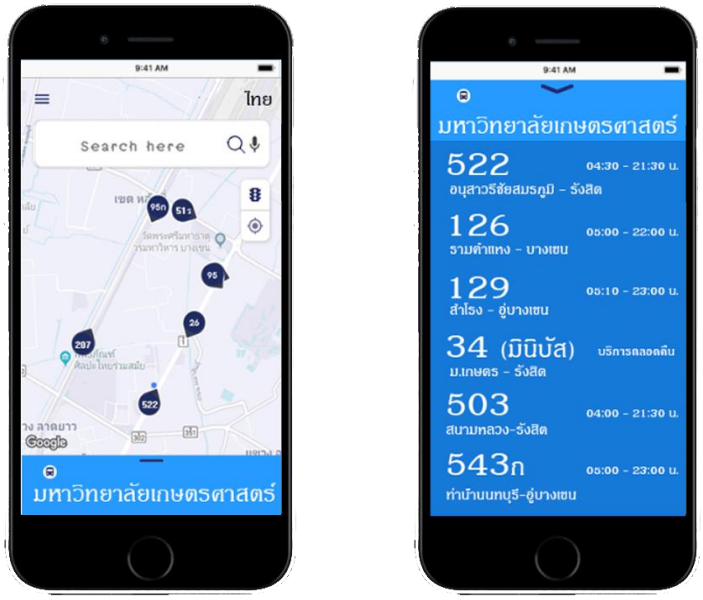
เมื่อเริ่มเปิดใช้งานตัวโปรแกรมก็จะพบกับหน้าจอให้ทำการ login เข้าสู่ระบบ โดยมีขั้นตอนการใช้งาน ดังต่อไปนี้



ระบบจะให้กรอกข้อมูลอีเมล์ (Email) ใส่ข้อมูลรหัสผ่าน (password) เพื่อทำการ เข้าสู่ระบบ หากยังไม่ได้ลงทะเบียนให้ทำการลงทะเบียนก่อน โดยการกด “Register” ระบบจะเปลี่ยนหน้าไปหน้าลงทะเบียน



ระบบจะให้กรอกข้อมูลอีเมล์ (Email) ใส่ข้อมูลรหัสผ่าน (Password) ยืนยันรหัสผ่าน (Repeat password) เมื่อทำการกรอกรายละเอียดเสร็จให้กดที่ “Register” เพื่อทำการลงทะเบียน จากนั้นระบบจะกลับมาที่หน้าจอ Login กรอก อีเมล์ (Email) กรอก รหัสผ่าน (Password) แล้วกดที่ปุ่ม Login



ระบบจะมาที่หน้าจอที่มีการหาสายรถโดยสารประจำทาง หน้าจอจะบอกตำแหน่งที่ผู้ใช้งานอยู่ เลื่อน ขึ้น จะพบสายรถโดยสารประจำทางที่วิ่งอยู่ในตำแหน่งที่ผู้ใช้งานอยู่ เลือกสายรถที่ต้องการ ระบบจะคำนวณเส้นทาง และคำนวณเวลาการมาถึงของรถโดยสารประจำทาง

ภาคผนวก ข

หนังสือตอบรับ บทความตีพิมพ์ผลงานวารสารวิชาการซายน์เทค มรภ.ภูเก็ต คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏภูเก็ต, ปีที่ 2 (1)

ที่ ศธ ๐๕๕๘/พิเศษ

มหาวิทยาลัยราชภัฏภูเก็ต
อ.เมือง จ.ภูเก็ต ๘๓๐๐๐

๑๔ มิถุนายน ๒๕๖๑

เรื่อง แจ้งผลการพิจารณาบทความเพื่อจัดพิมพ์เผยแพร่ในวารสารวิชาการชายน์เทค มรภ.ภูเก็ต

เรียน คุณกมลวรรณ พงษ์ศิลป์

ตามที่ท่านได้ส่งบทความเรื่อง **"การพัฒนาแบบจำลองสารสนเทศเพื่อเลือกเส้นทางโดยสารรถประจำทางบนพื้นฐานอัลกอริทึมพลวัต เอ สตาร์"** เพื่อพิจารณาจัดพิมพ์เผยแพร่ในวารสารวิชาการชายน์เทค มรภ.ภูเก็ต คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏภูเก็ต ปีที่ ๒ ฉบับที่ ๑ เดือนมกราคม – มิถุนายน ๒๕๖๑ นั้น

ในการนี้ กองบรรณาธิการวารสารวิชาการชายน์เทค มรภ.ภูเก็ต ขอเรียนให้ทราบว่าผู้ทรงคุณวุฒิได้พิจารณาบทความของท่านแล้ว เห็นสมควรให้ตีพิมพ์เผยแพร่บทความดังกล่าวในวารสารวิชาการชายน์เทค มรภ.ภูเก็ต ปีที่ ๒ ฉบับที่ ๑ เดือนมกราคม – มิถุนายน ๒๕๖๐

จึงเรียนมาเพื่อโปรดทราบ

ขอแสดงความนับถือ

(ผู้ช่วยศาสตราจารย์ ดร.วิภาวรรณ บัวทอง)
คณบดีคณะวิทยาศาสตร์และเทคโนโลยี
ปฏิบัติราชการแทน อธิการบดีมหาวิทยาลัยราชภัฏภูเก็ต

สำนักงานคณบดี คณะวิทยาศาสตร์และเทคโนโลยี
โทรศัพท์ / โทรสาร ๐๗๖ ๒๑๘ ๘๐๖
ผู้ประสานงาน นางสาวสุธิดา รัตนบุรี โทรศัพท์ ๐๘๑ ๐๙๙ ๖๐๔๖

การพัฒนาแบบจำลองสารสนเทศเพื่อเลือกเส้นทางโดยสารรถประจำทาง บนพื้นฐานอัลกอริทึมพลวัต เอ สตาร์

THE DEVELOPMENT OF BUS ROUTING SELECTION BASED ON DYNAMIC A* ALGORITHM

กมลวรรณ พงษ์ศิลป์[1]* และสุขสวัสดี ณัฏฐวุฒิสิทธิ์[2]

[1,2] สาขาวิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม

2410/2 ถ.พหลโยธิน เขตจตุจักร กรุงเทพฯ 10900

บทคัดย่อ

งานวิจัยนี้เป็นงานวิจัยเชิงทดลอง มีวัตถุประสงค์เพื่อศึกษาปัญหาและแนวทางการพัฒนากลไกการเลือกเส้นทางโดยสารรถประจำทาง โดยผู้วิจัยได้นำแนวคิดการคัดเลือกเส้นทางด้วยอัลกอริทึมพลวัต เอสตาร์มาใช้เพื่อค้นหาและคัดเลือกเส้นทางที่เหมาะสมแก่ผู้ใช้งาน ทำให้ช่วยคำนวณเวลา และวางแผนการเดินทางได้อย่างมีประสิทธิภาพ เครื่องมือที่ใช้ในงานวิจัย ได้แก่ แบบจำลองสารสนเทศเพื่อเลือกเส้นทางโดยสารรถประจำทางที่ถูกพัฒนาขึ้นด้วยภาษาสวิฟท์บนพื้นฐานอัลกอริทึมพลวัต เอสตาร์ กลุ่มตัวอย่าง ได้แก่ ข้อมูลการโดยสารรถประจำทางองค์การขนส่งมวลชนกรุงเทพ (ขสมก.) เส้นทางสาย 207, 51ร, 522 ประกอบด้วย (1) ค่าความเสี่ยงของเส้นทาง (2) ค่าความเหมาะสมของจุดขึ้น-ลงรถโดยสารประจำทาง และ (3) ค่านัยสำคัญของเส้นทางการโดยสารรถประจำทาง สำหรับใช้คำนวนหาเส้นทางที่เหมาะสม ผลการวิจัย พบว่า ผลลัพธ์จากการประเมินโดยผู้เชี่ยวชาญมีค่าเฉลี่ยในระดับดี (ค่าเฉลี่ย ($\overline{X}$) = 4.30 และค่าเบี่ยงเบนมาตรฐาน S.D. = 0.273) ประโยชน์ที่ได้รับจากงานวิจัยนี้สามารถนำไปประยุกต์ใช้กับการคัดเลือกเส้นทางโดยสารรถประจำทางอื่น ๆ รวมทั้งองค์ความรู้นี้ยังสามารถนำไปขยายผลทางวิชาการได้ต่อไปในอนาคต

คำสำคัญ : รถประจำทาง เส้นทาง อัลกอริทึม เอ สตาร์

Abstract

This research is experimental research. The purpose of this study was to investigate the problems and ways to develop a bus route selection mechanism. The researcher has adopted the concept of route selection with dynamic algorithms. Aster Star is used to find and select the appropriate route for the user. Help to calculate time And plan to travel efficiently. The research tool was a traffic information model developed by Swift, based on the Astartes dynamic algorithm. The samples were bus traffic data, mass

transit (Routes 207, 51ร, 522 consist of: (1) Route risk; (2) Approach to take-off bus; and (3) Significance of route. Routes The results showed that the results from the expert evaluation were good (mean ($\bar{X}$) = 4.30 and standard deviation S.D. = 0.273). This research can be applied to the selection of other bus routes as well as this knowledge can be extended to further academic results in the future.

