ต้นแบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา

Educational Service Web Database Prototype

โดย

นรพันธ์ ศิริอำพันธ์กุล

รหัสประจำตัว 46066208

อาจารย์ที่ปรึกษา

ผศ.ดร.พรฤดี เนติโสภากุล

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ภาคเรียนที่ 2 ปีการศึกษา 2548

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ	ต้นแบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา
นักศึกษา	นายนรพันธ์ ศิริอำพันธ์กุล
อาจารย์ที่ปรึกษา	ผศ.ดร.พรฤดี เนติโสภากุล
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2548

บทคัดย่อ

งานบริการการศึกษาจำเป็นต้องค้นหาและให้บริการข้อมูลต่างๆเกี่ยวกับสถานศึกษา ซึ่งมีเป็นจำนวนมาก การค้นหาโดยใช้เว็บเสิร์ชเอ็นจิ้นทั่วไปในปัจจุบันให้ผลลัพธ์ที่ไม่เกี่ยวข้องจำนวนมาก ทำให้เสียเวลาและความพยายามในการกลั่นกรองเฉพาะข้อมูลที่มีประโยชน์ในการค้นหาแต่ละครั้ง ดังนั้น แนวคิดของงานวิจัยนี้ คือการศึกษา และสร้างต้นแบบของฐานข้อมูลเว็บสำหรับงานบริการการศึกษา เพื่อใช้เป็นเครื่องมือที่รวบรวม และจัดเก็บข้อมูล เพื่อใช้ในงานบริการการศึกษา โดยใช้เทคโนโลยีเว็บเสิร์ชเอ็นจิ้น การค้นคืนเอกสารและการประมวลผลข้อความ และการพัฒนาฐานข้อมูลเว็บ ทำการค้นหาและจัดเก็บข้อมูลใน 2 มิติ ได้แก่มิติด้านสถานที่ตั้ง และมิติด้านสาขาวิชา โดยพัฒนาเว็บครอว์เลอร์สำหรับท่องไปยังเว็บเพจ กำหนดขอบเขตในการค้นหาและจัดเก็บไฮเปอร์ลิงค์ที่พบ ทั้งนี้ยังได้มีการพัฒนากระบวนการในการดึงข้อความระบุที่ตั้งสถาบันการศึกษามาจัดเก็บลงฐานข้อมูลโดยใช้การเปรียบเทียบรูปแบบข้อความ และการดึงข้อความระบุสาขาวิชา โดยการเปรียบเทียบข้อความกับรายชื่อสาขาวิชาที่จัดเก็บไว้ ซึ่งข้อมูลที่จัดเก็บได้นี้จะมีการนำเสนอผ่านทางเว็บเบราเซอร์ เพื่อให้ผู้ใช้สามารถค้นหาข้อมูลได้

Title	Educational Service Web Database Prototype
Student	Mr. Noraphan Siriamphankul
Advisor	Asst. Prof. Dr. Ponrudee Netisophakul
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2005

ABSTRACT

Educational service must search and provide information related to a numerous number of educational institutes. Using the current general search engine tools have too many unrelated result data. It causes time and effort in filtering only useful data for each search. Therefore, this research is to study and build a prototype of educational service web database that collects and organizes data for educational service usage. The related technologies are web search engines, information retrieval and text processing and web database development. This research is collecting data in Location dimension and Major dimension by using web crawler for crawl and extract hyperlink in domain specific webpage. Moreover, this research is developing algorithm for extract university's address by pattern matching techinque and extract university's major offered by matching with major in database.The Collected data represent via web browser that used by user.

กิตติกรรมประกาศ

การจัดทำโครงการพัฒนาระบบนี้ สามารถประสบความสำเร็จลุล่วงไปได้ ด้วยคำแนะนำต่างๆของ ผศ.ดร.พรฤดี เนติโสภากุล ซึ่งเป็นอาจารย์ที่ปรึกษาที่ได้สละเวลาในให้คำปรึกษาต่างๆ ผู้จัดทำต้องขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณ คุณพ่อ คุณแม่ที่คอยให้กำลังใจ รวมถึงสนับสนุนทุนทรัพย์ในการศึกษา

ขอขอบพระคุณคุณพ่อเปี่ยม คุณแม่จินตนา และคุณบุษยวรรณ ศรีมิตรานนท์ ที่คอยให้กำลังใจในการทำงานอยู่เสมอ

ขอขอบคุณ คุณจรัล วราสิทธิกุล ที่ช่วยแนะนำเทคนิควิธีการเขียนโปรแกรมเพื่อใช้ในการทำงาน

ขอขอบคุณพี่ๆน้องๆ ห้องปฏิบัติการ KMAKE ทุกท่านที่คอยเป็นกำลังใจอยู่เสมอ

ทั้งนี้ ขอขอบพระคุณทุกท่านที่ไม่ได้กล่าวนามมาในที่นี้ด้วย

นรพันธ์ ศิริอำพันธ์กุล

สารบัญ

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญตาราง

หน้า

ตารางที่

สารบัญตาราง(ต่อ)

หน้า

ตารางที่

สารบัญรูป

หน้า

รูปที่

สารบัญรูป(ต่อ)

หน้า

รูปที่

สารบัญรูป(ต่อ)

หน้า

รูปที่

บทที่ 1
บทนำ

ในปัจจุบันเทคโนโลยีสารสนเทศได้เข้ามามีบทบาทในการดำเนินชีวิต และการทำงานต่างๆมากขึ้น ซึ่งเทคโนโลยีเหล่านี้ได้ช่วยเพิ่มประสิทธิภาพในการดำเนินงานให้มีความถูกต้องและรวดเร็วมากยิ่งขึ้น ซึ่งรวมไปถึงการเผยแพร่ข้อมูลข่าวสารและการค้นหาข้อมูลต่างๆ ซึ่งได้อาศัยเทคโนโลยีทางด้านเครือข่ายและอินเทอร์เน็ตเข้ามาช่วย ทำให้สามารถกระจายข้อมูลข่าวสารได้อย่างรวดเร็ว

เนื่องด้วยความต้องการในการค้นหาข้อมูลต่างๆจากอินเทอร์เน็ตมีมากขึ้น อีกทั้งข้อมูลที่มีอยู่บนเครือข่ายอินเทอร์เน็ตนั้นมีเป็นจำนวนมาก เพื่อที่จะให้ได้ข้อมูลที่ต้องการนั้นได้มีการพัฒนาเว็บเสิร์ชเอ็นจิ้น(web search engine) ซึ่งเป็นเครื่องมือช่วยในการค้นหาเอกสารและข้อมูลที่มีอยู่มากมายบนอินเทอร์เน็ตให้ได้ตรงกับความต้องการ แต่ในบางครั้งเมื่อมีความต้องการค้นหาข้อมูลในงานด้านใดด้านหนึ่ง ผลลัพธ์ที่ได้จากการค้นหาโดยเว็บเสิร์ชเอ็นจิ้นทั่วไปนั้นมีจำนวนมากซึ่งอาจจะมีข้อมูลที่ไม่ได้เกี่ยวข้องกับสิ่งที่ต้องการมาด้วย ทำให้เสียเวลาในการเลือกหาข้อมูลที่ต้องการ จึงได้มีแนวคิดในการที่จะหาทางนำข้อมูลข่าวสารที่กระจัดกระจายอยู่ตามที่ต่างๆ มารวบรวมไว้ในแหล่งเดียวกัน เพื่อให้สะดวกในการค้นหาข้อมูล และเพื่อรองรับการใช้งานของผู้ใช้ที่มีความต้องการค้นหาข้อมูลในงานนั้นๆอยู่เป็นประจำ อีกทั้งเพื่อเป็นการลดจำนวนของเว็บเพจที่ไม่เกี่ยวข้องกับที่ต้องการอีกด้วย

ในงานบริการการศึกษา จะมีการให้ข้อมูลแก่ผู้มาขอรับบริการกลุ่มต่างๆ ซึ่งได้แก่ นักเรียน ,ทุน นักศึกษา อาจารย์และบุคคลทั่วไปที่มีความสนใจ แต่ในปัจจุบัน ข้อมูลข่าวสารของสถาบันการศึกษาต่างๆมีมากขึ้น อีกทั้งในแต่ละสถาบันการศึกษาได้มีการจัดทำเว็บไซต์เป็นของตัวเอง ทำให้ข้อมูลเกิดการกระจัดกระจายอยู่ตามเว็บไซต์ต่างๆดังกล่าวทำให้ต้องใช้เวลาในการรวบรวมข้อมูลข่าวสารจากสถาบันการศึกษาต่างๆมากขึ้น จึงได้นำเอาเทคโนโลยีด้านฐานข้อมูลมาช่วยในการจัดเก็บข้อมูลข่าวสารที่เกี่ยวกับสถาบันการศึกษาต่างๆเข้าไว้ด้วยกัน โดยทำการออกแบบฐานข้อมูลให้มีประสิทธิภาพ เพื่อให้เกิดความสะดวกและรวดเร็วในการค้นหาข้อมูล อีกทั้งสามารถเปรียบเทียบข้อมูลข่าวสารจากหลายๆสถาบันการศึกษาได้ โดยใช้เทคโนโลยีทางด้านเว็บเสิร์ชเอ็นจิ้น ซึ่งเป็นเครื่องมือช่วยในการค้นหาเอกสารและข้อมูลข่าวสารจากสถาบันการศึกษาต่างๆที่มีอยู่มากมายบนอินเทอร์เน็ต มารวบรวมไว้ในฐานข้อมูลเพื่อใช้ในงานบริการการศึกษา สำหรับการแนะแนวการศึกษาต่อในสถาบันการศึกษาต่างๆสำหรับผู้ที่สนใจ โดยฐานข้อมูลที่จะใช้ในการ

จัดเก็บนั้นจะต้องสามารถปรับปรุงข้อมูลที่จัดเก็บไว้แล้วได้ เมื่อมีการปรับปรุงเปลี่ยนแปลงหรือแก้ไขข้อมูลข่าวสารของสถาบันศึกษาแต่ละแห่ง เพื่อให้ข้อมูลมีความถูกต้องและทันสมัยอยู่เสมอ

1.1 วัตถุประสงค์

1. เพื่อศึกษา วิเคราะห์ และออกแบบ ฐานข้อมูลเว็บเพื่อใช้สำหรับงานบริการการศึกษา เพื่อเป็นแหล่งข้อมูลให้กับผู้ให้บริการในงานบริการการศึกษา ในการให้คำแนะนำกับผู้ที่มีความสนใจที่จะศึกษาต่อในต่างประเทศ พร้อมทั้งพัฒนาส่วนติดต่อกับผู้ให้บริการในงานบริการการศึกษา เพื่อให้การให้บริการได้อย่างรวดเร็ว ถูกต้องแม่นยำ น่าเชื่อถือ และง่ายต่อการเรียนรู้และใช้งาน
2. เพื่อลดเวลาการค้นหาข้อมูลข่าวสารของสถาบันการศึกษาในแต่ละแห่ง ซึ่งจะช่วยให้สามารถให้บริการได้รวดเร็วขึ้น อีกทั้งได้ข้อมูลที่ตรงกับความต้องการของผู้มาขอใช้บริการ
3. เพื่อศึกษากระบวนการในการคัดเลือกและจัดเก็บข้อมูลข่าวสารในเว็บไซต์ของสถาบันการศึกษาต่างๆ มารวบรวมไว้ในฐานข้อมูล โดยใช้เทคโนโลยีของเสิร์ชเอ็นจิ้น รวมถึงกระบวนการในการค้นหาและนำเสนอข้อมูลที่จัดเก็บไว้สำหรับใช้ในงานบริการการศึกษา เพื่อให้ได้ข้อมูลที่ถูกต้องและตรงตามความต้องการ

1.2 ขอบเขตการพัฒนา

ในการพัฒนาฐานข้อมูลเว็บสำหรับงานบริการการศึกษานี้ จะเน้นไปที่มุมมองของผู้ให้คำปรึกษาในการแนะแนวการศึกษาต่อเป็นหลัก ซึ่งจะต้องสามารถรองรับความต้องการของผู้ใช้บริการ ซึ่งได้แก่กลุ่มของนักศึกษา อาจารย์ และบุคคลทั่วไปที่มีความสนใจ โดยเน้นที่ผู้ที่สนใจศึกษาต่อต่างประเทศในแถบทวีปอเมริกาเหนือเป็นหลัก ทั้งนี้รวมไปถึงผู้สนใจศึกษาต่อยังประเทศอื่นๆที่ได้รับความนิยมด้วย

สำหรับเป้าหมายในการรวบรวมข้อมูลนั้น จะทำการรวบรวมข้อมูลที่เกี่ยวกับสถาบันการศึกษาใน 2 มิติ คือมิติทางด้านที่ตั้ง(Location) และมิติทางด้านสาขาวิชาที่เปิดสอน (Major) ซึ่งมีองค์ประกอบในการพัฒนาอยู่ 4 ส่วน มีรายละเอียดดังนี้

1. ส่วนการรวบรวมข้อมูล(Data Collection) ซึ่งจะใช้เทคโนโลยีของเว็บครอว์เลอร์ซึ่งเป็นส่วนประกอบของเสิร์ชเอ็นจิ้น
   - พัฒนา algorithm และเครื่องมือสำหรับท่องไปยังเว็บเพจต่างๆบนอินเทอร์เนต
   - มีการกำหนดขอบเขตของข้อมูลที่จะไปทำการค้นหาเพื่อนำมาจัดเก็บ

- พัฒนากระบวนการปรับปรุงข้อมูลให้มีความทันสมัย โดยทำการรวบรวมข้อมูลอยู่อย่างสม่ำเสมอ

2. ส่วนการดึงข้อมูล(Data Extraction) จะเป็นการดึงเอาข้อมูลที่ต้องการในงานบริการการศึกษาออกมาจากเว็บเพจของสถาบันการศึกษาต่างๆ ซึ่งจะทำการดึงข้อมูลเกี่ยวกับที่ตั้งของสถาบันการศึกษา และสาขาวิชาที่แต่ละสถาบันเปิดสอน โดยใช้เทคโนโลยีทางด้าน text - processing และการกำหนดรูปแบบของข้อความโดยใช้ regular expression

- พัฒนากระบวนการในการทำเปรียบเทียบรูปแบบ(Pattern Matching) ระหว่างข้อความภายในเอกสารกับ regular expression เพื่อให้ข้อมูลที่ดึงออกมานั้นมีความถูกต้องตรงตามต้องการ
- ปรับปรุงรูปแบบของ regular expression ให้ครอบคลุมรูปแบบต่างๆกันมากขึ้น

3. ส่วนจัดการฐานข้อมูล(Database Management)

- จัดเก็บข้อมูลที่ได้มาจากส่วนการรวบรวมข้อมูล
- ออกแบบฐานข้อมูลให้สามารถจัดเก็บและค้นคืนข้อมูลได้อย่างมีประสิทธิภาพ
- จัดการข้อมูลที่มีอยู่ในฐานข้อมูลให้มีมุมมองต่างๆกัน ได้แก่ มุมมองในเรื่องสถานที่ตั้ง(Location) และมุมมองทางด้านสาขาวิชา(Major) เพื่อช่วยสนับสนุนให้การค้นคืนข้อมูลนั้นทำได้อย่างรวดเร็วและช่วยรองรับความต้องการของผู้ใช้ในแต่ละเรื่องได้

4. ส่วนการสืบค้นข้อมูลและการแสดงผล

- รับข้อมูลจากผู้ใช้เพื่อนำไปค้นหาในฐานข้อมูล
- มีการแสดงผลที่ยืดหยุ่นได้ สามารถเลือกหัวข้อที่จะแสดงตามแต่ผู้ใช้บริการแต่ละรายต้องการ
- มีการแสดงอันดับของสถาบันการศึกษาที่น่าสนใจในแต่ละสาขาวิชา เพื่อช่วยประกอบในการตัดสินใจในการเลือกสถาบันการศึกษา

1.3 ขั้นตอนการดำเนินงาน

1. ศึกษากระบวนการทำงานของเสิร์ชเอ็นจิ้นในปัจจุบัน โดยศึกษากระบวนการของระบบค้นคืนสารสนเทศ และศึกษาองค์ประกอบของเสิร์ชเอ็นจิ้นที่มีอยู่ในปัจจุบัน

2. ศึกษากระบวนการเปรียบเทียบรูปแบบข้อความ(Pattern Matching) โดยใช้ Regular Expression รวมถึงศึกษารูปแบบในการเขียน regular expression เพื่อใช้งาน

3. ระบุความต้องการในการให้บริการในงานบริการการศึกษา และระบุขอบเขตของข้อมูลที่จะต้องจัดเก็บ

4. ระบุองค์ประกอบสำหรับใช้เสิร์ชเอ็นจิ้นสำหรับงานเฉพาะด้าน โดยออกแบบกระบวนการทำงานของเสิร์ชเอ็นจิ้นสำหรับงานเฉพาะด้าน และศึกษากระบวนการทำงานของ crawler ซึ่งเป็นส่วนประกอบของเสิร์ชเอ็นจิ้นที่ใช้ในการรวบรวมข้อมูลเข้ามาเก็บไว้ในฐานข้อมูล

5. ออกแบบ crawling algorithm เพื่อใช้ในการเก็บรวบรวมข้อมูลตามขอบเขตที่กำหนดไว้ และออกแบบฐานข้อมูลเว็บให้เหมาะสมกับการจัดเก็บและค้นคืนข้อมูล

6. ออกแบบ regular expression เพื่อใช้สำหรับเปรียบเทียบรูปแบบของข้อความ และออกแบบ algorithm ในการดึงข้อมูลที่ต้องการออกจากเอกสาร HTML

7. พัฒนาต้นแบบ crawler สำหรับงานบริการการศึกษา และพัฒนา module สำหรับเปรียบเทียบรูปแบบและดึงข้อมูลในหัวข้อต่างๆ เพื่อให้ crawler เรียกใช้งาน

8. พัฒนาระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา โดยปรับปรุงต้นแบบ crawler และปรับปรุงฐานข้อมูลให้มีความเหมาะสมและมีประสิทธิภาพมากขึ้น

9. ทดสอบการทำงานของระบบ โดยทดสอบความสามารถในการค้นหาข้อมูลเพื่อให้บริการในงานบริการการศึกษา และแก้ไขปรับปรุงข้อผิดพลาดเพื่อให้ระบบสามารถทำงานได้อย่างถูกต้อง

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. เกิดต้นแบบของเสิร์ชเอ็นจิ้นสำหรับการค้นหาข้อมูลของสถาบันการศึกษา ซึ่งสามารถนำไปใช้ประโยชน์ในงานบริการการศึกษาของหน่วยแนะแนวของสถานศึกษาและส่วนราชการได้ทันที
2. ช่วยอำนวยความสะดวกในการค้นหาข้อมูลของสถาบันการศึกษา ทำให้สามารถให้บริการให้อย่างรวดเร็วและมีความถูกต้องแม่นยำ
3. เกิดองค์ความรู้ใหม่ด้านฐานข้อมูลเว็บที่เป็นแบบไดนามิค ซึ่งสามารถปรับปรุงข้อมูลที่เก็บไว้ในฐานข้อมูล เมื่อมีการปรับปรุงเปลี่ยนแปลงหรือแก้ไขข้อมูลข่าวสารของสถาบันการศึกษา
4. สามารถนำกระบวนการไปปรับใช้ได้กับฐานข้อมูลเว็บสำหรับงานเฉพาะด้านอื่นๆ ได้

1.5 เครื่องมือที่ใช้ในการพัฒนาระบบ

1. ฮาร์ดแวร์

   Computer PC มีลักษณะดังนี้

   - CPU Intel Pentium4 processor 2.8 GHz
   - RAM 1 GB
   - Hard Disk 80 GB
   - CD-ROM 40X

2. ซอฟต์แวร์

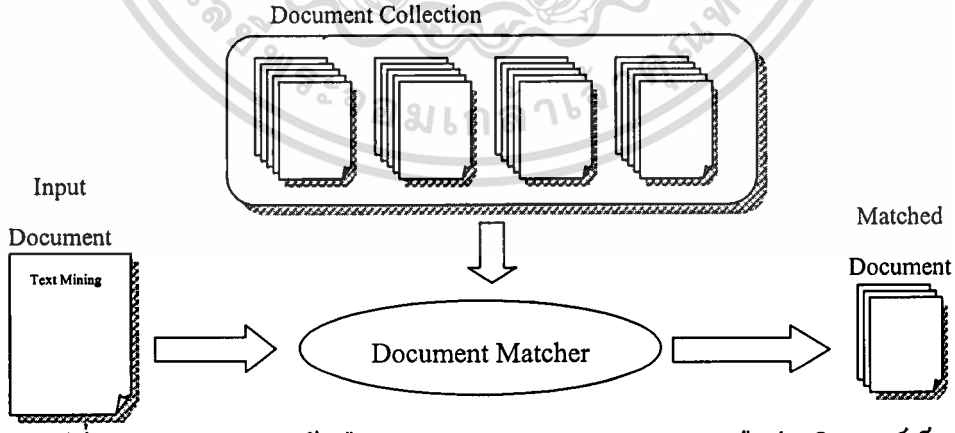
   - Microsoft Windows XP
   - Microsoft Office XP
   - Microsoft Visual Basic 6.0
   - Microsoft SQL Server 2000
   - Macromedia Dreamweaver MX
   - Internet Explorer
   - Internet Information Services(IIS)

บทที่ 2
ทฤษฎีที่ใช้ในระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา

2.1 ระบบค้นคืนสารสนเทศ(Information Retrieval)

ปัจจุบันเทคโนโลยีอินเทอร์เน็ต เติบโตอย่างรวดเร็ว และเอกสารประเภทเท็กซ์ (text document) ได้จัดทำขึ้นเพื่อเผยแพร่ ในเครือข่ายอินเทอร์เน็ตเป็นจำนวนมาก แต่ข้อมูลเหล่านี้มีการเปลี่ยนแปลงอยู่ตลอดเวลา อีกทั้งเว็บเพจนั้นยังมีความซับซ้อนและไม่มีรูปแบบที่แน่นอนเหมือนกับเอกสารทั่วไป ซึ่งความต้องการใช้เอกสารจากผู้ใช้มีเป็นจำนวนมาก แต่ผลของการค้นหาเอกสารบนเว็บส่วนใหญ่ มักจะได้ข้อมูลที่ตรงตามความต้องการของผู้ใช้งานเพียงเล็กน้อยเท่านั้น ดังนั้นจึงมีกระบวนการเพื่อสืบค้นเอกสารบนเว็บที่มีประสิทธิภาพ โดยการนำเอาหลักการของระบบจัดการฐานข้อมูล (Database Management System) และระบบค้นคืนสารสนเทศ (Information Retrieval System) มาใช้สร้างเป็นเครื่องมือในการค้นหาเอกสาร เพื่อดึงเอาเอกสารที่มีอยู่จำนวนมากมายมหาศาลออกมาให้ได้ถูกต้อง รวดเร็วและตรงตามความต้องการมากที่สุด ซึ่งเครื่องมือดังกล่าวนั้น ได้แก่ เว็บเสิร์ชเอ็นจิ้น (web search engine) ที่ใช้กันอยู่ในปัจจุบัน

ในการทำงานของระบบค้นคืนสารสนเทศสำหรับเอกสารที่มีอยู่บนเครือข่ายอินเทอร์เน็ต แสดงได้ดังรูป 2.1



รูปที่ 2.1 การของระบบค้นคืนสารสนเทศสำหรับเอกสารบนเครือข่ายอินเทอร์เน็ต

โดยเอกสารทั้งหมดจะถูกเก็บรวบรวมเอาไว้(Document Collection) เมื่อต้องการค้นคืนข้อมูลที่ต้องการ จะทำการกำหนดเงื่อนไขโดยใช้คำที่เป็นตัวระบุถึงความเกี่ยวข้องสัมพันธ์กับเอกสารที่ต้องการ และนำคำนั้นมาเปรียบเทียบกับเอกสารที่จัดเก็บเอาไว้ และแสดงผลลัพธ์ที่เป็น

เอกสารที่ตรงตามเงื่อนไขที่ผู้ใช้กำหนดมา ซึ่งหลักการนี้ได้นำไปใช้เป็นพื้นฐานในการทำงานของเสิร์ชเอ็นจิ้น(Sholom M. Weiss. 2005)

การค้นหาเอกสารบนเว็บนั้นแตกต่างจากการค้นหาเอกสารทั่วไป ประการแรกคือในเว็บนั้นเก็บเอกสารเป็นจำนวนมากกว่าพันล้านเพจ แต่ในระบบค้นคืนสารสนเทศโดยทั่วไปมักจะใช้กับเอกสารที่เก็บมีปริมาณไม่มากนักและมักจะเป็นเพียงการทำ index เพื่อช่วยในการค้นหาข้อมูลเท่านั้น ตัวอย่างเช่น การค้นหารายชื่อหนังสือหรือการค้นหาชื่อผู้แต่งหนังสือในห้องสมุด เป็นต้น ประการที่สองคือในระบบค้นคืนสารสนเทศแบบเดิมนั้นมักจะใช้กับนักวิจัยหรือผู้ที่มีความชำนาญที่ถูกฝึกฝนมาเพื่อให้สามารถใช้เครื่องมือค้นหาได้ สำหรับเอกสารก็ได้มีการเตรียมไว้อย่างดีและมีการเก็บเป็นหมวดหมู่ตามความสัมพันธ์ของหัวเรื่อง แต่เอกสารบนเว็บนั้นจะถูกสร้างขึ้นด้วยจุดประสงค์ที่หลากหลายและสร้างจากหลายๆแหล่ง อีกทั้งในการค้นหานั้นผู้ใช้ก็ไม่ได้ถูกฝึกฝนมาเพื่อให้ใช้เครื่องมือสืบค้น ดังนั้นเว็บเสิร์ชเอ็นจิ้น จะต้องใช้ระบบค้นคืนสารสนเทศที่มีความสลับซับซ้อนมากกว่าในสมัยก่อน เพื่อใช้ในการค้นหาข้อมูลบนอินเทอร์เน็ต

ในระบบค้นคืนสารสนเทศนั้นจะมีการใช้ query ที่เรียกว่า Searches โดยจะค้นหาส่วนของคำ (search terms) ในเอกสารที่ไม่มีโครงสร้าง ซึ่งผลของการค้นหานั้นจะมีการจัดเรียงลำดับเอกสารที่ตรงกับที่ค้นหามากที่สุด ซึ่งแตกต่างจากระบบฐานข้อมูลที่จะใช้ query ทั่วๆไป ซึ่งมักจะใช้กับข้อมูลที่มีโครงสร้างแน่นอน โดยการค้นหาจะใช้คำที่ต้องการค้นหาร่วมกับ Boolean operator ซึ่งผลที่ได้จะเป็นเอกสารที่มีคำที่ต้องการปรากฏอยู่ในเอกสารนั้นและไม่มีการจัดเรียงลำดับเหมือนกับระบบค้นคืนสารสนเทศ

ในระบบค้นคืนสารสนเทศนั้นเมื่อมีเอกสารใหม่จะต้องทำการสร้าง index เพื่อเพิ่มความรวดเร็วในการค้นหา ซึ่งหากไม่มีการปรับปรุง index อยู่อย่างสม่ำเสมอก็อาจจะค้นเอกสารที่ต้องการไม่พบได้ ซึ่งต่างจากระบบฐานข้อมูลที่จะสนับสนุนการเปลี่ยนแปลงของข้อมูลมากกว่า (Raghu Ramakrishnan. 2003)

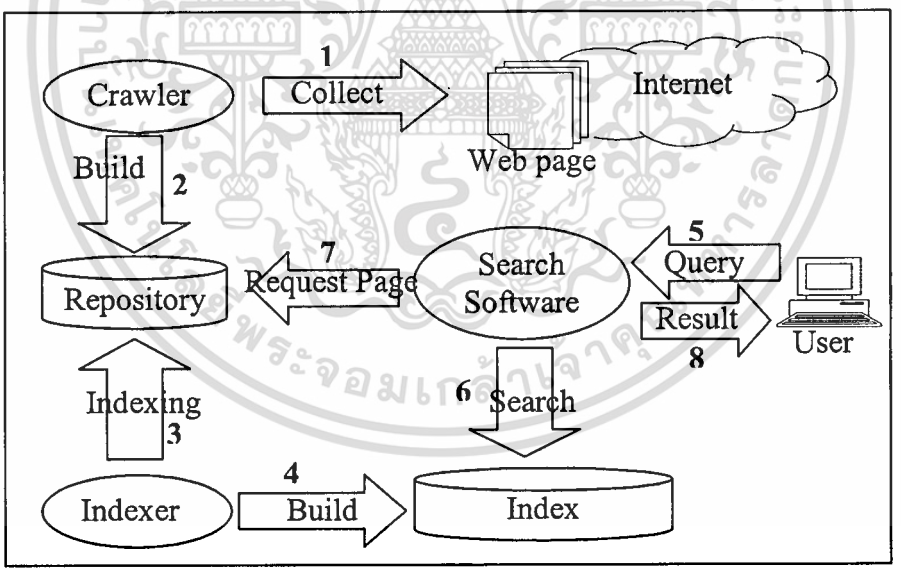
ในการค้นหาข้อมูล เว็บเพจบนเครือข่ายอินเทอร์เน็ตให้ได้ข้อมูลตรงตามความต้องการของผู้ใช้งานมากที่สุด ระบบค้นคืนสารสนเทศจะต้องสามารถแปลงข้อมูลที่ผู้ใช้งานต้องการค้นหาให้อยู่ในรูปแบบที่สามารถนำไปใช้ค้นหาข้อมูลได้ถูกต้องตรงกับความต้องการของผู้ใช้งานมากที่สุด โดยเป้าหมายหลักของระบบค้นคืนสารสนเทศ คือ เพื่อค้นคืนเอกสารทั้งหมดที่ตรงกับคำถามที่ผู้ค้นหาป้อนข้อมูลลงไป โดยจะต้องดึงเอกสารที่ไม่เกี่ยวข้องออกมาให้น้อยที่สุด หรือไม่ดึงเอกสารที่ไม่มีความเกี่ยวข้องกันออกมาเลย

2.2 การทำงานของ Search Engine

ในการทำงานของ Search Engine สามารถแบ่งองค์ประกอบต่างๆตามหน้าที่การทำงานได้เป็น 3 ส่วนหลัก ดังนี้

- Crawler หรือเรียกว่า Spider หรือ Robot (Martijn Koster. 1994) ทำหน้าที่เดินทางไปยังไซต์ต่างๆเพื่อสะสมไฟล์ HTML ของเว็บเพจ แล้วติดตามลิงค์จากเว็บเพจนั้นไปยังเว็บเพจอื่นๆ ภายหลังจากที่ crawler ได้อ่านเว็บเพจใดๆแล้ว จากนั้นจะกลับไปยังไซต์ที่เคยสำรวจแล้วเพื่อตรวจสอบการเปลี่ยนแปลงตามจังหวะเวลาที่กำหนด
- Indexer เป็นส่วนที่ทำหน้าที่สร้างดัชนีค้นหาจากไฟล์ HTML ที่ crawler หามา เว็บเพจใดๆจะสามารถสืบค้นได้จาก Search Engine ต่อเมื่อเว็บเพจนั้นผ่านการทำดัชนีมาแล้วเท่านั้น หากมีการเปลี่ยนแปลงกับเว็บเพจจะต้องแก้ไขข้อมูลดัชนีใหม่
- Search Engine Software เป็นโปรแกรมทำหน้าที่รับคำศัพท์ที่ต้องการค้นหาจากผู้ใช้และค้นหาเว็บเพจที่ตรงกับความต้องการในฐานข้อมูล

ซึ่งการทำงานทั้ง 3 ส่วนนี้สามารถแสดงเป็นภาพรวมได้ดังรูปที่ 2.2



รูปที่ 2.2 การทำงานของ Search Engine

โดยขั้นตอนการทำงานของ Search Engine เป็นดังนี้

1. crawler จะไปทำการอ่านเว็บเพจต่างๆที่อยู่บนอินเทอร์เน็ต โดยจะอาศัย hyperlink ในเพจนั้นเพื่อที่จะไปอ่านเพจอื่น ซึ่งสามารถที่จะระบุขอบเขตที่จะให้ crawler ไปอ่านเพจได้

2. หลังจากที่ crawler ได้อ่านเพจจากอินเทอร์เน็ตแล้ว จะทำการบีบอัดเว็บเพจนั้นแล้วเก็บลงใน repository

3. Indexer จะทำการอ่านเว็บเพจที่อยู่ใน repository เพื่อจัดทำเป็น index

4. indexer จะทำการสร้างฐานข้อมูลของ index ซึ่งจะช่วยให้สามารถค้นหาคำได้รวดเร็วยิ่งขึ้น

5. ผู้ใช้ต้องการค้นหาเอกสาร โดยจะส่งคำที่จะใช้ในการค้นหามาให้กับ search software

6. search software จะนำคำที่ผู้ใช้ป้อนเข้ามา ไปค้นหาในฐานข้อมูลของ index เพื่อหาเว็บเพจที่ตรงกับความต้องการของผู้ใช้

7. เมื่อพบคำที่ตรงกับที่ต้องการแล้ว search software จะไปดึงเอารายการของเอกสารออกมาจาก repository

8. search software จะทำหน้าที่จัดเรียงเอกสาร และส่งผลลัพธ์ไปให้กับผู้ใช้

ตัวอย่างของการทำงาน เช่น หลังจากที่ crawler ได้ไปอ่านเว็บเพจต่างๆแล้ว และมีการจัดทำ index เรียบร้อย เมื่อผู้ใช้ต้องการค้นหาคำว่า "University" จะส่งข้อความค้นหา(query string) ผ่านไปยัง search software จากนั้นตัว search software จะไปค้นหาคำที่ส่งมานั้นในฐานข้อมูล index ที่ indexer สร้างเอาไว้ ถ้าในฐานข้อมูล index มีคำว่า "University" ปรากฏอยู่ ก็จะส่งตำแหน่งที่เก็บอยู่ใน repository จากนั้น search software จะดึงเอารายการของเอกสารออกมาและส่งผลกลับไปยังผู้ใช้

2.3 Search Engine Robots

Search engine Robot หรือ Web Crawler คือโปรแกรมที่ทำหน้าที่ท่องไปตามเว็บไซต์เก็บเอาข้อมูล และลิงค์ที่เจออยู่ภายในหน้าเว็บเพจกลับมาทำเป็น index สำหรับใช้ช่วยในการค้นหาข้อมูลใน Search Engine(Avi Rappaport. 1999)

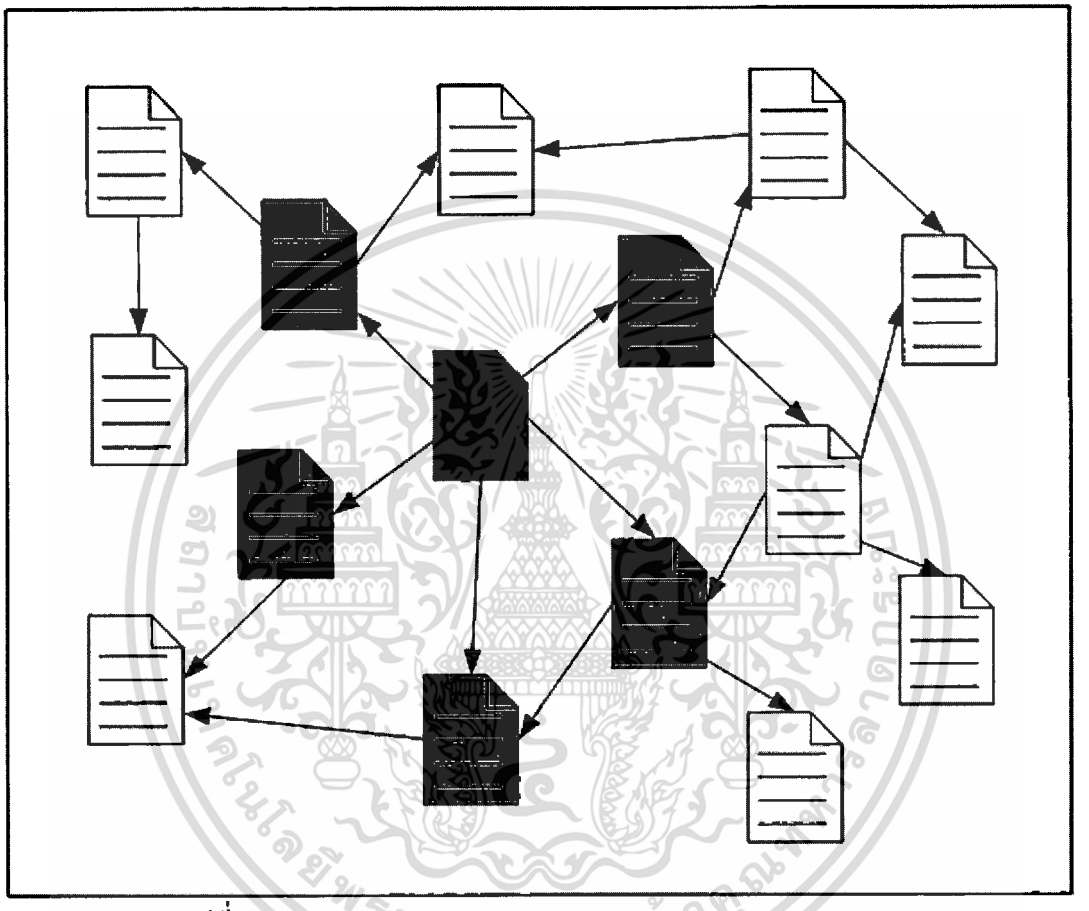
Search Engine ที่ได้รับความนิยมดังๆ ในต่างประเทศ เช่น Google,Altavista, Hotbot หรือ All The Web ล้วนแล้วแต่ใช้ Web Crawler เป็นตัวรวบรวมข้อมูลมาใส่ยังเว็บไซต์ของตัวเอง โดยลิงค์ที่ Web Crawler อ่านเจอทุกลิงค์จะถูก Web Crawler นำเอาไปเป็นอินพุทสำหรับท่องไปยังเว็บไซต์เหล่านั้นต่อไป

วิธีที่ Web Crawler ใช้ในการเดินทางท่องไปตามเว็บไซต์สามารถแบ่งออกได้ดังต่อไปนี้

2.3.1 Breadth-First Crawling

วิธีนี้เริ่มต้นการทำงานโดย web Crawler จะได้รับ URL เริ่มต้นมา 1 URL หรือมากกว่านั้น เพื่อใช้เป็นจุดเริ่มต้นในการเดินทางท่องไปตามเว็บไซต์เพื่อเก็บรวบรวมข้อมูล

หลังจาก Web Crawler ท่องไปยังเว็บไซต์เหล่านั้นแล้วก็จะค้นพบ URL ในเว็บเหล่านั้นอีก และก็จะนำ URL ทั้งหมดที่ค้นพบในเว็บไซต์เหล่านั้นกลับมาเก็บยังฐานข้อมูลของ Web Crawler ก่อนจะเดินทางไปยัง URL ที่เจอเหล่านั้นต่อไปตามลำดับ ดังแสดงได้ในรูปที่ 2.3

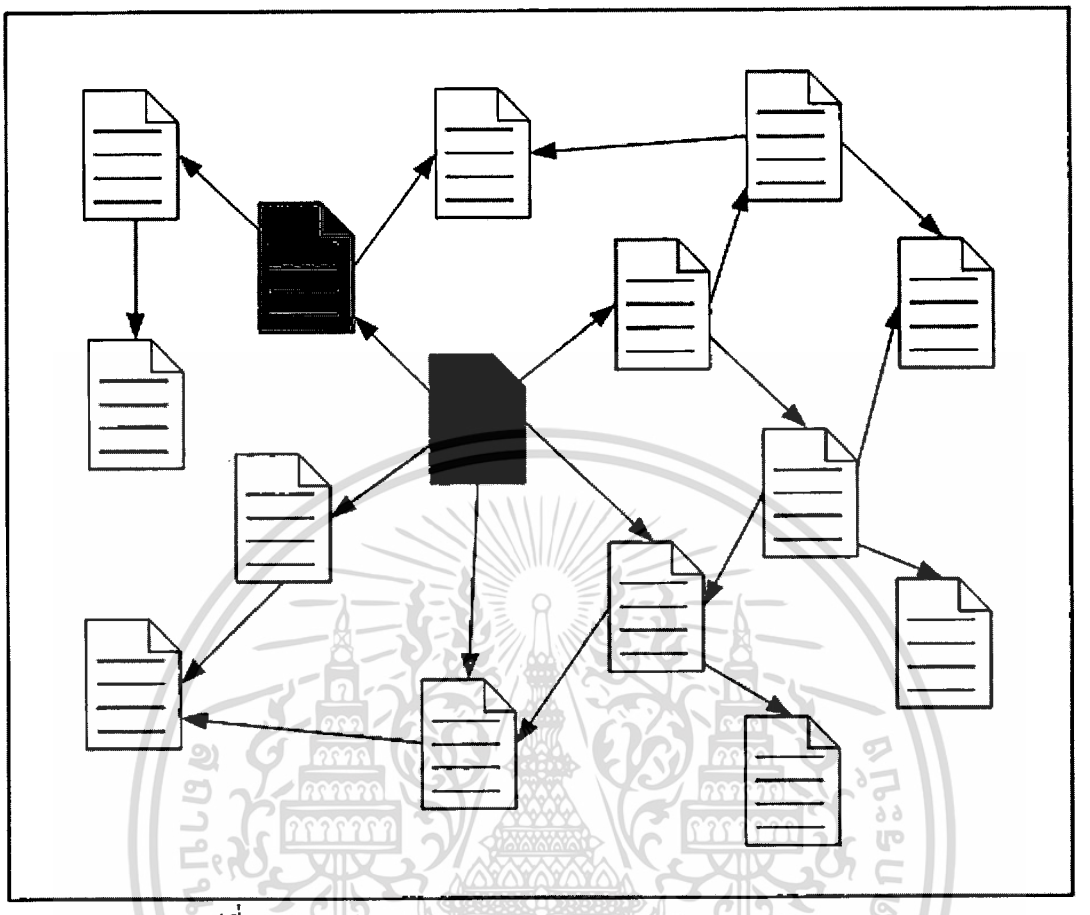


รูปที่ 2.3 การทำงานของ Web Crawler แบบ Breadth-First Crawling

จากรูปที่ 2.3 เริ่มต้นจะถูกแสดงด้วยสีที่เข้มที่สุด และ URL ลำดับถัดไปที่ Web Crawler จะท่องต่อไปจะถูกแสดงด้วยสีที่เข้มน้อยลงมาตามลำดับจนถึง URL สุดท้าย จะถูกแสดงด้วยสีขาวดังรูป

2.3.2 Depth-First Crawling

วิธีนี้เริ่มต้นการทำงานโดย Web Crawler จะได้รับ URL เริ่มต้นมา 1 URL หรือมากกว่านั้นเพื่อใช้เป็นจุดเริ่มต้นในการเดินทางออกไปเก็บข้อมูล และ URL ลำดับแรกในเพจต่อๆไปที่ถูก Web Crawler ค้นพบ จะถูกนำมาเป็นอินพุทสำหรับ Web Crawler ในการเดินทางออกไปเก็บข้อมูลต่อไป ดังแสดงได้ในรูปที่ 2.4



รูปที่ 2.4 การทำงานของ Web Crawler แบบ Depth-First Crawling

จากรูปที่ 2.4 URL แรกที่ web Crawler ใช้เป็นจุดเริ่มต้นจะถูกแสดงด้วยสีที่เข้มที่สุด และ URL ถัดไปที่ Web Crawler จะท่องไปจะถูกแสดงด้วยสีที่เข้มน้อยลงมาตามลำดับจนถึง URL สุดท้ายจะถูกแสดงด้วยสีขาวดังรูป

2.4 Search Engine สำหรับงานเฉพาะด้าน

เนื่องจากปริมาณเว็บในเครือข่ายอินเทอร์เน็ตในปัจจุบันมีมากมายมหาศาล การใช้ Web Search Engine ทั่วไป เช่น Google,AltaVista,Lycos นั้น ไม่สะดวกกับผู้ที่มีความต้องการในการค้นหาข้อมูลเฉพาะด้านเท่าใดนัก เนื่องจากปริมาณของผลลัพธ์ที่ได้มากการค้นหานั้นมีปริมาณมาก ซึ่งอาจจะมีหน้าเว็บเพจที่ไม่ได้มีความเกี่ยวข้องกับที่ต้องการมาด้วย จึงได้มีการพัฒนา Search Engine สำหรับงานเฉพาะด้าน(Specific Domain Web Search Engine) ซึ่งเป็นSearch Engine ที่ใช้ในการค้นหาเอกสารที่มีขอบเขตของเนื้อหาในแบบเดียวกัน เช่น การค้นหาข้อมูลการเปิดหลักสูตรการสอนในสถาบันการศึกษาต่างๆ Search Engine นี้จะทำการค้นหาข้อมูลในสถาบันการศึกษาที่มีอยู่บนอินเทอร์เน็ต แล้วส่งผลกลับมายังผู้ใช้ ซึ่งผลลัพธ์ของข้อมูลที่ได้นั้น จะได้มา

จากสถาบันการศึกษาเท่านั้น ทำให้ได้ข้อมูลที่ตรงตามต้องการ และลดจำนวนเพจที่ไม่เกี่ยวข้องกับที่ต้องการ

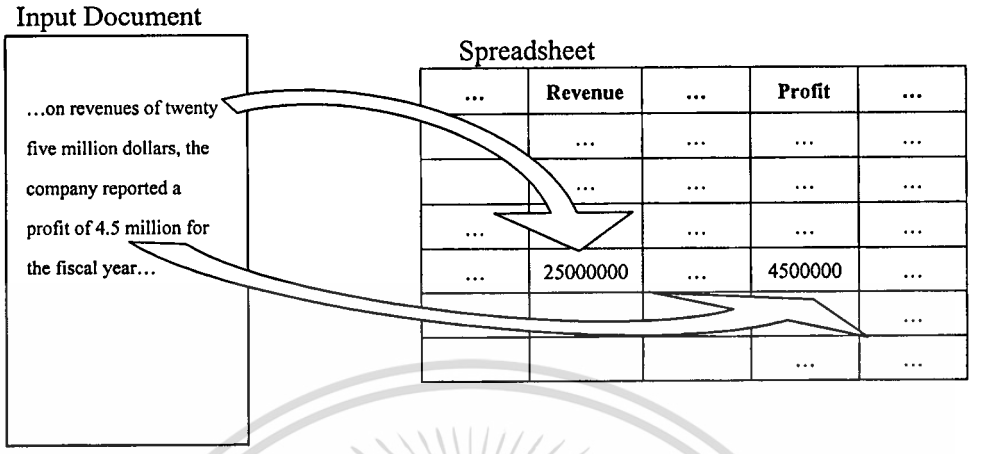
จากการที่ปริมาณของเว็บที่เข้าไปค้นหามีขอบเขตที่จำกัด ต่างจากการค้นหาเว็บในระบบอินเทอร์เน็ตทั้งหมด ทำให้การใช้ Search Engine สำหรับงานเฉพาะด้าน จะลดจำนวนเพจที่ไม่เกี่ยวข้อง และเพิ่มความเร็วในการค้นหา อีกทั้งในการจัดทำ index ก็ทำได้รวดเร็วขึ้นอีกด้วย

หลักการของ Search Engine ประเภทนี้จะเป็นเช่นเดียวกับ Search Engine ทั่วไป คือจะต้องอาศัยการทำงานของ crawler (robot หรือ spider) ในการเก็บรวมรวมเว็บเพจจากแหล่งต่างๆ ซึ่งเราสามารถกำหนดขอบเขตในการที่จะให้ crawler นั้นไปเก็บข้อมูลของเว็บเพจตาม domain name ที่ระบุได้ แต่ปัญหาของSearch Engineนี้คือ อาจจะไม่สามารถดึงข้อมูลที่น่าสนใจที่อยู่นอกเหนือจากขอบเขตที่ค้นหา แสดงให้กับผู้ใช้ได้(Michael Chau and Hsinchun Chen. 2003)

2.5 Information Extraction

Information Extraction เป็นสาขาย่อยของการประมวลผลภาษาธรรมชาติ(Natural Language Processing) โดยมุ่งเน้นในเรื่องการค้นหาความหมายของคำหรือประโยคที่มีในเอกสารที่ไม่มีโครงสร้างแน่นอน ในการพัฒนาโปรแกรมลักษณะนี้ขึ้น จะใช้การเปรียบเทียบคำในเอกสารกับรูปแบบประโยคที่จะจัดเก็บ(Extraction Pattern)(Peter Jackson. 2002) ทั้งนี้ ขึ้นอยู่กับขอบเขตของเรื่องที่สนใจในแต่ละเรื่อง ซึ่งจะสามารถระบุรูปแบบของข้อมูลนั้นได้ โดยมีการนำหลักการของ data mining มาประยุกต์ใช้เพื่อการทำนายข้อมูลที่จะได้มา (Gordon S. Linoff. 2001)

ในฐานข้อมูลนั้น ข้อมูลจะถูกจัดเก็บให้อยู่ในรูปของ field และ table แต่เมื่อข้อมูลนั้นไม่มีโครงสร้างที่แน่นอน เช่นการหาข้อมูลที่ต้องการในเอกสารต่างๆ จำเป็นต้องมีกระบวนการในการแยกเอาข้อมูลนั้นออกมาจากรูปแบบที่ไม่มีโครงสร้างมาให้ได้ ตัวอย่างเช่น ต้องการดึงข้อมูลยอดขายและรหัสโรงงานออกมาจากเอกสารของบริษัท มาจัดเก็บลงในฐานข้อมูล ซึ่งเอกสารเหล่านั้นไม่ได้มีการระบุตำแหน่งของข้อมูลที่ต้องการไว้เป็นที่แน่นอน และไม่สามารถใช้รูปแบบของการหาข้อมูลของเอกสารนี้ไปใช้กับเอกสารอื่นๆได้



รูปที่ 2.5 การดึงข้อมูลที่ต้องการจากเอกสารโดยใช้ Information Extraction

จากรูปที่ 2.5 เป็นตัวอย่างของการทำ Information Extraction เพื่อดึงข้อมูลรายได้และกำไรของบริษัทออกมาจากเอกสารที่แสดงเป็นข้อความ ซึ่งไม่มีโครงสร้างของข้อมูล ในกระบวนการของ Information Extraction นี้ จะต้องหาว่าข้อความที่แสดงถึงรายได้และกำไรของบริษัทนั้นอยู่ตรงส่วนใดของเอกสารและมีรูปแบบเป็นเช่นไร จากนั้นจึงทำการดึงข้อมูลนั้นออกมาจัดเก็บลงฐานข้อมูลที่มีโครงสร้างข้อมูลแน่นอน (Shalom M.Weiss. 2005)

2.6 Regular Expressions

Regular Expressions (regexs) หมายถึงรูปแบบของลำดับหรือกลุ่มของสัญลักษณ์ที่ใช้แทนลำดับ หรือกลุ่มของอักขระตามที่ต้องการ ซึ่งซอฟต์แวร์หลายๆตัวก็ได้นำหลักการนี้มาใช้สำหรับการทำ pattern-matching แต่อย่างไรก็ดี Regular Expressions เป็นเพียงรูปแบบที่ใช้งานทั่วๆไปเพื่อใช้อธิบายรูปแบบเท่านั้น ซึ่งรูปแบบนี้ไม่ได้ใช้เฉพาะเจาะจงสำหรับภาษาทางด้าน Programming ภาษาใดภาษาหนึ่งหรือ tool ตัวใดตัวหนึ่งเท่านั้น(Peter Jackson. 2002)

จากตัวอย่างต่อไปนี้ จะแสดงรูปแบบของลำดับของกลุ่มอักขระ(regular set) อย่างง่ายๆ ดังนี้

a(b|c)*a

ซึ่ง Regular Expression ที่เห็นนี้สามารถแสดงเป็นกลุ่มของคำในรูปแบบต่างๆ(set of string) ดังนี้

L = {aa, aba, aca, abba, abca, acba, acca, …}

โดยที่ (b|c) นั้นจะแทนการเลือกระหว่าง b หรือ c และ '*' นั้นจะหมายถึง อักขระนั้นสามารถเกิดขึ้นได้หลายครั้งหรือไม่เกิดขึ้นก็ได้

ตัวอย่างต่อไปนี้จะแสดงถึงการใช้ Regular Expreession เพื่อใช้สำหรับการระบุชื่อบุคคลดังนี้

{Mr.|Mrs.|Ms.|Dr.} {A|B|C|...|Z}. LASTNAME

โดยที่ LASTNAME นั้นหมายถึง last name ใดๆ เช่น last name ที่ปรากฏอยู่ในสมุดโทรศัพท์ เป็นต้น ส่วน element อื่นๆ เช่น "Mr." หรือ "A" หรือ "." นั้น จะมีความหมายในตัวเองอยู่แล้ว

สำหรับหลักการเบื้องต้นในการกระจายคำหรือข้อความ โดยใช้ Regular Expression จะกระทำเป็นขั้นตอนในแต่ละ module เป็นลำดับขั้น ดังแสดงในรูป 2.6

รูปที่ 2.6 Module ที่ใช้ในกระบวนการตัดคำหรือข้อความโดยใช้ Regular Expression

โดยก่อนเริ่มการทำงานจะต้องทำการจัดเตรียมขอบเขตของคำหรือประโยคของภาษาที่จะเกี่ยวข้องกับงานนั้นๆ เพื่อที่จะลดปริมาณของคำหรือข้อความที่ไม่เกี่ยวข้องกับขอบเขตที่สนใจอยู่

