

พัฒนาระบบไอทีเพื่อสถานีรถไฟฟ้ามหานคร : แอปพลิเคชันไอโอเอสเพื่อระบบบนแพลตฟอร์ม
ของสถานีรถไฟฟ้ามหานคร

PTT MIS for Life Station : an iOS application for the PTT life Station
Backend System



โดยงานพิเศษประจำหน่วยงานของกรมการศึกษาด้านเทคโนโลยี สถาบันส่งเสริม
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา ๒๕๕๖

พีทีทีเอ็มไอเอสฟอรัลไฟส์เตชัน : แอปพลิเคชันไอโอเอสเพื่อระบบแบคเอนด์
ของพีทีทีไลฟ์สเตชัน

PTT MIS for Life Station : an iOS application for the PTT life Station
Backend System



โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานคณะวิทยาศาสตร์ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่มีการนำไปใช้

ปีการศึกษา 2555

**PTT MIS for Life Station : an iOS application for the PTT life Station
Backend System**



**A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE
IN COMPUTER SCIENCE**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในเท่านั้น ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆก็ตาม

FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2012

หัวข้อโครงการพิเศษ พิธีที่เอ็มไอเอสฟอร์ไลฟ์สเตชัน : แอปพลิเคชันไอโอเอสเพื่อระบบแบค
แอนด์ของพีทีทีไลฟ์สเตชัน

PTT MIS for Life Station : an iOS application for the PTT life Station
Backend System

ชื่อนักศึกษา นายกันตภณ พงษ์พานิช 52050659
 นายพนิต ปัญญากร 52050755

ปริญญา วิทยาศาสตรบัณฑิต

สาขาวิชา วิทยาการคอมพิวเตอร์

ปีการศึกษา 2555

อาจารย์ที่ปรึกษา ดร.รุ่งรัตน์ เวียงศรีพนาวัลย์

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้
โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการ
คอมพิวเตอร์ ประจำปีการศึกษา 2555

คณะกรรมการสอบ	ลายมือชื่อ
ผศ.ศิริลักษณ์ อนันต์สถิตย์สิน (ประธานกรรมการ)	
ดร.สุวรรณ จันทิวาสารกิจ (กรรมการ)	
ดร.รุ่งรัตน์ เวียงศรีพนาวัลย์ (กรรมการและอาจารย์ที่ปรึกษา)	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการพิเศษ	พีทีทีเอ็มไอเอสฟอร์ไลฟ์สเตชัน : แอปพลิเคชันไอโอเอสเพื่อระบบแบคเอนด์ของพีทีทีไลฟ์สเตชัน		
ชื่อนักศึกษา	นายกันตภณ พงษ์พานิช		52050659
	นายพนิต ปัญญากร		52050755
ปริญญา	วิทยาศาสตรบัณฑิต		
สาขาวิชา	วิทยาการคอมพิวเตอร์		
ปีการศึกษา	2555		
อาจารย์ที่ปรึกษา	ดร.รุ่งรัตน์	เวียงศรีพนาวัลย์	

บทคัดย่อ

การพัฒนาแอปพลิเคชัน PTTMIS for Life Station บนระบบปฏิบัติการไอโอเอส ให้กับทางบริษัท ปตท. มีวัตถุประสงค์หลักเพื่ออำนวยความสะดวกให้กับผู้ดูแลระบบของแอปพลิเคชัน PTT Life Station ให้สามารถดูแลและจัดการข้อมูลผ่านทางอุปกรณ์ไอโอเอส เช่น ไอโฟนได้ทุกที่ทุกเวลา และมีวัตถุประสงค์รอง คือการใช้แอปพลิเคชัน PTTMIS นี้เป็นต้นแบบในการพัฒนาแอปพลิเคชันบนโทรศัพท์มือถือเพื่อผู้ดูแลระบบให้กับ ระบบอื่นๆ ของทางบริษัท แอปพลิเคชัน PTTMIS ประกอบไปด้วยเมนูทั้งหมดเก้าเมนู แอปพลิเคชันแรกจะเป็นเมนูที่เกี่ยวกับการจัดการข้อมูล (เพิ่ม ลบ หรือ แก้ไข) ที่ใช้ในแอปพลิเคชัน PTT Life Station ทั้งหมด ได้แก่ สถานีบริการน้ำมัน (Gas Station) ชนิดของน้ำมัน (Oil Type) โพรโมชัน (Promotion) สถานที่ท่องเที่ยว (Place Travel) ชนิดของสถานที่ท่องเที่ยว (Travel Type) ช่วงราคา (Price Range) ชนิดของร้านอาหาร (Travel Eat Type) และ แอปพลิเคชันอื่นๆ ที่ทาง แอปพลิเคชัน PTT Life station สามารถเชื่อมต่อไปได้ (Other Application) ส่วนเมนูสุดท้ายจะเกี่ยวกับรายละเอียดการใช้งานระบบของผู้ดูแลระบบ (Log Report) ในการใช้งานแอปพลิเคชันผู้ดูแลระบบสามารถจัดการข้อมูลได้ทุกที่ ไม่จำกัดเฉพาะในเครือข่ายอินเทอร์เน็ตของบริษัทเท่านั้น โดยการเชื่อมต่อระหว่างไอโฟนกับเว็บเซิร์ฟเวอร์จะทำผ่านโปรโตคอล https ในส่วนของการพัฒนาแอปพลิเคชันนั้นประกอบไปด้วยสองส่วนคือส่วนของ REST เว็บเซิร์ฟเวอร์ และส่วนของแอปพลิเคชันไอโอเอสซึ่งใช้ภาษา PHP และ Objective-C ในการพัฒนาตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	PTT MIS for Life Station : an iOS application for the PTT life Station Backend System		
Students	Mr. Guntapon Pongpanich	52050659	
	Mr. Panit Panyakorn	52050755	
Degree	Bachelor of Science		
Major Program	Computer Science		
Academic Year	2012		
Advisor	Dr. Rungrat Wiangsripanawan		

ABSTRACT

This special project aim was to develop an iOS application called “PTT MIS for Life Station (PTT MIS)” for the Petroleum Authority of Thailand (PTT). The application’s main objective is to provide a convenient way for the PTT Life Station application’s administrators to easily manage the system’s backend through the iOS device, in particular, the iPhone. Unlike the current web application system which allows administrators to manage the information only in the PTT’s intranet, the PTT MIS allows them to access the system anytime and anywhere. The second objective is to use this application as the prototype to promote the concept of mobile applications for administrators to other systems in the PTT. The PTT MIS application consists of nine menus: Gas Station, Oil Type, Promotion, Place Travel, Travel Type, Price Range, Travel Eat Type, Other Application and Log Report. The first eight menus are used for managing (add/update/delete) each type of the PTT Life Station’s information and the last menu is used for auditing. The communication between the server and the device is done through the https session. Hence, the application also allows the PTT Life Station’s administrators to manage the system from outside the PTT’s intranet network. The implementation consists of two parts: REST web service on the server and iOS application on the iOS device. PHP is used to implement the web service and Objective-C is used to implement the iOS application.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปัญหาพิเศษหัวข้อ โปรแกรมไอโฟนเพื่อระบบแบคเอนด์ของแอปพลิเคชันพีทีที โดฟีสเตชัน สำเร็จลุล่วงไปได้ด้วยดี เนื่องจากผู้จัดทำได้รับความช่วยเหลือจากบุคคลผู้มีพระคุณ หลายท่านดังนี้

ขอขอบพระคุณ บิดา มารดา และสมาชิกในครอบครัวทุกคน ผู้ซึ่งอยู่เบื้องหลัง คอยอบรม สั่งสอน ให้กำลังใจตลอดการจัดทำปัญหาพิเศษ

ขอขอบพระคุณ ดร.รุ่งรัตน์ เวียงศรีพนาวัลย์ อาจารย์ที่ปรึกษาในการทำหัวข้อโครงการ พิเศษนี้ ที่ได้ให้คำปรึกษาและคำแนะนำในการแก้ปัญหาต่างๆ รวมถึงการตรวจสอบและแก้ไขการ เขียนรายงานปัญหาพิเศษ

ขอขอบพระคุณ อาจารย์สาขาวิทยาการคอมพิวเตอร์ทุกท่าน ที่อบรมสั่งสอน ให้ความรู้ทั้ง ภาคทฤษฎีและภาคปฏิบัติให้แก่ทางคณะผู้จัดทำมาตลอดระยะเวลา 4 ปี จนกระทั่งปัญหาพิเศษนี้ได้ สำเร็จลุล่วงไปได้ด้วยดี

ขอขอบพระคุณ ดร.สันติพัฒน์ อรุณธารี ที่คอยให้คำปรึกษา และให้แนวทางในการจัดทำ ปัญหาพิเศษ

ขอขอบพระคุณ พี่ๆทุกคนจากบริษัท พีทีที ไอซีที โซลูชันส์ จำกัด ที่สนับสนุนอุปกรณ์ใน การจัดทำปัญหาพิเศษ รวมถึงให้ความรู้ทั้งด้านภาคทฤษฎีและภาคปฏิบัติในการจัดทำปัญหาพิเศษ สุดท้ายนี้ต้องขอบคุณ เพื่อนๆทุกคน ที่ให้คำปรึกษาและให้กำลังใจเสมอมา

นายกันตภณ พงษ์พานิช

นายพนิต ปัญญากร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VIII
สารบัญรูป	IX

บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของปัญหา	2
1.4 ประโยชน์ที่จะได้รับ	3
1.5 ขั้นตอนการดำเนินงาน	3
1.6 อุปกรณ์ที่ใช้ในการทำปัญหาพิเศษ	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	5
2.1 Mobile Application	5
2.2 Bring Your Own Device	5
2.3 Xcode	5
2.4 ภาษาที่ใช้ในการพัฒนา	6
2.4.1 Objective-C	6
2.4.2 PHP	6
2.5 เซิร์ฟเวอร์ที่ใช้พัฒนา	7
2.5.1 Appserv	7
2.6 REST เว็บเซอร์วิส	8
2.7 เฟรมเวิร์คที่ใช้พัฒนา	8
2.7.1 RestKit	8

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านอื่น
ไม่ว่ากรณีใดก็ตาม หากมีข้อผิดพลาดประการใด ขออภัยและต้องอภัยถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.7.2 Zend เฟรมเวิร์ค	8
2.8 เครื่องมือที่ใช้ทดสอบระบบ	9
2.8.1 Acunetix Web Vulnerability Scanner	9
บทที่ 3 วิธีดำเนินงาน	10
3.1 ระบบของงาน	10
3.1.1 Use Case Diagram	10
3.1.2 ER Diagram	11
3.1.3 Activity Diagram	12
3.1.3.1 การล็อกอิน	12
3.1.3.2 การเลือกเมนู	13
3.1.3.3 การดูข้อมูล	14
3.1.3.4 การเพิ่มข้อมูล	16
3.1.3.5 การแก้ไขข้อมูล	18
3.1.3.6 การลบข้อมูล	20
3.1.4 Sequence Diagram	22
3.1.4.1 การล็อกอิน	22
3.1.4.2 การแสดงรายการข้อมูล	23
3.1.4.3 การดูข้อมูล	23
3.1.4.4 การเพิ่มข้อมูล	24
3.1.4.5 การแก้ไขข้อมูล	25
3.1.4.6 การลบข้อมูล	26
3.2 การออกแบบงาน	27
3.2.1 หน้าจอล็อกอิน	27
3.2.2 หน้าจอเมนูหลัก	28
3.2.3 หน้าจอกรอกรายละเอียด Gas Station	28
3.2.4 หน้าจอกรอกรายละเอียด Place Travel	29
3.2.5 หน้าจอกรอกรายละเอียด Oil Type Setting	29
3.2.6 หน้าจอกรอกรายละเอียด Travel Place Type Setting	30

สารบัญ (ต่อ)

	หน้า
3.2.7 หน้าจอกรอกรายละเอียด Price Range Setting	30
3.2.8 หน้าจอกรอกรายละเอียด Food Type Setting	31
3.2.9 หน้าจอกรอกรายละเอียด Promotion	31
3.2.10 หน้าจอกรอกรายละเอียด Other Application	32
3.2.11 หน้าจอ Log Report	32
3.3 การพัฒนาแอปพลิเคชัน	33
3.3.1 การพัฒนา REST เว็บเซอร์วิส	33
บทที่ 4 ผลการดำเนินงานและการทดสอบระบบ	37
4.1 ผลการดำเนินงาน	37
4.2 การตรวจสอบโค้ด	62
4.2.1 ระดับ Medium	62
4.2.1.1 Apache httpd Remote Denial of Service	62
4.2.1.2 Apache httpOnly Cookie Disclosure	62
4.2.1.3 Error message on page	62
4.2.1.4 User credentials are sent in clear text	62
4.2.2 ระดับ Low	63
4.2.2.1 Error page Path disclosure	63
4.2.2.2 Possible sensitive directories	63
4.2.2.3 Session token in URL	63
4.2.3 ระดับ Informational alert	63
4.2.3.1 Broken links	63
4.2.3.2 Email address found	63
4.2.3.3 Password type input with auto-complete enabled	64
4.2.3.4 Possible internal IP address disclosure	64
4.2.3.5 Possible server path disclosure (Unix)	64
4.2.3.6 Possible username or password disclosure	64
4.3 การทดสอบการใช้งาน	64
4.3.1 ทดสอบการแสดงผลของข้อมูล	64

สารบัญ (ต่อ)

	หน้า
4.3.2 ทดสอบการจัดการข้อมูลภายในฐานข้อมูล	64
4.3.3 ทดสอบการเชื่อมต่อแอปพลิเคชันเข้ากับเว็บเซอร์วิส	64
บทที่ 5 บทสรุปและข้อเสนอแนะ	65
5.1 บทสรุป	65
5.2 ข้อจำกัดการพัฒนาแอปพลิเคชัน PTT MIS for Life Station	65
5.3 ข้อเสนอแนะ	66
5.4 งานในอนาคต	66
เอกสารอ้างอิง	67
ภาคผนวก ก. การติดตั้ง RestKit Version 0.10x ใน Xcode 4.x	69
ภาคผนวก ข. การติดตั้ง Zend เฟรมเวิร์ค ไลบรารี	76
ภาคผนวก ค. รูปแบบข้อมูล JSON	81
ภาคผนวก ง. การพัฒนาแอปพลิเคชัน	84
ง.1 การพัฒนา REST เว็บเซอร์วิส	84
ง.1.1 การสร้างส่วนติดต่อกับไคลเอนต์	84
ง.1.2 การสร้างส่วนที่จัดการกับข้อมูล	95
ง.2 การพัฒนา iOS แอปพลิเคชัน	103
ง.2.1 การสร้างส่วนติดต่อผู้ใช้	103
ง.2.1.1 การสร้างส่วนติดต่อผู้ใช้จาก MainStoryboard.storyboard	104
ง.2.1.2 การสร้างส่วนติดต่อผู้ใช้จาก Implementation File	121
ง.2.2 การติดต่อกับ REST เว็บเซอร์วิส	137
ง.2.3 ฟังก์ชันการทำงานของแอปพลิเคชัน	146
ง.2.3.1 คำสั่งที่ใช้ในการเปลี่ยนหน้า	146
ง.2.3.2 คำสั่งที่ใช้ในการจัดการส่วนติดต่อผู้ใช้	149
ง.2.3.3 คำสั่งที่จัดการฟอร์มในการกรอกข้อมูล	152
ภาคผนวก จ. การติดตั้งแอปพลิเคชันลงในเครื่อง iDevice	157
ภาคผนวก ฉ. คู่มือการใช้งานแอปพลิเคชัน	164

สารบัญตาราง

	หน้า
ตารางที่ 3.1 ข้อมูลฐานข้อมูลใน list ของแต่ละเมนู	13
ตารางที่ 3.2 ข้อมูลฐานข้อมูลของแต่ละเมนู	14
ตารางที่ 3.3 ข้อมูลฐานข้อมูลที่ทำการเพิ่มในแต่ละเมนู	17
ตารางที่ 3.4 ข้อมูลฐานข้อมูลที่ทำการแก้ไขในแต่ละเมนู	19
ตารางที่ 3.5 ข้อมูลฐานข้อมูลที่ทำการลบในแต่ละเมนู	21



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 ตัวอย่างการเรียกใช้เว็บเซอร์วิสแบบ REST	8
รูปที่ 3.1 Use Case Diagram	10
รูปที่ 3.2 โครงสร้าง ER Diagram	11
รูปที่ 3.3 Activity Diagram การล็อกอิน	12
รูปที่ 3.4 Activity Diagram การเลือกเมนู	13
รูปที่ 3.5 Activity Diagram การดูข้อมูล	14
รูปที่ 3.6 Activity Diagram การเพิ่มข้อมูล	16
รูปที่ 3.7 Activity Diagram การแก้ไขข้อมูล	18
รูปที่ 3.8 Activity Diagram การลบข้อมูล	20
รูปที่ 3.9 Sequence Diagram การล็อกอิน	22
รูปที่ 3.10 Sequence Diagram การแสดงรายการข้อมูล	23
รูปที่ 3.11 Sequence Diagram การดูข้อมูล	23
รูปที่ 3.12 Sequence Diagram การเพิ่มข้อมูล	24
รูปที่ 3.13 Sequence Diagram การแก้ไขข้อมูล	25
รูปที่ 3.14 Sequence Diagram การลบข้อมูล	26
รูปที่ 3.15 หน้าจอล็อกอิน	27
รูปที่ 3.16 หน้าจอเมนูหลัก	28
รูปที่ 3.17 หน้าจอกรอกรายละเอียด Gas Station	28
รูปที่ 3.18 หน้าจอกรอกรายละเอียด Place Travel	29
รูปที่ 3.19 หน้าจอกรอกรายละเอียด Oil Type Setting	29
รูปที่ 3.20 หน้าจอกรอกรายละเอียด Oil Type Setting	30
รูปที่ 3.21 หน้าจอกรอกรายละเอียด Price Range Setting	30
รูปที่ 3.22 หน้าจอกรอกรายละเอียด Food Type Setting	31
รูปที่ 3.23 หน้าจอกรอกรายละเอียด Promotion	31
รูปที่ 3.24 หน้าจอกรอกรายละเอียด Other Application	32
รูปที่ 3.25 หน้าจอ Log Report	32
รูปที่ 3.26 diagram การพัฒนา REST เว็บเซอร์วิส	33
รูปที่ 3.27 diagram การพัฒนาส่วนที่ติดต่อกับไคลเอนต์	34
รูปที่ 3.28 diagram การพัฒนาส่วนที่จัดการกับฐานข้อมูล	35

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านอื่น

ไม่ว่ากรณีใดๆก็ตาม หากมีข้อผิดพลาดใดๆ กรุณาแจ้งเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 3.29 diagram การพัฒนา iOS แอปพลิเคชัน	36
รูปที่ 4.1 การตั้งค่าภายใน Setting	38
รูปที่ 4.2 การกำหนดค่า IP Address	38
รูปที่ 4.3 การล็อกอินสำเร็จ	39
รูปที่ 4.4 การล็อกอินไม่สำเร็จ	39
รูปที่ 4.5 หน้ารายการเมนู	40
รูปที่ 4.6 หน้ารายการข้อมูลของแต่ละเมนู	41
รูปที่ 4.7 หน้าแสดงข้อมูลของแต่ละเมนู	42
รูปที่ 4.8 หน้าการเพิ่มข้อมูลของแต่ละเมนู	43
รูปที่ 4.9 หน้าการแก้ไขข้อมูลของแต่ละเมนู	44
รูปที่ 4.10 หน้าการลบข้อมูลของแต่ละเมนู	45
รูปที่ 4.11 หน้ารายการเมนู	46
รูปที่ 4.12 หน้ารายการข้อมูล Place Travel	46
รูปที่ 4.13 การค้นหาข้อมูล	47
รูปที่ 4.14 หน้าแสดงข้อมูล	47
รูปที่ 4.15 หน้าแสดงข้อมูลโดยไม่มีข้อมูลรูป	48
รูปที่ 4.16 หน้า Add ข้อมูล	48
รูปที่ 4.17 การเพิ่มข้อมูล	49
รูปที่ 4.18 การกรองข้อมูล	49
รูปที่ 4.19 หน้าแสดงข้อมูล	50
รูปที่ 4.21 การเลือกรูปจากเครื่อง iPhone	51
รูปที่ 4.22 การ Crop รูปที่เลือกจากเครื่อง iPhone	51
รูปที่ 4.23 หน้า Edit ข้อมูล	52
รูปที่ 4.24 การแก้ไขข้อมูล Text Field	52
รูปที่ 4.25 การแก้ไขข้อมูล Picker	53
รูปที่ 4.26 การบันทึกข้อมูล	53
รูปที่ 4.27 การกรองข้อมูล	54
รูปที่ 4.28 หน้าแสดงข้อมูล	54
รูปที่ 4.29 หน้ารายการข้อมูล Place Travel	55

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 4.30 การเลือกข้อมูลที่จะลบ	55
รูปที่ 4.31 ปุ่ม Delete	56
รูปที่ 4.32 Action Sheet ปุ่ม Delete	56
รูปที่ 4.33 หน้าแสดงข้อมูล	57
รูปที่ 4.34 หน้ารายการข้อมูล Place Travel	57
รูปที่ 4.35 การเปลี่ยนหน้าโดยใช้ Tab Bar	58
รูปที่ 4.36 การลือกเอาท์	59
รูปที่ 4.37 หน้าเมนู Log Report	59
รูปที่ 4.38 Log Report LOGIN	60
รูปที่ 4.39 Log Report INSERT	60
รูปที่ 4.40 Log Report UPDATE	61
รูปที่ 4.41 Log Report DELETE	61
รูปที่ ก-1 การติดตั้ง RestKit ใน Project	69
รูปที่ ก-2 การตั้งค่าภายใน Other Linker Flags	70
รูปที่ ก-3 การตั้งค่าภายใน Header Search Paths	71
รูปที่ ก-4 การตั้งค่าภายใน Target Dependencies	72
รูปที่ ก-5 การตั้งค่าภายใน Target Dependencies	73
รูปที่ ก-6 การตั้งค่าภายใน Link Binary With Libraries	74
รูปที่ ก-7 การตั้งค่าภายใน Link Binary With Libraries	75
รูปที่ ข-1 System Properties	76
รูปที่ ข-2 Environment Variables	76
รูปที่ ข-3 Edit User Variable	77
รูปที่ ข-4 Edit User Variable	77
รูปที่ ข-5 การสร้าง Project	77
รูปที่ ข-6 Project ที่สร้างจาก Zend เฟรมเวิร์ค	78
รูปที่ ข-7 คำสั่งของ Zend เฟรมเวิร์ค	78
รูปที่ ข-8 โครงสร้างของ Project ซ้ำงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้าน	79
รูปที่ ข-9 ไฟล์ httpd-vhosts.conf เนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้	80
รูปที่ ข-10 หน้า Index ของ Zend เฟรมเวิร์ค	80

สารบัญรูป (ต่อ)

	หน้า
รูปที่ ค-1 รูปแบบ Object ของ JSON	81
รูปที่ ค-2 รูปแบบ Array ของ JSON	81
รูปที่ ค-3 รูปแบบ Value ของ JSON	82
รูปที่ ค-4 รูปแบบ Number ของ JSON	82
รูปที่ ค-5 รูปแบบ String ของ JSON	83
รูปที่ ง-1 ตัวอย่าง Controller.php ส่วน method __construct	85
รูปที่ ง-2 ตัวอย่าง Controller.php ส่วนฟังก์ชันที่รองรับ HTTP ฟังก์ชัน	86
รูปที่ ง-3 ตัวอย่าง Controller.php ส่วนฟังก์ชันที่จัดการกับ response และ db adapter	87
รูปที่ ง-4 ตัวอย่าง Response.php ส่วนฟังก์ชันที่จัดการกับ accept header และ type	88
รูปที่ ง-5 ตัวอย่าง Response.php ส่วนฟังก์ชันที่จัดการกับ code และ body	89
รูปที่ ง-6 ตัวอย่าง Response.php ส่วนฟังก์ชันที่กำหนด content type และดึงข้อมูล	90
รูปที่ ง-7 ตัวอย่าง PlacetravelController.php	91
รูปที่ ง-8 ตัวอย่าง PlacetravelController.php ส่วนฟังก์ชัน indexAction	91
รูปที่ ง-9 ตัวอย่าง PlacetravelController.php ส่วนฟังก์ชัน getAction	92
รูปที่ ง-10 ตัวอย่าง PlacetravelController.php ส่วนฟังก์ชัน postAction()	92
รูปที่ ง-11 ตัวอย่าง PlacetravelController.php ส่วนฟังก์ชัน putAction	93
รูปที่ ง-12 ตัวอย่าง PlacetravelController.php ส่วนฟังก์ชัน deleteAction	94
รูปที่ ง-13 ตัวอย่าง Auth.php	96
รูปที่ ง-14 ตัวอย่าง Auth.php ส่วนฟังก์ชัน convertToArray	97
รูปที่ ง-15 ตัวอย่าง Placetravel.php	97
รูปที่ ง-16 ตัวอย่าง Placetravel.php ส่วนฟังก์ชัน getPlaceTravels	98
รูปที่ ง-17 ตัวอย่าง Placetravel.php ส่วนฟังก์ชัน getPlaceTravel	98
รูปที่ ง-18 ตัวอย่าง Placetravel.php ส่วนฟังก์ชัน createPlaceTravel	98
รูปที่ ง-19 ตัวอย่าง Placetravel.php ส่วนฟังก์ชัน editPlaceTravel	99
รูปที่ ง-20 ตัวอย่าง Placetravel.php ส่วนฟังก์ชัน deletePlaceTravel	99
รูปที่ ง-21 ตัวอย่าง Upload.php	100
รูปที่ ง-22 ตัวอย่าง Data.php การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้าน	101
รูปที่ ง-23 ตัวอย่าง Data.php ส่วนฟังก์ชัน logInsert	101
รูปที่ ง-24 ตัวอย่าง Data.php ส่วนฟังก์ชัน logUpdate	102

สารบัญรูป (ต่อ)

	หน้า
รูปที่ ง-25 ตัวอย่าง Data.php ส่วนฟังก์ชัน logDelete	102
รูปที่ ง-26 ตัวอย่าง Data.php ส่วนฟังก์ชัน logLogin และ logLogout	103
รูปที่ ง-27 ภาพรวมการทำงานของแอปพลิเคชัน	104
รูปที่ ง-28 หน้าลืออกอิน	104
รูปที่ ง-29 การกำหนดคลาส View Controller	105
รูปที่ ง-30 การใส่ Image View	105
รูปที่ ง-31 การกำหนดค่าให้กับ Image View และไฟล์เดอร์ที่ใช้ในการเก็บรูป	106
รูปที่ ง-32 หน้าลืออกอิน	106
รูปที่ ง-33 การสร้าง Text Field	107
รูปที่ ง-34 การเชื่อม Text Field เข้ากับตัวแปร	107
รูปที่ ง-35 การ Synthesize ตัวแปร	108
รูปที่ ง-36 การสร้าง Text View	108
รูปที่ ง-37 การเชื่อม Text View เข้ากับตัวแปร	109
รูปที่ ง-38 หน้าลืออกอิน	109
รูปที่ ง-39 การสร้าง Label	110
รูปที่ ง-40 การกำหนดค่าของ Label	110
รูปที่ ง-41 หน้าลืออกอิน	111
รูปที่ ง-42 การสร้าง Button	111
รูปที่ ง-43 การเชื่อม Button เข้ากับตัวแปร	112
รูปที่ ง-44 ฟังก์ชัน login	112
รูปที่ ง-45 การสร้าง Table View Controller	113
รูปที่ ง-46 Table View Controller	113
รูปที่ ง-47 หน้า Table View Controller กับหน้า PlaceTravelViewController	114
รูปที่ ง-48 การเชื่อมหน้า Table View Controller กับหน้า PlaceTravelViewController	114
รูปที่ ง-49 การกำหนดชื่อ Segue	114
รูปที่ ง-50 การสร้าง Navigation Controller	115
รูปที่ ง-51 การทำงานของ Navigation Controller	116
รูปที่ ง-52 การสร้าง Search Bar	117
รูปที่ ง-53 หน้า Table View Controller กับ Search Bar	117

สารบัญรูป (ต่อ)

	หน้า
รูปที่ ง-54 หน้า PlaceTravelViewController	118
รูปที่ ง-55 การเชื่อม Image View เข้ากับตัวแปร	118
รูปที่ ง-56 การสร้าง Switch	119
รูปที่ ง-57 การเชื่อม Switch เข้ากับตัวแปร	119
รูปที่ ง-58 การสร้าง Scroll View	120
รูปที่ ง-59 การเชื่อม Scroll View เข้ากับตัวแปร	120
รูปที่ ง-60 การกำหนดขนาดของ Scroll View	120
รูปที่ ง-61 ฟังก์ชัน create_tabbar	121
รูปที่ ง-62 ฟังก์ชัน tabBar	122
รูปที่ ง-63 tabBar	122
รูปที่ ง-64 ฟังก์ชัน createButton	122
รูปที่ ง-65 ฟังก์ชัน goForward	123
รูปที่ ง-66 การ Push หน้า PlaceTravelAddController	123
รูปที่ ง-67 ฟังก์ชัน createButton	123
รูปที่ ง-68 หน้า PlaceTravelViewController	124
รูปที่ ง-69 ฟังก์ชัน edit	124
รูปที่ ง-70 การ Push หน้า PlaceTravelEditController	124
รูปที่ ง-71 Activity Indicator	125
รูปที่ ง-72 การกำหนดค่า Activity Indicator	125
รูปที่ ง-73 การเรียกใช้งาน Activity Indicator	125
รูปที่ ง-74 ฟังก์ชัน Show Action Sheet	126
รูปที่ ง-75 การแสดง Action Sheet	126
รูปที่ ง-76 การประกาศตัวแปรในการสร้าง Label	127
รูปที่ ง-77 การ Initialize Label	127
รูปที่ ง-78 การกำหนดค่าภายใน Label	128
รูปที่ ง-79 คีย์บอร์ดปกติ	129
รูปที่ ง-80 ฟังก์ชัน textFieldDidBeginEditing	129
รูปที่ ง-81 คีย์บอร์ดมีปุ่ม Done	129
รูปที่ ง-82 การกำหนดค่าให้กับ Text View	130

สารบัญรูป (ต่อ)

	หน้า
รูปที่ ง-83 คีย์บอร์ดมีปุ่ม Done ของ Text View	130
รูปที่ ง-84 Date Picker	131
รูปที่ ง-85 ฟังก์ชัน textFieldShouldBeginEditing	131
รูปที่ ง-86 Date Picker ที่มี Toolbar	132
รูปที่ ง-87 ฟังก์ชันเมื่อกดปุ่ม done หรือ cancel	132
รูปที่ ง-88 โค้ดภายในฟังก์ชัน done	132
รูปที่ ง-89 การแสดงผลการเลือกข้อมูลจาก Date Picker	133
รูปที่ ง-90 Picker View	133
รูปที่ ง-91 การสร้าง Picker View	134
รูปที่ ง-92 ฟังก์ชัน numberOfComponentsInPickerView	134
รูปที่ ง-93 ฟังก์ชันการสร้าง Row ใน Picker View	134
รูปที่ ง-94 ฟังก์ชันในการสร้างชื่อ Title ใน Picker View	135
รูปที่ ง-95 ฟังก์ชันในการเลือก Row ใน Picker View	135
รูปที่ ง-96 ฟังก์ชันเมื่อกดปุ่ม done หรือ cancel	135
รูปที่ ง-97 การแสดงผลจากการเลือกข้อมูลใน Picker View	136
รูปที่ ง-98 ฟังก์ชัน done ของ Picker View	136
รูปที่ ง-99 การสร้าง Settings Bundle	136
รูปที่ ง-100 ไฟล์ Root.plist	137
รูปที่ ง-101 การกำหนดค่าภายในไฟล์ Root.plist	137
รูปที่ ง-102 การแสดงผลของ Settings Bundle	137
รูปที่ ง-103 ไฟล์ AppDelegate.h	138
รูปที่ ง-104 ไฟล์ AppDelegate.m	139
รูปที่ ง-105 ฟังก์ชัน requestData	140
รูปที่ ง-106 การสร้าง Activity Indicator	140
รูปที่ ง-107 การลบ Activity Indicator	140
รูปที่ ง-108 การทำงานเมื่อได้รับข้อมูลครบถ้วน	141
รูปที่ ง-109 ฟังก์ชัน requestData ซึ่งงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านอื่น	141
รูปที่ ง-110 การ import RestKit ลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้	141
รูปที่ ง-111 ฟังก์ชัน login	142

สารบัญรูป (ต่อ)

หน้า

รูปที่ ง-112 ข้อผิดพลาดในการติดต่อ REST เว็บเซอร์วิส	142
รูปที่ ง-113 การทำงานเมื่อได้รับข้อมูลครบถ้วน	142
รูปที่ ง-114 ฟังก์ชัน checkPassword	143
รูปที่ ง-115 การ Add ข้อมูล	144
รูปที่ ง-116 การ Edit ข้อมูล	145
รูปที่ ง-117 การ Delete ข้อมูล	145
รูปที่ ง-118 การกดปุ่มในหน้าจอ	146
รูปที่ ง-119 การเชื่อมปุ่มเข้ากับฟังก์ชันการทำงาน	146
รูปที่ ง-120 ฟังก์ชันการทำงานของปุ่มในหน้าจอ Menu	147
รูปที่ ง-121 การส่งข้อมูลผ่าน Segue	148
รูปที่ ง-122 การรับข้อมูลจาก Segue	148
รูปที่ ง-123 การกำหนดจำนวน Section	149
รูปที่ ง-124 การกำหนด Row ใน Section	149
รูปที่ ง-125 การแสดงข้อมูลใน Cell	150
รูปที่ ง-126 การแสดงผลของ Cell ใน Table View	150
รูปที่ ง-127 ฟังก์ชันในการค้นหาข้อมูล	151
รูปที่ ง-128 ตัวอย่างการค้นหาข้อมูล	151
รูปที่ ง-129 การกรองข้อมูล	152
รูปที่ ง-130 ตัวอย่างการกรองข้อมูล	153
รูปที่ ง-131 การนำเข้ารูปจากเครื่อง iPhone	153
รูปที่ ง-132 ตัวอย่างการนำเข้ารูปจากเครื่อง iPhone	154
รูปที่ ง-133 การสร้างรูปลงใน Image View	154
รูปที่ ง-134 ตัวอย่างการสร้างรูปลงใน Image View	155
รูปที่ ง-135 การประกาศค่าคงที่	155
รูปที่ ง-136 ฟังก์ชันในการเลื่อนหน้าจอขึ้น	156
รูปที่ ง-137 ฟังก์ชันในการเลื่อนหน้าจอลง	156
รูปที่ ง-138 การเรียกใช้งานฟังก์ชันการเลื่อนหน้าจอ	156
รูปที่ จ-1 การสร้าง Developer Certificate ขั้นตอนที่ 1	157
รูปที่ จ-2 การสร้าง Developer Certificate ขั้นตอนที่ 2	158

สารบัญรูป (ต่อ)

	หน้า
รูปที่ จ-3 การสร้าง Developer Certificate ขั้นตอนที่ 3.1	158
รูปที่ จ-4 การสร้าง Developer Certificate ขั้นตอนที่ 3.2	159
รูปที่ จ-5 การสร้าง Developer Certificate ขั้นตอนที่ 3.3	159
รูปที่ จ-6 การร้องขอ Developer Certificate	161
รูปที่ จ-7 การลงทำเบียนเครื่องที่จะทำการติดตั้ง	162
รูปที่ จ-8 การลงทะเบียนแอปพลิเคชัน	163
รูปที่ จ-9 การทดลองใช้แอปพลิเคชันบนตัวเครื่อง	163
รูปที่ ฉ-1 การตั้งค่าภายใน Setting	164
รูปที่ ฉ-2 การกำหนดค่า IP Address	164
รูปที่ ฉ-3 การใส่ข้อมูลหน้าล็อกอิน	165
รูปที่ ฉ-4 การล็อกอิน	165
รูปที่ ฉ-5 การเข้าใช้งานในแต่ละเมนู	166
รูปที่ ฉ-6 การล็อกเอาท์	167
รูปที่ ฉ-7 เมนู Place Travel	167
รูปที่ ฉ-8 การเลือกข้อมูลในเมนู Place Travel	168
รูปที่ ฉ-9 หน้าแสดงข้อมูลโดยไม่มีข้อมูลรูป	169
รูปที่ ฉ-10 การเลือกรูปจากเครื่อง iPhone	169
รูปที่ ฉ-11 การ Crop รูปที่เลือกจากเครื่อง iPhone	170
รูปที่ ฉ-12 การแก้ไขข้อมูล	170
รูปที่ ฉ-13 ข้อมูลของเมนู Place Travel	171
รูปที่ ฉ-14 การแก้ไขข้อมูลจังหวัด	172
รูปที่ ฉ-15 การแก้ไขข้อมูลสถานะบริการ	173
รูปที่ ฉ-16 การแก้ไขข้อมูลสถานะบริการ	173
รูปที่ ฉ-17 การบันทึกข้อมูล	174
รูปที่ ฉ.18 ข้อมูลถูกแก้ไขแล้ว	174
รูปที่ ฉ-19 หน้า Add ข้อมูล	175
เอกสารนี้รูปที่ ฉ-20 การเพิ่มข้อมูลับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้าน	175
ไม่ว่ากรณีรูปที่ ฉ-21 หน้าแสดงข้อมูลแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้	176
รูปที่ ฉ-22 การเลือกข้อมูลที่จะลบ	176

สารบัญรูป (ต่อ)

	หน้า
รูปที่ ฉ.23 การลบข้อมูล	177
รูปที่ ฉ.24 เมนู Gas Station	177
รูปที่ ฉ.25 ข้อมูลภายในเมนู Gas Station	178
รูปที่ ฉ.26 เมนู Oil Type	179
รูปที่ ฉ.27 ข้อมูลภายใน Oil Type	180
รูปที่ ฉ.28 เมนู Promotion	181
รูปที่ ฉ.29 ข้อมูลภายในเมนู Promotion	181
รูปที่ ฉ.30 เมนู Travel Type	182
รูปที่ ฉ.31 เมนู Price Range	183
รูปที่ ฉ.32 ข้อมูลภายในเมนู Price Range	183
รูปที่ ฉ.33 เมนู Travel Eat Type	184
รูปที่ ฉ.34 ข้อมูลภายในเมนู Travel Eat Type	184
รูปที่ ฉ.35 เมนู Other Application	185
รูปที่ ฉ.36 ข้อมูลภายในเมนู Other Application	185
รูปที่ ฉ.37 หน้าเมนู Log Report	186
รูปที่ ฉ.38 Log Report LOGIN	186
รูปที่ ฉ.39 Log Report INSERT	187
รูปที่ ฉ.40 Log Report UPDATE	187
รูปที่ ฉ.41 Log Report DELETE	188

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ปัจจุบัน เทคโนโลยีที่ใช้ในการผลิตและใช้งาน อุปกรณ์โมบายทั้งซอฟต์แวร์และฮาร์ดแวร์ มีความก้าวหน้าเป็นอย่างมาก ส่งผลให้ผู้ใช้สามารถใช้งานมัลติมีเดีย และ อินเทอร์เน็ต ผ่านหน้าจอสัมผัส ได้ง่าย สะดวก รวดเร็ว ทำให้การใช้งานอุปกรณ์โมบาย ไม่ว่าจะเป็นสมาร์ทโฟน หรือ แท็บเล็ต ไม่ได้ถูกจำกัดให้เป็นเพียงเพื่อความบันเทิง หรือ ประโยชน์ส่วนบุคคลอีกต่อไป องค์กร บริษัทที่มีความก้าวหน้าทางเทคโนโลยี ได้มีการนำอุปกรณ์โมบายเหล่านี้มาใช้ในการทำงานมากขึ้น บางแห่งอนุญาตให้พนักงานสามารถใช้งานระบบได้ผ่านทางอุปกรณ์โมบายของพนักงานเอง (Bring your own device – BYOD) ในจำนวนอุปกรณ์โมบายทั้งหมดนั้น iPhone จัดว่าเป็นอุปกรณ์ที่ได้รับความนิยมอย่างสูง เพราะนอกจาก ใช้งานง่าย มีหน้าจอที่สวยงาม และมีประสิทธิภาพที่สูงแล้ว กระบวนการในการนำแอปพลิเคชันไปใช้งานของอุปกรณ์ iOS ซึ่งได้แก่ iPhone iPod และ iPad นั้น มีความปลอดภัยและได้รับความน่าเชื่อถือ มากกว่า อุปกรณ์โมบายอื่นๆ

ด้วยเหตุนี้ทางบริษัท PTT ICT Solutions จึงได้มีแนวคิดที่จะทดลองใช้ iPhone มาช่วยในการทำงานของเจ้าหน้าที่ภายในองค์กร จากที่แต่เดิมต้องทำงานผ่านทางเครื่องคอมพิวเตอร์เท่านั้น โดยทางบริษัทได้เลือกให้ทำการพัฒนา แอปพลิเคชัน ในส่วนของการจัดการฐานข้อมูล weloveptt ซึ่งจะเก็บข้อมูลที่ใช้ทั้งหมดของ แอปพลิเคชัน PTT Life Station¹ ซึ่งในปัจจุบัน ผู้ดูแลระบบแบคเอนด์ของ PTT Life station สามารถทำการเพิ่ม แก้ไข หรือ เปลี่ยนแปลงข้อมูลได้จะต้องทำบนเว็บ แอปพลิเคชันผ่านทางเว็บเบราว์เซอร์เท่านั้น ซึ่งแม้จะสามารถเปิดเบราว์เซอร์จาก iPhone ได้ แต่การแสดงผลผ่านหน้าจอดูได้ลำบาก และ การใช้งานไม่สะดวก จึงต้องมีการพัฒนาแอปพลิเคชัน ที่จะช่วยให้ผู้ดูแลระบบสามารถจัดการข้อมูลผ่าน iPhone ได้โดยตรง

¹แอปพลิเคชันบน iPhone ของบริษัท PTT ICT Solutions ซึ่งช่วยให้ผู้ใช้ อุปกรณ์ iOS เช่น iPhone iPad และ iPod ทั่วประเทศไทย สามารถอัปเดตราคาน้ำมัน ค้นหาข้อมูลสถานีบริการน้ำมัน และสถานีบริการ NGV ของทางบริษัท ปตท. ได้ทุกที่ ทุกเวลา นอกจากข้อมูลเรื่องสถานีบริการน้ำมันแล้ว PTT Life Station ยังมีบริการแนะนำอื่นๆ เช่น แนะนำสถานที่ท่องเที่ยว และร้านอาหารต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นวัตถุประสงค์หลักของปัญหาพิเศษนี้ คือ การพัฒนาแอปพลิเคชัน PTT MIS for Life Station บน iPhone เพื่อให้ผู้ดูแลระบบในส่วนของการจัดการข้อมูล ของ แอปพลิเคชัน PTT Life Station สามารถ เพิ่มเติม แก้ไข ลบ และ เปลี่ยนแปลงข้อมูลต่างๆ ทั้งหมดที่ PTT Life Station ต้องใช้ได้ ผ่านทาง iPhone ซึ่งสามารถใช้งานได้ทุกที่ทุกเวลาไม่จำกัดอยู่แต่เพียงภายในอินทราเน็ตของบริษัท โดยจะต้องมีหน้าจอที่ใช้งานง่าย และ สวยงาม อีกทั้งจะต้องมีส่วนที่ใช้ในการบันทึกการใช้งานของผู้ดูแล (logging) ระบบเพื่อให้สามารถเรียกดูได้อีกด้วย อนึ่ง ตัวอย่างของข้อมูลที่ใช้ได้แก่ ข้อมูลเกี่ยวกับสถานีน้ำมัน ข้อมูลเกี่ยวกับสถานีแก๊ส NGV ชนิดของการบริการ สถานที่ท่องเที่ยว ประเภทของการท่องเที่ยว เป็นต้น

ในส่วนของการพัฒนาระบบนั้น จะแบ่งการพัฒนาออกเป็นสองส่วนคือ ส่วนของเว็บเซอร์วิส และ ส่วนของแอปพลิเคชันบน iPhone โดยในส่วนของเว็บเซอร์วิสนั้นจะใช้แบบ REST เนื่องจาก ได้รับความนิยมนในการใช้สร้าง มากกว่าแบบ SOAP ด้วยความง่ายต่อการสร้าง และมีการใช้งานโปรโตคอล HTTP อย่างมีประสิทธิภาพ นอกจากนี้ REST เว็บเซอร์วิสยังสามารถรองรับข้อมูลในการรับ-ส่งได้มากกว่าเว็บเซอร์วิสแบบ SOAP

1.2 วัตถุประสงค์

- 1.2.1 เพื่อให้ผู้ดูแลระบบสามารถจัดการข้อมูลผ่านส่วนติดต่อผู้ใช้ที่ใช้งานง่าย
- 1.2.2 เพื่อให้ผู้ดูแลระบบสามารถใช้งานแอปพลิเคชันที่มีการแสดงพอดีกับจอ iPhone
- 1.2.3 สามารถใช้แอปพลิเคชันภายนอกองค์กรได้
- 1.2.4 เพื่อศึกษาการทำงาน REST เว็บเซอร์วิสที่มีแนวโน้มในการใช้งานขององค์กรต่างๆ

1.3 ขอบเขตของปัญหา

- 1.3.1 ใช้แอปพลิเคชัน PTT MIS for Life Station ทำหน้าที่จัดการข้อมูลในฐานะข้อมูล โดยสามารถดู เพิ่ม แก้ไข และลบข้อมูลได้
- 1.3.2 ใช้ REST เว็บเซอร์วิสทำหน้าที่เป็นตัวรับส่งข้อมูลระหว่างฐานข้อมูลและแอปพลิเคชัน PTT MIS for Life Station
- 1.3.3 แอปพลิเคชันที่เป็นไคลเอนต์จะประกอบด้วย 9 เมนู โดยแบ่งเป็นเมนูที่ใช้จัดการข้อมูล 8 เมนู และเมนูที่ใช้ตรวจสอบการจัดการข้อมูล 1 เมนู
- 1.3.4 เมนูที่ใช้จัดการข้อมูลทั้ง 8 เมนูจะสามารถทำการเรียก, เพิ่ม, แก้ไข และลบ ข้อมูลใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ประโยชน์ที่จะได้รับ

- 1.4.1 ระบบมีส่วนติดต่อกับผู้ใช้ที่ใช้งานง่าย
- 1.4.2 ผู้ดูแลระบบได้ใช้ iPhone แอปพลิเคชัน ที่ดูได้ง่ายสบายตา
- 1.4.3 ประหยัดเวลาของผู้ดูแลระบบเนื่องจากสามารถใช้แอปพลิเคชันภายนอกองค์กรได้
- 1.4.4 ผู้พัฒนาได้เรียนรู้ขั้นตอนการพัฒนา iPhone แอปพลิเคชัน
- 1.4.5 ผู้พัฒนาได้เรียนรู้การออกแบบและการพัฒนา REST เว็บเซอร์วิส

1.5 ขั้นตอนการดำเนินงาน

- 1.5.1 ศึกษาภาษา Objective-C และเครื่องมือที่ใช้พัฒนา iPhone แอปพลิเคชัน
- 1.5.1 ศึกษาภาษา Objective-C และเครื่องมือที่ใช้พัฒนา iPhone แอปพลิเคชัน
- 1.5.2 ศึกษาการออกแบบและขั้นตอนการพัฒนา REST เว็บเซอร์วิส
- 1.5.3 ศึกษาข้อมูลที่จะทำการจัดการในฐานข้อมูล
- 1.5.4 ศึกษาข้อมูลรูปแบบ JSON
- 1.5.5 ออกแบบส่วนติดต่อกับผู้ใช้
- 1.5.6 พัฒนา REST เว็บเซอร์วิสเพื่อทำหน้าที่เป็นตัวรับส่งข้อมูลระหว่างฐานข้อมูลกับแอปพลิเคชัน
- 1.5.7 พัฒนา iPhone แอปพลิเคชันเพื่อจัดการข้อมูลในฐานข้อมูล
- 1.5.8 ทดสอบความถูกต้องของข้อมูลที่รับ ส่งจากฐานข้อมูลและส่งข้อมูลไปยัง โบบายแอปพลิเคชันผ่านเว็บเซอร์วิส
- 1.5.9 ทดสอบแอปพลิเคชันและแก้ไขข้อผิดพลาด
- 1.5.10 จัดทำเอกสาร

1.6 อุปกรณ์ที่ใช้ในการทำปัญหาพิเศษ

1.6.1 ฮาร์ดแวร์ :

- iMac ระบบปฏิบัติการ Mac OS X Mountain Lion ขึ้นไป
- iPhone 4 ระบบปฏิบัติการ iOS 5.0 ขึ้นไป

1.6.2 ซอฟต์แวร์ :

- Apache 2.2.14

- PHP 5.3.6

- MySQL 5.1.41

- phpMyAdmin 3.3.2deb1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีโทษตามกฎหมายสงวนลิขสิทธิ์ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Windows 7
- OS X Lion 10.7.4
- Xcode 4.5.2
- Eclipse Galileo
- Zend library 1.12.0



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 Mobile Application [1]

Mobile Application คือการพัฒนาระบบแอปพลิเคชันต่างๆ รวมถึงมัลติมีเดีย บนอุปกรณ์เคลื่อนที่ เช่น iPad และ Phone เพื่อเพิ่มประสิทธิภาพและความประทับใจในการนำเสนอข้อมูล รวมถึงการพัฒนาระบบใช้งานสำหรับองค์กร เช่น ระบบงานขาย ระบบตรวจสอบคลังสินค้า ระบบจองห้องพัก ระบบนำเสนอสำหรับอสังหาริมทรัพย์ ระบบ E-learning และ ระบบค้นหาแผนที่ เป็นต้น ในส่วนของผู้บริหาร สามารถดูรายงานสรุปการขาย รายงานความก้าวหน้าของโครงการ ยอดสินค้าคงเหลือ และรายงานอื่นๆตามต้องการ เป็นต้น

2.2 Bring Your Own Device [2]

Bring Your Own Device (BYOD) หมายถึงแนวโน้มที่พนักงานนำอุปกรณ์พกพาของตัวเองมาทำงาน และใช้อุปกรณ์ดังกล่าวเพื่อใช้ทำงานทรัพยากรหรือระบบที่มีการควบคุมการเข้าถึงของบริษัท เช่น อีเมล, ไฟล์เซิร์ฟเวอร์ และฐานข้อมูล บางคนใช้คำว่า Bring Your Own Technology (BYOT) เนื่องจากมันมีความหมายกว้างกว่าไม่จำกัดอยู่เฉพาะฮาร์ดแวร์เท่านั้น แต่ยังรวมถึงซอฟต์แวร์บนอุปกรณ์ด้วย เช่น เว็บเบราว์เซอร์, โปรแกรมเล่นมัลติมีเดีย และแอนตี้ไวรัส โดย BYOD มีความสำคัญมากขึ้นเรื่อยๆ ในโลกธุรกิจที่ประมาณ 90% ของพนักงานใช้อุปกรณ์ของตัวเองอยู่แล้วในที่ทำงานแม้จะใช้อย่างจำกัดก็ตาม บางคนเชื่อว่า BYOD อาจช่วยให้พนักงานมีประสิทธิภาพมากขึ้น

2.3 Xcode [3]

Xcode เป็นเครื่องมือของนักพัฒนาที่เอาไว้สร้างแอปพลิเคชันสำหรับเครื่องบนระบบปฏิบัติการ OS X และ iOS เทียบได้กับ Visual Studio ของฝั่งไมโครซอฟท์ ในส่วน Xcode เวอร์ชันล่าสุด ณ เวลาที่เขียนนี้ คือเวอร์ชัน 4.5.2 ต้องติดตั้งลงบนเครื่องที่มีระบบปฏิบัติการ OS X เวอร์ชัน 10.6 ขึ้นไป สามารถหาซื้อได้จาก Mac App Store ในราคา \$4.99 เหรียญสหรัฐ แต่ถ้าจะสมัครเป็นสมาชิกของ iOS หรือ Mac Developer Program สามารถดาวน์โหลดได้ฟรีโดย Xcode 4.5.2 ประกอบด้วย Xcode IDE Instruments และ SDK สำหรับ Mac OSX และ iOS

- Xcode IDE คือ สภาพแวดล้อมในการพัฒนาแอปพลิเคชัน ประกอบด้วยพื้นที่ทำงานสำหรับเขียนโค้ด พื้นที่สำหรับออกแบบหน้าจอส่วนติดต่อผู้ใช้เรียกว่า Interface builder มีคอมไพเลอร์ที่มีประสิทธิภาพสูงชื่อว่า LLVM Compiler 2.0 สามารถคอมไพล์โค้ดได้เร็วกว่า GCC

ถึง 2 เท่าทำให้สร้างแอปพลิเคชันให้ทำงานได้เร็วขึ้น มีระบบตรวจสอบโค้ดที่มีประสิทธิภาพแก้ไขโค้ดที่ผิดให้อัตโนมัติ มี Debugger Engine ที่ชื่อว่า LLDB สามารถดักจับโค้ดได้ขณะที่โปรแกรมกำลังรันทดสอบอยู่

- Instruments เป็นเครื่องมือที่ใช้ตรวจสอบประสิทธิภาพของแอปพลิเคชันที่พัฒนาเสร็จแล้ว โดยวัดการใช้งาน Memory CPU Time และ Overhead ต่างๆ

- SDK สำหรับ Mac OS X และ iOS คือ คลาส Library สำหรับใช้พัฒนาแอปพลิเคชัน แบ่งเป็น Mac OS X และ iOS

2.4 ภาษาที่ใช้ในการพัฒนา

2.4.1 Objective-C [4]

Objective-C เป็นภาษาโปรแกรมเชิงวัตถุ พัฒนาขึ้นจากภาษาซีโดยยังคงคุณลักษณะของภาษาซีไว้ครบทุกประการเพียงแต่เพิ่มระบบส่งข้อความ (messaging) แบบเดียวกับภาษาสมอลล์ทอล์ก ปัจจุบันภาษา Objective-C ถูกใช้มากใน Cocoa (API) Mac OS X GNUstep (API) และ Cocotron (API) เป็นต้น ซึ่งระบบเหล่านี้ได้รับการพัฒนาขึ้นโดยมีพื้นฐานมาจากมาตรฐาน OpenStep (API) ใน Nextstep (Operating system) ใช้ภาษา Objective-C เป็นภาษาหลัก

2.4.2 PHP [5]

PHP เป็นภาษาสคริปต์ที่ประมวลผลที่ฝั่งเซิร์ฟเวอร์ แล้วส่งผลลัพธ์ไปแสดงผลที่ฝั่งไคลเอนต์ผ่านบราวเซอร์เช่นเดียวกับ CGI และ ASP ซึ่ง PHP สนับสนุนระบบปฏิบัติการ Windows Linux และเว็บเซิร์ฟเวอร์ เช่น IIS PWS Apache OmniHTTPd และสนับสนุนระบบฐานข้อมูลได้หลายรูปแบบเช่น SQL Server MySQL mSQL Oracle Informix และ ODBC

ความสามารถพื้นฐานของ PHP เป็นภาษาสคริปต์ทั่วไปที่มีความสามารถในการรับข้อมูลจากฟอร์ม การสร้าง Content ในลักษณะ Dynamic รับส่ง Cookies สร้าง เปิด อ่าน และปิดไฟล์ในระบบ การรองรับระบบจัดการฐานข้อมูล และยังมีความสามารถในการรองรับโปรโตคอลหลายแบบทั้ง IMAP SNMP NNTP POP3 และ HTTP โดยมีไลบรารีสำหรับติดต่อกับแอปพลิเคชันได้มากมาย มีความยืดหยุ่นสูงสามารถนำไปสร้างแอปพลิเคชันได้หลากหลาย และอีกข้อดีหนึ่งที่โดดเด่นของ PHP ก็คือสามารถแทรกลงในแท็ก HTML ในตำแหน่งใดก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 เซิร์ฟเวอร์ที่ใช้พัฒนา

2.5.1 Appserv [6]

Appserv คือชุดโปรแกรมในลักษณะของ WAMP ในการสร้างเว็บเซิร์ฟเวอร์สำเร็จรูป (จำลองเซิร์ฟเวอร์) บนระบบปฏิบัติการ Microsoft Windows ซึ่งสร้างโดย คุณภาพงศ์ ปัญญาดี เป็นการรวมโปรแกรมจำนวน 4 ตัวในการสร้างเซิร์ฟเวอร์ ได้แก่ Apache HTTP Server PHP MySQL และ phpMyAdmin โดยมีเวอร์ชันปัจจุบันได้แก่ 2.4.9 (สำหรับ PHP4) และ 2.5.10 (สำหรับ PHP5) โดย AppServ ได้ให้ความสำคัญว่าทุกสิ่งทุกอย่างจะต้องให้เหมือนกับต้นฉบับ จึงไม่ได้ตัดทอนหรือเพิ่มเติมอะไรที่แปลกไปกว่า Official Release แต่อย่างไร เพียงแต่มีบางส่วนเท่านั้นที่ได้เพิ่มประสิทธิภาพการติดตั้งให้สอดคล้องกับการทำงานแต่ละคน โดยที่การเพิ่มประสิทธิภาพนี้ไม่ได้ไปยุ่งเกี่ยวในส่วนของ แพคเกจเดิมแต่เป็นการกำหนดค่า Config เท่านั้น เช่น Apache ก็จะไปอยู่ในส่วนของ httpd.conf, PHP จะอยู่ในส่วนของ php.ini และ MySQL จะเป็นในส่วน of my.ini ดังนั้นโปรแกรม AppServ จะสามารถทำงานและมีความเสถียรของระบบ ได้เหมือนกับ Official Release ทั้งหมด

ในการกำหนดค่า Config เพื่อให้การติดตั้งโปรแกรมต่างๆ ที่ได้กล่าวมาในข้างต้น เพื่อลดขั้นตอนการติดตั้งที่ยุ่งยากและใช้เวลานาน โดยผู้ใช้งานเพียงดับเบิลคลิก Setup ภายในเวลา 1 นาที ทุกอย่างก็จะติดตั้งเสร็จสมบูรณ์พร้อมที่จะทำงานได้ทันทีทั้งเว็บเซิร์ฟเวอร์ และเซิร์ฟเวอร์ฐานข้อมูลเหตุผลนี้จึงเป็นเหตุผลหลักที่หลายๆ คนทั่วโลก ได้เลือกใช้โปรแกรม AppServ แทนการที่จะต้องมาติดตั้งโปรแกรมต่างๆ ที่ละส่วน

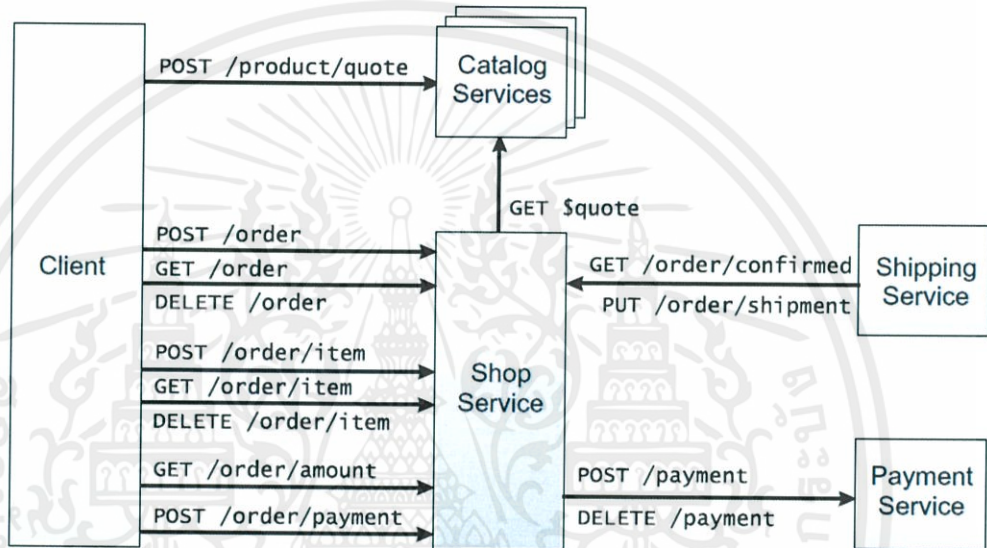
AppServ แบ่งออกเป็น 2 เวอร์ชันคือ

- 2.5.x คือเวอร์ชันที่นำ Package ใหม่ๆ นำมาใช้งาน โดยเฉพาะ เหมาะสำหรับนักพัฒนาที่ต้องการระบบใหม่ๆ หรือต้องการทดสอบ ทดลองใช้งานฟังก์ชันใหม่ ซึ่งอาจจะไม่ได้ความเสถียรของระบบได้ 100% เนื่องจากว่า Package จากนักพัฒนานั้นยังอยู่ในช่วงของขั้นทดสอบ ทดลองเพื่อหาข้อผิดพลาดอยู่
- 2.4.x คือเวอร์ชันที่นำ Package ที่มีความเสถียรเป็นหลักนำมาใช้งาน เหมาะสำหรับผู้ที่ต้องการความมั่นคงของระบบ โดยไม่ได้มุ่งเน้นที่จะใช้ฟังก์ชันใหม่ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 REST เว็บเซอร์วิส [7]

ปัจจุบันเว็บเซอร์วิสแบบ REST ได้รับความนิยมในการนำมาใช้สร้างเซอร์วิสมากกว่าแบบ SOAP ด้วยความเร็วในการประมวลผล ที่รวดเร็วกว่าและอาศัยเทคโนโลยีของเว็บเป็นหลัก และง่ายต่อการเรียกใช้เซอร์วิส โดยเรียกผ่านทาง URL และมีการทำงานโดยใช้ฟังก์ชัน 4 อย่างคือ GET POST PUT และ DELETE โดยผ่านโปรโตคอล HTTP และ REST แอปพลิเคชันนั้นจะคิวรี (Query) ข้อมูลผ่านทาง URL เหมือนกันกับ Web Page Request และ HTTP Server จะให้ผลลัพธ์จากการถูก Request ออกมาในรูปแบบของ Document ซึ่ง Document ที่ได้มาจะอยู่ในรูปแบบของ XML หรือ JSON



รูปที่ 2.1 ตัวอย่างการเรียกใช้เว็บเซอร์วิสแบบ REST

2.7 เฟรมเวิร์คที่ใช้พัฒนา

2.7.1 RestKit [8]

RestKit เป็นเฟรมเวิร์คสำหรับ iOS ที่มีจุดประสงค์ในการที่จะเชื่อมต่อกับ RESTful web service ซึ่งจะเป็นการเชื่อมต่อเข้ากับเว็บเซอร์วิสผ่าน HTTP request API และ HTTP response API โดย Restkit จะช่วยลดความซับซ้อนในการเขียนโค้ดโปรแกรม

2.7.2 Zend เฟรมเวิร์ค [9]

Zend เฟรมเวิร์ค เป็น PHP เฟรมเวิร์ค ที่พัฒนาขึ้นโดยทีมงานผู้พัฒนาภาษา PHP เอง โดย Zend ได้จัดเตรียมไลบรารีต่างๆมากมาย ที่จำเป็นในการพัฒนาระบบเว็บแอปพลิเคชันเช่น พัฒนาเว็บแอปพลิเคชันแบบ Model-View-Controller (MVC) การ Validate ข้อมูล การ Authentication ในรูปแบบต่างๆ รองรับการพัฒนาเว็บแอปพลิเคชันแบบ Multi Language และอื่นๆอีกมากมาย

2.8 เครื่องมือที่ใช้ทดสอบระบบ

2.8.1 Acunetix Web Vulnerability Scanner [10]

Acunetix Web Vulnerability Scanner เป็นเครื่องมือที่ใช้ทดสอบความปลอดภัยของเว็บแอปพลิเคชัน ซึ่งช่วยตรวจสอบช่องโหว่เช่น SQL injection, Cross site scripting และช่องโหว่อื่นๆ ที่อาจละเมิดความปลอดภัยได้ ซึ่ง Acunetix จะทำการตรวจสอบเว็บไซต์ หรือ เว็บแอปพลิเคชันที่สามารถเข้าใช้งานผ่านเว็บเบราว์เซอร์ และใช้โปรโตคอล HTTP/HTTPS นอกจากความสามารถในการหาช่องโหว่แล้ว Acunetix ยังสามารถบอกวิธีแก้ไขที่มีประสิทธิภาพด้วย ซึ่ง Acunetix ยังสามารถทำงานได้กับ Javascript Ajax และ Web 2.0 อีกด้วย ช่องโหว่ที่หาพบจะแบ่งออกเป็น 4 ระดับคือ Hard Medium Low และ Information