Keyword : Bus, Route, Algorithm, A star

---

1. บทนำ

ยุคประเทศไทย 4.0 รูปแบบกิจกรรมการดำเนินชีวิตประจำวัน และการทำธุรกิจของประชาชนได้เกิดการเปลี่ยนแปลงไปจากอดีตมาก ในปัจจุบันประชาชนมีการเดินทางติดต่อระหว่างกันตลอดเวลา [1] ซึ่งหนึ่งในรูปแบบกิจกรรมที่เป็นปัจจัยสำคัญต่อการดำเนินชีวิตได้แก่ การเดินทางไม่ว่าจะเป็นการขนส่งสินค้าและบริการการติดต่องาน หรือ การเดินทางท่องเที่ยวในสถานที่ต่าง ๆ [2] โดยเฉพาะในเมืองใหญ่ เช่น กรุงเทพมหานคร มีประชาชนที่ใช้บริการเส้นทางโดยสารด้วยระบบขนส่งสาธารณะในแต่ละวันเป็นจำนวนมากโดยเฉพาะรถโดยสารประจำทางขององค์การขนส่งมวลชนกรุงเทพ (ขสมก.) ซึ่งได้ถูกนำมาให้บริการแก่ประชาชนและช่วยแก้ไขปัญหาการจราจรได้อย่างดี

รถโดยสารประจำทางขององค์การขนส่งมวลชนกรุงเทพ (ขสมก.) เป็นการให้บริการขนส่งสาธารณะรูปแบบหนึ่งที่สำคัญสำหรับประชาชนไทย และมีบทบาทต่อการพัฒนาเศรษฐกิจและสังคมของประเทศ [3] ซึ่งรัฐบาลได้ให้ความสำคัญเป็นอันดับต้นของการพัฒนาประเทศ [4] แต่อย่างไรก็ดีจากการสำรวจทางสถิติ [5] พบว่าการให้บริการรถโดยสารขสมก.บางเส้นทางยังไม่สามารถอำนวยความสะดวกแก่ผู้โดยสารได้อย่างเพียงพอ ต้นเหตุของปัญหาส่วนใหญ่เกิดจากรถโดยสารประจำทางไม่สามารถกำหนดตารางเวลาที่แน่นอนของการเข้าสถานีจอดรถได้เพราะกรุงเทพมหานครมีสภาพการจราจรติดขัด [6] ส่งผลให้ความถี่ของการปล่อยรถโดยสารประจำทางบางเส้นทางเกิดการขาดช่วง จึงทำให้ผู้โดยสารต้องรอรถเป็นระยะเวลานานเนื่องจากไม่สามารถทราบกำหนดการที่แน่นอนของรถ [7]

ดังนั้นจากปัญหาดังกล่าว ผู้วิจัยได้พัฒนาแบบจำลองสารสนเทศสำหรับการคัดเลือกเส้นทางโดยสารรถประจำทางที่เหมาะสมบนพื้นฐานอัลกอริทึมพลวัต เอ สตาร์ เพื่อให้ผู้ใช้งานสามารถบริหารจัดการเส้นทางและคำนวณเวลา ประโยชน์ที่คาดจะได้รับครั้งนี้จะช่วยลดปัญหาการรอคิวรถโดยสารประจำทางเป็นเวลานานโดยผู้ให้และผู้รับบริการสามารถทราบกำหนดเวลาการเข้าจอดรถได้โดยประมาณ ทำให้สามารถวาง

แผนการเดินทางของตนเองได้ นอกจากนี้องค์ความรู้ด้านสารสนเทศนี้ยัง

สามารถนำไปประยุกต์ใช้ในระบบองค์การขนส่งมวลชนกรุงเทพ ระบุถึงเวลาการมาถึงของรถ ขสมก. ให้เกิดประสิทธิภาพมากขึ้นในอนาคต[8], [9]

2. วัตถุประสงค์

1. เพื่อศึกษาปัญหาและแนวทางการพัฒนาแบบจำลองการเลือกเส้นทางโดยสารรถประจำทางด้วยอัลกอริทึมพลวัต เอ สตาร์
2. เพื่อการพัฒนาแบบจำลองการเลือกเส้นทางโดยสารรถประจำทางด้วยอัลกอริทึมพลวัต เอ สตาร์

* *pbnyo100732@gmail.com*

3. วิธีดำเนินการวิจัย

ผู้วิจัยได้ใช้ระเบียบวิธีวิจัยเชิงทดลอง โดยพัฒนาแบบจำลองการคัดเลือกเส้นทางโดยสารรถประจำทางขสมก. สาย 207, สาย 51ร, และสาย 522 ที่มีเส้นทางการเดินรถโดยสารเชื่อมต่อกัน โดยมีจุดเริ่มต้นที่สถานีจอดรถโดยสารสาย 207 และ จุดปลายทางที่สถานีจอดรถโดยสารสาย 522 ดังแสดงในภาพที่ 1

ภาพที่ 1 ตัวอย่างเส้นทางรถโดยสารขสมก สาย 207, 51ร, 522

3.1 การออกแบบและพัฒนาแบบจำลอง

การทำงานของแบบจำลองสารสนเทศด้วยอัลกอริทึมพลวัต เอ สตาร์ มีขั้นตอนวิธีการทำงาน ดังสมการ (1) และมีเพิ่มค่าถ่วงน้ำหนักปรับเชิงพลวัต Dynamic Adjustment Weight (λ) เข้าไปในส่วนตัวแปรค่า Heuristic Function เพื่อเพิ่มความยืดหยุ่นของการเลือกเส้นทาง ดังสมการ (2)

$$F(n) = H(n) + C(n) \qquad (1)$$

จากสมการ (1) เปลี่ยนเป็นแบบพลวัตด้วยการแทรกค่าถ่วงน้ำหนักปรับเชิงพลวัต Dynamic Adjustment Weight (λ) ดังสมการ (2)

$$F^*(n) = (1+\lambda)\, H^*(n) + C^*(n) \qquad (2)$$

$F^*(n)$ = ค่าความเหมาะสมของการคัดเลือกเส้นทางจากโหนดต้นทางไปยังโหนดปลายทาง

$C^*(n)$ = ค่าใช้จ่ายในการเดินทางในแต่ละโหนดย่อย

$H^*(n)$ = ค่าฮิวริสติกที่ใช้ในการคัดเลือกเส้นทางในแต่ละโหนดย่อยว่าเหมาะสมในการเดินทางหรือไม่

λ = ค่าถ่วงน้ำหนักปรับเรียบเชิงพลวัตของสถานะเส้นทาง ซึ่งมีค่าเท่ากับ 0 และ 1 โดยเส้นทางสถานะไม่ปกติ = 0 และเส้นทางเสถานะปกติ =1 เพื่อแสดงให้เห็นว่าเส้นทางที่มีความเสี่ยงจะถูกนำมาใช้คำนวณ ในสมการ (2) ให้มีค่าลำดับในการเข้ารับการตรวจหาโหนด (Priority) มีค่ามากกว่าเส้นทางปกติ 2 เท่า

เพื่อให้ค่าน้ำหนักในการตัดสินสูง และอัลกอริทึมจะไม่เลือกไปยังเส้นทางนั้น ๆ

โดยหลักการทำงานของขั้นตอนวิธีการค้นหาและคัดเลือกเส้นทางด้วยอัลกอริทึมพลวัต เอ สตาร์จะให้ความสำคัญต่อการค้นหาเส้นทางที่สั้นที่สุดให้ก่อน (ในกรณีที่มีเส้นทางที่สั้นที่สุด) แต่หากพบว่าเส้นทางที่พบนั้นมีค่าความเสี่ยงที่ถูกระบุไว้ในตารางการกำหนดค่า(เช่น ความเสี่ยงของการจราจรติดขัด เส้นทางถูกปิดใช้ชั่วคราว หรือ ไม่สามารถใช้สัญจรได้) อัลกอริทึมจะใช้การค้นหาเส้นทางอื่นที่เหมาะสมที่สุดแทน ทำให้การค้นหาและถูกคัดเลือกเส้นทางโดยใช้มีความยืดหยุ่นและความเหมาะสมมากขึ้นสำหรับขั้นตอนการทำงานของอัลกอริทึมพลวัต แบบ เอ สตาร์ แสดงในรูปแบบรหัสเทียม (Psudo code) ดังภาพที่ 2

```
A* (start, goal)
1.  Closed set = the empty set
2.  Open set = includes start node
3.  G[start] = 0, H[start] = H_calc[start, goal]
4.  F[start] = H[start]
5.  While Open set ≠ Ø
6.  Do CurNode ← EXTRACT-MIN-F(Open set)
7.  if ( CurNode == goal ), then return BestPath
8.  For each Neighbor Node N of CurNode
9.  If ( N is in Closed set ), then Nothing
10. else if ( N is in Open set ),
11. calculate N's G, H, F
12. If ( G[N on the Open set] > calculated G[N] )
13. RELAX(N, Neighbor in Open set, w)
14. N's parent=CurNode& add N to Open set
15. else, then calculate N's G, H, F
16. N's parent = CurNode& add N to Open
```

ภาพที่ 2 แสดง Pseudo Code อัลกอริทึมแบบ เอ สตาร์ [9]

ผู้วิจัยได้นำขั้นตอนวิธีการทำงานของอัลกอริทึมพลวัต เอ สตาร์นี้มาประยุกต์ใช้กับการออกแบบจำลองสารสนเทศเพื่อคัดเลือกเส้นทางที่เหมาะสมของการโดยสารรถประจำทาง ขสมก. สาย 207, สาย 51ร, และ สาย 522 ที่มีเส้นทางการเดินรถโดยสารเชื่อมต่อกัน เพื่อช่วยลดระยะเวลาของการทำงานในภาพรวมซึ่งมีรายละเอียดของขั้นตอนแสดงในภาพที่ 3 ดังนี้

1) เริ่มต้นจากอัลกอริทึมตรวจสอบเส้นทางที่ต้องการค้นหาจากฐานข้อมูลเดิมที่เคยถูกเรียกใช้ หากพบว่าเคยมีการเรียกใช้ก่อนหน้าอัลกอริทึมจะแสดงผลลัพธ์ออกมาให้ผู้ใช้ทราบและบันทึกลงฐานข้อมูล