ในการทำงานนั้นจะเริ่มจาก Tokenizer ทำการแตกประโยคออกเป็นคำโดยแตกตามขอบเขตที่ได้ระบุไว้ในตอนแรก เมื่อได้คำที่เป็นส่วนหนึ่งของประโยคมาแล้ว(Part of speech: POS) จะทำการจัดกลุ่มของคำที่มีความเกี่ยวข้องกับขอบเขตที่ได้กำหนดไว้ ซึ่งในที่นี้จะเกี่ยวข้องในเรื่องของการหาชื่อ จากนั้นก็จะทำการค้นหารูปแบบที่ใกล้เคียงกับที่ระบุใน Reguler Expression และสุดท้าย เมื่อมีการค้นพบรูปแบบที่เกี่ยวข้องกับเนื้อหาที่ต้องการแล้ว ก็จำเป็นที่ต้องมีการจัดเก็บ template ของรูปแบบ เพื่อใช้เป็นแหล่งข้อมูลสำหรับการทำงานในเรื่องที่เกี่ยวข้องนั้นๆต่อไป

เพื่อให้การทำงานมีความถูกต้องแม่นยำมากยิ่งขึ้น มักจะมีการตรวจสอบโดยใช้คนเข้ามาช่วย เพื่อที่จะนำไปใช้สำหรับเทคนิคในการทำนาย ของ Machine-Learning โดยระบุจำนวนของ process ที่จำเป็นสำหรับการดึงข้อมูลออกของเอกสาร(Shalom M.Weiss. 2005)

สัญลักษณ์ของ Regular expression สรุปได้ดังนี้(A.M. Kuchling)

ตารางที่ 2.1 สัญลักษณ์ของ Regular Expression

สัญลักษณ์	คำอธิบาย
^	คำ/อักษรที่อยู่หน้าเครื่องหมายนี้ ต้องเป็นคำขึ้นต้นของข้อความที่นำมาตรวจสอบ
$	คำ/อักษรที่อยู่หน้าเครื่องหมายนี้ ต้องอยู่ตอนท้ายของข้อความที่นำมาตรวจสอบ
+	คำ/อักษรที่อยู่หน้าเครื่องหมายนี้ ต้องมีปรากฏในคำที่นำมาตรวจสอบ อย่างน้อย 1 ตัว
?	คำ/อักษรที่อยู่หน้าเครื่องหมายนี้ อาจจะมีปรากฏในคำที่นำมาตรวจสอบ หรือไม่ก็ได้ ถ้ามีจะมีกี่ตัวก็ได้
*	เหมือนกับ ?
|	เสนอทางเลือกอย่างใดอย่างหนึ่ง เช่น • (A|B) เป็นการบอกว่า จะใช้ A หรือ B ก็ได้ • (A|BC)DE จะเป็น ADE หรือ BCDE ก็ได้
[ ]	ใช้ระบุตำแหน่งในคำว่า ในตำแหน่งนี้จะมีตัวอักษรอะไรได้บ้าง เช่น • "[AB] เป็นการกำหนดว่า คำที่นำมาตรวจสอบ ต้องเป็นตัว Aหรือ ตัว B เท่านั้นจึงจะผ่าน มีความหมายเช่นเดียวกับ A|B • ^[a-zA-Z] เป็นการบอกว่า คำที่นำมาตรวจสอบต้องขึ้นต้นด้วยตัวอักษร จะเป็นตัวเล็ก คือ a ถึง z หรือ ตัวใหญ่ คือ A ถึง Z ก็ได้ • [0-9]% เป็นการบอกว่า ให้มีตัวเลข 1 ตัว เลขอะไรก็ได้ เลข 0 ถึง เลข 9 ต่อด้วยเครื่องหมาย % • ^[0-9]+$ ให้มีเฉพาะตัวเลข 0-9 อย่างน้อย 1 ตัว แต่ห้ามมีตัวอักษร • ^[AB]{3}[-][0-9]$ ขึ้นต้นด้วยตัว A หรือ B จำนวน 3 ตัว ต่อด้วยเครื่องหมาย – และจบด้วยตัวเลข 0-9 เช่น ABA-5 , AAA-3 เป็นต้น สิ่งต่อไปนี้จะไม่ผ่านหรือเป็นเท็จ เช่น AAABB เพราะ ตัวที่ 4 ไม่ใช่เครื่องหมาย – และตัวสุดท้ายไม่ใช่ตัวเลข ไม่ว่าตัวอักษร หรือสัญลักษณ์ใด ๆ ที่อยู่ภายในเครื่องหมาย [ ] จะกลายเป็นสัญลักษณ์ธรรมดา เช่น + กลายเป็นเครื่องหมายบวก แทนที่จะหมายถึงว่า ต้องมีตัวอักษรอย่างน้อย 1 ตัว

ตาราง 2.1 (ต่อ)

สัญลักษณ์	คำอธิบาย
.	ใช้แทนตัวอักษรใดก็ได้
\	สัญลักษณ์ที่ใช้กับอักขระพิเศษเพื่อแสดงว่าอักขระนั้นเป็นส่วนหนึ่งของข้อความ มีดังนี้ ^ , $, (,) , . , [, | , * , ? , + , \ , และ {
\s	ช่องว่าง หรือ whitespace
[^...]	ตัวอักษรตัวใดๆก็ได้ที่ไม่ได้อยู่ในกลุ่มของตัวอักษรที่เป็นตัวเลือก
{ }	แสดงจำนวนครั้งที่ซ้ำกัน เช่น • AB{2} หมายถึงให้มีตัว ข จำนวน 2 ตัว เช่น ABB • AB{2,} หมายถึงให้มีตัว ข อย่างน้อย 2 ตัว เช่น ABBBB • AB{3,5} หมายถึงให้มีตัว ข จำนวน 3-5 ตัวเท่านั้น คือ ABBB , ABBBB และ ABBBBB
( )	ใช้รวมกลุ่มเข้าด้วยกันเป็นส่วนเดียวกัน เช่น • A(BC)* หมายถึง ตัว A และอาจจะตามด้วยตัว BC หรือไม่มีตัว BC ก็ได้ เครื่องหมาย * แสดงว่าจะมีหรือไม่ก็ได้ • A(BC){1,5} หมายถึง ตัว A แล้วจะตามด้วย BC จำนวน 1-5 ชุด เช่น ABCBCBC หรือ ABCBC ก็ได้

ตัวอย่างการเขียน Regular Expression

ตารางที่ 2.2 ตัวอย่างการเขียน Regular Expression

รูปแบบการเขียน	คำอธิบาย
^be	ข้อความที่ขึ้นต้นด้วย be
^[be]	ข้อความที่ขึ้นต้นด้วย b หรือ e
[^be]	ข้อความใดๆที่ไม่ได้ขึ้นต้นด้วย b และ e
ion$	ข้อความที่ลงท้ายด้วยหรือจบท้ายด้วย ion
[ion]$	ข้อความที่ลงท้ายหรือจบท้ายด้วย i หรือ o หรือ n

ตาราง 2.2 (ต่อ)

รูปแบบการเขียน	คำอธิบาย
<H[1-6]>	ข้อความที่มี <H1> <H2> <H3> <H4> <H5> หรือ <H6> อยู่
<H[^4-6]>	ข้อความที่ไม่มี <H4> <H5> และ <H6> อยู่ด้วย
[0-9.]	ข้อความที่มีตัวเลขใดๆระหว่าง 0 ถึง 9 หรือ จุด อยู่ด้วย
^Subject$	ข้อความที่มีคำว่า Subject เท่านั้น
^(From|Subject|Date):	ข้อความขึ้นต้นด้วยคำว่า From หรือ Subject หรือ Date และตามด้วย :
^[0-9]+$	ข้อความที่มีตัวเลข 0-9 เท่านั้นและอย่างน้อยหนึ่งตัว
^[1-9][0-9]*$	ข้อความที่ขึ้นต้นด้วยเลข 1-9 และอาจจะต่อด้วย เลข 0-9 กี่ตัวก็ได้ ในกรณีนี้ ถ้าขึ้นต้นเป็นเลข 0 ก็จะไม่ผ่าน
^(0|[1-9][0-9]*)$	ข้อความที่อาจจะขึ้นต้นด้วยเลข 0 หรือเลข 1-9 ก็ได้ และอาจจะต่อด้วยเลข 0-9 ในกรณีนี้ ใช้ตรวจสอบการพิมพ์ที่เป็นตัวเลขตั้งแต่ 0 ขึ้นไป ถ้ามีตัวอักษร ก็จะไม่ผ่านการตรวจสอบ
^(0|-?[1-9][0-9]*)$	เป็นข้อความที่เหมือน ^(0|[1-9][0-9]*)$ เพียงแต่ ถ้าไม่ขึ้นต้นด้วยเลข 0 สามารถมีเครื่องหมาย ลบ ได้ หรือจะไม่มีเครื่องหมายลบ ก็ได้
^[0-9]+(\.[0-9]+)?$	ข้อความที่ขึ้นต้นด้วย 0-9 อย่างน้อย 1 ตัว และอาจจะมี จุดและต่อด้วยตัวเลข 0-9 อย่างน้อย 1 ตัว อย่างนี้ เป็นการบอกว่าจะทศนิยมหรือไม่มีก็ได้
^[0-9]+(\.[0-9]{2})?$	เป็นการบังคับว่า ข้อความนี้ถ้ามีทศนิยม ทศนิยมต้องมี 2 ตำแหน่งเท่านั้น เครื่องหมาย {} กำหนดว่าจะต้องมีซ้ำกี่ครั้ง
^[0-9]+(\.[0-9]{1,2})?$	เป็นข้อความที่อนุญาตให้มีทศนิยม 1 หรือ 2 ตำแหน่ง โดยระบุจำนวนทศนิยม ในระหว่างเครื่องหมาย { และ }
^[0-9]{1,3}(,[0-9]{3})*(\.[0-9]{1,2})?$	เป็นข้อความที่ต้องขึ้นต้นด้วยตัวเลข 0-9 หรือ อาจจะตามด้วยเครื่องหมาย “,” และตัวเลข 0-9 อีก 3 ตัว และอาจจะต่อด้วยทศนิยม 1 หรือ 2 ตำแหน่ง
^([0-9]+|[0-9]{1,3}(,[0-9]{3})*) (\.[0-9]{1,2})?$	ข้อความที่เหมือนข้างบน แต่มีการกำหนดให้การมีเครื่องหมายคอมม่า อาจจะมีหรือไม่มีก็ได้
([0-9]{1,3}(\.[0-9]{1,3}){3})	ข้อความที่ตรงตามรูปแบบของหมายเลข IP เช่น 127.0.0.1

2.7 วิธีการระบุ URL

เมื่อมีการสร้าง link ไปยังเอกสารหรือรูปภาพบนเว็บ จะต้องคำนึงถึงวิธีการที่จะอ้างไปยังสิ่งเหล่านั้น ซึ่งมีด้วยกัน 2 วิธีคือ(Jennifer Kyrnin. 2005)

- Absolute Paths
- Relative Paths

2.7.1 Absolute Path URLs

Absolute Path เป็นการระบุตำแหน่งที่ตั้งของสิ่งที่อ้างถึงอย่างเฉพาะเจาะจง ซึ่งจะระบุโดเมนเนมเข้าไปด้วย โดยทั่วไป เมื่อมีการอ้างถึงโดยใช้ Absolute path จะใช้ URL เป็นตัวระบุเลย ตัวอย่างเช่นเว็บปัจจุบันนี้มี Absolute path คือ http://webdesign.about.com/library /weekly/aa040502a.htm

เรามักจะใช้ Absolute path เมื่อสิ่งที่เราต้องการอ้างถึงนั้นอยู่ในโดเมนอื่น ตัวอย่างเช่น หากเราต้องการอ้างไปยัง site ของ Graphic Design Guide จะต้องระบุโดเมนเข้าไปด้วย ดังนี้คือ http://graphicdesign.about.com/

2.7.2 Relative Path URLs

Relative Path จะขึ้นอยู่กับตำแหน่งที่เก็บเว็บเพจที่มี link นั้นปรากฏอยู่ โดยหลักทั่วไปในการสร้าง link แบบ Relative Path คือ

- ถ้า link ไปยังเอกสารที่อยู่ใน directory เดียวกัน ไม่ต้องระบุ path
- ถ้า link ไปยังเอกสารที่อยู่ภายใต้ sub-directory ต้องระบุชื่อ sub-directory และคั่นด้วย "/"
- ถ้า link ไปยัง directory ที่สูงกว่าต้องระบุเป็น ../filename

2.8 การวัดประสิทธิภาพของ Web Search Engine

จุดมุ่งหมายของการค้นคืนสารสนเทศ และการทำ web search engine นั้น ก็เพื่อให้สามารถที่จะค้นหาเอกสารให้ได้อย่างมีประสิทธิภาพ ซึ่งมีคำที่ควรทราบ ดังนี้ (Ricardo Baeza-Yates. 1999)

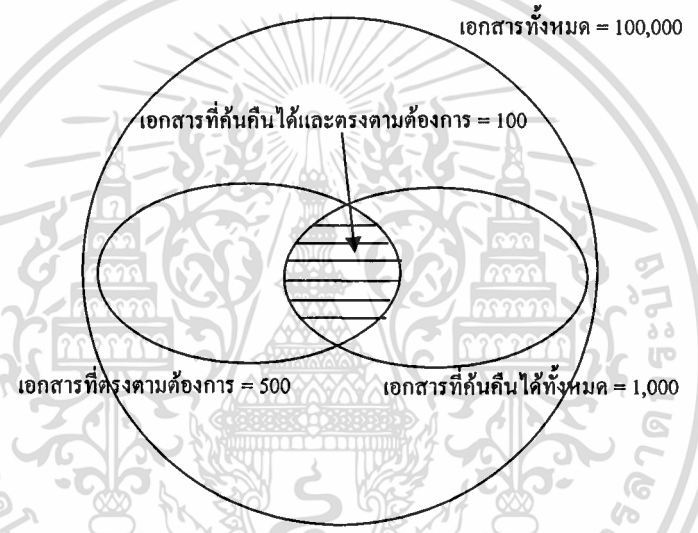
- เอกสารทั้งหมด(Collection) เป็นเอกสารที่ถูกเก็บไว้ในฐานข้อมูล ซึ่งรวบรวมได้จากการที่ crawler ไปอ่านเว็บเพจต่างๆ
- เอกสารที่ค้นคืนได้ทั้งหมด(answer set) เป็นรายการของเอกสารที่ถูกดึงขึ้นมาจากฐานข้อมูล ตามคำค้นหาที่ผู้ใช้ส่งมา
- เอกสารที่ตรงตามความต้องการ(Relevant Documents) เป็นเอกสารที่มีเนื้อหาตรงตามที่ผู้ใช้ต้องการ

- เอกสารที่ค้นคืนได้และตรงตามต้องการ(Relevant Doc in answer set) เป็นรายการเอกสารที่ถูกดึงขึ้นมาแล้วตรงกับความต้องการของผู้ใช้

ซึ่งในการวัดประสิทธิภาพของการค้นคืนสารสนเทศนั้น จะมองใน 2 มุมมองคือ

- Precision หมายถึงอัตราส่วนระหว่างจำนวนเพจที่ถูกค้นคืนได้และตรงตามต้องการเทียบกับจำนวนเพจที่ถูกค้นคืนได้ทั้งหมด

- Recall หมายถึงอัตราส่วนระหว่างจำนวนเพจที่ถูกค้นคืนได้และตรงตามต้องการเทียบกับจำนวนเพจที่ตรงความต้องการทั้งหมด



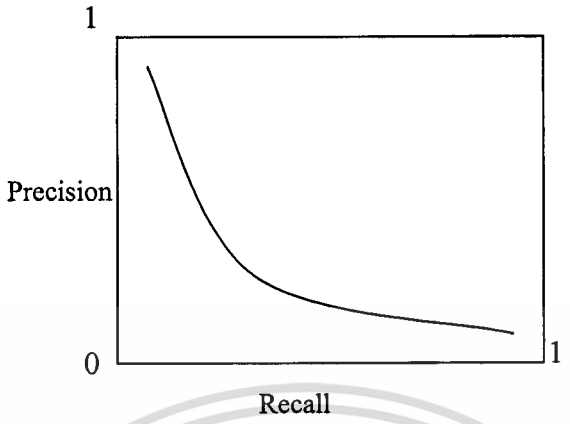
รูปที่ 2.7 กลุ่มของเอกสารเพื่อใช้หา Precision และ Recall

ดังตัวอย่างที่แสดงในรูปที่ 2.7 คือสมมติมีเอกสารอยู่ 100,000 เพจ มีเอกสารที่ตรงตามความต้องการอยู่ 500 เพจ แต่เมื่อทำการค้นคืนโดยใช้ web search engine ได้เอกสารที่ค้นคืนมาทั้งหมด 1,000 เพจ แต่ในเอกสารเหล่านี้มีเอกสารที่ตรงตามต้องการอยู่เพียง 100 เพจ ดังนั้น

Precision = 100/1,000 = 0.1

Recall= 100/500 = 0.2

ในทางอุดมคติ การค้นคืนสารสนเทศต้องได้เฉพาะเอกสารที่ตรงกับกรณีที่ค้นหาเท่านั้น นั่นก็คือค่า Precision และ Recall ต้องเป็น 100%(1.0) ซึ่งในทางปฏิบัติเป็นไปได้ยากมาก เพราะในทางปฏิบัติ ค่า Precision และ Recall จะมีความสัมพันธ์ผกผันกัน ดังแสดงในรูปที่ 2.8



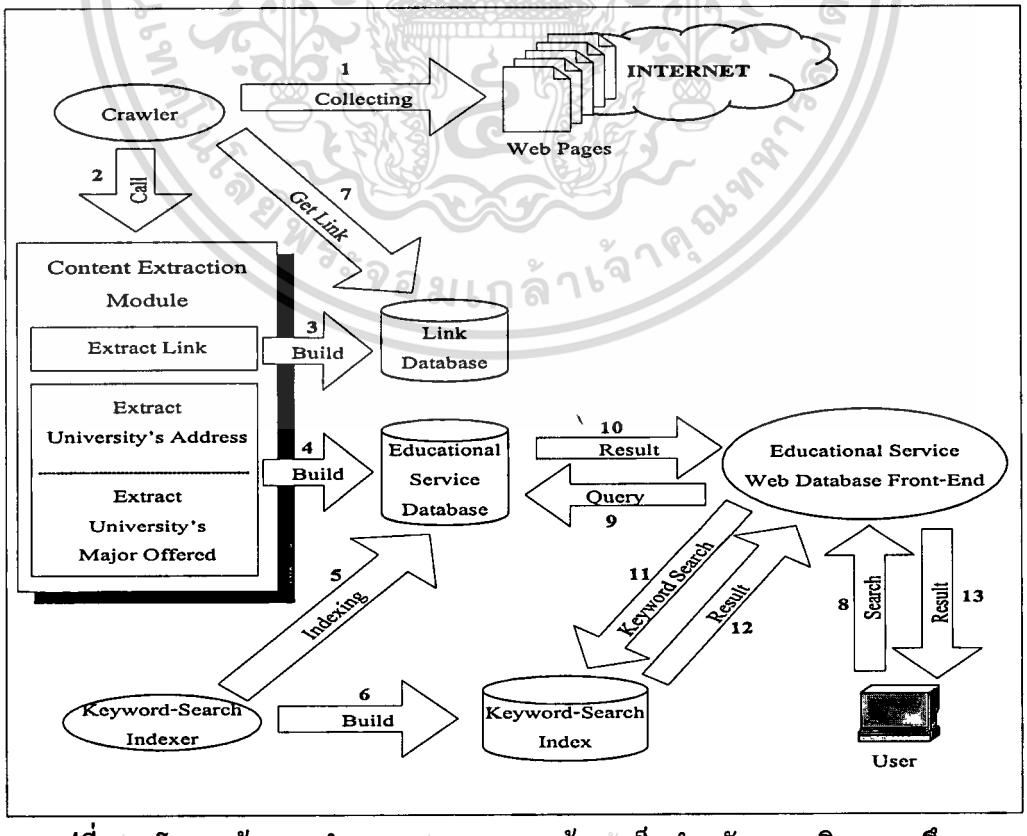
รูปที่ 2.8 กราฟแสดงความสัมพันธ์ระหว่าง Precision และ Recall

จากรูปที่ 2.8 แสดงถึงความสัมพันธ์ที่ผกผันกันระหว่างค่า Precision และค่า Recall โดยเมื่อต้องการให้ค่า Recall สูง จำเป็นที่จะต้องใช้เทอมในการค้นหามากขึ้น ซึ่งจะทำให้ได้เอกสารที่ตรงตามความต้องการมากขึ้น แต่เอกสารอื่นที่ถูกค้นคืนขึ้นมาก็จะมีมากขึ้นด้วย ซึ่งจะส่งผลให้ค่า Precision นั้นต่ำลง เป็นต้น

บทที่ 3
โครงสร้างการทำงานของระบบ

3.1 โครงสร้างของระบบ

ในการพัฒนาระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษานี้ ได้มีการนำหลักการของ Search Engine เข้ามาใช้งาน โดยที่จะเน้นไปในด้านการเก็บรวบรวมข้อมูลจากเอกสาร HTML ของสถาบันการศึกษาต่างๆ ที่มีอยู่บนเครือข่ายอินเทอร์เน็ต ทั้งนี้ ในด้านการทำงานโดยรวม เมื่อเปรียบเทียบกับการทำงานของ Search Engine ทั่วๆไปนั้น จะมีหลักการที่คล้ายๆกัน คือทำการรวบรวมเอกสารที่มีอยู่บนอินเทอร์เน็ตและจัดเก็บลงในฐานข้อมูล เพื่อรองรับการใช้งานของผู้ใช้ที่ต้องการค้นหาข้อมูล แต่ในการพัฒนาระบบฐานข้อมูลสำหรับงานบริการการศึกษา ซึ่งจะต้องทำการรวบรวมและจัดเก็บข้อมูลที่อยู่ภายในสถาบันการศึกษาแต่ละแห่งนั้น จำเป็นจะต้องมีการกรองเอาเฉพาะเอกสารที่ปรากฏอยู่ภายในเว็บของสถาบันการศึกษาเท่านั้น ซึ่งภาพรวมของการทำงานของระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษานั้น แสดงได้ดังรูปที่ 3.1



รูปที่ 3.1 โครงสร้างการทำงานของระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา

จากรูปที่ 3.1 แสดงโครงสร้างการทำงานโดยรวมของระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา ซึ่งสามารถอธิบายขั้นตอนการทำงานได้ดังนี้

1. Web Crawler จะเป็นโปรแกรมสำหรับท่องไปยังเว็บเพจเพื่อดึงเอกสาร HTML ยังตัวระบบ

2. Web Crawler ทำการเรียก Content Extraction Module ซึ่งเป็นโมดูลสำหรับดึงข้อความต่างๆที่อยู่ภายในเอกสาร ประกอบด้วย 3 ส่วนคือ

- Extract Link จะเป็นการดึงเอา hyperlink ที่ปรากฏในเอกสาร HTML
- Extract University's Address จะดึงข้อความระบุที่ตั้งของสถาบันการศึกษาที่ปรากฏในเอกสาร HTML โดยมีการคิดค่าน้ำหนักของข้อความ และใช้ Regular Expression ในการเปรียบเทียบรูปแบบข้อความเพื่อดึงข้อความระบุที่ตั้งสถาบันการศึกษาที่มีรูปแบบตรงกับที่กำหนดไว้
- Extract University's Major Offered จะดึงสาขาวิชาที่สถาบันการศึกษาเปิดสอน โดยเปรียบเทียบข้อความกับรายชื่อของสาขาวิชาที่ได้ทำการจัดเก็บเอาไว้ในฐานข้อมูล

3. ในส่วนการทำงานของ Extract Link จะค้นหา hyperlink ที่ปรากฏในเอกสาร HTML โดยพิจารณาจาก tag<A> และดึงข้อความมาจัดเก็บลงฐานข้อมูล

4. การทำงานในส่วนของ Extract University's Address และ Extract University's Major Offered จะดึงข้อความที่ระบุที่ตั้งสถาบันการศึกษาและสาขาวิชาที่สถาบันการศึกษาเปิดสอนออกจากเอกสาร HTML และนำข้อความเหล่านั้นจัดเก็บลงในฐานข้อมูลสำหรับงานบริการการศึกษา

5. ข้อความที่ถูกจัดเก็บในฐานข้อมูลสำหรับงานบริการการศึกษา จะถูก Keyword-Search Index ทำการตัดข้อความออกเป็นส่วนๆเพื่อสร้างเป็น index ของคำ

6. คำต่างๆที่ถูกจัดการโดย Keyword-Search Index ถูกจัดเก็บในฐานข้อมูล Index ซึ่งจะถูกใช้งานโดย Educational Service Web Database Front-End

7. เมื่อทำการประมวลผลเอกสาร HTML ของเว็บเพจนั้นๆเสร็จแล้ว Web Crawler จะทำการดึง hyperlink ที่ถูกจัดเก็บไว้มาทำการท่องไปยังเว็บเพจของ hyperlink นั้นต่อไป โดยการท่องเว็บเพจของ Web Crawler นั้นจะเป็นแบบ Breath-First Crawling

8. ในส่วนการติดต่อจากผู้ใช้ เริ่มจากการที่ผู้ใช้ทำการค้นหาข้อมูลจาก Front-End ของระบบ ซึ่งผู้ใช้สามารถทำการค้นหาได้ 2 วิธีคือ การท่องไปตาม link ภายใน Front-End และการที่ผู้ใช้ทำการค้นหาโดยใช้คำอิสระ(Keyword Search)

9. กรณีที่ผู้ใช้ค้นหาโดยการท่องไปตาม link ภายใน Front-End จะทำการดึงข้อมูลที่มีอยู่ในฐานข้อมูลได้โดยตรง

10. ข้อมูลที่ค้นคืนได้จะถูกส่งค่ากลับไปยัง Front-End

11. ในกรณีที่มีการใช้คำค้นหา Front-End จะไปทำการตรวจสอบคำจาก Keyword-Search Index ซึ่งจะทำการเปรียบเทียบคำที่ใช้ค้นหากับ index

12. ข้อมูลที่ค้นคืนได้จะเป็นข้อมูลที่ตรงกับคำที่ใช้ในการค้นหา ซึ่งจะถูกส่งกลับไปยัง Front-End

13. Front-End ทำการรวบรวมข้อมูลและแสดงผลกลับไปยังผู้ใช้

3.2 Algorithm ในการรวบรวมข้อมูลสำหรับงานบริการการศึกษา

ในการพัฒนาฐานข้อมูลเว็บสำหรับงานบริการการศึกษานั้น จะต้องทำการเก็บข้อมูลที่เป็นเอกสาร HTML จากแหล่งข้อมูลที่เป็นสถาบันการศึกษาต่างๆ โดยการทำงานนั้นจะต้องอาศัย web crawler ซึ่งเป็นโปรแกรมที่ช่วยในการเข้าถึงเอกสารต่างๆที่มีอยู่บนเครือข่ายอินเทอร์เน็ต ซึ่งในการทำงานเพื่อรวบรวมข้อมูลสำหรับงานบริการการศึกษานี้ สามารถแสดง algorithm ได้ดังนี้

```
1  PSEUDO CODE of Educational service web crawler
2  BEGIN
3     URLsearch = Starting URLs
4     ADD URLsearch into Link Database
5     REPEAT WHILE URLs in Link Database are not crawled
6        Download page p from URLsearch's HTML page
7        CALL   Content Extraction Module(p)
8        URLsearch = Next URL in Link Database that not crawled
9     END REPEAT
10 END
```

รูปที่ 3.2 Algorithm ในการทำงานของ Web Crawler สำหรับงานบริการการศึกษา

จากรูปที่ 3.2 เป็น Algorithm การทำงานของ Web Crawler สำหรับงานบริการการศึกษา ซึ่งสามารถอธิบายการทำงานได้ดังนี้

- กำหนด URL เริ่มต้นให้กับ crawler (บรรทัดที่ 3)
- เก็บรายชื่อ URL เริ่มต้นลงในฐานข้อมูลสำหรับเก็บ hyperlink(บรรทัดที่ 4)
- ทำการ download เอกสาร HTML ของ URL นั้นมา(บรรทัดที่ 6)

- ทำการเรียก Content Extraction Module พร้อมทั้งส่งเอกสาร HTML ที่ download มาแล้วไปด้วย(ดังแสดงรายละเอียดการทำงานในรูปที่ 3.3) เพื่อทำการดึงข้อมูลที่ต้องการออกจากเอกสาร(บรรทัดที่ 7)
- เมื่อการทำงานของ Content Extraction Module เสร็จเรียบร้อยแล้ว จะนำ URL ที่เก็บในฐานข้อมูลมาแทนใน URLsearch ซึ่งเป็น URL ที่ crawler ยังไม่ได้ download เอกสารมา(บรรทัดที่ 8)
- ทำงานซ้ำในบรรทัดที่ 6-8 จนกว่า URL ทั้งหมดที่เก็บในฐานข้อมูลนั้นถูกเรียกใช้โดย crawler

เมื่อเอกสาร HTML ถูกส่งมายัง Content Extraction Module จะมีการทำงานอยู่ด้วยกัน 3 ส่วนคือ

- Extract Link
- Extract University's Address
- Extract University's Major Offered

ซึ่งการทำงานโดยรวมของ Content Extraction Module นี้สามารถแสดง algorithm ได้ดังนี้

```
1  PSEUDO CODE of Content Extraction Module(p as HTML Page)
2  BEGIN
3      CALL Extract Link(p)
4      IF clue link word of Address contains in link word of p THEN
5              CALL Extract University's Address(p)
6      END IF
7      IF clue link word of Major contains in link word of p THEN
8              CALL Extract University's Major Offered(p)
9      END IF
10 END
```

รูปที่ 3.3 Algorithm ของ Content Extraction Module

จากรูปที่ 3.3 แสดง algorithm การทำงานของ Content Extraction Module ซึ่งสามารถอธิบายการทำงานได้ ดังนี้

- มีการเรียกใช้งาน Extract Link Module(ดังแสดงรายละเอียดการทำงานในรูปที่ 3.4) พร้อมทั้งส่งเอกสาร HTML ไปยัง Module นั้นด้วย(รูปที่ 3.3 บรรทัดที่ 3) เพื่อทำการดึงข้อความที่แสดง hyperlink ออกมาเพื่อทำการจัดเก็บ
- ทำการตรวจสอบข้อความที่ใช้เป็น link ของเอกสาร HTML ที่นำเข้ามา ว่ามีรูปแบบของข้อความ อยู่ในกลุ่มของข้อความที่ใช้เป็น link ที่อ้างถึงเอกสารที่มักจะปรากฏข้อมูลที่ตั้งของสถาบันการศึกษา(รูปที่ 3.3 บรรทัดที่ 4) ซึ่งได้แก่คำว่า
    - Contact
    - About
    - Address

  ซึ่งหากข้อความที่ใช้เป็น link ของเอกสารนั้นๆอยู่ในกลุ่มของข้อความเหล่านี้ ก็ทำให้คาดได้ว่า มีความเป็นไปได้ที่จะมีข้อมูลที่ตั้งของสถาบันการศึกษานั้นๆปรากฏอยู่
- กรณีที่ข้อความที่ใช้เป็น link อยู่ในกลุ่มของข้อความที่ใช้เป็น link ที่อ้างไปยังสถานที่ตั้งสถาบันการศึกษา จะทำการเรียกการทำงานของ Extract University's Address Module(ดังแสดงรายละเอียดการทำงานในรูปที่ 3.5) พร้อมทั้งส่งเอกสาร HTML นั้นไปด้วย(รูปที่ 3.3 บรรทัดที่ 5) เพื่อทำการดึงข้อความระบุที่ตั้งของสถาบันการศึกษาออกมาจัดเก็บ
- ทำการตรวจสอบข้อความที่ใช้เป็น link ของเอกสาร HTML ที่นำเข้ามา ว่ามีรูปแบบของข้อความ อยู่ในกลุ่มของข้อความที่ใช้เป็น link ที่อ้างถึงเอกสารที่มักจะปรากฏข้อมูลของสาขาวิชา(Major) ในสถาบันการศึกษา(รูปที่ 3.3 บรรทัดที่ 7) ซึ่งได้แก่คำว่า
    - Course
    - Program
    - Academic
    - Department
    - Faculty
    - Division
    - School

ซึ่งหากข้อความที่ใช้เพื่อเป็น link ของเอกสารนั้นๆอยู่ในกลุ่มของคำเหล่านี้ก็ทำให้คาดได้ว่า มีความเป็นไปได้ที่จะมีข้อมูลของสาขาวิชาที่มีอยู่ในสถาบันการศึกษานั้นๆปรากฏอยู่

- กรณีที่ข้อความที่ใช้เป็น link อยู่ในกลุ่มของข้อความที่ใช้เป็น link ที่อ้างไปยังสาขาวิชาของสถาบันการศึกษา จะทำการเรียกการทำงานของ Extract University's Major Offered Module(ดังแสดงรายละเอียดการทำงานในรูปที่ 3.6) พร้อมทั้งส่งเอกสาร HTML นั้นไปด้วย(รูปที่ 3.3 บรรทัดที่ 8) เพื่อทำการดึงข้อความระบุสาขาวิชาของสถาบันการศึกษาออกมาจัดเก็บ

เมื่อมีการเรียกใช้งาน Extract Link Module ซึ่งมีข้อมูลนำเข้าเป็นเอกสาร HTMLจะเป็นการทำงานเพื่อทำการดึงเอา hyperlink ที่ปรากฏในเอกสารที่นำเข้ามานั้นออกมาและจัดเก็บลงฐานข้อมูล โดย algorithm การทำงานของ Extract Link Module แสดงได้ดังนี้

```
1  PSEUDO CODE  of Extract Link(p as HTML Page)
2  BEGIN
3     REPEAT WHILE NOT End of Page p
4        IF  found TAG<A> or TAG<AREA>  THEN
5              Get Hyperlink from Attribute "HREF" in TAG
6              Verify Hyperlink with Web Exclusion List
7              IF   Hyperlink Allowed  AND Hyperlink not in Link Database
                    THEN
8                   ADD Hyperlink into Link Database
9              ELSE
10                  Discard Hyperlink
11             END IF
12       END IF
13   END REPEAT
14 END
```

รูปที่ 3.4 Algorithm ในการค้นหาและเปรียบเทียบ Hyperlink ภายในเอกสาร HTML

จากรูปที่ 3.4 แสดง algorithm ของ Extract Link Module ซึ่งจะทำหน้าที่ในการค้นหา hyperlink ที่พบในภายในเอกสาร โดยสามารถอธิบายการทำงานได้ดังนี้

- พิจารณาหา tag<A> หรือ tag<AREA> ที่มีในเอกสาร HTML ซึ่ง tag นี้จะเป็น tag ที่ใช้ในการระบุ hyperlink เพื่ออ้างไปยังเอกสารอื่นๆ(รูปที่ 3.4 บรรทัดที่ 4)
- กรณีที่พบ tag ดังกล่าว จะทำการดึง hyperlink ออกจาก tag โดยจะพิจารณาจาก Attribute "HREF" ภายใน tag<A> หรือ tag<AREA> (รูปที่ 3.4 บรรทัดที่ 5)
- นำ hyperlink นั้นไปทำการตรวจสอบเงื่อนไขในการจัดเก็บ link โดยที่เงื่อนไขต่างๆนั้นจะถูกเก็บอยู่ใน Web Exclusion List(รูปที่ 3.4 บรรทัดที่ 6) ซึ่งจะเป็นเงื่อนไขที่บอกว่า hyperlink ใดบ้างที่จะไม่จัดเก็บ ซึ่งในที่นี้ได้กำหนดเงื่อนไขใน Web Exclusion List ว่า hyperlink ใดๆที่มีโดเมนเป็น .com , .co , .biz , .org หรือการใช้ tag attribute"MAILTO" เป็นต้น จะไม่ทำการจัดเก็บลงฐานข้อมูล เช่น กรณี hyperlink "http://www.apply4admission.com" นั้นมีโดเมนเป็น .com จะไม่ทำการจัดเก็บ แต่หากเป็น hyperlink "http://www.harvard.edu" นั้นจะเห็นว่าโดเมนไม่ได้อยู่ในเงื่อนไขของ Web Exclusion List จะถือว่าผ่านตามเงื่อนไขที่กำหนดเอาไว้
- กรณีที่ hyperlink นั้นผ่านเงื่อนไขของ Web Exclusion List และ hyperlink นั้นยังไม่ถูกจัดเก็บไว้ใน Link Database จะนำ hyperlink รวมไปถึงข้อความที่ใช้แสดงเป็น link ที่ มาทำการจัดเก็บลงใน Link Database เพื่อให้ Crawler ใช้เป็น input ในการค้นหาข้อมูลต่อไป(รูปที่ 3.4 บรรทัดที่ 7 และ 8)

เมื่อมีการเรียกใช้งาน Extract University's Address Module จะเป็นการหาข้อความระบุที่ตั้งของสถาบันการศึกษา ในเอกสาร HTML ซึ่ง algorithm แสดงการทำงานของ Extract University's Address Module นั้น แสดงได้ดังนี้

```
1  PSEUDO CODE of  Extract University's Address(p as HTML Page)
2  BEGIN
3        SPLIT content in page p by list of tag that Address in
4        REPEAT WHILE NOT empty of split content's list
5                COMPUTE split content weight WITH content clue of Address
6                 IF  weight of split content >= threshold  THEN
7                    IF split content MATCH WITH Address's Regular Expression
                        THEN
8                       ADD split content into Educational service Database
9                    END IF
10                END IF
11       END REPEAT
12 END
```

รูปที่ 3.5 Algorithm ของ Extract University's Address Module

จากรูปที่ 3.5 แสดง algorithm การทำงานของ Extract University's Address Module ซึ่งทำหน้าที่ในการค้นหาและเปรียบเทียบรูปแบบของข้อความที่ระบุถึงที่ตั้งของสถาบันการศึกษาและนำมาจัดเก็บลงฐานข้อมูล โดยสามารถอธิบายการทำงานได้ ดังนี้

- เมื่อรับเอกสาร HTML เข้ามา จะทำการแยกเนื้อหาเอกสารออกเป็นส่วนๆ(รูปที่ 3.5 บรรทัดที่ 3) โดยแบ่งจาก list ที่เก็บรวบรวม tag ที่เป็นไปได้ที่ข้อความที่แสดงที่ตั้งสถาบันการศึกษานั้นจะปรากฏอยู่ ได้แก่
  - ระหว่าง tag <P> และ </P>
  - ระหว่าง tag <DIV> และ</DIV>
  - ระหว่าง tag<SPAN> และ </SPAN>
  - ระหว่าง tag<TD> และ </TD>
  - ระหว่าง tag<UL> และ </UL>
  - ระหว่าง tag<BLOCKQUOTE> และ </BLOCKQUOTE>

  ซึ่งเนื้อหาเอกสารที่แยกออกมาแล้วนั้นก็จะเป็นเนื้อหาที่ปรากฏอยู่ภายใต้ tag ดังกล่าว

- ทำการตรวจสอบแต่ละข้อความที่แยกออกมาจากเอกสารในขั้นตอนที่แล้ว มาทำการหาค่าน้ำหนักของข้อความ(รูปที่ 3.5 บรรทัดที่ 5) โดยเปรียบเทียบข้อความนั้นกับข้อความที่ใช้ระบุรูปแบบของข้อความแสดงที่ตั้ง(content clue of Address) ตัวอย่างเช่น รูปแบบของข้อความที่ใช้ระบุที่ตั้งของสถาบันการศึกษานั้น จะประกอบไปด้วย ชื่อสถาบันการศึกษา เลขที่ ชื่อถนน ชื่อเมือง ชื่อรัฐ ชื่อประเทศ รหัสไปรษณีย์ โดยจะนำแต่ละส่วนมาเปรียบเทียบกับข้อความ เช่น นำชื่อของสถาบันการศึกษาที่เก็บไว้ในฐานข้อมูลมาเทียบกับในข้อความว่ามีปรากฏหรือไม่ หรือนำเอาคำที่ใช้ระบุถึงชื่อของถนนมาเปรียบเทียบ เช่นคำว่า Street , Road , Avenue เป็นต้น ซึ่งหากทำการเปรียบเทียบในแต่ละส่วนและพบรูปแบบนั้นปรากฏอยู่ ก็จะเพิ่มค่าน้ำหนัก
- เมื่อค่าน้ำหนักรวมของข้อความนั้นมากกว่าหรือเท่ากับค่าที่ได้ตั้งไว้(threshold) ก็จะนำข้อความนั้นมาเปรียบเทียบกับ Regular Expression สำหรับที่ตั้งสถาบันการศึกษา(รูปที่ 3.5 บรรทัดที่ 7) สำหรับ Regular Expression สำหรับการหาที่อยู่ของสถาบันการศึกษานี้จะแสดงในหัวข้อถัดไป
- เมื่อพบข้อความที่มีรูปแบบตรงหรือใกล้เคียงกับรูปแบบของ Regular Expression แล้ว จะนำข้อความนั้นจัดเก็บลงฐานข้อมูลสำหรับงานบริการการศึกษา เพื่อใช้งานต่อไป(รูปที่ 3.5 บรรทัดที่ 8)

เมื่อมีการเรียกใช้งาน Extract University's Major Offered Module จะเป็นการทำงานเพื่อหาข้อความที่ระบุถึงสาขาวิชา ในเอกสาร HTML ซึ่ง algorithm แสดงการทำงานของ Extract University's Major Offered Module นั้น แสดงได้ดังนี้

```
1  PSEUDO CODE of  Extract University's Major Offered(p as HTML Page)
2  BEGIN
3        SPLIT content in page p by list of tag that Major in
4        REPEAT WHILE NOT empty of split content's list
5              VERIFY split content WITH Major's List
6              IF split content MATCH WITH Major in List >=1 THEN
7                    SPLIT phrase without tag from split content
8                    REPEAT WHILE NOT empty of phrase
9                          VERIFY phrase WITH Major's List
10                         IF  phrase contains in Major's List THEN
11                               ADD phrase into Educational service Database
12                         END IF
13                   END REPEAT
14             ELSE
15             END IF
16       END REPEAT
17 END
```

รูปที่ 3.6 Algorithm ของ Extract University's Major Offered Module

จากรูปที่ 3.6 แสดง algorithm ในการทำงานของ Extract University's Major offered Module ซึ่งทำหน้าที่ในการค้นหาและเปรียบเทียบรูปแบบของข้อความที่ระบุถึงสาขาวิชาของสถาบันการศึกษา และนำมาจัดเก็บลงฐานข้อมูล โดยสามารถอธิบายการทำงานได้ ดังนี้

- ทำการแยกเนื้อหาเอกสารออกเป็นส่วนๆ(รูปที่ 3.6 บรรทัดที่ 3) โดยแบ่งจาก list ที่เก็บรวบรวม tag ที่เป็นไปได้ที่ข้อความที่แสดงสาขาวิชาต่างๆปรากฏอยู่ ได้แก่
  - ระหว่าง tag <P> และ </P>
  - ระหว่าง tag <UL> และ</UL>
  - ระหว่าง tag<TABLE> และ </TABLE>

  ซึ่งเนื้อหาเอกสารที่แยกออกมาแล้วนั้นก็จะเป็นเนื้อหาที่ปรากฏอยู่ภายใต้ tag ดังกล่าว

- ทำการตรวจสอบแต่ละข้อความที่แยกออกมาแล้ว เปรียบเทียบกับ list ของ Major ต่างๆที่มีเก็บรวบรวมไว้ในฐานข้อมูล(รูปที่ 3.6 บรรทัดที่ 5)
- หากพบรายการสาขาวิชาในข้อความนั้น จะคาดว่ามีความเป็นไปได้ที่จะมีข้อมูลของสาขาวิชาที่สถาบันการศึกษานั้นเปิดสอนอยู่(รูปที่ 3.6 บรรทัดที่ 6) จะทำการแยกข้อความเป็นส่วนย่อยๆ โดยเก็บเฉพาะส่วนของข้อความ (phrase) ที่อยู่ระหว่าง tag เท่านั้น(รูปที่ 3.6 บรรทัดที่ 7)
- ทำการเปรียบเทียบส่วนของข้อความนั้นกับรายชื่อของสาขาวิชาย่อยที่เก็บในฐานข้อมูล(รูปที่ 3.6 บรรทัดที่ 9)
- กรณีที่ส่วนของข้อความนั้นปรากฏเป็นส่วนหนึ่งของรายชื่อของสาขาวิชา ถือว่าส่วนของข้อความนั้นเป็นข้อความที่ระบุถึงสาขาวิชาของสถาบันการศึกษา จะทำการจัดเก็บข้อความนั้นลงในฐานข้อมูลสำหรับงานบริการการศึกษาเพื่อใช้งานต่อไป(รูปที่ 3.6 บรรทัดที่ 10 และ 11)

3.3 Regular Expression สำหรับงานบริการการศึกษา

ในการหารูปแบบของข้อความต่างๆเพื่อจัดเก็บลงฐานข้อมูลสำหรับงานบริการการศึกษานั้น ได้มีการใช้ Regular Expression เพื่อใช้ในการเปรียบเทียบรูปแบบของข้อความสำหรับงานต่างๆ ดังต่อไปนี้

3.3.1 Regular Expression สำหรับการหาข้อความที่เป็นไปได้ที่จะมีที่ตั้งของสถาบันการศึกษาปรากฏอยู่

ในการหาข้อความที่ระบุสถานที่ตั้งของสถาบันการศึกษานั้น จะทำการค้นหาข้อความที่อยู่ในรูปของภาษา HTML ซึ่งโดยทั่วไปแล้ว ข้อความที่ระบุที่ตั้งของสถาบันการศึกษานั้น มักจะปรากฏอยู่ภายใต้ HTML Tag ซึ่งส่วนใหญ่จะได้แก่ <P>, <DIV>,<SPAN> เป็นต้น ซึ่งจะต้องนำเอาข้อความที่อยู่ระหว่าง Tag เหล่านั้นออกมา แล้วทำการหารูปแบบของที่ตั้งสถาบันการศึกษาต่อไป ซึ่ง Regular Expression สำหรับการหาข้อความที่เป็นไปได้ที่จะมีที่ตั้งของสถาบันการศึกษาปรากฏอยู่นั้น สามารถแสดงได้ดังรูปที่ 3.7

```
( ((<P)(\s(.)+)*>(.)*(</P>)) | ((<DIV)(\s(.)+)*>(.)*(</DIV>)) | ((<SPAN)(\s(.)+)*>(.)*(</SPAN>)) |
((<TD)(\s(.)+)*>(.)*(</TD>)) | ((<UL)(\s(.)+)*>(.)*(</UL>)) |
((<BLOCKQUOTE)(\s(.)+)*>(.)*(</BLOCKQUOTE>)) )
```

รูปที่ 3.7 Regular Expression แสดงรูปแบบข้อความที่เป็นไปได้ที่จะปรากฏที่ตั้งของสถาบันการศึกษา

จาก Regular Expression ในรูป 3.7 สามารถอธิบายได้ดังนี้

- ข้อความที่ขึ้นต้นด้วย <P อาจมีเว้นวรรคแล้วตามด้วยตัวอักษรใดๆ และเครื่องหมาย > ตามด้วยข้อความใดๆ และปิดท้ายข้อความด้วย </P> หรือ
- ข้อความที่ขึ้นต้นด้วย <DIV และปิดท้ายด้วย </DIV> หรือ
- ข้อความที่ขึ้นต้นด้วย <SPAN และปิดท้ายด้วย </SPAN> หรือ
- ข้อความที่ขึ้นต้นด้วย <TD และปิดท้ายด้วย </TD> หรือ
- ข้อความที่ขึ้นต้นด้วย <UL และปิดท้ายด้วย </UL> หรือ
- ข้อความที่ขึ้นต้นด้วย <BLOCKQUOTE และปิดท้ายด้วย </BLOCKQUOTE>

ซึ่งข้อความที่อยู่ภายใต้ข้อความขึ้นต้นและปิดท้ายประโยค จะเป็นข้อความที่มีความเป็นไปได้ที่จะมีข้อความที่ระบุที่ตั้งสถาบันการศึกษาปรากฏอยู่ และจะนำไปเปรียบเทียบกับ Regular Expression สำหรับการหาที่ตั้งของสถาบันการศึกษาต่อไป

3.3.2 Regular Expression สำหรับการหาที่ตั้งของสถาบันการศึกษา

ในการหาข้อความที่ระบุที่ตั้งของสถาบันการศึกษาที่อยู่ในเอกสาร HTML นั้น ได้มีการใช้รูปแบบของ Regular Expression เพื่อเปรียบเทียบรูปแบบของที่ตั้ง โดยกำหนดจากรูปแบบของที่ตั้งที่แสดงโดยทั่วไปของสถาบันการศึกษาหลายๆแห่ง และสร้างเป็นรูปแบบทั่วไปขึ้นมา และนำข้อความในเอกสาร HTML นั้นมาเทียบกับรูปแบบที่ได้กำหนดไว้แล้ว เพื่อที่จะดึงเอาข้อความที่มีส่วนใกล้เคียงกับรูปแบบที่กำหนด มาจัดเก็บลงในฐานข้อมูลในส่วนของที่ตั้งของสถาบันการศึกษา

สำหรับ Regular Expression ของการหาที่ตั้งของสถาบันการศึกษานี้ สามารถเขียนให้อยู่ในรูปแบบทั่วไปได้ดังนี้

```
(University Name)(Separator)(Any Phase){0,1}(Separator)(HomeNo){0,1}
(Separator)(StreetAddress){0,1}(Separator)(City){0,1}(Separator)
(State/Province)(Separator)(ZIP)(Separator)(Country){0,1}(Separator)(Tel){0,1}
```

รูปที่ 3.8 Regular Expression แสดงรูปแบบของที่ตั้งของสถาบันการศึกษา

จาก Regular Expression ในรูปที่ 3.8 จะแสดงรายละเอียดของรูปแบบข้อความแต่ละส่วนได้ดังนี้

ตาราง 3.1 คำอธิบายรายละเอียดของ Regular Expression ในการหาที่ตั้งของสถาบันการศึกษา

ข้อความ	Regular Expression	คำอธิบาย
University Name	[A-Z][a-zA-Z]+ (\s[a-zA-Z]+)+	ขึ้นต้นด้วยตัวอักษรภาษาอังกฤษตัวใหญ่แล้วตามด้วยตัวอักษรภาษาอังกฤษอย่างน้อย 1 ตัว และจะมีเว้นวรรค ตามด้วยตัวอักษรภาษาอังกฤษอย่างน้อย 1 ตัว ซึ่งข้อความด้านหลังจะต้องปรากฏอย่างน้อย 1 ครั้ง
HomeNo	[0-9]+((-|/)[0-9]+)*	ขึ้นต้นด้วยตัวเลขตั้งแต่ 1 ตัวขึ้นไป และอาจมีชุดของเครื่องหมาย – หรือ / ตามด้วยตัวเลขตั้งแต่ 1 ตัวขึ้นไป จำนวนกี่ชุดก็ได้
StreetAddress	[A-Z](.)+(\s(.)+)+	ขึ้นต้นด้วยตัวอักษรภาษาอังกฤษตัวใหญ่แล้วตามด้วยตัวอักษรใดๆอย่างน้อย 1 ตัว แล้วตามด้วย เว้นวรรค และตัวอักษรใดๆอย่างน้อย 1 ตัว ซึ่งข้อความชุดหลังนี้จะต้องปรากฏอย่างน้อย 1 ครั้ง
City	[A-Z][a-zA-Z]+ (\s[a-zA-Z]+)+	ขึ้นต้นด้วยตัวอักษรภาษาอังกฤษตัวใหญ่แล้วตามด้วยตัวอักษรภาษาอังกฤษอย่างน้อย 1 ตัว แล้วตามด้วย เว้นวรรค และตัวอักษรภาษาอังกฤษอย่างน้อย 1 ตัว ซึ่งข้อความชุดหลังนี้จะต้องปรากฏอย่างน้อย 1 ครั้ง
State/Province	([A-Z]{2,3}) | ([A-Z][a-zA-Z]+ (\s[A-Z][a-zA-Z]+)*)	ขึ้นต้นด้วยตัวอักษรภาษาอังกฤษตัวใหญ่ 2 หรือ 3 ตัว หรืออาจจะขึ้นต้นด้วยตัวอักษรภาษาอังกฤษตัวใหญ่แล้วตามด้วยตัวอักษรภาษาอังกฤษตัวใหญ่หรือตัวเล็กอย่างน้อย 1 ตัว และอาจมีเว้นวรรคตามด้วยตัวอักษรเรียงต่อกันอีกก็ได้
ZIP	([A-Z]|[0-9])+(\s ([A-Z]|[0-9])+)*	ขึ้นต้นข้อความด้วยตัวเลขหรือตัวอักษรภาษาอังกฤษตัวใหญ่อย่างน้อย 1 ตัว และอาจมีเว้นวรรคตามด้วยตัวเลขหรือข้อความเรียงต่อกันอีกก็ได้

ตาราง 3.1(ต่อ)

ข้อความ	Regular Expression	คำอธิบาย
Country	[A-Z][a-zA-Z]+ (\s[A-Z][a-zA-Z]+)*	ขึ้นต้นข้อความด้วยอักษรภาษาอังกฤษตัวใหญ่แล้วตามด้วยอักษรภาษาอังกฤษ อย่างน้อย 1 ตัว อาจจะมีเว้นวรรคแล้วตามด้วยตัวอักษร หรือไม่มีก็ได้
Tel	[0-9]+((-|.)[0-9]+)*	ขึ้นต้นด้วยตัวเลขตั้งแต่ 1 ตัวขึ้นไป และอาจมีเครื่องหมาย – หรือ / ตามด้วยตัวเลขก็ได้
Separator	(\, | \. | \\| - | \s)+	อักขระพิเศษที่ปรากฏอยู่ระหว่างข้อความ และต้องมีอย่างน้อย 1 ตัว
Any Phrase	(.)+	ขึ้นต้นและจบข้อความด้วยตัวอักษรใดๆก็ได้และต้องมีตัวอักษรอย่างน้อย 1 ตัว

