

ระบบจัดหมวดหมู่เนื้อหาอัตโนมัติสำหรับเว็บไซต์ด้านการศึกษา
 AUTOMATIC CONTENT CATEGORIZATION FOR TRAVEL WEB SITE



นายณัฐวุฒิ เขียวอ้าย รหัส 47380021

นายเอกณรงค์ นามอินทร์ รหัส 47380057

ภาควิชา คณะศึกษาศาสตร์ วันที่รับ..... 17 พ.ย. 2554 เลขทะเบียน..... 15705642 เลขเรียกหนังสือ..... นร. มหาวิทยาลัยนเรศวร ๘๖๓/๕
--

๒๕๕๓

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาระดับปริญญาวิทยาศาสตรบัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะศึกษาศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2553

15705642

นร.

๘๖.๓๖/๕

๒๕๕๓



ใบรับรองปริญญาโท

ชื่อหัวข้อโครงการ	ระบบจัดหมวดหมู่เนื้อหาอัตโนมัติสำหรับเว็บไซต์ด้านการท่องเที่ยว		
ผู้ดำเนินโครงการ	นายณัฐฉัตร	เชียวอ้าย	รหัสสถิติ 47380021
	นายเอกณรงค์	นามอินทร์	รหัสสถิติ 47380057
ที่ปรึกษาโครงการ	นายสิริภพ	กชรัตน์	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2553		

.....
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยราชภัฏบรจรัม อนุมัติให้ปริญญาโทฉบับนี้เป็นส่วนหนึ่ง
ของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

.....
.....
.....ที่ปรึกษาโครงการ
(นายสิริภพ กชรัตน์)

.....
.....กรรมการ
(นายกาญจพงศ์ สอนคม)

.....
.....กรรมการ
(นายเศรษฐา คั้งคำวานิช)

ชื่อหัวข้อโครงการ..... ระบบจัดหมวดหมู่เนื้อหาอัตโนมัติสำหรับเว็บไซต์ด้านการท่องเที่ยว
ผู้ดำเนินโครงการ นายณัฐวุฒิ เขียวอ้าย รหัส 47380021
นายเอกณรงค์ นามอินทร์ รหัส 47380057
ที่ปรึกษาโครงการ อาจารย์สิรภพ กษรรัตน์
สาขาวิชา วิศวกรรมคอมพิวเตอร์
ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา 2553

.....

บทคัดย่อ

โครงการนี้เป็นการออกแบบอัลกอริทึมและพัฒนาโปรแกรมสำหรับการจัดหมวดหมู่ให้กับเนื้อหาในเว็บไซต์ด้านการท่องเที่ยวแบบอัตโนมัติ โดยใช้หลักการเปรียบเทียบค่าสำคัญของแต่ละหมวดหมู่เพื่อเป็นตัวชี้ว่าแต่ละเนื้อหาควรอยู่ในหมวดหมู่ใด สำหรับโปรแกรมที่พัฒนาขึ้นนี้จะประกอบด้วยส่วนที่ใช้จัดการหมวดหมู่และส่วนที่ใช้สืบค้นเนื้อหาข้อมูลตามหมวดหมู่ โดยส่วนที่ใช้จัดการหมวดหมู่จะใช้จัดการค่าสำคัญรวมถึงค่าเหมือนค่าคล้ายของแต่ละหมวดหมู่ และความสัมพันธ์ระหว่างหมวดหมู่ เพื่อเอาไว้ใช้ในกระบวนการระบุหมวดหมู่แบบอัตโนมัติ ซึ่งถ้ายังมีข้อมูลเหล่านี้มากก็จะยิ่งทำให้การสืบค้นเอกสารตามหมวดหมู่มีความถูกต้องมากขึ้น สำหรับโปรแกรมจะถูกออกแบบในลักษณะเว็บแอปพลิเคชัน (Web Application) โดยใช้ภาษา ASP.NET C# และ Microsoft Visual Studio 2010 เป็นเครื่องมือในการพัฒนา และใช้ Microsoft SQL Server เป็นตัวจัดการฐานข้อมูล

สำหรับอัลกอริทึมที่พัฒนาขึ้น จะแบ่งเป็นวิธีที่ใช้การตัดคำและวิธีที่ไม่ใช้การตัดคำ โดยหากไม่ใช้การตัดคำโปรแกรมจะสามารถระบุหมวดหมู่ที่เกี่ยวข้องกับเอกสารตัวอย่างได้อย่างแม่นยำอิงตามค่าสำคัญที่กำหนดไว้ แต่จะใช้ในการระบุหมวดหมู่ให้กับเนื่อนานกว่าวิธีที่ใช้การตัดคำ ซึ่งวิธีที่ใช้การตัดคำอาจเกิดความคิดพลาดในการเปรียบเทียบค่าสำคัญ แต่ก็ใช้เวลาน้อยกว่าในการระบุหมวดหมู่ให้กับเนื้อหา

Project Title Automatic Content Categorization for Travel Web Site
Name Mr. Nuttawut Khew-ai ID 47380021
 Mr. Aeknarong Nam-in ID 47380057
Project Advisor Mr. Sirapop Khotcharrat
Major Computer Engineering
Department Electrical and Computer Engineering
Academic Year 2010

.....

Abstract

This project has the purpose to design the algorithm and develop a program for automatic categorization of the content of a travel web site based on category keywords comparing in order to assign categories to the content. The program composes of the category management module and the content browsing module. The category management module is used for managing the category keywords, category synonym keywords, and relations between each category used for the automatic categorization process. The accuracy of the system depend on the number of keywords and category relations data in which the more data, the more accuracy. Our program has been implemented using ASP.NET C# with Microsoft Visual Studio 2010 as a development tool and Microsoft SQL Server as a DBMS.

For our algorithm, there are two methods, one that uses word segmentation and one that does not use word segmentation. The method that does not use word segmentation has more accuracy than the other one but uses more time for categorization process. In contrast, the method that uses word segmentation uses less time for categorization process but may lead to the missing keyword comparing cause by word segmentation.

กิตติกรรมประกาศ

โครงการวิศวกรรมคอมพิวเตอร์สำเร็จได้ด้วยดีก็เนื่องด้วยความอนุเคราะห์จากท่านอาจารย์ที่ปรึกษา คือ อาจารย์สิรภพ กษรัตน์ ผู้ซึ่งให้ความรู้ คำแนะนำและเอาใจใส่เป็นอย่างดีในตลอดระยะเวลาการดำเนินโครงการ คณะผู้จัดทำจึงขอกราบขอบพระคุณอย่างสูงไว้ ณ โอกาสนี้ด้วย ขอขอบคุณบิดามารดา และเพื่อนๆ ทุกคนที่คอยเป็นกำลังใจอีกทั้งขอขอบคุณทุกๆ ท่านที่มีส่วนช่วยทำให้โครงการนี้ประสบความสำเร็จได้ด้วยดี



นายณัฐวุฒิ

เจิวฮ้าย

นายเอกณเรงค์

นามอินทร์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ฉ
สารบัญรูปภาพ	ช
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.4 ขอบเขตของโครงการ	2
1.5 ขั้นตอนการดำเนินงาน	2
1.6 รายละเอียดงบประมาณตลอดโครงการ.....	2
บทที่ 2 หลักการและทฤษฎี	4
2.1 ความหมายของการจำแนกหมวดหมู่เอกสาร	5
2.2 การเรียนรู้ด้วยคอมพิวเตอร์ (machine learning)	6
2.3 การตัดคำภาษาไทย	7
2.4 Inverted Index	8
2.5 ระบบฐานข้อมูล	9
2.6 ภาษา SQL	10
2.7 วิธีการทำนอร์มัลไลซ์	22
2.8 ภาษาซีชาร์ป	23
2.9 เอเอสพีคอตเน็ต (ASP.NET)	25
2.10 การใช้งาน Class String.....	27

สารบัญ (ต่อ)

	หน้า
บทที่ 3 วิธีการดำเนินงาน	33
3.1 วิเคราะห์ความต้องการของระบบจัดหมวดหมู่เนื้อหาอัตโนมัติสำหรับเว็บด้านการ ท่องเที่ยว	34
3.2 การออกแบบระบบ	35
3.3 การออกแบบเว็บไซต์	54
<hr/>	
บทที่ 4 ผลการทดลอง	67
4.1 ผลการทำงานของระบบ	67
4.2 วิเคราะห์ผลการทดลอง	78
บทที่ 5 บทสรุป	83
5.1 ผลการทำงาน	83
5.2 ปัญหาและแนวทางแก้ไข	84
5.3 ข้อเสนอแนะ	84
เอกสารอ้างอิง	85
ภาคผนวก ก. ภาคผนวก ก Microsoft SQL Server.....	86
ประวัติผู้เขียนโครงการ	90

สารบัญตาราง

ตารางที่	หน้า
3.1 ตารางแสดงหมวดหมู่สถานที่ท่องเที่ยว	52
3.2 ตารางแสดงการเก็บเอกสารข้อมูลสถานที่ท่องเที่ยว	52
3.3 ตารางแสดงการเก็บเอกสารการท่องเที่ยวตามหมวดหมู่ของสถานที่ท่องเที่ยว	53
3.4 ตารางแสดงการเก็บข้อมูลสมาชิก.....	53
3.5 ตารางแสดงการเก็บข้อมูลคำคล้าย	54
4.1 ตารางแสดงจำนวนของหมวดหมู่เปรียบเทียบกับเวลาที่ใช้ในการวิเคราะห์หมวดหมู่เอกสาร โดยไม่ใช้วิธีตัดคำ.....	80
4.2 ตารางแสดงจำนวนของหมวดหมู่เปรียบเทียบกับเวลาที่ใช้ในการวิเคราะห์หมวดหมู่เอกสาร โดยใช้วิธีตัดคำ.....	81
4.3 ตารางแสดงเวลาเฉลี่ยของที่ใช้ในการวิเคราะห์หมวดหมู่เอกสาร โดยใช้วิธีตัดคำ และไม่ตัดคำ.....	82



สารบัญรูปภาพ

รูปที่	หน้า
2.1 ภาพแสดงการจัดกลุ่ม	4
2.2 การจำแนกหมวดหมู่	4
2.3 แสดงการตัดคำแบบ N-gram	7
2.4 การสร้างดัชนีแบบ Inverted Index	8
2.5 แสดงลำดับขั้นการทำงานอัลกอริทึม	22
3.1 ขั้นตอนในการพัฒนาระบบ	33
3.2 ภาพแสดง Context Diagram DFD Level 0.....	37
3.3 ภาพแสดง Context Diagram DFD Level 1.....	38
3.4 ภาพแสดง Context Diagram DFD Level 2 : Login	39
3.5 ภาพแสดง Context Diagram DFD Level 2 : เพิ่ม/ลบ/แก้ไข Category	40
3.6 ภาพแสดง Context Diagram DFD Level 2 : เพิ่ม/ลบ/แก้ไข Document	40
3.7 แสดงผลรวมค่าแวกเตอร์ที่น้อยที่สุดของทุกช่วงภาพแสดง Context Diagram DFD Level 2 : เพิ่ม/ลบ/แก้ไข Relation	41
3.8 ภาพแสดง Context Diagram DFD Level 2 : เพิ่ม/ลบ/แก้ไข คำคล้าย คำใกล้เคียง.....	41
3.9 ภาพแสดง Context Diagram DFD Level 2 : จัดหมวดหมู่อัตโนมัติ	42
3.10 ภาพแสดง Context Diagram DFD Level 2 : การค้นหา.....	43
3.11 รูปแสดงการจัดหมวดหมู่โดยเปรียบเทียบตัวอักษร.....	44
3.12 รูปแสดงการจัดหมวดหมู่โดยการตัดคำ.....	45
3.13 แสดงภาพ ER Diagram ของตารางหมวดหมู่สถานที่ท่องเที่ยว.....	46
3.14 แสดงภาพ ER Diagram ของตารางเก็บเอกสารข้อมูลสถานที่ท่องเที่ยว.....	47
3.15 แสดงภาพ ER Diagram ของตารางเก็บเอกสารการท่องเที่ยวตามหมวดหมู่ของสถานที่ ท่องเที่ยว.....	48
3.16 แสดงภาพ E-R Diagram ของตารางเก็บข้อมูลสมาชิก.....	49
3.17 แสดงภาพ ER Diagram ของตารางเก็บข้อมูลคำคล้าย.....	50
3.18 แสดงความสัมพันธ์ตารางฐานข้อมูลระบบจัดหมวดหมู่ข้อมูลการท่องเที่ยวอัตโนมัติ.....	51
3.19 รูปแสดง โครงสร้างของเว็บไซต์จัดหมวดหมู่อัตโนมัติสำหรับการท่องเที่ยว.....	55
3.20 รูปแสดง หน้าหลักเว็บไซต์จัดหมวดหมู่อัตโนมัติสำหรับการท่องเที่ยว.....	56

สารบัญรูปรูปภาพ (ต่อ)

3.21	รูปแสดงหน้า Administrator.....	57
3.22	รูปแสดงหน้า About.....	57
3.23	รูปแสดงส่วนเพิ่มหมวดหมู่.....	58
3.24	รูปแสดงส่วนแก้ไขหมวดหมู่.....	59
3.25	รูปแสดงส่วนเพิ่มคำเหมือนคำคล้าย.....	60
3.26	รูปแสดงส่วนแก้ไขคำเหมือนคำคล้าย.....	61
3.27	รูปแสดงส่วนเพิ่มและลบความสัมพันธ์ระหว่างหมวดหมู่.....	62
3.28	รูปแสดงส่วนเพิ่มเอกสาร.....	63
3.29	รูปแสดงส่วนแก้ไขเอกสาร 1.....	64
3.30	รูปแสดงส่วนแก้ไขเอกสาร 2.....	65
4.1	รูปแสดงการเพิ่มชุดหมวดหมู่ตัวอย่าง.....	68
4.2	รูปแสดงการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่พระบรมมหาราชวัง.....	68
4.3	รูปแสดงการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดจุฬามณี.....	69
4.4	รูปแสดงการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดเจดีย์หลวงวิหาร.....	69
4.5	รูปแสดงการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดเชียงมั่น.....	70
4.6	รูปแสดงการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดพระศรีมหาธาตุวรมหาวิหาร.....	70
4.7	รูปแสดงการเพิ่มเอกสารที่ 1 และทำการวิเคราะห์ข้อมูลเพื่อจัดหมวดหมู่.....	72
4.8	รูปแสดงการเพิ่มเอกสารที่ 2 และทำการวิเคราะห์ข้อมูลเพื่อจัดหมวดหมู่.....	73
4.9	รูปแสดงการเพิ่มเอกสารที่ 3 และทำการวิเคราะห์ข้อมูลเพื่อจัดหมวดหมู่.....	73
4.10	รูปแสดงการเพิ่มเอกสารที่ 4 และทำการวิเคราะห์ข้อมูลเพื่อจัดหมวดหมู่.....	74
4.11	รูปแสดงการเพิ่มเอกสารที่ 5 และทำการวิเคราะห์ข้อมูลเพื่อจัดหมวดหมู่.....	74
4.12	รูปแสดงการค้นหาคำตามหมวดหมู่เพื่อค้นหาเอกสารที่ 1 เมื่อเลือกหมวดหมู่ พระบรมมหาราชวัง.....	75
4.13	รูปแสดงการค้นหาคำตามหมวดหมู่เพื่อค้นหาเอกสารที่ 2 เมื่อเลือกหมวดหมู่วัดจุฬามณี.....	75
4.14	รูปแสดงการค้นหาคำตามหมวดหมู่เพื่อค้นหาเอกสารที่ 3 เมื่อเลือกหมวดหมู่วัดเจดีย์หลวงวิหาร	76
4.15	รูปแสดงการค้นหาคำตามหมวดหมู่เพื่อค้นหาเอกสารที่ 4 เมื่อเลือกหมวดหมู่วัดเชียงมั่น.....	76
4.16	รูปแสดงการค้นหาคำตามหมวดหมู่เพื่อค้นหาเอกสารที่ 5 เมื่อเลือกหมวดหมู่วัดพระศรีรัตนมหา ธาตุ.....	77

สารบัญรูปภาพ (ต่อ)

4.17 รูปแสดงการเพิ่มความสัมพันธ์ระหว่างหมวดหมู่วัดจุฬามณีกับพระบรมมหาราชวัง.....	77
4.18 รูปแสดงการค้นหาเอกสารหลังเพิ่มความสัมพันธ์ระหว่างหมวดหมู่วัดจุฬามณีกับพระบรมมหาราชวัง.....	78
ก-1 แสดงขั้นตอนการติดตั้งโปรแกรม SQL Server Management Studio Express.....	84
ก-2 แสดงขั้นตอนการติดตั้งโปรแกรม.....	84
ก-3 แสดงขั้นตอนการเพิ่ม Name และ Company.....	85
ก-4 แสดงขั้นตอนการเลือก path.....	85
ก-5 แสดงขั้นตอนการเริ่มติดตั้งโปรแกรม.....	86
ก-6 แสดงขั้นตอนการเริ่มติดตั้งโปรแกรมขั้นตอนที่2.....	86
ก-7 แสดงขั้นตอนการเริ่มติดตั้งโปรแกรมขั้นตอนที่3.....	87
ก-7 แสดงขั้นตอนเสร็จสิ้นการติดตั้งโปรแกรม.....	87



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของโครงการ

ในปัจจุบันมีผู้ที่มีความสนใจในการหาค้นหาข้อมูลด้านการท่องเที่ยวมากขึ้น ทำให้มีเว็บไซต์ด้านการท่องเที่ยวเพื่อรองรับความต้องการของผู้ที่สนใจมากขึ้นตามไปด้วย อีกทั้งเอกสารข้อมูลด้านการท่องเที่ยวในรูปแบบอิเล็กทรอนิกส์มีปริมาณและเนื้อหาที่หลากหลายมากขึ้น

การเข้าถึงข้อมูลด้านการท่องเที่ยวสามารถเข้าถึงได้หลากหลายวิธี เช่น การค้นหาแบบ Full-Text Search ซึ่งค้นหาโดยใช้คีย์เวิร์ดในการค้นหา และอีกวิธีคือการค้นหาตามหมวดหมู่ของข้อมูลด้านการท่องเที่ยว ซึ่งต้องทำการจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวก่อนที่จะนำมาใช้ในการค้นหา โดยการค้นหาข้อมูลด้านการท่องเที่ยวตามหมวดหมู่นั้นต้องอาศัยคนในการวิเคราะห์และจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวให้อยู่ตามหมวดหมู่ที่ต้องการ ซึ่งจะไม่มีปัญหาถ้าข้อมูลด้านการท่องเที่ยวที่ต้องการจัดหมวดหมู่นั้นมีปริมาณน้อย แต่จะมีปัญหาในการจัดหมวดหมู่ในกรณีที่ข้อมูลด้านการท่องเที่ยวที่ต้องการจัดหมวดหมู่นั้นมีจำนวนมาก ทำให้ต้องใช้จำนวนบุคลากรในการจัดหมวดหมู่มากขึ้น อีกทั้งอาจเกิดข้อผิดพลาดในการจัดหมวดหมู่ขึ้นได้

จากปัญหาการจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวที่มีปริมาณมาก เพื่อใช้สำหรับการค้นหาตามหมวดหมู่นั้นใช้เวลานานในการจัดหมวดหมู่ และใช้ยังใช้บุคลากรจำนวนมากในการจัดหมวดหมู่ จึงได้คิดอัลกอริทึมในการจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวแบบอัตโนมัติ โดยใช้หลักการเทียบค่าของข้อมูลการท่องเที่ยวกับหมวดหมู่ข้อมูลการท่องเที่ยว และได้พัฒนาเว็บแอปพลิเคชันต้นแบบสำหรับส่วนผู้ดูแลระบบที่ใช้ในการจัดหมวดหมู่อัตโนมัติขึ้นมา

1.2 วัตถุประสงค์ของโครงการ

1.2.1 ออกแบบอัลกอริทึมสำหรับจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวแบบอัตโนมัติ โดย
ใช้หลักการเปรียบเทียบค่าของข้อมูลการท่องเที่ยวกับหมวดหมู่ข้อมูลการท่องเที่ยวในฐานข้อมูล

1.2.2 สร้างเว็บแอปพลิเคชันต้นแบบเพื่อใช้ในการจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวโดย
อัตโนมัติ

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1.3.1 เว็บไซต์อิเล็กทรอนิกส์ต้นแบบสามารถนำไปประยุกต์ใช้สำหรับส่วนผู้ดูแลระบบเว็บไซต์ด้านท่องเที่ยวให้สามารถจัดหมวดหมู่ข้อมูลหรือเนื้อหาได้อย่างถูกต้อง ประหยัดเวลา และลดจำนวนบุคลากรที่ใช้ในการจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยว
- 1.3.2 ประหยัดเวลาของผู้ดูแลระบบในการจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยว

1.4 ขอบเขตของโครงการ

- 1.4.1 สร้างเว็บไซต์อิเล็กทรอนิกส์ต้นแบบสำหรับจัดการข้อมูลด้านการท่องเที่ยวอัตโนมัติโดยใช้ภาษา ASP.NET, C# ในการพัฒนา
- 1.4.2 มีระบบ Login สำหรับผู้จัดการระบบจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยว
- 1.4.3 ทำการ เพิ่ม ลบ แก้ไข หมวดหมู่ข้อมูลด้านการท่องเที่ยวได้
- 1.4.4 ทำการ เพิ่ม ลบ แก้ไข เอกสารข้อมูลด้านการท่องเที่ยวได้
- 1.4.5 ทำการ เพิ่ม ลบ แก้ไข ความสัมพันธ์ของหมวดหมู่ข้อมูลด้านการท่องเที่ยวได้
- 1.4.6 ทำการ เพิ่ม ลบ แก้ไข คำใกล้เคียงที่ใช้ในการเปรียบเทียบกับข้อมูลด้านการท่องเที่ยวที่ยังไม่ได้จัดหมวดหมู่
- 1.4.7 ทำการ เพิ่ม ลบ แก้ไข ความสัมพันธ์ระหว่างหมวดหมู่สถานที่ท่องเที่ยว และกิจกรรมท่องเที่ยว
- 1.4.8 ทำการ ค้นหาแบบเลือกตามหมวดหมู่ข้อมูลการท่องเที่ยวได้
- 1.4.9 จัดหมวดหมู่เอกสารด้านการท่องเที่ยวโดยใช้การเปรียบเทียบคำกับข้อมูลด้านการท่องเที่ยวในฐานข้อมูล

1.5 ขั้นตอนการดำเนินงาน

กิจกรรม	ม.ค.54	ก.พ.54	มี.ค.-เม.ย.54	พ.ค.54
ศึกษาการเขียนภาษา ASP.NET ,C# และ Microsoft SQL Server Management Studio Express	←→			
ออกแบบอัลกอริทึมที่ใช้ในการจัดหมวดหมู่ข้อมูลการท่องเที่ยว	←→			
ออกแบบเว็บเพจ		←→		
เขียนโปรแกรมและจัดทำฐานข้อมูล		←→		
ทดสอบโปรแกรม และ วิเคราะห์โครงการ			←→	
นำเสนอผลงานและจัดทำรายงาน				←→

1.6 รายละเอียดงบประมาณตลอดโครงการ

1.6.1 ค่ากระดาษรายงาน	100	บาท
1.6.2 ค่าหนังสือ	400	บาท
1.6.3 ค่าซอฟต์แวร์ / วัสดุ	700	บาท
1.6.4 ค่าถ่ายเอกสาร/เข้ารูปเล่ม	800	บาท
รวมเป็นเงินทั้งสิ้น	2,000	บาท

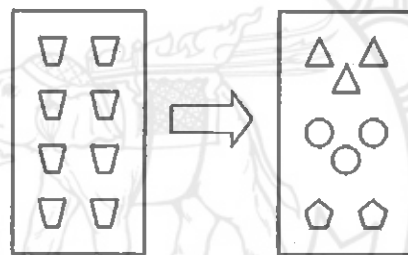
(สองพันบาทถ้วน)

บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

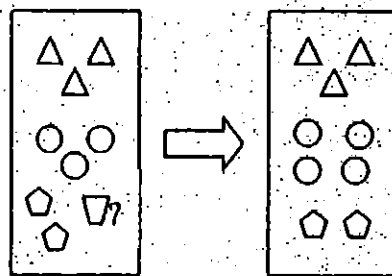
การสืบค้นสารสนเทศ (Information Retrieval) [1] นั้นได้รับความสนใจอย่างมากเนื่องจากการเพิ่มปริมาณของข้อมูลดิจิทัลที่สามารถนำมาใช้งานได้ตามความต้องการในการเข้าถึงข้อมูลเหล่านั้นด้วยวิธีการที่หลากหลาย การแบ่งเอกสารตามเนื้อหาเพื่อการสืบค้นสารสนเทศนั้น อาจแบ่งได้ 2 ลักษณะ คือ การจัดกลุ่ม (Clustering) และการจำแนกหมวดหมู่ (Classification หรือ Categorization)

การจัดกลุ่มของเอกสาร คือ การแบ่งกลุ่มตามเนื้อหาของเอกสาร โดยไม่มีการกำหนดกลุ่มหรือหมวดหมู่ของเอกสารไว้ก่อน ซึ่งจะเป็นการแบ่งกลุ่มตามลักษณะของเอกสาร โดยเอกสารที่มีลักษณะเหมือนกันจะอยู่ด้วยกัน ดังแสดงในรูปที่ 2.1



รูปที่ 2.1 ภาพแสดงการจัดกลุ่ม

การจำแนกหมวดหมู่ของเอกสาร คือ การแบ่งกลุ่มตามเนื้อหาของเอกสารโดยที่มีการกำหนดกลุ่มหรือหมวดหมู่ของเอกสารไว้ก่อน โดยจะเปรียบเทียบเอกสารกับต้นแบบในแต่ละหมวดหมู่ เอกสารจะถูกจัดอยู่ในหมวดหมู่ที่ต้นแบบมีลักษณะคล้ายกับตัวมันเองมากที่สุด ดังแสดงในรูปที่ 2.2



รูปที่ 2.2 การจำแนกหมวดหมู่

2.1 ความหมายของการจำแนกหมวดหมู่เอกสาร [1]

การจำแนกหมวดหมู่เอกสาร (Text Categorization) คือ กิจกรรมในการแยกเอกสารซึ่งประกอบด้วยภาษาธรรมชาติให้อยู่ภายใต้หมวดหมู่ที่กำหนดไว้ก่อน โดยใช้ใจความสำคัญของเอกสาร วิธีการที่นิยมใช้ในการจำแนกหมวดหมู่เอกสารคือ วิธีการทางด้านวิศวกรรมความรู้ (Knowledge Engineering) ซึ่งเป็นการกำหนดชุดของกฎที่สร้างจากความรู้ของผู้เชี่ยวชาญเพื่อใช้แยกประเภทของเอกสารภายใต้หมวดหมู่ที่กำหนด โดยการกำหนดชุดของกฎดังกล่าวต้องกระทำโดยคน ต่อมาในทศวรรษที่ 90 วิธีการนี้ได้รับความนิยมลดน้อยลงเมื่อเทียบกับวิธีการทางการเรียนรู้ด้วยคอมพิวเตอร์ (Machine Learning) ซึ่งเป็นการสร้างตัวจำแนกหมวดหมู่เอกสารอัตโนมัติ (Automatic Text Classifier) ด้วยเครื่องคอมพิวเตอร์โดยการเรียนรู้ด้วยวิธีการเชิงอุปนัยจากชุดของเอกสารที่ได้จำแนกประเภทไว้ก่อน และลักษณะเฉพาะของหมวดหมู่ที่เกี่ยวข้องข้อดีของวิธีการทางการเรียนรู้ด้วยคอมพิวเตอร์ คือ ความถูกต้องของผลลัพธ์ที่ใกล้เคียงกับการจำแนกหมวดหมู่ของเอกสารที่ทำโดยคน และการประหยัดแรงงานคนเป็นอย่างมาก เพราะไม่ต้องใช้ผู้เชี่ยวชาญในการสร้างตัวจำแนกประเภทเอกสาร หรือในการปรับเปลี่ยนหมวดหมู่ของเอกสาร

การจำแนกหมวดหมู่เอกสารสามารถแบ่งประเภทได้หลายลักษณะขึ้นอยู่กับแง่มุมในการพิจารณาแบ่งการจำแนกหมวดหมู่เอกสารไว้ ดังนี้

2.1.1 การจำแนกแบบหมวดหมู่เดียว (Single-Label Text Categorization) และการจำแนกแบบหลายหมวดหมู่ (Multi-label Text Categorization)

2.1.1.1 การจำแนกแบบหมวดหมู่เดียว เป็นการกำหนดหมวดหมู่ให้กับเอกสารเพียงเอกสารละหนึ่งหมวดหมู่เท่านั้น

2.1.1.2 การจำแนกแบบหลายหมวดหมู่ เป็นการกำหนดจำนวนหมวดหมู่ให้กับเอกสารตั้งแต่ 0 ไปจนถึง N หมวดหมู่

2.1.2 การจำแนกโดยใช้หมวดหมู่เป็นตัวหลัก (Category-Pivoted Categorization, CPC) และการจำแนกโดยใช้เอกสารเป็นตัวหลัก (Document-Pivoted Categorization, DPC)

2.1.2.1 การจำแนกโดยใช้หมวดหมู่เป็นตัวหลัก เป็นการพิจารณาหมวดหมู่ $c_i \in C$ แต่ละหมวดหมู่เพื่อหาเอกสาร $d_j \in D$ ทั้งหมดที่ควรอยู่ภายใต้หมวดหมู่ c_i

2.1.2.2 การจำแนกโดยใช้เอกสารเป็นตัวหลัก เป็นการพิจารณาเอกสาร $d_j \in D$ เพื่อหาหมวดหมู่ $c_i \in C$ ทั้งหมดให้กับเอกสาร d_j

2.1.3 การจำแนกหมวดหมู่แบบบังคับ (Hard Categorization) และการจำแนกหมวดหมู่แบบจัดลำดับ (Ranking Categorization)

2.1.3.1 การจำแนกหมวดหมู่แบบบังคับ เป็นการจำแนกหมวดหมู่โดยตัดสินใจเลือกค่าผลลัพธ์ของฟังก์ชันเป้าหมายว่าจะให้มีค่าเป็น T หรือ F ซึ่งเป็นการจำแนกหมวดหมู่แบบอัตรโนมิตี

2.1.3.2 การจำแนกหมวดหมู่แบบจัดลำดับ เป็นการจำแนกหมวดหมู่โดยจัดลำดับค่าผลลัพธ์ของฟังก์ชันเป้าหมายเรียงลำดับตามลำดับหมวดหมู่ซึ่งเป็นการจำแนกหมวดหมู่แบบกึ่งอัตรโนมิตี

2.2 การเรียนรู้ด้วยคอมพิวเตอร์ (Machine learning)

การเรียนรู้ด้วยคอมพิวเตอร์เป็นสาขาหนึ่งของปัญญาประดิษฐ์ (Artificial Intelligent) ที่เกี่ยวข้องกับการออกแบบและพัฒนาอัลกอริทึมและวิธีการที่จะทำให้คอมพิวเตอร์มีความสามารถในการเรียนรู้ โดยทั่วไปวิธีการเรียนรู้มีอยู่ 2 ประเภท ได้แก่ การเรียนรู้เชิงอุปนัย (Inductive Learning) และการเรียนรู้เชิงอนุมาน (Deductive Learning) การเรียนรู้ด้วยคอมพิวเตอร์เชิงอุปนัยเป็นการค้นหากฎ ลักษณะแบบแผน หรือข้อสรุปต่างๆ จากการสังเกตกลุ่มข้อมูลขนาดใหญ่ ส่วนการเรียนรู้เชิงอนุมาน เป็นการหาข้อสรุปจากหลักฐาน หรือข้อเท็จจริงที่มีอยู่ หลักสำคัญของการวิจัยทางด้านการเรียนรู้ด้วยคอมพิวเตอร์ คือ การสกัดเอาความรู้หรือสารสนเทศจากข้อมูลโดยอัตรโนมิตีด้วยวิธีการคำนวณหรือวิธีการทางสถิติ ดังนั้นการเรียนรู้ด้วยคอมพิวเตอร์นั้นจึงมีความสัมพันธ์อย่างใกล้ชิดกับการทำเหมืองข้อมูล (Data Mining) และสถิติ

อัลกอริทึมสำหรับการเรียนรู้ด้วยคอมพิวเตอร์ถูกจัดให้อยู่ในกลุ่มของวิทยาศาสตร์หรือวิธีการที่เกี่ยวข้องกับการแบ่งแยกประเภท (Taxonomy) ซึ่งมีที่มาจากผลลัพธ์ที่ได้จากอัลกอริทึมประเภทของอัลกอริทึมอาจแบ่งได้ ดังต่อไปนี้

2.2.1 การเรียนรู้โดยอาศัยตัวอย่าง (Supervised Learning)

เป็นการเรียนรู้โดยใช้อัลกอริทึมเพื่อสร้างฟังก์ชันที่จะทำข้อมูลเข้าให้เป็นผลลัพธ์ที่ต้องการ การจำแนกประเภทเป็นรูปแบบหนึ่งของการเรียนรู้โดยอาศัยตัวอย่าง ตัวเรียนรู้อาจศึกษาพฤติกรรมของฟังก์ชันที่จะทำการกำหนดเวกเตอร์ $[X_1, X_2, \dots, X_n]$ ให้อยู่ภายใต้ประเภทใดประเภทหนึ่งจากหลายประเภท โดยสังเกตจากตัวอย่างข้อมูลเข้าและตัวอย่างผลลัพธ์ของฟังก์ชัน