- ระดับ Hard เป็นช่องโหว่ที่มีความอันตรายสูงสุด ส่งผลให้เว็บไซต์อาจถูก hack หรือ ขโมยข้อมูล
- ระดับ Medium เป็นช่องโหว่ที่เกิดจากการกำหนดค่าต่างๆของเซิร์ฟเวอร์ที่ผิดพลาด หรือ การเขียนโค้ดที่เปิดช่องโหว่ให้ผู้บุกรุกเจาะระบบเข้ามาได้
- ระดับ Low เป็นช่องโหว่ที่เกิดจากที่ไม่มีการเข้ารหัสข้อมูล หรือตำแหน่งไฟล์ (directory path) ถูกเปิดเผย
- ระดับ Informational Alert เป็นการแจ้งเตือนต่างๆไป เช่น เว็บอาจไม่ได้มีข้อมูลรั่วไหล หรืออีเมลล์ถูกเปิดเผย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

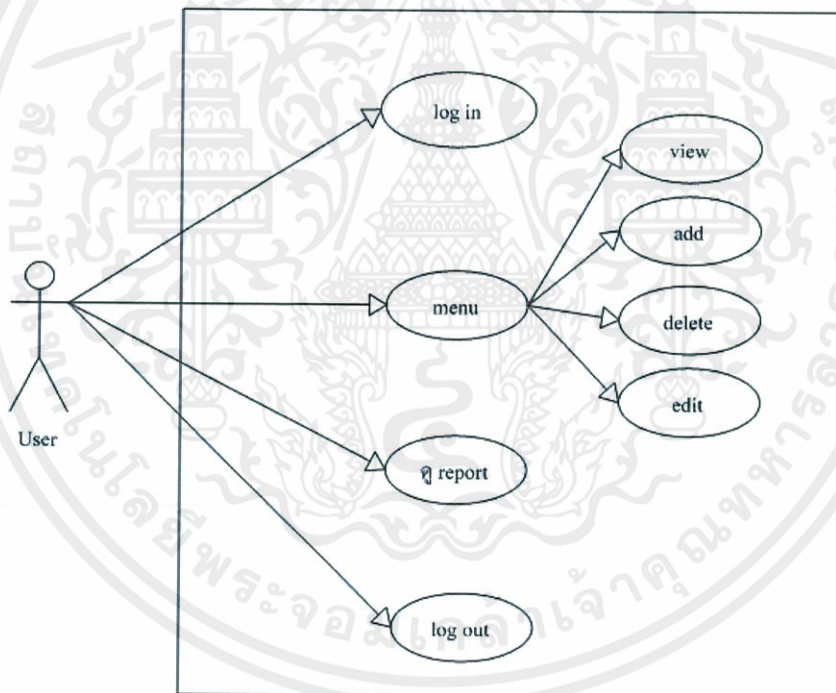
บทที่ 3

วิธีดำเนินงาน

3.1 ระบบของงาน

3.1.1 Use Case Diagram

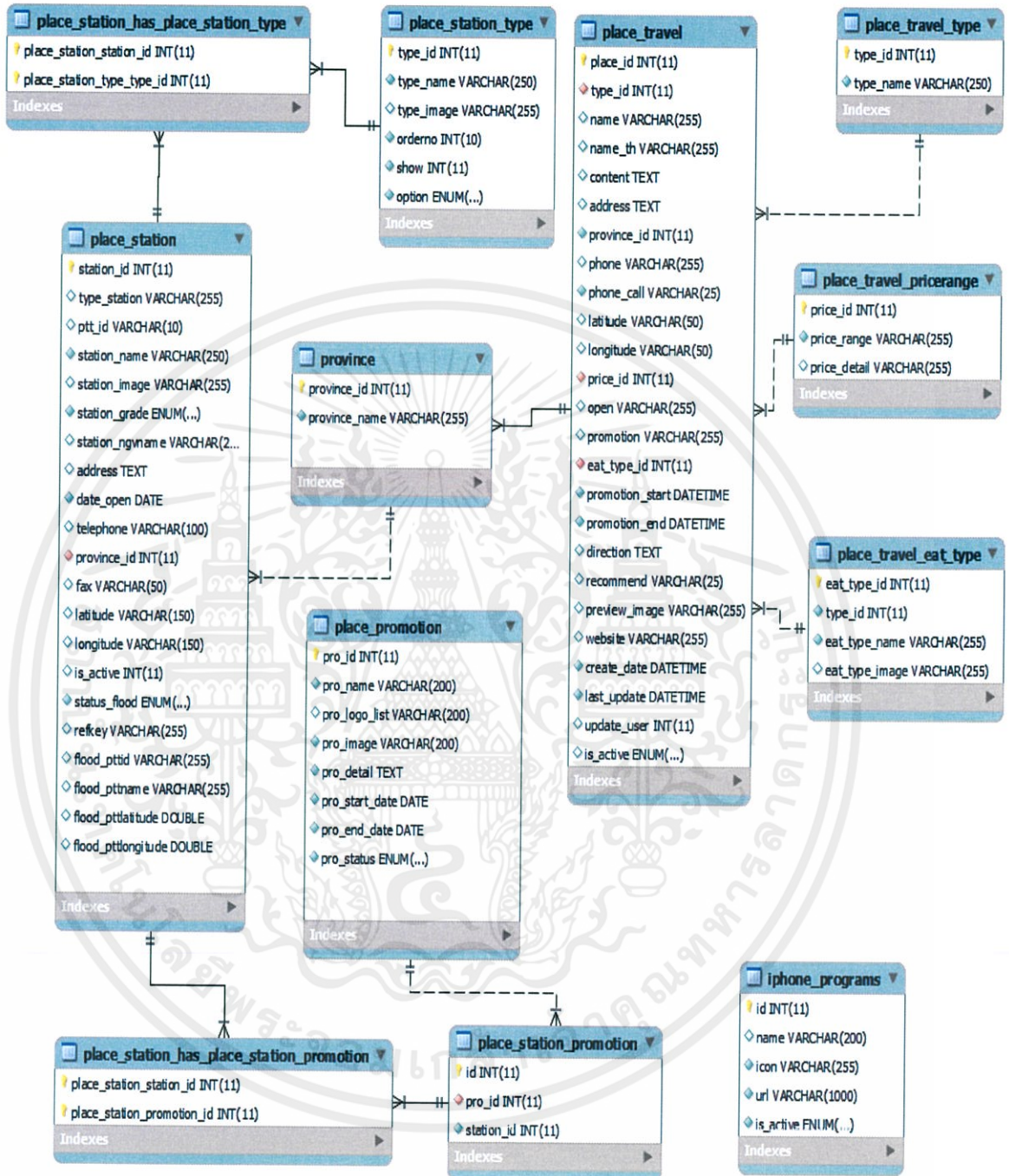
ผู้ใช้สามารถใช้งานแอปพลิเคชันโดยการล็อกอิน (log in) เข้าไปยังแอปพลิเคชัน และเมื่อการล็อกอินไม่มีข้อผิดพลาดอะไรผู้ใช้ก็จะสามารถที่จะทำงานเลือกเมนูรายการที่ต้องการ และในแต่ละเมนูนั้น ผู้ใช้สามารถที่จะทำการดูข้อมูล เพิ่มข้อมูล ลบข้อมูล และ แก้ไขข้อมูล โดยจะประกอบไปด้วยเมนูเมื่อผู้ใช้ต้องการดูประวัติการเข้าใช้งานก็สามารถทำได้เพราะได้มีการเก็บบันทึกการทำงานเอาไว้ และถ้าผู้ใช้ใช้งานแอปพลิเคชันเสร็จแล้วก็จะสามารถทำการล็อกเอาท์ (log out) ออกจากแอปพลิเคชันเพื่อป้องกันคนอื่นแอบเข้ามาใช้งาน



รูปที่ 3.1 Use Case Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 ER Diagram



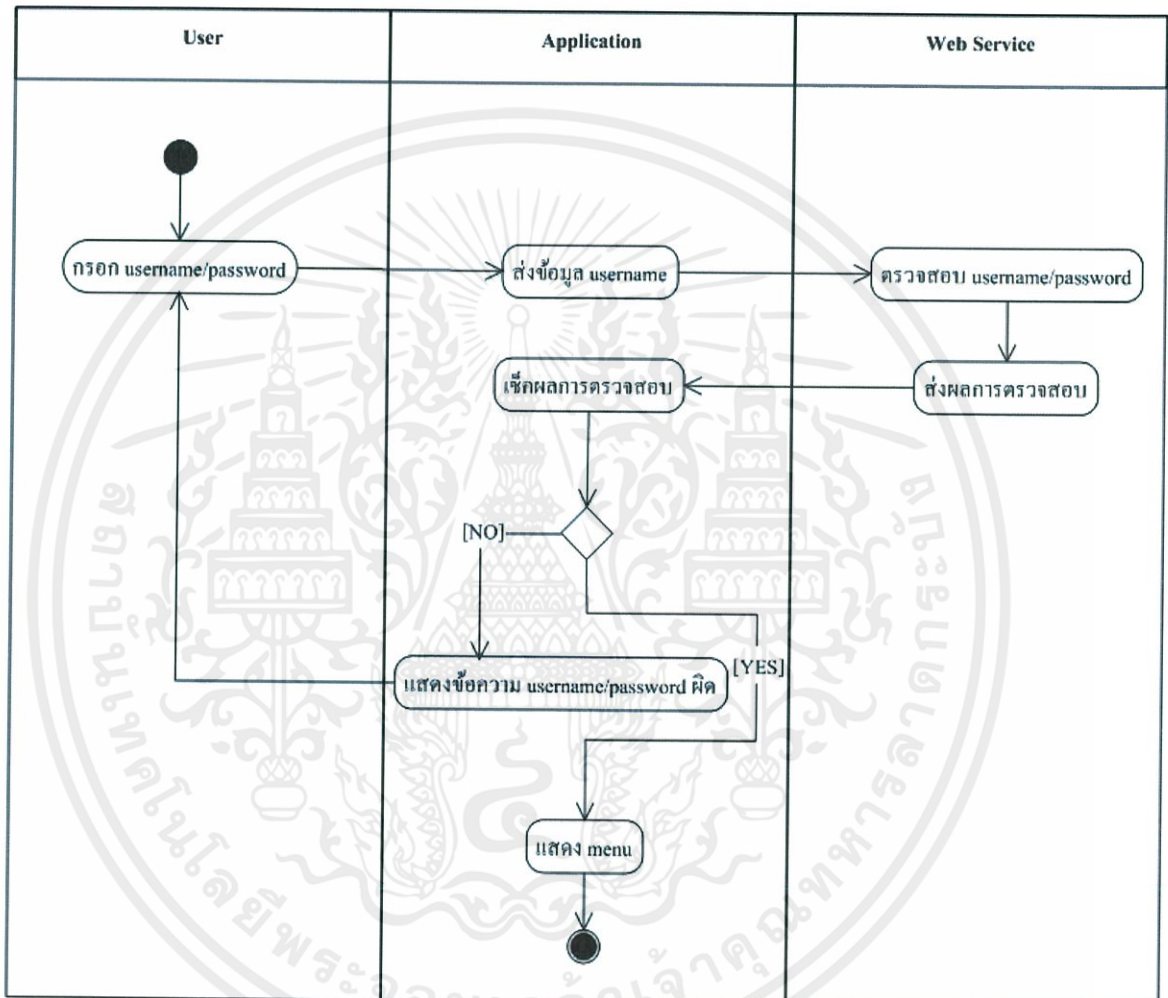
รูปที่ 3.2 โครงสร้าง ER Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 Activity Diagram

Activity Diagram จะประกอบไปด้วยการลือกอิน, การเลือกเมนู, การดูข้อมูล, การเพิ่มข้อมูล, การแก้ไขข้อมูล และการลบข้อมูลซึ่งในแต่ละ Activity Diagram จะมีตารางข้อมูลของฐานข้อมูลที่ใช้ในแต่ละ Activity Diagram ประกอบอยู่ด้วย

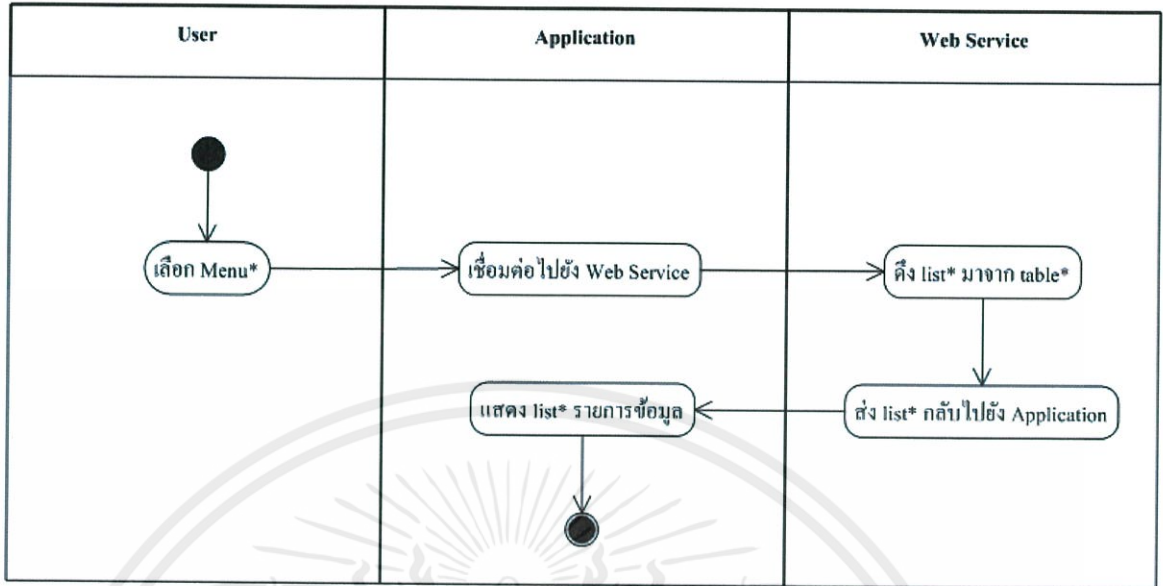
3.1.3.1 การลือกอิน



รูปที่ 3.3 Activity Diagram การลือกอิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3.2 การเลือกเมนู



รูปที่ 3.4 Activity Diagram การเลือกเมนู

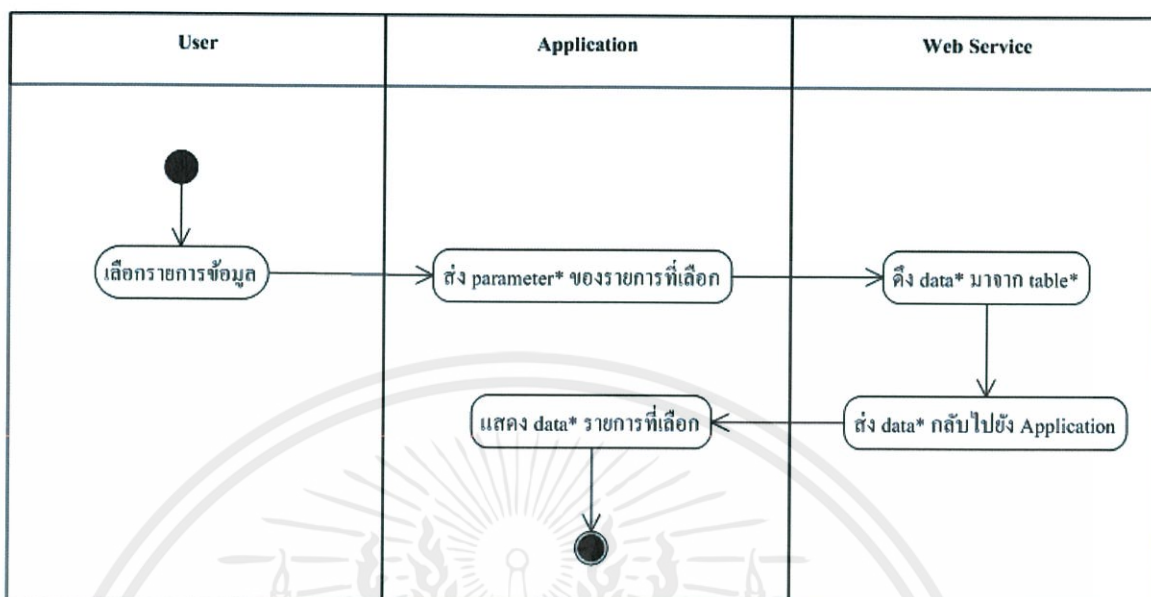
- Menu* คือ รายชื่อเมนูในแอปพลิเคชัน
- table* คือ table ที่อยู่ในฐานข้อมูล weloveptt
- list* คือ ข้อมูลที่อยู่ภายใน table*

ตารางที่ 3.1 ข้อมูลฐานข้อมูลใน list ของแต่ละเมนู

Menu*	table* (ชื่อ table)	list* (ชื่อ field)
Gas Station	place_station	station_id, station_name
Place Travel	place_travel	place_id, name
Oil Type Setting	place_station_type	type_id, type_name
Travel Place Type Setting	place_travel_type	type_id, type_name
Price Range Setting	place_travel_pricerange	price_id, price_range
Food Type Setting	place_travel_eat_type	eat_type_id, type_id
Promotion	place_promotion	pro_id, pro_name
Other Application	iphone_programs	id, name
LogReport	audittrail	Datetime, id, user, action, field

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3.3 การดูข้อมูล



รูปที่ 3.5 Activity Diagram การดูข้อมูล

parameter* คือ ข้อมูลที่ส่ง ไปยังเว็บเซอร์วิส

table* คือ table ที่อยู่ในฐานข้อมูล weloveptt

data* คือ ข้อมูลที่อยู่ภายใน table*

ตารางที่ 3.2 ข้อมูลฐานข้อมูลของแต่ละเมนู

Menu	table* (ชื่อ table)	data* (ชื่อ filed)	parameter*
Gas Station	place_station	station_id, is_active, station_name, address, station_ngvname, type_station, date_open, telephone, fax, latitude, longitude, status_flood	station_id
	province	province_name	
	place_station_type	type_name	

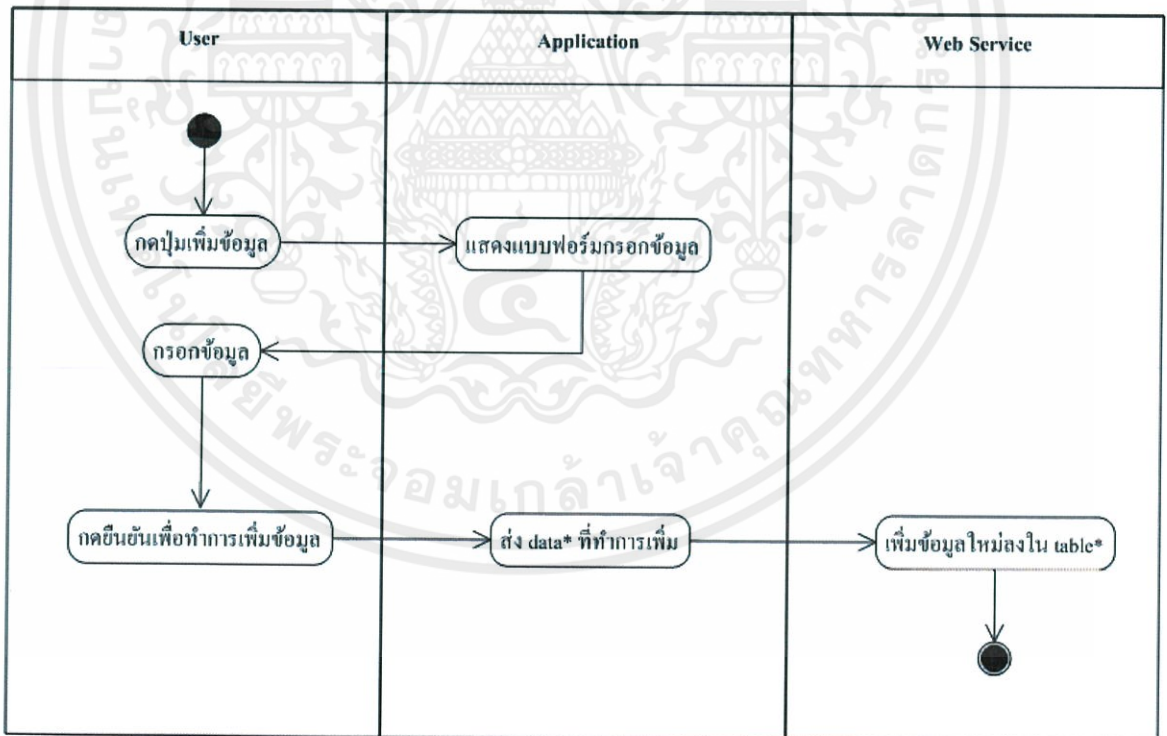
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Menu	table* (ชื่อ table)	data* (ชื่อ filed)	parameter*
Place Travel	place_travel	place_id, name_th, preview_image, name, content, address, phone, phone-call, latitude, longitude, open, is_active, pro_name, pro_start_date, website, pro_end_date, direction, recommend, create_date, last_update, update_user,	place_id
	place_travel_pricerange	price_range	
	province	province_name	
	place_travel_eat_type	eat_type_name	
	place_travel_type	type_name	
Oil Type Setting	place_station_type	type_id, type_name, type_image, orderno, show, option	type_id
Travel Place Type Setting	place_travel_type	type_id, type_name	type_id
Price Range Setting	place_travel_pricerange	price_id, price_range, price_detail	price_id
Food Type Setting	place_travel_eat_type	eat_type_id, type_id, eat_type_name, eat_type_image	eat_type_id

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Menu	table* (ชื่อ table)	data* (ชื่อ filed)	parameter*
Promotion	place_promotion	pro_id, pro_name, pro_logo_list, pro_image, pro_detail, pro_start_date, pro_end_date, pro_status	pro_id
Other Application	iphone_programs	id, name, icon, url, is_active	id
LogReport	audittrail	id,datetime,script,user, action,table,field, keyword,oldvalue, newvalue	id

3.1.3.4 การเพิ่มข้อมูล



รูปที่ 3.6 Activity Diagram การเพิ่มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังขอสงวนสิทธิ์ในข้อมูลที่มีเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

data* คือ ข้อมูลที่จะทำการเพิ่มลงไปยังฐานข้อมูล

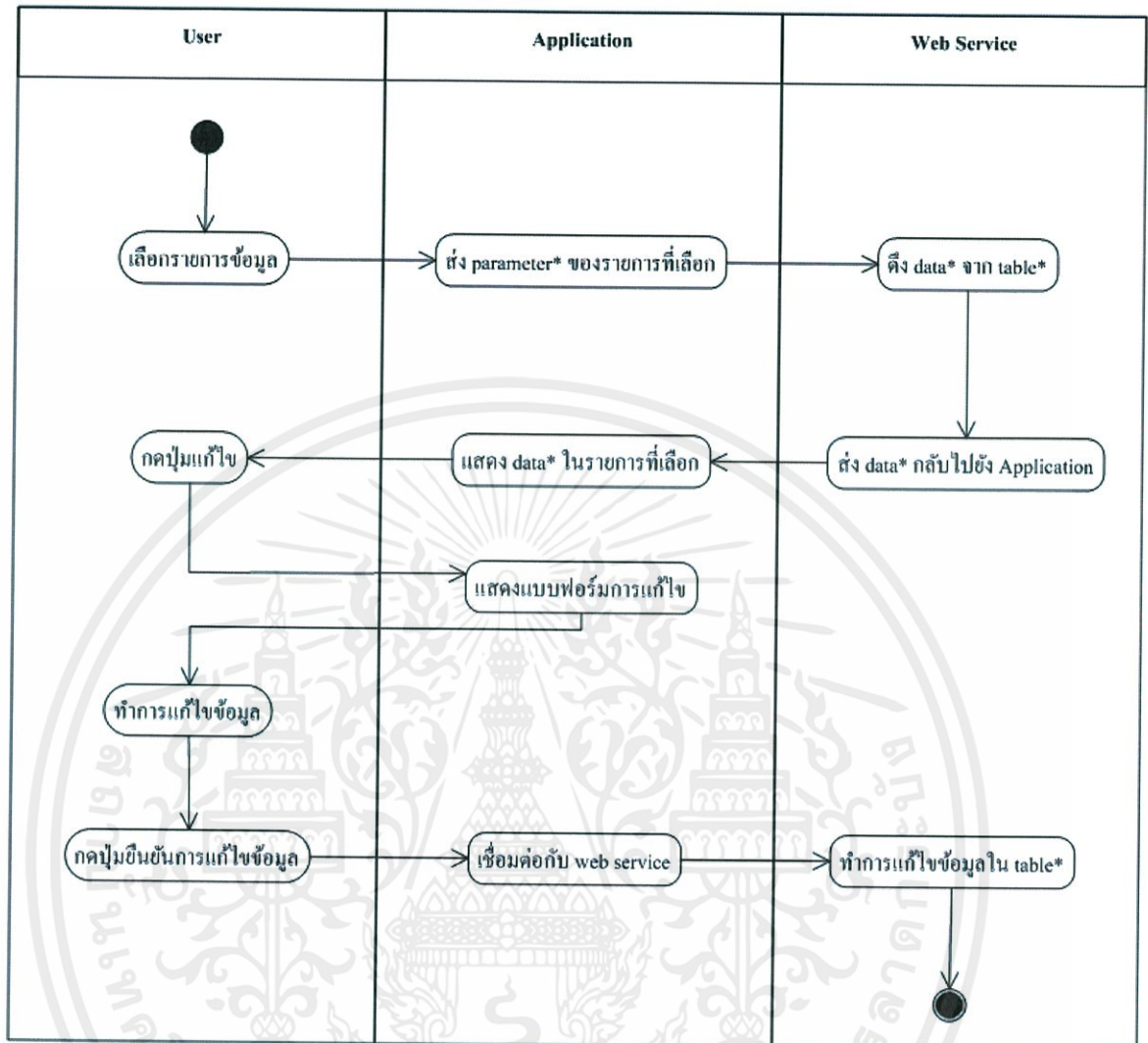
table* คือ table ที่ทำการเพิ่มข้อมูลเข้าไป

ตารางที่ 3.3 ข้อมูลฐานข้อมูลที่ทำการเพิ่มในแต่ละเมนู

Menu	table* (ชื่อ table)	data* (ชื่อ filed)
Gas Station	place_station	station_id, is_active, station_name, station_ngvname, type_station, address, date_open, telephone, fax, latitude, longitude, status_flood
	province	province_name
	place_station_type	type_name
Place Travel	place_travel	place_id, name_th, preview_image, name, content, address, phone, phone-call, latitude, longitude, open, pro_name, pro_start_date, pro_end_date, direction, recommend, website, create_date, last_update, update_user, is_active
	place_travel_pricerange	price_range
	province	province_name
	place_travel_eat_type	eat_type_name
	place_travel_type	type_name
Oil Type Setting	place_station_type	type_id, type_name, type_image, orderno, show, option
Travel Place Type Setting	place_travel_type	type_id, type_name
Price Range Setting	place_travel_pricerange	price_id, price_range, price_detail
Food Type Setting	place_travel_eat_type	eat_type_id, type_id, eat_type_name, eat_type_image
Other Application	iphone_programs	id, name, icon, url, is_active

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3.5 การแก้ไขข้อมูล



รูปที่ 3.7 Activity Diagram การแก้ไขข้อมูล

data* คือ ข้อมูลที่จะทำการแก้ไขในฐานข้อมูล

table* คือ table ที่ทำการแก้ไขข้อมูล

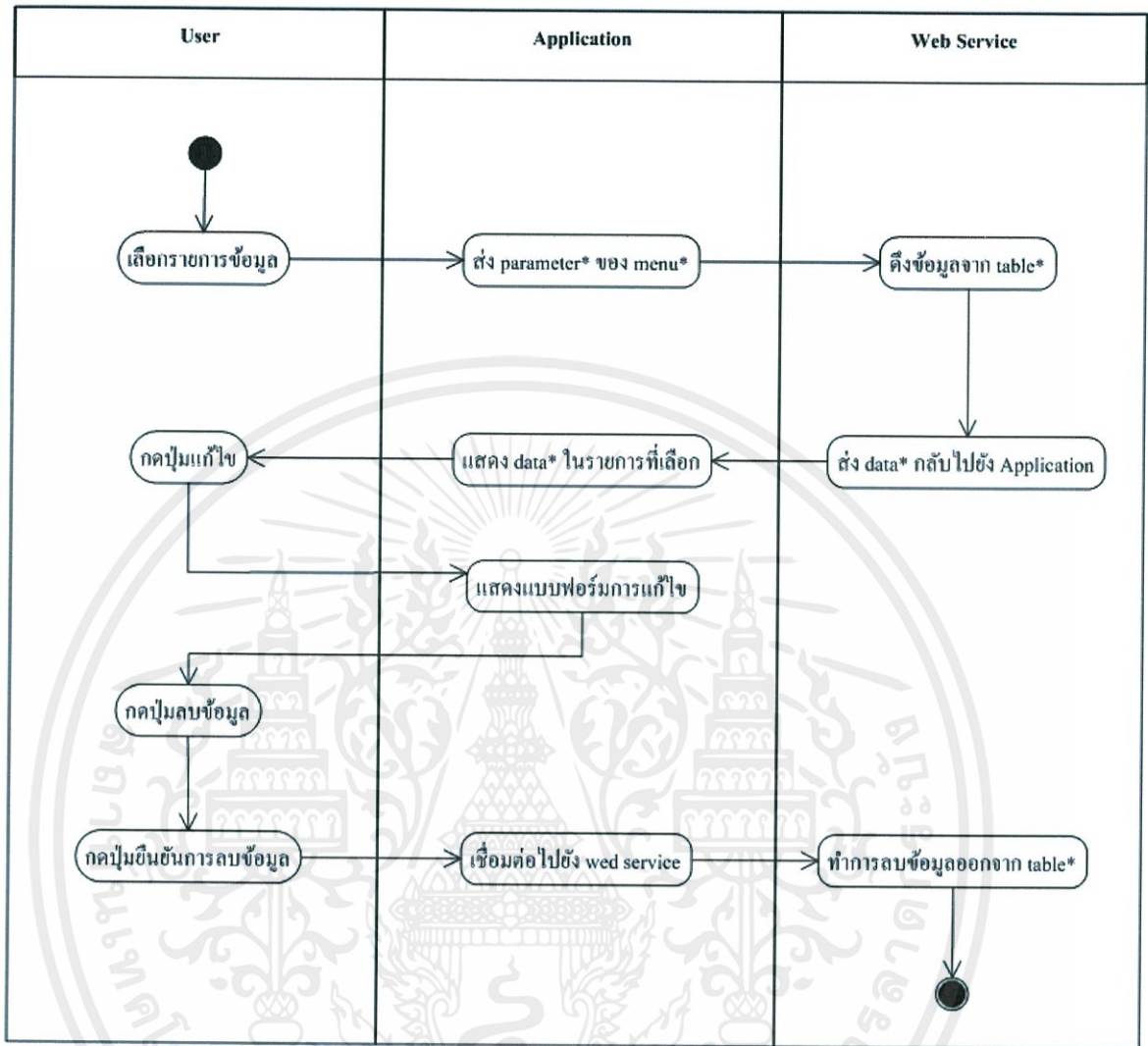
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.4 ข้อมูลฐานข้อมูลที่ทำการแก้ไขในแต่ละเมนู

Menu	table* (ชื่อ table)	data* (ชื่อ filed)
Gas Station	place_station	station_id, is_active, station_name, station_ngvname, type_station, address, date_open, telephone, fax, latitude, longitude, status_flood
	province	province_name
	place_station_type	type_name
Place Travel	place_travel	place_id, name_th, preview_image, name, content, address, phone, phone-call, latitude, longitude, open, pro_name, pro_start_date, pro_end_date, direction, recommend, website, create_date, last_update, update_user, is_active
	place_travel_pricerange	price_range
	province	province_name
	place_travel_eat_type	eat_type_name
	place_travel_type	type_name
Oil Type Setting	place_station_type	type_id, type_name, type_image, orderno, show, option
Travel Place Type Setting	place_travel_type	type_id, type_name
Price Range Setting	place_travel_pricerange	price_id, price_range, price_detail
Food Type Setting	place_travel_eat_type	eat_type_id, type_id, eat_type_name, eat_type_image
Other Application	iphone_programs	id, name, icon, url, is_active

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3.6 การลบข้อมูล



รูปที่ 3.8 Activity Diagram การลบข้อมูล

data* คือ ข้อมูลที่จะทำการลบในฐานข้อมูล

table* คือ table ที่ทำการลบข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

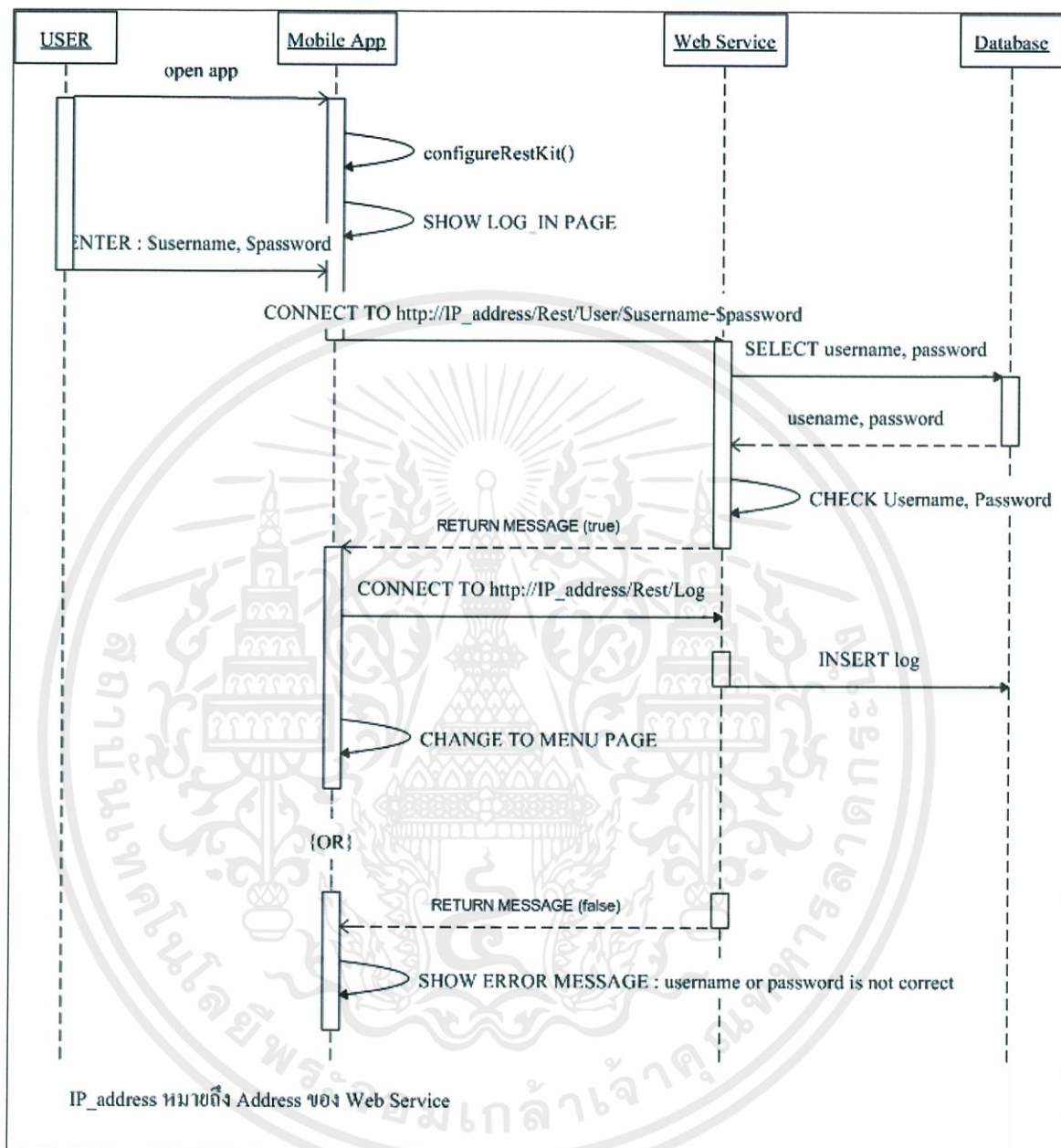
ตารางที่ 3.5 ข้อมูลฐานข้อมูลที่ทำการลบในแต่ละเมนู

Menu	table* (ชื่อ table)	data* (ชื่อ filed)
Gas Station	place_station	station_id, is_active, station_name, station_ngvname, type_station, address, date_open, telephone, fax, latitude, longitude, status_flood
	province	province_name
	place_station_type	type_name
Place Travel	place_travel	place_id, name_th, preview_image, name, content, address, phone, phone-call, latitude, longitude, open, pro_name, pro_start_date, pro_end_date, direction, recommend, website, create_date, last_update, update_user, is_active
	place_travel_pricerange	price_range
	province	province_name
	place_travel_eat_type	eat_type_name
	place_travel_type	type_name
Oil Type Setting	place_station_type	type_id, type_name, type_image, orderno, show, option
Travel Place Type Setting	place_travel_type	type_id, type_name
Price Range Setting	place_travel_pricerange	price_id, price_range, price_detail
Food Type Setting	place_travel_eat_type	eat_type_id, type_id, eat_type_name, eat_type_image
Other Application	iphone_programs	id, name, icon, url, is_active

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4 Sequence Diagram

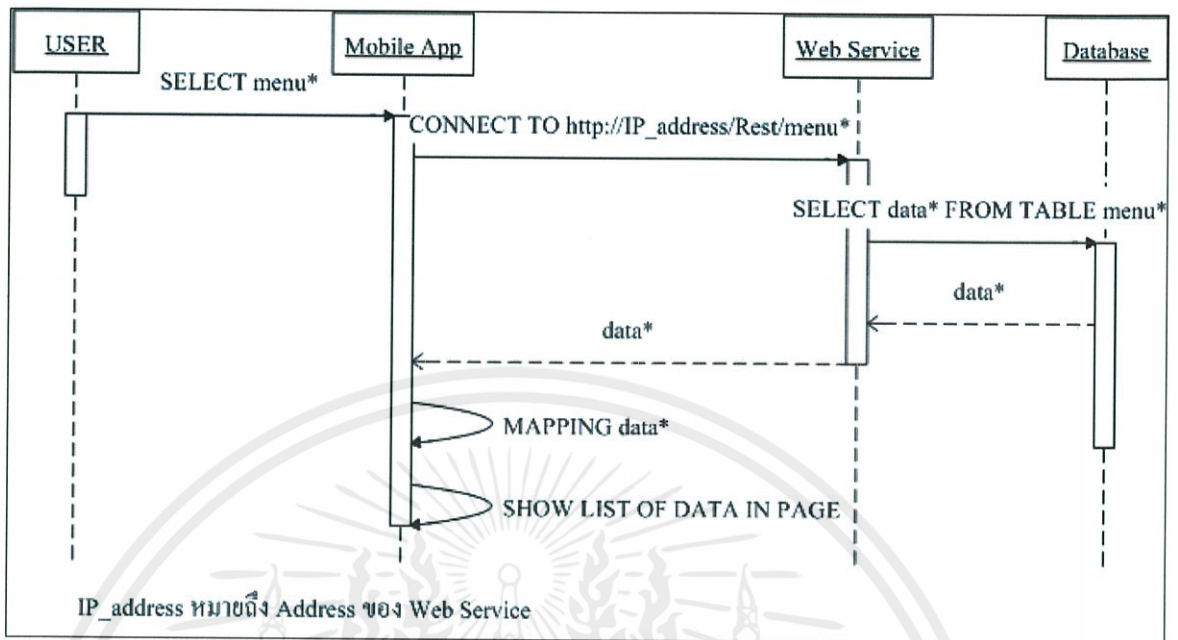
3.1.4.1 การล็อกอิน



รูปที่ 3.9 Sequence Diagram การล็อกอิน

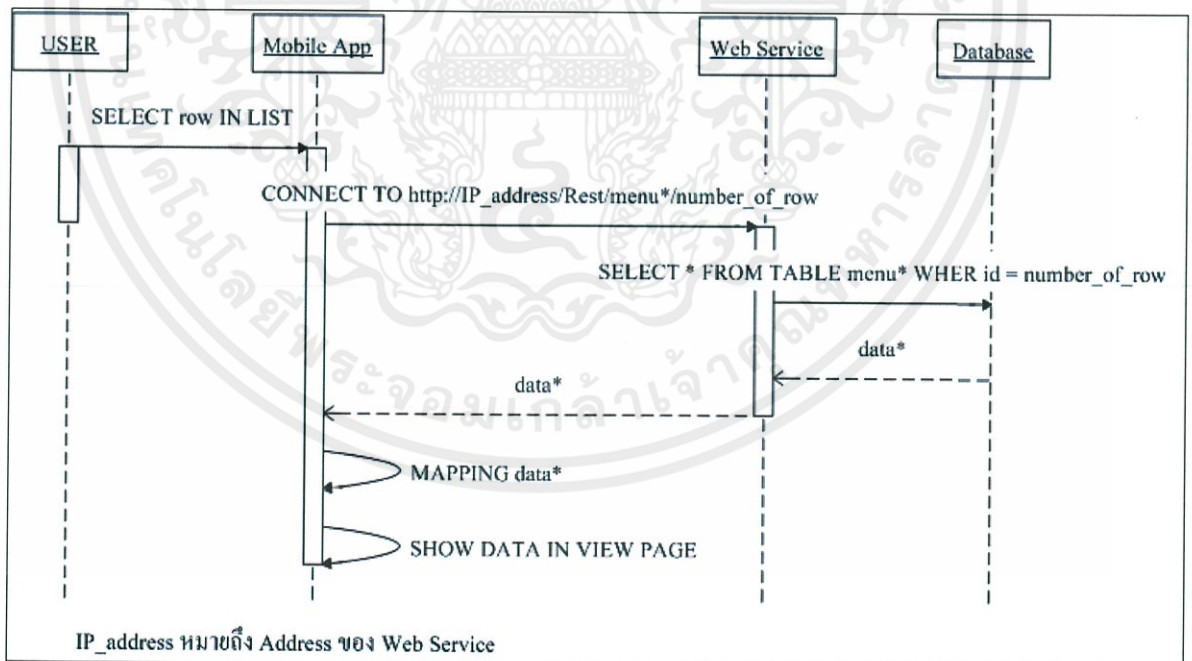
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4.2 การแสดงรายการข้อมูล



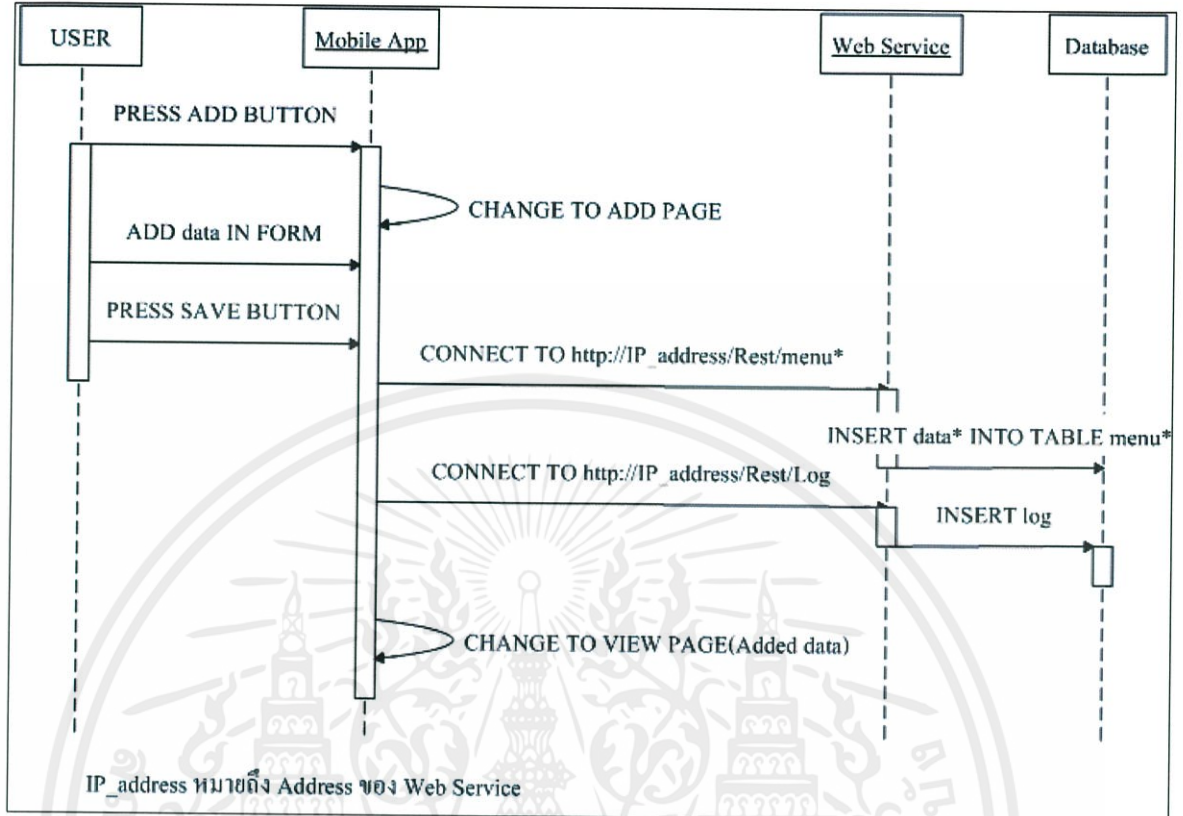
รูปที่ 3.10 Sequence Diagram การแสดงรายการข้อมูล

3.1.4.3 การดูข้อมูล



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3.11 Sequence Diagram การดูข้อมูล ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

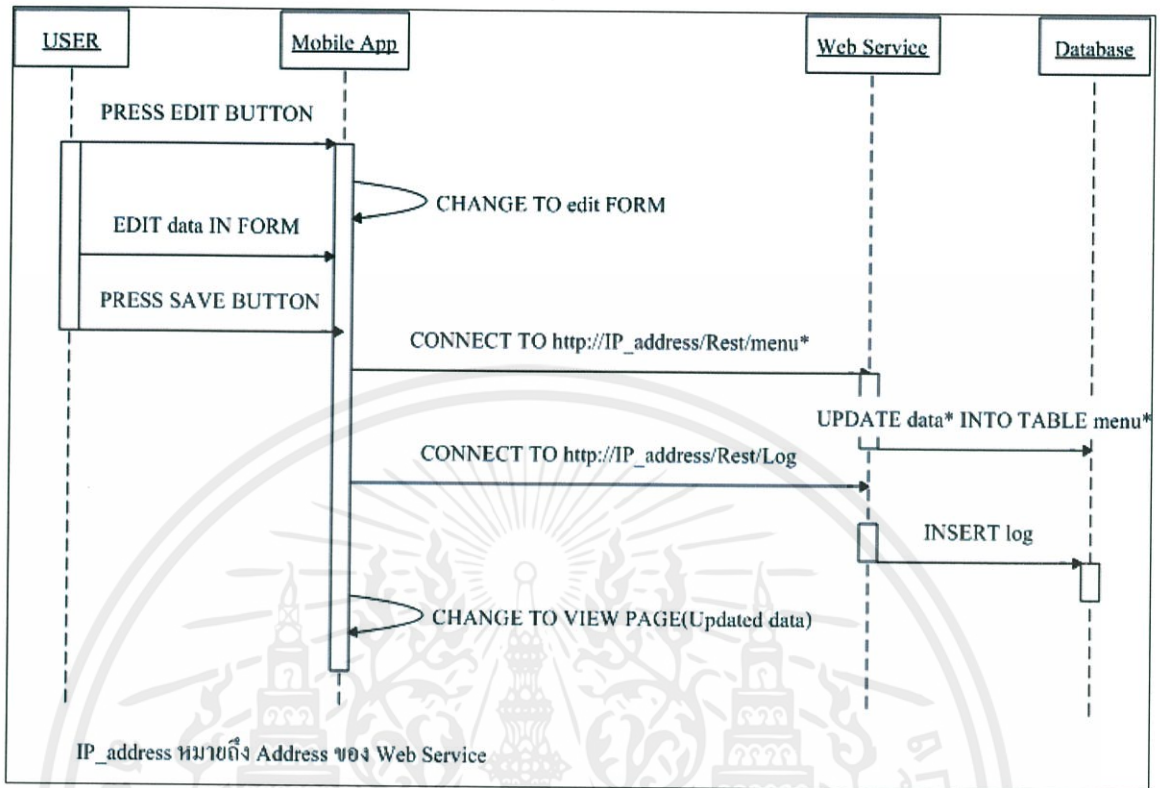
3.1.4.4 การเพิ่มข้อมูล



รูปที่ 3.12 Sequence Diagram การเพิ่มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

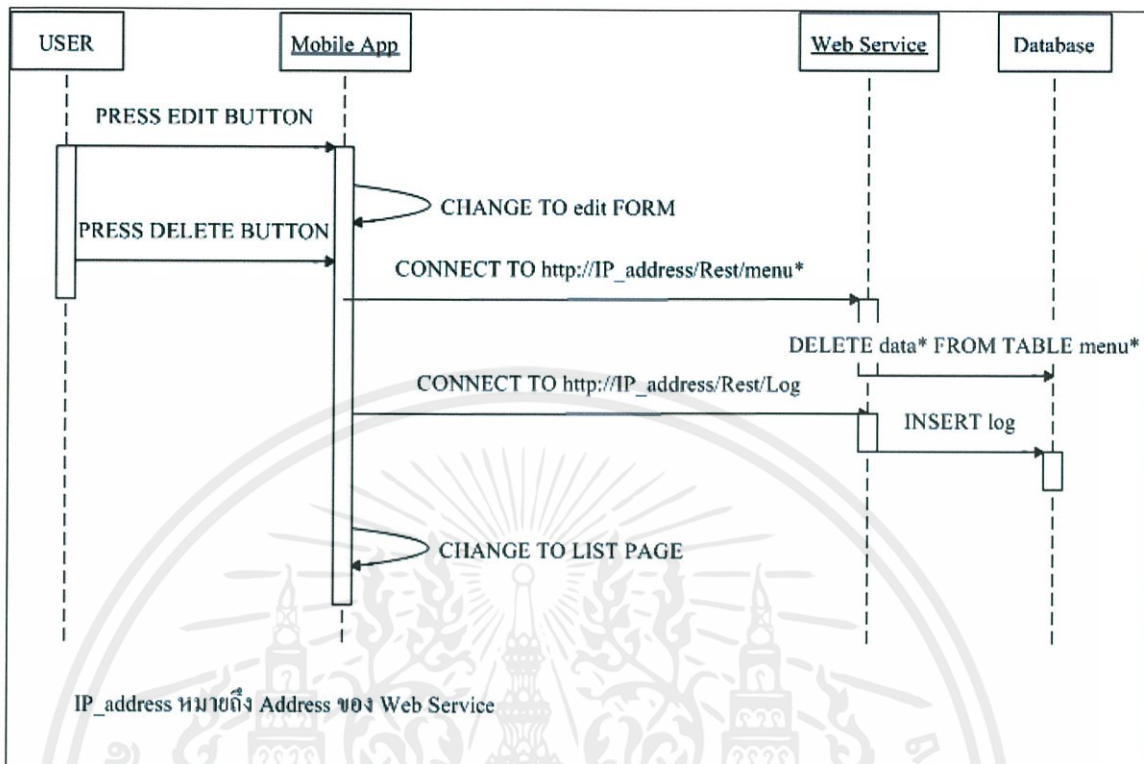
3.1.4.5 การแก้ไขข้อมูล



รูปที่ 3.13 Sequence Diagram การแก้ไขข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4.6 การลบข้อมูล



รูปที่ 3.14 Sequence Diagram การลบข้อมูล

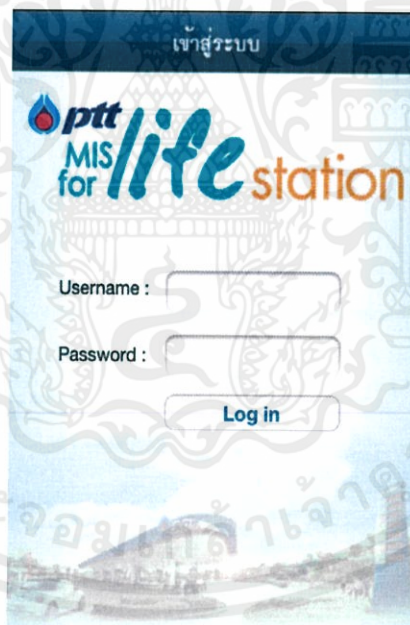
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบงาน

เมนูต่างๆภายในแอปพลิเคชันจะประกอบไปด้วย 9 เมนูดังนี้

1. Gas Station จะเป็นข้อมูลของสถานีน้ำมันทั้งหมดของ ปตท.
2. Oil Type จะเป็นข้อมูลประเภทน้ำมันของ ปตท.
3. Promotion จะเป็นข้อมูลโปรโมชั่นต่างๆของ ปตท.
4. Place Travel จะเป็นข้อมูลการท่องเที่ยวต่างๆ
5. Travel Type จะเป็นข้อมูลประเภทของการท่องเที่ยว
6. Price Range จะเป็นข้อมูลราคาของสถานที่ท่องเที่ยวต่างๆ
7. Travel Eat Type จะเป็นข้อมูลของสถานที่ต่างๆที่อยู่ในแต่ละประเภทของ Travel Type
8. Other Application จะเป็นข้อมูลของแอปพลิเคชันอื่นๆที่เกี่ยวข้องกับ ปตท.
9. Log Report จะเป็นข้อมูลการกระทำต่างๆภายในแอปพลิเคชัน PTT MIS for Life Station

3.2.1 หน้าจอล็อกอิน



รูปที่ 3.15 หน้าจอล็อกอิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 หน้าจอเมนูหลัก



รูปที่ 3.16 หน้าจอเมนูหลัก

3.2.3 หน้าจอกรอกรายละเอียด Gas Station

รูปที่ 3.17 หน้าจอกรอกรายละเอียด Gas Station

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 หน้าจอกรอกรายละเอียด Place Travel

MIS for Life Station

PLACE TRAVEL

- การใช้งาน ON
- ชื่อภาษาอังกฤษ
- ชื่อภาษาไทย
- รายละเอียด
- ที่อยู่

กลับมาที่นี่

รูปที่ 3.18 หน้าจอกรอกรายละเอียด Place Travel

3.2.5 หน้าจอกรอกรายละเอียด Oil Type Setting

MIS for Life Station

Oil Type Setting

- ชื่อประเภท
- รูป
- คำอธิบายแสดง
- แสดง/ซ่อน ON
- Option

กลับมาที่นี่

รูปที่ 3.19 หน้าจอกรอกรายละเอียด Oil Type Setting

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6 หน้าจอรอกรายละเอียด Travel Place Type Setting



รูปที่ 3.20 หน้าจอรอกรายละเอียด Oil Type Setting

3.2.7 หน้าจอรอกรายละเอียด Price Range Setting



รูปที่ 3.21 หน้าจอรอกรายละเอียด Price Range Setting

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.8 หน้าจอรอกรายละเอียด Food Type Setting

รูปที่ 3.22 หน้าจอรอกรายละเอียด Food Type Setting

3.2.9 หน้าจอรอกรายละเอียด Promotion

รูปที่ 3.23 หน้าจอรอกรายละเอียด Promotion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.10 หน้าจอรอกรายละเอียด Other Application

รูปที่ 3.24 หน้าจอรอกรายละเอียด Other Application

3.2.11 หน้าจอ Log Report

รูปที่ 3.25 หน้าจอ Log Report

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

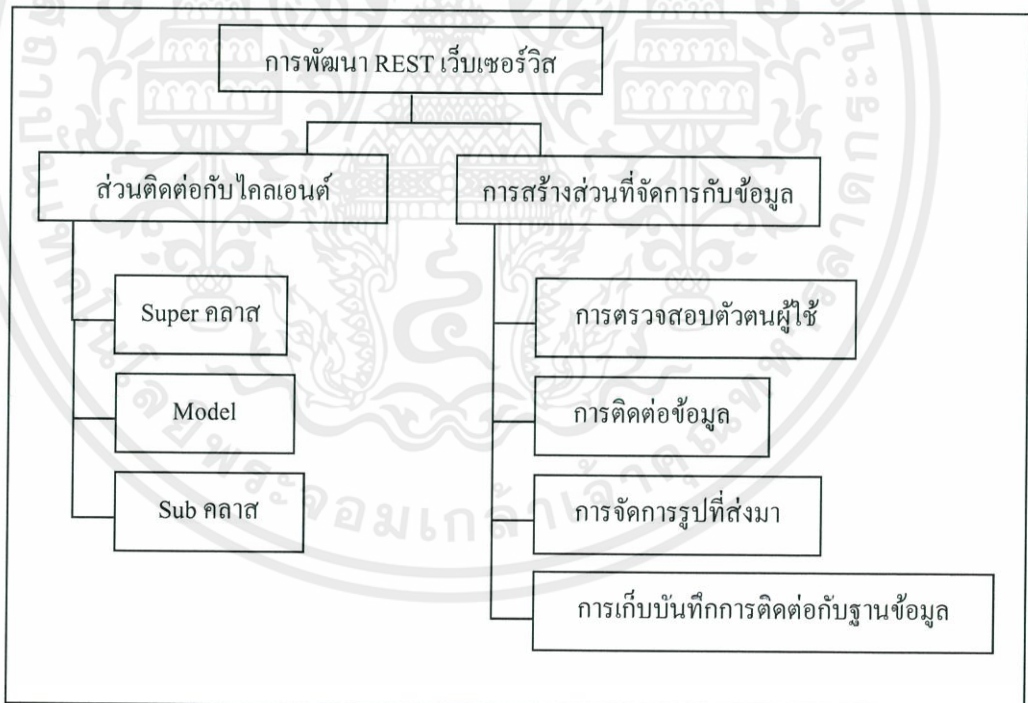
3.3 การพัฒนาแอปพลิเคชัน

การพัฒนาแอปพลิเคชันนั้นสามารถแบ่งได้เป็นสองส่วนคือ ส่วนของเว็บเซอร์วิสและส่วน
โมบายแอปพลิเคชัน

3.3.1 การพัฒนา REST เว็บเซอร์วิส

ในการพัฒนาเว็บเซอร์วิส REST สามารถแบ่งขั้นตอนออกเป็น 2 ส่วนใหญ่คือ การสร้าง
ส่วนที่ติดต่อกับไคลเอนต์ และ การสร้างส่วนที่จัดการกับข้อมูล โดยจะมีการใช้ Zend เฟรมเวิร์ค
(ที่มีโครงสร้างแบบ MVC) ในการพัฒนาเว็บเซอร์วิส

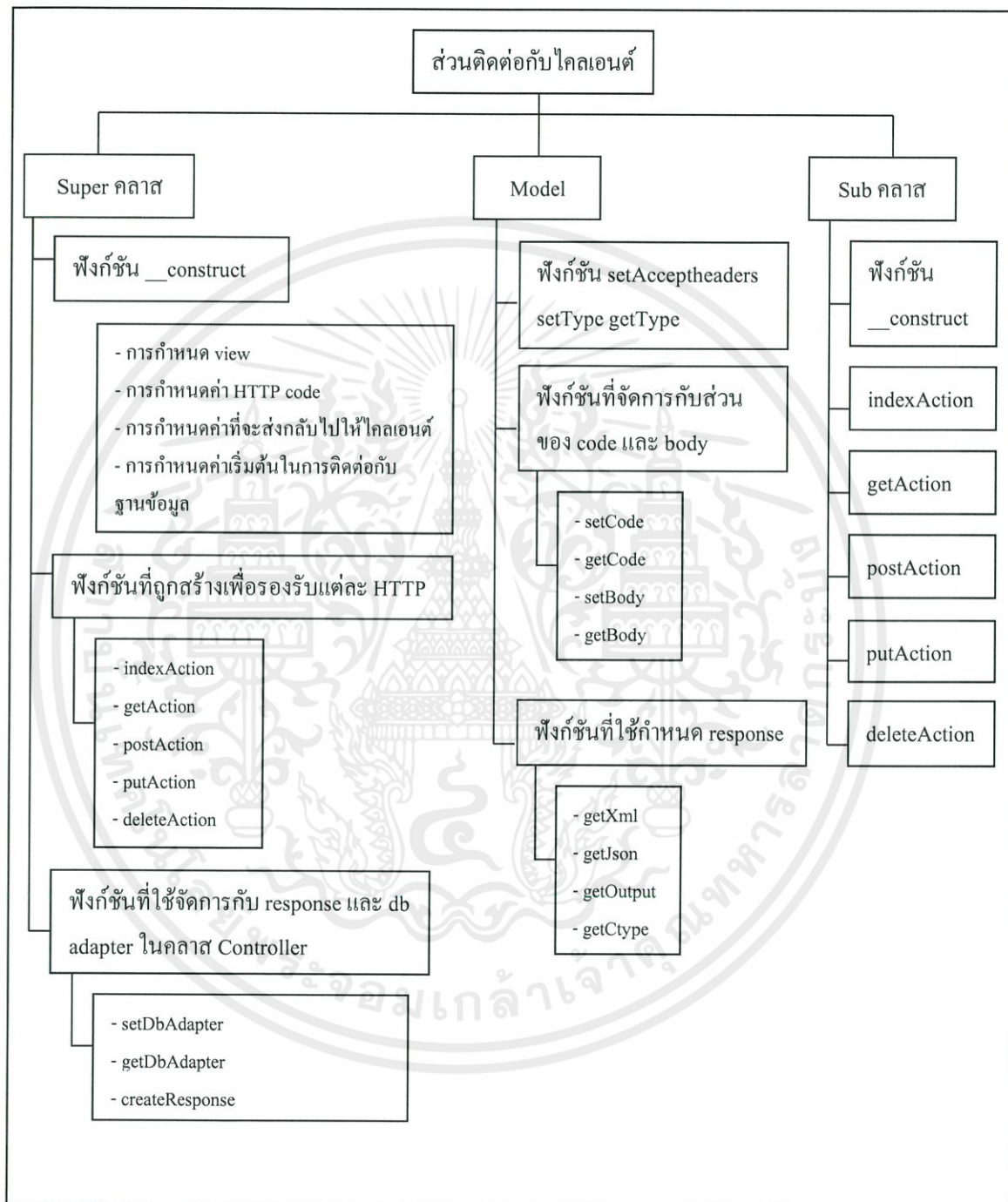
ส่วนที่ติดต่อกับไคลเอนต์นั้น ในการพัฒนาจะแบ่งการพัฒนาออกเป็นสามส่วน คือ ส่วน
ของ Super class ส่วน Model และ ส่วนของ Sub class ในส่วนของการสร้างส่วนที่จัดการข้อมูล
จะประกอบไปด้วยส่วนคือ ส่วนของการตรวจสอบตัวตนของผู้ใช้ ส่วนของการติดต่อข้อมูล ส่วนของ
การจัดการรูปภาพ และ ส่วนที่ใช้ในการบันทึกการติดต่อกับฐานข้อมูล ดังรูปที่ 3.26