อนึ่ง ในการพัฒนาระบบงานในส่วนของการเปรียบเทียบที่ตั้งสถาบันการศึกษาโดยใช้ Regular Expression นี้เห็นว่า รูปแบบของข้อความแต่ละส่วนนั้นมีความใกล้เคียงกัน เช่น ชื่อสถาบันการศึกษากับชื่อเมือง หรือเลขที่บ้านกับเบอร์โทรศัพท์ เป็นต้น ซึ่งเมื่อพัฒนาระบบขึ้นและได้สร้าง Regular Expression ตามรูปแบบทั้งหมดนี้อาจจะเกิดความยุ่งยาก จึงได้กำหนดรูปแบบของ Regular Expression เพื่อใช้ในการพัฒนาสำหรับการหาที่ตั้งสถาบันการศึกษาใหม่ ดังรูปที่ 3.9

```
[A-Z][a-zA-Z]+(\s[a-zA-Z]+)+(\,|\.|\\|-|\s)+(.)+[0-9]+((-|.)[0-9]+)*
```

รูปที่ 3.9 Regular Expression ที่ใช้ในการพัฒนาสำหรับการหาที่ตั้งสถาบันการศึกษา

จากรูปที่ 3.9 จะเป็น Regular Expression ที่มีรูปแบบของข้อความที่ขึ้นต้นด้วยชื่อของสถาบันการศึกษาและลงท้ายข้อความด้วยเบอร์โทรศัพท์ ทั้งนี้ เพื่อให้ง่ายในการพัฒนา ซึ่งสามารถที่จะปรับปรุงรูปแบบของ Regular Expression ให้มีรายละเอียดมากขึ้น โดยอ้างอิงมาจากรูปแบบข้อความที่ตั้งสถาบันการศึกษาที่กล่าวมาแล้วข้างต้นได้(รูปที่ 3.8 และตารางที่ 3.1)

3.3.3 Regular Expression สำหรับการหาข้อความที่เป็นไปได้ที่จะมีรายการของสาขาวิชาปรากฏอยู่

ในการหาข้อความที่ปรากฏรายการของสาขาวิชาที่สถาบันการศึกษานั้นเปิดสอนจะต้องทำการค้นหาข้อความภายใน HTML Tag เช่นเดียวกับการหาข้อความระบุที่ตั้ง

สถาบันการศึกษา ซึ่งใน HTML Tag ที่มักมีรายการของสาขาวิชาปรากฏอยู่ มักจะได้แก่ <P>,<UL>,<TABLE> เป็นต้น ซึ่งจะนำข้อความที่ได้เหล่านั้นมาเปรียบเทียบเพื่อหาว่าภายในข้อความนั้นๆปรากฏชื่อของสาขาวิชาหรือไม่ ในขั้นตอนต่อไป สำหรับ Regular Expression สำหรับการหาข้อความที่เป็นไปได้ที่มีรายการของสาขาวิชาปรากฏอยู่ในเอกสารนั้น สามารถแสดงได้ ดังรูปที่ 3.10

```
( ((<P)(\s(.)+)*>(.)*(</P>)) | ((<UL)(\s(.)+)*>(.)*(</UL>)) | ((<TABLE)(\s(.)+)*>(.)*(</TABLE>)) )
```

รูปที่ 3.10 Regular Expression แสดงรูปแบบข้อความที่เป็นไปได้ที่จะมีรายการของสาขาวิชา

จาก Regular Expression ในรูปที่ 3.10 สามารถอธิบายได้ดังนี้

- ข้อความที่ขึ้นต้นด้วย <P อาจมีเว้นวรรคแล้วตามด้วยตัวอักษรใดๆ และเครื่องหมาย > ตามด้วยข้อความใดๆ และปิดท้ายข้อความด้วย </P> หรือ
- ข้อความที่ขึ้นต้นด้วย <UL และปิดท้ายด้วย </UL> หรือ
- ข้อความที่ขึ้นต้นด้วย <TABLE และปิดท้ายด้วย </TABLE>

ซึ่งข้อความที่ได้นั้น เป็นข้อความที่มีความเป็นไปได้ที่อาจจะปรากฏรายการของสาขาวิชาอยู่ ซึ่งจะต้องไปทำการเปรียบเทียบกับรายชื่อของสาขาวิชาย่อยก่อน ซึ่งหากมีปรากฏบางสาขาวิชา ก็ถือว่าข้อความนั้นน่าจะมีสาขาวิชาอื่นๆปรากฏอยู่ด้วย และจะนำข้อความนั้นไปเปรียบเทียบกับ Regular Expression สำหรับระบุแต่ละสาขาวิชา ต่อไป

3.3.4 Regular Expression สำหรับการหาข้อความที่ระบุแต่ละสาขาวิชา

หลังจากที่ได้ข้อความที่คาดว่าน่าจะมีสาขาวิชาปรากฏอยู่จากหัวข้อที่ 3.3.3 ก็จะนำข้อความนั้นมาทำแบ่งข้อความนั้นออกเป็นส่วนย่อยๆอีก โดยจะพิจารณาแต่ละข้อความที่อยู่ระหว่าง HTML Tag ซึ่ง Regular Expression สำหรับการดึงเอาแต่ละข้อความออกมาเพื่อตรวจสอบนั้น แสดงได้ดังรูปที่ 3.11

```
((<[a-zA-Z]+(\s(.)+)*>(.)*(<(/){0,1}[a-zA-Z]*>)
```

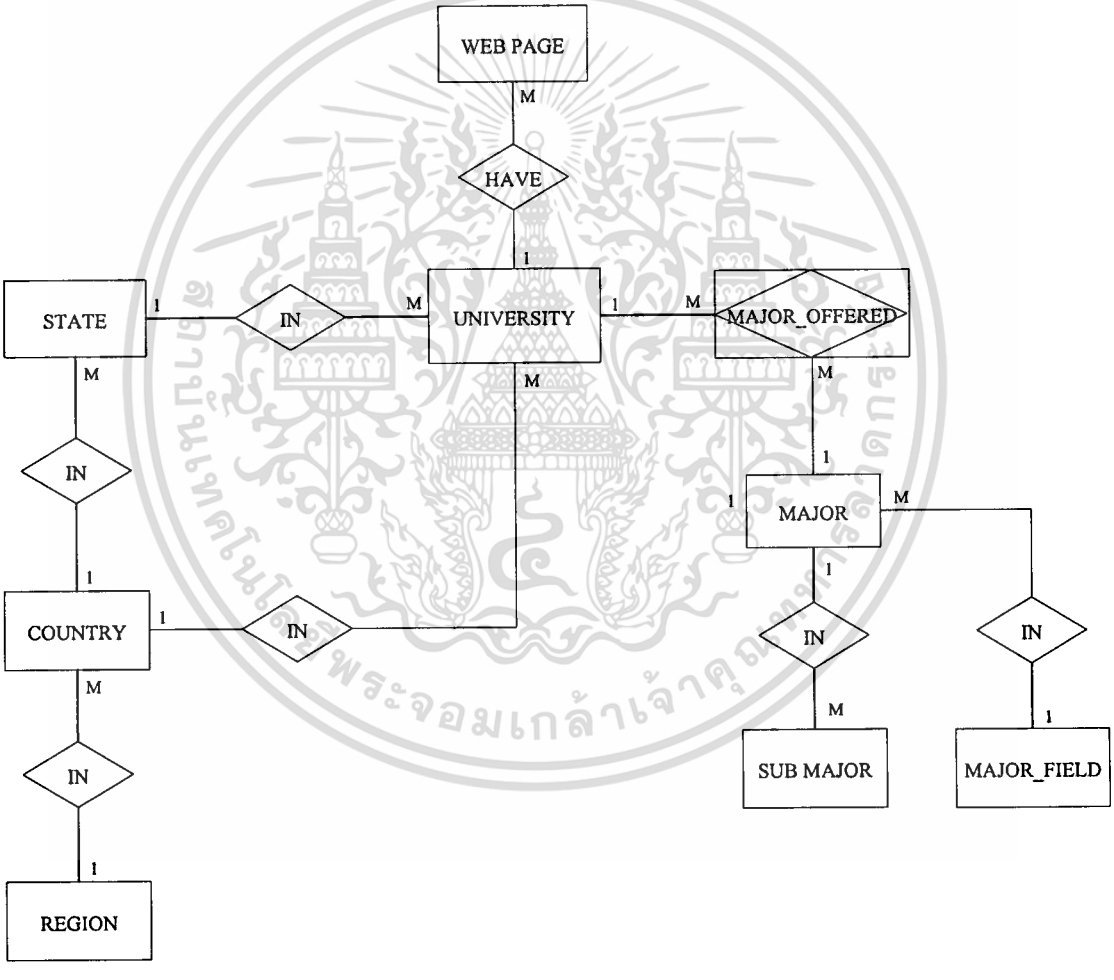
รูปที่ 3.11 Regular Expression แสดงรูปแบบข้อความที่ต้องการแบ่งเป็นส่วนย่อยๆ

จาก Regular Expression ในรูปที่ 3.11 จะแสดงถึงข้อความที่ขึ้นต้นด้วย HTML Tag ใดๆ และตามด้วยข้อความใดๆ ปิดท้ายด้วย HTML Tag ซึ่งหากพบข้อความรูปแบบนี้ จะ

นำเอาข้อความที่อยู่ระหว่าง HTML Tag นั้นมาทำการเปรียบเทียบกับสาขาวิชาที่เก็บไว้ในฐานข้อมูล

3.4 ความสัมพันธ์ของข้อมูลในระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา

ในการพัฒนาระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา มีความสัมพันธ์ระหว่างข้อมูลที่มีในระบบ ซึ่งสามารถระบุเป็นเอนทิตี และแสดงความสัมพันธ์ระหว่างเอนทิตีต่างๆ โดยใช้ Entity Relationship Diagram ได้ดังนี้



รูปที่ 3.12 Entity Relationship Diagram แสดงความสัมพันธ์ระหว่างเอนทิตีของระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา

ซึ่งเอนทิตีที่ปรากฏอยู่ใน Entity Relationship Diagram นั้นสามารระบุรายละเอียดของแต่ละเอนทิตีได้ดังนี้

ตารางที่ 3.2 เอนทิตีที่เกี่ยวข้องในระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา

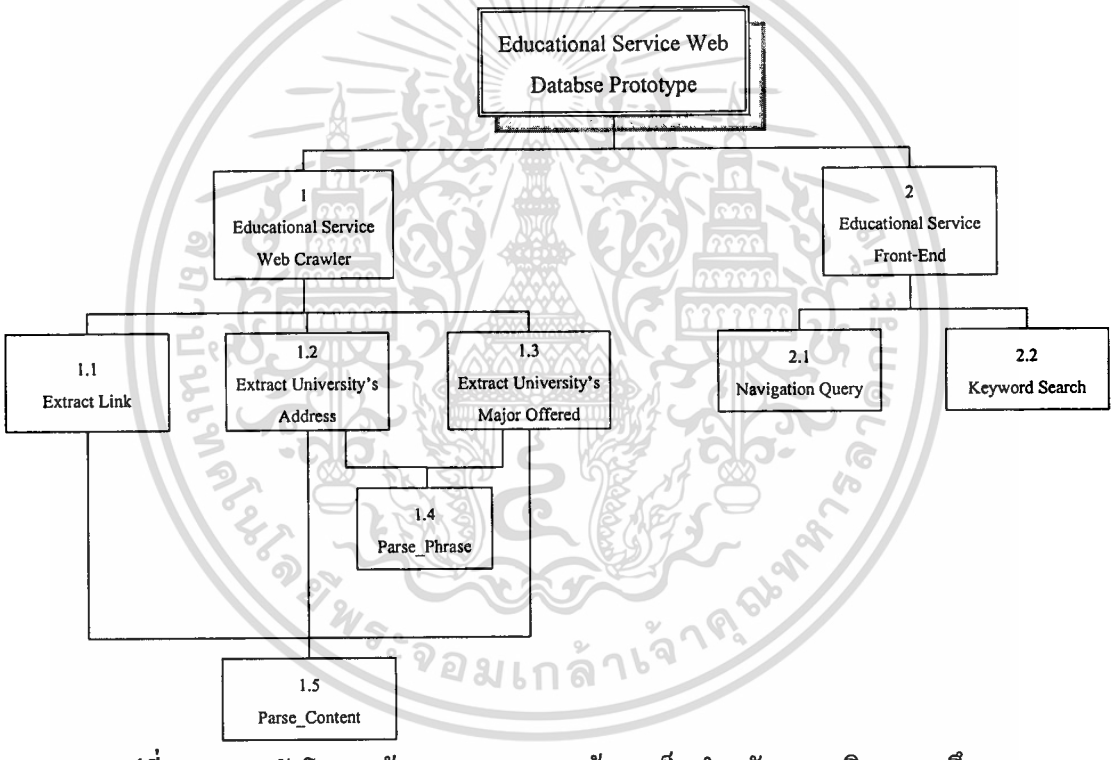
ลำดับที่	เอนทิตี	คำอธิบายรายละเอียด
1	UNIVERSITY	เก็บข้อมูลสถาบันการศึกษาต่างๆ
2	MAJOR_FIELD	เก็บกลุ่มของสาขาวิชา
3	MAJOR	เก็บรายชื่อสาขาวิชาต่างๆ
4	SUB MAJOR	เก็บรายชื่อสาขาวิชาย่อย
5	MAJOR_OFFERED	เก็บสาขาวิชาที่สถาบันการศึกษาเปิดสอน
6	STATE	เก็บข้อมูลรัฐ/เมือง
7	COUNTRY	เก็บข้อมูลประเทศ
8	REGION	เก็บข้อมูลทวีปต่างๆ
9	WEB PAGE	เก็บข้อมูล web page ที่มีอยู่ในสถาบันการศึกษา

บทที่ 4

รายละเอียดของระบบงาน

4.1 รายละเอียดของระบบงาน

ในการพัฒนาระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษาได้มีการแบ่งการทำงานออกเป็นส่วนๆ ดังแสดงในรูปที่ 4.1



รูปที่ 4.1 แผนผังโครงสร้างของระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา

จากรูปที่ 4.1 แสดงถึงแผนผังโครงสร้างการทำงานของระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา โดยได้แบ่งส่วนของการพัฒนาออกเป็นสองส่วนหลักๆ คือ

1. Educational Service Web Crawler จะพัฒนาในส่วนของการรวบรวมข้อมูลที่มีอยู่จากเว็บเพจมาเก็บลงในฐานข้อมูล

2. Educational Service Front-End พัฒนาในส่วนของการติดต่อกับผู้ใช้ผ่านทาง Web Browser

โดยในส่วนของ Educational Service Web Crawler นี้ได้มีการแบ่งการทำงานเป็นส่วนย่อยๆ คือ

- Extract Link จะทำการค้นหา Hyperlink และเปรียบเทียบเงื่อนไขการจัดเก็บ Hyperlink ที่มีในเอกสาร HTML

- Extract University's Address จะทำการค้นหาข้อความที่ระบุที่ตั้งของสถาบันการศึกษา และการดึงข้อมูลระบุที่ตั้ง ในเอกสาร HTML มาจัดเก็บ

- Extract University's Major Offered จะทำการค้นหาข้อความที่ระบุสาขาวิชาของสถาบันการศึกษา และการดึงข้อมูลสาขาวิชา ในเอกสาร HTML มาจัดเก็บ

- Parse_Phrase จะทำการตัดข้อความที่ระบุที่ตั้ง และสาขาวิชามาสร้างเป็น index เพื่อใช้สำหรับการค้นหาแบบ Keyword Search

- Parse_Content จะทำการตัดคำหรือข้อความในเอกสาร HTML ตาม Tag ที่กำหนด

สำหรับ Educational Service Front-End จะเป็นส่วนของการค้นคืนข้อมูลที่จัดเก็บอยู่ในฐานข้อมูลและนำเสนอไปยังผู้ใช้ ซึ่งในการทำการค้นคืนข้อมูลนั้นสามารถทำได้ 2 วิธีคือ

- Navigation Query จะเป็นการค้นคืนข้อมูลจาก hyperlink ที่ได้ทำการจัดหมวดหมู่ไว้แล้ว เช่น ข้อมูลที่แบ่งตามแต่ละทวีป หรือแต่ละประเทศ เป็นต้น

- Keyword Search จะค้นคืนจากคำที่ผู้ใช้ป้อนเข้ามา แล้วนำไปเทียบกับ index ของคำที่จัดเก็บไว้ในฐานข้อมูล เพื่อที่จะนำข้อมูลที่ตรงกับ keyword แสดงไปยังผู้ใช้

ซึ่งในการอธิบายรายละเอียดของการทำงานแต่ละส่วนนั้น จะอธิบายในส่วนที่มีความซับซ้อน ดังจะแสดงต่อไปนี้

4.1.1 กระบวนการในการค้นหา Hyperlink และเปรียบเทียบเงื่อนไขการจัดเก็บ Hyperlink ในเอกสาร HTML (รูปที่ 4.1 ส่วนงาน 1.1)

ในกระบวนการค้นหา Hyperlink นี้ เป็นการทำงานใน Module ซึ่งจะเรียกใช้โดย crawler ทำเพื่อค้นหารายชื่อ URL ที่เป็น link ที่ปรากฏอยู่ในเอกสาร HTML โดยมีรูปแบบในการค้นหาคือ เมื่อพบ tag<A> หรือ tag<AREA> จะทำการเก็บข้อความที่ปรากฏอยู่ใน attribute "HREF" ซึ่งเป็น Attribute ระบุ link ที่จะเชื่อมโยงไปยังเอกสารต่างๆ เพื่อที่จะนำ link เหล่านั้นไปเป็น input สำหรับให้ Web Crawler ใช้ในการท่องไปยังเอกสารต่างๆบนเครือข่ายอินเทอร์เน็ต ซึ่งจาก Algorithm ที่แสดงนั้นมีตัวแปรดังนี้

- Link จะเก็บรายชื่อ URL ที่เป็น hyperlink อยู่ภายในเครื่องหมาย tag<A> หรือ tag<AREA> และอยู่ใน Attribute “HREF”
- Link_Word เก็บข้อความที่แสดงถึง hyperlink ของ Link ซึ่งจะเป็นข้อความที่ปรากฏอยู่ระหว่าง tag<A> และ </A>
- Ch เก็บอักขระ 1 ตัวอักษรเพื่อใช้ในการเปรียบเทียบตัวอักษร
- Tag เก็บค่าภายใน Tag ที่อยู่ภายใน tag<A> หรือ tag<AREA>

โดยรูปแบบในการพิจารณาเก็บ hyperlink นั้น จะพิจารณาจากเครื่องหมาย tag เปิด(“<”) และตัวอักขระที่ปรากฏหลังเครื่องหมายเป็น “a” ซึ่ง จะถือว่าเป็น anchor tag ที่ใช้แสดง hyperlink ทั้งนี้ รวมไปถึง tag <AREA> ด้วย ซึ่งเป็น tag ที่แสดง Hyperlink โดยระบุพื้นที่บนรูปภาพ ซึ่งเมื่อพบ tag ดังกล่าว ก็จะทำการเก็บข้อความหลังจาก Attribute “HREF” ซึ่งหาก tag ที่พบเป็น tag อื่นๆ จะไม่ทำการพิจารณา

เนื่องจากการเขียน hyperlink นั้นมี 2 ประเภท นั่นคือ Absolute path จะเป็นการอ้างถึง URL แบบใช้ชื่อเต็ม เช่น “http://www.kmitl.ac.th/news/news.html” เป็นต้น อีกประเภทหนึ่งคือ Relative path ซึ่งจะเป็นการอ้างถึงเอกสาร HTML ซึ่งอยู่ภายในโดเมนเดียวกัน ตัวอย่างเช่น "./news.html" เป็นต้น ซึ่งกระบวนการค้นหา hyperlink จะต้องมีวิธีในการแปลงจาก Hyperlink ที่เป็น Relative path ให้เป็น Absolute path

เมื่อทำการแปลง hyperlink เป็น Absolute path แล้ว จะทำการเปรียบเทียบ hyperlink กับเงื่อนไขในการจัดเก็บ link ที่อยู่ใน Web Exclusion ซึ่งเป็นตัวกำหนดว่ารายชื่อของ hyperlink ใดที่จะไม่จัดเก็บ เช่น hyperlink ที่มีโดเมนเนม เป็น .com หรือ .co หรือ hyperlink ประเภท mailto เป็นต้น ซึ่งหาก hyperlink นั้นผ่านข้อกำหนดใน Web Exclusion ก็จะจัดเก็บลงใน Link Database

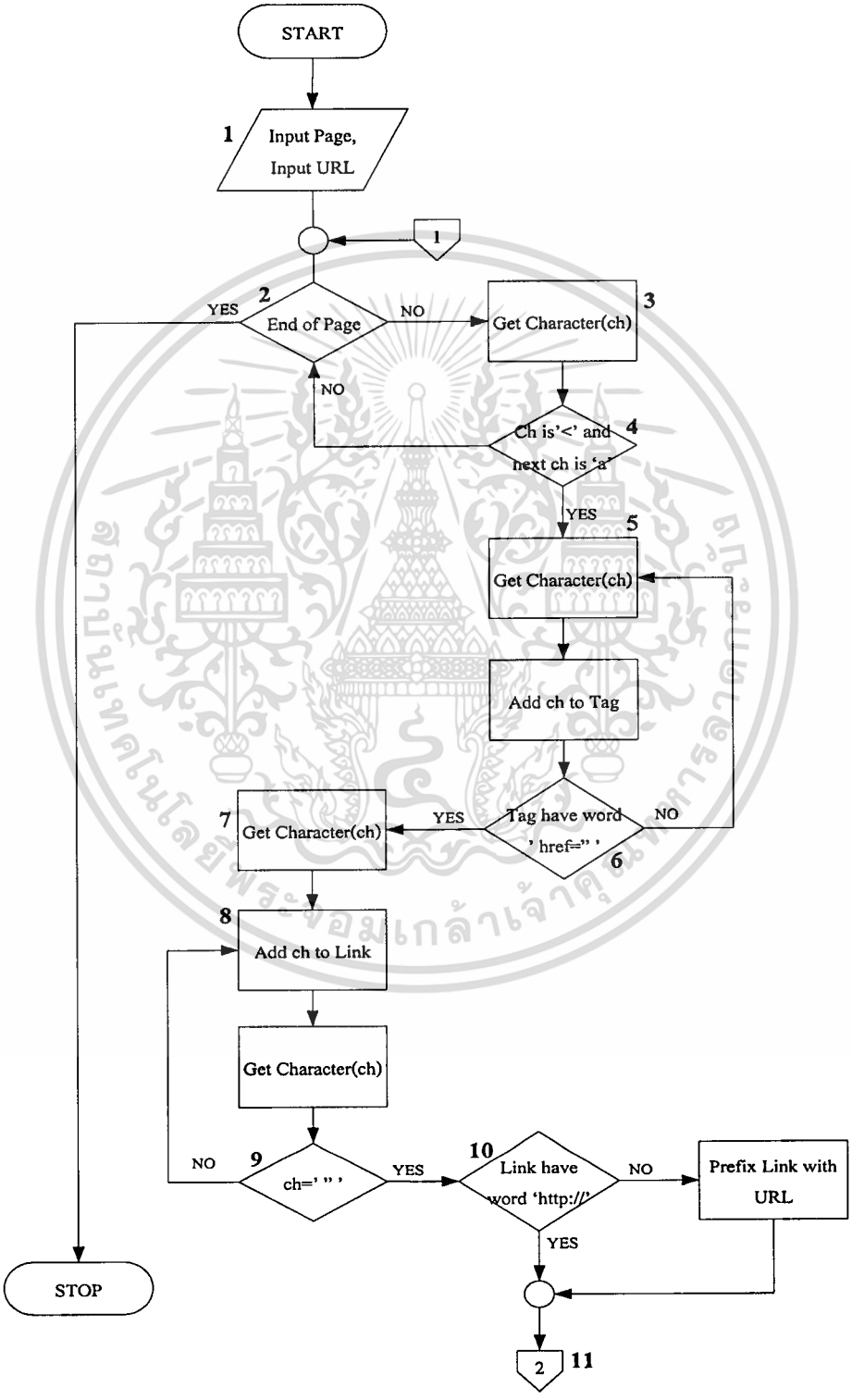
ตัวอย่างเช่น

<a href=”http://www.kmitl.ac.th”> จะทำการจัดเก็บ “http://www.kmitl.ac.th“ ใน Link

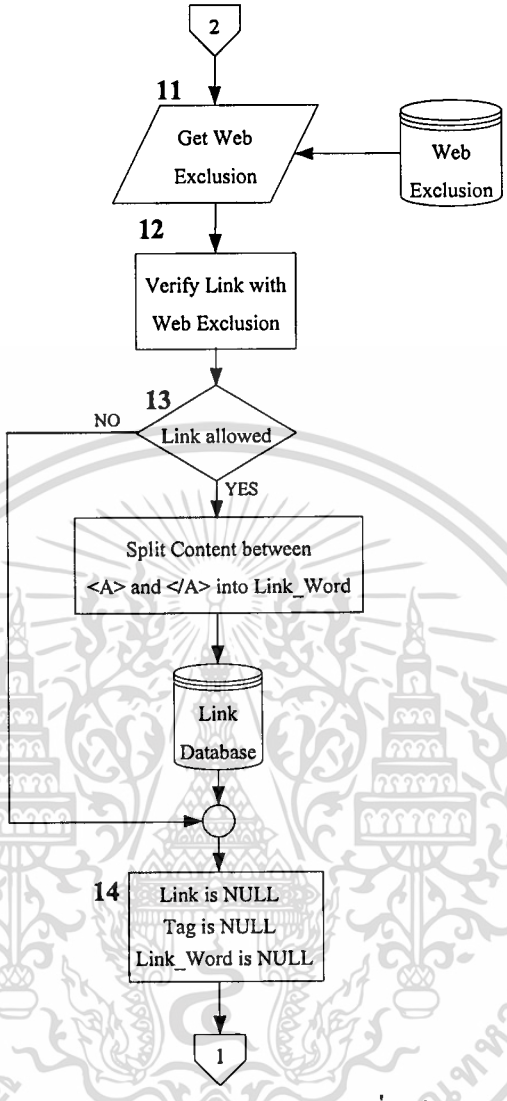
<area SHAPE="polygon" COORDS="380, 139, 394, 153, 409, 141, 403, 133" HREF="./dc.html"> กรณีนี้จะเก็บ “dc.html” และนำหน้าด้วยชื่อของ URL ของหน้าเอกสารที่พบ hyperlink นี้ เช่น ในปัจจุบันอยู่ที่ “http://www.kmitl.ac.th/news” จะเก็บ hyperlink นี้เป็น “http://www.kmitl.ac.th/news/dc.html” เป็นต้น

กรณี Link เก็บ “mailto:curry@kmitl.ac.th” จะไม่ผ่านข้อกำหนดใน Web Exclusion ดังนั้น จะไม่จัดเก็บ Link นี้

สำหรับ Algorithm ในการค้นหา Hyperlink และเปรียบเทียบเงื่อนไขการจัดเก็บ Hyperlink ในเอกสาร HTML สามารถแสดงได้ดังนี้



รูปที่ 4.2 การค้นหา Hyperlink และเปรียบเทียบเงื่อนไขการจัดเก็บ Hyperlink ในเอกสาร HTML



รูปที่ 4.2(ต่อ) การค้นหา Hyperlink และเปรียบเทียบเงื่อนไขการจัดเก็บ Hyperlink ในเอกสาร HTML

จากรูปที่ 4.2 สามารถอธิบายการทำงานได้ดังนี้

1. รับ input เป็นเอกสาร HTML รวมถึงURL ของเอกสาร HTMLที่ได้มาจากการรวบรวมของ Web Crawler
2. ตรวจสอบดูว่าถึงตำแหน่งสุดท้ายของเอกสาร(end of page) หรือยัง
   - กรณียังไม่ใช่ end of page ให้ทำขั้นตอนที่ 3 ต่อไป
   - กรณีที่เป็น end of page ให้จบการทำงาน
3. ทำการอ่านตัวอักขระที่มีในเอกสารทีละ 1 ตัว

4. เปรียบเทียบอักขระที่อ่านเข้ามาเพื่อดูว่ามีเครื่องหมายเปิด Tag ซึ่งเป็นเครื่องหมาย “<” และตัวอักขระต่อมาเป็นตัวอักษร “a” หรือไม่

- กรณีที่ไม่ใช่อักขระแสดงเครื่องหมายเปิด Tag (“<”) และตัวอักขระต่อมาเป็นตัวอักษร “a” ให้ย้อนกลับไปทำขั้นตอนที่ 2

- กรณีที่เป็นตัวอักขระแสดงเครื่องหมายเปิด Tag(“<”) และตัวอักขระต่อมาเป็นตัวอักษร “a” ถือว่าพบ tag <A> หรือ tag<AREA> แล้ว ให้ทำขั้นตอนที่ 5

5. ทำการอ่านตัวอักขระมา 1 ตัวและเก็บลงใน Tag

6. ตรวจสอบคำใน Tag ว่ามีคำว่า ‘ href=” ‘ หรือไม่

- กรณีที่พบ ให้ทำขั้นตอนที่ 7 ต่อไป

- กรณีไม่พบ ให้ย้อนกลับไปทำขั้นตอนที่ 5

7. ทำการอ่านตัวอักขระมา 1 ตัว

8. เก็บอักขระที่อ่านมาลงตัวแปร Link และอ่านอักขระเข้ามาใหม่อีก 1 ตัว

9. ตรวจสอบว่าอักขระที่อ่านเข้ามาเป็น “ ’ “ ซึ่งหมายถึงการสิ้นสุดข้อความภายใน Attribute “href” แล้วหรือไม่

- กรณีที่พบแล้ว จะทำในขั้นตอนที่ 10

- กรณียังไม่พบ จะย้อนกลับไปทำขั้นตอนที่ 8

10. ตรวจสอบ Link ว่ามีคำว่า “http://” ซึ่งหมายถึงเป็นการอ้าง hyperlink แบบ Absolute path แล้วหรือไม่

- กรณีที่มีคำว่า “http://” อยู่แล้ว จะทำในขั้นตอนที่ 11 ต่อไป

- กรณีไม่พบ แสดงว่ามีการอ้าง hyperlink แบบ Relative path ซึ่งจะต้องนำ URL ของหน้าเพจปัจจุบันนี้ใส่ด้านหน้าและตามด้วยข้อความใน Link เพื่อทำให้เป็น Absolute path แล้วทำขั้นตอนที่ 11 ต่อไป

11. อ่านค่า Web Exclusion มาจากฐานข้อมูล ซึ่งเป็นข้อมูลที่เก็บเงื่อนไขและข้อกำหนดต่างๆไว้ เช่น ไม่อนุญาตให้มีการเก็บ link ที่มี โดเมนเนมเป็น .com ,.co หรือ link ประเภท mailto เป็นต้น

12. ตรวจสอบ Link กับข้อกำหนดในการจัดเก็บ link ของ Web Exclusion

13. หาก Link นั้นได้ผ่านข้อกำหนดในการจัดเก็บ link จะทำการตัดข้อความที่อยู่ระหว่าง tag<A> และ </A> ซึ่งเป็นข้อความที่แสดงถึง hyperlink ออกมา และนำ Link พร้อมทั้งข้อความที่แสดงถึง hyperlink ของ Link นั้น จัดเก็บลงใน Link Database เพื่อใช้ในการท่องไปยังเอกสาร HTML ต่างๆ โดย Web Crawler

14. ลบค่าที่อยู่ใน Tag , Link และ Link_Word เพื่อให้สามารถรับข้อมูลใน Tag ต่อไปได้ และย้อนกลับไปยังขั้นตอนที่ 2

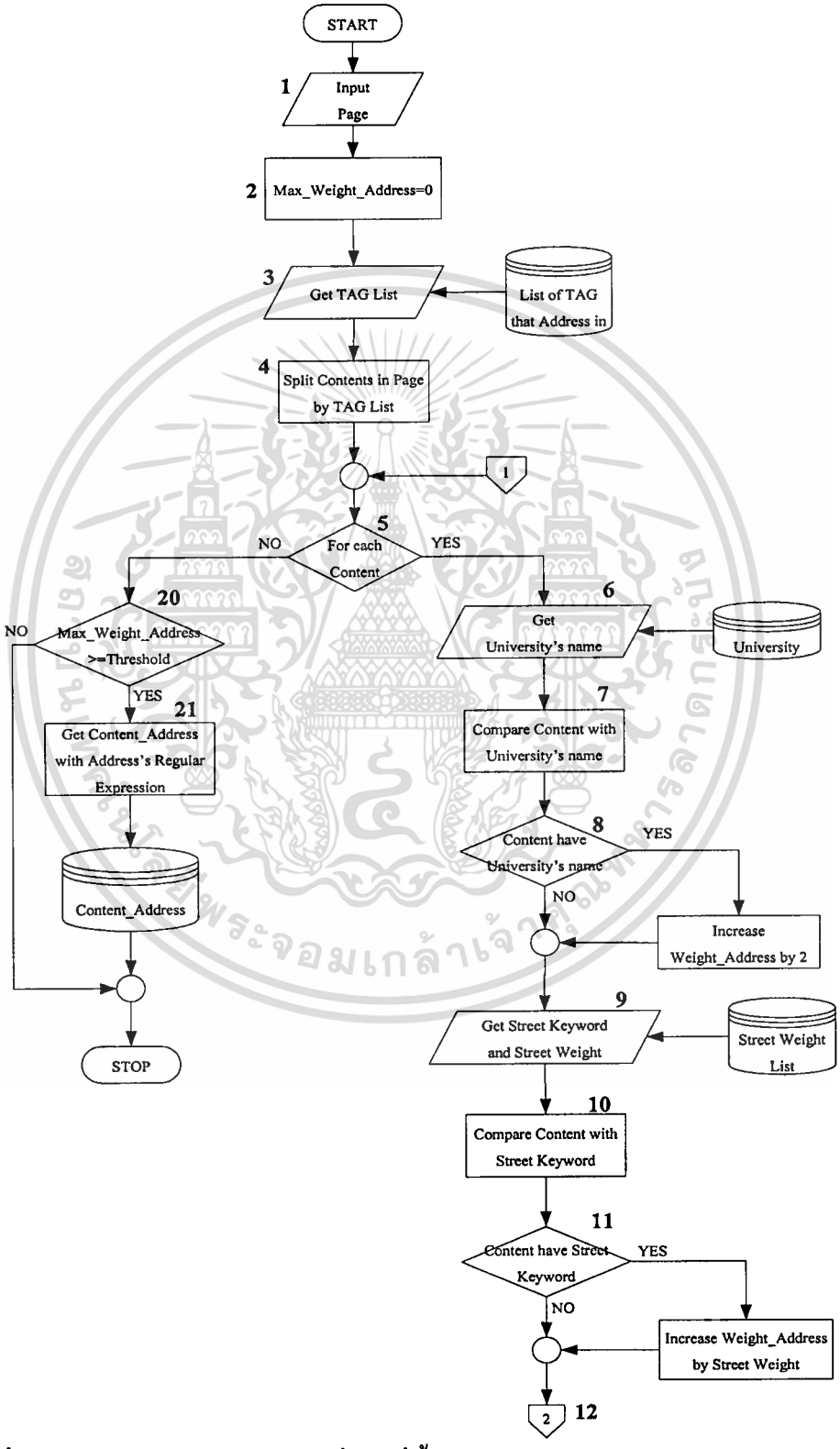
4.1.2 กระบวนการในการค้นหาข้อความที่ระบุที่ตั้งของสถาบันการศึกษา และการดึงข้อมูลระบุที่ตั้ง ในเอกสาร HTML มาจัดเก็บ (รูปที่ 4.1 ส่วนงาน 1.2)

ในกระบวนการค้นหาข้อความที่ระบุที่ตั้งของสถาบันการศึกษา และการดึงข้อมูลออกมาจากเอกสาร HTML นั้น เป็นการทำงานภายใน Module ซึ่งถูกเรียกใช้โดย Crawler เพื่อที่จะทำการจัดเก็บข้อมูลในส่วนของที่ตั้งของสถาบันการศึกษา โดยพิจารณาจากข้อความที่ปรากฏในเอกสาร HTML แล้วทำการกลั่นกรองเพื่อหาข้อมูลในส่วนที่ต้องการ เพื่อที่จะนำไปใช้ในการให้บริการในงานบริการการศึกษาต่อไป ซึ่งในการออกแบบ algorithm สำหรับกระบวนการนี้ ได้มีตัวแปรดังนี้

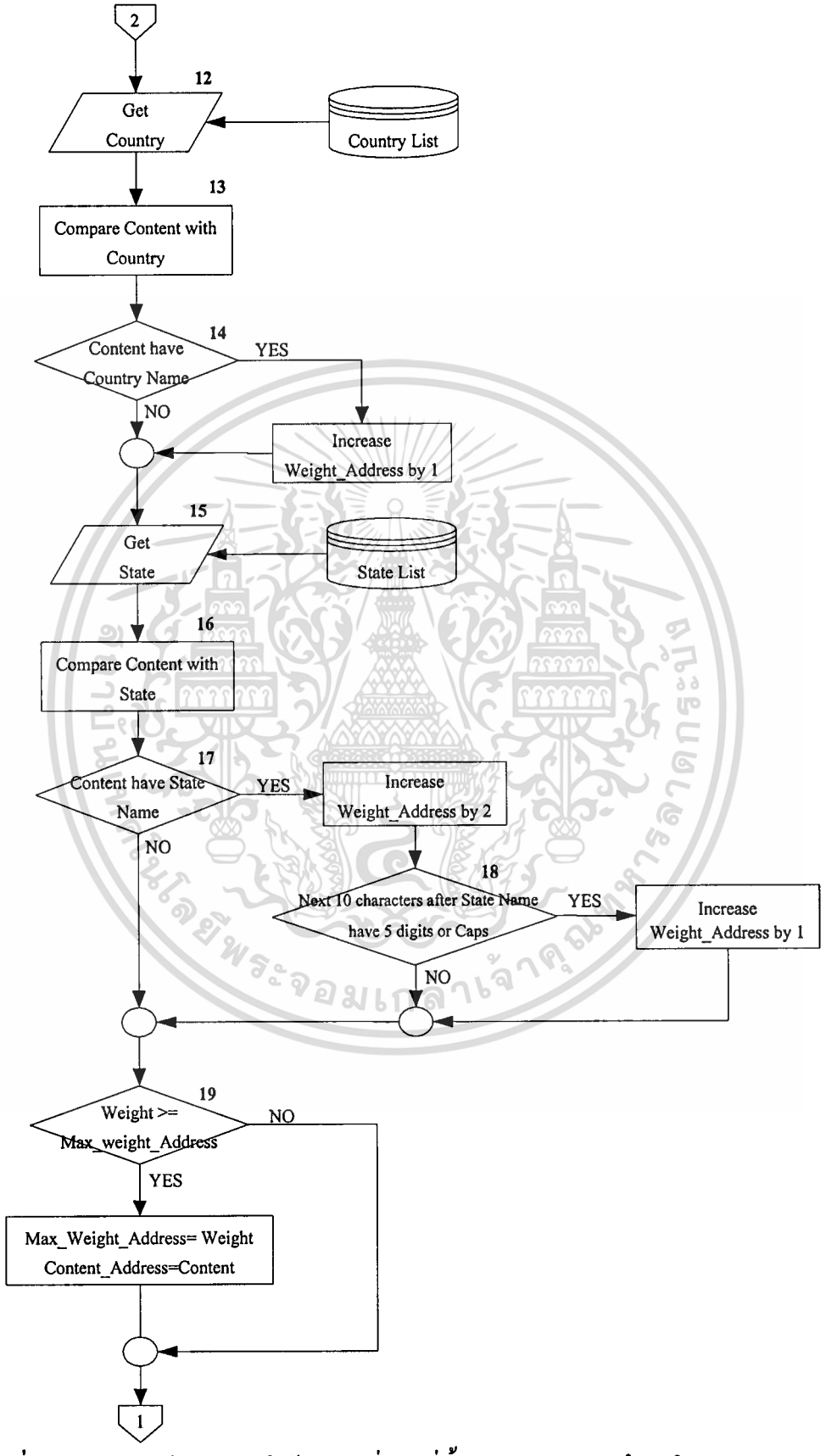
- Max_Weight_Address เก็บค่าน้ำหนักที่มากที่สุดของข้อความที่มีความเป็นไปได้ว่าจะมีข้อมูลที่ตั้งของสถาบันการศึกษาปรากฏอยู่
- Content จะเก็บข้อความที่ถูกแยกออกเป็นส่วนๆ
- TAG เป็นตัวแปรแบบ list ที่รวบรวม tag ที่มีความเป็นไปได้ที่ข้อความแสดงที่ตั้งของสถาบันการศึกษาจะปรากฏอยู่
- Weight_Address เก็บค่าน้ำหนักที่ได้ของแต่ละข้อความ
- Content_Address เก็บข้อความที่มีค่าน้ำหนักมากที่สุด
- Threshold เก็บค่าน้ำหนักที่เพียงพอที่จะระบุว่าเป็นข้อความที่แสดงที่ตั้งของสถาบันการศึกษา

โดยในการทำงานนั้นจะทำการรับเอกสาร HTML เข้าไป แล้วทำการแบ่งข้อความเป็นส่วนๆตาม tag ที่กำหนดไว้ จากนั้นจึงนำแต่ละข้อความมา หาค่าน้ำหนัก เพื่อเป็นการกรองข้อความที่ไม่มีความเป็นไปได้ที่จะเป็นข้อความที่ระบุที่ตั้งของสถาบันการศึกษา โดยมีการให้ค่าน้ำหนักของปัจจัยในการพิจารณาเป็นส่วนๆ ซึ่งได้แก่ ข้อความนั้นๆควรจะต้องประกอบด้วย ชื่อสถาบันการศึกษา คำที่ใช้เพื่อระบุชื่อถนน เช่น Street, Road เป็นต้น ชื่อของรัฐ/เมือง ชื่อประเทศ ซึ่งค่าน้ำหนักของข้อความที่ได้นั้นจะต้องมีค่ามากเพียงพอกับจุดที่ได้ตั้งไว้(Threshold) เพื่อสันนิษฐานว่าข้อความนั้นอาจจะปรากฏข้อความแสดงที่ตั้งของสถาบันการศึกษาปรากฏอยู่ จากนั้นจะทำการเปรียบเทียบกับรูปแบบของข้อความที่แสดงที่ตั้ง โดยใช้ Regular Expression ซึ่งจะประกอบไปด้วย ชื่อของสถาบันการศึกษา เลขที่ ชื่อถนน ชื่อเมือง ชื่อรัฐ/เมือง ชื่อประเทศ และรหัสไปรษณีย์ ซึ่งข้อความที่นำมาเปรียบเทียบนี้อาจจะมีรูปแบบที่ตรง หรืออาจจะแตกต่างไปก็ได้

สำหรับ algorithm ในการค้นหาข้อความที่ระบุที่ตั้งของสถาบันการศึกษา และการดึงข้อมูล ในเอกสาร HTML มาจัดเก็บนั้น สามารถแสดงได้ดังนี้



รูปที่ 4.3 การค้นหาและดึงข้อความที่ระบุที่ตั้งของสถาบันการศึกษาในเอกสาร HTML



รูปที่ 4.3(ต่อ) การค้นหาและดึงข้อความที่ระบุที่ตั้งของสถาบันการศึกษาในเอกสาร HTML

จากรูปที่ 4.3 สามารถอธิบายการทำงานได้ดังนี้

1. รับ input เข้ามาเป็นเอกสาร HTML ที่ได้มาจากการท่องไปยังเว็บเพจต่างๆของ crawler

2. กำหนดค่าเริ่มต้นของตัวแปร Max_Weight_Address ซึ่งเป็นตัวแปรเก็บค่าน้ำหนักมากที่สุดของข้อความ โดยให้มีค่าเป็น 0

3. อ่านค่า TAG ที่เก็บรวบรวมไว้ในฐานข้อมูล ซึ่งเป็น tag ที่เป็นไปได้ที่ข้อความแสดงที่ตั้งของสถาบันการศึกษาจะปรากฏอยู่ ซึ่ง tag ที่มักปรากฏข้อความ ได้แก่ tag<P> , tag<DIV>, tag<SPAN> , tag<TD> เป็นต้น

4. ทำการแยกข้อความในเอกสาร HTML ออกเป็นส่วนๆ โดยแบ่งตาม TAG ที่อ่านขึ้นมา ซึ่งข้อความที่แยกได้นั้นจะเป็นข้อความซึ่งอยู่ภายใต้ tag ดังกล่าว โดยจะอยู่ระหว่าง tagเปิด (<...>) และ tag ปิด(</...>)

5. ทำซ้ำในแต่ข้อความที่แยกออกมาได้ เพื่อหาค่าน้ำหนักของข้อความที่คาดว่าจะเป็นข้อความที่ระบุถึงที่ตั้งของสถาบันการศึกษา

6. อ่านชื่อของสถาบันการศึกษาที่เก็บไว้ในฐานข้อมูลขึ้นมา

7. นำข้อความมาเปรียบเทียบกับชื่อของสถาบันการศึกษา

8. กรณีข้อความนั้นปรากฏชื่อของสถาบันการศึกษา จะให้ค่าน้ำหนักของชื่อสถาบันการศึกษาเป็น 2 และเพิ่มค่าน้ำหนักให้กับตัวแปร Weight_Address และทำในขั้นตอนที่ 9

หากไม่พบชื่อของสถาบันการศึกษาในข้อความ จะไปทำในขั้นตอนที่ 9 ต่อไป

9. อ่าน keyword ที่ใช้เพื่อระบุชื่อถนน เช่นคำว่า Street , Road, St., Avenue, Ave. เป็นต้นออกมาจากฐานข้อมูล และอ่านค่าน้ำหนักของ keyword แต่ละตัว ทั้งนี้ เนื่องมาจากว่า keyword ที่ใช้นั้นอาจจะหมายถึงความหมายอื่นได้ด้วย ดังนั้นจึงต้องมีการกำหนดค่าน้ำหนักของ keyword เพื่อเป็นตัวกำหนดว่าคำใดที่มีความเป็นไปได้ในการเป็น keyword ที่ใช้ระบุชื่อของถนนมากกว่ากัน เช่น คำว่า Street และ St. มีความหมายระบุชื่อของถนน แต่คำว่า Street จะมีค่าน้ำหนักมากกว่า เพราะว่าคำว่า St. นี้อาจจะหมายถึงคำอื่นด้วยก็ได้ เช่นเป็นคำย่อของคำว่า Saint เป็นต้น

10. นำข้อความมาเปรียบเทียบกับ keyword ที่ใช้เพื่อระบุถึงชื่อถนน

11. กรณีข้อความนั้นปรากฏ keyword ที่ใช้ระบุชื่อถนน จะทำการเพิ่มค่าน้ำหนักให้กับตัวแปร Weight_Address โดยค่าน้ำหนักขึ้นอยู่กับค่าน้ำหนักของ keyword ที่ใช้ระบุชื่อถนนนั้นๆ และจะทำในขั้นตอนที่ 12 ต่อไป

หากไม่พบ keyword ของชื่อถนน จะไปทำในขั้นตอนที่ 12 ต่อไป

12. อ่านชื่อประเทศจากฐานข้อมูล

13. เปรียบเทียบข้อความกับชื่อประเทศ

14. กรณีข้อความนั้นปรากฏชื่อประเทศ จะทำการให้ค่าน้ำหนักของชื่อประเทศเป็น 1 และทำการเพิ่มค่าน้ำหนักให้กับกับตัวแปร Weight_Address และทำในขั้นตอนที่ 15

หากไม่พบชื่อประเทศ จะไปทำในขั้นตอนที่ 15 ต่อไป

15. อ่านชื่อของรัฐ/เมืองขึ้นมาจากฐานข้อมูล โดยนำชื่อรัฐที่อยู่ภายในประเทศในข้อ 12

16. เปรียบเทียบข้อความกับชื่อรัฐ/เมือง

17. กรณีข้อความนั้นปรากฏชื่อรัฐ/เมือง จะให้ค่าน้ำหนักของชื่อรัฐ/เมืองเป็น 1 และทำการเพิ่มค่าน้ำหนักให้กับกับตัวแปร Weight_Address และทำในขั้นตอนที่ 18

หากไม่พบชื่อรัฐ/เมือง จะไปทำในขั้นตอนที่ 18 ต่อไป

18. ตรวจสอบอักขระที่ถัดจากชื่อรัฐ/เมือง ไปอีก 10 ตัวอักษร เพื่อดูว่ามีตัวเลขหรืออักษรภาษาอังกฤษตัวพิมพ์ใหญ่และตัวเลขเรียงต่อกัน 5 ตัวอักษรหรือไม่ ทั้งนี้ เพื่อใช้ในพิจารณารหัสไปรษณีย์ ซึ่งโดยทั่วไปมักจะมีรูปแบบเป็นตัวเลขหรืออักษรภาษาอังกฤษตัวพิมพ์ใหญ่ผสมตัวเลขเรียงต่อกัน และมักปรากฏต่อจากชื่อรัฐ/เมือง

- กรณีที่พบชุดของตัวเลขหรือชุดของตัวเลขผสมกับตัวอักษรภาษาอังกฤษตัวพิมพ์ใหญ่ ให้ตั้งสมมติฐานว่าเป็นรหัสไปรษณีย์ จะให้ค่าน้ำหนักเป็น 1 และทำการเพิ่มค่าน้ำหนักให้กับตัวแปร Weight_Address จากนั้นจะทำในขั้นตอนที่ 19 ต่อไป

- กรณีไม่พบ จะทำในขั้นตอนที่ 19 ต่อไป

19. เปรียบเทียบค่าน้ำหนักของข้อความในตัวแปร Weight_Address กับค่าน้ำหนักที่มากที่สุดซึ่งเก็บในตัวแปร Max_Weight_Address

- กรณีที่ค่าน้ำหนักของข้อความมากกว่าค่าน้ำหนักมากที่สุดใน Max_Weight_Address จะทำการปรับค่า Max_Weight_Address ให้เป็นค่าของน้ำหนักในตัวแปร Weight_Address นั้น และ เก็บข้อความนั้นลงในตัวแปร Content_Address และกลับไปทำซ้ำในขั้นตอนที่ 5 โดยใช้ข้อความต่อไป

- กรณีที่ค่าน้ำหนักของ Weight_Address มีค่าน้อยกว่า จะวนกลับไปทำซ้ำในขั้นตอนที่ 5 จนกว่าจะหมดทุกข้อความ

20. กรณีที่เปรียบเทียบครบทุกข้อความแล้ว จะนำค่าน้ำหนักที่มากที่สุดที่เก็บในตัวแปร Max_Weight_Address มาเปรียบเทียบกับค่า Threshold ซึ่งเป็นจุดที่ระบุถึงค่าน้ำหนักที่เพียงพอที่จะสรุปได้ว่าเป็นข้อความที่ปรากฏที่ตั้งของสถาบันการศึกษา ซึ่งในการพัฒนานี้ได้กำหนดค่า Threshold ไว้ที่ 5 จากค่าน้ำหนักสูงสุดที่เป็นไปได้ ซึ่งมีค่าเท่ากับ 8

- กรณีที่ค่าน้ำหนักมากที่สุดนั้นมากกว่าหรือเท่ากับค่า threshold จะทำในขั้นตอนที่ 21

- กรณีค่าน้ำหนักมากที่สุดน้อยกว่าค่า threshold จะถือว่ายังไม่เป็นข้อความที่ปรากฏที่ตั้งของสถาบันการศึกษา จะไม่จัดเก็บ และจบการทำงาน

21. นำข้อความที่มีค่าน้ำหนักมากที่สุด ซึ่งเก็บในตัวแปร Content_Address มาเปรียบเทียบกับรูปแบบของ Regular Expression ของที่ตั้งสถาบันการศึกษา เพื่อที่จะทำการดึงข้อมูลที่ตั้งออกมาเพื่อจัดเก็บ

ในการใช้ Regular Expression นั้น เนื่องจากว่า ข้อความได้มานั้น อาจจะปรากฏข้อความอื่นๆนอกเหนือจากข้อความที่แสดงที่ตั้งด้วย ดังนั้นจึงต้องทำการใช้ Regular Expression เพื่อทำการดึงเอาเฉพาะข้อมูลในส่วนที่เป็นที่ตั้งออกมาจัดเก็บ ซึ่งเมื่อดึงข้อมูลนั้นออกมาได้แล้ว จะทำการจัดเก็บลงในฐานข้อมูล และจบการทำงาน