2.2.2 การเรียนรู้โดยไม่อาศัยตัวอย่าง (Unsupervised Learning)

เป็นการเรียนรู้โดยการจำลองแบบของชุดข้อมูลเข้าจากลักษณะเฉพาะของข้อมูล โดยไม่ต้องใช้ตัวอย่างผลลัพธ์ที่ถูกจัดประเภทไว้แล้ว

2.2.3 การเรียนรู้กึ่งอาศัยตัวอย่าง (Semi-supervised Learning)

เป็นการเรียนรู้ที่อาศัยทั้งตัวอย่างที่ยังไม่ได้จัดประเภทและตัวอย่างที่ถูกจัดประเภทไว้แล้ว เพื่อสร้างฟังก์ชันหรือตัวจำแนกประเภทที่เหมาะสม

การวิเคราะห์การคำนวณและประสิทธิภาพของอัลกอริทึมที่ใช้สำหรับการเรียนรู้ด้วยคอมพิวเตอร์เป็นแขนงหนึ่งของทฤษฎีทางวิทยาการคอมพิวเตอร์ที่รู้จักกันในชื่อ ทฤษฎีการเรียนรู้เกี่ยวกับการคำนวณ (Computational Learning Theory)

2.3 การตัดคำภาษาไทย

ในการตัดคำภาษาไทยนั้นมีหลากหลายวิธีแต่วิธีที่นิยมคือการใช้เทคนิคการตัดคำแบบ N-gram มาประยุกต์ใช้ในการตัดคำ เพื่อเดาคำ (Suggestion) ที่มีการกรอกผิดพลาดขึ้นมา สำหรับภาษาไทยการหาขอบเขตคำค่อนข้างเป็นปัญหาเนื่องจากลักษณะการเขียนภาษาไทยนั้นไม่มีการใช้ตัวอักษรหรือสัญลักษณ์ที่นำมาใช้คั่นระหว่างคำหรือว่ามีการวรรคระหว่างคำเหมือนภาษาอังกฤษ นั่นคือต้องมีกระบวนการตัดคำ (Word segmentation) ที่เหมาะสมก่อนเป็นอันดับแรก หากจะให้นิยามของการตัดคำ คือ การแบ่งตัวอักษรจากข้อความ (String) เพื่อหาขอบเขตของแต่ละหน่วยคำ (Morpheme) เนื่องจากส่วนใหญ่ภาษาไทยมีการเขียนในลักษณะที่ติดกัน โดยไม่มีการใช้เครื่องหมายวรรคตอนคั่นระหว่างคำเหมือนภาษาอังกฤษดังที่กล่าว ไปซึ่งใช้ช่องว่าง (Space) คั่นระหว่างคำ แต่ภาษาไทยจะมีการเว้นวรรคเป็นระยะเพื่อให้ผู้อ่านทำความเข้าใจกับความหมายของคำการตัดคำโดยใช้ N-Gram โดยทำการนำบางส่วนของข้อความนั้นออกมาเป็นข้อความตามค่า N ซึ่งเรียกวิธีนี้ว่า N-Gram เข้ามาใช้ในการตัดคำแทน และเราจะทำการเพิ่มช่องว่างข้างหน้า และท้ายของข้อความ (ใช้ _ ในการแสดงช่องว่าง) เพื่อเพิ่มโอกาสในการจับคู่ของคำที่อยู่ข้างต้น และท้ายของข้อความ แต่ในภาษาไทยนั้นเราไม่สามารถกำหนดได้ว่า 1 ตัวอักษรคือ 1 Gram เนื่องจากภาษาไทยมีสระและวรรณยุกต์ ดังนั้นคำที่ประกอบไปด้วยสระ และวรรณยุกต์จะถือว่าเป็น 1 Gram เช่น มีข้อความ “สวัสดิ” ทำการตัดคำแบบ N-Gram ได้ดังรูปที่ 2.3

bi-grams :	_ส, สวิ, วิส, สดี, ดี_
tri-grams :	_สวิ, สวิส, วิสดี, สดี_ , ดี_
quad-grams :	_สวิส, สวิสดี, วิสดี_ , สดี_
	_ . ดี_

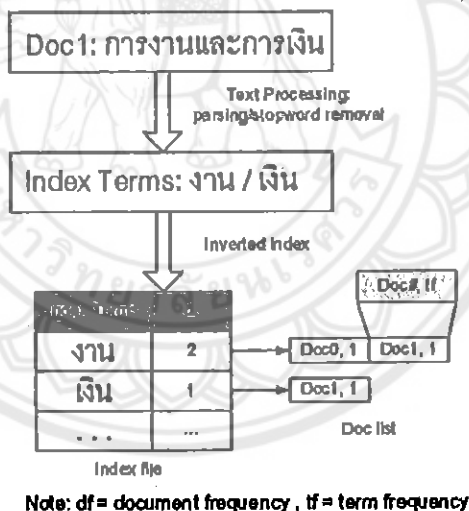
รูปที่ 2.3 แสดงการตัดคำแบบ N-gram

การตัดคำเพื่อสร้างดัชนีค่านั้น ควรใช้แบบจำลอง N-Gram เพื่อเพิ่มความสามารถของการตัดคำ หรือควรรนำหลักการตัดคำโดยใช้กฎ (Rule base approach) และหลักการตัดคำโดยใช้คลังข้อมูล (Corpus based approach) มาผสมผสานกัน เพื่อเพิ่มความถูกต้อง และแม่นยำในการค้นหา

2.4 Inverted Index

Inverted Index หมายถึง โครงสร้างการเก็บคำสำคัญเพื่อการสืบค้น โดยมีการเรียงตามลำดับตัวอักษรและการเชื่อมโยงไปยังรายการอ้างอิงของเอกสารที่ปรากฏคำนั้นๆอยู่

Inverted Index เป็นการมองเนื้อหาของเอกสาร ในมุมกลับกัน แทนที่จะเป็นว่า เอกสารนี้มีคำอะไรบ้าง เป็น เอกสารนี้ถูกอ้างอิงมาจากเอกสารอื่นด้วยคำอะไรบ้าง ซึ่งสามารถทำได้โดยการนำ Matrix ของเนื้อหา มา Inverse เพื่อให้ Index ของคำค้นของไซค์หนึ่งนั้นระบุว่า สอดคล้องกับเอกสารอื่นใดบ้าง



รูปที่ 2.4 การสร้างดัชนีแบบ Inverted Index

การสร้างดัชนีคำ ควรนำเทคนิคอื่นมาใช้ในการวิเคราะห์คำให้ได้ผลมากยิ่งขึ้น เนื่องจากจะมีคำบางประเภทที่ทำให้หาคำในพจนานุกรม เช่น คำกำกวม คำที่ไม่รู้จัก เป็นต้น ควรนำเทคนิคการตัดคำหลายๆ แบบมาประยุกต์ใช้ เช่น การทำเหมืองข้อความ (Text Mining) หรืออาจใช้ การจัดกลุ่มเอกสาร (Document Clustering) ซึ่งเป็นวิธีที่จะจัดกลุ่มเอกสารปริมาณมาก ๆ แบบอัตโนมัติ เพื่อแยกเอกสารออกเป็นกลุ่มตามความคล้ายคลึงและความสัมพันธ์กันซึ่งขึ้นอยู่กับข้อมูลที่ปรากฏในเอกสารแต่ละฉบับ

2.5 ระบบฐานข้อมูล [2]

Database หรือ ระบบฐานข้อมูล คือ ที่อยู่ของข้อมูลที่มีความสัมพันธ์กัน หรือเปรียบเทียบได้ว่าเป็นคลังของข้อมูลก็ได้ ข้อมูลเหล่านี้จะถูกจัดเก็บร่วมกันอย่างมีระบบและมีรูปแบบ ทำให้ง่ายต่อการประมวลผลและการจัดการ โดยปกติแล้วการใช้งานจะต้องมีโปรแกรมเพื่อจัดการฐานข้อมูลที่มีอยู่ ซึ่งเรียกว่า DBMS สำหรับฐานข้อมูลที่ได้รับการนิยมนามากที่สุดในปัจจุบันจะเป็นแบบ Relation ซึ่งจะจัดเก็บ ข้อมูลอยู่ในรูปของตาราง (Table) โดยที่ฐานข้อมูลในแต่ละตารางจะมีความสัมพันธ์ซึ่งกันและกัน

2.5.1 DBMS (Database Management System) หรือ ระบบฐานจัดการฐานข้อมูล

โปรแกรม DBMS จะมีหน้าที่ในการจัดการและควบคุมความถูกต้องความซ้ำซ้อนและความสัมพันธ์ระหว่างข้อมูลต่างๆ ภายในฐานข้อมูลแทนโปรแกรมเมอร์ ทำให้ผู้ใช้สามารถเรียกใช้ข้อมูลจากฐานข้อมูลได้ โดยไม่จำเป็นต้องทราบถึง โครงสร้างทางกายภาพของข้อมูลในระดับลึก เนื่องจากโปรแกรม DBMS จะมีส่วนของ Query Language ที่เป็นภาษาที่ประกอบด้วยคำสั่งต่างๆ ที่ใช้จัดการและเรียกใช้ข้อมูลจากฐานข้อมูล ซึ่งสามารถนำไปใช้ร่วมกับภาษาคอมพิวเตอร์อื่นๆ เพื่อพัฒนาโปรแกรมที่ใช้สำหรับเรียกใช้ฐานข้อมูลมาประมวลผล เช่น Visual Basic, Delphi, AS

2.5.2 องค์ประกอบของฐานข้อมูล

มี 5 ประเภท คือ

1. ฮาร์ดแวร์(Hardware)
2. โปรแกรม (Program หรือ Software) ซึ่งมีหน้าที่ควบคุมดูแลการสร้างฐานข้อมูล การเรียกใช้ข้อมูลและการจัดทำรายงานเรียกว่า โปรแกรมระบบจัดการฐานข้อมูล (Database Management System: DBMS)
3. ข้อมูล (Data)
4. บุคลากร (People Ware) คือ ผู้ใช้งาน (User) พนักงานปฏิบัติการ (Operator) นักวิเคราะห์และออกแบบระบบ(System Analyst) ผู้เขียนโปรแกรมประยุกต์ใช้งาน (Programmer) และผู้บริหารฐานข้อมูล (Database Administrator: DBA)
5. ขั้นตอนการปฏิบัติงาน(Procedure) เป็นขั้นตอนและวิธีการต่างๆ ในการปฏิบัติงานเพื่อการทำงานที่ถูกต้องและเป็นไปตามขั้นตอนที่ได้กำหนดไว้ จึงควรทำเอกสารที่ระบุขั้นตอนการทำงานของหน้าที่ต่างๆ ในระบบฐานข้อมูลทั้งขั้นตอนปกติและขั้นตอนในสถานะที่ระบบเกิดปัญหา (Failure)

2.5.3 โครงสร้างของฐานข้อมูล ประกอบด้วย

- บิต (Bit: Binary Digit) คือ หน่วยของข้อมูลที่เล็กที่สุดที่เก็บอยู่ในหน่วยความจำภายในคอมพิวเตอร์ ซึ่ง Bit จะแทนด้วยตัวเลขหนึ่งตัว คือ 0 หรือ 1 อย่างใดอย่างหนึ่ง เรียกตัวเลข 0 หรือ 1 ว่าเป็น บิต 1 บิต
- Character คือ ตัวอักษรแต่ละตัว / ตัวเลข / เครื่องหมาย
- Field คือ เขตข้อมูล / ชุดข้อมูลที่ให้แทนความหมายของสื่อ โครงสร้างเช่นชื่อของบุคคล ชื่อของวัสดุสิ่งของ
- Record คือ ระเบียบ หรือรายการข้อมูล เช่น ระเบียบของพนักงานแต่ละคน
- Table /File คือ ตาราง หรือแฟ้มข้อมูล ประกอบขึ้นด้วยระเบียบต่างๆ เช่น ตารางข้อมูลของบุคคล ตารางข้อมูลของวัสดุสิ่งของ
- Database คือ ฐานข้อมูล ประกอบด้วยตารางและแฟ้มข้อมูลต่างๆ ที่เกี่ยวข้องหรือ มีความสัมพันธ์กัน

2.6 ภาษา SQL

SQL (ออกเสียงว่า "ess-que-el" หรือ "ซี-เคิลว") ย่อมาจาก Structured Query Language เป็นภาษามาตรฐานที่ใช้ในการติดต่อกับฐานข้อมูล กิดค้นโดยนักวิทยาศาสตร์ของ IBM ในทศวรรษ 1970 ในปัจจุบัน ANSI (American National Standard Institute) ได้ประกาศให้ SQL ภาษามาตรฐานสำหรับระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System ย่อเป็น RDBMS) คำสั่งของ SQL ทำงานต่างของฐานข้อมูล เช่น การปรับปรุงข้อมูล การดึงข้อมูล เพื่อแสดงผลที่ต้องการ

ระบบฐานข้อมูลในปัจจุบันเกือบทั้งหมดเป็นแบบระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS) เช่น Oracle, Sybase, Microsoft SQL Server, Microsoft Access, DB2 ของ IBM เป็นต้น ถึงแม้ว่า ฐานข้อมูลแบบ RDBMS ทุกระบบ จะสามารถใช้คำสั่งพื้นฐานของ SQL เหมือนกัน แต่ทุกระบบอาจมีคำสั่งพิเศษของตัวเองที่ ระบบอื่นอาจจะใช้ไม่ได้

2.5.1 Relational Database

ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) เป็นโครงสร้างฐานข้อมูลที่สามารถกำหนดความสัมพันธ์ขึ้นเมื่อใดก็ได้โดยไม่จำเป็นต้องกำหนดไว้ตั้งแต่เริ่มต้นออกแบบฐานข้อมูล และเปลี่ยนโครงสร้างได้ง่าย ฐานข้อมูลเก็บข้อมูลและสารสนเทศในอ็อบเจกต์ เรียกว่า Table

Table ประกอบด้วย คอลัมน์ (Column) และแถวข้อมูล (Row) โดยคอลัมน์ประกอบด้วย ชื่อคอลัมน์ ประเภทข้อมูล และคุณสมบัติอื่นๆ (เช่น การกำหนดเป็น primary key, ค่าเริ่มต้น) แถวเก็บข้อมูลของคอลัมน์

สามารถเปรียบกับฐานข้อมูล dBase ได้ดังนี้

Relational Database	dBase
Table	ไฟล์
คอลัมน์ (Column)	ฟิลด์
แถวข้อมูล (Row)	เรคคอร์ด

Table ตัวอย่างชื่อ "weather" มีข้อมูลคอลัมน์ชื่อ city, state, high และ low มีแถวข้อมูลใน

Table คือ

city	State	high	Low
Phoenix	Arizona	105	90
Tucson	Arizona	101	92
Flagstaff	Arizona	88	69
San Diego	California	77	60
Albuquerque	New Mexico	80	72

2.6.2 การเลือกข้อมูล (SELECTION statement)

SELECTION statement ใช้ในการสร้างคิวรี หรือการสอบถามกับฐานข้อมูล และเลือกดึงข้อมูลที่ตรงกับเกณฑ์ที่กำหนด

ไวยากรณ์

SELECT column1 [, column2...]

FROM tablename

[WHERE condition];

[] คิวเลือก

column	เป็นคอลัมน์ที่ต้องการให้แสดงผล ถ้าต้องการเลือกทุกคอลัมน์ใน Table ให้ใช้ * เช่น SELECT * FROM weather;
Tablename	เป็นการระบุ Table ที่ต้องการดึงข้อมูล
WHERE	WHERE clause เป็นตัวเลือกซึ่งจะมีหรือไม่ก็ได้ใช้กำหนด หรือเงื่อนไขในการเลือกเฉพาะข้อมูลที่ต้องการ

Condition ที่ใช้ใน WHERE clause คือ

=	เท่ากับ
>	มากกว่า
<	น้อยกว่า
>=	มากกว่าหรือเท่ากับ
<=	น้อยกว่าหรือเท่ากับ
◇	ไม่เท่ากับ
LIKE	(ดูหมายเหตุ 1)

หมายเหตุ

1. LIKE เป็นแบบแผนของ operator การจับคู่ สามารถใช้ใน condition ของ WHERE clause ซึ่ง LIKE เป็น operator ที่มีความสามารถสูงในการในการเลือกข้อมูล เครื่องหมายเปอร์เซ็นต์ "%" สามารถเป็น wild card ในการเลือก โดยการจับคู่ตัวอักษรไม่จำกัดที่อยู่หน้าหรือหลังตัวอักษรที่กำหนด เช่น 'R%' ให้มีอักษร R อยู่ซ้ายสุด หรือหน้า '%R' ให้มีอักษร R อยู่ขวาสุดหรือท้าย '%R%' ให้มีอักษร R อยู่ตรงกลาง

2. SQL statement ต้องลงท้ายประโยคคำสั่งด้วยเครื่องหมาย semi colon (;) เพื่อแสดงการสิ้นสุดคำสั่ง

ตัวอย่าง

```
SELECT first, last, city
FROM empinfo
WHERE first LIKE 'Er%';
```

SQL statement จะจับคู่เฉพาะ first ที่ขึ้นต้นด้วย 'ER' ข้อความหรือตัวอักษร ใน WHERE clause ต้องอยู่ในเครื่องหมาย single quote (')

```
SELECT first, last
FROM empinfo
WHERE last LIKE '%s';
```

SQL statement จะจับคู่เฉพาะ first ที่ลงท้ายด้วย 's'

```
SELECT * FROM empinfo
WHERE first = 'Eric';
```

SQL statement จะเลือกแถวที่ข้อมูลในคอลัมน์ first คือ 'Eric'

ตัวอย่าง Table ชื่อ "empinfo"

2.6.3 การสร้าง Table

CREATE TABLE statement ใช้ในการสร้าง Table ขึ้นมาใหม่

ไวยากรณ์ 1

รูปแบบการสร้าง Table อย่างง่าย

```
CREATE TABLE tablename
```

```
(column1 datatype,
```

```
column2 datatype,
```

```
column3 datatype, ...);
```

ไวยากรณ์ 2

รูปแบบการสร้าง Table ที่มีข้อกำหนด หรือ constraint

```
CREATE TABLE tablename
```

```
(column1 datatype [constraint],
```

```
column2 datatype [constraint],
```

```
column3 datatype [constraint],...);
```

[] ตัวเลือก

tablename	ระบุชื่อ Table
column	ระบุคอลัมน์ที่ต้องการ
datatype	กำหนดประเภทข้อมูลของคอลัมน์
constraint	ข้อกำหนด หรือคุณสมบัติของคอลัมน์

ตัวอย่าง

```
CREATE TABLE employee
```

```
(first varchar(15),
```

```
last varchar(20),
```

```
age number(3),
```

```
address varchar(30),
```

```
city varchar(20),
```

```
state varchar(20));
```

การสร้าง Table ใหม่ให้พิมพ์คำสั่ง CREATE TABLE ตามด้วยชื่อ Table แล้วพิมพ์เครื่องหมายวงเล็บเปิด ตามด้วยชื่อคอลัมน์ แล้วกำหนด ประเภทข้อมูล ถ้าต้องการกำหนด constraint แล้วพิมพ์เครื่องหมายจุลภาค (,) จากนั้นให้พิมพ์คอลัมน์อื่นต่อไป โดยแบ่งแต่ละคอลัมน์

ด้วยจุดภาค จนถึงคอลัมน์สุดท้าย ให้พิมพ์ตามเครื่องหมายวงเล็บปิด ตามด้วยเครื่องหมาย semicolon (;)

2.6.4 การกำหนดคอลัมน์

ชื่อคอลัมน์ สามารถกำหนดเป็นตัวอักษร หมายเลข และเส้นใต้ แต่ไม่เกิน 30 ตัวอักษร และไม่เป็น reserve word ของ SQL (เช่น SELECT, CREATE เป็นต้น)

ประเภทข้อมูล ต้องกำหนดให้เหมาะสมกับข้อมูลที่จะเก็บในคอลัมน์ เช่น first เป็นการเก็บชื่อพนักงาน ควรกำหนดเป็น CHAR หรือ VARCHAR (variable-length character), age เก็บอายุ ควรกำหนดเป็น NUMBER เป็นต้น

ประเภทข้อมูลที่ใช้อยู่

ประเภทข้อมูล	คำอธิบาย
CHAR(size)	ตัวอักษรชนิดความยาวคงที่ size ระบุจำนวนตัวอักษรได้ไม่เกิน 255 ไบต์
VARCHAR(size)	ตัวอักษรชนิดความยาวแปรผันในการเก็บ size ระบุจำนวนตัวอักษร
NUMBER(size)	ตัวเลข size กำหนดจำนวนหลักที่ต้องการ
DATE	วันที่
NUMBER(size,d)	ตัวเลขชนิดมีทศนิยม size กำหนดจำนวนหลักทั้งหมดรวมถึงทศนิยม d กำหนดตำแหน่งทศนิยม

Constraint

constraint เป็นตัวเลือกในการกำหนดคอลัมน์ โดย constraint เป็นกฎหรือคุณสมบัติ เช่น กำหนดเป็น primary key หรือควบคุมค่าที่ป้อนเข้ามา เช่น not null ใช้เป็นการระบุว่าคอลัมน์ต้องมีค่าเสมอ ห้ามเป็นช่องว่าง unique ใช้เป็นการบังคับว่าค่าของคอลัมน์ต้องเป็นค่าแบบไม่ซ้ำ check ใช้ในการตรวจสอบค่าก่อนนำเข้า

การกำหนด constraint สำหรับ primary key และ foreign key

2.6.5 การกำหนด Key

Key เป็นข้อกำหนดคุณสมบัติของคอลัมน์ใน Table ซึ่งทำหน้าที่เป็น index เพื่อเพิ่มความเร็วในการค้นหาข้อมูล และใช้กำหนดความสัมพันธ์ระหว่าง Table

Primary key สร้างจากคอลัมน์ (หรือประกอบด้วยหลายคอลัมน์) ที่มีค่าของคอลัมน์ในทุกแถวข้อมูลมีค่าไม่ซ้ำกัน เป็นการควบคุมค่าที่เก็บในคอลัมน์ และใช้ในการเชื่อมโยงกับ Table อื่น ในแต่ละ Table กำหนด primary key ได้ 1 key

Foreign key เป็นคอลัมน์ใน Table ที่ primary key อยู่ที่ Table อื่น หมายถึงว่า ค่าของคอลัมน์ทั้งหมดใน Table ต้องมีค่าสอดคล้องกับค่าของคอลัมน์ที่เป็น primary key ของ Table ที่สัมพันธ์กัน

Index สามารถกำหนดขึ้นเพื่อใช้ในการเพิ่มความเร็วในการค้นหาข้อมูล การเรียงลำดับข้อมูล แต่ควรสร้างความจำเป็น เนื่องจากถ้ามีคอลัมน์ที่เป็น Index มากจะทำให้การทำงานช้าลง

การกำหนด primary key

ตัวอย่าง

```
CREATE TABLE employee
(employee_id (10),
first varchar(15),
last varchar(20),
age number(3),
address varchar(30),
city varchar(20),
state varchar(20),
CONSTRAINT employee_primary_key PRIMARY KEY (employee_id));
```

หรือ

```
CREATE TABLE employee
(employee_id (10) NOT NULL
CONSTRAINT employee_primary_key PRIMARY KEY (employee_id),
first varchar(15),
last varchar(20),
age number(3),
address varchar(30),
```

```
city varchar(20),
state varchar(20));
```

การกำหนด foreign key

ตัวอย่าง

```
CREATE TABLE employee
```

```
(employee_id (10),
```

```
first varchar(15),
```

```
last varchar(20),
```

```
age number(3),
```

```
address varchar(30),
```

```
city varchar(20),
```

```
state varchar(20),
```

```
CONSTRAINT employee_foreign_key PRIMARY KEY (state));
```

การกำหนด index

ไวยากรณ์

```
CREATE INDEX indexname
```

```
ON tablename (column);
```

indexname	กำหนดชื่อ index
tablename	ที่ Table ที่ต้องการกำหนดสร้าง index
column	ชื่อคอลัมน์ที่กำหนดเป็น index

ตัวอย่าง

```
CREATE INDEX Index_city
```

```
ON employee (city);
```

2.6.6 การแก้ไข Table

ALTER statement ใช้ในการปรับปรุงคอลัมน์ใน Table ที่สร้างเสร็จแล้ว

ไวยากรณ์ 1

การปรับปรุงคอลัมน์

```
ALTER TABLE tablename
ADD (column1 datatype [constraint]
[,column2 datatype [constraint],
column3 datatype [constraint],... ] );
[ ] ตัวเลือก
```

ไวยากรณ์ 2

การปรับปรุง constraint ของ Table

```
ALTER TABLE tablename
ADD (tableconstraint );
```

tablename	ระบุชื่อ Table
column	ระบุคอลัมน์ที่ต้องการ
datatype	กำหนดประเภทข้อมูลของคอลัมน์
constraint	ข้อกำหนด หรือคุณสมบัติของคอลัมน์
tableconstraint	กำหนด constraint ของ Table เช่น กำหนด primary key หรือ foreign key

ตัวอย่าง การปรับปรุง column

```
ALTER TABLE employee
ADD(employee_id(10),
first varchar(20),
edu varchar(20),
```

ตัวอย่าง การปรับปรุง primary key

```
ALTER TABLE employee
ADD (CONSTRAINT employee_primary_key1 PRIMARY KEY (employee_id);
```

การปรับปรุง Table ใหม่ให้พิมพ์คำสั่ง CREATE TABLE ตามด้วยชื่อ Table ตามด้วย ADD แล้วพิมพ์เครื่องหมายวงเล็บเปิด ตามด้วยชื่อคอลัมน์ แล้วกำหนด ประเภทข้อมูล ถ้าต้องการ กำหนด constraint แล้วพิมพ์เครื่องหมายจุลภาค (,) จากนั้นให้พิมพ์คอลัมน์อื่นต่อไป โดยแบ่งแต่ละคอลัมน์ด้วยจุลภาค จนถึงคอลัมน์สุดท้าย ให้พิมพ์ตามเครื่องหมายวงเล็บปิด ตามด้วย เครื่องหมาย semicolon (;) ถ้าเป็นการปรับปรุง constraint ของ Table ให้ constraint ของ Table ใน วงเล็บ และแก้ไขให้พิมพ์ชื่อคอลัมน์เดิมและการปรับปรุง ข้อกำหนดจะเข้าไปแทนที่ข้อกำหนดเดิม

2.6.7 การป้อนข้อมูล

INSERT statement ใช้ในการป้อนหรือเพิ่มแถวข้อมูลเข้า Table

ไวยากรณ์

```
INSERT INTO tablename
(column1,column2,...)
VALUES(value1,value2,...);
```

tablename	ระบุชื่อ Table
column	ระบุคอลัมน์ที่ต้องการ เพิ่มค่า
value	ค่าที่ต้องการป้อน

หมายเหตุ แต่ละค่าของ value กับ column จะต้องตรงกัน

ตัวอย่าง

```
INSERT INTO employee
(first, last, age, address, city, state)
VALUES ('Luke', 'Duke', 45, '2130 Boars Nest', 'Hazard Co', 'Georgia');
NOTE: ข้อมูลที่เป็นข้อความ (string) ต้องอยู่ในเครื่องหมาย single quote ('')
```

การป้อนข้อมูลเข้าสู่ Table ให้พิมพ์คำสั่ง INSERT INTO ตามด้วยชื่อ Table แล้วพิมพ์ เครื่องหมายวงเล็บเปิด ตามด้วยชื่อคอลัมน์ แล้วพิมพ์เครื่องหมายจุลภาค (,) จากนั้นให้พิมพ์คอลัมน์ อื่นต่อไป โดยแบ่งแต่ละคอลัมน์ด้วยจุลภาค จนถึงคอลัมน์สุดท้ายที่กำหนด ให้พิมพ์เครื่องหมาย วงเล็บปิด ให้พิมพ์คำสั่ง VALUES แล้วพิมพ์เครื่องหมายวงเล็บเปิด ตามด้วยค่าที่ต้องป้อนเข้าในแต่ละ คอลัมน์ แล้วพิมพ์เครื่องหมายจุลภาค (,) จากนั้นให้พิมพ์ค่าอื่นต่อไป โดยแบ่งแต่ละค่าด้วย จุลภาค จนถึงค่าสุดท้าย ให้พิมพ์ตามเครื่องหมายวงเล็บปิด ตามด้วยเครื่องหมาย semicolon (;)

การป้อนข้อมูลจาก table อื่น

ตัวอย่าง ตามตัวอย่างเป็นการป้อนข้อมูลเข้าสู่ table "employee" ด้วยข้อมูลจาก table "temp_emp"

```
INSERT INTO employee
(empid, first, last, age, dept)
SELECT empno, firstname, lastname, age, department
FROM temp_emp
WHERE department IN ('Sales','Account','IT');
```

การป้อนข้อมูลที่เป็น Date

ตัวอย่าง การป้อนค่าคอลัมน์ hiredate ด้วย '25-AUG-99'

```
INSERT INTO empid, first, last, hiredate, dept)
VALUE (1150, 'Luke', 'Duke', '25-AUG-99', 'Sales')
```

ตัวอย่าง ในฐานข้อมูล Oracle สามารถป้อนค่าด้วยฟังก์ชัน TO_DATE

```
INSERT INTO empid, first, last, hiredate, dept)
VALUE (1150, 'Luke', 'Duke', TO_CHAR('25/10/1999','dd/mm/yyyy'), 'Sales');
```

2.6.8 การปรับปรุงข้อมูล

UPDATE statement ใช้ในการปรับค่าหรือเปลี่ยนข้อมูลตามเกณฑ์ที่กำหนด

ไวยากรณ์

```
UPDATE tablename
SET columnname1 = newvalue1 [, "columnname2" = "newvalue2" ...]
WHERE column1 OPERATOR value [AND|OR column2 OPERATOR value
];
```

[] คิวเลือก

tablename	ระบุชื่อ Table
columnname	ระบุคอลัมน์ที่ต้องการแก้ไขค่า
newvalue	ค่าใหม่ที่ต้องการแก้ไข
columnname	ชื่อคอลัมน์ที่กำหนดเงื่อนไข
value	ค่าของเงื่อนไข

ตัวอย่าง

```
UPDATE phone_book
SET area_code = 623
WHERE prefix = 979;
```

```
UPDATE phone_book
SET last_name = 'Smith', prefix=555, suffix=9292
WHERE last_name = 'Jones';
```

```
UPDATE employee
SET age = age+1
WHERE first_name='Mary' AND last_name='Williams';
```

การปรับปรุงข้อมูล ให้พิมพ์คำสั่ง UPDATE ตามด้วยชื่อ Table แล้วตามด้วยคำสั่ง SET ชื่อคอลัมน์ที่ต้องการเปลี่ยนค่า ตามเครื่องหมายเท่ากับ ค่าใหม่ที่นำแทนที่ ถ้าปรับปรุงหลายคอลัมน์ ให้ชื่อคอลัมน์ เท่ากับ ค่าใหม่ แล้วแยกด้วยเครื่องหมายจุลภาค (ยกเว้นค่าชุดสุดท้าย) แล้วตามด้วย WHERE แล้วตามด้วยเงื่อนไข ถ้ามีหลายเงื่อนไขให้เชื่อมด้วย AND หรือ OR แล้วปิดท้ายคำสั่งด้วยเครื่องหมาย semicolon (;)

2.6.9 การลบข้อมูล

DELETE statement เป็นคำสั่งที่ใช้การลบข้อมูล

ไวยากรณ์

```
DELETE FROM tablename
```

```
WHERE column OPERATOR value [AND|OR column OPERATOR value];
```

[] ตัวเลือก

tablename	ระบุชื่อ Table
columnname	ชื่อคอลัมน์ที่กำหนดเงื่อนไข
value	ค่าของเงื่อนไข

ตัวอย่าง

```
DELETE FROM employee;
```

NOTE: ถ้าใช้คำสั่ง DELETE statement โดยไม่มี WHERE clause จะเป็นการลบข้อมูลทั้งหมดใน Table

```
DELETE FROM employee
```

```
WHERE lastname = 'May';
```

```
DELETE FROM employee
```

```
WHERE firstname = 'Mike' or firstname = 'Eric';
```

การลบเป็นการลบแถวข้อมูลออกจาก Table ให้ป้อนคำสั่ง DELETE FROM ตามด้วย ชื่อ Table ตามด้วย WHERE และเงื่อนไขในการลบไป ถ้ามีหลายเงื่อนไขให้เชื่อมด้วย AND หรือ OR แล้วปิดท้ายคำสั่งด้วยเครื่องหมาย semicolon (;)

2.6.10 การลบ Table

DROP statement ที่ใช้ในการลบ Table หรือยกเลิก

ไวยากรณ์

```
DROP TABLE tablename ;
```

ตัวอย่าง

```
DROP TABLE employee;
```

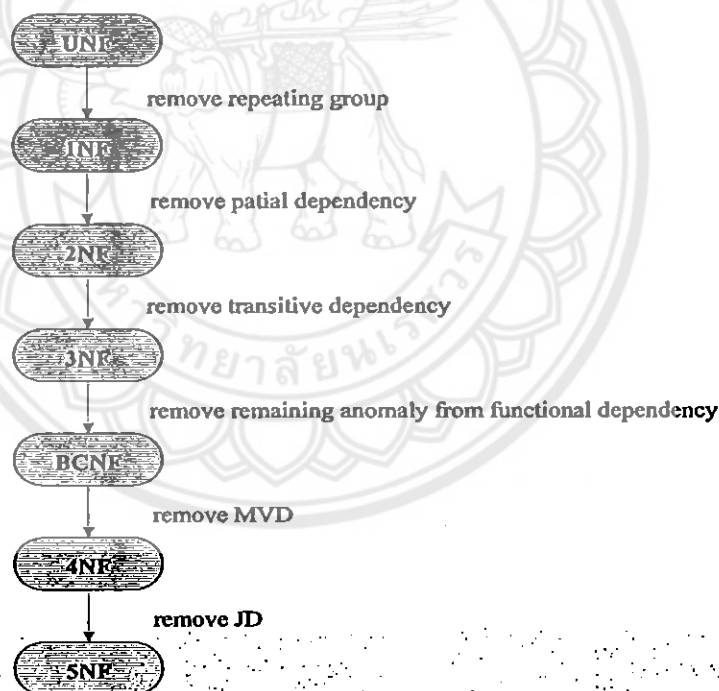
การลบ Table ให้พิมพ์คำสั่ง DROP TABLE ตามชื่อ Table ที่ต้องการลบแล้วปิดท้ายคำสั่งด้วยเครื่องหมาย semicolon (;)

2.7 วิธีการทำนอร์มัลไลซ์

จากปัญหา Update Anomalies (ทั้ง Insertion, Modification และ Deletion) ที่เกิดขึ้นเพราะความซ้ำซ้อนกันของข้อมูลและตารางเก็บข้อมูลที่มีโครงสร้างไม่ดีพอ ทำให้มีการค้นหาวิธีที่จะทำให้ทราบก่อนว่าตารางที่มีอยู่นั้นจะทำให้เกิดปัญหา Update Anomalies หรือไม่เมื่อนำไปใช้เก็บข้อมูลแล้ว รวมทั้งยังใช้สำหรับตรวจสอบด้วยว่าเป็นตารางที่มีโครงสร้างดีเพียงพอที่จะไม่ทำให้เกิดปัญหาอื่นๆตามมาแล้วหรือไม่ ซึ่งวิธีที่ใช้ทดสอบตารางก็คือการทำนอร์มัลไลซ์

นอร์มัลไลซ์ (Normalization) จึงเป็นวิธีที่ใช้ในการปรับโครงสร้างของตารางเพื่อให้ได้ตารางที่สามารถเก็บข้อมูลได้โดยไม่มีปัญหาใดๆตามมาภายหลังซึ่งการทำนอร์มัลไลซ์นั้นถูกแบ่งออกเป็นหลายขั้นตอนโดยเรียกแต่ละขั้นตอนว่าระดับ 1, 2, 3, ...