รูปที่ 3.26 diagram การพัฒนา REST เว็บเซอร์วิส

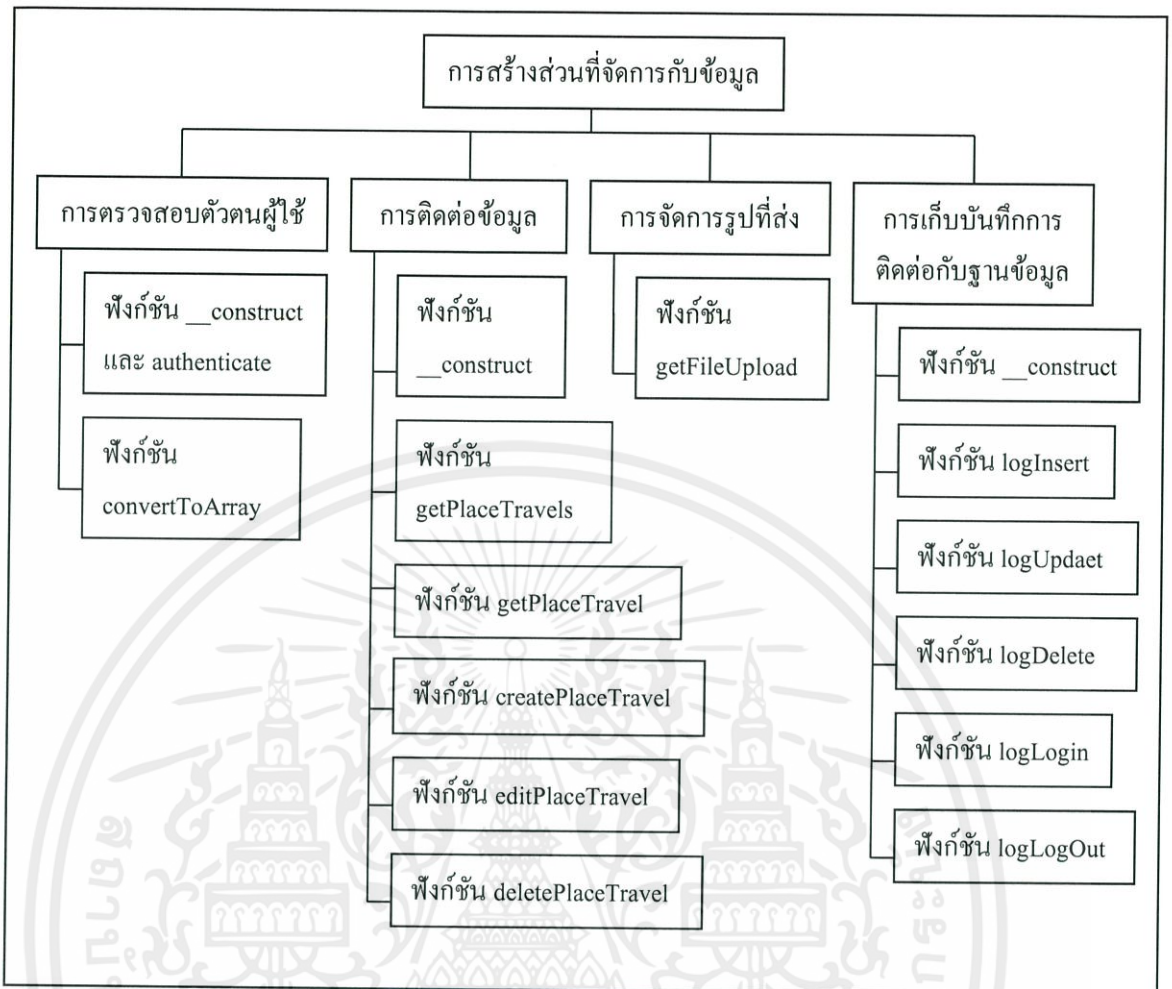
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับรูปที่ 3.27 และ รูปที่ 3.28 แสดง ภาพรวมของเมธอดต่างๆ ที่ใช้ในการพัฒนาของ ส่วนที่ติดต่อกับไคลเอนต์ และ ส่วนที่จัดการกับข้อมูล ตามลำดับ รายละเอียดเพิ่มเติม และ โค้ดที่ใช้ ในการพัฒนาส่วนต่างๆ นั้น สามารถดูเพิ่มเติมได้จาก ภาคผนวก ง.1



รูปที่ 3.27 diagram การพัฒนาส่วนที่ติดต่อกับไคลเอนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.28 diagram การพัฒนาส่วนที่จัดการกับฐานข้อมูล

3.3.2 การพัฒนา iOS แอปพลิเคชัน

การพัฒนา iOS แอปพลิเคชัน สามารถแบ่งขั้นตอนออกเป็น 3 ส่วนใหญ่คือ การสร้างส่วนติดต่อกับผู้ใช้ การติดต่อกับ REST เว็บเซอร์วิส และ ฟังก์ชันการทำงานของแอปพลิเคชัน

โดยใน รูปที่ 3.29 จะแสดงให้เห็นภาพรวมของการพัฒนา iOS แอปพลิเคชัน ว่ามีการทำงานอะไรบ้าง ซึ่งสามารถรายละเอียดเพิ่มเติมรวมถึงโค้ดที่ใช้พัฒนาได้ใน ภาคผนวก ง.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.29 diagram การพัฒนา iOS แอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงานและการทดสอบระบบ

4.1 ผลการดำเนินงาน

หลังจากที่พัฒนาแอปพลิเคชันสำเร็จแล้ว ผลลัพธ์ที่ได้คือ แอปพลิเคชัน PTT MIS for Life Station ที่สามารถ เรียกดู เพิ่ม แก้ไข และ ลบ ข้อมูลต่างๆ ทั้งหมดของแอปพลิเคชัน PTT Life Station ทั้งหมด 8 ส่วน และ ส่วนของ Log Report ที่จะทำการเก็บข้อมูลการใช้งานของผู้ใช้แอปพลิเคชันนี้ โดยข้อมูลต่างๆ ทั้งหมดของแอปพลิเคชัน สามารถแบ่งได้ออกเป็น 8 ส่วน ดังนี้

1. ข้อมูลเกี่ยวกับสถานีบริการน้ำมัน (เป็นข้อมูลทั้งหมดของแต่ละสถานี เช่น ชื่อสถานี สถานีที่ตั้ง วันที่ทำการเปิดบริการ และ บริการต่างๆ ภายในสถานี) สามารถเรียกใช้งานได้ที่เมนู Gas Station
2. ข้อมูลเกี่ยวกับบริการต่างๆ ที่ให้บริการในแต่ละสถานี เช่น น้ำมันประเภทต่างๆ และ ร้านค้าที่เปิดในสถานีบริการน้ำมัน สามารถเรียกใช้ได้เมนู Oil Type
3. ข้อมูลโปรโมชั่นต่างๆ ภายในสถานี สามารถเรียกใช้ได้เมนู Promotion
4. ข้อมูลเกี่ยวกับสถานที่ท่องเที่ยว เช่น ชื่อ ที่ตั้ง และ รูปของสถานที่ท่องเที่ยว สามารถเรียกใช้งานได้ที่เมนู Place Travel
5. ข้อมูลประเภทของสถานที่ท่องเที่ยว ตัวอย่างเช่น เป็นร้านอาหาร หรือ ร้านนั่งดื่ม สามารถเรียกใช้ได้เมนู Travel Type
6. ข้อมูลเกี่ยวกับช่วงราคา เป็นช่วงราคาของร้านอาหารแต่ละร้านไว้ สามารถเรียกใช้ได้เมนู Price Range
7. ข้อมูลประเภทร้านอาหาร สามารถเรียกใช้ได้เมนู Travel Eat Type
8. ข้อมูลเกี่ยวกับแอปพลิเคชัน โดยจะเป็นแอปพลิเคชันอื่นๆ ที่ทางเครือ ปตท. ให้บริการอยู่ สามารถเรียกใช้ได้เมนู Other Application

สำหรับส่วนของ Log Report ผู้ใช้สามารถเรียกใช้ได้เมนู Log Report

ในการเข้าใช้งานแอปพลิเคชันจะมีการพิสูจน์ตัวตนของผู้ใช้กับเว็บเซิร์ฟเวอร์ผ่านทางหน้าจอล็อกอิน การเชื่อมต่อกับเว็บเซิร์ฟเวอร์ตลอดการใช้งานจะผ่านโปรโตคอล https นอกจากนี้ แอปพลิเคชันอนุญาตให้ผู้ใช้สามารถตั้งค่าหมายเลข IP Address ของ Rest เว็บเซอร์วิส ในกรณีที่ไม่ต้องการใช้ค่า Default IP address ได้อีกด้วย

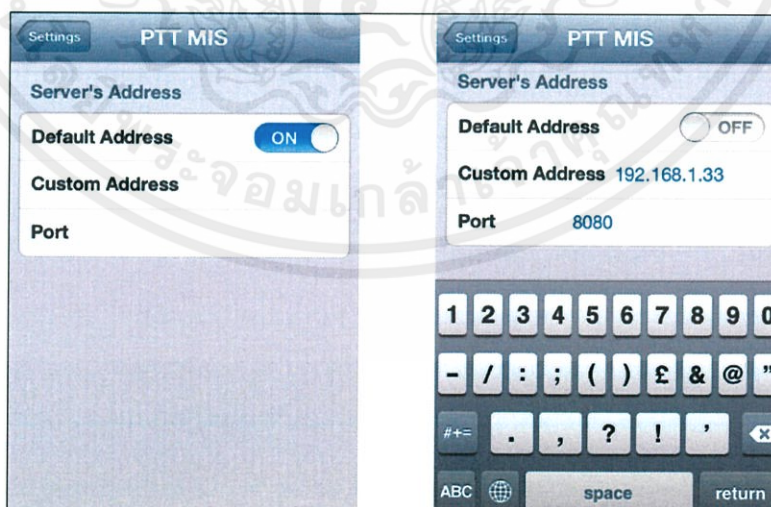
เอกสารนี้เป็นเอกสารที่ความไว้สำหรับการใช้งานเพื่อการสื่อสารเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเริ่มต้นใช้งานแอปพลิเคชัน PTT MIS for Life Station ถ้าผู้ใช้ไม่ต้องการใช้ค่า Default ผู้ใช้จะต้องไปตั้งค่าหมายเลข IP Address ของ Rest เว็บเซอร์วิสซึ่งอยู่ภายในเมนู Setting ของเครื่อง ดังรูปที่ 4.1 ซึ่งการกำหนดค่า IP Address ของ Rest เว็บเซอร์วิส ถ้า Default Address เป็น On จะเป็นการติดต่อไปยังเซิร์ฟเวอร์หลักของทาง ปตท. แต่ถ้าเป็น Off จะเป็นการติดต่อไปยังเซิร์ฟเวอร์ชั่วคราวซึ่งจะต้องกำหนดค่า IP Address ในช่อง Custom Address และ กำหนดค่า Port ในช่อง Port ดังรูปที่ 4.1



รูปที่ 4.1 การตั้งค่าภายใน Setting

การกำหนดค่า IP Address ของ Rest เว็บเซอร์วิส ถ้า Default Address เป็น On จะเป็นการติดต่อไปยังเซิร์ฟเวอร์หลักของทาง ปตท. แต่ถ้าเป็น Off จะเป็นการติดต่อไปยังเซิร์ฟเวอร์ชั่วคราวซึ่งจะต้องกำหนดค่า IP Address ในช่อง Custom Address และ กำหนดค่า Port ในช่อง Port ดังรูปที่ 4.2



รูปที่ 4.2 การกำหนดค่า IP Address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องขออนุญาตทุกครั้งที่มีการนำไปใช้

เมื่อเข้าแอปพลิเคชัน PTT MIS for Life Station จะพบกับหน้าล็อกอินซึ่งถ้าทำการใส่ค่า Username และ Password ถูกต้องก็จะเข้าไปยังหน้ารายการเมนู ดังรูปที่ 4.3



รูปที่ 4.3 การล็อกอินสำเร็จ

ในกรณีที่มีการใส่ Username หรือ Password ผิดพลาดแอปพลิเคชันจะมีข้อความขึ้นมาแจ้งเตือนดังรูปที่ 4.4



รูปที่ 4.4 การล็อกอินไม่สำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายในหน้าเมนูจะประกอบด้วยเมนูหลัก 9 เมนูซึ่งเมนูต่างๆยกเว้นเมนู Log Report จะมีการทำงานคล้ายๆกันตัวอย่างเช่นในรูปที่ 4.6 จะเป็นหน้ารายการข้อมูลของแต่ละเมนู และรูปที่ 4.7 จะเป็นหน้าแสดงผลข้อมูลของแต่ละเมนู



รูปที่ 4.5 หน้ารายการเมนู

4.1.1 การเพิ่ม การแก้ไข และ การลบข้อมูล

แอปพลิเคชันอนุญาตให้ผู้ใช้สามารถทำการ เพิ่มข้อมูลในแต่ละเมนูได้โดยการ กดปุ่ม + ที่มุมขวาบนของหน้าจอ ซึ่งในแต่ละเมนูจะแสดงหน้าจอการเพิ่มข้อมูล ดังรูปที่ 4.8 สำหรับการแก้ไขข้อมูล และ การลบข้อมูลของข้อมูลในแต่ละเมนูนั้นสามารถทำได้โดยการกดปุ่ม Edit ซึ่งอยู่บนแถบขวาบนของหน้าจอที่แสดงข้อมูลในรูปที่ 4.7 ซึ่ง เมื่อกดปุ่ม Edit แต่ละเมนูจะแสดงหน้าจอให้ผู้ใช้ทำการแก้ไข หรือ ลบข้อมูล ดังรูปที่ (ใส่รูปรวมหน้าจอ การ Edit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นทรัพย์สินของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

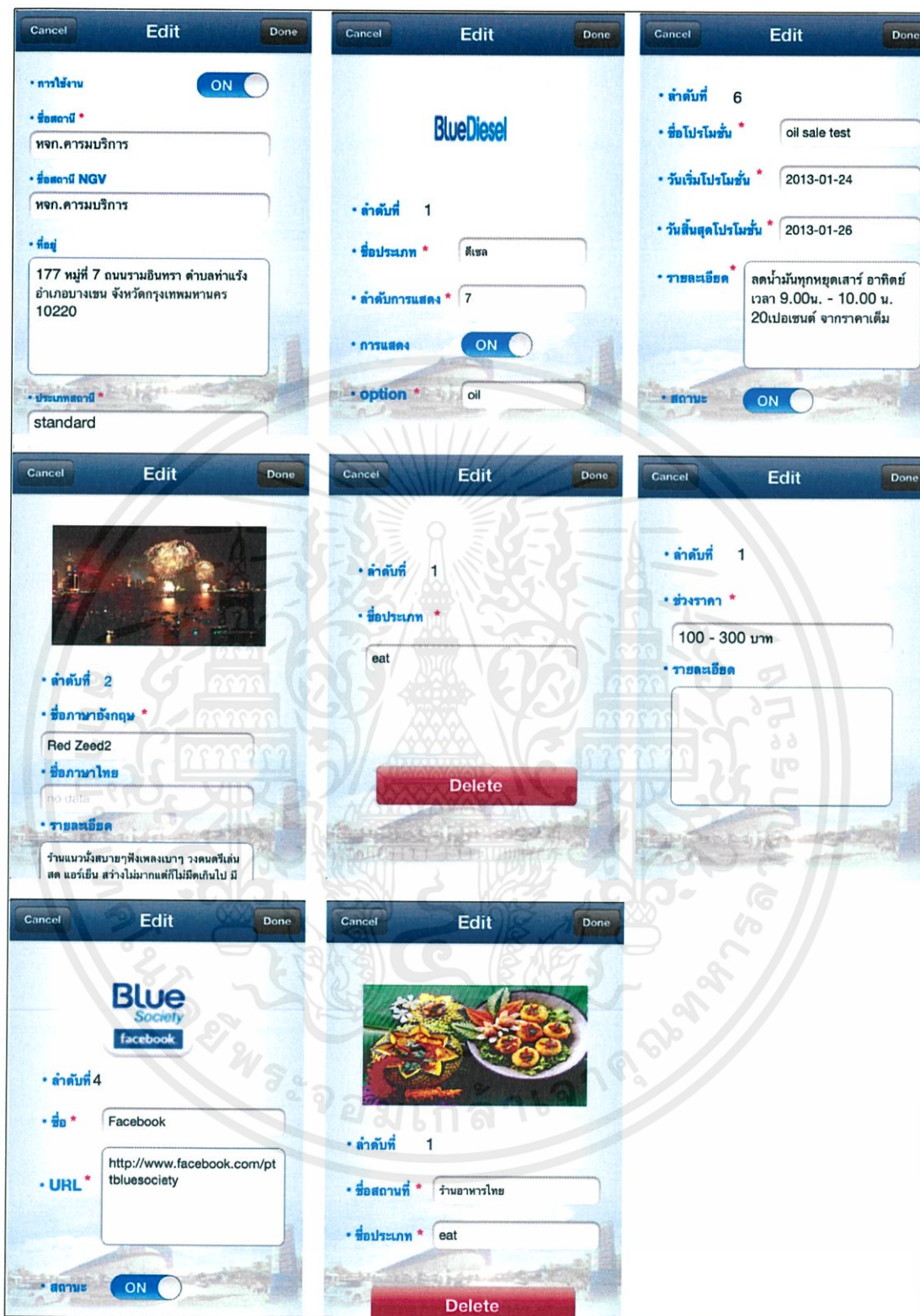
รูปที่ 4.6 หน้ารายการข้อมูลของแต่ละเมนู

<p>Gas Station Info Edit</p> <ul style="list-style-type: none"> • ลำดับที่: 1 • การใช้งาน: ใช้งานจริง • ชื่อสถานี: พท.การบริการ • ชื่อสถานี NGV: พท.การบริการ • ประเภทสถานี: standard • ที่อยู่: 177 หมู่ที่ 7 ถนนรามอินทรา ตำบลท่าเรือ อำเภอหนองแขม จังหวัดกรุงเทพมหานคร 10220 • จังหวัด: กทม. • วันที่เปิด: 2011-08-24 • เบอร์โทรศัพท์: 025109209 • เบอร์ FAX: 029436636 • Latitude: 13.8561 • Longitude: 100.629 • สถานะการทำงาน: เปิดจริง 	<p>Oil Type Info Edit</p> <p>BlueDiesel</p> <ul style="list-style-type: none"> • ลำดับที่: 1 • ชื่อประเภท: ดีเซล • ลำดับการแสดงผล: 7 • การแสดงผล: Show • option: oil 	<p>Promotion Info Edit</p>  <ul style="list-style-type: none"> • ลำดับที่: 6 • ชื่อโปรโมชั่น: oil sale test • วันเริ่มโปรโมชั่น: 2013-01-24 • วันสิ้นสุดโปรโมชั่น: 2013-01-26 • รายละเอียด: ลดน้ำมันทุกจุดสาขา อาทิศจักรเวลา 9.00น. - 10.00 น. 20เปอร์เซ็นต์ จากราคาเดิม • สถานะ: online
<p>Place Travel Info Edit</p>  <ul style="list-style-type: none"> • ลำดับที่: 2 • ชื่อภาษาอังกฤษ: Red Zeed2 • ชื่อภาษาไทย: no_data • รายละเอียด: ร้านเหล้ามีบรรยากาศดีเพลงเบาๆ รสน้ำดื่มอร่อย มีที่นั่งสำหรับนั่งพักผ่อนเป็นโซนๆ มีโต๊ะ 2 ชั้น และเป็นโต๊ะ กระจกมองประมาณครึ่งมีที่จอดรถและมีวิวภาพวิวและสวนข้างชาชามีไฟพุ่มแสงเป็นสีขาวอยู่ 	<p>Travel Type Info Edit</p> <ul style="list-style-type: none"> • ลำดับที่: 1 • ชื่อประเภท: eat 	<p>Price Range Info Edit</p> <ul style="list-style-type: none"> • ลำดับที่: 1 • ช่วงราคา: 100 - 300 บาท • รายละเอียด: no_data
<p>Travel Eat Type Info Edit</p>  <ul style="list-style-type: none"> • ลำดับที่: 1 • ชื่อสถานที่: ร้านอาหารไทย • ชื่อประเภท: eat 	<p>Other Application Info Edit</p> <p>Blue Society facebook</p> <ul style="list-style-type: none"> • ลำดับที่: 4 • ชื่อ: Facebook • URL: http://www.facebook.com/pttbluesociety • สถานะ: active 	

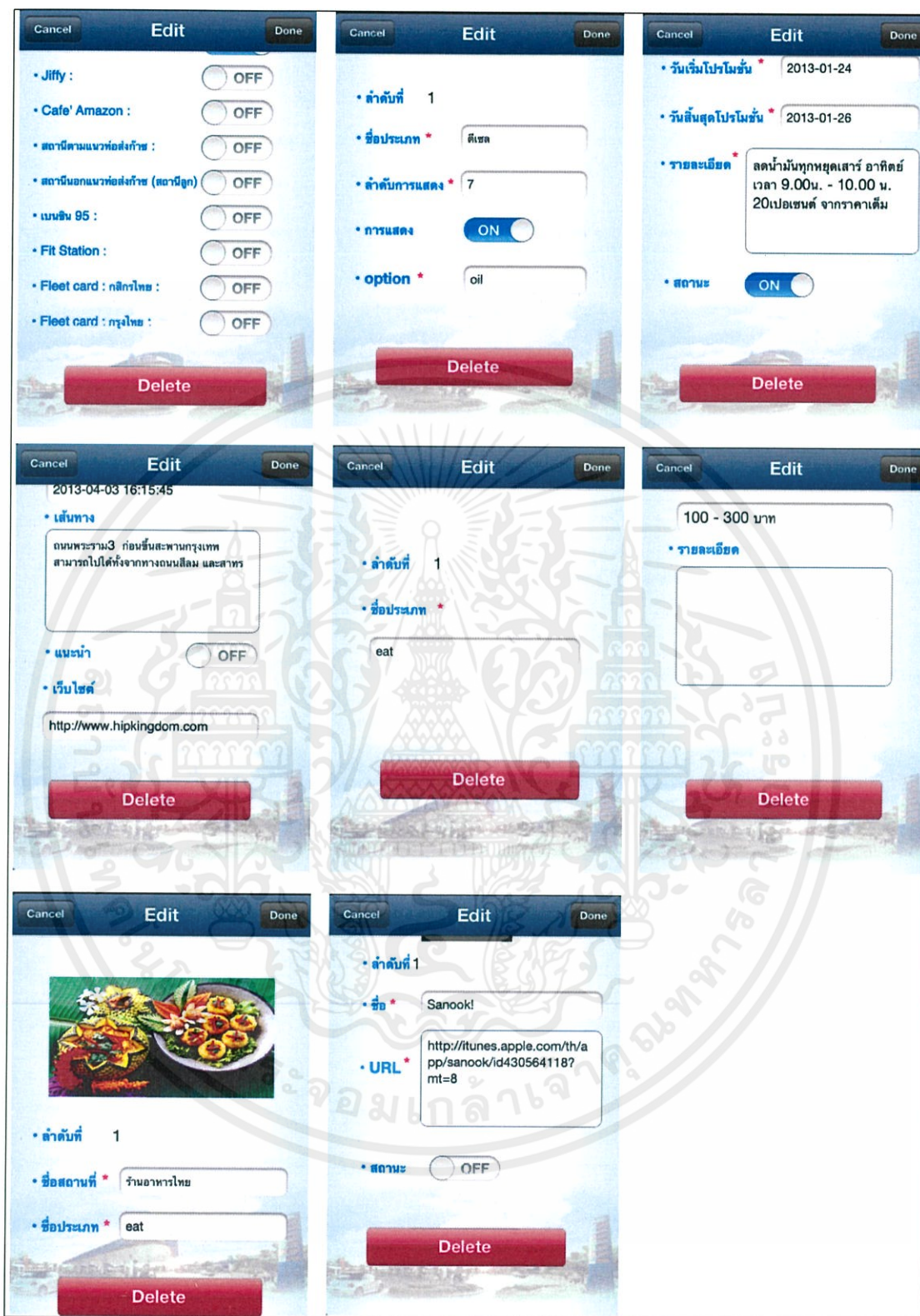
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.7 หน้าแสดงข้อมูลของแต่ละเมนู ให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.8 หน้าการเพิ่มข้อมูลของแต่ละเมนู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

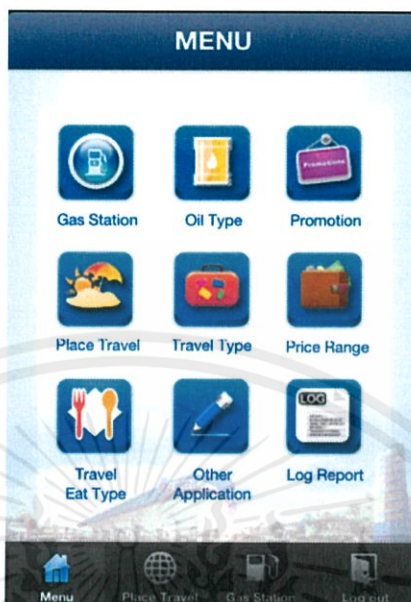


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานวิจัยการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 4.9 หน้าการแก้ไขข้อมูลของแต่ละเมนู
 ไม่ว่าจะแก้ไขข้อมูลใดๆ ก็ตาม อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



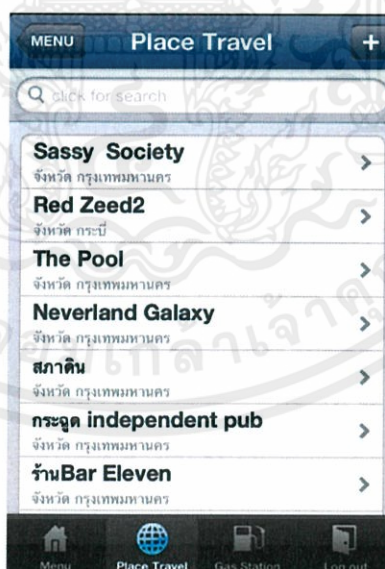
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่ควรเอาไปใช้ประโยชน์ด้านการค้า
รูปที่ 4.10 หน้าการลบข้อมูลของแต่ละเมนู
 ไม่ว่าจะพิมพ์ใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจากทั้ง 8 เมนูจะมีการทำงานคล้ายกัน ซึ่งในตัวอย่างการใช้งานต่อไปนี้จะขอ ยกตัวอย่างแสดงการใช้งานในส่วนของเมนู Place Travel



รูปที่ 4.11 หน้ารายการเมนู

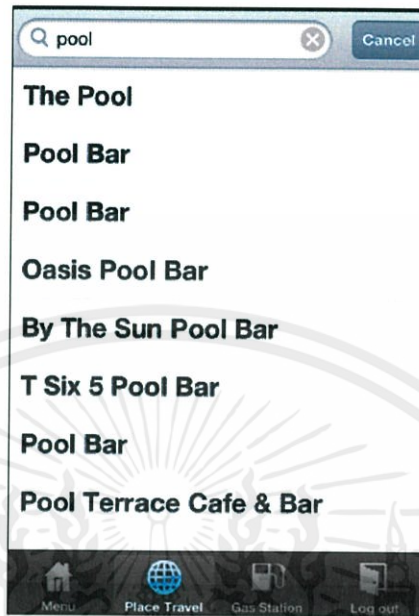
จากรูปที่ 4.11 เมื่อทำการกดไปที่เมนู Place Travel จะได้หน้าต่างดังรูปที่ 4.12 ซึ่งจะเป็น ข้อมูลชื่อและจังหวัดของสถานที่ท่องเที่ยวต่างๆ



รูปที่ 4.12 หน้ารายการข้อมูล Place Travel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากหน้ารายการข้อมูล Place Travel จะสามารถค้นหาข้อมูลเพื่อประหยัดเวลาในการค้นหาได้ดังตัวอย่างรูปที่ 4.13 จะเป็นการค้นหาสถานที่ที่มีคำว่า Pool ประกอบอยู่ด้วย



รูปที่ 4.13 การค้นหาข้อมูล

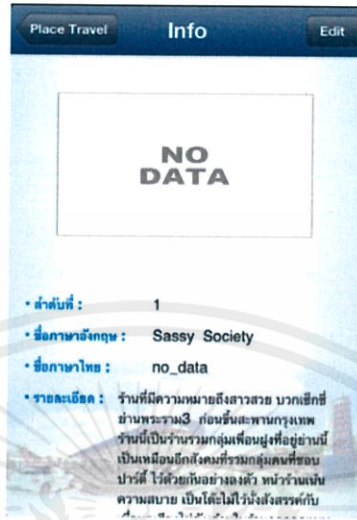
เมื่อทำการเลือกข้อมูลสถานที่ที่ต้องการก็จะแสดงข้อมูลในสถานที่นั้นๆ ในอีกหน้าหนึ่งดังรูปที่ 4.14



รูปที่ 4.14 หน้าแสดงข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานภายในหน่วยงานเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.14 ถ้าภายในสถานที่ท่องเที่ยวที่ไม่มีข้อมูลรูปภาพอยู่จะแสดงรูป NO DATA แทนดังรูปที่ 4.15



รูปที่ 4.15 หน้าแสดงข้อมูลโดยไม่มีข้อมูลรูป

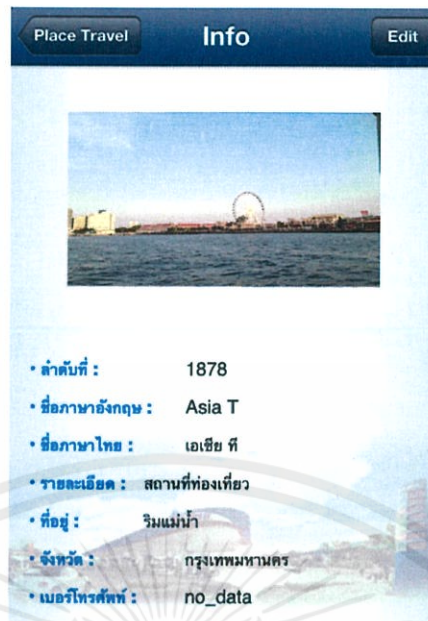
- การเพิ่มข้อมูล

จากหน้ารายการข้อมูลดังรูปที่ 4.12 เมื่อทำการกดปุ่ม + จะเป็นการย้ายไปหน้า Add เพื่อทำการเพิ่มข้อมูลใหม่เข้าไปยังฐานข้อมูล ดังรูปที่ 4.16



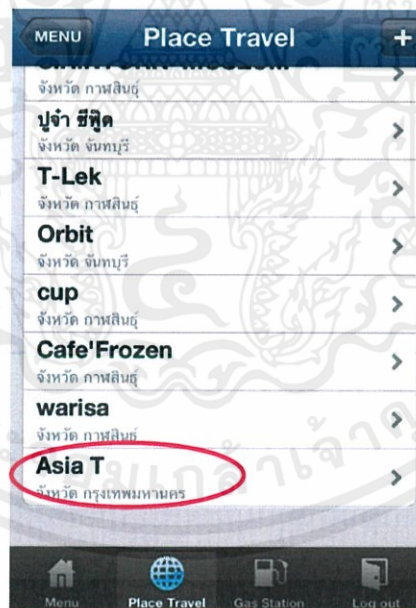
รูปที่ 4.16 หน้า Add ข้อมูล

ในการใส่ข้อมูลในหน้า Add นั้นจะทำเหมือนกันกับในหน้า Edit ทั้งหมด เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การเข้าถึงเอกสารที่เกี่ยวข้องกับนโยบายด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 หน้าแสดงข้อมูล

เมื่อทำการกลับมาที่รายการข้อมูลสถานที่ท่องเที่ยวก็จะเห็นได้ว่าข้อมูลได้ถูกเพิ่มมาแล้วดังรูปที่ 4.20

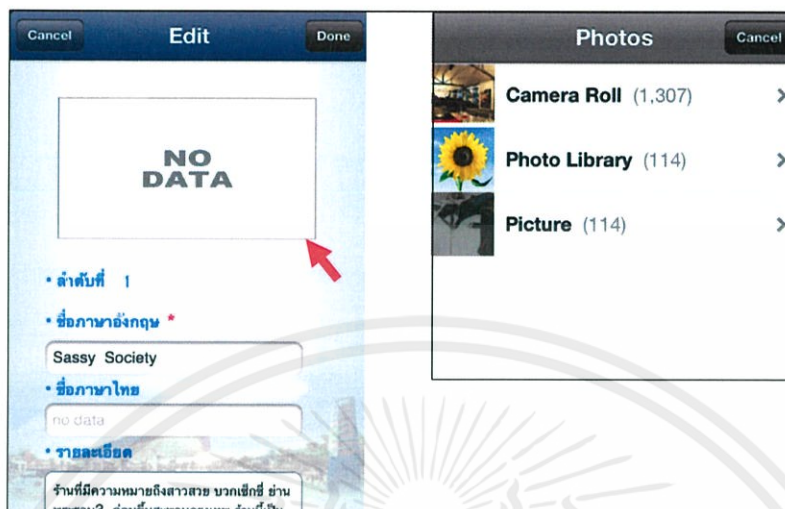


รูปที่ 4.20 หน้ารายการข้อมูล Place Travel

- การแก้ไขข้อมูล

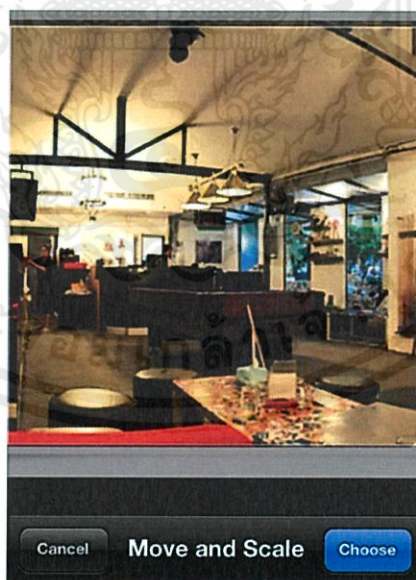
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
จากหน้าแสดงข้อมูลในรูปที่ 4.19 เมื่อทำการกดปุ่ม Edit จะสามารถทำการแก้ไขข้อมูลได้
ไม่ว่ากรณีใดๆทั้งสิ้น ยกเว้นให้มีเหตุผลเป็นสำคัญ และต้องอ้างอิงถึงเจ้าของเอกสารที่ทรงทั้งหมดที่เรานำไปใช้
อย่างเช่นในรูปที่ 4.21 จะเป็นการเพิ่มรูปเข้าไปในรายการข้อมูล วิธีการคือทำการกดที่รูปได้เลย ก็

จะสามารถเลือกรูปจากเครื่องมาใช้ได้ โดยรูปที่เลือกก่อนนำมาใช้งานนั้นสามารถ Crop ตามต้องการได้



รูปที่ 4.21 การเลือกรูปจากเครื่อง iPhone

เมื่อทำการเลือกรูปจากเครื่อง iPhone ได้แล้วจะมีให้ทำการ Crop รูปได้เลยซึ่งสามารถ Crop ได้ตามความต้องการ เมื่อทำการ Crop เสร็จแล้วก็ทำการกดปุ่ม Choose เพื่อนำรูปที่เลือกเข้าไปยังแอปพลิเคชันได้เลยดังรูปที่ 4.22 และ รูปที่ 4.23

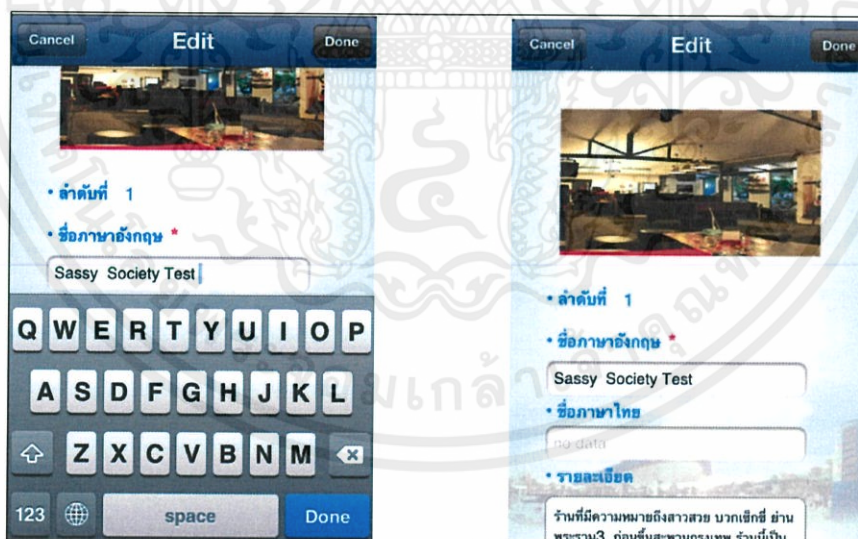


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในห้องเรียนเท่านั้น ไม่ควรนำออกไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



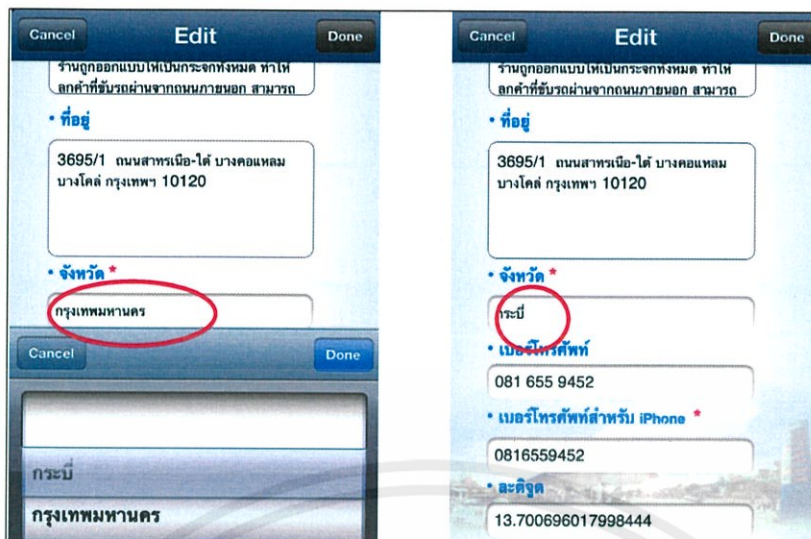
รูปที่ 4.23 หน้า Edit ข้อมูล

ในการแก้ไขข้อมูลนั้นสามารถทำได้โดยการคลิกที่ช่อง Text Field เพื่อทำการแก้ไขได้เลย ดังรูปที่ 4.24 โดยเมื่อทำการแก้ไขเสร็จแล้วให้ทำการกดปุ่ม Done ก็จะเป็นการซ่อนคีย์บอร์ด



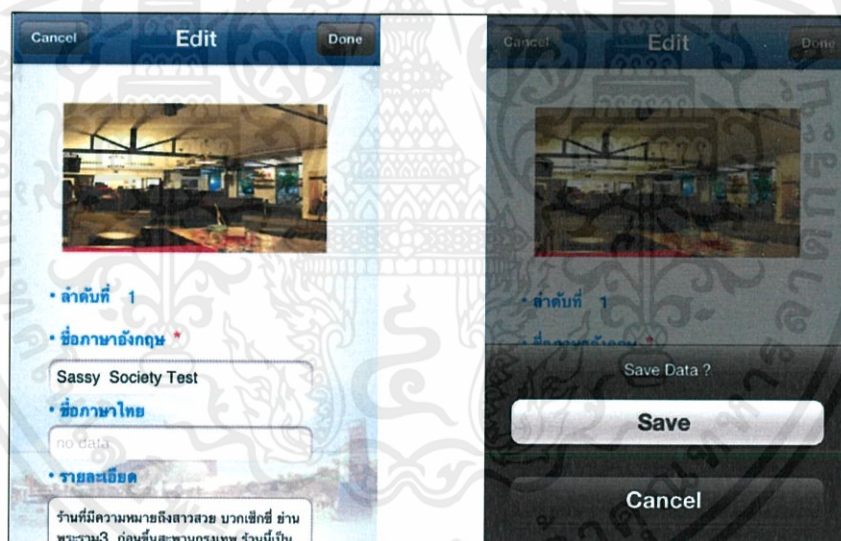
รูปที่ 4.24 การแก้ไขข้อมูล Text Field

ในส่วน of ข้อมูลที่เป็น Picker เมื่อทำการเลือกข้อมูลได้แล้วจะต้องทำการกดที่ปุ่ม Done เพื่อทำการยืนยันข้อมูลที่เลือกใส่ลงไปใน Text Field ดังรูปที่ 4.25 ซึ่งถ้ากดปุ่ม Cancel จะเป็นการซ่อน Picker



รูปที่ 4.25 การแก้ไขข้อมูล Picker

เมื่อทำการแก้ไขข้อมูลเสร็จหมดแล้วทำการกดปุ่ม Done เพื่อทำการบันทึกข้อมูล จะมี Action Sheet ขึ้นมาถามถึงความแน่ใจในการทำการบันทึกข้อมูลดังรูปที่ 4.26



รูปที่ 4.26 การบันทึกข้อมูล

หลังจากที่ทำการกดปุ่ม Save ในรูปที่ 4.26 จะมีการกรองข้อมูลถ้าหากลิ้มใส่ข้อมูลที่เป็น นั่นก็คือข้อมูลที่มี * กำกับไว้ จะทำการแจ้งเตือนให้กลับไปทำการกรอกข้อมูลในส่วนนั้นดัง รูปที่ 4.27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.27 การกรองข้อมูล

เมื่อไม่ลืมกรอกข้อมูลอะไรแล้วหลังจากทำการ Save ข้อมูลลงฐานข้อมูล ก็จะย้ายหน้าไปยังหน้าแสดงข้อมูลให้ทันทีเพื่อให้เห็นถึงข้อมูลที่ถูกแก้ไขเรียบร้อยแล้วดังรูปที่ 4.28



รูปที่ 4.28 หน้าแสดงข้อมูล

เมื่อทำการกลับมาที่รายการข้อมูลสถานที่ท่องเที่ยวก็จะเห็นได้ว่าข้อมูลได้ถูกแก้ไขแล้วดังเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาคือเท่านั้น ไม่อนุญาตให้เผยแพร่เชิงพาณิชย์การซ้ำไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

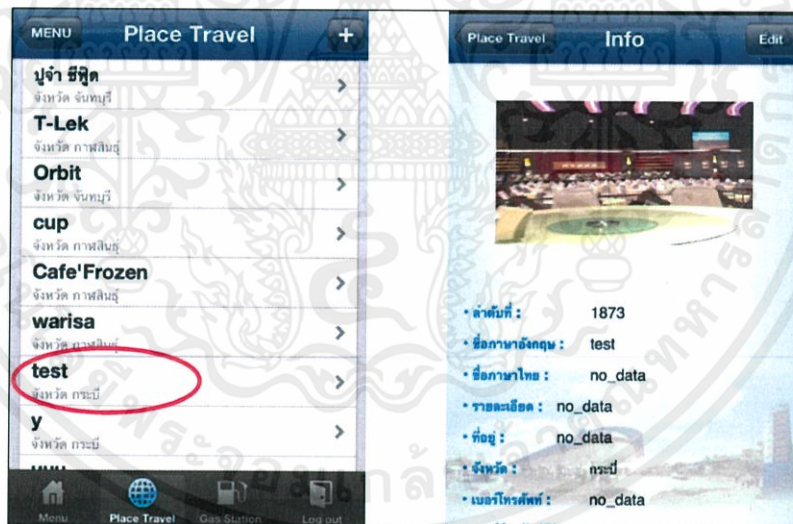
รูปที่ 4.29



รูปที่ 4.29 หน้ารายการข้อมูล Place Travel

- การลบข้อมูล

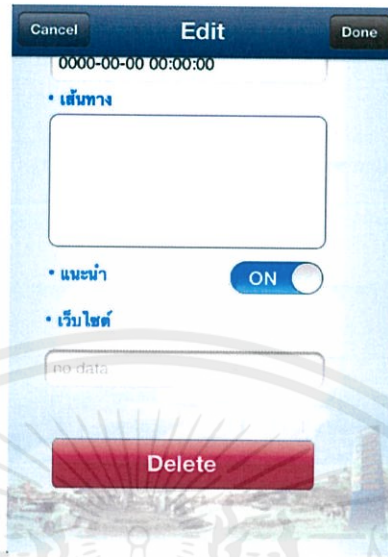
ในการลบข้อมูลนั้นจะทำโดยการเลือกข้อมูลที่จะทำการลบ ในหน้ารายการข้อมูลดังรูปที่ 4.30 จะเป็นตัวอย่างในการเลือกข้อมูลที่ชื่อว่า Test มาทำการลบ



รูปที่ 4.30 การเลือกข้อมูลที่จะลบ

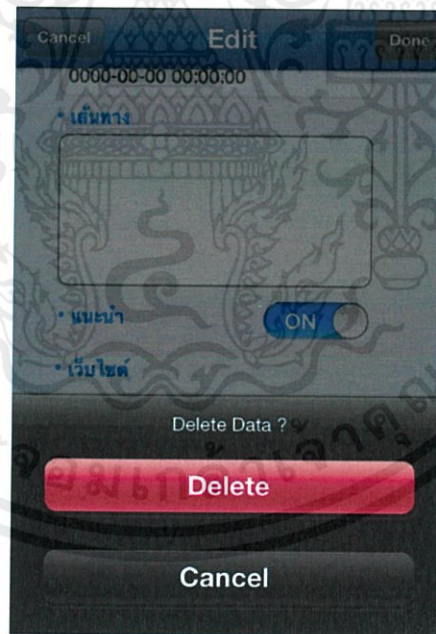
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือกข้อมูลที่จะลบได้แล้วนั้น ปุ่ม Delete จะอยู่ภายในหน้า Edit ซึ่งจะอยู่ด้านล่างสุดดังรูปที่ 4.31



รูปที่ 4.31 ปุ่ม Delete

เมื่อทำการกดปุ่ม Delete แล้วจะมี Action Sheet ขึ้นมาถามเพื่อความแน่ใจดังรูปที่ 4.32



รูปที่ 4.32 Action Sheet ปุ่ม Delete

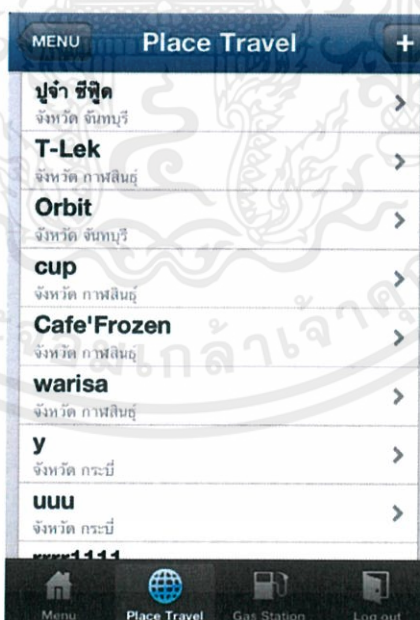
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการกด Delete จะย้ายมาที่หน้าแสดงข้อมูลแล้วแสดง Action Sheet ขึ้นมาแจ้งเตือนว่า Delete Complete เพื่อเป็นการบอกว่าข้อมูลนั้นได้ถูกลบออกจากฐานข้อมูลแล้วดังรูปที่ 4.33



รูปที่ 4.33 หน้าแสดงข้อมูล

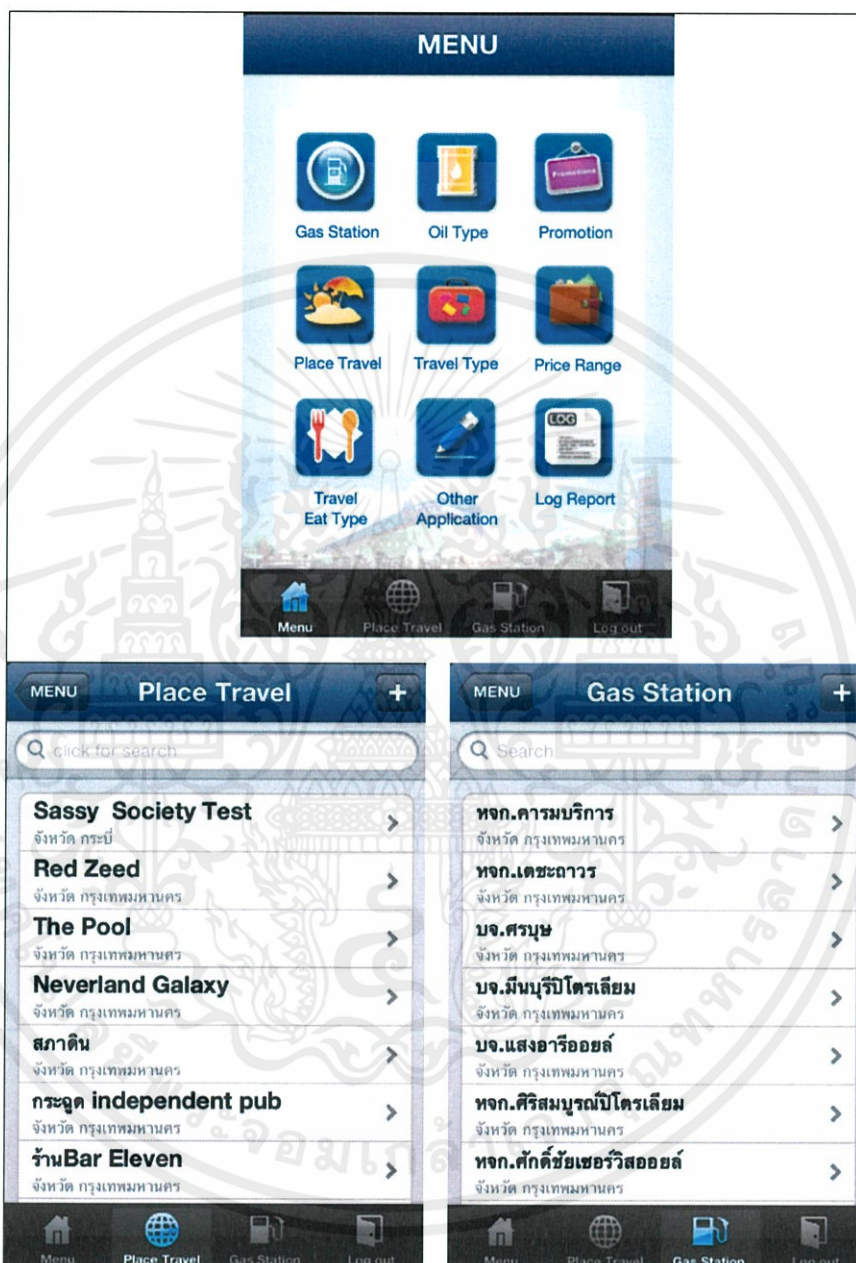
จากรูปที่ 4.33 เมื่อทำการกดปุ่ม OK ใน Action Sheet จะกลับมาที่หน้ารายการข้อมูล Place Travel ซึ่งจะเห็นได้ว่าข้อมูล Test ได้ถูกลบออกไปแล้วดังรูปที่ 4.34



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.34 หน้ารายการข้อมูล Place Travel ให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 การใช้ Tab Bar

Tab Bar จะใช้ในการย้ายไปหน้าที่มีการใช้บ่อยซึ่งจะประกอบด้วยหน้าเมนู หน้า Place Travel และ หน้า Gas Station โดยมีการใช้งานดังรูปที่ 4.35



รูปที่ 4.35 การเปลี่ยนหน้าโดยใช้ Tab Bar

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

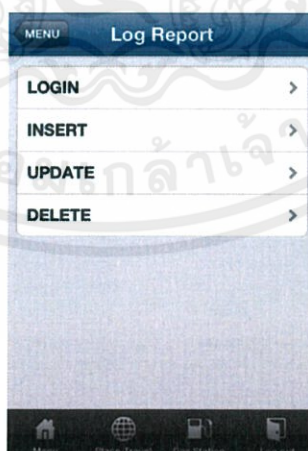
ส่วนปุ่ม Log out ที่อยู่ภายใน Tab Bar นั้นเมื่อกดไปแล้วจะทำการย้อนกลับมาที่หน้าล็อกอินดังรูปที่ 4.36 โดยการล็อกเอาทันนั้นจะเกิดขึ้นได้ในกรณีที่เมื่อกดปุ่ม Log out กดปุ่ม Home ของ iPhone หรือ เมื่อน้ำจอ Lock คับไปเองเพื่อเพิ่มความปลอดภัยมากยิ่งขึ้น



รูปที่ 4.36 การล็อกเอาทัน

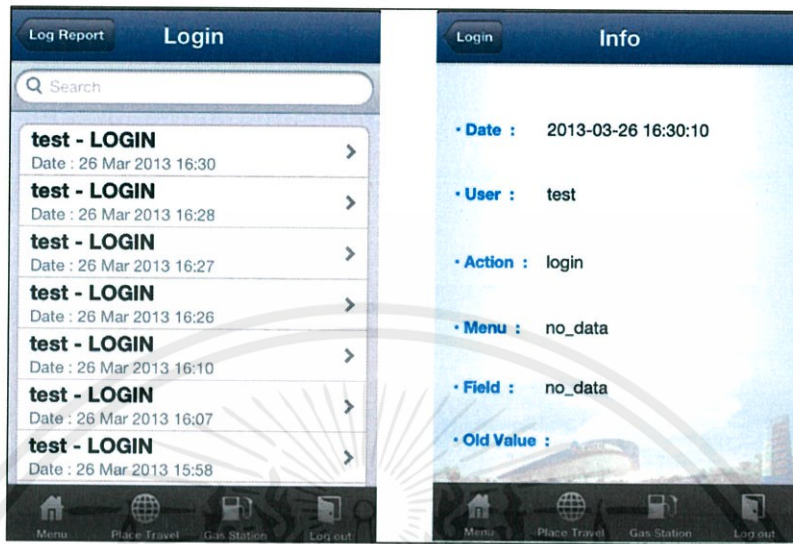
4.1.3 Log Report

เมนู Log Report จะเป็นการเรียกข้อมูลจากฐานข้อมูลมาคว่าผู้ดูแลระบบคนไหนการแก้ไขข้อมูลอะไรบ้างเนื่องจากทุกครั้งที่มีการล็อกอิน การเพิ่มข้อมูล การแก้ไขข้อมูล และการลบข้อมูลจะมีการเก็บข้อมูลการกระทำต่างเอาไว้ในฐานข้อมูล โดย Log Report จะมีการแยกหมวดหมู่ตาม Action ดังรูปที่ 4.37



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้รูปที่ 4.37 หน้าเมนู Log Report ญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LOGIN จะเป็นการดูว่าผู้ดูแลระบบคนไหนเข้ามาใช้งานบ้างและเข้ามาใช้เมื่อไรดังรูปที่ 4.38



รูปที่ 4.38 Log Report LOGIN

INSERT จะเป็นการดูว่าผู้ดูแลระบบคนไหนทำการเพิ่มข้อมูลอะไรบ้างดังรูปที่ 4.39



รูปที่ 4.39 Log Report INSERT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UPDATE จะเป็นการดูว่าผู้ดูแลระบบคนไหนทำการแก้ไขข้อมูลอะไรบ้างดังรูปที่ 4.40



รูปที่ 4.40 Log Report UPDATE

DELETE จะเป็นการดูว่าผู้ดูแลระบบคนไหนทำการลบข้อมูลอะไรบ้างดังรูปที่ 4.41



รูปที่ 4.41 Log Report DELETE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การตรวจสอบโค้ด

การตรวจสอบโค้ดเป็นการทดสอบการทำงานโดยใช้ซอฟต์แวร์ Acunetix ซึ่งใช้ในการทดสอบเว็บเซิร์ฟเวอร์โดยผลลัพธ์ที่ได้มีดังนี้ พบจุดบกพร่องทั้งหมด 39 จุด สามารถแบ่งได้เป็น 3 ระดับ คือ Medium 25% Low 17% และ Information 56%

4.2.1 ระดับ Medium

ระดับ Medium เป็นช่องโหว่ที่เกิดจากการกำหนดค่าต่างๆของเซิร์ฟเวอร์ที่ผิดพลาด การเขียนโค้ดที่เปิดช่องโหว่ให้ผู้บุกรุกเจาะระบบเข้ามาได้ โดยปัญหาที่พบในระดับนี้ได้แก่ Apache httpd Remote Denial of Service Apache httpOnly Cookie Disclosure Error message on page และ User credentials are sent in clear text

4.2.1.1 Apache httpd Remote Denial of Service

Apache httpd Remote Denial of Service เกิดจากการทำงานที่ผิดพลาดของ Apache httpd server ทำให้เกิดช่องโหว่ที่สามารถโจมตีด้วยวิธี Denial of Service ได้ วิธีการแก้ไขนี้สามารถทำได้โดยทำการอัปเดตตัว Apache HTTP Server ให้เป็นเวอร์ชัน 2.2.20 หรือเวอร์ชันใหม่กว่านั้น

4.2.1.2 Apache httpOnly Cookie Disclosure

Apache httpOnly Cookie Disclosure เป็นข้อบกพร่องที่เกิดใน Apache HTTP Server ตั้งแต่เวอร์ชัน 2.2.x จนถึง 2.2.21 ที่ไม่ได้มีการปกปิดข้อมูลส่วน header ของ request เมื่อเกิดข้อผิดพลาด (เช่น 400 page not found) ทำให้ข้อมูลถูกก็รั่วไหลได้ การแก้ไขสามารถทำได้โดย อัปเดต Apache HTTP Server ให้เป็นเวอร์ชัน 2.2.22 หรือเวอร์ชันใหม่กว่านั้น

4.2.1.3 Error message on page

Error message on page เมื่อหน้าเว็บเกิดข้อผิดพลาดจะทำการแสดงรายละเอียดของข้อผิดพลาดนั้น ทำให้ผู้ไม่หวังดีอาจใช้ข้อมูลเหล่านั้นทำการเจาะเข้ามาในระบบได้ ต้องทำการแก้ไขที่ตัวโค้ด เช่นการทำ exception หรือการตั้งค่า environment ให้ไม่ทำการแสดงรายละเอียดของข้อผิดพลาด

4.2.1.4 User credentials are sent in clear text

User credentials are sent in clear text ข้อบกพร่องนี้เกิดจากการที่ส่งข้อมูลที่เป็นความลับของผู้ใช้ผ่านทางทั่วไปที่ไม่มีการเข้ารหัส (HTTP) ซึ่งสามารถแก้ไขได้โดยการส่งข้อมูลผ่านช่องทางที่เข้ารหัส (HTTPS)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 ระดับ Low

ระดับ Low เป็นช่องโหว่ที่เกิดจากที่ไม่มีการเข้ารหัสข้อมูล หรือ ตำแหน่งไฟล์ (directory path) ถูกเปิดเผย โดยปัญหาที่พบในระดับนี้คือ Error page Path disclosure, Possible sensitive directories, Session token in URL

4.2.2.1 Error page Path disclosure

Error page Path disclosure เกิดจากการที่ผู้ไม่หวังดีทำการเรียกหน้าเว็บที่ไม่มีแล้วได้รับรายละเอียดของข้อผิดพลาด เมื่อทำเข้าไปเรื่อยๆผู้ไม่หวังดีอาจสามารถเดาถึงโครงสร้างของไฟล์ที่อยู่ในเซิร์ฟเวอร์ได้ แก้ไขโดยการกำหนดค่าเซิร์ฟเวอร์ไม่ให้ส่งรายละเอียดของข้อผิดพลาดให้กับผู้ใช้

4.2.2.2 Possible sensitive directories

Possible sensitive directories การสร้างไดเรกทอรีที่มีความสำคัญกับระบบ เช่น การสำรองข้อมูล ส่วนใช้งานของผู้ดูแลระบบ หรือข้อมูลในฐานข้อมูล ซึ่งสิ่งพวกนี้อาจไม่ได้เปิดเผยในเว็บไซต์ แต่ผู้ไม่หวังดีอาจจะเดาที่อยู่ของไดเรกทอรีเหล่านี้ได้ แก้ไขโดยจำกัดการเข้าถึงไดเรกทอรีเหล่านี้หรือนำออกไปจากเซิร์ฟเวอร์

4.2.2.3 Session token in URL

Session token in URL เกิดจากการที่เว็บแอปพลิเคชันมีการเก็บ session token และเก็บไว้ใน URL ซึ่ง URL อาจถูกทำการบันทึกไว้ทำให้ข้อมูลรั่วไหลได้ แก้ไขโดยใช้การเก็บข้อมูลด้วยคุกกี้แทน

4.2.3 ระดับ Informational alert

ระดับ Informational alert คือการแจ้งเตือนต่างๆไป เช่น เว็บอาจไม่ได้มีข้อมูลรั่วไหล หรือ อีเมลถูกเปิดเผย โดยพบข้อบกพร่องดังนี้ Broken links, Email address found, Password type input with auto-complete enabled, Possible internal IP address disclosure, Possible server path disclosure (Unix), Possible username or password disclosure

4.2.3.1 Broken links

Broken links คือลิงค์ที่ไม่สามารถใช้งานได้ เมื่อกดไปแล้วไม่มีหน้าดังกล่าว แก้ไขโดยลบลิงค์ดังกล่าวออก

4.2.3.2 Email address found

Email address found คือการแสดงที่อยู่อีเมล อาจทำให้โดนสแปมได้ แก้ไขโดยลบที่อยู่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
อีเมล
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3.3 Password type input with auto-complete enabled

Password type input with auto-complete enabled เกิดจากการที่ระบบมีการจำ ชื่อผู้ใช้ และ รหัสผ่าน อัตโนมัติ ทำให้อาจโดยผู้ไม่หวังดีสามารถขโมยข้อมูลเหล่านี้ผ่าน browser cache

4.2.3.4 Possible internal IP address disclosure

Possible internal IP address disclosure คือมีการเปิดเผย IP Address ในหน้าเว็บซึ่งอาจทำให้ข้อมูลโครงสร้างของเน็ตเวิร์กภายในถูกเปิดเผยด้วย แก๊ไขโดยไม่แสดง IP Address ในหน้าเว็บ

4.2.3.5 Possible server path disclosure (Unix)

Possible server path disclosure (Unix) มีการแสดงชื่อไดเรกทอรีที่ทำให้รู้ถึงโครงสร้างของเซิร์ฟเวอร์

4.2.3.6 Possible username or password disclosure

Possible username or password disclosure มีการแสดง ชื่อผู้ใช้ หรือ รหัสผ่านในหน้าเว็บ (ผลการตรวจอาจเป็น false positive คือ ไม่ได้ผิดพลาดแต่ระบบตรวจเจอ)

4.3 การทดสอบการใช้งาน

การทดสอบโดยการใช้งานจะเป็นการลองใช้แอปพลิเคชันเพื่อดูว่าจะได้ผลลัพธ์ที่ถูกต้องหรือไม่

4.3.1 ทดสอบการแสดงผลของข้อมูล

ในการทดสอบการแสดงผลของข้อมูล ได้มีการทดสอบการแสดงผลของข้อมูลรูปภาพ, ข้อมูลข้อความ และข้อมูลภายในของป๊อปอัพต่างๆภายในแอปพลิเคชัน

4.3.2 ทดสอบการจัดการข้อมูลภายในฐานข้อมูล

ในการทดสอบการจัดการข้อมูลภายในฐานข้อมูล ได้มีการตรวจสอบฐานข้อมูลว่ามีข้อมูลถูกต้องครบถ้วนตามที่ได้จัดการฐานข้อมูลผ่านทางแอปพลิเคชัน

4.3.3 ทดสอบการเชื่อมต่อแอปพลิเคชันเข้ากับเว็บเซอร์วิส

ในการทดสอบการเชื่อมต่อแอปพลิเคชันเข้ากับเว็บเซอร์วิส ได้มีการทดสอบการเชื่อมต่อแอปพลิเคชันเข้ากับเว็บเซอร์วิส IP Address อื่นเพื่อความถูกต้องในกรณีที่มีการเปลี่ยน IP Address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 บทสรุป

ในการพัฒนาแอปพลิเคชัน PTT MIS for Life Station บนระบบปฏิบัติการ iOS มีจุดประสงค์เพื่อให้การจัดการฐานข้อมูล (ที่สนับสนุนแอปพลิเคชัน PTT Life Station) จากสถานที่ภายนอกองค์กร ซึ่งการใช้งานระบบจัดการข้อมูลเดิม (เว็บแอปพลิเคชัน) บน iPhone เกิดปัญหาในการใช้งาน เนื่องจากขนาดของหน้าเว็บแอปพลิเคชันไม่พอดีกับหน้าจอของโทรศัพท์ซึ่งมีขนาดเล็กกว่าทำให้ต้องขยายดูหน้าจอเฉพาะจุดในการกรอกข้อมูลทำให้เสียเวลา แต่การใช้งานแอปพลิเคชัน PTT MIS for Life Station จะช่วยให้ผู้ใช้ใช้งานได้ง่ายขึ้นเนื่องจากส่วนติดต่อของผู้ใช้ถูกสร้างให้มีความเหมาะสมกับหน้าจอของโทรศัพท์

สำหรับการติดต่อระหว่างแอปพลิเคชันกับฐานข้อมูลจะทำผ่านเว็บเซอร์วิสแบบ REST ซึ่งเหตุผลที่สร้างเว็บเซอร์วิสแบบ REST นั้นก็เพราะว่า แนวโน้มของเทคโนโลยีเว็บเซอร์วิสแบบ REST กำลังจะมาแทนที่เว็บเซอร์วิสแบบ SOAP จึงทำให้ทางผู้พัฒนาต้องการพัฒนาเว็บเซอร์วิสแบบ REST เพื่อเป็นความรู้แก่ทางผู้พัฒนาเอง