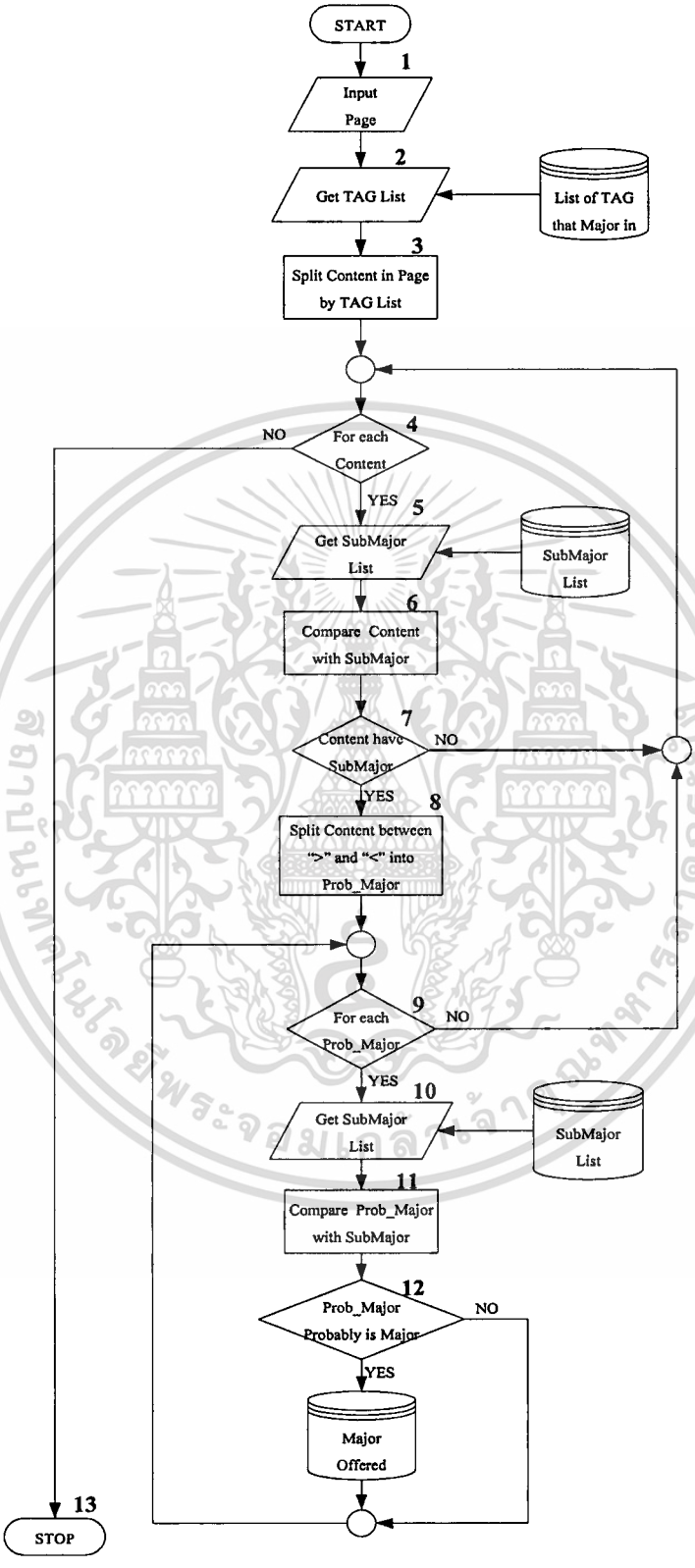
ทั้งนี้ ในการใช้ค่าน้ำหนักเพื่อพิจาณาข้อความที่อาจจะปรากฏข้อความระบุที่ตั้งสถาบันการศึกษา จะเห็นว่าได้กำหนดค่าน้ำหนักในส่วนของชื่อสถาบันการศึกษา(University Name) และชื่อรัฐ/เมือง(State Name) ให้มีค่าน้ำหนักเป็น 2 เนื่องจากว่าข้อความที่ระบุที่ตั้งสถาบันการศึกษา มักจะปรากฏชื่อสถาบันการศึกษา และชื่อรัฐ/เมือง ด้วยเสมอ ซึ่งเมื่อพบข้อความดังกล่าวแล้ว มีความเป็นไปได้ที่ข้อความนั้นอาจจะปรากฏที่ตั้งสถาบันการศึกษาก็ได้ แต่ก็ต้องจำเป็นต้องอาศัยค่าน้ำหนักจากการเปรียบเทียบข้อความในส่วนอื่นเพิ่มเติมด้วย

4.1.3 กระบวนการในการค้นหาข้อความที่ระบุสาขาวิชาของสถาบันการศึกษา และการดึงข้อมูลสาขาวิชา ในเอกสาร HTML มาจัดเก็บ (รูปที่ 4.1 ส่วนงาน 1.3)

กระบวนการค้นหาข้อความที่ระบุสาขาวิชาของสถาบันการศึกษา และการดึงข้อมูลสาขาวิชาในเอกสาร HTML นี้ เป็นการทำงานใน Module ซึ่งจะถูกเรียกใช้โดย crawler ทำเพื่อค้นหาข้อมูลเกี่ยวกับสาขาวิชาที่สถาบันการศึกษาเปิดสอนอยู่ โดยพิจารณาจากเอกสาร HTML ที่ crawler ท่องไป แล้วทำการดึงเอาข้อความที่เป็นไปได้ที่จะเป็นสาขาวิชา มาจัดเก็บลงในฐานข้อมูล ซึ่งใน algorithm ของกระบวนการนี้ ประกอบด้วยตัวแปรดังต่อไปนี้

- Content จะเก็บข้อความที่ถูกแยกออกเป็นส่วนๆ
- TAG เป็นตัวแปรแบบ list ที่รวบรวม tag ที่มีความเป็นไปได้ที่ข้อความที่มีรายการของสาขาวิชาปรากฏอยู่
- Prob_Major เป็นตัวแปรเก็บข้อความที่ถูกแยกออกจาก Content เป็นส่วนๆ เพื่อนำมาเปรียบเทียบกับสาขาวิชาย่อย

สำหรับ algorithm ในการค้นหาข้อความที่ระบุสาขาวิชาของสถาบันการศึกษา และการดึงข้อมูลสาขาวิชาในเอกสาร HTMLนั้น สามารถแสดงได้ดังนี้



รูปที่ 4.4 การค้นหาข้อความที่ระบุสาขาวิชาของสถาบันการศึกษา และการดึงข้อมูลสาขาวิชาในเอกสาร HTML

จากรูป 4.4 สามารถอธิบายรายละเอียดการทำงานได้ ดังนี้

1. รับ input เข้ามาเป็นเอกสาร HTML ซึ่งได้จากการท่องไปยังเว็บเพจต่างๆของ crawler

2. อ่านค่า TAG ที่รวบรวมไว้ ซึ่งเป็น tag ที่เป็นไปได้ที่ข้อความแสดงสาขาวิชาของสถาบันการศึกษาจะปรากฏอยู่ ซึ่งในกรณีของการหาข้อความที่แสดงที่อยู่นั้น tag ที่มักปรากฏข้อความ มักได้แก่ tag<P> , tag<TABLE>, tag<UL> เป็นต้น

3. ทำการแยกข้อความในเอกสาร HTML ออกเป็นส่วนๆ โดยแบ่งตาม TAG ซึ่งข้อความที่แยกได้นั้นจะเป็นข้อความซึ่งอยู่ภายใต้ tag ดังกล่าวโดยจะอยู่ระหว่าง tagเปิด(<...>) และ tag ปิด (</...>)

การทำงานในขั้นตอนที่ 4-7 จะเป็นการหาข้อความที่มีสาขาวิชาปรากฏอยู่ โดยมีขั้นตอนดังต่อไปนี้

4. ทำซ้ำในแต่ละข้อความที่แยกออกมาได้จากขั้นตอนที่ 3 เพื่อตรวจสอบว่าแต่ละข้อความนี้บรรจุสาขาวิชาอยู่หรือไม่

5. อ่านรายการของสาขาวิชาย่อย(SubMajor) จากฐานข้อมูล

6. ใช้ฟังก์ชั่นเปรียบเทียบข้อความ เพื่อทดสอบว่ามีสาขาวิชาย่อยบรรจุอยู่ในข้อความหรือไม่

7. กรณีที่พบว่ามีสาขาวิชาบรรจุอยู่ในข้อความที่แยกออกมานั้น จะดำเนินการในขั้นตอนที่ 8 ต่อไป หากไม่พบ จะกลับไปทำซ้ำในขั้นตอนที่ 4 กับข้อความชุดใหม่

ตัวอย่างเช่น กรณีที่มีข้อความที่ถูกแยกจากขั้นตอนที่ 3 ได้เป็นดังนี้

```
<P>
<br><a href="http://www.alliedhealth.unc.edu/">Allied Health</A> <br>
<A href="http://www.aims.unc.edu/">Anesthesiology </A><br>
<A href="http://www.unc.edu/depts/anthro/">Anthropology</A><br>
<A href="http://www.unc.edu/depts/art/">Art</A><br>
<A href="http://www.unc.edu/depts/asia/">Asian Studies</A><br>
<A href="http://www.bio.unc.edu/">Biology</A><br>
<A href="http://www.bme.unc.edu">Biomedical Engineering</A><br>
<A href="http://www.sph.unc.edu/bios/">Biostatistics</A><br>
</P>
```

รูปที่ 4.5 ข้อความในเอกสาร HTML ที่ถูกแยกออกเพื่อหาสาขาวิชา

จากข้อความที่ได้ในรูปที่ 4.5 จะนำข้อความชุดนี้ไปเปรียบเทียบกับสาขาวิชาย่อยที่อ่านมาจากฐานข้อมูลโดยใช้ฟังก์ชั่นเปรียบเทียบข้อความ ซึ่งจากข้อความตัวอย่าง เมื่อพบข้อความที่แสดงถึงสาขาวิชา เช่น Art หรือ Biology ก็จะนำข้อความชุดนี้ไปดำเนินการในขั้นตอนที่ 8 ต่อไป

8. ทำการแยกข้อความที่อยู่ระหว่างเครื่องหมาย ">" และ "<" ออกมาเก็บลงในตัวแปร Prob_Major ตัวอย่างเช่น จากข้อความในรูปที่ 4.5 จะทำการตัดข้อความ Allied Health , Anesthesiology เป็นต้น มาจัดเก็บลงในตัวแปร Prob_Major

การทำงานในขั้นตอนที่ 9-12 จะทำการเปรียบเทียบแต่ละข้อความที่เก็บอยู่ในตัวแปร Prob_Major มาเปรียบเทียบกับรายชื่อของสาขาวิชาย่อยที่เก็บอยู่ในฐานข้อมูล ดังแสดงขั้นตอนการทำงานได้ดังนี้

9. ทำซ้ำในแต่ละข้อความที่ดึงออกมาได้จาก Content เพื่อทำการเปรียบเทียบความเป็นไปได้ของข้อความที่แสดงถึงสาขาวิชา

10. อ่านรายการสาขาวิชาย่อย(SubMajor) พร้อมทั้งรายการสาขาวิชาที่สาขาวิชาย่อยนั้นๆ สังกัดอยู่ จากฐานข้อมูล

11. เปรียบเทียบข้อความใน Prob_Major กับสาขาวิชาย่อย และตัดสินใจว่าข้อความนั้นเป็นสาขาวิชาย่อยของสาขาวิชาใด

12. กรณีที่เปรียบเทียบแล้วเห็นว่า ข้อความใน Prob_Major นั้น ตรงกับสาขาวิชาย่อยที่จัดเก็บไว้ หรือเป็นข้อความส่วนหนึ่งของสาขาวิชาย่อยที่จัดเก็บไว้ จะถือว่าข้อความนั้นเป็นข้อความแสดงสาขาวิชาที่สถาบันการศึกษาเปิดสอน และทำการจัดเก็บลงฐานข้อมูล จากนั้นจะวนการทำงานในขั้นตอนที่ 9 ต่อไป

ตัวอย่างเช่น ข้อความที่เก็บในตัวแปร Prob_Major มีค่าเป็น Biology นำมาเปรียบเทียบกับรายชื่อสาขาวิชาย่อยในฐานข้อมูล ซึ่งมีข้อความ Biology ปรากฏในฐานข้อมูลอยู่ด้วย ซึ่งถือว่าข้อความนี้ระบุถึงสาขาวิชา ดังนั้น จะทำการเก็บข้อความดังกล่าวลงในฐานข้อมูล

13. เมื่อทำการเปรียบเทียบหมดทุกข้อความแล้ว จะจบการทำงาน

4.1.4 กระบวนการในการตัดคำหรือข้อความ ในเอกสาร HTML (รูปที่ 4.1 ส่วนงาน 1.5)

ในกระบวนการตัดคำหรือข้อความในเอกสาร HTML นี้ ทำเพื่อทำการเก็บข้อความที่ปรากฏอยู่ภายในเอกสาร HTML โดยเก็บข้อความซึ่งอยู่ระหว่างสัญลักษณ์ tag ที่กำหนดไว้เพื่อที่จะนำข้อความเหล่านั้นมาใช้ใน Module ของการดึงข้อมูลเพื่อจัดเก็บลงฐานข้อมูล กระบวนการนี้เป็นส่วนหนึ่งของ Module สำหรับใช้ในส่วนการตัดข้อความที่ใช้แสดงถึง hyperlink

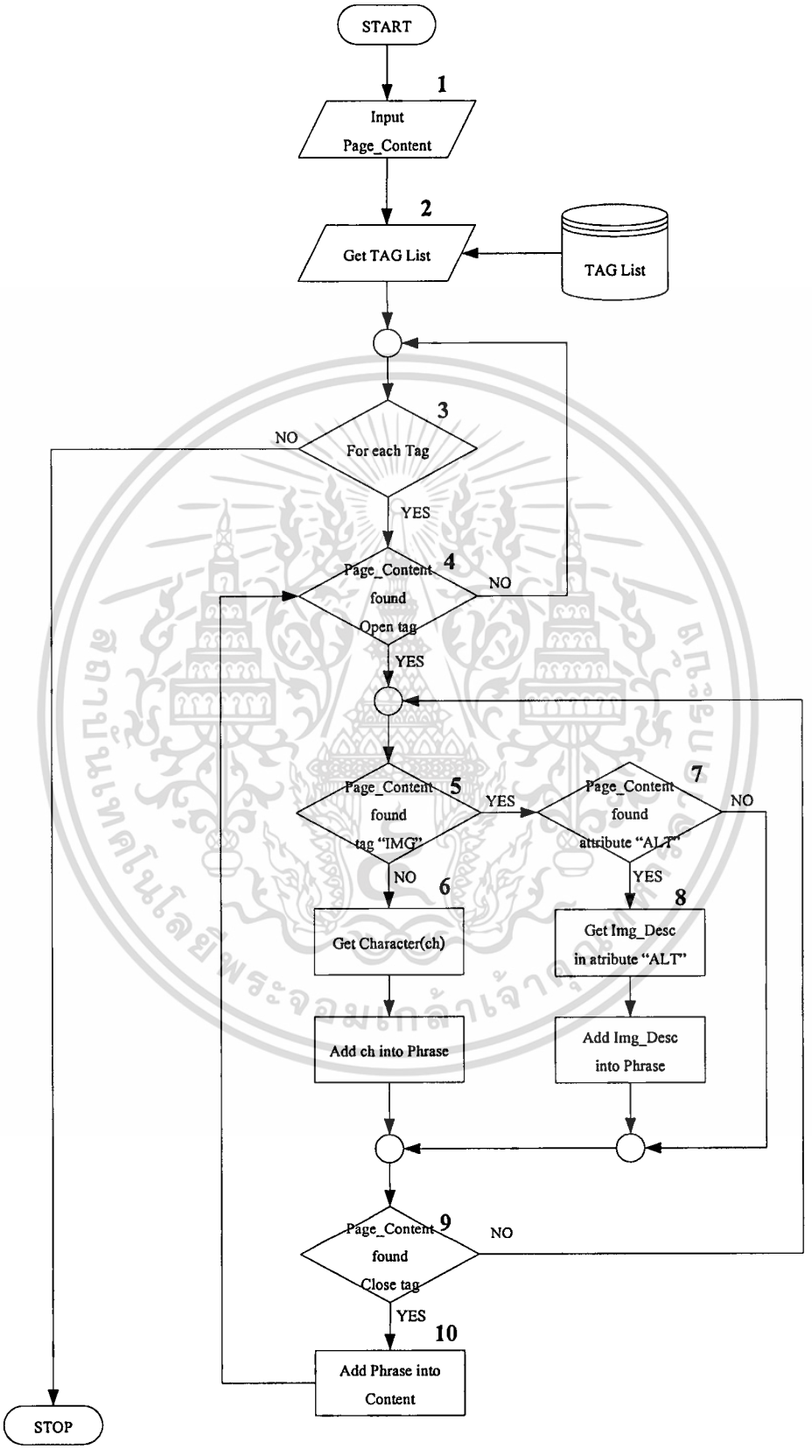
ส่วนของการตัดข้อความเพื่อหาที่อยู่ของสถาบันการศึกษา และการตัดข้อความเพื่อหารายการของสาขาวิชา ซึ่งจาก Algorithm ที่แสดงนั้นมีตัวแปรดังนี้

- Page_Content เก็บเอกสาร HTML ที่รับเข้ามา
- TAG เป็น list ที่รวบรวม tag ที่กำหนดไว้เพื่อเป็นตัวแบ่งข้อความในเรื่องต่างๆ
- Img_Desc เก็บข้อความที่เป็น attribute "ALT" ของ tag "IMG"
- Phrase จะเก็บคำหรือข้อความที่อยู่ระหว่างเครื่องหมาย tag เปิด และ tag ปิด
- Ch เก็บอักขระ 1 ตัวอักษร
- Content เป็น list ที่เก็บข้อความที่อยู่ระหว่าง tag ที่ได้จากกระบวนการตัดข้อความ โดยจะนำข้อความนี้ไปใช้งานต่อในModule ต่างๆ

โดยรูปแบบในการพิจารณาเก็บข้อความนั้น จะพิจารณาจาก tag เปิด(<...>) และ tag ปิด(</...>) ซึ่ง tag เหล่านั้นจะถูกกำหนดในแต่ละส่วน เช่นในส่วนของการหาที่อยู่ของสถาบันการศึกษา จะพิจารณา tag<P> , tag<DIV>, tag<SPAN> แต่หากเป็นส่วนการหาสาขาวิชาจะพิจารณา tag<P>, tag<TABLE>, tag<UL> เป็นต้น ซึ่งเมื่อพบ tag เปิดจะทำการเก็บข้อความไปจนกว่าจะพบ tag ปิด ซึ่งจะถือว่าข้อมูลที่อยู่ระหว่าง tag นั้นสิ้นสุดลง

แต่กรณีที่พบ tag<IMG> จะทำการเก็บข้อความที่ปรากฏใน Attribute "ALT" ซึ่งเป็น Attribute อธิบายรูปภาพ ซึ่งบางเว็บเพจนั้นอาจมีการใช้รูปภาพอธิบายความหมายแทนการใช้ข้อความ จึงต้องทำการเก็บข้อความภายใน tag <IMG> ซึ่งเมื่อพบคำว่า "ALT" จะเริ่มทำการเก็บข้อความภายใน Attribute นี้ ตัวอย่างเช่น <IMG src="/pic.jpg" alt="Show Picture"> จะเห็นว่ามี Attribute "ALT" อยู่ ดังนั้นจะทำการเก็บข้อความ "Show Picture" ลงในตัวแปร Phrase เป็นต้น

สำหรับ Algorithm ในการตัดคำหรือข้อความเอกสาร HTML สามารถแสดงได้ดังนี้



รูปที่ 4.6 การตัดคำหรือข้อความ ในเอกสาร HTML

จากรูปที่ 4.6 สามารถอธิบายการทำงานได้ดังนี้

1. รับ input เป็นเอกสาร HTML

2. อ่านค่า TAG ที่รวบรวมไว้ในฐานข้อมูล ทั้งนี้ ขึ้นอยู่กับการใช้งานในแต่ละส่วน ซึ่งได้แก่ tag ที่ใช้สำหรับการตัดข้อความเพื่อหาข้อมูลที่อยู่ของสถาบันการศึกษา และ tag ที่ใช้สำหรับการตัดข้อความเพื่อหาข้อมูลสาขาวิชา

3. ทำซ้ำในแต่ละ tag ที่อ่านขึ้นมา โดยตัดข้อความในเอกสาร HTML ตาม tag นั้นๆ

4. ตรวจสอบว่าพบข้อความแสดง tag เปิด ตาม tag ที่กำหนดไว้หรือไม่ เช่นตรวจสอบว่าพบ tag ที่กำหนดไว้คือ tag<P> เป็นต้น

- กรณีที่พบ tag ที่กำหนดในข้อความ จะทำในขั้นตอนที่ 5
- กรณีไม่พบ tag ที่กำหนดในข้อความ จะทำซ้ำในขั้นตอนที่ 3 โดยใช้ tag ต่อไป

5. ตรวจสอบว่าข้อความต่อมานั้นมี tag<IMG> หรือไม่

- กรณีที่พบ tag<IMG> อยู่ในข้อความด้วย จะไปทำขั้นตอนที่ 7
- กรณีไม่พบ จะทำขั้นตอนที่ 6

6. อ่านค่าตัวอักขระมา 1 ตัวอักษร และเก็บค่าลงในตัวแปร Phrase และทำขั้นตอนที่ 9

7. กรณีเป็น tag<IMG> จะตรวจสอบว่าภายใน tag<IMG> นั้นมี Attribute "ALT" หรือไม่

- กรณีที่พบ Attribute "ALT" ให้ทำขั้นตอนที่ 8
- กรณีไม่พบ ให้ทำขั้นตอนที่ 9 ต่อไป

8. ทำการอ่านข้อความที่อยู่ภายใน Attribute "ALT" และเก็บลงตัวแปร Phrase

9. ตรวจสอบว่าพบข้อความแสดง tagปิด ตาม tag ที่กำหนดไว้หรือไม่

- กรณีพบ tag ปิดตามที่กำหนดไว้ ให้ทำขั้นตอนที่ 10
- กรณีไม่พบ ให้ทำซ้ำในขั้นตอนที่ 5

10. เก็บค่าในตัวแปร Phrase ลงในตัวแปร Content เพื่อนำไปใช้งาน แล้วกลับไปทำซ้ำในขั้นตอนที่ 4 จนกว่าจะไม่พบข้อความแสดง tagเปิดตามที่กำหนดไว้

4.2 รายละเอียดฐานข้อมูลสำหรับใช้ในการพัฒนาระบบ

จากเอนทิตีที่เกี่ยวข้องของระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา สามารถแสดงรายละเอียดของรีเลชั่นต่างๆ ออกมาเป็นตารางเพื่อใช้ในการพัฒนาระบบงานต่อไปได้ ทั้งนี้ การพัฒนาระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา ได้เพิ่มตารางเพื่อช่วยในการทำงานให้สะดวกขึ้น ซึ่งประกอบด้วยตารางดังนี้

ตารางที่ 4.1 ตารางที่ใช้ในการพัฒนาระบบงาน

ลำดับที่	ตาราง	คำอธิบายรายละเอียด
1	UNIVERSITY	เก็บข้อมูลสถาบันการศึกษาต่างๆ เช่น ชื่อ , URL
2	MAJOR_FIELD	เก็บกลุ่มของสาขาวิชา เช่น เช่นกลุ่ม Engineer , Business
3	MAJOR	เก็บรายชื่อสาขาวิชาต่างๆ
4	SUB MAJOR	เก็บรายชื่อสาขาวิชาย่อย
5	UNI_MAJOR_OFFER	เก็บสาขาวิชาที่แต่ละสถาบันการศึกษาเปิดสอน
6	STATE	เก็บข้อมูลรัฐหรือเมืองของสถาบันการศึกษา
7	COUNTRY	เก็บข้อมูลประเทศ
8	REGION	เก็บข้อมูลทวีปต่างๆ
9	WEB PAGE	เก็บข้อมูล web page ที่มีอยู่ในสถาบันการศึกษา
10	CLUE_LINK	เก็บข้อความเพื่อใช้ตรวจสอบคำที่ใช้เป็น link ของเว็บเพจนั้นๆ ว่ามีรูปแบบของคำที่อยู่ในกลุ่มของคำที่ใช้เป็น link ที่อ้างถึงเอกสารที่มักจะปรากฏข้อมูลทั้งในส่วนของที่อยู่และสาขาวิชาหรือไม่
11	STREET_WEIGHT_LIST	เก็บ keyword ในการพิจารณาข้อความที่ระบุถึงชื่อถนน
12	CONTENT_IN_TAG	เก็บ tag ที่ใช้ในการแยกข้อความภายในเอกสาร HTML ซึ่ง tag ที่จัดเก็บเป็น tag ที่มักพบข้อความที่มีข้อมูลในเรื่องที่สนใจปรากฏอยู่
13	CONTENT_MAJOR_TAG	เก็บ tag ที่ใช้แยกข้อความที่เป็นสาขาวิชาออกจากเอกสาร HTML ซึ่งเป็น tag ที่มักพบรายการของสาขาวิชาปรากฏอยู่
14	WEB_EXCLUSION	ข้อกำหนดในการจัดเก็บ link เพื่อใช้กรอง link ที่จะจัดเก็บให้อยู่ภายในกลุ่มของสถาบันการศึกษา
15	TMP	ตารางชั่วคราวเก็บรายชื่อ link ที่พบ

ตารางที่ 4.1 (ต่อ)

ลำดับที่	ตาราง	คำอธิบายรายละเอียด
16	TMP_U	ตารางชั่วคราวเก็บรายชื่อ link ที่พบอยู่ภายใน webpage ของสถาบันการศึกษา
17	WEB_DUMP_TMP	เก็บเอกสาร HTML ของเว็บเพจที่ Crawler ท่องไป
18	WEB_DUMP_U	เก็บเอกสาร HTML ของเว็บเพจสถาบันการศึกษา
19	START_WEB	เก็บรายชื่อเว็บไซต์เริ่มต้น
20	TOP_REFERENCE	ตารางอ้างอิง 10 อันดับสถาบันการศึกษาในแต่ละสาขาวิชา
21	TOP_DETAIL	แสดงรายละเอียดแต่ละอันดับในแต่ละสาขาวิชา

โดยรายละเอียดของแต่ละตารางสามารถดังแสดงได้ ต่อไปนี้

ตารางที่ 4.2 ตาราง UNIVERSITY

Attribute Name	Description	Data Type	Required	Key	Reference
Uni_No	รหัสสถาบันการศึกษา	Varchar(20)	Y	PK	
Uni_Name	ชื่อสถาบันการศึกษา	Varchar(500)	Y		
Uni_URL	เว็บไซต์สถาบันการศึกษา	Varchar(500)	Y		
Uni_Address	ที่อยู่สถาบันการศึกษา	Varchar(500)	N		
Uni_State	เมืองหรือรัฐที่สถาบันการศึกษาตั้งอยู่	Number(9)	N	FK	STATE
Uni_Country	ประเทศที่สถาบันตั้งอยู่	Varchar(4)	Y	FK	COUNTRY

ตารางที่ 4.3 ตาราง MAJOR_FIELD

Attribute Name	Description	Data Type	Required	Key	Reference
Field_ID	รหัสกลุ่มสาขาวิชา	Number(2)	Y	PK	
Field_Name	ชื่อกลุ่มสาขาวิชา	Varchar(100)	Y		

ตารางที่ 4.4 ตาราง MAJOR

Attribute Name	Description	Data Type	Required	Key	Reference
Major_ID	รหัสสาขาวิชา	Varchar(5)	Y	PK	
Major_Name	ชื่อสาขาวิชา	Varchar(100)	Y		
Major_Field_ID	รหัสกลุ่มสาขาวิชา	Number(2)	Y	FK	MAJOR_ FIELD

ตารางที่ 4.5 ตาราง SUBMAJOR

Attribute Name	Description	Data Type	Required	Key	Reference
SubMaj_ID	รหัสสาขาวิชาย่อย	Varchar(5)	Y	PK	
SubMaj_Name	ชื่อสาขาวิชาย่อย	Varchar(200)	Y		
Major_ID	รหัสสาขาวิชา	Varchar(5)	Y	FK	MAJOR

ตารางที่ 4.6 ตาราง UNI_MAJOR_OFFER

Attribute Name	Description	Data Type	Required	Key	Reference
Uni_No	รหัสสถาบันการศึกษา	Varchar(20)	Y	PK	UNIVERSITY
Major_ID	รหัสสาขาวิชา	Varchar(5)	Y	PK	MAJOR
Uni_Major_Offer	ชื่อสาขาวิชาที่ สถาบันการศึกษาเปิดสอน	Varchar(200)	Y		
Page_U_ID	รหัสหน้าเว็บเพจ	Varchar(50)	Y	FK	WEBPAGE

ตารางที่ 4.7 ตาราง STATE

Attribute Name	Description	Data Type	Required	Key	Reference
State_ID	รหัสรัฐหรือเมือง	Number(5)	Y	PK	
State_Name	ชื่อรัฐหรือเมือง	Varchar(100)	Y		
State_ABB	ชื่อย่อของรัฐหรือเมือง	Varchar(10)	N		
Country_NO	รหัสประเทศ	Number(5)	Y	FK	COUNTRY

ตารางที่ 4.8 ตาราง COUNTRY

Attribute Name	Description	Data Type	Required	Key	Reference
Country_NO	รหัสประเทศ	Varchar(4)	Y	PK	
Country_Name	ชื่อประเทศ	Varchar(100)	Y		
Country_Domain	รหัสโดเมน	Varchar(5)	Y		
Region_NO	รหัสทวีป	Number(4)	Y	FK	REGION

ตารางที่ 4.9 ตาราง REGION

Attribute Name	Description	Data Type	Required	Key	Reference
Region_ID	รหัสทวีป	Number(4)	Y	PK	
Region_Name	ชื่อทวีป	Varchar(50)	Y		

ตารางที่ 4.10 ตาราง WEBPAGE

Attribute Name	Description	Data Type	Required	Key	Reference
Page_U_ID	รหัสหน้าเว็บเพจ	Varchar(50)	Y	PK	
Index_ID	รหัสลำดับชั้นของเว็บเพจ	Number(9)	Y		
Page_URL	ชื่อ URL ของเว็บเพจ	Varchar(500)	Y		
Link_Word	คำที่ใช้เพื่อแสดงถึงlink	Varchar(500)	N		
Prev_Link	รหัสเว็บเพจที่เชื่อมมายังเว็บเพจปัจจุบัน	Varchar(20)	Y	FK	WEBPAGE
Uni_Domain	รหัสสถาบันการศึกษา ระบุว่าเว็บเพจอยู่ในสถาบันใด	Varchar(20)	Y	FK	UNIVERSITY

ตารางที่ 4.11 ตาราง CLUE_LINK

Attribute Name	Description	Data Type	Required	Key	Reference
Clue_Link_ID	รหัสคำที่ใช้ตรวจสอบ link	Number(4)	Y	PK	
Clue_Link_Type	กลุ่มเรื่องที่จะตรวจสอบ (Address,Major)	Varchar(50)	Y		
Clue_Link_Word	คำที่ใช้ตรวจสอบกับข้อความที่ใช้เป็น link	Varchar(50)	Y		

ตารางที่ 4.12 ตาราง STREET_WEIGHT_LIST

Attribute Name	Description	Data Type	Required	Key	Reference
Street_ID	รหัสของชื่อถนน	Number(4)	Y	PK	
Street_Word	คำที่ใช้ระบุชื่อถนน	Varchar(50)	Y		
Street_Weight	ค่าน้ำหนัก	Number(3)	Y		

ตารางที่ 4.13 ตาราง CONTENT_IN_TAG

Attribute Name	Description	Data Type	Required	Key	Reference
Tag_ID	รหัส Tag	Number(4)	Y	PK	
Tag_Open	Tag เปิด	Varchar(20)	Y		
Tag_Close	Tag ปิด	Varchar(20)	Y		
Content_Type	ประเภทของการใช้ Tag เพื่อแยกข้อความ	Varchar(50)	Y		

ตารางที่ 4.14 ตาราง CONTENT_MAJOR_TAG

Attribute Name	Description	Data Type	Required	Key	Reference
Content_Major_ID	รหัส Tag สำหรับแยกสาขาวิชา	Number(4)	Y	PK	
Content_Major_TagOpen	Tag เปิด เพื่อแยกข้อความที่เป็นสาขาวิชา	Varchar(20)	Y		
Content_Major_TagClose	Tag ปิด เพื่อแยกข้อความที่เป็นสาขาวิชา	Varchar(20)	Y		

ตารางที่ 4.15 ตาราง WEB_EXCLUSION

Attribute Name	Description	Data Type	Required	Key	Reference
List_ID	รหัสข้อกำหนดในการเก็บ Link	Number(9)	Y	PK	
List_Exclusion	ข้อกำหนดในการเก็บ link	Varchar(100)	Y		

ตารางที่ 4.16 ตาราง TMP

Attribute Name	Description	Data Type	Required	Key	Reference
URL_ID	รหัสเว็บเพจ	Number(9)	Y	PK	
Index_ID	รหัสลำดับชั้นของเว็บเพจ	Number(9)	Y		
URL_Search	ชื่อ URL ของเว็บเพจ	Varchar(500)	Y		
Crawl_Status	สถานการณ์ Crawl	Varchar(2)	Y		
Link_Word	คำที่ใช้เพื่อแสดงถึงlink	Varchar(500)	N		
Prev_Link	รหัสเว็บเพจที่เชื่อมมายังเว็บเพจปัจจุบัน	Number(9)	Y	FK	TMP

ตารางที่ 4.17 ตาราง TMP_U

Attribute Name	Description	Data Type	Required	Key	Reference
Tmp_U_Id	รหัสเว็บเพจของสถาบันการศึกษา	Number(9)	Y	PK	
Index_ID	รหัสลำดับชั้นของเว็บเพจ	Number(9)	Y		
Tmp_U_URL	ชื่อ URL ของเว็บเพจ	Varchar(500)	Y		
Crawl_Status	สถานการณ์ Crawl	Varchar(2)	Y		
Link_Word	คำที่ใช้เพื่อแสดงถึงlink	Varchar(500)	N		
Prev_Link	รหัสเว็บเพจที่เชื่อมมายังเว็บเพจปัจจุบัน	Number(9)	Y	FK	TMP_U
Uni_Domain	รหัสสถาบันการศึกษา ระบุว่าเว็บเพจอยู่ในสถาบันใด	Varchar(20)	Y	FK	UNIVERSITY

ตารางที่ 4.18 ตาราง WEB_DUMP_TMP

Attribute Name	Description	Data Type	Required	Key	Reference
URL_ID	รหัสเว็บเพจ	Varchar(50)	Y	PK	TMP
URL_Content	เนื้อความเอกสารเว็บเพจ	Text	Y		
URL_Content_Len	ความยาวของเอกสาร	Number(9)	N		
Crawl_Date	วันเวลาในการ Crawl	Date	Y		

ตารางที่ 4.19 ตาราง WEB_DUMP_U

Attribute Name	Description	Data Type	Required	Key	Reference
Page_U_ID	รหัสเว็บเพจสถาบันการศึกษา	Varchar(50)	Y	PK	WEBPAGE
URL_Content	เนื้อความเอกสารเว็บเพจ	Text	Y		
URL_Content_Len	ความยาวของเอกสาร	Number(9)	N		
Crawl_Date	วันเวลาในการ Crawl	Date	Y		

ตารางที่ 4.20 ตาราง START_WEB

Attribute Name	Description	Data Type	Required	Key	Reference
Startweb_ID	รหัสเว็บไซต์เริ่มต้น	Number(4)	Y	PK	
Startweb_URL	URL ของเว็บไซต์เริ่มต้น	Varchar(100)	Y		
Startweb_Desc	คำอธิบายประกอบเว็บไซต์	Varchar(200)	N		

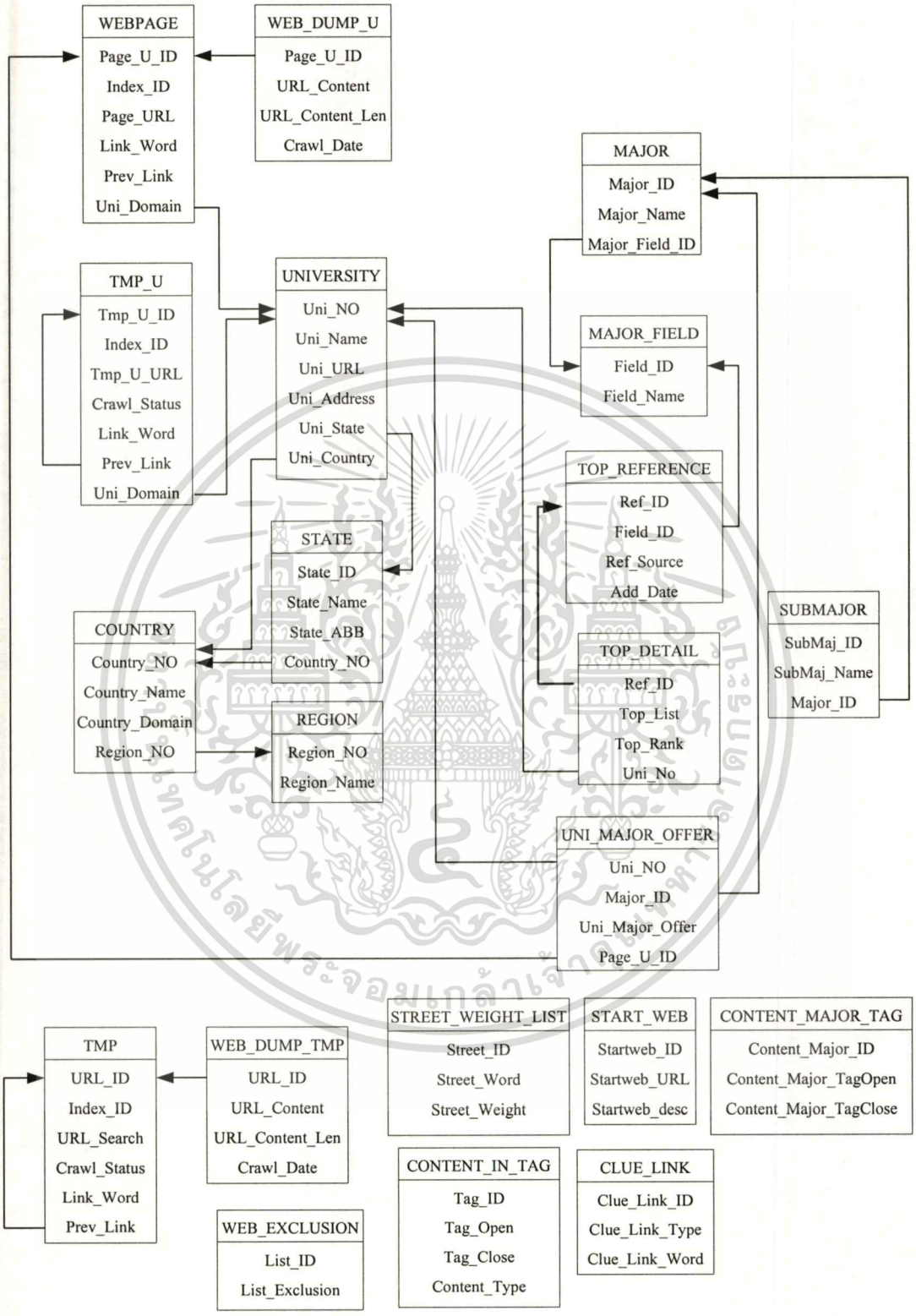
ตารางที่ 4.21 ตาราง TOP_REFERENCE

Attribute Name	Description	Data Type	Required	Key	Reference
Ref_ID	รหัสอ้างอิง	Number(9)	Y	PK	
Field_ID	รหัสกลุ่มสาขาวิชา	Number(2)	Y	FK	MAJOR_ FIELD
Ref_Source	แหล่งอ้างอิง	Varchar(100)	N		
Add_Date	วันที่บันทึก	Date	Y		

ตารางที่ 4.22 ตาราง TOP_DETAIL

Attribute Name	Description	Data Type	Required	Key	Reference
Ref_ID	รหัสอ้างอิง	Number(9)	Y	PK	
Top_List	อันดับในการบันทึก	Number(2)	Y	PK	
Top_Rank	อันดับใน Top Ten	Number(2)	N		
Uni_No	รหัสสถาบันการศึกษา	Varchar(20)	Y	FK	UNIVERSITY

ซึ่งจากตารางที่ใช้สำหรับพัฒนาระบบฐานข้อมูลสำหรับงานบริการการศึกษานี้ สามารถแสดงความสัมพันธ์ระหว่างตารางข้อมูลที่ได้สร้างขึ้นเพื่อใช้ในการพัฒนา โดยแสดงดังรูปที่ 4.7



รูปที่ 4.7 ความสัมพันธ์ระหว่างตารางข้อมูลที่ใช้ในการพัฒนาระบบฐานข้อมูลสำหรับงานบริการการศึกษา

บทที่ 5
การพัฒนาระบบงาน

ในการพัฒนาระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษานี้ ได้แบ่งส่วนการทำงานดังนี้คือ

- ส่วนการเก็บรวบรวมและดึงข้อมูลจากเอกสาร HTMLจากเครือข่ายอินเทอร์เน็ต ได้พัฒนาโดยใช้ภาษา Visual Basic 6.0 สร้าง Web Crawler เพื่อใช้ในการท่องไปยังเว็บไซต์ของสถาบันการศึกษาต่างๆและรวบรวมข้อมูลที่ต้องการมาจัดเก็บลงในฐานข้อมูล
- ส่วนการสืบค้นข้อมูลและการแสดงผล จะพัฒนาเป็นแบบ online เพื่อใช้งานผ่าน Web Browser โดยพัฒนาด้วยภาษา ASP และ HTML
- ส่วนจัดการฐานข้อมูล ได้มีการใช้ระบบจัดการฐานข้อมูล Microsoft SQL Server 2000 สำหรับเก็บข้อมูลต่างๆของสถาบันการศึกษา

ซึ่งการพัฒนาระบบงานในแต่ละส่วน สามารถแสดงได้ดังต่อไปนี้

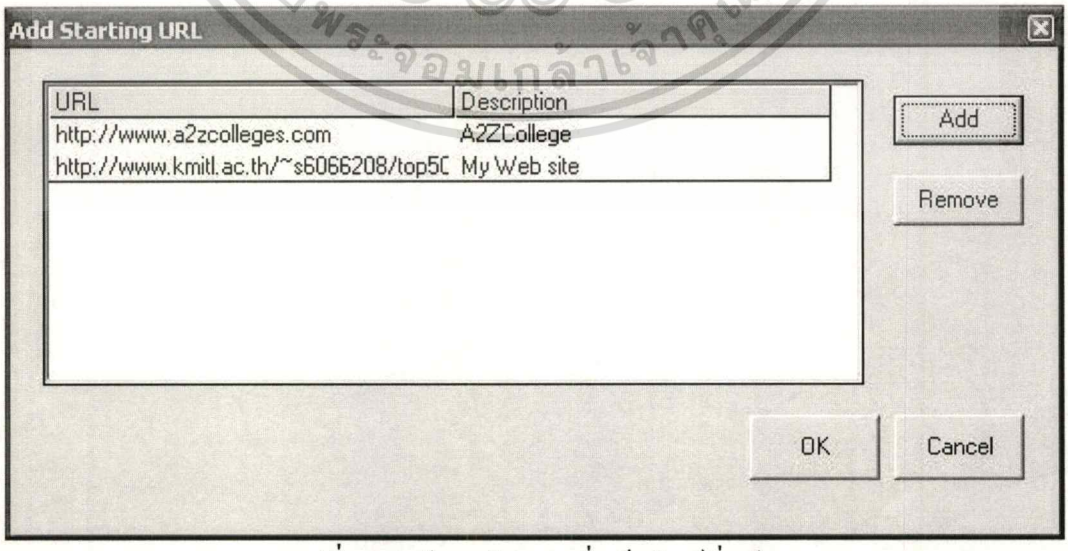
5.1 การพัฒนาส่วนการเก็บรวมรวมข้อมูลและประมวลผลเอกสาร HTML สำหรับงานบริการการศึกษา

ในส่วนการเก็บรวบรวมข้อมูลและประมวลผลเอกสาร HTML สำหรับงานบริการการศึกษานี้ จะเป็นทำงานอยู่บนเครื่องที่จัดเก็บฐานข้อมูลงานบริการการศึกษา โดยได้มีการพัฒนาโปรแกรม Web Crawler เพื่อใช้ในการรวบรวมเอกสาร HTML ของแต่ละสถาบันการศึกษา และดึงเฉพาะข้อมูลในส่วนที่ต้องการ ซึ่งได้แก่ข้อมูลที่ตั้งสถาบันการศึกษา และข้อมูลสาขาวิชาที่สถาบันการศึกษานั้นเปิดสอน โดยการทำงานของ Web Crawler นี้จะทำการท่องไปยังเว็บเพจต่างๆ โดยการอ่านข้อมูลในรูปของภาษา HTML ซึ่งหากภายในเว็บเพจนั้นมี hyperlink ที่เชื่อมต่อไปยังเอกสารอื่นๆ โดยใช้ tag <A> โปรแกรมจะทำการเก็บรายชื่อ hyperlink นั้นเพื่อใช้เป็นข้อมูลนำเข้าสำหรับการท่องไปยังเอกสารอื่นๆต่อไป และเมื่อโปรแกรมพบเว็บเพจที่มีข้อมูลที่ต้องการ ก็จะทำการดึงเอาข้อมูลที่ต้องการนั้นมาจัดเก็บลงฐานข้อมูล

สำหรับการทำงานของโปรแกรม Web Crawler เพื่อใช้สำหรับรวบรวมและประมวลผลเอกสาร HTML สามารถแสดงการทำงานได้ ดังนี้

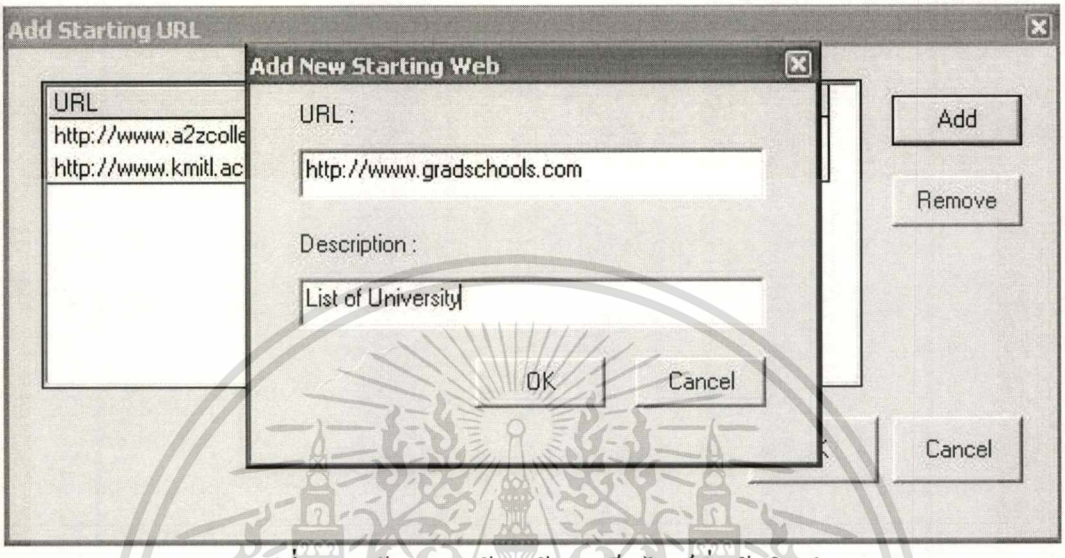
รูปที่ 5.1 หน้าจอหลักเมื่อเริ่มใช้งานโปรแกรม Web Crawler

โดยเมื่อเริ่มทำงานนั้นจะต้องมีการกำหนดเว็บไซต์เริ่มต้นให้กับ Web Crawler จากปุ่ม  ซึ่งจะปรากฏหน้าจอสำหรับเลือกรายชื่อของเว็บไซต์เริ่มต้น ดังรูปที่ 5.2



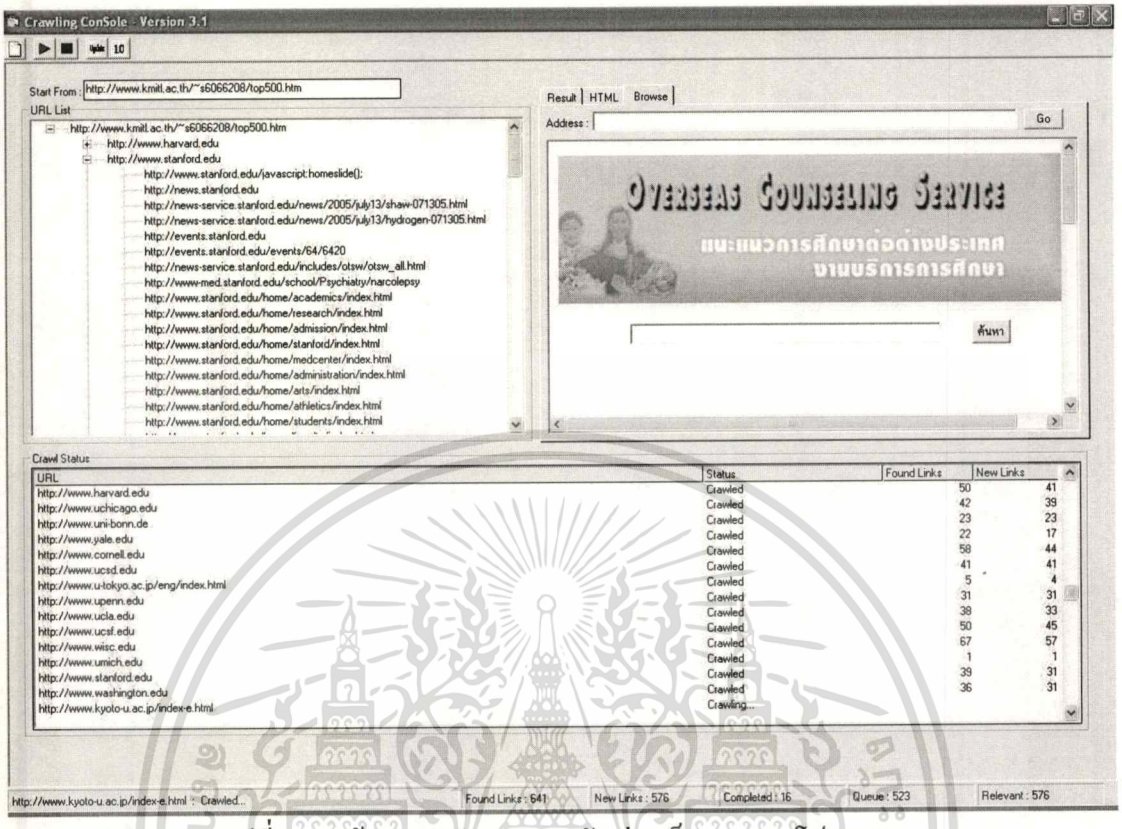
รูปที่ 5.2 หน้าจอเลือกรายชื่อเว็บไซต์เริ่มต้น

หากต้องการกำหนดรายชื่อเว็บไซต์ใหม่เข้าไป กดปุ่ม Add จะปรากฏหน้าจอการป้อนรายชื่อเว็บไซต์เริ่มต้นใหม่ที่ต้องการ ดังรูปที่ 5.3



รูปที่ 5.3 หน้าจอการป้อนข้อมูลเว็บไซต์เริ่มต้นใหม่

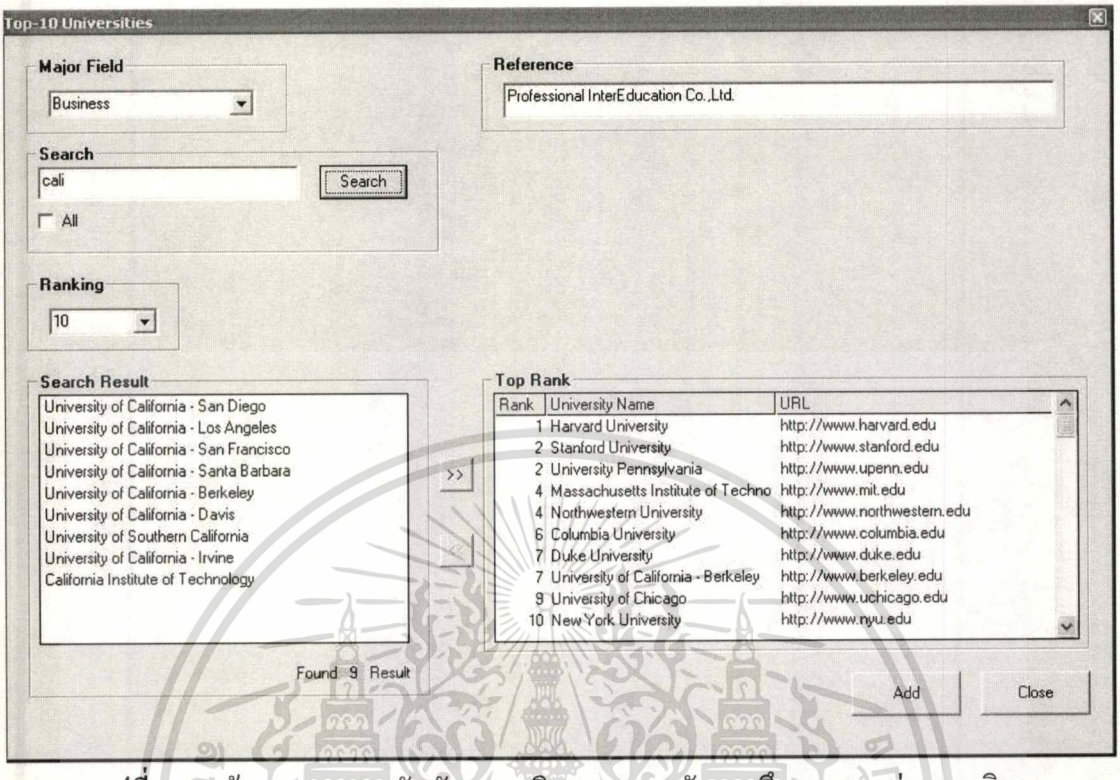
เมื่อเริ่มการทำงาน โดยกดปุ่ม โปรแกรมจะท่องไปยังเว็บเพจต่างๆ โดยเริ่มจากเว็บไซต์เริ่มต้นที่เลือกไว้ ซึ่งตัวโปรแกรมจะทำการอ่านเว็บเพจนั้นเป็นภาษา HTML และค้นหา URL ที่เป็น Hyperlink ในเอกสาร จากนั้นจึงจัดเก็บลงในฐานข้อมูลเพื่อใช้เป็นข้อมูลนำเข้าสำหรับการท่องไปยังเอกสารอื่นๆต่อไป สำหรับหน้าจอการทำงานสำหรับท่องเว็บเพจของโปรแกรมนั้นแสดงได้ดังรูปที่ 5.4



