

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบค้นหาเส้นทางเดินทางโดยรถโดยสารประจำทางอัจฉริยะ

INTELLIGENT ROUTING FOR TRAVELING BY BUSES



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2550

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบค้นหาเส้นทางเดินทางโดยรถโดยสารประจำทางอัจฉริยะ

INTELLIGENT ROUTING FOR TRAVELING BY BUSES

ผู้จัดทำ

1. นายฉัตรชัย ผู้เจริญยิ่ง รหัสนักศึกษา 48015330

2. นายวรวุฒ ชาติศิริ รหัสนักศึกษา 48015358



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบค้นหาเส้นทางเดินทางโดยรถโดยสารประจำทางอัจฉริยะ

นายฉัตรชัย ผู้เจริญยิ่ง	48015330
นายวรวุฒ ชาติศรี	48015358
ผศ. ดร. วิศิษฎ์ หิรัญกิตติ	อาจารย์ที่ปรึกษา
ปีการศึกษา 2550	

บทคัดย่อ

การคมนาคมการเดินทางในอนาคตจะมีการพัฒนาโดยนำเอาปัญญาประดิษฐ์มาประยุกต์ใช้มากขึ้น ระบบการคมนาคมสาธารณะรถโดยสารประจำทางได้แก่ รถเมล์ รถไฟฟ้า ฯลฯ ก็เช่นเดียวกัน โครงการนี้ได้พัฒนาระบบค้นหาเส้นทางการเดินทางด้วยรถประจำทางเพื่ออำนวยความสะดวกในการเดินทางซึ่งจะครอบคลุม รถเมล์ รถไฟฟ้า ฯลฯ โดยจะเป็นการประยุกต์ใช้ปัญญาประดิษฐ์ในการค้นหาเส้นทางการเดินทางด้วยรถประจำทางอย่างชาญฉลาด โดยสามารถสอบถามว่าจะเดินทางไปยังสถานที่แห่งหนึ่งจะขึ้นรถเมล์สายใดได้บ้าง แล้วต้องไปต่อรถสายใดอีกจนถึงจุดหมาย โดยสามารถเลือกค้นหาในเงื่อนไขต่างๆเช่น เปลี่ยนจำนวนสายน้อยที่สุด ใช้ระยะทางน้อยที่สุด และสามารถเลือกประเภทของยานพาหนะที่ต้องการเดินทางได้ด้วย

Intelligent Routing for Travelling by Buses

Mr. Chatchai Phucharoenyong 48015330

Mr. Worawech Chatsri 48015358

Asst. Prof. Dr. Visit Hirankitti Advisor

Academic Year 2007

ABSTRACT

The future transportation will be advanced by more adaption of artificial intelligence. Similarly, this is also the case for fixed/route public mass transportation best such as buses, trains, In this thesis we have developed an intelligent system for finding routes for traveling via fixed route transportation, such as buses and trains. We applied artificial intelligence for finding the best routes. The user can ask which route to take by which bus number and which bus number to take further until he/she can reach his/her destination under different criteria, for example, minimum change of buses, a shortest route, specific types of buses to take, and so on.



กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้คงไม่อาจเสร็จได้ด้วยดี หากไม่ได้รับความช่วยเหลือ และร่วมมือจากหลาย ๆ ฝ่ายด้วยกัน บุคคลท่านแรกที่ต้องกล่าวถึงเพราะเป็นส่วนสำคัญที่ทำให้วิทยานิพนธ์นี้เสร็จลงได้ก็คือ ผศ. ดร. วิศิษฐ์ หิรัญกิตติ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้ความเอาใจใส่ แนะนำ และช่วยเหลือเสมอมา ซึ่งต้องขอขอบพระคุณเป็นอย่างมาก .

ขอขอบคุณเพื่อน ๆ ทุกคนที่ให้คำแนะนำ ปรึกษา ช่วยเหลือ ให้กำลังใจ ช่วยแก้ไขปัญหา และต้องขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้คณะผู้จัดทำ มีวันนี้ ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูคณะผู้จัดทำ มาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ เอาใจใส่เสมอมา ในทุก ๆ ด้านอันหาที่เปรียบมิได้ คณะผู้จัดทำ ขอระลึกในพระคุณอันสุดประมาณ และขอกราบขอบพระคุณมา ณ ที่นี้

นายฉัตรชัย ผู้เจริญยิ่ง
นายวรเวช ชาตีสรี

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VIII
สารบัญรูปภาพ	IX
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของปัญหา	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	1
1.4 ขอบเขตของโครงการ	2
1.5 เนื้อหาของปริิญาพนธ์	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 ภาษา XML (Extensible Markup Language)	3
2.1.1 ลักษณะโครงสร้างของ XML	3
2.1.1.1 Tag	4
2.1.1.2 Element	4
2.1.1.3 Content	5
2.1.1.4 Attribute	5
2.1.2 ตัวกำหนดโครงสร้างของ XML ด้วย DTD (Document Type Definition)	5
2.1.3 การแปลงรูปแบบเอกสาร XML (Transformation XML)	5
2.2 ภาษา SVG (Scalable Vector Graphic)	6
2.2.1 การอธิบายภาพโดยใช้ภาษา SVG	6
2.2.2 Vector Graphic Shapes	7
2.2.2.1 rect	7
2.2.2.2 circle	7
2.2.2.3 ellipse	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.2.2.4 line	8
2.2.2.5 polygon	8
2.2.2.6 polyline	8
2.2.2.7 path	9
2.2.3 ประโยชน์ในการใช้ SVG	9
2.3 ภาษา Java Script	11
2.3.1 JavaScript คืออะไร	11
2.3.2 ลักษณะการทำงานของ JavaScript	12
2.3.2.1 Navigator JavaScript	12
2.3.2.2 LiveWire JavaScript	12
2.4 DOM (Document Object Model)	13
2.4.1 DOM Parsing	14
2.5 AJAX (Asynchronous Javascript and XML)	15
2.5.1 โครงสร้างของ AJAX เว็บแอปพลิเคชัน	15
2.5.2 หลักการในการทำงานของเทคโนโลยี AJAX	17
2.5.3 ฝั่ง server ต้องจัดส่งเพียงแต่ข้อมูลที่ร้องขอ	20
2.5.4 การตอบสนองต่อผู้ใช้ต้องมีความลื่นไหลและต่อเนื่อง	21
2.5.5 อาศัยการเขียน โค้ดเป็นหลักและต้องมีระเบียบ	23
2.6 ระบบพิกัดตำแหน่งบนพื้นโลก	25
2.6.1 พิกัดกริด (Grid Coordinates)	25
2.6.2 พิกัดกริดยูนิเวอร์ซัลทรานส์เวอร์สเมอร์เคเตอร์ (Universal Transverse Mercator)	26
บทที่ 3 การนำทฤษฎีปัญหาประคิษฐ์มาประยุกต์ใช้งาน	29
3.1 การแทนข้อมูลแผนที่ดิจิทัล	29
3.2 การค้นหาเส้นทาง	30
3.3 การค้นหาเส้นทางที่ดีที่สุดโดยใช้วิธีแอสตาร์	31
3.4 การนำอัลกอริทึมการค้นหาแบบแอสตาร์มาประยุกต์ใช้งานกับโครงการงาน	32

สารบัญ (ต่อ)

	หน้า
3.4.1 การค้นหาเส้นทางที่มีระยะทางสั้นที่สุดในการเดินทาง.....	32
3.4.2 การค้นหาเส้นทางที่มีจำนวนการต่อสายรถน้อยที่สุดในการเดินทาง.....	32
3.4.3 การค้นหาเส้นทางที่มีจำนวนการต่อสายรถน้อยที่สุด และมีระยะทางสั้นที่สุดในการเดินทาง.....	32
3.4.4 การค้นหาเส้นทางโดยระบุชนิดของยานพาหนะที่ใช้ในการเดินทาง.....	33
3.4.5 การค้นหาเส้นทางแบบมีจุดผ่านหลายจุด.....	34
บทที่ 4 การออกแบบและพัฒนา.....	35
4.1 การออกแบบ โครงสร้างระบบรถประจำทางอัจฉริยะ.....	35
4.2 Use Case Diagram.....	36
4.3 Class Diagram.....	38
4.4 Sequence Diagram ของการค้นหาเส้นทาง.....	39
4.5 Sequence Diagram ของการเลือกดูเส้นทางสายรถประจำทาง.....	39
4.6 Sequence Diagram ของการเลือกดูข้อมูลสถานที่.....	40
บทที่ 5 การออกแบบ User Interface.....	41
5.1 UI ส่วนของผู้ใช้งาน.....	41
5.1.1 ส่วนกำหนดจุดต้นทางและจุดปลายทาง.....	41
5.1.2 ส่วนแสดงผลลัพธ์และจำนวนเส้นทางทั้งหมด.....	41
5.1.3 ส่วนแสดงรายละเอียดแต่ละเส้นทาง.....	41
5.1.4 ส่วนแสดงชั้นข้อมูลของสถานที่สำคัญ.....	42
5.1.5 ส่วนแผนที่กรุงเทพฯ.....	42
5.1.6 ส่วนแสดงสายรถประจำทางต่างๆ.....	42
5.2 UI ส่วนของการเพิ่มเส้นทางยานพาหนะ.....	42
5.2.1 ส่วนเพิ่มเส้นรถไฟฟ้า BTS และ MRT.....	43
5.2.2 ส่วนเพิ่มเส้นรถประจำทาง.....	43
5.2.1 ส่วนแผนที่กรุงเทพฯ.....	43
5.3 UI ส่วนของการเพิ่มสถานที่สำคัญ.....	43

สารบัญ (ต่อ)

	หน้า
5.3.1 ส่วนเพิ่มสถานที่สำคัญ.....	44
5.3.2 ส่วนแผนที่กรุงเทพฯ.....	44
บทที่ 6 การทดลองและผลการทดลอง	45
6.1 ความต้องการเบื้องต้นในการทดลอง.....	45
6.2 ทดลองเพิ่มเส้นทางรถประจำทาง.....	46
6.3 ทดลองค้นหาเส้นทางจากต้นทางไปยังปลายทางแบบไม่มีจุดผ่าน.....	49
6.3.1 ทดลองค้นหาเส้นทางแบบที่มีระยะทางสั้นที่สุด.....	49
6.3.2 ทดลองค้นหาเส้นทางแบบที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุด.....	50
6.3.3 ทดลองค้นหาเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุด และระยะทางสั้นที่สุด.....	51
6.4 ทดลองค้นหาเส้นทางจากต้นทางไปยังปลายทางแบบมีจุดผ่าน 3 จุด.....	52
6.4.1 ทดลองค้นหาเส้นทางที่มีระยะทางสั้นที่สุด.....	52
6.4.2 ทดลองค้นหาเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุด.....	53
6.4.3 ทดลองค้นหาเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุด และระยะทางสั้นที่สุด.....	54
6.5 ผลการทดลอง.....	55
บทที่ 7 บทสรุป	56
7.1 สรุปผลการทดลอง.....	56
7.2 วิจารณ์สิ่งที่ได้จากโครงการ.....	56
7.3 ปัญหาอุปสรรคและแนวทางแก้ไข.....	56
7.4 การวิจัยและพัฒนาต่อ.....	56
บรรณานุกรม	57
ภาคผนวก ก	58

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงคำสั่งที่ใช้ในการสร้างรูปร่าง.....	9
2.2 SVG Element.....	10
2.3 SVG Attribute.....	10
2.4 แสดงส่วนประกอบที่เป็นเทคโนโลยีต่างๆซึ่งถูกนำมาสร้าง AJAX.....	24



สารบัญรูปภาพ

รูปที่	หน้า
2.1 การใช้งาน DOM	14
2.2 แสดงวิธีการเข้าถึงข้อมูลโดยวิธี DOM Parsing	14
2.3 แสดงแบบจำลอง Web Application แบบทั่วไป	16
2.4 แสดงแบบจำลอง Web Application ที่พัฒนาโดย AJAX	17
2.5 แสดงแบบจำลองการสื่อสาร ระหว่าง Web browser กับ Server แบบปกติ	18
2.6 แสดงแบบจำลองการสื่อสาร ระหว่าง Web browser กับ Server ที่ใช้ AJAX	19
2.7 แสดงไดอะแกรมข้อมูลที่ทำกรสื่อสารในเว็บแอปพลิเคชันปกติ	20
2.8 แสดงไดอะแกรมข้อมูลที่ทำกรสื่อสารในเว็บแอปพลิเคชันที่ใช้ AJAX	20
2.9 แสดงกราฟเส้นเพื่อให้เห็นการใช้ข้อมูลระหว่างแบบปกติและ AJAX	21
2.10 แสดงลักษณะการสื่อสารที่เป็นแบบ Synchronous ในเว็บแอปพลิเคชันแบบเดิม	22
2.11 แสดงลักษณะการสื่อสารที่เป็นแบบ Asynchronous ในเว็บแอปพลิเคชันแบบ AJAX	22
2.12 แสดงแผนภาพเพื่อให้เห็นลักษณะการทำงานของเว็บแอปพลิเคชันแบบ AJAX	25
3.1 ความสัมพันธ์ระหว่าง path ของถนนและเส้นทาง	29
3.2 โครงสร้างของถนนและการกำหนดโหนดตามทฤษฎีกราฟ	30
3.3 ส่วนประกอบของ heuristic function $f(n)$	31
3.4 ความสัมพันธ์ระหว่างประเภทยานพาหนะกับเส้นทางบนกราฟ	33
3.5 กราฟสำหรับการค้นหาแบบมีจุดผ่านหลายจุด	34
4.1 แสดงโครงสร้างของระบบค้นหาเส้นทางรถประจำทางอัจฉริยะ	35
4.2 ยูสเคสไดอะแกรม	36
4.3 คลาสไดอะแกรม	38
4.4 ซีควเอนไดอะแกรมของการค้นหาเส้นทาง	39
4.5 ซีควเอนไดอะแกรมของการค้นหาเส้นทางสายรถประจำทาง	39
4.6 ซีควเอนไดอะแกรมของการค้นหาสถานที่	40
5.1 แสดง User Interface สำหรับบุคคลทั่วไป	41
5.2 แสดง User Interface สำหรับเพิ่มเส้นทางรถประจำทาง	42
5.3 แสดง User Interface สำหรับเพิ่มสถานที่สำคัญ	43
6.1 แสดง User Interface สำหรับเพิ่มเส้นทางรถประจำทาง	46
6.2 แสดงไดอะล็อกการสร้างเส้นทางรถประจำทาง	46

สารบัญรูปภาพ (ต่อ)

รูปที่	หน้า
6.3 แสดงการกำหนดจุดเริ่มต้นของเส้นทาง	47
6.4 แสดงการใช้งานปุ่มกลับทิศทางของเส้นทาง	47
6.5 แสดงการเพิ่มความยาวของเส้นทาง	48
6.6 แสดงตัวช่วยในการเพิ่มความยาวของเส้นทาง	48
6.7 แสดงการใส่เส้นทางที่สมบูรณ์	48
6.8 ผลการค้นหาเส้นทางแบบที่มีระยะทางสั้นที่สุด	49
6.9 ผลการค้นหาเส้นทางแบบที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุด	50
6.10 ผลการค้นหาเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุดและระยะทางสั้นที่สุด	51
6.11 ผลการค้นหาเส้นทางที่มีระยะทางสั้นที่สุดแบบมีจุดผ่าน 3 จุด	52
6.12 ผลการค้นหาเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุดแบบมีจุดผ่าน 3 จุด	53
6.13 ผลการค้นหาเส้นทางที่มีการเปลี่ยนสายรถน้อยที่สุดและระยะทางสั้นที่สุดแบบมีจุดผ่าน 3 จุด	54
ก.1 แสดงขั้นตอนภายหลังจากการดับเบิลคลิกไฟล์ python-2.4.3.msi	59
ก.2 แสดงขั้นตอนการเลือกกำหนดผู้ใช้งาน	59
ก.3 แสดงขั้นตอนการกำหนดไคเรกทอรีสำหรับติดตั้งโปรแกรมภาษาไพธอน	60
ก.4 แสดงรายการคอมโพเนนต์ที่จะถูกติดตั้งลงในระบบ	61
ก.5 แสดงขั้นตอนการดำเนินการติดตั้ง	61
ก.6 แสดงขั้นตอนเสร็จสิ้นการลงโปรแกรม Python	62
ก.7 แสดงขั้นตอนภายหลังจากการดับเบิลคลิกไฟล์ appserv-win32.x.x.x.exe	62
ก.8 แสดงขั้นตอนเงื่อนไขการใช้งานโปรแกรม	63
ก.9 แสดงขั้นตอนการเลือกไคเรกทอรีที่จะทำการติดตั้งโปรแกรม AppServ	64
ก.10 แสดงขั้นตอนการเลือกแพ็คเกจคอมโพเนนต์	64
ก.11 แสดงขั้นตอนการกำหนดค่าคอนฟิกของ Apache Web Server	65
ก.12 แสดงขั้นตอนการกำหนดค่าคอนฟิกของ MySQL Database	65
ก.13 แสดงขั้นตอนเสร็จสิ้นการลงโปรแกรม AppServ	66
ก.14 แสดงขั้นตอนภายหลังจากการดับเบิลคลิกไฟล์ MySQL-python-1.2.2.win32-py2.4.exe	66
ก.15 แสดงขั้นตอนการค้นหาโปรแกรมไพธอนเวอร์ชัน 2.4 สำหรับ MySQL-Python	67
ก.16 แสดงขั้นตอนพร้อมลงโปรแกรมสู่ระบบ	67
ก.17 แสดงขั้นตอนเสร็จสิ้นการลงโปรแกรม MySQL-Python	68

สารบัญรูปภาพ (ต่อ)

รูปที่	หน้า
ก.18 แสดงขั้นตอนภายหลังจากการคลิกไฟล์.....	68
ก.19 แสดงขั้นตอนการค้นหาโปรแกรมไพธอนเวอร์ชัน 2.4 สำหรับ mod_python.....	69
ก.20 แสดงความพร้อมที่จะลงโปรแกรม mod_python.....	69
ก.21 แสดงขั้นตอนการเลือก apache ที่จะทำการแก้ไขค่าคอนฟิก.....	70
ก.22 แสดงขั้นตอนสิ้นสุดการลงโปรแกรม mod_python.....	70



บทที่ 1

บทนำ

1.1 ความเป็นมาของปัญหา

เนื่องจากรถโดยสารประจำทางในปัจจุบันนั้นผู้โดยสารจะไม่ทราบเลยว่าเมื่อต้องการไปที่แห่งหนึ่งจำเป็นจะต้องขึ้นรถโดยสารประจำทางสายใดบ้างและขึ้นสายใดใกล้ที่สุด โดยเฉพาะกับการเดินทางโดยรถโดยสารประจำทางภายในกรุงเทพมหานครที่มีถนนหลายสายมาก และมีรถโดยสารประจำทางหลายสาย ทำให้ผู้โดยสารทั้งที่รู้จักเส้นทางและไม่รู้จักเส้นทางเกิดความยากลำบากในการเลือกเดินทาง บ่อยครั้งที่ผู้โดยสารมักขึ้นรถผิดสายแล้วไปลงผิดที่ โครงการนี้จึงมุ่งเน้นที่จะช่วยอำนวยความสะดวกและช่วยแก้ปัญหาเกี่ยวกับการเดินทางด้วยรถโดยสารประจำทาง ทำให้ผู้โดยสารรู้ว่าจะต้องขึ้นรถสายใด เปลี่ยนรถที่ไหนแล้วต้องขึ้นรถสายใดไปต่อ นอกจากนี้ยังสามารถค้นหาเส้นทางที่มีระยะทางสั้นที่สุด, เส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุด และเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุดและระยะทางสั้นที่สุดได้อีกด้วย เพื่อช่วยให้ผู้โดยสารวางแผนการเดินทางด้วยรถโดยสารประจำทางได้อย่างมีประสิทธิภาพ

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่ออำนวยความสะดวกแก่ประชาชนทั่วไปในการใช้บริการรถโดยสารประจำทาง
- 1.2.2 เพื่อเป็นต้นแบบของระบบขนส่งในอนาคต
- 1.2.3 เพื่อศึกษาข้อดีข้อเสียของระบบรถโดยสารประจำทางในปัจจุบัน
- 1.2.4 เพื่อศึกษาหลักการของ A* ในปัญญาประดิษฐ์

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1.3.1 ทำให้เข้าใจระบบรถโดยสารประจำทางในปัจจุบันและนำข้อดีข้อเสียนั้นไปพัฒนาเป็นระบบให้บริการรถโดยสารประจำทางอัจฉริยะ
- 1.3.2 ทำให้ประชาชนเกิดความสะดวกในการใช้บริการรถประจำทาง
- 1.3.3 ทำให้เข้าใจการค้นหาเส้นทางโดยใช้วิธี A*

1.4 ขอบเขตของโครงการ

- 1.4.1 ผู้ใช้สามารถค้นหาเส้นทางรถโดยสารประจำทางได้โดยการกำหนดจุดต้นทางและจุดปลายทาง
- 1.4.2 ผู้ใช้สามารถค้นหาเส้นทางได้ทั้งแบบระยะทางน้อยที่สุด เปลี่ยนสายรถน้อยที่สุด และรวมทั้ง 2 แบบคือ ระยะทางน้อยที่สุดและเปลี่ยนสายรถน้อยที่สุด
- 1.4.3 ผู้ใช้สามารถทราบระยะทางในการเดินทางหลังจากค้นหาเส้นทางแล้ว
- 1.4.4 ผู้ใช้สามารถเพิ่มจุดที่ต้องการผ่าน เพิ่มขึ้นจากจุดต้นทางและปลายทางในการค้นหา
- 1.4.5 ผู้ใช้สามารถเลือกประเภทของรถประจำทางที่จะใช้ในการเดินทางได้

1.5 เนื้อหาของปริิณญาณินพณ์

เนื้อหาในปริิณญาณินพณ์เล่มนี้จะกล่าวถึงทฤษฎีของแผนที่ ระบบพิกัดทางภูมิศาสตร์ การแปลงพิกัด การเก็บข้อมูลแผนที่ การแสดงผลแผนที่ การควบคุมการแสดงผลของแผนที่ การนำทฤษฎีปัญญาประดิษฐ์มาประยุกต์ใช้งาน และการออกแบบพัฒนา ตามลำดับ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 ภาษา XML (Extensible Markup Language)

XML ถูกกำหนดโดยกลุ่มทำงาน XML ของสถาบัน World Wide Web Consortium (W3C) กลุ่มทำงานกลุ่มนี้ได้บรรยายถึงภาษา XML ไว้ว่า Extensible Markup Language เป็นฟอร์แมตที่อธิบายถึงรายละเอียดของโครงสร้างและแบบของข้อมูล เป็นภาษาหรือชุดคำสั่งเกี่ยวกับข้อมูลบนเว็บที่ทำให้การพัฒนาในส่วนของการสร้างข้อมูลจากหลากหลายแอปพลิเคชัน มานำเสนอบนเครื่องเดสก์ทอปด้วย XML จะทำให้การจัดการข้อมูลหรือเรียกใช้ข้อมูลจากแอปพลิเคชันต่างๆ จะเข้าสู่มาตรฐานเดียวกัน

XML และ HTML เป็นส่วนหนึ่งของภาษา SGML ซึ่ง XML จะให้รายละเอียดเกี่ยวกับข้อมูล เช่น ชื่อเมือง อุณหภูมิ ความกดอากาศ ส่วน HTML เป็นการกำหนด Tag ต่างๆ ที่จะกำหนดรูปแบบการแสดงผลของข้อมูลบนหน้าเว็บ ซึ่งข้อมูลจะสามารถแสดงออกมาได้หลายรูปแบบขึ้นอยู่กับข้อกำหนดของ HTML

2.1.1 ลักษณะโครงสร้างของ XML

XML เป็นการใช้อ็ความเพื่อบ่งบอกโครงสร้างของเอกสารพิจารณาตัวอย่างรูปแบบโครงสร้างของหนังสือ เมื่อหนังสือประกอบด้วยจำนวนบท 2 บท ในแต่ละบทประกอบด้วยเนื้อความ

Begin Book

Begin Chapter 1

Text for Chapter 1

End Chapter 1

Begin Chapter 2

Text for Chapter 2

End Chapter 2

End Book

หนังสือที่มีอยู่ในปัจจุบันจะมีโครงสร้างที่มีรายละเอียดที่ซับซ้อนมากกว่านี้ เช่น บทนำ สารบัญ เป็นต้น เช่นเดียวกัน ภายในส่วนเนื้อความยังประกอบด้วยโครงสร้างย่อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือ ย่อหน้า (Paragraph) แต่ละย่อหน้ายังประกอบขึ้นจากประโยค คำและตัวอักษรด้วย ลักษณะของเอกสาร XML นั้น สามารถอธิบายโดยใช้ตัวอย่างที่ 1 ได้ ดังนี้

```
<?xml version="1.0" encoding="windows-874"?>
<note>
  <to> Tiny </to>
  <from>Michael</from>
  <heading>กรุณาติดต่อ Mr.Michael Tiny</heading>
  <body>Mr.Tiny โทรศัพท์ทางไกลมาบอกว่ามีปัญหาเรื่องภาษีศุลกากร
</body>
</note>
```

บรรทัดที่ 1 นั้นหมายความว่าเราประกาศเอกสารนี้เป็นเอกสาร XML เวอร์ชันที่ 1.0 และมีการเข้ารหัสอักขระเป็น windows-874 เพื่อให้ใช้ภาษาไทยได้ บรรทัดที่ 2 คือ Element ตัวแรกหรือตัวแม่ของเอกสาร หรือที่เรียกว่า root element ที่ทุกๆ เอกสาร XML ต้องมีโดยที่เอกสารหนึ่งเอกสารมีได้เพียง root element เดียวเท่านั้น บรรทัดที่ 3 ถึง 6 มี element 4 ตัวคือ to, from, heading และ body ซึ่งเป็น element ลูกของ element note ส่วนบรรทัดสุดท้ายก็คือแท็กปิดของ element แม่ นั่นเองและทุกๆ element ในเอกสาร XML ต้องมีทั้งแท็กเปิดและแท็กปิด

2.1.1.1 Tag tag ใน XML มีความหมายในลักษณะเดียวกับที่ใช้ใน HTML Tag คือข้อความที่อยู่ระหว่างสัญลักษณ์ < และ > ส่วนชนิดของ tag จะมีทั้ง tag เปิดและ tag ปิด แต่ก็จะมีที่เป็นแท็กเปิดและแท็กปิดอยู่ในตัวเดียวกันด้วยเช่น <ชื่อแท็ก /> ซึ่งแท็กเปิดเป็นตัวบอกว่าเริ่ม element และแท็กปิดเป็นตัวบอกว่าจบ element และความแตกต่างระหว่างแท็กเปิดกับแท็กปิดคือแท็กเปิดสามารถใส่ข้อมูลอธิบายเพิ่มเติมซึ่งเรียกว่า attribute แต่แท็กปิดไม่สามารถทำได้