2) หากพบว่ายังไม่มีการค้นหาของเส้นทางที่ต้องการในฐานข้อมูล อัลกอริทึมจะเริ่มต้นค้นหาตามรูปแบบพลวัต เอ สตาร์

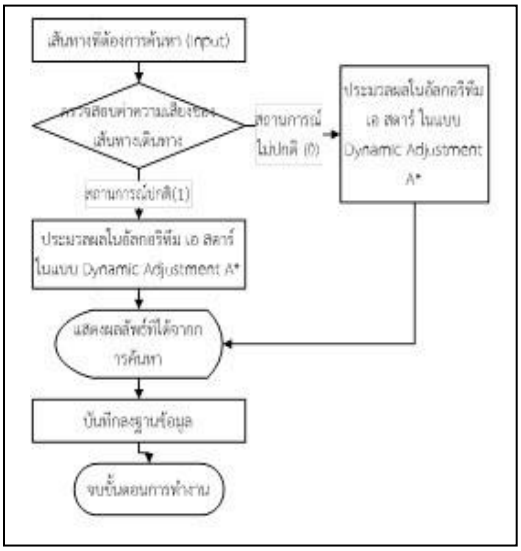
3) จากนั้นอัลกอริทึมทำการเช็คค่าความเสี่ยงของเส้นทางที่เลือกมานั้นว่ามีสถานะเป็นอะไร

$\lambda = 0$ ไม่สามารถเดินทางได้ สถานการณ์ไม่ปกติ (0)

$\lambda = 1$ ไม่สามารถเดินทางได้ สถานการณ์ปกติ (1)

4) หากอัลกอริทึมพบว่ามีเส้นทางใดเส้นทาง หนึ่งไม่สามารถเดินทางได้ ($\lambda = 0$) อัลกอริทึมจะทำการคัดเลือกเส้นทางอื่นที่แทน

5) หากอัลกอริทึมไม่พบเส้นทางที่จะเดินทางได้อัลกอริทึมจะเสนอเส้นทางที่เหมาะสมอื่นให้แทน



ภาพที่ 3 ขั้นตอนการทำงานอัลกอริทึมพลวัต เอ สตาร์

สำหรับในส่วนผู้ใช้งาน ผู้วิจัยออกแบบจำลองผู้ใช้งาน (Use Case View) ด้วยภาษาสวิฟท์ โดยใช้ XCODE 5.0 บน MAC OSX 10.0 เป็นเครื่องมือพัฒนา ดังภาพที่ 4

ภาพที่ 4 ตัวอย่างแบบจำลองผู้ใช้งาน

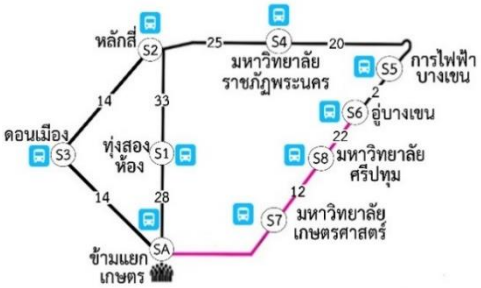
3.2 การทดสอบแบบจำลอง

ผู้วิจัยได้ทดสอบแบบจำลองที่ถูกพัฒนาขึ้นโดยใช้เส้นทางกรณีศึกษารถ ขสมก. รถประจำทางสาย 207, 51ร และ 522 ซึ่งมีตารางเวลาการเดินรถประจำทาง (ตารางที่ 1) และสร้างสถานการณ์จำลอง จำนวน 2 สถานการณ์ ได้แก่ จำลองสถานการณ์ปกติ (ใช้ทุกเส้นทางได้ตามปกติ) และ สถานการณ์ที่มีความเสี่ยงที่ทำให้ไม่สามารถใช้เส้นทางได้ สำหรับใช้ทดสอบการทำงานของอัลกอริทึมพลวัต เอ สตาร์

ตารางที่ 1 ตารางเวลาการเดินรถประจำทาง

รถประจำทางสายที่	เส้นทางระหว่างโหนด	Cost (C*)
207	ข้ามแยกเกษตร (SA) – ทุ่งสองห้อง (S1)	28
207	ทุ่งสองห้อง (S1) – หลักสี่ (S2)	33
207	หลักสี่ (S2) - ดอนเมือง (S3)	14
207	ดอนเมือง (S3) - ข้ามแยกเกษตร (SA)	14
51ร	หลักสี่ (S2) - มหาวิทยาลัยราชภัฏพระนคร (S4)	25
51ร	มหาวิทยาลัยราชภัฏพระนคร (S4) – การไฟฟ้าบางเขน (S5)	20
51ร	การไฟฟ้าบางเขน (S5) – อู่บางเขน (S6)	2
522	ข้ามแยกเกษตร (SA) – มหาวิทยาลัยเกษตรศาสตร์ (S7)	12
522	ข้ามแยกเกษตร (SA) – มหาวิทยาลัยเกษตรศาสตร์ (S7)	12
522	มหาวิทยาลัยเกษตรศาสตร์ (S7) – มหาวิทยาลัยศรีปทุม (S8)	26
522	มหาวิทยาลัยศรีปทุม (S7) – อู่บางเขน (S6)	22

จากตารางที่ 1 เส้นทางรถ ขสมก. รถประจำทางสาย 207 ขึ้นข้ามแยกเกษตร ลงต่อรถที่ป้ายรถเมล์หลักสี่ รถประจำทางสาย 51ร ลงอู่บางเขน ใช้เวลาในการเดินทาง 1 ชั่วโมง 5 นาที และ สายรถประจำทางสาย 522 ขึ้นข้ามแยกเกษตร ถึง อู่บางเขน ใช้เวลา 45 นาที ผู้วิจัยได้นำมาสร้างแบบจำลองเส้นทางลักษณะกราฟ (Graph Model) เพื่อใช้ในการทดสอบการทำงานของอัลกอริทึมดังแสดงในภาพที่ 5



ภาพที่ 5 ตัวอย่างแบบจำลองเส้นทางรูปแบบกราฟที่ได้จากตารางเส้นทางการเดินรถประจำทาง

จากนั้นผู้วิจัยได้สร้างตารางการทดสอบที่มีการกำหนดค่า Heuristic Function และค่าถ่วงน้ำหนักปรับเชิงพลวัต Dynamic Adjustment Weight (λ) เข้าไปในส่วนตัวแปรค่าความเสี่ยงสำหรับทดสอบสถานการณ์จำลอง 2 รูปแบบ เพื่อใช้วัดและประเมินประสิทธิภาพของแบบจำลองที่ได้พัฒนาขึ้น ดังแสดงในตารางที่ 2

ตารางที่ 2 ค่าตัวแปรที่ถูกกำหนดขึ้นเพื่อใช้ทดสอบเส้นทางโดยสารรถประจำทางด้วยสถานการณ์จำลอง

ลำดับ	เส้นทาง	รถประจำทางสายที่	Cost (C*)	λ	Heuristic (H*)	Priority (F*)
1	SA-S1-S2	207	61	0	3	64
2	SA-S1-S2-S3	207	75	1	0	76
3	S2-S4-S5-S6	51ร	47	1	2	50
4	SA-S7-S8-S6	522	60	1	1	62

จากตารางที่ 2 กำหนดให้สถานีต้นทาง (SA) มีรถประจำทางสาย 207 กับ 522 และสถานีปลายทางสาย 522 กับ 51ร โดยเส้นทางที่ใช้ทดสอบได้แก่ เส้นทางที่สั้นที่สุด เริ่มต้น SA ไปยัง ปลายทาง S6 โดยผู้โดยสารสามารถเลือกเส้นทาง โดยใช้เส้นทางที่เหมาะสม เพื่อให้เดินทางจากต้นทางไปยังปลายทางได้ภายในเวลาที่ต้องการได้อย่างเหมาะสม โดยมีการกำหนดระดับค่าความเสี่ยงของเส้นทาง (H*) เช่น ความเสี่ยงจากจราจรแออัด ความเสี่ยงจากการปิดกันชั่วคราวหรือความเสี่ยงจากธรรมชาติเป็นต้น และมีการกำหนดค่าระยะทางจากสถานีจอดรถแต่ละโหนด (C*) และกำหนดค่าตัวเลือกที่เหมาะสม (F*) เพื่อให้การคำนวณเส้นทางที่ถูกคัดเลือกด้วย อัลกอริทึม เอ สตาร์ เกิดประสิทธิภาพมากยิ่งขึ้น

4. ผลการทดสอบแบบจำลอง

ผลการทดสอบด้วยเส้นทางกรณีศึกษา รถ ขสมก. ประจำทางสาย 207, 51ร และ 522 ด้วยสถานการณ์จำลอง 2 สถานการณ์ ได้แก่ จำลองสถานการณ์ปกติ (ใช้ทุกเส้นทางได้ตามปกติ) และ สถานการณ์ที่มีความเสี่ยงที่ทำให้ไม่สามารถใช้เส้นทางได้ มาใช้ในการทดสอบการทำงานของอัลกอริทึมพลวัต เอ สตาร์ (ตารางที่ 2) พบว่า

4.1 ผลการทดสอบด้วยสถานการณ์จำลอง

เป็นสถานการณ์ปกติของการโดยสารรถประจำทาง ซึ่งสามารถใช้ได้กับทุกเส้นทางโดยสารรถประจำทาง ซึ่งผลจากการทดสอบอัลกอริทึมสามารถแสดงผลลัพธ์ได้ถูกต้องตามลำดับขั้นการทำงาน ดังนี้

การค้นหาครั้งที่ 1 เริ่มจาก SA โดยให้ค่า Heuristic ของปลายทาง (H*) มีค่าเท่ากับ 0