รูปที่ 5.4 หน้าจอการทำงานสำหรับท่องเว็บเพจของโปรแกรม

โดยหน้าจอการทำงาน ด้านบนจะเป็นช่องที่แสดงชื่อของเว็บไซต์เริ่มต้น และจะมีส่วนที่แสดงถึงลำดับชั้นของเว็บเพจแต่ละหน้า โดยที่ชื่อของ URL จะแสดงเรียงเป็นลำดับชั้น เพื่อให้ดูเข้าใจง่ายยิ่งขึ้น กรอบด้านล่างนั้นจะแสดงรายชื่อของ URL ที่ได้ทำการท่องไปแล้ว พร้อมทั้งบอกถึงจำนวน hyperlink ที่พบภายในแต่ละเอกสารด้วย โดยแต่ละ hyperlink นั้นจะถูกเก็บอยู่ในฐานข้อมูล และจะเสร็จสิ้นการทำงานของโปรแกรม ก็ต่อเมื่อกดหยุด ที่ปุ่ม  หรือโปรแกรมได้ทำการท่องไปยังเว็บเพจตาม URL ที่มีในฐานข้อมูลนั้นหมดแล้ว

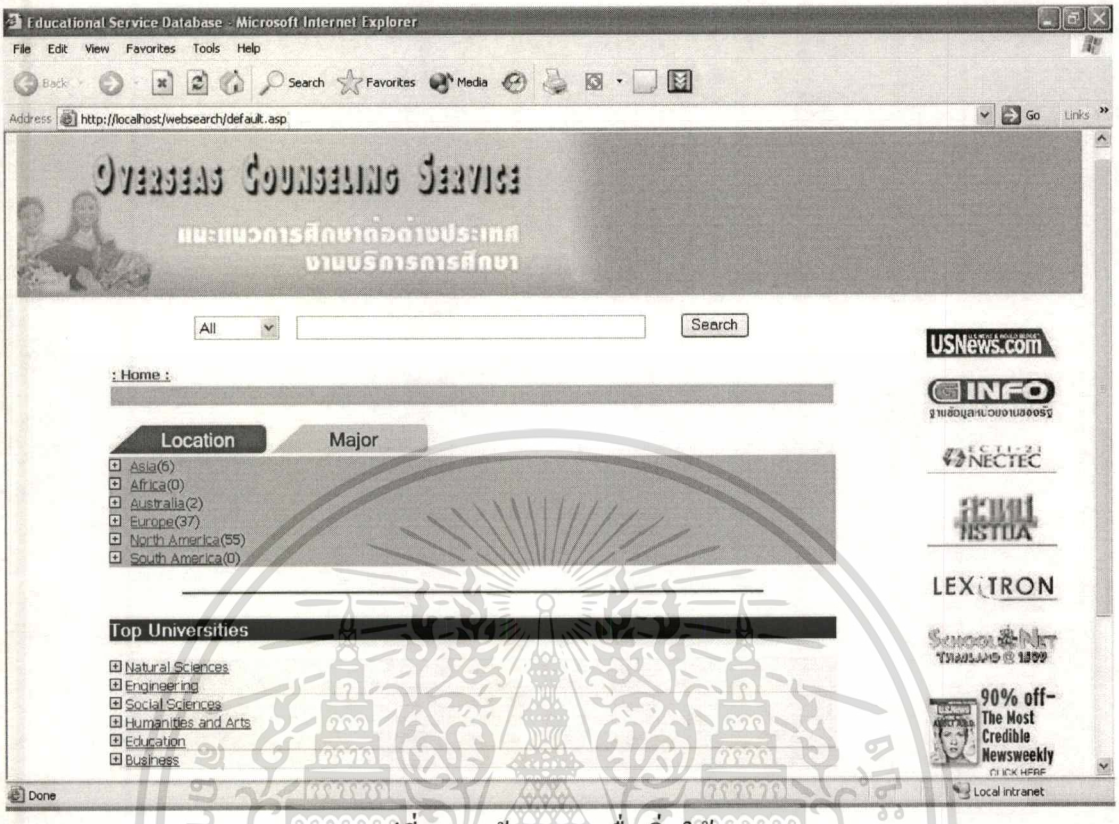
สำหรับการทำงานของโปรแกรมนี้ ยังมีส่วนหน้าจอสำหรับกรอกอันดับสถาบันการศึกษาที่ได้รับความนิยม ตามกลุ่มของสาขาวิชา ซึ่งใช้ข้อมูลที่อ้างอิงมาจากแหล่งข้อมูลที่ได้รับความเชื่อถือ สำหรับหน้าจอการกรอกอันดับความนิยมของสถาบันการศึกษาตามกลุ่มสาขาวิชา สามารถแสดงได้ดังรูปที่ 5.5



รูปที่ 5.5 หน้าจอการกรอกอันดับความนิยมของสถาบันการศึกษาตามกลุ่มสาขาวิชา

5.2 การพัฒนาส่วนการสืบค้นข้อมูลและการแสดงผล

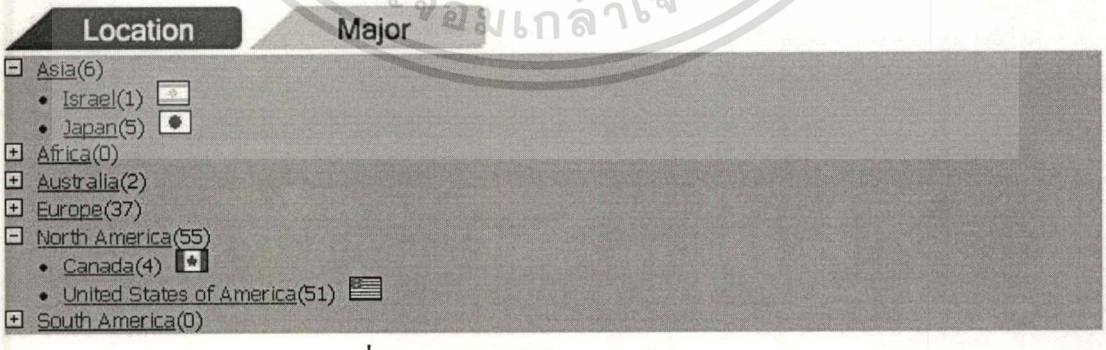
ในส่วนติดต่อกับผู้ใช้นี้ จะพัฒนาเพื่อใช้บนเครือข่ายอินเทอร์เน็ต เพื่อให้ผู้ใช้สามารถใช้งานข้อมูลที่จัดเก็บได้จากโปรแกรม Web Crawler ที่ได้พัฒนาขึ้น ซึ่งได้มีการออกแบบหน้าจอแบบ Graphic User Interface เพื่อให้ผู้ใช้ระบบสามารถใช้งานได้ง่าย และสะดวกยิ่งขึ้น โดยหน้าจอการทำงาน แสดงได้ดังรูปที่ 5.6



รูปที่ 5.6 หน้าจอแรกเมื่อเริ่มใช้งาน

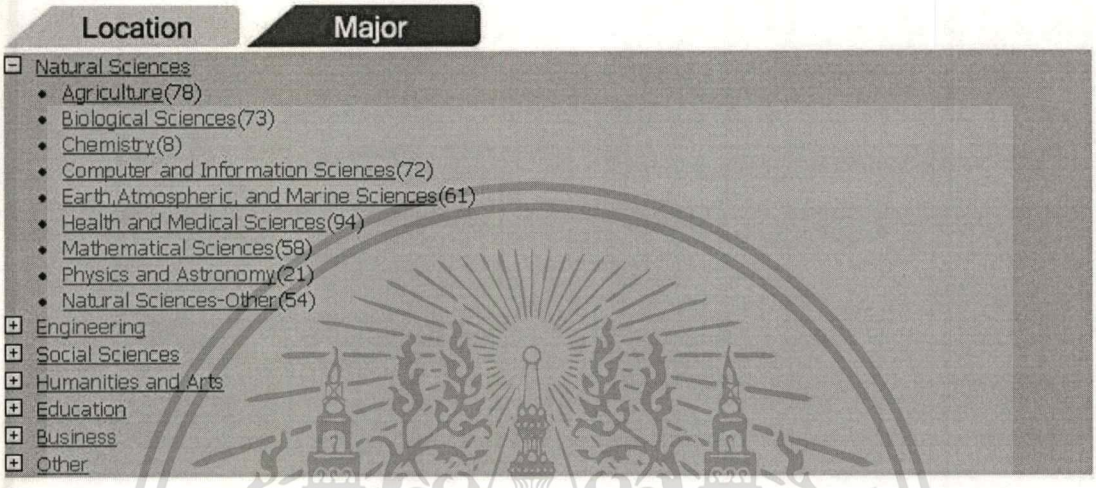
โดยหน้าจอการแสดงผล จะสามารถแสดงข้อมูลใน 2 มิติคือ

- มิติด้าน Location จะแบ่งสถาบันการศึกษาออกตามสถานที่ตั้ง โดยแบ่งเป็นทวีป และประเทศ โดยผู้ใช้สามารถเลือกพื้นที่ตามที่ผู้ใช้ต้องการดูข้อมูลได้ ดังแสดงในรูปที่ 5.7



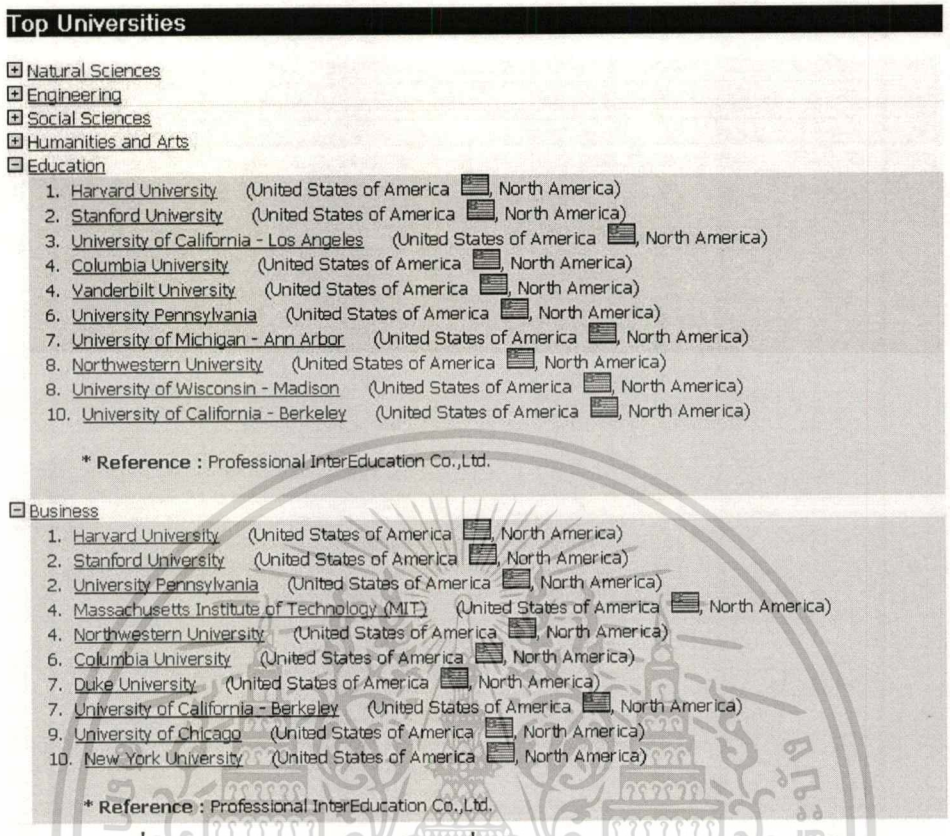
รูปที่ 5.7 การแสดงข้อมูลมิติด้าน Location

- มิติด้าน Major จะแบ่งสถาบันการศึกษาออกตามสาขาวิชาที่มีการเปิดสอนอยู่ ซึ่งผู้ใช้สามารถที่จะเลือกสาขาวิชาที่ต้องการศึกษาต่อ เพื่อดูว่ามีสถาบันการศึกษาใดเปิดสอนอยู่บ้าง ดังแสดงในรูปที่ 5.8



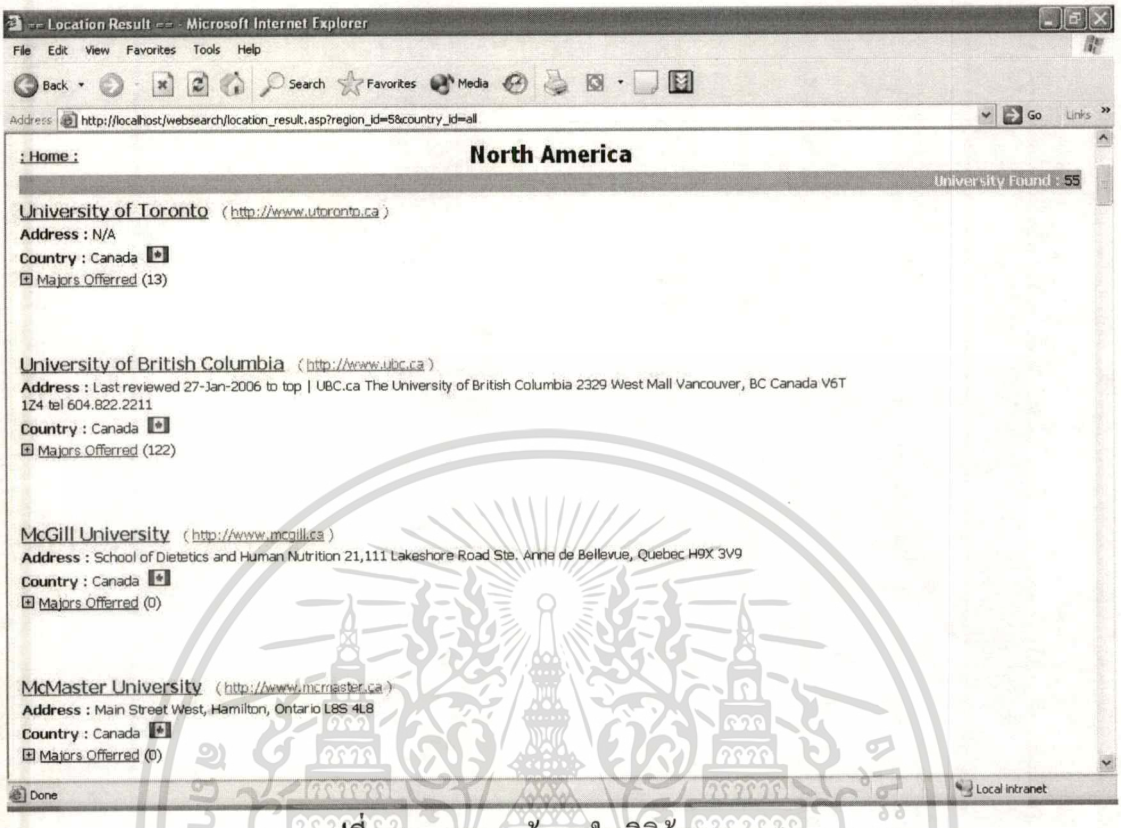
รูปที่ 5.8 การแสดงข้อมูลมิติด้าน Major

ทั้งนี้ ยังมีส่วนของการแสดงอันดับความนิยมของสถาบันการศึกษาตามกลุ่มของสาขาวิชาต่างๆ ซึ่งได้รวบรวมจากแหล่งข้อมูลมานำเสนอ เพื่อช่วยในการตัดสินใจของผู้ใช้ในการเลือกสถาบันการศึกษา ดังแสดงในรูปที่ 5.9



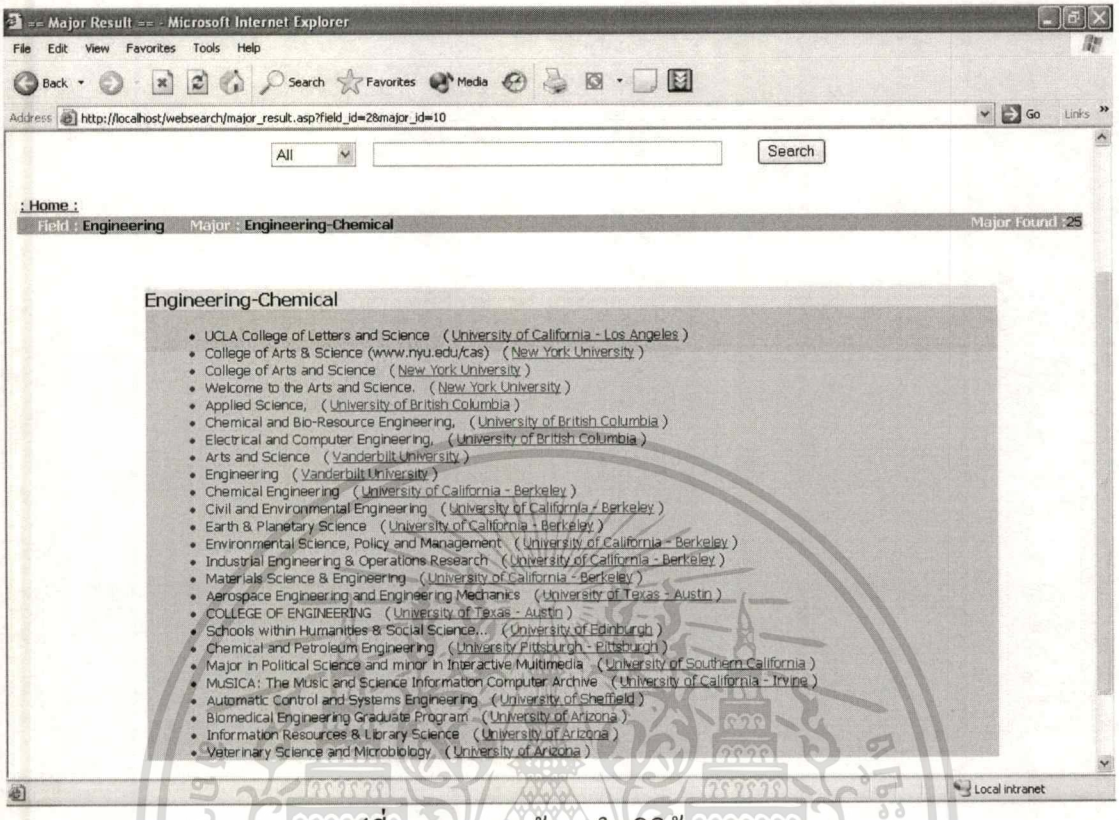
รูปที่ 5.9 แสดงสถาบันการศึกษาที่ได้รับความนิยม ตามกลุ่มสาขาวิชา

เมื่อผู้ใช้ทำการเลือกหัวข้อในการค้นหาแล้ว จะแสดงผลการค้นหาตามมิติที่ผู้ใช้เลือก ดังแสดงในรูปที่ 5.10 และ รูปที่ 5.11



รูปที่ 5.10 ผลการค้นหาในมิติด้าน Location

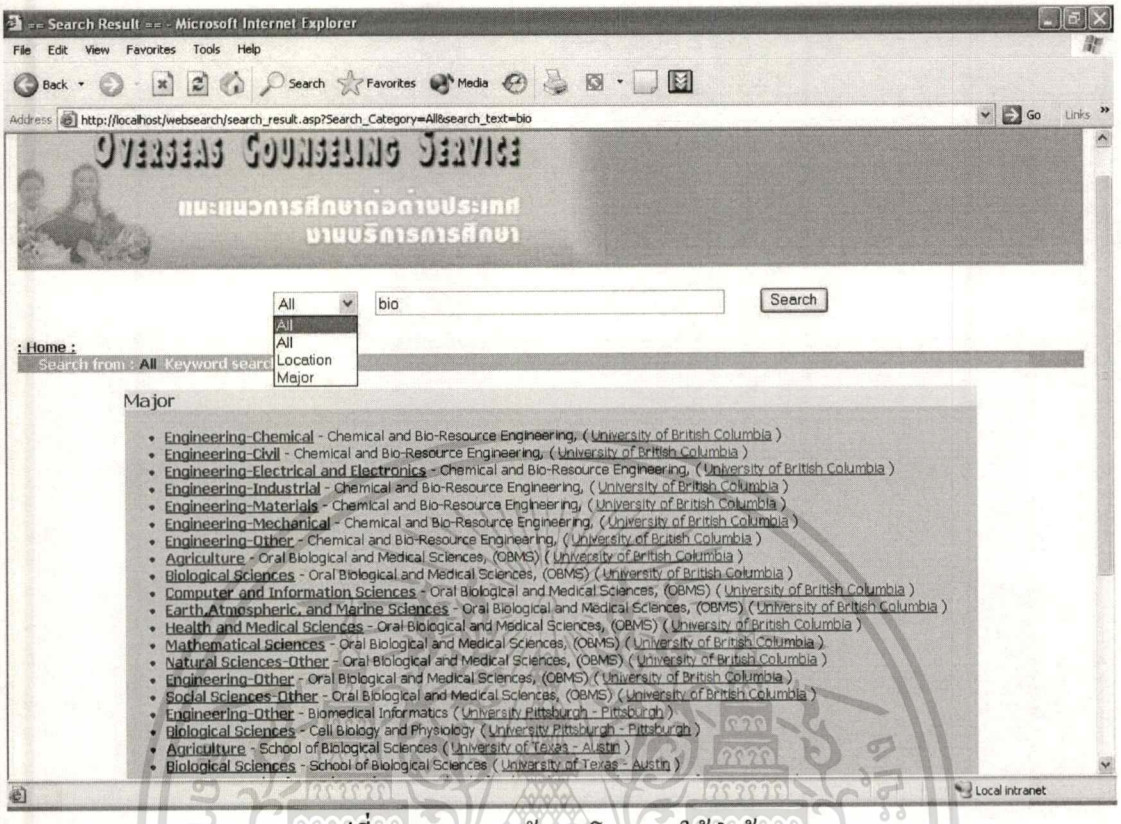
จากรูปที่ 5.10 นี้ เป็นผลของการค้นหาข้อมูลในมิติด้าน Location ซึ่งผู้ใช้ทำการเลือกทวีปหรือประเทศที่สนใจ โดยผลของการค้นหานี้จะแสดงรายชื่อของสถาบันการศึกษา ที่ตั้ง ประเทศ และรายชื่อของสาขาวิชาที่สถาบันการศึกษานั้นๆเปิดสอน



รูปที่ 5.11 ผลการค้นหาในมิติด้าน Major

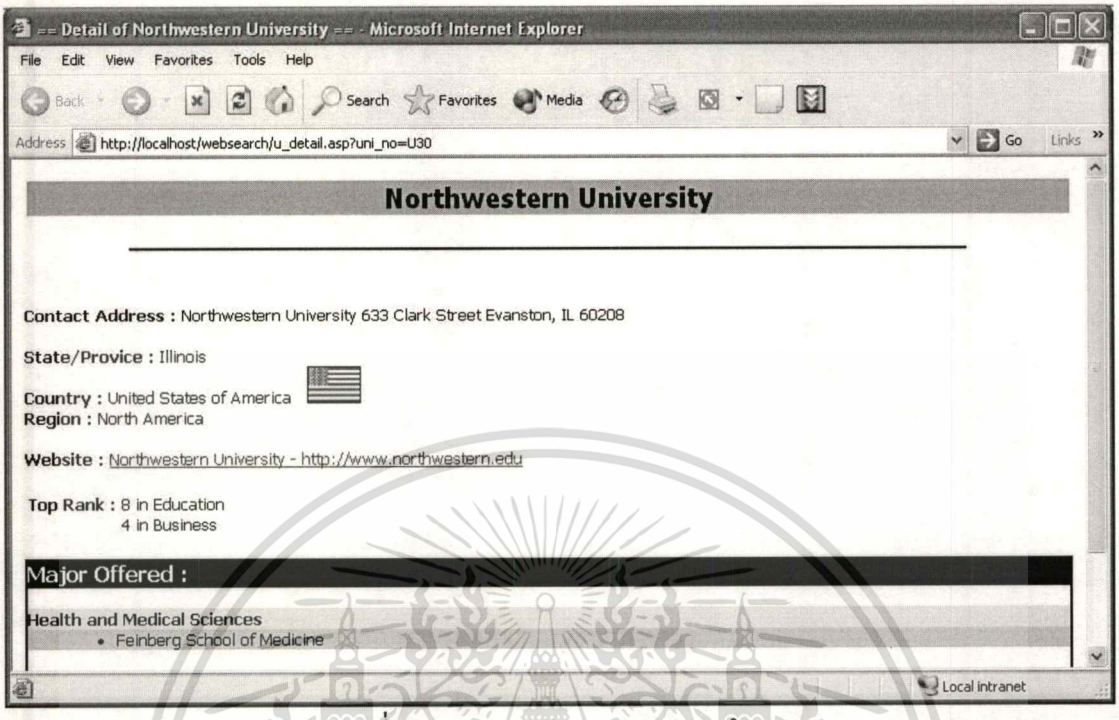
จากรูปที่ 5.11 เป็นผลการค้นหาข้อมูลในมิติด้าน Major ซึ่งผู้ใช้ทำการเลือกกลุ่มสาขาวิชาหรือสาขาวิชาที่สนใจ โดยผลของการค้นหาจะทำการแสดงชื่อของสาขาวิชาที่แต่ละสถาบันการศึกษาเปิดสอน โดยจัดกลุ่มตามสาขาวิชา

ทั้งนี้ ในการค้นหาข้อมูลสามารถแสดงผลการค้นหาจากการใช้คำค้นหาได้ โดยผู้ใช้สามารถพิมพ์คำค้นหาที่สนใจ และเลือกมิติค้นหาว่าจะค้นหาตามมิติของ Location หรือ Major หรือค้นหาจากทั้ง 2 มิติ ซึ่งหน้าจอแสดงผล ดังแสดงในรูปที่ 5.12



รูปที่ 5.12 ผลการค้นหาโดยการใช้คำค้นหา

เมื่อผู้ใช้ต้องการดูรายละเอียดของแต่ละสถาบันการศึกษา สามารถเลือกดูตามชื่อของสถาบันการศึกษานั้นๆได้ โดยจะมีการแสดงหน้าจอรายละเอียดของสถาบันการศึกษานั้นออกมา ดังรูปที่ 5.13



รูปที่ 5.13 รายละเอียดสถาบันการศึกษา

โดยหน้าจอนี้จะแสดงรายละเอียด ดังนี้

- ที่ตั้ง
- ชื่อรัฐ/เมือง
- ประเทศ
- ทวีป
- เว็บไซต์ของสถาบันการศึกษา
- อันดับความนิยมตามสาขา
- รายชื่อสาขาวิชาที่เปิดสอน

ซึ่งข้อมูลที่นำเสนอ เป็นข้อมูลที่ถูกรวบรวมมาโดยโปรแกรม Web Crawler และจัดเก็บไว้ในฐานข้อมูลสำหรับงานบริการการศึกษา

บทที่ 6

ผลการทำงานของระบบ

6.1 การทดสอบการทำงานของระบบ

ในการทำงานของต้นแบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา โดยการรวบรวมข้อมูลที่เกี่ยวกับสถาบันการศึกษา ซึ่งได้แก่ ข้อมูลที่ตั้งและสาขาวิชาของแต่ละสถาบันการศึกษา ซึ่งได้นำ Web Crawler ที่ได้พัฒนาขึ้นมาใช้ในการรวบรวมข้อมูล โดยได้กำหนดเงื่อนไขในการทดสอบการทำงาน ดังนี้

- ในการทำงานของ Web Crawler ได้ทำการกำหนดเว็บเพจเริ่มต้นจาก โดยใช้ URL คือ “http://www.kmitl.ac.th/~s6066208/top500.htm“ ดังตัวอย่างหน้าเว็บเพจในรูปที่ 6.1

Top 500 World Universities (1-100)

World Rank	Institution	Country	Total Score	Score on	Score on	Score on	Score on	Score on
1	Harvard University	USA	100	98.61	100	100	100	100	60.57
2	Stanford University	USA	77.24	41.23	72.16	96.1	75.23	72.28	68.12
3	University of Cambridge	UK	76.18	100	93.41	56.62	58.54	70.2	73.17
4	University of California - Berkeley	USA	74.24	70.01	76	74.13	75.64	72.73	45.08
5	Massachusetts Institute of Technology (MIT)	USA	72.36	74.12	78.86	73.62	69.11	64.62	47.54
6	California Institute of Technology	USA	68.98	59.31	66.52	64.79	66.65	53.19	100
7	Princeton University	USA	63.63	60.95	76.81	65.38	52.06	46.81	67.25
8	University of Oxford	UK	61.39	64.42	59.08	53.14	55.32	65.17	58.96
9	Columbia University	USA	61.19	77.76	58.84	57.29	51.63	68.29	37.04
10	University of Chicago	USA	60.52	72.23	81.87	55.26	46.61	54.06	32.69
11	Yale University	USA	58.62	52.22	44.5	63.6	58.09	63.61	50.38
12	Cornell University	USA	55.55	46.62	52.37	60.53	47.19	66.17	33.65
13	University of California - San Diego	USA	53.75	17.78	34.71	63.6	59.41	67.23	47.89
14	Tokyo University	Japan	51.9	36.12	14.41	44.55	55.04	91.9	49.85
15	University Pennsylvania	USA	51.77	35.56	35.11	61.16	44.63	72.6	34.01
16	University of California - Los Angeles	USA	51.59	27.4	32.76	60.53	48.12	79.88	24.82

Done

รูปที่ 6.1 เว็บเพจเริ่มต้นสำหรับการเก็บข้อมูลโดย Web Crawler

จากรูปที่ 6.1 เป็นเว็บเพจเริ่มต้นในการเก็บรวมรวมข้อมูลของ Web Crawler ซึ่งเป็นเว็บเพจที่ได้จัดทำขึ้นโดยรวบรวมรายชื่อสถาบันการศึกษาต่างๆ

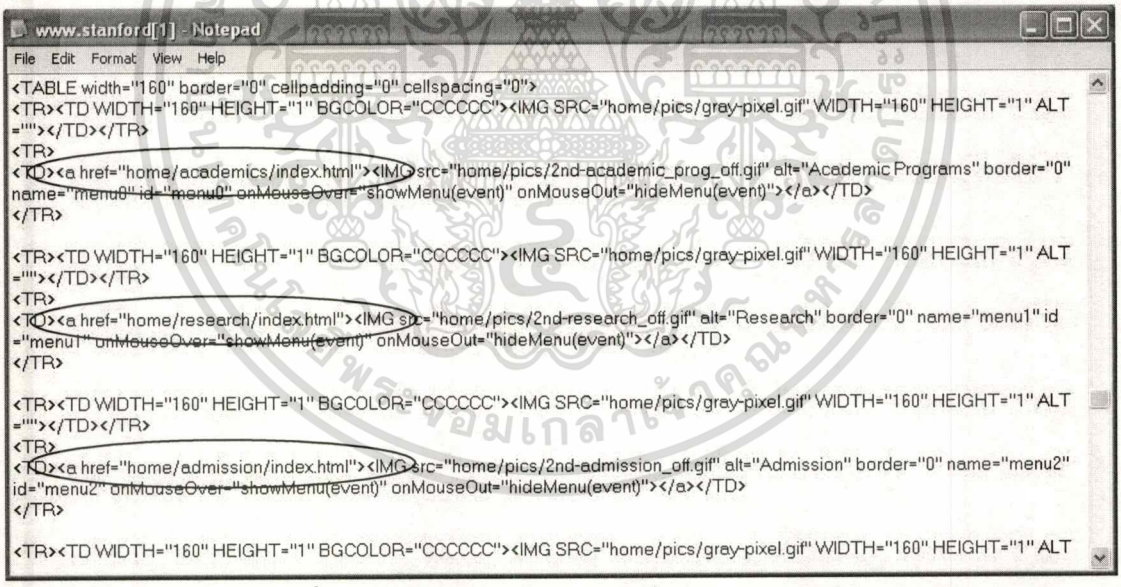
- จำนวนเว็บไซต์ของสถาบันการศึกษาที่ใช้ในการทดสอบมีจำนวน 100 แห่ง
- Web Crawler ทำการท่องไปยังเว็บเพจต่างๆได้เป็นจำนวน 11,814 เว็บเพจ

ซึ่งในการทดสอบการทำงานของระบบนั้น จะทำการตรวจสอบข้อมูลที่ถูกจัดเก็บมาได้และสามารถแสดงผลให้กับผู้ใช้งานได้ รวมไปถึงการวัดประสิทธิภาพในการดึงข้อมูลของโปรแกรมในส่วนของที่ตั้งสถาบันการศึกษา และส่วนของสาขาวิชาที่สถาบันการศึกษานั้นเปิดสอน

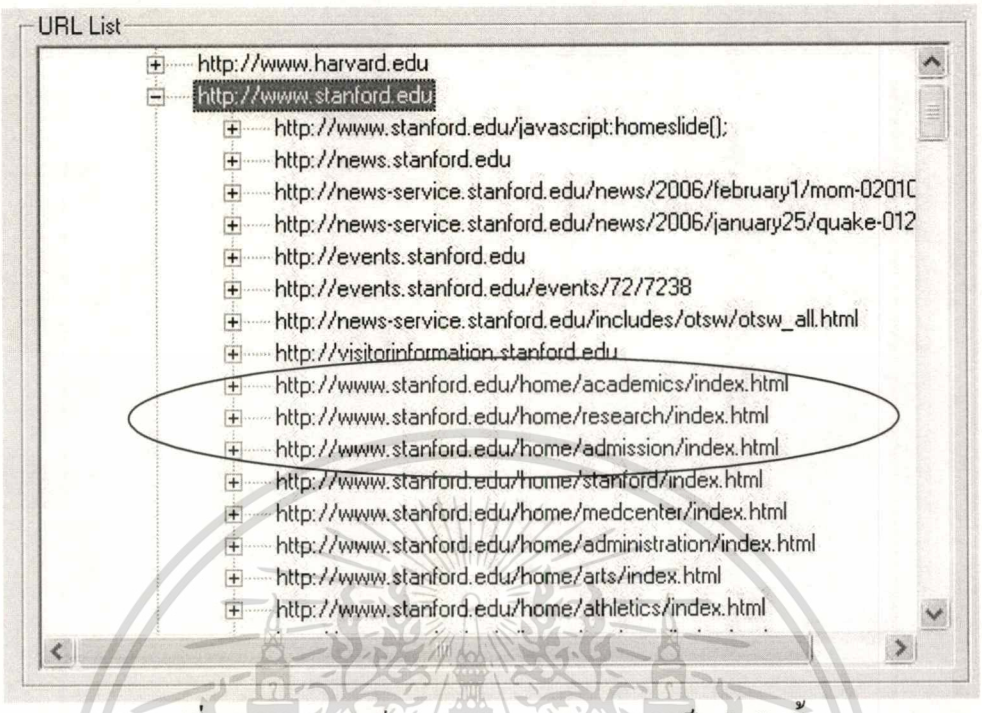
6.2 ผลการทำงานของระบบ

6.2.1 ผลการทำงานในการรวบรวม Hyperlink ที่พบในเอกสาร HTML

ในการรวบรวม hyperlink ที่พบในเอกสาร HTML นี้ จะทำการแสดงตัวอย่างของ hyperlink ที่พบในเอกสาร และผลที่ได้จากการจัดเก็บ ดังรูปที่ 6.2 และ รูปที่ 6.3



รูปที่ 6.2 ตัวอย่างเอกสาร HTML ที่ปรากฏ hyperlink

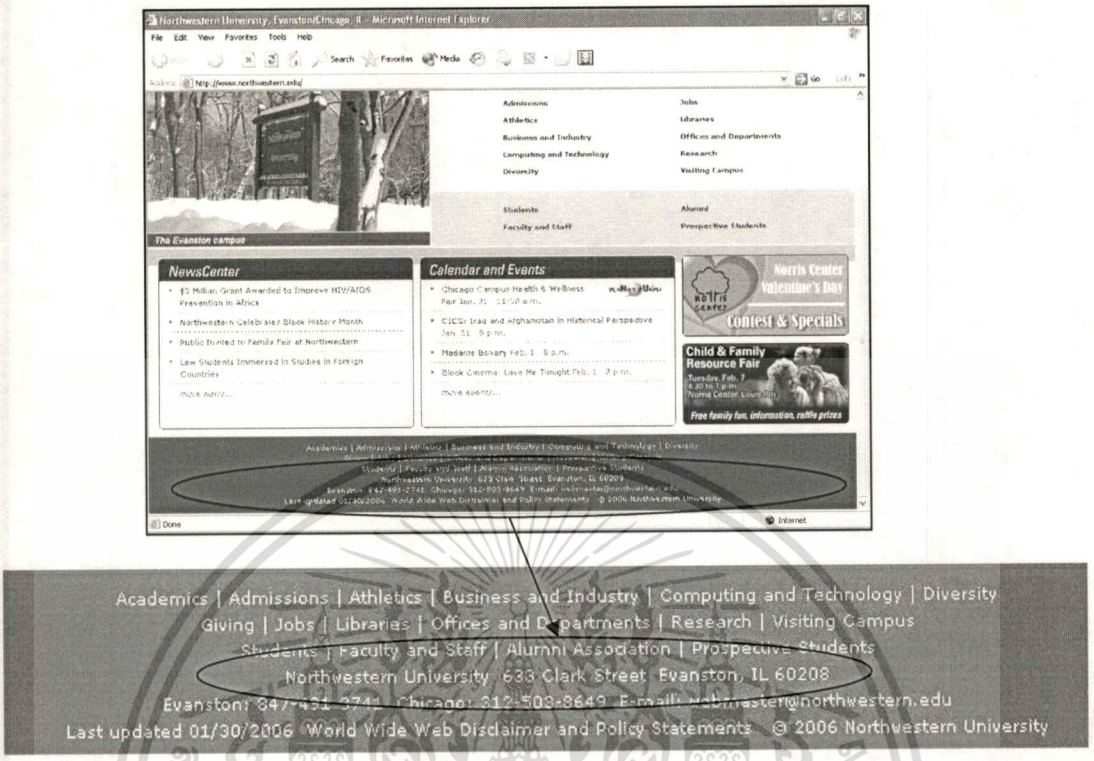


รูปที่ 6.3 ผลการจัดเก็บ hyperlink และนำเสนอเป็นลำดับชั้น

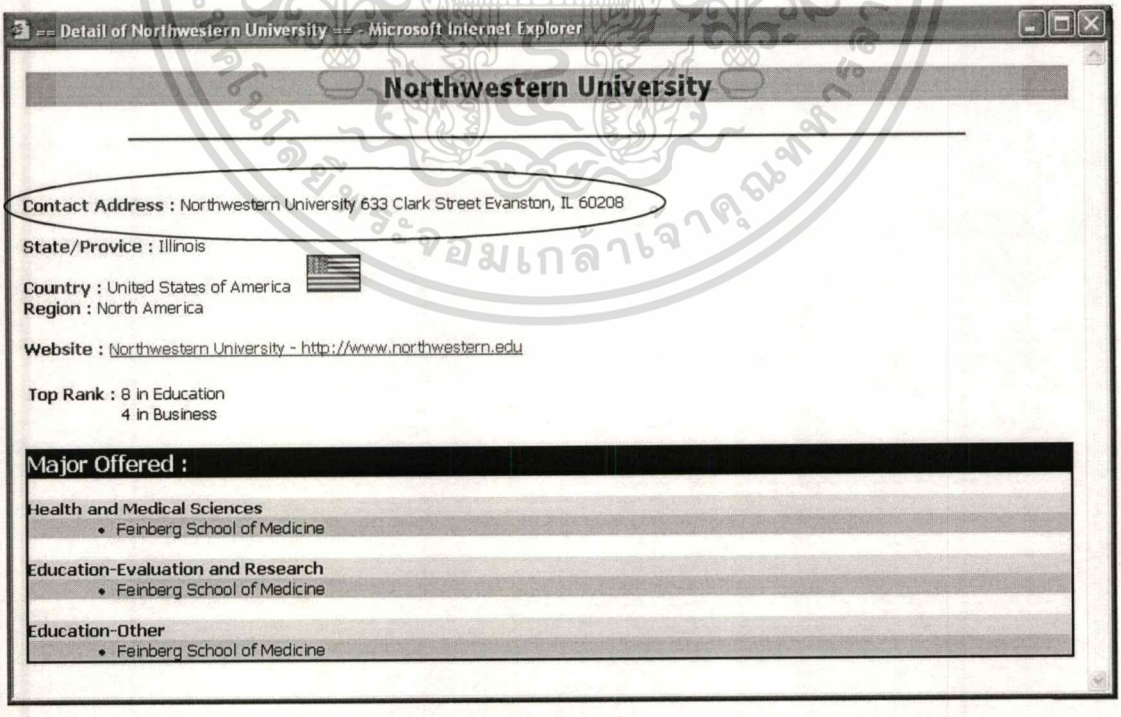
จากรูปที่ 6.2 เป็นตัวอย่างเอกสาร HTML ของ Stanford University ซึ่งจะปรากฏ hyperlink ภายใต้ tag<A> และในกรณีนี้มีการอ้างถึงเอกสารอื่น โดยใช้ Relative Path ซึ่งตัวโปรแกรมจะทำการแปลงจาก Relative Path ให้เป็น Absolute Path แล้วทำการจัดเก็บลงฐานข้อมูล ซึ่งจากรูปที่ 6.3 เป็นรายชื่อของ URL ที่เป็น hyperlink ที่อยู่ในฐานข้อมูล โดยจะมีการแสดงเป็นลำดับชั้นเพื่อให้ทราบว่า URL นี้เป็น hyperlink ที่มาจากเอกสารใด ซึ่งโปรแกรม Web Crawler จะใช้ URL ที่เก็บได้นี้มาใช้เป็นข้อมูลนำเข้าเพื่อท่องไปยังเว็บเพจนั้นๆต่อไป

6.2.2 ผลการทำงานในการรวบรวมข้อมูลที่ตั้งสถาบันการศึกษา

ในการรวบรวมข้อมูลที่ตั้งของสถาบันการศึกษานั้น จะแสดงผลทางหน้าจอส่วนที่ติดต่อกับผู้ใช้ผ่านทาง Web Browser โดยจะแสดงตัวอย่างข้อมูลที่ตั้งของสถาบันการศึกษาที่ปรากฏในเว็บเพจ และผลของข้อมูลที่จัดเก็บได้ โดยโปรแกรม ดังรูปที่ 6.4 และรูปที่ 6.5



รูปที่ 6.4 ข้อมูลที่ตั้งสถาบันการศึกษาที่พบในเว็บเพจ

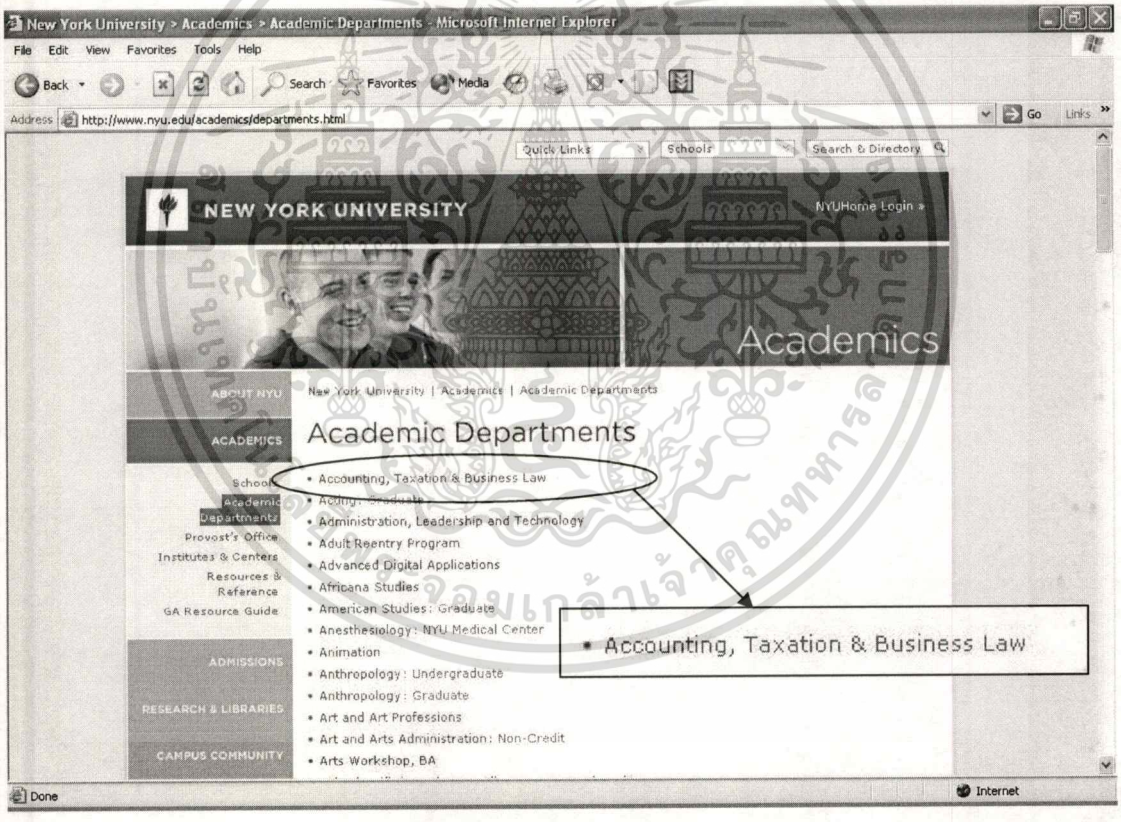


รูปที่ 6.5 ผลการจัดเก็บข้อมูลที่ตั้งของสถาบันการศึกษา

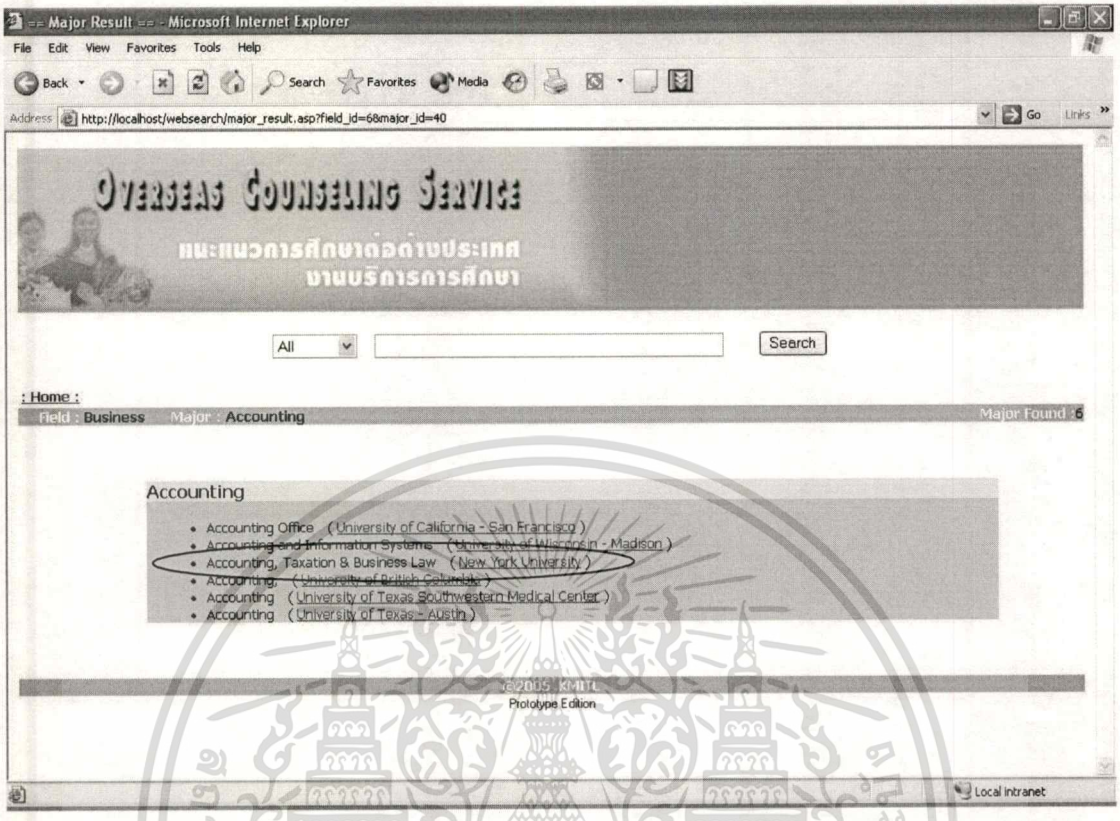
จากรูปที่ 6.4 เป็นตัวอย่างเว็บเพจของ Northwestern University ที่ปรากฏข้อมูลที่ตั้งสถาบันการศึกษาอยู่ที่ด้านล่างของเว็บเพจ ซึ่งโปรแกรมสามารถทำการดึงข้อมูลที่ตั้งนี้มาจัดเก็บได้ และแสดงผลทางหน้าจอตามรูปที่ 6.5

6.2.3 ผลการทำงานในการรวบรวมข้อมูลสาขาวิชาของสถาบันการศึกษา

ในการรวบรวมข้อมูลสาขาวิชาของสถาบันการศึกษานั้น จะแสดงผลทางหน้าจอส่วนที่ติดต่อกับผู้ใช้ผ่านทาง Web Browser เช่นเดียวกับข้อมูลที่ตั้งสถาบันการศึกษา โดยจะแสดงตัวอย่างข้อมูลสาขาวิชาของสถาบันการศึกษาที่ปรากฏในเว็บเพจ และผลของข้อมูลที่จัดเก็บได้ โดยโปรแกรม ดังรูปที่ 6.6 และรูปที่ 6.7



รูปที่ 6.6 ข้อมูลสาขาวิชาที่พบในเว็บเพจ



รูปที่ 6.7 ผลการจัดเก็บข้อมูลสาขาวิชาของสถาบันการศึกษา

จากรูปที่ 6.6 เป็นตัวอย่างเว็บเพจของ New York University ที่มีรายชื่อของสาขาวิชาที่เปิดสอนอยู่ ซึ่งโปรแกรมสามารถทำการดึงข้อมูลสาขาวิชานั้นมาจัดเก็บได้ และแสดงผลทางหน้าจอตามรูปที่ 6.7

6.3 การวัดประสิทธิภาพในการดึงข้อมูลมาจัดเก็บ

สำหรับในการวัดประสิทธิภาพการดึงข้อมูลนี้ จะใช้ตัววัดประสิทธิภาพของการดึงข้อมูล 2 ตัว ได้แก่

- ค่า Precision จะเป็นอัตราส่วนระหว่างจำนวนข้อมูลที่โปรแกรมสามารถจัดเก็บมาได้และตรงตามความต้องการ ต่อจำนวนข้อมูลที่จัดเก็บได้ทั้งหมด

- ค่า Recall จะเป็นอัตราส่วนระหว่างจำนวนข้อมูลที่โปรแกรมสามารถจัดเก็บมาได้และตรงตามความต้องการ ต่อจำนวนข้อมูลที่ตรงตามความต้องการทั้งหมด

ซึ่งในการคำนวณเพื่อวัดประสิทธิภาพการดึงข้อมูลนี้ จะวัดใน 2 ส่วนคือ ในส่วนของที่ตั้งของสถาบันการศึกษา และส่วนของสาขาวิชาที่สถาบันการศึกษานั้นๆเปิดสอน ดังแสดงรายละเอียดการวัดประสิทธิภาพได้ดังนี้

6.3.1 การวัดประสิทธิภาพการดึงข้อมูลที่ตั้งของสถาบันการศึกษา

ในการวัดประสิทธิภาพของการดึงข้อมูลที่ตั้งของสถาบันการศึกษา จะวัดประสิทธิภาพจากตัววัด ต่อไปนี้

- Precision เป็นอัตราส่วนข้อมูลที่ตั้งสถาบันการศึกษาที่จัดเก็บได้ และอยู่ภายใต้เงื่อนไขที่กำหนด ต่อ ข้อมูลที่โปรแกรมจัดเก็บมาได้ทั้งหมดในส่วนของที่ตั้งสถาบันการศึกษา

- Recall เป็นอัตราส่วนข้อมูลที่ตั้งสถาบันการศึกษาที่จัดเก็บได้ และอยู่ภายใต้เงื่อนไขที่กำหนด ต่อ จำนวนสถาบันการศึกษาที่มีที่ตั้งปรากฏอยู่บนเว็บเพจ

โดยเงื่อนไขในการวัดประสิทธิภาพการดึงข้อมูลที่ตั้งของสถาบันการศึกษานี้ ได้มีการใช้ฮิวริสติก (Heuristics) เพื่อช่วยในการพิจารณาจำนวนข้อมูลที่จะนำมาใช้ในการคำนวณ อีกทั้งได้มีการตั้งสมมติฐานว่า ในทุกๆสถาบันการศึกษาจะต้องมีเว็บเพจที่ระบุที่ตั้งของสถาบันการศึกษา ดังนั้น จำนวนสถาบันการศึกษาที่มีข้อมูลที่ตั้งปรากฏบนเว็บเพจจะมีค่าเท่ากับ 100 ซึ่งผลของการวัดประสิทธิภาพสำหรับแต่ละฮิวริสติก สามารถแสดงได้ ดังนี้