เมื่อตารางผ่านการทำนอร์มัลไลซ์แต่ละระดับแล้วตารางนั้นก็จะเป็นอยู่ในรูปแบบที่เรียกว่า Normal Form (NF) อาทิถ้าเป็นตารางที่ผ่านการทำนอร์มัลไลซ์ระดับที่ 1 แล้วก็หมายถึงตารางที่เป็น 1NF หรือ Normal Form ระดับที่ 1



รูปที่ 2.5 แสดงลำดับขั้นการทำนอร์มัลไลซ์

การทำนอร์มัลไลซ์สามารถทำได้ถึงระดับที่ 5 หรือ 5NF โดยยิ่งตารางผ่านการทำนอร์มัลไลซ์สูงเท่าใดก็จะลดปัญหาที่เกิดขึ้นจากการเก็บและใช้ข้อมูลได้มากขึ้นเท่านั้น

อย่างไรก็ตามการทำนอร์มัลไลซ์นั้นมีข้อเสียอยู่เหมือนกัน เนื่องจากการทำนอร์มัลไลซ์แต่ละระดับจะทำให้ตารางเพิ่มมากขึ้น ซึ่งมีผลกระทบต่อการคิวรีข้อมูลมาใช้งานภายหลังที่อาจจะต้อง

เรียกใช้ตารางจำนวนหนึ่งแทนที่จะอ่านจากตารางที่ไม่ผ่านการทำนอร์มัลไลซ์เพียงตารางเดียว ดังนั้นการทำนอร์มัลไลซ์ต้องพิจารณาปัจจัยเหล่านี้ให้ถี่ถ้วน

2.8 ภาษาซีชาร์ป[3]

ภาษาซีชาร์ป (C# Programming Language) เป็นภาษาโปรแกรมเชิงวัตถุทำงานบนคอตเน็ตเฟรมเวิร์ก พัฒนาโดยบริษัทไมโครซอฟท์และมี Anders Hejlsberg เป็นหัวหน้าโครงการ โดยมีรากฐานมาจากภาษาซีพลัสพลัสและภาษาอื่นๆ (โดยเฉพาะภาษาเคลไพและจาวา) โดยปัจจุบันภาษาซีชาร์ปเป็นภาษามาตรฐานรองรับโดย ECMA และ ISO

ตัวอย่างต่อไปนี้ ก็คือตัวอย่างโปรแกรม Hello world ใน C#:

```
public class ExampleClass
{
    public static void Main ()
    {
        System.Console.WriteLine ("Hello, world!");
    }
}
```

ผลของการทำงานคือการเขียนคำว่า *Hello, world!* บนเครื่อง (Console) ที่ใช้งาน. ในแต่ละบรรทัดมีความหมายดังนี้:

```
public class ExampleClass
```

บรรทัดนี้คือการประกาศ Class, *public* หมายถึงวัตถุที่สร้างในโครงการ (Project) อื่นๆ สามารถเข้าใช้งาน Class นี้ได้ ไม่จำกัด. ข้อมูลต่างๆ ที่อยู่ผ่านได้หน่วยของบรรทัดนี้ จะใช้ในการทำงานของ Class นี้

```
public static void Main ()
```

บรรทัดนี้เป็นจุดที่ใช้ในการเริ่มการทำงานของโปรแกรม เมื่อโปรแกรมทำงาน โดยสามารถเรียกใช้จากโปรแกรมอื่นได้โดยการใช้ไวยากรณ์ `ExampleClass.Main ()`. (`public static void` เป็นส่วนที่ใช้ในการทำงาน ซึ่งต้องเรียนรู้ในการเขียนขั้นสูง)

```
System.Console.WriteLine ("Hello, world!");
```

ในบรรทัดนี้ เป็นการทำงาน เพื่อแสดงผลออกมา *Console* คือโปรแกรมระบบ, ซึ่งก็คือโปรแกรมระบบแบบสั่งคำสั่งที่ละบรรทัด (เช่น DOS) ที่สามารถรับข้อมูลและแสดงผลเป็นข้อความได้. จากที่เราเขียนโปรแกรมจะทำการเรียก *Console*. โดยใช้คำสั่ง *WriteLine*, ซึ่งทำให้สามารถส่งค่าข้อความออกมาแสดงผลทางโปรแกรมระบบได้

ไมโครซอฟท์ส่งมาตรฐานภาษาซีชาร์ปให้กับ ECMA และได้รับการยอมรับเป็นมาตรฐาน ECMA ในเดือนธันวาคม ค.ศ. 2001 ในชื่อว่า *ECMA-334 C# Language Specification* ใน ค.ศ. 2003 ภาษาซีชาร์ปได้รับการยอมรับเป็นมาตรฐาน ISO (ISO/IEC 23270).

มาตรฐาน ISO/IEC 23270:2003 ระบุรูปแบบ และกำหนดการแปล (ตีความ) โปรแกรมที่เขียนด้วยภาษาซีชาร์ป. โดยตัวมาตรฐานได้ระบุ:

- รูปแบบการนำเสนอ (the representation of C# programs)
- ไวยากรณ์ (the syntax and constraints of the C# language)
- กฎการตีความสำหรับแปลโปรแกรมภาษาซีชาร์ป (the semantic rules for interpreting C# programs)
- ข้อห้าม และข้อจำกัด ของเครื่องมือที่สร้างตามข้อกำหนดของซีชาร์ป (the restrictions and limits imposed by a conforming implementation of C#)
- กลไกในการแปลงโปรแกรมภาษาซีชาร์ป เพื่อใช้ในระบบประมวลผลข้อมูล (the mechanism by which C# programs are transformed for use by a data-processing system)
- กลไกในการเรียกให้โปรแกรมภาษาซีชาร์ปทำงาน เพื่อใช้ในระบบประมวลผลข้อมูล (the mechanism by which C# applications are invoked for use by a data-processing system)
- กลไกในการแปลงข้อมูลเข้า เพื่อใช้กับโปรแกรมภาษาซีชาร์ป (the mechanism by which input data are transformed for use by a C# application)
- กลไกในการแปลงข้อมูลออก หลังจากถูกประมวลผล โดยโปรแกรมภาษาซีชาร์ป (the mechanism by which output data are transformed after being produced by a C# application)
- the size or complexity of a program and its data that will exceed the capacity of any specific data-processing system or the capacity of a particular processor;
- all minimal requirements of a data-processing system that is capable of supporting a conforming implementation.

นอกจากนี้ตัวมาตรฐานไม่ได้กล่าวถึง โครงสร้างข้อมูล และคิวไลบรารีกลางของ .NET Framework ซึ่งเป็นหัวใจสำคัญของการเขียนโปรแกรมด้วยภาษาซีชาร์ป

2.9 เอเอสพีดอตเน็ต (ASP.NET) [3]

เอเอสพีดอตเน็ต (ASP.NET) คือเทคโนโลยีสำหรับพัฒนาเว็บไซต์ เว็บแอปพลิเคชัน และเว็บเซอร์วิส ซึ่งเป็นส่วนหนึ่งของดอตเน็ตเฟรมเวิร์ก พัฒนาโดยไมโครซอฟท์

ASP.NET เป็นรุ่นถัดจาก Active Server Pages (ASP) แม้ว่า ASP.NET นั้นจะใช้ชื่อเดิมจาก ASP แต่ทั้งสองเทคโนโลยีนั้นแตกต่างกันอย่างสิ้นเชิง โดยไมโครซอฟท์นั้นได้สร้าง ASP.NET ขึ้นมาใหม่หมดบนฐานจาก Common Language Runtime (CLR) ซึ่งทำให้ผู้พัฒนาสามารถเลือกใช้ภาษาใดก็ได้ที่รองรับโดยดอตเน็ตเฟรมเวิร์กเช่น C# และ VB.NET เป็นต้น ปัจจุบันรุ่นล่าสุดคือ ASP.NET 4.0 ซึ่งรวมอยู่ใน .NET Framework 4.0

2.9.1 รูปแบบไฟล์ ASPX

ASPX เป็นชื่อรูปแบบไฟล์ของหน้าแบบฟอร์ม ASP.NET โดยทั่วไปแล้วในไฟล์จะมีรหัสแบบ HTML หรือ XHTML ซึ่งใช้กำกับรูปแบบฟอร์ม หรือ เนื้อหาในหน้าเว็บ และในส่วนของโค้ดนั้น อาจจะอยู่ในหน้าเดียวกันในแท็ก หรือ บล็อก `<% -- รหัสที่ใช้ -- %>` (โดยในกรณีนี้จะคล้ายกับเทคโนโลยีที่ใช้พัฒนาเว็บ อย่าง PHP และ JSP) หรือแยกอยู่ในหน้าโค้ดออกมาต่างหาก (Code behind) ASP.NET รองรับการเขียน โค้ดในบรรทัดเดียวกันทั้งหมดในไฟล์ ASPX แต่วิธีนี้นั้นเป็นวิธีที่ไม่แนะนำ

ไฟล์รูปแบบอื่นๆ ที่เกี่ยวข้องกับ ASP.NET มีดังนี้:

- ASCX - control ที่พัฒนาขึ้นมาเอง
- ASMX - หน้าเว็บเซอร์วิส
- ASHX - ตัวจัดการ HTTP ด้วยตนเอง

2.9.2 คุณสมบัติของ ASP.NET

1. ใช้ ภาษาใดๆในการเขียน Script ก็ได้ ใน ASP เราสามารถใช้ได้เฉพาะภาษาที่เป็น Script อย่าง VBScript และ Jscript แต่ใน ASP.NET เราจะได้ใช้ภาษาที่เป็นรูปแบบของภาษาเต็มๆซึ่งในปัจจุบันมี 3 ภาษา คือ C# VB.NET และ Jscript.NET ที่ออกมาเป็นพื้นฐาน แต่ในอนาคต Microsoft มีแผนที่จะเพิ่มภาษาให้ครบถ้วนทุกภาษา

2. มีความยืดหยุ่นในการเขียนโปรแกรมสูงขึ้น เราสามารถใช้ภาษาในการเขียน ASP.NET ได้มากกว่า 1 ภาษา ภายใน File เดียวกัน ทำให้สามารถเลือกภาษาที่ง่ายที่สุดในการเขียนในแต่ละส่วนได้ เช่น การจวนลูปของ VB นั้นง่ายกว่า C# แต่การใช้งาน Function C# ง่ายกว่า (ขึ้นอยู่กับมุมมอง

15705642

๒/๕

๘๔ ๘๖๑๕

๒๕๕๓

ของแต่ละคนด้วย) ก็แยกเขียนในส่วนของกรวนลูด้วย VB และการเขียนในส่วนของ Function ด้วย C# เป็นต้น แต่มีเงื่อนไขคือต้องกำหนดให้ชัดเจนว่าส่วนไหนใช้ภาษาอะไรเขียน

3. ลักษณะการแปลภาษาและนามสกุล File ที่เปลี่ยนไป ลักษณะการแปลภาษาใน ASP รุ่นเก่าๆนั้นเป็นแบบ Interpreter คือแปลไปทำงานไป แต่สำหรับ ASP.NET นี้เป็นแบบ Compiler คือแปลคำสั่งให้เรียบร้อยก่อนแล้วค่อยทำงาน (แต่ก็แปลในลักษณะ JUST-IN-TIME หรือ JIT คือต้องการใช้ส่วนไหนก็ค่อยแปล เช่นถ้า Function ใดยังไม่ถูกเรียกใช้งาน ก็จะไม่มีการแปล Function นั้น วิธีนี้โปรแกรมจะทำงานเร็วขึ้น) นอกจากนี้นามสกุลของ File ก็มีการเปลี่ยนแปลงจาก .asp มาเป็น .aspx

4. รูปแบบและการใช้งาน Compiler ที่ง่ายขึ้น รูปแบบ ของคอมไพเนนต์ จะเน้นไปที่ XML มากที่สุด และที่สำคัญคือการ ใช้งาน คอมไพเนนต์ ใน ASP.NET นั้นเราสามารถ Upload File ไปไว้ใน Directory ที่ผู้ดูแล Server หรือเรียกสั้นว่า Admin กำหนดให้หลังจากนั้น คอมไพเนนต์ จะมีการติดตั้งตัวเองโดยอัตโนมัติ ลดปัญหาที่มีใน ASP Version เก่าๆ ได้เป็นอย่างดี เนื่องจากใน ASP Version เก่าๆนั้นการติดตั้งคอมไพเนนต์ จะกระทำโดยผู้ดูแล Server เท่านั้น ทำให้เราประสบความลำบากหากต้องการใช้งาน คอมไพเนนต์ ที่ Server นั้น ไม่ได้ติดตั้งเอาไว้ให้ใช้

5. มี Library ให้เลือกใช้งานมากขึ้น ใน ASP Version เก่าๆนั้น Application บางอย่างสร้างได้ไม่สะดวกนัก ต้องอาศัยคอมไพเนนต์ต่างๆมาเพิ่มเติม แต่ใน ASP.NET ได้เพิ่ม Library ในส่วนเหล่านี้ให้กลายเป็นพื้นฐานของการใช้งาน เช่น Library ที่เกี่ยวกับการส่ง mail upload เป็นต้น ทำให้เราสามารถสร้าง Application ได้หลายหลากยิ่งขึ้น

6. มี Control ทำให้การใช้งานบางอย่างง่ายขึ้น เป็นส่วนพิเศษที่เพิ่มเติมขึ้นมา ส่วนที่เรียกว่า Control นี้จะช่วยให้เราสามารถสร้าง website ได้อย่างง่ายและมีประสิทธิภาพมากขึ้น จึงไม่ต้องกังวลว่า Browser รุ่นนั้นรุ่นนี้จะรับภาษาที่เราเขียนขึ้นมาหรือไม่

7. สามารถเรียกข้อมูลจาก Server ได้ แต่เดิมนั้น Server เรียกข้อมูลได้จากเครื่องผู้ใช้เท่านั้น แต่ใน ASP.NET เครื่อง Server สามารถเรียกข้อมูลจากเครื่องServer ด้วยกันเองได้ เช่น เราอาจเขียน โปรแกรม ส่งให้ดึงข้อมูลของตลาดหุ้นในเวปไซด์ www.yahoo.com มาแสดงบนเวปเพจของเราเองได้ เป็นต้น

8. ไม่ขึ้นกับ Hardware เนื่องจากเป็นระบบใน .NET Framework ดังนั้นจึงมีคุณสมบัติของ Common Language Runtime (CLR) ทำให้มีการ Compile โปรแกรม เป็น ภาษามาตรฐาน ที่เรียกว่า ILก่อน ดังนั้นไม่ว่าคุณจะใช้เครื่อง Palm Notebook PDA หรือเล่น WAP ผ่านทางโทรศัพท์มือถือ ก็ไม่เกิดปัญหาขึ้น ซึ่งในอนาคตจะให้รองรับอุปกรณ์ได้มากขึ้น

9. ง่ายต่อการหาจุดผิดพลาดในโปรแกรม เมื่อก่อนเวลาเกิด Error เครื่องจะบอกแค่ว่ามีผิดพลาดชนิดใด และ ผิดที่บรรทัด ใด แต่ใน ASP.NET นี้จะมีการแสดงรายละเอียดที่มากขึ้น

พร้อมทั้งแนวทางแก้ไข นอกจากนี้ยังสามารถแสดงว่าตัวแปรต่างๆ ในสภาวะนั้นมีค่าเป็นเท่าไร ซึ่งจะขึ้นประโยชน์ต่อการหาจุดผิดพลาดต่างๆ

10. สามารถเขียนโปรแกรมกำกับเหตุการณ์ต่างๆ ใน Web page ได้ เราสามารถเขียนโปรแกรมเพื่อกำกับเหตุการณ์หรืออีเวนต์ ต่างๆ ได้ ตั้งแต่ โหลดหน้า Web page ซึ่งใน Asp เดิมเราต้องเขียนด้วย Client – Side Script เท่านั้น

11. แยกส่วนที่เป็น Html กับส่วนของ Script ออกจากกันอย่างชัดเจน ส่วนที่เป็น Html กับ Script ASP จะรวมอยู่ด้วยกัน แต่ ASP.NET นี้จะมีการแยกอย่างชัดเจนว่าส่วนไหนเป็น Html และส่วนไหนเป็น Script

2.10 การใช้งาน Class String [4]

คลาส String ใช้ช่วยในการจัดการกับข้อความ ซึ่งหากมีการประกาศตัวแปรชนิด String ก็จะสามารถใช้ชื่อประกอบต่างๆ ของคลาส String ได้

2.10.1 การเปรียบเทียบ String

การเปรียบเทียบ String คือการตรวจสอบว่า String ทั้งสองตัวเป็น Object ที่เก็บค่าข้อมูลเดียวกันหรือไม่ มีการใช้คำสั่งในการเปรียบเทียบ ดังนี้

2.10.1.1 รูปแบบคำสั่งที่ 1 การใช้เครื่องหมายเท่ากับ (==)

```
string1 == string2
```

เมื่อ string1 และ string2 คือ ชื่อตัวแปร Object แบบ String

ตัวอย่างคำสั่งเช่น

```
String studentWord1 = "HELLO";
```

```
String studentWord2 = "Hello";
```

```
if(studentWord1 == studentWord2)
```

```
System.out.println("String ทั้งสองมีค่าเท่ากัน");
```

```
else
```

```
System.out.println("String ทั้งสองมีค่า ไม่เท่ากัน");
```

คำอธิบาย

ผลลัพธ์จากตัวอย่างข้างบน จะทำการแสดงข้อความว่า "String ทั้งสองมีค่าไม่เท่ากัน"

ออกทางจอภาพ เพราะว่าตัวแปร String ทั้งสองเก็บค่าข้อมูลไม่เหมือนกัน

2.10.1.2 รูปแบบคำสั่งที่ 2 การใช้คำสั่ง equals

```
string1.equals(string2)
```

ตัวอย่างคำสั่งเช่น

```
String studentWord1 = "HELLO";
String studentWord2 = "HELLO";
if(studentWord1.equals( studentWord2))
    System.out.println("String ทั้งสองมีค่าเท่ากัน");
else
    System.out.println("String ทั้งสองมีค่าไม่เท่ากัน");
```

คำอธิบาย

ผลลัพธ์จากตัวอย่างข้างบน จะทำการแสดงข้อความว่า "String ทั้งสองมีค่าเท่ากัน" ออกทางจอภาพ เพราะว่าตัวแปร String ทั้งสองเก็บค่าข้อมูลเหมือนกัน

2.10.1.3 รูปแบบคำสั่งที่ 3 การใช้คำสั่ง compareTo

```
string1.compareTo(string2)
```

การใช้คำสั่งนี้จะทำการเปรียบเทียบตัวอักษรใน sting1 เทียบกับ sting2 ทีละตัวในตำแหน่งเดียวกัน ไปเรื่อยๆ ซึ่งถ้ามีข้อมูลเดียวกันจำนวนเท่ากันก็จะ return ค่าเป็น 0 แต่ถ้ามีตัวใดตัวหนึ่ง หรือมีขนาดข้อมูลต่างกันก็จะ return เป็นผลต่างของรหัส Ascii ของข้อมูลใน String ทั้งสอง ซึ่งมีผลลัพธ์ ดังนี้

เป็นลบ (-) ถ้า String1 มีค่าน้อยกว่า String2

เป็นบวก (+) ถ้า String1 มีค่ามากกว่า String2

เป็นศูนย์ (0) ถ้า String1 มีค่าเท่ากับ String2

ตัวอย่างคำสั่งเช่น

```
String studentWord1 = "HELLO";
String studentWord2 = "Hello";
int check = studentWord1. compareTo( studentWord2);
if(check == 0)
    System.out.println("String ทั้งสองมีค่าเท่ากัน");
else
    System.out.println("String ทั้งสองมีค่าไม่เท่ากัน");
```

คำอธิบาย

ผลลัพธ์จากตัวอย่างข้างบน จะทำการแสดงข้อความว่า “String ทั้งสองมีไม่ค่าเท่ากัน” ออกทางจอภาพ เพราะคำสั่ง `compareTo` จะเปรียบเทียบว่าตัวแปร `String` ทั้งสองเก็บค่าข้อมูลเหมือนกันหรือไม่ ซึ่งมีค่าตัวอักษรต่างกันโดย `studentWord1` เก็บตัวอักษรพิมพ์ใหญ่ (มีรหัส Ascii น้อยกว่า `studentWord2`) ทำให้ได้ผลลัพธ์เป็นค่าลบ (ตัวแปร `check` มีค่าเท่ากับ -32)

2.10.2 การดึงข้อความบางส่วนจาก String

เป็นการดึงข้อความ หรือตัดคำใน `String` โดยการบอกตำแหน่งเริ่มต้นและสุดท้ายของข้อความที่ต้องการตัด

รูปแบบคำสั่ง

```
String string2 = string1.substring(n, m);
```

`n` คือตัวเลขที่บอกตำแหน่งเริ่มต้นใน `string1` ของข้อความที่ต้องการ

`m` คือตัวเลขที่บอกตำแหน่งสุดท้ายใน `string1` ของข้อความที่ต้องการ

ตัวอย่างคำสั่งเช่น

```
String studentWord1 = "WELCOME to Java";
```

```
String myString = studentWord1.substring(0,7);
```

```
System.out.println("My string is " + myString);
```

ผลลัพธ์ทางจอภาพ

```
My string is WELCOME
```

การดึงตัวอักษรใน String

เป็นการใช้คำสั่ง `charAt()` เพื่อดึงเอาตัวอักษรที่อยู่ใน `String` โดยการระบุตำแหน่งที่ต้องการ ผลลัพธ์ที่ได้จะ `return` กำเป็นตัวอักษรในตำแหน่งนั้นใน `String`

รูปแบบคำสั่ง

```
char charName = string1.charAt( n );
```

`n` คือ เลขจำนวนเต็ม บอกตำแหน่งของอักษรใน `string1`

ตัวอย่างคำสั่งเช่น

```
String studentWord1 = "WELCOME";
```

```
char letter = studentWord1.charAt(3);
```

```
System.out.println("Character in studentWord1 is " + letter);
```

ผลลัพธ์ทางจอภาพ

```
Character in studentWord1 is C
```

2.10.3 การตัดช่องว่างใน String

ใช้คำสั่ง trim() เป็นตัดช่องว่างหน้าและหลังของข้อความที่อยู่ใน String
รูปแบบคำสั่ง

```
String string2 = string1.trim();
```

ตัวอย่างคำสั่งเช่น

```
String studentWord1 = "    My name is Somsri    ";
String myString = studentWord1.trim();
System.out.println("My string is " + myString);
```

ผลลัพธ์ทางจอภาพ

```
My string is My name is Somsri
```

2.10.4 การค้นหาใน String

ในการค้นหาข้อความใน String จะใช้ Method 2 Method ดังนี้

indexOf() ใช้ในการค้นหาข้อความที่พบเป็นชุดแรก

lastIndexOf() ใช้ในการค้นหาข้อความที่พบเป็นชุดสุดท้าย

รูปแบบคำสั่ง

```
int pos = string1.indexOf( text ); หรือ
```

```
int pos = string1.indexOf( text , n);
```

```
int pos = string1.lastIndexOf( text ); หรือ
```

```
int pos = string1.lastIndexOf( text , n );
```

pos คือ เลขจำนวนเต็มแสดงตำแหน่งที่ค้นพบ

n คือ เลขจำนวนเต็ม บอกตำแหน่งเริ่มต้นใน string1 ที่ต้องการค้นหา

text คือ ข้อความที่ต้องการค้นหาใน string1

ผลลัพธ์ที่ได้ คือ ตัวเลขแสดงตำแหน่งที่ค้นพบ

ตัวอย่างคำสั่งเช่น

```
String studentWord1 = "Welcome to Java, Hello everybody, My name is Somsri, Hello!!!";
int pos1 = studentWord1.indexOf("Hello");
int pos2 = studentWord1.indexOf("Hello", 25);
int pos3 = studentWord1.lastIndexOf("Hello");
int pos4 = studentWord1.lastIndexOf("Hello", 25);
System.out.println("pos1 = " + pos1);
System.out.println("pos2 = " + pos2);
System.out.println("pos3 = " + pos3);
```

```
System.out.println("pos4 = " + pos4);
```

ผลลัพธ์ทางจอภาพ

```
pos1 = 17
```

```
pos1 = 53
```

```
pos1 = 53
```

```
pos1 = 17
```

อธิบายโปรแกรม

จากตัวอย่างโปรแกรมข้างต้น ข้อความใน String ที่ชื่อ studentWord1 มีค่าเท่ากับ

Welcome to Java, Hello everybody, My name is Somsri, Hello!!! และ ในโปรแกรมต้องการค้นหา

ข้อความ Hello ซึ่งปรากฏอยู่ใน studentWord1 ตำแหน่งที่ 17 และ 53 ดังนั้นเมื่อใช้คำสั่ง

studentWord1.indexOf("Hello") จะ ได้ผลลัพธ์เท่ากับ 17 เพราะเป็นตำแหน่งแรกที่ค้นพบ และ

คำสั่ง studentWord1.lastIndexOf("Hello") จะ ได้ผลลัพธ์เท่ากับ 53 เพราะเป็นตำแหน่งสุดท้ายที่

ค้นพบ

2.10.5 StringTokenizer

StringTokenizer เป็น Class ที่อยู่ใน package ที่ชื่อ java.util โดยทำหน้าที่ในการแบ่งข้อความออกเป็นส่วนๆตามสัญลักษณ์ที่กำหนด หรือสัญลักษณ์มาตรฐาน เช่น ช่องว่าง(Space) แท็บ (Tab) และขึ้นบรรทัดใหม่ (Newline) สัญลักษณ์ที่ใช้แบ่งข้อความเหล่านี้ เรียกว่า delimiter ส่วนข้อความที่แบ่งออกมาได้เรียกว่า token

รูปแบบคำสั่ง

```
StringTokenizer(String s, String delim, boolean returnTokens)
```

```
StringTokenizer(String s, String delim)
```

ใช้ในการแบ่ง String ที่ชื่อ s ออกเป็น Token หรือเป็นคำๆด้วยสัญลักษณ์ที่ส่งมากับตัวแปร String ที่ชื่อ delim

```
StringTokenizer(String s)
```

ใช้ในการแบ่ง String ที่ชื่อ s ออกเป็น Token หรือเป็นคำๆด้วยสัญลักษณ์มาตรฐาน

```
boolean hasMoreTokens()
```

เป็นคำสั่งในการตรวจสอบว่ามี Token หลงเหลืออยู่ใน StringTokenizer อีกหรือไม่ ซึ่งจะ

return ค่ากลับเป็นจริงถ้ามี Token เหลืออยู่ มิเช่นนั้นจะเป็นเท็จ

```
String nextToken()
```

เป็นคำสั่งอ่านค่า Token ถัดไปใน StringTokenizer

```
String nextToken(String delim)
```

เป็นคำสั่งอ่านค่า Token ถัดไปใน StringTokenizer โดยมีการแบ่งค่าตามสัญลักษณ์ใน String ที่ชื่อ delim

String count(Tokens)

เป็นคำสั่งที่ใช้นับจำนวน Token ทั้งหมดที่ถูกแบ่งได้ทั้งหมดใน StringTokenizer

ตัวอย่างโปรแกรม

```
import java.util.StringTokenizer;
```

```
class TestToken {
```

```
    public static void main(String[] args)
```

```
    {
```

```
        String input = "My name is Somsri";
```

```
        StringTokenizer st= new StringTokenizer(input);
```

```
        System.out.println("Count of Tokens = " + st.countTokens());
```

```
        int i=0;
```

```
        while(st.hasMoreTokens()){
```

```
            i++;
```

```
            System.out.println("Token"+(i) +" = " + st.nextToken());
```

```
        }
```

```
    }
```

```
}
```

ผลลัพธ์ทางจอภาพ

```
Token1 = My
```

```
Token2 = name
```

```
Token3 = is
```

```
Token4 = Somsri
```

คำอธิบาย

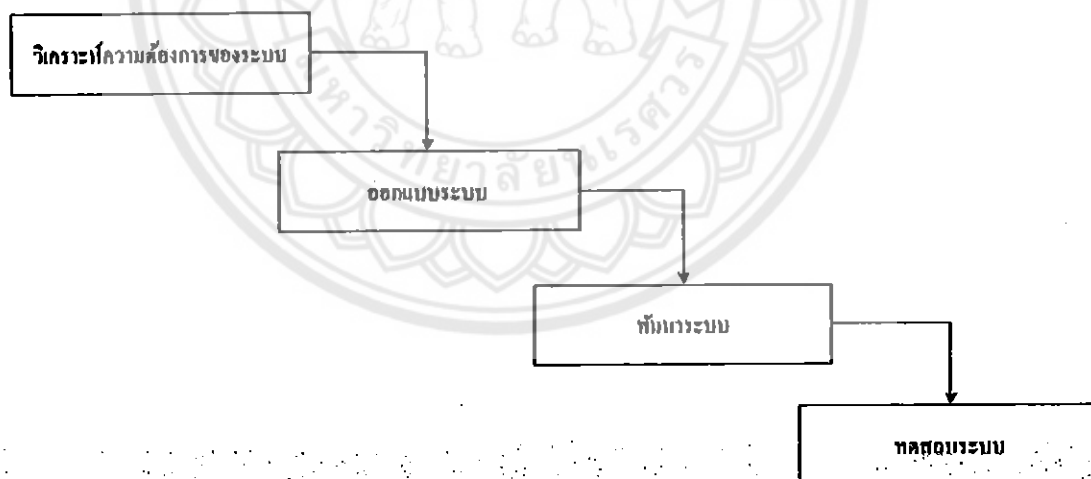
ผลลัพธ์ของโปรแกรมข้างต้นจะแยก String ที่ชื่อ input ได้ทั้งหมด 4 คำซึ่งถูกแบ่งด้วย สัญลักษณ์มาตรฐาน (ช่องว่าง) ด้วยคำสั่ง StringTokenizer st= new StringTokenizer(input) เก็บไว้ในตัวแปร Object ของ Class StringTokenizer ที่ชื่อ st จากนั้นวนลูปตรวจสอบว่ามี Token ใน st ยังเหลืออยู่หรือไม่ แล้วแสดงผล Token แต่ละคำด้วยคำสั่ง st.nextToken() จนกว่าจะหมด Token ใน st ซึ่งได้ Token ทั้งหมด 4 คำดังผลลัพธ์ที่แสดงทางจอภาพ