ตารางที่ 3 : แสดงการค้นหาครั้งที่ 1 ในสถานการณ์ปกติ

ลำดับ	เส้นทาง	รถประจำทางสายที่	Cost (C*)	Heuristic (H*)	λ
1	SA-S1-S2	207	61	3	0
2	SA-S1-S2-S3	207	75	0	1
3	S2-S4-S5-S6	51ร	47	2	1
4	SA-S7-S8-S6	522	60	1	1

จากตารางที่ 3 เส้นทาง SA-S1-S2-S3 เส้นทางที่ H* มีค่าเป็น 0 เส้นทางนั้นไม่ได้ผ่านจุดปลายทางระบบจะตัดทิ้ง ดังนั้น ระบบจะทำการคัดเลือก SA S1 S2 ได้แก่ (1) (3) เส้นทางสาย S2-S6 ซึ่งวิ่งโดยรถสาย 51ร (4) SA-S6 ซึ่งโดยรถประจำทางสาย 522 เท่านั้น

การค้นหาครั้งที่ 2 จะนำเส้นทางจากการค้นหาครั้งแรกที่สั้นที่สุดไปเชื่อมเส้นทางถัดไป โดยจะนำเส้นทางจากการค้นหาก่อนหน้ามาเปรียบเทียบ

ตารางที่ 4: แสดงการค้นหาครั้งที่ 2 ในสถานการณ์ปกติ

ลำดับ	เส้นทาง	รถประจำทางสายที่	Cost (C*)	λ	Heuristic (H*)	Priority (F*)
1	SA-S1-S2	207	61	0	3	64
2	S2-S4-S5-S6	51ร	47	1	2	50
3	SA-S7-S8-S6	522	60	1	1	62

จากตารางที่ 4 ผลจากการค้นหาตารางที่ 4 พบว่า เส้นทาง SA ปลายทาง S6 ถูกคัดเลือกโดยค่า F* ที่น้อยที่สุด ซึ่งมีค่าเท่ากับ 62 ดังนั้น ระบบจะแนะนำให้ผู้โดยสารเลือกเส้นทางโดยสารประจำทางสาย 522 หรือหากเกิดกรณีที่สาย 522 เกิดขัดข้อง ระบบจะแนะนำสายทางเลือกสาย 207 ต่อรถ 51ร SA-S2 และ S2-S6 แทนที่เป็นลำดับถัดไป

5. สรุปและอภิปรายผลการวิจัย

5.1 ผลการวิจัยเป็นไปตามวัตถุประสงค์ของการพัฒนาแบบจำลองการเลือกเส้นทางโดยสารรถประจำทางรูปแบบพลวัตด้วยอัลกอริทึม เอ สตาร์ โดยผลการประเมินด้วยผู้เชี่ยวชาญ พบว่า มีค่าเฉลี่ยในระดับดี (ค่าเฉลี่ย ($\bar{X}$) = 4.30 และค่าเบี่ยงเบนมาตรฐาน S.D. = 0.273) เนื่องจากระบบได้ถูกออกแบบให้ผู้ใช้งานมีความคล่องตัวในการนำไปใช้ [3] และการคำนวณเพื่อระบุตำแหน่งเส้นทางปัจจุบันของรถโดยสารประจำทางมีความถูกต้องและมีการคัดเลือกเส้นทางการเดินทางได้อย่างเหมาะสม

5.2 การประยุกต์ใช้ ค่า H* จะถูกจัดเก็บเป็นค่าพารามิเตอร์ในฐานข้อมูลโดยผู้ดูแลระบบ โดยมีการกำหนดตามเส้นทางจริงที่ใช้ในการเดินทาง ซึ่งในอนาคตค่านี้จะมีการใช้ข้อมูลจากเส้นทางการจราจรที่ถูกเก็บรวบรวมโดยสำนักงานพัฒนารัฐบาลดิจิทัลองค์การมหาชน (สพร., 2561)

6. เอกสารอ้างอิง

[1] วีรพันธ์ รุจิเกียรติกำจร, และวีรินทร์ หวังจิรนิรันดร์. (2556). *การศึกษาค่าใช้จ่ายในการเดินทางและค่าใช้จ่ายด้านพลังงานของการเดินทางแต่ละรูปแบบ: กรณีศึกษาเส้นทางสะพานใหม่-สีลม*. วารสารวิจัยพลังงาน มหาวิทยาลัยจุฬาลงกรณ์, ปีที่10 ฉบับที่ 1 (มกราคม – เมษายน) 2556.

[2] Burke, Paul J. & Nishitateno, Shuhei. (2013*). Gasoline Prices, Gasoline Consumption, and New-Vehicle Fuel Economy: Evidence for a large sample of countries.* Energy Economics 36 2013 : 363-370.

[3] Lane, B. W. (2010). *The relationship between recent gasoline price fluctuations and transit ridership in majorUS cities.* Journal of Transport Geography 18 2010 : 214-225.

[4] ชนินทร์ เขียวสนั่น. (2547). การส่งเสริมระบบขนส่งมวลชนในเขตเมืองชั้นใน กรณีศึกษาพฤติกรรมการเดินทางของผู้ใช้รถยนต์ส่วนบุคคลในย่านธุรกิจถนนสีลม. (วิทยานิพนธ์ปริญญามหาบัณฑิต).สาขาวิชาการวางแผนชุมชนเมืองและสภาพแวดล้อม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

[5] ธเนศ ขุมทรัพย์. (2549). ความสัมพันธ์ระหว่างค่าใช้จ่ายด้านที่อยู่อาศัยค่าใช้จ่ายในการเดินทางเข้าสู่แหล่งงาน และที่ตั้ง ที่อยู่อาศัย : กรณีศึกษา ผู้ที่ทางานในอาคารสำนักงานย่านถนนสาทร. (วิทยานิพนธ์ปริญญามหาบัณฑิต). สาขาวิชาเคหการ ภาควิชาเคหการ คณะสถาปัตยกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย.

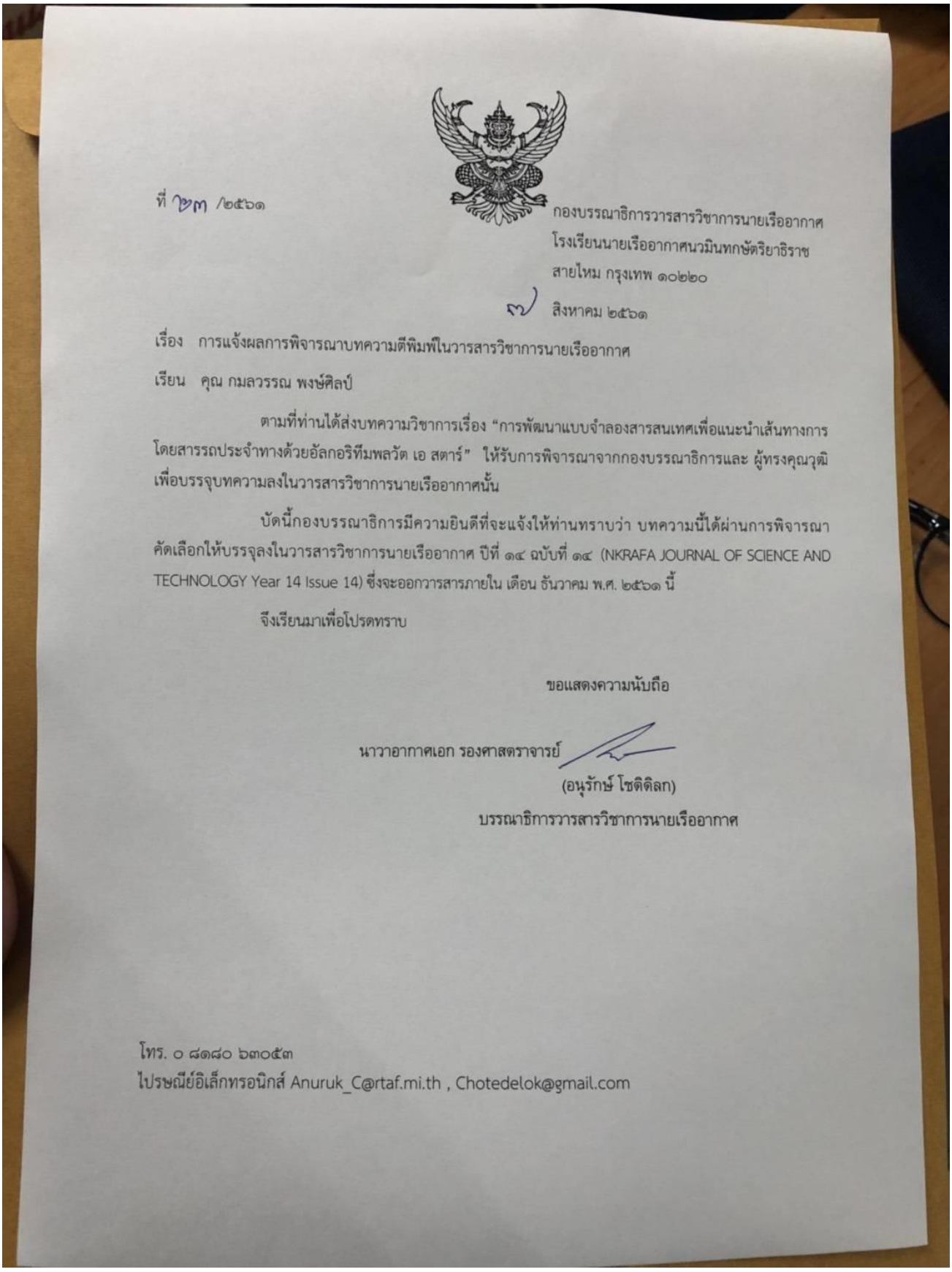
[6] อรอนงค์ กฤตยาเกียรณ์. (2545). การจัดทาพื้นที่จอดรถยนต์เพื่อสนับสนุนโครงการระบบขนส่งมวลชน. (วิทยานิพนธ์ปริญญามหาบัณฑิต). สาขาวิชาการวางผังเมือง ภาควิชาการวางแผนภาคและเมือง คณะสถาปัตยกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย.