2.1.1.2 Element Element เป็นส่วนขยายอธิบายความหมายเพิ่มเติมได้อีกและมีความสัมพันธ์ แบบ element แม่กับ element ลูกใน element สามารถใส่ข้อมูลเข้าไปได้โดยข้อมูลที่ใส่เข้าไประหว่างแท็กเปิด กับแท็กปิดเรียกว่า Content และข้อมูลที่ใส่เพิ่มเข้าไปในแท็กเปิดเพื่ออธิบายคุณสมบัติลักษณะของ element เพิ่มเรียกว่า Attribute ทุก element ในเอกสารต้องมีทั้งแท็กเปิดและแท็กปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.3 Content เนื้อความหรือ Content ถือได้ว่าเป็นข้อมูลเพื่อใช้ในการแสดงให้ผู้อ่านเอกสารได้เห็นหรือ กล่าวอีกนัยหนึ่งคือ Content อยู่หลัง Tag เปิด และจบก่อนที่จะถึง Tag ปิดนั่นเอง

2.1.1.4 Attribute Element ในเอกสาร XML สามารถมี Attribute ในแท็กเปิดได้ และมีได้มากกว่า 1 ตัวเหมือนกับในเอกสาร HTML โดยที่ Attribute จะอธิบายถึงคุณสมบัติหรือบอกลักษณะของelement นั้นเพิ่ม คือถ้ากล่าวถึงสิ่งของหากเราไม่ได้ให้ความหมายเพิ่มเติมเราก็จะไม่มีทางรู้หรือแยกแยะสิ่งที่คล้ายๆ กันออกจากกันได้ เช่น ถ้าพูดถึง บท ในหนังสือ แค่นี้เป็นความหมายโดยรวมแต่ถ้าบอกว่าบทที่ 1 ในหนังสือ เลข 1 ในที่นี้คือความหมายเพิ่มเติมให้กับบท

2.1.2 ตัวกำหนดโครงสร้างของ XML ด้วย DTD (Document Type Definition)

Document Type Definition ใช้ในการกำหนดโครงสร้างของเอกสาร XML โดยเมื่อเปิดเอกสาร XML ด้วย DTD ตัวประมวลผลจะตรวจสอบเอกสารว่าถูกต้องตรงกับ DTD ที่ได้กำหนดโครงสร้างไว้หรือไม่ เช่น ถ้าไม่ประกาศ Element หรือ Attribute ไว้ใน DTD แล้วจะไม่สามารถใช้ Element หรือ Attribute นั้นๆ ในเอกสาร XML ได้

DTD มีรูปแบบทั่วไปตามนี้ `<!DOCTYPE Name DTD> Name` ตามตัวอย่างเป็นที่กำหนดชื่อของ Document Element (หรือ Root Element) ชื่อของ Document Element จะต้องตรงกับชื่อที่กำหนดไว้ที่นั่นเท่านั้น DTD

2.1.3 การแปลงรูปแบบเอกสาร XML (Transformation XML)

XSL ซึ่งย่อมาจาก Extensible Style sheet Language (XSL) เป็นภาษาที่ใช้ในการแปลงจากรูปแบบ(Transformation) ของเอกสาร XML ไปอยู่ในรูปแบบอื่นๆ และในขณะเดียวกันก็ทำการประยุกต์รูปแบบการจัดข้อความ (style) ด้วย XSL stylesheet (สไตลชีต) ถูกเขียนในรูปแบบที่เหมาะสมกับ XML แต่ต้องทำการกำหนดลำดับของ Element ไว้ล่วงหน้าในการกำหนดสิ่งที่ต้องการให้กระทำในการแปลงรูปแบบโดยความหมายจะถูกกำหนดโดยตัวประมวลผล XSL เองซึ่งควรจะเป็นไปตามมาตรฐานของ W3C ดังนั้นจาก XSL ซึ่งเป็นแอปพลิเคชันหนึ่งของ XML และอยู่ในรูปแบบที่เหมาะสมกับ XML ดังนั้นจึงสามารถประมวลผล XSL สไตลชีตได้เหมือนกับเอกสาร XML ครั้งแรก XSL เป็นข้อเสนอที่ถูกระบุทั้งทางที่การแปลงรูปแบบ XML และไวยากรณ์ในการจัดการรูปแบบบนแพลตฟอร์มที่เป็นอิสระกัน (platform-independent styling grammar) แต่อย่างไรก็ตามในส่วนของ การแปลงรูปแบบก้าวหน้าอย่างรวดเร็วและเป็นประโยชน์อย่างมาก ขณะที่ไม่มีแอปพลิเคชันในส่วนของ การจัดการรูปแบบมากนักและส่วนใหญ่จะกล่าวว่ามีประโยชน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางด้านการทำงานน้อย ดังนั้นจากข้อเสนอในตอนแรกจึงสามารถแบ่งออกได้เป็น 2 ส่วนที่แตกต่างกันคือ ส่วนในการแปลงรูปแบบเอกสาร ซึ่งปัจจุบันถูกเรียกว่า XSLT (XSL Transformation Language) แต่ในที่นี้จะพิจารณาเฉพาะในส่วนของการแปลงรูปแบบเท่านั้นเพื่อแปลงรูปแบบจากเอกสาร XML แบบหนึ่งไปเป็น XML อีกแบบหนึ่ง เช่นใช้ในการแปลงเอกสาร GML ไปเป็น SVG

2.2 ภาษา SVG (Scalable Vector Graphic)

SVG (Scalable Vector Graphics Language) เป็นภาษาที่ใช้อธิบายภาพกราฟิก 2 มิติและอยู่ในรูปแบบ XML และกราฟิกที่อธิบายนั้นเป็นแบบเวกเตอร์ด้วย สามารถจัดกลุ่ม กำหนดลักษณะและอื่นๆ ของการอธิบายรูปกราฟิก และสามารถที่จะทำให้กราฟิกนั้นมีการทำงานที่มีการปฏิสัมพันธ์กับผู้ใช้งาน โดยใช้ภาษาจาวาสคริปต์ต่างๆ เข้าช่วยและสามารถทำงานในรูปแบบแอนิเมชันได้อีกด้วย

และเนื่องจาก SVG เป็น XML ดังนั้นจึงมีการสร้าง DOM สำหรับ SVG ขึ้นมาโดยอ้างอิงตามมาตรฐานของ W3C ซึ่งทำให้ SVG มีคุณสมบัติไดนามิกและการปฏิสัมพันธ์ตามไปด้วย แม้แต่การจัดการเหตุการณ์ที่เกิดขึ้นที่ตัวกราฟิก อย่างเช่น onmouseover() หรือ onclick() ก็สามารถกำหนดลงในวัตถุของ SVG ได้อีกด้วย

2.2.1 การอธิบายภาพโดยใช้ภาษา SVG

ลักษณะการเขียนภาษา SVG มีลักษณะเป็น XML ซึ่งต้องมี tag เปิดและปิด และมี tag ที่อยู่ใน tag ด้วย ซึ่งจะสังเกตได้ว่า SVG ไฟล์ จะต้องมี <svg> อยู่เป็น tag นอกสุดหรือเป็น root element ซึ่งมีตัวอย่างดังต่อไปนี้

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 20000303 Stylable//EN"
"http://www.w3.org/TR/2003/WD-SVG-20000303/DTD/svg-20000303-
stylable.dtd">
<svg xml:space="preserve" width="5.5in" height=".5in">
...
...
</svg>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในบรรทัดแรกจะเป็นการบอกให้ผู้ที่นำ SVG ไฟล์นี้ไปอ่านทราบว่าเอกสารต่อไปนี้อ้างอิงจากมาตรฐาน XML เวอร์ชัน 1.0 โดยข้อมูลที่อยู่ในแต่ละแท็กจะใช้มาตรฐาน iso-8859-1 ในการเขียนบรรทัดที่สองจะอ้างอิงถึง DTD หรือ Document Type Definition (DTD เป็นเอกสารที่ใช้บอกถึงชนิดของข้อมูลที่จะใช้ใน XML และยังระบุอีกด้วยว่าเอกสารนั้นๆ มีโครงสร้างเป็นอย่างไร) ใน<svg> Element สามารถที่จะบรรจุ text, shapes และ paths ได้ เช่น

2.2.2 Vector Graphic Shapes

2.2.2.1 rect ใช้สำหรับสร้างและปรับแต่งรูปสี่เหลี่ยม

ตัวอย่าง

```
<rect x="20" y="20" rx="20" ry="20" width="250" height="100" style="fill:red;stroke:black;stroke-width:5; opacity:0.5"/>
```

คำอธิบาย แอตทริบิวต์

x, y : ใช้กำหนดตำแหน่งของรูปสี่เหลี่ยมที่สร้าง

rx, ry : ใช้กำหนดส่วนโค้งบริเวณมุมสี่เหลี่ยม

width, height : ใช้กำหนดขนาดความกว้างและความสูงตามลำดับ

style : ใช้ในการกำหนดลักษณะของวัตถุ

fill, stroke : ใช้กำหนดสีของวัตถุและสีของเส้นขอบวัตถุตามลำดับ

stroke-width : ใช้กำหนดความหนาของเส้น

opacity : ใช้กำหนดความโปร่งใสของวัตถุมีค่าระหว่าง 0-1

2.2.2.2 circle ใช้สำหรับสร้างและปรับแต่งรูปวงกลม

ตัวอย่าง

```
<circle cx="100" cy="50" r="40" stroke="black" stroke-width="2" fill="red" />
```

คำอธิบายแอตทริบิวต์

cx, cy : ใช้กำหนดจุดศูนย์กลางของวงกลม หากไม่กำหนดมีค่าเป็น 0, 0

r : ใช้กำหนดรัศมีของวงกลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2.3 ellipse ใช้สำหรับสร้างและปรับแต่งรูปวงรี

ตัวอย่าง

```
<ellipse cx="300" cy="150" rx="200" ry="80" style="
fill:rgb(200,100,50);stroke:rgb(0,0,100);stroke-width:2" />
```

คำอธิบาย แอตทริบิวต์

cx, cy : ใช้กำหนดจุดศูนย์กลางของวงกลม หากไม่กำหนดจะมีค่าเป็น 0, 0
rx, ry : ใช้กำหนดรัศมีในแนวนอนและแนวตั้งตามลำดับ

2.2.2.4 line ใช้สำหรับสร้างและปรับแต่งเส้นตรง

ตัวอย่าง

```
<line x1="0" y1="0" x2="300" y2="300"
style="stroke:rgb(99,99,99);stroke-width:2" />
```

คำอธิบาย แอตทริบิวต์

x1, y1 : ใช้กำหนดตำแหน่งจุดเริ่มต้นของเส้นตรง

x2, y2 : ใช้กำหนดตำแหน่งจุดสิ้นสุดของเส้นตรง

2.2.2.5 polygon ใช้สำหรับสร้างและปรับแต่งรูปหลายเหลี่ยม

ตัวอย่าง

```
<polygon points="220,100 300,210 170,250"
style="fill:#cccccc;stroke:#000000;stroke-width:1" />
```

คำอธิบาย แอตทริบิวต์

point ใช้กำหนดตำแหน่งของมุมแต่ละมุม (x,y)

2.2.2.6 polyline ใช้สำหรับสร้างและปรับแต่งรูปเส้นตรงหลายเส้นเชื่อมต่อกัน

ตัวอย่าง

```
<polyline point="0,0 0,20 20,20 20,40 40,40 40,60"
style="fill:#cccccc;stroke:#000000;stroke-width:1" />
```

คำอธิบาย แอตทริบิวต์

point ใช้กำหนดตำแหน่งจุดเชื่อมต่อของเส้น (x,y)

2.2.2.7 path ใช้สำหรับสร้างและปรับแต่งรูปร่างต่างๆโดยจะมีคำสั่งที่ใช้ในการสร้างรูปร่างดังนี้

ตารางที่ 2.1 แสดงคำสั่งที่ใช้ในการสร้างรูปร่าง

M = moveto	S=smooth curveto
- A = elliptical Arc	L = lineto
- V = vertical lineto	T = smooth quadratic Belzier curveto
Q=quadratic Belzier curve	C = curveto
- H = horizontal lineto	Z = closepath

ตัวอย่าง

```
<path d="M250 150 L150 350 L350 350 Z" />
```

คำอธิบาย

จากตัวอย่างเป็นการสร้าง path โดยเริ่มต้นที่จุด 250,150 จากนั้นลากเส้นตรงไปยังจุด 150,350 จากนั้นลากเส้นตรงไปยังจุด 350,350 จากนั้นทำการปิด path โดยกลับไปอยู่ที่จุด 250,150

2.2.3 ประโยชน์ในการใช้ SVG

- SVG สามารถอ่านและแก้ไขได้โดยโปรแกรมหลายโปรแกรม เช่น notepad
- สามารถเปลี่ยนขนาดของภาพได้โดยคุณภาพของภาพไม่ลดลง
- SVG สามารถทำงานร่วมกับภาษาจาวาได้
- SVG เป็นมาตรฐานที่เปิดเผยให้คนทั่วไปทราบ
- ข้อความใน SVG มีความสามารถในการเลือกและค้นหาจึงเหมาะแก่การนำมาทำเป็นแผนที่

โดย element และ attribute ของ SVG ที่ใช้งานบ่อยมีดังต่อไปนี้

ตารางที่ 2.2 SVG Element

SVG	Action
<svg>	เป็น tag ที่ใช้ในการกำหนดพื้นที่ทั้งหมดของภาพและเป็น tag ที่เก็บ tag ต่างๆ ของ SVG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 SVG Element (ต่อ)

<path>	ใช้ในการวาดเส้น
<rect>	ใช้ในการวาดสี่เหลี่ยม
<g>	ใช้ในการรวม tag ต่างๆ ให้อยู่ในกลุ่มเดียวกัน
<desc>	ใช้ในการเพิ่มเติมข้อมูลหรือคำอธิบาย
<defs>	ใช้ในการกำหนดค่าต่างๆ
<script>	ใช้ในการเขียน script จะเขียนใน tag นี้
<text>	ใช้ในการแสดงผลแบบอักษร
<cursor>	ใช้ในการเปลี่ยนรูปของ mouse cursor
<line>	ใช้ในการวาดเส้นอย่างง่าย
<textPath>	ใช้ในการเขียนอักษรให้เป็นไปตามเส้นทางที่กำหนด
<image>	ใช้ในการแสดงรูปภาพในรูปแบบต่างๆ เช่น jpeg, gif, png เป็นต้น

ตารางที่ 2.3 SVG Attribute

Attribute	Action
fill	จัดการกับสีของ object นั้นๆ
id	ใช้ในการแยกแยะ object ต่างๆ
Width	ขนาดความกว้างของ object
Height	ขนาดความสูงของ object
X	ตำแหน่งพิกัด x
Y	ตำแหน่งพิกัด y
Xmlns	ใช้ในการอ้างอิงถึง namespace
Visibility	กำหนดการปรากฏอยู่ของ object
Style	กำหนดลักษณะของ object
point-event	กำหนดการปฏิสัมพันธ์กับ object
viewBox	กำหนดขนาดในการแสดงผลของภาพ
stroke-width	กำหนดขนาดของเส้น
Stroke	กำหนดสีของเส้น
zoomAndPan	กำหนดความสามารถในการย่อหรือขยาย
OnClick	mouse event
onmouseover	mouse event

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 SVG Attribute (ต่อ)

onmouseout	mouse event
Transform	เคลื่อนย้ายหรือหมุนปรับอัตราส่วนของภาพ
xlink:href	เป็นการอ้างถึง URL

2.3 ภาษา Java Script

จาวาสคริปต์ เป็นภาษาสคริปต์ ที่สมบูรณ์ใกล้เคียงกับภาษาสำหรับการเขียนโปรแกรมปกติ จาวาสคริปต์มีคุณสมบัติพื้นฐานต่างๆ เช่นเดียวกับภาษาคอมพิวเตอร์ทั่วไป ไม่ว่าจะเป็นการกำหนดตัวแปรการกำหนดฟังก์ชันการควบคุมการทำงานของ โปรแกรมสแตทเมนต์ และชุดของโอเปอเรเตอร์ต่างๆ นอกจากนี้จาวาสคริปต์ยังมีคุณสมบัติในการเขียนโปรแกรมแบบอ็อบเจกต์ บางอย่างไว้ด้วยดังเช่น จาวาสคริปต์สามารถเข้าไปใช้กับอ็อบเจกต์ ซึ่งปรากฏอยู่บนเว็บหรือจะแก้ไข Attribute รวมทั้งอีเวนต์ไว้ให้เรียกใช้งานด้วย จาวาสคริปต์นั้นช่วยให้เพจ HTML สามารถทำงานได้ดังเช่นนักพัฒนาเว็บอาจจะสร้างให้เว็บเพจของตนสามารถตอบสนองต่อเหตุการณ์ หรืออีเวนต์ของผู้ใช้ได้ไม่ว่าจะเป็นการคลิกเมาส์ หรือว่าการเรียกฟอร์มของ HTML ข้อดีของจาวาสคริปต์ที่สำคัญอีกประการหนึ่งคือ โครงสร้างของภาษากลายภาษาจาวา ซึ่งง่ายต่อการเรียนรู้และใช้งาน

ความแตกต่างระหว่างภาษาจาวาและจาวาสคริปต์ที่เห็นได้ชัดเจนคือ จาวาสคริปต์มีโครงสร้างของภาษาที่ไม่ซับซ้อนเหมือนกับจาวา นักพัฒนาเว็บที่คุ้นเคยกับ Tag HTML และเคยใช้งานภาษาในการเขียนภาษาซี สามารถเรียนรู้และใช้งานภาษาสคริปต์ได้ในเวลาไม่นานนอกจากนั้น โปรแกรมที่เขียนด้วยภาษาสคริปต์นั้น ไม่จำเป็นที่จะต้องคอมไพล์เหมือนโปรแกรมจาวาทั่วไป ทั้งนี้เนื่องจากเป็นภาษาสคริปต์ โปรแกรมเว็บเบราว์เซอร์จึงมีตัวทำหน้าที่ปฏิบัติตามสคริปต์ที่เขียนไว้ทันที ในการพัฒนาแผนที่โดยใช้ภาษา GML เราสามารถใช้ภาษาจาวาสคริปต์ในการเข้าถึง Element หรือ Attribute ของโครงสร้างภาษา SVG เพื่อสร้างส่วนของการปฏิสัมพันธ์กับผู้ใช้แผนที่ เช่น การย่อ หรือ ขยายแผนที่ การเลื่อนแผนที่หรือ การนำข้อมูลรายละเอียดของสถานที่ต่างๆ ขึ้นมาแสดงโดยใช้ภาษา SVG

2.3.1 JavaScript คืออะไร

JavaScript เป็นภาษายุคใหม่สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง เราสามารถเขียน โปรแกรม JavaScript เพิ่มเข้าไปในเว็บเพจเพื่อใช้ประโยชน์สำหรับงานด้านต่าง ๆ ทั้งการคำนวณ การแสดงผล การรับ-ส่งข้อมูล และที่สำคัญคือสามารถโต้ตอบกับผู้ใช้ได้อย่างทันทีทันใด นอกจากนี้ยังมีความสามารถด้านอื่น ๆ อีกหลายประการที่ช่วยสร้างความน่าสนใจให้กับเว็บเพจของเราได้อย่างดี

2.3.2 ลักษณะการทำงานของ JavaScript

JavaScript เป็นภาษาสคริปต์เชิงวัตถุ หรือเรียกว่า Object Oriented Programming ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนเอกสารด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ ทำงานร่วมกับ ภาษา HTML และภาษาจาวาได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server) โดยมีลักษณะการทำงานดังนี้

2.3.2.1 Navigator JavaScript เป็น Client-Side JavaScript ซึ่งหมายถึง

JavaScript ที่ถูกแปลทางฝั่งไคลเอนต์ (หมายถึงฝั่งเครื่องคอมพิวเตอร์ของผู้ใช้ไม่ว่าจะเป็นเครื่องพีซี เครื่องแมคอินทอช หรืออื่นๆ) จึงมีความเหมาะสมต่อการใช้งานของผู้ใช้ทั่วไป เป็นส่วนใหญ่

2.3.2.2 LiveWire JavaScript เป็น Server-Side JavaScript ซึ่งหมายถึง

JavaScript ที่ถูกแปลทางฝั่งเซิร์ฟเวอร์ (หมายถึงฝั่งเครื่อง คอมพิวเตอร์ของผู้ให้บริการเว็บ โดยอาจจะเป็นเครื่องของซัน ซิลิคอนกราฟิก หรืออื่น ๆ) สามารถใช้ได้เฉพาะกับ LiveWire ของเน็ตสเคป โดยตรง

ตัวแปร (Variable) หมายถึง ชื่อหรือสัญลักษณ์ที่ตั้งขึ้นสำหรับการเก็บค่าใดๆ ที่ไม่คงที่ โดยการจองเนื้อที่ในหน่วยความจำของระบบเครื่องที่เก็บข้อมูลซึ่งสามารถอ้างอิงได้ มีขนาดขึ้นอยู่กับชนิดของข้อมูลและค่าของข้อมูล ซึ่งค่าในตัวแปรนี้สามารถเปลี่ยนแปลงได้ตามคำสั่งในการประมวลผล

การตั้งชื่อ (Identifier or Name) เป็นชื่อที่ตั้งขึ้นมาเพื่อกำหนดให้เป็นชื่อของโปรแกรมหลัก, ฟังก์ชัน, ตัวแปร, ค่าคงที่, คำสั่ง และคำสงวน โดยมีหลักการตั้งชื่อว่า

- ขึ้นต้นด้วยตัวอักษรในภาษาอังกฤษ ตามด้วยตัวอักษรหรือตัวเลขใด ๆ ก็ได้
- ห้ามเว้นช่องว่าง
- ห้ามใช้สัญลักษณ์พิเศษ ยกเว้นขีดล่าง () และ ดอลลาร์ (\$)
- สำหรับความยาวของชื่อใน JavaScript จะมีความยาวเท่าใดก็ได้ แต่ที่นิยมใช้ไม่เกิน 20 ตัวอักษร
- การตั้งชื่อมีข้อพึงระวังว่า จะต้องไม่ซ้ำกับคำสงวน (Reserve word) และตัวอักษรของชื่อจะจำแนกแตกต่างกันระหว่างอักษรตัวพิมพ์เล็กกับอักษรตัวพิมพ์ใหญ่
- ควรจะตั้งชื่อโดยให้ชื่อนั้นมีสื่อความหมายให้เข้ากับข้อมูล สามารถอ่านและเข้าใจได้

คำสงวน (Reserve word) เป็นคำที่มีความหมายเฉพาะตัวในภาษา JavaScript สงวนไม่ให้มีการตั้งชื่อซ้ำกับชื่อโปรแกรม, ฟังก์ชัน, ตัวแปร, ค่าคงที่ และคำสั่ง คำสงวนสามารถเรียกใช้ได้ทันทีโดยไม่ต้องมากำหนดความหมายใหม่แต่อย่างใด

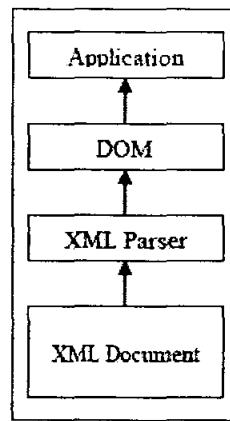
ชนิดของข้อมูลของตัวแปร (Data Type) เป็นการกำหนดประเภทค่าของข้อมูลให้กับตัวแปร เพื่อให้เหมาะสมกับการอ้างอิงข้อมูลจากตัวแปรในการใช้งาน ชนิดข้อมูลของตัวแปรนั้นมีอยู่ด้วยกัน 4 ชนิด ได้แก่

- Number หมายถึง ข้อมูลชนิดตัวเลข ประกอบด้วย เลขจำนวนเต็ม (Integer) และเลขจำนวนจริง (Floating)
- Logical หมายถึง ข้อมูลทางตรรกะ มี 2 สถานะ คือ จริง (True) และเท็จ (False)
- string หมายถึง ข้อมูลที่เป็นข้อความ ซึ่งจะต้องกำหนดไว้ในเครื่องหมายคำพูด ("...")
- Null หมายถึง ไม่มีค่าข้อมูลใดๆ ซึ่งค่า null ใช้สำหรับการยกเลิกพื้นที่เก็บค่าของตัวแปรออกจากหน่วยความจำ

2.4 DOM (Document Object Model)

การใช้งาน XML นั้นเราสามารถที่จะทำอะไรกับข้อมูลในแอปพลิเคชันที่เราเขียนขึ้นได้ ไม่ใช่เพียงการเข้าถึงเท่านั้นแต่ยังรวมถึงการแก้ไขและเพิ่มเติมเอกสาร XML ที่เรามีอยู่อีกด้วย จึงเกิดมี Document Object Model (DOM) ขึ้นเพื่อช่วยในการทำงานกับเอกสาร XML ทั้งหมด DOM มักจะถูกใส่เข้าไปเป็นชั้นที่กั้นกลางระหว่าง XML parser กับแอปพลิเคชันที่ต้องการใช้ข้อมูลในเอกสาร XML ซึ่งหมายความว่า parser จะอ่านข้อมูลจากเอกสาร XML ให้กับ DOM จากนั้น DOM จะถูกใช้โดยแอปพลิเคชันระดับสูงกว่า แอปพลิเคชันสามารถทำงานกับเอกสาร XML ได้ตามที่ต้องการ รวมถึงการใส่ DOM เข้าไปใน DOM ตัวอื่น ซึ่งในแต่ละภาษาก็จะมีแบบจำลอง DOM เป็นของตนเอง

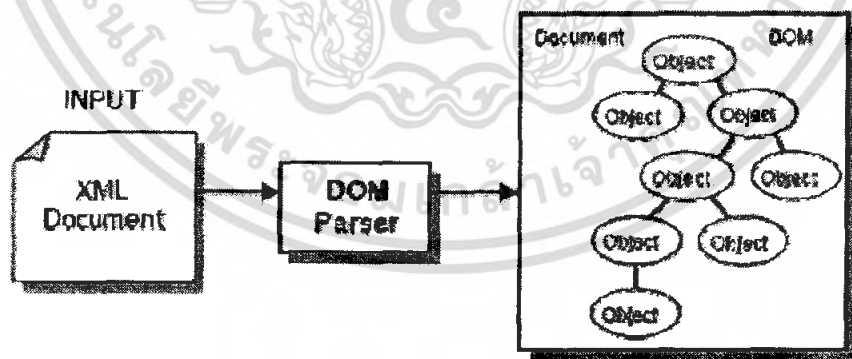
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 การใช้งาน DOM