บทที่ 3

หลักการงานของระบบจัดหมวดหมู่เนื้อหาอัตโนมัติสำหรับเว็บด้านการท่องเที่ยว

จากการมองเห็นปัญหาการในการจัดหมวดหมู่เนื้อหาโดยอาศัยการวิเคราะห์จากองค์ความรู้ที่มีอยู่คนนั้นหากมีข้อมูลมากจะใช้เวลานาน อีกทั้งอาจเกิดความผิดพลาดในการจัดหมวดหมู่ได้เนื่องจากคนอาจจะมีองค์ความรู้ไม่ครอบคลุมข้อมูลเนื้อหาเกี่ยวกับการท่องเที่ยวที่มีอยู่มากมาย โครงการนี้จึงได้พิจารณาถึงแนวทางในการพัฒนาการจัดหมวดหมู่ข้อมูลการท่องเที่ยวอัตโนมัติ ซึ่งทำให้เพิ่มวิธีในการจัดหมวดหมู่ข้อมูลการท่องเที่ยว อีกทั้งยังช่วยลดเวลาในการจัดหมวดหมู่ของผู้ดูแลระบบได้ โครงการนี้เป็นอีกทางเลือกหนึ่งที่จะช่วยเพิ่มความสะดวกให้แก่ผู้ดูแลระบบในการจัดหมวดหมู่ และเพิ่มความถูกต้องในการจัดหมวดหมู่ข้อมูลการท่องเที่ยว ทำให้เกิดความสะดวกและลดเวลาในการค้นหาข้อมูลในการหาข้อมูล

ซึ่งขั้นตอนในการพัฒนาระบบแบ่งเป็น 4 ขั้นตอน ดังภาพที่ 3-1 โดยแต่ละขั้นตอนมีรายละเอียดดังนี้



รูปที่ 3.1 ขั้นตอนในการพัฒนาระบบ

3.1 วิเคราะห์ความต้องการของระบบจัดหมวดหมู่เนื้อหาอัตโนมัติสำหรับเว็บด้านการท่องเที่ยว

3.1.1 การจัดหมวดหมู่โดยใช้การวิเคราะห์ของคนเป็นหลัก ในการจัดหมวดหมู่โดยอาศัยการวิเคราะห์ตามองค์ความรู้ของของแต่ละคนนั้นอาจมีไม่เพียงพอ ทำให้เกิดการวิเคราะห์ผิดพลาด อีกทั้งข้อมูลการท่องเที่ยวที่มีอยู่ปริมาณมากทำให้ใช้เวลาในการวิเคราะห์แยกหมวดหมู่นั้นใช้เวลานาน

3.1.2 การจัดหมวดหมู่อัตโนมัติ ในการจัดหมวดหมู่อัตโนมัตินั้นจะทำการจัดหมวดหมู่โดยอ้างอิงข้อมูลหมวดหมู่และคำที่ใช้ในการวิเคราะห์จัดหมวดหมู่ โดยข้อมูลดังกล่าวจะเก็บไว้ในฐานข้อมูลแล้วนำมาเปรียบเทียบกับข้อมูลเอกสารที่ผู้ดูแลระบบเพิ่มเข้ามา ทำให้ช่วยลดเวลาในการวิเคราะห์หมวดหมู่ของผู้ดูแลระบบได้

3.1.3 ฟังก์ชันที่ต้องการ

ผู้ใช้งานทั่วไป

- ผู้ใช้งานทั่วไปสามารถค้นหาข้อมูลตามหมวดหมู่และค้นหาโดยใช้คีย์เวิร์ด จากเอกสารการท่องเที่ยวได้

ผู้ดูแลระบบ

- ผู้ดูแลระบบสามารถเพิ่มและลบเอกสารข้อมูลการท่องเที่ยว
- ผู้ดูแลระบบสามารถเพิ่มและลบหมวดหมู่ของสถานที่ท่องเที่ยวและกิจกรรมท่องเที่ยวได้
- ผู้ดูแลระบบสามารถเพิ่มและลบความสัมพันธ์ระหว่างหมวดหมู่ของสถานที่ท่องเที่ยว

และกิจกรรมท่องเที่ยวได้

- ผู้ดูแลระบบสามารถเพิ่มและลบคำเหมือนและคำคล้ายได้
- ผู้ดูแลระบบสามารถเพิ่มคีย์เวิร์ดที่เกี่ยวข้องกับเอกสารได้
- ผู้ดูแลระบบสามารถเพิ่มและลบหมวดหมู่ของสถานที่ท่องเที่ยวและกิจกรรมท่องเที่ยวได้

ระบบจัดหมวดหมู่ข้อมูลการท่องเที่ยวอัตโนมัติ

- ระบบสามารถจัดหมวดหมู่ข้อมูลการท่องเที่ยวได้อัตโนมัติ

3.2 การออกแบบระบบ

จากการศึกษาและรวบรวมข้อมูลของการพัฒนาระบบจัดหมวดหมู่ข้อมูลการท่องเที่ยวอัตโนมัติ นั้น ได้ทำการจำลองข้อมูล วิเคราะห์ออกแบบระบบ ออกแบบฐานข้อมูล และออกแบบหน้าจอการทำงาน ออกแบบการติดต่อผู้ใช้งาน และออกแบบเครื่องมือการทดสอบ โดยแต่ละขั้นตอนมีรายละเอียดดังนี้

3.2.1 แผนภาพกระแสข้อมูล (Data Flow Diagram) โดยแผนภาพกระแสข้อมูลจะแสดงความสัมพันธ์ระหว่างโปรเซส (Process) กับข้อมูล (Data) ที่เกี่ยวข้องโดยข้อมูลในแผนภาพกระแสข้อมูลจะทำให้ทราบถึง

- ข้อมูลมาจากไหน
- ข้อมูลไปที่ไหน
- ข้อมูลเก็บไว้ที่ไหน
- เกิดเหตุการณ์ใดกับข้อมูลบ้างระหว่างทาง

โดยได้แบ่งระดับในการเขียนแผนภาพกระแสข้อมูลดังนี้

3.2.1.1 ระดับที่ 0 หรือเรียกว่า DFD Level 0 หรือ Context Diagram เป็นแผนภาพที่แสดงให้เห็นว่าระบบโดยรวมนั้นดำเนินการได้อย่างไรบ้างเพื่อเพิ่มความเข้าใจในระบบงานที่จะพัฒนาให้ตรงกันของผู้พัฒนาระบบและผู้ใช้งานระบบ สามารถแสดงรายละเอียดที่เกี่ยวข้องกับแผนภาพกระแสข้อมูล และแสดงแผนภาพกระแสข้อมูลระดับ 0 ได้ดังรูปที่ 3.2

3.2.1.2 ระดับที่ 1 หรือเรียกว่า DFD Level 1 เป็นการอธิบายถึงกระบวนการทำงานต่างๆ ภายในระบบซึ่งเป็นที่รวมของโปรเซสหลักและข้อมูลหลักๆ ที่เกี่ยวข้อง และแสดงแผนภาพกระแสข้อมูลระดับ 1 ได้ดังรูปที่ 3.3

3.2.1.3 ระดับที่ 2 หรือเรียกว่า DFD Level 2 จะแสดงโปรเซสย่อย (Sub Process) ของแผนภาพกระแสข้อมูลระดับที่ 1 และแสดงแผนภาพกระแสข้อมูลระดับ 2 ได้ดังรูปที่ 3.4

จากการวิเคราะห์ระบบจัดหมวดหมู่ข้อมูลการท่องเที่ยวอัตโนมัติ นั้น ได้แบ่งระดับของผู้ใช้งานออกเป็น 2 ระดับด้วยกัน คือ

- ผู้ใช้ทั่วไป คือ ผู้ที่สามารถค้นหาข้อมูลตามหมวดหมู่และค้นหาโดยใช้ 키워ด์ จากเอกสารการท่องเที่ยวได้
- ผู้จัดการระบบ คือ ผู้ที่สามารถเพิ่มลบเอกสารการท่องเที่ยว เพิ่มลบหมวดหมู่ สถานที่ท่องเที่ยวและกิจกรรม เพิ่มความสัมพันธ์ระหว่างหมวดหมู่ เพิ่มและลบคำเหมือนและคำคล้าย และค้นหาข้อมูลตามหมวดหมู่และค้นหาโดยใช้ 키워ด์ จากเอกสารการท่องเที่ยวได้

แสดงโปรเซสการทำงานจากระบบจัดหมวดหมู่เนื้อหาข้อมูลการท่องเที่ยวอัตโนมัติ

List of Process

1. Login
2. เพิ่ม/แก้ไข Category
3. เพิ่ม/แก้ไข Document
4. เพิ่ม/แก้ไข ความสัมพันธ์ของหมวดหมู่สถานที่ท่องเที่ยว กิจกรรมท่องเที่ยว
5. เพิ่ม/แก้ไข คำคล้าย คำใกล้เคียง
6. ระบบจัดหมวดหมู่อัตโนมัติ
7. แสดงผลการค้นหา

List of Data

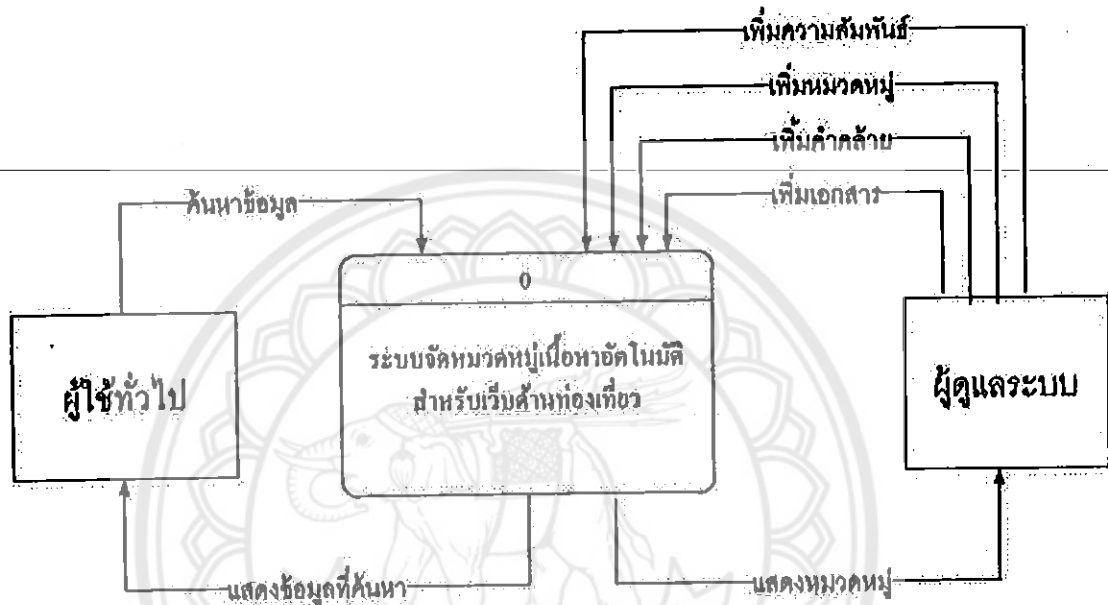
1. ข้อมูลการท่องเที่ยวที่ยังไม่ได้จัดหมวดหมู่
2. ข้อมูล Login
3. ข้อมูลหมวดหมู่สถานที่ท่องเที่ยว กิจกรรมท่องเที่ยว
4. ข้อมูลที่จัดหมวดหมู่แล้ว

List of External entity

1. ผู้ดูแลระบบ
2. ผู้ใช้ทั่วไป

Context Diagram DFD Level 0

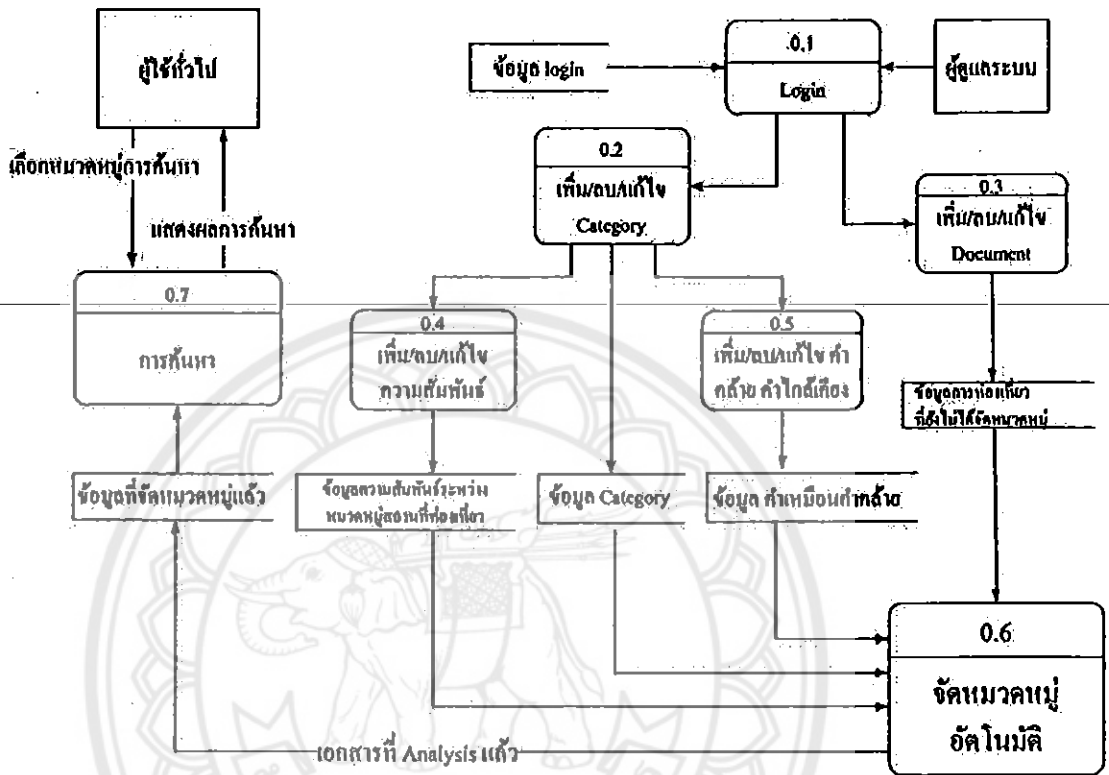
Context Diagram หรือ Data Flow Level 0 แสดงกระบวนการทำงานหลักของระบบจัดหมวดหมู่ข้อมูลการท่องเที่ยวแบบอัตโนมัติ โดยแสดงความสัมพันธ์ระหว่างผู้ใช้ทั่วไปและผู้ดูแลระบบ กับระบบจัดหมวดหมู่เนื้อหาด้านการท่องเที่ยวอัตโนมัติ



รูปที่ 3.2 ภาพแสดง Context Diagram DFD Level 0

Context Diagram DFD Level 1

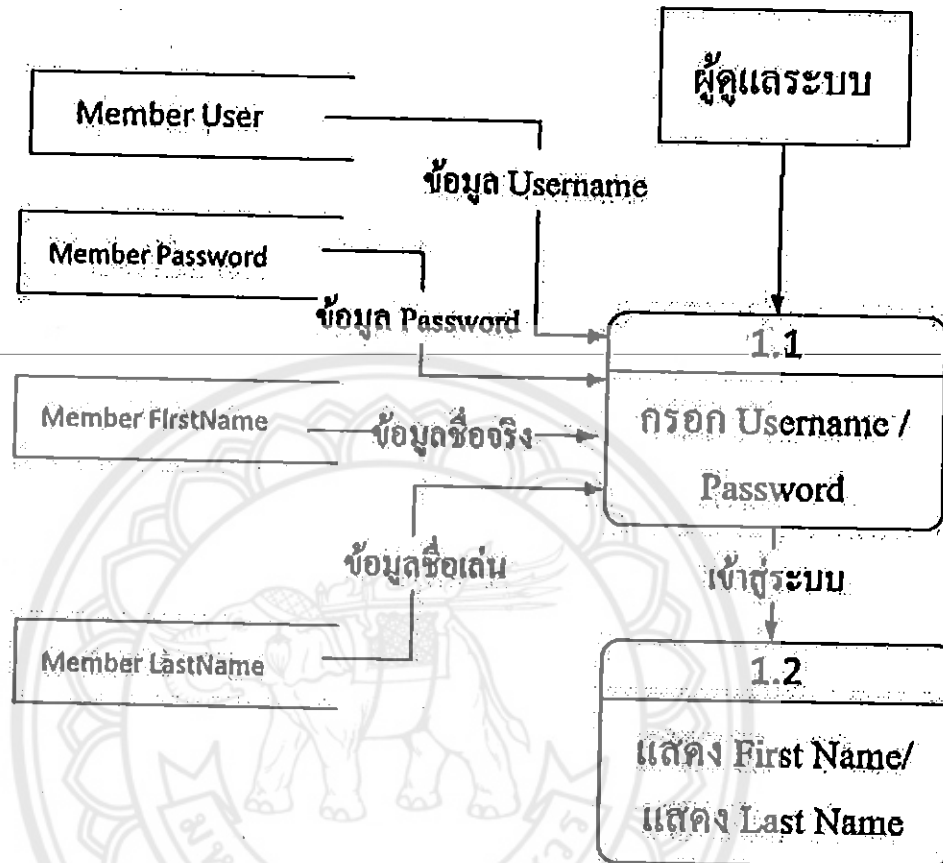
Data Flow Level 0 แสดงกระบวนการทำงานภายในของระบบจัดหมวดหมู่ข้อมูลการ
ท่องเที่ยวแบบอัตโนมัติ โดยแสดงความสัมพันธ์ระหว่างกระบวนการทำงานย่อย 7 กระบวนการ



รูปที่ 3.3 ภาพแสดง Context Diagram DFD Level 1

Context Diagram DFD Level 2: Login

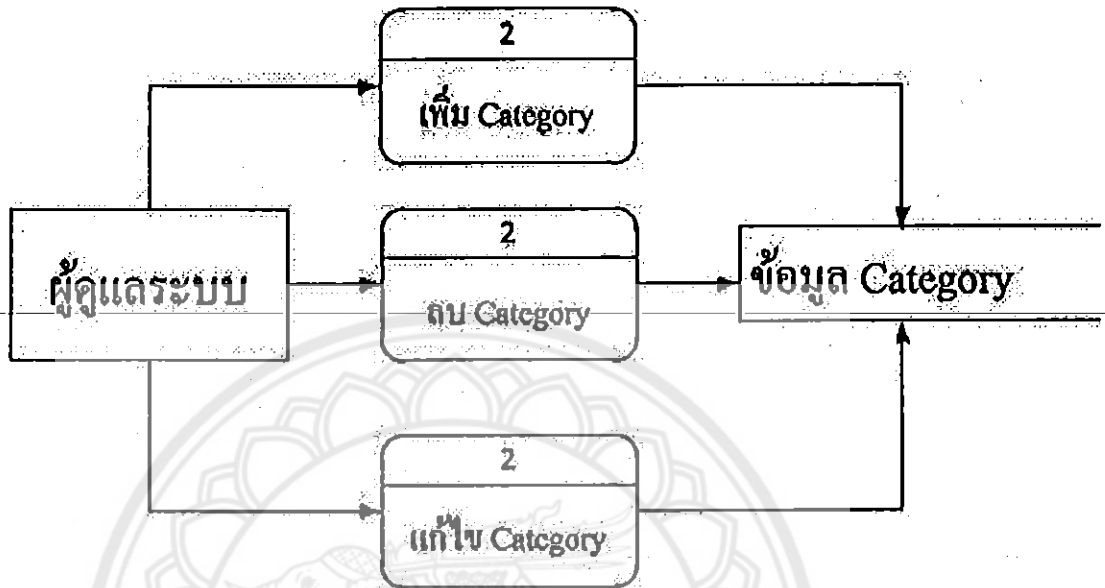
Data Flow Level 2 แสดงกระบวนการทำงานของกระบวนการเข้าสู่ระบบของผู้ดูแลระบบ



รูปที่ 3.4 ภาพแสดง Context Diagram DFD Level 2 : Login

Context Diagram DFD Level 2: เพิ่ม/ลบ/แก้ไข Category

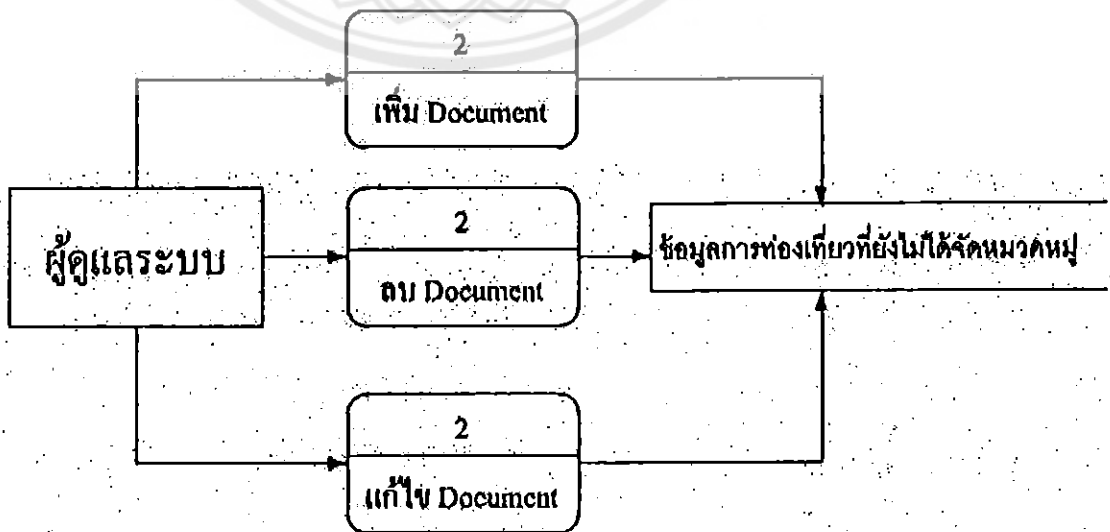
Data Flow Level 2 ของกระบวนการเพิ่ม ลบแก้ไข Category แสดงกระบวนการทำงาน กระบวนการทำงานกระบวนการเพิ่ม ลบแก้ไข Category (หมวดหมู่ข้อมูลด้านการท่องเที่ยว)



รูปที่ 3.5 ภาพแสดง Context Diagram DFD Level 2 : เพิ่ม/ลบ/แก้ไข Category

Context Diagram DFD Level 2: เพิ่ม/ลบ/แก้ไข Document

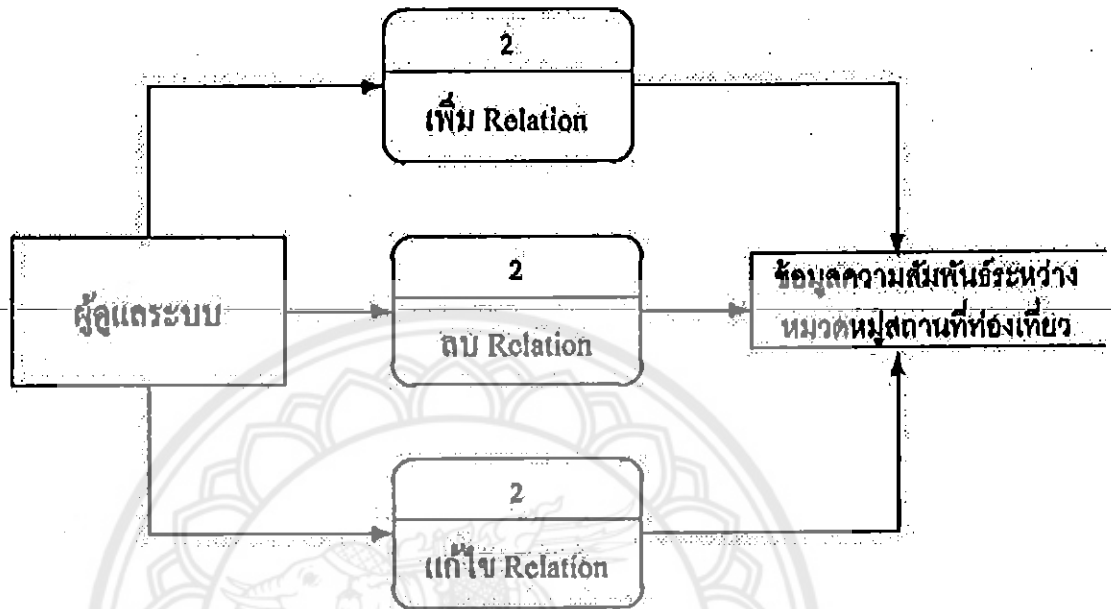
Data Flow Level 2 แสดงกระบวนการทำงานกระบวนการทำงานของการ เพิ่ม/ลบ/แก้ไข Document (เอกสารข้อมูลด้านการท่องเที่ยว)



รูปที่ 3.6 ภาพแสดง Context Diagram DFD Level 2 : เพิ่ม/ลบ/แก้ไข Document

Context Diagram DFD Level 2: เพิ่ม/ลบ/แก้ไข Relation

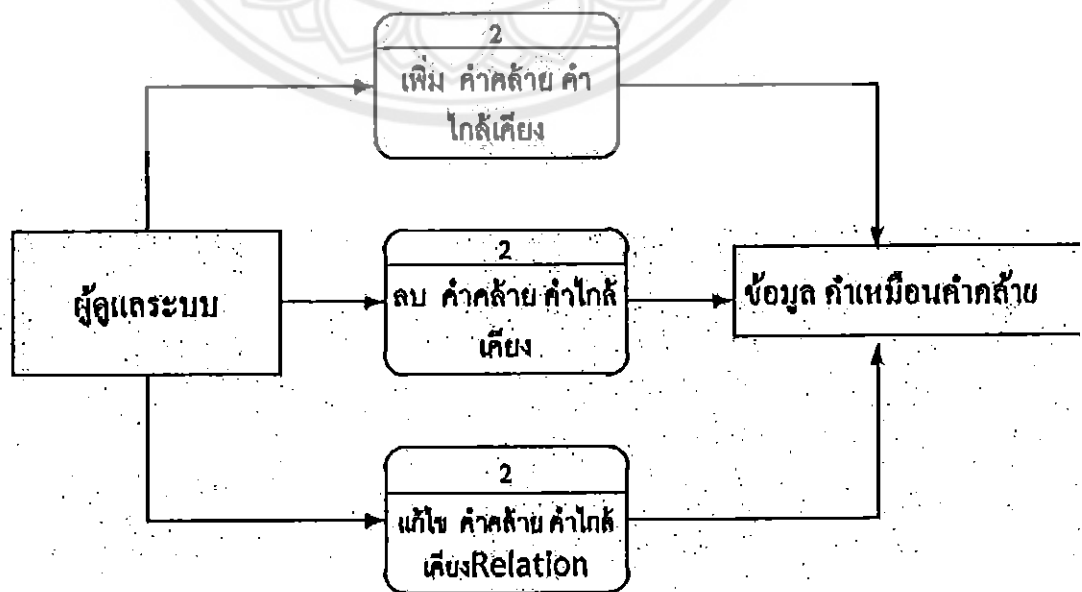
Data Flow Level 2 แสดงกระบวนการทำงานกระบวนการทำงานของการ เพิ่ม/ลบ/แก้ไข Relation (ความสัมพันธ์ของข้อมูลด้านการท่องเที่ยว)



รูปที่ 3.7 ภาพแสดง Context Diagram DFD Level 2 : เพิ่ม/ลบ/แก้ไข Relation

Context Diagram DFD Level 2: เพิ่ม/ลบ/แก้ไข คำคล้าย คำใกล้เคียง

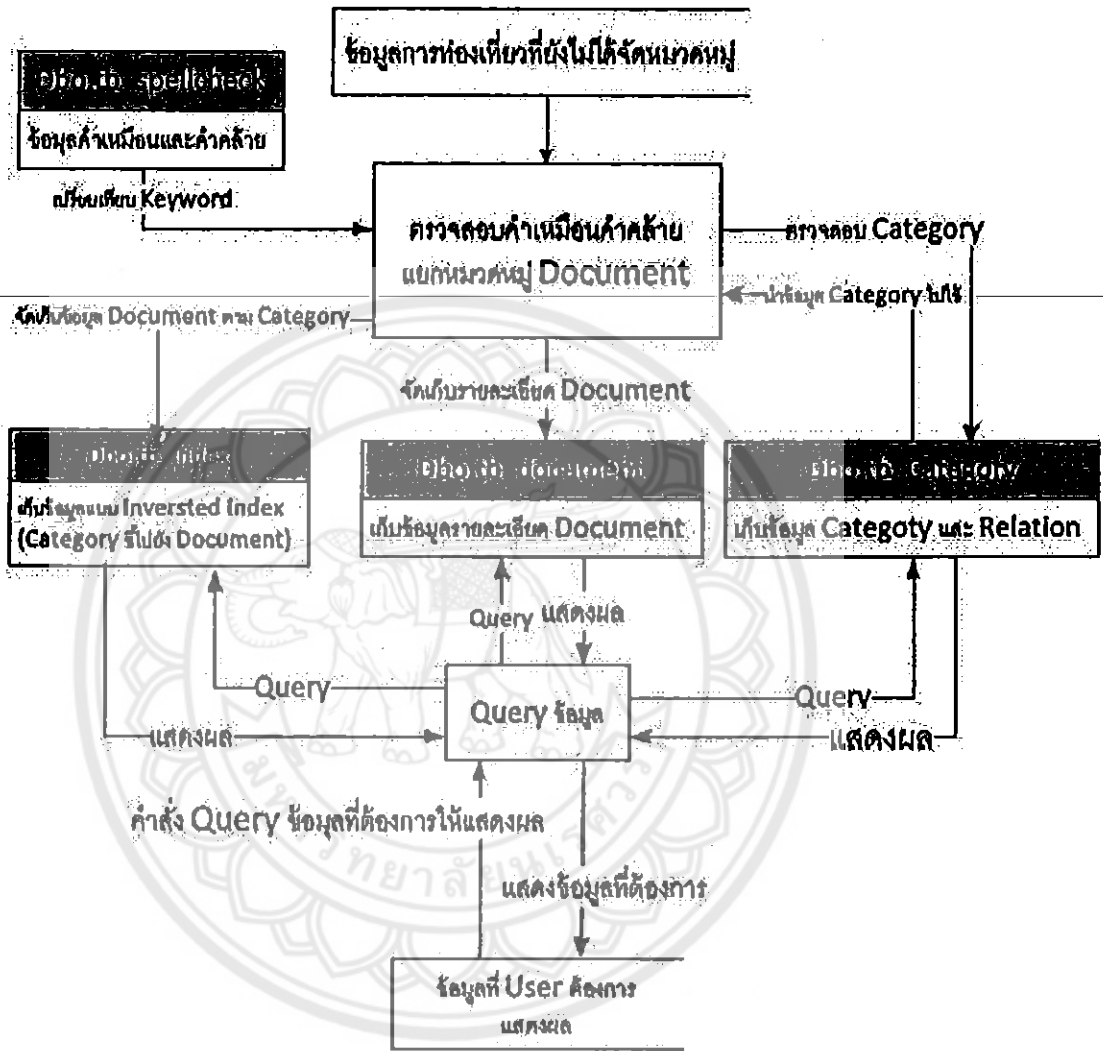
Data Flow Level 2 แสดงกระบวนการทำงานกระบวนการทำงานของการ เพิ่ม/ลบ/แก้ไข คำที่มีความหมายคล้าย หรือ คำที่มีความหมายใกล้เคียง ที่ถูกจัดเก็บไว้ในฐานข้อมูล



รูปที่ 3.8 ภาพแสดง Context Diagram DFD Level 2 : เพิ่ม/ลบ/แก้ไข คำคล้าย คำใกล้เคียง

Context Diagram DFD Level 2: จัดหมวดหมู่อัตโนมัติ

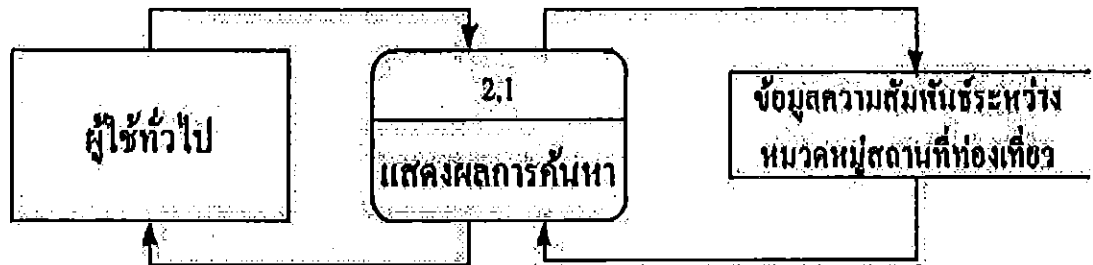
Data Flow Level 2 ของกระบวนการจัดหมวดหมู่อัตโนมัติ แสดงกระบวนการทำงานและการเชื่อมโยงของตารางของฐานข้อมูลกับกระบวนการจัดหมวดหมู่อัตโนมัติ



รูปที่ 3.9 ภาพแสดง Context Diagram DFD Level 2 : จัดหมวดหมู่อัตโนมัติ

Context Diagram DFD Level 2: การค้นหา

Data Flow Level 2 แสดงกระบวนการทำงานของการค้นหาและนำข้อมูลค้นหา
 ที่องที่ขวออกมาแสดงผลให้กับผู้ใช้ทั่วไป