[7] ปราชญา สมชื่อ, และภูมิรพี สืบวงษ์รอด. (2557). *ระบบติดตามรถยนต์ด้วยจีพีเอสผ่านแอพพลิเคชันบนระบบปฏิบัติการแอนดรอยด์.* ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา ตาก.

[8] สุรเชษฐ์ กานต์ประชา, และเศรษฐา ตั้งค้าวานิช. (2557). *โครงการวิจัย เรื่อง ระบบติดตามและประมาณเวลาการเดินรถไฟฟ้าด้วยสมาร์ทโฟนผ่านเครือข่าย 3G.* คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

[9] สุขสวัสดี ณัฏฐวุฒิสิทธิ์, และประสงค์ ปราณีตพลกรัง. (2554). *การพัฒนาแบบจำลองการจัดเส้นทางการเดินทางที่เหมาะสมในรูปแบบพลวัตของอัลกอริทึมเอสตาร์.* บทความวิจัยการประชุมทางวิชาการระดับชาติด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศครั้งที่ 7.

ภาคผนวก ค
หนังสือตอบรับ บทความตีพิมพ์ผลงานวารสาร
วิชาการนายเรืออากาศ, ปีที่ 14 (14)



ที่ ๒๓ /๒๕๖๑

กองบรรณาธิการวารสารวิชาการนายเรืออากาศ
โรงเรียนนายเรืออากาศนวมินทกษัตริยาธิราช
สายไหม กรุงเทพ ๑๐๒๒๐

๗ สิงหาคม ๒๕๖๑

เรื่อง การแจ้งผลการพิจารณาบทความตีพิมพ์ในวารสารวิชาการนายเรืออากาศ

เรียน คุณ กมลวรรณ พงษ์ศิลป์

ตามที่ท่านได้ส่งบทความวิชาการเรื่อง "การพัฒนาแบบจำลองสารสนเทศเพื่อแนะนำเส้นทางการโดยสารรถประจำทางด้วยอัลกอริทึมพลวัต เอ สตาร์" ให้รับการพิจารณาจากกองบรรณาธิการและ ผู้ทรงคุณวุฒิ เพื่อบรรจุบทความลงในวารสารวิชาการนายเรืออากาศนั้น

บัดนี้กองบรรณาธิการมีความยินดีที่จะแจ้งให้ท่านทราบว่า บทความนี้ได้ผ่านการพิจารณาคัดเลือกให้บรรจุลงในวารสารวิชาการนายเรืออากาศ ปีที่ ๑๔ ฉบับที่ ๑๔ (NKRAFA JOURNAL OF SCIENCE AND TECHNOLOGY Year 14 Issue 14) ซึ่งจะออกวารสารภายใน เดือน ธันวาคม พ.ศ. ๒๕๖๑ นี้

จึงเรียนมาเพื่อโปรดทราบ

ขอแสดงความนับถือ

นาวาอากาศเอก รองศาสตราจารย์
(อนุรักษ์ โชติดิลก)
บรรณาธิการวารสารวิชาการนายเรืออากาศ

โทร. ๐ ๘๑๘๐ ๖๓๐๕๓
ไปรษณีย์อิเล็กทรอนิกส์ Anuruk_C@rtaf.mi.th , Chotedelok@gmail.com

การพัฒนาแบบจำลองสารสนเทศเพื่อแนะนำเส้นทางการโดยสารรถประจำทาง ด้วยอัลกอริทึมพลวัต เอ สตาร์

The Development of Bus Routing Recommendation Using Dynamic A* Algorithm

[1]กมลวรรณ พงษ์ศิลป์* และ [2]สุขสวัสดี ณัฏฐวุฒิสิทธิ์

[1,2]สาขาวิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม

2410/2 ถ.พหลโยธิน เขตจตุจักร กรุงเทพฯ 10900

E-mail : pbnyo100732@gmail .com

บทคัดย่อ (Abstract)

งานวิจัยนี้เป็นงานวิจัยเชิงทดลอง มีวัตถุประสงค์เพื่อศึกษาปัญหาและแนวทางการพัฒนากลไกการเลือกเส้นทางโดยสารรถประจำทาง โดยผู้วิจัยได้นำแนวคิดการคัดเลือกเส้นทางด้วยอัลกอริทึมพลวัต เอสตาร์มาใช้เพื่อค้นหาและคัดเลือกเส้นทางที่เหมาะสมแก่ผู้ใช้งาน ทำให้ช่วยคำนวณเวลา และวางแผนการเดินทางได้อย่างมีประสิทธิภาพ เครื่องมือที่ใช้ในงานวิจัย ได้แก่ แบบจำลองสารสนเทศเพื่อแนะนำเส้นทางโดยสารรถประจำทางที่ถูกพัฒนาขึ้นด้วยภาษาสวิฟท์บนพื้นฐานอัลกอริทึมพลวัต เอ สตาร์ กลุ่มตัวอย่าง ได้แก่ ข้อมูลการโดยสารรถประจำทางองค์การขนส่งมวลชนกรุงเทพ (ขสมก.) เส้นทางสาย 207, 51ร, 522 ประกอบด้วย (1) ค่าความเสี่ยงของเส้นทาง (2) ค่าความเหมาะสมของจุดขึ้น-ลงรถโดยสารประจำทาง และ (3) ค่านัยสำคัญของเส้นทางการโดยสารรถประจำทาง สำหรับใช้คำนวณ หาเส้นทางที่เหมาะสม ผลการวิจัย พบว่า ผลลัพธ์จากการประเมินโดยผู้เชี่ยวชาญมีค่าเฉลี่ยในระดับดี (ค่าเฉลี่ย ($\bar{X}$) = 4.30 และค่าเบี่ยงเบนมาตรฐาน S.D. = 0.273) ประโยชน์ที่ได้รับจากงานวิจัยนี้สามารถนำไปประยุกต์ใช้กับการคัดเลือกเส้นทางโดยสารรถประจำทางอื่น ๆ รวมทั้งองค์ความรู้นี้ยังสามารถนำไปขยายผลทางวิชาการได้ต่อไปในอนาคต

คำสำคัญ : การคัดเลือก, รถประจำทาง, เส้นทาง, อัลกอริทึม, เอ สตาร์

Abstract

This research is experimental research. The purpose of this study was to investigate the problems and ways to develop a bus route selection mechanism. The researcher has adopted the concept of route selection with dynamic algorithms. Aster Star is used to find and select the appropriate route for the user. Help to calculate time And plan to travel efficiently. The research tool was a traffic information model developed by Swift, based on the Astartes dynamic algorithm. The samples were bus traffic data, mass transit (Routes 207, 51ร , 522 consist of: (1) Route risk; (2) Approach to take-off bus; and (3) Significance of route. Routes The results showed that

the results from the expert evaluation were good (mean ($\bar{X}$) = 4.30 and standard deviation S.D. = 0.273). This research can be applied to the selection of other bus routes as well as this knowledge can be extended to further academic results in the future.

Keyword : Selection, Bus, Route, Algorithm, A* star

1. บทนำ

ยุคประเทศไทย 4.0 รูปแบบกิจกรรมการดำเนินชีวิตประจำวัน และ การทำธุรกิจของประชาชนได้เกิดการเปลี่ยนแปลงไปจากอดีตมาก [1] ในปัจจุบันประชาชนมีการเดินทางติดต่อระหว่างกันตลอดเวลา ซึ่งหนึ่งในรูปแบบกิจกรรมที่เป็นปัจจัยสำคัญต่อการดำเนินชีวิต [2] ได้แก่ การเดินทางไม่ว่าจะเป็นการขนส่งสินค้าและบริการการติดต่องาน หรือ การเดินทางท่องเที่ยวในสถานที่ต่าง ๆ [3] โดยเฉพาะในเมืองใหญ่ เช่น กรุงเทพมหานคร มีประชาชนที่ใช้บริการเส้นทางโดยสารด้วยระบบขนส่งสาธารณะในแต่ละวันเป็นจำนวนมากโดยเฉพาะรถโดยสารประจำทางขององค์การขนส่งมวลชนกรุงเทพ (ขสมก.) [4] ซึ่งได้ถูกนำมาให้บริการประชาชนและช่วยแก้ไขปัญหาการจราจรได้อย่างดี [5]

รถโดยสารประจำทางขององค์การขนส่งมวลชนกรุงเทพ (ขสมก.) เป็นการให้บริการขนส่งสาธารณะรูปแบบหนึ่งที่สำคัญสำหรับประชาชนไทย และมีบทบาทต่อการพัฒนาเศรษฐกิจและสังคมของประเทศ [6] ซึ่งรัฐบาลได้ให้ความสำคัญเป็นอันดับต้นของการพัฒนาประเทศ แต่อย่างไรก็ดีจากการสำรวจทางสถิติพบว่าการให้บริการรถโดยสาร ขสมก. บางเส้นทางยังไม่สามารถอำนวยความสะดวกแก่ผู้โดยสารได้อย่างเพียงพอ ต้นเหตุของปัญหาส่วนใหญ่เกิดจากรถโดยสารประจำทางไม่สามารถกำหนดตารางเวลาที่แน่นอนของการเข้าสถานีจอดรถได้เพราะกรุงเทพมหานครมีสภาพการจราจรติดขัด ส่งผลให้ความถี่ของการปล่อยรถโดยสารประจำทางบางเส้นทางเกิดการขาดช่วง จึงทำให้ผู้โดยสารต้องรอรถเป็นระยะเวลานานเนื่องจากไม่สามารถทราบกำหนดการที่แน่นอนของรถ [7], [8], [9]