2.4.1 DOM Parsing

DOM (Document Object Model) เป็นการทำงานแบบ Tree-Based parser และใช้การเข้าถึงข้อมูลด้วยวิธีการ Random-Access คือจะประมวลโครงสร้างของเอกสาร XML ให้เป็นโครงสร้างต้นไม้ (Tree structure) แบบ hierarchical object model ซึ่งในโครงสร้างต้นไม้จะประกอบไปด้วยอีลิเมนต์โหนด (element node) โดยภายในแต่ละอีลิเมนต์โหนดประกอบไปด้วยข้อมูลที่เกี่ยวข้องกับอีลิเมนต์นั้นตามโครงสร้างและข้อมูลในเอกสาร XML ซึ่งจะประกอบไปด้วยข้อมูลทั้งหมดของเอกสาร XML เพื่อให้แอปพลิเคชันสามารถเข้าหาจุดต่างๆ ของ tree structure ได้ โดยที่ DOM จะโหลด XML ทั้งหมด (ข้อมูลใน Root Node ทั้งหมด) เข้ามาเป็น Tree ในหน่วยความจำหลักของคอมพิวเตอร์ (memory) ก่อนจึงทำงานได้ดังแสดงในรูป



รูปที่ 2.2 แสดงวิธีการเข้าถึงข้อมูลโดยวิธี DOM Parsing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

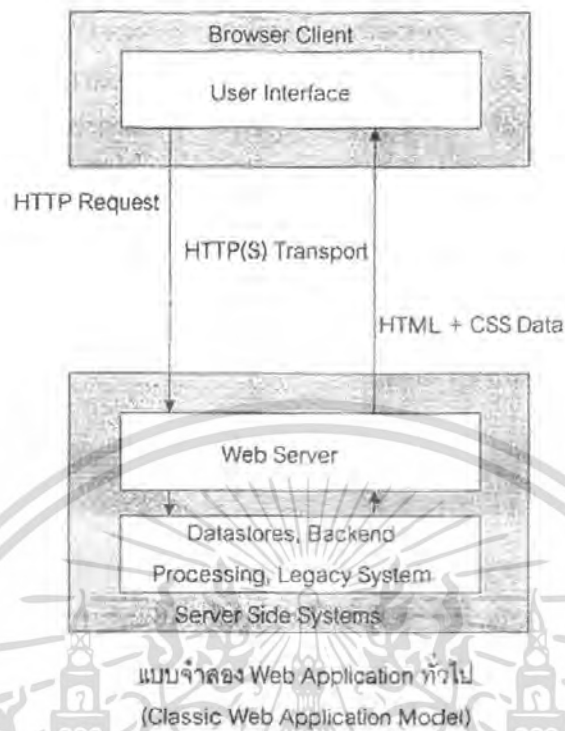
2.5 AJAX (Asynchronous Javascript and XML)

AJAX ย่อมาจาก “Asynchronous Javascript And XML” จากคำที่นำมารวมกันคือ เทคโนโลยีต่างๆ แสดงให้เห็นว่า AJAX นั้นไม่ใช่เทคโนโลยีใหม่ แต่เป็นการนำเทคโนโลยีต่างๆ มารวมเข้าไว้ด้วยกัน ซึ่งได้แก่ DHTML\CSS\JavaScript\DOM/XML และ XMLHttpRequest Object เป็นส่วนประกอบ โดยแนวคิดของเทคโนโลยีนี้ คือ การต้องการให้ผู้ที่ใช้เว็บแอปพลิเคชันสามารถใช้งานได้รวดเร็วเหมือน แอปพลิเคชันทั่วๆ ไปที่ไม่ได้ทำงานบนเว็บ โดยให้การโต้ตอบกับ user ใช้เวลาน้อยที่สุด ซึ่งทำให้เกิดเทคโนโลยีนี้ เพื่อการแสดงผลข้อมูลที่รวดเร็ว

2.5.1 โครงสร้างของ AJAX เว็บแอปพลิเคชัน

ในเว็บแอปพลิเคชันแบบเดิมนั้นการทำงานจะเกิดจากการที่ web browser ของ ผู้ใช้บริการร้องขอข้อมูลเว็บเพจจาก web server ซึ่ง server จะจัดการนำหน้าเพจที่ร้องขอ จัดส่งกลับมาให้ web browser จัดการนำหน้านั้นขึ้นมาแสดงผล เมื่อทำการส่งเพจเสร็จสิ้น การสื่อสารนั้นถือเป็นอันจบการติดต่อระหว่าง server กับ ผู้ใช้บริการ

ดังนั้นการขอเพจหน้าใหม่ เท่ากับว่าต้องเริ่มทำการติดต่อใหม่ ผู้ใช้บริการต้องเข้าไปขอเพจจาก server ใหม่ แล้ว server ก็จัดส่งเพจใหม่ทั้งเพจกลับไปทั้งหน้า แล้ว web browser ก็ดึงเพจหน้านั้นขึ้นมาแสดงผลแทนที่เพจเก่าทั้งหน้า ผลที่เกิดขึ้นกับการแสดงผล คือ เกิดการรอกการเปลี่ยนหน้าเพจทั้งหน้า จะเกิดการกระพริบจากการเปลี่ยนเพจ (refresh) เท่ากับว่าการที่จะใช้งานเพจนี้ ต้องรอทั้งให้ server จัดส่งหน้าเพจทั้งหน้าที่เป็นรูปร่าง โครงสร้างเพจแล้วมาให้ และยังคงให้ web browser ทำการ refresh การแสดงผล ซึ่งลักษณะการทำงานแบบนี้ทำให้เกิดความล่าช้า สิ้นเปลืองเวลา ในกรณีที่เพจนั้นมีข้อมูลที่จะต้องส่งมามาก แต่เพียงต้องการแค่ข้อมูลบางส่วนที่เปลี่ยนแปลงในหน้าเดิม เท่ากับว่า server ต้องส่งหน้านั้นกลับเข้ามาอีกทั้งหน้าแต่เปลี่ยนแค่ข้อมูลส่วนนั้น ในการส่งเพจอย่างนี้ทำให้สิ้นเปลือง แบนด์วิดท์ของการสื่อสาร โดยไม่จำเป็น

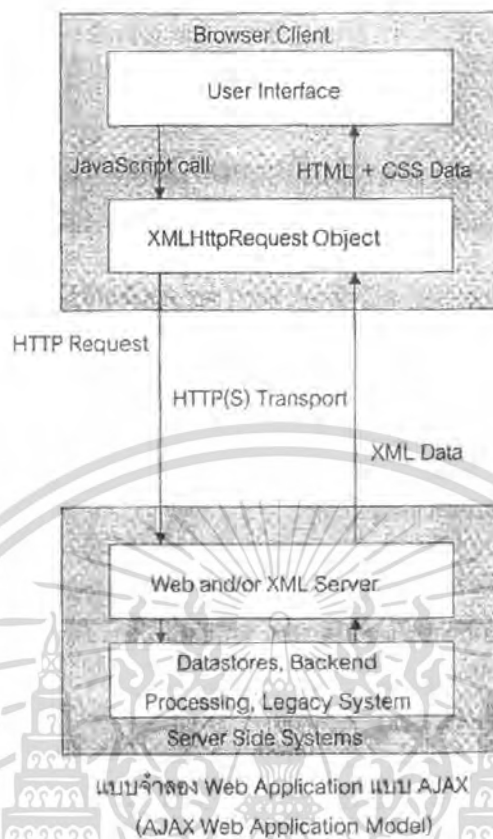


รูปที่ 2.3 แสดงแบบจำลอง Web Application แบบทั่วไป

ปัญหาต่างๆเหล่านี้จึงก่อให้เกิด การใช้งาน web application ได้ไม่เหมาะสมกับความต้องการ ในบางงานที่ต้องการการตอบสนองที่รวดเร็ว ความสามารถของเว็บแอปพลิเคชันเดิมๆ จึงถูกจำกัดขอบเขตให้เหมาะสมกับเฉพาะ ในบางงานเท่านั้น เพราะถ้าเทียบกับแอปพลิเคชันในเครื่องทั่วไปแล้ว ความรวดเร็วยังต่างกันอยู่

จากข้อบกพร่องที่เป็นจุดด้อยตรงนี้ จึงได้คิดค้นวิธีที่จะทำให้การตอบสนองของเว็บแอปพลิเคชันนั้นรวดเร็วกว่าเดิม ก่อเกิดการนำเทคโนโลยีต่างๆมารวมกันเพื่อแก้ไขรูปแบบโครงสร้างการทำงานเสียใหม่ เป็นเทคโนโลยีที่ชื่อว่า AJAX ซึ่งเว็บแอปพลิเคชันของ AJAX นั้นจะช่วยลดการ refresh หน้าเพจ ทำให้การแสดงผลนั้นมีความเรียบลื่น และจัดการรูปแบบของเอกสารที่ส่งมาจาก server ให้มีขนาดเล็กลง โดยเปลี่ยนจากการส่งหน้าเพจ มาเป็นการส่งชิ้นข้อมูล XML เล็กๆเพียงในส่วนที่ต้องการเท่านั้น ทำให้การใช้ แบนด์วิท นั้นลดลงกว่าแบบเดิม การส่งข้อมูลจึงเกิดความรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



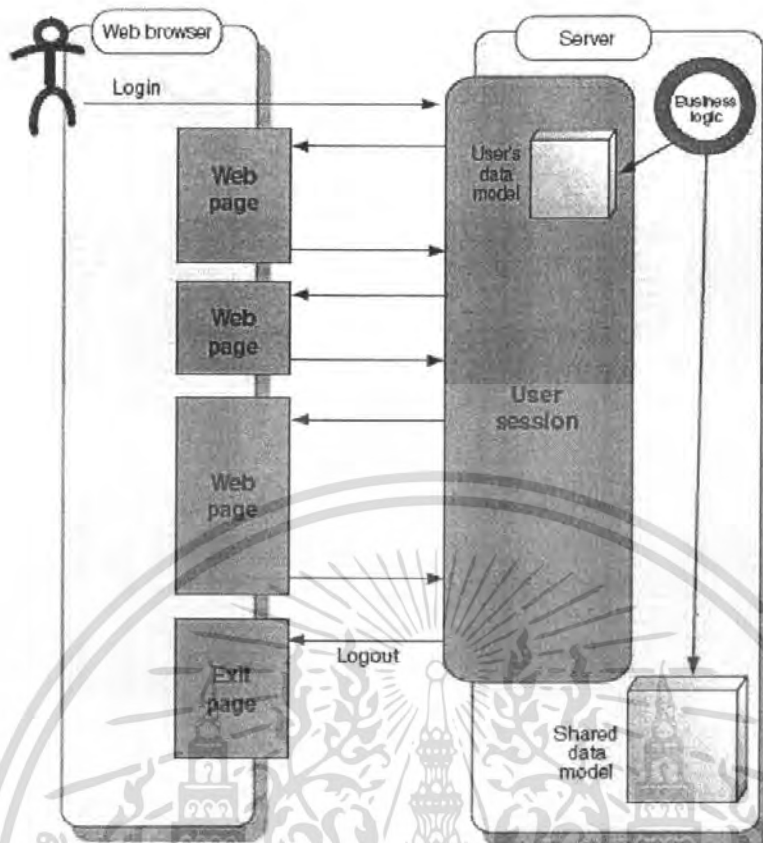
รูปที่ 2.4 แสดงแบบจำลอง Web Application ที่พัฒนาโดย AJAX

จากรูปในส่วน XMLHttpRequest Object นั้นเป็นส่วนสำคัญที่ AJAX จำเป็นต้องมีเป็นส่วนประกอบหลัก เพื่อใช้ปรับปรุงวิธีการรับส่งข้อมูลจากแบบเก่า โดยมีหน้าที่ติดต่อร้องขอและรับข้อมูลจาก server ซึ่งข้อมูลที่ใช้ในการสื่อสาร เป็นข้อมูล XML เฉพาะส่วนที่ต้องการ ส่วนของการแสดงผลนั้นใช้ javascript เข้ามาช่วยจัดการการแสดงผล โดยนำข้อมูลที่รับมาขึ้นมาแสดงผลเฉพาะส่วนที่เกิดการเปลี่ยนแปลงทำให้ไม่ต้อง refresh หน้าเว็บ เพราะส่วนที่ไม่เปลี่ยนแปลงยังคงอยู่ ถือเป็นจุดเด่นของเทคโนโลยีนี้

2.5.2 หลักการในการทำงานของเทคโนโลยี AJAX

ฝั่งผู้ใช้บริการ (browser host) ต้องใช้ Application สำหรับ web browser ตามแบบจำลองแบบเก่าจะมีหน้าที่แค่แสดงผลหน้าเพจเท่านั้น web browser ไม่สามารถรู้ส่วนเกี่ยวข้องของกลไกการนำเพจนั้นมาแสดงผล การตอบสนองต่อ user นั้นมาจาก server ดังแสดงในรูป

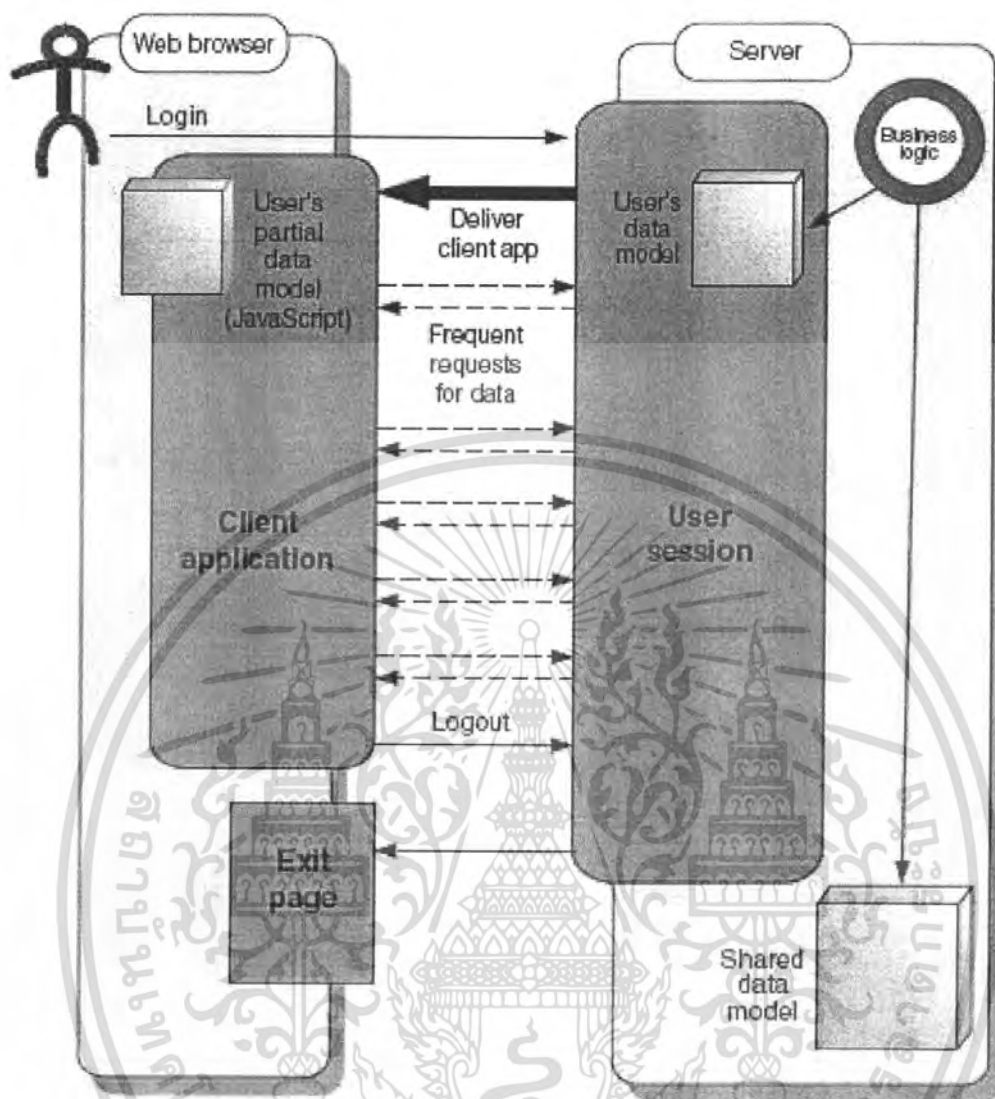
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แสดงแบบจำลองการสื่อสาร ระหว่าง Web browser กับ Server แบบปกติ

จากรูป เมื่อ user เริ่ม login เพื่อร้องขอเว็บเพจจาก server การทำงานส่วนใหญ่ เพื่อให้ได้ตอบสนองต่อ user จะเกิดที่ฝั่ง server จากนั้น user จะได้น้ำหน้าเว็บเพจพร้อมกับการตอบสนองมาจาก server นำมาแสดงผลที่ web browser และเมื่อ user ร้องขอเว็บเพจเพิ่มเติม ก็จะได้รับเว็บเพจขึ้นมาอีกเป็นหน้าใหม่มาแทนที่หน้าเดิม ซึ่งเท่ากับข้อมูลในการสื่อสารเป็นเพจ เปลี่ยนไปเรื่อยๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 แสดงแบบจำลองการสื่อสาร ระหว่าง Web browser กับ Server ที่ใช้ AJAX

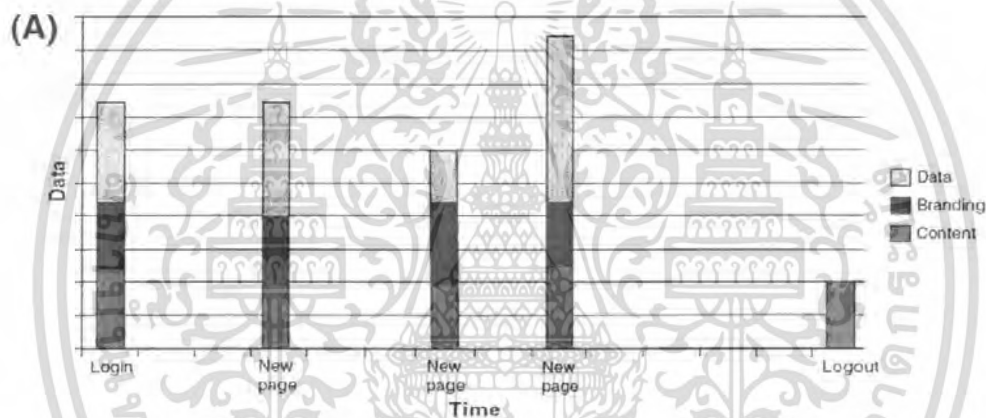
แต่สำหรับในเทคโนโลยี AJAX จากรูปนั้น เมื่อ user เริ่ม login และร้องขอข้อมูลจาก server ครั้งแรกนั้น เอกสารที่ส่งมายัง web browser นั้นจะเป็น javascript เอกสารนี้จะยังคงอยู่กับ web browser ตลอดเวลาเป็นตัว Application เพื่อใช้งานโต้ตอบ user โดยแบ่งใช้เป็น 2 กรณีคือ ตอบสนองแบบทันทีทันใดโดยไม่ติดต่อ server ใช้กับกรณีงานที่สามารถตอบเองได้โดยไม่ต้องติดต่อ server และอีกแบบคือตอบสนองโดยติดต่อกับ server เพื่อดึงข้อมูลที่ต้องการก่อน มักใช้กับงานฐานข้อมูลหรือการใช้ทรัพยากรอื่นที่ต้องติดต่อกับ server เพื่อเอาข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

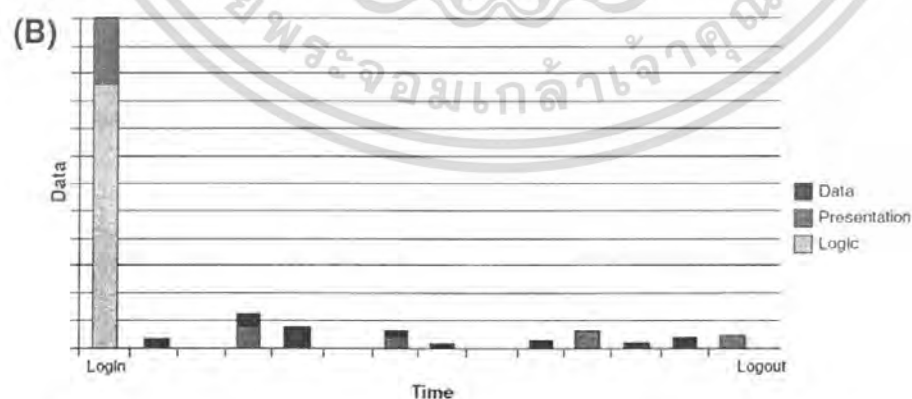
2.5.3 ฝั่ง server ต้องจัดส่งเพียงแค่ข้อมูลที่ร้องขอ

ในเว็บแอปพลิเคชันแบบเก่า นั้น จะได้รับเพจที่มีลักษณะหน้าเหมือนกันซ้ำๆ โดยไม่จำเป็น ซึ่งหน้าเพจที่คล้ายกันนี้ จะมีลักษณะเป็น template โครงสร้างแบบเดียวกัน ดังนั้น server จัดส่งเพจไปนั้นข้อมูลที่ส่งไปนั้นก็จะเป็นเพจเดิมๆ ที่ต้องจัดส่งเกินความจำเป็น ยกตัวอย่างในกรณีเว็บสั่งสินค้าที่ต้องการอัปเดตเฉพาะราคาในตระกร้าจะต้องทำการอัปเดตเพจทั้ง เพจใหม่ เพื่อเพียงเปลี่ยนแค่ราคาในตระกร้า แต่ส่วนอื่นๆก็ยังคงเหมือนเดิม ซึ่ง server จะต้อง จัดส่งข้อมูลที่เกินความจำเป็นให้เปลืองแบนด์วิดท์

แต่ในเว็บแอปพลิเคชันที่ใช้เทคโนโลยี AJAX จะช่วยทำงานได้ดีกว่าเดิม โดยการส่งคำร้องขอไปยัง server แบบ Asynchronous โดยส่วนสำคัญหลักทั้งโครงสร้าง หน้าตา เดิม จะอยู่ที่ฝั่งผู้ใช้บริการเรียบร้อยแล้ว ดังนั้น server จึงจัดส่งข้อมูลที่ตรงกับคำร้องขอเท่านั้น

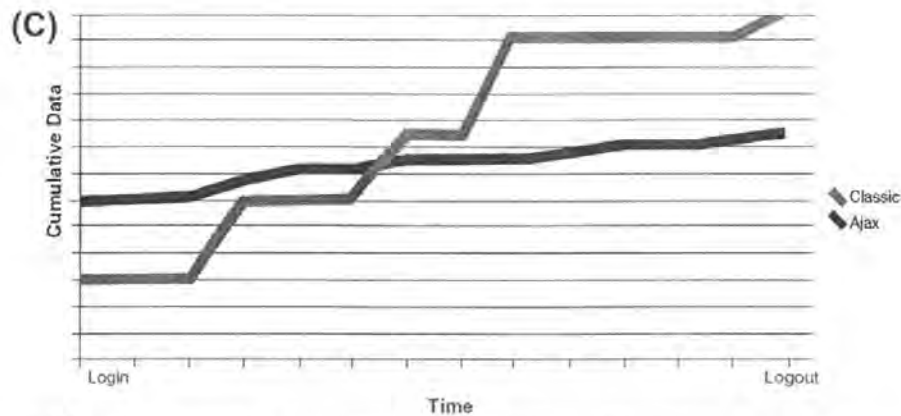


รูปที่ 2.7 แสดง โดอะแกรมข้อมูลที่ทำการสื่อสารในเว็บแอปพลิเคชันปกติ



รูปที่ 2.8 แสดง โดอะแกรมข้อมูลที่ทำการสื่อสารในเว็บแอปพลิเคชันที่ใช้ AJAX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 แสดงกราฟเส้นเพื่อให้เห็นการใช้ข้อมูลระหว่างแบบปกติและ AJAX

ใน AJAX Application นั้นจะเห็นว่าการส่งข้อมูลนั้น จะหนักไปที่ส่วนโหลดครั้งแรกสุด ด้วยส่วนประกอบของ AJAX ที่ส่งไปไว้ที่ฝั่งผู้ให้บริการ ในการเริ่ม login แต่ในแง่ของ traffic ที่สะสมนั้นจะน้อยกว่าในแบบเดิม เวลาเฉลี่ยในการโต้ตอบผู้ให้บริการก็เร็วกว่า แอมการใช้ bandwidth ก็น้อยกว่าด้วย

2.5.4 การตอบสนองคือผู้ใช้ต้องมีความถี่ไหลและต่อเนื่อง

ในส่วนของเว็บเบราว์เซอร์นั้นจะมี input 2 อย่างที่เป็นกลไกในการทำงานคือ hyperlinks และ html forms

hyperlinks นั้นถูกสร้างบน server และนำมาด้วย common gateway interface \ (CGI) เป็นพารามิเตอร์ชี้ตำแหน่ง dynamic server pages ซึ่ง hyperlinks นี้เราสามารถตกแต่งด้วยรูปภาพ และ Cascade style sheet (CSS) เพื่อใช้สำหรับการโต้ตอบเบื้องต้น เช่น ในเวลาที่เราเมาส์ไปอยู่เหนือลิงค์ก็เกิดการเปลี่ยนแปลงที่ลิงค์ เช่น สี ลึน รูปร่างลิงค์ โดยในการออกแบบ hyperlinks ควรให้ดูกลมกลืน กับส่วนของ user interface อื่นๆด้วย

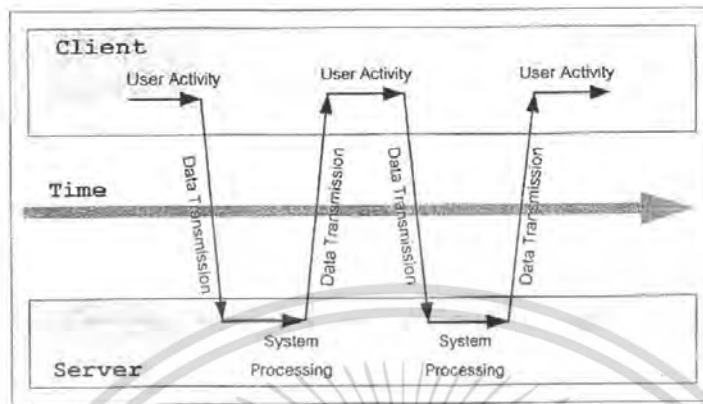
Forms controls เป็นส่วนที่เป็นคอมโพเนนท์พื้นฐานของ user interface จำพวก textbox , checkbox , radio button และ drop down list เป็นต้น

ทางเลือกที่จะช่วยเข้ามาจัดการ hyperlinks และ form controls ก็ Javascript function เป็นเทคนิคพื้นฐานในเว็บเพจซึ่งใช้งานได้ เช่น ใช้ตรวจสอบ empty fields , ค่าตัวเลขที่เกินช่วง นำไปใช้งานเพิ่มลูกเล่นให้กับเว็บเพจ เช่น แจ้งข้อความเตือน