5.2 ข้อจำกัดการพัฒนาแอปพลิเคชัน PTT MIS for Life Station

ข้อจำกัดในการพัฒนาแอปพลิเคชัน PTT MIS for Life Station คือเป็นการพัฒนาแอปพลิเคชันที่ต้องทำการบนโปรแกรม Xcode ซึ่งมีให้ใช้เฉพาะระบบปฏิบัติการ OS X บนเครื่อง Macintosh เท่านั้น นอกจากนั้นทางผู้พัฒนาไม่เคยมีประสบการณ์ในการพัฒนาบนแพลตฟอร์มนี้มาก่อนจึงทำให้ต้องใช้เวลาในการศึกษานาน จึงจะสามารถทำการพัฒนาได้

ส่วนข้อจำกัดในการพัฒนาเว็บเซอร์วิสก็เป็นเช่นเดียวกับการพัฒนา iOS แอปพลิเคชัน คือผู้พัฒนาไม่มีความรู้ในการทำเว็บเซอร์วิสแบบ REST โดยนอกจากผู้พัฒนาต้องศึกษาการพัฒนาเว็บเซอร์วิสแบบ REST แล้วผู้พัฒนายังต้องทำการศึกษาเทคโนโลยีที่จะใช้พัฒนาด้วย

และสุดท้าย ข้อจำกัดในเรื่องฐานข้อมูล เนื่องจากเป็นฐานข้อมูลที่ทางบริษัทมีอยู่แล้ว ทำให้ผู้พัฒนาต้องศึกษาโครงสร้างของฐานข้อมูลเพื่อที่จะสามารถจัดการข้อมูลได้ถูกต้องตามระบบเดิม (เว็บแอปพลิเคชัน) และยังมีปัญหาในกรณีของโครงสร้างของฐานข้อมูลที่ถูกสร้างมาอย่างผิดวิธี โดยส่วนนี้ผู้พัฒนาเองก็ไม่สามารถแก้ไขได้ ต้องหาจัดการข้อมูลให้ถูกต้องเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ข้อเสนอแนะ

- ควรมีการพัฒนาบนระบบปฏิบัติการมือถืออื่นๆ เช่น Andriod เพื่อให้ครอบคลุมกับทุก Platform มากยิ่งขึ้นเนื่องจากผู้ดูแลระบบอาจจะใช้มือถือหลายระบบปฏิบัติการ

5.4 งานในอนาคต

- การเพิ่มข้อมูลของโรงพยาบาลต่างๆพร้อมเส้นทางที่เชื่อมไปยัง Google Map เพื่อรองรับในอนาคตที่อาจจะมีการเพิ่มฟังก์ชันกรณีฉุกเฉินในแอปพลิเคชัน PTT Life Station



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] Anonymous. (2011). MobileApplication. สืบค้นเมื่อ 30 มีนาคม 2013, จาก http://www.treconwebsite.com/index.php?option=com_content&view=category&id=50&Itemid=90
- [2] Anonymous. (2013). BYOD คือ. สืบค้นเมื่อ 23 เมษายน 2013, จาก <http://th.wikipedia.org/wiki/BYOD>
- [3] takato. (2011). Xcode คือ. สืบค้นเมื่อ 30 มีนาคม 2013, จาก <http://takato.exteen.com/20110826/xcode>
- [4] Anonymous. (2013). Objective-C คือ. สืบค้นเมื่อ 30 มีนาคม 2013, จาก <http://th.wikipedia.org/wiki/ภาษาอ็อบเจกทีฟ-ซี>
- [5] Anonymous. (2010). PHP คือ. สืบค้นเมื่อ 30 มีนาคม 2013, จาก <http://www.hellomyweb.com/index.php/main/content/135>
- [6] ภาณุพงศ์ ปัญญาดี. (2006). Appserv คือ. สืบค้นเมื่อ 30 มีนาคม 2013, จาก <http://www.appservnetwork.com/modules.php?name=Content&pa=showpage&pid=3>
- [7] Hatem Hamad, Motaz Saad, and Ramzi Abeb. (2010). Performance Evaluation of RESTful Web Services for Mobile Devices, paper presented in *the International Arab of e-Technology*, Vol.1, No.3, January 2010
- [8] Anonymous. (2012). Restkit. สืบค้นเมื่อ 30 มีนาคม 2013, จาก www.raywenderlich.com/13097/intro-to-restkit-tutorial
- [9] Anonymous. (ไม่ปรากฏปีพิมพ์). Zend Framework. สืบค้นเมื่อ 30 มีนาคม 2013, จาก <http://framework.zend.com/manual/1.12/en/introduction.overview.html>
- [10] sharecodetips. (2011). Json คือ. สืบค้นเมื่อ 30 มีนาคม 2013, จาก <http://sharecodetips.wordpress.com/>
- [11] Anonymous. (2013). Acunetix is. สืบค้นเมื่อ 30 มีนาคม 2013, จาก <http://www.acunetix.com/vulnerability-scanner/wvsmanual.pdf>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



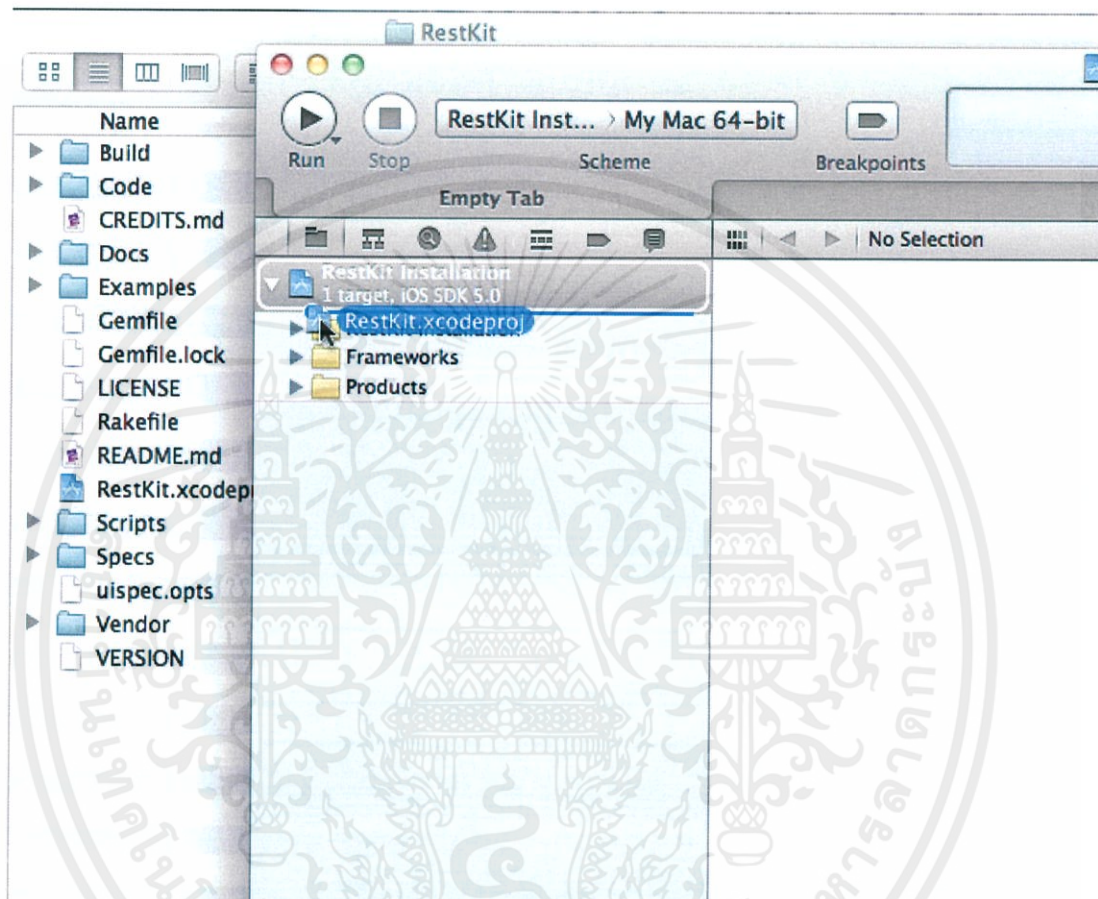
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

1. การติดตั้ง RestKit version 0.10x ใน Xcode 4.x

การติดตั้ง RestKit ติดตั้งได้โดยการลาก file RestKit.xcodeproj มาวางใส่ภายใน Project Navigator ดังรูปที่ ก-1 เพื่อทำการติดตั้ง Restkit ลงใน Project

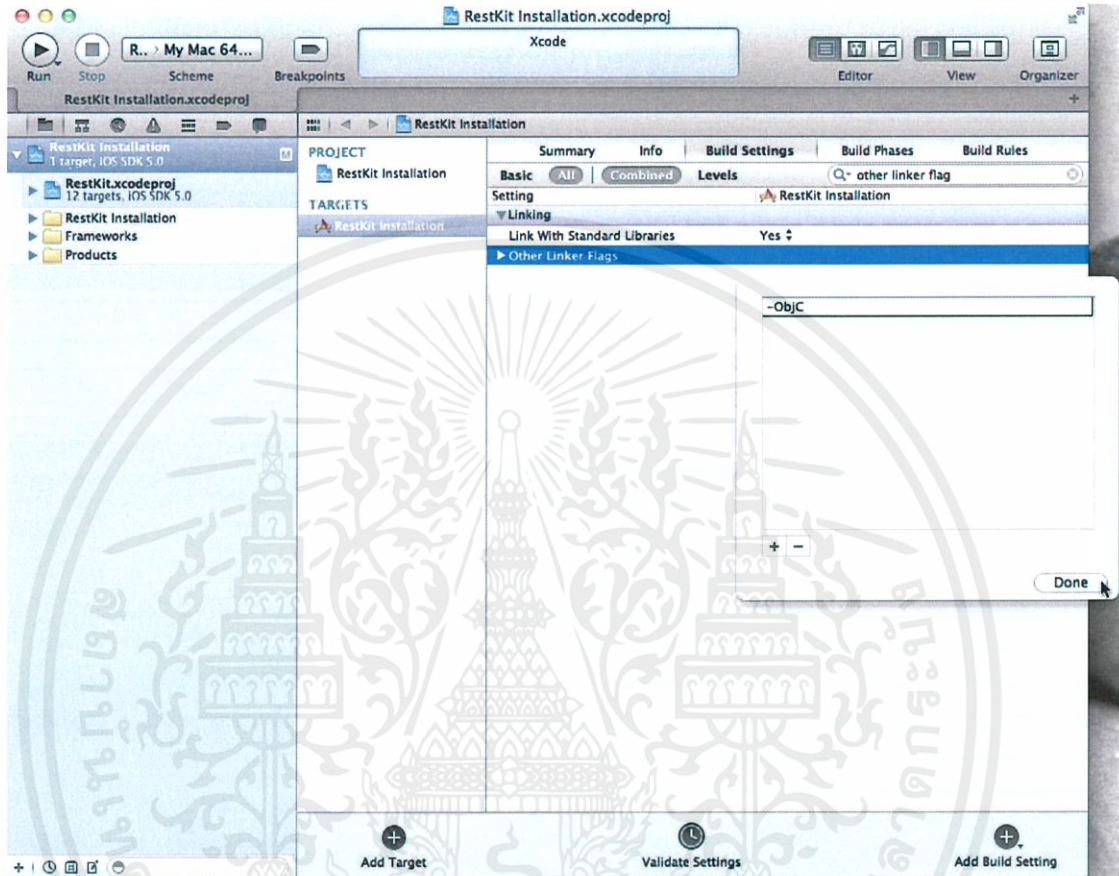


รูปที่ ก-1 การติดตั้ง RestKit ใน Project

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การตั้งค่า RestKit

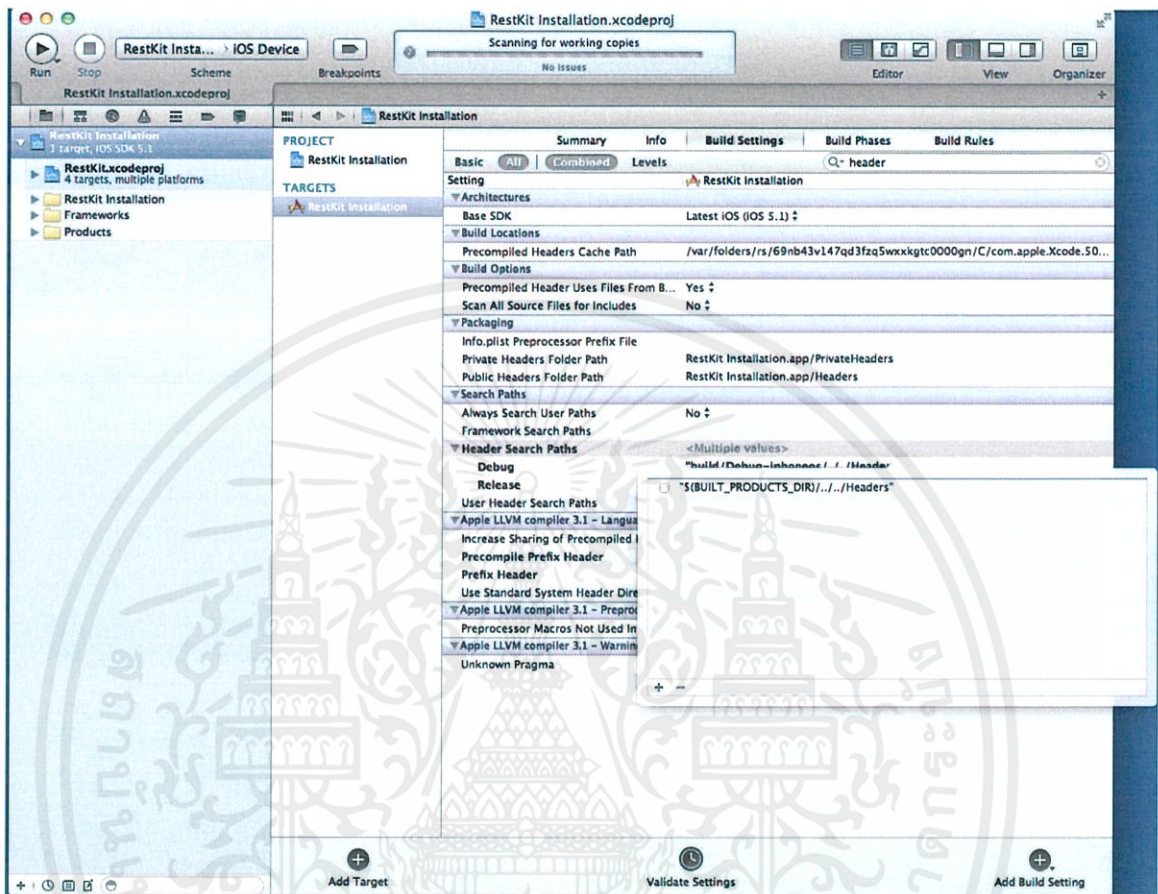
ภายใน Project Navigator ทำการเปิด Project จากนั้นไปในส่วนของ Targets เพื่อทำการตั้งค่าโดยไปที่ Build Setting และทำการค้นหาหัวข้อ “ Other Linker Flags ” จากนั้นทำการใส่ค่า -Objc กับ -all_load ดังรูปที่ ก-2



รูปที่ ก-2 การตั้งค่าภายใน Other Linker Flags

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากใส่ค่า Other Linker Flags เสร็จแล้ว จากนั้นทำการค้นหา “Header Search Paths” แล้วทำการใส่ค่า “\$(BUILT_PRODUCTS_DIR)/../Headers” ลงไปโดยใส่ค่า double quotes ลงไปด้วยดังรูปที่ ก-3

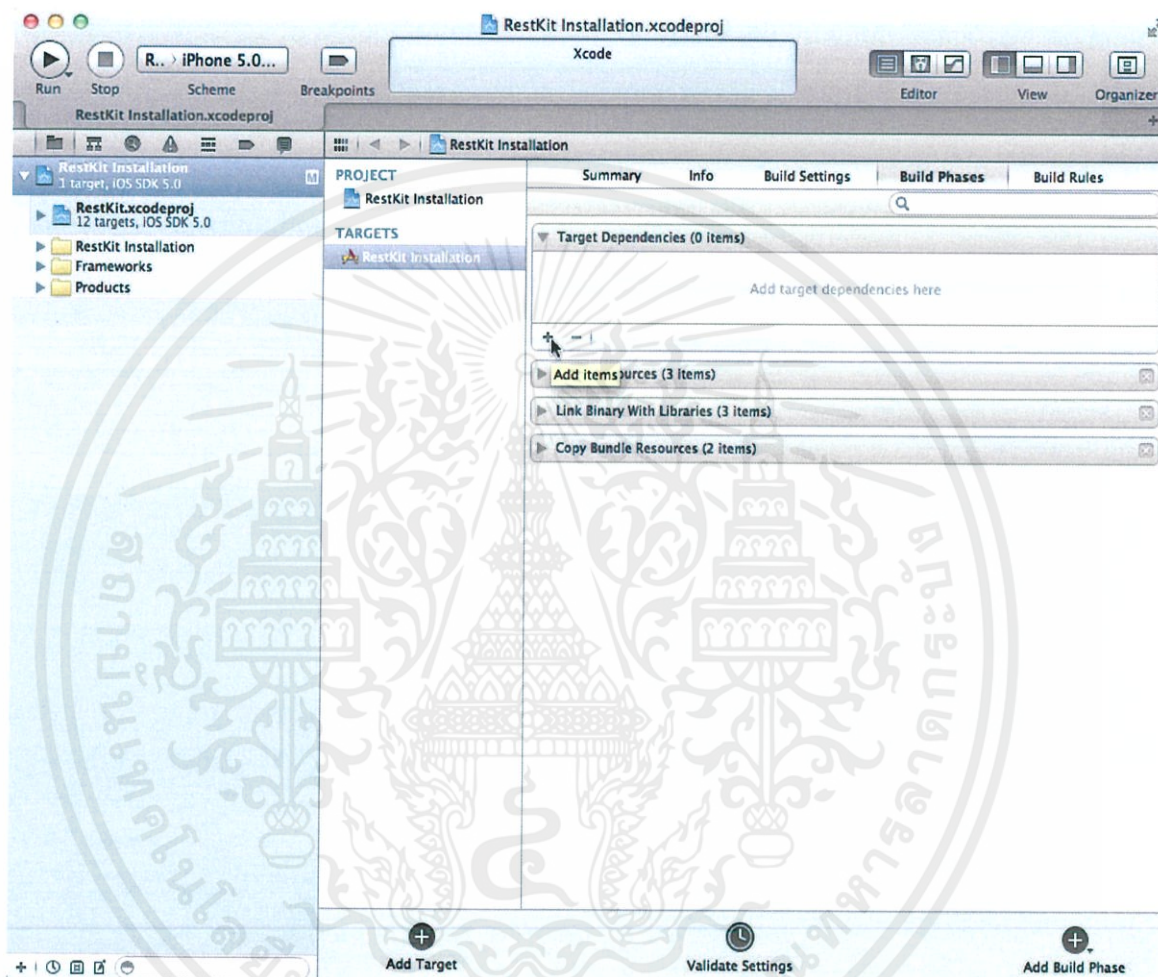


รูปที่ ก-3 การตั้งค่าภายใน Header Search Paths

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

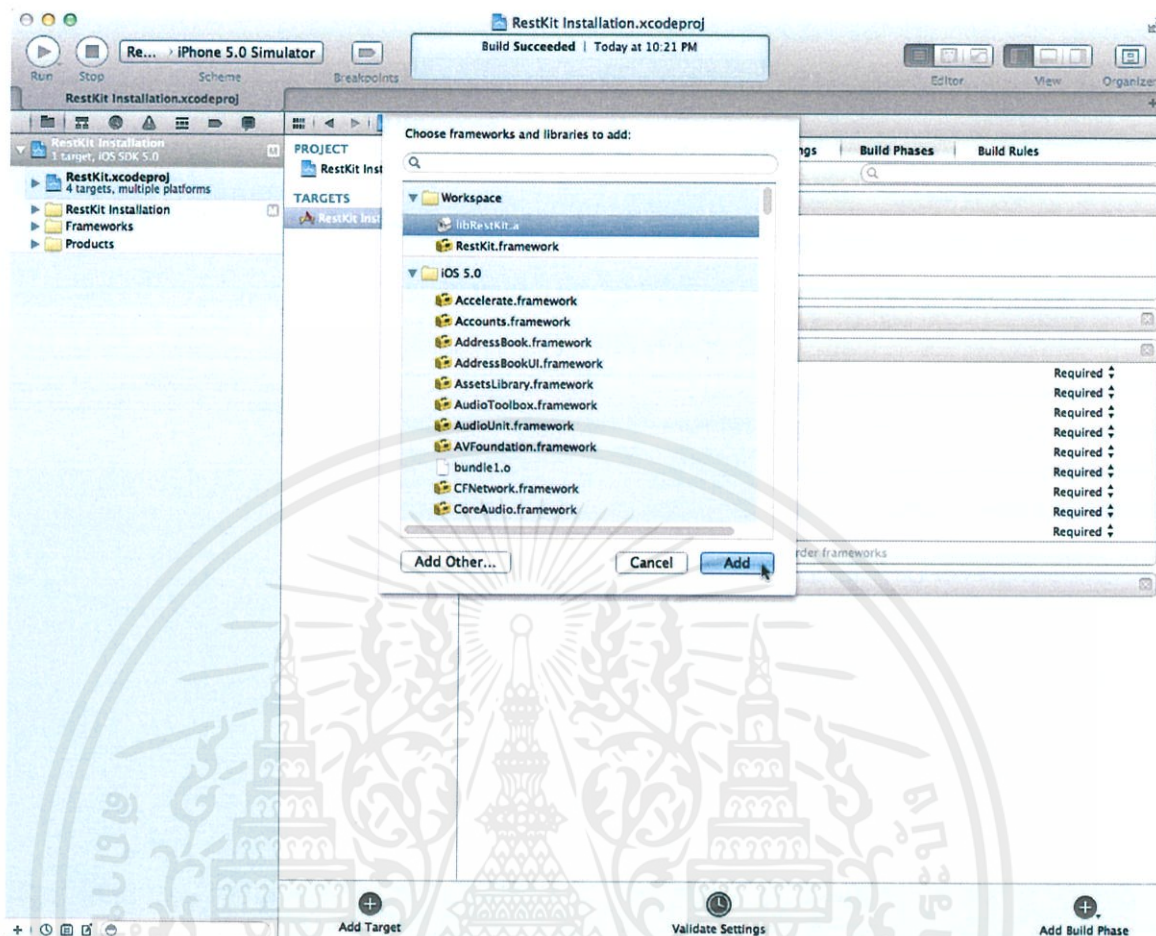
หลังจากทำการกำหนดค่าภายใน Header Search Paths เสร็จแล้วก็ทำการเปลี่ยนไปตั้งค่าจาก Build Setting เป็น Build Phases ภายในหัวข้อ Target Dependencies โดยทำการคลิก + เพื่อทำการเพิ่มค่าที่อยู่ภายใน Workspace ได้ดังรูปที่ ก-4 และ ก-5 ซึ่ง

- libRestKit.a เป็น RestKit สำหรับ iOS static library หรือ
- RestKit.framework เป็น RestKit สำหรับ OS X framework เลือกอย่างใดอย่างหนึ่ง



รูปที่ ก-4 การตั้งค่าภายใน Target Dependencies

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

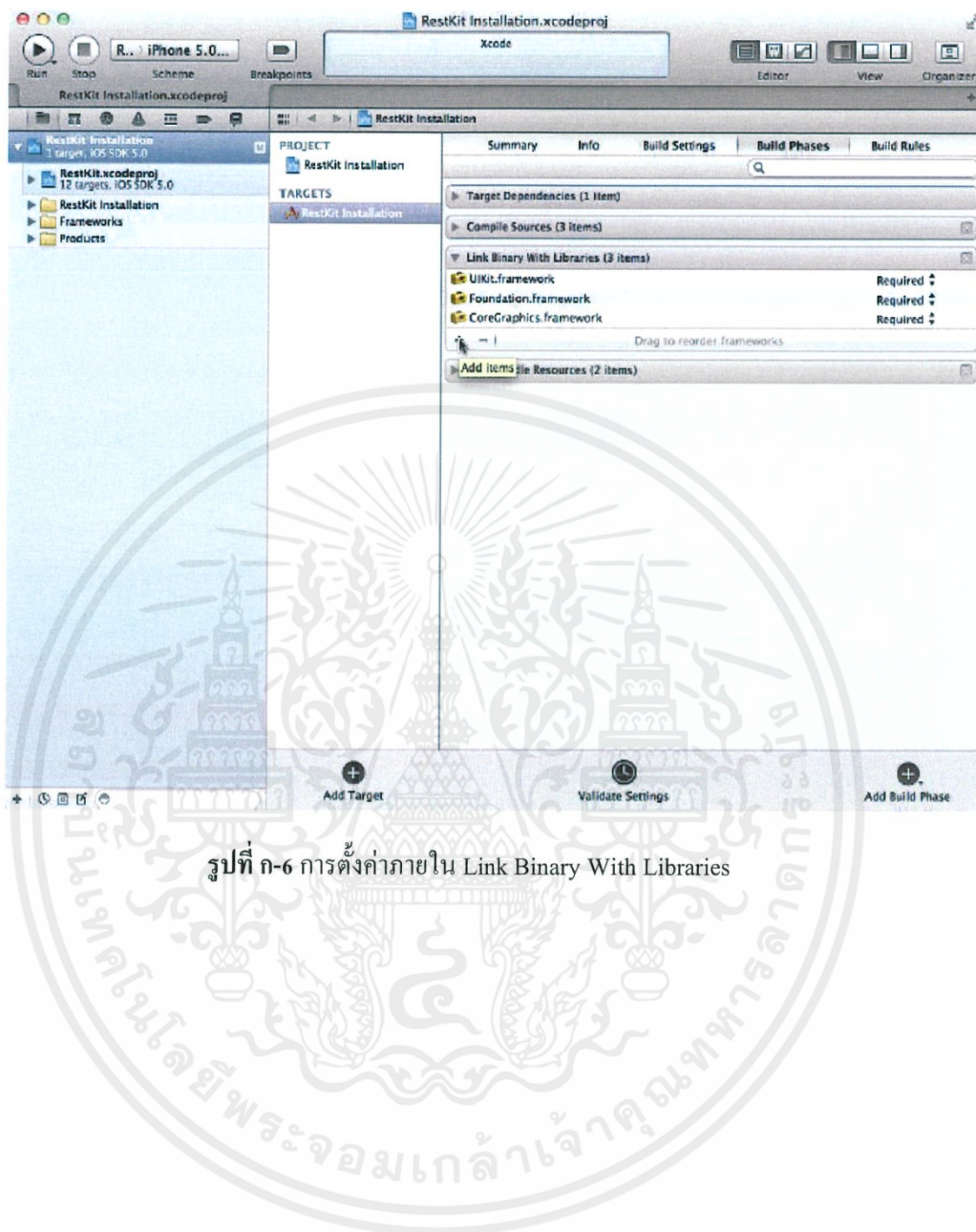


รูปที่ ก-5 การตั้งค่าภายใน Target Dependencies

หลังจากที่ทำการตั้งค่าในส่วนของ Target Dependency เสร็จแล้วจากนั้นไปทำการตั้งค่าภายในส่วนของ Link Binary With Libraries ดังรูปที่ ก-6 และ ก-7 โดยทำการเพิ่มค่า Frameworks ดังต่อไปนี้

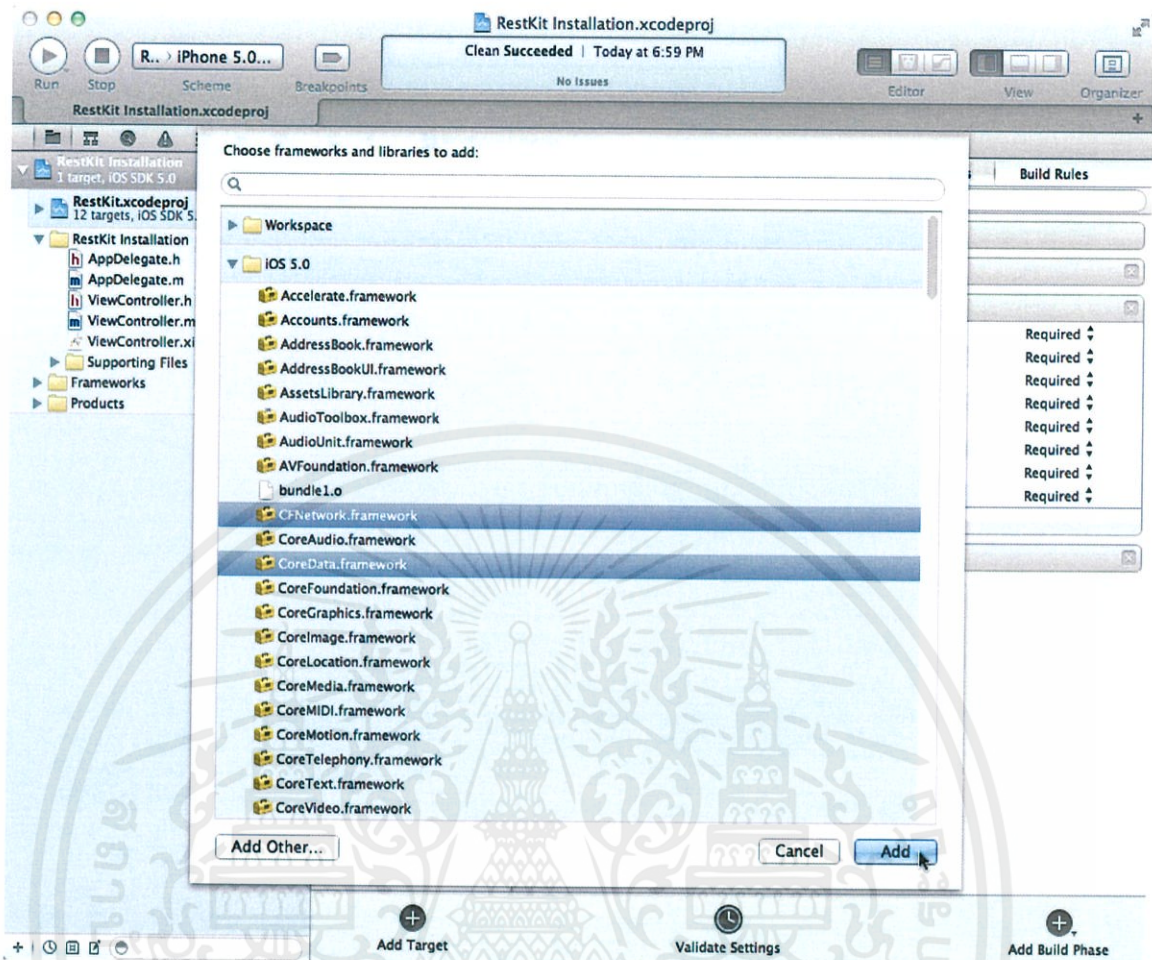
- CFNetwork.framework สำหรับ iOS
- CoreData.framework
- MobileCoreServices.framework สำหรับ iOS
- Security.framework
- SystemConfiguration.framework
- QuartzCore.framework สำหรับ iOS
- libxml2.dylib

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก-6 การตั้งค่าภายใน Link Binary With Libraries

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก-7 การตั้งค่าภายใน Link Binary With Libraries

3. การใช้งาน RestKit

การเรียกใช้งาน RestKit ให้ทำการใส่ `#import <RestKit/RestKit.h>` ไปที่ส่วนของ Header ซึ่งจะทำให้สามารถเรียกใช้ฟังก์ชันของ Restkiy ได้เรียบร้อยแล้ว

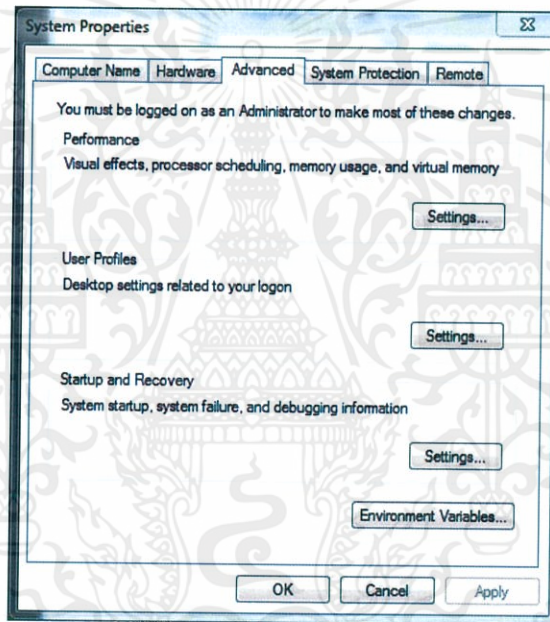
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

การติดตั้ง Zend เฟรมเวิร์ค ไลบรารี

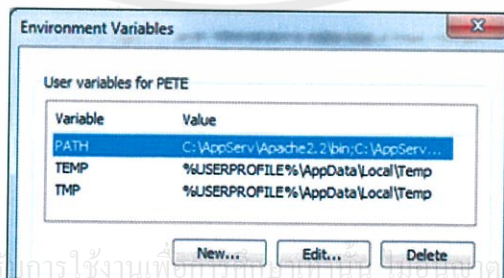
การติดตั้ง Zend เฟรมเวิร์คนี้เป็นการติดตั้งบน Apache server โดยใช้ระบบปฏิบัติการวินโดวส์

1. ทำการดาวน์โหลด Zend framework เวอร์ชันล่าสุดจากทางเว็บไซต์ <http://framework.zend.com/downloads/latest>
2. ทำการแตกไฟล์เตอร์ที่ได้ดาวน์โหลดมาและทำการติดตั้ง command line ของ Zend เฟรมเวิร์ค
- 2.1 ถ้าเป็นระบบปฏิบัติการวินโดวส์ ให้ทำการเพิ่ม path ของ zf.bat โดยไปที่ Control Panel -> System -> Advanced system settings -> Environment Variables ดังรูปที่ ข-1



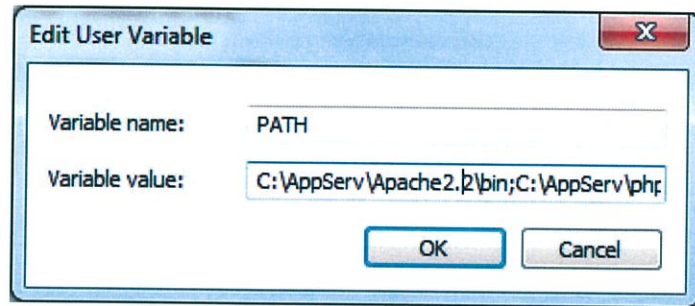
รูปที่ ข-1 System Properties

- 2.2 ทำการคลิกที่ PATH จากนั้นกดปุ่ม Edit ในรูป ข-2 แล้วจะพบกับหน้าต่างดังรูปที่ ข-3



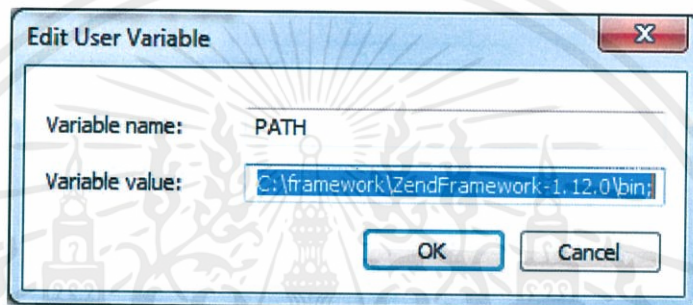
รูปที่ ข-2 Environment Variables

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาด้านนี้ ไม่ใช่มุ่งสู่ตลาดหน้าไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกหรือเผยแพร่เอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข-3 Edit User Variable

- 2.3 ใส่ Path ที่อยู่ของไฟล์ zf.bat ต่อท้าย path เดิม แล้วกดปุ่ม OK ตัวอย่างของ path เช่น
C:\framework\ZendFramework-1.12.0\bin;



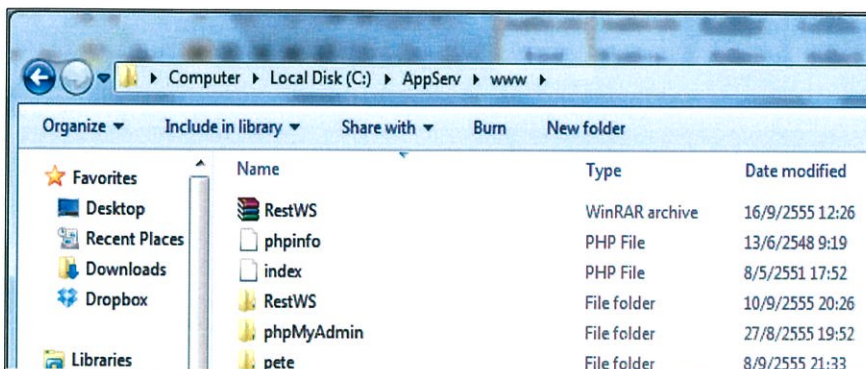
รูปที่ ข-4 Edit User Variable

- 2.4 ถ้าเป็นระบบปฏิบัติการที่เป็น Unix-Based สามารถใช้คำสั่งผ่าน command line ของระบบปฏิบัติการได้เลย โดยใช้คำสั่งดังนี้ `alias zf.sh=path/to/ZendFramework/bin/zf.sh`.
3. ทำการสร้าง Project โดยใช้ command line ดังนี้ `zf create project name-of-project` ตัวอย่างเช่น `zf create project RestWS` (โดยจะต้องเรียกใช้คำสั่งนี้ที่โฟลเดอร์ที่ใช้ในการสร้าง Project ดังรูปที่ ข-5

```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Users\PETE>cd ../../
C:\>cd AppServ/www
C:\AppServ\www>zf createproject RestWS
```

รูปที่ ข-5 การสร้าง Project

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข-6 Project ที่สร้างจาก Zend เฟรมเวิร์ค

ในกรณีที่ต้องการดู Command ทั้งหมดของ Zend เฟรมเวิร์คเราสามารถให้คำสั่ง `zf` เพื่อดูได้

ดั่งรูปที่ ข-7

```

Administrator: C:\Windows\system32\cmd.exe

C:\AppServ\www>zf
An Error Has Occurred
An action and provider is required.

Zend Framework Command Line Console Tool v1.12.0
Usage:
  zf [--global-opts] action-name [--action-opts] provider-name [--provider-opts]
  [provider parameters ...]
  Note: You may use "?" in any place of the above usage string to ask for more
  specific help information.
  Example: "zf ? version" will list all available actions for the version prov
  ider.

Providers and their actions:
  Version
  zf show version mode[=mini] name-included[=1]
  Note: There are specialties, use zf show version.? to get specific help on t
  hem.

  Config
  zf create config
  zf show config
  zf enable config
  Note: There are specialties, use zf enable config.? to get specific help on
  them.
  zf disable config
  Note: There are specialties, use zf disable config.? to get specific help on
  them.

  Phpinfo
  zf show phpinfo

  Manifest
  zf show manifest

  Profile
  zf show profile

  Project
  zf create project path name-of-profile file-of-profile
  zf show project
  Note: There are specialties, use zf show project.? to get specific help on t
  hen.
  
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่ออกเสียงเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ ข-7 คำสั่งของ Zend เฟรมเวิร์ค
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากทำการสร้าง Project เสร็จแล้วจะเห็น โครงสร้างของ Project ดังรูปที่ ข-8

RestWS

```

|-- application
| |-- Bootstrap.php
| |-- configs
| | `-- application.ini
| |-- controllers
| | |-- ErrorController.php
| | `-- IndexController.php
| |-- models
| `-- views
|   |-- helpers
|   `-- scripts
|-- error
|   |-- error.phtml
|   `-- index
|       |-- index.phtml
|-- library
|-- public
|   |-- .htaccess
|   |-- index.php
|-- tests
|   |-- application
|   |-- bootstrap.php
|   |-- library
|   |-- bootstrap.php
|   `-- phpunit.xml

```

รูปที่ ข-8 โครงสร้างของ Project

3.1 ทำการแก้ไขไฟล์ httpd.conf ที่อยู่ภายใน AppServ\Apache2.2\conf โดยนำ comment (เครื่องหมาย #) ออกในบรรทัดต่อไปนี้

```
Include conf/extra/httpd-vhosts.conf
```

```
LoadModule rewrite_module modules/mod_rewrite.so
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ทำการแก้ไขไฟล์ httpd-vhosts.conf ที่อยู่ภายใน AppServ\Apache2.2\conf\extra โดยเพิ่มเติมคำสั่งดังรูปที่ ข-9 โดยในส่วนของ ServerName คือให้ใส่ชื่อเซิร์ฟเวอร์ของเรา

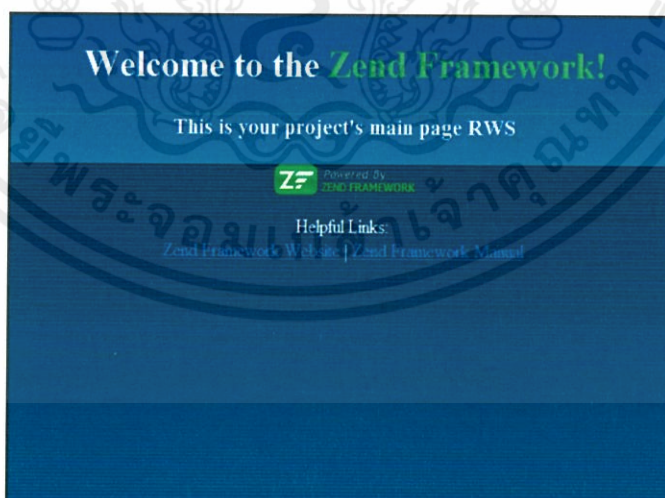
```
<VirtualHost *:80>
  ServerName rws
  DocumentRoot /path/to/RestWS/public

  SetEnv APPLICATION_ENV "development"

  <Directory /path/to/RestWS/public>
    DirectoryIndex index.php
    AllowOverride All
    Order allow,deny
    Allow from all
  </Directory>
</VirtualHost>
```

รูปที่ ข-9 ไฟล์ httpd-vhosts.conf

- 3.3 ทำการแก้ไข host file โดยเพิ่มคำสั่งดังต่อไปนี้ 127.0.0.1 rws (ในขั้นตอนนี้ทำเพื่อทดสอบ ในระหว่างการพัฒนาระบบ)
4. ทำการทดสอบจากการพิมพ์ ServerName ที่เบราว์เซอร์ก็จะปรากฏหน้าจอดังรูปที่ ข-10



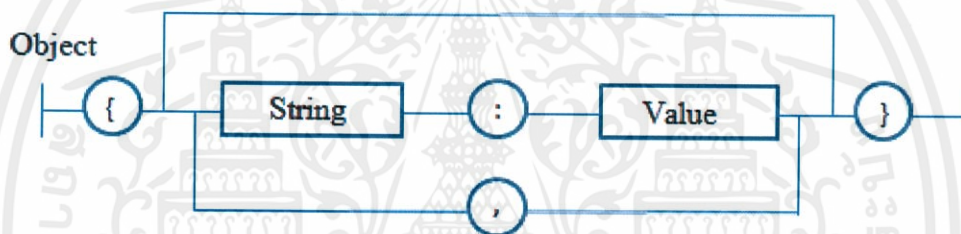
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ ข-10 หน้า Index ของ Zend เฟรมเวิร์ค หน้าไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

JSON (JavaScript Object Notation) คือ รูปแบบของข้อมูลที่ใช้สำหรับแลกเปลี่ยนข้อมูลที่มีขนาดเล็ก ซึ่งสามารถทำความเข้าใจได้ง่าย มันถูกกำหนดภายใต้ภาษา JavaScript JSON เป็นรูปแบบข้อมูลตัวอักษรที่มีความเป็นอิสระอย่างสมบูรณ์ แต่จะมีหลักการการเขียนที่คุ้นเคยกับนักเขียนโปรแกรมภาษาต่างๆได้ไม่ว่าจะเป็นภาษา C C++ C# Java Javascript Perl Python และอื่นๆโดยใน JSON มีรูปแบบได้แก่

1. Object

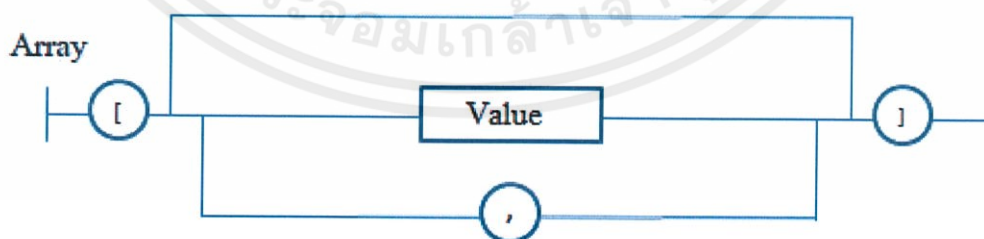
Object นั้นเป็นชุดของข้อมูลที่มีชื่อข้อมูลและค่าของข้อมูลนั้นคู่กัน ซึ่งจะถูกเริ่มต้นด้วยเครื่องหมาย { และจะปิดท้ายข้อมูลด้วยเครื่องหมาย } ข้อมูลแต่ละค่าจะมีเครื่องหมาย : กำกับระหว่างชื่อข้อมูลกับค่าของข้อมูล และแต่ละข้อมูลจะมีเครื่องหมาย , คั่น



รูปที่ ค-1 รูปแบบ Object ของ JSON

2. Array

Array เป็นลำดับของข้อมูล ซึ่งจะถูกเริ่มต้นด้วยเครื่องหมาย [และจะจบด้วยเครื่องหมาย] แต่ละค่าของข้อมูลจะถูกคั่นด้วยเครื่องหมาย ,

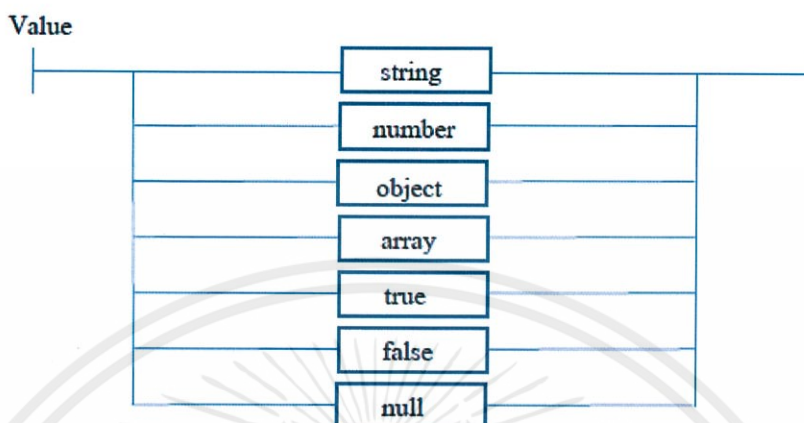


รูปที่ ค-2 รูปแบบ Array ของ JSON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Value

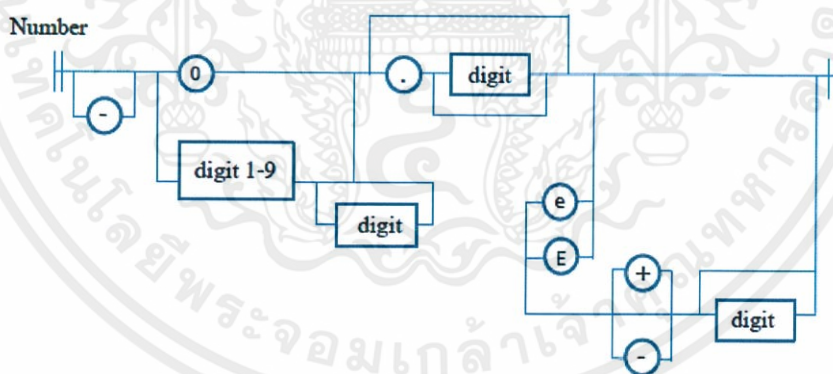
Value เป็น String ที่อยู่ในเครื่องหมาย “ ” หรือตัวเลข หรือค่าทางตรรกศาสตร์ true, false หรือค่า null หรือ object หรือ array ซึ่งโครงสร้างสามารถวางซ้อนกันได้



รูปที่ ค-3 รูปแบบ Value ของ JSON

4. Number

Number นั้นมีความคล้ายคลึงกับ Number ในภาษา C หรือภาษา Java อย่างมาก แต่จะไม่สามารถใช้เลขฐาน 8 กับเลขฐาน 16 ได้ และช่องว่าง (Whitespace) สามารถที่จะใส่ไว้ระหว่างสัญลักษณ์ต่างๆได้

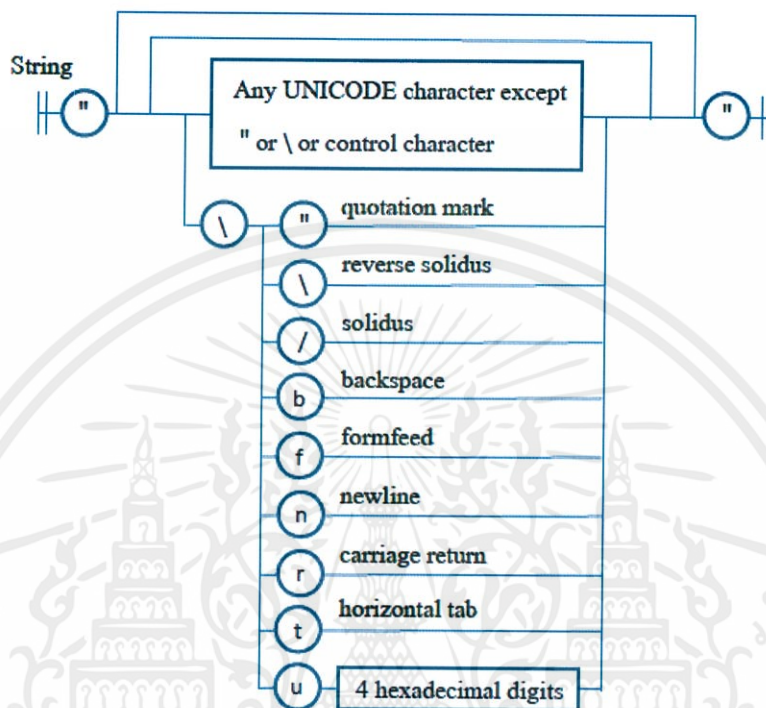


รูปที่ ค-4 รูปแบบ Number ของ JSON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. String

String เป็นลำดับของตัวอักษรตั้งแต่ 0 ตัวอักษรหรือมากกว่า ซึ่งอยู่ภายใต้เครื่องหมาย “ ” และจะใช้เครื่องหมาย \ ในการใส่เครื่องหมายกำกับต่างๆ ซึ่งจะมีลักษณะคล้ายกับ String ในภาษา C หรือภาษา Java



รูปที่ ก-5 รูปแบบ String ของ JSON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง.

ง.1 การพัฒนา REST เว็บเซอร์วิส

ในการพัฒนาเว็บเซอร์วิส REST สามารถแบ่งขั้นตอนออกเป็น 2 ส่วนใหญ่คือ การสร้างส่วนที่ติดต่อกับไคลเอนต์ และการสร้างส่วนที่จัดการกับข้อมูล โดยจะใช้ Zend เฟรมเวิร์ค (ที่มีโครงสร้างแบบ MVC) ในการพัฒนาเว็บเซอร์วิส

ง.1.1 การสร้างส่วนติดต่อกับไคลเอนต์

ตามโครงสร้างของ MVC (Model-View-Controller) ส่วนที่จะติดต่อกับไคลเอนต์ก็คือ Controller แบ่งเป็นการสร้าง Controller เป็น ส่วนของ super คลาส และ sub คลาส นอกจากนี้ยังมี ส่วนของ Model ที่เก็บข้อมูลที่จะส่งกลับไปให้ไคลเอนต์ด้วย

• Super คลาส

super คลาส คือ คลาสที่กำหนดคุณสมบัติ (ตัวแปร หรือ ฟังก์ชัน) ต่างๆขึ้นมาเพื่อให้คลาสอื่นสามารถสืบทอดคุณสมบัติไปใช้งานได้ ในระบบนี้จะมี 1 คลาส คือ Mis_Rest_Controller.php (คลาสจะอยู่ในไดเรกทอรี Rest/Library/Mis/Rest/Controller.php) ซึ่งจะทำหน้าที่รับ request จากไคลเอนต์ จากนั้นจะทำการเช็คค่า request ที่ส่งมาใช้ HTTP ฟังก์ชันอะไร ก็จะทำการเรียกฟังก์ชัน (ที่สร้างไว้รองรับแต่ละ HTTP ฟังก์ชัน) โดยในกรณีของคลาสนี้ จะทำการสร้างฟังก์ชันสำหรับทุก HTTP ฟังก์ชัน ให้มีการส่ง response กลับเป็น HTTP Status Code 501 มีความหมายว่า Not Implement ที่เพราะเหตุว่าคลาสอื่นที่สืบทอดไป หากทำการสร้างฟังก์ชันรองรับแค่บาง HTTP ฟังก์ชัน (เช่น รองรับแค่ GET) เมื่อไคลเอนต์ทำการ request ด้วย HTTP ฟังก์ชันที่ไม่ได้รองรับ (เช่น POST) คลาสที่สืบทอดไปก็จะสามารถเรียกใช้ฟังก์ชันของ super คลาส แทนได้ ซึ่งผลลัพธ์ที่ไคลเอนต์จะได้รับคือ Not Implement ซึ่งหมายความว่าฟังก์ชันนี้ยังไม่ได้ถูกสร้าง

ฟังก์ชัน __construct

ฟังก์ชัน __construct ซึ่งจะเป็นฟังก์ชันที่ทำงานเมื่อคลาสถูกสร้างขึ้น แบ่งการทำงานออกเป็น 4 ส่วนคือ การกำหนด view การกำหนดค่า HTTP code การกำหนดค่าที่จะส่งกลับไปให้ไคลเอนต์และ การกำหนดค่าเริ่มต้นในการติดต่อกับฐานข้อมูลดังรูปที่ ง-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1 <?php
2
3 class Mis_Rest_Controller extends Zend_Rest_Controller
4 {
5     private $_dbAdapter;
6
7     public function __construct(Zend_Controller_Request_Abstract $request,
8     Zend_Controller_Response_Abstract $response, array $invokeArgs = array())
9     {
10         parent::__construct($request,$response,$invokeArgs);
11
12         if (Zend_Controller_Action_HelperBroker::hasHelper('layout')){
13             $this->_helper->layout->disableLayout();
14         }
15         $this->_helper->viewRenderer->setNoRender(true);
16
17         //set the response code
18         $this->_response->setHttpResponseCode(501);
19
20         //create an instance of our response object
21         $this->_output = new Mis_Rest_Response();
22
23         //set code for the output/response object
24         $this->_output->setCode(501);
25
26         //set the body of the output/response object
27         $this->_output->setBody('Method Not Implemented');
28
29         //pass the Accept Headers
30         $this->_output->setAcceptheaders($this->_request->getHeader('Accept'));
31
32         //set type
33         $this->_output->setType();
34
35         $this->setDbAdapter();
36     }

```

รูปที่ ง-1 ตัวอย่าง Controller.php ส่วน method __construct

ส่วนประกอบในฟังก์ชันมีรายละเอียดดังนี้

- บรรทัดที่ 11-13 จะเป็นการกำหนดให้ไม่มีการสร้าง view (ส่วนที่แสดงผลให้ผู้ใช้เห็น)
- บรรทัดที่ 17 เป็นการกำหนด HTTP code ให้เป็น 501 (Not implement)
- บรรทัดที่ 20 ทำการสร้างอ็อบเจกต์ของคลาส Response.php (คลาสที่เก็บข้อมูลที่จะส่งเป็น response กลับไปให้ ไคลเอนต์)
- บรรทัดที่ 23 กำหนดค่า code ให้มีค่า 501
- บรรทัดที่ 27 กำหนดค่า body ให้มีค่าเป็นฟังก์ชัน Not Implement
- บรรทัดที่ 30 กำหนดค่า accept header (ต้องการรับข้อมูลกลับเป็นฟอร์แมตอะไร) โดยรับค่ามาจาก header ของ request
- บรรทัดที่ 33 กำหนด type (ฟอร์แมตที่จะส่งข้อมูลกลับ) default เป็น JSON
- บรรทัดที่ 35 กำหนดค่าให้กับ db adapter (อ็อบเจกต์ที่ใช้ติดต่อกับฐานข้อมูล)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชันที่ถูกสร้างเพื่อรองรับแต่ละ HTTP method

ในส่วนของฟังก์ชันที่สร้างเพื่อรองรับ HTTP ฟังก์ชันจะแบ่งออกเป็น 5 ฟังก์ชันคือ

indexAction รองรับ HTTP ฟังก์ชัน GET ที่ไม่มีพารามิเตอร์ (บรรทัดที่ 43-49)

getAction รองรับ HTTP ฟังก์ชัน GET ที่มีพารามิเตอร์ (บรรทัดที่ 51-57)

postAction รองรับ HTTP ฟังก์ชัน POST (บรรทัดที่ 59-65)

putAction รองรับ HTTP ฟังก์ชัน PUT (บรรทัดที่ 67-73)

deleteAction รองรับ HTTP ฟังก์ชัน DELETE (บรรทัดที่ 75-81)

```

43     public function indexAction()
44     {
45         $this->_response->setHeader('Content-type',
46             $this->_output->getContentType(),
47             true);
48         $this->_response->setBody($this->_output->getOutput());
49     }
50
51     public function getAction()
52     {
53         $this->_response->setHeader('Content-type',
54             $this->_output->getContentType(),
55             true);
56         $this->_response->setBody($this->_output->getOutput());
57     }
58
59     public function postAction()
60     {
61         $this->_response->setHeader('Content-type',
62             $this->_output->getContentType(),
63             true);
64         $this->_response->setBody($this->_output->getOutput());
65     }
66
67     public function putAction()
68     {
69         $this->_response->setHeader('Content-type',
70             $this->_output->getContentType(),
71             true);
72         $this->_response->setBody($this->_output->getOutput());
73     }
74
75     public function deleteAction()
76     {
77         $this->_response->setHeader('Content-type',
78             $this->_output->getContentType(),
79             true);
80         $this->_response->setBody($this->_output->getOutput());
81     }
82

```

รูปที่ ง-2 ตัวอย่าง Controller.php ส่วนฟังก์ชันที่รองรับ HTTP ฟังก์ชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชันที่ใช้จัดการกับ response และ db adapter ในคลาส Controller

ในส่วนนี้จะแบ่งออกเป็น 3 ฟังก์ชัน คือ

method setDbAdapter เป็นฟังก์ชันที่ใช้กำหนดค่าต่างเช่น ชื่อhost ชื่อผู้ใช้ รหัสผ่านของฐานข้อมูลให้กับ db adapter (บรรทัดที่ 91-105)

method getDbAdapter เป็นฟังก์ชันที่เรียกอ็อบเจกต์ของ db adapter (บรรทัดที่ 107-110)

method createResponse เป็นฟังก์ชันที่ใช้สร้าง response ที่จะส่งกลับไปให้ไคลเอนต์ (บรรทัดที่ 112-123)

```

91     public function setDbAdapter()
92     {
93         $config = new Zend_Config_Ini('../application/configs/application.ini',
94             'development');
95         $this->_dbAdapter = new Zend_Db_Adapter_Pdo_Mysql(array(
96             'host' => $config->database->hostname,
97             'username' => $config->database->account->username,
98             'password' => $config->database->account->password,
99             'dbname' => $config->database->databasename
100         ));
101
102         $this->_dbAdapter->query("SET character_set_results=utf8");
103         $this->_dbAdapter->query("SET character_set_client=utf8");
104         $this->_dbAdapter->query("SET character_set_connection=utf8");
105     }
106
107     public function getDbAdapter()
108     {
109         return $this->_dbAdapter;
110     }
111
112     public function _createResponse($code, $result)
113     {
114         $this->_output->setCode($code);
115         $this->_output->setBody($result);
116
117         $this->_response->setHttpResponseCode($code);
118         $this->_response->setHeader('content-type',
119             $this->_output->getCtype(),
120             true);
121         $this->_response->setBody($this->_output->getOutput('auto'));
122     }
123 }

```

รูปที่ ง-3 ตัวอย่าง Controller.php ส่วนฟังก์ชันที่จัดการกับ response และ db adapter

• model ที่ใช้เก็บข้อมูลของ response

ตามโครงสร้างของ MVC ส่วนที่เป็น model จะเป็นส่วนที่เก็บข้อมูลและ business logic ซึ่งในระบบนี้คลาสที่เก็บข้อมูลของ response ก็คือ Mis_Rest_Response.php (คลาสจะอยู่ในไดเรกทอรี Rest\library\Mis\Rest\ Response.php) มีหน้าที่จัดการกับ ข้อมูลที่จะส่งกลับ ฟอ์แมตของข้อมูล และ รหัสของ response

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน setAcceptheaders setType getType

จากตัวอย่าง (ในรูปที่ 3.28) จะเป็นฟังก์ชันการทำงานในคลาส Mis_Rest_Response.php ซึ่งประกอบไปด้วยฟังก์ชัน setAcceptheaders setType และ getType

ฟังก์ชัน setAcceptheaders เป็นการกำหนดค่าของประเภทข้อมูลที่ไคลเอนต์ ต้องการรับ (บรรทัดที่ 18-21)

ฟังก์ชัน setType เป็นการกำหนดรูปแบบที่ข้อมูลจะส่งกลับโดยเช็คจาก accept header (บรรทัดที่ 23-36)

ฟังก์ชัน getType เรียกค่า type ของคลาส (บรรทัดที่ 38-41)

```

1  <?php
2
3  class Mis_Rest_Response
4  {
5      private $_code = 200;
6      private $_body = NULL;
7      private $_type = 'json';
8      private $_json = NULL;
9      private $_xml = NULL;
10     private $_ctype = 'application/json; charset=UTF-8';
11     private $_acceptheaders = NULL;
12
13     public function __construct($option = array())
14     {
15         //.....
16     }
17
18     public function setAcceptheaders($headerstring)
19     {
20         $this->_acceptheaders = explode(',', $headerstring);
21     }
22
23     public function setType($type = 'auto')
24     {
25         if($type == 'auto' && $this->_acceptheaders){
26             if(in_array('application/json', $this->_acceptheaders)){
27                 $this->_type = 'json';
28                 $this->_ctype = 'application/json; charset=UTF-8';
29             }
30             elseif(in_array('application/xml', $this->_acceptheaders)){
31                 $this->_type = 'xml';
32                 $this->_ctype = 'application/xml; charset=UTF-8';
33             }
34         }
35     }
36
37
38     public function getType()
39     {
40         return $this->_type;
41     }

```

รูปที่ 3-4 ตัวอย่าง Response.php ส่วนฟังก์ชันที่จัดการกับ accept header และ type

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชันที่จัดการกับส่วนของ code และ body

ข้อมูลใน response ที่จะส่งกลับไปให้ไคลเอนต์แบ่งได้ออกเป็น 2 ส่วน คือ code และ body ทั้งหมดมี 4 ฟังก์ชัน ที่ใช้จัดการคือฟังก์ชัน setCode getCode setBody และ getBody

ฟังก์ชัน setCode เป็นการกำหนดค่า HTTP code ที่จะส่งกลับไปบอกไคลเอนต์ (บรรทัดที่ 43-46)

ฟังก์ชัน getCode เป็นการเรียกค่า code ของคลาส (บรรทัดที่ 48-51)

ฟังก์ชัน setBody เป็นส่วนที่กำหนดข้อมูลที่จะส่งให้ไคลเอนต์ (บรรทัดที่ 53-76)

ฟังก์ชัน getBody เป็นการเรียกค่า body ของคลาส (บรรทัดที่ 78-81)