6.3.1.1 ผลการทดลอง

ฮิวริสติกที่ 1 เว็บเพจของสถาบันการศึกษาจะปรากฏข้อความระบุที่ตั้งที่หน้าเว็บเพจแรก

จากฮิวริสติกดังกล่าว สามารถแสดงจำนวนข้อมูลต่างๆได้ดังนี้

- จำนวนเว็บเพจของสถาบันการศึกษาที่ปรากฏข้อความระบุที่ตั้งในหน้าแรกของเว็บเพจ เท่ากับ 53
- จำนวนข้อมูลที่ตั้งที่โปรแกรมจัดเก็บได้ทั้งหมด โดยยังไม่ได้ทำการตรวจสอบว่าเป็นข้อมูลที่ตั้งจริงหรือไม่ เท่ากับ 43
- จำนวนข้อมูลที่ตั้งที่โปรแกรมจัดเก็บได้ และข้อมูลนั้นปรากฏในหน้าเพจแรกเว็บสถาบันการศึกษา เท่ากับ 33

จากข้อมูลดังกล่าว สามารถคำนวณอัตราส่วนเพื่อวัดประสิทธิภาพของการดึงข้อมูลที่ตั้งของสถาบันการศึกษาได้ ดังนี้

- Precision = 33/43
  = 0.79
- Recall = 33/100
  = 0.33

ในกรณีนี้ อัตราส่วนของข้อมูลที่ตั้งที่พบในหน้าเว็บเพจแรก ต่อจำนวนเว็บเพจของสถาบันการศึกษาทั้งหมดที่มีที่ตั้งปรากฏอยู่ในหน้าแรก(Addresses in First page Ratio - AIF) เป็นดังนี้

- AIF Ratio = 33/53

  = 0.62

<u>ฮิวริสติกที่ 2</u> เว็บเพจของสถาบันการศึกษาจะปรากฏข้อความระบุที่ตั้งที่หน้าเว็บเพจแรก หรือ เว็บเพจที่มีคำที่ใช้เป็น link ปรากฏคำว่า “Address” ”Contact” หรือ ”About”

จากฮิวริสติกดังกล่าว สามารถแสดงจำนวนข้อมูลต่างๆได้ดังนี้

- จำนวนเว็บเพจของสถาบันการศึกษาที่ปรากฏข้อความระบุที่ตั้งที่หน้าเว็บเพจแรก หรือ เว็บเพจที่มีคำที่ใช้เป็น link ปรากฏคำว่า “Address” ”Contact” หรือ ”About” เท่ากับ 84
- จำนวนข้อมูลที่ตั้งที่โปรแกรมจัดเก็บได้ โดยยังไม่ได้ทำการตรวจสอบว่าเป็นข้อมูลที่ตั้งจริงหรือไม่ เท่ากับ 43
- จำนวนข้อมูลที่ตั้งที่โปรแกรมจัดเก็บได้ และเป็นข้อมูลที่ตั้งจริง และข้อมูลนั้นปรากฏข้อความระบุที่ตั้งที่หน้าเว็บเพจแรก หรือ เว็บเพจที่มีคำที่ใช้เป็น link ปรากฏคำว่า “Address” ”Contact” หรือ ”About” เท่ากับ 42

จากข้อมูลดังกล่าว สามารถคำนวณอัตราส่วนเพื่อวัดประสิทธิภาพของการดึงข้อมูลที่ตั้งของสถาบันการศึกษาได้ ดังนี้

- Precision = 42/43

  = 0.98

- Recall = 42/100

  = 0.42

ในกรณีนี้ อัตราส่วนของข้อมูลที่ตั้งที่พบในหน้าเว็บเพจแรกหรือ เว็บเพจที่มีคำที่ใช้เป็น link ปรากฏคำว่า “Address” ”Contact” หรือ ”About” ต่อจำนวนเว็บเพจของสถาบันการศึกษาทั้งหมดที่มีที่ตั้งปรากฏอยู่ในหน้าแรกหรือ เว็บเพจที่มีคำที่ใช้เป็น link ปรากฏคำว่า “Address” ”Contact” หรือ ”About” (Addresses in Linked Page Ratio - AIL) เป็นดังนี้

- AIL Ratio = 42/84

  = 0.50

6.3.1.2 วิเคราะห์ผลการทดลอง

จากการทดลองของทั้ง 2 ฮิวริสติกข้างต้น สามารถวิเคราะห์ผลการทดลองได้ดังนี้

- ค่า Precision ที่ได้จากของทั้ง 2 ฮิวริสติก จะเห็นว่า ในฮิวริสติกแรกนั้น เงื่อนไขการพบข้อมูลที่ตั้งของสถาบันการศึกษานั้นมีเพียงการพบที่เว็บเพจแรกเท่านั้น ซึ่งทำให้จำนวนของข้อมูลที่จัดเก็บได้และเป็นข้อมูลที่ปรากฏในหน้าเว็บเพจแรกนั้นมีค่าเป็น 33 จากจำนวนข้อมูลที่จัดเก็บไว้ทั้งหมดจำนวน 43 เว็บเพจ ทำให้ค่า Precision ที่ได้จากฮิวริสติกแรกมีค่าเป็น 0.79 แต่จากฮิวริสติกที่ 2 นั้นได้เพิ่มเงื่อนไขการพบข้อมูลที่ตั้งสถาบันการศึกษามากขึ้น คือ นอกเหนือจากที่ปรากฏในหน้าแรกของเว็บเพจแล้วนั้น ยังพบในเว็บเพจซึ่งมาจาก link ที่ปรากฏคำว่า "Address" "Contact" หรือ "About" ซึ่งทำให้จำนวนของข้อมูลที่จัดเก็บได้และตรงตามเงื่อนไขมีค่าเพิ่มเป็น 42 และทำให้ค่า Precision เพิ่มเป็น 0.98 นั่นแสดงว่า ในการดึงข้อมูลโดยใช้โปรแกรมที่พัฒนาขึ้นภายใต้เงื่อนไขตามฮิวริสติกที่ 2 นั้น ข้อมูลที่จัดเก็บได้โดยส่วนใหญ่เป็นข้อมูลซึ่งระบุที่ตั้งของสถาบันการศึกษา

- ค่า Recall ที่ได้จากทั้ง 2 ฮิวริสติก จะเห็นว่า กรณีของฮิวริสติกที่ 1 ที่มีการกำหนดเงื่อนไขการปรากฏของข้อมูลที่ตั้งเพียงในหน้าแรกนั้น โปรแกรมสามารถดึงมาจัดเก็บได้ตามเงื่อนไข มี 33 เว็บเพจ และกรณีของฮิวริสติกที่ 2 ที่มีการกำหนดเงื่อนไขการที่ปรากฏของข้อมูลที่ตั้งในหน้าแรกของเว็บเพจและในเว็บเพจซึ่งมาจาก link ที่ปรากฏคำว่า "Address" "Contact" หรือ "About" จะเห็นว่าจำนวนข้อมูลที่ตั้งจะเพิ่มเป็น 42 ซึ่งค่า Recall ที่ได้นั้น จะเป็นอัตราส่วนระหว่างข้อมูลที่ตั้งที่ตรงตามเงื่อนไข ต่อจำนวนสถาบันการศึกษาที่มีข้อมูลที่ตั้งปรากฏบนเว็บเพจจะมีค่าเท่ากับ 100 ตามที่ได้ตั้งสมมติฐานไว้ ซึ่งจะได้ค่า Recall ของฮิวริสติกที่ 1 เท่ากับ 0.33 และค่า Recall ของฮิวริสติกที่ 2 เท่ากับ 0.42 นั่นคือ หากมีการกำหนดเงื่อนไขในการปรากฏของข้อมูลที่ตั้งมากขึ้น จะมีโอกาสในการพบข้อมูลที่ตั้งของสถาบันการศึกษามากขึ้น

- อัตราส่วนของข้อมูลที่พบ ต่อข้อมูลตามเงื่อนไขในฮิวริสติกซึ่งได้แก่ อัตราส่วนของข้อมูลที่ตั้งที่พบในหน้าเว็บเพจแรก ต่อจำนวนเว็บเพจของสถาบันการศึกษาทั้งหมดที่มีที่ตั้งปรากฏอยู่ในหน้าแรก(Addresses in First page Ratio - AIF) และ อัตราส่วนของข้อมูลที่ตั้งที่พบในหน้าเว็บเพจแรกหรือ เว็บเพจที่มีคำที่ใช้เป็น link ปรากฏคำว่า "Address" "Contact" หรือ "About" ต่อจำนวนเว็บเพจของสถาบันการศึกษาทั้งหมดที่มีที่ตั้งปรากฏอยู่ในหน้าแรกหรือ เว็บเพจที่มีคำที่ใช้เป็น link ปรากฏคำว่า "Address" "Contact" หรือ "About" (Addresses in Linked Page Ratio - AIL) นั้น จะเห็นว่า กรณีของฮิวริสติกที่ 1 ที่มีการกำหนดเงื่อนไขการปรากฏของข้อมูลที่ตั้งเพียงในหน้าแรกนั้น สถาบันการศึกษาที่มีเว็บเพจหน้าแรกปรากฏที่ตั้งจะมีอยู่เพียง 53 เว็บเพจ และ

ที่โปรแกรมสามารถดึงมาจัดเก็บได้ตามเงื่อนไขในฮิวริสติกที่ 1 มี 33 เว็บเพจ ทำให้ค่า AIF Ratio ที่ได้มีค่าเป็น 0.62 แต่หากใช้เงื่อนไขตามฮิวริสติกที่ 2 จะเห็นว่าจำนวนของเว็บเพจที่ปรากฏข้อมูลที่ตั้งจะเพิ่มเป็น 84 แต่จำนวนข้อมูลที่จัดเก็บได้และตรงตามเงื่อนไขมีเพียง 42 เว็บเพจ ทำให้ค่า AIL Ratio ที่ได้ มีค่าเป็น 0.50 แม้ว่าจำนวนข้อมูลที่สามารถจัดเก็บได้จริงจะมีมากขึ้นก็ตาม นั่นคือ หากมีการกำหนดเงื่อนไขในการปรากฏของข้อมูลที่ตั้งมากขึ้น จำนวนของข้อมูลที่ตั้งของสถาบันการศึกษาที่ปรากฏตามหน้าเว็บเพจก็จะมีมากขึ้น แต่โปรแกรมที่ทำการจัดเก็บอาจจะไม่สามารถจัดเก็บมาได้หมด ทั้งนี้อาจขึ้นอยู่กับ ความลึกของหน้าเว็บเพจที่ปรากฏข้อมูลที่ตั้ง อยู่ในระดับที่ลึกเกินไป ทำให้โปรแกรม Web Crawler ยังไม่สามารถดึงข้อมูลส่วนนั้นออกมาได้ ซึ่งต้องใช้เวลาเพื่อให้โปรแกรมท่องไปยังเว็บเพจต่างๆจนถึงหน้าเว็บเพจนั้น หรืออาจเกิดจากการค่าน้ำหนักของข้อความน้อยกว่าที่กำหนดไว้ ทำให้ไม่สามารถจัดเก็บลงฐานข้อมูลได้

6.3.2 การวัดประสิทธิภาพการดึงข้อมูลสาขาวิชาของสถาบันการศึกษา

ในการวัดประสิทธิภาพของการดึงข้อมูลสาขาวิชาของสถาบันการศึกษานี้ จะพิจารณาข้อมูลในส่วนที่เป็นเว็บเพจที่ปรากฏรายชื่อสาขาวิชาของสถาบันการศึกษา โดยโปรแกรมในส่วนการดึงสาขาวิชาจะทำการท่องไปยังเว็บเพจที่ใช้ข้อความเป็น link ปรากฏคำว่า ซึ่งได้แก่คำว่า “Course” “ Program” “Academic” “Department” “Faculty” “Division” และ “School” ซึ่งการรวบรวมเว็บเพจที่มีความเป็นไปได้ว่าจะเป็นเว็บเพจที่แสดงรายชื่อสาขาวิชาของสถาบันการศึกษาโดยโปรแกรมส่วนการดึงสาขาวิชา ได้เว็บเพจจำนวน 285 เว็บเพจ และจากการตรวจสอบในแต่ละเว็บเพจ พบว่า เว็บเพจที่ปรากฏรายชื่อสาขาวิชา เป็นจำนวน 117 เว็บเพจ แต่ในจำนวนนี้จะมีบางเว็บเพจที่ปรากฏชื่อสาขาวิชา แต่เว็บเพจนั้นไม่ใช่หน้าเว็บเพจหลักที่จะแสดงรายชื่อสาขาวิชา ซึ่งอาจจะปรากฏสาขาวิชาในรูปของเมนู หรืออยู่ภายใต้ combo box บนหน้าเว็บเพจ ทั้งนี้ จากการตรวจสอบพบเว็บเพจที่เป็นหน้าหลักในการแสดงรายชื่อของ major จำนวน 98 เว็บเพจ แต่เนื่องจากเว็บเพจระบุสาขาวิชาในแต่ละสถาบันการศึกษาสามารถปรากฏได้หลายเว็บเพจ ดังนั้นจึงทำการนับจำนวนของสถาบันการศึกษาที่โปรแกรมดึงข้อมูลส่วนสาขาวิชาจัดเก็บมาได้ทั้งหมด ซึ่งนับได้ 52 แห่ง จาก 285 เว็บเพจ นอกจากนี้ ได้มีการตั้งสมมติฐานว่า ในทุกๆสถาบันการศึกษาจะต้องมีเว็บเพจที่ระบุสาขาวิชาที่เปิดสอนอย่างน้อย 1 เว็บเพจ ดังนั้น จำนวนสถาบันการศึกษาที่มีเว็บเพจระบุสาขาวิชาอย่างน้อย 1 เว็บเพจ จะมีค่าเท่ากับ 100 ทั้งนี้ ในการวัดประสิทธิภาพการดึงข้อมูลสาขาวิชาของสถาบันการศึกษา ได้นิยามตัววัดประสิทธิภาพการดึงข้อมูลสาขาวิชา ดังนี้

- Precision เป็นอัตราส่วนของจำนวนสถาบันการศึกษาที่มีเว็บเพจระบุสาขาวิชาอย่างน้อย 1 เว็บเพจ และโปรแกรมสามารถจัดเก็บได้ ต่อ จำนวนสถาบันการศึกษาที่โปรแกรมดึงข้อมูลส่วนสาขาวิชาสามารถจัดเก็บมาได้

- Recall เป็นอัตราส่วนของจำนวนสถาบันการศึกษาที่มีเว็บเพจระบุสาขาวิชาอย่างน้อย 1 เว็บเพจ และโปรแกรมสามารถจัดเก็บได้ ต่อ จำนวนสถาบันการศึกษาทั้งหมดที่มีเว็บเพจระบุสาขาวิชาอย่างน้อย 1 เว็บเพจ

โดยผลการวัดประสิทธิภาพการดึงข้อมูลสาขาวิชาของสถาบันการศึกษา สามารถแสดงได้ ดังนี้

6.3.2.1 ผลการทดลอง

จากการทำงานของโปรแกรมเพื่อดึงข้อมูลสาขาวิชาที่เปิดสอนในแต่ละสถาบันการศึกษา แสดงข้อมูลได้ดังนี้

- จำนวนสถาบันการศึกษาที่มีเว็บเพจระบุสาขาวิชาอย่างน้อย 1 เว็บเพจ และโปรแกรมสามารถจัดเก็บได้ เท่ากับ 36
- จำนวนสถาบันการศึกษาที่โปรแกรมส่วนจัดเก็บสาขาวิชาจัดเก็บมาได้ซึ่งในจำนวนนี้อาจจะไม่ปรากฏเว็บเพจหลักที่ระบุสาขาวิชา ซึ่งนับได้เท่ากับ 52

จากข้อมูลดังกล่าว สามารถคำนวณคำนวณอัตราส่วนเพื่อวัดประสิทธิภาพของการดึงข้อมูลสาขาวิชาของสถาบันการศึกษาได้ ดังนี้

- Precision = 36/52
  = 0.69
- Recall = 36/100
  = 0.36

6.3.2.2 วิเคราะห์ผลการทดลอง

จากการวัดประสิทธิภาพในการดึงข้อมูลสาขาวิชาของสถาบันการศึกษา จะเห็นว่า จากการที่โปรแกรมทำการดึงข้อมูลสาขาวิชาของสถาบันการศึกษา พบสถาบันการศึกษาที่มีเว็บเพจสาขาวิชาที่จัดเก็บมาได้ เท่ากับ 36 จากสถาบันการศึกษาที่เก็บได้ทั้งหมด 52 แห่ง ซึ่งค่า Precision นั้นจะมีค่าเป็น 0.69 ซึ่งถือว่าโปรแกรมสามารถค้นพบเว็บเพจที่ระบุสาขาวิชา และเก็บรวบรวมข้อมูลจากเว็บเพจนั้นได้ในระดับหนึ่ง และจากสมมติฐานที่ว่า ในทุกๆสถาบันการศึกษาจะต้องมีเว็บเพจที่ระบุสาขาวิชาที่เปิดสอนอย่างน้อย 1 เว็บเพจ ซึ่งมีค่าเป็น 100 ทำให้ค่า Recall ที่ได้ มีค่าเป็น 0.36 จะเห็นได้ว่า อัตราส่วนที่ได้นั้น มีค่ายังไม่มาก ทั้งนี้ อาจเกิดจากการที่ Web

Crawler ที่ท่องเว็บเพจแต่ละสถาบันการศึกษา ยังไม่พบเว็บเพจที่ระบุสาขาวิชาของสถาบันการศึกษานั้นๆ อาจเพราะบางสถาบันมีการแสดงเว็บเพจระบุสาขาวิชาไว้ในระดับที่ลึกมาก ซึ่งต้องใช้เวลาในการทำงานของโปรแกรมที่มากขึ้น อีกทั้งอาจเกิดจากการที่สถาบันการศึกษานั้นใช้ข้อความที่เป็น link นอกเหนือจากที่กำหนดในเงื่อนไข ทำให้ไม่สามารถเก็บข้อมูลนั้นมาได้

6.4 ข้อจำกัดในการทำงานของระบบ

ในการทำงานของระบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษานั้นมีข้อจำกัดในการทำงาน ดังนี้

1. ในการค้นหาและดึงข้อความระบุที่ตั้งและสาขาวิชาของสถาบันการศึกษา โปรแกรม Web Crawler จะทำการประมวลผลเอกสาร HTML โดยการเปรียบเทียบข้อความที่เป็นภาษาอังกฤษเท่านั้น ซึ่งหากเว็บเพจของบางสถาบันการศึกษามีการใช้ภาษาอื่น เช่น ภาษาญี่ปุ่น หรือ ภาษาฝรั่งเศส ระบบจะไม่สามารถทำการเปรียบเทียบข้อความเพื่อดึงข้อมูลในส่วนที่ต้องการออกมาได้

2. การเริ่มต้นทำงานของโปรแกรม Web Crawler จะต้องทำการระบุเว็บเพจเริ่มต้นให้กับตัวโปรแกรม ซึ่งเป็นเว็บเพจที่ได้รวบรวมรายชื่อสถาบันการศึกษาต่างๆไว้ ซึ่งหากมีการใช้เว็บเพจเริ่มต้นที่ไม่ได้มีการรวบรวมสถาบันการศึกษาไว้ จะส่งผลให้การรวบรวมข้อมูลทำได้ไม่ดี

3. ในการค้นหาและดึงข้อมูลระบุสาขาวิชาที่เปิดสอนของสถาบันการศึกษา ได้มีการเปรียบเทียบข้อความที่เหมือนหรือเป็นส่วนหนึ่งของชื่อสาขาวิชาที่ได้จัดเก็บไว้ในฐานข้อมูล ซึ่งหากมีกรณีที่เว็บเพจของสถาบันการศึกษามีการอ้างถึงชื่อสาขาวิชา แต่สาขาวิชานั้นไม่ได้ทำการเปิดสอนในสถาบันการศึกษานั้น หากข้อความที่เป็นสาขาวิชาที่อ้างถึงนั้นเหมือนหรือมีส่วนหนึ่งที่ตรงกับรายชื่อสาขาวิชาที่จัดเก็บเอาไว้ โปรแกรมก็จะทำการดึงข้อความนั้นมาจัดเก็บในส่วนของสาขาวิชาที่เปิดสอนด้วย ซึ่งจะได้ผลลัพธ์ของข้อมูลที่ไม่ถูกต้อง

4. ในการค้นหาและดึงข้อมูลระบุที่ตั้งสถาบันการศึกษา โปรแกรมจะทำเปรียบเทียบรูปแบบของข้อความ ซึ่งในบางสถาบันการศึกษามีการแสดงข้อความระบุที่ตั้งสถาบันการศึกษาไว้ในหลายๆเว็บเพจ และอาจมีการแสดงข้อความที่ตั้งในรูปแบบที่แตกต่างกัน หรือมีการระบุที่ตั้งของหน่วยงานต่างๆภายในสถาบันการศึกษาในเว็บเพจนั้นด้วย ซึ่งโปรแกรมจะทำการจัดเก็บข้อความแรกที่มีรูปแบบของข้อความตรงกับที่กำหนดไว้เท่านั้น

5. ในการทำงานของโปรแกรมจะประมวลผลข้อความในเว็บเพจทีละเอกสาร ซึ่งจากการรวบรวมข้อมูลจากเว็บเพจจำนวน 11,814 เว็บเพจ จาก 100 สถาบันการศึกษา ใช้เวลาในการทำงานกว่า 70 ชั่วโมง ซึ่งถือว่าทำงานได้ช้า

บทที่ 7
บทสรุปและข้อเสนอแนะ

7.1 บทสรุป

การพัฒนาต้นแบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษา ซึ่งเป็นฐานข้อมูลซึ่งเก็บรวบรวมข้อมูลที่เกี่ยวกับสถาบันการศึกษาต่างๆ ซึ่งได้แก่ข้อมูลที่ตั้งของสถาบันการศึกษา และสาขาวิชาที่สถาบันการศึกษานั้นๆเปิดสอน เพื่อให้ผู้ที่มีความสนใจหรือต้องการทราบข้อมูลของสถาบันการศึกษาสามารถเรียกดูได้อย่างรวดเร็วและมีความถูกต้องแม่นยำ ในการพัฒนาระบบงานนี้ได้นำความรู้ในเรื่องของเว็บเสิร์ชเอ็นจิ้น การจัดการฐานข้อมูล การประมวลผลข้อความ มาเป็นแนวทางในการพัฒนา โดยมีการพัฒนางานใน 4 ส่วน คือ

- ส่วนการรวบรวมข้อมูล(Data Collection)
- ส่วนการดึงข้อมูล(Data Extraction)
- ส่วนจัดการฐานข้อมูล(Database Management)
- ส่วนการสืบค้นข้อมูลและการแสดงผล

โดยในการทำงานแต่ละส่วนนั้นได้มีการพัฒนาเพื่อตอบสนองต่อความต้องการของผู้ใช้ ซึ่งมีรายละเอียด ดังต่อไปนี้

- ในส่วนการรวบรวมข้อมูลและส่วนการดึงข้อมูล ได้มีการพัฒนา algorithm และเครื่องมือต่างๆเพื่อใช้ในการกำหนดขอบเขตในการรวบรวมข้อมูล และพัฒนากระบวนการเปรียบเทียบ รูปแบบ(Pattern Matching)เพื่อใช้ในการดึงข้อมูล ซึ่งทั้ง 2 ส่วนนี้ได้พัฒนาเป็นโปรแกรมด้วยภาษา Visual Basic 6.0

- ในส่วนจัดการฐานข้อมูล ได้มีการใช้ระบบจัดการฐานข้อมูล Microsoft SQL Server 2000 โดยจัดเก็บในรูปแบบฐานข้อมูลเชิงสัมพันธ์ ทำให้เกิดความสะดวกและความถูกต้องในการเพิ่มเติม แก้ไข ปรับปรุงเปลี่ยนแปลงข้อมูล

- ส่วนการสืบค้นข้อมูลและการแสดงผล ได้มีการใช้ภาษา ASP ในการพัฒนาระบบในส่วนของการแสดง ผลผ่านทาง Web Browser และได้ออกแบบหน้าจอการทำงานเพื่อให้ง่ายต่อการใช้งาน และนำข้อมูลที่จัดเก็บได้ แสดงไปยังผู้ใช้ได้อย่างรวดเร็วและถูกต้อง

ซึ่งจากการทำงานของระบบที่ได้พัฒนาขึ้นนี้ สามารถทำการค้นหาและจัดเก็บข้อมูล ซึ่งได้แก่ข้อมูลที่ตั้งของสถาบันการศึกษา และข้อมูลสาขาวิชาที่สถาบันการศึกษานั้นเปิดสอน ออกมา

จากเว็บเพจต่างๆได้ โดยดูได้จากผลการค้นหาที่แสดงให้กับผู้ใช้งาน แต่ทั้งนี้ปริมาณของข้อมูลที่จัดเก็บได้ขึ้นอยู่กับเวลาในการทำงานของโปรแกรมเพื่อใช้ในการท่องไปยังเว็บไซต์ต่าง ซึ่งหากมีระยะเวลาการทำงานของโปรแกรมที่นานขึ้น จำนวนของเว็บเพจที่มีข้อมูลที่ต้องการก็จะถูกพบมากขึ้นด้วย

7.2 ข้อเสนอแนะ

ในการพัฒนาต้นแบบฐานข้อมูลเว็บสำหรับงานบริการการศึกษานี้ยังมีข้อบกพร่องบางประการที่สามารถนำไปพัฒนาต่อเพื่อแก้ไขให้สมบูรณ์ หรือพัฒนาเพิ่มเพื่อประสิทธิภาพของระบบให้ระบบสมบูรณ์มากขึ้น ดังนี้

1. ปรับปรุง algorithm ทั้งในส่วนของการตรวจสอบ hyperlink การตรวจสอบข้อความระบุที่ตั้งสถาบันการศึกษา หรือข้อความระบุสาขาวิชา เพื่อให้โปรแกรมทำงานได้อย่างรวดเร็วยิ่งขึ้น และสามารถเก็บข้อมูลได้อย่างถูกต้องแม่นยำยิ่งขึ้น

2. เพิ่มรูปแบบของข้อความ(Regular Expression) ในการเปรียบเทียบข้อความระบุที่ตั้งของสถาบันการศึกษา เพื่อรองรับรูปแบบของข้อความระบุที่ตั้งสถาบันการศึกษาที่มีหลากหลายรูปแบบ รวมไปถึงการเพิ่มเงื่อนไขในการตรวจสอบข้อความที่ใช้เป็น link สำหรับการค้นหาข้อมูลที่ตั้งและสาขาวิชา โดยทำให้มีความครอบคลุมข้อความที่เป็นไปได้มากขึ้น

3. ในการพัฒนาระบบงานในส่วนของการเปรียบเทียบที่ตั้งสถาบันการศึกษาโดยใช้ Regular Expression นี้เห็นว่า รูปแบบของข้อความแต่ละส่วนนั้นมีความใกล้เคียงกัน เช่น ชื่อสถาบันการศึกษากับชื่อเมือง หรือเลขที่บ้านกับเบอร์โทรศัพท์ เป็นต้น ซึ่งจากการพัฒนาระบบงานได้กำหนดรูปแบบของ Regular Expression เพื่อให้ง่ายต่อการพัฒนา ซึ่งมีรูปแบบของข้อความที่ขึ้นต้นด้วยชื่อของสถาบันการศึกษาและลงท้ายข้อความด้วยเบอร์โทรศัพท์ ซึ่งในการกำหนดรูปแบบของ Regular Expression เพื่อใช้ในการพัฒนานี้ สามารถที่จะปรับปรุงให้มีรายละเอียดที่มากขึ้นได้ เพื่อนำมาเปรียบเทียบรูปแบบของข้อความให้มีความชัดเจนและถูกต้องมากยิ่งขึ้น

4. เพิ่มมิติในการดึงข้อมูล ซึ่งได้แก่ ข้อมูลคุณสมบัติผู้สมัครที่ต้องมีในการสมัครเพื่อศึกษาต่อของแต่ละสถาบันการศึกษา หรือ ข้อมูลค่าใช้จ่ายในการศึกษา เป็นต้น ทั้งนี้อาจจะสามารถนำหลักการในการเปรียบเทียบรูปแบบข้อความตามที่ได้มีการพัฒนาในส่วนของการดึงข้อมูลที่ตั้งสถาบัน การศึกษามาปรับใช้กับข้อมูลในแต่ละมิติ

5. ปรับปรุงกระบวนการ update ข้อมูลให้มีความเป็นอัตโนมัติมากขึ้น ซึ่งอาจจะทำการ update ตามช่วงเวลาที่ตั้งไว้ เป็นต้น

บรรณานุกรม

Baeza-Yates, Ricardo and Ribeiro-Neto, Berthier. 1999. **Modern Information Retrieval.** Boston: Addison-Wesley.

Chau, Michael and Chen, Hsinchun. 2003. **Personalized and Focused Web Spiders.** [Online]. Available: Http://ai.bpa.arizona.edu/~mchau/papers/WebSpiders.pdf.

Educational Testing Service. 2004. **2004-2005 Guide to the Use of Scores.** [Online]. Available: http://ftp.ets.org/pub/gre/994994.pdf.

Graduate Management Admission Council. 2004. **GMAT Schools' Guide October 2004 – December 2005 Graduate Management Admission Test.** [Online]. Available: http://www.gmac.com/gmac/theGMAT/tools/GMATScoreReportingServiceBooklet.htm.

Granade, Charles. 1985. **Graduate Programs in Engineering and Applied Sciences 1986.** 20 th ed. Princeton: Peterson's Guides.

Huang, Lan. **A Survey On Web Information Retrieval Technologies.** [Online]. Available: http://ranger.uta.edu/~alp/ix/readings/SurveyOfWebSearchEngines.pdf.

Jackson, Peter and Moulinier, Isabelle. 2002. **Natural Language Processing for Online Applications : Text Rettrieval, Extraction and Catagorization.** Philadelphia: John Benjamins.

Kyrnin, Jennifer. 2005. **Absolute and Relative Paths.** [Online]. Available : http://webdesign.about.com/od/beginningtutorials/a/aa040502a.htm.

Koster, Martijn. 1994. **About Web Robots.** [Online]. Available: http://www.robotstxt.org/wc/faq.html.

Kuchling, A.M. . **Regular Expression HOWTO.** [Online]. Available: http://www.amk.ca/python/howto/regex.

Linoff, Gordon S. and Berry, Michael J.A. . 2001. **Mining the Web : Transforming Customer Data into Customer Value.** New York: John Wiley & Sons.

Ramakrishnan, Raghu. 2003. **Database Management System.** New York: Mcgraw-Hill.

Rappaport, Avi. 1999. **Robots & Spiders & Crawlers :How Web and intranet search engines follow links to build indexes.** [Online]. Available: http://www.cs.uiowa.edu/~hshen/Robots.pdf.

Sonnenreich, Wes And Macinta, Tim. 1998. **Web developer.com Guide to search engine.** New York: Wiley Computer.

Weiss, Sholom M. . 2005. **Text Mining: predictive methods for analyzing unstructured information.** Boston: Allyn and Bacon.

ภาคผนวก ก
คู่มือติดตั้งระบบ

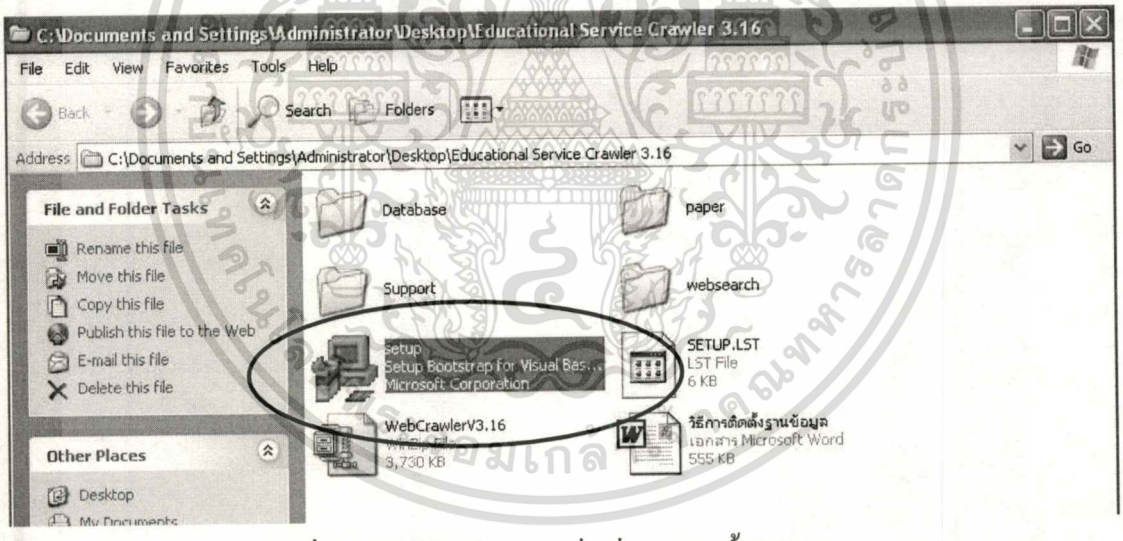
ก.1 การติดตั้งโปรแกรม Educational Service Web Crawler

ก.1.1 สิ่งที่ต้องการในการติดตั้งโปรแกรม

- โปรแกรมติดตั้ง Educational Service Web Crawler

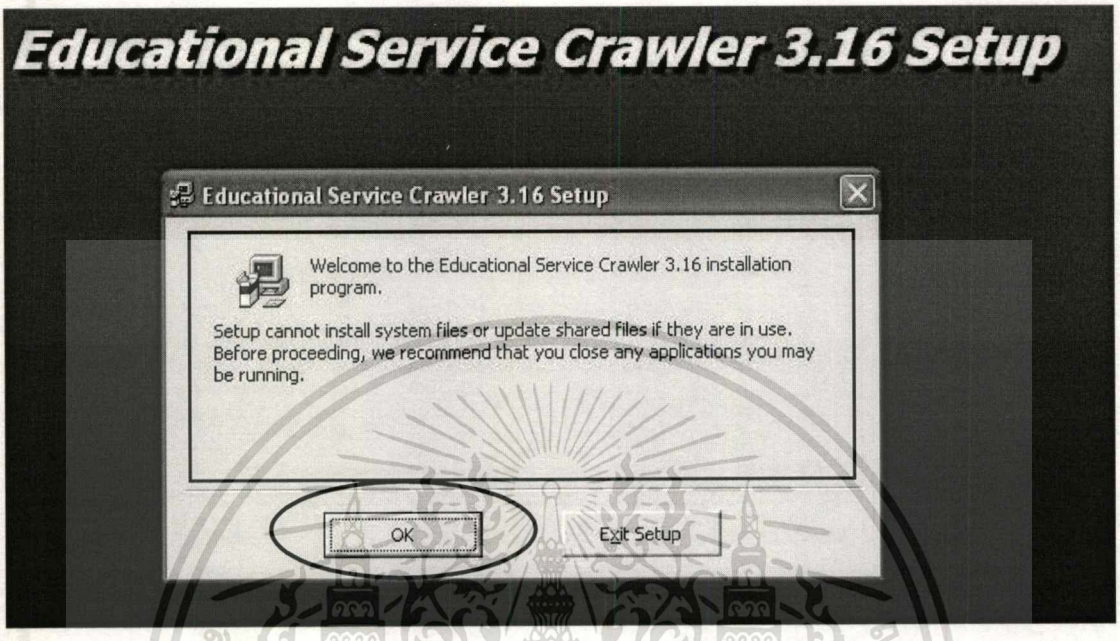
ก.1.2 ขั้นตอนการติดตั้งโปรแกรม

1. เริ่มการติดตั้งโปรแกรม โดยเลือกไฟล์ Setup ที่มีอยู่ในโฟลเดอร์ Educational Service Web Crawler ดังรูปที่ ก.1

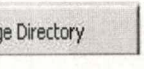


รูปที่ ก.1 เลือกไฟล์ Setup เพื่อเริ่มการติดตั้งโปรแกรม

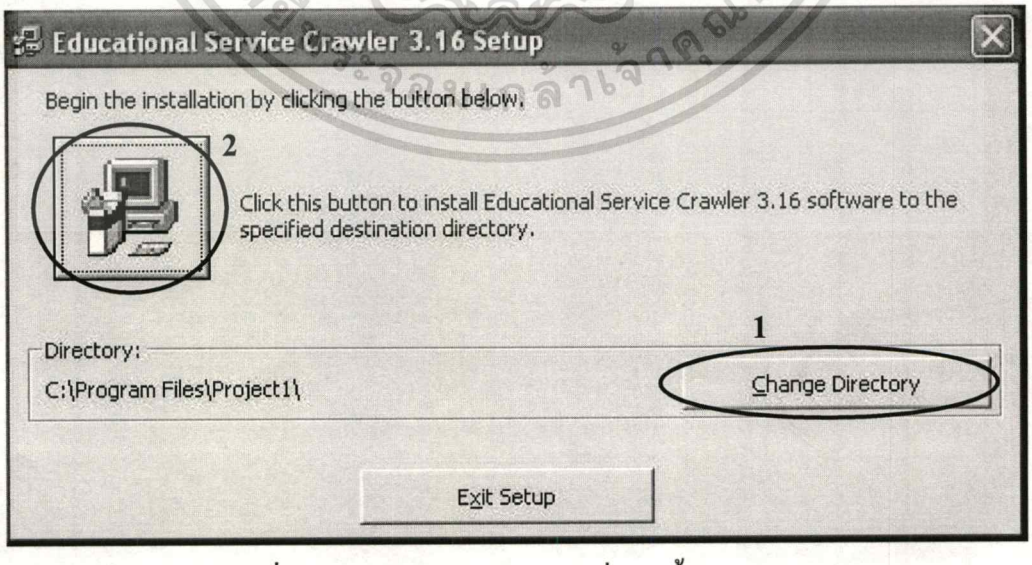
2. เมื่อเข้าสู่การติดตั้ง จะปรากฏข้อความต้อนรับ กดที่ปุ่ม OK ดังรูปที่ ก.2



รูปที่ ก.2 ข้อความต้อนรับในการติดตั้งโปรแกรม

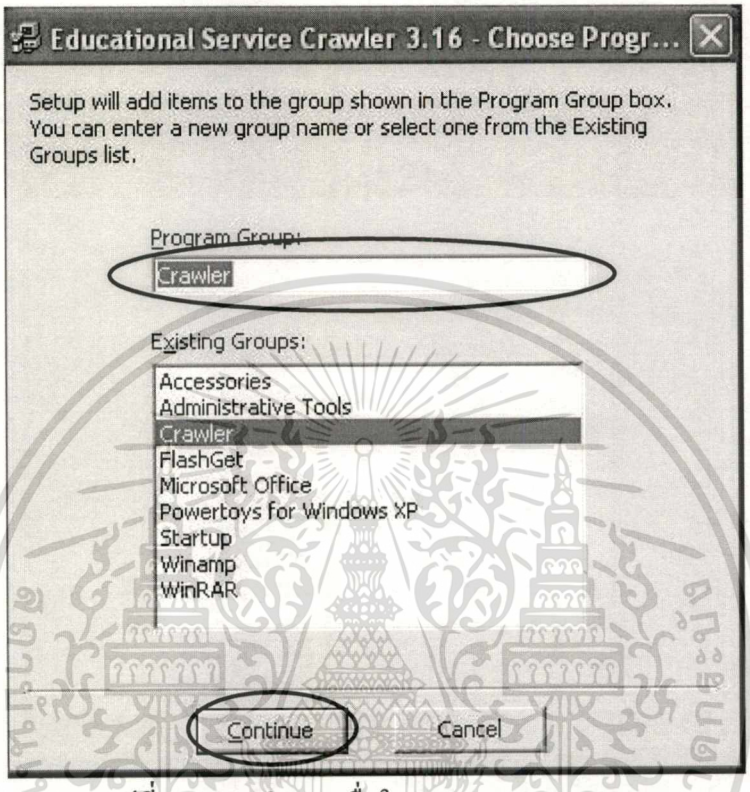
3. ทำการระบุ directory ที่จะจัดเก็บโปรแกรม โดยสามารถกดที่ปุ่ม 

จากนั้นกดปุ่ม  เพื่อเริ่มติดตั้งโปรแกรม ดังรูปที่ ก.3



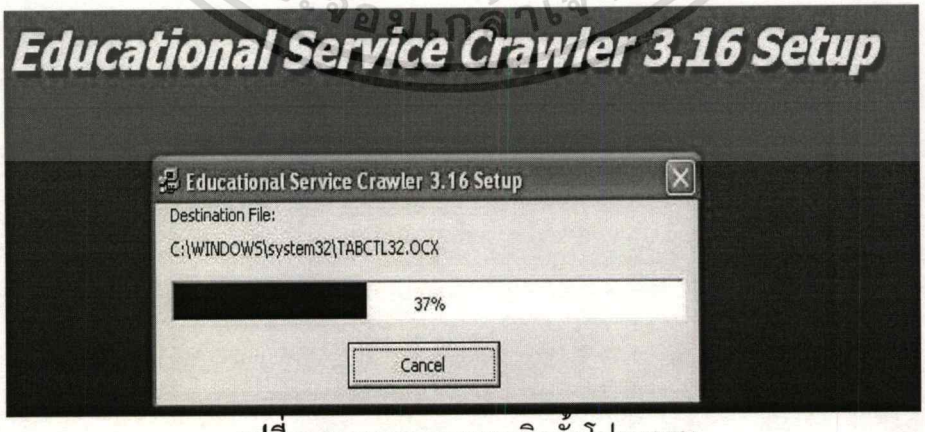
รูปที่ ก.3 การระบุ directory และเริ่มติดตั้งโปรแกรม

4. กำหนดชื่อใน Program Group Box โดยสามารถเปลี่ยนชื่อได้ จากนั้นกดที่ปุ่ม Continue ดังรูปที่ ก.4



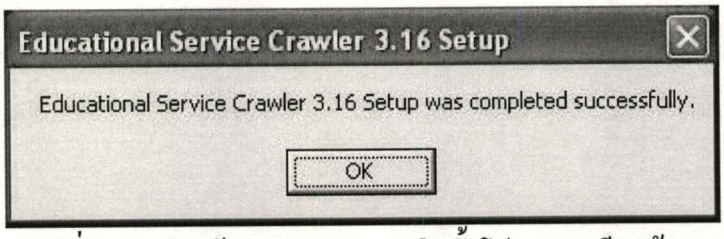
รูปที่ ก.4 การกำหนดชื่อใน Program Group Box

5. ขณะที่ทำการติดตั้งโปรแกรมจะแสดงแถบสถานะการติดตั้งโปรแกรม ดังรูปที่ ก.5



รูปที่ ก.5 แถบสถานะการติดตั้งโปรแกรม

6. เมื่อโปรแกรมติดตั้งเรียบร้อยจะแสดงกล่องข้อความ ดังรูปที่ ก.6



รูปที่ ก.6 กล่องข้อความแสดงการติดตั้งโปรแกรมเรียบร้อย

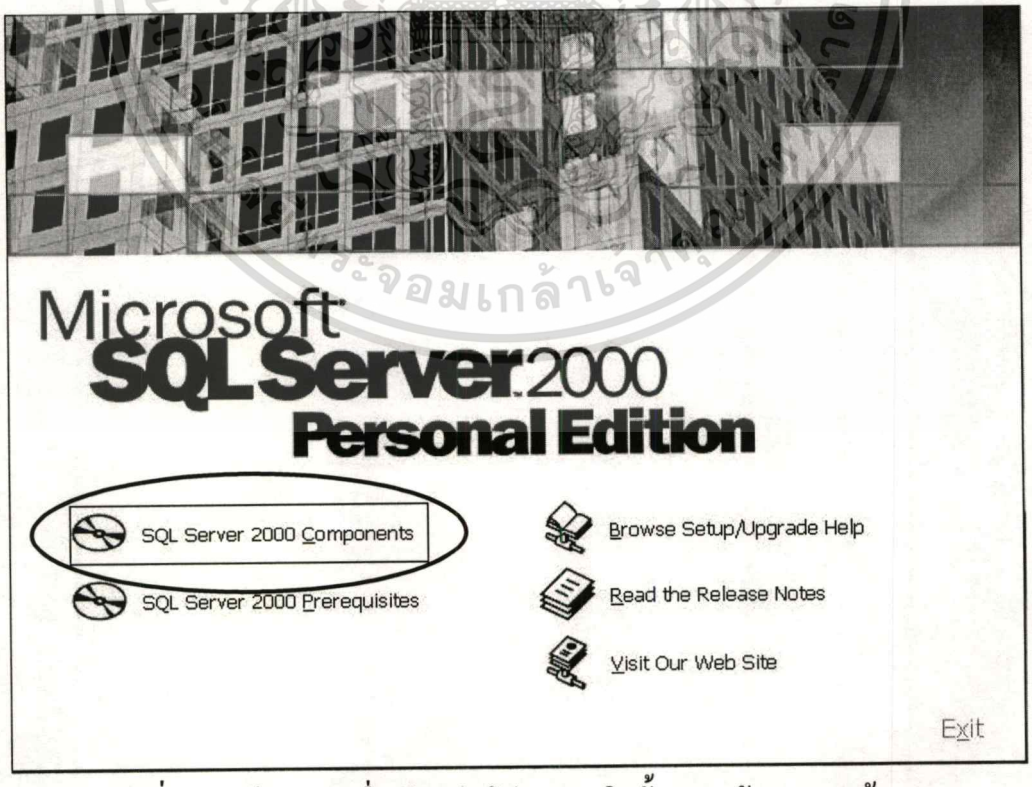
ก.2 การติดตั้งระบบจัดการฐานข้อมูล Microsoft SQL Server 2000

ก.2.1 สิ่งที่ต้องการในการติดตั้ง

- แผ่นโปรแกรมระบบจัดการฐานข้อมูล Microsoft SQL Server 2000 Personal Edition

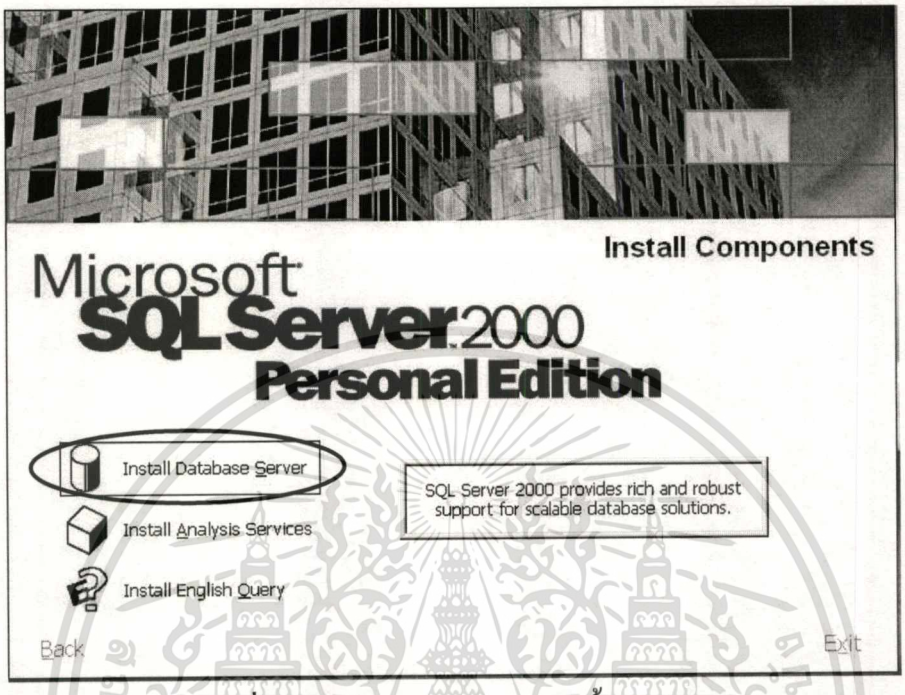
ก.2.2 ขั้นตอนการติดตั้ง

1. เมื่อเปิดแผ่นโปรแกรมติดตั้งระบบจัดการฐานข้อมูล จะขึ้นหน้าจอเมนูดังรูป ก.7 ทำการเลือกหัวข้อ SQL Server 2000 Components



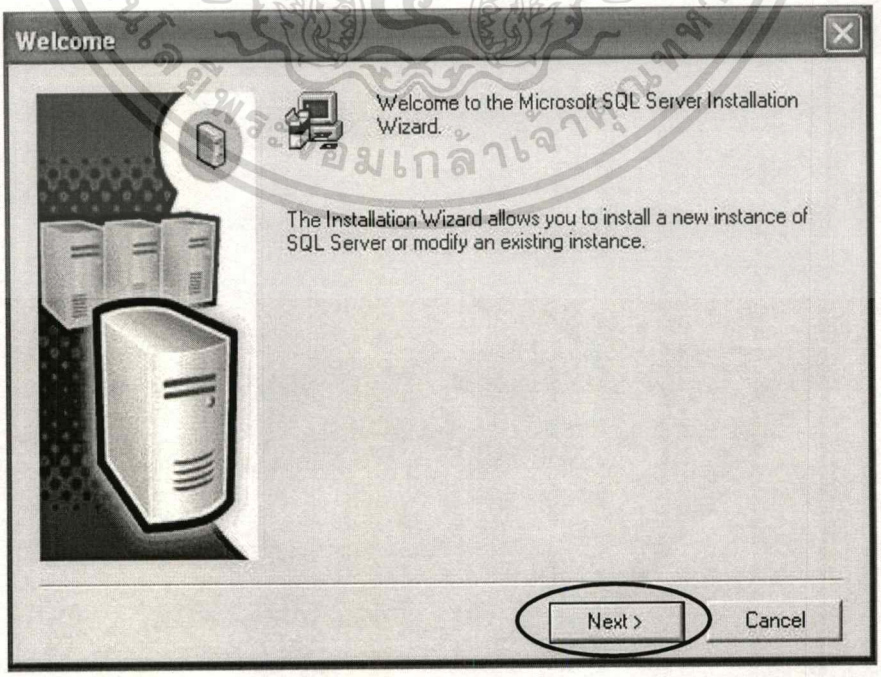
รูปที่ ก.7 หน้าจอเมนูเมื่อเปิดแผ่นโปรแกรมติดตั้งระบบจัดการฐานข้อมูล

2. เลือก Install Database Server ดังรูปที่ ก.8



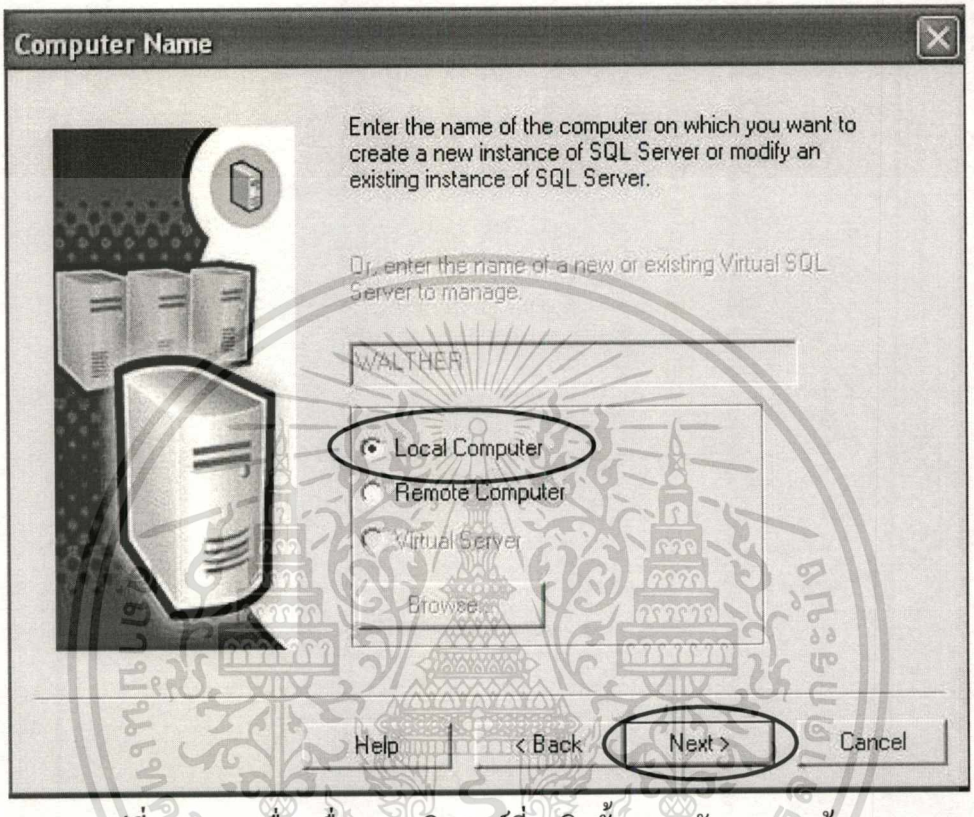
รูปที่ ก.8 หน้าจอเมนูเลือกการติดตั้งฐานข้อมูล