ดังนั้นรูปแบบสื่อสารของ AJAX จะเป็นแบบ Asynchronous ซึ่งต่างจากแอปพลิเคชันแบบเดิมซึ่งเป็นแบบ Synchronous โดยการทำงานติดต่อก็จะอาศัยส่วนทำงานติดต่อ XMLHttpRequest Object ซึ่งจะถูกนำมาอยู่ที่ฝั่ง web browser ทำหน้าที่ช่วยงานด้านการเรียกขอข้อมูล โดยจะรับการร้องขอที่มีเข้ามาแล้วส่งคำร้องขอไปยัง server

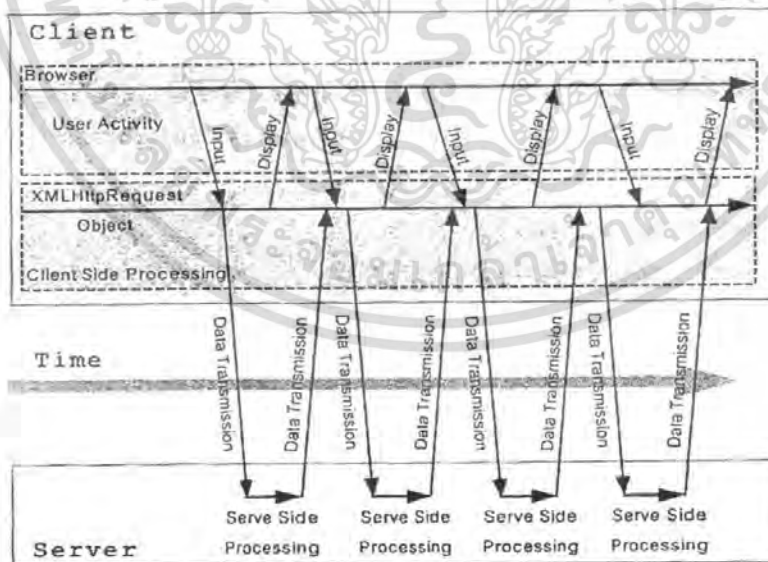
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และมีการเชื่อมต่อกับ server อยู่ตลอดเวลา เมื่อ user เรียกขอข้อมูลก็จะได้รับการตอบสนองก่อน โดยไม่ต้องรอการตอบสนองเบื้องต้นจาก server



รูปที่ 2.10 แสดงลักษณะการสื่อสารที่เป็นแบบ Synchronous ในเว็บแอปพลิเคชันแบบเดิม

จากรูปเป็นการทำงานแบบ Synchronous ในเว็บแอปพลิเคชันแบบเดิม การทำงานเริ่มจาก user เป็นผู้ส่งคำร้องไปยัง server จากนั้น user ต้องรอการตอบสนองจาก server ระหว่างที่รอ user ไม่สามารถที่จะส่งคำร้องระหว่างที่รออีกไม่ได้ ต้องรอจนกว่า server ทำการตอบสนองคำร้องที่ขอมาก่อนหน้าก่อน แต่เมื่อมีการตอบสนองแล้ว การสื่อสาร ก็เป็นอันสิ้นสุด



รูปที่ 2.11 แสดงลักษณะการสื่อสารที่เป็นแบบ Asynchronous ในเว็บแอปพลิเคชันแบบAJAX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปเป็นการทำงานแบบ Asynchronous ของเว็บแอปพลิเคชันแบบ AJAX การทำงานเริ่มจาก user เป็นผู้ส่งคำร้องโดยใช้ javascript ส่งคำร้องไปที่ XMLHttpRequest Object จากนั้น XMLHttpRequest Object จะเลือกว่าจะส่งคำร้องไปยัง server หรือไม่ หากบางกรณีที่สามารถตอบสนองได้ทันที ก็ให้ทำการตอบสนองเลย เช่น การแก้ไขข้อมูลในหน่วยความจำ หรือการแก้ไขข้อมูลที่ XMLHttpRequest มีอยู่แล้ว เป็นต้น หากคำร้องที่ XMLHttpRequest Object นั้นจำเป็นต้องขอข้อมูลจาก server แล้วนั้นก็ทำการส่งคำร้องไปยัง server เช่น ข้อมูลที่จะต้องดึงมาจากฐานข้อมูล หรือข้อมูลที่ต้องอาศัยการประมวลผลจาก server เป็นต้น และเมื่อ server ตอบสนองกลับมาที่ XMLHttpRequest object แล้ว XMLHttpRequest object จะส่งต่อการตอบสนองไปยัง user ด้วย javascript และถึงแม้ XMLHttpRequest object จะได้รับการตอบสนองจาก server แล้วแต่การสื่อสารระหว่างกันนั้นยังคงดำเนินต่อไปเพื่อรอการส่งคำร้องขอข้อมูลอยู่เหมือนเดิม จนกว่า user จะทำการปิดเว็บเพจไป

2.5.5 อาศัยการเขียนโค้ดเป็นหลักและต้องมีระเบียบ

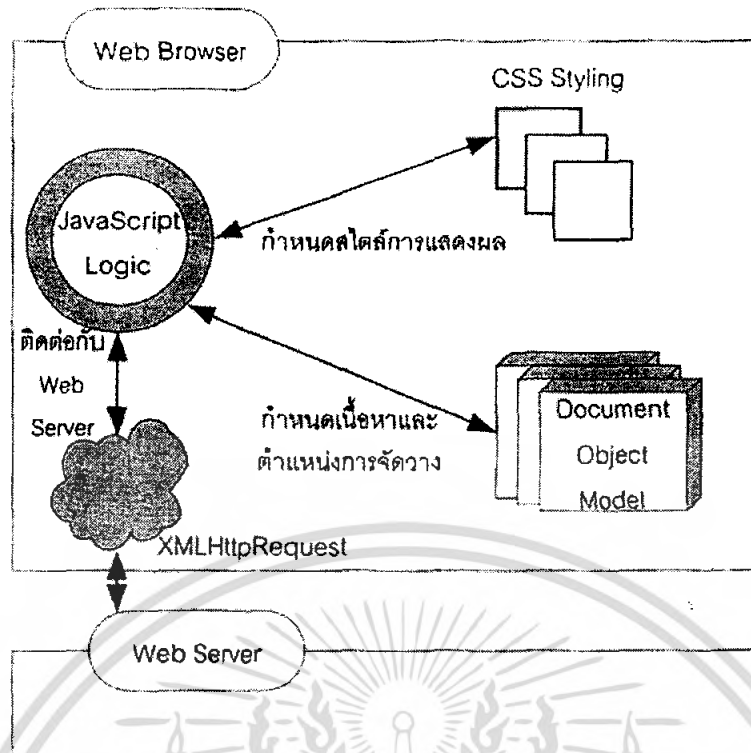
การใช้งานของ javascript ในเว็บแอปพลิเคชันแบบเดิมนั้นจะจำกัดอยู่ในงานบางเวลาหรือบางช่วงเท่านั้นขึ้นอยู่กับกิจกรรมที่ user กระทำ แต่ในเว็บแอปพลิเคชันของ AJAX นั้น javascript จะถูกใช้งานอยู่แทบตลอดเวลาความเร็วในการทำงานจึงเป็นไปอย่างต่อเนื่อง และใช้หน่วยความจำอยู่ตลอดเวลาเช่นกันจะเห็นว่าเว็บแอปพลิเคชันที่ใช้ AJAX นี้จะใช้งาน javascript ค่อนข้างหนัก ดังนั้นการเขียนโค้ดสำหรับ AJAX จึงจำเป็นต้องมีความเข้าใจ javascript มาก เพื่อให้ได้โค้ดที่เป็นระเบียบ สามารถปรับปรุงแก้ไขได้ง่ายภายหลัง จึงจะทำให้เว็บแอปพลิเคชันมีประสิทธิภาพ

โค้ดที่ทำการส่งมาเมื่อ user เข้าใช้แอปพลิเคชันมันจะต้องรันอยู่จนกว่ามันจะถูกปิดลง ไม่มีการหยุดลง ไม่มีการล้าง ไม่มีการหลุดจาก memory นั่นหมายถึงการใช้งานที่ค่อนข้างหนัก เพื่อให้ตรงวัตถุประสงค์ จึงต้องเขียนให้ประสิทธิภาพสูง สามารถแก้ไขได้ มีความเป็นระเบียบเพื่อให้เข้าใจได้ นั่นคือความสำเร็จแก่ server แอปพลิเคชันของ AJAX จะมีการผสมผสานฟังก์ชันเป็นส่วนของโค้ดซึ่งสื่อสารที่มีประสิทธิภาพ กับ server ในระหว่างที่ user ใช้งาน

ตารางที่ 2.4 แสดงส่วนประกอบที่เป็นเทคโนโลยีต่างๆซึ่งถูกนำมาสร้าง AJAX

Javascript	ใช้ในการควบคุมการแสดงผลข้อมูล รวมทั้งการโต้ตอบกับ user กล่าวได้ว่าการทำงานหลักๆที่เกี่ยวข้องกับการตอบสนอง user จะเป็นหน้าที่ของ javascript ทั้งหมด
Cascade Style Sheet (CSS)	กำหนดโครงสร้างหรือลักษณะของการแสดงผลของ webpage ที่มีความคล้ายคลึงกันเช่น สีอักษร ขนาด เป็นต้น ซึ่งการทำเพจที่มีลักษณะซ้ำๆคล้ายคลึงกัน การกำหนดลักษณะที่ซ้ำๆจะทำให้เสียเวลา จึงใช้วิธีการกำหนด style sheet ซึ่งเป็นการกำหนดรูปแบบไว้ก่อน ช่วยให้เว็บเพจมีความเป็นระเบียบ และช่วยการแสดงผลที่มีลักษณะซ้ำๆก็ได้อย่างรวดเร็ว
Documet Object Model (DOM)	เป็นสิ่งที่มาจัดการกับข้อมูลในเอกสาร html สามารถจัดการกับเอกสาร html และสามารถลงลึกไปจัดการกับโค้ดในภาษา html ได้ เป็นแพลตฟอร์มและภาษากลางระหว่างโปรแกรมกับสคริปต์ต่างๆ สามารถเข้าถึงและปรับปรุง content โครงสร้างและสไตล์ของเอกสารต่างๆได้ DOMนั้นทำให้ AJAX สามารถจัดการเอกสาร html ได้อย่างมีประสิทธิภาพ อีกทั้งทำให้โค้ดของเว็บแอปพลิเคชัน AJAX มีความเป็นระเบียบ ง่ายต่อการจัดการ
XMLHttpRequest Object	ควบคุมการแลกเปลี่ยนข้อมูลระหว่าง web browser กับ web server โดยข้อมูลที่แลกเปลี่ยนกันนั้นจะอยู่ในรูปของเอกสาร xml และข้อความสั้นๆ

รูปต่อไปนี้แสดงให้เห็นว่า การทำงานหลักของ AJAX จะไปเน้นหนักที่ฝั่ง browser โดยมีภาระต่างๆในการจัดการแสดงผล ผังอยู่กับ user ในการติดต่อสื่อสารระหว่างกันของ user กับ server จะอาศัยส่วนจัดการการติดต่อคือ XMLHttpRequest Object มาเชื่อมการติดต่อในการร้องขอและรับข้อมูล ส่วนการจัดการกับเนื้อหาและตำแหน่งการวางภายในหน้าเพจจะขึ้นกับ DOM ทำหน้าที่ส่วน CSS จะคอยควบคุมการสไตล์การแสดงผลของ webpage แต่โดยทั้งหมดในหน้าที่หลักจะมี javascript คอยจัดการเพื่อนำข้อมูลออกมาแสดงผลยังหน้าเว็บเพจต่อไป



รูปที่ 2.12 แสดงแผนภาพเพื่อให้เห็นลักษณะการทำงานของเว็บแอปพลิเคชันแบบ AJAX

2.6 ระบบพิกัดตำแหน่งบนพื้นโลก

ในบทนี้จะขอก้าวถึงทฤษฎีที่เกี่ยวข้องที่ใช้กับงานวิจัยและพัฒนาในโครงการนี้ ซึ่งจะกล่าวถึง ระบบการกำหนดพิกัดตำแหน่งบนพื้นโลก ที่มีความสำคัญต่อการใช้หน่วยวัดต่างๆ บนแผนที่ โดยจะกล่าวถึงเพียงพิกัดเดียวคือพิกัดกริดเท่านั้น เนื่องจากพิกัดอื่นๆ ไม่ได้ถูกใช้ในโครงการนี้

2.6.1 พิกัดกริด (Grid Coordinates)

พิกัดกริดเป็นพิกัดตารางสี่เหลี่ยมประกอบด้วยเส้นตรง 2 ชุด คือ เส้นตรงขนานกันในแนวนอน และเส้นตรงขนานกันในแนวตั้ง เส้นตรงทั้ง 2 ชุดนี้จะตัดกันเป็นรูปสี่เหลี่ยมมุมฉาก

เส้นตรงขนานดังกล่าวจะพิมพ์ไว้บนแผ่นแผนที่ ที่เรียกว่า เส้นกริด แต่ละเส้นมีตัวเลขแสดงค่า พิกัดกริดนับจากจุดศูนย์กลางกำเนิด (Origin) ของระบบพิกัดกริดนั้น ขนาดระยะห่างระหว่างเส้น

กริดคู่หนึ่งๆ ย่อมคงที่เสมอและจะมีระยะจริงตรงกับภูมิประเทศเท่าใดนั้นขึ้นอยู่กับขนาด มาตราส่วนของแผนที่ที่ผลิตขึ้นใช้ จุดศูนย์กลางกำเนิดของระบบพิกัดกริดจะใช้ศูนย์

กำเนิดจริง (True Origin) หรือศูนย์กำเนิดสมมติ (False Origin) ย่อมขึ้นอยู่กับความเหมาะสมของแต่ละระบบ

ระบบพิกัดกริดมีอยู่หลายชนิดด้วยกัน ในที่นี้จะกล่าวถึงเฉพาะพิกัดกริดที่นิยมใช้ในกิจการทหาร (Military Grid) ยุคปัจจุบันนี้เท่านั้น

2.6.2 พิกัดกริดยูนิเวอร์ซัลทรานสเวอร์สมเมอร์เคเตอร์ (Universal Transverse Mercator)

แผนที่มาตราส่วนใหญ่และมาตราส่วนปานกลางที่ใช้ในกิจการทหารส่วนมาก นอกจากจะมี ระบบพิกัดภูมิศาสตร์ (Geographic Coordinates) แล้วยังมีระบบกริดที่ใช้ช่วยในการกำหนดตำแหน่ง และใช้อ้างอิงในการบอกตำแหน่งอีกด้วย การบอกตำแหน่งโดยอาศัยระบบกริดมีส่วนดีและสะดวกกว่า ใช้ระบบพิกัดภูมิศาสตร์ เพราะตารางกริดมีขนาดเท่ากันและรูปร่างเหมือนกันทุกตาราง และพิกัดกริด ให้ค่าเป็นระยะทางซึ่งง่ายแก่การกำหนดดีกว่าค่ามุม ระบบ UTM กริด มีลักษณะโดยย่อดังนี้

- ใช้ร่วมกับโปรเจกชันแบบ TRANSVERSE MERCATOR (Gauss Kruger) โดยแบ่งพิภพ ออกเป็นโซนละ 6 องศา ตามลองจิจูด โซนที่ 1 อยู่ระหว่างลองจิจูด 180 องศาตะวันตก กับลองจิจูด 174 องศาตะวันตก นับต่อเนื่องไปทางตะวันออกครบพิภพรวม 60 โซน ซึ่งโซนที่ 60 จะอยู่ระหว่างลองจิจูด 174 องศาตะวันออกกับลองจิจูด 180 องศาตะวันออก
- สำหรับประเทศไทยใช้รัศมีของพิภพตามค่า Equatorial semi-axis $a = 6,377,276.34518$ เมตร ของ EVEREST SPHEROID
- ระบบพิกัด UTM กริด ครอบคลุมบริเวณตั้งแต่ละติจูด 80 องศาใต้ ถึงละติจูด 84 องศาเหนือ
- หน่วยที่ใช้ในการวัดเป็นเมตร โดยมีจุดศูนย์กำเนิดอยู่ที่จุดตัดกันระหว่างเส้นศูนย์สูตรกับเส้น เมริเดียนย่านกลาง (Central Meridian) ของแต่ละ โซน
- ค่าพิกัดมี 2 ค่า คือ พิกัดทางเหนือ (Northing) ใช้อักษรย่อว่า N และ พิกัดทางตะวันออก (Easting) ใช้อักษรย่อว่า E
- ค่าพิกัดของจุดศูนย์กำเนิดของแต่ละ โซนเป็นค่าพิกัดสมมติเพื่อหลีกเลี่ยงค่าพิกัดที่เป็นลบ โดยกำหนดให้

พิกัดของจุดศูนย์กำเนิดของแต่ละ โซนทางซีกโลกเหนือ

False northing = 0 เมตร

False easting = 500, 000 เมตร

พิกัดของจุดศูนย์กำเนิดของแต่ละ โซนทางซีกโลกใต้

False northing = 10,000,000 เมตร

False easting = 500, 000 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แต่ละโซนมีขนาดพื้นที่เท่ากัน แผนที่ที่คลุมบริเวณของแต่ละโซนมีขนาดเท่ากัน สำหรับ แผนที่มาตราส่วนใหญ่จะมีส่วนเหลือมลักันออกไปสองข้างรอยต่อโซนเข้าละ 30 ลิปดา หรือ 25 ไมล์ เพื่อประ โยชน์ในงานสำรวจด้านวิศวกรรมและการตรวจการ ยิงของปืนใหญ่
- เส้นกริดในทางตั้งจะขนานกับเมริเดียนย่านกลางของแต่ละโซน ส่วนเส้นกริดทาง แนวนอน จะขนานกับเส้นศูนย์สูตร
- การกำหนดโซนของกริด (Grid Zone Designation)
- ระหว่างละติจูด 80 องศาใต้กับละติจูด 84 องศาเหนือแบ่งออกเป็น 20 ส่วนๆ ละ 8 องศา เฉพาะส่วน บนสุดเท่านั้นที่มีขนาด 12 องศา แต่ละส่วนใช้อักษรกำกับ เริ่มจากอักษร C ที่เป็นส่วนใต้สุด (ระหว่าง ละติจูด 80-72 องศาใต้) ขึ้นไปตามลำดับถึงอักษร X ยกเว้น อักษร I กับ O
- ระหว่างลองจิจูด 180 องศาตะวันตกเวียนไปทางตะวันออกถึงลองจิจูด 180 องศา ตะวันออก แบ่ง ออกเป็น 60 ส่วนๆ ละ 6 องศา แต่ละส่วนใช้ตัวเลขกำกับ เริ่มส่วนที่ 1 ระหว่างลองจิจูด 180 องศา ตะวันตกกับ 174 องศาตะวันตก นับไปทางตะวันออก จนถึงส่วนที่ 60 ซึ่งเป็นส่วนสุดท้ายอยู่ระหว่าง ลองจิจูด 174 องศาตะวันออกกับ 180 องศาตะวันออก (ลองจิจูด 180 องศาตะวันตกกับลองจิจูด 180 องศาตะวันออกเป็นเส้น เดียวกัน)
- การกำหนดรูปสี่เหลี่ยมจัตุรัสแสนเมตร(100,000 meter square identification) - เริ่มจาก ลองจิจูด 180 องศาตะวันตกนับไปตามเส้นศูนย์สูตรไปทางตะวันออกทุกๆ ระยะ 100,000 เมตร ให้ อักษรกำกับ เริ่มจากอักษร A ถึง Z ยกเว้นอักษร I กับ O จะมีชุด ตัวอักษรซ้ำกันทุกๆ 8 องศา หรือ 3 โซน
- ตามแนวเหนือ-ใต้ ซีกโลกภาคเหนือเฉพาะโซนหมายเลขที่ เริ่มจากเส้นศูนย์สูตรขึ้นไป ทุกๆ ระยะ 100,000 เมตร ให้ตัวอักษรกำกับเริ่มจากอักษร A ขึ้นไป ตามลำดับถึงอักษร V ยกเว้น I กับ O ส่วนโซน หมายเลขคู่เริ่มนับจากจุดที่อยู่ใต้เส้นศูนย์สูตรลงไป 500,000 เมตร ระยะ 100,000 เมตรแรกใช้อักษร A กำกับและให้อักษรกำกับทุกๆ ระยะ 100,000 เมตรขึ้นมาตามลำดับจนถึงอักษร V ยกเว้นอักษร I กับ O จะมีชุดอักษรซ้ำกัน ทุกๆ ระยะ 2,000,000 เมตร
- การบอกค่าพิกัดกริดของระบบ UTM กริดที่สมบูรณ์จะต้องบอกตามลำดับดังต่อไปนี้
 - บอกให้ทราบชื่อโซนของกริด (Grid Zone Designation) เช่น 3P
 - บอกให้ทราบชื่อรูปสี่เหลี่ยมจัตุรัสแสนเมตร เช่น (100,000 meter square identification) เช่น MN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บอกค่าพิกัดกริดของจุดที่พิจารณา ตามขนาดความละเอียดที่ต้องการ เช่น 24,
2142, 218427, 21834279.....

- การบอกค่าพิกัด UTM กริดของจุดใดๆ ขณะปฏิบัติการในโซนเดียวกัน เพื่อความรวดเร็ว นิยมบอกเป็นค่าพิกัด E และ N เฉพาะหลักหมื่น หลักพัน (ซึ่งในแผนที่ 1:50,000 ชุด L 708 และ L 7017 จะพิมพ์ไว้เป็นตัวเลขใหญ่กำกับเส้นกริด) และหลักร้อย ซึ่งประมาณได้ในแผนที่ เช่น 218427218 คือค่า E (8 คือ หลักร้อยที่ประมาณได้) 427 คือค่า N (7 คือ หลักร้อยที่ประมาณได้)
- การบอกค่าพิกัดของระบบ UTM กริดของจุดใดๆ ในโซนเดียวกันซึ่งต้องการค่า E และ N ที่สมบูรณ์ จะต้องเขียนให้ครบตามค่าพิกัดที่นับจากศูนย์สมมติ เช่น พิกัดของจุด ก. E = 748,600 เมตร N = 1,546,415 เมตร พิกัดของจุด ข. E = 801,502 เมตร N = 1,643,712 เมตร
- ปกติเส้นกริดของระบบ UTM จะมีค่าประจำอยู่ทุกเส้นระยะระหว่างเส้นสามารถอ่านประมาณได้ใกล้เคียงถึง 1/10 ของระยะระหว่างระหว่างเส้นกริดนั้น ถ้าต้องการอ่านให้ละเอียดถูกต้องกว่านั้นก็จำเป็นต้องใช้ Coordinate Scale ช่วยในการอ่านในรูปที่แสดงไว้จะเห็นตัวอย่างของ Coordinate Scale ที่ใช้อ่านพิกัดกริดของแผนที่มาตราส่วนต่างๆ ซึ่งผู้ใช้แผนที่สามารถสร้างขึ้นใช้เองได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

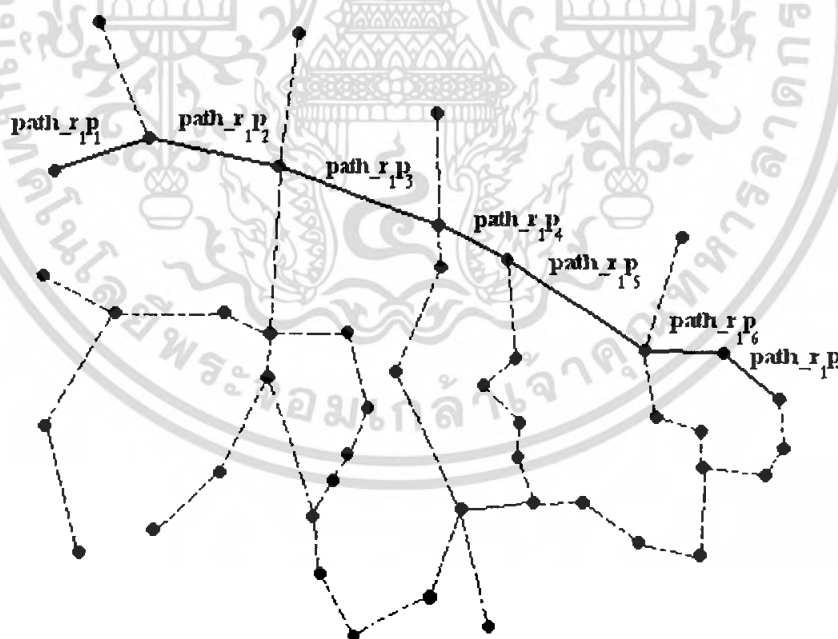
การนำทฤษฎีปัญหาประติมากรรมมาประยุกต์ใช้งาน

3.1 การแทนข้อมูลแผนที่

ในการค้นหาเส้นทางการเดินทางจากจุด 2 จุดบนแผนที่นั้น เราจำเป็นต้องแทนข้อมูลของเส้นทางบนแผนที่ให้อยู่ในรูปกราฟ จากนั้นการวางแผนหาเส้นทางการเดินทางที่สั้นที่สุดนั้นก็จะเป็นเพียงการค้นหาเส้นทางที่สั้นที่สุดที่เชื่อมระหว่างจุดสถานที่ 2 จุดบนกราฟนั่นเอง

ในการแทนข้อมูลของเส้นทางบนแผนที่ให้อยู่ในรูปกราฟนั้น ทำได้โดย กำหนดตำแหน่งต่างๆ ด้วยระบบพิกัดตำแหน่ง (X,Y) เมื่อกำหนดพิกัดแล้วจะแทนความหมายของพิกัดนั้นๆด้วยสัญลักษณ์ การแทนข้อมูลของถนนมีดังนี้ พิกัดตำแหน่ง (X,Y) ให้มีชื่อ pt_id สำหรับ path[pt_id1, pt_id2] ให้มีชื่อ path_id ที่เชื่อมระหว่างพิกัดชื่อ pt_id1 และ pt_id2 ตามลำดับ โดยมี length คือระยะระหว่างพิกัด และ road[path_id1, path_id2, path_id3, ...] ให้มีชื่อ road_id ซึ่งได้จากการเชื่อมหลายๆ path ตั้งแต่ path_id1 ถึง path_id3 และอื่นๆตามลำดับเข้าด้วยกัน