รูปที่ 3.10 ภาพแสดง Context Diagram DFD Level 2 : การค้นหา

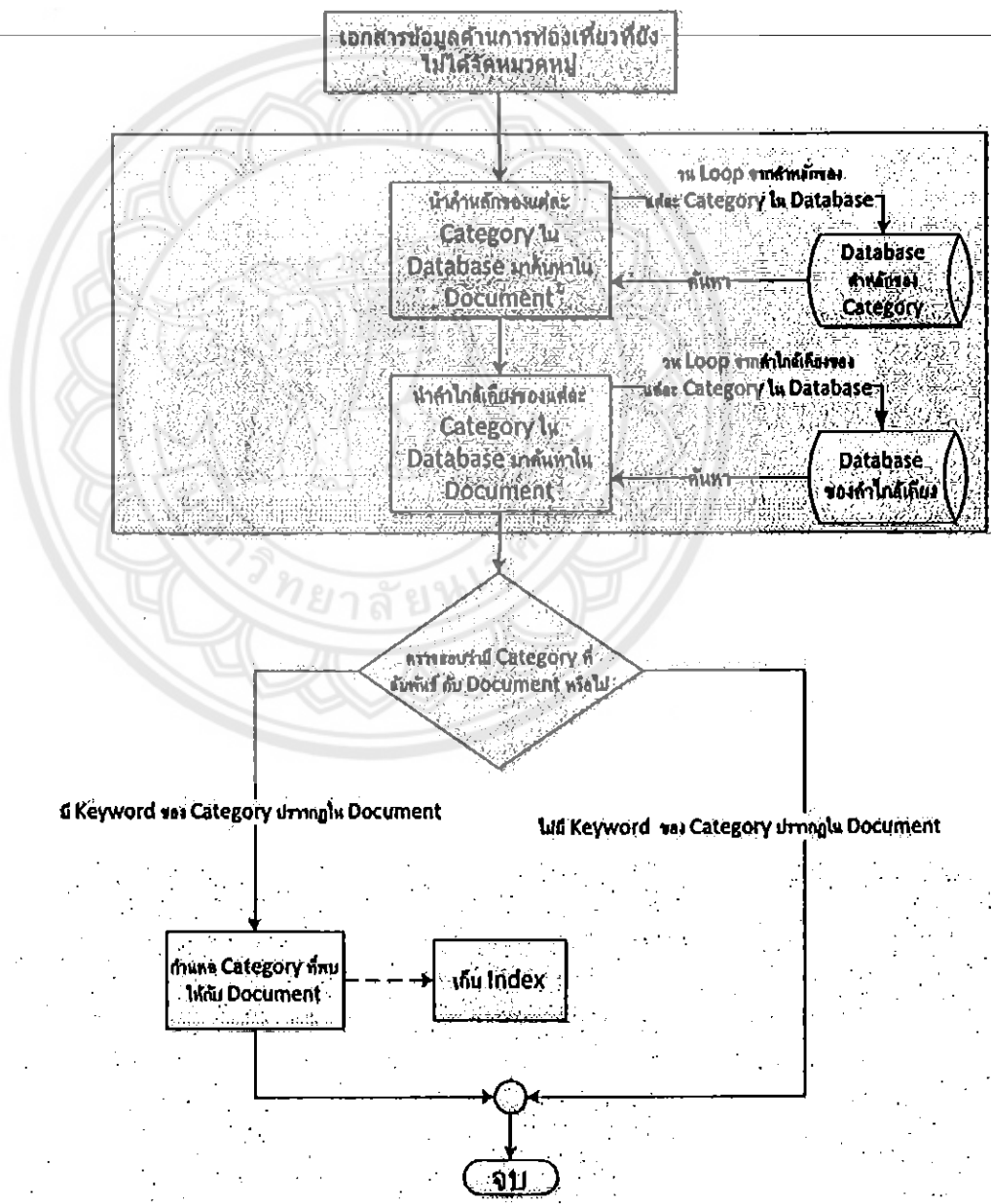


3.2.2 ออกแบบกระบวนการจัดหมวดหมู่อัตโนมัติ

กระบวนการจัดหมวดหมู่ข้อมูลอัตโนมัตินั้นมีหลากหลายวิธี แต่ที่ได้ออกแบบกระบวนการจัดหมวดหมู่ไว้ 2 วิธีคือ

3.2.2.1 การจัดหมวดหมู่โดยไม่ใช้วิธีการตัดคำ

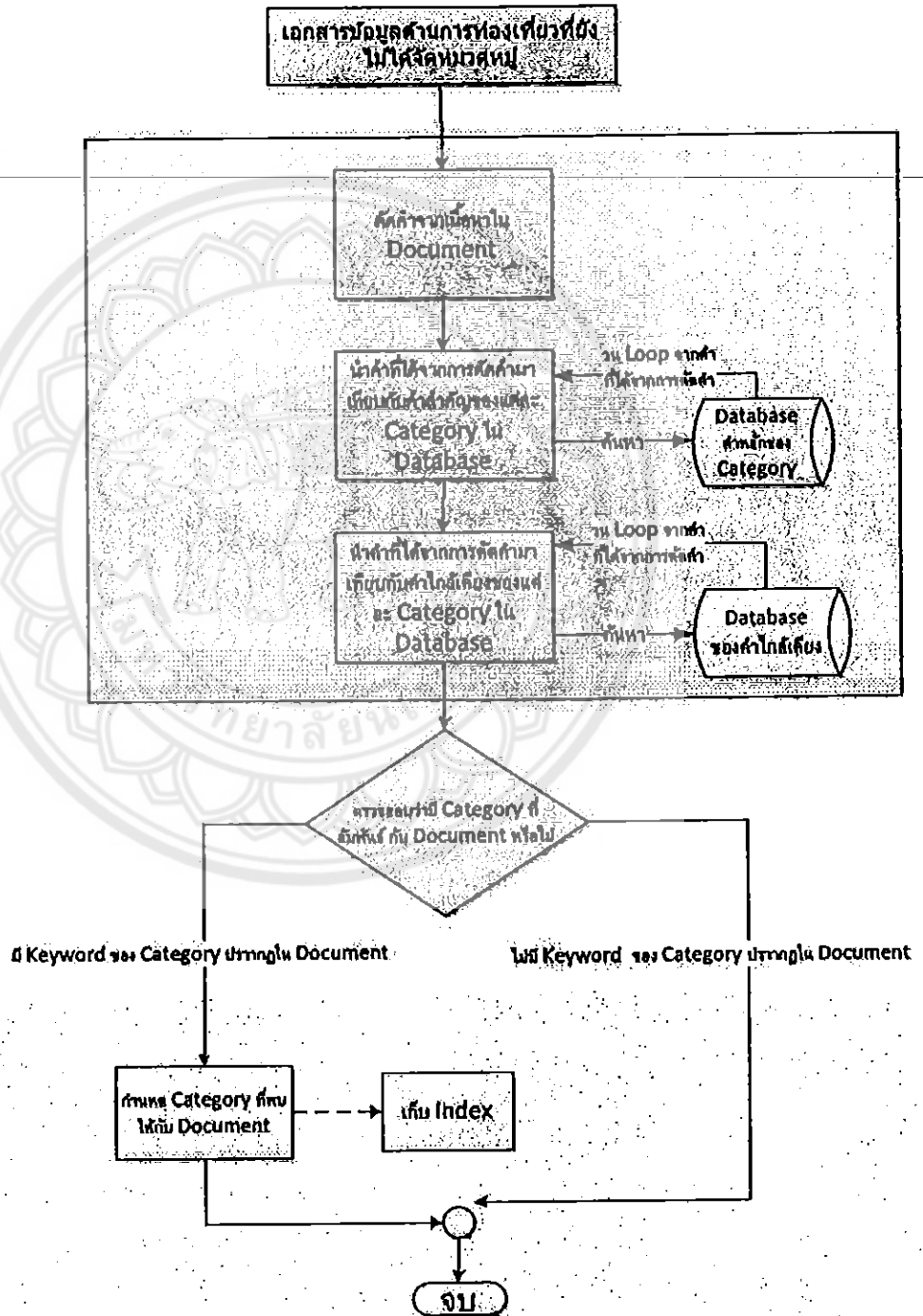
ขั้นตอนกระบวนการจัดหมวดหมู่โดยไม่ใช้วิธีการตัดคำ เป็นการนำคำหลักและคำใกล้เคียงของแต่ละ Category ใน Database มาค้นหาในเอกสารข้อมูลด้านการท่องเที่ยว ถ้ามีคีย์เวิร์ดของ Category และคำใกล้เคียงในเอกสาร ก็ทำการจัดหมวดหมู่ให้กับเอกสารตาม ID ของ Category ที่ค้นเจอแล้วทำการเก็บในฐานข้อมูลเพื่อใช้ในการค้นหาต่อไป



รูปที่ 3.11 รูปแสดงการจัดหมวดหมู่โดยเปรียบเทียบตัวอักษร

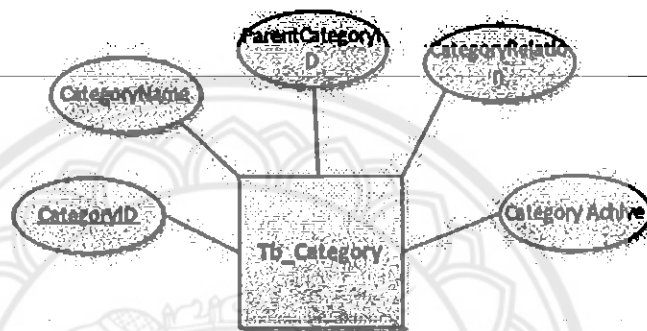
3.2.2.2 การจัดหมวดหมู่โดยการใช้การตัดคำ

ขั้นตอนกระบวนการจัดหมวดหมู่โดยใช้วิธีการตัดคำเป็นการนำคำหลักและคำใกล้เคียงของแต่ละ Category ใน Database มาค้นหาใน เอกสารที่ผ่านการตัดคำในส่วน ของเนื้อหา ถ้ามีคีย์เวิร์ดของ Category และคำใกล้เคียงในเอกสาร ก็ทำการจัดหมวดหมู่ ให้กับเอกสารตาม ID ของ Category ที่ค้นเจอแล้วทำการเก็บในฐานข้อมูล



รูปที่ 3.12 รูปแสดงการจัดหมวดหมู่โดยการใช้การตัดคำ

3.2.3 ER Diagram (The Entity Relationship Diagram) แบบจำลองอี-อาร์ เป็นเครื่องมือที่จะช่วยในการออกแบบเพื่ออธิบายข้อมูลต่างๆ ในรูปแบบของเอนทิตีและความสัมพันธ์ระหว่างเอนทิตีเหล่านั้นอีกทั้งยังเป็นที่ยอมรับแพร่หลายในการออกแบบข้อมูล เนื่องจากสามารถให้หลักการที่ใช้ในการแปลงความต้องการของผู้ใช้ไปอยู่ในรูปแบบแผนผัง พร้อมทั้งรายละเอียดที่จำเป็นในการสร้างฐานข้อมูล สามารถแสดงแผนภาพ อีอาร์ โคอะแกรม และพจนานุกรมข้อมูล (Data Dictionary)



รูปที่ 3.13 แสดงภาพ ER Diagram ของ ตารางหมวดหมู่สถานที่ท่องเที่ยว

ER Diagram ของ ตารางหมวดหมู่สถานที่ท่องเที่ยว

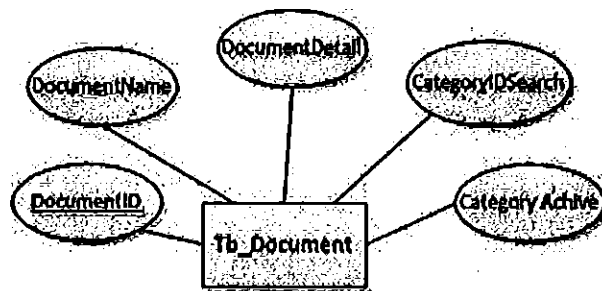
ส่วนประกอบที่ 1 : Entity

- หมวดหมู่สถานที่ท่องเที่ยว

ส่วนประกอบที่ 2 : Attribute

- ID ของหมวดหมู่สถานที่ท่องเที่ยว/กิจกรรมท่องเที่ยว
- ชื่อของหมวดหมู่สถานที่ท่องเที่ยว/กิจกรรมท่องเที่ยว
- แสดงโหนดที่เป็น Parent
- เก็บ CategoryID ที่มีความเกี่ยวข้องกัน
- สถานะถูกใช้งาน 1 ถูกใช้งาน 0 ไม่ถูกใช้งาน

3.2.3.1 ER Diagram ของตารางเก็บเอกสารข้อมูลสถานที่ท่องเที่ยว



รูปที่ 3.14 แสดงภาพ ER Diagram ของตารางเก็บเอกสารข้อมูลสถานที่ท่องเที่ยว

ER Diagram ของ ตารางเก็บเอกสารข้อมูลสถานที่ท่องเที่ยว ประกอบด้วย

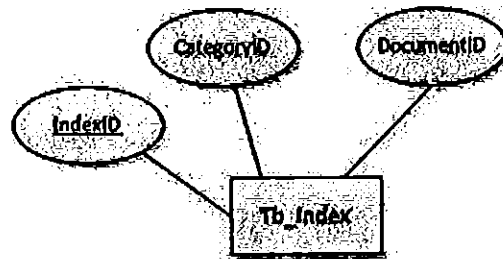
ส่วนประกอบที่ 1 : Entity

- เอกสารข้อมูลสถานที่ท่องเที่ยว

ส่วนประกอบที่ 2 : Attribute

- ID ของเอกสารที่ Admin เพิ่ม
- ชื่อของเอกสาร
- เนื้อหาของเอกสาร
- แสดงเอกสารถูกจัดอยู่ใน CategoryID ใดบ้าง
- คำอธิบายเอกสารเพิ่มเติม (ใช้ในการค้นหาแบบธรรมดา)
- แสดงหมวดหมู่สถานที่ท่องเที่ยวเป็น String
- สถานะถูกใช้งาน 1 ถูกใช้งาน 0 ไม่ถูกใช้งาน

3.2.3.2 ER Diagram ของตารางเก็บเอกสารการท่องเที่ยวตามหมวดหมู่ของสถานที่ท่องเที่ยว



รูปที่ 3.15 แสดงภาพ ER-Diagram ของตารางเก็บเอกสารการท่องเที่ยวตามหมวดหมู่ของสถานที่ท่องเที่ยว

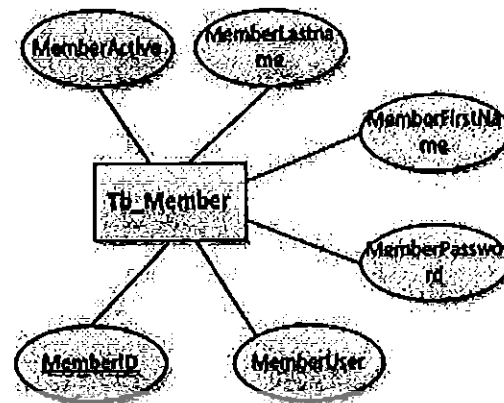
ER Diagram ของ ตารางเก็บเอกสารการท่องเที่ยวตามหมวดหมู่ของสถานที่ท่องเที่ยว ประกอบด้วย ส่วนประกอบที่ 1 : Entity

- เอกสารการท่องเที่ยวตามหมวดหมู่ของสถานที่ท่องเที่ยว

ส่วนประกอบที่ 2 : Attribute

- ลำดับเอกสารที่ถูกรับเพิ่ม
- ID ของหมวดหมู่สถานที่ท่องเที่ยว/กิจกรรมท่องเที่ยว
- ID ของเอกสารที่ถูกรับเพิ่มตามหมวดหมู่สถานที่ท่องเที่ยว/กิจกรรมท่องเที่ยว

3.2.3.3 ER Diagram ของตารางเก็บข้อมูลสมาชิก



รูปที่ 3.16 แสดงภาพ E-R Diagram ของตารางเก็บข้อมูลสมาชิก

ER Diagram ของ ตารางเก็บข้อมูลสมาชิก ประกอบด้วย

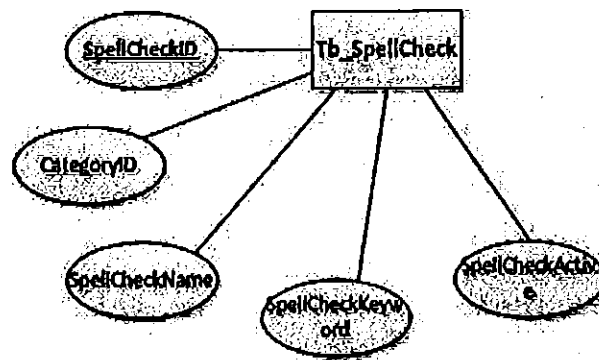
ส่วนประกอบที่ 1 : Entity

- ข้อมูลสมาชิก

ส่วนประกอบที่ 2 : Attribute

- ID ของสมาชิก
- Username
- Password
- ชื่อจริงของสมาชิก ใช้แสดงหลังจาก Login ผ่าน
- นามสกุลของสมาชิก ใช้แสดงหลังจาก Login ผ่าน
- สถานะถูกใช้งาน 1 ถูกใช้งาน 0 ไม่ถูกใช้งาน

3.2.3.4 ER Diagram ของตารางเก็บข้อมูลคำคล้าย



รูปที่ 3.17 แสดงภาพ ER Diagram ของตารางเก็บข้อมูลคำคล้าย

ER Diagram ของ ตารางเก็บข้อมูลคำคล้าย ประกอบด้วย

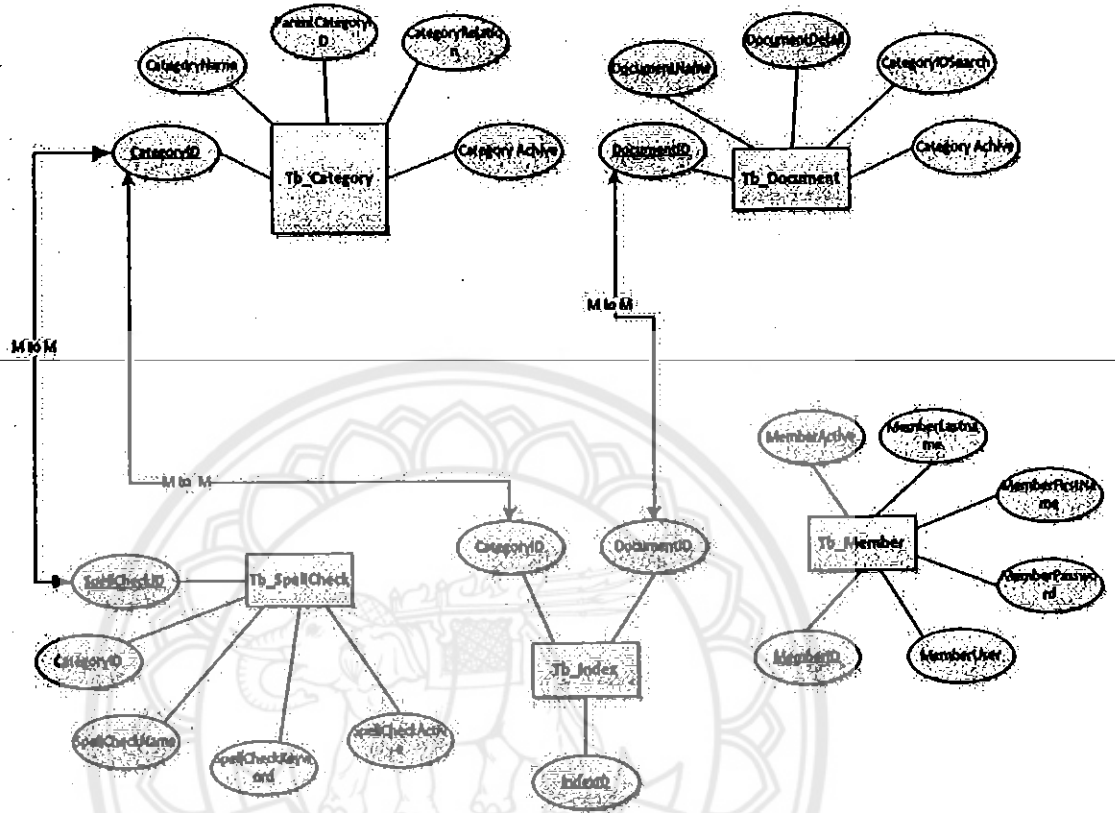
ส่วนประกอบที่ 1 : Entity

- ข้อมูลคำคล้าย

ส่วนประกอบที่ 2 : Attribute

- ID ของคำที่ใช้ในการตรวจสอบกับเนื้อหาของเอกสารที่เพิ่ม
- ID ของหมวดหมู่สถานที่ท่องเที่ยว/กิจกรรมท่องเที่ยว
- คำที่ใช้ในการตรวจสอบกับเนื้อหาของเอกสารที่เพิ่ม
- คำที่ต้องการให้แสดงเพิ่มเติม เชื่อมโยงกับ SpellCheckName
- สถานะถูกใช้งาน 1 ถูกใช้งาน 0 ไม่ถูกใช้งาน

3.2.3.5 ความสัมพันธ์ระหว่างตารางฐานข้อมูลระบบจัดหมวดหมู่ข้อมูลการท่องเที่ยวอัตโนมัติ



รูปที่ 3.18 แสดงความสัมพันธ์ตารางฐานข้อมูลระบบจัดหมวดหมู่ข้อมูลการท่องเที่ยวอัตโนมัติ

3.2.4 Data Dictionary

tb_Category : ตารางหมวดหมู่สถานที่ท่องเที่ยว

ตารางที่ 3.1 แสดงตารางหมวดหมู่สถานที่ท่องเที่ยว

ชื่อ field	ชนิดข้อมูล	คำอธิบาย
CategoryID (PK)	Int,not null	ID ของหมวดหมู่สถานที่ท่องเที่ยว/กิจกรรมท่องเที่ยว
CategoryName	Nvarchar(max), null	ชื่อของหมวดหมู่สถานที่ท่องเที่ยว/กิจกรรมท่องเที่ยว
ParentCategoryID	Int,null	แสดงโหนดที่เป็น Parent
CategoryRelation	Nvarchar(max), null	เก็บ CategoryID ที่มีความเกี่ยวข้องกัน
CategoryActive	Int,null	สถานะถูกใช้งาน 1 ถูกใช้งาน 0 ไม่ถูกใช้งาน

tb_Document : ตารางเก็บเอกสารข้อมูลสถานที่ท่องเที่ยว

ตารางที่ 3.2 แสดงตารางเก็บเอกสารข้อมูลสถานที่ท่องเที่ยว

ชื่อ field	ชนิดข้อมูล	คำอธิบาย
DocumentID (PK)	Int,not null	ID ของเอกสารที่ Admin เพิ่ม
DocumentName	Nvarchar(max), null	ชื่อของเอกสาร
DocumentDetail	Nvarchar(max), null	เนื้อหาของเอกสาร
CategoryIDSearch	Nvarchar(max), null	แสดงเอกสารถูกจัดอยู่ใน CategoryID ใบบ้าง
DocumentKeyword	Nvarchar(max), null	คำอธิบายเอกสารเพิ่มเติม (ใช้ในการค้นหาแบบ ธรรมชาติ)
CategoryNameSearch	Nvarchar(max), null	แสดงหมวดหมู่สถานที่ท่องเที่ยวเป็น String
DocumentActive	Int,null	สถานะถูกใช้งาน 1 ถูกใช้งาน 0 ไม่ถูกใช้งาน

tb_index : ตารางเก็บเอกสารการท่องเที่ยวตามหมวดหมู่ของสถานที่ท่องเที่ยว

ตารางที่ 3.3 แสดงตารางเก็บเอกสารการท่องเที่ยวตามหมวดหมู่ของสถานที่ท่องเที่ยว

ชื่อ field	ชนิดข้อมูล	คำอธิบาย
IndexID (PK)	Int,not null	ลำดับเอกสารที่ถูกเพิ่ม
CategoryID	Int,null	ID ของหมวดหมู่สถานที่ท่องเที่ยว/กิจกรรมท่องเที่ยว
DocumentID	Nvarchar(max), null	ID ของเอกสารที่ถูกจัดตามหมวดหมู่สถานที่ท่องเที่ยว/ กิจกรรมท่องเที่ยว

tb_Member : ตารางเก็บข้อมูลสมาชิก

ตารางที่ 3.4 แสดงตารางเก็บข้อมูลสมาชิก

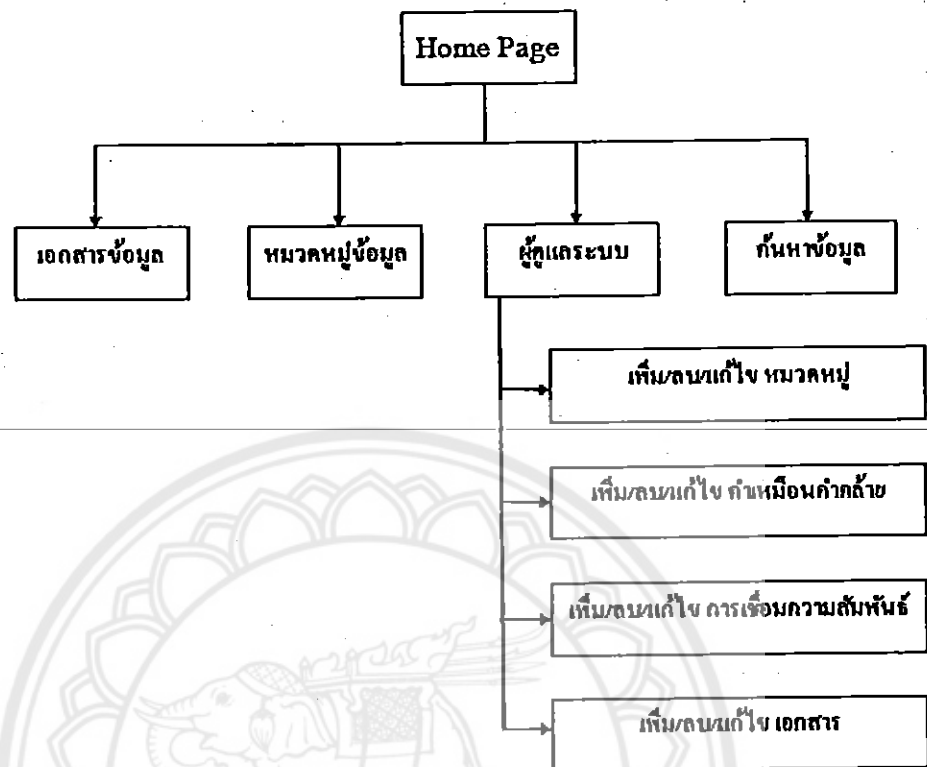
ชื่อ field	ชนิดข้อมูล	คำอธิบาย
MemberID (PK)	Int,not null	ID ของสมาชิก
MemberUser	Nvarchar(max), null	Username
MemberPassword	Nvarchar(max), null	Password
MemberFirstName	Nvarchar(max), null	ชื่อจริงของสมาชิก ใช้แสดงหลังจาก Login ผ่าน
MemberLastname	Nvarchar(max), null	นามสกุลของสมาชิก ใช้แสดงหลังจาก Login ผ่าน
MemberActive	Int,null	สถานะถูกใช้งาน 1 ถูกใช้งาน 0 ไม่ถูกใช้งาน

tb_SpellCheck : ตารางเก็บข้อมูลคำคล้าย
 ตารางที่ 3.5 ตารางเก็บข้อมูลคำคล้าย

ชื่อ field	ชนิดข้อมูล	คำอธิบาย
SpellCheckID (PK)	Int,not null	ID ของคำที่ใช้ในการตรวจสอบกับเนื้อหาของเอกสารที่เพิ่ม
CategoryID	Int,not null	ID ของหมวดหมู่สถานที่ท่องเที่ยว/กิจกรรมท่องเที่ยว
SpellCheckName	Nvarchar(max), null	คำที่ใช้ในการตรวจสอบกับเนื้อหาของเอกสารที่เพิ่ม
SpellCheckKeyword	Nvarchar(max), null	คำที่ต้องการให้แสดงเพิ่มเติม เชื่อมโยงกับ SpellCheckName
SpellCheckActive	Int,null	สถานะถูกใช้งาน 1 ถูกใช้งาน 0 ไม่ถูกใช้งาน

3.3 การออกแบบเว็บไซต์

ในการทำเว็บไซต์จะใช้ ASP.NET และ ภาษา C# โดยใช้การเขียนผ่าน โปรแกรม Microsoft Visual Studio 2010 ซึ่งเป็นโปรแกรมที่มีชุดเครื่องมือพัฒนาที่ถูกออกแบบมาเพื่อช่วยเหลือนักพัฒนาซอฟต์แวร์ในด้านต่างๆ รวมถึงด้านการพัฒนาเว็บแอปพลิเคชัน จากนั้นทำการร่างโครงสร้างของเว็บไซต์ให้สอดคล้องกับระบบของข้อมูลและการใช้งานให้ใช้งานได้ง่ายเหมาะสมสำหรับทุกคนที่เข้ามาใช้งานดังนี้



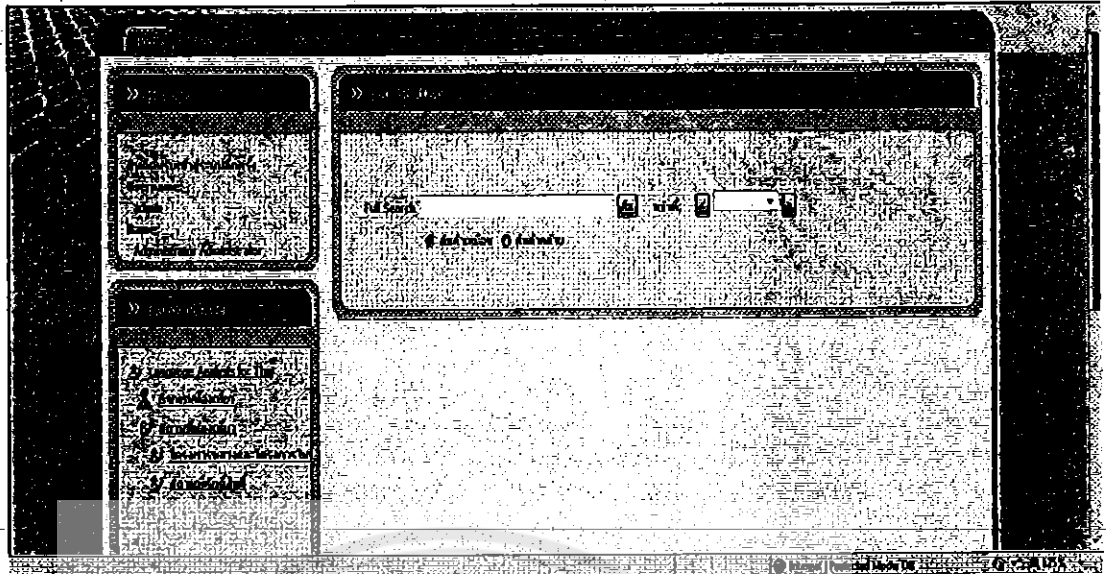
รูปที่ 3.19 รูปแสดง โครงสร้างของเว็บไซต์จัดหมวดหมู่อัตโนมัติสำหรับการท่องเที่ยว

ขั้นตอนการใช้งานเว็บไซต์

หลังจากออกแบบโครงสร้างของเว็บไซต์แล้วทำให้ได้ส่วนต่างๆของเว็บไซต์ตามที่ร่างไว้เบื้องต้น จะมีหลักการทำงานและวิธีการใช้งานดังนี้

3.3.1 หน้าหลักของเว็บไซต์จัดหมวดหมู่อัตโนมัติสำหรับการท่องเที่ยว

หน้าหลักจะมีหน้าตาและองค์ประกอบทั้งหมดดังนี้



รูปที่ 3.20 รูปแสดงหน้าหลักเว็บไซต์จัดหมวดหมู่อัตโนมัติสำหรับการท่องเที่ยว

จากรูปที่เห็นคือหน้าหลักของเว็บไซต์จัดหมวดหมู่อัตโนมัติสำหรับการท่องเที่ยว เมนูด้านซ้ายจะประกอบไปด้วย

- เมนู HOME
เมนูนี้จะใช้สำหรับการเรียกกลับไปยังหน้าแรกของเว็บไซต์
- เมนู Administrator
เมนูผู้ดูแลระบบ ใช้สำหรับผู้ดูแลเว็บไซต์เพื่อเข้าไปแก้ไขข้อมูลต่างๆ อัพเดทเว็บไซต์ เพิ่มข้อมูล หรือลบข้อมูลต่างๆภายในเว็บไซต์โดยจะต้องมีการยืนยันตัวตนโดยการต้องผ่านการ Login จะต้องกรอก Username และ Password ให้ถูกต้องจึงจะมีสิทธิ์เข้าไปได้
- เมนู about
เมนู about ใช้สำหรับการเรียกดูข้อมูลรายละเอียดของผู้พัฒนาเว็บไซต์
- เมนู CategoryName
เมนู CategoryName ใช้สำหรับการค้นหาเอกสารที่ตรงกับหมวดหมู่