ดังนั้นจากปัญหาดังกล่าว ผู้วิจัยได้พัฒนาแบบจำลองสารสนเทศสำหรับการคัดเลือกเส้นทางโดยสารรถประจำทางที่เหมาะสมบนพื้นฐานอัลกอริทึมพลวัต เอ สตาร์ [10] เพื่อให้ผู้ใช้งานสามารถบริหารจัดการเส้นทาง [11] และคำนวณเวลา เพื่อลดระยะทางหรือเวลาในการเดินทาง [12] ประโยชน์ที่คาดจะได้รับครั้งนี้จะช่วยลดปัญหาการรอคิวรถโดยสารประจำทางเป็นเวลานาน โดยผู้ให้บริการและผู้รับบริการสามารถทราบกำหนดเวลาการเข้าจอดรถได้โดยประมาณ [13] ทำให้สามารถวางแผนการเดินทางของตนเองได้ นอกจากนี้องค์ความรู้ด้านสารสนเทศนี้ยังสามารถนำไปประยุกต์ใช้ในระบบขนส่งมวลชนให้เกิดประสิทธิภาพมากขึ้นในอนาคต[14] [15]

2. วัตถุประสงค์

1. เพื่อศึกษาปัญหาและแนวทางการพัฒนาแบบจำลองการเลือกเส้นทางโดยสารรถประจำทางด้วยอัลกอริทึมพลวัต เอ สตาร์

2. เพื่อการพัฒนาแบบจำลองการเลือกเส้นทางโดยสารรถประจำทางด้วยอัลกอริทึมพลวัต เอ สตาร์

3. วิธีดำเนินการวิจัย

ผู้วิจัยได้ใช้ระเบียบวิธีวิจัยเชิงทดลอง โดยพัฒนาแบบจำลองการคัดเลือกเส้นทางโดยสารรถประจำทางขสมก.สาย 207, สาย 51ร, และสาย 522 ที่มีเส้นทางการเดินรถโดยสารเชื่อมต่อกัน โดยมีจุดเริ่มต้นที่สถานีจอดรถโดยสารสาย 207 และ จุดปลายทางที่สถานีจอดรถโดยสารสาย 522 ดังแสดงในรูปที่ 1

รูปที่ 1 ตัวอย่างเส้นทางรถโดยสารขสมกสาย 207, 51ร, 522

3.1 การออกแบบและพัฒนาแบบจำลอง

การทำงานของแบบจำลองสารสนเทศด้วยอัลกอริทึมพลวัตเอ สตาร์ มีขั้นตอนวิธีการทำงาน ดังสมการ (1)

$$F(n) = H(n) + C(n) \qquad (1)$$

จากสมการ (1) เปลี่ยนเป็นแบบพลวัตด้วยการแทรกค่าถ่วงน้ำหนักปรับเชิงพลวัต Dynamic Adjustment Weight (λ) เข้าไปในส่วนตัวแปรค่า Heuristic Function เพื่อเพิ่มความยืดหยุ่นของการเลือกเส้นทาง ดังสมการ (2)

$$F^*(n) = (1+\lambda)\, H^*(n) + C^*(n) \qquad (2)$$

$F^*(n)$ = ค่าความเหมาะสมของการคัดเลือกเส้นทางจากโหนดต้นทางไปยังโหนดปลายทาง

$C^*(n)$ = ค่าใช้จ่ายในการเดินทางในแต่ละโหนดย่อย

$H^*(n)$ = ค่าฮิวริสติกที่ใช้ในการคัดเลือกเส้นทางในแต่ละโหนดย่อยว่าเหมาะสมในการเดินทางหรือไม่

λ = ค่าถ่วงน้ำหนักปรับเรียบเชิงพลวัตของสถานะเส้นทาง ซึ่งมีค่าเท่ากับ 0 และ 1 โดยเส้นทางสถานะปกติ =0 และเส้นทางเสี่ยงสถานะ =1 เพื่อแสดงให้เห็นว่าเส้นทางที่มีความเสี่ยงจะถูกนำมาใช้คำนวณ ในสมการ (2) ให้มีค่าลำดับในการเข้ารับการตรวจหาโหนด (Priority) มีค่ามากกว่าเส้นทางปกติ 2 เท่า เพื่อให้ค่าน้ำหนักในการตัดสินสูง และอัลกอริทึมจะไม่เลือกไปยังเส้นทางนั้น ๆ [15]

โดยหลักการทำงานของขั้นตอนวิธีการค้นหาและคัดเลือกเส้นทางด้วยอัลกอริทึมพลวัต เอ สตาร์จะให้ความสำคัญต่อการค้นหาเส้นทางที่สั้นที่สุดให้ก่อน (ในกรณีที่มีเส้นทางที่สั้นที่สุด) แต่หากพบว่าเส้นทางที่พบนั้นมีค่าความเสี่ยงที่ถูกระบุไว้ในตาราง

การกำหนดค่า (เช่น ความเสี่ยงของการจราจรติดขัด เส้นทางถูกปิดใช้ชั่วคราว หรือ

ไม่สามารถใช้สัญจรได้) อัลกอริทึมจะใช้การค้นหาเส้นทางอื่นที่เหมาะสมที่สุดแทน ทำให้การค้นหาและถูกคัดเลือกเส้นทางโดยใช้มีความยืดหยุ่นและความเหมาะสมมากขึ้น [10] สำหรับขั้นตอนการทำงานของอัลกอริทึมพลวัต แบบ เอ สตาร์ แสดงในรูปแบบรหัสเทียม (Psudo code) ดังรูปที่ 2

```
A* (start, goal)
1.  Closed set = the empty set
2.  Open set = includes start node
3.  G[start] = 0, H[start] = H_calc[start, goal]
4.  F[start] = H[start]
5.  While Open set ≠ Ø
6.  do CurNode ← EXTRACT-MIN-F(Open set)
7.  if ( CurNode == goal ), then return BestPath
8.  For each Neighbor Node N of CurNode
9.  If ( N is in Closed set ), then Nothing
10. else if ( N is in Open set ),
11. calculate N's G, H, F
12. If ( G[N on the Open set] > calculated G[N])
13. RELAX(N, Neighbor in Open set, w)
14. N's parent=CurNode& add N to Open set
15. else, then calculate N's G, H, F
16. N's parent = CurNode& add N to Open
```

รูปที่ 2 แสดง Pseudo Code อัลกอริทึมแบบ เอ สตาร์

ผู้วิจัยได้นำขั้นตอนวิธีการทำงานของอัลกอริทึมพลวัต เอ สตาร์นี้มาประยุกต์ใช้กับการออกแบบจำลองสารสนเทศเพื่อคัดเลือกเส้นทางที่เหมาะสมของการโดยสารรถประจำทาง ขสมก. สาย 207, สาย 51ร, และ สาย 522 ที่มีเส้นทางการเดินรถโดยสารเชื่อมต่อกัน เพื่อช่วยลดระยะเวลาของการทำงานในภาพรวมซึ่งมีรายละเอียดของขั้นตอนแสดงในรูปที่ 3 ดังนี้

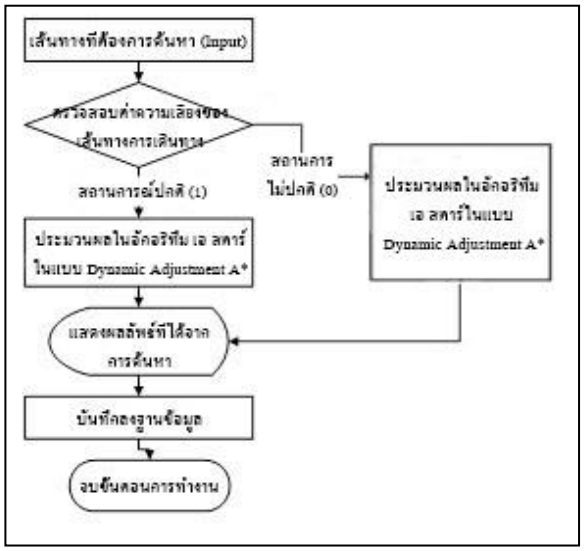
1) เริ่มต้นจากอัลกอริทึมตรวจสอบเส้นทางที่ต้องการค้นหาจากฐานข้อมูลเดิมที่เคยถูกเรียกใช้ หากพบว่าเคยมีการเรียกใช้ก่อนหน้าอัลกอริทึมจะแสดงผลลัพธ์ออกมาให้ผู้ใช้ทราบและบันทึกลงฐานข้อมูล

2) หากพบว่ายังไม่มีการค้นหาของเส้นทางที่ต้องการในฐานข้อมูล อัลกอริทึมจะเริ่มต้นค้นหาตามรูปแบบพลวัต เอ สตาร์

3) จากนั้นอัลกอริทึมทำการเช็คค่าความเสี่ยงของเส้นทาง (แลมด้า) ที่เลือกมานั้นว่ามีสถานะเป็น (λ= 0 หรือ 1)