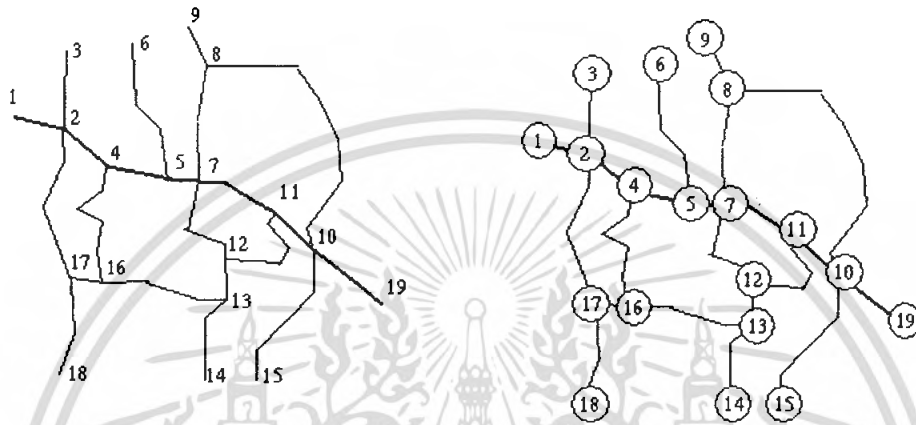
จากด้านล่างเป็นตัวอย่างเส้นทางซึ่งเป็นกลุ่มของ path บนถนนเส้นหนึ่งในโครงข่ายถนน คือ [path_{r1p1}, path_{r1p2}, path_{r1p3}, path_{r1p4}, path_{r1p5}, path_{r1p6}, path_{r1p7}] โดยที่ r_i คือชื่อถนน และ p_i คือชื่อ path ต่างๆของถนน



รูปที่ 3.1 ความสัมพันธ์ระหว่าง path ของถนนและเส้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากลักษณะโครงข่ายของถนน เราจะทำการกำหนดให้บริเวณจุดต่อหรือจุดตัดของถนน อันได้แก่ สี่แยก สามแยก ปลายของถนน เป็นพิกัดตำแหน่ง และให้พิกัดดังกล่าวเป็นโหนด โดยแต่ละโหนดจะเชื่อมต่อกับด้วยเอจ (Edge) ซึ่งอาจจะเป็น path หรือถนน ตามทฤษฎีของกราฟ การเชื่อมต่อของถนนเป็นกราฟแสดงได้ดังรูปด้านล่าง ในรูปเป็นตัวอย่างของถนนสายหนึ่ง (เส้นทึบ) ซึ่งประกอบไปด้วย path ที่ผ่านโหนดต่างๆดังนี้ [1, 2, 4, 5, 7, 11, 10, 19] จากลักษณะกราฟของแผนที่ก็จะกลายเป็นข้อมูลสำคัญที่ใช้ในการค้นหาเส้นทาง



รูปที่ 3.2 โครงสร้างของถนนและการกำหนดโหนดตามทฤษฎีกราฟ

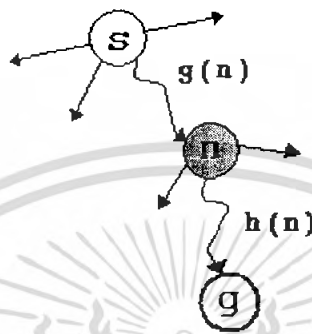
3.2 การค้นหาเส้นทาง

การค้นหาเส้นทางบนแผนที่ เป็นการค้นหาโหนดที่เป็นจุดเชื่อมของเส้นถนน โดยที่เมื่อนำโหนดทั้งหมดกับเส้นเชื่อมโหนดซึ่งก็คือ path มาต่อกันก็จะได้เป็นเส้นทางที่จะนำไปสู่จุดหมาย จากที่ได้อธิบายถึงการเชื่อมต่อของถนนทำให้เกิดเป็นกราฟในหัวข้อที่ผ่านมาจะเป็นผลให้ทุกๆ โหนดมีการเชื่อมต่อกัน สามารถเดินทางจากจุดหนึ่งไปยังอีกจุดหนึ่งได้ โดยทั่วไปแล้วสิ่งที่เราสนใจก็คือการเดินทางไปในกราฟนั้น เพื่อให้ได้เส้นทางใดที่ดีที่สุด ได้แก่ มีระยะทางสั้นที่สุด เป็นต้น

การค้นหาเส้นทางที่ดีที่สุด ได้ใช้วิธีการค้นหาแบบฮิวริสติก (Heuristic Search) ซึ่งมีการนำเอาฟังก์ชันฮิวริสติก (heuristic function) มาช่วยตัดสินใจเลือกเส้นทางที่จะนำไปสู่จุดหมายได้ โดยประหยัดทั้งเวลาและหน่วยความจำ ซึ่งใช้วิธีที่เรียกว่า เอสตาร์ (A* Search) ถือว่าเป็นวิธีที่เหมาะสมที่สุดสำหรับการค้นหาเส้นทาง เพราะสามารถรับประกันว่าให้คำตอบที่ดีที่สุด

3.3 การค้นหาเส้นทางที่ดีที่สุดโดยใช้วิธีเอสตาร์

การค้นหาแบบเอสตาร์ใช้ฟังก์ชันฮิวริสติก(Heuristic function) $f(n) = g(n) + h(n)$ ในการค้นหาเส้นทางให้ไปสู่จุดหมายอย่างรวดเร็ว โดย $g(n)$ หมายถึงค่าใช้จ่าย (Cost) จริงที่ใช้ในการเดินทางจากโหนดเริ่มต้น s จนถึงโหนดปัจจุบัน n และค่า $h(n)$ หมายถึงค่าใช้จ่ายประมาณที่ใช้ในการเดินทางจากโหนดปัจจุบัน n ไปถึงโหนดจุดหมาย g ดังรูป



รูปที่ 3.3 ส่วนประกอบของ heuristic function $f(n)$

อัลกอริทึมในการค้นหาแบบเอสตาร์จะใช้ $f(n)$ ในการตัดสินใจเลือกโหนดของกราฟ เริ่มจากโหนดเริ่มต้น s เพื่อค้นหาโหนดลูกแต่ละชั้น ซึ่งแต่ละชั้นจะทำให้เกิดเส้นทางที่ยังค้นหาไม่จบเส้นทางเหล่านี้มีโหนดปลาย ซึ่งแต่ละอันถือว่าเป็นโหนด n ของเส้นทางนั้นๆ ถ้าเส้นทางใดในบรรดาเส้นทางเหล่านี้ให้ค่า $f(n)$ ของโหนด n ของตนเป็นค่าที่น้อยที่สุดในกลุ่มโหนด n อื่นๆ อัลกอริทึมจะเลือกขยายขอบเขตการค้นหาต่อในเส้นทางนั้น โดยขยายการค้นหาที่โหนด n ของเส้นทางที่เลือกลงไปยังโหนดลูกของมัน ถ้ามีโหนดลูกตัวใดปรากฏในเส้นทางเส้นทางที่กำลังค้นหาค้างอยู่ก็จะถูกตัดทิ้งไป การค้นหาโดยเอสตาร์จะทำทีละชั้นเช่นนี้ซ้ำแล้วซ้ำอีกไปเรื่อยๆ จนกระทั่งพบโหนดจุดหมาย g เส้นทางที่เกิดขึ้นที่เชื่อมจาก s ไป g จะถือว่าเป็นเส้นทางที่ดีที่สุดที่เลือกโดยวิธีเอสตาร์ วิธีเอสตาร์จะสามารถเลือกเส้นทางที่ดีที่สุด ได้ก็ต่อเมื่อ ฟังก์ชัน $g(n)$ ซึ่งเป็นการคิดค่าใช้จ่ายจากโหนดเริ่มต้น s ถึงโหนดปัจจุบัน n จะต้องเป็นฟังก์ชันที่คิดค่าออกมาแล้วเป็นค่าใช้จ่ายจริงของเส้นทางนั้นๆ ที่สะสมมาตั้งแต่โหนดเริ่มต้น s จนถึงโหนดปัจจุบัน n ส่วน $h(n)$ ซึ่งเป็นการประมาณการค่าใช้จ่ายจากโหนดปลาย n ถึงโหนดจุดหมาย g จะต้องเป็นการประมาณค่าที่ไม่เกินจากค่าใช้จ่ายจริงของเส้นทางที่ค่าใช้จ่ายน้อยที่สุดที่เชื่อมจาก n ถึง g

3.4 การนำอัลกอริทึมการค้นหาแบบเอสตาร์มาประยุกต์ใช้งานกับโครงงาน

3.4.1 การค้นหาเส้นทางที่มีระยะทางสั้นที่สุดในการเดินทาง

สำหรับการค้นหาแบบนี้จะให้ $g(n)$ เป็นการคำนวณค่าระยะทางจริงที่ใช้ในการเดินทางจากโหนดเริ่มต้นจนถึงโหนดปัจจุบัน และให้ $h(n)$ เป็นการคำนวณระยะทางจากโหนดปัจจุบัน ของเส้นทางที่อยู่ระหว่างการค้นหาให้เป็นพิคัด (X_1, Y_1) ไปยังจุดหมาย ซึ่งกำหนดเป็นพิคัด (X_2, Y_2)

$$g(n) = \text{ระยะทางจริงที่ใช้ในการเดินทางจากโหนดเริ่มต้นจนถึงโหนดปัจจุบัน} \quad (1.1)$$

$$h(n) = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \quad (1.2)$$

3.4.2 การค้นหาเส้นทางที่มีจำนวนการต่อสายร่อน้อยที่สุดในการเดินทาง

สำหรับการค้นหาเส้นทางในลักษณะนี้จะให้ $g(n)$ เป็นค่าจำนวนสายรถที่ใช้จริงจากโหนดเริ่มต้นจนถึงโหนดปัจจุบัน ส่วน $h(n)$ นั้นเนื่องจากโหนดปัจจุบันของเส้นทางที่กำลังทำการค้นหาอยู่นั้นไม่มีข่าวสารใดๆที่จะบอกได้ว่าการเดินทางหลังจากนี้จะต้องมีการเปลี่ยนสายรถอีกกี่สายเราจึงกำหนดให้ $h(n)$ เป็น 0

$$g(n) = \text{จำนวนสายรถที่ใช้จริงจากโหนดเริ่มต้นจนถึงโหนดปัจจุบัน} \quad (1.3)$$

$$h(n) = 0 \quad (1.4)$$

3.4.3 การค้นหาเส้นทางที่มีจำนวนการต่อสายร่อน้อยที่สุดและมีระยะทางสั้นที่สุดในการเดินทาง

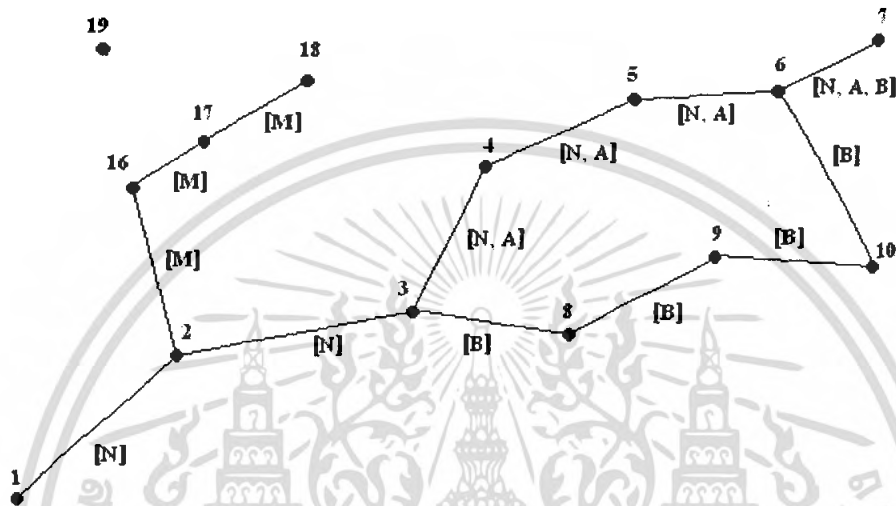
สำหรับการค้นหาเส้นทางในลักษณะนี้จะให้ $g(n)$ เป็นการคำนวณค่าระยะทางจริงที่ใช้ในการเดินทางจากโหนดเริ่มต้นจนถึงโหนดปัจจุบันรวมทั้งจำนวนสายรถที่ใช้จริงจากโหนดเริ่มต้นจนถึงโหนดปัจจุบัน (ซึ่งเราจะให้น้ำหนักกับสายรถมากกว่าระยะทาง) ส่วน $h(n)$ เป็นการคำนวณระยะทางจากโหนดปัจจุบันไปยังโหนดจุดหมาย

$$g(n) = \text{ระยะทาง} + \text{จำนวนสายรถที่ใช้} \quad (1.5)$$

$$h(n) = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \quad (1.6)$$

3.4.4 การค้นหาเส้นทางโดยระบุชนิดของยานพาหนะที่ใช้ในการเดินทาง

เนื่องจากเส้นทางที่จะใช้เดินทางบนกราฟนั้น เกิดจากการที่มีพาหนะชนิดใดชนิดหนึ่งเดินทางผ่านเส้นทางนั้น นั่นคือหากเส้นทางใดไม่มียานพาหนะเดินทางผ่าน เส้นทางนั้นก็จะไม่สามารถใช้เดินทางได้นั่นเอง ส่วนยานพาหนะที่มีอยู่ในระบบมี 4 ชนิดด้วยกัน คือ รถประจำทางแบบธรรมดา, รถประจำทางแบบปรับอากาศ, รถไฟฟ้าบีทีเอส และ รถไฟฟ้าเอ็มอาร์ที ซึ่งตัวอย่างเส้นทางของยานพาหนะเป็นดังรูป



รูปที่ 3.4 ความสัมพันธ์ระหว่างประเภทยานพาหนะกับเส้นทางบนกราฟ

จากรูปข้างต้น เป็นกราฟที่เกิดจากการเดินทางของยานพาหนะต่างๆ ซึ่งแทนด้วยอักษรต่างๆคือ อักษร N แทน รถประจำทางแบบธรรมดา อักษร A แทนรถประจำทางแบบปรับอากาศ อักษร B แทนรถไฟฟ้าแบบ BTS และอักษร M แทนรถไฟฟ้าแบบ MRT ส่วนเส้นที่เชื่อมโยงระหว่างโหนดก็เกิดจากการที่มียานพาหนะเดินทางจากโหนดหนึ่งไปยังอีกโหนดหนึ่ง ตัวอย่างเช่น รถประจำทางแบบธรรมดา (N) มีเส้นทางการเดินทางดังนี้ คือ เริ่มต้นจากโหนดหมายเลข 1 ไปต่อยังโหนดหมายเลข 2 ไปต่อยังโหนดหมายเลข 3 จนถึงโหนดหมายเลข 7 ทำให้เกิดเส้นเชื่อม โยงระหว่างโหนดต่างๆขึ้นนั่นเอง

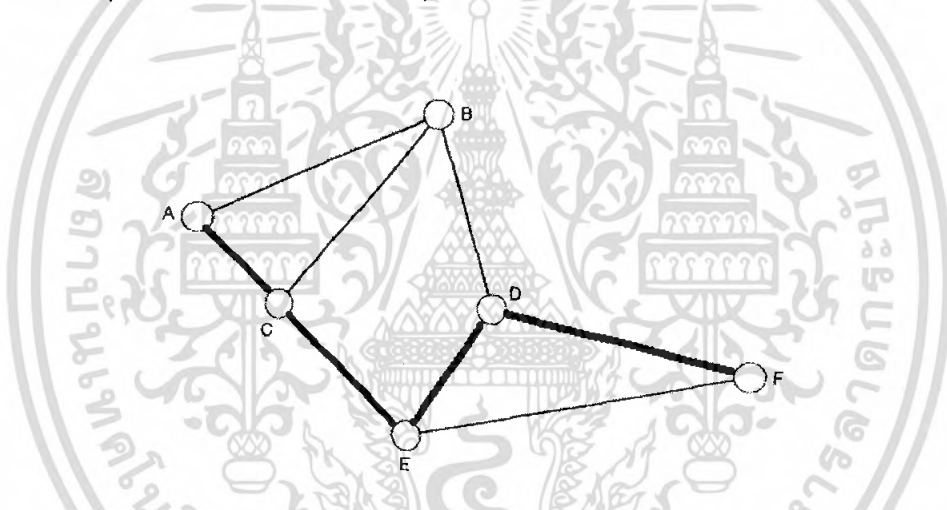
มีความเป็นไปได้ที่ยานพาหนะที่เดินทางระหว่างโหนดใดๆก็อาจจะเดินทางซ้ำกับยานพาหนะคันอื่นได้เช่น เส้นเชื่อมโยงระหว่างโหนดหมายเลข 4 กับโหนดหมายเลข 5 เกิดจากที่มียานพาหนะ 2 ชนิด (N และ A) ใช้เส้นทางร่วมกันในการเดินทาง

ส่วนการค้นหาเส้นทางโดยระบุชนิดของยานพาหนะที่ใช้ในการเดินทางนั้น คือ การที่เราจะเดินทางจากโหนดหนึ่งไปยังโหนดหนึ่งโดยระบุให้เดินทางไปตามเส้นทางที่มี

ยานพาหนะตามที่ระบุเท่านั้น ตัวอย่างเช่น หากเราต้องการเดินทางจากโหนดหมายเลข 1 ไปยังโหนดหมายเลข 7 ด้วยยานพาหนะรถประจำทางแบบธรรมดาหรือรถประจำทางแบบปรับอากาศ (N หรือ A) เราก็จะสามารถเดินทางไปได้ด้วยเส้นทาง 1, 2, 3, 4, 5, 6, 7 เพราะเป็นเส้นทางที่มียานพาหนะ N หรือ A ตามที่ปรากฏอยู่

3.4.5 การค้นหาเส้นทางแบบมีจุดผ่านหลายจุด

จากวิธีการค้นหาแบบ A^* ที่ได้กล่าวมาแล้วจะเห็นได้ว่า เราจะเสร็จสิ้นการค้นหาที่ต่อเมื่อค้นพบโหนดเป้าหมายหมาย แต่ในการค้นหาแบบมีจุดผ่านหลายจุดนั้น ไม่เพียงแต่โหนดสุดท้ายที่ตรวจสอบจะต้องเป็นโหนดเป้าหมายแล้ว ในเส้นทางที่เราตรวจสอบอยู่จะต้องเป็นเส้นทางที่ผ่านจุดที่ต้องการด้วย ซึ่งเราสามารถอธิบายด้วยทฤษฎีเซตทางคณิตศาสตร์ได้ คือ เราจะเสร็จสิ้นการค้นหาที่ต่อเมื่อ เซตของจุดหมาย (เซตของจุดที่ผ่านรวมกับจุดหมาย) เป็นสับเซต (Subset) ของเซตของเส้นทางที่ตรวจสอบอยู่ในขณะนั้น และโหนดสุดท้ายที่ตรวจสอบเป็นโหนดจุดหมาย ดังตัวอย่าง



รูปที่ 3.5 กราฟสำหรับการค้นหาแบบมีจุดผ่านหลายจุด

จากรูปข้างต้น เราต้องการเดินทางจากโหนด A ไปยังโหนด F โดยกำหนดให้ต้องผ่านโหนด E และ D ด้วย เราจะแทนข้อมูลในรูปของเซตได้คือให้ S แทนเซตของโหนดเริ่มต้น ให้ G คือแทนเซตของโหนดเป้าหมาย และ P แทนเซตของโหนดเส้นทางที่ตรวจสอบ ดังนี้ $S = \{A\}$, $G = \{E, D, F\}$ และ $P = \{A, C, E, D, F\}$ จะเห็นได้ว่าเซต G เป็นสับเซตของเซต P และโหนดสุดท้ายที่ตรวจสอบเป็นโหนดเป้าหมาย

วิธีการค้นหาแบบต่างๆ ที่ได้อธิบายไปนี้ นำมาใช้ในการค้นหาเส้นทางต่อสายรถเมล์ตามเงื่อนไขต่างๆ ได้แก่ ระยะทางสั้นที่สุด เปลี่ยนสายรถน้อยที่สุดในการเดินทาง เป็นต้น

บทที่ 4

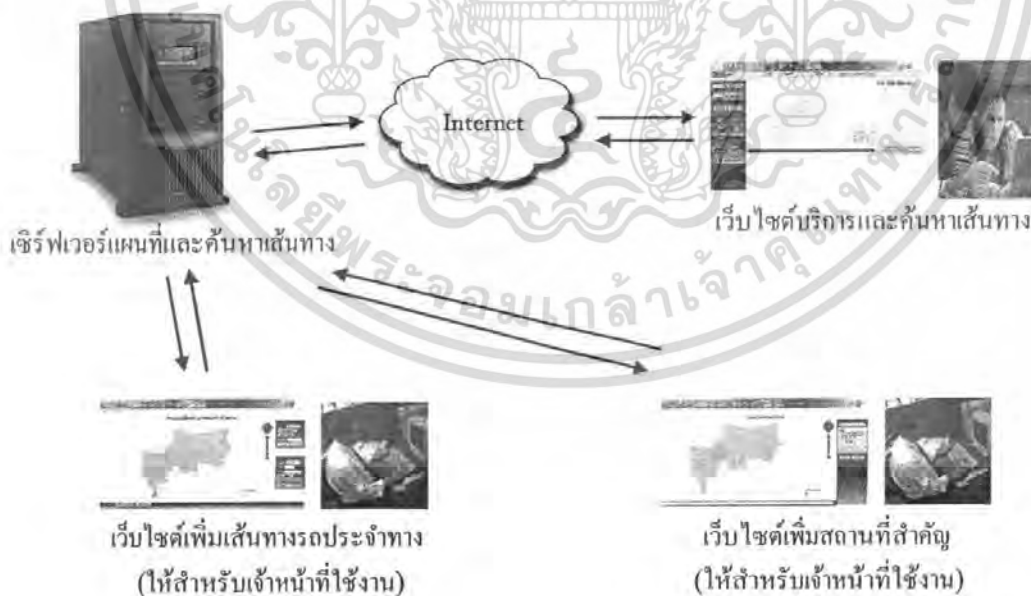
การออกแบบและพัฒนา

ในบทนี้จะขอกล่าวถึงการออกแบบโครงสร้างของระบบค้นหาเส้นทางรถประจำทาง
อัจฉริยะ ตลอดจนการมีส่วนประกอบต่างของระบบอย่างละเอียด

4.1 การออกแบบโครงสร้างระบบรถประจำทางอัจฉริยะ

ระบบจัดสร้างรถแท็กซี่ให้บริการผู้โดยสารที่พัฒนาขึ้นประกอบด้วย 5 ส่วน (ดังรูปข้างล่าง)
คือ

- โคลเอนต์บน Webpage เป็นส่วนของการค้นหาเส้นทาง และให้บริการข้อมูลแก่ผู้ใช้
- โคลเอนต์บน Webpage ส่วนการเพิ่มสายรถประจำทาง เป็นส่วนของการเพิ่มเส้นทางรถ
ประจำทาง เส้นทางรถไฟฟ้า BTS รถไฟฟ้าใต้ดิน MRT
- โคลเอนต์บน Webpage ส่วนการเพิ่มสถานที่สำคัญ เป็นส่วนของการเพิ่มสถานที่สำคัญ
ต่างๆ โดยแยกตามประเภท
- เซิร์ฟเวอร์แผนที่และ AI เป็นส่วนที่ให้บริการค้นหาเส้นทางและส่งผลลัพธ์แก่โคลเอนต์
การทำงานของระบบค้นหาเส้นทางรถประจำทางอัจฉริยะมีลำดับขั้นตอนดังต่อไปนี้



รูปที่ 4.1 แสดงโครงสร้างของระบบค้นหาเส้นทางรถประจำทางอัจฉริยะ

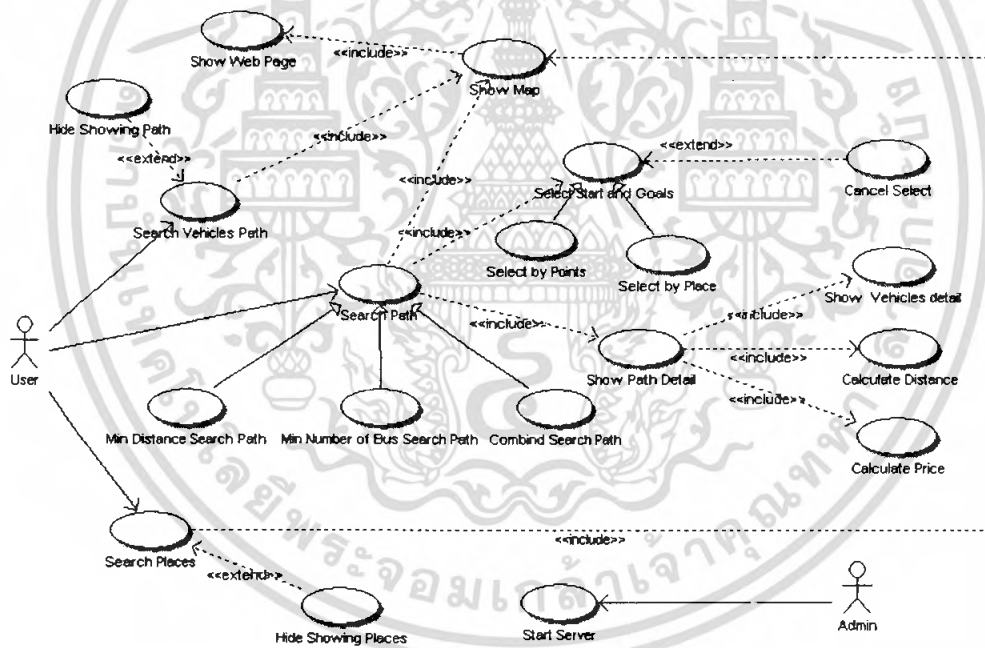
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งมีลำดับขั้นตอนการทำงานดังนี้

- ผู้ใช้เรียกใช้โปรแกรมผ่านเว็บเบราว์เซอร์ เว็บเบราว์เซอร์จะทำการโหลดหน้าเว็บเพจ สคริปต์ภาษาจาวาทำการเรียกแผนที่จากเซิร์ฟเวอร์
- ผู้ใช้ทำการกำหนดจุดต้นทางและจุดหมายปลายทางและทำการส่งข้อมูลไปยังเซิร์ฟเวอร์
- เมื่อเซิร์ฟเวอร์ได้รับข้อมูลมา จะทำการประมวลผลเพื่อหาเส้นทางที่สั้นที่สุดและจะทำการส่งข้อมูลกลับมายังฝั่งไคลเอนต์
- เมื่อฝั่งไคลเอนต์ได้รับข้อมูลมา จะทำการนำข้อมูลดังกล่าวมาแสดงผลบนแผนที่ ส่วนข้อมูลเลขรถประจำทาง ระยะทาง ค่าโดยสาร จะแสดงที่ส่วนล่างของแผนที่

ระบบจะทำงานดังกล่าวทุกครั้งเมื่อมีการค้นหาเส้นทางของฝั่งไคลเอนต์

4.2 Use Case Diagram



รูปที่ 4.2 ยูสเคสไดอะแกรม

● ผู้ที่กระทำกับระบบ (Actor)

User คือ ผู้ที่เข้ามาใช้บริการของระบบ

Admin คือ ผู้ที่เริ่มต้นการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **ยูสเคส (Use Case)**

Search Path คือ การค้นหาเส้นทาง

Min Distance Search Path คือ การค้นหาเส้นทางที่มีระยะทางสั้นที่สุด

Min Number of Bus Search Path คือ การค้นหาเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุด

Show Map คือ การแสดงแผนที่

Show Web Page คือ หน้าเว็บเพจของระบบ

Select Start and Goals คือ การเลือกต้นทางและปลายทางที่ต้องการค้นหาเส้นทาง

Select by Points คือ การเลือกต้นทางและปลายทางจากจุดบนถนน

Select by Places คือ การเลือกต้นทางและปลายทางจากสถานที่

Cancel Select คือ การยกเลิกต้นทางและปลายทางที่เลือกไว้

Show Path Detail คือ การแสดงรายละเอียดของเส้นทาง

Show Vehicle Detail คือ การแสดงรายละเอียดของยานพาหนะ

Calculate Distance คือ การคำนวณระยะทางที่ใช้ในการเดินทาง

Calculate Price คือ การคำนวณจำนวนเงินรวมที่ใช้ในการเดินทาง

Search Places คือ การค้นหาตำแหน่งและรายละเอียดของสถานที่ต่างๆ

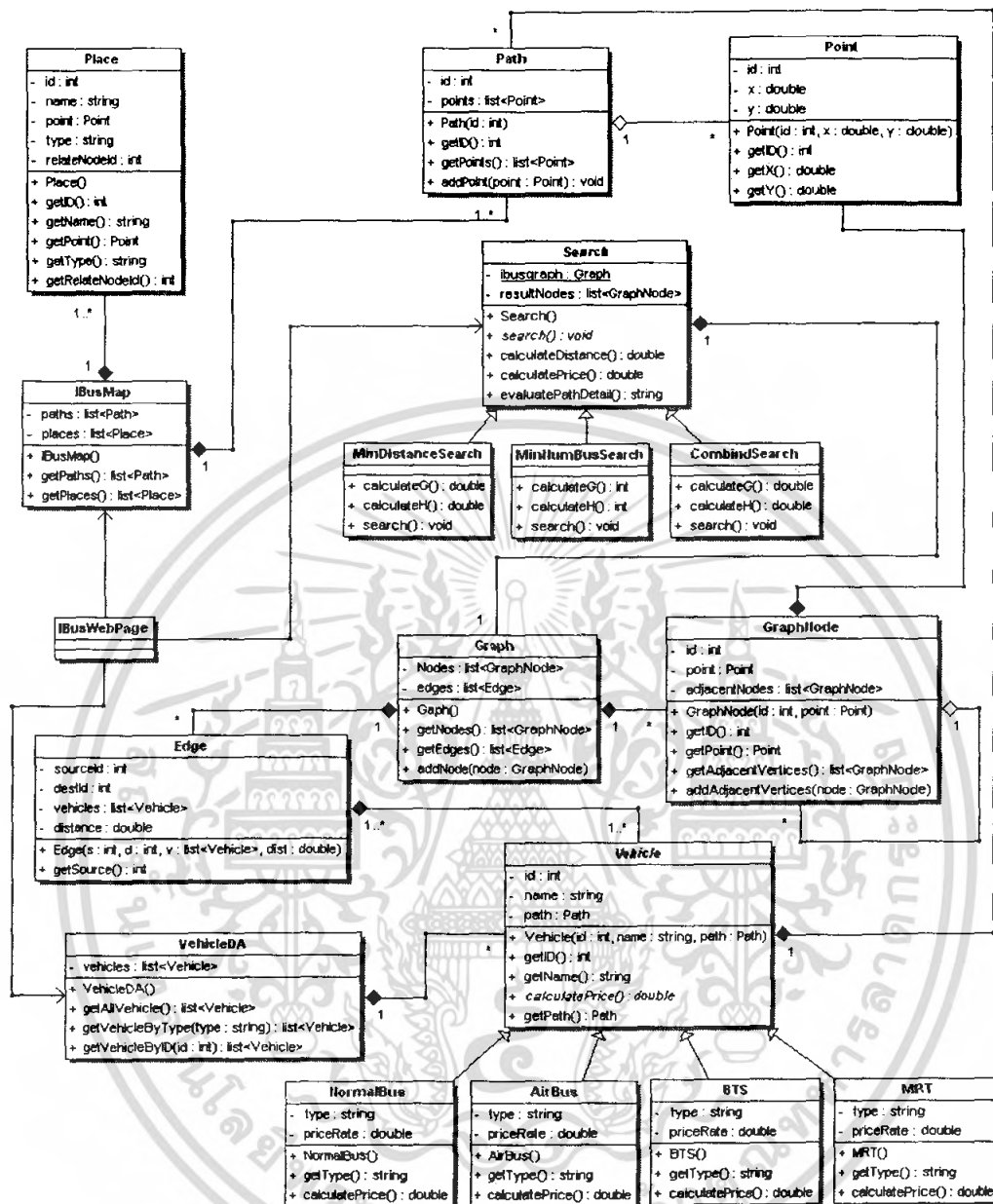
Hide Showing Places คือ การซ่อนสถานที่ที่แสดงอยู่

Search Vehicle Path คือ การค้นหาเส้นทางการเดินทางของสายรถประจำทาง

Hide Showing Places คือ การซ่อนเส้นทางสายรถที่แสดงอยู่

Start Server คือ การเริ่มต้นการทำงานของระบบ

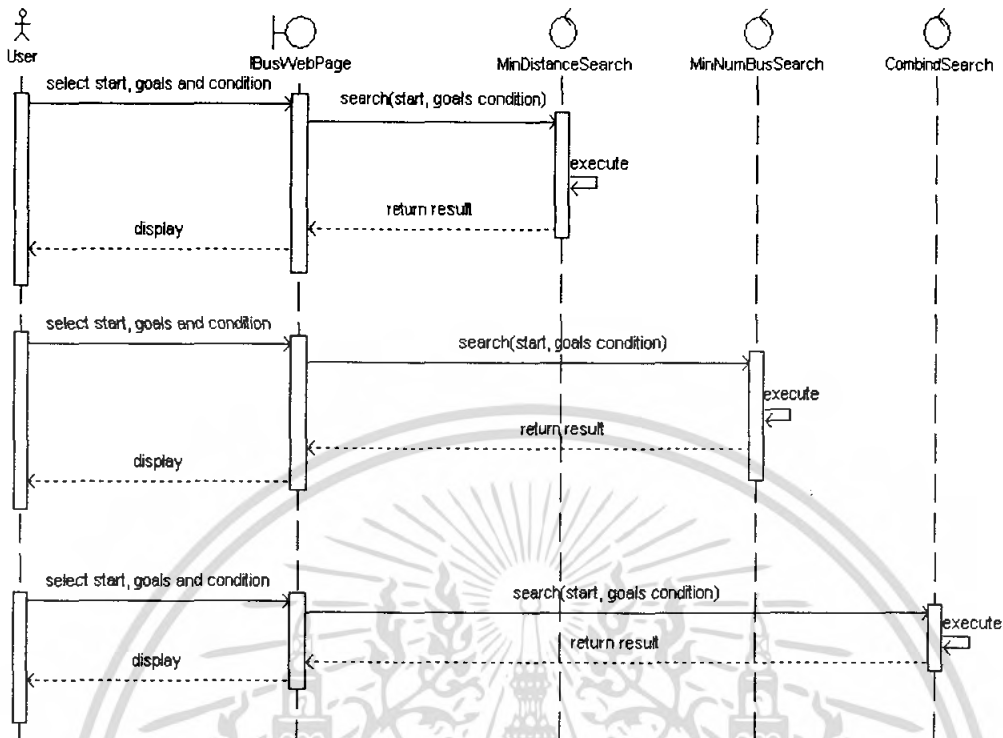
4.3 Class Diagram



รูปที่ 4.3 คลาสไดอะแกรม

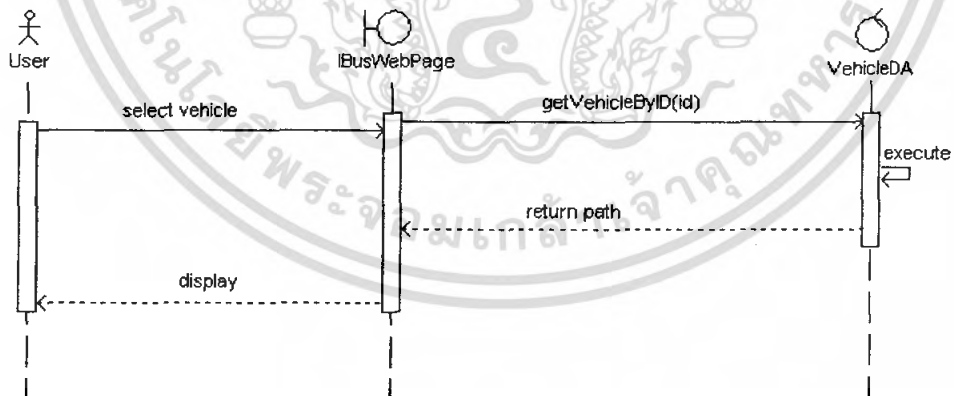
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 Sequence Diagram ของการค้นหาเส้นทาง



รูปที่ 4.4 ซีควเอนโคอะแกรมของการค้นหาเส้นทาง

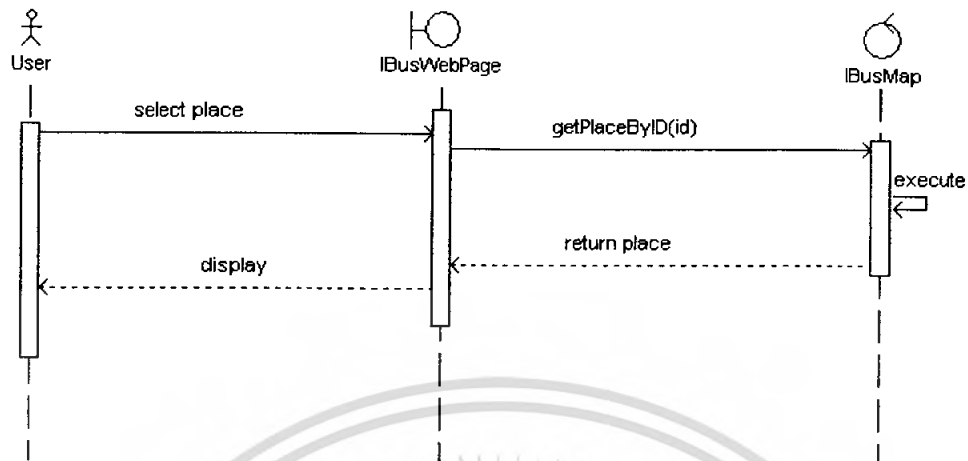
4.5 Sequence Diagram ของการเลือกดูเส้นทางสายรถประจำทาง



รูปที่ 4.5 ซีควเอนโคอะแกรมของการค้นหาเส้นทางสายรถประจำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 Sequence Diagram ของการเลือกดูข้อมูลสถานที่



รูปที่ 4.6 ซีควีนโคแอมของการค้นหาสถานที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การออกแบบ User Interface

5.1 UI ส่วนของผู้ใช้งาน

หน้าดั่งค้นหาเส้นทางนั้นถูกออกแบบมาให้สามารถใช้กับหน้าจอ Touch Screen ได้อย่างสะดวกซึ่งประกอบด้วย 5 ส่วนหลักดังนี้



รูปที่ 5.1 แสดง User Interface สำหรับบุคคลทั่วไป

5.1.1 ส่วนกำหนดจุดต้นทางและจุดปลายทาง

เป็นส่วนที่ให้ผู้ใช้งานได้กำหนดจุดต้นทางและจุดหมายปลายทาง โดยสามารถกำหนดได้ถึงระดับป้าย ดังรูป

5.1.2 ส่วนแสดงผลพัทธ์และจำนวนเส้นทางทั้งหมด

เป็นส่วนที่ผู้ใช้งานจะสามารถเลือกดูเส้นทางได้หลายๆเส้นทางซึ่งสามารถเรียงลำดับได้ทั้งแบบระยะทางใกล้ที่สุดหรือเรียงจากการเปลี่ยนสายรถน้อยที่สุด

5.1.3 ส่วนแสดงรายละเอียดแต่ละเส้นทาง

เป็นส่วนที่ผู้ใช้งานจะได้ข้อมูลรายละเอียดเกี่ยวกับสายรถว่ามีจำนวนกี่สาย เป็นรถประเภทใดบ้าง เลขรถเลขใดบ้าง ราคาเท่าไร ระยะทางในสายนั้นก็กิโล และรถวิ่งในช่วงเวลาใดบ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.4 ส่วนแสดงชั้นข้อมูลของสถานที่สำคัญ

เป็นส่วนชั้นข้อมูลสถานที่สำคัญซึ่งผู้ใช้สามารถเลือกที่จะแสดงหรือไม่แสดงก็ได้ โดยชั้นข้อมูลอยู่ทั้งหมดดังนี้

- โรงพยาบาล
- สถานีตำรวจ
- สถานีอนามัย
- กรม
- ตลาด
- สถานีรถไฟ
- มหาวิทยาลัย

5.1.5 ส่วนแผนที่กรุงเทพฯ

เป็นส่วนที่ผู้ใช้สามารถที่จะเลื่อนไปมาหรือซูมเข้าออกได้อย่างอิสระเพื่อดูแผนที่การเดินทางที่ผู้ใช้สนใจ โดยในส่วนนี้จะมี Navigator Bar เพื่ออำนวยความสะดวกอยู่

5.1.6 ส่วนแสดงสายรถประจำทางต่างๆ

เป็นส่วนที่ผู้ใช้สามารถที่จะดูเส้นทางรถประจำทางว่ามีสายใดบ้าง เส้นทางเป็นเช่นไร และสามารถดูเส้นทางของรถประจำทางทั้งหมดได้

5.2 UI ส่วนของการเพิ่มเส้นทางยานพาหนะ

จะเป็นส่วนที่สำคัญที่มีหน้าที่ให้เจ้าหน้าที่ทำการเพิ่มเส้นทางต่างลงไปในระบบ



รูปที่ 5.2 แสดง User Interface สำหรับเพิ่มเส้นทางรถประจำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งมีส่วนประกอบดังต่อไปนี้

5.2.1 ส่วนเพิ่มเส้นทางรถไฟฟ้า BTS และ MRT

เป็นส่วนที่ให้เจ้าหน้าที่สามารถเพิ่มเส้นทางของรถไฟฟ้า BTS และ MRT โดยในส่วนนี้ยังสามารถเพิ่มเส้นทางประเภทโคกก็ได้ที่อยู่ในรูปแบบของสถานีเจ้ารถไฟราง เรือ หรือรถตู้ เป็นต้น

5.2.2 ส่วนเพิ่มเส้นทางรถประจำทาง

เป็นส่วนที่ให้เจ้าหน้าที่สามารถเพิ่มเส้นทางของรถประจำทางทั้งธรรมดาและปรับอากาศ โดยระบบออกแบบมาให้สามารถเพิ่มเส้นทางอย่างง่ายดาย ด้วยตัวช่วยที่จะบอกว่าปัจจุบันมีถนนเส้นใดต่ออยู่กับถนนเส้นนั้น

5.2.1 ส่วนแผนที่กรุงเทพฯ

เป็นส่วนที่ผู้ใช้สามารถที่จะเลื่อนไปมาหรือซูมเข้าออก ได้อย่างอิสระเพื่อดูแผนที่ยานพาหนะที่ผู้ใช้สนใจ โดยในส่วนนี้จะมี Navigator Bar เพื่ออำนวยความสะดวกอยู่ และยังสามารถที่จะขยายหรือลดขนาดเส้นถนนเพื่อให้สามารถมองได้ง่ายขึ้น

5.3 UI ส่วนของการเพิ่มสถานที่สำคัญ

จะเป็นส่วนที่สำคัญที่มีหน้าที่ให้เจ้าหน้าที่ทำการเพิ่มสถานที่สำคัญต่างๆไปในระบบ



รูปที่ 5.3 แสดง User Interface สำหรับเพิ่มสถานที่สำคัญ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งมีส่วนประกอบดังต่อไปนี้

5.3.1 ส่วนเพิ่มสถานที่สำคัญ

เป็นส่วนที่ให้เจ้าหน้าที่สามารถเพิ่มสถานที่สำคัญโดยกำหนดจุดของสถานที่สำคัญอ้างอิงกับจุดบนเส้นถนน

5.3.2 ส่วนแผนที่กรุงเทพฯ

เป็นส่วนที่ผู้ใช้สามารถที่จะเลื่อนไปมาหรือซูมเข้าออกได้อย่างอิสระเพื่อดูแผนที่การเดินทางที่ผู้ใช้สนใจ โดยในส่วนนี้จะมี Navigator Bar เพื่ออำนวยความสะดวก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การทดลอง และผลการทดลอง

ในโครงการนี้เราได้พัฒนาระบบค้นหาเส้นทางรถประจำทางอัจฉริยะที่ยังเป็นต้นแบบ ดังนั้นการทดสอบระบบจึงยังไม่ได้ทดสอบในสภาพแวดล้อมจริงที่จะต้องมีผู้ใช้เข้ามาใช้งานระบบจำนวนมากพร้อมๆกัน

6.1 อุปกรณ์คอมพิวเตอร์ที่ใช้ในการทดลอง

ประกอบด้วยเครื่องคอมพิวเตอร์ PC 2 ชุดซึ่งใช้เป็นเครื่องไคลเอนต์และเซิร์ฟเวอร์ตามลำดับ ดังนี้

1. เครื่องไคลเอนต์

- CPU Intel Pentium 4 2.0 GHz
- RAM 256 MB
- 20 GB Hard Drive
- Microsoft Internet Explorer 7.0
- SVG Viewer 3.0

2. เครื่องเซิร์ฟเวอร์

- CPU Intel Pentium 4 3.0 GHz
- RAM 512 MB
- 20 GB Hard Drive
- Microsoft Internet Explorer 7.0
- SVG Viewer 3.0
- Apache 2.2
- Python 2.4
- Mod Python 3.3.1
- MySQL DBMS 2.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 ทดลองเพิ่มเส้นทางรถประจำทาง

ระบบเพิ่มสายรถประจำทางนั้นออกแบบมาเพื่อให้สามารถใช้งานง่ายที่สุด ดังนี้



รูปที่ 6.1 แสดง User Interface สำหรับเพิ่มเส้นทางรถประจำทาง

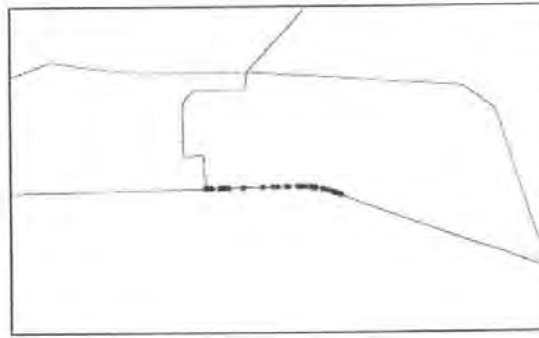
โดยเริ่มจากเลือกประเภทของรถประจำทาง ใส่หมายเลข ใส่ชื่อรถ เมื่อใส่เสร็จแล้วก็ทำการคลิกปุ่มเริ่มใส่เส้นทาง ดังรูป



รูปที่ 6.2 แสดงไดอะล็อกการสร้างเส้นทางรถประจำทาง

เมื่อคลิกปุ่มเริ่มใส่เส้นทางแล้วก็จะเป็นการเริ่มต้นเพิ่มเส้นทาง โดยในการกำหนดเส้นทางนั้นให้ทำการคลิกที่เส้นถนนบนแผนที่ เมื่อคลิกที่เส้นถนนแล้วจะแสดงจุดขึ้นมาบนถนนเพื่อให้เราสามารถกำหนดจุดเริ่มต้นของสายรถประจำทาง ดังรูปข้างล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.3 แสดงการกำหนดจุดเริ่มต้นของเส้นทาง

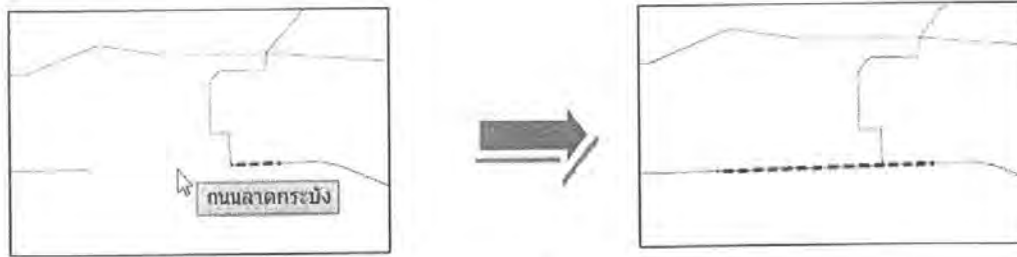
เมื่อกำหนดจุดแล้วโปรแกรมจะทำการลากจุดนั้นให้เกิดมาเป็นเส้นทางขึ้นมา แต่ทิศทางในบางครั้งอาจไม่เป็นไปตามที่เราต้องการเราสามารถปรับแก้ทิศทางด้วยการคลิกปุ่มกลับเส้นทางดังรูป



รูปที่ 6.4 แสดงการใช้งานปุ่มกลับทิศทางของเส้นทาง

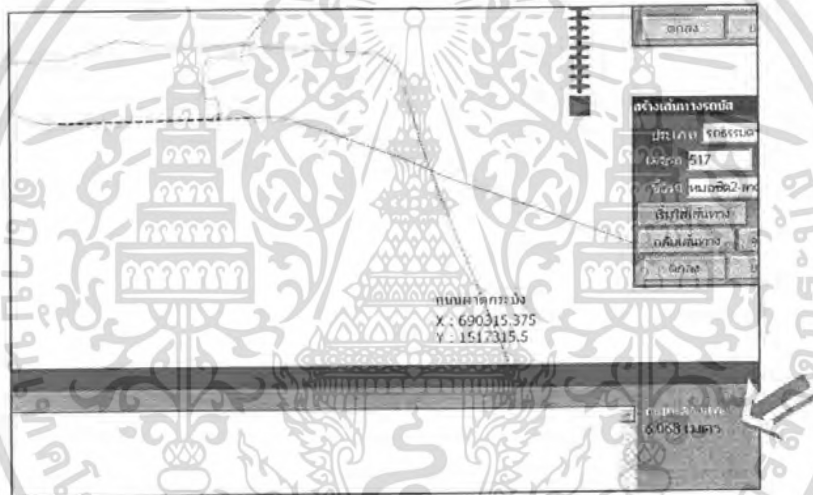
เมื่อเราได้ทิศทางที่ต้องการแล้วเราสามารถเพิ่มความความยาวของเส้นทางโดยการคลิกที่เส้นทางที่อยู่ติดกันกับเส้นสีแดง เส้นทางก็จะยาวขึ้นดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



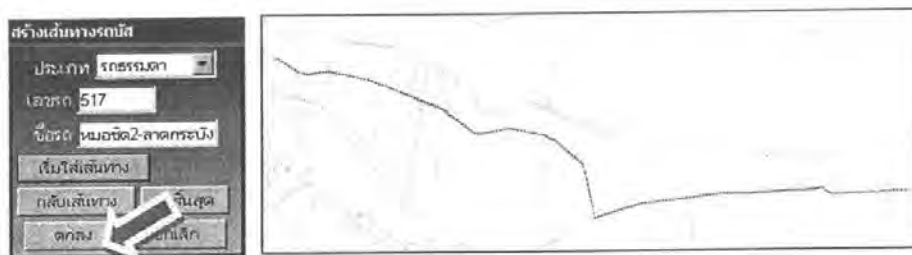
รูปที่ 6.5 แสดงการเพิ่มความยาวของเส้นทาง

แต่ในลักษณะบางประการของแผนที่ ซึ่งจะมีเส้นทางบางเส้นทางที่มีขนาดเล็กมากจนไม่สามารถคลิกเลือกได้ ทำให้ไม่สามารถต่อเส้นทางนั้นได้ ระบบเพิ่มสายรถประจำทางมีตัวช่วยคือ ส่วนแสดงผลว่าจุดข้างหน้าสุดของเส้นทางอยู่ติดกับถนนใดบ้าง และแต่ละถนนมีระยะทางเท่าใด ซึ่งเราสามารถคลิกเลือกที่ตัวช่วยนี้ได้ทันที ดังรูป



รูปที่ 6.6 แสดงตัวช่วยในการเพิ่มความยาวของเส้นทาง

เมื่อได้เส้นทางที่ต้องการแล้วก็ให้ทำการคลิกที่ปุ่มตกลง เพื่อเป็นการสิ้นสุดการเพิ่มเส้นทาง ดังรูปข้างล่าง



รูปที่ 6.7 แสดงการใส่เส้นทางที่สมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 ทดลองค้นหาเส้นทางจากต้นทางไปยังปลายทางแบบไม่มีจุดผ่าน

6.3.1 ทดลองค้นหาเส้นทางแบบที่มีระยะทางสั้นที่สุด



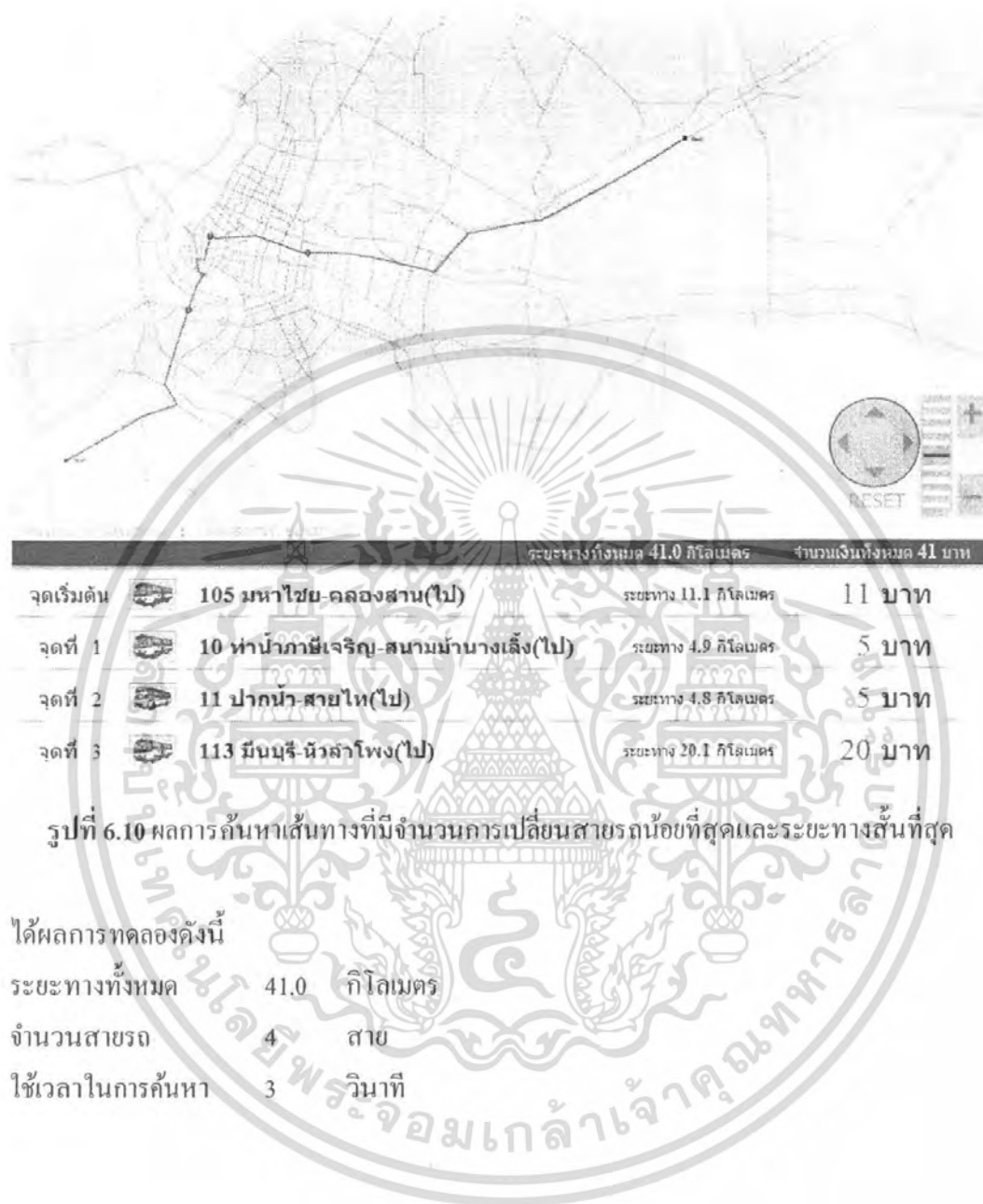
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.2 ทดลองค้นหาเส้นทางแบบที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.3 ทดลองค้นหาเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุดและระยะทางสั้นที่สุด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4 ทดลองค้นหาเส้นทางจากต้นทางไปยังปลายทางแบบมีจุดผ่าน 3 จุด