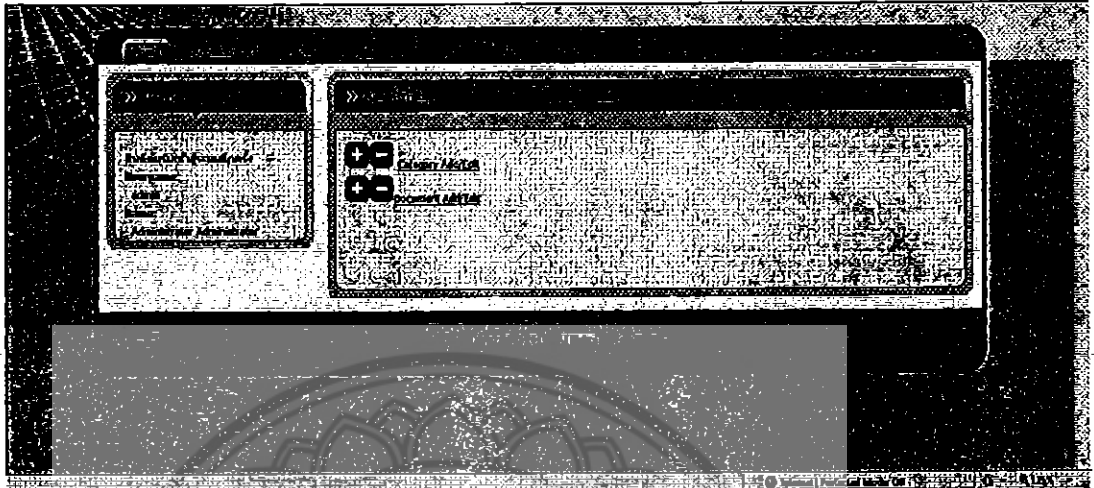
3.3.2 หน้า Administrator

เป็นส่วนของผู้ดูแลระบบเพื่อเข้ามาทำการปรับปรุงเว็บไซต์โดยสิ่งที่สามารถปรับปรุงได้จะมีคังเมนูต่อไปนี้

- เมนู Category Add/Edit
เมนูนี้จะใช้สำหรับการจัดการในส่วนของหมวดหมู่คือ เพิ่ม ลบ แก้ไข หมวดหมู่ เพิ่ม ลบ แก้ไข คำเหมือนคำคล้าย เพิ่ม ลบ แก้ไข ความสำคัญของข้อมูล

- เมนู Document Add/Edit

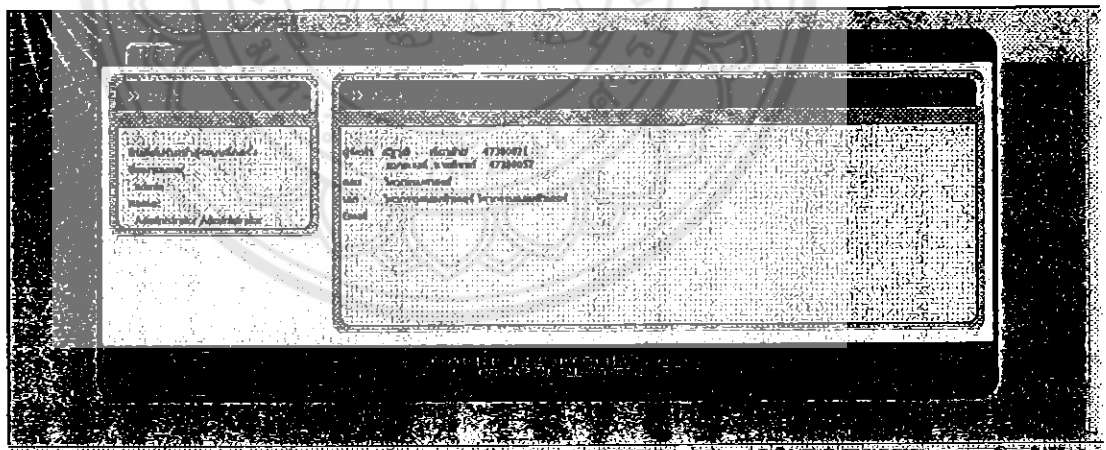
เมนูนี้จะใช้สำหรับการจัดการ ในส่วนของเอกสารข้อมูลคือ เพิ่ม ลบ แก้ไข เอกสาร



รูปที่ 3.21 รูปแสดงหน้า Administrator

3.3.3 หน้า About

เป็นหน้าที่แสดงข้อมูลรายละเอียดของผู้พัฒนาเว็บไซต์ เช่น ผู้จัดทำ คณะ สาขาวิชา



รูปที่ 3.22 รูปแสดงหน้า About

3.3.4 เมนู Category Add/Edit

เมนูนี้จะใช้สำหรับการจัดการ ในส่วนของหมวดหมู่คือ เพิ่ม แก้ไข หมวดหมู่ เพิ่ม แก้ไข คำเหมือนคำคล้าย เพิ่ม แก้ไข ความสัมพันธ์ของข้อมูล

- ส่วนเพิ่มหมวดหมู่

ส่วนนี้จะเป็นการเพิ่มหมวดหมู่เข้าไปในฐานข้อมูล โดยผ่านหน้าเว็บไซต์

The screenshot shows a web application interface for adding a category. On the left is a sidebar with a tree view of categories. The main area contains a form with the following fields and controls:

- CategoryID:** A text input field.
- CategoryName:** A text input field.
- ParentCategoryID:** A dropdown menu with "Root" selected.
- CategoryList:** A dropdown menu showing a tree view of categories, including "Language Analysis for The Administration", "ภาษาเขียน", "ภาษาพูด", "ภาษาเขียน-คำขวัญ", and "ภาษาพูด".
- CategoryActive:** A dropdown menu with "Show" selected.
- Buttons:** "Add", "Update", and "Cancel" buttons.

รูปที่ 3.23 รูปแสดงส่วนเพิ่มหมวดหมู่

โดยหน้าของการเพิ่มข้อมูลจะมีช่องให้กรอกข้อมูลต่างๆที่จะใช้เป็นหัวข้อหมวดหมู่ รายละเอียดต่างๆดังนี้

CategoryID: IDของหมวดหมู่ที่จะเพิ่มในฐานข้อมูล ไม่สามารถกรอกได้ เมื่อทำการเพิ่มข้อมูลจะใส่หมายเลขIDให้อัตโนมัติ

CategoryName: ชื่อที่จะใช้เป็นหัวข้อหมวดหมู่

ParentCategoryID: เป็นการเลือกว่าจะให้หัวข้อหมวดหมู่ใหม่นั้นอยู่ภายใต้หัวข้อหมวดหมู่ที่มีอยู่เดิมตรงตำแหน่งไหน ถ้าเลือก Root หัวข้อหมวดหมู่ใหม่จะไม่อยู่ภายใต้หัวข้อหมวดหมู่เดิม ถ้าเลือก list อื่นๆที่ปรากฏ หัวข้อหมวดหมู่ใหม่ที่จะเพิ่มจะไปอยู่ในหัวข้อหมวดหมู่ที่เลือก

CategoryActive: เลือกว่าจะใช้งานหัวข้อให้แสดงหรือไม่แสดงบน CategoryName ด้านซ้ายมือ เมื่อกรอกข้อมูลทั้งหมดเสร็จสิ้นแล้วให้คลิกปุ่ม "บันทึก" เพื่อบันทึกข้อมูลลงฐานข้อมูลเป็นอันเสร็จสิ้น

ส่วนแก้ไขหมวดหมู่

ส่วนนี้จะเป็นการแก้ไขข้อมูลหมวดหมู่แล้วทำการจัดเก็บในฐานข้อมูลโดยผ่านทางเว็บไซต์

รูปที่ 3.24 รูปแสดงส่วนแก้ไขหมวดหมู่

หน้าของการแก้ไขข้อมูลจะมี list หัวข้อหมวดหมู่ตรง CategoryName ให้เลือกข้อมูลที่จะแก้ไข รายละเอียดต่าง ๆ ดังนี้

CategoryID: ID ของหมวดหมู่ที่จะเพิ่มในฐานข้อมูล ไม่สามารถกรอกได้ เมื่อทำการเพิ่มข้อมูลจะใส่หมายเลข ID ให้อัตโนมัติ

CategoryName: ชื่อที่จะแก้ไข ทำการใส่ข้อมูลทับลงไปแทนชื่อเดิม ในกรณีที่จะแก้ไขชื่อหมวดหมู่

ParentCategoryID: เป็นการเลือกที่จะให้หัวข้ออยู่ภายใต้หมวดหมู่ไหน ถ้าต้องการเปลี่ยนสามารถเลือกตาม list ที่มีเพื่อเปลี่ยนหมวดหมู่

CategoryActive: เลือกที่จะใช้งานหัวข้อให้แสดงหรือไม่แสดงบน CategoryName ด้านซ้ายมือ เมื่อกรอกข้อมูลทั้งหมดเสร็จสิ้นแล้วให้กดปุ่ม “บันทึก” เพื่อบันทึกข้อมูลลงฐานข้อมูลเป็นอันเสร็จสิ้น

- ส่วนเพิ่มคำเหมือนคำคล้าย

- จากรูปที่ 3.25 สามารถเพิ่มคำเหมือนคำคล้ายให้กับหัวข้อที่เลือกได้โดยการเลือก SpellCheckMap เพื่อไปดูหน้าตาใหม่

SpellCheckID	CategoryName	SpellCheckName	SpellCheckKeyword	SpellCheckActive	เปิด
313	พจนานุกรมภาษาไทย	พจนานุกรมภาษาไทย (พจนานุกรมราชบัณฑิตยสถาน)			เปิด
314	พจนานุกรมภาษาไทย	พจนานุกรมภาษาไทย			เปิด
315	พจนานุกรมภาษาไทย	พจนานุกรมภาษาไทย (พจนานุกรมราชบัณฑิตยสถาน)			เปิด
316	พจนานุกรมภาษาไทย	พจนานุกรมภาษาไทย (พจนานุกรมราชบัณฑิตยสถาน)			เปิด
317	พจนานุกรมภาษาไทย	พจนานุกรมภาษาไทย			เปิด
318	พจนานุกรมภาษาไทย	พจนานุกรมภาษาไทย (พจนานุกรมราชบัณฑิตยสถาน)			เปิด
319	พจนานุกรมภาษาไทย	พจนานุกรมภาษาไทย (พจนานุกรมราชบัณฑิตยสถาน)			เปิด
320	พจนานุกรมภาษาไทย	พจนานุกรมภาษาไทย (พจนานุกรมราชบัณฑิตยสถาน)			เปิด
321	พจนานุกรมภาษาไทย	พจนานุกรมภาษาไทย			เปิด

SpellCheckID	123
CategoryID	ภาษาไทย
SpellCheckName	พจนานุกรมภาษาไทย
SpellCheckKeyword	
SpellCheckActive	เปิด

รูปที่ 3.25 รูปแสดงส่วนเพิ่มคำเหมือนคำคล้าย

โดยหน้าของการเพิ่มคำเหมือนคำคล้ายจะมีช่องให้กรอกข้อมูลรายละเอียดต่างๆดังนี้

SpellCheckID: ID ของหมวดหมู่ในฐานะข้อมูล ของหมวดหมู่ที่จะทำการเพิ่มคำเหมือนคำคล้าย

CategoryID: แสดงว่าหมวดหมู่ที่เลือกอยู่ภายใต้หมวดหมู่ไหน

SpellCheckName: ใส่คำเหมือนคำคล้ายที่ต้องการ

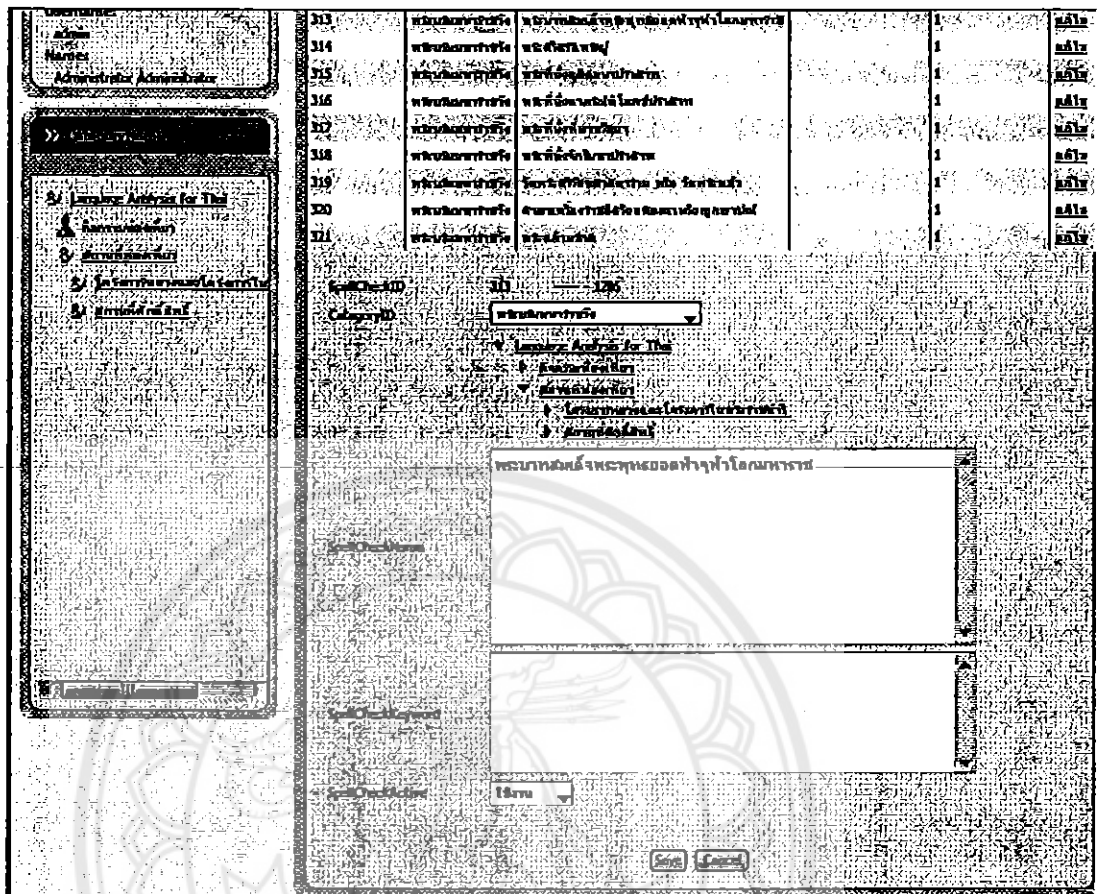
SpellCheckKeyword: ใส่คำอธิบายที่ต้องการให้แสดง (สำหรับการค้นหาแบบการค้นหาธรรมชาติ)

CategoryActive: เลือกว่าจะใช้งานหรือไม่ใช้งาน

เมื่อกรอกข้อมูลทั้งหมดเสร็จสิ้นแล้วให้คลิกปุ่ม "Save" เพื่อบันทึกข้อมูลลงฐานข้อมูลเป็นอันเสร็จสิ้น

- ส่วนแก้ไขคำเหมือนคำคล้าย

จากรูปที่ 3.26 สามารถแก้ไขคำเหมือนคำคล้ายให้กับหัวข้อที่เลือกได้โดยการเลือกแก้ไขคำเหมือนคำคล้ายที่ต้องการแก้ไข



รูปที่ 3.26 รูปแสดงส่วนแก้ไขคำเหมือนคำคล้าย

โดยหน้าของการแก้ไขคำเหมือนคำคล้ายจะมีช่องให้เลือกแก้ไขแล้วกรอกข้อมูลรายละเอียดต่างๆดังนี้

แก้ไข : เลือกคำที่ต้องการแก้ไข

SpellCheckID : IDของหมวดหมู่ในฐานข้อมูล

CategoryID : แสดงว่าหมวดหมู่ที่เลือกอยู่ภายใต้หมวดหมู่ไหน

SpellCheckName : ใส่คำเหมือนคำคล้ายที่ต้องการแก้ไข

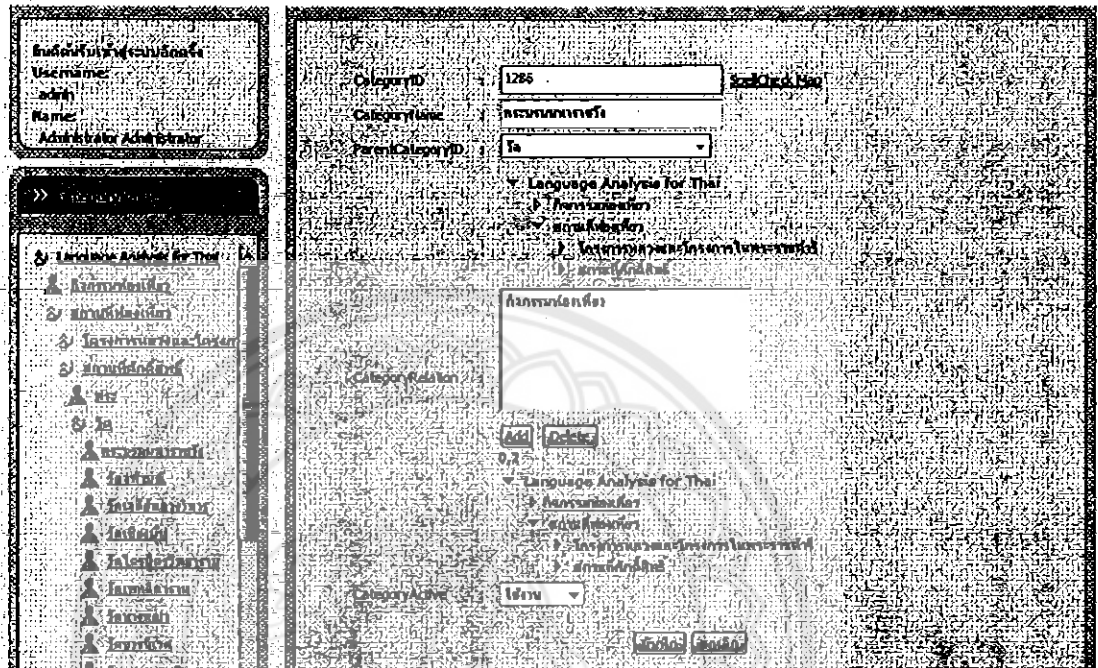
SpellCheckKeyword : ใส่คำอธิบายที่ต้องการให้แสดง (สำหรับการค้นหาแบบการค้นหาธรรมดา)

CategoryActive: เลือกว่าจะใช้งานหรือไม่ใช้งาน

เมื่อกรอกข้อมูลทั้งหมดเสร็จสิ้นแล้วให้คลิกปุ่ม "Save" เพื่อบันทึกข้อมูลลงฐานข้อมูลเป็นอันเสร็จสิ้น

ส่วนเพิ่มและลบความสัมพันธ์ระหว่างหมวดหมู่

ส่วนนี้จะเป็นการเพิ่มความสัมพันธ์ระหว่างหมวดหมู่แล้วเก็บในฐานข้อมูลโดยผ่านหน้าเว็บไซต์



รูปที่ 3.27 รูปแสดงส่วนเพิ่มและลบความสัมพันธ์ระหว่างหมวดหมู่

โดยหน้าของการเพิ่มและลบความสัมพันธ์ระหว่างหมวดหมู่จะมีlist หัวข้อหมวดหมู่ตรง CategoryName ให้เลือกข้อมูลที่จะเพิ่มหรือลบความสัมพันธ์ระหว่างหมวดหมู่ รายละเอียดดังนี้

CategoryID : IDของหมวดหมู่ในฐานข้อมูล ไม่สามารถกรอกได้ เมื่อทำการเพิ่มข้อมูลจะได้อัตโนมัติ

CategoryName : ชื่อหมวดหมู่ที่จะทำการเพิ่มความสัมพันธ์

CategoryRelation : เพิ่มหรือลบความสัมพันธ์ของหมวดหมู่ที่เลือกใน CategoryName โดยเลือกจากหมวดหมู่ข้างล่าง ถ้าเพิ่มข้อมูลให้ทำการกด Add ถ้าต้องการลบข้อมูลให้กดที่หมวดหมู่ในกล่อง CategoryRelation แล้วกด Delete

CategoryActive : เลือกว่าจะใช้งานหัวข้อให้แสดงหรือไม่แสดงบน CategoryName ด้านซ้ายมือ เมื่อกรอกข้อมูลทั้งหมดเสร็จสิ้นแล้วให้กดปุ่ม “บันทึก” เพื่อบันทึกข้อมูลลงฐานข้อมูลเป็นอันเสร็จสิ้น

3.3.5 เมนู Document Add/Edit

เมนูนี้จะใช้สำหรับการจัดการในส่วนของเอกสารคือ เพิ่ม แก้ไข เอกสาร

- ส่วนเพิ่มเอกสาร

ส่วนนี้จะเป็นการเพิ่มเอกสารในฐานข้อมูลโดยผ่านหน้าเว็บไซต์

รูปที่ 3.28 รูปแสดงส่วนเพิ่มเอกสาร

โดยหน้าของการเพิ่มเอกสารจะมีช่องให้กรอกข้อมูลต่างๆรายละเอียดดังนี้

DocumentID: IDของเอกสารที่จะเพิ่มในฐานข้อมูล ไม่สามารถกรอกได้ เมื่อทำการเพิ่มข้อมูลจะได้หมายเลขIDให้อัตโนมัติ

DocumentName: ชื่อเอกสาร

DocumentDetail: รายละเอียดข้อมูลของเอกสาร

ปุ่ม Analysis: เป็นปุ่มที่เอาไว้วิเคราะห์ข้อมูลจาก DocumentName, DocumentDetail เพื่อเปรียบเทียบกับ CategoryName, SpellCheckName เพื่อทำการจัดหมวดหมู่ข้อมูล

CategoryIndexSearch: เมื่อคลิกปุ่ม Analysis ข้อมูลที่ได้จากการจัดหมวดหมู่จะปรากฏตรงกล่อง CategoryIndexSearch เพื่อบอกว่าข้อมูลที่วิเคราะห์ได้ตรงกับหมวดหมู่ไหน โดยแสดงเป็น ID ของ Category

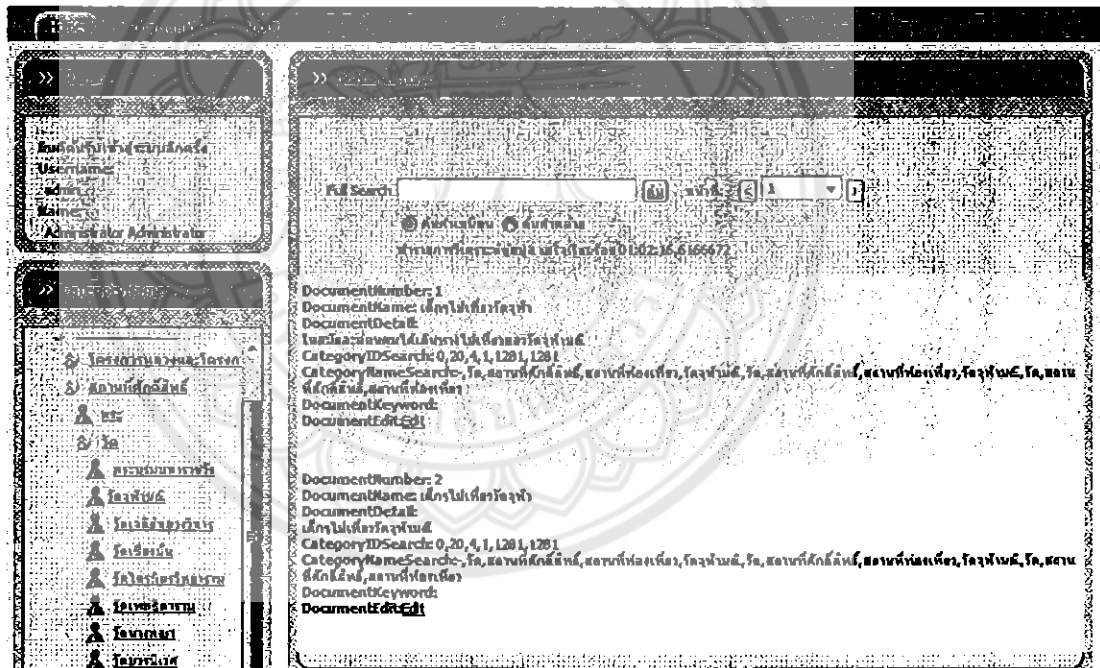
Text Analysis: แปลง ID ของ CategoryID ใน CategoryIndexSearch เป็นข้อความ

DocumentKeyword: เพิ่มคำอธิบายให้กับเอกสาร

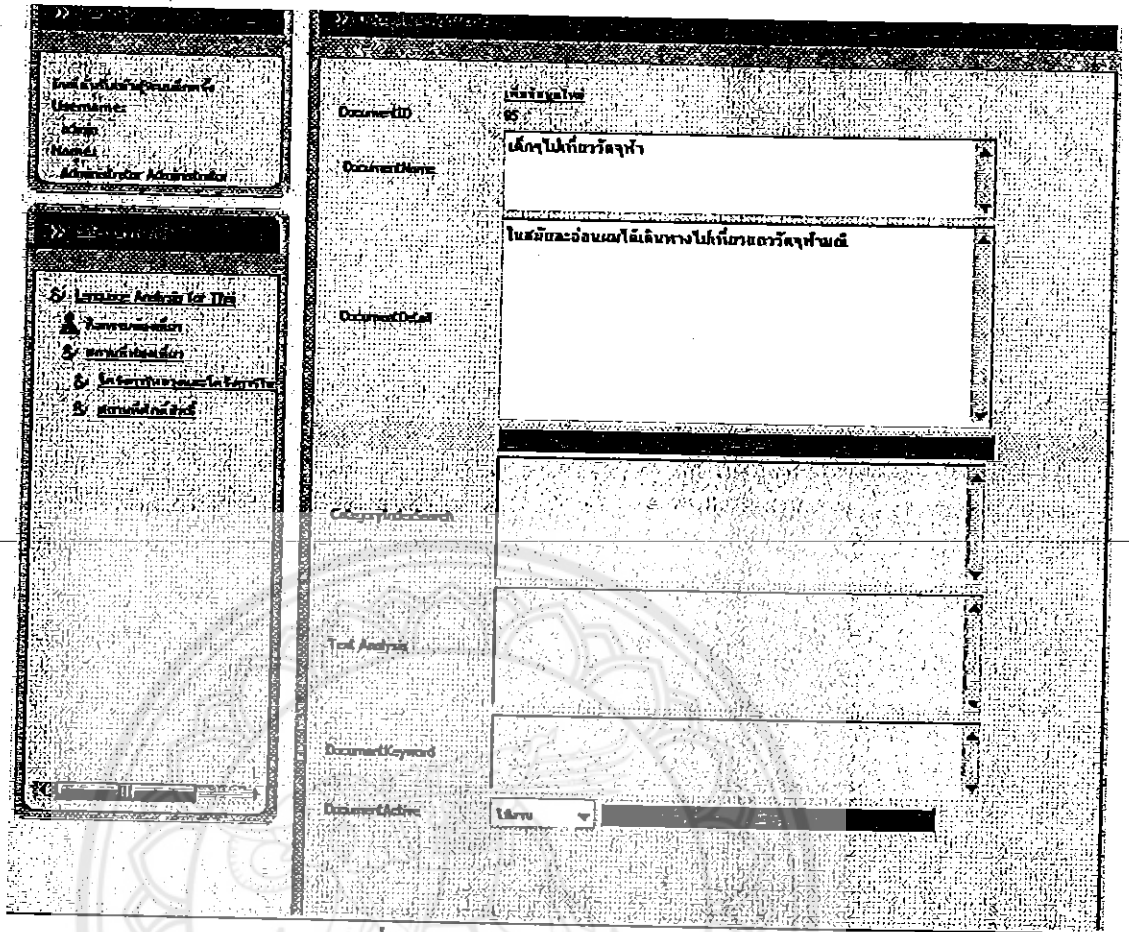
CategoryActive: เลือกว่าจะใช้งานหัวข้อให้แสดงหรือไม่แสดงบน CategoryName ด้านซ้ายมือ เมื่อรอกข้อมูลทั้งหมดเสร็จสิ้นแล้วให้คลิกปุ่ม "Save" เพื่อบันทึกข้อมูลลงฐานข้อมูลเป็นอันเสร็จสิ้น

- ส่วนแก้ไขเอกสาร

ส่วนนี้จะเป็นการแก้ไขเอกสารในฐานข้อมูลโดยผ่านหน้าเว็บไซต์



รูปที่ 3.29 รูปแสดงส่วนแก้ไขเอกสาร 1



รูปที่ 3.30 รูปแสดงส่วนแก้ไขเอกสาร 2

จากรูป 3.30 สามารถแก้ไขเอกสาร โดยเลือกที่หมวดหมู่จะปรากฏเอกสารที่เกี่ยวข้อง เลือก Edit ตรง DocumentEdit สำหรับเอกสารที่ต้องการแก้ไข จะเข้าสู่หน้าแก้ไขเอกสารตามรูป4.11 เลือกแก้ไขเอกสารในส่วนที่ปรากฏ

DocumentID: IDของเอกสารที่จะเพิ่มในฐานข้อมูล ไม่สามารถกรอกได้ เมื่อทำการเพิ่มข้อมูลจะใส่หมายเลขIDให้อัตโนมัติ

DocumentName: ชื่อเอกสารที่ต้องการแก้ไข

DocumentDetail: รายละเอียดข้อมูลของเอกสารที่ต้องการแก้ไข

ปุ่มAnalysis : เป็นปุ่มที่เอาไว้วิเคราะห์ข้อมูลจาก DocumentName , DocumentDetail เพื่อเปรียบเทียบกับ CategoryName , SpellCheckName เพื่อทำการจัดหมวดหมู่ข้อมูล

CategoryIndexSearch: เมื่อกดปุ่มAnalysis ข้อมูลที่ได้จากการจัดหมวดหมู่จะปรากฏตรงกล่อง CategoryIndexSearch เพื่อบอกว่าข้อมูลที่วิเคราะห์ได้ตรงกับหมวดหมู่ไหน โดยแสดงเป็นID ของ Category

Text Analysis: แปลงID ของ CategoryID ใน CategoryIndexSearch เป็นข้อความ

DocumentKeyword: เพิ่มคำอธิบายให้กับเอกสาร

CategoryActive: เลือกว่าจะใช้งานหัวข้อให้แสดงหรือไม่แสดงบน CategoryName ด้านซ้ายมือ
เมื่อกรอกข้อมูลทั้งหมดเสร็จสิ้นแล้วให้กดปุ่ม "Save" เพื่อบันทึกข้อมูลลงฐานข้อมูลเป็นอันเสร็จ
สิ้น



บทที่ 4

ผลการทดลอง

4.1 ผลการทำงานของระบบ

ตัวอย่างการใช้งานเว็บไซต์จัดหมวดหมู่อัตโนมัติสำหรับการท่องเที่ยว

ทำการเพิ่มหมวดหมู่ สำหรับใช้ทำเป็นกลุ่มตัวอย่าง

พระบรมมหาราชวัง

วัดจุฬามณี

วัดเจดีย์หลวงวิหาร

วัดเชียงมั่น

วัดไคร่มิตรวิทยาราม

วัดเทพธิดาราม

วัดนางพญา

วัดบวรนิเวศ

วัดเบญจมบพิตร

วัดปราสาท

วัดพระศรีมหาธาตุวรมหาวิหาร

วัดพระสิงห์วรวิหาร

วัดมหาธาตุ

วัดราชคี่ร์หิรัญยาราม

วัดราชนัคคาราม

วัดราชบพิตร

วัดราชบูรณะ

วัดราชบูรณะหรือวัดเลียบ (กรุงเทพฯ)