3. จะปรากฏกล่องข้อความต้อนรับ กดปุ่ม Next > ดังรูปที่ ก.9



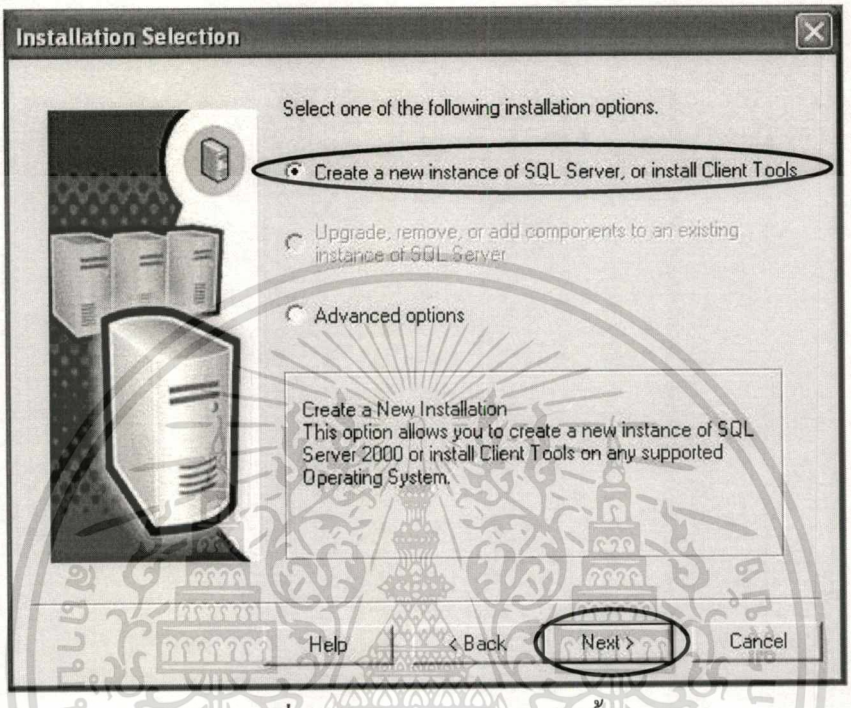
รูปที่ ก.9 กล่องข้อความต้อนรับในการติดตั้งระบบจัดการฐานข้อมูล

4. ระบุชื่อเครื่องคอมพิวเตอร์ที่จะติดตั้งระบบจัดการฐานข้อมูล ในที่นี้เลือก Local System แล้วกดปุ่ม Next > ดังรูปที่ ก.10



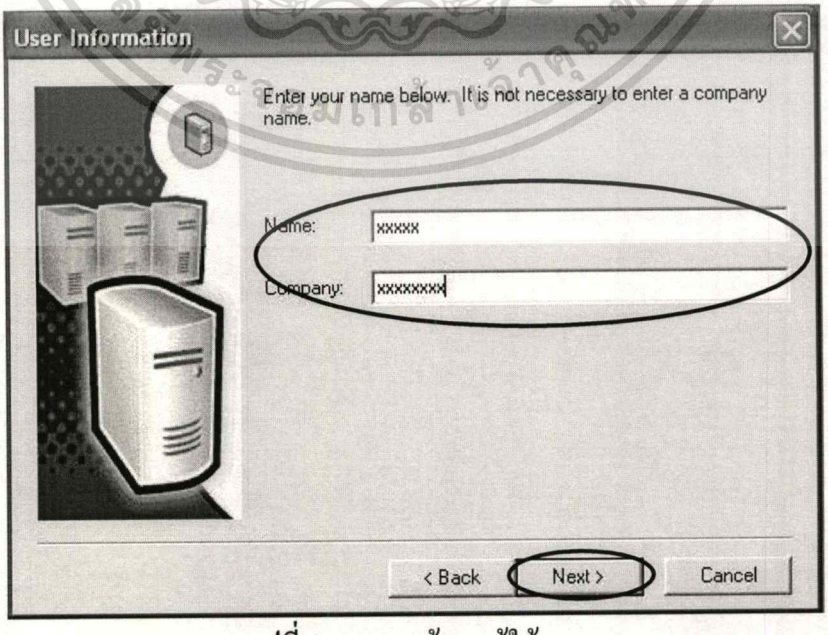
รูปที่ ก.10 ระบุชื่อเครื่องคอมพิวเตอร์ที่จะติดตั้งระบบจัดการฐานข้อมูล

5. เลือกรูปแบบในการติดตั้ง โดยในที่นี้เลือกเป็น Create a new instance of SQL Server, or install Client Tools จากนั้นกดปุ่ม Next > ดังรูปที่ ก.11



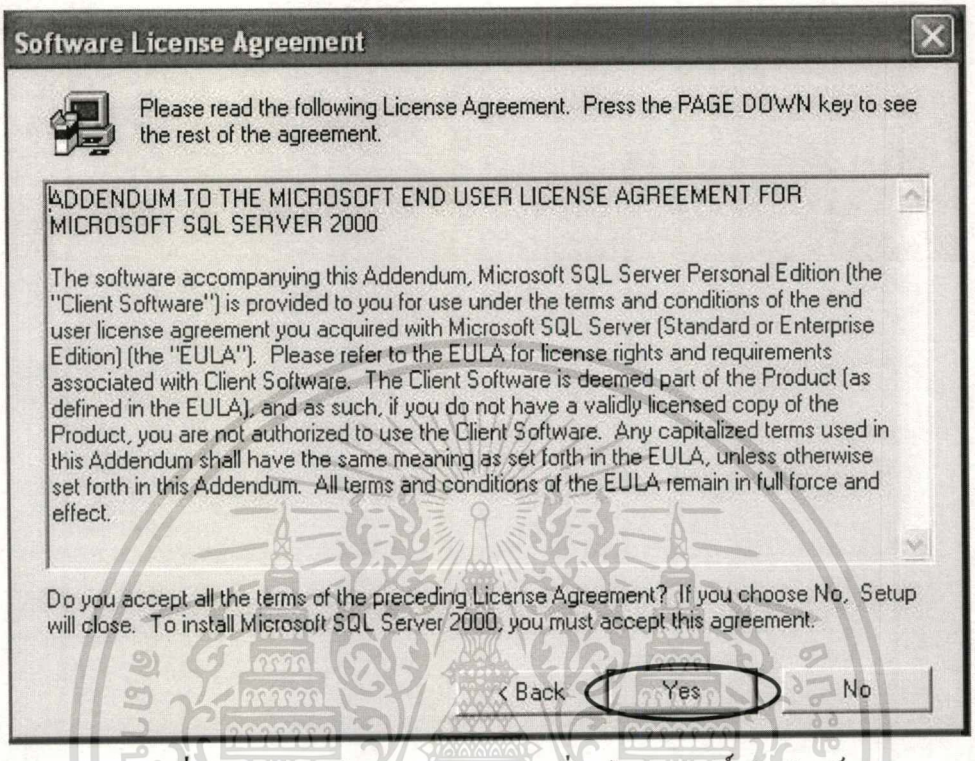
รูปที่ ก.11 เลือกรูปแบบการติดตั้ง

6. ระบุข้อมูลผู้ใช้ จากนั้นกดที่ปุ่ม Next > ดังรูปที่ ก.12



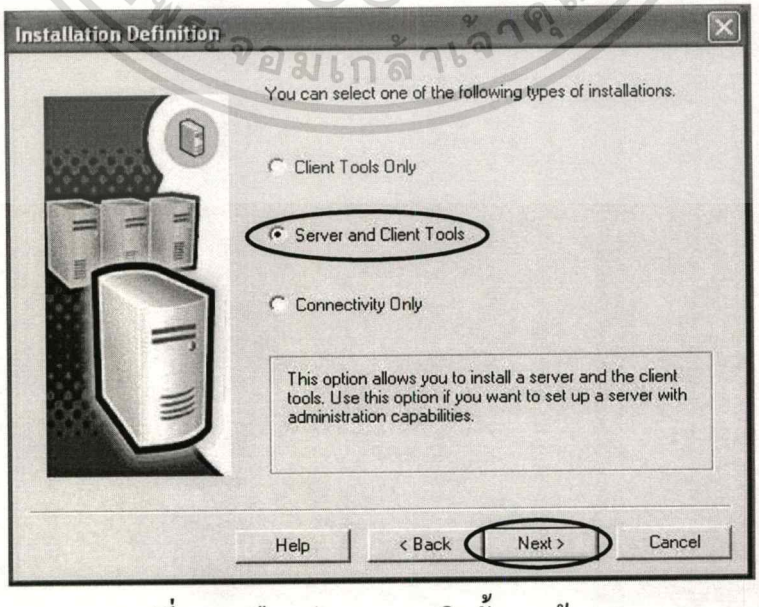
รูปที่ ก.12 ระบุข้อมูลผู้ใช้

7. จะปรากฏกล่องข้อความการยอมรับเงื่อนไขลิขสิทธิ์ผลิตภัณฑ์ กดปุ่ม Yes ดังรูปที่ ก.13



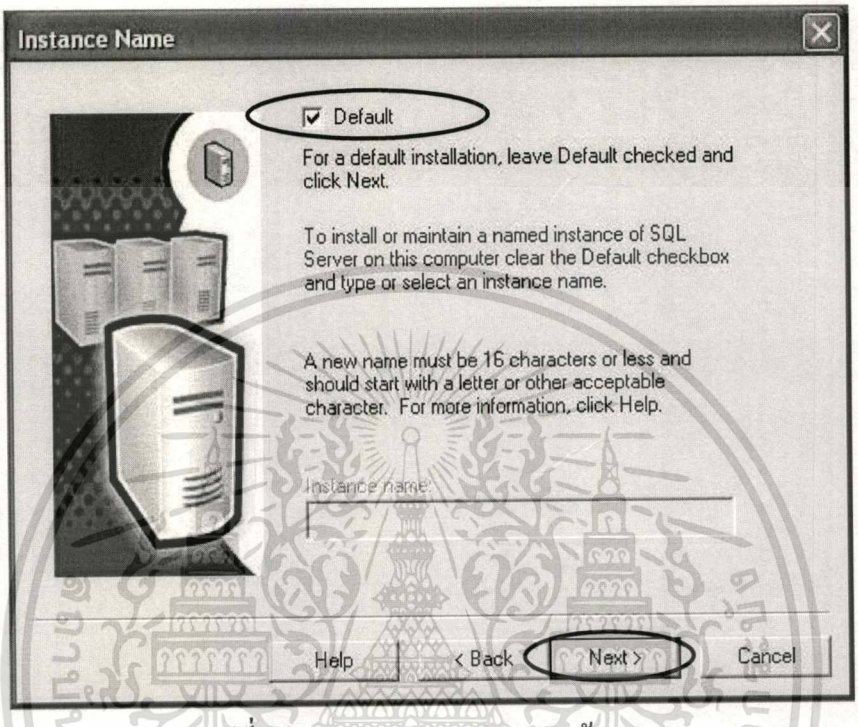
รูปที่ ก.13 กล่องข้อความการยอมรับเงื่อนไขลิขสิทธิ์ผลิตภัณฑ์

8. เลือกประเภทของการติดตั้ง ในที่นี้เลือกเป็น Server and Client Tools แล้วกดปุ่ม Next > ดังรูปที่ ก.14



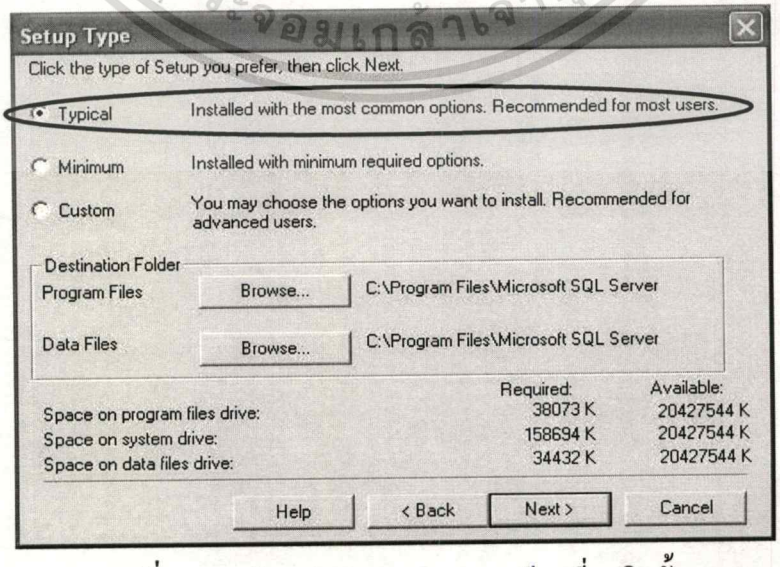
รูปที่ ก.14 เลือกประเภทการติดตั้งฐานข้อมูล

9. ระบุชื่อ Instance ของฐานข้อมูล ในที่นี้เลือกที่ Default แล้วกดปุ่ม Next > ดังรูปที่ ก.15



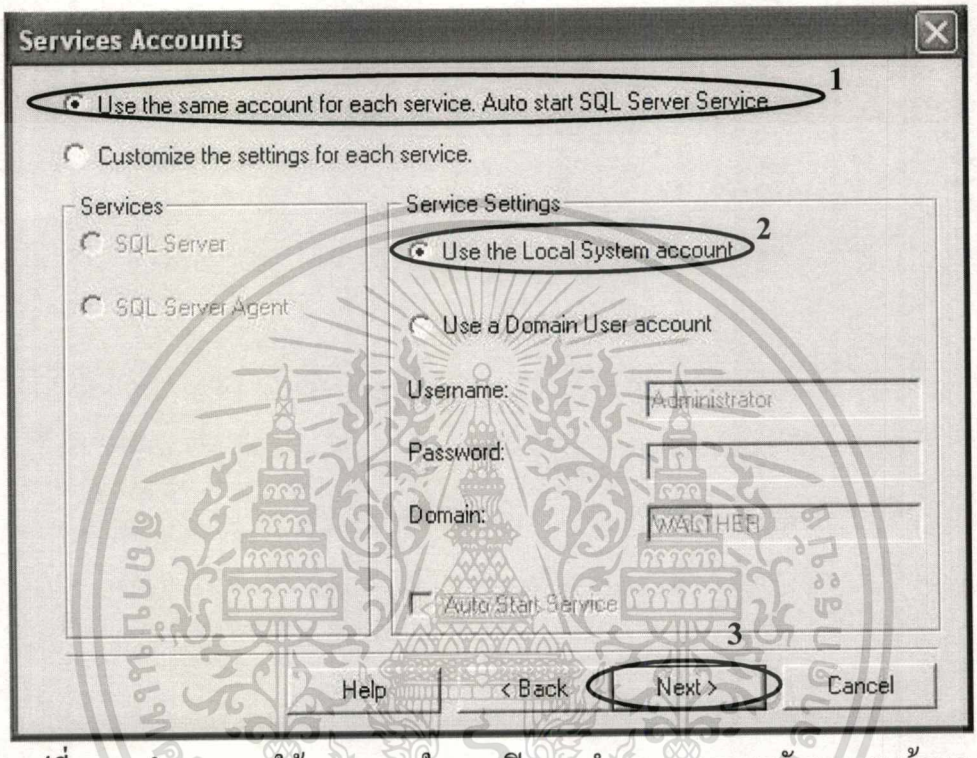
รูปที่ ก.15 ระบุ Instance ของฐานข้อมูล

10. เลือกประเภทของรายละเอียดที่จะติดตั้ง โดยเลือกเป็นแบบ Typical และกดปุ่ม Next > ดังรูปที่ ก.16



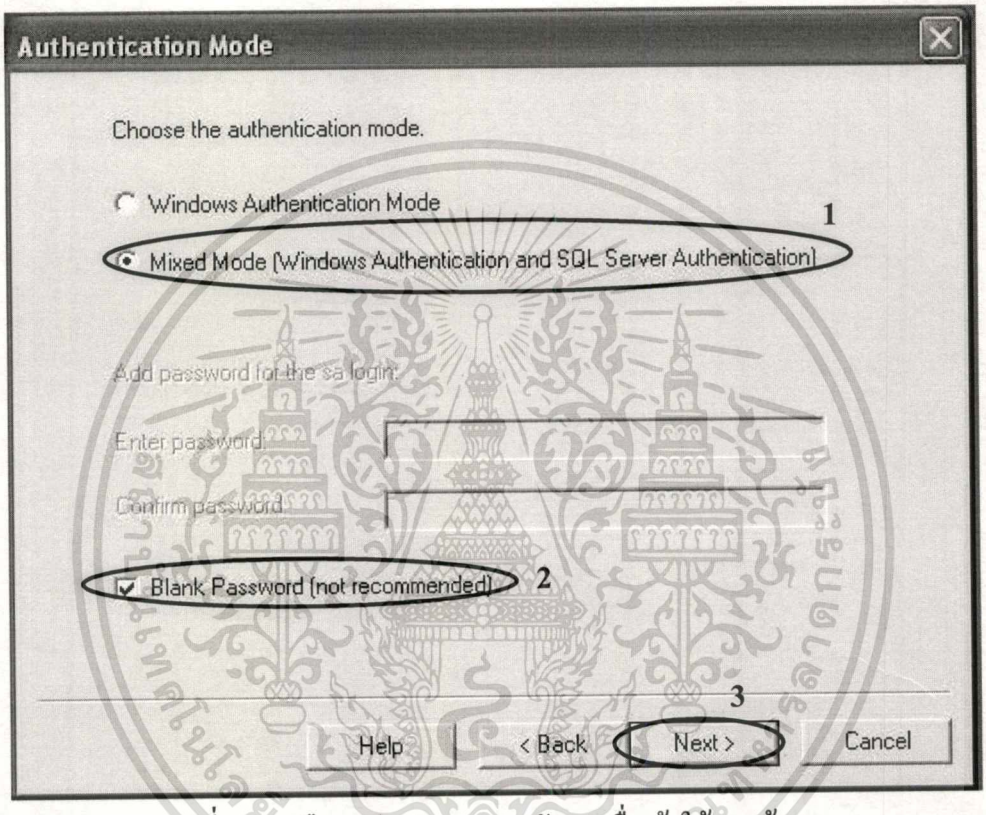
รูปที่ ก.16 เลือกประเภทของรายละเอียดที่จะติดตั้ง

11. กำหนดการใช้ Account เพื่อเปิดการทำงานของระบบจัดการฐานข้อมูล ในที่นี้เลือก Use the same account for each service. Auto start SQL Server Service และเลือก Use the Local System account จากนั้นกดปุ่ม Next > ดังรูปที่ ก.17



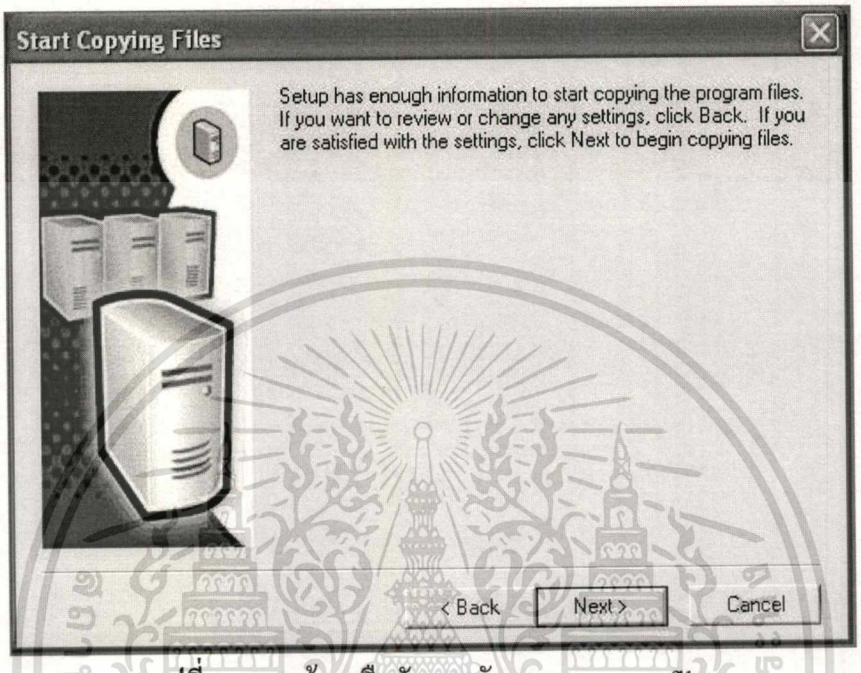
รูปที่ ก.17 กำหนดการใช้ Account ในการเปิดการทำงานของระบบจัดการฐานข้อมูล

12. เลือกรูปแบบการระบุตัวตนเพื่อเข้าใช้ฐานข้อมูล ซึ่งในที่นี้เลือก Mixed Mode (Windows Authentication and SQL Server Authentication) จากนั้นทำการกำหนด password ให้กับ username “sa” ซึ่งเป็น username ของระบบจัดการฐานข้อมูล วิ่งในที่นี้เลือกเป็น Blank Password จากนั้นกดปุ่ม Next > ดังรูป ก.18



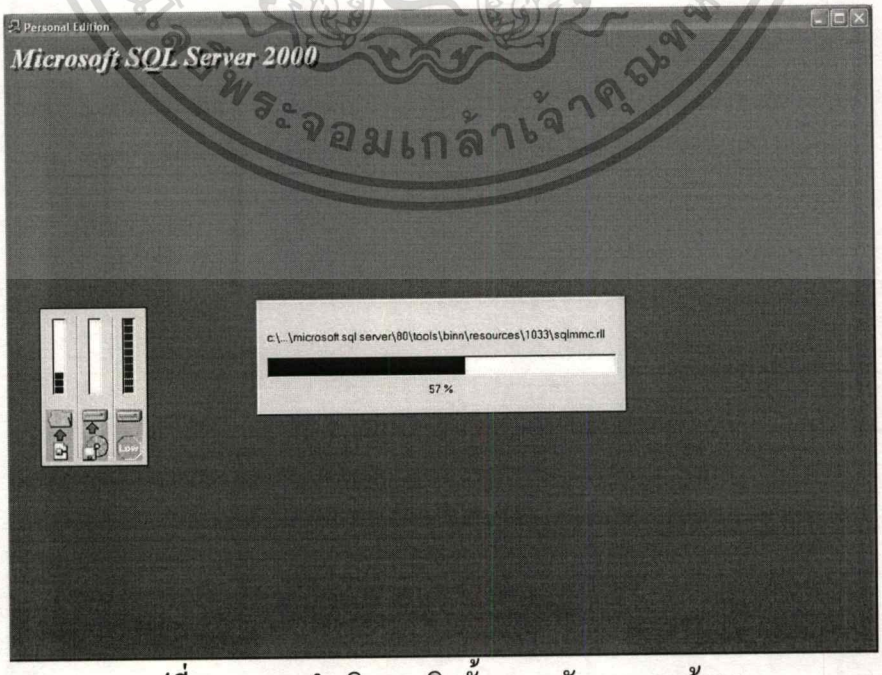
รูปที่ ก.18 เลือกรูปแบบการระบุตัวตนเพื่อเข้าใช้ฐานข้อมูล

13. จะปรากฏหน้าจอเพื่อยืนยันการคัดลอก program files โดยกดปุ่ม Next > ดังรูปที่ ก.19



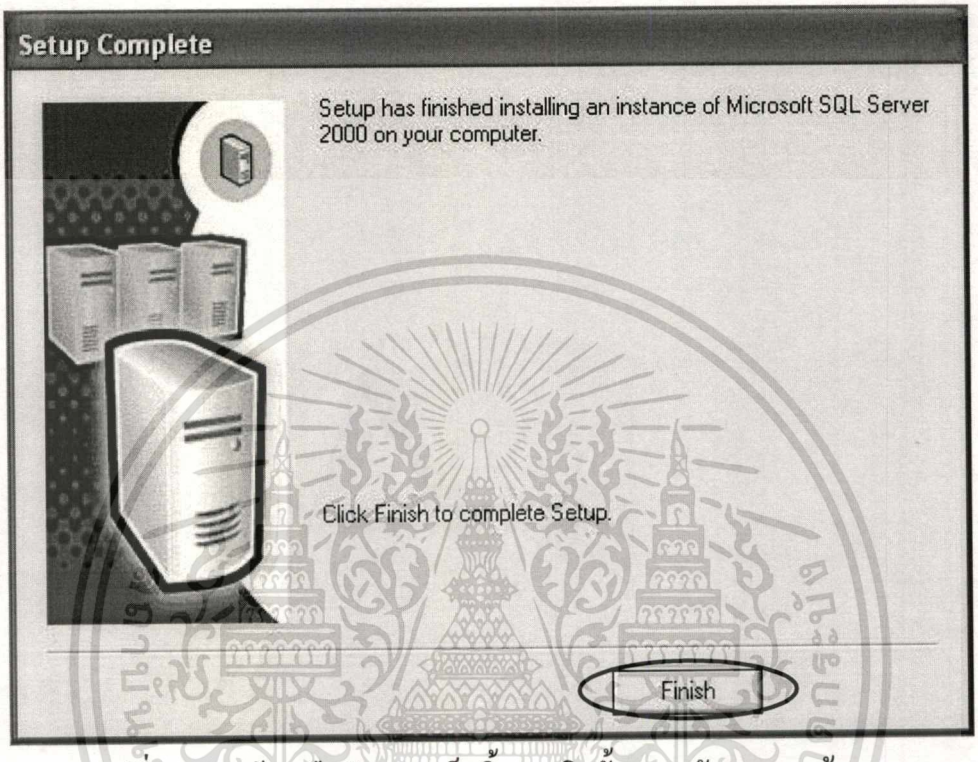
รูปที่ ก.19 หน้าจอยืนยันการคัดลอก program files

14. เริ่มดำเนินการติดตั้งระบบจัดการฐานข้อมูล ดังรูปที่ ก.20



รูปที่ ก.20 การดำเนินการติดตั้งระบบจัดการฐานข้อมูล

15. เมื่อติดตั้งเสร็จเรียบร้อยแล้ว จะขึ้นหน้าจอแสดงการเสร็จสิ้นการติดตั้งระบบจัดการฐานข้อมูล จากนั้นกดปุ่ม Finish เพื่อจบการติดตั้ง ดังรูปที่ ก.21



รูปที่ ก.21 หน้าจอยืนยันการเสร็จสิ้นการติดตั้งระบบจัดการฐานข้อมูล

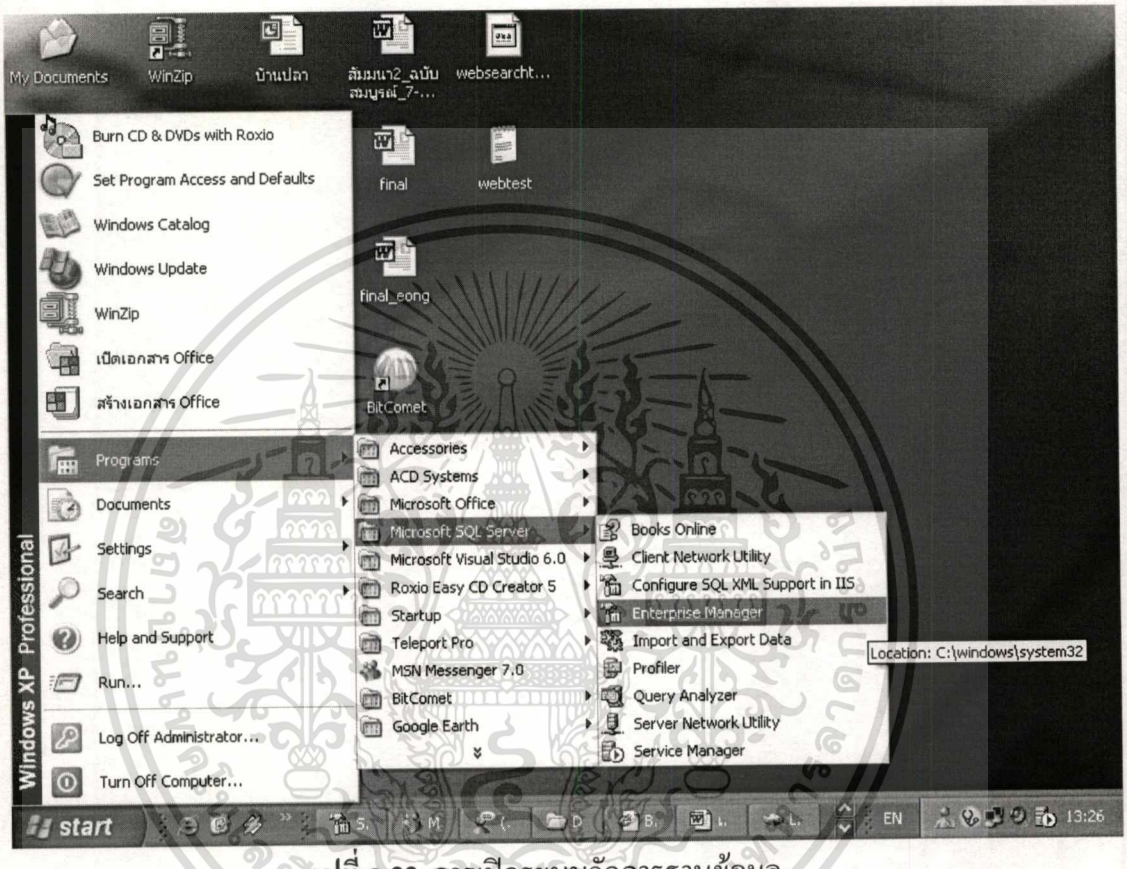
ก.3 การติดตั้งฐานข้อมูลสำหรับงานบริการการศึกษา

ก.3.1 สิ่งที่ต้องการในการติดตั้ง

- ระบบจัดการฐานข้อมูล SQL Server 2000
- ไฟล์ websearchDB.sql

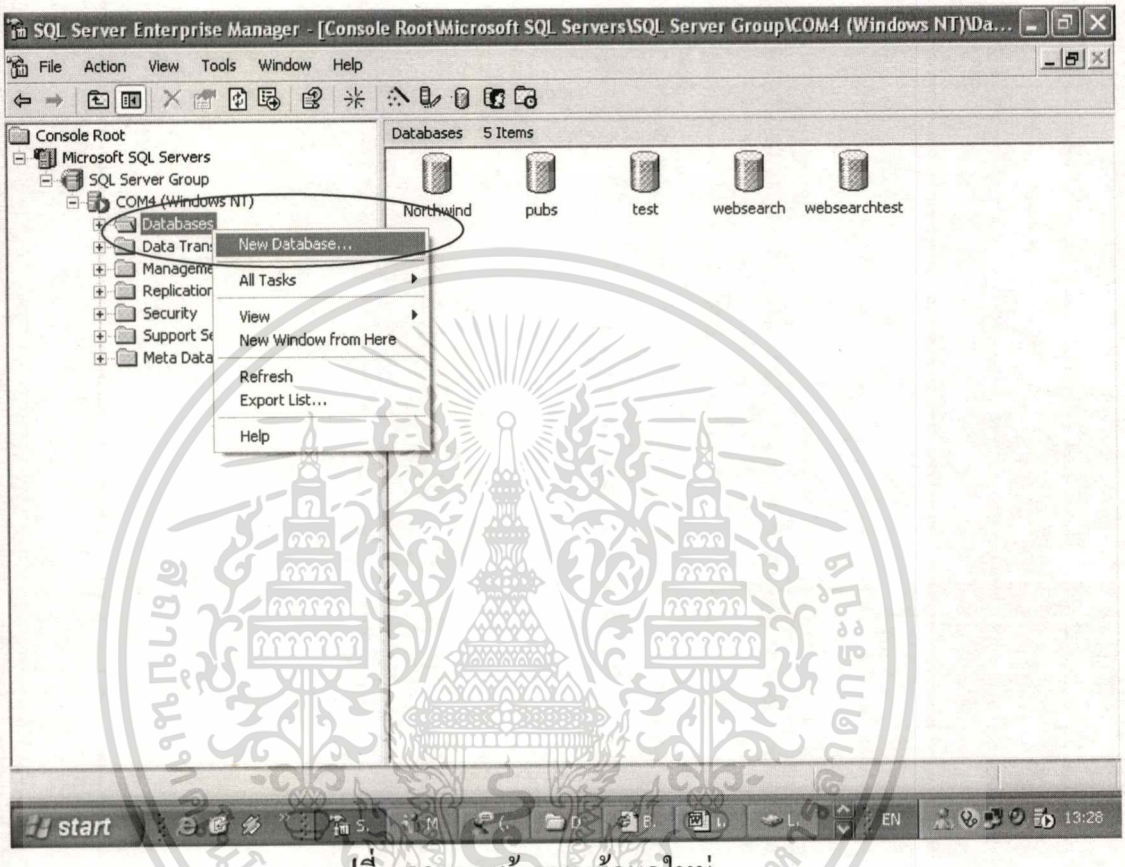
ก.3.2 ขั้นตอนการติดตั้ง

1. เปิดระบบจัดการฐานข้อมูล โดยเข้าไปที่ Start -> Programs -> Microsoft SQL Server -> Enterprise Manager ดังรูปที่ ก.22



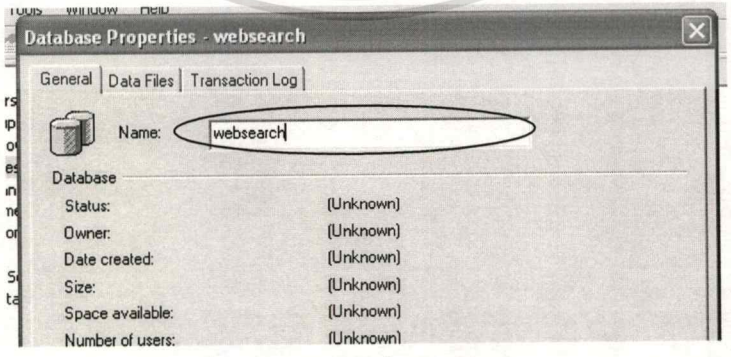
รูปที่ ก.22 การเปิดระบบจัดการฐานข้อมูล

2. สร้างฐานข้อมูลใหม่ โดยคลิกขวาตรง Databases และเลือก New Database… ดังรูปที่ ก.23



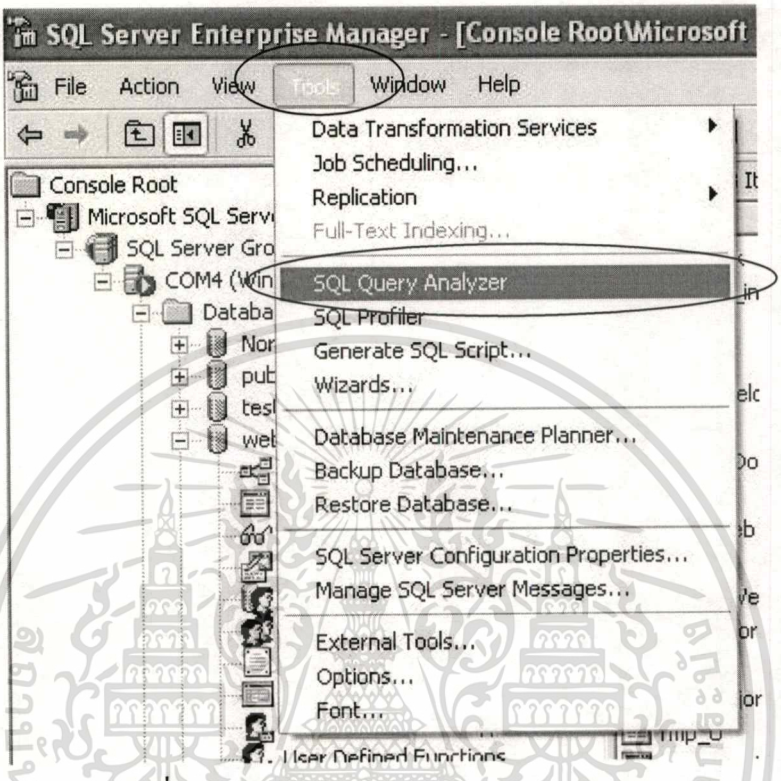
รูปที่ ก.23 การสร้างฐานข้อมูลใหม่

3. กำหนดชื่อฐานข้อมูลใหม่ โดยใช้ชื่อว่า “websearch” ดังรูปที่ ก.24



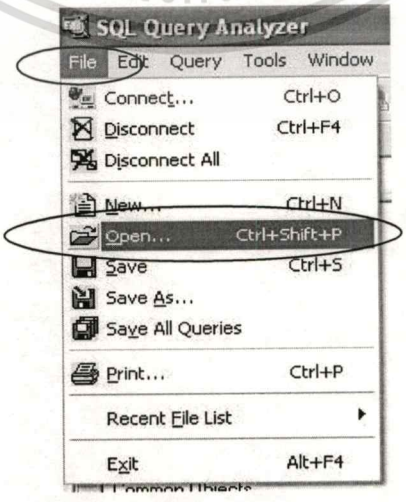
รูปที่ ก.24 กำหนดชื่อฐานข้อมูลใหม่

4. ทำการเรียกโปรแกรม Query Analyzer โดยเลือก Tools -> Query Analyzer ดังรูปที่ ก.25



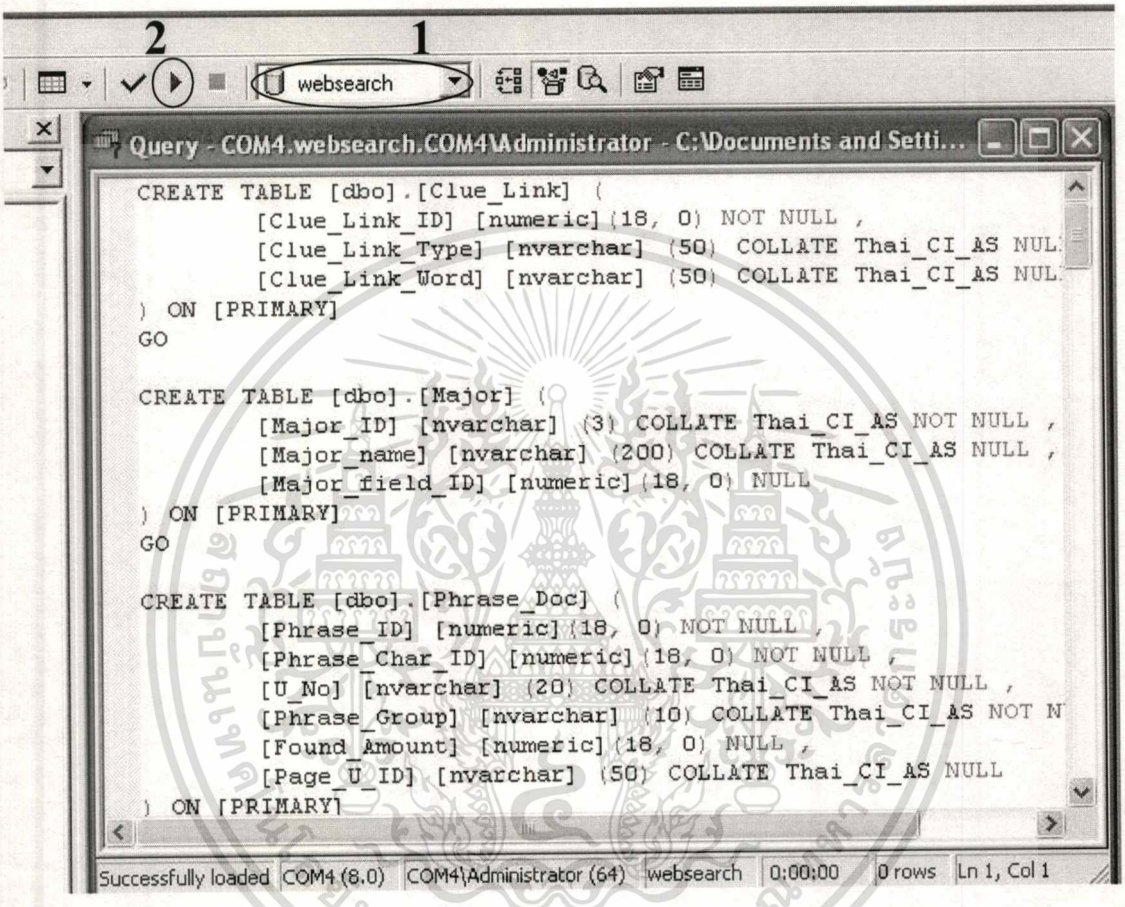
รูปที่ ก.25 การเรียกใช้โปรแกรม Query Analyzer

5. เมื่อเปิด Query Analyzer แล้ว ทำการเปิด SQL script สำหรับสร้างตารางในฐานข้อมูล โดยเลือก File->Open... แล้วทำการเลือกไฟล์ websearchDB.sql ที่มีอยู่ในโปรแกรมติดตั้งระบบ ดังแสดงการเลือกเมนูดังรูปที่ ก.26



รูปที่ ก.26 การเลือกไฟล์ SQL Script เพื่อสร้างตารางในฐานข้อมูล

6. เมื่อเลือกไฟล์แล้ว ทำการเลือกฐานข้อมูลที่จะจัดเก็บโดยเลือกฐานข้อมูลชื่อ websearch และสั่ง execute เพื่อทำการสั่งให้ทำงาน SQL Scipt สำหรับสร้างตารางในฐานข้อมูลรวมทั้งบันทึกข้อมูลเริ่มต้นที่ต้องใช้ในการทำงานของโปรแกรม ดังแสดงขั้นตอนการทำงานในรูปที่ ก.27



รูปที่ ก.27 การสั่งงาน SQL Script

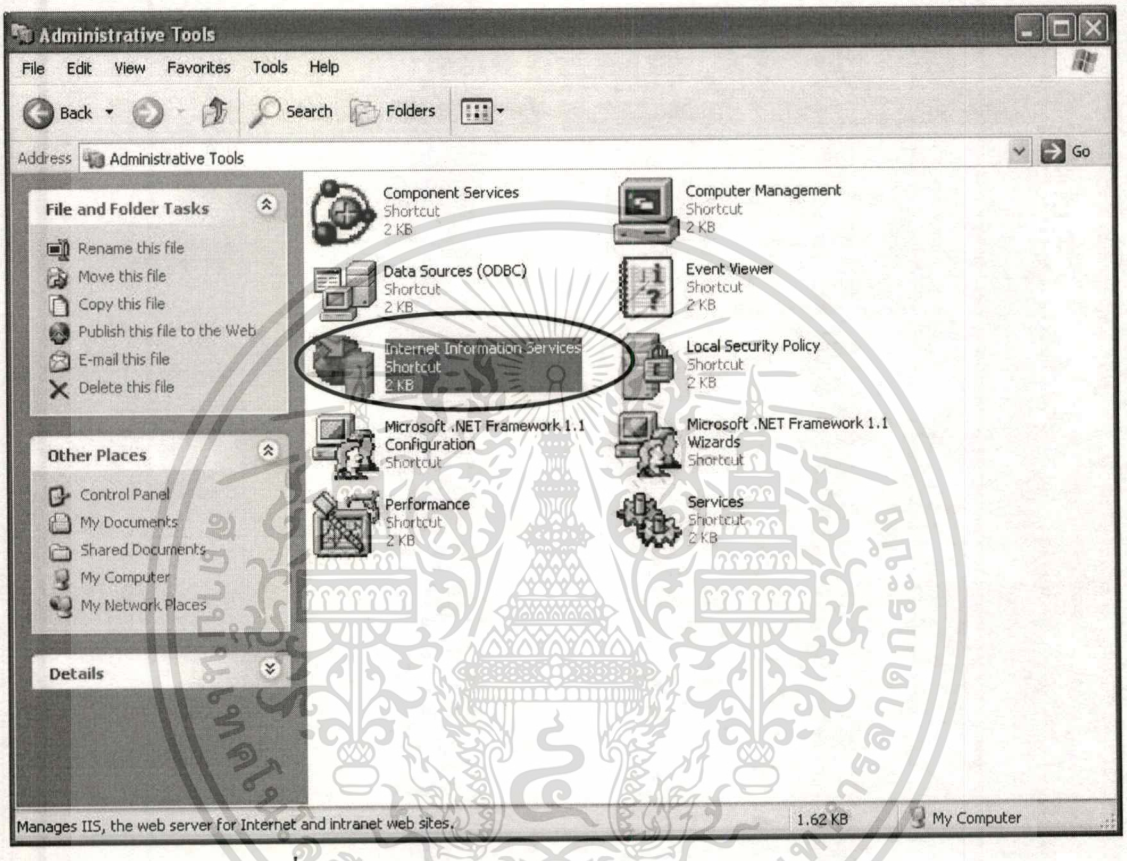
ก.4 การติดตั้งส่วนการค้นหาข้อมูลและแสดงผล

ก.4.1 สิ่งที่ต้องการในการติดตั้ง

- Internet Information Services
- โฟลเดอร์จัดเก็บหน้าเว็บเพจของส่วนการค้นหาและแสดงผลที่ได้พัฒนาขึ้น(โฟลเดอร์ websearch ที่อยู่ในโปรแกรมติดตั้งระบบ)

ก.4.2 ขั้นตอนการติดตั้ง

1. เปิด Internet Information Services โดยไปที่ Start -> Control Panel -> Administrative Tools -> Internet Information Services ดังแสดงในรูปที่ ก.28

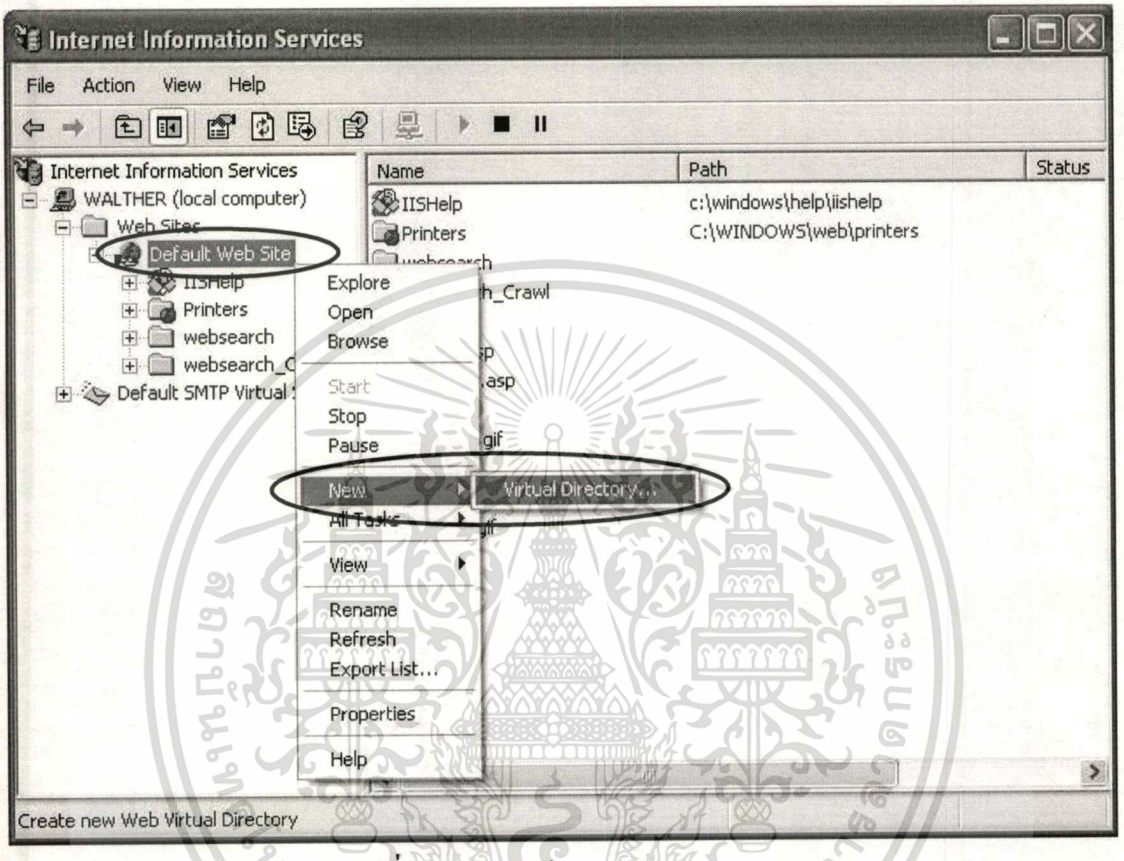


รูปที่ ก.28 การเรียกใช้งาน Internet Information Services

2. การติดตั้งโฟลเดอร์ websearch ให้สามารถใช้งานผ่าน web browser ได้นั้น สามารถทำได้ 2 วิธี คือ

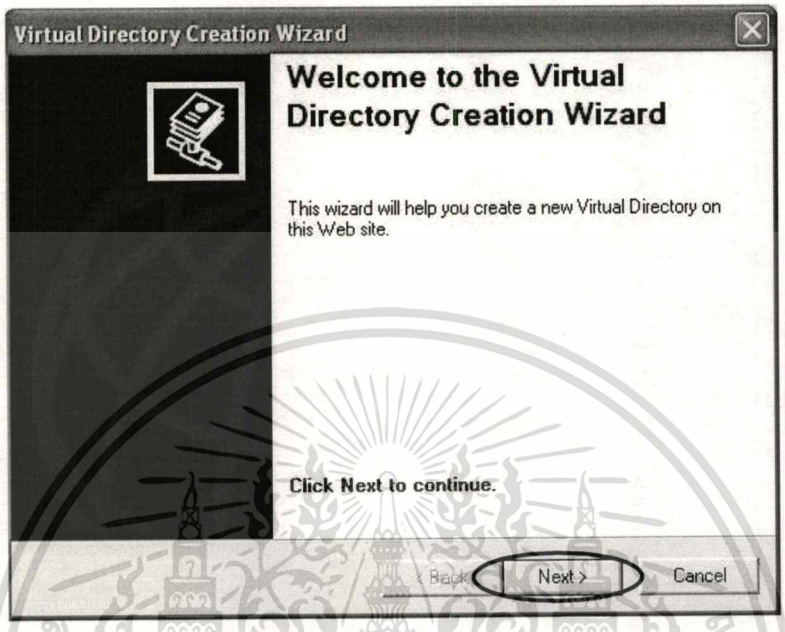
- นำโฟลเดอร์ websearch ไปวางไว้ที่ c:\Inetpub\wwwroot\
- ทำการกำหนด Virtual Directory ให้กับโฟลเดอร์ websearch

ในการกำหนด Virtual Directory นั้น จะทำการคลิกขวาที่ Default Web Site แล้วเลือก New -> Virtual Directory… ดังรูปที่ ก.29



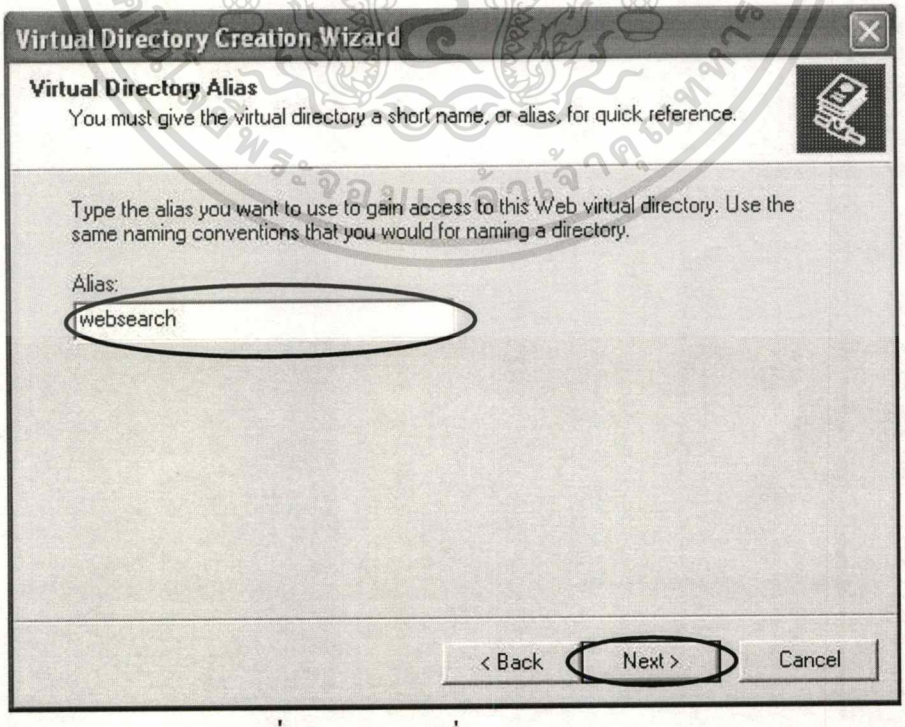
รูปที่ ก.29 การสร้าง Virtual Directory

3. จะปรากฏข้อความต้อนรับ กดปุ่ม Next > ดังรูปที่ ก.30



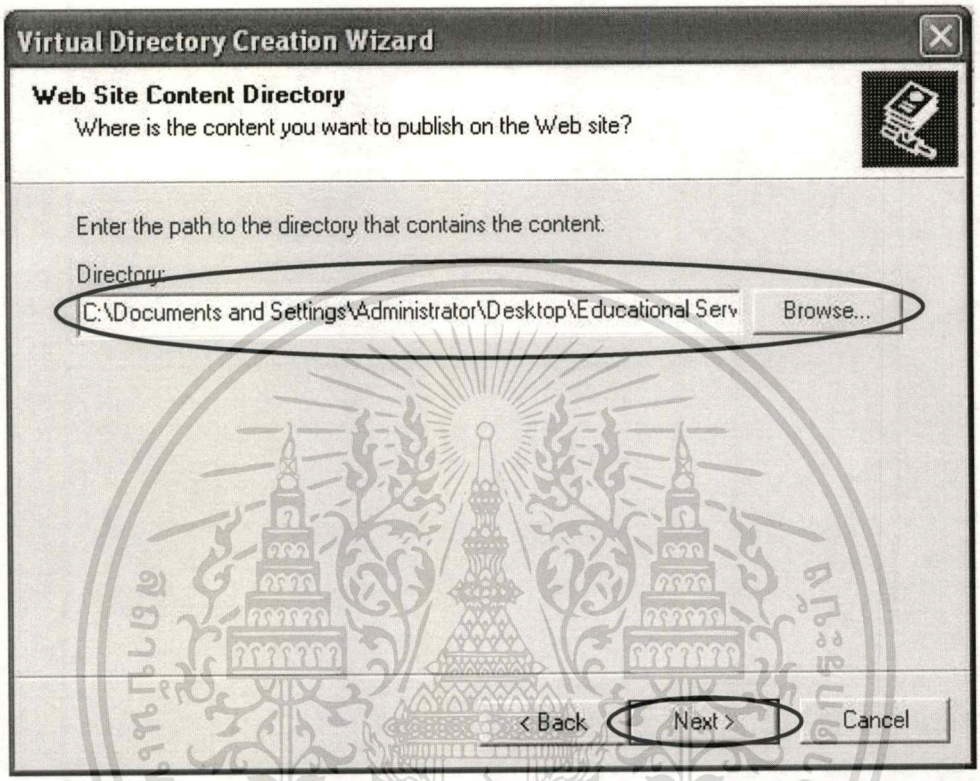
รูปที่ ก.30 หน้าจอแสดงข้อความต้อนรับ

4. ระบุชื่อที่จะใช้แสดงเป็น Virtual Directory และกดปุ่ม Next > ดังรูปที่ ก.31



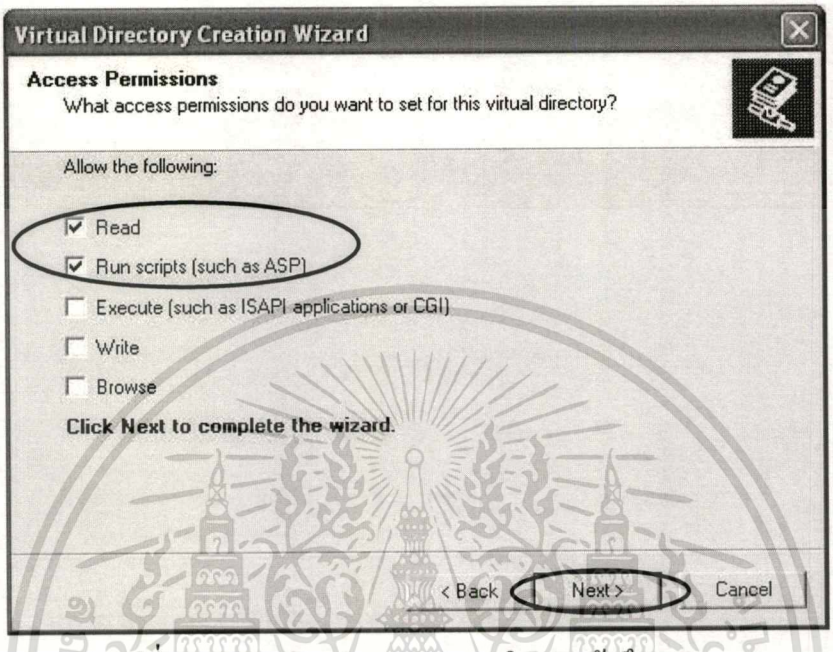
รูปที่ ก.31 การระบุชื่อ Virtual Directory