6.4.1 ทดลองค้นหาเส้นทางที่มีระยะทางสั้นที่สุด



รูปที่ 6.11 ผลการค้นหาเส้นทางที่มีระยะทางสั้นที่สุดแบบมีจุดผ่าน 3 จุด

ได้ผลการทดลองดังนี้

ระยะทางทั้งหมด	81.9	กิโลเมตร
จำนวนสายรถ	12	สาย
ใช้เวลาในการค้นหา	20	วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4.2 ทดลองค้นหาเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุด



		ระยะทางทั้งหมด 90.1 กิโลเมตร		จำนวนเงินทั้งหมด 128 บาท	
จุดที่ 4		136	หมอชิต2-ตุลาการกรร(กลับ)	ระยะทาง 9.3 กิโลเมตร	27 บาท
จุดที่ 5		136	ตุลาการกรร-หมอชิต2(ไป)	ระยะทาง 7.9 กิโลเมตร	5 บาท
จุดที่ 6		11	ประเวศ-มานูญครอง(กลับ)	ระยะทาง 13.6 กิโลเมตร	26 บาท
จุดที่ 7		11	มานูญครอง-ประเวศ(ไป)	ระยะทาง 9.5 กิโลเมตร	9 บาท
จุดที่ 8		113	มีนบุรี-ห้าเสาโพง(ไป)	ระยะทาง 13.8 กิโลเมตร	12 บาท

รูปที่ 6.12 ผลการค้นหาเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุดแบบมีจุดผ่าน 3 จุด

ได้ผลการทดลองดังนี้

ระยะทางทั้งหมด 90.1 กิโลเมตร

จำนวนสายรถ 9 สาย

ใช้เวลาในการค้นหา 40 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4.3 ทดลองค้นหาเส้นทางที่มีจำนวนการเปลี่ยนสายรถน้อยที่สุดและระยะทางสั้นที่สุด



ได้ผลการทดลองดังนี้

ระยะทางทั้งหมด	90.1	กิโลเมตร
จำนวนสายรถ	9	สาย
ใช้เวลาในการค้นหา	35	วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.5 ผลการทดลอง

จากผลการทดลองเราสามารถสรุปได้ดังต่อไปนี้

1. เวลาในการค้นหาเส้นทางเมื่อกำหนดเพียงจุดต้นทางกับจุดปลายทาง ทำงานได้รวดเร็ว
2. จุดต้นทางจุดปลายทางและจุดที่ต้องผ่านจะทำงานช้ามากขึ้นมากเมื่อกำหนดให้จำนวนจุดที่ต้องการผ่านมีมากกว่า 5 จุดขึ้นไป
3. เวลาการค้นหาเส้นทางที่ใช้เวลามากเมื่อจำนวนสายรถประจำทางในระบบมีจำนวนมากขึ้น
4. เวลาในการเริ่มต้นระบบในฝั่งเซิร์ฟเวอร์ทำงานได้รวดเร็ว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

บทสรุป

7.1 สรุปผลการทดลอง

ระบบค้นหาเส้นทางโดยใช้ A* สามารถทำงานได้ดี และจะสามารถใช้งานร่วมกับ User Interface ส่วนระบบแสดงผลแผนที่นั้นสามารถทำงานได้รวดเร็ว แต่เมื่อแสดงชั้นข้อมูลสถานที่ ซึ่งมีจำนวนเส้นของภาพมากขึ้น ความเร็วในการแสดงผลจะช้าลงมากซึ่งจะต้องมีการปรับปรุงต่อไป

7.2 วิจารณ์สิ่งที่ได้จากโครงการ

1. ได้ศึกษาระบบเส้นทางรถเมล์ที่มีอยู่ในกรุงเทพฯ และปริมณฑล
2. ได้ศึกษาการออกแบบ User Interface เพื่อใช้ในหน้าจอ Touch Screen
3. ได้ศึกษาการเขียน โปรแกรมค้นหาเส้นทางรถเมล์ที่มีความซับซ้อน
4. ได้ศึกษาการพัฒนาเว็บไซต์ด้วย AJAX
5. ได้ศึกษาการเขียน โปรแกรมเพื่อแสดงผลและควบคุมแผนที่
6. ได้ศึกษาการแก้ปัญหาต่างๆ ที่เกิดกับแผนที่ เช่น เส้นทางขาด เส้นทางซ้อนกัน เป็นต้น

7.3 ปัญหาอุปสรรคและแนวทางแก้ไข

1. เกิดความยากลำบากในการกำหนดเส้นทางเมื่ออ้างอิงตำแหน่งจากป้าย
2. เกิดความยากลำบากในการเพิ่มเส้นทางรถประจำทาง จึงได้จัดทำระบบเพิ่มเส้นทางรถประจำทางขึ้นเพื่อแก้ไขปัญหานี้ และเพิ่มความสะดวกในการเพิ่มสายรถต่างๆ ให้มีความสะดวกมากขึ้นกว่าเดิม
3. เกิดความยากลำบากในการปรับปรุงแผนที่เพื่อให้สามารถแสดงข้อมูลที่มีความละเอียดสูง และรายละเอียดครบถ้วน

7.4 การวิจัยและพัฒนาต่อ

1. เมื่อระบบนำไปใช้จริงจะต้องทำการออกแบบ User Interface ให้เหมาะกับประชาชน และต้องเข้าใจง่าย
2. ปรับปรุงระบบค้นหาเส้นทางให้มีความรวดเร็วมากขึ้น เพื่อให้สามารถรองรับการเข้ามาใช้งานของผู้ใช้ที่มีจำนวนมากๆ
3. ปรับปรุงการแสดงผลแผนที่ให้แสดงผลได้เร็วขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- ชัยยา กงศรี และ วิวัฒน์ หาญจิต. 2549. **จีไอเอสเพื่อการบรรเทาสาธารณภัยฉุกเฉิน**
 ปรินูญานิพนธ์วิศวกรรมศาสตร์บัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์
 คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ประภากร ลาภประสพ, วิศิษฐ์ หิรัญกิตติ และชมกิมปาน. 2543. **ระบบแผนที่ชาญฉลาด**
 การประชุมเสนอผลงานระดับบัณฑิต ครั้งที่ 1 มหาวิทยาลัยเชียงใหม่ เชียงใหม่
- Sylvain, H. 2007. **CherryPy Essentials Rapid Python Web Application Development**,
 Birmingham, UK
- Mark, R. 2006. **Rapid Web Applications with TurboGears: Using Python to Create Ajax-
 Powered Sites**, Boston, US
- Tyler, M. 2005. **Web Mapping Illustrated**, Sebastopol, US
- Ben Coppin. 2004. **Artificial Intelligence Illuminated**. Canada : Jones and Barlett.

ภาคผนวก ก

คู่มือการติดตั้งระบบและติดตั้งโปรแกรม

1 ความต้องการของระบบทางด้านซอฟต์แวร์

1.1 โปรแกรมภาษา Python เวอร์ชัน 2.4 สำหรับ Windows

ดาวน์โหลดโปรแกรม Python 2.4 สำหรับ Windows ได้จาก

<http://www.python.org>

1.2 โปรแกรม AppServ เวอร์ชัน 2.5 สำหรับ Windows

ดาวน์โหลดโปรแกรม AppServ เวอร์ชัน 2.5 (หรือเวอร์ชันที่สูงกว่านี้) ได้จาก

<http://www.appservnetwork.com>

1.3 โปรแกรม MySQL-Python เวอร์ชัน 1.2.2 สำหรับ Windows และ Python 2.4

ดาวน์โหลดโปรแกรม MySQL-Python เวอร์ชัน 1.2.2 สำหรับ Windows และ Python 2.4 ได้จาก <http://sourceforge.net/projects/mysql-python>

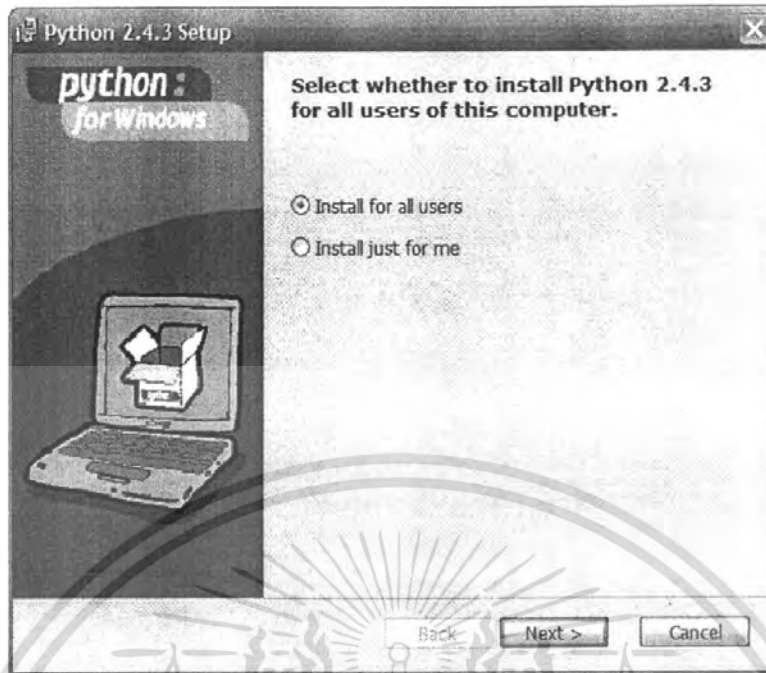
1.4 โปรแกรม mod_python เวอร์ชัน 3.2 สำหรับ Windows และ Apache 2.0

ดาวน์โหลดโปรแกรม mod_python เวอร์ชัน 1.2.2 สำหรับ Windows และ Python 2.4 ได้จาก <http://www.modpython.org>

2 การติดตั้งโปรแกรม

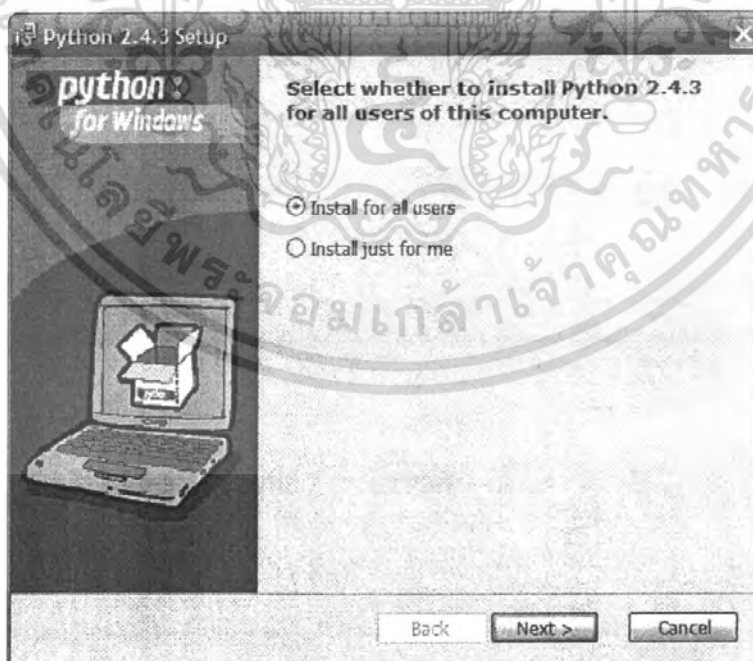
2.1 การติดตั้งโปรแกรมภาษา Python

ขั้นที่ 1 ดับเบิลคลิกที่ไฟล์ python-2.4.3.msi ซึ่งปรากฏข้อความดังรูปต่อไปนี้



รูปที่ ก.1 แสดงขั้นตอนภายหลังจากการดับเบิลคลิกไฟล์ python-2.4.3.msi

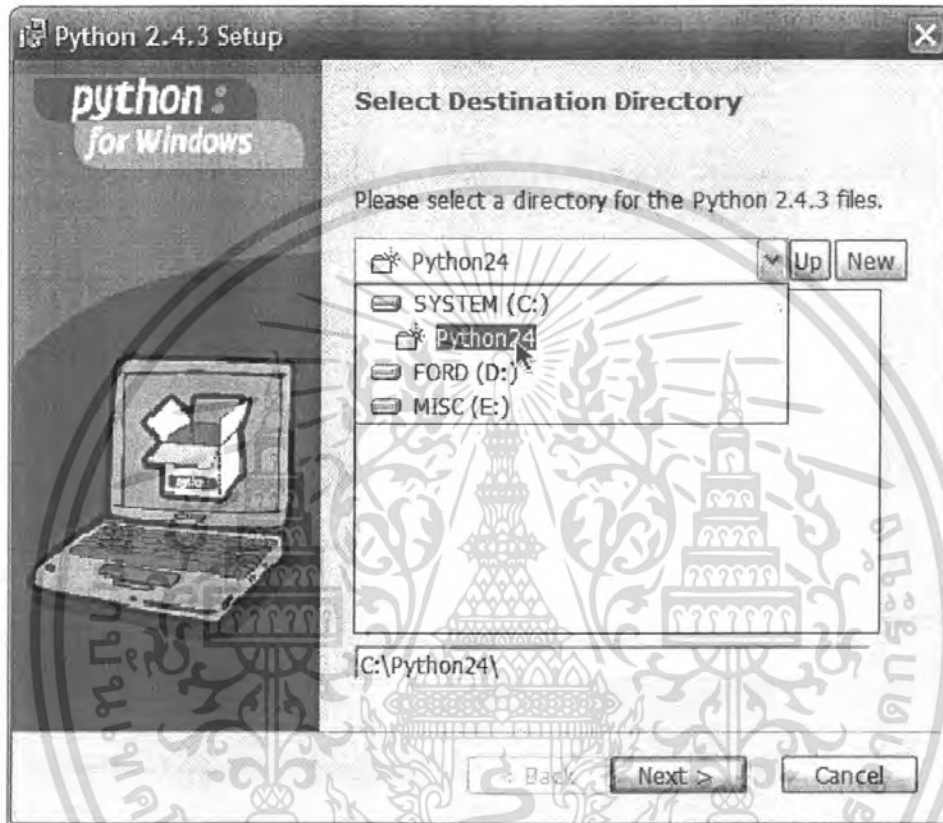
ขั้นที่ 2 เลือก “Install for all users” เพื่อให้ผู้ใช้งานทุกคนใช้งาน โปรแกรมไพธอน
ได้ หลังจากนั้นให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป



รูปที่ ก.2 แสดงขั้นตอนการเลือกกำหนดผู้ใช้งาน

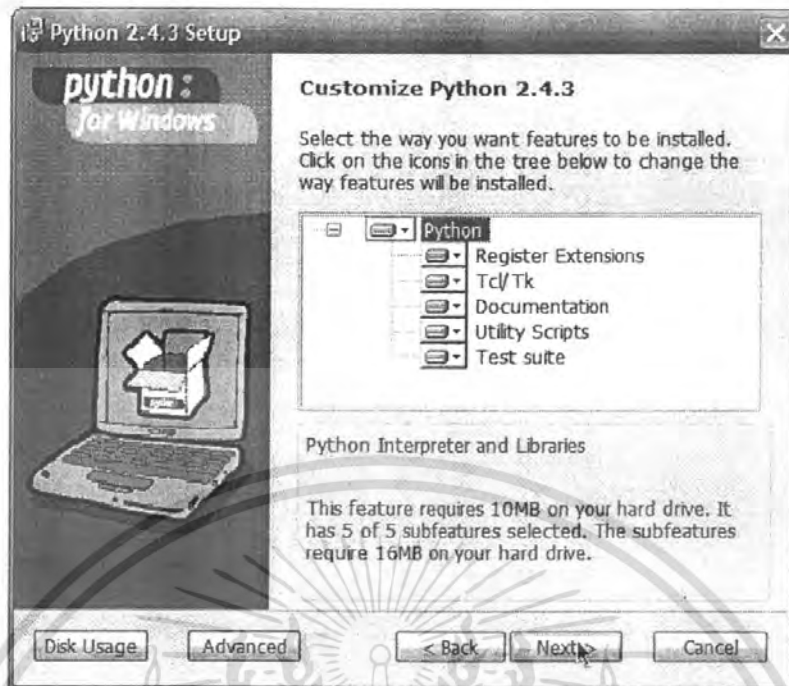
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นที่ 3 เลือกไดเรกทอรีที่ต้องการติดตั้ง เป็นขั้นตอนที่ท่านต้องกำหนดว่าจะติดตั้งโปรแกรมไพธอนไว้ที่ไดเรกทอรีใดภายในระบบปฏิบัติการของท่าน โดยปกติค่าที่กำหนดจากไพธอนจะเป็น C:\Python24 หลังจากนั้นให้คลิกปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป



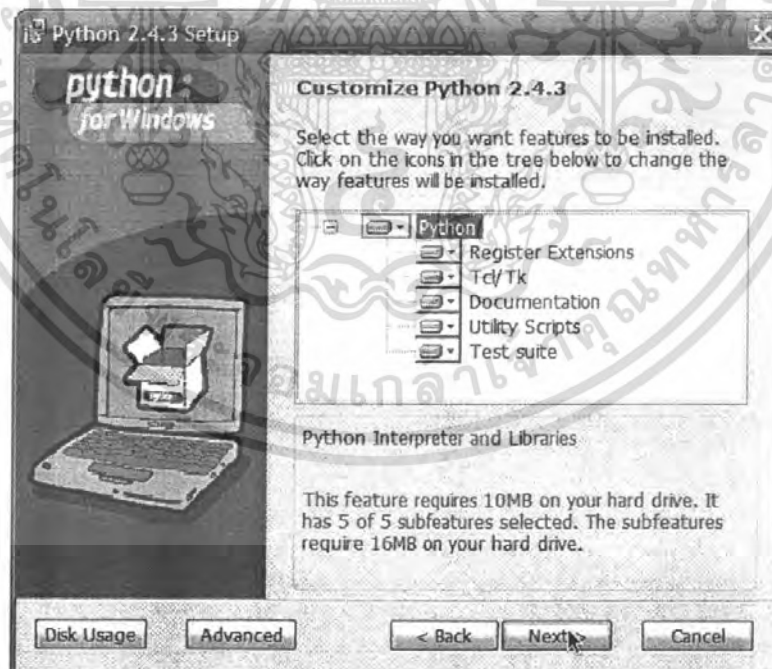
รูปที่ ก.3 แสดงขั้นตอนการกำหนด ไดเรกทอรีสำหรับติดตั้ง โปรแกรมภาษาไพธอน

ขั้นที่ 4 เลือกคอมโพเนนต์ที่จะติดตั้งลงในระบบ โดยเลือกติดตั้งทุกตัวเลือก หลังจากนั้นให้คลิกปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป



รูปที่ ก.4 แสดงรายการคอม โปเนนต์ที่จะถูกติดตั้งลงในระบบ

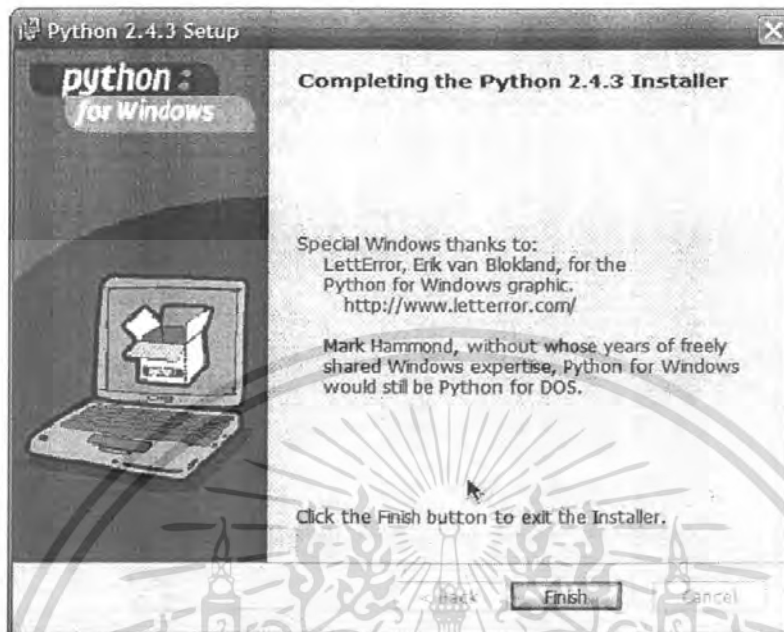
ขั้นที่ 5 ขั้นตอนการดำเนินการติดตั้งโปรแกรมภาษาไพธอนลงสู่ระบบคอมพิวเตอร์



รูปที่ ก.5 แสดงขั้นตอนการดำเนินการติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

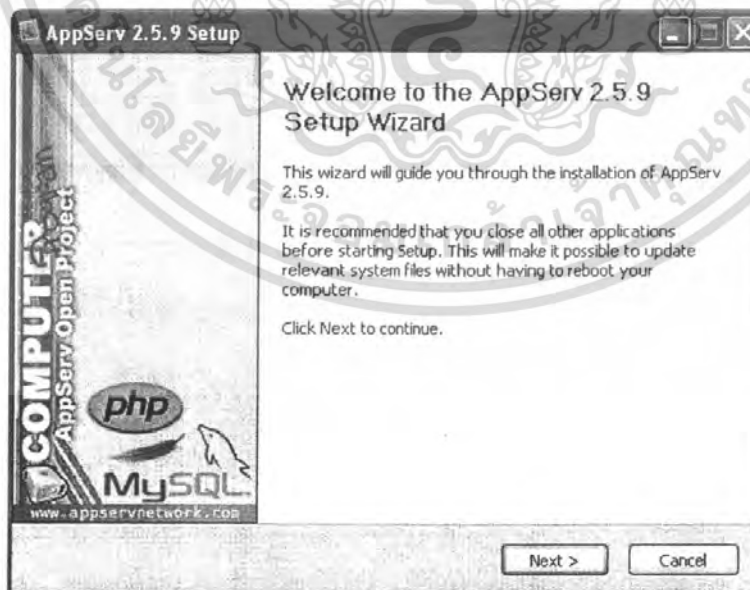
ขั้นที่ 6 ขั้นตอนการติดตั้งเสร็จสมบูรณ์ ให้กดปุ่ม Finish เพื่อเสร็จสิ้นการติดตั้ง



รูปที่ ก.6 แสดงขั้นตอนเสร็จสิ้นการลงโปรแกรม Python

2.2 ติดตั้งโปรแกรม AppServ

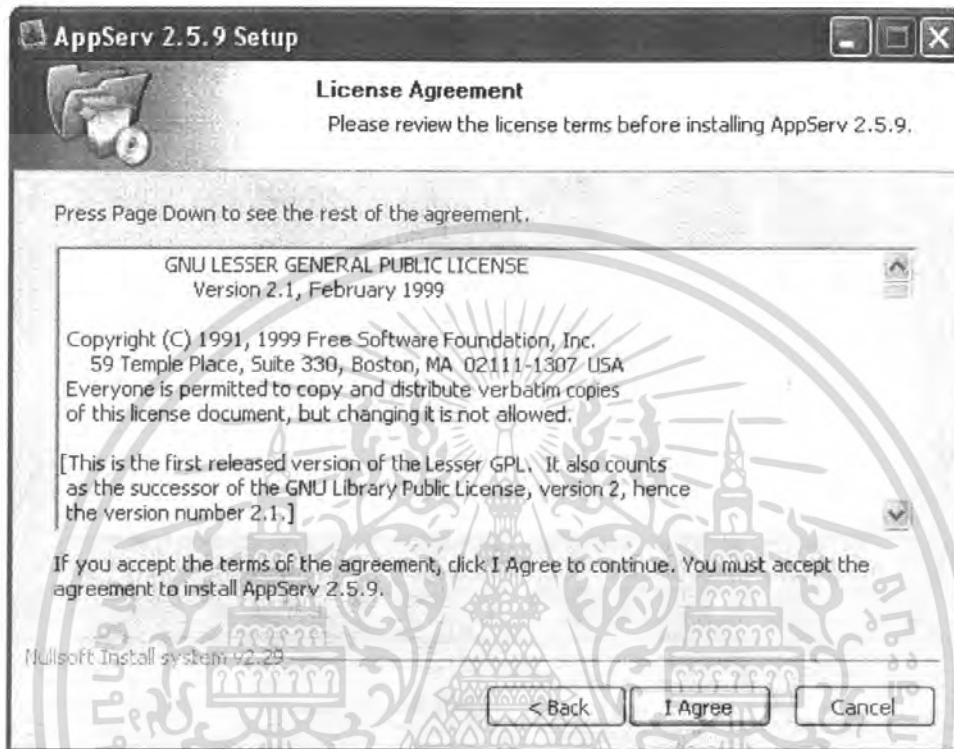
ขั้นที่ 1 ดับเบิลคลิกที่ไฟล์ appserv-win32-x.x.x.exe เพื่อทำการติดตั้ง จะปรากฏหน้าจอตามรูปที่ 8