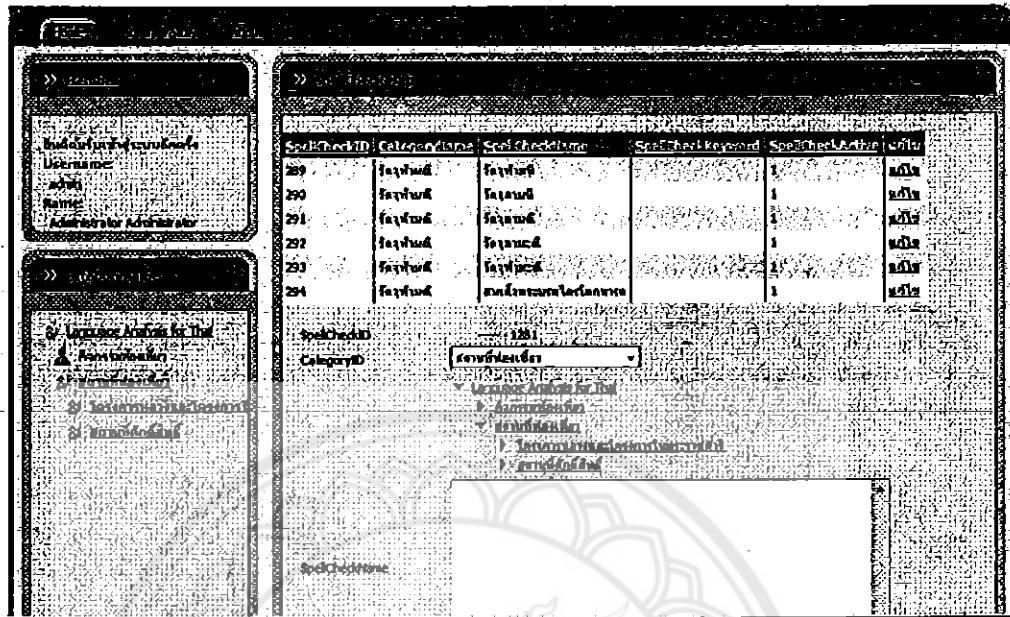
วัดสระเกศและภูเขาทอง

วัดสวนคอก

ทำการเพิ่ม Keyword หมวดหมู่ตัวอย่างตามหัวข้อ 4.2.4 เมนูCategory Add/Edit

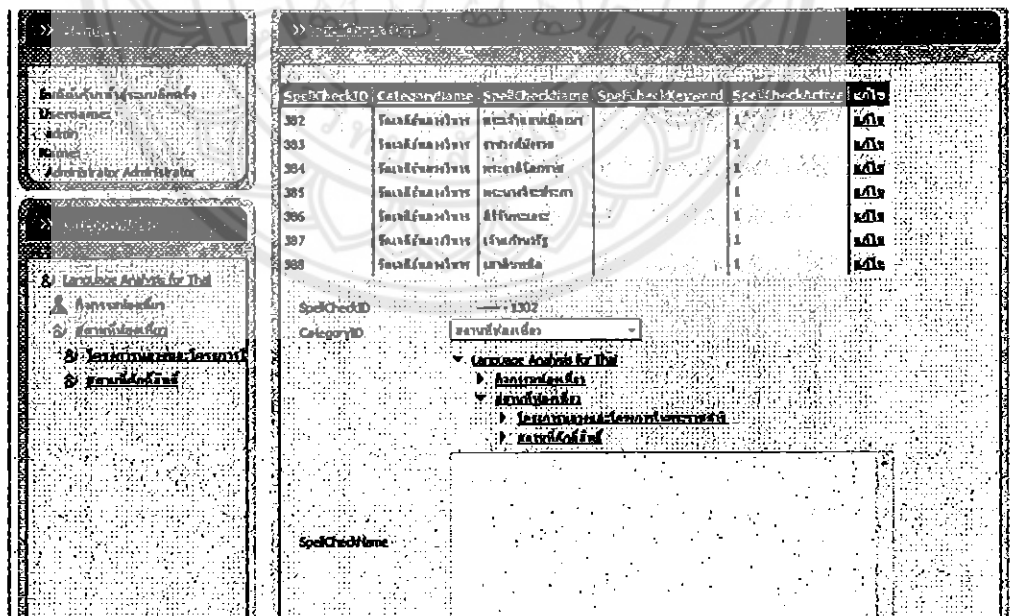
จะได้ผลลัพธ์ตามรูปที่ 4.1

ทำการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดจุฬามณี จะได้ผลลัพธ์ตามรูปที่ 4.3



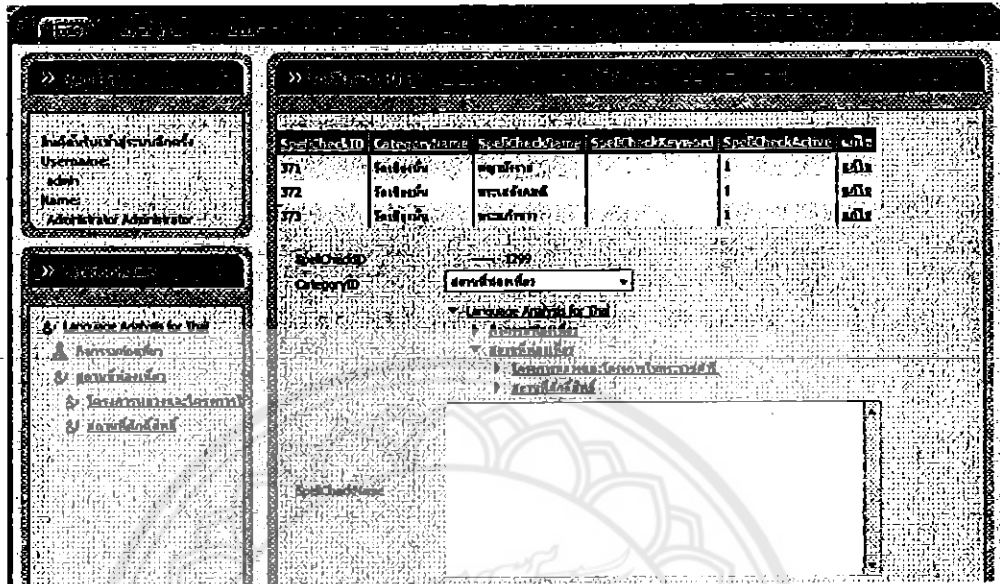
รูปที่ 4.3 รูปแสดงการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดจุฬามณี

ทำการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดเจดีย์หลวงวิหาร จะได้ผลลัพธ์ตามรูปที่ 4.4



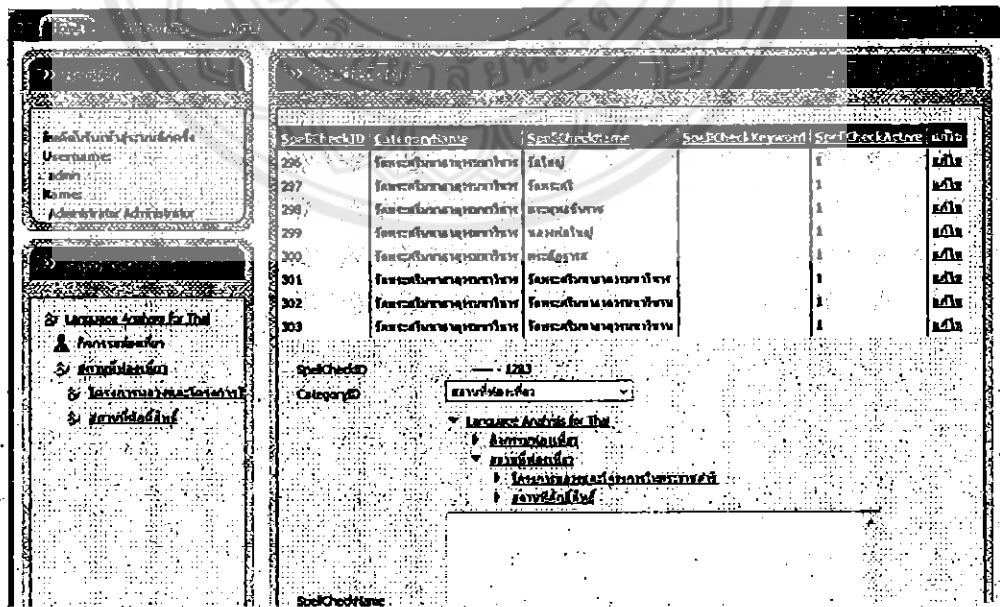
รูปที่ 4.4 รูปแสดงการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดเจดีย์หลวงวิหาร

ทำการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดเชิงม้น จะได้ผลลัพธ์ตามรูปที่ 4.5



รูปที่ 4.5 รูปแสดงการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดเชิงม้น

ทำการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดพระศรีมหาธาตุวรวิหาร จะได้ผลลัพธ์ตามรูปที่ 4.6



รูปที่ 4.6 รูปแสดงการเพิ่มคำเหมือนคำคล้ายของหมวดหมู่วัดพระศรีมหาธาตุวรวิหาร

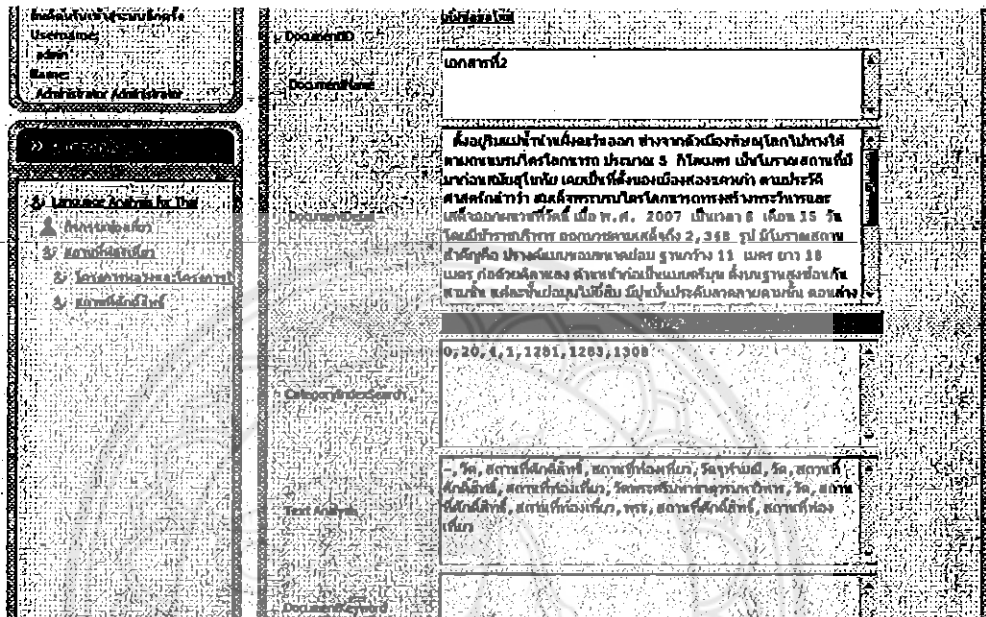
ทำการเพิ่มเอกสารที่ต้องการเพื่อให้แอปพลิเคชันทำการ Analysis เพื่อจัดหมวดหมู่
อัตโนมัติเลือกเอกสารตัวอย่าง 5 เอกสารเพื่อทำการจัดหมวดหมู่

เอกสารที่ 1 พระบรมมหาราชวัง เมื่อรัชกาลที่ 1 ทรงเสด็จขึ้นครองราชย์เป็นปฐมกษัตริย์
แห่งพระบรมราชจักรีวงศ์ในปี 2325 ทรงให้ย้ายราชธานีจากฝั่งธนบุรีมายังฝั่งพระนครจากนั้นทรง
โปรดเกล้าฯ ให้สร้างพระบรมมหาราชวังเป็นศูนย์กลางสำหรับพระนครใหม่ ภายใน
พระบรมมหาราชวังประกอบด้วยหมู่พระมหามนเทียรพระมหาปราสาท พระที่นั่งหอและอาคาร
อื่นๆซึ่งมีการรื้อถอน ซ่อมแซม และต่อเติมแก้ไขเปลี่ยนแปลง หรือสร้างใหม่บ้างตามความจำเป็น
และเหมาะสมในแต่ละรัชสมัย

เอกสารที่ 2 ตั้งอยู่ริมแม่น้ำน่านฝั่งตะวันออก ห่างจากตัวเมืองพิษณุโลกไปทางใต้ตามถนน
บรมไตรโลกนารถ ประมาณ 5 กิโลเมตร เป็นโบราณสถานที่มีมาก่อนสมัยสุโขทัย เคยเป็นที่ตั้งของ
เมืองสองแควเก่า ตามประวัติศาสตร์กล่าวว่า สมเด็จพระบรมไตรโลกนารถทรงสร้างพระวิหารและ
เสด็จออกผนวชที่วัดนี้ เมื่อ พ.ศ. 2007 เป็นเวลา 8 เดือน 15 วัน โดยมีข้าราชการบริพาร ออกบวชตาม
เสด็จถึง 2,348 รูป มีโบราณสถานสำคัญคือ ปรากฏแบบขอมขนาดย่อม ฐานกว้าง 11 เมตร ยาว 18
เมตร ก่อด้วยศิลาแลง ด้านหน้าก่อเป็นแบบศรีมูขุ ตั้งบนฐานสูงซ้อนกันสามชั้น แต่ละชั้นย่อมุมไม้
ยี่สิบ มีปูนปั้นประดับลวดลายตามชั้น คอนล่างแถบหน้ากระดานและบัวหน้ากระดานเป็นลายหงส์
เหมือนกับองค์ปราสาทที่วัดพระศรีรัตนมหาธาตุ จังหวัดลพบุรี สมัยที่ยังสมบูรณ์อยู่มีกำแพงแก้ว
ล้อมรอบ ใกล้เคียงกันมีมณฑปพระพุทธบาทจำลองซึ่งสมเด็จพระนารายณ์มหาราชได้โปรดให้
สร้างขึ้น แผ่นจารึกหน้ามณฑปมีใจความสรุปได้ว่า เมื่อ พ.ศ. 2221 สมเด็จพระนารายณ์มหาราชทรง
มีพระบรมราชโองการให้ใช้ผ้าทาบรอยพระพุทธบาท สลักลงบนแผ่นหิน พระราชทานไว้เป็นที่
กราบไหว้ของฝูงชน

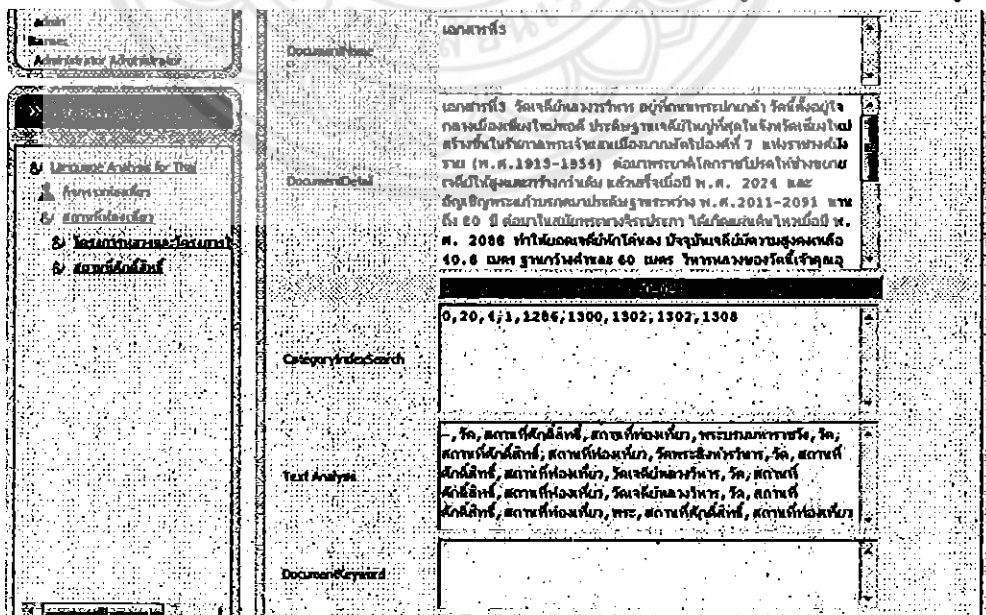
เอกสารที่ 3 วัดเจดีย์หลวงวิหาร อยู่ที่ถนนพระปกเกล้า วัดนี้ตั้งอยู่ใจกลางเมืองเชียงใหม่
พอดี ประดิษฐานเจดีย์ใหญ่ที่สุดในจังหวัดเชียงใหม่ สร้างขึ้นในรัชกาลพระเจ้าแสนเมืองมหายศ
องค์ที่ 7 แห่งราชวงศ์มังราย (พ.ศ.1913-1954) ต่อมาพระยาติโลกราชโปรดให้ช่างขยายเจดีย์ให้สูง
และกว้างกว่าเดิม แล้วเสร็จเมื่อปี พ.ศ. 2024 และอัญเชิญพระแก้วมรกตมาประดิษฐานระหว่าง พ.ศ.
2011-2091 นานถึง 80 ปี ต่อมาในสมัยพระนางจิระประภา ได้เกิดแผ่นดินไหวเมื่อปี พ.ศ. 2088 ทำ
ให้ยอดเจดีย์หัก โคนลง ปัจจุบันเจดีย์มีความสูงคงเหลือ 40.8 เมตร ฐานกว้างด้านละ 60 เมตร วิหาร
หลวงของวัดนี้เจ้าคุณอุบาลีคุณูปมาจารย์ (สิริจันทะเถระ) และเจ้าแก้วรัฐเป็นผู้สร้างขึ้นในปี พ.ศ.
2471 หน้าประตูทางเข้าวิหาร มีบันไดนาคเลื้อยลงมาถึง ใช้ทางเกี่ยวกระหวัดขึ้นไปเป็นซุ้มประตู
วิหาร นาคคู่นี้เป็นฝีมือเก่าแก่ที่มีมาตั้งแต่เดิมได้ชื่อว่าเป็นนาคที่สวยงามที่สุดของภาคเหนือ ในวัดเจดีย์
หลวงนี้ยังมี เสาอินทขิล หรือ เสาหลักเมือง สร้างขึ้นเมื่อครั้งพ่อขุนเม็งรายมหาราชสร้างเมือง
เชียงใหม่ เมื่อปี พ.ศ. 1839 ประดิษฐานอยู่ในวิหารจตุรมุขทรงไทยหลังเตี้ย เสาอินทขิลนี้สร้างด้วย

ทำการเพิ่มเอกสารที่ 2 และทำการวิเคราะห์เพื่อจัดหมวดหมู่ จะได้ผลลัพธ์ตามรูปที่ 4.8



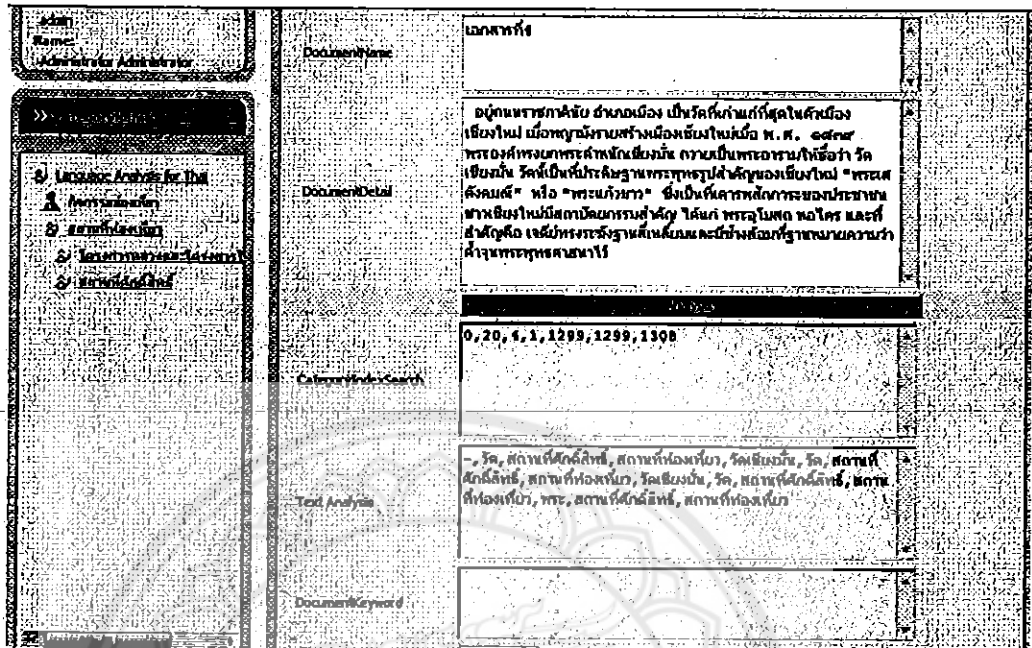
รูปที่ 4.8 รูปแสดงการเพิ่มเอกสารที่ 2 และทำการวิเคราะห์ข้อมูลเพื่อจัดหมวดหมู่

ทำการเพิ่มเอกสารที่ 3 และทำการวิเคราะห์เพื่อจัดหมวดหมู่ จะได้ผลลัพธ์ตามรูปที่ 4.9



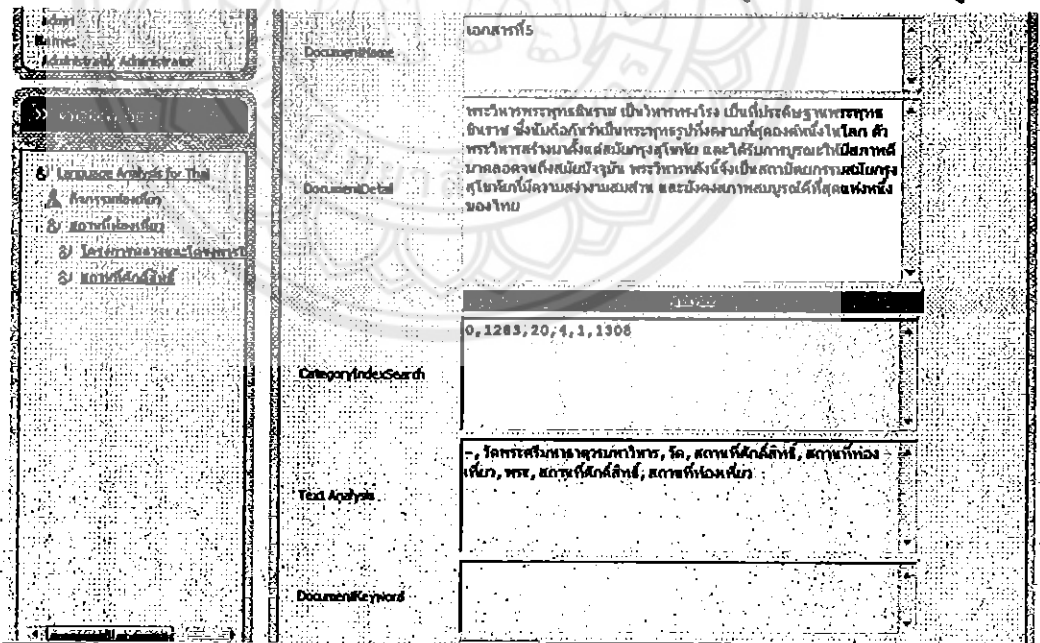
รูปที่ 4.9 รูปแสดงการเพิ่มเอกสารที่ 3 และทำการวิเคราะห์ข้อมูลเพื่อจัดหมวดหมู่

ทำการเพิ่มเอกสารที่ 4 และทำการวิเคราะห์เพื่อจัดหมวดหมู่ จะได้ผลลัพธ์ตามรูปที่ 4.10



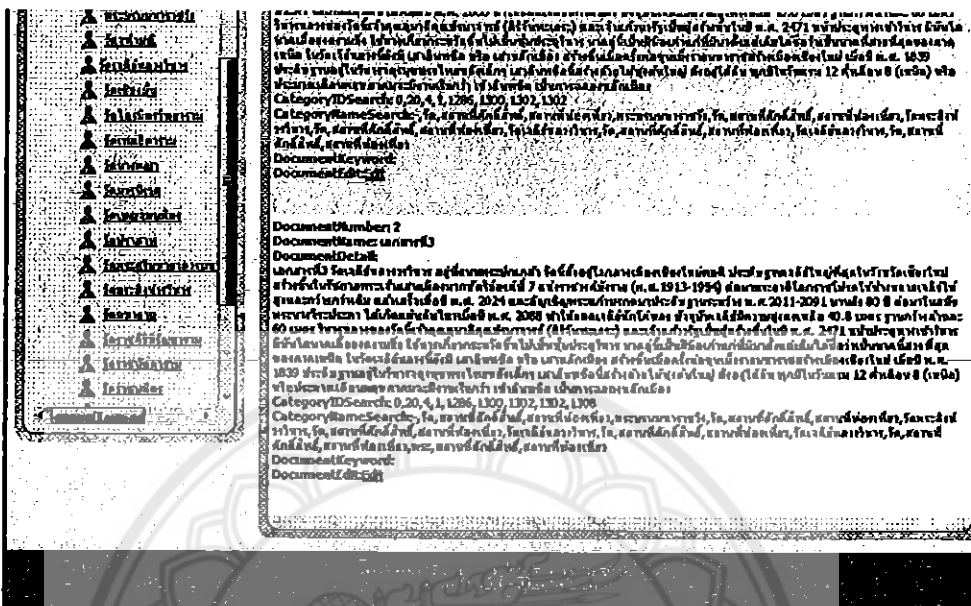
รูปที่ 4.10 รูปแสดงการเพิ่มเอกสารที่ 4 และทำการวิเคราะห์ข้อมูลเพื่อจัดหมวดหมู่

ทำการเพิ่มเอกสารที่ 5 และทำการวิเคราะห์เพื่อจัดหมวดหมู่ จะได้ผลลัพธ์ตามรูปที่ 4.11



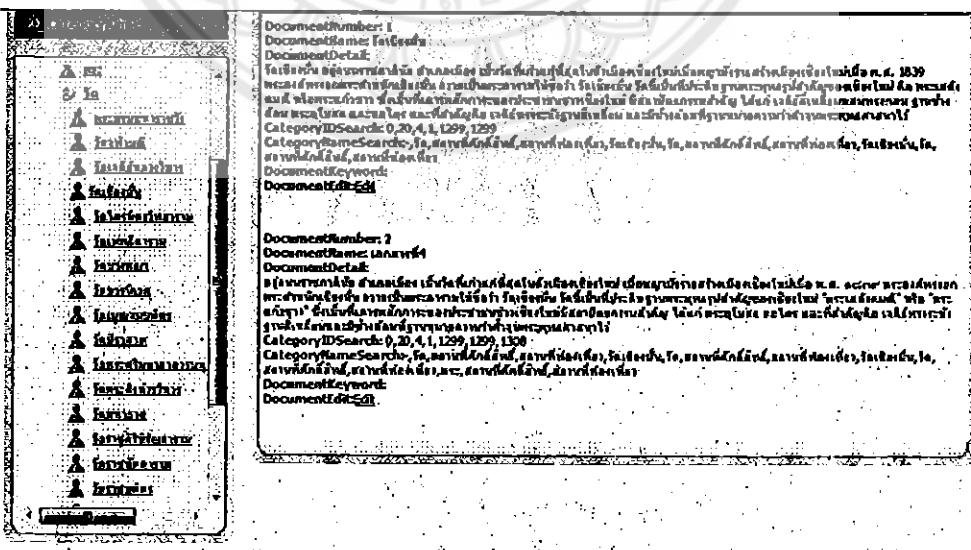
รูปที่ 4.11 รูปแสดงการเพิ่มเอกสารที่ 5 และทำการวิเคราะห์ข้อมูลเพื่อจัดหมวดหมู่

แสดงการค้นหาตามหมวดหมู่เพื่อค้นหาเอกสารที่ 3 เมื่อเลือกหมวดหมู่วัดเจดีย์หลวง
วิหาร จะได้ผลลัพธ์ตามรูปที่ 4.14



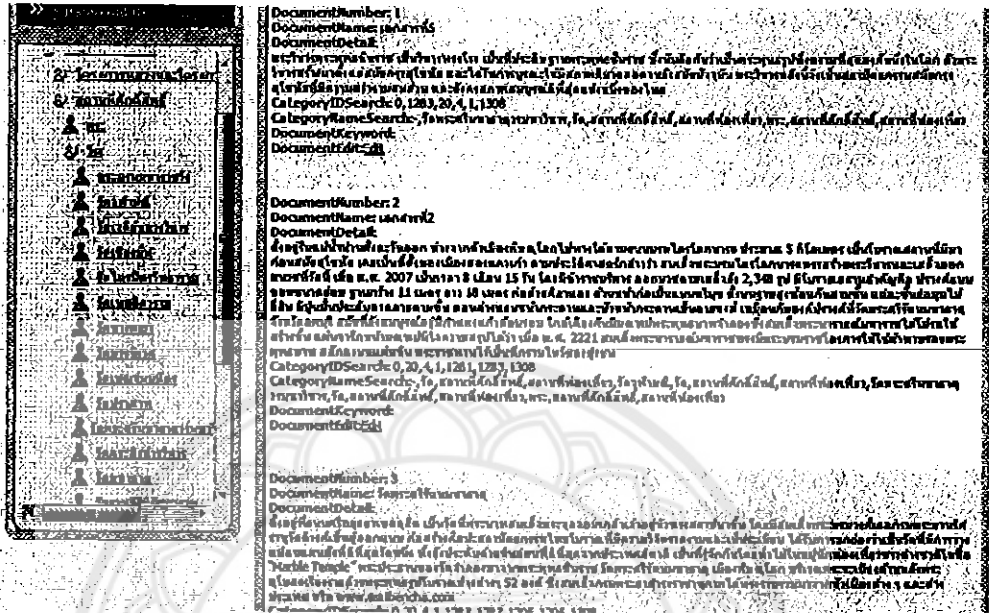
รูปที่ 4.14 รูปแสดงการค้นหาตามหมวดหมู่เพื่อค้นหาเอกสารที่ 3 เมื่อเลือกหมวดหมู่วัดเจดีย์หลวงวิหาร

แสดงการค้นหาตามหมวดหมู่เพื่อค้นหาเอกสารที่ 4 เมื่อเลือกหมวดหมู่วัดเชียงมั่น จะได้
ผลลัพธ์ตามรูปที่ 4.15



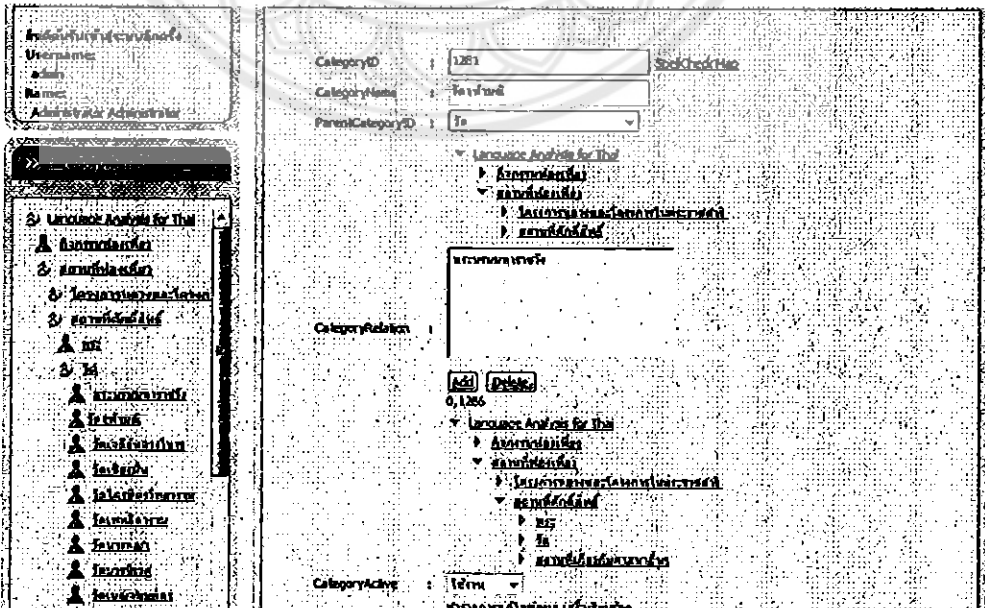
รูปที่ 4.15 รูปแสดงการค้นหาตามหมวดหมู่เพื่อค้นหาเอกสารที่ 4 เมื่อเลือกหมวดหมู่วัดเชียงมั่น

แสดงการค้นหาตามหมวดหมู่เพื่อค้นหาเอกสารที่ 5 เมื่อเลือกหมวดหมู่พระศรีรัตนมหาธาตุ
จะได้ผลลัพธ์ตามรูปที่ 4.16

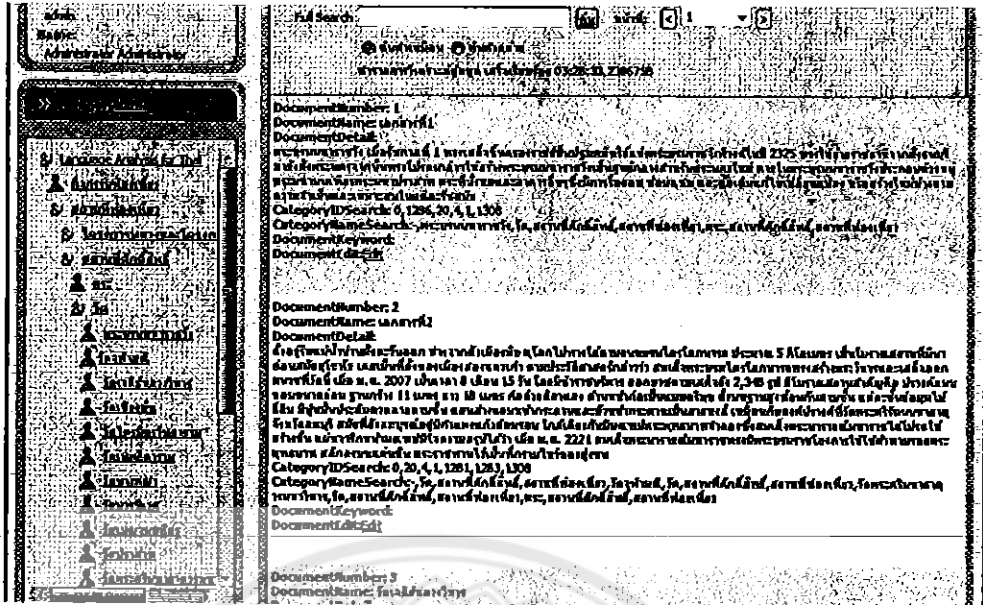


รูปที่ 4.16 รูปแสดงการค้นหาตามหมวดหมู่เพื่อค้นหาเอกสารที่ 5 เมื่อเลือกหมวดหมู่
วัดพระศรีรัตนมหาธาตุ

การเพิ่มความสัมพันธ์ระหว่างหมวดหมู่ ยกตัวอย่างการเพิ่มความสัมพันธ์ระหว่าง
หมวดหมู่วัดจุฬามณีกับพระบรมมหาราชวัง จะได้ผลลัพธ์ตามรูปที่ 4.17 และ รูปที่ 4.18



รูปที่ 4.17 รูปแสดงการเพิ่มความสัมพันธ์ระหว่างหมวดหมู่วัดจุฬามณีกับพระบรมมหาราชวัง