```

43     public function setCode($code = 200)
44     {
45         $this->_code = $code;
46     }
47
48     public function getCode()
49     {
50         return $this->_code;
51     }
52
53     public function setBody($data = NULL)
54     {
55
56         $body = NULL;
57         //if($data instanceof Zend_Db_Table_Rowset){
58         if($data instanceof Zend_Db_Statement_Pdo){
59             // $data = $data->toArray();
60             $cnt = 0;
61             foreach($data as $row){
62                 $body[$cnt] = array();
63                 foreach($row as $column => $value){
64                     if($value==NULL || $value==' ' || $value == "โง่โง่" )
65                         $body[$cnt][$column] = "no_data";
66                     else
67                         $body[$cnt][$column] = $value;
68                 }
69                 $cnt++;
70             }$this->_body = $body;
71         }
72         if(is_array($data)){
73             $this->_body = $data;
74         }
75     }
76

```

รูปที่ ง-5 ตัวอย่าง Response.php ส่วนฟังก์ชันที่จัดการกับ code และ body

ฟังก์ชันที่ใช้กำหนด response

ในส่วนของฟังก์ชันที่ใช้กำหนด response จะประกอบไปด้วยฟังก์ชัน getXml getJson getOutput และ getCtype

ฟังก์ชัน getXml เป็นการดึงข้อมูลในรูปแบบของ XML ข้อมูลประกอบด้วยส่วน code และ body (บรรทัดที่ 83-104)

ฟังก์ชัน `getJson` เป็นการดึงข้อมูลในรูปแบบของ JSON ข้อมูลประกอบด้วยส่วน `code` และ `body` (บรรทัดที่ 106-112)

ฟังก์ชัน `getOutput` เป็นการเรียกผลลัพธ์ที่จะส่งกลับให้ไคลเอนต์โดยจะใช้ฟังก์ชัน `getXml` และ `getJson` (บรรทัดที่ 114-128)

ฟังก์ชัน `getCtype` เป็นการเรียกค่า `content type` ซึ่งคือประเภทข้อมูลที่จะส่งกลับไปที่ไคลเอนต์ (บรรทัดที่ 130-133)

```

83     public function getXml()
84     {
85         $xml = '<response>';
86         $xml .= '<code>'.$this->_code.'</code>';
87         if(is_string($this->_body)){
88             $xml .= '<msg>'.$this->_body.'</msg>';
89         }
90         elseif(is_array($this->_body)){
91             $xml .= '<msg>';
92             foreach($this->_body as $items){
93                 $xml .= '<item>';
94                 foreach($items as $key => $val){
95                     $xml .= "<$key>$val</$key>";
96                 }
97                 $xml .= '</item>';
98             }
99             $xml .= '</msg>';
100         }
101         $xml .= '</response>';
102         $this->_xml = simplexml_load_string($xml);
103         return $this->_xml->asXML();
104     }
105
106     public function getJson()
107     {
108         $this->_json = new stdClass();
109         $this->_json->code = $this->_code;
110         $this->_json->body = $this->_body;
111         return Zend_Json::encode($this->_json);
112     }
113
114     public function getOutput($type = NULL)
115     {
116         if(!is_null($type)){
117             $this->setType($type);
118         }
119         switch($this->_type){
120             case 'json':
121                 return $this->getJson();
122                 break;
123             case 'xml':
124                 return $this->getXml();
125                 break;
126             default:
127                 break;
128         }
129     }
130
131     public function getCtype()
132     {
133         return $this->_ctype;
134     }

```

รูปที่ ง-6 ตัวอย่าง `Response.php` ส่วนฟังก์ชันที่กำหนด `content type` และดึงข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• Sub คลาส

Sub คลาส เป็นคลาสที่ทำการสืบทอดคุณสมบัติ (ตัวแปร หรือ ฟังก์ชัน) มาจาก super คลาส ซึ่งในระบบนี้จะมีทั้งหมด 11 คลาส แต่จะขอยกตัวอย่างมาเพียง 1 คลาส คือ Mis_Rest_PlacetravelController.php (คลาสจะอยู่ในไดเรกทอรี Rest\application\modules\Rest\controllers\PlacetravelController.php) ซึ่งเป็นคลาสที่หน้าจัดการข้อมูลเกี่ยวกับสถานที่ท่องเที่ยว ทั้งการเรียกข้อมูลของสถานที่ การเพิ่ม การแก้ไข และการลบข้อมูลสถานที่ท่องเที่ยว ซึ่งคลาสจะต้องสร้างฟังก์ชันเพื่อรองรับแต่ละ HTTP method

ฟังก์ชัน __construct

ฟังก์ชัน __construct เป็น ฟังก์ชันที่จะเริ่มทำงานเมื่อ object ของคลาสถูกสร้างขึ้น โดยในฟังก์ชันจะทำการสร้างอ็อบเจกต์ของ model Placetravel (ส่วนที่เก็บข้อมูลของสถานที่ท่องเที่ยวและติดต่อกับฐานข้อมูล)

```

1  <?php
2
3  class Rest_PlacetravelController extends Mis_Rest_Controller
4  {
5      private $_placeTravel;
6      private $_upload;
7      private $_param = array();
8      private $_user;
9
10     public function __construct(Zend_Controller_Request_Abstract $request,
11                                 Zend_Controller_Response_Abstract $response, array $invokeArgs = array())
12     {
13         parent::__construct($request, $response, $invokeArgs);
14         $this->_placeTravel = new Rest_Model_Placetravel($this->getDbAdapter());
15     }

```

รูปที่ ง-7 ตัวอย่าง PlacetravelController.php

ฟังก์ชัน indexAction

การทำงานใน indexAction จะเป็นฟังก์ชันที่รองรับ GET method โดยทำการเรียกฟังก์ชัน getPlaceTravels (เรียกข้อมูลรายการสถานที่ท่องเที่ยวทั้งหมด) ของ model Placetravel จากนั้นนำผลลัพธ์ (\$result) ไปสร้าง response เพื่อส่งกลับไปให้ไคลเอนต์

```

21     public function indexAction()
22     {
23         $result = $this->_placeTravel->getPlaceTravels();
24         $this->_createResponse(200, $result);
25     }
26
27

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ การนำเอกสารนี้ไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีก

รูปที่ ง-8 ตัวอย่าง PlacetravelController.php ส่วนฟังก์ชัน indexAction

ฟังก์ชัน getAction

การทำงานใน getAction จะเป็นฟังก์ชันที่รองรับ GET method โดยทำการเรียกฟังก์ชัน getPlacetravel (เรียกข้อมูลของสถานที่เดียว) ของ model Placetravel จากนั้นจะนำผลลัพธ์ไปสร้าง response

```

29 public function getAction()
30 {
31     $result = $this->_placeTravel->getPlaceTravel($this->_request->get('id'));
32
33     $this->_createResponse(200,$result);
34 }

```

รูปที่ ง-9 ตัวอย่าง PlacetravelController.php ส่วนฟังก์ชัน getAction

ฟังก์ชัน postAction

การทำงานส่วนของฟังก์ชัน postAction จะเป็นฟังก์ชันที่รองรับ POST method ทำหน้าที่สำหรับการเพิ่มรายการสถานที่ท่องเที่ยว โดยแบ่งเป็นส่วนของการ authentication และการจัดการกับข้อมูลที่จะเพิ่มลงในฐานข้อมูล

```

36 public function postAction()
37 {
38     //get user account
39     $this->_user = $this->_request->getParam('user');
40     //get pass
41     $this->_pass = $this->_request->getParam('pass');
42
43     //authentication
44     $auth = new Mis_Rest_Auth($this->_user,
45                             $this->_pass,$this->getDbAdapter());
46     $auth_result = $auth->authenticate();
47
48     if($auth_result->isValid())
49     {
50         //insert data to database
51         $this->_param = $this->_request->getParam('data');
52
53         if(array_key_exists('preview_image',$this->_param)&&
54            $this->_param['preview_image'] != 'no_data'){
55
56             $img_name=explode(".", $this->_param['preview_image']);
57             $this->_param['preview_image'] =
58             $img_name[0]."_".date("U").".".$img_name[1];
59
60             //path
61             $image_path = $this->_param['preview_image'];
62
63             //upload picture to folder
64             $this->_upload = new Rest_Model_Upload();
65             $this->_upload->getFileUpload('file',$image_path);
66         }
67
68         $result =
69         $this->_placeTravel->createPlaceTravel($this->_param,$this->_user);
70     }
71     else{
72         foreach ($auth_result->getMessages() as $message) {
73             $result = array('message' => $message);
74         }
75     }
76
77     $this->_createResponse(200,$result);
78 }

```

รูปที่ ง-10 ตัวอย่าง PlacetravelController.php ส่วนฟังก์ชัน postAction()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยมีรายละเอียดของฟังก์ชัน putAction ดังนี้

- บรรทัดที่ 81-83 เป็นการรับข้อมูล username password ที่ไคลเอนต์ส่งมา
- บรรทัดที่ 86-88 เป็นการตรวจสอบว่า request ที่เข้ามาเป็น request ของผู้ใช้งานในระบบ

หรือไม่

- บรรทัดที่ 90-114 หลังจากยืนยันแล้วว่าเป็นผู้ใช้งานในระบบก็จะทำการแก้ไขข้อมูล
- บรรทัดที่ 98-110 เป็นการตรวจสอบว่ามีการส่งรูปมาด้วยหรือไม่ ถ้ามีจะทำการสร้างอ็อบเจกต์ของ model upload เพื่อใช้จัดการรูปที่ส่งมา

- บรรทัดที่ 112 เป็นการเรียกฟังก์ชัน editPlacetravel เพื่อแก้ไขข้อมูล
- บรรทัดที่ 115-119 ในกรณีที่ไม่ผ่านการ authentication จะส่งข้อความกลับไปบอก

ไคลเอนต์

- บรรทัดที่ 121 เป็นการสร้าง response กลับไปให้ไคลเอนต์

ฟังก์ชัน deleteAction

การทำงานส่วนของฟังก์ชัน deleteAction จะเป็นฟังก์ชันที่รองรับ DELETE method ทำหน้าที่สำหรับการลบข้อมูลรายการสถานที่ท่องเที่ยวโดยแบ่งเป็นส่วนของการ authentication และการลบข้อมูลออกจากฐานข้อมูล

```

124     public function deleteAction()
125     {
126         //get user account
127         $this->_user = $this->_request->getParam('user');
128         //get pass
129         $this->_pass = $this->_request->getParam('pass');
130
131         //authentication
132         $auth =
133             new Mis_Rest_Auth($this->_user, $this->_pass, $this->getDbAdapter());
134         $auth_result = $auth->authenticate();
135
136         if($auth_result->isValid())
137         {
138             $result = $this->_placeTravel->deletePlaceTravel
139                 ($this->_request->get('id'), $this->_user);
140         }
141         else{
142             foreach ($auth_result->getMessages() as $message) {
143                 $result = array('message' => $message);
144             }
145         }
146
147         $this-> createResponse(200, $result);
148     }

```

รูปที่ ง-12 ตัวอย่าง PlacetravelController.php ส่วนฟังก์ชัน deleteAction

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยมีรายละเอียดของฟังก์ชัน putAction ดังนี้

- บรรทัดที่ 127-129 เป็นการรับข้อมูล username password ที่ไคลเอนต์ส่งมา
- บรรทัดที่ 132-134 เป็นการตรวจสอบว่า request เป็น request ของผู้ใช้ในระบบหรือไม่
- บรรทัดที่ 136-140 หลังจากยืนยันแล้วว่าเป็นผู้ใช้ในระบบก็จะทำการแก้ไขข้อมูล
- บรรทัดที่ 138 เป็นการเรียกฟังก์ชัน deletePlacetravel เพื่อลบข้อมูล
- บรรทัดที่ 141-145 ในกรณีที่ไม่ผ่านการ authentication จะส่งข้อความกลับไปบอก

ไคลเอนต์

- บรรทัดที่ 121 เป็นการสร้าง response กลับไปให้ไคลเอนต์

ง.1.2 การสร้างส่วนที่จัดการกับข้อมูล

ในส่วนนี้เราแยกออกเป็น 4 ส่วน คือ การตรวจสอบตัวตนผู้ใช้ การติดต่อข้อมูลกับฐานข้อมูล การจัดการรูปที่ส่งมา และการเก็บบันทึกการติดต่อกับฐานข้อมูล

• การตรวจสอบตัวตนผู้ใช้

ในระบบเว็บเซอร์วิสนี้จะทำการตรวจสอบตัวตนผู้ใช้ ในกรณีที่ผู้ใช้ต้องการ เพิ่ม แก้ไข หรือ ลบข้อมูล เท่านั้นสำหรับการเรียกข้อมูลจะไม่มีตรวจสอบ ยกเว้นกรณีของการเรียกข้อมูลที่เป็นบันทึกการทำงาน (หรือ log report) ซึ่งคลาสที่ใช้ตรวจสอบตัวตนนั้นคือ Auth.php (คลาสนี้จะอยู่ใน Rest\library\Mis\Rest\Auth.php) โดยคลาสนี้จะทำการอิมพลิเมนต์อินเตอร์เฟซ Zend_Auth_Adapter_Interface จากไลบรารีของ Zend ซึ่ง Zend_Auth_Adapter_Interface กำหนดไว้ว่า คลาสที่อิมพลิเมนต์จะต้องทำการสร้าง method authenticate เพื่อใช้ในการตรวจสอบตัวตน

ฟังก์ชัน __construct และ authenticate

ในฟังก์ชัน __construct ในเป็นส่วนที่รับค่า username password และ dbAdapter และในฟังก์ชัน authenticate จะเป็นส่วนที่นำ username และ password ที่รับมา เช็คกับ username และ password ในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1 <?php
2
3 class Mis_Rest_Auth implements Zend_Auth_Adapter_Interface
4 {
5     protected $_username;
6     protected $_password;
7     private $_dbAdapter;
8
9     public function __construct($username,$password,$dbAdapter)
10    {
11        $this->_username = trim($username);
12        $this->_password = trim($password);
13        $this->_dbAdapter = $dbAdapter;
14    }
15
16
17    public function authenticate()
18    {
19        //encrypt password
20        $this->_password = md5($this->_password);
21
22        $user_db = $this->_dbAdapter->query('SELECT * FROM member_tb WHERE username =
23        ?',array($this->_username));
24        $user_array = $this->convertToArray($user_db);
25        if(is_null($user_array)){
26            return new
27            Zend_Auth_Result(-1,$this->_username,array('access'=>'invalid','message'=>'us
28            ername or password invalid'));
29        }
30        else{
31            if($this->_password==$user_array[0]['password']&&$user_array[0]['group_id']==
32            '-1'){
33                return new
34                Zend_Auth_Result(1,$this->_username,array('access'=>'valid','message'=>'a
35                ccount is valid'));
36            }
37            else{
38                return new
39                Zend_Auth_Result(-3,$this->_username,array('access'=>'invalid','message'=
40                >'username or password invalid'));
41            }
42        }
43    }
44 }

```

รูปที่ ง-13 ตัวอย่าง Auth.php

โดยมีรายละเอียดของ method __construct และ authenticate ดังนี้

- บรรทัดที่ 9-15 method __construct เป็นส่วนที่รับ username password จากไคลเอนต์ และสร้างอ็อบเจกต์ของ db adapter

- บรรทัดที่ 17-36 เป็นส่วนของฟังก์ชัน authentication ใช้ตรวจสอบผู้ใช้

- บรรทัดที่ 20 เป็นการเข้ารหัส password ด้วย md5

- บรรทัดที่ 22 เป็นการดึงข้อมูลของผู้ใช้โดยใช้ username ที่ส่งมาเป็นเงื่อนไขในการค้น

- บรรทัดที่ 23 แปลงผลลัพธ์ที่ได้จากฐานข้อมูลให้เป็นอาร์เรย์

- บรรทัดที่ 24-26 เช็คข้อมูลผู้ใช้มีค่าเป็น null หรือไม่ ถ้าใช่แสดงว่า username ที่ส่งมา

นั้นผิด

- บรรทัดที่ 27-35 ในกรณีที่ username ถูกต้อง จะทำการเช็ค password ต่อ ถ้า password ถูก ก็จะส่งผลกลับมาว่า success หรือ 1 (ในบรรทัดที่ 30)

ฟังก์ชัน convertToArray

เอกสารนี้เผยแพร่และสงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น ผู้จัดทำขอสงวนสิทธิ์ในข้อมูลและข้อมูลอ้างอิงที่ปรากฏในเอกสารฉบับนี้ไว้เป็นของตนเอง

ฟังก์ชัน convertToArray จะเป็นการนำผลลัพธ์ที่ได้จากการติดต่อกับฐานข้อมูล

(Zend_Db_Statement_Pdo) มาแปลงเป็นอาร์เรย์เพื่อนำผลลัพธ์ไปใช้ต่อ โดยการแปลงนั้น จะทำ

การเช็คก่อนว่าข้อมูลที่ส่งมาเป็น Zend_Db_Statement_Pdo หรือไม่ ถ้าใช่ก็จะทำการแปลงเป็นอาร์เรย์แล้วส่งค่ากลับไป ถ้าไม่ก็จะส่งค่า null กลับไปแทน

```

37     public function convertToArray($data = NULL) {
38         $array = NULL;
39         if($data instanceof Zend_Db_Statement_Pdo){
40             //$data = $data->toArray();
41             $cnt = 0;
42             foreach($data as $row){
43                 $array[$cnt] = array();
44                 foreach($row as $column => $value){
45                     if($value==NULL || $value=='')
46                         || $value == "ไม่มี" || $value == "(null)" )
47                         $array[$cnt][$column] = "no_data";
48                     else
49                         $array[$cnt][$column] = $value;
50                 }
51                 $cnt++;
52             }
53         }
54         return $array;
55     }

```

รูปที่ ง-14 ตัวอย่าง Auth.php ส่วนฟังก์ชัน convertToArray

• การติดต่อข้อมูล

ในส่วนของคลาสนี้จะเป็นส่วนที่ไปติดต่อกับฐานข้อมูลจริงๆ โดยจะมีทั้งหมด 11 คลาส แต่จะขอยกตัวอย่างมาเพียงคลาสเดียว คือ คลาส Placetravel.php (คลาสนี้จะอยู่ใน Rest\application\modules\Rest\models\Placetravel.php) โดยจะเป็นคลาสที่จัดการกับข้อมูลที่เป็นสถานที่ท่องเที่ยว มีฟังก์ชันที่เกี่ยวข้องทั้งหมด 5 ฟังก์ชัน คือ getPlacetravels getPlaceTravel createPlaceTravel editPlaceTravel และ deletePlaceTravel

ฟังก์ชัน __construct

ฟังก์ชัน __construct จะเป็นการทำงานเมื่อคลาสนี้ถูกสร้างอ็อบเจกต์ขึ้นมา โดยในฟังก์ชันนี้จะทำการรับอ็อบเจกต์ของ dbAdapter มาเพื่อใช้ติดต่อกับฐานข้อมูล

```

1  <?php
2
3  class Rest_Model_Placetravel extends Mis_Rest_Data
4  {
5      private $ _menu = 'Place Travel';
6      private $ _dbAdapter;
7      private $ _result;
8      private $ _table = 'place_travel';
9
10     public function __construct($dbAdapter)
11     {
12         parent::__construct($dbAdapter);
13         $this->_dbAdapter = $dbAdapter;
14     }

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ในการอ้างอิงเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกหรือทำซ้ำโดยไม่ได้รับอนุญาตจากเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ ง-15 ตัวอย่าง Placetravel.php

ฟังก์ชัน getLocationTravels

ฟังก์ชัน getLocationTravels จะเป็นการเรียกข้อมูลทุกเรคคอร์ดมาจาก table place_table

```

16 public function getLocationTravels()
17 {
18     $this->_result = $this->_dbAdapter->query('SELECT
a.name,a.place_id,b.province_name FROM '.$this->_table.' a LEFT JOIN province b
ON a.province_id = b.province_id ORDER BY place_id ASC');
19     return $this->_result;
20 }

```

รูปที่ ง-16 ตัวอย่าง Placetravel.php ส่วนฟังก์ชัน getLocationTravels

ฟังก์ชัน getLocationTravel

ฟังก์ชัน getLocationTravel จะเรียกข้อมูลเพียงหนึ่งเรคคอร์ดออกมาจาก table place_table โดยใช้ id ที่ส่งมาเป็นเงื่อนไข

```

22 public function getLocationTravel($id)
23 {
24     $this->_result = $this->_dbAdapter->query('SELECT a.place_id, a.preview_image ,
a.name, a.name_th, a.content, a.address, a.phone, a.phone_call, a.latitude,
a.longitude, a.open, a.promotion, a.promotion_start, a.promotion_end,
a.direction, a.recommend, a.create_date,a.last_update, a.website, a.update_user,
a.is_active, b.province name, c.type name, d.eat_type name, e.price_range FROM
'.$this->_table.' a INNER JOIN province b ON a.province_id = b.province_id INNER
JOIN place_travel_type c ON a.type_id = c.type_id LEFT JOIN place_travel_eat_type
d ON c.type_id = d.type_id AND a.eat_type_id = d.eat_type_id LEFT JOIN
place_travel_pricerange e ON a.price_id = e.price_id WHERE a.place_id = ?',
array($id));
25     return $this->_result;
26 }

```

รูปที่ ง-17 ตัวอย่าง Placetravel.php ส่วนฟังก์ชัน getLocationTravel

ฟังก์ชัน createPlaceTravel

ในฟังก์ชัน createPlaceTravel จะทำการเพิ่มข้อมูลลงใน table place_travel จากนั้นจะทำการบันทึกผลโดยใช้ฟังก์ชัน logInsert

```

28 public function createPlaceTravel(array $data = array(), $user)
29 {
30     $row = $this->_dbAdapter->insert($this->_table, $data);
31
32     //log insert
33     $this->logInsert($user, $row, $data, $this->_menu);
34 }

```

รูปที่ ง-18 ตัวอย่าง Placetravel.php ส่วนฟังก์ชัน createPlaceTravel

ฟังก์ชัน editPlaceTravel

ฟังก์ชัน editPlaceTravel จะทำการแก้ไขข้อมูลใน table place_travel โดยเลือกแก้ไขเฉพาะเรคคอร์ดที่ไคลเอนต์ส่งมา (\$id) จากนั้นจะทำการบันทึกผลโดยใช้ฟังก์ชัน logUpdate

```

36 public function editPlaceTravel($id,array $data = array(),$user)
37 {
38     //query old value
39     $resultTemp = $this->_result = $this->_dbAdapter->query('SELECT * FROM
place_travel WHERE place_id = ?', array($id));
40
41     $where[] = "place_id = $id";
42     $row = $this->_dbAdapter->update($this->_table, $data, $where);
43
44     //log update
45     $this->logUpdate($user,$row,$data,$resultTemp,$this->_menu);
46 }

```

รูปที่ 3-19 ตัวอย่าง Placetravel.php ส่วนฟังก์ชัน editPlaceTravel

ฟังก์ชัน deletePlaceTravel

ฟังก์ชัน deletePlaceTravel จะทำการลบข้อมูลใน table place_travel โดยเลือกเฉพาะเรคคอร์ดที่ไคลเอนต์ส่งมา (\$id) จากนั้นทำการบันทึกผลด้วยฟังก์ชัน logDelete

```

48 public function deletePlaceTravel($id,$user)
49 {
50     $where[] = "place_id = $id";
51     $row = $this->_dbAdapter->delete($this->_table, $where);
52
53     //log delete
54     $this->logDelete($user,$row,$id,$this->_menu);
55 }
56
57
58 }

```

รูปที่ 3-20 ตัวอย่าง Placetravel.php ส่วนฟังก์ชัน deletePlaceTravel

• การจัดการรูปที่ส่งมา

เนื่องจากข้อมูลที่ PTT Life Station ใช้มีส่วนที่เป็นรูปภาพด้วย ตัวเว็บเซอร์วิสต้องสามารถรองรับในการเก็บข้อมูลที่เป็นรูปได้ โดยจะมีเมนูที่เก็บข้อมูลรูปทั้งหมด 4 เมนู คือ PlaceTravel PlaceTravelEatType PlaceStationType และ OtherApplication โดยทั้ง 4 เมนูนี้อาจใช้คลาส Upload (คลาสจะอยู่ในไดเรกทอรี Rest\application\modules\Rest\models\Upload.php) เป็นตัวจัดการ

ฟังก์ชัน getFileUpload

จากตัวอย่างคลาสนี้จะทำการเซฟรูปที่รับมาลงใน path ที่ไคลเอนต์ส่งมาให้ (\$image_path) ในกรณีที่เป็นการเปลี่ยนรูป ระบบจะทำการลบรูปเดิมออกแล้วเซฟรูปใหม่แทน (ข้อมูลที่อยู่ของรูปเดิมคือ \$old_path)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1 <?php
2
3 class Rest_Model_Upload
4 { private $ftp;
5
6     function __construct()
7     {
8         //...
9     }
10
11    function getFileUpload($param=NULL,$image_path,$old_path=NULL)
12    {
13        $config = new Zend_Config_Ini('../application/configs/application.ini',
14            'development');
15
16        $this->ftp = new Zend_File_Transfer_Adapter_Http();
17        $file_name = $this->ftp->getFileName('file',false);
18
19        $this->ftp->addFilter('Rename',array('target'=>'/var/www/weloveptt.com'.$image_p
20            ath,'overwrite'=>true));
21        $this->ftp->setDestination($config->uploads->base_path);
22        try{
23            $this->ftp->receive();
24
25            //delete old file
26            if(!is_null($old_path)){
27                $sub = substr($old_path, 0, 4);
28                if($sub == 'http'){
29                    $path = '/var/www/weloveptt.com'.$old_path;
30                    unlink($path);
31                }
32            }
33        }
34        catch(Zend_File_Transfer_Exception $e){
35            error_log($e->getMessage());
36        }
37    }
38 }

```

รูปที่ ง-21 ตัวอย่าง Upload.php

• การเก็บบันทึกการติดต่อกับฐานข้อมูล (Log Report)

เนื่องจากการเปลี่ยนแปลงข้อมูล ไม่ว่าจะเป็นการเพิ่ม แก้ไข หรือลบข้อมูล มีผลต่อผู้ใช้ แอปพลิเคชัน PTT Life Station ทำให้ต้องมีการเก็บบันทึกการเปลี่ยนแปลงเพื่อที่จะสามารถ ตรวจสอบได้ในกรณีที่เกิดข้อผิดพลาดโดยจะสร้างคลาส Data (คลาสจะอยู่ในไดเรกทอรี Rest\library\Mis\Rest\Data.php) เป็น super คลาส ให้ model อื่นๆสืบทอดไปใช้

ฟังก์ชัน __construct

ในฟังก์ชัน __construct เป็นการรับค่า db adapter ที่ถูกส่งมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1 <?php
2
3 class Mis_Rest_Data
4 {
5     private $_log;
6     private $_dbAdapter;
7
8     public function __construct($dbAdapter)
9     {
10         $this->_dbAdapter = $dbAdapter;
11     }

```

รูปที่ ง-22 ตัวอย่าง Data.php

ฟังก์ชัน logInsert

ฟังก์ชัน logInsert จะเป็นการบันทึกรายละเอียดการเพิ่มข้อมูลลงฐานข้อมูล ข้อมูลที่บันทึกประกอบด้วย เวลา ชื่อผู้ใช้ ชื่อtable ชื่อfield และข้อมูลที่ทำกรเพิ่ม

```

32 public function logInsert($user,$row=0,$data,$table)
33 {
34     if($row == 1){
35         $this->_log = new Rest_Model_Log($this->_dbAdapter);
36         $date = date('Y-m-d H:i:s');
37         foreach($data as $key => $value){
38             $insert_data = array(
39                 'datetime' => $date,
40                 'script' => 'iPhone',
41                 'user' => $user,
42                 'action' => 'insert',
43                 'table' => $table,
44                 'field' => $key,
45                 'newvalue' => $value);
46
47             $this->_log->createLog($insert_data);
48         }
49     }
50 } //end logInsert

```

รูปที่ ง-23 ตัวอย่าง Data.php ส่วนฟังก์ชัน logInsert

ฟังก์ชัน logUpdate

ฟังก์ชัน logUpdate จะเป็นการบันทึกรายละเอียดการแก้ไขข้อมูลลงฐานข้อมูล ข้อมูลที่บันทึกประกอบด้วย เวลา ชื่อผู้ใช้ ชื่อtable ชื่อfield ข้อมูลก่อนถูกแก้ไข และข้อมูลหลังจากแก้ไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

52 public function logUpdate($user,$row=0,$data,$resultTemp,$table)
53 {
54     if($row == 1){
55         $this->_log = new Rest_Model_Log($this->_dbAdapter);
56         $date = date('Y-m-d H:i:s');
57         $array = $this->convertToArray($resultTemp);
58         foreach ($data as $key => $value){
59             if($array[0][$key] != $value){
60                 $insert_data = array(
61                     'datetime' => $date,
62                     'script' => 'iPhone',
63                     'user' => $user,
64                     'action' => 'update',
65                     'table' => $table,
66                     'field' => $key,
67                     'oldvalue' => $array[0][$key],
68                     'newvalue' => $value);
69
70                 $this->_log->createLog($insert_data);
71             }
72         }
73     }
74 } //end logUpdate

```

รูปที่ ง-24 ตัวอย่าง Data.php ส่วนฟังก์ชัน logUpdate

ฟังก์ชัน logDelete

ฟังก์ชัน logDelete จะเป็นการบันทึกรายละเอียดการลบข้อมูลลงฐานข้อมูล ข้อมูลที่บันทึกประกอบด้วย เวลา ชื่อผู้ใช้ ชื่อ table และ เรคคอร์ดที่ถูกลบ

```

76 public function logDelete($user,$row=0,$id,$table)
77 {
78     if($row == 1){
79         $this->_log = new Rest_Model_Log($this->_dbAdapter);
80         $date = date('Y-m-d H:i:s');
81         $delete_data = array(
82             'datetime' => $date,
83             'script' => 'iPhone',
84             'user' => $user,
85             'action' => 'delete',
86             'table' => $table,
87             'field' => 'id',
88             'oldvalue' => $id);
89
90         $this->_log->createLog($delete_data);
91     }
92 } //end logDelete

```

รูปที่ ง-25 ตัวอย่าง Data.php ส่วนฟังก์ชัน logDelete

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน logLogin และ logLogout

ฟังก์ชัน logLogin เป็นการทำการบันทึกเมื่อมีผู้ใช้ทำการล็อกอินเข้าระบบมา และฟังก์ชัน logLogout เป็นการบันทึกเมื่อผู้ใช้ล็อกเอาต์ออกจากระบบ ข้อมูลที่ทำการบันทึกคือ ชื่อผู้ใช้ และเวลาที่เข้า/ออกระบบ

```

96     $this->_log = new Rest_Model_Log($this->_dbAdapter);
97     $date = date('Y-m-d H:i:s');
98     $insert_data = array(
99         'datetime' => $date,
100        'script' => 'iPhone',
101        'user' => $user,
102        'action' => 'login');
103
104     $this->_log->createLog($insert_data);
105
106
107 }//end logLogin
108
109 public function logLogout($user)
110 {
111     $this->_log = new Rest_Model_Log($this->_dbAdapter);
112     $date = date('Y-m-d H:i:s');
113     $insert_data = array(
114         'datetime' => $date,
115         'script' => 'iPhone',
116         'user' => $user,
117         'action' => 'logout');
118
119     $this->_log->createLog($insert_data);
120
121
122 }//end logLogin

```

รูปที่ ง-26 ตัวอย่าง Data.php ส่วนฟังก์ชัน logLogin และ logLogout

ง.2 การพัฒนา iOS แอปพลิเคชัน

ง.2.1 การสร้างส่วนติดต่อผู้ใช้

สำหรับการสร้างส่วนติดต่อผู้ใช้งานของ iOS แอปพลิเคชันนั้นสามารถทำได้ภายในโปรแกรม Xcode ผ่านทางไฟล์ที่มีชื่อว่า MainStoryboard.storyboard ซึ่งใช้สำหรับการสร้างส่วนติดต่อผู้ใช้ภายในแอปพลิเคชัน หรือจะทำการสร้างโดยการเขียนโค้ดโปรแกรมผ่านทางคลาสโดยตรง โดยคลาสนั้นอยู่ในภาษา Objective-C นั้นจะประกอบด้วย 2 ไฟล์ด้วยกันนั่นคือ .h กับ .m ซึ่ง .h จะหมายถึง interface file ส่วน .m จะหมายถึง implementation file โดยในที่นี้จะขอแสดงถึงการสร้างในส่วนของ LoginViewController MainMenuController PlaceTravelListController PlaceTravelViewController, PlaceTravelAddController และ PlaceTravelEditController ดังรูปที่ ง-27 เนื่องจากภายในคลาสอื่นๆจะมีการสร้างคล้ายกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-27 ภาพรวมการทำงานของแอปพลิเคชัน

3.2.1.1 การสร้างส่วนติดต่อผู้ใช้จาก MainStoryboard.storyboard

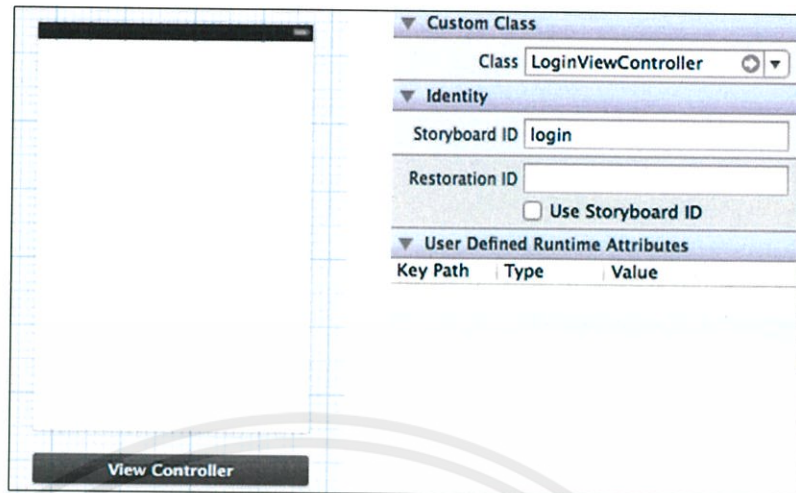
- การกำหนดคลาสให้กับ View Controller

ในการสร้างส่วนติดต่อผู้ใช้ในแต่ละหน้านั้นจะเป็นการสร้างในหน้าที่ชื่อว่า View Controller ซึ่งในแต่ละ View Controller นั้นจะต้องมีการกำหนดการเชื่อมต่อหน้า View Controller แต่ละหน้าเข้ากับคลาสแต่ละคลาส อย่างเช่นในรูปที่ 3-28 ซึ่งเป็นหน้า Login จะเป็นการเชื่อมต่อหน้า View Controller เข้ากับคลาส LoginViewController โดยทำการเลือกได้ที่ Custom Class ดังรูปที่ 3-29



รูปที่ 3-28 หน้าล็อกอิน

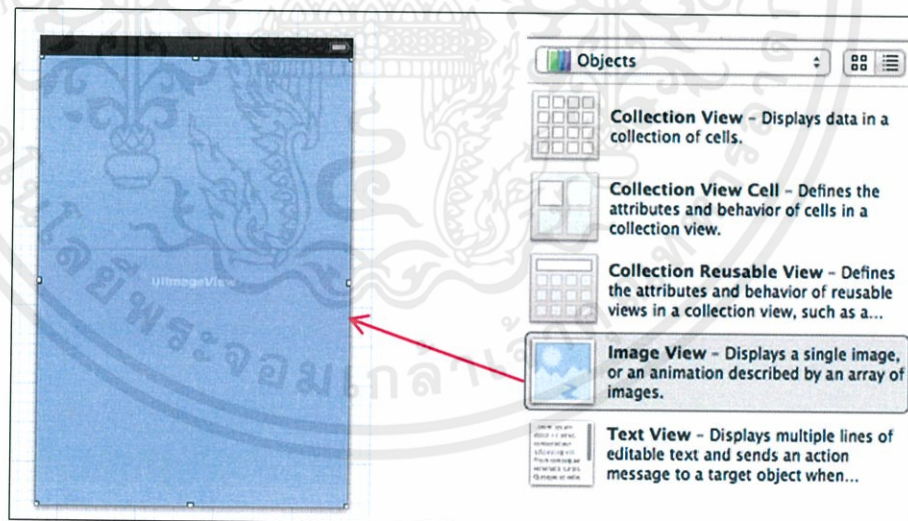
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในโครงการเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องแจ้งถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง-29 การกำหนดคลาส View Controller

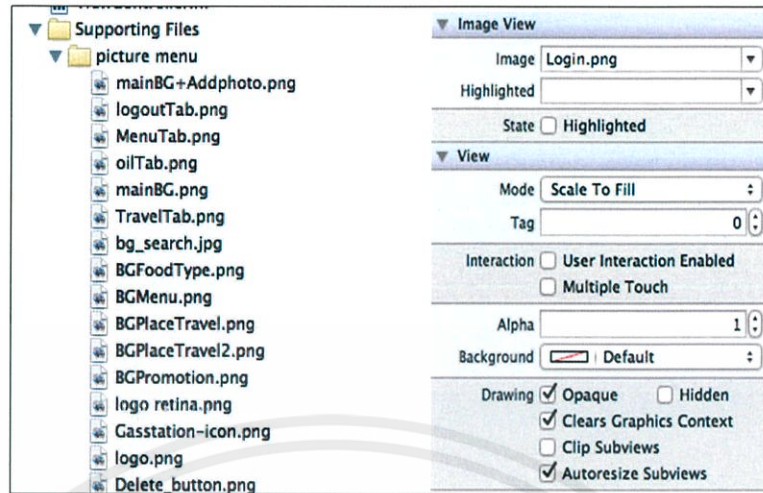
- การสร้างรูปภาพพื้นหลัง

รูปภาพพื้นหลังในแต่ละ View Controller จะทำการสร้างได้โดยการลาก Item ที่ชื่อว่า Image View มาใส่ไว้ในหน้า View Controller ดังรูปที่ ง-30 โดยหลังจากที่ทำการใส่ Image View เสร็จแล้วก็ต้องทำการใส่รูปโดยสามารถกำหนดรูปและค่าต่างๆได้โดยรูปที่สามารถนำมาใส่ได้นั้นจะเป็นรูปที่ทำการลากมาใส่ไว้ไฟล์เดอร์ Supporting File แล้ว ดังรูปที่ ง-31



รูปที่ ง-30 การใส่ Image View

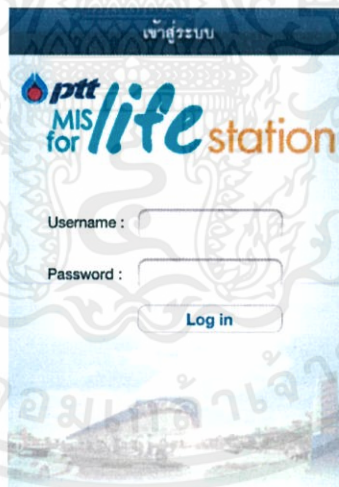
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง-31 การกำหนดค่าให้กับ Image View และ โฟลเดอร์ที่ใช้ในการเก็บรูป

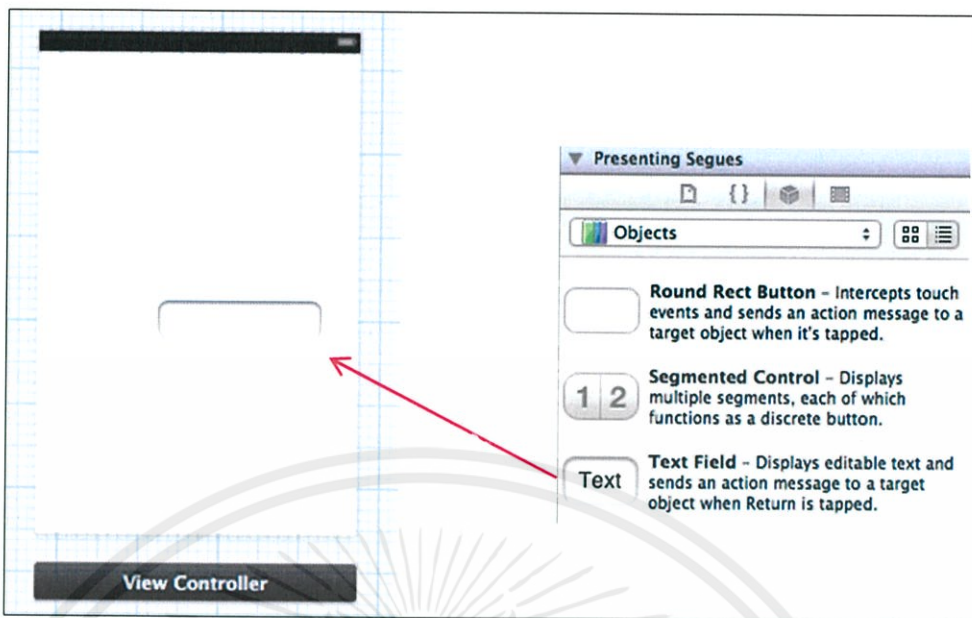
- การสร้าง Text Field

Text Field เป็นช่องที่ใช้สำหรับการกรอกข้อมูลที่เป็นข้อความ เช่น ในรูปที่ ง-32 จะเป็นตัวอย่างของหน้าล็อกอินนั่นก็คือใช้ในการกรอกค่า Username และ Password โดยการสร้างนั้นจะเริ่มจากการลาก Item ที่ชื่อว่า Text Field มาใส่ไว้ที่หน้า View Controller ดังรูปที่ ง-33



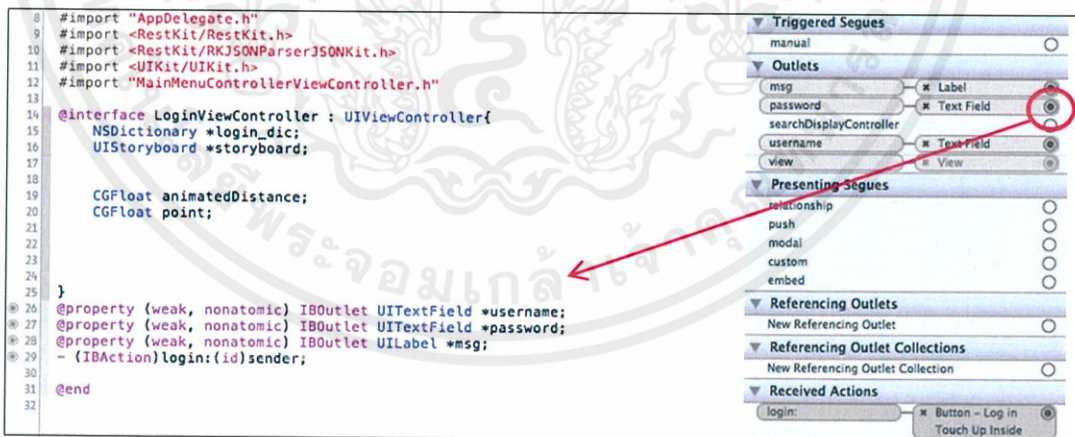
รูปที่ ง-32 หน้าล็อกอิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง-33 การสร้าง Text Field

หลังจากที่ทำการลาก Text Field ใส่ในหน้า View Controller แล้ว ก็ต้องทำการเชื่อม Text Field เข้ากับตัวแปรที่อยู่ภายในแต่ละคลาส เช่นในรูปที่ ง-34 จะเป็นการเชื่อมเข้ากับตัวแปรในคลาส LoginViewController โดยทำได้จากการที่ลากจากช่องในส่วนของ Outlets เข้ากับตัวแปรที่อยู่ในไฟล์ LoginViewController.h โดยเมื่อทำการเชื่อม Text Field เข้ากับตัวแปร เสร็จแล้วจะมีจุดสีดำแสดงขึ้นมาเพื่อเป็นการบอกว่าการเชื่อม Text Field เข้ากับตัวแปรแล้ว ดังรูปที่ ง-34



รูปที่ ง-34 การเชื่อม Text Field เข้ากับตัวแปร

จากรูปที่ ง-34 ในส่วนของโค้ดในไฟล์ LoginViewController.h เป็นการประกาศตัวแปรต่างๆ โดยในบรรทัดที่ 26-28 เป็นการประกาศถึงตัวแปรที่ใช้ในการเชื่อมเข้ากับ Item ต่างๆ ที่ใช้ในการสร้าง ส่วนติดต่อผู้ใช้ โดยหลังจากการประกาศตัวแปรที่ใช้ในการเชื่อมเข้ากับ Item ต่างๆ ที่ใช้ในการสร้างส่วนติดต่อผู้ใช้ แล้วยังต้องทำการ Synthesize ภายในไฟล์ LoginViewController.m ด้วยดัง

รูปที่ ง-35 บรรทัดที่ 19-21 คือทุกๆที่เป็นตัวแปร @property จะต้องทำการ Synthesize โดยการ Synthesize จะเป็นการกำหนด Setter Method กับ Getter Method ให้กับตัวแปร

```

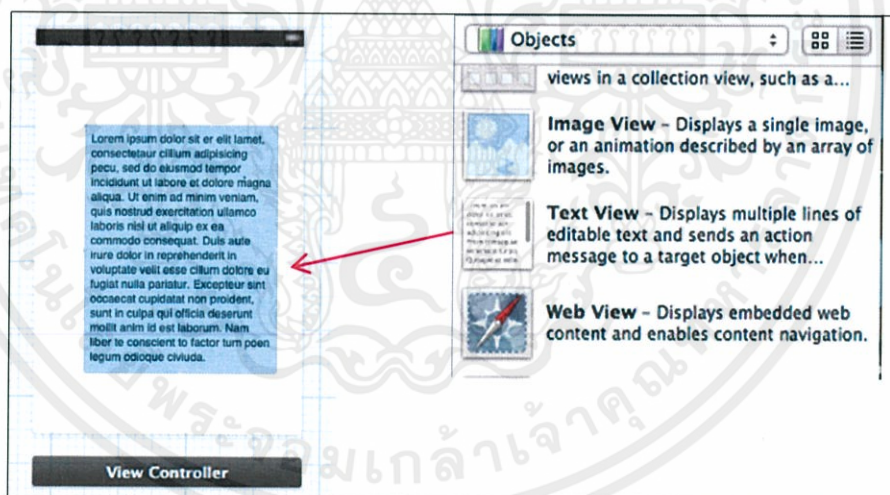
9   #import "LoginViewController.h"
10
11  @interface LoginViewController ()
12
13  @end
14
15  @implementation LoginViewController
16
17  static const CGFloat KEYBOARD_ANIMATION_DURATION = 0.3;
18
19  @synthesize username;
20  @synthesize password;
21  @synthesize msg;

```

รูปที่ ง-35 การ Synthesize ตัวแปร

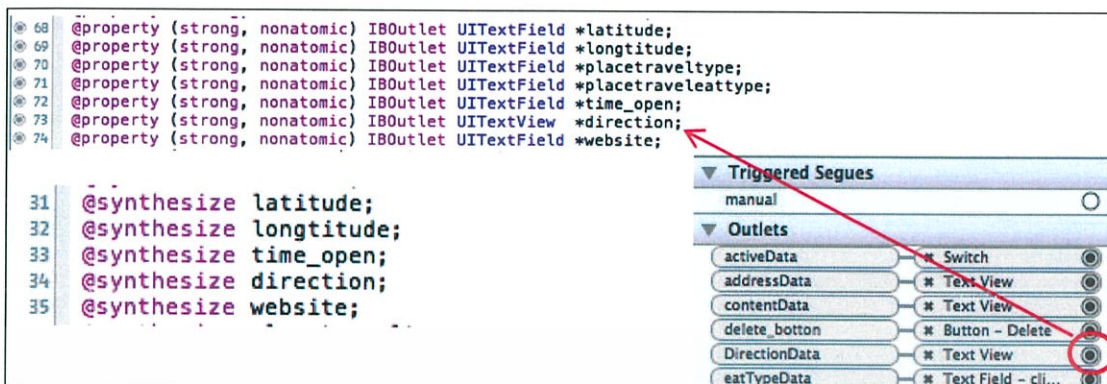
- การสร้าง Text View

Text View นั้นจะคล้ายกับ Text Field แต่จะต่างกันตรงที่ Text View นั้นจะมี Scroll View อยู่ภายในด้วยซึ่งจะเหมาะกับการพิมพ์ข้อความที่มีความยาวมากหลายบรรทัด โดยการสร้าง Text View นั้นก็จะทำเหมือนกับ Text Field ต่างกันเพียงแค่เปลี่ยน ไปเลือก Item ที่ชื่อว่า Text View แทน ดังรูปที่ ง-36 และเมื่อทำการวาง Text View เสร็จแล้วก็ต้องทำการเชื่อมความสัมพันธ์ระหว่าง Text View กับตัวแปรด้วยดังรูปที่ ง-37



รูปที่ ง-36 การสร้าง Text View

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง-37 การเชื่อม Text View เข้ากับตัวแปร

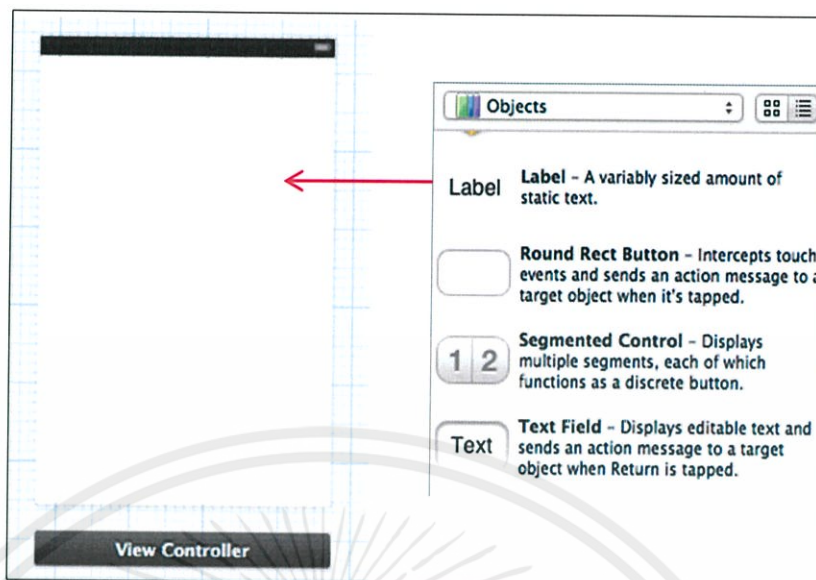
- การสร้าง Label

Label นั้นจะเป็นการสร้างข้อความที่แสดงโดยที่ผู้ใช้ ไม่สามารถทำการพิมพ์ หรือ แก้ไขได้ ดังในรูปที่ ง-38 โดยสร้างได้จากกรณีที่ลาก Item ที่ชื่อว่า Label มาใส่ไว้ในหน้า View Controller ดังรูปที่ ง-39



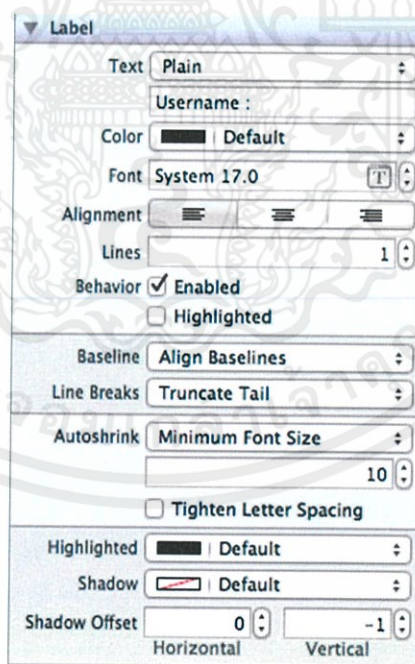
รูปที่ ง-38 หน้าล็อกอิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง-39 การสร้าง Label

โดยเมื่อทำการลาก Label ใสในหน้า View แล้วจะสามารถทำการใส่ข้อความและทำการกำหนดค่าต่างๆใน Label ได้เช่น กำหนดสี และกำหนดฟอนต์ เป็นต้นดังรูปที่ ง-40

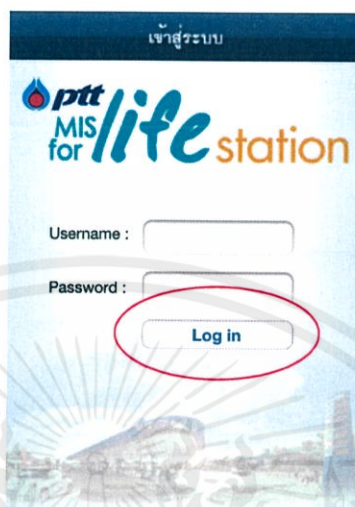


รูปที่ ง-40 การกำหนดค่าของ Label

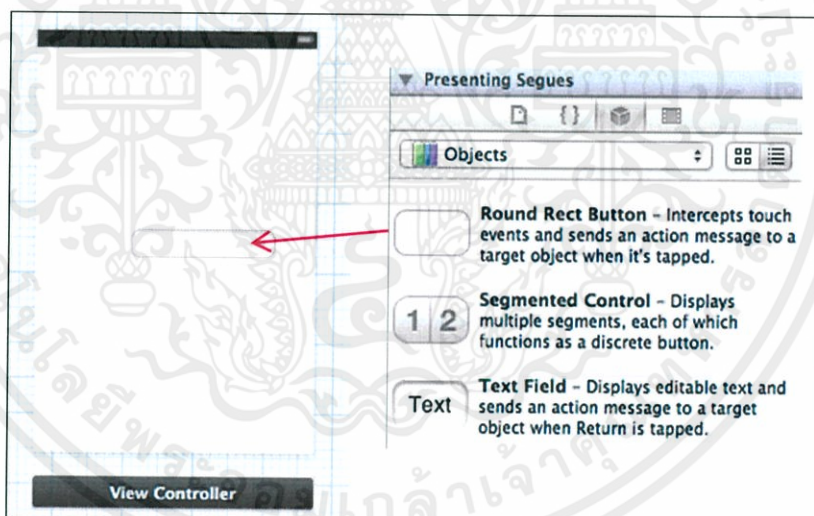
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้นหากมีการนำเอกสารนี้ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Button

Button นั้นเป็นปุ่มที่ให้ผู้สเซอร์ทำการกดได้ดังรูปที่ ง-41 โดยในการสร้างปุ่ม Button นั้นสามารถทำได้จากการลาก Item ที่ชื่อว่า Round Rect Button มาใส่ในหน้า View ดังรูปที่ ง-42



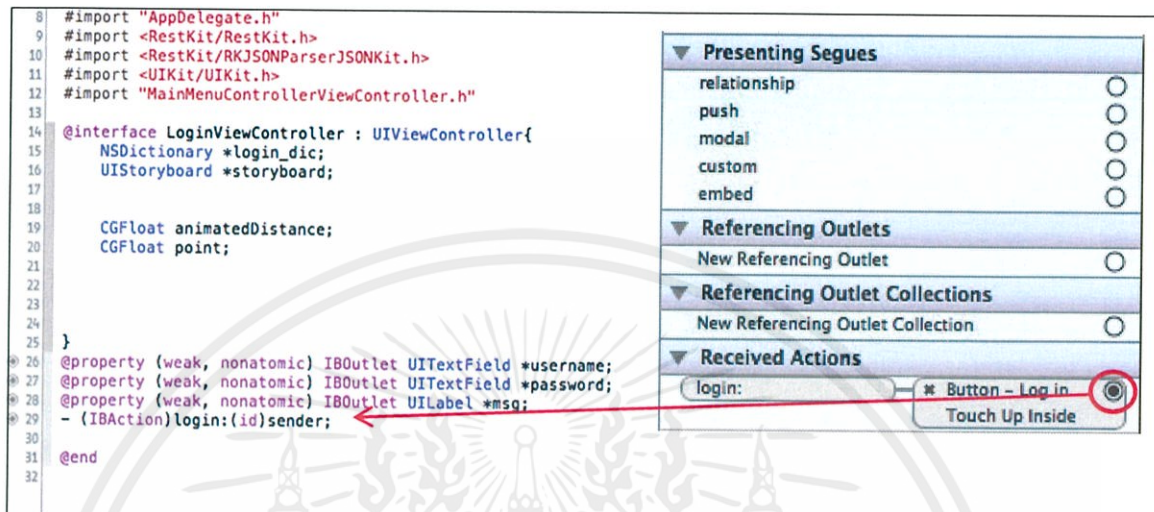
รูปที่ ง-41 หน้าล็อกอิน



รูปที่ ง-42 การสร้าง Button

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

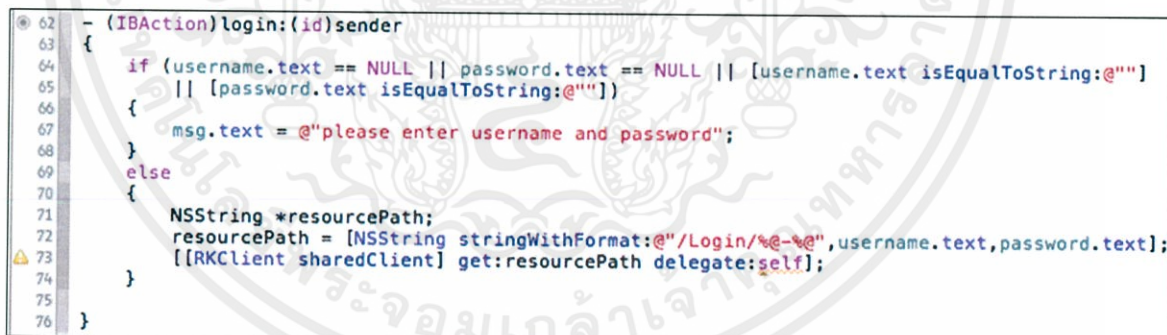
หลังจากที่ทำการใส่ปุ่ม Button ลงในหน้า View Controller ไปแล้วก็ต้องทำการเชื่อมความสัมพันธ์ระหว่างปุ่ม กับ ฟังก์ชันในการทำงานโดยทำการลากจากในหัวข้อ Received Actions ไปเชื่อมกับฟังก์ชัน login ที่อยู่ในไฟล์ LoginViewController.h ดังรูปที่ ง-43



รูปที่ ง-43 การเชื่อม Button เข้ากับตัวแปร

โดยฟังก์ชัน login นั้นจะมีการอิมพลีเมนต์อยู่ในไฟล์ LoginViewController.m

ดังรูปที่ ง-44

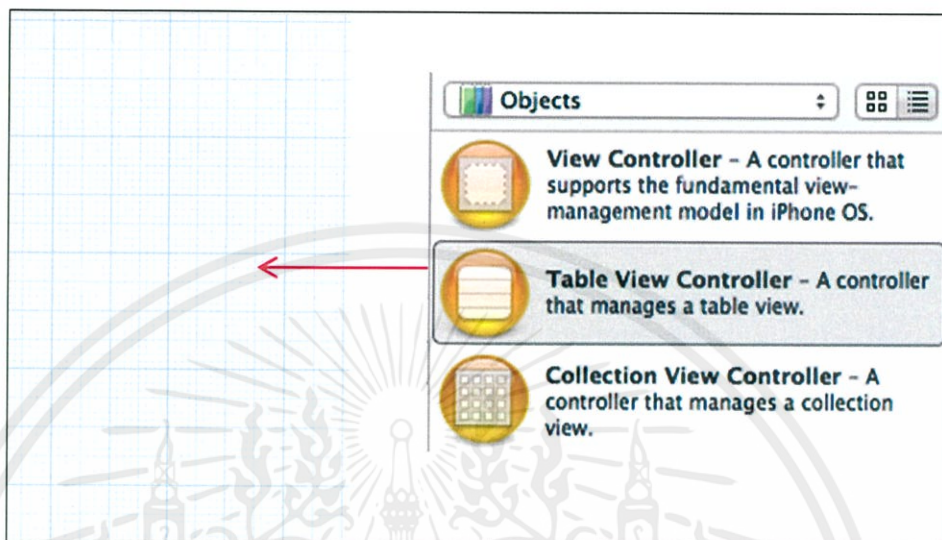


รูปที่ ง-44 ฟังก์ชัน login

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Table View Controller

ในการสร้าง Table View สามารถทำได้โดยการลาก Item ที่ชื่อว่า Table View Controller มาวางใส่ในพื้นที่ Canvas (พื้นที่สำหรับการสร้างหน้า View ต่างๆ) ดังรูปที่ ง-45



รูปที่ ง-45 การสร้าง Table View Controller

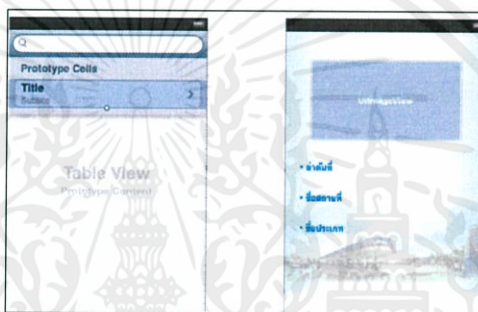


รูปที่ ง-46 Table View Controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Segue

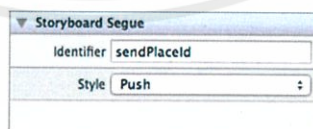
Segue นั้นเป็นตัวเชื่อมต่อระหว่างปุ่มที่ทำการกดได้กับหน้า View Controller ซึ่งจากรูปที่ ง-47 จะเป็นการตัวอย่างของการสร้าง Segue จากหน้า Table View Controller กับหน้า PlaceTravel ViewController ซึ่งจะทำการเปลี่ยนหน้าเมื่อทำการกดไปที่ Cell ของ Table View Controller โดยสร้างได้จากการคลิกที่ Cell ในหน้า Table View Controller ดังรูปที่ ง-48 จากนั้นกด Ctrl ค้างไว้แล้วลากเส้นเชื่อมไปใส่ไว้ที่หน้า PlaceTravelViewController ที่ต้องการจากนั้นทำการเลือก Action ของ Segue เป็นแบบ Push เพื่อให้มีปุ่ม Back กลับมายังหน้าก่อนหน้านี้อีก และสามารถกำหนดชื่อ Segue เพื่อนำไปอ้างอิงใน โค้ดโปรแกรมได้ โดยกำหนดในส่วนของ Storyboard Segue ดังรูปที่ ง-49



รูปที่ ง-47 หน้า Table View Controller กับหน้า PlaceTravelViewController



รูปที่ ง-48 การเชื่อมหน้า Table View Controller กับหน้า PlaceTravelViewController



รูปที่ ง-49 การกำหนดชื่อ Segue

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Navigation Controller

Navigation Controller นั้นจะเป็นการเชื่อมต่อหน้า View Controller ต่างๆเข้าด้วยกันแล้วจะเก็บไป Stack ซึ่งถ้ามีหน้าใหม่มาแทนหน้าปัจจุบันจะทำการ Push Stack และถ้าพิจารณาหน้าปัจจุบันเช่นการ Back กลับออกกลับไปหน้าที่แล้วจะเป็นการ Pop Stack ซึ่ง Navigation Controller นั้นจะใช้ในกรณีที่อยู่ในแอปพลิเคชันมีการเปลี่ยนหน้ากลับไปมาซึ่งวิธีในการสร้าง Navigation Controller นั้นสามารถทำได้โดยการลาก Item ที่ชื่อว่า Navigation Controller มาวางใส่ใน Canvas ดังรูปที่ ง-50

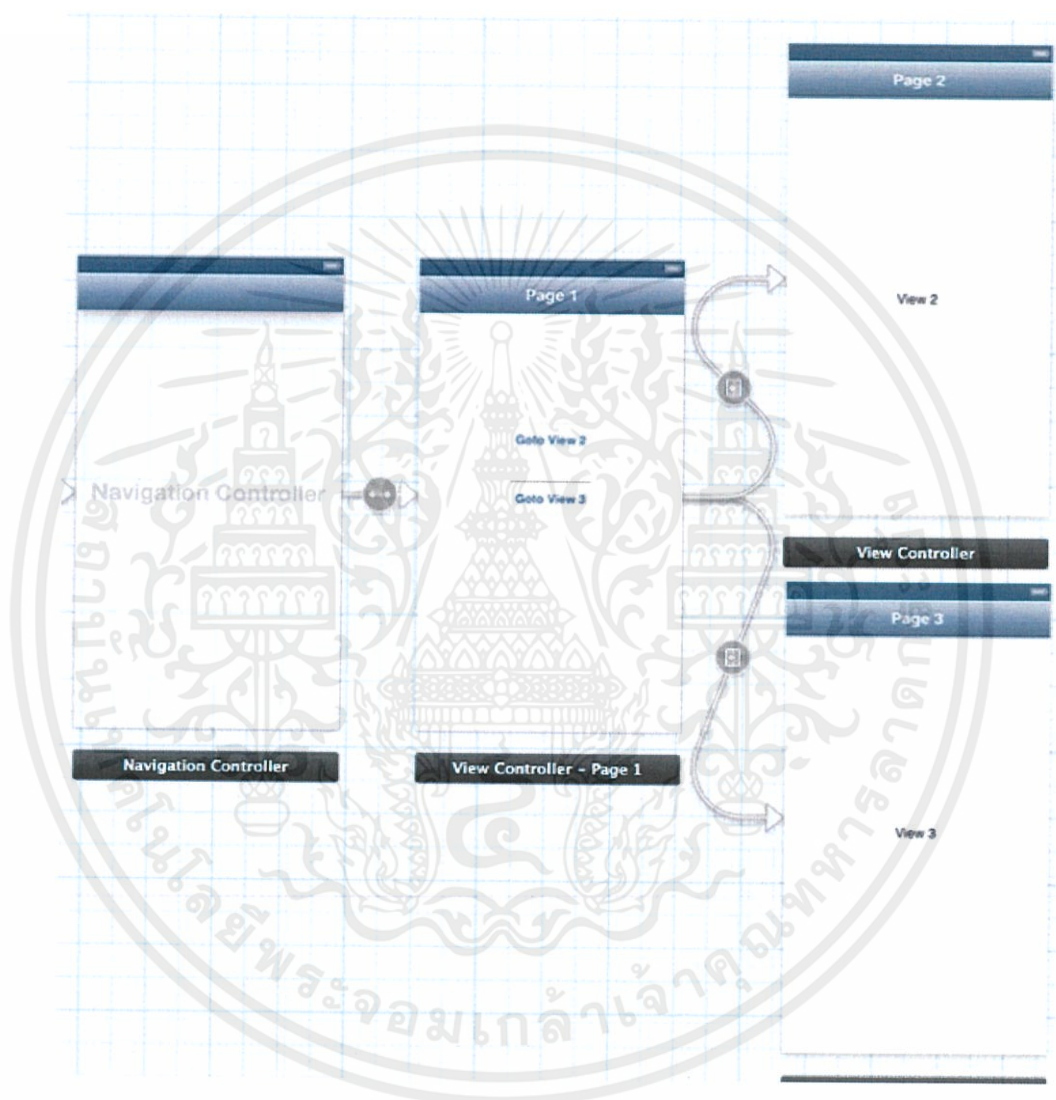


รูปที่ ง-50 การสร้าง Navigation Controller

จากรูปที่ ง-51 จะเป็นตัวอย่างของการทำงานของ Navigation Controller โดยจากรูปจะเห็นว่า Page 1 จะเป็นหน้าหลักซึ่งจะเชื่อมโยงไปยังหน้า Page 2 กับ Page 3 ซึ่งการเชื่อมจะใช้ Segue ในการเชื่อมโดยการกดที่ปุ่มที่ต้องการแล้วกด Ctrl ค้างไว้จากนั้นลากไปยังหน้า Page ที่ต้องการ ซึ่งจากรูปคือปุ่ม Goto View 2 จะเชื่อม Segue ไปยังหน้า Page 2 และ ปุ่ม Goto View 3 จะเชื่อม Segue ไปยังหน้า Page 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานภายในรูปที่ ง-51 จะเห็นได้ว่า Page 1 เป็นหน้าหลักซึ่งเมื่อกดปุ่ม Goto View 2 ก็จะทำการ Push หน้า Page 2 มาวางทับหน้า Page 1 และในการที่ใช้ Navigation Controller เมื่อทำการ Push หน้ามาวางทับแล้วนั้นทาง Navigation Controller จะมีการสร้างปุ่ม Navigation Bar ในการ Back กลับไปยังหน้าที่แล้วให้อัตโนมัติซึ่งจากในรูปก็คือจากหน้า Page 2 ก็จะมีปุ่ม Back ให้กลับมายังหน้า Page 1 ได้ซึ่งก็จะเป็นการ Pop เอาหน้า Page 2 ออก

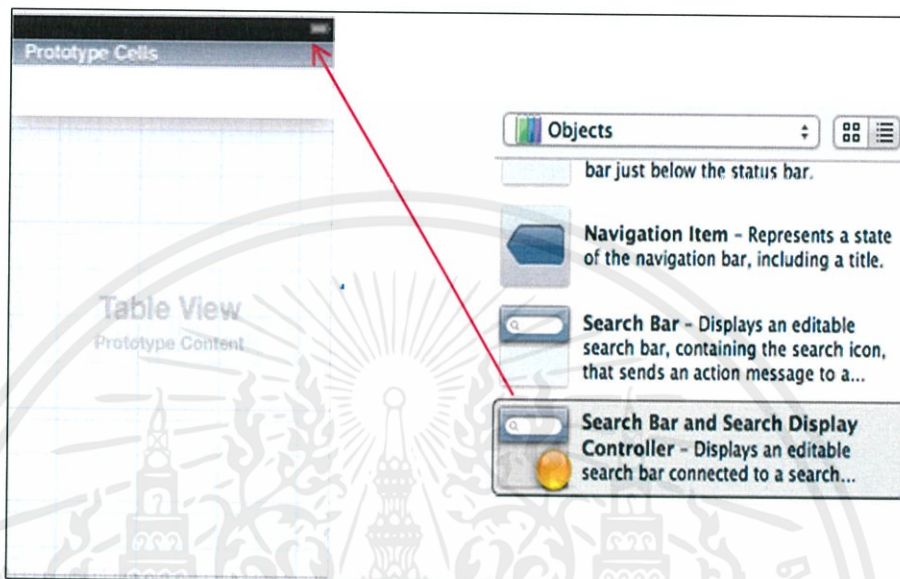


รูปที่ ง-51 การทำงานของ Navigation Controller

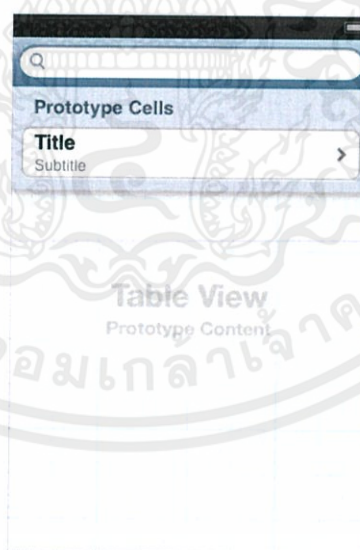
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Search Bar

ในการสร้าง Search Bar ไว้ภายใน Table View นั้นสามารถทำได้โดยการลาก Item ที่ชื่อว่า Search Bar and Search Display มาใส่ไว้ด้านบนของ Table View Controller ที่เคยสร้างไว้แล้วดังรูปที่ ง-52



รูปที่ ง-52 การสร้าง Search Bar



รูปที่ ง-53 หน้า Table View Controller กับ Search Bar

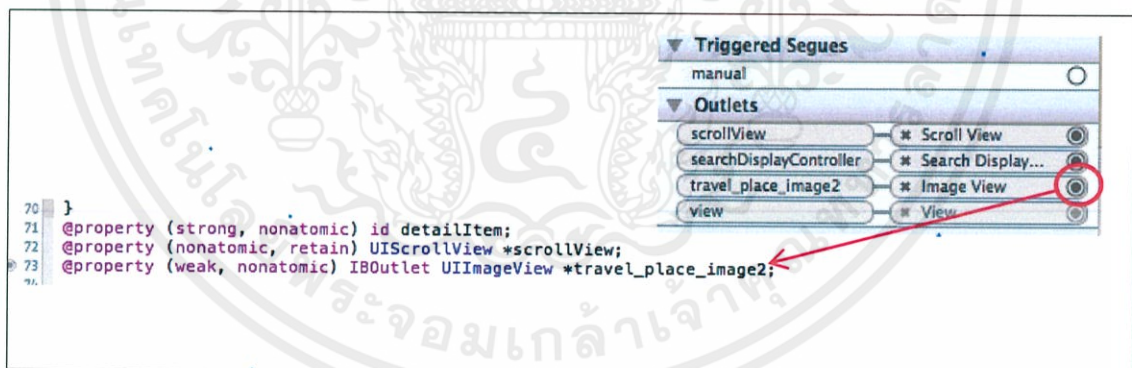
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Image View

Image View นั้นจะต่างจากการสร้าง Image View ในรูปภาพพื้นหลังนั่นก็คือรูปที่จะนำมาแสดงนั้นจะมาจากโค้ดโปรแกรมซึ่งหลังจากทำการสร้าง Image View ลงใน View Controller แล้ว ก็จะต้องเชื่อม Image View เข้ากับตัวแปรในแต่ละคลาสด้วย เช่นในรูปที่ ง-55 จะเป็นตัวอย่างของการเชื่อม Image View เข้ากับตัวแปรในคลาส PlaceTravelViewController



รูปที่ ง-54 หน้า PlaceTravelViewController

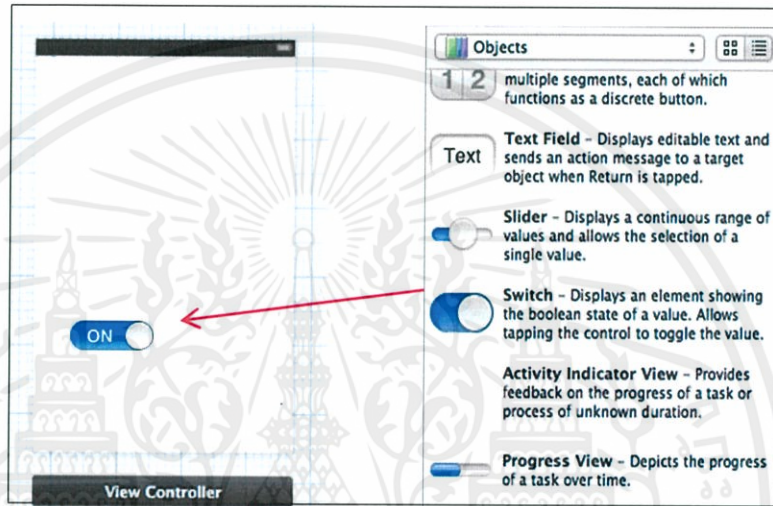


รูปที่ ง-55 การเชื่อม Image View เข้ากับตัวแปร

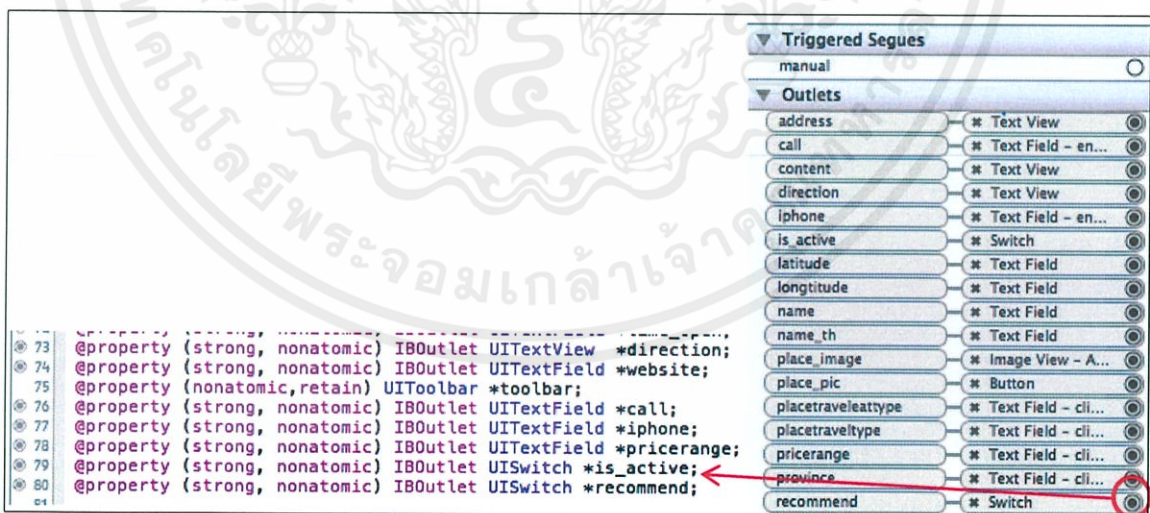
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้างปุ่ม Switch

ปุ่ม Switch นั้นจะเป็นปุ่มที่เมื่อทำการกดลงไปในนั้นจะมีการเปลี่ยนแปลงค่าอยู่ 2 ค่านั้นก็คือค่า On กับ Off โดยการสร้างปุ่ม Switch นั้นจะสร้างคล้ายๆกับการสร้าง Text Field อย่างเช่นในรูปที่ ๓-56 จะเป็นการสร้าง Switch โดยการลาก Item ที่ชื่อว่า Switch มาวางใส่ในหน้า View Controller แล้วก็ต้องมีการเชื่อมความสัมพันธ์ระหว่างปุ่ม Switch กับตัวแปรภายในคลาส ด้วยดังรูปที่ ๓-57 โดยจากรูปจะเป็นตัวอย่างการเชื่อมความสัมพันธ์ระหว่างปุ่ม Switch กับ คลาส PlaceTravelAddController



รูปที่ ๓-56 การสร้าง Switch



รูปที่ ๓-57 การเชื่อม Switch เข้ากับตัวแปร

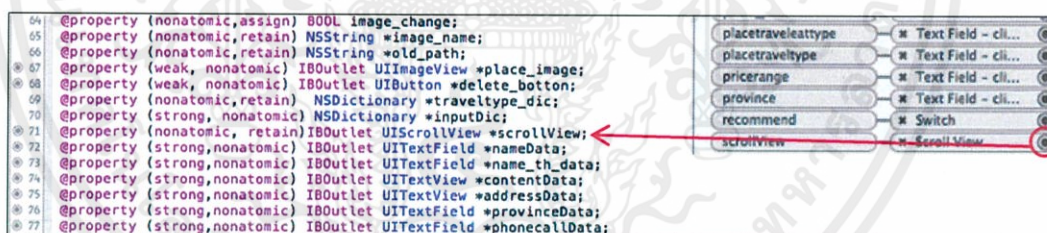
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Scroll View

Scroll View นั้นเป็นการใช้เพื่อให้สามารถเลื่อนหน้าจอได้เวลาที่มีเนื้อหารายละเอียดเกินหน้าจอปกติ โดยวิธีการสร้างนั้นทำได้โดยการลาก Item ที่ชื่อว่า Scroll View ไปวางใส่ใน View Controller ดังรูปที่ ง-58 และเมื่อทำการวางเสร็จแล้วก็ต้องทำการเชื่อมความสัมพันธ์กับตัวแปรภายใน คลาสด้วย ดังรูปที่ ง-59 ส่วนการกำหนดขนาดของ Scroll View ว่าจะให้มีขนาดเท่าไรสามารถทำได้จากการเขียนโค้ดในไฟล์ .m ดังรูปที่ ง-60 และเนื้อหาที่จะให้อยู่ภายใน Scroll View นั้นจะต้องทำการลากมาวางทับ Scroll View แทนการวางแบบปกติที่วางตรงส่วนไหนก็ได้



รูปที่ ง-58 การสร้าง Scroll View



รูปที่ ง-59 การเชื่อม Scroll View เข้ากับตัวแปร