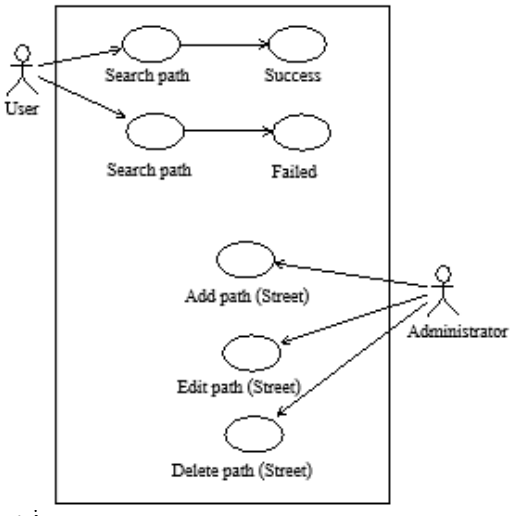
4) หากอัลกอริทึมพบว่ามีเส้นทางใดเส้นทาง หนึ่งไม่สามารถเดินทางได้ (λ มีค่าเท่ากับ 0) อัลกอริทึมจะทำการคัดเลือกเส้นทางอื่นแทน

5) หากอัลกอริทึมไม่พบเส้นทางที่จะเดินทางได้อัลกอริทึมจะเสนอเส้นทางที่เหมาะสมอื่นให้แทน



รูปที่ 3 ขั้นตอนการทำงานอัลกอริทึมพลวัต เอ สตาร์

สำหรับออกแบบในลักษณะ Client – Server เพื่อผู้ใช้สามารถใช้งานผ่านเว็บไปยังเครื่องแม่ข่าย (Server) ได้โดยมีการออกแบบ Use Case View ที่ประกอบด้วย Use Case Diagram และบุคคลที่เกี่ยวข้องได้แก่ ผู้ใช้งานและผู้ดูแลระบบ ดังรูปที่ 4



รูปที่ 4 แสดง Use Case Diagram ของระบบ

สำหรับในส่วนผู้ใช้งาน ผู้วิจัยออกแบบจำลองผู้ใช้งาน (Use Case View) ด้วยภาษาสวิฟท์ เป็นภาษาโปรแกรมเปรียบเสมือนภาษาตัวกลาง โดยใช้ XCODE 5.0 บน MAC OSX 10.0 เป็นเครื่องมือพัฒนา ดังรูปที่ 5

รูปที่ 5 ตัวอย่างแบบจำลองผู้ใช้งาน

3.2 การทดสอบแบบจำลอง

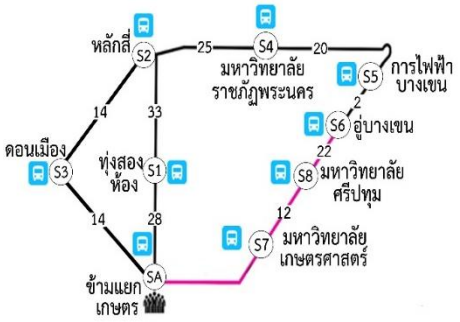
ผู้วิจัยได้ทดสอบแบบจำลองที่ถูกพัฒนาขึ้นโดยใช้เส้นทางกรณีศึกษารถ ขสมก. รถประจำทางสาย 207, 51ร และ 522 ซึ่งมีตารางเวลาการเดินรถประจำทาง (ตารางที่ 1) และสร้างสถานการณ์จำลอง จำนวน2 สถานการณ์ ได้แก่ จำลองสถานการณ์ปกติ (ใช้ทุกเส้นทางได้ตามปกติ) และ สถานการณ์ที่มีความเสี่ยงที่ทำให้ไม่สามารถใช้เส้นทางได้ สำหรับใช้ทดสอบการทำงานของอัลกอริทึมพลวัต เอ สตาร์

ตารางที่ 1 ตารางเวลาการเดินรถประจำทาง

รถประจำทางสายที่	เส้นทางระหว่างโหนด	Cost (C*)
207	ข้ามแยกเกษตร (SA) – ทุ่งสองห้อง (S1)	28
207	ทุ่งสองห้อง (S1) – หลักสี่ (S2)	33
207	หลักสี่ (S2) - ดอนเมือง (S3)	14

รถประจำทางสายที่	เส้นทางระหว่างโหนด	Cost (C*)
207	ดอนเมือง (S3) - ข้ามแยกเกษตร (SA)	14
51ร	มหาวิทยาลัยราชภัฏพระนคร (S4) – การไฟฟ้าบางเขน (S5)	20
51ร	การไฟฟ้าบางเขน (S5) – อู่บางเขน (S6)	2
522	ข้ามแยกเกษตร (SA) – มหาวิทยาลัยเกษตรศาสตร์ (S7)	12
522	มหาวิทยาลัยศรีปทุม (S7) – อู่บางเขน (S6)	22

จากตารางที่ 1 เส้นทางรถ ขสมก. รถประจำทางสาย 207 ขึ้นข้ามแยกเกษตร ลงต่อรถที่ป้ายรถเมล์หลักสี่ รถประจำทางสาย 51ร ลงอู่บางเขน ใช้เวลาในการเดินทาง 1 ชั่วโมง 5 นาที และ สายรถประจำทางสาย 522 ขึ้นข้ามแยกเกษตร ถึง อู่บางเขน ใช้เวลา 45 นาที ผู้วิจัยได้นำมาสร้างแบบจำลองเส้นทางลักษณะกราฟ (Graph Model) เพื่อใช้ในการทดสอบการทำงานของอัลกอริทึมดังแสดงในรูปที่ 6



รูปที่ 6 ตัวอย่างแบบจำลองเส้นทางรูปแบบกราฟที่ได้จากตารางเส้นทางการเดินรถประจำทาง

จากนั้นผู้วิจัยได้สร้างตารางการทดสอบที่มีการกำหนดค่าHeuristic Function และค่าถ่วงน้ำหนักปรับเชิงพลวัต Dynamic Adjustment Weight (λ) เข้าไปในส่วนตัวแปรค่าความเสี่ยงสำหรับทดสอบสถานการณ์จำลอง2 รูปแบบเพื่อใช้วัดและประเมินประสิทธิภาพของแบบจำลองที่ได้พัฒนาขึ้น ดังแสดงในตารางที่ 2

ตารางที่ 2 ค่าตัวแปรที่ถูกกำหนดขึ้นเพื่อใช้ทดสอบเส้นทางโดยสารรถประจำทางด้วยสถานการณ์จำลอง

ลำดับ	เส้นทาง	รถประจำทางสายที่	Cost (C*)	Heuristic (H*)	λ
1	SA-S1-S2	207	61	3	0
2	SA-S1-S2-S3	207	75	0	1
3	S2-S4-S5-S6	51ร	47	2	1
4	SA-S7-S8-S6	522	60	1	1

จากตารางที่ 2 กำหนดให้สถานีต้นทาง (SA) มีรถประจำทางสาย 207 กับ 522 และสถานีปลายทางสาย 522 กับ 51ร โดยเส้นทางที่ใช้ทดสอบได้แก่ เส้นทางที่สั้นที่สุด เริ่มต้น SA ไปยัง ปลายทาง S6 โดยผู้โดยสารสามารถเลือกเส้นทาง โดยใช้เส้นทางที่เหมาะสม เพื่อให้

เดินทางจากต้นทางไปยังปลายทางได้ภายในเวลาที่ต้องการได้อย่างเหมาะสม โดยมีการกำหนดระดับค่าความเสี่ยงของเส้นทาง (H*) เช่น ความเสี่ยงจากจราจรแออัด ความเสี่ยงจากการปิดกันชั่วคราวหรือความเสี่ยงจากธรรมชาติ เป็นต้น และมีการกำหนดค่าระยะทางจากสถานีจอดรถแต่ละโหนด (C*) และกำหนดค่าตัวเลือกที่เหมาะสม (F*) เพื่อให้การคำนวณเส้นทางที่ถูกคัดเลือกด้วย อัลกอริทึม เอ สตาร์ เกิดประสิทธิภาพมากยิ่งขึ้น

4. ผลการทดสอบแบบจำลอง

ผลการทดสอบด้วยเส้นทางกรณีศึกษารถ ขสมก. ประจำทางสาย 207, 51ร และ 522 ด้วยสถานการณ์จำลอง 2 สถานการณ์ ได้แก่ จำลองสถานการณ์ปกติ (ใช้ทุกเส้นทางได้ตามปกติ) และ สถานการณ์ที่มีความเสี่ยงที่ทำให้ไม่สามารถใช้เส้นทางได้ โดยนำเส้นทางที่แสดงในรูปแบบกราฟ (รูปที่ 6) มาใช้ในการทดสอบการทำงานของอัลกอริทึมพลวัต เอ สตาร์ (ตารางที่ 2) พบว่า

4.1 ผลการทดสอบด้วยสถานการณ์จำลองเป็นสถานการณ์ปกติของการโดยสารรถประจำทาง ซึ่งสามารถใช้ได้กับทุกเส้นทางโดยสารรถประจำทาง ซึ่งผลจากการทดสอบ อัลกอริทึมสามารถแสดงผลลัพธ์ได้ถูกต้องตามลำดับขั้นการทำงาน ดังนี้

การค้นหาครั้งที่ 1 เริ่มจาก SA โดยให้ค่า Heuristic ของปลายทาง (H*) มีค่าเท่ากับ 0

ตารางที่ 3 : แสดงการค้นหาครั้งที่ 1 ในสถานการณ์ปกติ

ลำดับ	เส้นทาง	รถประจำทางสายที่	Cost (C*)	λ	Heuristic (H*)	Priority (F*)
1	SA-S1-S2	207	61	0	3	64
2	SA-S1-S2-S3	207	75	1	0	76
3	S2-S4-S5-S6	51ร	47	1	2	50
4	SA-S7-S8-S6	522	60	1	1	62

จากตารางที่ 3 เส้นทาง SA-S1-S2-S3 เส้นทางที่ H* มีค่าเป็น 0 เส้นทางนั้นไม่ได้ผ่านจุดปลายทางระบบจะตัดทิ้ง ดังนั้น ระบบจะทำการคัดเลือก SA S1 S2 ได้แก่ (1) (3) เส้นทางสาย S2-S6 ซึ่งวิ่งโดยรถสาย 51ร (4) SA-S6 ซึ่งโดยรถประจำทางสาย 522 เท่านั้น