รูปที่ 4.18 รูปแสดงการค้นหาเอกสารหลังเพิ่มความสัมพันธ์ระหว่างหมวดหมู่ฉุกเฉินกับ
พระบรมมหาราชวัง

จากตัวอย่างข้างต้นนั้นเราสามารถนำมาใช้ในการจัดการเพิ่มหมวดหมู่เอกสารท่องเที่ยวอัตโนมัติในลักษณะอื่นๆ ได้อีก ถ้ามีข้อมูลเพิ่มมากขึ้นการจัดหมวดหมู่ในปริมาณมากนั้นถ้าจัดการโดย User อาจเกิดข้อผิดพลาดจากการเพิ่มข้อมูลของ User การจัดหมวดหมู่อัตโนมัตินั้นจะช่วยให้เราสามารถทำการจัดหมวดหมู่ได้อย่างถูกต้องและประหยัดในการจัดเก็บเอกสารที่ข้อมูลตรงกับหมวดหมู่ไว้ในหมวดเดียวกัน การจัดเอกสารไว้ในหมวดเดียวกันนั้นทำให้สามารถค้นหาข้อมูลที่ต้องการได้รวดเร็วและคัดส่วนที่ไม่เกี่ยวข้องกับการค้นหาออกไป

4.2 วิเคราะห์ผลการทดลอง

จากการทดสอบเปรียบเทียบการเพิ่มเอกสารที่ 1 ถึงเอกสารที่ 5 แล้วทำการวิเคราะห์จัดหมวดหมู่โดยไม่ตัดค่า ในกรณีที่จำนวนของหมวดหมู่มีจำนวน 50 -1250 หมวดหมู่ ได้ผลการทดสอบดังตารางที่ 4.1 และแสดงกราฟค่าเวลาเฉลี่ยในการวิเคราะห์เอกสารดังกราฟที่ 4.1

และการทดสอบเปรียบเทียบการเพิ่มเอกสารที่ 1 ถึงเอกสารที่ 5 แล้วทำการวิเคราะห์จัดหมวดหมู่โดยการตัดค่า ในกรณีที่จำนวนของหมวดหมู่มีจำนวน 50 -1250 หมวดหมู่ ได้ผลการทดสอบดังตารางที่ 4.2 และแสดงกราฟค่าเวลาเฉลี่ยในการวิเคราะห์เอกสารดังกราฟที่ 4.2

จากการวิเคราะห์หมวดหมู่เอกสาร ตารางที่ 4.1 และตาราง 4.2 พบปัจจัย 2 อย่างที่ทำให้ใช้เวลาในการวิเคราะห์เอกสารคือ จำนวนหมวดหมู่ที่เพิ่มมากขึ้น เมื่อมีหมวดหมู่เพิ่มมากขึ้นจะใช้เวลาในการวิเคราะห์เอกสารเพิ่มขึ้น และขนาดของเอกสารก็เป็นปัจจัยที่ทำให้ใช้เวลาในการวิเคราะห์เอกสารเพิ่มขึ้น ถ้าเอกสารมีขนาดใหญ่จะใช้นานกว่าเอกสารที่มีขนาดเล็กกว่า

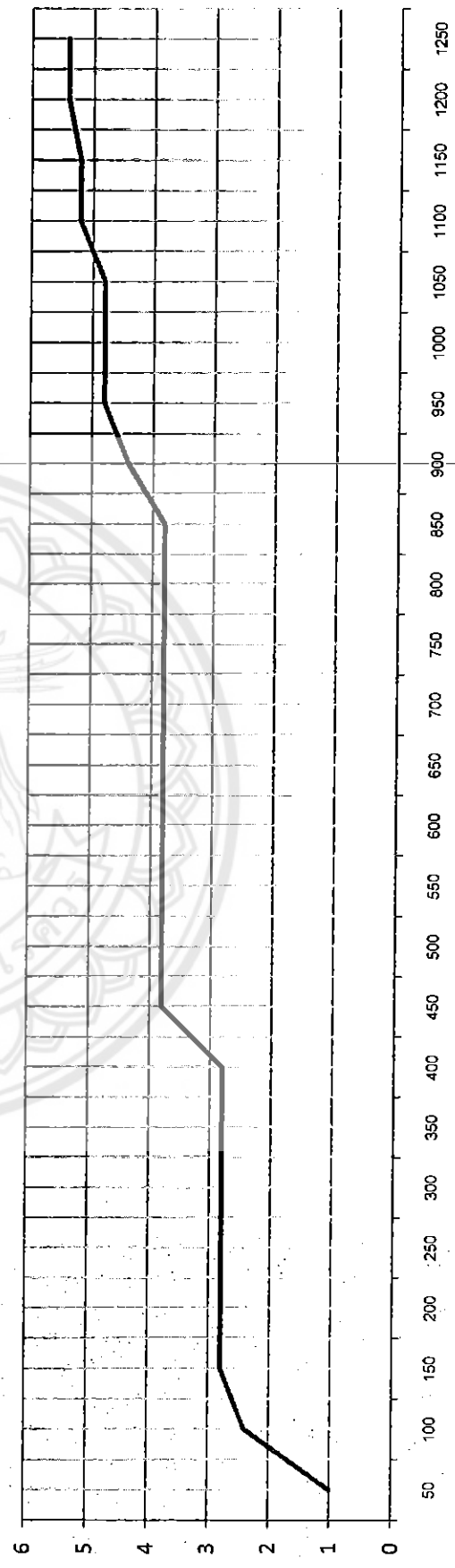
จากตารางที่ 4.3 และกราฟที่ 4.3 แสดงการเปรียบเทียบเวลาเฉลี่ยของที่ใช้ในการวิเคราะห์หมวดหมู่เอกสาร โดยใช้วิธีตัดคำและไม่ตัดคำพบว่า การวิเคราะห์จัดหมวดหมู่โดยการใช่วิธีตัดคำนั้น จะใช้เวลาในการวิเคราะห์จัดหมวดหมู่ น้อยกว่าเมื่อมีหมวดหมู่เอกสารมากกว่า 1000 หมวดหมู่ ส่วนในกรณีที่มีหมวดหมู่ น้อยกว่า 1000 หมวดหมู่ จะใช้เวลาในการวิเคราะห์หมวดหมู่ใกล้เคียงกัน



ตารางที่ 4.1 ตารางแสดงจำนวนของหมวดหมู่เปรียบเทียบกับเวลาที่ใช้ในการวิเคราะห์หมวดหมู่เอกสาร โดยไม่ใช้วิธีตัดค่า

เอกสาร ที่	จำนวนของหมวดหมู่เปรียบเทียบกับเวลาที่ใช้ในการวิเคราะห์หมวดหมู่เอกสาร(s)																								
	50	100	150	200	250	300	350	400	450	500	550	600	650	700	750	800	850	900	950	1000	1050	1100	1150	1200	1250
1	1	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4
2	1	3	4	4	4	4	4	4	5	5	5	5	5	5	5	5	5	6	6	6	6	7	7	7	7
3	1	3	4	4	4	4	4	4	5	5	5	5	5	5	5	5	5	6	6	6	6	7	7	7	7
4	1	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4
5	1	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	5
ค่าเฉลี่ย	1	2.4	2.8	2.8	2.8	2.8	2.8	2.8	3.8	3.8	3.8	3.8	3.8	3.8	3.8	3.8	3.8	4.4	4.8	4.8	4.8	5.2	5.2	5.4	5.4

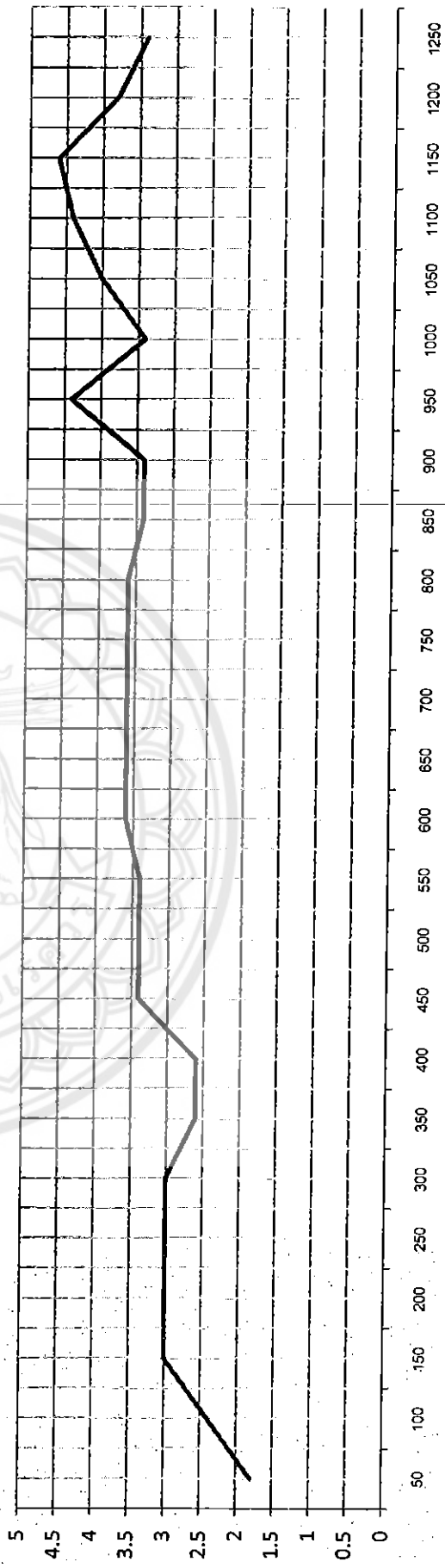
กราฟที่ 4.1 ตารางแสดงจำนวนของหมวดหมู่เปรียบเทียบกับเวลาที่ใช้ในการวิเคราะห์หมวดหมู่เอกสาร โดยไม่ใช้วิธีตัดค่า



ตารางที่ 4.2 ตารางแสดงจำนวนของหมวดหมู่เปรียบเทียบเกี่ยวกับเวลาที่ใช้ในการวิเคราะห์หมวดหมู่เอกสารโดยใช้วิธีตัดค่า

เอกสาร ที่	จำนวนของหมวดหมู่เปรียบเทียบเกี่ยวกับเวลาที่ใช้ในการวิเคราะห์หมวดหมู่เอกสาร(๑)																								
	50	100	150	200	250	300	350	400	450	500	550	600	650	700	750	800	850	900	950	1000	1050	1100	1150	1200	1250
1	1	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	3	4	3	2	3	3	3	
2	2	3	5	5	5	5	3	3	5	5	5	5	5	5	5	4	4	4	5	5	5	6	6	5	
3	2	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	3	6	3	5	5	6	4	
4	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	4	4	3	4	3	3	2	
5	2	2	2	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	4	5	5	3	
ค่าเฉลี่ย	1.8	2.4	3	3	3	3	2.6	2.6	3.4	3.4	3.4	3.6	3.6	3.6	3.6	3.6	3.4	3.4	4.4	3.4	4	4.4	4.6	3.8	3.4

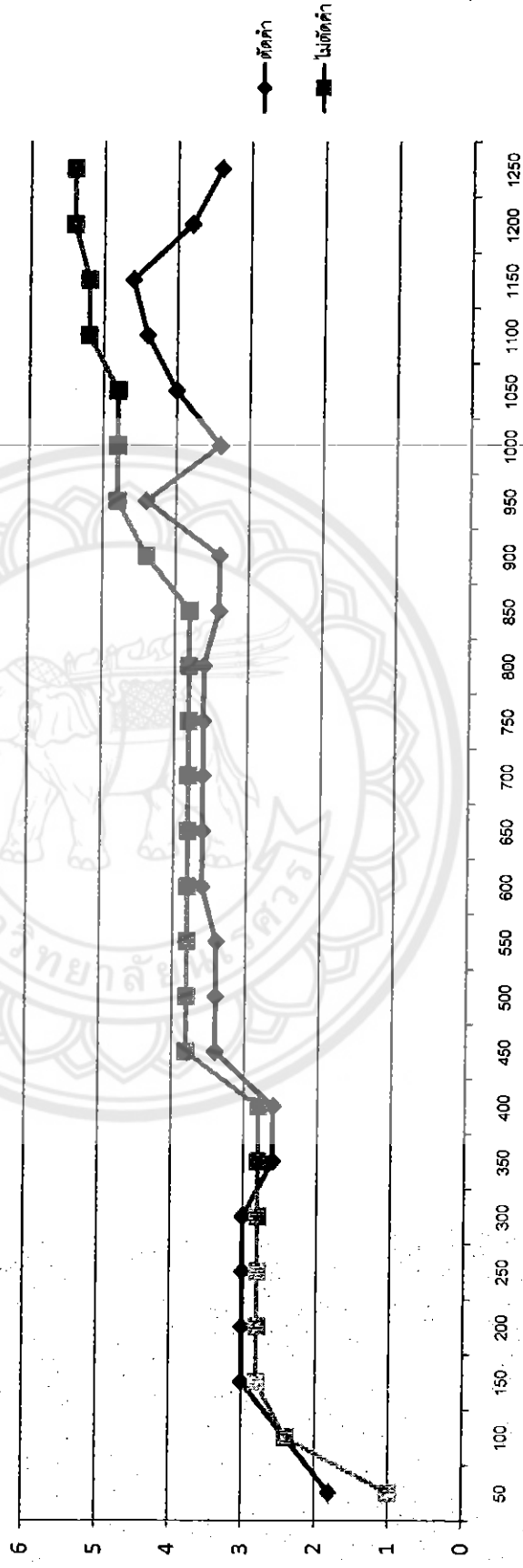
กราฟที่ 4.2 กราฟแสดงจำนวนเฉลี่ยของหมวดหมู่เปรียบเทียบเกี่ยวกับเวลาที่ใช้ในการวิเคราะห์หมวดหมู่เอกสารโดยใช้วิธีตัดค่า



ตารางที่ 4.3 ตารางแสดงเวลาเฉลี่ยของที่ใช้ในการวิเคราะห์หาค่าความหนาแน่นของวัสดุโดยวิธีตัดค่าและไม้ตัดค่า

เวลาเฉลี่ย	จำนวนของหาค่าความหนาแน่นที่เปรียบเทียบกับเวลาที่ใช้ในการวิเคราะห์หาค่าความหนาแน่นของวัสดุ(s)																								
	50	100	150	200	250	300	350	400	450	500	550	600	650	700	750	800	850	900	950	1000	1050	1100	1150	1200	1250
ไม้ตัดค่า	2.4	2.4	2.8	2.8	2.8	2.8	2.8	2.8	3.8	3.8	3.8	3.8	3.8	3.8	3.8	3.8	3.8	4.4	4.8	4.8	4.8	5.2	5.4	5.4	
ตัดค่า	1.8	2.4	3	3	3	3	2.6	2.6	3.4	3.4	3.4	3.6	3.6	3.6	3.6	3.6	3.4	3.4	4.4	3.4	4	4.4	4.6	3.8	3.4

กราฟที่ 4.3 กราฟแสดงเวลาเฉลี่ยของที่ใช้ในการวิเคราะห์หาค่าความหนาแน่นของวัสดุโดยวิธีตัดค่าและไม้ตัดค่า



บทที่ 5 บทสรุป

โครงการนี้พัฒนาขึ้นเพื่อช่วยในการจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวแบบอัตโนมัติ ซึ่งได้แนวคิดมาจากการที่ปัจจุบันนี้ข้อมูลด้านการท่องเที่ยวมีจำนวนเพิ่มขึ้นอย่างรวดเร็วจึงจำกัลดนกลายเป็นคลังข้อมูลขนาดใหญ่ ดังนั้นการได้มาซึ่งข้อมูลต่างๆ จำเป็นต้องใช้วิธีการสืบค้นข้อมูลแบบ Full Text Search ในการค้นหาข้อมูล อย่างไรก็ตามรูปแบบหนึ่งของการสืบค้นข้อมูลที่ได้รับคามนิยมอย่างมากคือ ระบบค้นหาข้อมูลตามหมวดหมู่จากหมวดหมู่ซึ่งจะทำให้สามารถค้นหาข้อมูลที่เฉพาะเจาะจงตามหมวดหมู่

จากการทดลองสร้างต้นแบบ Web application การจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวอัตโนมัติ โดยใช้ฐานข้อมูลทดสอบจากฐานข้อมูลที่กำหนด และใช้การเทียบค่าเป็นตัวเปรียบเทียบสำหรับการวิเคราะห์ค่าในการจัดหมวดหมู่ จากผลการทดลองเมื่อทำการเลือกคุณลักษณะของคำที่ใช้เป็นคีย์เวิร์ด (keyword) ในการเปรียบเทียบได้เหมาะสมจะส่งผลกระทบต่อความถูกต้องในการจำแนกหมวดหมู่ข้อมูลด้านการท่องเที่ยว และพบปัจจัยที่ส่งผลให้การเพิ่มเอกสารใช้เวลานานขึ้น 2 ปัจจัยคือ ขนาดของเอกสาร และขนาดของหมวดหมู่ ทำให้ต้องคิดหาวิธีการแก้ไขปัญหาที่เกิดขึ้นจากกระบวนการระบุหมวดหมู่โดยไม่ใช้การคัดคำ ทำให้ได้กระบวนการระบุหมวดหมู่โดยใช้การคัดคำซึ่งไม่มีปัญหาเรื่องขนาดของเอกสาร และขนาดของหมวดหมู่ แต่จะมีปัญหาเรื่องความถูกต้องของการคัดคำ ดังนั้นแต่ละวิธีการจะมีข้อดีและข้อเสียที่แตกต่างกันไป การเลือกใช้งานควรเลือกใช้ให้เหมาะสม

5.1 สรุปผลการทำงานของต้นแบบ Web application การจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวอัตโนมัติ

- 5.1.1 สามารถจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวได้อัตโนมัติ
- 5.1.2 สามารถค้นหาข้อมูลตามหมวดหมู่ได้อย่างรวดเร็วและมีความถูกต้องของข้อมูล
- 5.1.3 สามารถเพิ่มเติมหมวดหมู่ แก้ไข และปรับปรุงหมวดหมู่ให้ข้อมูลมีการอัปเดตอยู่เสมอเพื่อความถูกต้องของข้อมูล
- 5.1.4 สามารถค้นหาข้อมูลด้านการท่องเที่ยวแบบ keyword search โดยใช้ SQL Select Statement แบบปกติได้
- 5.1.5 สามารถสร้างความสัมพันธ์ระหว่างหมวดหมู่ข้อมูลการท่องเที่ยวแบบทางเดียวและสองทาง ได้โดยการเพิ่มความสัมพันธ์ระหว่างหมวดหมู่ที่ต้องการ

5.2 ปัญหาและแนวทางแก้ไขปัญหา

ในการทำงานที่ผ่านมาสามารถสรุปปัญหาที่พบต่างๆในการทำงานที่เกิดขึ้นได้ดังนี้

5.2.1 ปัญหาด้านการจัดหมวดหมู่

การจัดหมวดหมู่การเทียบคำโดยใช้คุณลักษณะของคำเป็นคีย์เวิร์ด (keyword) นั้น ถ้าคำที่เลือกมาใช้เป็นคีย์เวิร์ด (keyword) ผิดพลาด หรือมีคำสำหรับเปรียบเทียบไม่เพียงพอ ผลลัพธ์จากการจัดหมวดหมู่ที่ได้จะคลาดเคลื่อนจากความถูกต้อง

5.1.2 ปัญหาด้านฐานข้อมูล

จากการทดสอบใช้งานพบข้อดีและข้อเสียของกระบวนการระบุหมวดหมู่ทั้ง 2 แบบดังนี้

- ใช้กระบวนการระบุหมวดหมู่โดยไม่ใช้การตัดคำ

ข้อดีคือ สามารถนำคีย์เวิร์ด (keyword) ของแต่ละหมวดหมู่มาเปรียบเทียบกับเอกสารแล้วสามารถระบุหมวดหมู่ได้อย่างแม่นยำ เนื่องจากกระบวนการเปรียบเทียบที่ละหมวดหมู่เป็นการไล่เทียบทีละตัวอักษรจึงไม่มีข้อผิดพลาดในการเทียบคีย์เวิร์ด (keyword)

ข้อเสียคือ รองรับจำนวนข้อมูลที่ใช่เป็นหัวคีย์เวิร์ด (keyword) ได้จำกัด เนื่องจากต้องใช้เวลาในการเปรียบเทียบคีย์เวิร์ด (keyword) ก่อนข้างาน ทำให้ไม่สามารถรองรับข้อมูลจำนวนมากได้ จึงเหมาะสำหรับฐานข้อมูลขนาดเล็ก

- ใช้กระบวนการระบุหมวดหมู่โดยใช้การตัดคำ

ข้อดีคือ สามารถใช้งานได้ในระบบที่มีปริมาณข้อมูลจำนวนมากได้ เพราะใช้การตัดคำแล้วนำคำที่ได้มาเปรียบเทียบกับหมวดหมู่เนื้อหาข้อมูล

ข้อเสียคือ การเปรียบเทียบโดยใช้การตัดคำอาจมีข้อผิดพลาดในการเทียบคีย์เวิร์ด (keyword) เกิดจากการที่คำตัดคำไม่สามารถตัดคำได้ละเอียดพอ

5.3 ข้อเสนอแนะ

5.3.1 คำที่ใช้เป็นคีย์เวิร์ด (keyword) นั้นควรจะครอบคลุมถึงคำจำกัดความของสถานที่นั้นๆได้อย่างละเอียด

5.3.2 สามารถนำส่วนตัดคำ ICU4.NET ไปพัฒนาในการจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวอัตโนมัติสำหรับฐานข้อมูลขนาดใหญ่

5.3.3 สามารถนำกระบวนการจัดหมวดหมู่ข้อมูลด้านการท่องเที่ยวไปใช้สำหรับจัดหมวดหมู่ข้อมูลด้านอื่นๆได้ เช่น ข้อมูลด้านกีฬา ข้อมูลด้านข่าว เป็นต้น

เอกสารอ้างอิง

- [1] นางสาวณิชชาพร สุระ. (2549). การจำแนกหมวดหมู่เอกสารภาษาไทยอัตโนมัติโดยใช้
อัลกอริทึม FPTC.
วทบ.ม., สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, กรุงเทพมหานคร
- [2] ศศ.ศิริลักษณ์ วิจารณ์กิจอำนวย. (2538). ระบบฐานข้อมูล ใน ระบบฐานข้อมูล (DATABASE
SYSTEM) (หน้า 1-11)
- [3] วิกิพีเดีย สารานุกรมเสรี. ภาษาซีชาร์ป. สืบค้นเมื่อ 22 พฤษภาคม 2554,
จาก <http://th.wikipedia.org/wiki/ภาษาซีชาร์ป/>
- [4] มหาวิทยาลัยมหาสารคาม. การใช้งาน String Class. สืบค้นเมื่อ 22 พฤษภาคม 2554,
จาก https://cyberclass.msu.ac.th/cyberclass/cyberclass/Ch09_9b19.doc



ภาคผนวก ก

Microsoft SQL Server

Microsoft SQL Server (อ่านว่า ซีควอลเซิร์ฟเวอร์ หรือ เอสคิวแอลเซิร์ฟเวอร์) คือระบบจัดการฐานข้อมูลพัฒนาโดยไมโครซอฟท์ ซึ่งใช้ภาษา T-SQL ในการดึงเรียกข้อมูล

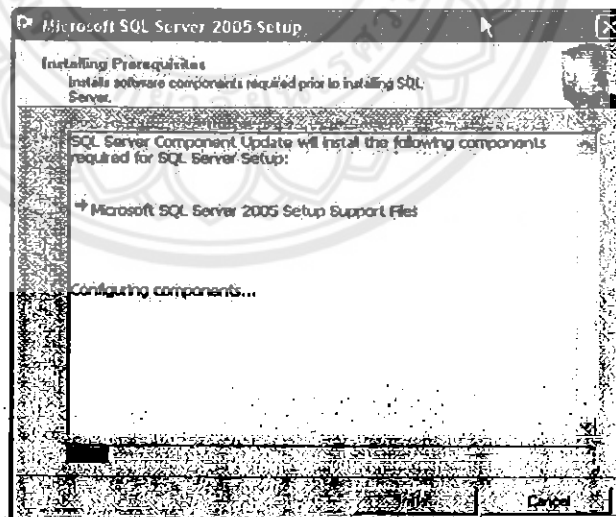
การติดตั้ง SQL Server Management Studio Express

SQL Server Management Studio Express เป็นโปรแกรมช่วยในการจัดการฐานข้อมูลใน SQL SERVER ที่ได้ติดตั้งไว้

1. ทำการ Run File ติดตั้ง SQLServer2005_SSMSEE.msi จากนั้นให้Click ที่Check Box “ I accept the terms in the license agreement” จากนั้นคลิกปุ่ม Next ตามรูปที่ ก-1 และตามรูปที่ ก-2

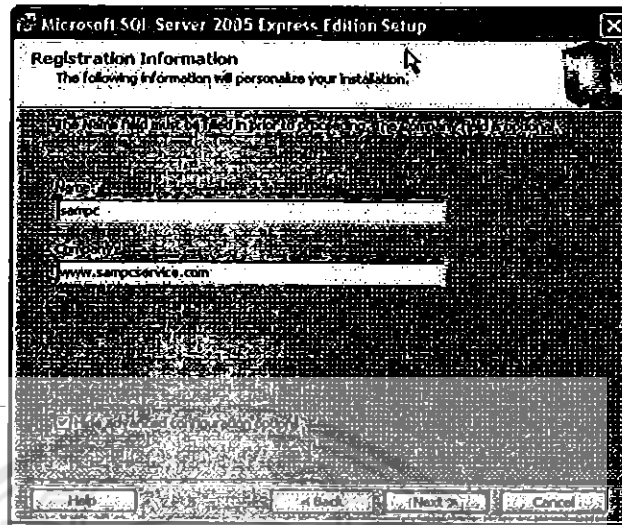


รูปที่ ก-1 แสดงขั้นตอนการติดตั้งโปรแกรม SQL Server Management Studio Express



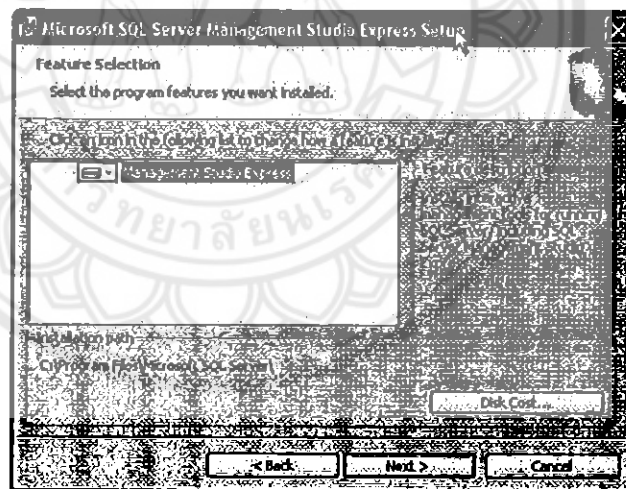
รูปที่ ก-2 แสดงขั้นตอนการติดตั้งโปรแกรม

2. ทำการป้อนชื่อ ในช่อง Name และ ชื่อกิจการในช่อง Company จากนั้นกดปุ่ม Next ตามรูปที่ ก-3



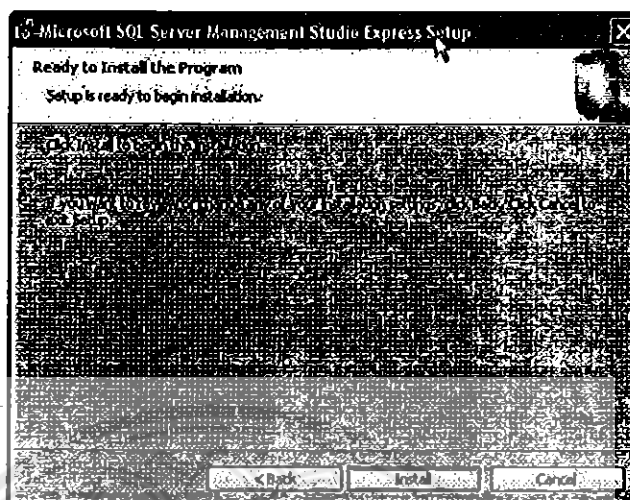
รูปที่ ก-3 แสดงขั้นตอนการเพิ่ม Name และ Company

3. ให้ทำการเลือก Feature ของ โปรแกรมดังรูป แล้วกดปุ่ม Next ตามรูปที่ ก-4



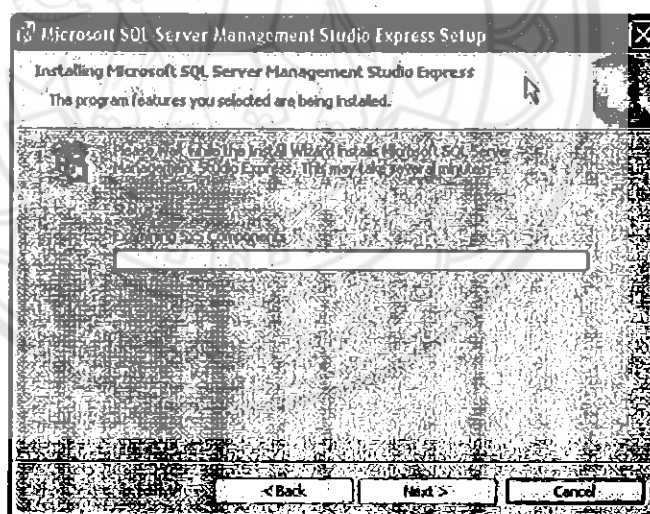
รูปที่ ก-4 แสดงขั้นตอนการเลือก path

4. ให้ทำการกด Next ตามรูปที่ ก-5



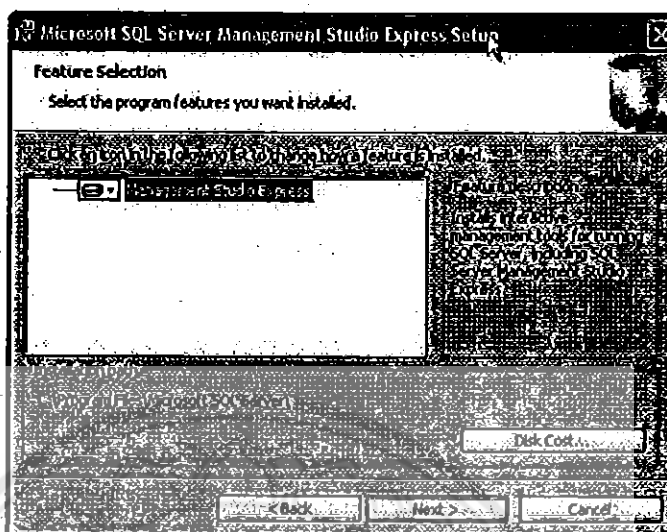
รูปที่ ก-5 แสดงขั้นตอนการเริ่มติดตั้ง โปรแกรม

5. ให้ทำการกด Next ตามรูปที่ ก-6



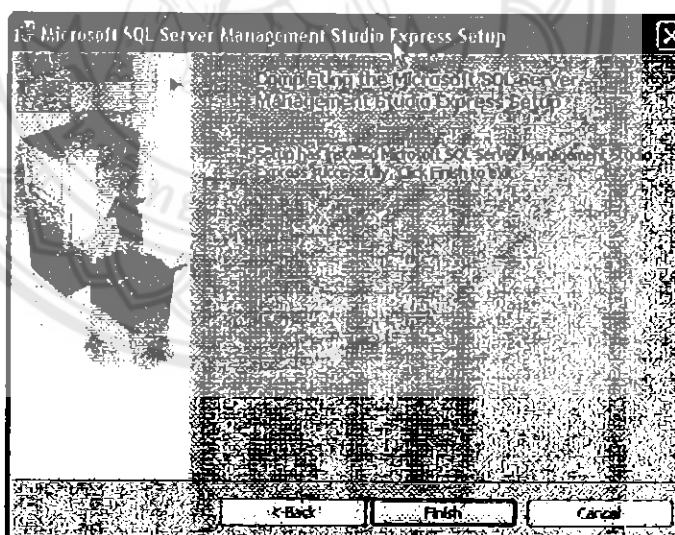
รูปที่ ก-6 แสดงขั้นตอนการเริ่มติดตั้งโปรแกรมขั้นตอนที่ 2

6. ให้ทำการกด Next ตามรูปที่ ก-7



รูปที่ ก-7 แสดงขั้นตอนการเริ่มติดตั้ง โปรแกรมขั้นตอนที่3

7. เสร็จสิ้นการติดตั้ง ตามรูปที่ ก-8



รูปที่ ก-8 แสดงขั้นตอนเสร็จสิ้นการติดตั้ง โปรแกรม

ประวัติผู้เขียนโครงการ

ชื่อ



นายณัฐภูมิเชียวชัย

ภูมิลำเนา 29 หมู่ 22 ต.จิว อ.เทิง จ.เชียงราย 57160

ประวัติการศึกษา

- จบการศึกษาระดับมัธยมศึกษาจาก โรงเรียนเทิง
วิทยาคม

- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 7

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยนเรศวร

E-Mail: Donovan_x64@hotmail.com



ชื่อนายเอกฉรงค์ นามอินทร์

ภูมิลำเนา 225 หมู่ 6 ต.บ้านถ้ำ อ.คอกคำใต้ จ.พะเยา 56210

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจาก โรงเรียนคอกคำใต้

วิทยาคม

- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 7

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยนเรศวร

E-Mail: Hongsa_x57@hotmail.com