```

956: //---set the viewable frame of the scroll view---
957: scrollView.frame = CGRectMake(0, 0, 320, 460);
958: // //---set the content size of the scroll view---
959: [scrollView setContentSize:CGSizeMake(320, 1880)];
960:
  
```

รูปที่ ง-60 การกำหนดขนาดของ Scroll View

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ง.2.1.2 การสร้างส่วนติดต่อผู้ใช้จาก Implementation File

- การสร้าง TabBar

TabBar เป็นแถบที่สร้างขึ้นมาเพื่อความสะดวกในการเปลี่ยนหน้าไปยังหน้าที่ต้องการได้รวดเร็วยิ่งขึ้นโดยการสร้างจะเป็นการสร้าง ดังรูปที่ ง-61 โดยในบรรทัดที่ 163 จะเป็นการตั้งค่าความยาว กับความสูงของ TabBar และในบรรทัดที่ 165-172 จะเป็นการตั้งชื่อ และกำหนดรูปให้ในแต่ละ Item ใน TabBar โดย TabBar Item อันแรกจะเป็น Root หมายถึงหน้าที่สร้าง TabBar นั้นขึ้นมา ซึ่งในที่นี้ก็คือหน้า Menu จะเป็น Root ของ TabBar ส่วนในบรรทัดที่ 173 จะเป็นการนำค่าที่ทำกรตั้งค่าให้กับ TabBar ต่างๆมาเก็บไว้ในอาร์เรย์เพื่อนำไปเพิ่มลงใน View Controller ในบรรทัดที่ 175-176

```

160 -(void)create_tabbar
161 {
162
163     tabbar = [[UITabBar alloc] initWithFrame:CGRectMake(0, 430, 320, 50)];
164     tabbar.delegate = self;
165     first = [[UITabBarItem alloc] initWithTitle:@"Menu"
166     image:[UIImage imageNamed:@"MenuTab.png"] tag:0];
167     UITabBarItem *second = [[UITabBarItem alloc]
168     initWithTitle:@"Place Travel" image:[UIImage imageNamed:@"TravelTab.png"] tag:1];
169     UITabBarItem *third = [[UITabBarItem alloc]
170     initWithTitle:@" Gas Station " image:[UIImage imageNamed:@"oilTab.png"] tag:2];
171     UITabBarItem *fourth = [[UITabBarItem alloc]
172     initWithTitle:@"Log out" image:[UIImage imageNamed:@"logoutTab.png"] tag:3];
173     NSArray *tabArray = [NSArray arrayWithObjects:first,second,third,fourth, nil];
174
175     tabbar.items = tabArray;
176     [self.navigationController.view addSubview:tabbar];
177
178 }

```

รูปที่ ง-61 ฟังก์ชัน create_tabbar

เมื่อทำการเลือก Item ใน TabBar จะมีการมาตรวจเช็คกว่า Item ที่เลือกนั้นคืออะไร แล้วเมื่อเลือกแล้วจะให้ทำอะไรต่อไป ดังรูป ง-62 อย่างเช่นในบรรทัดที่ 77-81 เมื่อทำการเลือก TabBar Item PlaceStationList จะทำการ push นำหน้าของ PlaceStationList มาวางทับหน้าปัจจุบัน และในบรรทัดที่ 89-91 เมื่อทำการเลือก TabBar Item สุดท้าย (Log out) ระบบจะกำหนดให้ซ่อนหน้าต่างทั้งหมด ซึ่งก็คือการสื่อกเอาที่ออกไปหน้าสื่อกอินนั่นเอง

```

73 - (void)tabBar:(UITabBar *)tabBar didSelectItem:(UITabBarItem *)item
74 {
75     [[self navigationController] popToRootViewControllerAnimated:NO];
76
77     if (item.tag == 2) {
78         PlaceStationListController *listStation =
79         [storyboard instantiateViewControllerWithIdentifier:@"PlaceStationList"];
80         [listStation setTabbar:tabbar];
81         [[self navigationController] pushViewController:listStation animated:NO];
82     }
83     else if (item.tag == 1) {
84         PlaceTravelListController *listTravel =
85         [storyboard instantiateViewControllerWithIdentifier:@"PlaceTravelList"];
86         [listTravel setTabbar:tabbar];
87         [[self navigationController] pushViewController:listTravel animated:NO];
88     }
89     else if (item.tag == 3){
90         [self.navigationController dismissViewControllerAnimated:YES completion:nil];
91     }
92 }
93 }

```

รูปที่ ง-62 ฟังก์ชัน TabBar



รูปที่ ง-63 TabBar

- การสร้าง Bar Button Item

Bar Button Item นั้นเป็นปุ่มที่อยู่ใน Bar ด้านบนซึ่งสามารถเพิ่มปุ่มเข้าไปได้เช่น ปุ่ม add edit เป็นต้น ซึ่งในรูปที่ ง-64 นี้จะเป็นตัวอย่างการสร้าง Bar Button Item ประเภท add

```

53 -(void)createButton
54 {
55     UIBarButtonItem *addButton = [[UIBarButtonItem alloc] initWithBarButtonSystemItem:
56     UIBarButtonItemAdd target:self action:@selector(goForward)];
57     self.navigationItem.rightBarButtonItem = addButton;
58 }
59

```

รูปที่ ง-64 ฟังก์ชัน createButton

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ ง-64 เป็นการสร้าง Bar Button Item ประเภท add นั้นในส่วนของ action ในบรรทัดที่ 55 คือเป็นการบอกว่าเมื่อทำการกดปุ่มนี้แล้วจะทำให้ทำฟังก์ชันอะไรต่อเช่นในที่นี้จะหมายถึงเมื่อกดปุ่มแล้วจะให้ไปทำฟังก์ชัน goForward ดังรูปที่ ง-65

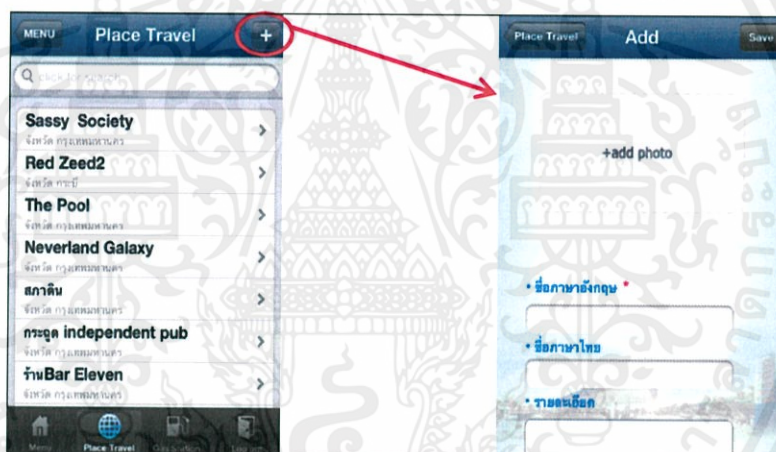
```

301 -(void)goForward
302 {
303     PlaceTravelAddViewController *placeTravelAddController =
304     [[self storyboard] instantiateViewControllerWithIdentifier:@"PlaceTravelAdd"];
305     [[self navigationController] pushViewController:placeTravelAddController animated:YES];
306 }
307 }
308

```

รูปที่ ง-65 ฟังก์ชัน goForward

ในฟังก์ชัน goForward ในที่นี้นั้นจะเป็นการบอกถึงการ Push หน้า PlaceTravelAdd Controller มาวางทับหน้าปัจจุบัน ดังรูปที่ ง-66



รูปที่ ง-66 การ Push หน้า PlaceTravelAddController

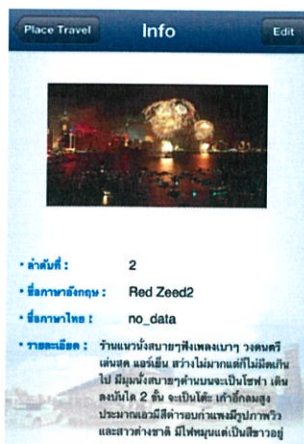
จากรูปที่ ง-67 จะเป็นการสร้าง Bar Button Item ประเภท edit

```

1045 -(void)createButton
1046 {
1047     //Edit Button
1048     UIBarButtonItem *editButton = [[UIBarButtonItem alloc] initWithBarButtonSystemItem:
1049     UIBarButtonSystemItemEdit target:self action:@selector(edit)];
1050
1051     self.navigationItem.rightBarButtonItem = editButton;
1052 }
1053

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง-68 หน้า PlaceTravelViewController

จากรูปที่ ง-67 จะเป็นการสร้าง Bar Button Item ประเภท edit และเมื่อทำการกดปุ่มนี้แล้วก็จะไปทำงานต่อที่ฟังก์ชัน edit ดังรูปที่ ง-69 โดยฟังก์ชัน edit จะทำการ push หน้าหน้า PlaceTravelEditController มาวางทับหน้าปัจจุบันพร้อมทั้งในการเรียกหน้า placeTravelEditController นั้นจะมีการเรียกฟังก์ชัน setInputDic ที่อยู่ใน PlaceTravelEditController ด้วยซึ่งจะมีการส่งข้อมูลที่มีอยู่ในหน้า PlaceTravelViewController เพื่อไปแสดงในหน้า PlaceTravelEditController ต่อไป ดังรูปที่ ง-70

```

974 -(void)edit
975 {
976     PlaceTravelEditViewController *placeTravelEditController =
977     [[self storyboard] instantiateViewControllerWithIdentifier:@"editplacetravel"];
978     [placeTravelEditController setInputDic:place_dic];
979
980     [[self navigationController] pushViewController:placeTravelEditController animated:NO];
981
982 }
983
984
  
```

รูปที่ ง-69 ฟังก์ชัน edit

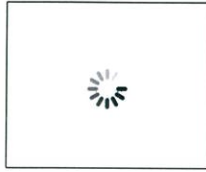


เอกสารนี้เป็นเอกสารที่สงวนไว้ใช้ประโยชน์ด้านการค้า

รูปที่ ง-70 การ Push หน้า PlaceTravelEditController ทุกครั้งที่มีการนำไปใช้

- การสร้าง Activity Indicator

ในระหว่างการโหลดข้อมูลจาก REST เว็บเซอร์วิสนั้นเมื่อระหว่างที่ยังโหลดข้อมูลไม่เสร็จ จะมีตัว Activity Indicator ขึ้นมาแสดงเพื่อบอกว่าข้อมูลยังโหลดไม่เสร็จดังรูปที่ ง-71



รูปที่ ง-71 Activity Indicator

```

1171 -(void)setActivityIndicator
1172 {
1173     overlayView = [[UIView alloc] initWithFrame:[UIScreen mainScreen].bounds];
1174     overlayView.backgroundColor = [UIColor colorWithRed:0 green:0 blue:0 alpha:0.3];
1175     activityIndicator = [[UIActivityIndicatorView alloc] initWithActivityIndicatorStyle:
1176                         UIActivityIndicatorViewStyleWhiteLarge];
1177 }
1178
  
```

รูปที่ ง-72 การกำหนดค่า Activity Indicator

จากรูปที่ ง-72 เป็นการกำหนดค่าในการสร้าง Activity Indicator โดยในบรรทัดที่ 1173 จะเป็นการ Initialize ในการสร้าง Activity Indicator ในบรรทัดที่ 1174 จะเป็นการกำหนดสีของพื้นหลัง และในบรรทัดที่ 1175 จะเป็นการกำหนดขนาดของ Activity Indicator

จากรูปที่ ง-73 จะเป็นการเรียกใช้งาน Activity Indicator โดยในบรรทัดที่ 712-715 จะเป็นการสร้าง Activity Indicator ส่วนบรรทัดที่ 720-721 จะเป็นการลบ Activity Indicator ออก

```

711 -(void)requestDidLoad:(RKRequest*)request{
712     activityIndicator.center = overlayView.center;
713     [overlayView addSubview:activityIndicator];
714     [activityIndicator startAnimating];
715     [self.navigationController.view addSubview:overlayView];
716 }
717
718 -(void)request:(RKRequest*)request didReceiveResponse:(RKResponse *)response{
719     [activityIndicator removeFromSuperview];
720     [overlayView removeFromSuperview];
721 }
722
  
```

รูปที่ ง-73 การเรียกใช้งาน Activity Indicator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Action Sheet

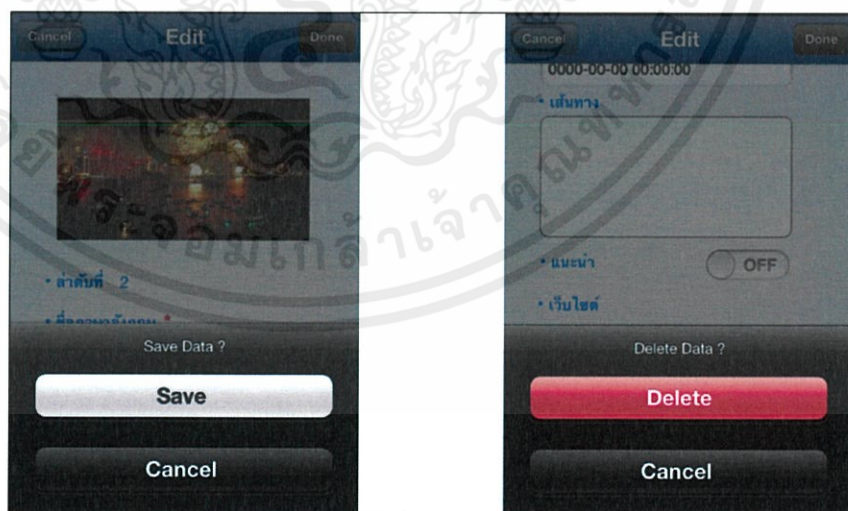
Action Sheet เป็นป๊อปอัพที่จะแสดงขึ้นมาเมื่อทำการกดปุ่ม Button โดยจากรูปที่ ง-74 จะเป็นการแสดง Action Sheet ขึ้นมาเมื่อทำการกดปุ่ม Save หรือ Delete โดย Action Sheet ที่ขึ้นมา นั้นจะเป็นการถามเพื่อตรวจสอบความแน่ใจของผู้ใช้ในการทำ Action นั้นๆ โดยในบรรทัดที่ 1184-1186 กับ บรรทัดที่ 1196-1198 จะเป็นการกำหนดคำที่จะแสดงขึ้นมา และกำหนดประเภทของ Button ที่แสดงขึ้นมาอย่างเช่น ถ้าเป็น Button Delete ในบรรทัดที่ 1197 ก็จะเป็น Button ที่มีสีแดง เป็นต้น

```

1179 -(IBAction)showActionSheet
1180 {
1181     if (Edit)
1182     {
1183
1184         UIAlertController *confirmsave = [[UIAlertSheet alloc] initWithTitle:@"Save Data ?"
1185         delegate:self cancelButtonTitle:@"Cancel " destructiveButtonTitle:nil
1186         otherButtonTitles:@"Save ", nil];
1187
1188         confirmsave.actionSheetStyle = UIAlertControllerStyleBlackOpaque;
1189         [confirmsave showInView:[UIApplication sharedApplication].keyWindow];
1190     }
1191 }
1192
1193 else
1194 {
1195     Edit = YES;
1196     UIAlertController *confirmsave = [[UIAlertSheet alloc] initWithTitle:@"Delete Data ?"
1197     delegate:self cancelButtonTitle:@"Cancel " destructiveButtonTitle:@"Delete"
1198     otherButtonTitles:nil, nil];
1199
1200     confirmsave.actionSheetStyle = UIAlertControllerStyleBlackOpaque;
1201     [confirmsave showInView:[UIApplication sharedApplication].keyWindow];
1202 }
1203 }
1204 }
1205 }

```

รูปที่ ง-74 ฟังก์ชัน Show Action Sheet



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการ **รูปที่ ง-75 การแสดง Action Sheet** อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Label

การสร้าง Label นั้นสามารถสร้างได้จากการโค้ดโปรแกรมได้ด้วยโดยไม่ต้องสร้างผ่านทาง MainStoryboard.storyboard เพียงอย่างเดียว โดยการสร้างจะทำได้จากการประกาศตัวแปรที่จะทำเป็น Label ไว้ในไฟล์ .h ซึ่งก็คือไฟล์ interface ดังรูปที่ ง-76 ซึ่งในที่นี้จะเป็นตัวอย่างของคลาส PlaceTravelViewController

```

21     UILabel *placeId;
22     UILabel *placeIdData;
23     UILabel *longitude;
24     UILabel *longitudeData ;
25     UILabel *PriceRange ;
26     UILabel *PriceRangeData ;
27     UILabel *Open ;
28     UILabel *OpenData ;
29     UILabel *Promotion ;
30     UILabel *PromotionData ;
31     UILabel *PromotionStart ;
32     UILabel *PromotionStartData ;
33     UILabel *PromotionEnd ;
34     UILabel *PromotionEndData ;
35     UILabel *Direction ;
36     UILabel *DirectionData ;
37     UILabel *Recommend ;
38     UILabel *RecommendData ;
39     UILabel *Website ;
40     UILabel *WebsiteData;
41     UILabel *CreateDate ;
42     UILabel *CreateDateData ;
43     UILabel *UpdateDate;
44     UILabel *UpdateDateData ;

```

รูปที่ ง-76 การประกาศตัวแปรในการสร้าง Label

โดยหลังจากทำการประกาศตัวแปรในไฟล์ .h แล้ว จะต้องมาทำการ Initialize ในไฟล์ .m ด้วยดังรูปที่ ง-77

```

986 -(void)setupConfigview
987 {
988     format = [[NSDateFormatter alloc] init];
989     placeId = [[UILabel alloc] init];
990     placeIdData = [[UILabel alloc]init];
991     name_th = [[UILabel alloc] init];
992     name = [[UILabel alloc] init];
993     nameData = [[UILabel alloc] init];
994     name_th_data= [[UILabel alloc] init];
995     content = [[UILabel alloc] init];
996     contentData = [[UILabel alloc] init];
997     address = [[UILabel alloc] init];
998     addressData = [[UILabel alloc] init];
999     province = [[UILabel alloc] init];
1000    provinceData = [[UILabel alloc] init];
1001    phonecall = [[UILabel alloc] init];
1002    phonecallData = [[UILabel alloc] init];
1003    iphonecall = [[UILabel alloc] init];
1004    iphonecallData = [[UILabel alloc] init];
1005    latitude=[[UILabel alloc] init];
1006    latitudeData=[[UILabel alloc] init];
1007    longitude = [[UILabel alloc] init];
1008    longitudeData= [[UILabel alloc]init];
1009    PriceRange = [[UILabel alloc] init];
1010    PriceRangeData = [[UILabel alloc]init];
1011    Open = [[UILabel alloc] init];
1012    OpenData = [[UILabel alloc]init];
1013    Promotion = [[UILabel alloc] init];
1014    PromotionData = [[UILabel alloc]init];
1015    PromotionStart = [[UILabel alloc] init];
1016    PromotionStartData = [[UILabel alloc]init];
1017    PromotionEnd = [[UILabel alloc] init];
1018    PromotionEndData = [[UILabel alloc]init];

```

รูปที่ ง-77 การ Initialize Label

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้นหลังจากการประกาศตัวแปร และทำการ Initialize เสร็จแล้ว ต่อไปจะเป็นการทำการสร้าง Label ขึ้นมาดังรูปที่ ง-78 โดยจากรูปในบรรทัดที่ 170 และ 193 ในส่วนที่เป็น .text จะเป็นการ

กำหนดข้อความภายใน Label ซึ่งในที่นี้จะแตกต่างกันตรงที่ใน placeId นั้นจะเป็นการกำหนดข้อความเข้าไปตรงๆเลย ส่วน placeIdData นั้นจะเป็นการเอาข้อมูลที่เก็บอยู่ในตัวแปร place_dic มาแสดง ส่วนในบรรทัดอื่นๆนั้นจะเป็นการกำหนดค่าต่างๆดังนี้ ในบรรทัดที่ 173 กับ 180 จะเป็นการกำหนดสีพื้นหลังกับสีตัวอักษร ในบรรทัดที่ 174 กับ 175 จะเป็นการกำหนดฟอนต์ของตัวอักษรในบรรทัดที่ 177 จะเป็นการตัดคำในการขึ้นบรรทัดใหม่ ในบรรทัดที่ 183 จะเป็นการกำหนดตำแหน่งของ Label ใน Scroll Viewว่าจะให้สร้างที่ตำแหน่งใด และในบรรทัดที่ 184 จะเป็นการเอา Label ที่สร้างมานี้ใส่ลงในหน้า Scroll View

```

168 //Place ID
169
170 placeId.text = @"• ลำดับที่ : ";
171
172 placeId.numberOfLines = 0;
173 placeId.backgroundColor = [UIColor whiteColor];
174 placeId.font = [UIFont fontWithName:@"Helvetica" size:14];
175 placeId.font = [UIFont boldSystemFontOfSize:14];
176
177 CGSize size = [placeId.text sizeWithFont: placeId.font constrainedToSize:
178 CGSizeMake(250, 5000) lineBreakMode: UILineBreakModeHeadTruncation];
179
180 placeId.textColor = [UIColor colorWithRed:(0/255.0) green:(114/255.0)
181 blue:(201/255.0) alpha:1];
182
183 placeId.frame = CGRectMake(15, 215, size.width, size.height);
184 [self.scrollView addSubview:placeId];
185
186 [placeId setBackgroundColor:[UIColor clearColor]];
187
188
189
190
191 //Place ID data
192
193 placeIdData.text = [[[place_dic objectForKey:@"body"] objectAtIndex:0]
194 objectForKey:@"place_id"];
195
196 placeIdData.numberOfLines = 0;
197 placeIdData.backgroundColor = [UIColor whiteColor];
198 placeIdData.font = [UIFont fontWithName:@"Helvetica" size:14];
199
200 size = [placeIdData.text sizeWithFont:placeIdData.font constrainedToSize:
201 CGSizeMake(250, 5000) lineBreakMode: UILineBreakModeWordWrap];
202
203 placeIdData.frame = CGRectMake(width1, 215, size.width, size.height);
204 [self.scrollView addSubview:placeIdData];
205
206 [placeIdData setBackgroundColor:[UIColor clearColor]];
207

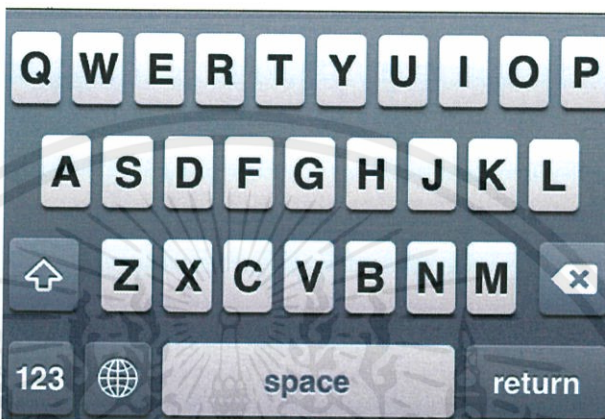
```

รูปที่ ง-78 การกำหนดค่าภายใน Label

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้างปุ่ม Done ในคีย์บอร์ด

เมื่อทำการกด Text Field นั้นโดยปกติแล้วจะไม่มีปุ่ม Done มาให้ดังรูปที่ ง-79 ดังนั้นจึงต้องทำการสร้างปุ่ม Done ให้กับคีย์บอร์ดเพื่อเป็นการบอกว่าทำการใส่ข้อมูลลงใน Text Field เสร็จเรียบร้อยแล้วดังรูปที่ ง-80 ในฟังก์ชัน textFieldDidBeginEditing นั้นจะทำงานเมื่อมีการกดไปที่ Text Field

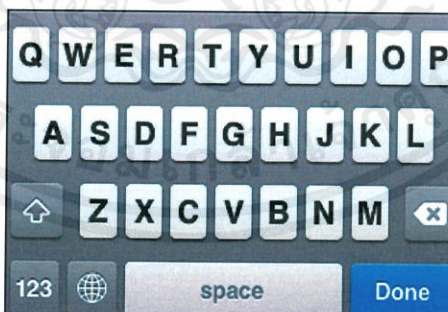


รูปที่ ง-79 คีย์บอร์ดปกติ

```

114 -(void)textFieldDidBeginEditing:(UITextField*)textField
115 {
116     [self moveUp:textField with:@"textField"];
117     [textField setReturnKeyType:UIReturnKeyDone];
118 }
119
120
  
```

รูปที่ ง-80 ฟังก์ชัน textFieldDidBeginEditing

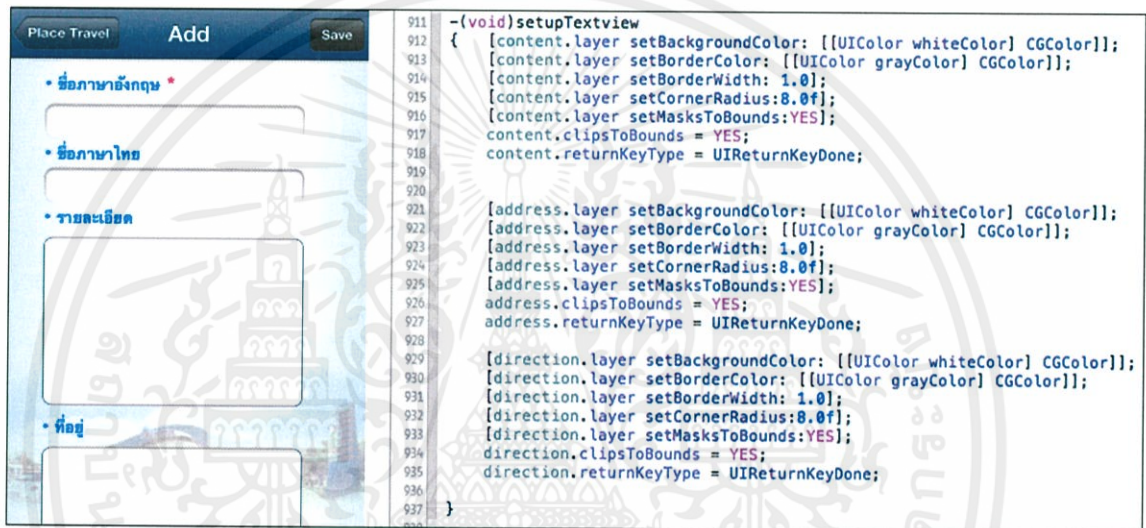


รูปที่ ง-81 คีย์บอร์ดมีปุ่ม Done

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Text View ให้คล้ายกับ Text Field

จากหัวข้อการสร้าง Text View ใน MainStoryboard.storyboard นั้นรูปร่างหน้าของ Text View ที่ออกมานั้นจะเป็นเรียบๆ สีขาวไม่มีขอบ โดยถ้าหากต้องการสร้าง Text View ให้ดูมีขอบเหมือนกับ Text Field สามารถทำได้ดังนี้ โดยหลังจากที่ทำการเชื่อมความสัมพันธ์ระหว่าง Text View กับตัวแปร เสร็จแล้วก็ต้องมาทำการเขียนโค้ดในไฟล์ .m เพื่อกำหนดขอบและเส้นสีขอบให้กับ Text View ดังรูปที่ ง-82 อย่างเช่นในบรรทัดที่ 912-917 จะเป็นการกำหนดค่าของสีพื้นหลัง, สีเส้นขอบ และ กำหนดขอบ ส่วนในบรรทัดที่ 918 จะเป็นการสร้างปุ่ม Done ในคีย์บอร์ดของ Text View ดังรูปที่ ง-83



รูปที่ ง-82 การกำหนดค่าให้กับ Text View

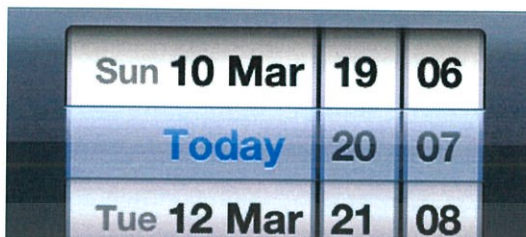


รูปที่ ง-83 คีย์บอร์ดมีปุ่ม Done ของ Text View

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การสร้าง Date Picker

Date Picker เป็นส่วนที่ใช้ในการเลือกวัน/เดือน/ปี ดังรูปที่ ง-84



รูปที่ ง-84 Date Picker

การสร้าง Date Picker นั้นสามารถทำได้จากการเขียนโค้ดโปรแกรมดังรูปที่ ง-85

```

128 -(BOOL)textFieldShouldBeginEditing:(UITextField*)textField
129 {
130     picker.hidden = YES;
131     province_picker.hidden = YES;
132     pricerange_picker.hidden = YES;
133     place_travel_type_picker.hidden = YES;
134     food_type_picker.hidden = YES;
135     toolbar.hidden = YES;
136
137     if (textField == textfield_Date || textField == textfield_end_Date )
138     {
139         if (temp2 != nil)
140         {
141             [temp2 resignFirstResponder];
142         }
143         temp = textField;
144         if(picker==nil){
145
146             picker = [[UIDatePicker alloc] init];
147             [picker setFrame:CGRectMake(0,274,320,162)];
148
149             toolbar = [[UIToolbar alloc] initWithFrame:CGRectMake(0, 274-44, 320, 44)];
150             cancel = [[UIBarButtonItem alloc] initWithBarButtonSystemItem:
151                 UIBarButtonSystemItemCancel target:self action:@selector(cancel)];
152
153             done = [[UIBarButtonItem alloc] initWithBarButtonSystemItem:
154                 UIBarButtonSystemItemDone target:self action:@selector(save)];
155
156             UIBarButtonItem *blank = [[UIBarButtonItem alloc] initWithBarButtonSystemItem:
157                 UIBarButtonSystemItemFixedSpace target:nil action:nil];
158
159             blank.width = 188;
160             [toolbar setItems:[NSArray alloc] initWithObjects:cancel,blank,done, nil]];
161             [self.view addSubview:picker];
162             [self.view addSubview:toolbar];
163
164         }
165         else if (picker.hidden==YES) {
166             picker.hidden = NO;
167             toolbar.hidden = NO;
168         }
169
170         else {
171             picker.hidden = YES;
172             toolbar.hidden = YES;
173
174         }
175         return NO;
176     }

```

รูปที่ ง-85 ฟังก์ชัน textFieldShouldBeginEditing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ฟังก์ชัน textFieldShouldBeginEditing ในรูปที่ ง-85 จะถูกเรียกใช้เมื่อ Text Field ของ Date Picker ถูกกด และในบรรทัดที่ 137 ระบบจะมีการตรวจสอบว่า Text Field ที่ทำการกดไปนั้นเป็น Text Field ช่องไหนโดยถ้าเป็น textfield_Date หรือ textfield_end_Date โดยในบรรทัดที่ 146-147

จะเป็นการ Initialize Date Picker และ การกำหนดตำแหน่งของ Date Picker ส่วนในบรรทัดที่ 149-162 จะเป็นการสร้าง Toolbar ไว้ด้านบน Date Picker ดังรูปที่ ง-86 ซึ่งบรรทัดที่ 149 จะเป็นการกำหนดตำแหน่งของ Toolbar บรรทัดที่ 150 และ 153 จะเป็นการสร้างปุ่ม ในการบันทึกค่าใน Date Picker ที่เลือกหรือยกเลิกการใช้งานของ Date Picker โดยจะมีฟังก์ชันเมื่อทำการกดปุ่ม done กับ cancel ดังรูปที่ ง-87 โดยถ้าเป็นการบันทึกค่า จะไปเรียกฟังก์ชัน done บรรทัดที่ 156-159 จะแสดงโค้ดในการสร้าง UIBarButtonItem ที่เป็นพื้นที่ว่างๆเพื่อสร้างระยะห่างระหว่างปุ่ม cancel กับปุ่ม save โดยกำหนดความกว้างไว้ที่ 188 ในบรรทัดที่ 160 จะเป็นการใส่ค่าปุ่ม cancel ปุ่มว่าง และปุ่ม save ไว้ใน Toolbar โดยใน 165-173 จะเป็นการตรวจสอบการแสดงผลของ Picker กับ Toolbar ว่า จะให้แสดงเมื่อไร



รูปที่ ง-86 Date Picker ที่มี Toolbar

```

562 -(void)save
563 {
564     [self done:temp];
565 }
566
583 -(void)cancel
584 {
585     picker.hidden = YES;
586     toolbar.hidden = YES;
587     province_picker.hidden = YES;
588     place_travel_type_picker.hidden = YES;
589     food_type_picker.hidden = YES;
590     pricerange_picker.hidden = YES;
591 }

```

รูปที่ ง-87 ฟังก์ชันเมื่อกดปุ่ม done หรือ cancel

```

495 else
496 {
497     NSDateFormatter *format = [[NSDateFormatter alloc] init];
498     [format setDateFormat:@"yyyy-MM-dd HH:mm:ss"];
499     NSString *date = [format stringFromDate:[picker date]];
500
501     [temp setText:date];
502     picker.hidden = YES;
503     toolbar.hidden = YES;
504 }

```

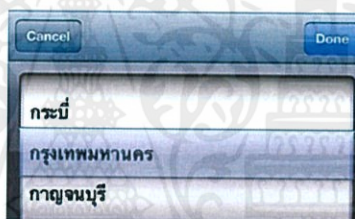
รูปที่ ง-88 โค้ดภายในฟังก์ชัน done

จากรูปที่ ง-88 จะเป็น โค้ดในส่วนของ Date Picker ที่อยู่ในฟังก์ชัน done โดยจากรูป ง-88 เอกสารนี้ ในบรรทัดที่ 496-498 จะเป็นการกำหนดค่าข้อมูลวัน/เดือน/ปี ที่เลือกให้อยู่ในฟอร์มตามที่ต้องการ ไม่ว่าจะกรณีกำหนดในบรรทัดที่ 497 บรรทัดที่ 500 จะเป็นการเอาค่าและนำค่าที่ถูกแปลงไปแสดงใน Text Field ดังรูปที่ ง-89 โดยจะมีการซ่อน Picker และ Toolbar หลังจากที่ถูกกดปุ่ม done (บรรทัดที่ 501-502)

รูปที่ ง-89 การแสดงผลการเลือกข้อมูลจาก Date Picker

- การสร้าง Picker View

Picker View คือส่วนที่ใช้ในการเลือกข้อมูลต่างๆที่มีหลายค่า รูปที่ ง-90 จะเป็นตัวอย่างของการสร้าง Picker View ของจังหวัดในประเทศไทย



รูปที่ ง-90 UIPickerView

จากรูปที่ ง-91 การสร้าง UIPickerView จะเหมือนกับการสร้าง DatePicker แต่ต่างกันตรงที่มีการตรวจว่า Text Field ที่กดเลือกมานั้นใช่ province หรือไม่ และในบรรทัดที่ 185 จะมีการใช้ Initialize UIPickerView แทน DatePicker

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

178     else if (textField == province)
179     {
180         if (temp2 != nil) {
181             [temp2 resignFirstResponder];
182         }
183         temp = textField;
184         if(province_picker==nil){
185             province_picker = [[UIPickerView alloc] init];
186             [province_picker setFrame:CGRectMake(0,274,320,162)];
187             province_picker.showsSelectionIndicator = YES;
188             province_picker.delegate = self;
189
190             toolbar=[[UIToolbar alloc] initWithFrame:CGRectMake(0, 274-44, 320, 44)];
191             cancel = [[UIBarButtonItem alloc] initWithBarButtonSystemItem:
192                 UIBarButtonSystemItemCancel target:self action:@selector(cancel)];
193
194             done = [[UIBarButtonItem alloc] initWithBarButtonSystemItem:
195                 UIBarButtonSystemItemDone target:self action:@selector(save)];
196
197             UIBarButtonItem *blank = [[UIBarButtonItem alloc] initWithBarButtonSystemItem:
198                 UIBarButtonSystemItemFixedSpace target:nil action:nil];
199
200             blank.width = 188;
201             [toolbar setItems:[NSArray alloc] initWithObjects:cancel,blank,done, nil]];
202             [self.view addSubview:province_picker];
203             [self.view addSubview:toolbar];
204         }
205     }
206     else if (province_picker.hidden==YES) {
207         province_picker.hidden = NO;
208         toolbar.hidden = NO;
209     }
210 }
211 else {
212     province_picker.hidden = YES;
213     toolbar.hidden = YES;
214 }
215 }
216 return NO;
217 }
218 }
219 }
220 }

```

รูปที่ ง-91 การสร้าง Picker View

การนำข้อมูลใส่ใน Picker View นั้นทำผ่านโค้ดได้โดยต้องทำผ่านฟังก์ชันต่างๆ โดยเริ่มจากรูปที่ ง-92 จะเป็นการกำหนดค่าจำนวน Component ใน Picker View ซึ่งในที่นี้จะทำการสร้าง Picker View ของจังหวัด ดังนั้นจึงใช้แค่ Component 1 แลวก้เพียงพอแล้ว

```

361 - (NSInteger)numberOfComponentsInPickerView:(UIPickerView *)pickerView {
362     if (pickerView == province_picker )
363     { return 1; }
364 }

```

รูปที่ ง-92 ฟังก์ชัน numberOfComponentsInPickerView

จากรูปที่ ง-93 จะเป็นฟังก์ชันในการสร้าง Row ใน Component ซึ่งในที่นี้ก็คือจำนวนของจังหวัดในประเทศไทย โดยจะทำการนับจากข้อมูลที่ถูกเก็บอยู่ในตัวแปร province_dic ในบรรทัดที่

377

```

373 - (NSInteger)pickerView:(UIPickerView *)pickerView numberOfRowsInComponent:(NSInteger)component
374 {
375     if (pickerView == province_picker )
376     {return [[province_dic objectForKey:@"body"] count];}
377 }

```

รูปที่ ง-93 ฟังก์ชันการสร้าง Row ใน Picker View

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่ถูกต้องเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ ง-94 จะเป็นฟังก์ชันในการสร้างชื่อ title ที่จะแสดงในแต่ละ Rows ซึ่งในที่นี้ก็คือชื่อของจังหวัดในประเทศไทย โดยจะทำการสร้างจากข้อมูลที่ถูกเก็บอยู่ในตัวแปร province_dic ในบรรทัดที่ 397-398

```

393 - (NSString *)pickerView:(UIPickerView *)pickerView
394 titleForRow:(NSInteger)row forComponent:(NSInteger)component {
395
396     if (pickerView == province_picker )
397     {return [[[province_dic objectForKey:@"body"]
398         objectAtIndex:row] objectForKey:@"province_name"];
399     }

```

รูปที่ ง-94 ฟังก์ชันในการสร้างชื่อ Title ใน Picker View

จากรูปที่ ง-95 จะเป็นฟังก์ชันที่ทำงานเมื่อเราทำการกดเลือก Row หรือ ชื่อจังหวัด โดยในบรรทัดที่ 426 จะทำการเก็บลำดับ Row ที่เลือกเอาไว้ในตัวแปร select_province โดยที่ลำดับของ Row นั้นจะเริ่มจาก 0

```

421 -(void)pickerView:(UIPickerView *)pickerView
422 didSelectRow:(NSInteger)row inComponent:(NSInteger)component
423 {
424
425     if (pickerView == province_picker )
426     {select_province = row;
427     }

```

รูปที่ ง-95 ฟังก์ชันในการเลือก Row ใน Picker View

จากรูปที่ ง-96 เมื่อทำการเลือกข้อมูลใน Picker View เสร็จแล้วต้องการที่จะ save หรือ cancel ก็จะทำฟังก์ชันนั้นๆเหมือนดังใน Date Picker

```

562 -(void)save
563 {
564     [self done:temp];
565 }
566
567
568
569
570
571
572
573 -(void)cancel
574 {
575     picker.hidden = YES;
576     toolbar.hidden = YES;
577     province_picker.hidden = YES;
578     place_travel_type_picker.hidden = YES;
579     food_type_picker.hidden = YES;
580     pricerange_picker.hidden = YES;
581 }

```

รูปที่ ง-96 ฟังก์ชันเมื่อกดปุ่ม done หรือ cancel

เมื่อทำการกดปุ่ม save ก็จะมาทำฟังก์ชัน done เพื่อทำการแสดงข้อมูลใน Picker View ที่ทำการเลือกลงใน Text Field ดังรูปที่ ง-97 โดยจากรูปที่ ง-98 ในบรรทัดที่ 455-456 จะเป็นการนำข้อมูลภายในตัวแปร province_dic โดยใช้ลำดับที่ของข้อมูลจากตัวแปร select_province เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง-97 การแสดงผลจากการเลือกข้อมูลใน Picker View

```

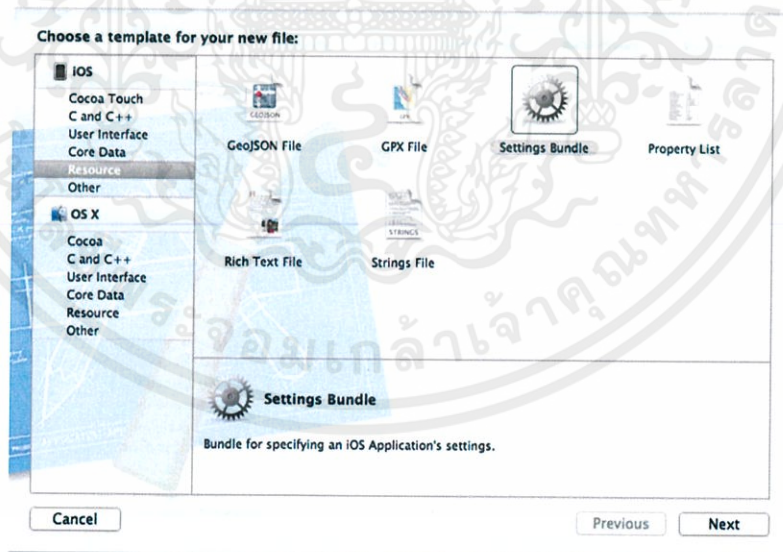
452 -(void)done:(UITextField*)textField
453 {
454     if (textField == province)
455     {
456         province.text = [[[province_dic objectForKey:@"body"]
457             objectAtIndex:selected_province ] objectForKey:@"province_name"];
458         province_picker.hidden = YES;
459         toolbar.hidden = YES;
460     }

```

รูปที่ ง-98 ฟังก์ชัน done ของ Picker View

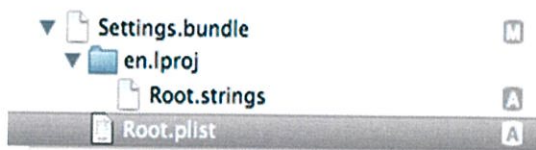
- การสร้างการติดต่อแอปพลิเคชันผ่านทางเครื่อง iPhone

การติดต่อ Application ผ่านทางตัวเครื่อง iPhone นั้นทำได้โดยไปตั้งค่าภายใน Setting ซึ่งเป็นการทำเพื่อให้ง่ายต่อการติดต่อไปที่ IP Address ของ REST เว็บเซอร์วิสโดยสามารถเปลี่ยนใน Setting ในเครื่อง iPhone ได้เลย โดยที่ไม่ต้องทำการ Build แอปพลิเคชันลงเครื่อง iPhone ใหม่ ทำได้โดยการคลิกขวาเพื่อทำการเพิ่มไฟล์เข้าไปใหม่ใน Project โดยไฟล์ที่ทำการเพิ่มลงไปนั้นจะเป็นประเภท Resource ที่ชื่อว่า Settings Bundle ดังรูปที่ ง-99



รูปที่ ง-99 การสร้าง Settings Bundle

หลังจากทำการเพิ่มไฟล์ Setting Bundle แล้วต้องเข้าไปที่ไฟล์ Root.plist เพื่อทำการตั้งค่าต่างๆที่จะแสดงในหน้า Setting ดังรูปที่ ง-100

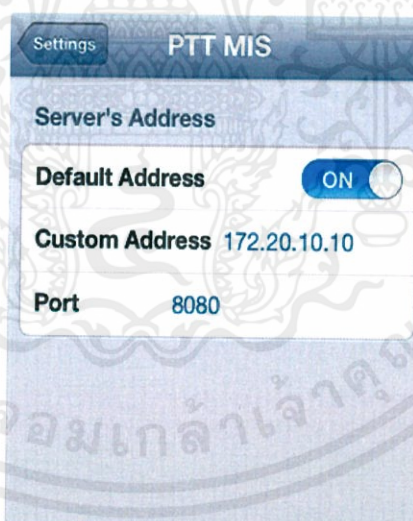


รูปที่ ง-100 ไฟล์ Root.plist

เมื่อเข้ามาที่ไฟล์ Root.plist ซึ่งจะมีให้ตั้งค่าต่างๆอย่างเช่น ใน Preference Item จะเป็นการตั้งค่าในการแสดง Item ต่างๆในหน้า Setting โดยในที่นี้จะยกตัวอย่างของ Item 3 ที่กำหนดชื่อว่า Port และมีตัวแปรในการอ้างอิงคือ port ซึ่งเป็นประเภท Text Field ดังรูปที่ ง-101 ซึ่งจะได้ผลลัพธ์ออกมาในรูปที่ ง-102

Key	Type	Value
▼ iPhone Settings Schema	Dictionary	(1 item)
▼ Preference Items	Array	(4 items)
▶ Item 0 (Group - Server's	Dictionary	(2 items)
▶ Item 1 (Toggle Switch - Default	Dictionary	(4 items)
▶ Item 2 (Text Field - Custom	Dictionary	(3 items)
▼ Item 3 (Text Field - Port)	Dictionary	(3 items)
Type	String	Text Field
Title	String	Port
Identifier	String	port

รูปที่ ง-101 การกำหนดค่าภายในไฟล์ Root.plist



รูปที่ ง-102 การแสดงผลของ Settings Bundle

ง.2.2 การติดต่อกับ REST เว็บเซอร์วิส

• การเชื่อมต่อแอปพลิเคชันเข้ากับ REST เว็บเซอร์วิส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า จากรูปที่ ง-103 เป็นการเชื่อมต่อไปยัง REST เว็บเซอร์วิส โดยเริ่มจากการโค้ดโปรแกรมในไฟล์ AppDelegate.h ซึ่งจะมีการใช้คำสั่ง `#import <RestKit/RestKit.h>` (บรรทัดที่ 9) เพื่อให้สามารถใช้ ฟังก์ชันต่างๆของทาง RestKit เพื่อทำการติดต่อกับ REST Web Service ได้ และมีการ

ประกาศตัวแปร Global Variable 3 ตัวคือ account password_account และ ip_image (ในบรรทัดที่ 15-17) เพื่อให้สะดวกต่อการนำไปใช้ในคลาสอื่นๆ เนื่องจากในการเรียกดูจาก REST เว็บเซอร์วิส นั้นจะต้องอ้างอิงไปยัง IP Address ด้วยเพื่อให้ง่ายเวลาเปลี่ยน IP Address จึงทำการเปลี่ยนแก้ที่ คลาส AppDelegate เพียงที่เดียวเท่านั้น ส่วนใน account และ password_account นั้นจะใช้ในการ ตรวจสอบเวลาที่จะทำการ Add Edit หรือ Delete ข้อมูลโดยจะมีการส่ง account และ password_account ไปตรวจสอบก่อนเพื่อป้องกันการทำการ Action ต่างๆที่มาจากการแฮคเข้า ระบบ

```

9  #import <RestKit/RestKit.h>
10 #import <UIKit/UIKit.h>
11
12
13
14
15 NSString *account;
16 NSString *password_account;
17 NSString *ip_image;
18
19 @interface AppDelegate : UIResponder <UIApplicationDelegate>
20
21
22 @property (strong, nonatomic) UIWindow *window;
23
24
25
26
27 @end
28

```

รูปที่ ง-103 ไฟล์ AppDelegate.h

รูปที่ ง-104 เป็นการแสดงโค้ดโปรแกรมในไฟล์ AppDelegate.m ซึ่งเป็นส่วนที่ใช้ในการ เชื่อมต่อแอปพลิเคชันไปที่เว็บเซอร์วิสในกรณีที่มีการเปลี่ยน IP Address ชั่วคราวโดยไม่ต้อง build แอปพลิเคชัน ใหม่ โดยการเชื่อมต่อนั้นจะมี 2 รูปแบบโดยจาก IP Address ที่เป็นค่า Default หรือ Custom IP Address ดังในบรรทัดที่ 35-44 ที่เป็น Default กับ Custom IP Address ซึ่งในบรรทัดที่ 24-32 จะเป็นการรับการกำหนดค่า IP Address จาก Setting ในเครื่อง iPhone ที่เคยสร้างไว้ในหัวข้อ การสร้างการติดต่อ แอปพลิเคชันผ่านทางเครื่อง iPhone และในบรรทัดที่ 37-42 จะเป็นการติดต่อ ไปยัง REST เว็บเซอร์วิส ด้วย Default IP Address และในบรรทัดที่ 43-62 จะเป็นการติดต่อไปยัง REST เว็บเซอร์วิสด้วย Custom IP Address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

17 - (BOOL)application:(UIApplication *)application
18 didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
19 {
20     sleep(1);
21     // Override point for customization after application launch.
22
23     //config application
24     NSUserDefaults *defaults = [NSUserDefaults standardUserDefaults];
25     NSArray *key = [NSArray arrayWithObjects:@"defaultAddress", @"customAddress", @"port", nil];
26     NSArray *object = [NSArray arrayWithObjects:
27         @"YES", @"", @"", nil];
28     NSDictionary *appDefaults = [NSDictionary dictionaryWithObjects:
29         object forKeys:key];
30     [defaults registerDefaults:appDefaults];
31     [defaults synchronize];
32
33     //get preference
34     BOOL enable = [defaults boolForKey:@"defaultAddress"];
35
36     if (enable) {
37         //ip server
38         ip_image = @"http://xxxxxxxxxxx/weloveptt.com";
39
40         //rest
41         [RKClient clientWithBaseURLString:@"http://xxxxxxxxxxx/Rest"];
42     } else {
43         NSString *address = [defaults stringForKey:@"customAddress"];
44         NSString *port = [defaults stringForKey:@"port"];
45         //ip server
46         ip_image = [NSString stringWithFormat:@"http://%/weloveptt.com",address];
47
48         if ([port isEqualToString:@""])
49         {
50             //rest
51             [RKClient clientWithBaseURLString:[NSString stringWithFormat:
52                 @"http://%/Rest",address]];
53         }
54         else
55         {
56             //rest
57             [RKClient clientWithBaseURLString:[NSString stringWithFormat:
58                 @"http://%:@%/Rest",address,port]];
59         }
60     }
61 }
62
63 //username
64 account = [[NSString alloc] init];
65 //password
66 password_account = [[NSString alloc] init];
67
68
69
70 return YES;
71 }