5. ระบุ Directory ที่อ้างถึง ซึ่งจะระบุ Directory ของโฟลเดอร์ websearch จากนั้นกดปุ่ม Next > ดังรูปที่ ก.32



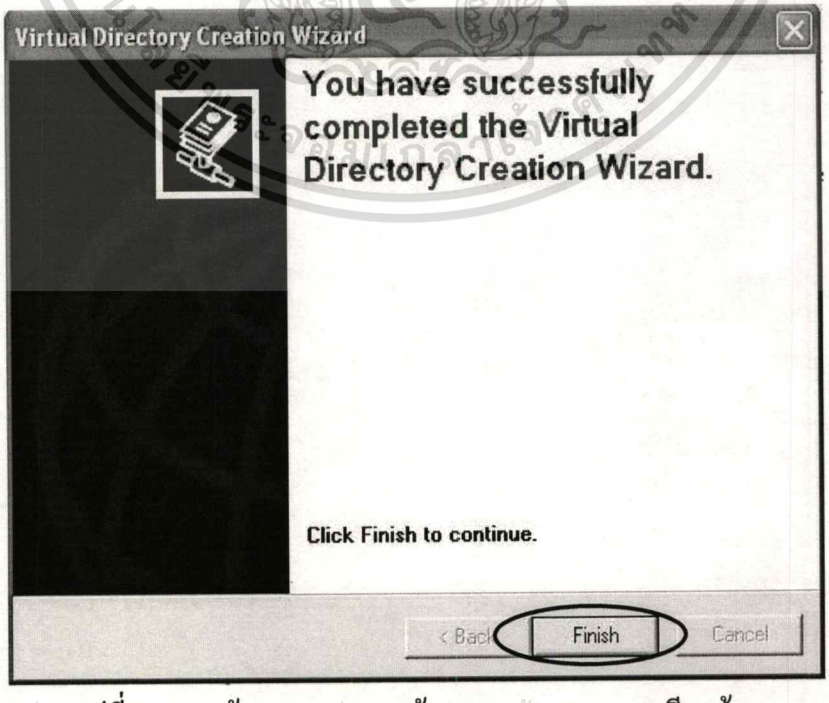
รูปที่ ก.32 การระบุ Directory ที่อ้างถึง

6. กำหนด Permission ในการเข้าถึงเอกสาร โดยในที่นี้ให้เลือกที่ Read และ Run scripts(such as ASP) แล้วกดปุ่ม ดังรูปที่ ก.33



รูปที่ ก.33 การกำหนด Permission ในการเข้าถึงเอกสาร

7. จะปรากฏหน้าจอแสดงการสร้าง Virtual Directory เรียบร้อย จากนั้นกดปุ่ม Finish เพื่อจบการทำงาน ดังรูปที่ ก.34

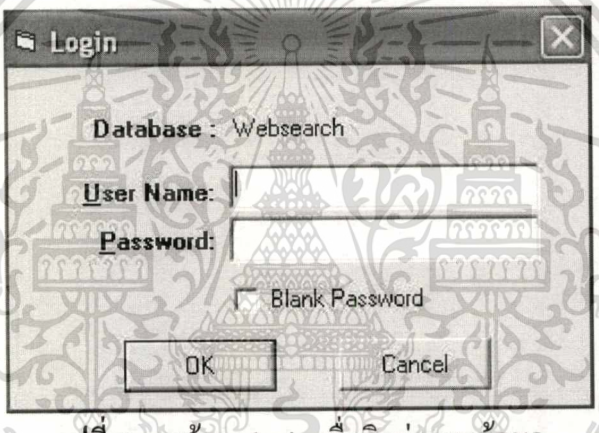


รูปที่ ก.34 หน้าจอแสดงการสร้าง Virtual Directory เรียบร้อย

ภาคผนวก ข
คู่มือการใช้งานระบบ

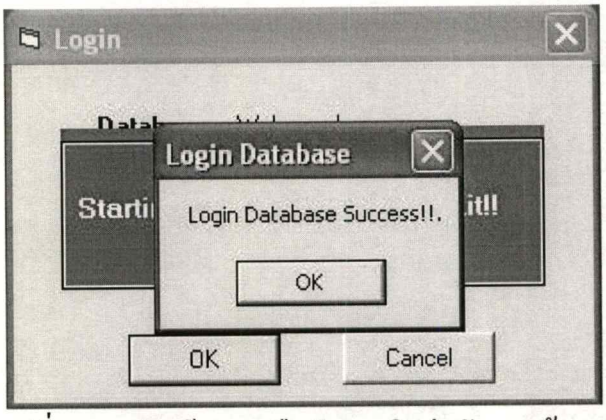
ข.1 การใช้งานโปรแกรม Educational Service Web Crawler

เมื่อมีการเรียกใช้งานโปรแกรม Educatioanal Service Web Crawler จะปรากฏหน้าจอสำหรับการ login เพื่อติดต่อฐานข้อมูล โดยผู้ใช้ทำการป้อน username และ password ที่สามารถติดต่อกับฐานข้อมูลได้ โดยหน้าจอการ login สามารถแสดงได้ดังรูปที่ ข.1



รูปที่ ข.1 หน้าจอ login เพื่อติดต่อฐานข้อมูล

เมื่อป้อนข้อมูล Username และ password ถูกต้อง จะสามารถทำการติดต่อกับฐานข้อมูลได้ โปรแกรมจะปรากฏกล่องข้อความยืนยันการติดต่อกับฐานข้อมูล ดังรูปที่ ข.2

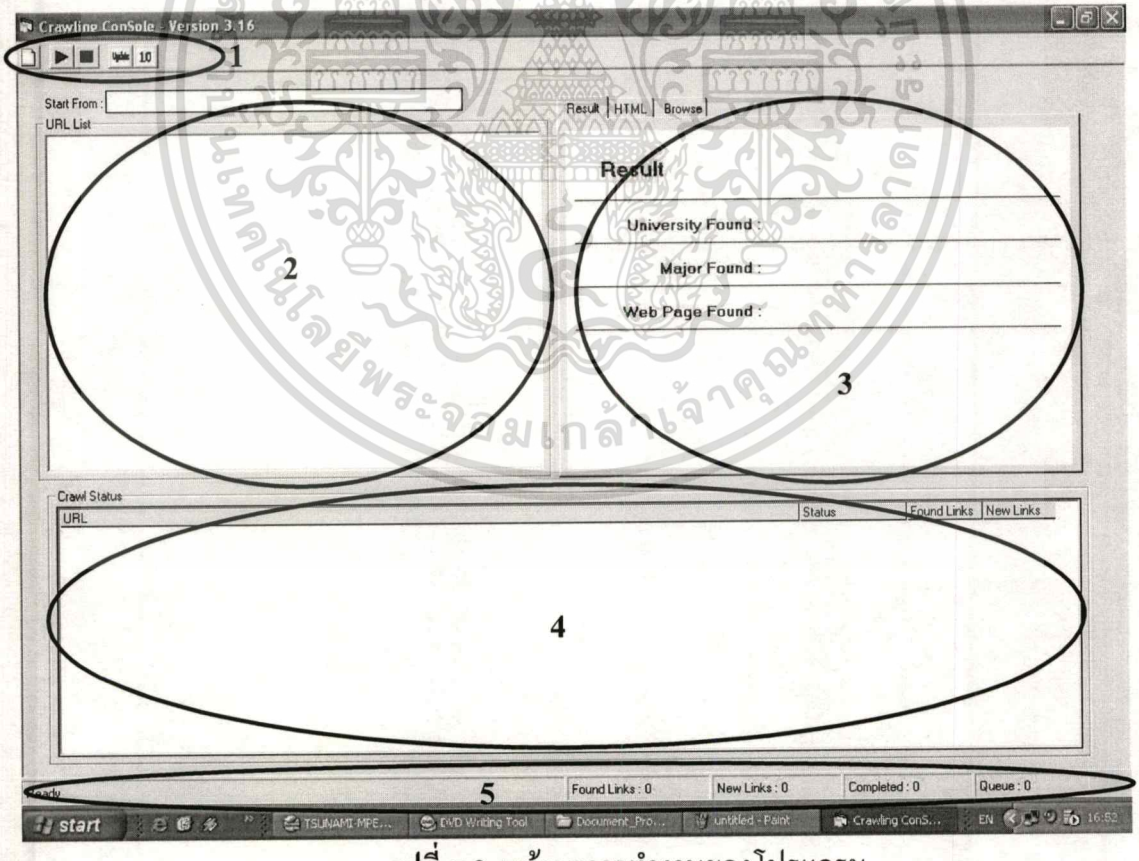


รูปที่ ข.2 กล่องข้อความยืนยันการติดต่อกับฐานข้อมูล

สำหรับหน้าจอการทำงานจะประกอบไปด้วยส่วนต่างๆ ดังนี้

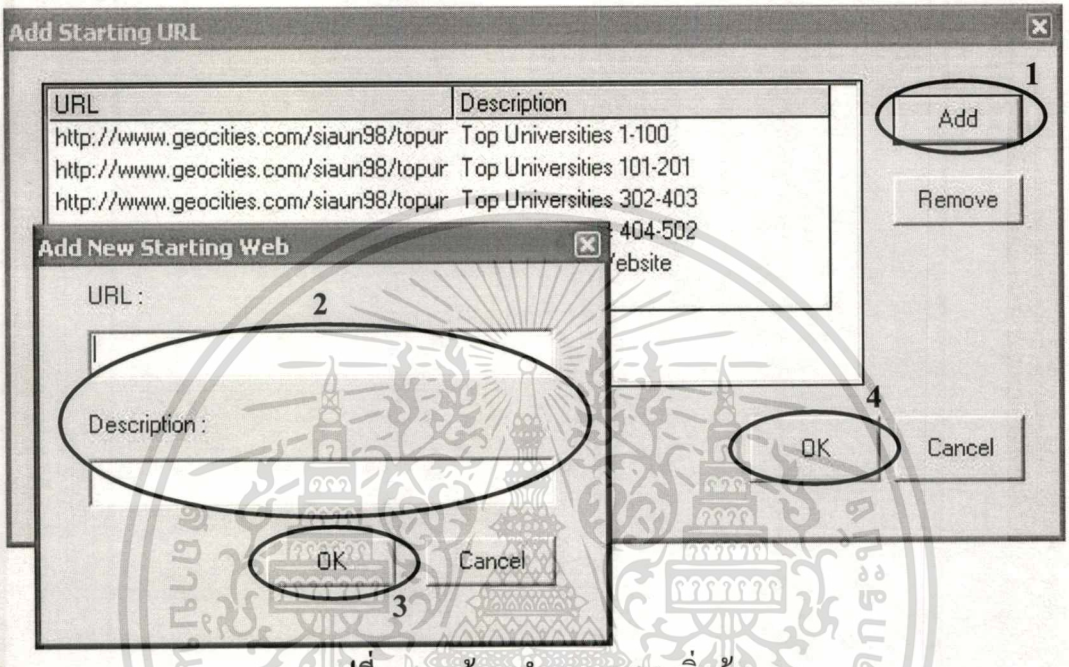
1. เมนู สำหรับสั่งงานโปรแกรม
2. URL List จะแสดงรายชื่อ URL ที่จัดเก็บอยู่ในฐานข้อมูล
3. ส่วนแสดงข้อมูล จะแสดงผลการจัดเก็บข้อมูลของโปรแกรม ,การแสดง HTML ของ URL และ การแสดงหน้าเว็บเพจของ URL
4. Crawl Status แสดงรายชื่อของ URL ที่ Crawler ทำการท่องไป พร้อมทั้งระบุสถานะการท่องไปของ URL นั้นด้วย
5. Status Bar แสดงสถานะการทำงานของโปรแกรม โดยแสดงสถานการณ์ทำงานของ URL ปัจจุบัน แสดงจำนวนของ hyperlink ที่พบในเอกสาร, hyperlink ที่พบใหม่, จำนวน URL ที่ท่องไปโดย Crawler และ จำนวนของ URL ที่ยังอยู่ในฐานข้อมูลและยังไม่ได้ทำการท่องไปโดย Crawler

โดยหน้าจอการทำงานแสดงได้ดังรูปที่ ข.3



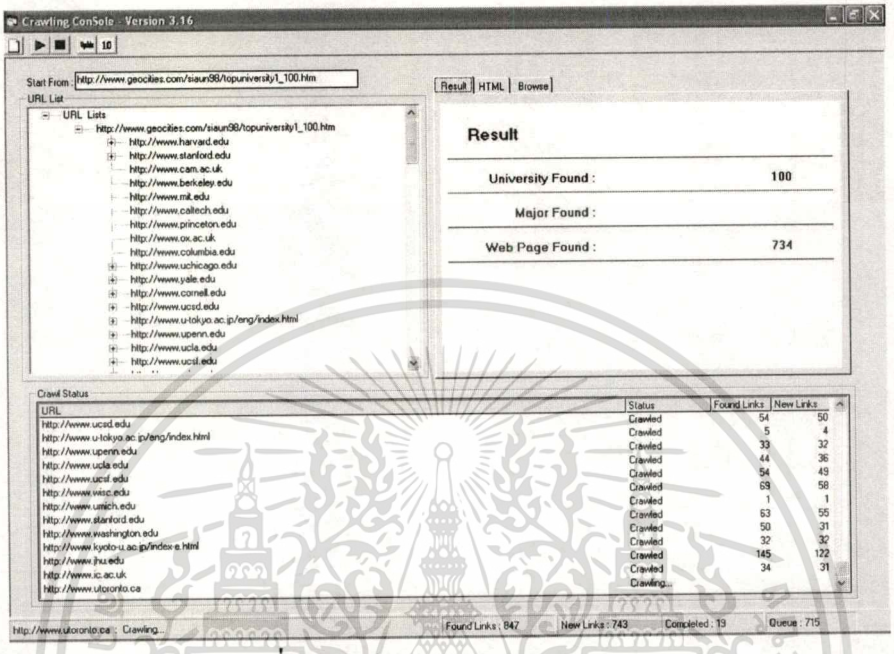
รูปที่ ข.3 หน้าจอการทำงานของโปรแกรม

ในการรวบรวมข้อมูลจากเว็บเพจต่างๆ จะต้องกำหนด URL เริ่มต้นให้กับโปรแกรม โดยกดปุ่ม  ที่เมนูเพื่อกำหนด URL เริ่มต้น โดยหากต้องการเพิ่มรายชื่อ URL ใหม่ กดที่ปุ่ม Add และพิมพ์ชื่อ URL พร้อมคำอธิบาย และกด OK ดังรูปที่ ข.4



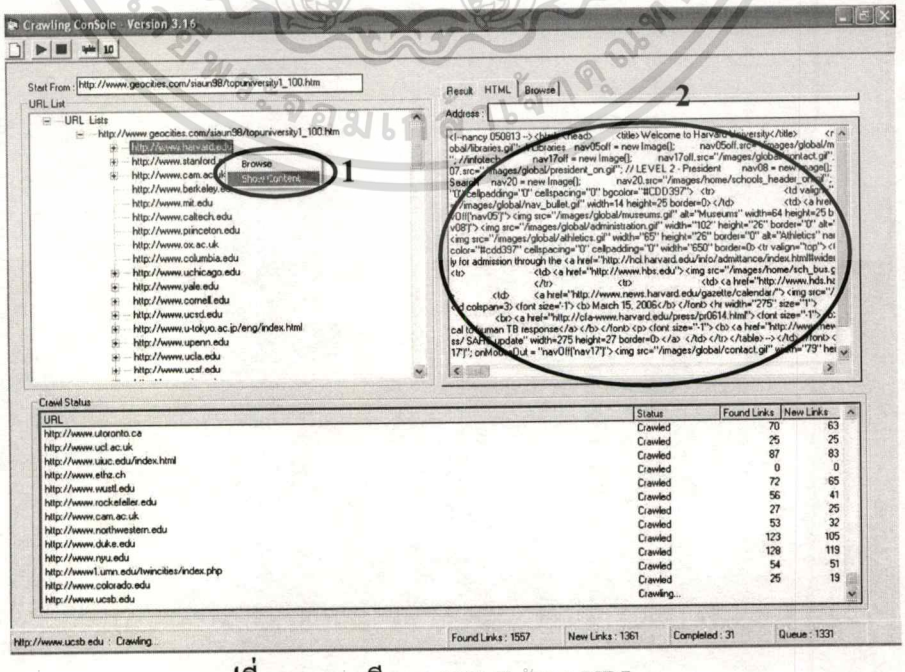
รูปที่ ข.4 หน้าจอกำหนด URL เริ่มต้น

กดปุ่ม ▶ เพื่อเริ่มให้ Crawler ท่องไปยังเว็บเพจ โดยสามารถดูสถานะการทำงานของโปรแกรมได้ ดังรูปที่ ข.5



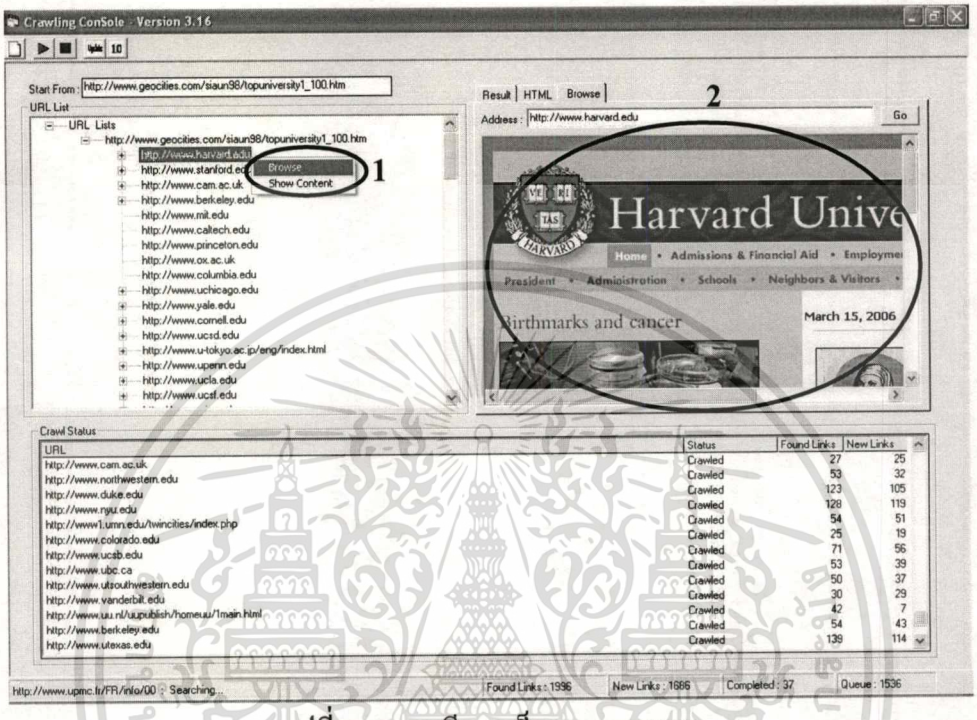
รูปที่ ข.5 หน้าจอขณะโปรแกรมทำงาน

โปรแกรมสามารถดู HTML ของ URL นั้นๆได้ โดยคลิกขวาที่ URL ใน URL List แล้วเลือก Show Content ซึ่งจะแสดง HTML ทางด้านขวามือในส่วนการแสดงข้อมูล ดังรูปที่ ข.6



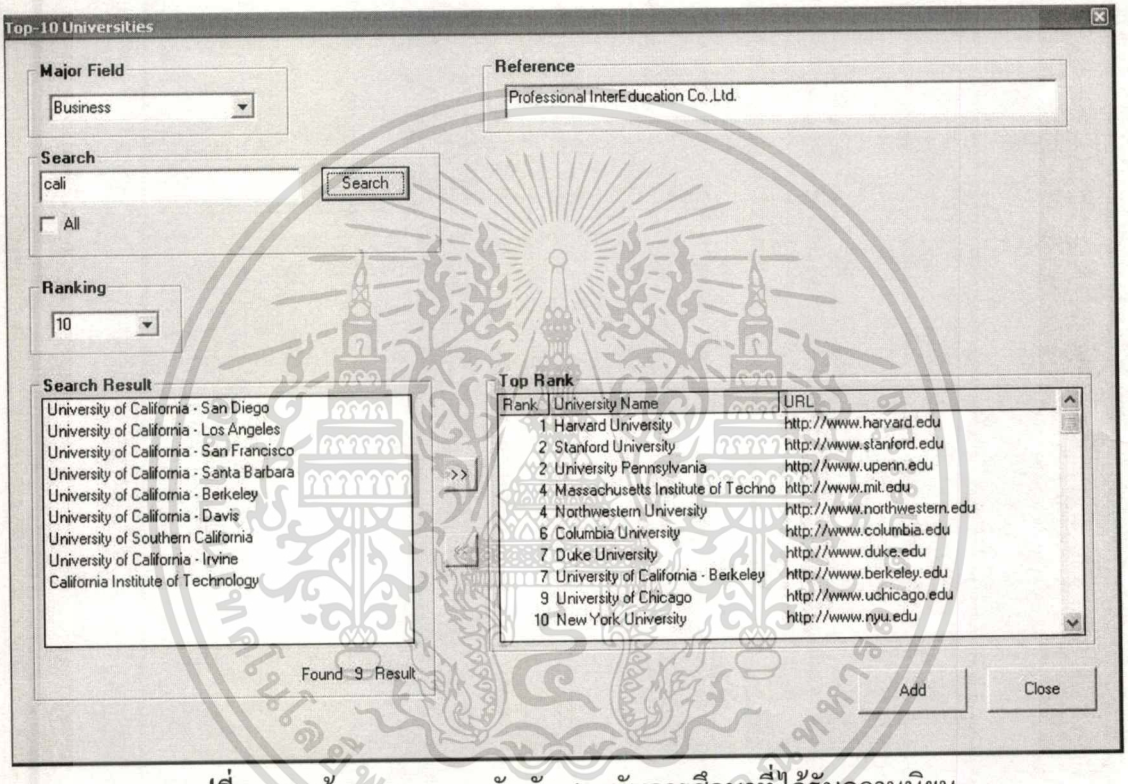
รูปที่ ข.6 การเรียกดู HTML ของ URL

ทั้งนี้ โปรแกรมยังสามารถเรียกดูเว็บเพจของ URL นั้นได้ โดยคลิกขวาที่ URL ใน URL List ซึ่งจะแสดงหน้าเว็บเพจทางด้านขวาในส่วนการแสดงข้อมูล ดังรูป ข.7



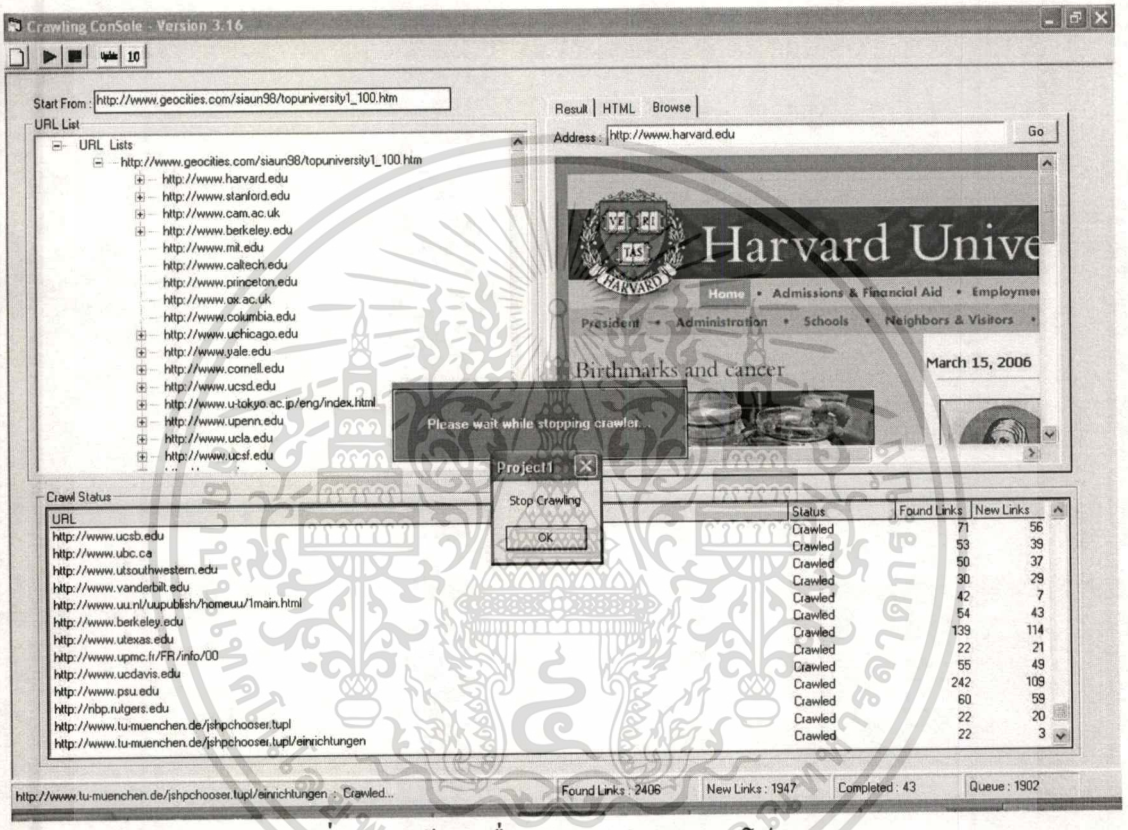
รูปที่ ข.7 การเรียกดูเว็บเพจของ URL

โปรแกรมสามารถทำการกรอกอันดับของสถาบันการศึกษาที่ได้รับความนิยม ตามกลุ่มของสาขาวิชา ซึ่งจะใช้ข้อมูลที่อ้างอิงมาจากแหล่งข้อมูลที่เชื่อถือได้ โดยกดที่ปุ่ม 10 ซึ่งในหน้าจอการทำงานนั้นจะมีการระบุกลุ่มสาขาวิชาที่จะบันทึก ค้นหารายชื่อสถาบันการศึกษา และระบุอันดับความนิยม และกดปุ่ม >> เพื่อจัดอันดับ ทั้งนี้ สามารถกรอกข้อมูลของแหล่งอ้างอิงได้ด้วย สำหรับหน้าจอการกรอกอันดับสถาบันการ ศึกษาที่ได้รับความนิยม แสดงได้ดังรูปที่ ข.8



รูปที่ ข.8 หน้าจอการกรอกอันดับสถาบันการศึกษาที่ได้รับความนิยม

ในการปรับปรุงข้อมูลของโปรแกรม สามารถกดปุ่ม  เพื่อให้ Crawler ทำการท่องไปยังเว็บเพจที่เคยรวบรวมไว้อีกครั้งเพื่อหาข้อมูลใหม่ และดึงข้อมูลนั้นมาจัดเก็บ ซึ่งหากต้องการให้โปรแกรมหยุดการทำงาน สามารถสั่งหยุดการทำงานโดยปุ่ม  ซึ่งจะปรากฏกล่องข้อความดังรูปที่ ข.9



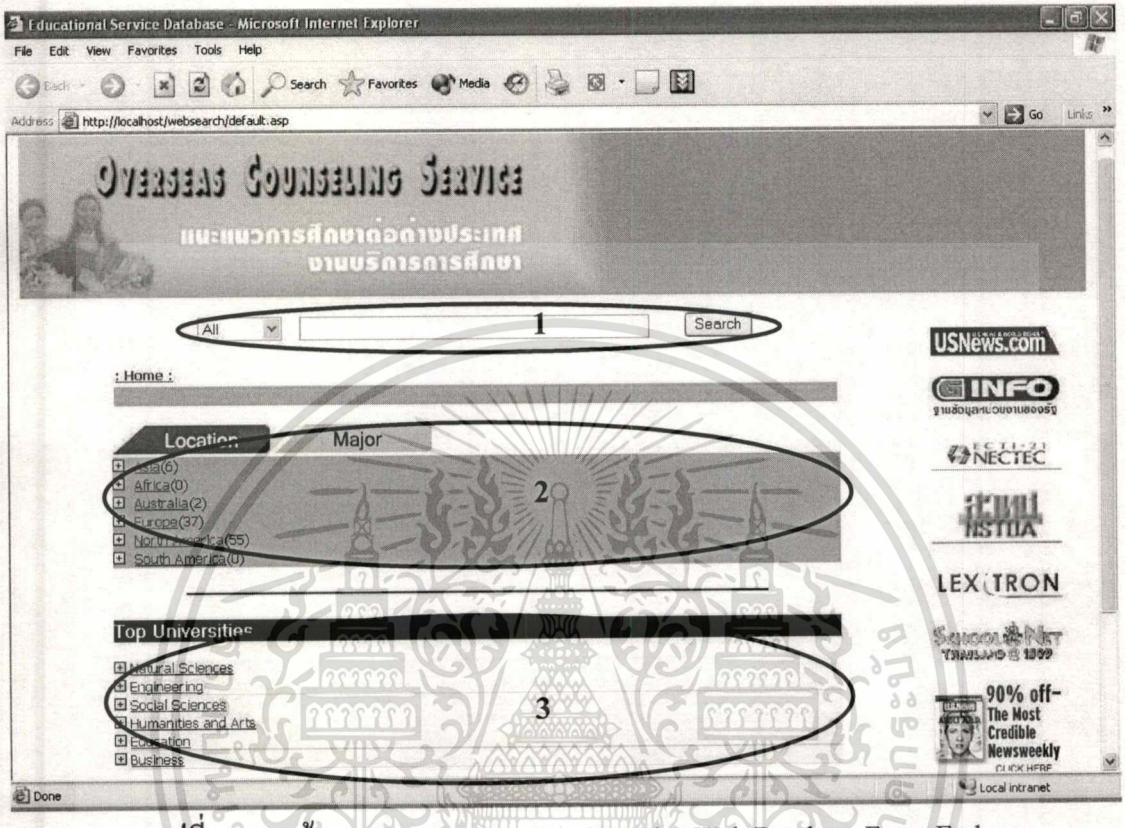
รูปที่ ข.9 หน้าจอเมื่อหยุดการทำงานของโปรแกรม

ข.2 การใช้งาน Educational Service Web Database Front-End

Educational Service Web Database Front-End เป็นส่วนการค้นหาข้อมูลและการแสดงผล ซึ่งจะแสดงข้อมูลที่ได้มาจากการรวบรวมข้อมูลของโปรแกรม Web Crawler และแสดงผลผ่าน Web Browser โดยมีส่วนการทำงานดังนี้

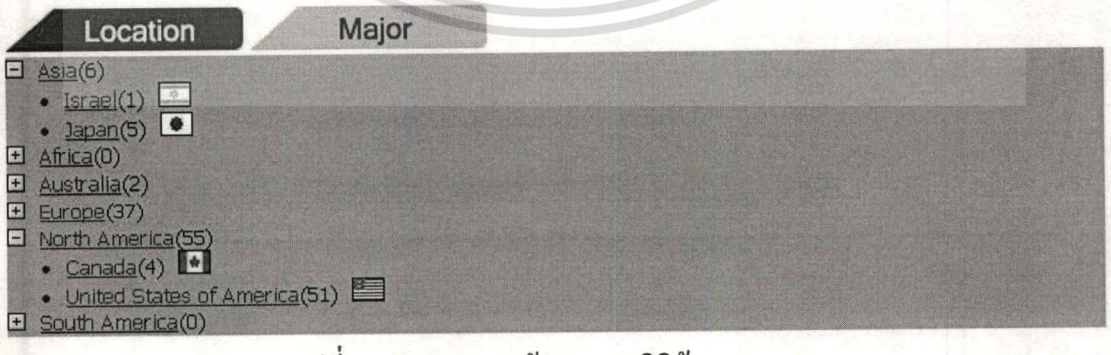
1. ส่วนการค้นหาข้อมูลโดยใช้คำค้นหา ผู้ใช้สามารถใส่คำค้นหา พร้อมทั้งเลือกมิติในการค้นหาได้ด้วย
2. การแสดงผลของข้อมูลในมิติทางด้าน Location และทางด้าน Major
3. อันดับสถาบันการศึกษาที่ได้รับความนิยม ตามกลุ่มของสาขาวิชา ซึ่งข้อมูลจะได้มาจากการกรองข้อมูลโดยโปรแกรม Web Crawler

สำหรับหน้าจอการทำงานสามารถแสดงได้ดังรูปที่ ข.10



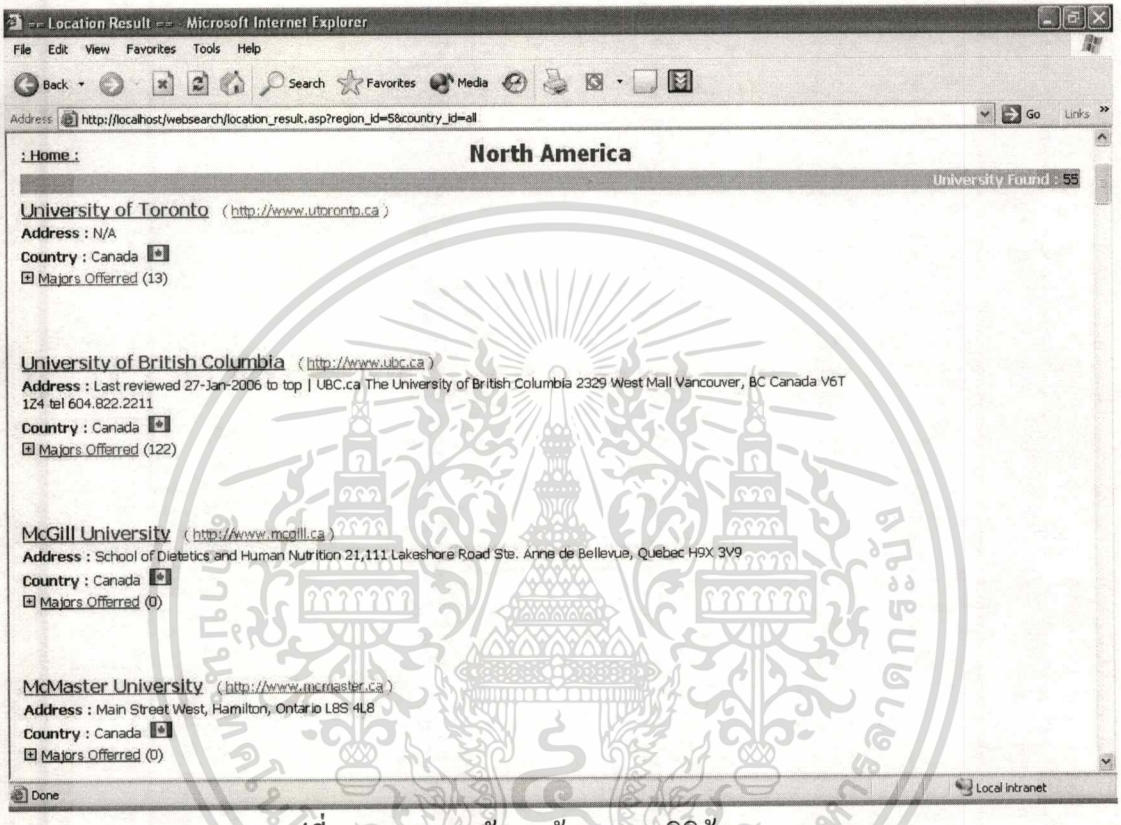
รูปที่ ข.10 หน้าจอแรกของ Educational Service Web Database Front-End

ในการค้นหาข้อมูลตามมิติด้าน Location นั้น จะมีการแสดงรายละเอียดของชื่อทวีป และชื่อประเทศ ดังรูปที่ ข.11



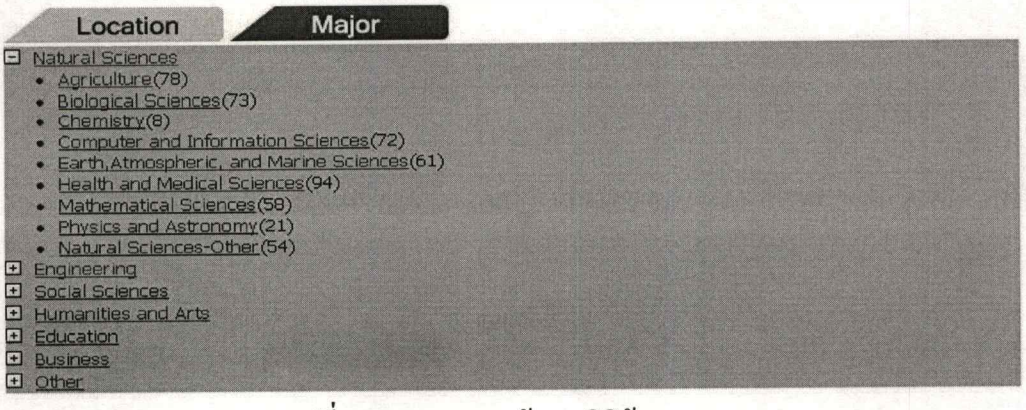
รูปที่ ข.11 การแสดงข้อมูลตามมิติด้าน Location

เมื่อเลือกทวีปหรือประเทศที่จะค้นหาข้อมูล จะแสดงผลของสถาบันการศึกษาที่ตั้งอยู่ตามทวีปหรือประเทศนั้นๆ อีกทั้งยังสามารถเลือกดูสาขาวิชาที่เปิดสอนของสถาบันการศึกษานั้นๆได้อีกด้วย ดังแสดงในรูปที่ ข.12



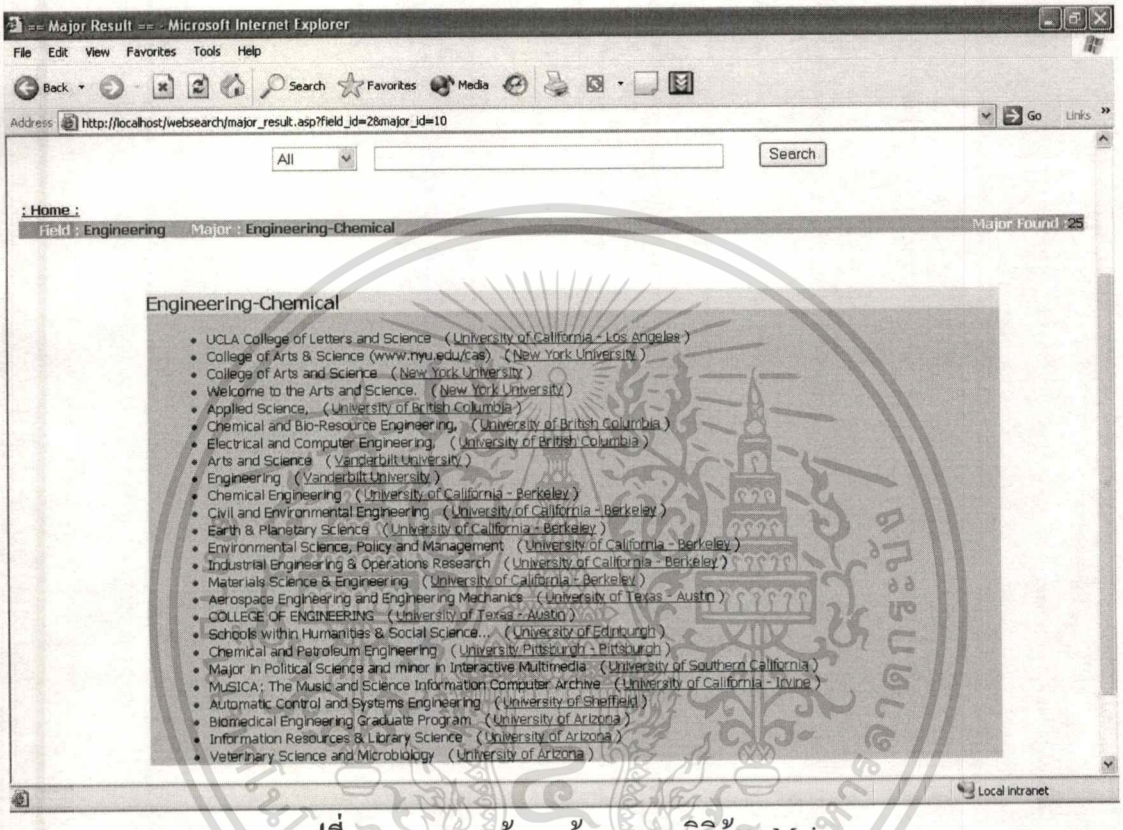
รูปที่ ข.12 ผลการค้นหาข้อมูลตามมิติด้าน Location

ในการค้นหาข้อมูลตามมิติด้าน Major นั้น จะมีการแสดงรายละเอียดของกลุ่มสาขาวิชาและรายชื่อสาขาวิชา ดังรูปที่ ข.13



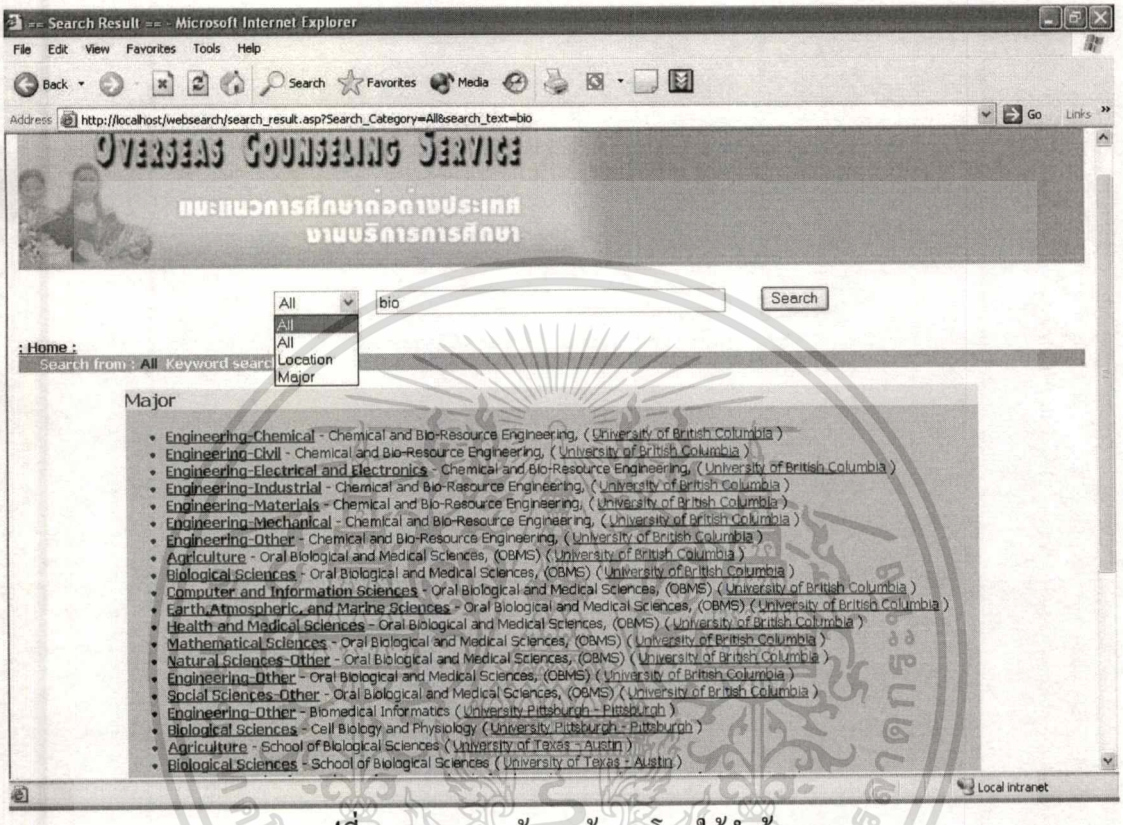
รูปที่ ข.13 การแสดงข้อมูลมิติด้าน Major

เมื่อเลือกกลุ่มสาขาวิชาหรือสาขาวิชาที่สนใจแล้ว จะแสดงผลการค้นหาข้อมูลในมิติด้าน Major โดยจะมีการแสดงชื่อของสาขาวิชาที่แต่ละสถาบันการศึกษาเปิดสอน โดยจัดกลุ่มตามสาขาวิชา ดังรูปที่ ข.14



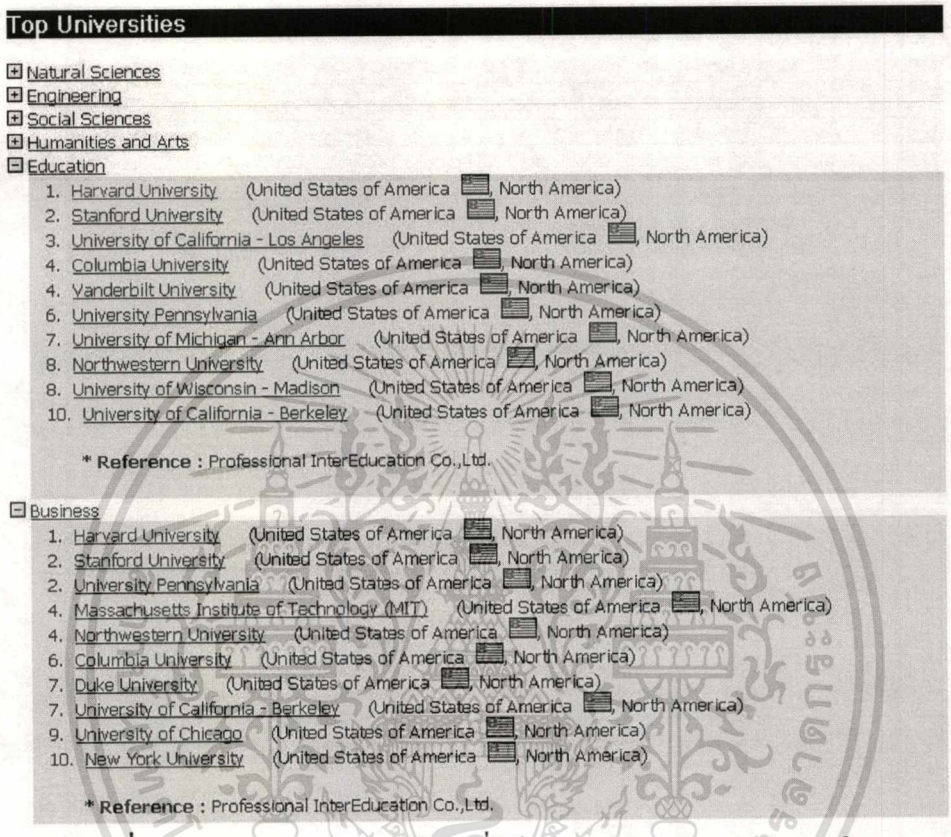
รูปที่ ข.14 ผลการค้นหาข้อมูลตามมิติด้าน Major

กรณีที่ผู้ใช้ค้นหาข้อมูลโดยใช้คำค้นหา ผู้ใช้สามารถเลือกมิติในการค้นหาข้อมูลได้ด้วย ซึ่งเมื่อทำการค้นหา จะแสดงผลของการค้นหา โดยแยกตามมิติของข้อมูลที่ค้นหามาได้ ดังรูปที่ ข.15



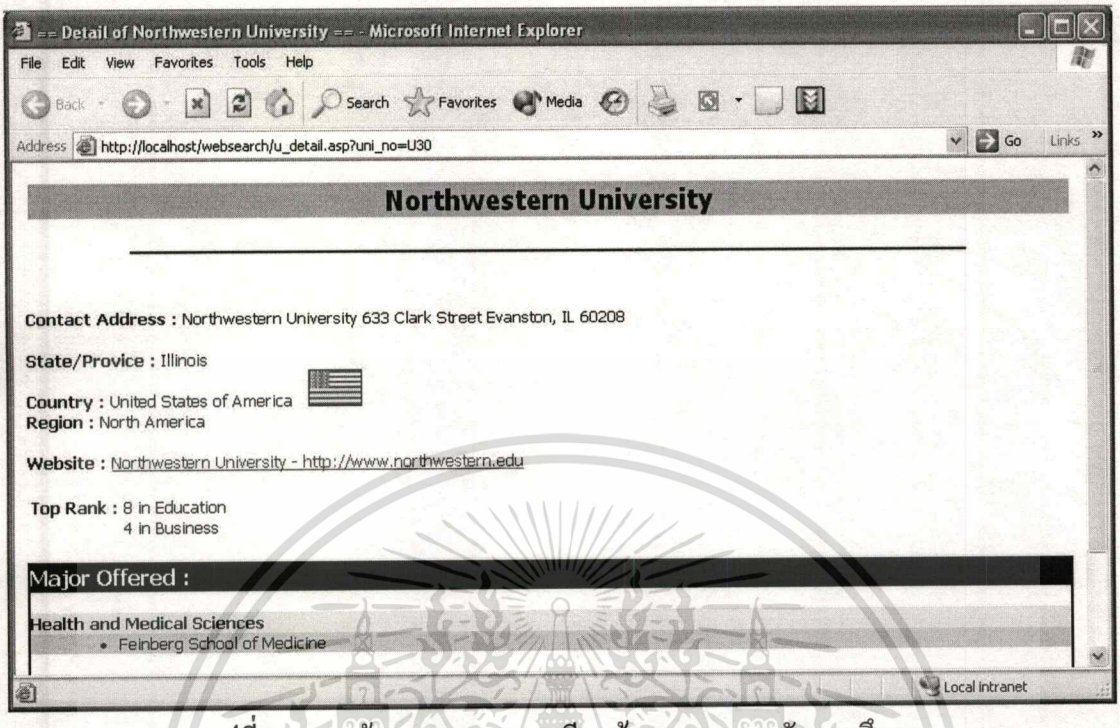
รูปที่ ข.15 ผลการค้นหาข้อมูลโดยใช้คำค้นหา

ในส่วนของการแสดงผลข้อมูลอันดับสถาบันการศึกษาที่ได้รับความนิยมนั้น ได้แสดงข้อมูลโดยแยกตามกลุ่มของสาขาวิชา และสามารถเลือกดูรายละเอียดในแต่ละกลุ่มได้ ดังรูปที่ ข.16



รูปที่ ข.16 อันดับสถาบันการศึกษาที่ได้รับความนิยม ตามกลุ่มสาขาวิชา

ทั้งนี้ ผู้ใช้สามารถดูรายละเอียดของแต่ละสถาบันการศึกษาได้ โดยคลิกที่ชื่อของงสถาบันการศึกษานั้นๆ ซึ่งจะปรากฏหน้าจอแสดงรายละเอียดของสถาบันการศึกษา ทั้งข้อมูลที่ตั้งและสาขาวิชา ดังรูปที่ ข.17



รูปที่ ข.17 หน้าจอแสดงรายละเอียดข้อมูลของสถาบันการศึกษา

ประวัติผู้เขียน

ชื่อ นายนรพันธ์ ศิริอำพันธ์กุล

วัน/เดือน/ปี เกิด 15 กันยายน พ.ศ. 2524

สถานที่เกิด จ.บุรีรัมย์

ประวัติการศึกษา

ปริญญาตรี คณะวิทยาศาสตร์และศิลปศาสตร์ สาขาระบบสารสนเทศคอมพิวเตอร์ มหาวิทยาลัยบูรพา วิทยาเขตสารสนเทศจันทบุรี