รูปที่ ก.7 แสดงขั้นตอนภายหลังการดับเบิลคลิกไฟล์ appserv-win32.x.x.x.exe

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

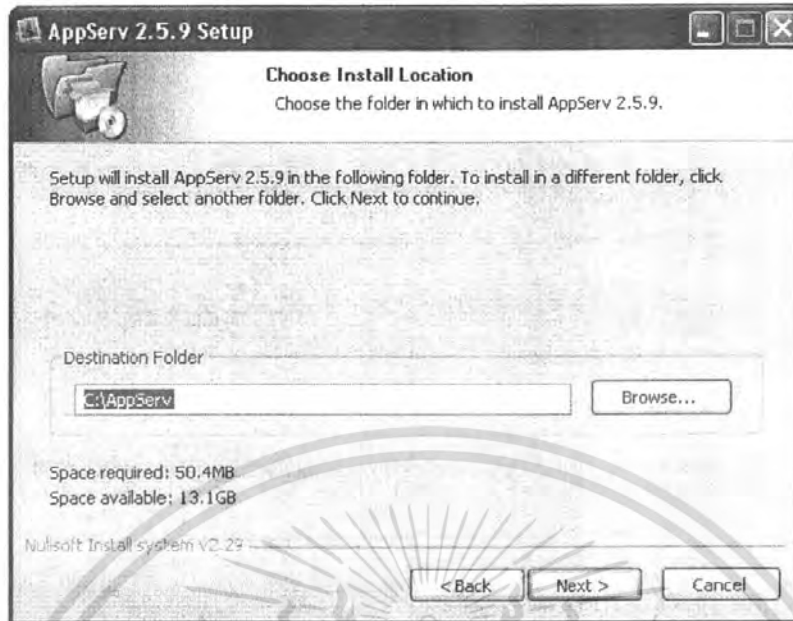
ขั้นตอนที่ 2 ขั้นตอนเงื่อนไขการใช้งานโปรแกรม โดยโปรแกรม AppServ ได้แจกจ่ายในรูปแบบ GNU License หากผู้ติดตั้งอ่านเงื่อนไขต่างๆเสร็จแล้ว หากยอมรับเงื่อนไขให้กด Next เพื่อสู่ขั้นตอนต่อไป



รูปที่ ก.8 แสดงขั้นตอนเงื่อนไขการใช้งานโปรแกรม

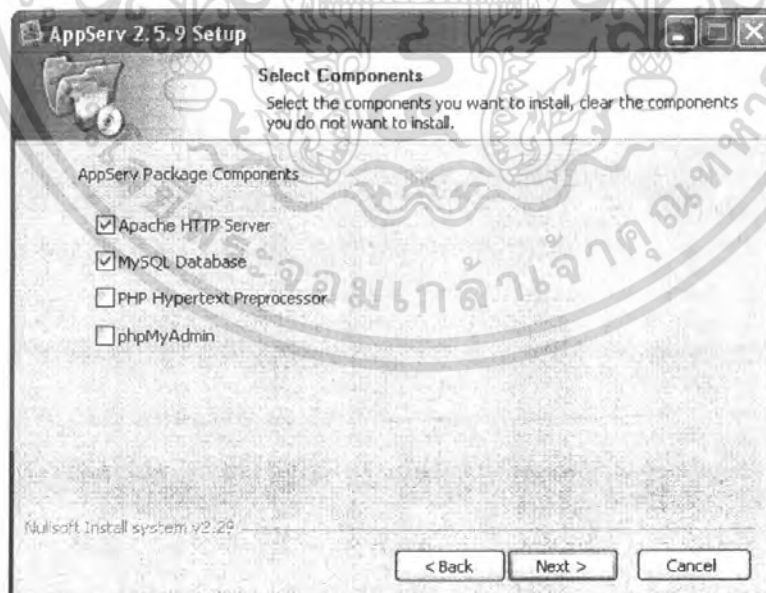
ขั้นตอนที่ 3 ขั้นตอนการเลือกไดเรกทอรีที่ต้องการติดตั้ง โดยค่าเริ่มต้นจะกำหนดอยู่ที่ C:\Appserv หากต้องการเปลี่ยนไดเรกทอรีที่ติดตั้ง ให้กด Browse แล้วเลือกไดเรกทอรีที่ต้องการ เมื่อเลือกไดเรกทอรีเสร็จสิ้น ให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.9 แสดงขั้นตอนการเลือกไดเรกทอรีที่จะทำการติดตั้งโปรแกรม AppServ

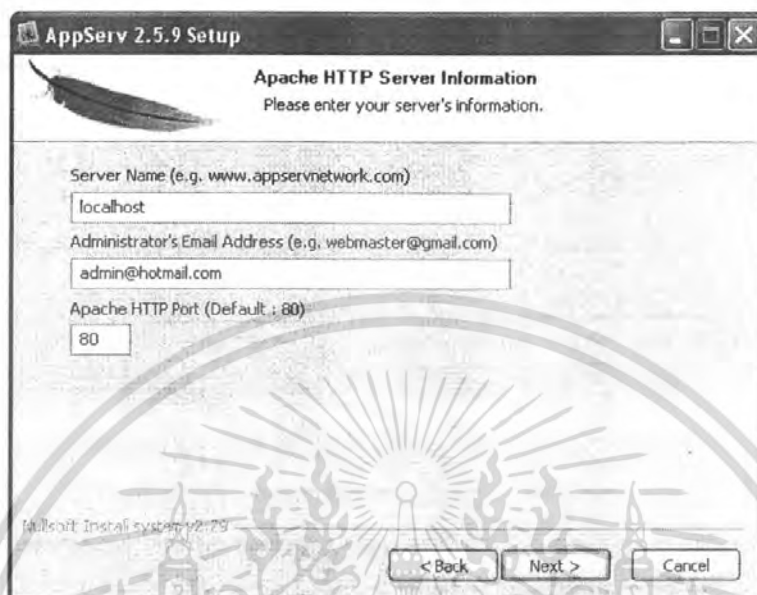
ขั้นตอนที่ 4 ขั้นตอนการแพ็คเกจคอมโพเนนต์ที่ต้องการติดตั้ง โดยค่าเริ่มต้นจะให้เลือกทุกแพ็คเกจ แต่สำหรับการติดตั้งระบบนี้ใช้เพียง 2 แพ็คเกจ คือ ApacheHTTPServer และ MySQL Database หลังจากนั้นให้คลิกปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป



รูปที่ ก.10 แสดงขั้นตอนการเลือกแพ็คเกจคอมโพเนนต์

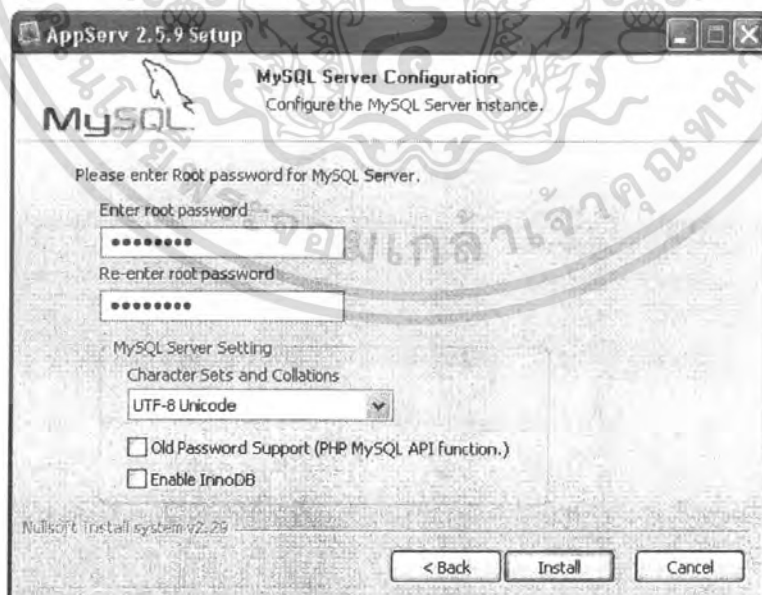
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 5 ขั้นตอนการกำหนดค่าคอนฟิกของ Apache Web Server มีอยู่ด้วยกันทั้งหมด 3 ส่วน เมื่อใส่ข้อมูลเสร็จแล้วให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป



รูปที่ ก.11 แสดงขั้นตอนการกำหนดค่าคอนฟิกของ Apache Web Server

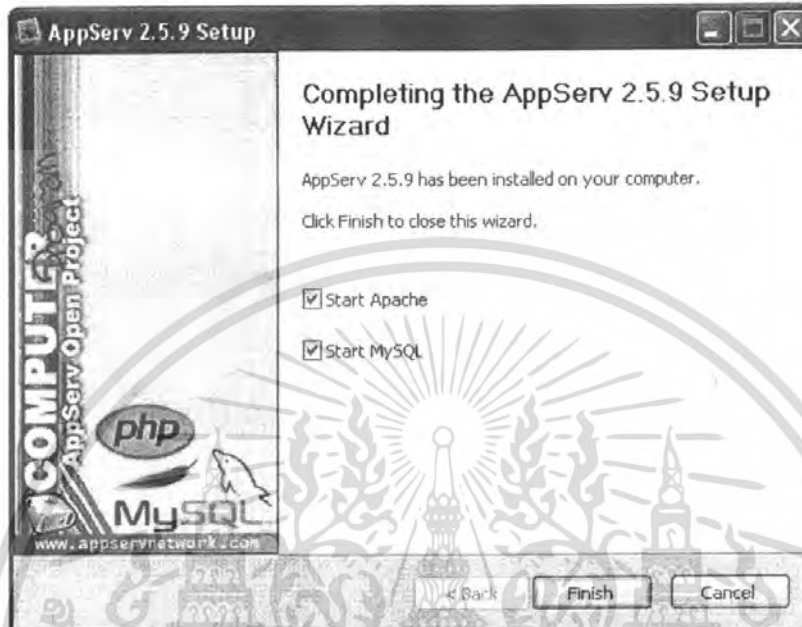
ขั้นตอนที่ 6 ขั้นตอนการกำหนดค่าคอนฟิกของ MySQL Database มีอยู่ด้วยกันทั้งหมด 3 ส่วน เมื่อใส่ข้อมูลเสร็จแล้วให้กดปุ่ม Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป



รูปที่ ก.12 แสดงขั้นตอนการกำหนดค่าคอนฟิกของ MySQL Database

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

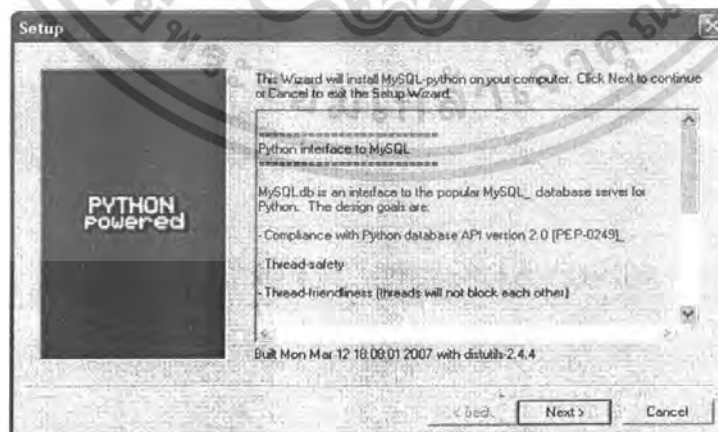
ขั้นตอนที่ 7 สิ้นสุดขั้นตอนการติดตั้งโปรแกรม AppServ สำหรับขั้นตอนสุดท้ายนี้ จะมีให้เลือกว่าต้องการสั่งให้มีการรัน Apache และ MySQL ทันทีหรือไม่ จากนั้นให้คลิกปุ่ม Finish เพื่อเสร็จสิ้นการติดตั้งโปรแกรม AppServ



รูปที่ ก.13 แสดงขั้นตอนเสร็จสิ้นการลงโปรแกรม AppServ

2.3 การติดตั้งโปรแกรม MySQL-Python

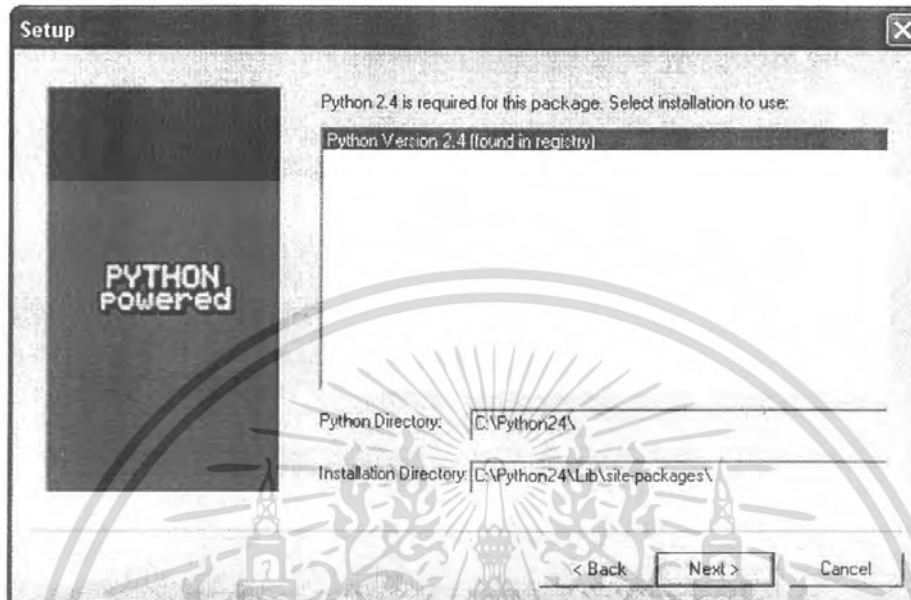
ขั้นที่ 1 ดับเบิลคลิกที่ไฟล์ MySQL-python-1.2.2.win32-py2.4.exe จะปรากฏหน้าจอจดังรูป



รูปที่ ก.14 แสดงขั้นตอนภายหลังการดับเบิลคลิกไฟล์ MySQL-python-1.2.2.win32-py2.4.exe

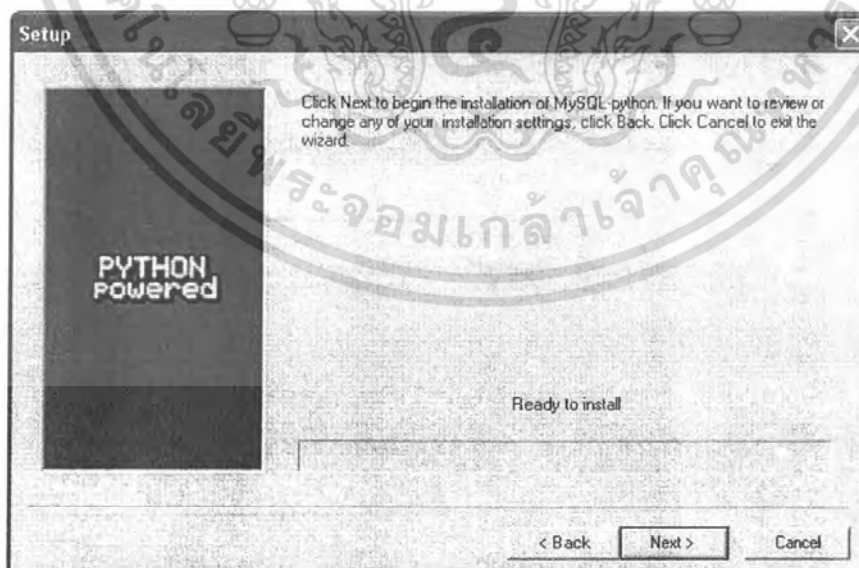
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นที่ 2 แสดงขั้นตอนการเลือกโปรแกรมไพธอน โดยในระบบต้องมีโปรแกรมไพธอน 2.4 อยู่ ให้คลิก Next เพื่อเข้าสู่ขั้นตอนต่อไป



รูปที่ ก.15 แสดงขั้นตอนการค้นหาโปรแกรมไพธอนเวอร์ชัน 2.4 สำหรับ MySQL-Python

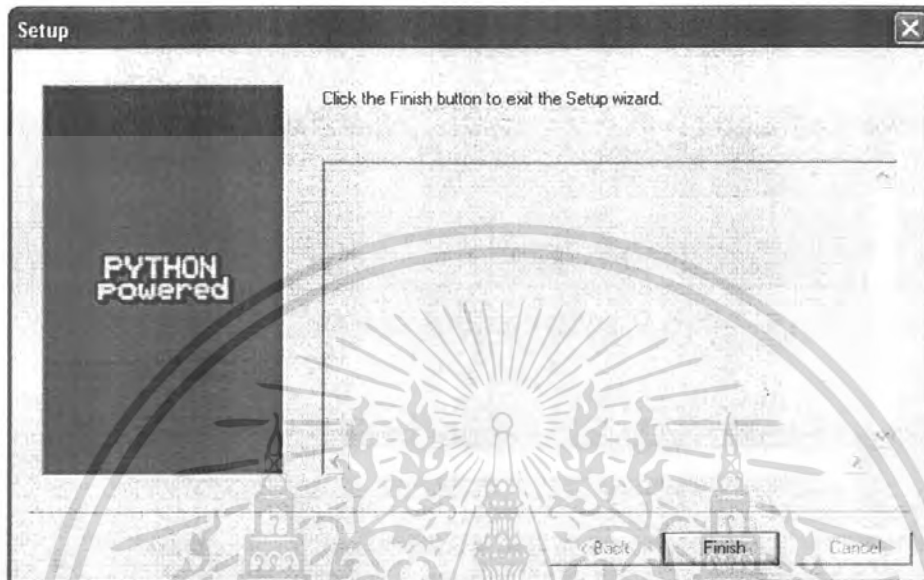
ขั้นตอนที่ 3 แสดงขั้นตอนพร้อมที่จะลงโปรแกรมสู่ระบบ ให้คลิก Next เพื่อเข้าสู่ขั้นตอนต่อไป



รูปที่ ก.16 แสดงขั้นตอนพร้อมลงโปรแกรมสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

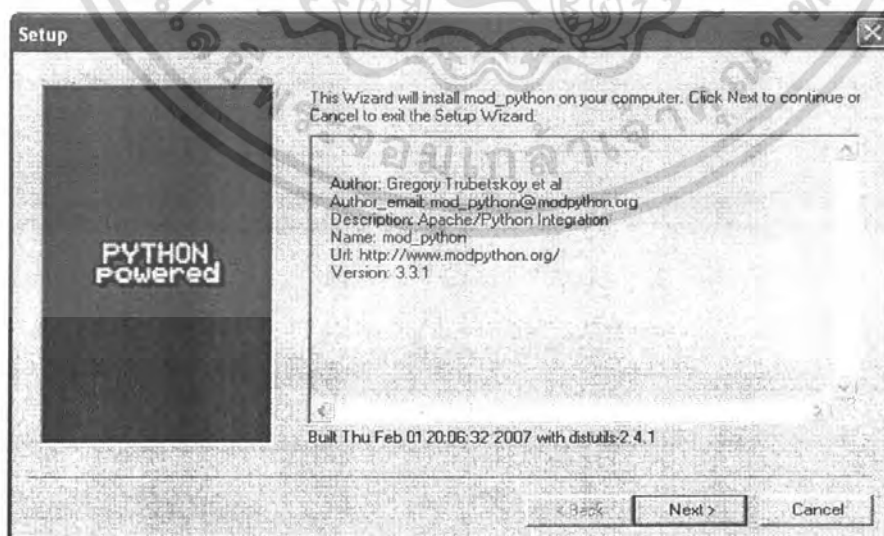
ขั้นที่ 4 แสดงขั้นตอนเสร็จสิ้นการลงโปรแกรม ให้คลิก Finish เพื่อเสร็จสิ้นการลงโปรแกรม



รูปที่ ก.17 แสดงขั้นตอนเสร็จสิ้นการลงโปรแกรม MySQL-Python

2.4 ติดตั้งโปรแกรม mod_python

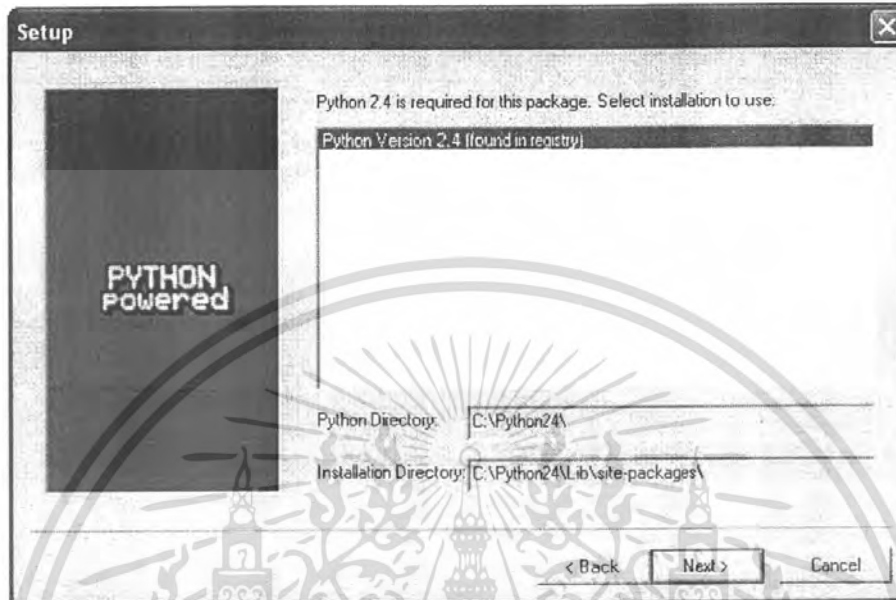
ขั้นตอนที่ 1 ให้ดับเบิลคลิกที่ไฟล์ mod_python-3.3.1.win32-py2.4-Apache2.2.exe จะปรากฏหน้าต่างดังรูป



รูปที่ ก.18 แสดงขั้นตอนภายหลังการดับเบิลคลิกไฟล์

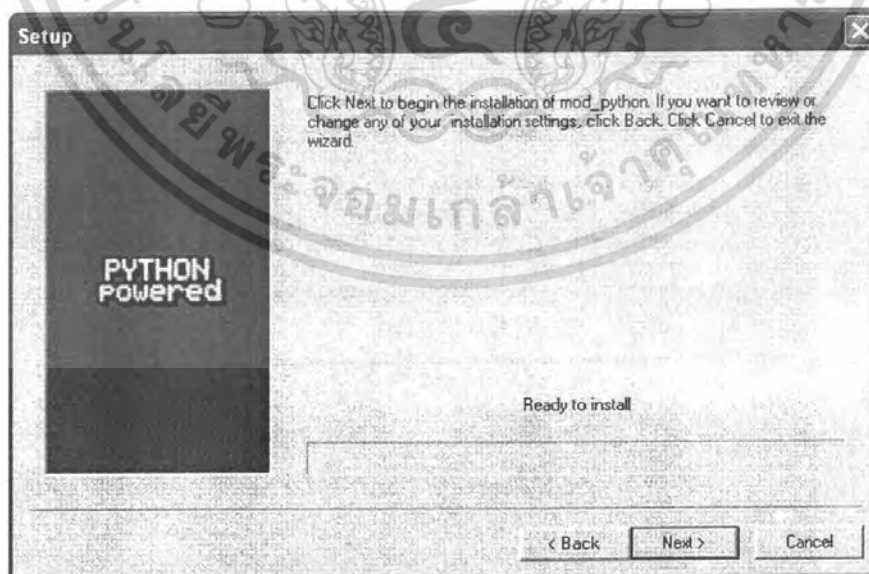
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 แสดงขั้นตอนการเลือกโปรแกรมไพธอน ซึ่งบนระบบต้องมีโปรแกรมไพธอนเวอร์ชัน 2.4 อยู่ ให้คลิก Next เพื่อเข้าสู่ขั้นตอนต่อไป



รูปที่ ก.19 แสดงขั้นตอนการค้นหาโปรแกรมไพธอนเวอร์ชัน 2.4 สำหรับ mod_python

ขั้นตอนที่ 3 แสดงขั้นตอนการแสดงความพร้อมที่จะลงโปรแกรม ให้คลิก Next เพื่อเข้าสู่ขั้นตอนต่อไป



รูปที่ ก.20 แสดงความพร้อมที่จะลงโปรแกรม mod_python

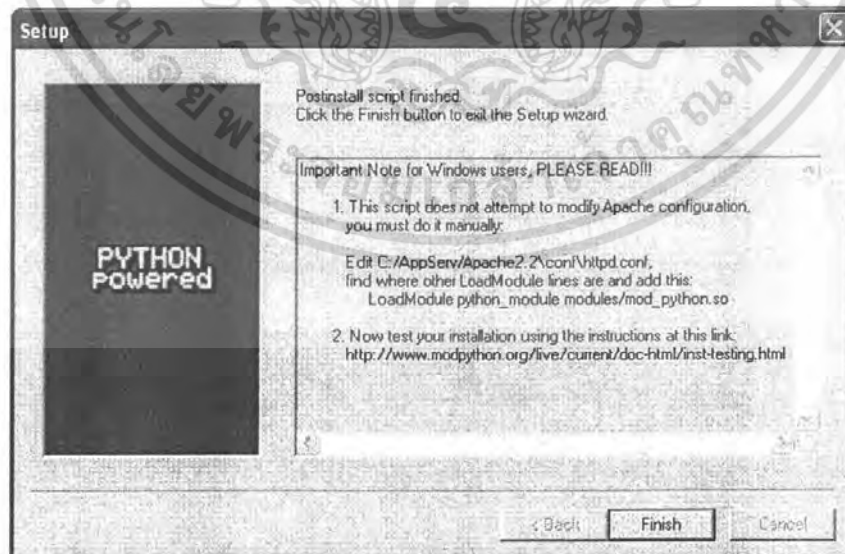
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 4 แสดงขั้นตอนการเลือกปรับปรุงค่าคอนฟิกของ apache โดยให้เราเลือกไดเรกทอรีของ apache ที่อยู่ในระบบเรา



รูปที่ ก.21 แสดงขั้นตอนการเลือก apache ที่จะทำการแก้ไขค่าคอนฟิก

ขั้นตอนที่ 5 ขั้นตอนสิ้นสุดการลงโปรแกรม ให้คลิก Next เพื่อสิ้นสุดการลงโปรแกรม



รูปที่ ก.22 แสดงขั้นตอนสิ้นสุดการลงโปรแกรม mod_python

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 ติดตั้งระบบ

ให้คัดลอกไฟล์ทั้งหมดจาก D:\www\ ไปไว้ที่ไดเรกทอรี C:\AppServ\www\ และ
คัดลอกไฟล์เตอร์ ibusdb จาก D:\MySQL\data\ ไปไว้ที่ C:\AppServ\MySQL\data\

2.7 แก้ไขค่าคอนฟิกของ apache

ทำให้เปิดไฟล์ httpd.conf ที่อยู่ในไดเรกทอรี C:\AppServ\Apache2.2\conf\ ขึ้นมา
แก้ไข ซึ่งอาจเปิดด้วยโปรแกรม Notepad ก็ได้ โดยเพิ่มข้อความดังรูปลงไปท้ายสุดของ
ไฟล์

```
LoadModule python_module modules/mod_python.so
<Directory /AppServ/www>
  AddHandler mod_python .py
  PythonHandler mod_python.publisher
  PythonDebug on
</Directory>
```

รูปที่ ก.23 แสดงค่าคอนฟิก