```

รูปที่ ง-104 ไฟล์ AppDelegate.m

• การ Get ข้อมูล

ในการ Get ข้อมูลจากฐานข้อมูลโดยผ่านทาง REST เว็บเซอร์วิสนั้นมีอยู่ 2 แบบด้วยกันคือ การ Get ข้อมูลแบบไม่ส่งพารามิเตอร์ไปซึ่งจะเป็นการ Get ข้อมูลมาทั้งหมด และการ Get ข้อมูลแบบส่งพารามิเตอร์ไปซึ่งจะเป็นการ Get ข้อมูลแบบเฉพาะเจาะจง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การ Get ข้อมูลแบบไม่ส่งพารามิเตอร์

จากรูปที่ ง-105 จะเป็นการ Get ข้อมูลแบบไม่ส่งพารามิเตอร์ซึ่งในบรรทัดที่ 219 จะเป็นการส่ง Resource Path ในการ Get ข้อมูลจาก REST เว็บเซอร์วิส

```

216 -(void)requestData
217 {
218     NSString *resourcePath;
219     resourcePath = @"/Placetravel";
220     [[RKClient sharedClient] get:resourcePath delegate:self];
221
222 }
223

```

รูปที่ ง-105 ฟังก์ชัน requestData

จากรูปที่ ง-106 เป็นการสร้าง Activity Indicator เมื่อเริ่มโหลดข้อมูลจาก REST เว็บเซอร์วิส

```

246 -(void)requestDidStartLoad:(RKRequest*) request
247 {
248     overlayView = [[UIView alloc] initWithFrame:[UIScreen mainScreen].bounds];
249     overlayView.backgroundColor = [UIColor colorWithRed:0 green:0 blue:0 alpha:0.3];
250     activityIndicator = [[UIActivityIndicatorView alloc]
251         initWithActivityIndicatorStyle:UIActivityIndicatorViewStyleWhiteLarge];
252     activityIndicator.center = overlayView.center;
253     [overlayView addSubview:activityIndicator];
254     [activityIndicator startAnimating];
255     [self.navigationController.view addSubview:overlayView];
256 }
257

```

รูปที่ ง-106 การสร้าง Activity Indicator

จากรูปที่ ง-107 เป็นการลบ Activity Indicator เมื่อโหลดข้อมูลจาก REST เว็บเซอร์วิสเสร็จ

```

258 -(void)request:(RKRequest*)request didReceiveResponse:(RKResponse *)response{
259     [activityIndicator removeFromSuperview];
260     [overlayView removeFromSuperview];
261 }

```

รูปที่ ง-107 การลบ Activity Indicator

จากรูปที่ ง-108 จะแสดงฟังก์ชันที่ทำงานเมื่อการ Get ข้อมูลมานั้นสำเร็จไม่มีข้อผิดพลาดอะไรโดยในบรรทัดที่ 229-230 จะเป็นการเก็บข้อมูลที่ได้รับมาจาก REST เว็บเซอร์วิส ซึ่งได้มาในรูปแบบของ JSON เก็บลงในตัวแปร place_dic และในบรรทัดที่ 232 จะเป็นการสร้างกล่องข้อมูลในอาร์เรย์ตามจำนวนของข้อมูล place_dic เพื่อใช้ในการ Search ข้อมูลของตัวแปร filteredPlaceTravel

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาดูเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ ใช้ขึ้นต้นในการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

224 -(void)request:(RKRequest*)request didLoadResponse:(RKResponse *)response{
225     if ([request isGET]) {
226         if ([response isOK]) {
227
228
229             RKJSONParserJSONKit *parser = [RKJSONParserJSONKit new];
230             place_dic = [parser objectFromstring:[response bodyAsString] error:nil];
231
232             self.filteredPlaceTravel = [NSMutableArray arrayWithCapacity:
233                 [[place_dic objectForKey:@"body"] count]];
234
235             [self.tableView reloadData];
236             [self.searchDisplayController.searchResultsTableView reloadData];
237
238             if (isAdd) {
239                 [self performSegueWithIdentifier:@"sendPlaceId" sender:self];
240             }
241         }
242     }
243 }
244 }

```

รูปที่ ง-108 การทำงานเมื่อได้รับข้อมูลครบถ้วน

- การ Get ข้อมูลแบบส่งพารามิเตอร์

การ Get ข้อมูลแบบส่งพารามิเตอร์นั้นจะต่างกับแบบไม่ส่งพารามิเตอร์ตรง Resource Path เท่านั้นดังรูปที่ ง-109 ซึ่งจะมีการส่งพารามิเตอร์ไปด้วยในบรรทัดที่ 68-69

```

64 -(void)requestData{
65
66     NSString *resourcePath = [NSString stringWithFormat:@"%s/Placetravel"];
67
68     [[RKClient sharedClient] get:resourcePath queryParameters:[NSDictionary
69         dictionaryWithObject:_detailItem forKey:@"id"] delegate:self];
70
71 }

```

รูปที่ ง-109 ฟังก์ชัน requestData

• การตรวจสอบรหัสผ่าน

ในการตรวจสอบรหัสผ่านนั้นจะเป็นการส่งรหัสผ่านไปตรวจสอบกับ REST เว็บเซอร์วิส แล้วทาง REST เว็บเซอร์วิส จะส่งผลลัพธ์กลับมาให้ทางไคลเอนต์ โดยเริ่มจากรูปที่ ง-110 จะเห็นได้ว่า คลาสไหนก็ตามที่มีการติดต่อกับ REST Web Service นั้นจะต้องมีการ import RestKit เพื่อให้สามารถติดต่อกับทาง REST เว็บเซอร์วิสได้ และ import AppDelegate เพื่อทำการใช้ค่า account กับ password_account ตัวเดียวกัน

```

8 #import "AppDelegate.h"
9 #import <RestKit/RestKit.h>
10 #import <RestKit/RKJSONParserJSONKit.h>
11 #import <UIKit/UIKit.h>
12 #import "MainMenuControllerViewController.h"
13
14 @interface LoginViewController : UIViewController{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ควรเผยแพร่หรือทำซ้ำโดยไม่ได้รับอนุญาต

รูปที่ ง-110 การ import RestKit

จากรูปที่ ง-111 จะเป็นฟังก์ชันที่ทำงานเมื่อทำการกดปุ่ม Log in โดยถ้าช่อง Username และ Password ไม่เป็นค่า Null แล้วจะทำการติดต่อกับ REST เว็บเซอร์วิส โดยส่ง username และ password ไปตรวจสอบดังในบรรทัดที่ 71-73

```

62  -(IBAction)login:(id)sender
63  {
64      if (username.text == NULL || password.text == NULL || [username.text isEqualToString:@""]
65          || [password.text isEqualToString:@""])
66      {
67          msg.text = @"please enter username and password";
68      }
69      else
70      {
71          NSString *resourcePath;
72          resourcePath = [NSString stringWithFormat:@"/Login/%@-%@", username.text, password.text];
73          [[RKClient sharedClient] get:resourcePath delegate:self];
74      }
75  }
76  }

```

รูปที่ ง-111 ฟังก์ชัน login

จากรูปที่ ง-112 จะเป็นกรณีที่มีการติดต่อกับ REST เว็บเซอร์วิส มีข้อผิดพลาดโดย error.code = -1003 จะหมายถึงการที่ไม่ได้เปิดใช้งาน internet หรือ wifi และ -1004 จะเป็นการบอกว่าไม่พบ REST เว็บเซอร์วิส ตาม IP Address ที่ระบุไว้ในคลาส AppDelegate

```

118  -(void)request:(RKRequest *)request didFailLoadWithError:(NSError *)error
119  {
120      if (error.code == -1003 || error.code == -1004)
121      {
122          msg.text = @"Error url not found";
123      }
124  }
125  }

```

รูปที่ ง-112 ข้อผิดพลาดในการติดต่อกับ REST เว็บเซอร์วิส

จากรูปที่ ง-113 เมื่อทำการส่ง username กับ password ไปตรวจสอบแล้วทาง REST เว็บเซอร์วิส ก็จะส่งผลลัพธ์กลับมาให้ซึ่งก็คือการ Get ข้อมูลจากทาง REST เว็บเซอร์วิส โดยจะเก็บผลลัพธ์ที่ REST เว็บเซอร์วิส ส่งกลับมาในตัวแปร login_dic ดังในบรรทัดที่ 111 และทำการเรียกฟังก์ชันตรวจสอบผลลัพธ์ที่เก็บในตัวแปร login_dic

```

104  -(void)request:(RKRequest*)request didLoadResponse:(RKResponse *)response
105  {
106      if ([request isGET])
107      {
108          if ([response isOK])
109          {
110              RKJSONParserJSONKit *parser = [RKJSONParserJSONKit new];
111              login_dic = [parser objectFromString:[response bodyAsString] error:nil];
112              [self checkPassword];
113          }
114      }
115  }
116  }

```

รูปที่ ง-113 การทำงานเมื่อได้รับข้อมูลครบถ้วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ขอสงวนสิทธิ์ในนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น ถือว่าเป็นฉบับร่างและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ ง-114 จะเป็นการตรวจสอบผลลัพธ์ที่ถูกเก็บในตัวแปร login_dic (บรรทัดที่ 81-82) ถ้าผลลัพธ์มีค่าเท่ากับ valid จะทำการเปลี่ยนหน้าไปยังหน้า MainMenuController ดังในบรรทัด

ที่ 84-88 และในบรรทัดที่ 89-93 จะเป็นการกำหนดค่าให้กับตัวแปร `account` และ `account_password` ซึ่งเป็นตัวแปร Global Variable เพื่อนำไปใช้ในคลาสอื่นๆในการ Add Edit หรือ Delete แต่ถ้าผลลัพธ์ไม่ใช่ `valid` ก็จะทำการแสดงข้อความที่ REST เว็บเซอร์วิส ส่งกลับมาให้ผู้ใช้ เห็นดังในบรรทัดที่ 97-98

```

79 - (void)checkPassword
80 {
81     if ([[login_dic objectForKey:@"body"] objectAtIndex:0]objectForKey:@"access"
82         isEqualToString:@"valid"])
83     {
84         MainMenuControllerViewController *mainMenu = [storyboard
85             instantiateViewControllerWithIdentifier:@"mainMenu"];
86         UINavigationController *navigation = [[UINavigationController alloc]
87             initWithRootViewController:mainMenu];
88         [self presentViewController:navigation animated:YES completion:nil];
89         account = username.text;
90         password_account = password.text;
91         username.text = @"";
92         password.text = @"";
93         msg.text = @"";
94     }
95     else
96     {
97         msg.text = [[[login_dic objectForKey:@"body"] objectAtIndex:0]
98             objectForKey:@"message"];
99     }
100 }

```

รูปที่ ง-114 ฟังก์ชัน checkPassword

• การ Add ข้อมูล

จากรูปที่ ง-115 จะเป็นการ Add ข้อมูลไปที่ฐานข้อมูลโดยผ่าน REST เว็บเซอร์วิส เริ่มจากการ `setValue` ในแต่ละค่าที่ต้องการ Add เข้ากับชื่อ Field ในฐานข้อมูล โดยเก็บค่าไว้ในตัวแปร `paramsRequest` (บรรทัดที่ 632-666) บรรทัดที่ 665-666 จะเป็นการเพิ่มค่า `account` และ `password_account` ไปด้วยเพื่อทำการพิสูจน์ตัวตนว่าการ Add นี้มาจากผู้ใช้ที่ถูกต้องนอกจากนี้ยังทำการ `setValue imagePath` เข้ากับ Field ในฐานข้อมูลที่ชื่อว่า `preview_image` โดยเก็บค่าไว้ในตัวแปร `paramsRequest` (บรรทัดที่ 675) ในบรรทัดที่ 676-678 จะแสดงโค้ดการ Add ข้อมูลของรูปเข้าไปยัง Folder ที่ทำการเก็บรูป ส่วนในบรรทัดที่ 683 จะเป็นคำสั่งในการ Post เพื่อทำการ Add ข้อมูลผ่าน REST เว็บเซอร์วิส โดยจะส่งข้อมูลที่ถูกรวบรวมทั้งหมดภายในตัวแปร `paramsRequest` ไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

628 //sending
629 RKParams* paramsRequest = [RKParams params];
630
631 //set values
632 [paramsRequest setValue:name.text forParam:@"data[name]"];
633 [paramsRequest setValue:name_th.text forParam:@"data[name_th]"];
634 [paramsRequest setValue:[[[province_dic objectForKey:@"body"]objectAtIndex:
635 select_province ] objectForKey:@"province_id"] forParam:@"data[province_id]"];
636
637 [paramsRequest setValue:[[[travelttype_dic objectForKey:@"body"]objectAtIndex:
638 select_travelttype ] objectForKey:@"type_id"] forParam:@"data[type_id]"];
639
640 [paramsRequest setValue: content.text forParam:@"data[content]"];
641 [paramsRequest setValue:address.text forParam:@"data[address]"];
642 [paramsRequest setValue:call.text forParam:@"data[phone]"];
643 [paramsRequest setValue:iphone.text forParam:@"data[phone_call]"];
644 [paramsRequest setValue:latitude.text forParam:@"data[latitude]"];
645 [paramsRequest setValue:longitude.text forParam:@"data[longitude]"];
646 [paramsRequest setValue:[[[pricerange_dic objectForKey:@"body"]objectAtIndex:
647 select_pricerange ] objectForKey:@"price_id"] forParam:@"data[price_id]"];
648
649 [paramsRequest setValue:time_open.text forParam:@"data[open]"];
650 [paramsRequest setValue:textfield_promotion_name.text forParam:@"data[promotion]"];
651 [paramsRequest setValue:[[[eattype_array objectAtIndex:select_travelttype
652 objectForKey:@"body"]objectAtIndex:select_travleattype ]
653 objectForKey:@"eat_type_id"] forParam:@"data[eat_type_id]"];
654
655
656 [paramsRequest setValue:textfield_Date.text forParam:@"data[promotion_start]"];
657 [paramsRequest setValue:textfield_end_Date.text forParam:@"data[promotion_end]"];
658 [paramsRequest setValue:direction.text forParam:@"data[direction]"];
659 [paramsRequest setValue:rec forParam:@"data[recommnd]"];
660
661 [paramsRequest setValue:website.text forParam:@"data[website]"];
662 [paramsRequest setValue:active forParam:@"data[is_active]"];
663
664 //username
665 [paramsRequest setValue:account forParam:@"user"];
666 [paramsRequest setValue:password_account forParam:@"pass"];
667
668 //attach image
669 NSData *image_data = [NSData dataWithData:[UIImagePNGRepresentation([place_image image])]];
670 if (image_data == nil) {
671
672     NSLog(@"error data is null");
673 }
674 else {
675     [paramsRequest setValue:imagePath forParam:@"data[preview_image]"];
676     RKParamsAttachment* attachment = [paramsRequest setData:image_data forParam:@"file"];
677     attachment.MIMETYPE = @"image/png";
678     attachment.fileName = image_name;
679 }
680
681 //fire request
682 [[RKClient sharedClient] post:@"/Placettravel" params:paramsRequest delegate:self];
683
684
685
686 }

```

รูปที่ ง-115 การ Add ข้อมูล

• การ Edit ข้อมูล

จากรูปที่ ง-116 จะเป็นการ Edit ข้อมูลภายในฐานข้อมูล ซึ่งจะคล้ายๆกับการ Add ข้อมูล แต่จะต่างกันตรงในบรรทัดที่ 1037-1043 จะเป็นการทำการเพิ่มรูปเข้าไปใหม่แทนรูปเก่า และในบรรทัดที่ 1047-1048 จะเป็นการกำหนด resourcePath ซึ่งจะเห็นว่าใน resourcePath จะมีการใส่ method PUT เข้าไปด้วยและมีการส่งพารามิเตอร์ไปด้วยเพื่อเป็นการเจาะจงไปว่าจะทำการ Edit ข้อมูลของ id ที่เท่าไร และในบรรทัดที่ 1050 ก็จะเป็นการ Edit ข้อมูลโดยอาศัยการ Post ไปยัง REST เว็บเซอร์วิส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

997     RKParams* paramsRequest = [RKParams params];
998
999     //set values
1000     [paramsRequest setValue:nameData.text forParam:@"data[name]"];
1001     [paramsRequest setValue:name_th_data.text forParam:@"data[name_th]"];
1002     [paramsRequest setValue:[{[province_dic objectForKey:@"body"]
1003     objectAtIndex:select_province } objectForKey:@"province_id"]
1004     forParam:@"data[province_id]"];
1005
1006     [paramsRequest setValue:[{[traveltype_dic objectForKey:@"body"]
1007     objectAtIndex:select_traveltype } objectForKey:@"type_id"]
1008     forParam:@"data[type_id]"];
1009
1010     [paramsRequest setValue:contentData.text forParam:@"data[content]"];
1011     [paramsRequest setValue:addressData.text forParam:@"data[address]"];
1012     [paramsRequest setValue:phonecallData.text forParam:@"data[phone]"];
1013     [paramsRequest setValue:iphonecallData.text forParam:@"data[phone_call]"];
1014     [paramsRequest setValue:latitudeData.text forParam:@"data[latitude]"];
1015     [paramsRequest setValue:longitudeData.text forParam:@"data[longitude]"];
1016     [paramsRequest setValue:[{[pricerange_dic objectForKey:@"body"]
1017     objectAtIndex:select_pricerange } objectForKey:@"price_id"]
1018     forParam:@"data[price_id]"];
1019
1020     [paramsRequest setValue:OpenData.text forParam:@"data[open]"];
1021     [paramsRequest setValue:phonecallData.text forParam:@"data[promotion]"];
1022     [paramsRequest setValue:[{[eattype_array objectForKey:@"body"]
1023     objectAtIndex:select_eattype } objectForKey:@"eat_type_id"]
1024     forParam:@"data[eat_type_id]"];
1025
1026     [paramsRequest setValue:PromotionStartData.text forParam:@"data[promotion_start]"];
1027     [paramsRequest setValue:PromotionEndData.text forParam:@"data[promotion_end]"];
1028     [paramsRequest setValue:DirectionData.text forParam:@"data[direction]"];
1029     [paramsRequest setValue:rec forParam:@"data[recommend]"];
1030     [paramsRequest setValue:WebsiteData.text forParam:@"data[website]"];
1031     [paramsRequest setValue:active forParam:@"data[is_active]"];
1032     //username
1033     [paramsRequest setValue:account forParam:@"user"];
1034     [paramsRequest setValue:password_account forParam:@"pass"];
1035
1036     //attach image
1037     NSData *image_data = [NSData dataWithData:[UIImagePNGRepresentation([place_image image])]];
1038     if (image_change) {
1039         [paramsRequest setValue:imagePath forParam:@"data[preview_image]"];
1040         [paramsRequest setValue:old_path forParam:@"old_path"];
1041         RKParamsAttachment* attachment = [paramsRequest setData:image_data forParam:@"file"];
1042         attachment.MIMETYPE = @"image/png";
1043         attachment.fileName = image_name;
1044     }
1045
1046     //fire request
1047     NSString *resourcePath = [NSString stringWithFormat: @"/Placetraavel/%@?_method=PUT",
1048     [[_inputDic objectForKey:@"body"] objectAtIndex:0] objectForKey:@"place_id"];
1049
1050     [[RKClient sharedClient] post:resourcePath params:paramsRequest delegate:self];
1051 }

```

รูปที่ ง-116 การ Edit ข้อมูล

• การ Delete ข้อมูล

จากรูปที่ ง-117 จะเป็นการ Delete ข้อมูลภายในฐานข้อมูลซึ่งก็จะทำคล้ายๆกับการ Edit ข้อมูลแต่ต่างกันตรงบรรทัดที่ 1319 ที่จะเปลี่ยนจาก method PUT เป็น method Delete

```

1314     RKParams* paramsRequest = [RKParams params];
1315     //username
1316     [paramsRequest setValue:account forParam:@"user"];
1317     [paramsRequest setValue:password_account forParam:@"pass"];
1318
1319     NSString *resourcePath [NSString stringWithFormat: @"/Placetraavel/%@?_method=DELETE",
1320     [[_inputDic objectForKey:@"body"] objectAtIndex:0] objectForKey:@"place_id"];
1321
1322     [[RKClient sharedClient] post:resourcePath params:paramsRequest delegate:self];

```

รูปที่ ง-117 การ Delete ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ง.2.3 ฟังก์ชันการทำงานของแอปพลิเคชัน

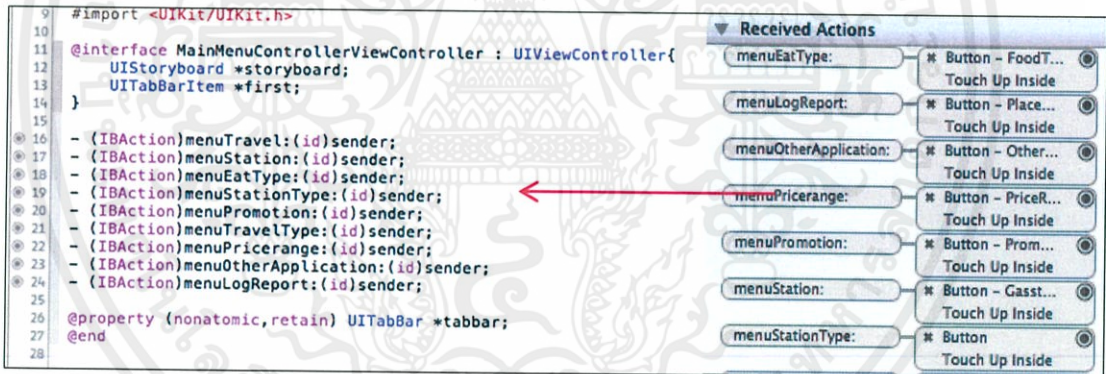
ง.2.3.1 คำสั่งที่ใช้ในการเปลี่ยนหน้า

- การเปลี่ยนหน้าจากการเลือกเมนู

เนื่องจากการสร้างปุ่ม Button ไว้ในหน้า View Controller แล้ว และมีการเชื่อมปุ่มแต่ละปุ่มเข้ากับฟังก์ชันต่างๆดังรูปที่ ง-119



รูปที่ ง-118 การกดปุ่มในหน้าเมนู



รูปที่ ง-119 การเชื่อมปุ่มเข้ากับฟังก์ชันการทำงาน

โดยแต่ละปุ่มจะมีฟังก์ชันการทำงานดังรูป ง-120 นั่นก็คือจะทำการ Push เอาหน้าของเมนูที่เลือกมาวางทับลงบนหน้าเมนูหลัก โดยจากรูปที่ ง-120 ในบรรทัดที่ 95-100 จะเป็นตัวอย่างของ Action เมื่อทำการกดปุ่ม Button ของ PlaceTravel โดยจะทำการ Push เอาหน้า PlaceTravelList มาวางทับหน้า เมนูหลักและในบรรทัดที่ 98 จะเป็นการกำหนดให้มี TabBar ในหน้า PlaceTravelList

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

95 - (IBAction)menuTravel:(id)sender {
96     PlaceTravelListController *listTravel =
97     [storyboard instantiateViewControllerWithIdentifier:@"PlaceTravelList"];
98     [listTravel setTabbar:tabbar];
99     [[self navigationController] pushViewController:listTravel animated:YES];
100 }
101
102 - (IBAction)menuStation:(id)sender {
103     PlaceStationListController *listStation =
104     [storyboard instantiateViewControllerWithIdentifier:@"PlaceStationList"];
105     [listStation setTabbar:tabbar];
106     [[self navigationController]pushViewController:listStation animated:YES];
107 }
108 }
109
110 - (IBAction)menuEatType:(id)sender {
111     PlaceTravelEatTypeListController *listEatType =
112     [storyboard instantiateViewControllerWithIdentifier:@"PlaceTravelEatType"];
113     [listEatType setTabbar:tabbar];
114     [[self navigationController]pushViewController:listEatType animated:YES];
115 }
116
117 - (IBAction)menuStationType:(id)sender {
118     PlaceStationTypeListController *listStationType =
119     [storyboard instantiateViewControllerWithIdentifier:@"PlaceStationTypeList"];
120     [listStationType setTabbar:tabbar];
121     [[self navigationController]pushViewController:listStationType animated:YES];
122 }
123 }
124
125 - (IBAction)menuPromotion:(id)sender {
126     PromotionListController *listPromotion =
127     [storyboard instantiateViewControllerWithIdentifier:@"PromotionList"];
128     [listPromotion setTabbar:tabbar];
129     [[self navigationController]pushViewController:listPromotion animated:YES];
130 }
131
132 - (IBAction)menuTravelType:(id)sender {
133     PlaceTravelTypeListController *listTravelType =
134     [storyboard instantiateViewControllerWithIdentifier:@"PlaceTravelTypeList"];
135     [listTravelType setTabbar:tabbar];
136     [[self navigationController]pushViewController:listTravelType animated:YES];
137 }
138
139 - (IBAction)menuPricerange:(id)sender {
140     PriceRangeListController *listPricerange =
141     [storyboard instantiateViewControllerWithIdentifier:@"PricerangeList"];
142     [listPricerange setTabbar:tabbar];
143     [[self navigationController]pushViewController:listPricerange animated:YES];
144 }
145
146 - (IBAction)menuOtherApplication:(id)sender {
147     OtherApplicationListController *listOtherApp =
148     [storyboard instantiateViewControllerWithIdentifier:@"OtherAppList"];
149     [listOtherApp setTabbar:tabbar];
150     [[self navigationController]pushViewController:listOtherApp animated:YES];
151 }
152
153 - (IBAction)menuLogReport:(id)sender
154 {
155     LogReportListController *listLogReport =
156     [storyboard instantiateViewControllerWithIdentifier:@"LogReportMenu"];
157     [listLogReport setTabbar:tabbar];
158     [[self navigationController]pushViewController:listLogReport animated:YES];
159 }
160 }

```

รูปที่ ง-120 ฟังก์ชันการทำงานของปุ่มในหน้า Menu

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การส่งข้อมูลไปยังคลาสอื่น

จากรูปที่ ง-121 จะเป็นฟังก์ชันที่มีการส่งข้อมูลไปยังปลายทางที่ Segue ซึ่งอยู่ซึ่งจากรูปจะเป็นตัวอย่างของการส่งค่า place_id ไป Segue ที่ชื่อ sendPlaceId ที่เคยกำหนดไว้ใน หัวข้อการสร้าง Segue อย่างเช่นในบรรทัดที่ 276 283 และ 293 จะเป็นการส่งค่า place_id ไปยังคลาส PlaceTravel ViewController ซึ่งเป็นหน้าที่ Segue sendPlaceId ที่อยู่และในคลาส PlaceTravelViewController จะมีฟังก์ชัน setDetailItem คอยรับค่า place_id อยู่ดังรูปที่ ง-122

```

263 - (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender
264 {
265     if ([[segue identifier] isEqualToString:@"sendPlaceId"])
266     {
267
268         // In order to manipulate the destination view controller, another check o
269
270         if(sender == self.searchDisplayController.searchResultsTableView)
271         {
272             NSIndexPath *indexPath = [self.searchDisplayController.
273                 searchResultsTableView indexPathForSelectedRow];
274             NSString *object = [[filteredPlaceTravel objectAtIndex:
275                 [indexPath row]]objectForKey:@"place_id"];
276             [[segue destinationViewController] setDetailItem:object];
277
278         }
279         else if (isAdd ) {
280             NSString *object = [[[place_dic objectForKey:@"body"
281                 lastObject] objectForKey:@"place_id"];
282             [[segue destinationViewController] setDetailItem:object];
283             self.isAdd = NO;
284
285         }
286     }
287
288     else
289     {
290         NSIndexPath *indexPath = [self.tableView indexPathForSelectedRow];
291         NSString *object = [[[place_dic objectForKey:@"body"
292             objectAtIndex:[indexPath row]] objectForKey:@"place_id"];
293         [[segue destinationViewController] setDetailItem:object];
294
295     }
296 }
297
298 }
299 }

```

รูปที่ ง-121 การส่งข้อมูลผ่าน Segue

```

53 - (void)setDetailItem:(id)newDetailItem
54 {
55     if (_detailItem != newDetailItem) {
56         _detailItem = newDetailItem;
57         NSLog(@"view of sendPlaceID = %@",_detailItem);
58     }
59 }
60
61 }

```

รูปที่ ง-122 การรับข้อมูลจาก Segue

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ง.2.3.2 คำสั่งที่ใช้ในการจัดการส่วนติดต่อผู้ใช้

- การแสดงข้อมูลภายใน Cell ของ Table View

หลังจากที่ทำการสร้าง Table View Controller ในหัวข้อการสร้าง Table View Controller แล้วในการที่จะใส่ข้อมูลลงใน Table View นั้นจะทำได้ตามจากการเขียนโค้ดในไฟล์ .m ซึ่งในที่นี้คือไฟล์ PlaceTravelListController โดยจากรูปที่ ง-123 จะเป็นฟังก์ชันที่กำหนดถึงจำนวน Section ภายใน Table View

```

107 - (NSInteger)numberOfSectionsInTableView:(UITableView *)tableView
108 {
109     // Return the number of sections.
110     return 1;
111 }
112

```

รูปที่ ง-123 การกำหนดจำนวน Section

จากรูปที่ ง-124 จะเป็นฟังก์ชันที่กำหนดถึงจำนวน Row ในแต่ละ Section ว่าจะให้มีจำนวนเท่าใดโดยในบรรทัดที่ 121 จะเป็นการกำหนด Row ตามจำนวนของข้อมูลที่ถูกเก็บอยู่ในตัวแปร place_dic หรือ ในบรรทัดที่ 117-118 จะเป็นการกำหนด Row ตามจำนวนข้อมูลที่ทำกร Search จากตัวแปร filteredPlaceTravel

```

114 - (NSInteger)tableView:(UITableView *)tableView numberOfRowsInSection:(NSInteger)section
115 {
116     // Return the number of rows in the section.
117     if (tableView == self.searchDisplayController.searchResultsTableView) {
118         return [filteredPlaceTravel count];
119     } else {
120         return [[place_dic objectForKey:@"body"] count];
121     }
122 }
123
124
125
126 }

```

รูปที่ ง-124 การกำหนด Row ใน Section

จากรูปที่ ง-125 จะเป็นฟังก์ชันที่นำข้อมูลมาแสดงใน Cell ของแต่ละ Row ดังรูปที่ ง-126 โดยในบรรทัดที่ 137-143 จะเป็นการแสดงข้อมูลที่อยู่ในตัวแปร filteredPlaceTravel ซึ่งเป็นข้อมูลที่ได้จากการ Search ซึ่งข้อมูลที่แสดงจะเป็นชื่อสถานที่และแสดง ซึ่งจะเป็นการแสดงจังหวัดที่ตั้งของสถานที่นั้นๆ (ตัวแปร detailTextLabel) และในบรรทัดที่ 147-153 จะแสดงข้อมูลภายใน Cell จากข้อมูลที่เกี่ยวข้องในตัวแปร place_dic ซึ่งเป็นตัวแปรที่เก็บข้อมูลของสถานที่ทั้งหมด และมี detailTextLabel ซึ่งจะเป็นการแสดงข้อมูลจังหวัดที่ตั้งของสถานที่นั้นๆ

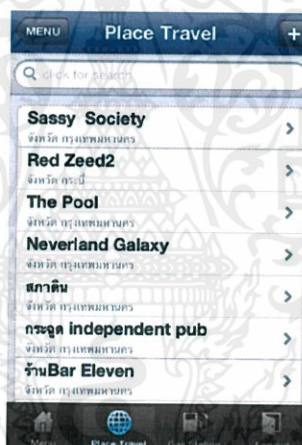
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

128 - (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:
129 (NSIndexPath *)indexPath
130 {
131     static NSString *CellIdentifier = @"Cell";
132     UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:CellIdentifier];
133     if (cell == nil) {
134         cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault
135             reuseIdentifier:CellIdentifier];
136     }
137     if (tableView == self.searchDisplayController.searchResultsTableView)
138     { cell.textLabel.text = [[filteredPlaceTravel objectAtIndex:indexPath.row]
139         objectForKey:@"name"];
140
141         cell.detailTextLabel.text = [NSString stringWithFormat: @"จังหวัด %@",
142             [[filteredPlaceTravel objectAtIndex:indexPath.row] objectForKey:@"province_name"]];
143     }
144     else {
145
146         cell.textLabel.text = [[[place_dic objectAtIndex:indexPath.row] objectForKey:@"name"];
147     }
148
149     cell.detailTextLabel.text = [NSString stringWithFormat: @"จังหวัด %@",
150         [[[place_dic objectAtIndex:indexPath.row] objectForKey:@"province_name"]];
151
152     return cell;
153 }
154 }

```

รูปที่ ง-125 การแสดงข้อมูลใน Cell



รูปที่ ง-126 การแสดงผลของ Cell ใน Table View

- การค้นหาข้อมูล

จากรูปที่ ง-127 จะเป็นการค้นหาข้อมูลซึ่งจากรูปจะเป็นตัวอย่างการค้นหาข้อมูลของคลาส PlaceTravelListController โดรนในบรรทัดที่ 316-319 จะเป็นการเก็บค่าที่พิมพ์ในช่อง Search เพื่อนำไปใช้ในการค้นหาในตัวแปร place_dic แล้วเก็บผลลัพธ์ไว้ในตัวแปร filteredPlaceTravel และในบรรทัดที่ 323-341 จะเป็นการ Reload ข้อมูลเพื่อใช้ในการค้นหาข้อมูลใหม่

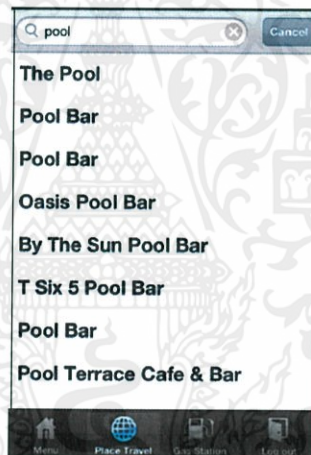
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

310 #pragma mark Content Filtering
311 -(void)filterContentForSearchText:(NSString*)searchText scope:(NSString*)scope {
312     // Update the filtered array based on the search text and scope.
313     // Remove all objects from the filtered search array
314     [self.filteredPlaceTravel removeAllObjects];
315     // Filter the array using NSPredicate
316     NSPredicate *predicate = [NSPredicate predicateWithFormat:
317         @"SELF.name contains[c] %@", searchText];
318     filteredPlaceTravel = [NSMutableArray arrayWithArray:[place_dic
319         objectForKey:@"body"] filteredArrayUsingPredicate:predicate];
320 }
321
322 #pragma mark - UISearchDisplayController Delegate Methods
323 -(BOOL)searchDisplayController:(UISearchDisplayController *)controller
324 shouldReloadTableForSearchString:(NSString *)searchString
325 {
326     // Tells the table data source to reload when text changes
327     [self filterContentForSearchText:searchString scope:
328         [[self.searchDisplayController.searchBar scopeButtonTitles]
329         objectAtIndex:self.searchDisplayController.searchBar.selectedScopeButtonIndex]];
330     // Return YES to cause the search result table view to be reloaded.
331     return YES;
332 }
333
334 -(BOOL)searchDisplayController:(UISearchDisplayController *)
335 controller shouldReloadTableForSearchScope:(NSInteger)searchOption
336 {
337     // Tells the table data source to reload when scope bar selection changes
338     [self filterContentForSearchText:self.searchDisplayController.searchBar.text scope:
339         [[self.searchDisplayController.searchBar scopeButtonTitles]
340         objectAtIndex:searchOption]];
341     // Return YES to cause the search result table view to be reloaded.
342     return YES;
343 }

```

รูปที่ ง-127 ฟังก์ชันในการค้นหาข้อมูล



รูปที่ ง-128 ตัวอย่างการค้นหาข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ง.2.3.3 คำสั่งที่จัดการฟอร์มในการกรอกข้อมูล

- การกรอกข้อมูลในการ Add และ Edit

เมื่อ User ทำการทำการกดปุ่ม Save ข้อมูลแล้วก็จะมีการกรอกข้อมูลก่อนทำการ add หรือ edit ว่าใส่ข้อมูลครบถ้วนไหม ซึ่งถ้าครบข้อมูลครบถ้วนแล้วก็จะอนุญาตให้ทำการ add หรือ edit ข้อมูลได้

```

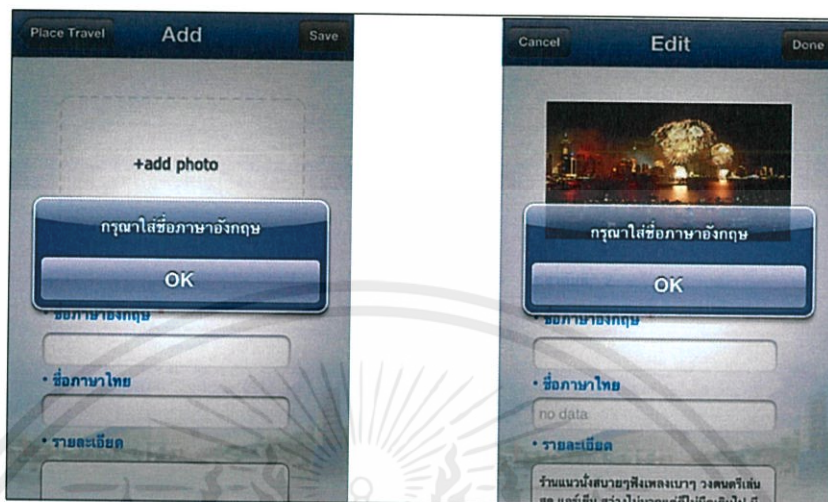
827 -(void)actionSheet:(UIActionSheet *)actionSheet clickedButtonAtIndex:(NSInteger)buttonIndex {
828
829     if (buttonIndex == 0) {
830
831         if ([name.text isEqualToString:@""])
832         { UIAlertView *message = [[UIAlertView alloc] initWithTitle:@"กรุณาใส่ชื่อภาษาอังกฤษ"
833                               message:nil
834                               delegate:nil
835                               cancelButtonTitle:nil
836                               otherButtonTitles:nil];
837
838             message.delegate = self;
839             [message addButtonWithTitle:@"OK"];
840             [message show];
841
842         }
843         else if ([province.text isEqualToString:@""])
844         { UIAlertView *message = [[UIAlertView alloc] initWithTitle:@"กรุณาเลือกจังหวัด"
845                               message:nil
846                               delegate:nil
847                               cancelButtonTitle:nil
848                               otherButtonTitles:nil];
849
850             message.delegate = self;
851             [message addButtonWithTitle:@"OK"];
852             [message show];
853
854         }
855         else if ([iphone.text isEqualToString:@""])
856         { UIAlertView *message = [[UIAlertView alloc] initWithTitle:
857                               @"กรุณาใส่หมายเลขโทรศัพท์สำหรับ iPhone"
858                               message:nil
859                               delegate:nil
860                               cancelButtonTitle:nil
861                               otherButtonTitles:nil];
862
863             message.delegate = self;
864             [message addButtonWithTitle:@"OK"];
865             [message show];
866
867         }
868         else if ([pricerange.text isEqualToString:@""])
869         { UIAlertView *message = [[UIAlertView alloc] initWithTitle:@"กรุณาเลือกราคา"
870                               message:nil
871                               delegate:nil
872                               cancelButtonTitle:nil
873                               otherButtonTitles:nil];
874
875             message.delegate = self;
876             [message addButtonWithTitle:@"OK"];
877             [message show];
878
879         }
880         else if ([placetraveltype.text isEqualToString:@""])
881         { UIAlertView *message = [[UIAlertView alloc] initWithTitle:
882                               @"กรุณาเลือกประเภทร้านอาหาร"
883                               message:nil
884                               delegate:nil
885                               cancelButtonTitle:nil
886                               otherButtonTitles:nil];
887
888             message.delegate = self;
889             [message addButtonWithTitle:@"OK"];
890             [message show];
891
892         }
893         else
894         { [self send];
895         }
896     }
897 }

```

รูปที่ ง-129 การกรอกข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งนี้จากรูปที่ ง-129 จะมีการตรวจสอบข้อมูลว่ามีกรอกรหัสข้อความลงใน Text Field ที่กำหนด
ไว้หรือไม่อย่างเช่นในบรรทัดที่ 831-840 จะเป็นการตรวจในช่อง Text Field ของ name ซึ่งถ้าไม่มี

ข้อมูลจะแสดงข้อความมาแจ้งเตือนว่า กรุณาใส่ชื่อภาษาอังกฤษ ดังรูปที่ ง-130 และมีปุ่มให้ OK ให้กดเพื่อกลับไปทำการพิมพ์ข้อความลงในช่อง Text Field ของ name



รูปที่ ง-130 ตัวอย่างการกรองข้อมูล

- การเลือกรูปจากในเครื่อง iPhone

จากรูปที่ ง-131 จะเป็นการเลือกรูปจากในเครื่อง iPhone มายังแอปพลิเคชันเพื่อที่จะทำการเพิ่มรูปที่ทำการเลือกมานั้นไปยัง Path ที่เก็บรูปในฝั่งเซิร์ฟเวอร์ และในบรรทัดที่ 958 จะมีการเก็บชื่อรูปไว้ในตัวแปร image_name ด้วยเพื่อใช้ในการเพิ่ม Path ที่เก็บรูปลงในฐานข้อมูล

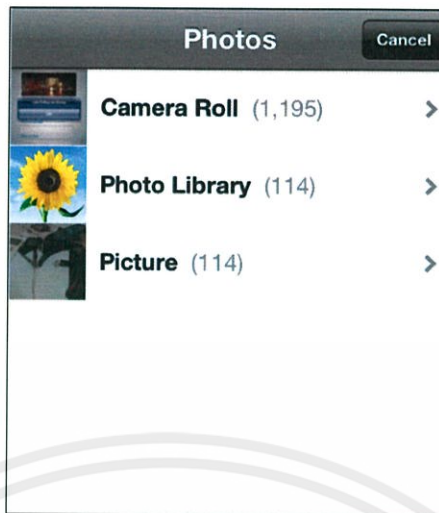
```

939 - (IBAction)selectPicture:(id)sender {
940
941     //image view & imagePicker
942     UIImagePickerController *imgpicker = [[UIImagePickerController alloc] init];
943     imgpicker.delegate = self;
944     imgpicker.allowsEditing = YES;
945     imgpicker.sourceType = UIImagePickerControllerSourceTypePhotoLibrary;
946     [self presentViewController:imgpicker animated:YES];
947 }
948
949 -(void)imagePickerController:(UIImagePickerController *)picker
950 didFinishPickingMediaWithInfo:(NSDictionary *)info
951 {
952     UIImage* temp = [info objectForKey:UIImagePickerControllerEditedImage];
953     place_image.image = temp;
954
955     NSURL *imageurl = [info valueForKey:UIImagePickerControllerReferenceURL];
956     ALAssetsLibrary *library = [[ALAssetsLibrary alloc] init];
957     [library assetForURL:imageurl resultBlock:^(ALAsset *asset){
958         image_name = [NSString stringWithFormat:@"%d", asset.defaultRepresentation.filename];
959     }failureBlock:nil];
960
961     [picker dismissViewControllerAnimated:YES completion:nil];
962 }
963
964

```

รูปที่ ง-131 การนำเข้ารูปจากเครื่อง iPhone

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง.132 ตัวอย่างการนำเข้รูปจากเครื่อง iPhone

- การใส่ข้อมูลรูปภาพลงใน Image View

จากรูปที่ ง-133 จะเป็นการแสดงรูปภาพ โดยในจะเป็นการนำรูปมาจากจาก URL ที่โดย URL ได้มาจาก ตัวแปร place_dic ที่เก็บข้อมูลจากฐานข้อมูลซึ่งในการแสดงรูปนั้นจะมีการตรวจเช็ค ด้วยว่า URL ที่เอามานั้นถ้าเป็นคำว่า no_data ก็ จะสร้างรูป NO_DATA.png ขึ้นมาดังในบรรทัดที่ 136-140

```

132 // check_preview_image
133 NSString *text = [[[place_dic objectForKey:@"body"] objectAtIndex:0]
134                   objectForKey:@"preview_image"];
135
136 if([[[[place_dic objectForKey:@"body"] objectAtIndex:0] objectForKey:@"preview_image"]
137     isEqualToString:@"no_data"])
138 {
139     UIImage *image = [UIImage imageNamed: @"NO_DATA.png"];
140     self.travel_place_image2.image = image;
141 }
142 else
143 {
144     if ([text rangeOfString:@"http"].location == NSNotFound) {
145
146         self.travel_place_image2.image = [UIImage initWithData:
147             [NSData dataWithContentsOfURL:[NSURL URLWithString:
148                 [NSString stringWithFormat:@"%@", ip_image,
149                 [[[place_dic objectForKey:@"body"] objectAtIndex:0]
150                 objectForKey:@"preview_image"]]]]];
151
152     }
153     else {
154         text = [text stringByReplacingOccurrencesOfString:@"https"withString:@"http"];
155
156         self.travel_place_image2.image = [UIImage initWithData:
157             [NSData dataWithContentsOfURL:[NSURL URLWithString:
158                 [NSString stringWithFormat:@"%@",text]]]];
159
160     }
161 }

```

รูปที่ ง-133 การสร้างรูปลงใน Image View

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น การนำเอกสารนี้ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง-134 ตัวอย่างการสร้างรูปลงใน Image View

- การเลื่อนหน้าจอเมื่อกดพิมพ์ข้อความ

เมื่อทำการกดที่ Text Field เพื่อที่จะทำการพิมพ์ข้อความนั้นซึ่งหากไม่มีการเลื่อนหน้าจอขึ้นจะทำให้คีย์บอร์ดที่ใช้ในการพิมพ์ข้อความนั้นบัง Text Field ทำให้ไม่สามารถมองเห็นข้อความที่ทำการพิมพ์ไปแล้วว่าพิมพ์อะไรไปบ้างซึ่งการเลื่อนหน้าจอขึ้นสามารถทำได้โดยเริ่มจากรูปที่ ง-135 ในบรรทัดที่ 17 จะเป็นการประกาศค่าคงที่ในการคำนวณค่าคีย์บอร์ด

```

9  #import "LoginViewController.h"
10
11  @interface LoginViewController ()
12
13  @end
14
15  @implementation LoginViewController
16
17  static const CGFloat KEYBOARD_ANIMATION_DURATION = 0.3;
18
19  @synthesize username;
20  @synthesize password;
21  @synthesize msg;

```

รูปที่ ง-135 การประกาศค่าคงที่

หลังจากทำการประกาศค่าคงที่แล้ว ก็ต้องทำการสร้างฟังก์ชันดังรูปที่ ง-136 เพื่อใช้ในการเลื่อนหน้าจอขึ้นเพื่อไม่ให้คีย์บอร์ดบัง Text Field

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

145 -(void)moveUp:(id)input
146 {   animatedDistance = 0;
147     int bounds;
148     UITextField *textField;
149
150     bounds = 230;
151     textField = (UITextField*)input;
152     CGRect textFieldRect = [self.view.window convertRect:textField.bounds fromView:textField];
153     if(textFieldRect.origin.y > bounds)
154     {   animatedDistance = textFieldRect.origin.y-bounds;
155         CGRect viewFrame = self.view.frame;
156         viewFrame.origin.y -= animatedDistance;
157
158         [UIView beginAnimations:nil context:NULL];
159         [UIView setAnimationBeginsFromCurrentState:YES];
160         [UIView setAnimationDuration:KEYBOARD_ANIMATION_DURATION];
161         [self view] setFrame:viewFrame;
162         [UIView commitAnimations];
163     }
164
165 }
166
167 }

```

รูปที่ ง-136 ฟังก์ชันในการเลื่อนหน้าจอขึ้น

จากรูปที่ ง-137 จะเป็นฟังก์ชันที่ทำการเลื่อนจอลงมา อย่างเช่นในกรณีเมื่อทำการกดปุ่ม Done ที่คีย์บอร์ดเป็นการบอกว่าพิมพ์ข้อความลง Text Field เสร็จแล้วก็จะทำการเลื่อนหน้าจอให้กลับลงมาเป็นเหมือนเดิม

```

-(void)moveTextFieldDown
{   NSLog(@"move down");
    CGRect viewFrame = self.view.frame;
    viewFrame.origin.y += animatedDistance;

    [UIView beginAnimations:nil context:NULL];
    [UIView setAnimationBeginsFromCurrentState:YES];
    [UIView setAnimationDuration:KEYBOARD_ANIMATION_DURATION];

    [self.view setFrame:viewFrame];

    [UIView commitAnimations];
}