การค้นหาครั้งที่ 2 จะนำเส้นทางจากการค้นหาครั้งแรกที่สั้นที่สุดไปเชื่อมเส้นทางถัดไป โดยจะนำเส้นทางจากการค้นหาก่อนหน้ามาเปรียบเทียบ

ตารางที่ 4 : แสดงการค้นหาครั้งที่ 2 ในสถานการณ์ปกติ

ลำดับ	เส้นทาง	รถประจำทางสายที่	Cost (C*)	λ	Heuristic (H*)	Priority (F*)
1	SA-S1-S2	207	61	0	3	64
2	S2-S4-S5-S6	51ร	47	1	2	50
3	SA-S7-S8-S6	522	60	1	1	62

จากตารางที่ 4 ผลจากการค้นหาตารางที่ 4 พบว่า เส้นทาง SA ปลายทาง S6 ถูกคัดเลือกโดยค่า F* ที่น้อยที่สุด ซึ่งมีค่าเท่ากับ 62 ดังนั้นระบบจะแนะนำให้ผู้โดยสารเลือกเส้นทางโดยสารประจำทางสาย 522 หรือหากเกิดกรณีที่สาย 522 เกิดขัดข้อง ระบบจะแนะนำสายทางเลือกสาย 207 ต่อรถ 51ร SA-S2 และ S2-S6 แทนที่เป็นลำดับถัดไป

5. สรุปและอภิปรายผลการวิจัย

ผลการวิจัยเป็นไปตามวัตถุประสงค์ของการพัฒนาแบบจำลองการเลือกเส้นทางโดยสารรถประจำทางรูปแบบพลวัตด้วยอัลกอริทึม เอสตาร์ โดยผลการประเมินด้วยผู้เชี่ยวชาญ พบว่ามีค่าเฉลี่ยในระดับดี (ค่าเฉลี่ย ($\overline{X}$) = 4.30 และค่าเบี่ยงเบนมาตรฐาน (S.D. = 0.273) เนื่องจากระบบได้ถูกออกแบบให้ผู้ใช้งานมีความคล่องตัวในการนำไปใช้ และการคำนวณเพื่อระบุตำแหน่งเส้นทางปัจจุบันของรถโดยสารประจำทางมี

6. เอกสารอ้างอิง

[1] วีรพันธ์ รุจิเกียรติกาจร และวีรินทร์ หวังจิรนิรันดร์. (2556**).** การศึกษาค่าใช้จ่ายในการเดินทางและค่าใช้จ่ายด้านพลังงานของการเดินทางแต่ละรูปแบบ: กรณีศึกษาเส้นทางสะพานใหม่-สีลม. วารสารวิจัยพลังงาน, 10(1): 1-11.

[2] Paul J. Burke and Nishitateno, Shuhei. (2011). Gasoline prices, gasoline consumption, and new-vehicle fuel economy: Evidence for a large sample of countries. Australian National University Coombs Building 9.

[3] Lane B. W. (2010). The relationship between recent gasoline price fluctuations and transit ridership in major US cities. Journal of Transport Geography 18: 214-225.

[4] ชนินทร์ เขียวสนั่น. (2547). การส่งเสริมระบบขนส่งมวลชนในเขตเมืองชั้นในกรณีศึกษา พฤติกรรมการเดินทางของผู้ใช้รถยนต์ส่วนบุคคลในย่านธุรกิจถนนสีลม (วิทยานิพนธ์ปริญญามหาบัณฑิต, สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง).

[5] ธเนศ ขุมทรัพย์. (2549). ความสัมพันธ์ระหว่างค่าใช้จ่ายด้านที่อยู่อาศัยค่าใช้จ่ายในการเดินทางเข้าสู่แหล่งงาน และที่ตั้ง ที่อยู่อาศัย :กรณีศึกษา ผู้ที่ทางานในอาคารสานักงานย่านถนนสาทร (วิทยานิพนธ์

ปริญญามหาบัณฑิต, จุฬาลงกรณ์มหาวิทยาลัย).

[6] อรอนงค์ กฤตยาเกียรณ์. (2545). การจัดทาพื้นที่จอดรถยนต์เพื่อสนับสนุนโครงการระบบขนส่งมวลชน (วิทยานิพนธ์ปริญญามหาบัณฑิต, จุฬาลงกรณ์มหาวิทยาลัย).

[7] สุรเชษฐ์ กานต์ประชา และเศรษฐา ตั้งค้าวานิช.(2557). ระบบติดตามและประมาณเวลาการเดินรถไฟฟ้าด้วยสมาร์ทโฟนผ่านเครือข่าย 3G (โครงการวิจัยคณะวิศวกรรมศาสตร์, มหาวิทยาลัยนเรศวร).

[8] สุขสวัสดี ณัฏฐวุฒิสิทธิ์ และประสงค์ ปราณีตพลกรัง.(2554). การพัฒนาแบบจาลองการจัดเส้นทางการเดินทางที่เหมาะสมในรูปแบบพลวัตของอัลกอริทึม เอ สตาร์. ในการประชุมทางวิชาการระดับชาติด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศครั้งที่ 7. กรุงเทพฯ: คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

[9] สัญญา เครือหงษ์ และสุรางค์รัตน์ เชาว์โคกสูง. (2551). การพัฒนาเกมแนวต่อสู้โดยใช้ชุดคาสั่งซีดีเอ็กซ์เอนจินและอัลกอริทึมแบบ เอ-สตาร์. วารสารเทคโนโลยีสารสนเทศ มหาวิทยาลัยพระจอมเก้าพระนครเหนือ. 4(7): 38-46.

[10] วิภาดา เพชรรัตน์. (2554). สร้างระบบที่สามารถหาเส้นทางในการเดินทางไปพบปะลูกค้าที่มีระยะทางรวมที่สั้นที่สุดหรือใช้เวลาที่น้อยที่สุดสาหรับลดค้าใช้จ่ายในการเดินทาง (วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต, มหาวิทยาลัยสงขลานครินทร์).

[11] วีรชัย สว่างทุกข์. (2557). การใช้ระบบติดตามจีพีเอสแบบเปิดเผยรหัสต้นฉบับควบคู่สมาร์ตโฟน เพื่อใช้ติดตามรถขนส่ง กรณีศึกษาน้าดื่มทิพย์เขลางค์. วารสารวิชาการคณะเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยราชภัฏลาปาง. 7(1): 40-51

[12] วิภาดา เพชรรัตน์. (2554). ระบบนาทางอัจฉริยะ : กรณีศึกษา บริษัทวานิชรุ่งเรืองอินเตอร์เทรด จากัด (สาขาภาคใต้). (วิทยาศาสตรมหาบัณฑิต , มหาวิทยาลัยสงขลานครินทร์).

[13] พีระเดช สารวมรัมย์.(2558). การพัฒนาระบบติดตามพิกัดผู้ป่วยกลุ่มเสี่ยง STROKE and STEMI ที่ต้องได้รับบริการจากระบบบริการการแพทย์ฉุกเฉินแบบเคลื่อนที่. (วิทยาศาสตรมหาบัณฑิต , มหาวิทยาลัยรังสิต).

[14] Ahmed Ahmed, Elshaimaa Nada, Wafaa Al- Mutiri. (2017). University buses routing and tracking system. International Journal of Computer Science & InformationTechnology (IJCSIT). 9(1) : 95-104

[15] เพชรไพลิน รัสดีดวง, และกฤติกา สังขวดี, ปัญญา สังขวดี. แอปพลิเคชันส่งเสริมกีฬาวอลเลย์บอลบนระบบปฏิบัติการแอนดรอยด์. การประชุมวิชาการระดับชาติราชภัฏเพชรบุรีวิจัยศิลปวัฒนธรรม ครั้งที่ 4. เพชรบุรีฯ: มหาวิทยาลัยราชภัฏเพชรบุรี.

ประวัติย่อผู้วิจัย

ชื่อ – สกุล	กมลวรรณ พงษ์ศิลป์
วัน เดือน ปีเกิด	10 กรกฎาคม 2532
สถานที่เกิด	จังหวัดสมุทรปราการ
วุฒิการศึกษา	พ.ศ.2558 คณะสถาปัตยกรรมและการออกแบบ สาขาออกแบบเซรามิกส์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
ประสบการณ์ทำงาน	
พ.ศ 2557-ปัจจุบัน	ทำงานที่บริษัท เวล ทู ดี จำกัด
สถานที่อยู่ปัจจุบัน	99/29 ซ.ลาดปลาเค้า 72 แขวงอนุสาวรีย์ เขตบางเขน กทม. 10220
ผลงานวิชาการ	กมลวรรณ พงษ์ศิลป์, สุขสวัสดี ณัฏฐวุฒิสิทธิ์. (2561). การพัฒนาแบบจำลองสารสนเทศเพื่อเลือกเส้นทางโดยสารรถประจำทางบนพื้นฐานอัลกอริทึมพลวัต เอ สตาร์. วารสารวิชาการซายน์เทค มรภ.ภูเก็ต คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏภูเก็ต, ปีที่ 2(1)
	กมลวรรณ พงษ์ศิลป์, สุขสวัสดี ณัฏฐวุฒิสิทธิ์. (2561). การพัฒนาแบบจำลองสารสนเทศเพื่อแนะนำเส้นทางการโดยสารรถประจำทางด้วยอัลกอริทึมพลวัต เอ สตาร์. วารสารวิชาการนายเรืออากาศ, ปีที่ 14(14)