```

รูปที่ ง-137 ฟังก์ชันในการเลื่อนหน้าจอลง

การเรียกใช้ฟังก์ชันเลื่อนหน้าจอขึ้น และ เลื่อนหน้าจอลงนั้น จะใช้ดังรูปที่ ง-138 โดยในบรรทัดที่ 136 จะเป็นการเลื่อนหน้าจอขึ้นเมื่อทำการกดไปที่ Text Field ส่วนในบรรทัดที่ 142 จะเป็นการเลื่อนหน้าจอลงเมื่อทำการพิมพ์ข้อความให้ Text Field เสร็จแล้ว

```

134 -(void)textFieldDidBeginEditing:(UITextField*)textField
135 {
136     [self moveUp:textField];
137     [textField setReturnKeyType:UIReturnKeyDone];
138 }
139
140 -(void)textFieldDidEndEditing:(UITextField *)textField
141 {
142     [self moveTextFieldDown];
143 }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ ง-138 การเรียกใช้งานฟังก์ชันการเลื่อนหน้าจอ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

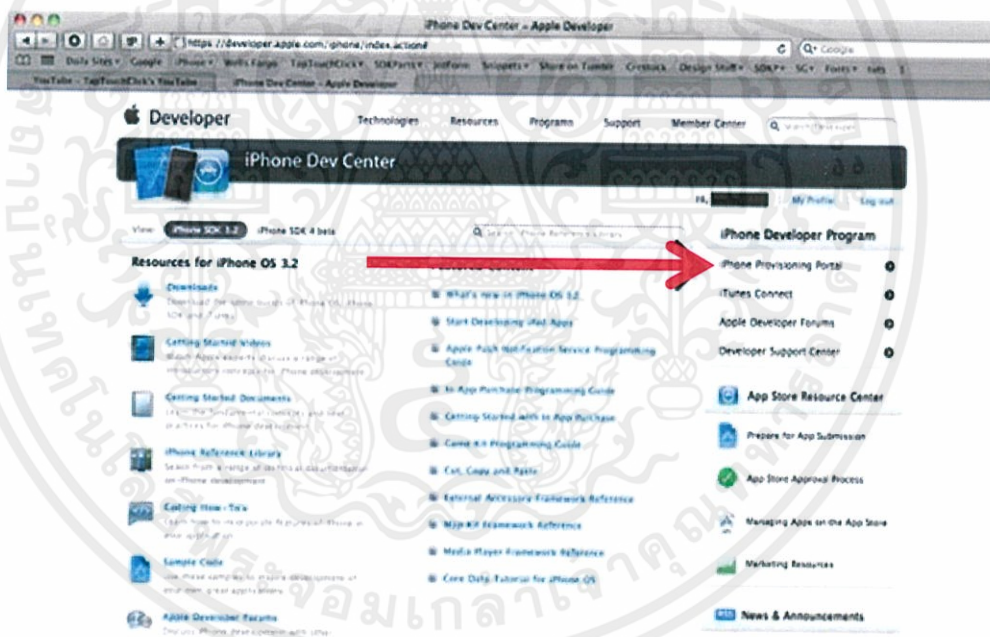
ภาคผนวก จ.

การติดตั้งแอปพลิเคชันลงในเครื่อง iDevice

ในการจะพัฒนาแอปพลิเคชันบน iDevice สามารถทำการทดลองใช้งานแอปพลิเคชันบน Simulator ที่มาพร้อมกับโปรแกรม Xcode ได้ แต่ก็สามารถทำการทดลองใช้งานบนเครื่องได้ โดยในการติดตั้งจะใช้ข้อมูล 3 ส่วนคือ ข้อมูลเกี่ยวกับแอปพลิเคชัน (AppID) ข้อมูลเกี่ยวกับผู้พัฒนา (Developer Certificate) และ ข้อมูลเกี่ยวกับเครื่องที่จะทำการติดตั้ง (unique device identifier หรือ UDID) โดยการติดตั้งแอปพลิเคชันลงบนอุปกรณ์ทั้ง iPhone iPad และ iPod สามารถทำได้โดยมี 5 ขั้นตอนหลักๆ คือ การสร้าง Developer Certificate การติดตั้ง Developer Certificate การลงทะเบียนตัวเครื่องที่จะติดตั้ง การลงทะเบียนแอปพลิเคชัน และ การทดลองใช้แอปพลิเคชันบนตัวเครื่อง

• การสร้าง Developer Certificate

1. ทำการ log in เข้าไปที่ Provisioning Portal (<http://developer.apple.com/iphone>)



รูปที่ จ-1 การสร้าง Developer Certificate ขั้นตอนที่ 1

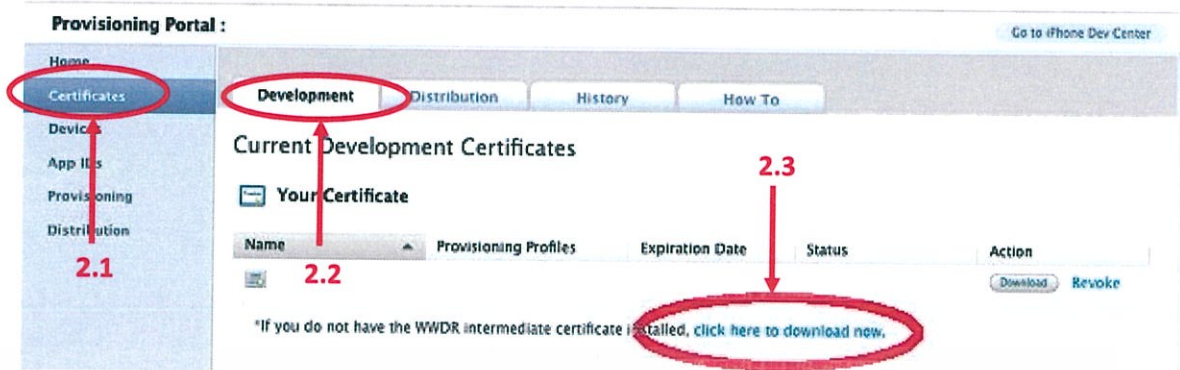
2. ทำการติดตั้ง WWDR Intermediate Certificate

2.1 เลือก Certificate ที่แถบด้านซ้ายมือ

2.2 เลือกแถบ Development

2.3 คลิกดาวน์โหลด “WWDR Intermediate Certificate.”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ลงเนื้อหาใดๆของเอกสารทุกครั้งที่มีการนำไปใช้

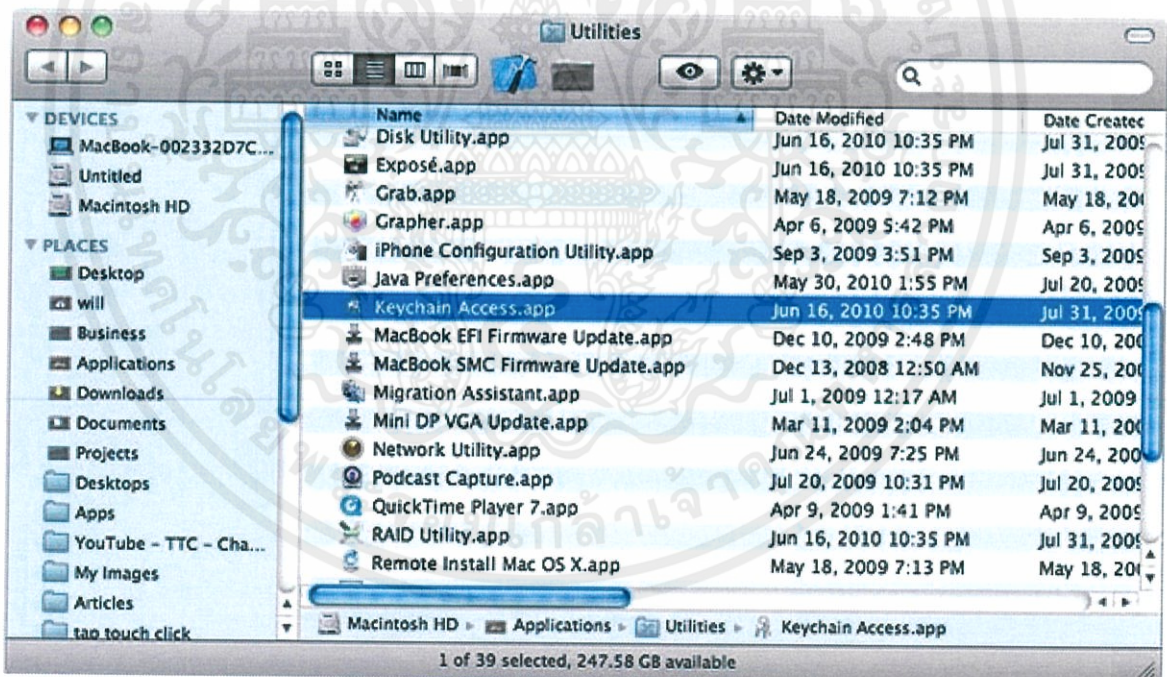


รูปที่ จ-2 การสร้าง Developer Certificate ขั้นตอนที่ 2

2.4 เมื่อทำการดาวน์โหลดเรียบร้อยแล้วให้ดับเบิลคลิกที่ไฟล์ เพื่อเอาไฟล์เข้าไปไว้ใน โปรแกรม Keychain Access

3. ทำการสร้าง Developer Certificate (โดยในการสร้าง Developer Certificate จะต้องใช้ “Certificate Signing Request” ซึ่งสร้างได้จากโปรแกรม Keychain Access)

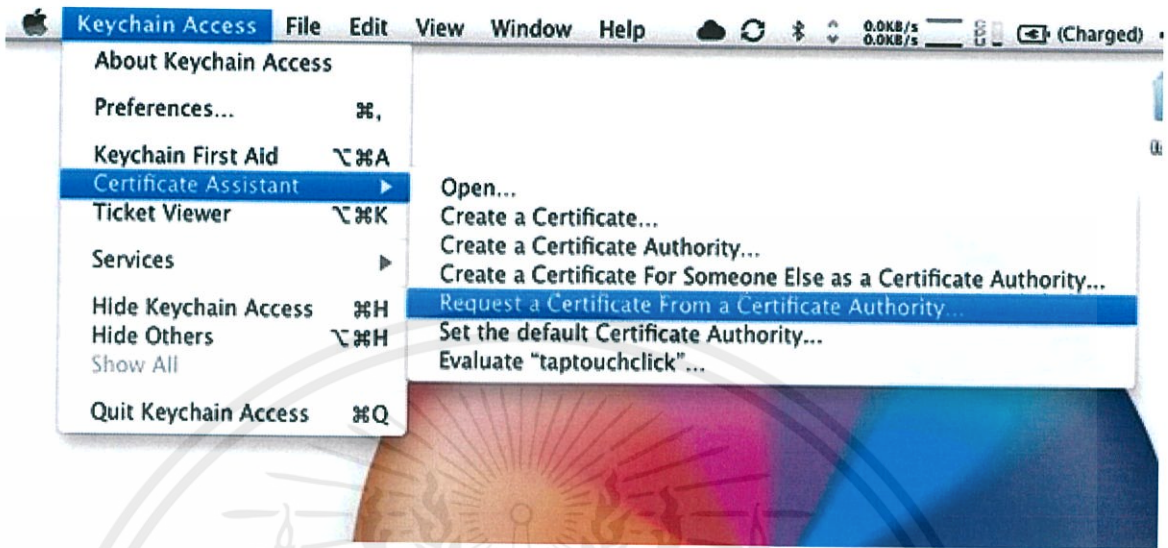
3.1 เปิดโปรแกรม Keychain Access โดยคลิกที่ Application->Utilities->Keychain Access



รูปที่ จ-3 การสร้าง Developer Certificate ขั้นตอนที่ 3.1

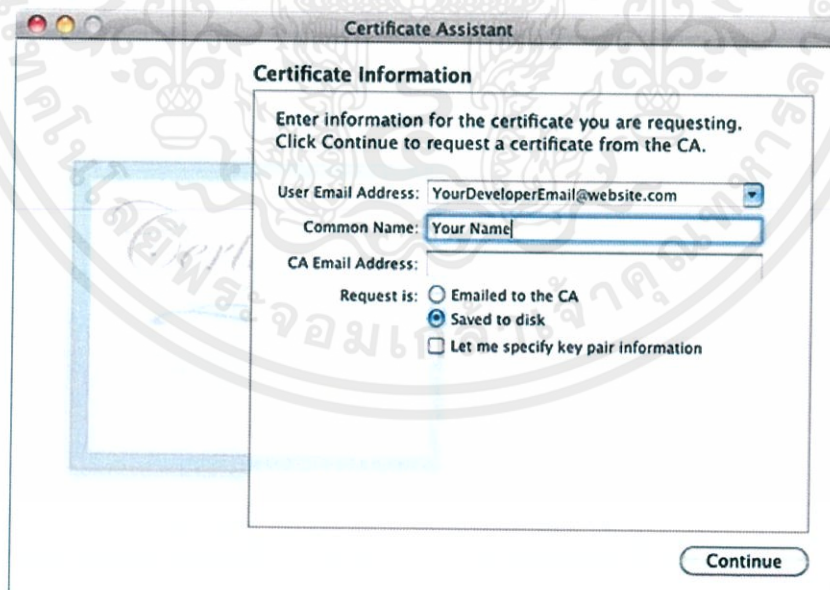
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 คลิกเลือกเมนู Keychain Access -> Certificate Assistant -> Request Certificate from A Certificate Authority



รูปที่ จ-4 การสร้าง Developer Certificate ขั้นตอนที่ 3.2

3.3 หลังจากเลือก Request Certificate from A Certificate Authority จะพบหน้าต่างแสดงขึ้นมา ให้ทำการกรอกข้อมูลโดยเว้นช่อง CA email Address ไว้ ในส่วนของ Request is ให้เลือก Save to disk แล้วกด Continue เพื่อบันทึกไฟล์



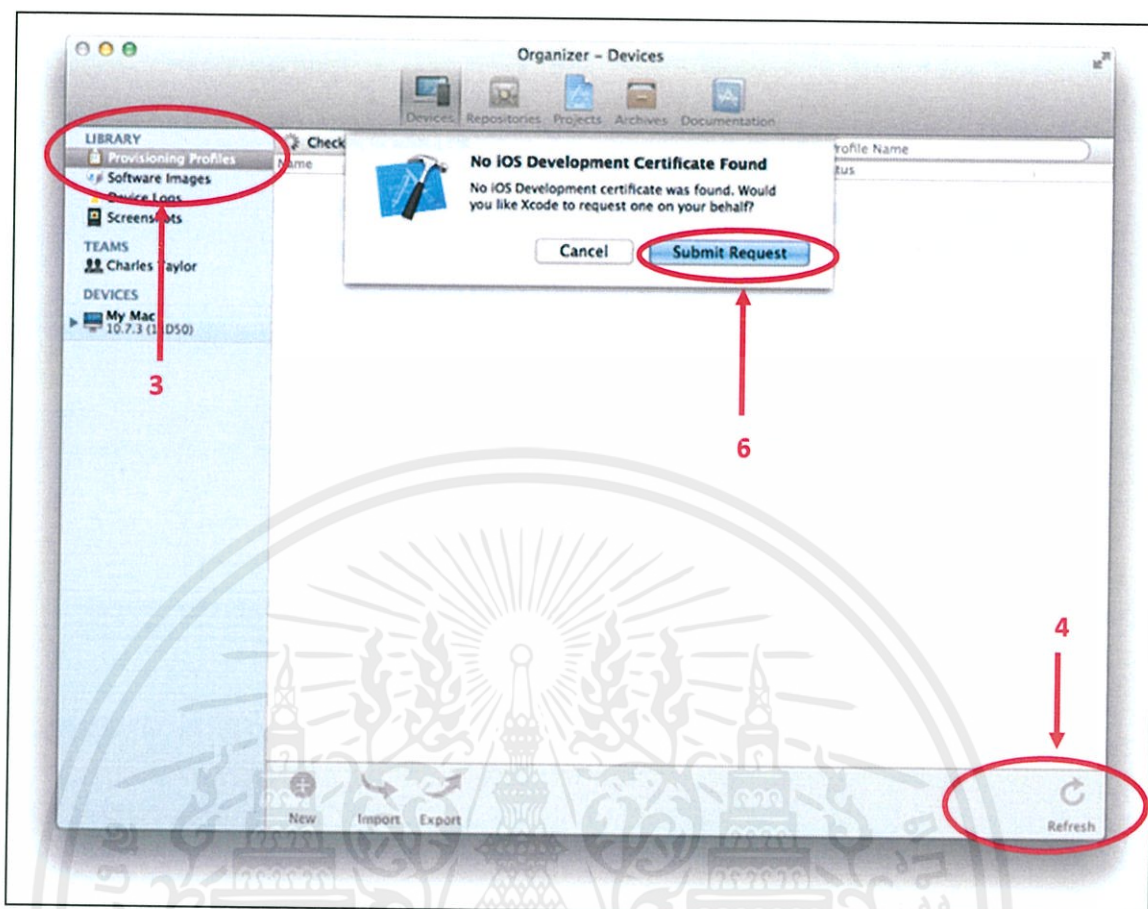
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ จ-5 การสร้าง Developer Certificate ขั้นตอนที่ 3.3
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 กลับไปที่ Provisioning Portal แล้วเลือก New Certificate จากนั้นทำการอัปโหลดไฟล์ Certificate Signing Request ที่สร้างไว้ เมื่อทำการสร้างเสร็จสถานะของ Certificate จะขึ้นว่า Pending และเมื่อสถานะเปลี่ยนเป็น Available หมายความว่า การสร้าง Developer Certificate เสร็จแล้ว

• การร้องขอ Developer Certificate

1. เลือกแถบเมนู Window -> Organizer
2. เลือก Devices
3. ในแถบด้านซ้ายให้เลือก Library แล้วเลือก Provisioning Profiles
4. คลิกปุ่ม refresh ที่อยู่ด้านล่างของหน้าต่าง
5. กรอก username และ password ของ developer program account เพื่อทำการ log in
6. หลังจากที่กรอกข้อมูลแล้วจะมีหน้าต่างขึ้นมาถามว่าต้องการให้ Xcode ร้องขอ Developer Certificate หรือไม่ ให้ทำการกดตกลง (อาจมีหน้าต่างแสดงขึ้นมาอีกเพื่อร้องขอ certificate ประเภทอื่นๆ ให้กดตกลง)
7. เมื่อทำขั้นตอนที่ 6 เสร็จเรียบร้อย จะมีหน้าต่างถามว่าต้องการ export developer certificate ออกมาหรือไม่ ให้เลือก export (ในการสร้าง certificate ค่า private key ของเครื่องจะถูกเก็บไว้ใน certificate ส่วนค่า public key จะถูกเก็บไว้ในระบบ provisioning portal ดังนั้นเมื่อผู้พัฒนาเปลี่ยนเครื่อง Mac แล้วต้องการใช้ certificate เดิม จะต้องทำการ export ตัว certificate แล้วไปใส่ไว้ในเครื่องใหม่)
8. ทำการกำหนดชื่อไฟล์และรหัสผ่าน จากนั้นกด save (รหัสผ่านที่กำหนดนี้ จะใช้เพื่อนำ certificate ออกจากเครื่อง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

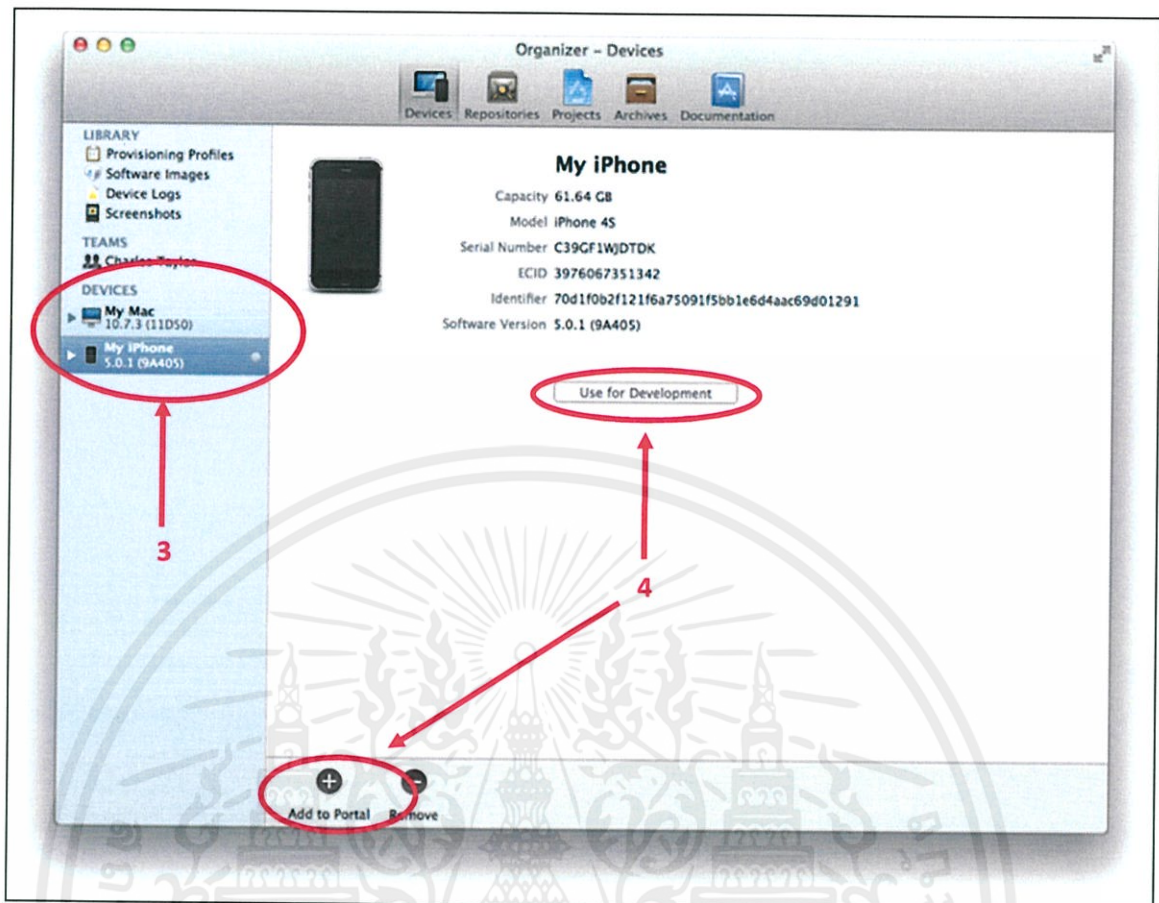


รูปที่ จ-6 การร้องขอ Developer Certificate

• การลงทำเขียนตัวเครื่องที่จะติดตั้ง

1. ทำการต่ออุปกรณ์ iDevice เข้ากับเครื่อง Mac
2. ทำการเปิด Device Organizer (Window->Organizer->Devices)
3. ดูในส่วนของ Devices (อยู่ในแถบด้านซ้าย) จากนั้นเลือก ชื่ออุปกรณ์ที่ต้องการ
4. กดเลือกปุ่ม “ Use for Development ” (ในขั้นตอนนี้จะทำเฉพาะครั้งแรกเท่านั้น หากเคยทำมาก่อนปุ่มนี้จะไม่แสดงให้กด แต่ให้เรากดปุ่ม “ Add to Portal ” ที่อยู่ด้านล่างหน้าต่างแทน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

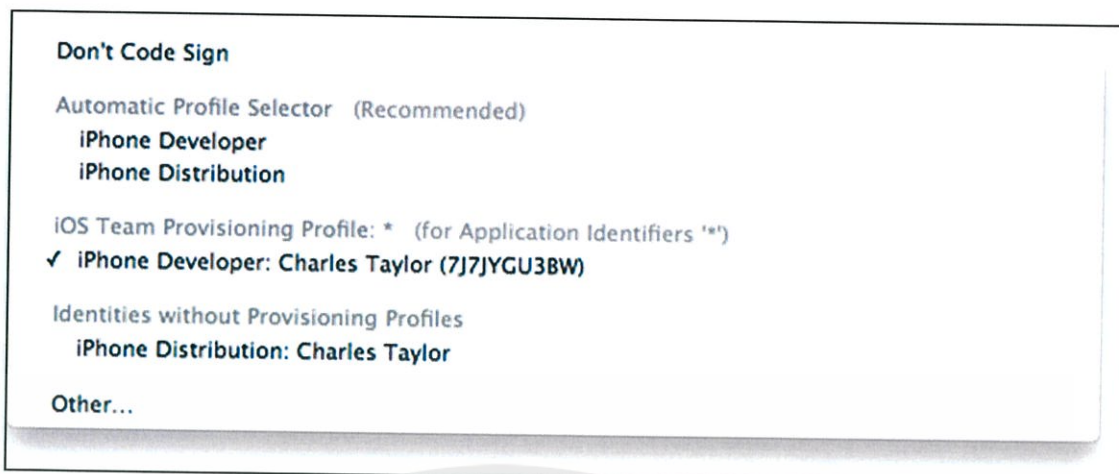


รูปที่ จ-7 การลงทะเบียนเครื่องที่จะทำการติดตั้ง

• การลงทะเบียนแอปพลิเคชัน

1. เลือกแถบเมนู project
2. คลิกเลือก Build Setting
3. คลิก All
4. ทำการค้นหา “ Code Signing ” ในช่องค้นหา
5. ใน pop-up เมนู Code Signing Identity ให้ดูในส่วนของ iOS Team Provisioning Profile แล้วเลือก certificate ที่ขึ้นต้นด้วย “ iPhone Developer ” แล้วตามด้วยชื่อผู้พัฒนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ จ-8 การลงทะเบียนแอปพลิเคชัน

• การทดลองใช้แอปพลิเคชันบนตัวเครื่อง

1. เลือกอุปกรณ์ที่ต้องการติดตั้งแอปพลิเคชัน
2. กดปุ่ม Run



รูปที่ จ-9 การทดลองใช้แอปพลิเคชันบนตัวเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ฉ.

การเริ่มต้นใช้งานแอปพลิเคชัน PTT MIS for Life Station จะต้องไปตั้งค่าหมายเลข IP Address ของ Rest เว็บเซอร์วิสก่อนโดยทำการกดไปที่แอปพลิเคชัน Setting จากนั้นกดไปที่ PTT MIS ดังรูปที่ ฉ-1



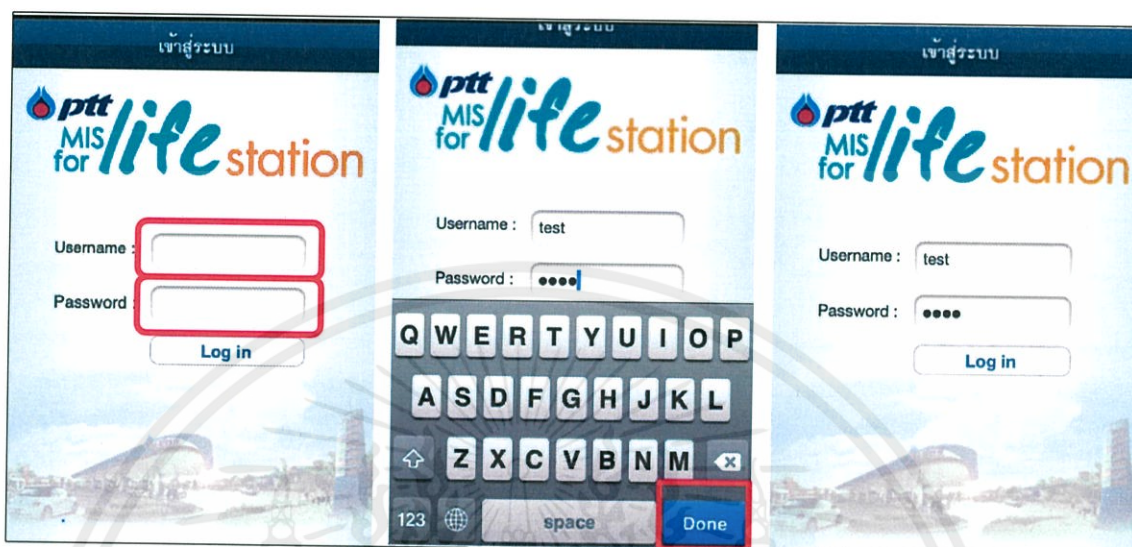
รูปที่ ฉ-1 การตั้งค่าภายใน Setting

การกำหนดค่า IP Address ของ Rest เว็บเซอร์วิส ถ้า Default Address เป็น On จะเป็นการติดต่อไปยังเซิร์ฟเวอร์หลักของทาง ปตท. แต่ถ้าเป็น Off จะเป็นการติดต่อไปยังเซิร์ฟเวอร์ชั่วคราว ซึ่งจะต้องกำหนดค่า IP Address ในช่อง Custom Address และ กำหนดค่า Port ในช่อง Port โดยการเปลี่ยนค่า Default Address ทำได้โดยโดยการกดไปที่ปุ่ม ON OFF ได้เลยและในการเปลี่ยนค่า Custom Address และ Port สามารถกดไปที่ช่องข้างหลังก็จะมีคีย์บอร์ดขึ้นมาให้ใส่หมายเลข IP Address และเมื่อทำการใส่ข้อมูลเสร็จแล้วให้กดปุ่ม Home และทำการกดเข้าใช้งานที่แอปพลิเคชัน PTT MIS ดังรูปที่ ฉ-2



รูปที่ ฉ-2 การกำหนดค่า IP Address

เมื่อเข้าแอปพลิเคชัน PTT MIS for Life Station ให้ทำการกดไปที่ช่องที่อยู่ด้านหลัง Username และ Password เพื่อทำการใส่ข้อมูล โดยเมื่อทำการใส่ข้อมูลที่ต้องการเสร็จแล้วให้ทำการกดปุ่ม Done ในคีย์บอร์ดเพื่อให้คีย์บอร์ดหายไป



รูปที่ ๓-3 การใส่ข้อมูลหน้าล็อกอิน

เมื่อทำการใส่ข้อมูลเสร็จแล้วให้ทำการกดปุ่ม Log in เพื่อทำการเข้าสู่หน้าเมนู แต่ในกรณีถ้าทำการใส่ข้อมูล Username หรือ Password ผิดพลาดจะมีข้อความขึ้นมาแจ้งเตือนแทน



รูปที่ ๓-4 การล็อกอิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในหน้าเมนูสามารถทำการเข้าใช้งานในแต่ละเมนูได้โดยการกดไปที่ไอคอนของเมนูที่ต้องการได้เลย หรืออาจจะกดปุ่มที่อยู่ด้านล่างซึ่งเป็นเมนูที่มีการเข้าใช้งานบ่อยเพื่อให้ง่ายต่อการเปลี่ยนไปหน้าเมนูที่ต้องการ



รูปที่ ๓-5 การเข้าใช้งานในแต่ละเมนู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนปุ่ม Log out นั้นเมื่อกดไปแล้วจะทำการย้อนกลับมาที่หน้าล็อกอินโดยการคลิกเอาที่นั้นจะเกิดขึ้นได้ในกรณีที่เมื่อกดปุ่ม Log out กดปุ่ม Home ของ iPhone หรือ เมื่อหน้าจอ Lock ดับไปเอง



รูปที่ ๑-6 การล็อกเอาท์

• เมนู Place Travel

ในเมนู Place Travel นั้นจะเป็นเมนูที่ทำการจัดการข้อมูลของสถานที่ท่องเที่ยวต่างๆ โดยสามารถเข้าใช้งานโดยการกดไปที่ไอคอน Place Travel หรือ กดทางปุ่มด้านล่างเพื่อเข้าสู่หน้ารายการข้อมูลของเมนู Place Travel



รูปที่ ๑-7 เมนู Place Travel

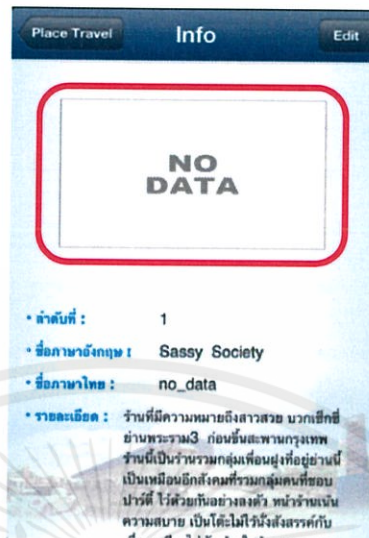
ในหน้ารายการข้อมูลของเมนู Place Travel นั้นจะพบกับรายการข้อมูลต่างๆ โดยต้องทำการกดไปที่รายการข้อมูลที่ต้องการเพื่อเข้าไปดูข้อมูลนั้นๆ อย่างเช่น ในรูปที่ ๘-8 จะเป็นตัวอย่างในการกดเข้าไปดูข้อมูลของ Red Zeed2 โดยที่สามารถคลิกที่แถบค้นหาเพื่อทำการพิมพ์ชื่อรายการข้อมูลที่ต้องการและทำการกดรายการข้อมูลที่ต้องการ



รูปที่ ๘-8 การเลือกข้อมูลในเมนู Place Travel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าภายในสถานที่ท่องเที่ยวที่ไม่มีข้อมูลรูปภาพอยู่จะแสดงรูป NO DATA แทน



รูปที่ ๙-๑ หน้าแสดงข้อมูลโดยไม่มีข้อมูลรูป

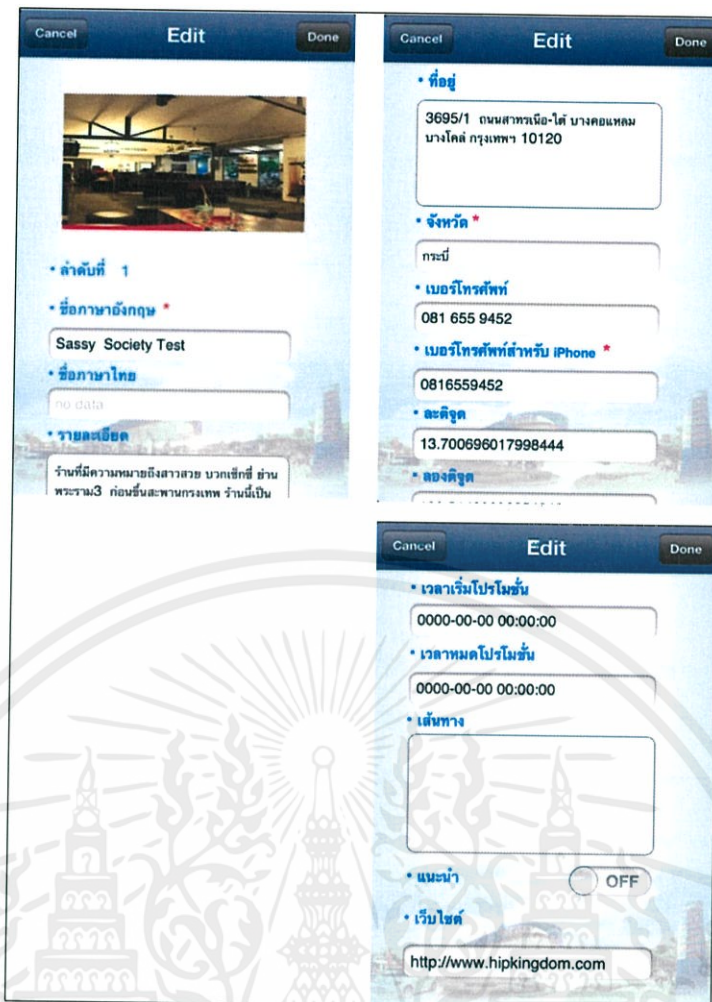
- การแก้ไขข้อมูล

จากหน้าแสดงข้อมูลเมื่อทำการกดปุ่ม Edit จะสามารถทำการแก้ไขข้อมูลได้อย่างเช่นในรูปที่ ๙-10 จะเป็นการเพิ่มรูปเข้าใหม่เข้าไปแทนรูปเดิมซึ่งทำได้โดยการกดที่รูปได้เลย ก็จะสามารถเลือกรูปจากเครื่องมาใส่ได้



รูปที่ ๙-10 การเลือกรูปจากเครื่อง iPhone

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
เมื่อทำการเลือกรูปจากเครื่อง iPhone ได้แล้วจะมีให้ทำการ Crop รูปได้เลยซึ่งสามารถ Crop ได้โดยใช้ 2 นิ้วในการย่อขยายรูป เมื่อทำการ Crop เสร็จแล้วก็ทำการกดปุ่ม Choose เพื่อนำรูปที่เลือกเข้าไปยังแอปพลิเคชันได้เลย



รูปที่ ฉ-13 ข้อมูลของเมนู Place Travel

- | | |
|------------------------------|--|
| - ชื่อภาษาอังกฤษ | เป็นชื่อสถานที่ภาษาอังกฤษ |
| - ชื่อภาษาไทย | เป็นชื่อสถานที่ภาษาไทย |
| - รายละเอียด | เป็นรายละเอียดของสถานที่ |
| - ที่อยู่ | เป็นที่อยู่ของสถานที่ |
| - จังหวัด | เป็นจังหวัดที่ตั้งของสถานที่ |
| - เบอร์โทรศัพท์ | เป็นเบอร์โทรศัพท์ของสถานที่ |
| - เบอร์โทรศัพท์สำหรับ iPhone | เป็นเบอร์โทรศัพท์ที่สามารถกดโทรผ่านแอปพลิเคชัน |

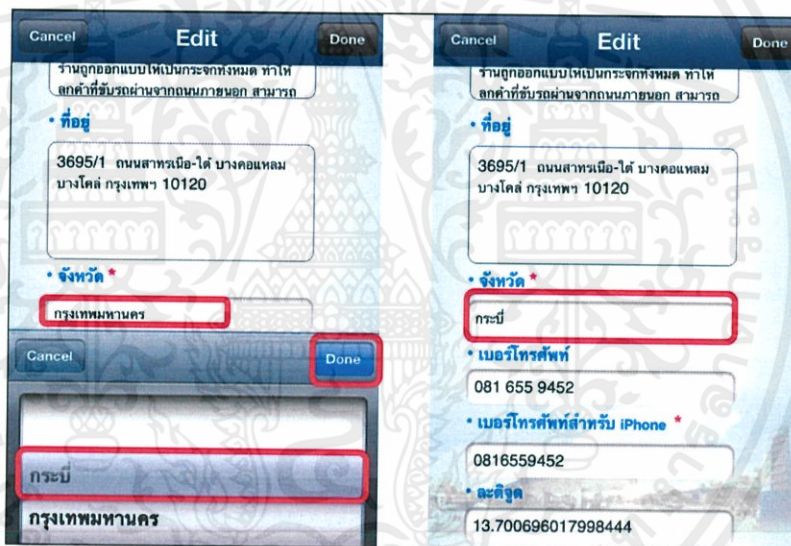
PTT Life Station ได้เลย

- | | |
|-------------------|--|
| - ละติจูด | เป็นค่าละติจูดของสถานที่ |
| - ลองจิจูด | เป็นค่าลองจิจูดของสถานที่ |
| - ราคา | เป็นช่วงราคาค่าบริการโดยประมาณของสถานที่ |
| - เวลาเปิด | เป็นเวลาเปิดของสถานที่ |
| - ประเภทสถานที่ | เป็นประเภทของสถานที่ |
| - ประเภทร้านอาหาร | เป็นประเภทของร้านอาหารที่มีอยู่ในประเภทสถานที่ |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่
ไม่ว่ากรณีใดๆทั้งสิ้น ห้ามมิให้คัดแปลงเนื้อหา
เอกสารทุกครั้งที่มีการนำไปใช้

- สถานะบริการ	เป็นสถานะการให้บริการว่าสถานที่นั้นปิดหรือเปิด
- ชื่อโปรโมชัน	เป็นชื่อโปรโมชันที่ของสถานที่
- เวลาเริ่มโปรโมชัน	เป็นวันที่และเวลาเริ่มโปรโมชัน
- เวลาหมดโปรโมชัน	เป็นวันที่และเวลาหมดโปรโมชัน
- เส้นทาง	เป็นเส้นทางการเดินทางไปยังสถานที่
- แนะนำ	เป็นสถานะการแนะนำสถานที่
- เว็บไซต์	เป็นเว็บไซต์ของสถานที่

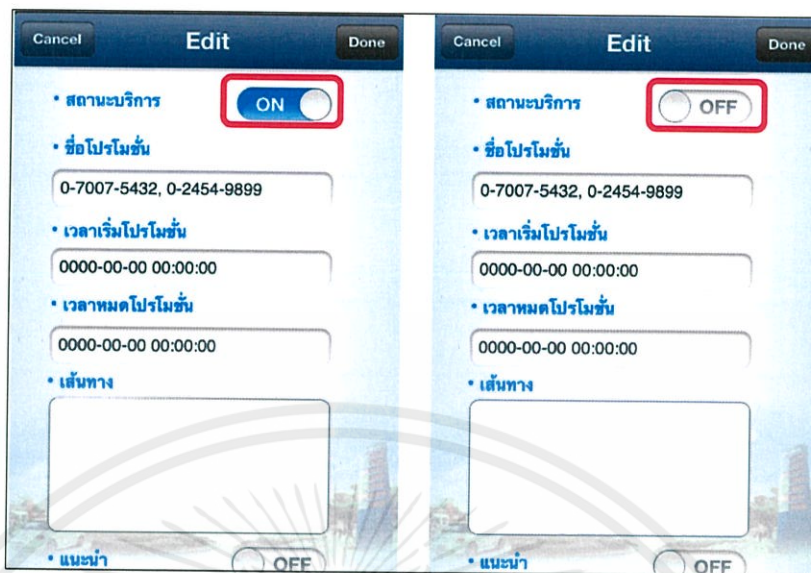
ในส่วนของคุณสมบัติที่เป็นจังหวัดเมื่อทำการกดไปแล้วจะมีหน้าต่างแสดงขึ้นมาให้ทำการเลื่อนขึ้นลงเพื่อเลือกค่าจังหวัดที่ต้องการ โดยเมื่อทำการเลือกข้อมูลได้แล้วจะต้องทำการกดที่ปุ่ม Done เพื่อทำการยืนยันข้อมูลที่เลือกใส่ลงไปในช่วงข้อมูลจังหวัด แต่ถ้ากดปุ่ม Cancel จะเป็นการยกเลิกการเลือกค่าจังหวัด



รูปที่ ๑-14 การแก้ไขข้อมูลจังหวัด

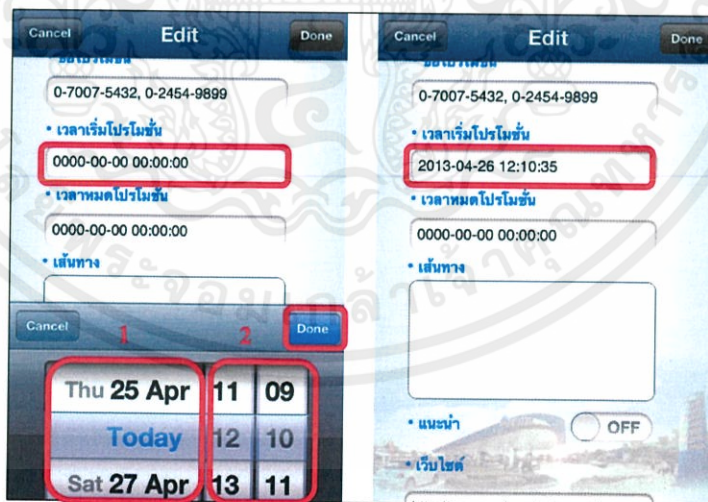
ในส่วนของคุณสมบัติที่เป็นสถานะบริการจะเป็นปุ่ม ON OFF โดยสามารถทำการเปลี่ยนค่าโดยการกดไปที่ปุ่มเลขซึ่งเมื่อทำการกดไปแล้วจะเปลี่ยนค่าจะ ON กลายเป็น OFF หรือ เปลี่ยนจาก OFF กลายเป็น ON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ๑-15 การแก้ไขข้อมูลสถานะบริการ

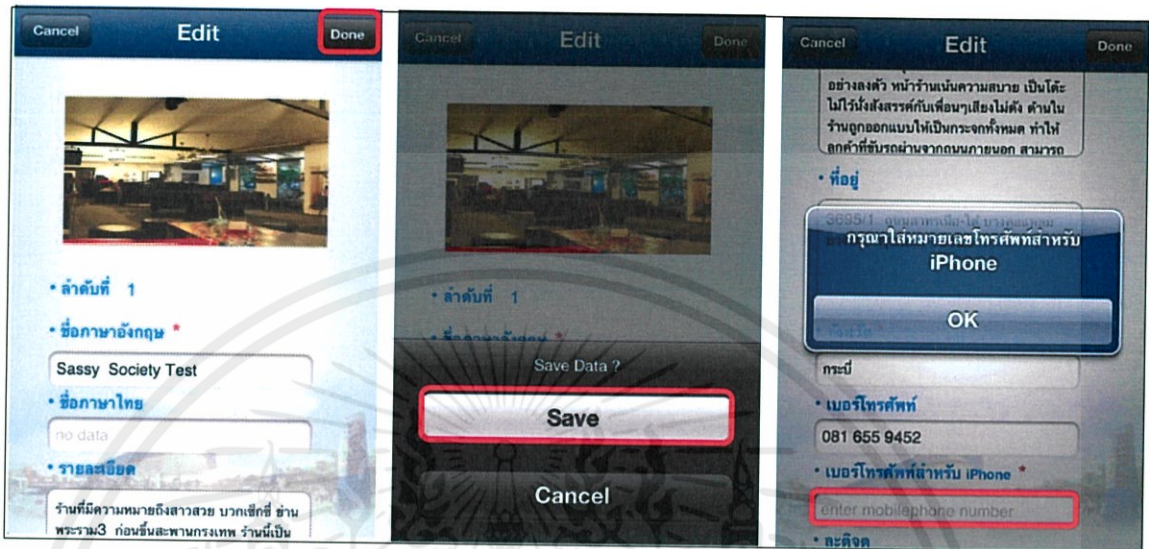
ในส่วนของคุณสมบัติที่เป็นเวลาเริ่มโปรโมชั่นนั้นเมื่อทำการกดไปแล้วจะมีหน้าต่างแสดงขึ้นมาเพื่อให้ทำการเลื่อนขึ้นลงเพื่อทำการเลือกวันที่ในส่วนที่ 1 และในส่วนที่ 2 จะเป็นการเลือกเวลาเริ่มโปรโมชั่นดังรูปที่ ๑-16 โดยเมื่อทำการเลือกข้อมูลได้แล้วจะต้องทำการกดที่ปุ่ม Done เพื่อทำการยืนยันข้อมูลที่เลือกใส่ลงไปในช่วงข้อมูลเวลาเริ่มโปรโมชั่นแต่ถ้ากดปุ่ม Cancel จะเป็นการยกเลิกการเลือกค่าเวลาเริ่มโปรโมชั่น



รูปที่ ๑-16 การแก้ไขข้อมูลสถานะบริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการแก้ไขข้อมูลเสร็จหมดแล้วทำการกดปุ่ม Done เพื่อทำการบันทึกข้อมูล จะมีหน้าต่างขึ้นมาถามถึงความแน่ใจในการทำการบันทึกข้อมูล หลังจากที่ทำการกดปุ่ม Save ถ้าหากลืมใส่ข้อมูลที่มี * กำกับไว้ จะทำการแจ้งเตือนให้กลับไปทำการกรอกข้อมูลในส่วนนั้น



รูปที่ ๑-17 การบันทึกข้อมูล

เมื่อทำการ Save ข้อมูลลงฐานข้อมูล ก็จะย้ายหน้าไปยังหน้าแสดงข้อมูลให้ทันทีเพื่อให้เห็นถึงข้อมูลที่ถูกรแก้ไขเรียบร้อยแล้ว เมื่อทำการกลับมาที่รายการข้อมูลก็จะเห็นได้ว่าข้อมูลได้ถูกรแก้ไขแล้ว



รูปที่ ๑.18 ข้อมูลถูกรแก้ไขแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

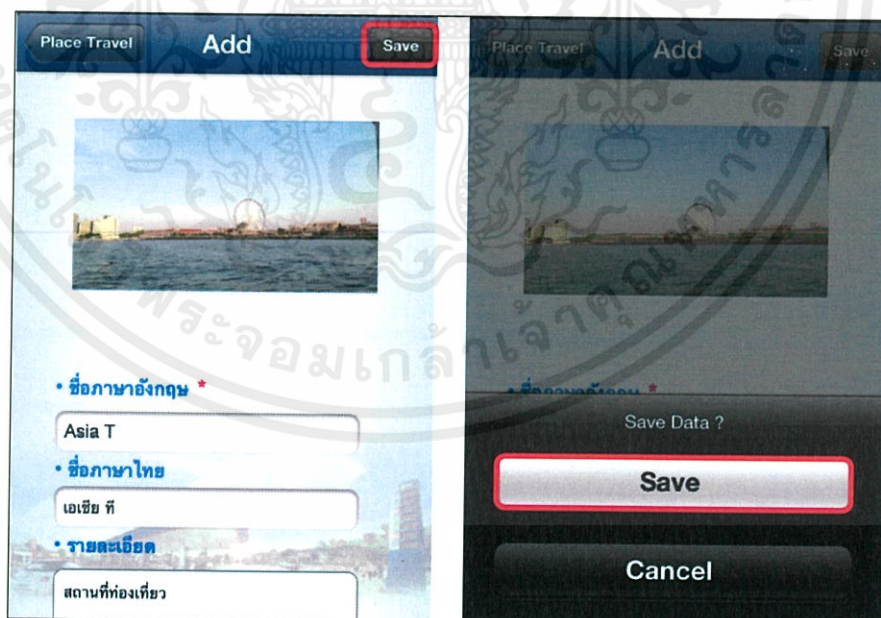
- การเพิ่มข้อมูล

จากหน้ารายการข้อมูลเมื่อทำการกดปุ่ม + จะเป็นการย้ายไปหน้า Add เพื่อทำการเพิ่มข้อมูลใหม่เข้าไปยังฐานข้อมูล



รูปที่ ๑-19 หน้า Add ข้อมูล

ในการใส่ข้อมูลในหน้า Add นั้นจะทำเหมือนกันกับในหน้า Edit ทั้งหมดโดยจะต้องทำการใส่ข้อมูลในส่วนที่มี * กำกับไว้ให้ครบถ้วน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้รูปที่ ๑-20 การเพิ่มข้อมูล อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

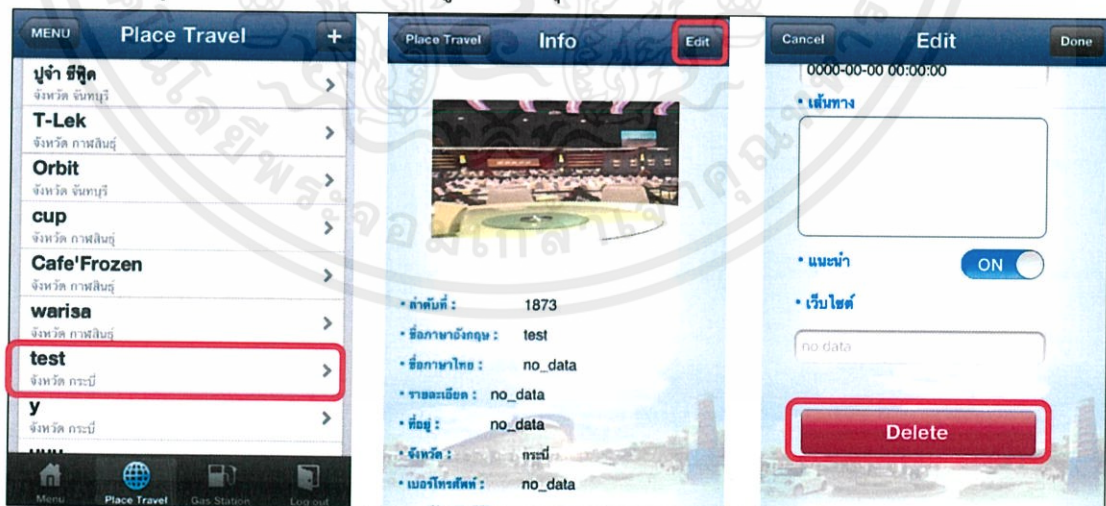
เมื่อทำการ Save ข้อมูลลงฐานข้อมูลก็จะย้ายหน้าไปยังหน้าแสดงข้อมูลให้ทันทีเพื่อให้เห็นถึงข้อมูลที่ถูกเพิ่มเข้าไปเรียบร้อยแล้ว เมื่อทำการกลับมาที่รายการข้อมูลสถานที่ท่องเที่ยวก็จะเห็นได้ว่าข้อมูลได้ถูกเพิ่มมาแล้ว



รูปที่ ฉ-21 หน้าแสดงข้อมูล

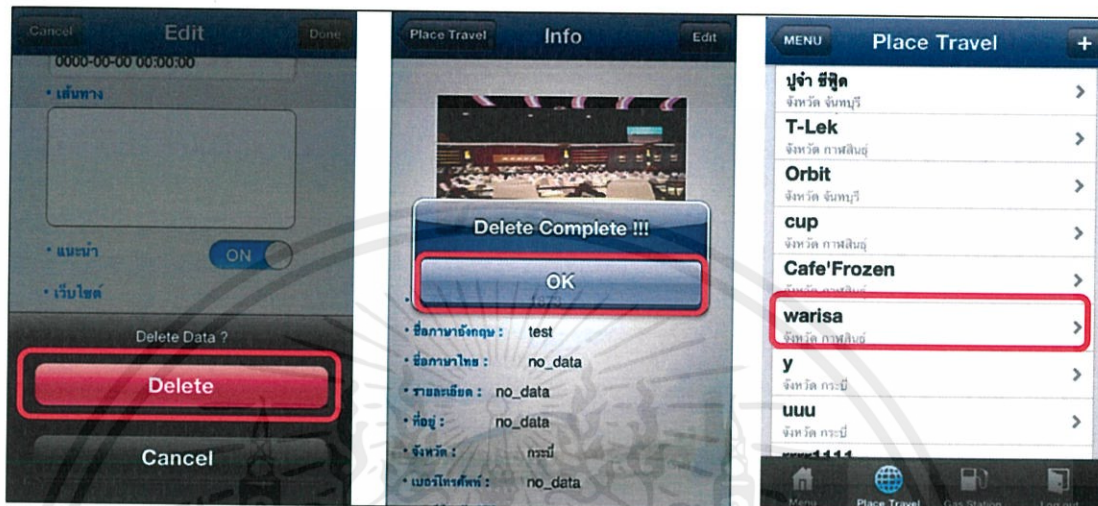
- การลบข้อมูล

ในการลบข้อมูลนั้นจะทำโดยการเลือกข้อมูลที่จะทำการลบ ในหน้ารายการข้อมูลดังรูปที่ ฉ-22 จะเป็นตัวอย่างในการเลือกข้อมูลที่ชื่อว่า Test มาทำการลบ เมื่อเลือกข้อมูลที่จะลบได้แล้วนั้น ปุ่ม Delete จะอยู่ภายในหน้า Edit ซึ่งจะอยู่ด้านล่างสุด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการรูปที่ ฉ-22 การเลือกข้อมูลที่จะลบ อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการกดปุ่ม Delete แล้วจะมีหน้าขึ้นมาถามเพื่อความแน่ใจ เมื่อทำการกดปุ่ม Delete จะย้ายมาที่หน้าแสดงข้อมูลแล้วแสดงหน้าขึ้นมาแจ้งเตือนว่า Delete Complete เพื่อเป็นการบอกว่า ข้อมูลนั้น ได้ถูกลบออกจากฐานข้อมูล หากทำการกดปุ่ม OK ก็จะกลับมาที่หน้ารายการข้อมูลของ Place Travel ซึ่งจะเห็นได้ว่าข้อมูล Test ได้ถูกลบออกไปแล้ว



รูปที่ ฉ.23 การลบข้อมูล

• เมนู Gas Station

ในเมนู Gas Station นั้นจะเป็นเมนูที่ทำการจัดการข้อมูลของสถานีปั้มน้ำมันต่างๆ โดยสามารถเข้าใช้งานโดยการกดไปที่ไอคอน Gas Station หรือ กดทางปุ่มด้านล่างเพื่อเข้าสู่หน้ารายการข้อมูลของเมนู Gas Station



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ รูปที่ ฉ-24 เมนู Gas Station อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการดูข้อมูล, การเพิ่มข้อมูล, การแก้ไขข้อมูล และการลบข้อมูลนั้นจะทำคล้ายๆกันกับในส่วนของเมนู Place Travel โดยในส่วนข้อมูลของเมนู Gas Station มีดังนี้

The image shows two overlapping screenshots of a mobile application interface for a 'Gas Station' menu. The top-left screenshot is the 'Info' screen, and the top-right is the 'Edit' screen. The bottom-left screenshot shows a detailed 'Edit' form with input fields for various fields, and the bottom-right shows the 'Edit' screen with toggle switches for fuel types.

Info Screen Data:

- ลำดับที่ : 1
- การใช้งาน : ใช้งานจริง
- ชื่อสถานี : หจก.คารมบริการ
- ชื่อสถานี NGV : หจก.คารมบริการ
- ประเภทสถานี : standard
- ที่อยู่ : 177 หมู่ที่ 7 ถนนรามอินทรา ตำบลท่าแร้ง อำเภอบางเขน จังหวัดกรุงเทพมหานคร 10220
- จังหวัด : กระบี่
- วันที่เปิด : 2011-08-24
- เบอร์โทรศัพท์ : 025109209
- เบอร์ FAX : 029436636
- Latitude : 13.8561
- Longitude : 100.629
- สถานะการให้บริการ : เผ้าระวัง

Edit Screen (Top Right) Data:

- การใช้งาน : ON
- ชื่อสถานี * : หจก.คารมบริการ
- ชื่อสถานี NGV : หจก.คารมบริการ
- ที่อยู่ : 177 หมู่ที่ 7 ถนนรามอินทรา ตำบลท่าแร้ง อำเภอบางเขน จังหวัดกรุงเทพมหานคร 10220
- ประเภทสถานี * : standard

Edit Screen (Bottom Left) Data:

- ประเภทสถานี * : standard
- วันที่เปิด : 2011-08-24
- จังหวัด : กรุงเทพมหานคร
- เบอร์โทรศัพท์ : 025109209
- เบอร์ FAX : 029436636
- latitude : 13.8561
- longitude : 100.629

Edit Screen (Bottom Right) Data:

- latitude : 13.8561
- longitude : 100.629
- สถานะการให้บริการ * : เปิดตามปกติ
- ดีเซล : ON
- แก๊สโซล 91 : ON
- แก๊สโซล 95 : ON
- แก๊สโซล E20 : ON
- แก๊สโซล E85 : OFF
- เบนซิน 91 : OFF

รูปที่ ๑-25 ข้อมูลภายในเมนู Gas Station

- การใช้งาน เป็นการบอกถึงว่าสถานีปั้มน้ำมันยังเปิดใช้งานไหม

- ชื่อสถานี เป็นการบอกถึงชื่อของสถานีปั้มน้ำมัน

- ชื่อสถานี NGV เป็นการบอกถึงชื่อของสถานีที่มี NGV

- ที่อยู่ เป็นการบอกถึงสถานที่ตั้งของสถานี

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการใช้งานภายในองค์กรเท่านั้น ไม่สามารถนำออกเผยแพร่โดยไม่ได้รับอนุญาตจากทางบริษัทฯ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังห้ามนำไปใช้ในการทำสื่อโฆษณาหรือสิ่งอื่นใดที่มิได้มีการนำไปใช้

- ประเภทสถานี เป็นการบอกถึงว่าสถานีเป็นประเภทไหนโดยจะประกอบด้วย Platinum Standard และNGV
- วันที่เปิด เป็นการบอกถึงวันที่จดทะเบียนสถานี
- จังหวัด เป็นการบอกถึงจังหวัดที่ตั้งของสถานี
- เบอร์โทรศัพท์ เป็นการบอกถึงเบอร์โทรศัพท์ของสถานี
- เบอร์ FAX เป็นการบอกถึงเบอร์ FAX ของสถานี
- latitude เป็นการบอกถึงละติจูดของสถานี
- longitude เป็นการบอกถึงลองจิจูดของสถานี
- สถานะการให้บริการ เป็นการบอกถึงสถานะของสถานีโดยจะประกอบด้วย เปิดตามปกติ, ปิดให้บริการ และเฝ้าระวัง
- ส่วนที่เหลือจะเป็นประเภทของน้ำมันซึ่งสามารถเลือกให้สถานีนั้นๆมีน้ำมันประเภทไหนก็ปรับให้เป็น ON โดยที่จำนวนประเภทน้ำมันนั้นจะปรับเพิ่มลดให้เองตามที่กำหนดภายในเมนู Oil Type

• เมนู Oil Type

ในเมนู Oil Type นั้นจะเป็นเมนูที่ทำการจัดการข้อมูลประเภทน้ำมันต่างๆโดยสามารถเข้าใช้งานโดยการกดไปที่ไอคอน Oil Type



รูปที่ ฉ-26 เมนู Oil Type

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการดูข้อมูล, การเพิ่มข้อมูล, การแก้ไขข้อมูล และการลบข้อมูลนั้นจะทำคล้ายๆกันกับในส่วนของเมนู Place Travel โดยในส่วนข้อมูลของเมนู Oil Type มีดังนี้



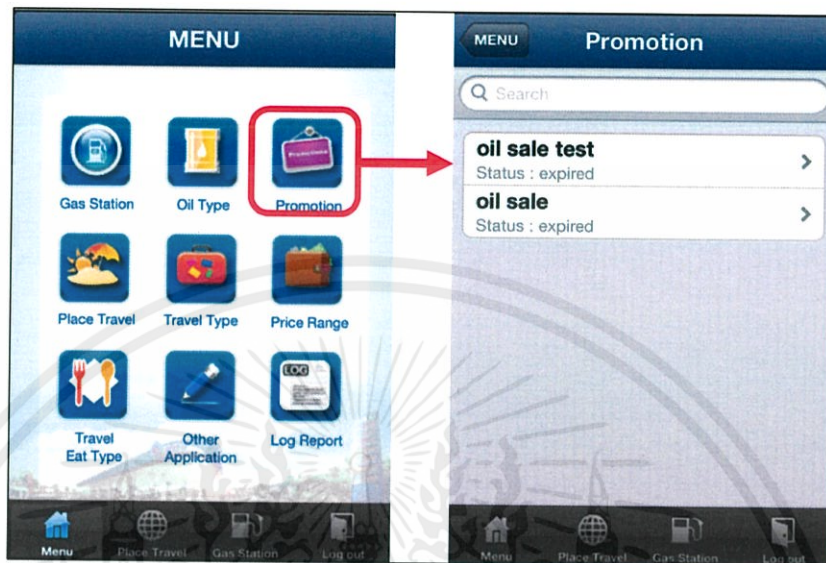
รูปที่ ๓-27 ข้อมูลภายใน Oil Type

- ชื่อประเภท เป็นการบอกถึงชื่อของประเภทน้ำมัน
- ลำดับการแสดงผล เป็นการบอกถึงลำดับที่จะให้แสดงภายในแอปพลิเคชัน PTT Life Station
- การแสดงผล เป็นการบอกถึงการแสดงประเภทน้ำมันนั้นๆให้เห็นหรือไม่
- option เป็นการเลือกประเภทของน้ำมัน โดยประกอบไปด้วย Oil NGV Option และ Map

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• เมนู Promotion

ในเมนู Promotion นั้นจะเป็นเมนูที่ทำการจัดการข้อมูลโปรโมชั่นต่างๆที่มีภายในสถานีปั๊มน้ำมัน โดยสามารถเข้าใช้งานโดยการกดไปที่ไอคอน Promotion



รูปที่ ๑-28 เมนู Promotion

ในการดูข้อมูล, การแก้ไขข้อมูล และการลบข้อมูลนั้นจะทำคล้ายๆกันกับในส่วนของเมนู Place Travel โดยในส่วนข้อมูลของเมนู Oil Type มีดังนี้



รูปที่ ๑-29 ข้อมูลภายในเมนู Promotion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกร ใช้งานเพื่อการศึกษาค้นคว้า ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

- ชื่อ โปรโมชั่น เป็นการบอกถึงชื่อของ โปรโมชั่น
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังเป็นให้คิดแปลงเนื้อหา และต้องอ้างอิงเจ้าของเอกสารทุกครั้งในการนำไปใช้

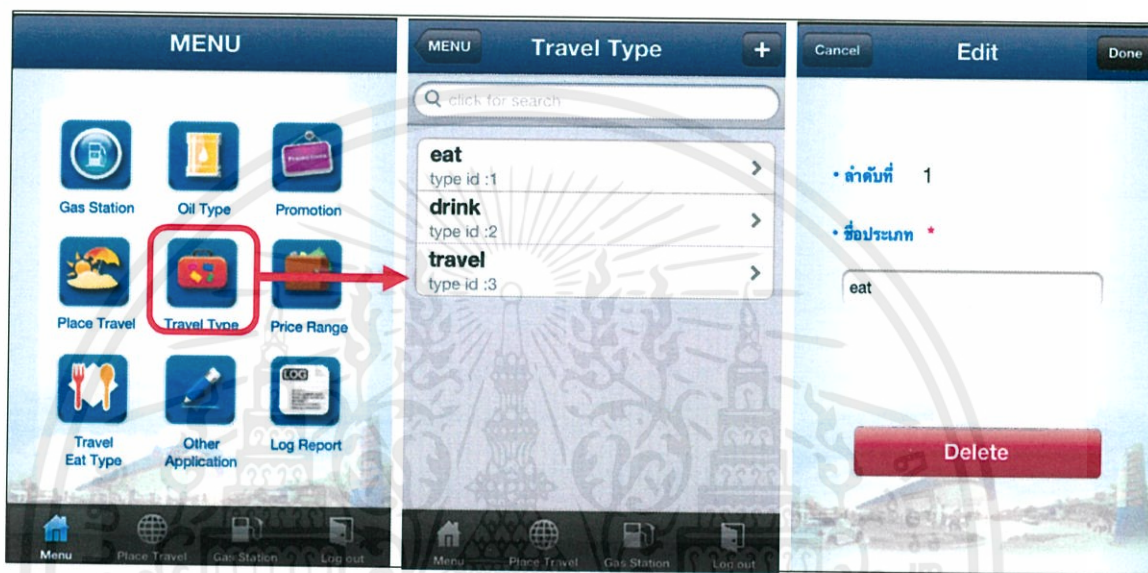
- วันเริ่มโปรโมชั่น เป็นการบอกถึงวันที่เริ่มโปรโมชั่น

- วันสิ้นสุดโปรโมชั่น เป็นการบอกถึงวันที่สิ้นสุด โปรโมชั่น

- รายละเอียด เป็นการอธิบายถึงรายละเอียดของโปรโมชัน
- สถานะ เป็นการตั้งค่าแสดงผลของโปรโมชันนั้นๆ

• เมนู Travel Type

ในเมนู Travel Type นั้นจะเป็นเมนูที่ทำการจัดการข้อมูลประเภทของสถานที่ท่องเที่ยวต่างๆ โดยสามารถเข้าใช้งานโดยการกดไปที่ไอคอน Travel Type



รูปที่ ๓-30 เมนู Travel Type

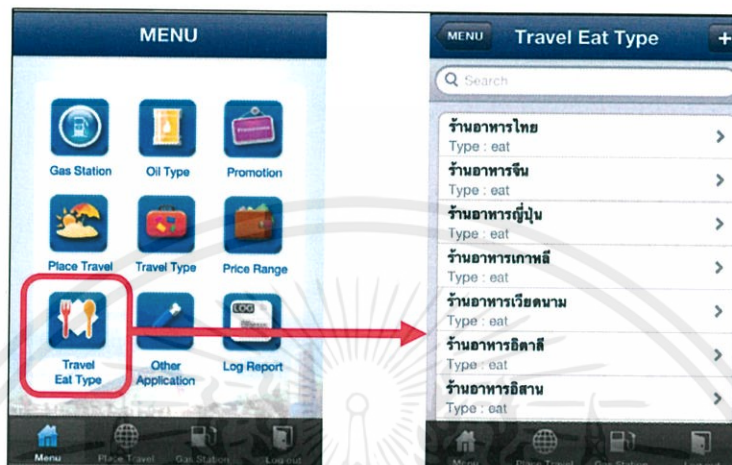
ในการดูข้อมูล, การเพิ่มข้อมูล, การแก้ไขข้อมูล และการลบข้อมูลนั้นจะทำคล้ายๆกันกับในส่วนของเมนู Place Travel โดยในส่วนข้อมูลของเมนู Travel Type มีดังนี้

- ชื่อประเภท เป็นการบอกถึงชื่อประเภทของสถานที่ท่องเที่ยว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• เมนู Travel Eat Type

ในเมนู Travel Eat Type นั้นจะเป็นเมนูที่ทำการจัดการข้อมูลประเภทร้านอาหารที่อยู่ภายในแต่ละประเภทการท่องเที่ยวในเมนู Travel Type ที่โดยสามารถเข้าใช้งาน โดยการกดไปที่ ไอคอน Travel Eat Type



รูปที่ ฉ-33 เมนู Travel Eat Type

ในการดูข้อมูล, การเพิ่มข้อมูล, การแก้ไขข้อมูล และการลบข้อมูลนั้นจะทำคล้ายๆกันกับในส่วนของเมนู Place Travel โดยในส่วนของข้อมูลของเมนู Travel Eat Type มีดังนี้



รูปที่ ฉ-34 ข้อมูลภายในเมนู Travel Eat Type

- ชื่อสถานที่ เป็นการบอกถึงชื่อประเภทของร้านอาหาร
 - ชื่อประเภท เป็นการบอกถึงชื่อประเภทการท่องเที่ยว

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ในการใช้งานเพื่อธุรกิจเท่านั้น ไม่สามารถนำไปใช้เพื่อประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• เมนู Other Application

ในเมนู Other Application นั้นจะเป็นเมนูที่ทำการจัดการข้อมูลของแอปพลิเคชันอื่นที่มีอยู่ภายในแอปพลิเคชัน PTT Life Station โดยสามารถเข้าใช้งานโดยการกดไปที่ไอคอน Other Application



รูปที่ ฉ-35 เมนู Other Application

ในการดูข้อมูล, การเพิ่มข้อมูล, การแก้ไขข้อมูล และการลบข้อมูลนั้นจะทำคล้ายกันกับในส่วนของเมนู Place Travel โดยในส่วนข้อมูลของเมนู Other Application มีดังนี้

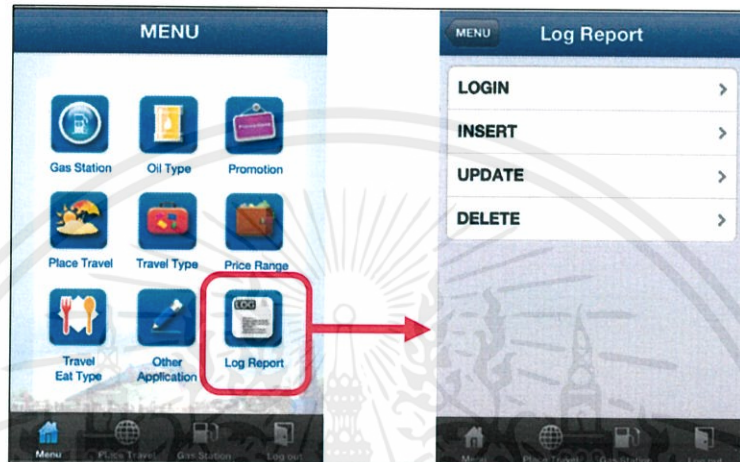


รูปที่ ฉ-36 ข้อมูลภายในเมนู Other Application

- | | |
|---------|---|
| - ชื่อ | เป็นการบอกถึงชื่อแอปพลิเคชันอื่น |
| - URL | เป็นการบอกถึงลิงค์ที่อ้างอิงไปยังแอปพลิเคชันนั้นๆ |
| - สถานะ | เป็นการบอกถึงการแสดงผลของแอปพลิเคชัน |

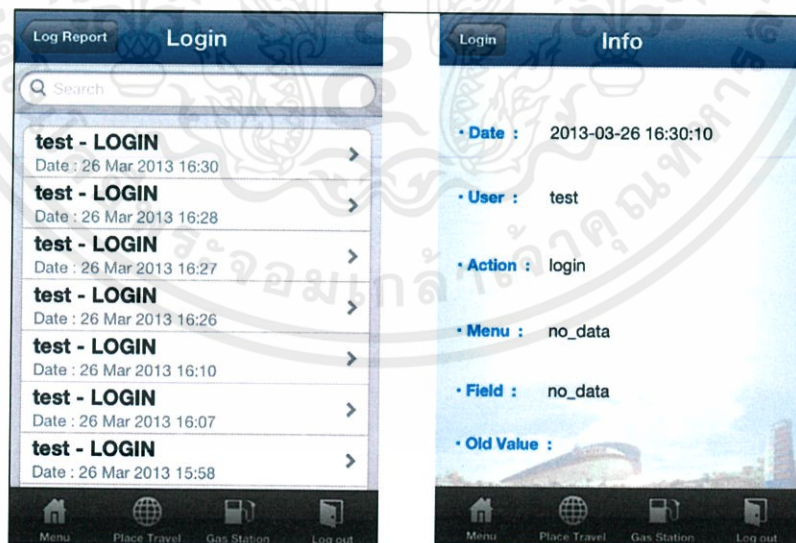
• เมนู Log Report

เมนู Log Report จะเป็นการเรียกข้อมูลจากฐานข้อมูลมาว่าผู้ดูแลระบบคนไหนการแก้ไขข้อมูลอะไรบ้างเนื่องจากทุกครั้งที่มีการล็อกอิน, การเพิ่มข้อมูล, การแก้ไขข้อมูล และการลบข้อมูล จะมีการเก็บข้อมูลการกระทำต่างเอาไว้ในฐานข้อมูล โดย Log Report จะมีการแยกหมวดหมู่ตาม Action ดังรูปที่ ฉ-37



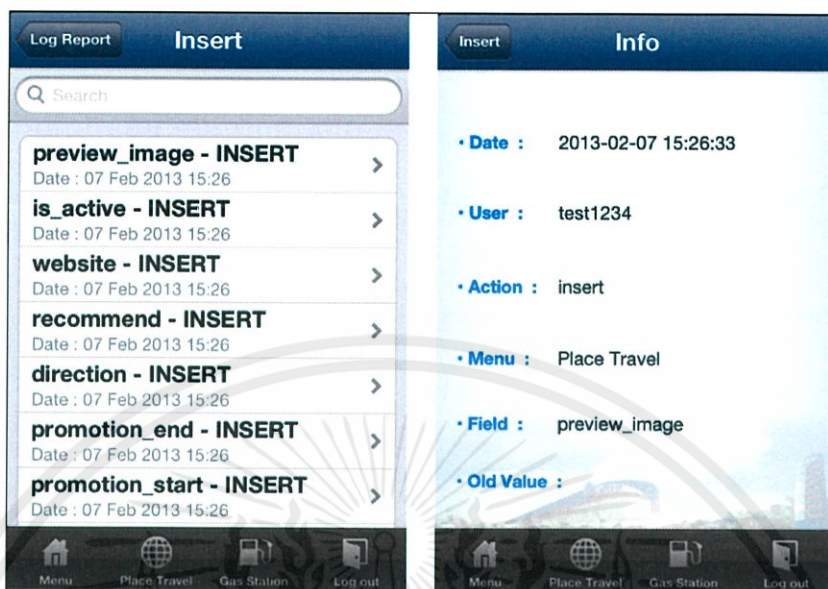
รูปที่ ฉ-37 หน้าเมนู Log Report

LOGIN จะเป็นการดูว่าผู้ดูแลระบบคนไหนเข้ามาใช้งานบ้างและเข้ามาใช้เมื่อไร
 ดังรูปที่ ฉ-38



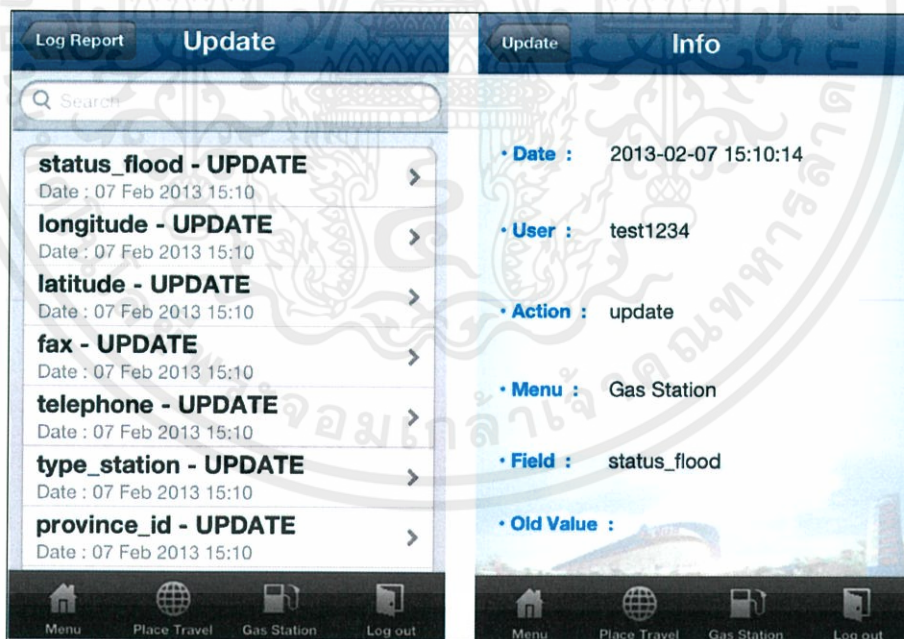
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้รูปที่ ฉ-38 Log Report LOGIN กรุณาอย่าให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INSERT จะเป็นการดูว่าผู้ดูแลระบบคนไหนทำการเพิ่มข้อมูลอะไรบ้างดังรูปที่ จ-39



รูปที่ จ-39 Log Report INSERT

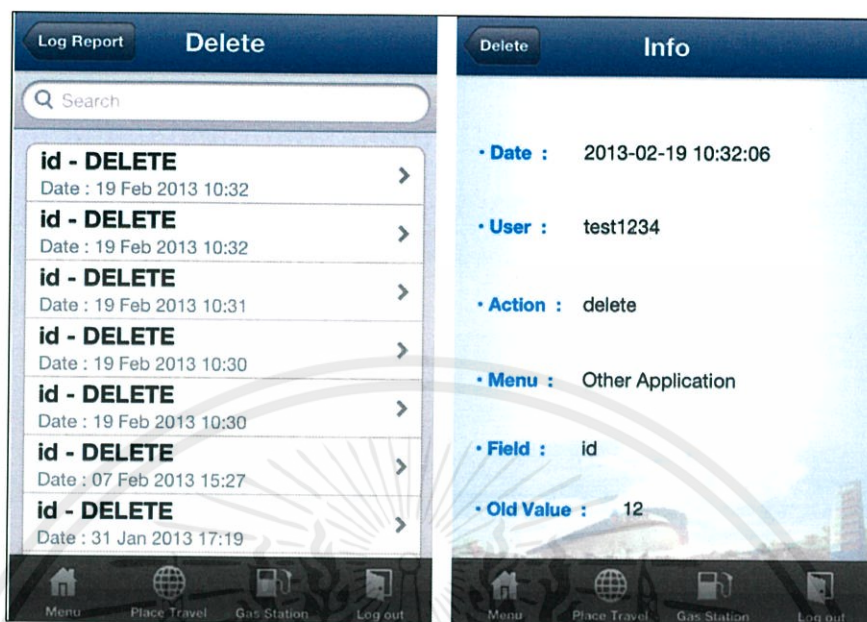
UPDATE จะเป็นการดูว่าผู้ดูแลระบบคนไหนทำการแก้ไขข้อมูลอะไรบ้างดังรูปที่ จ-40



รูปที่ จ-40 Log Report UPDATE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DELETE จะเป็นการดูว่าผู้ดูแลระบบคนไหนทำการลบข้อมูลอะไรบ้างดังรูปที่ ฉ-41



รูปที่ ฉ-41 Log Report DELETE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้