ระบบการค้นหาเอกสารอ้างอิงโดยใช้ Ontology

ONTOLOGY – AWARE SEARCH & NAVIGATION FOR BIBLIOGRAPHY RECORDS

นายกิตติกร หวานใจ รหัส 49360075

นายธีระพล สันติสำราญวิไล รหัส 49360815

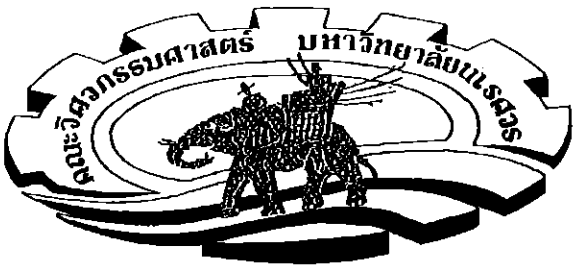
ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 11 ม.ค. 2555
เลขทะเบียน 1672 9646
เลขเรียกหนังสือ ปร.
มหาวิทยาลัยนเรศวร T1673 ร 2552

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2552



ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน ระบบการค้นหาเอกสารอ้างอิงโดยใช้ Ontology

ผู้ดำเนินโครงงาน นายกิตติกร หวานใจ รหัส 49360075

นายธีระพล สันติสำราญวิไล รหัส 49360815

อาจารย์ที่ปรึกษา ดร.วรลักษณ์ คงเด่นฟ้า

สาขาวิชา วิศวกรรมคอมพิวเตอร์

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2552

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

.....ประธานกรรมการ

(ดร.วรลักษณ์ คงเด่นฟ้า)

.....กรรมการ

(รองศาสตราจารย์ ดร.ไพศาล มุณีสว่าง)

.....กรรมการ

(อ.ภาณุพงศ์ สอนคม)

หัวข้อโครงงาน	ระบบการค้นหาเอกสารอ้างอิงโดยใช้ Ontology		
ผู้ดำเนินโครงงาน	นายกิตติกร	หวานใจ	รหัส 49360075
	นายธีระพล	สันติสำราญวิไล	รหัส 49360815
อาจารย์ที่ปรึกษา	ดร.วรลักษณ์ ดงเด่นฟ้า		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2552		

.....

บทคัดย่อ

จากความแพร่หลายและก้าวหน้าของการค้นคว้าข้อมูลบนโลกอินเตอร์เน็ต มีนักวิชาการจำนวนมากนำข้อมูลมาไว้บนอินเตอร์เน็ต เพื่อช่วยให้ผู้สนใจสามารถที่จะแลกเปลี่ยนข้อมูลกันได้ง่ายขึ้น ทำให้ผู้จัดทำเล็งเห็นถึงความสำคัญของการค้นคว้าหาข้อมูล จึงมีความต้องการที่จะพัฒนาเว็บแอปพลิเคชั่นที่สามารถช่วยเหลือการค้นหาและสืบค้นข้อมูลเอกสารผลงานวิชาการ สำหรับผู้ที่สนใจในสาขาวิชา Computer Science หรือ Computer Engineering ซึ่งทำให้การค้นคว้านั้นมีความสะดวกในการค้นหาข้อมูล และข้อมูลที่ได้นั้นมีความน่าเชื่อถือ สามารถที่จะนำข้อมูลต่างๆ เหล่านั้นไปใช้ในการศึกษาและวิจัยในเรื่องต่างๆได้อย่างมีประสิทธิภาพ โดยสิ่งสำคัญที่ผู้จัดทำต้องการเสนอก็จะเป็นในเรื่องของ Search และ Navigation ซึ่งระบบการ Search ทำให้ผู้ใช้สามารถที่จะค้นหาข้อมูลที่ต้องการจากเว็บไซต์ได้อย่างครอบคลุมโดยใช้หลักการ Ontology มาใช้ในการค้นหา และตรงตามความต้องการมากยิ่งขึ้นและ Navigation จะเป็นส่วนที่ช่วยให้ผู้ใช้สามารถที่เข้าถึงส่วนต่างๆของเว็บไซต์ได้ง่ายและตรงตามความต้องการ อีกทั้งยังสามารถทราบถึงข้อมูลที่มีความสัมพันธ์กันของสิ่งที่ต้องการค้นคว้าอีกด้วย

จากข้างต้นการพัฒนาเว็บแอปพลิเคชั่นจะทำให้สามารถช่วยเหลือการสืบค้นข้อมูลของผู้ที่สนใจในเรื่องของ Computer Science หรือ Computer Engineering สามารถค้นหาข้อมูลได้ง่ายและมีความสะดวกรวดเร็ว โดยสามารถที่จะตอบสนองความต้องการของผู้ใช้ได้เป็นอย่างดีและสามารถนำข้อมูลนั้นๆไปใช้ได้อย่างมีประสิทธิภาพ

Project Title	ONTOLOGY – AWARE SEARCH & NAVIGATION		
Name	Mr Kittikorn	Wanjai	ID. 49360075
	Mr Teerapon	Santisumranvilai	ID. 49360815
Project Advisor	Woralak Kongdenfha, Ph.D.		
Major	Computer Engineering.		
Department	Electrical and Computer Engineering.		
Academic Year	2552		

.....

ABSTRACT

The advancement of Information and Communication Technology (ICT) encourages academics and researchers around the world to exchange information and collaborate with each other. In particular, research documents have been distributed over the Internet. Academics and researchers can then search and collect such documents easily comparing to the traditional way of paper-based collections. However, the vast amount of information available on the Internet raises the need of search and navigation methods to facilitate the tasks of Internet users. In this thesis, the authors therefore propose a web-based application that enables academics and researchers to search and navigate through research documents available on the Internet. We observe that while the search method can facilitate the document finding task, it is not easy for users to choose appropriate keywords for searching. We therefore provide the navigation feature to allow researchers to browse through a collection of documents as well as tag clouds to allow filtering of search results. To improve the accuracy of search results, we use the concept of ontology. While we focused on the concepts related to Computer Science and Engineering, the proposed method can be adopted to search documents in other contexts. Our experimental results have shown that the ontology-based search method can improve the accuracy of search results, and the variety of navigation methods allows users to have different views of information, which is useful when performing different types of data analysis.

กิตติกรรมประกาศ

โครงงานนี้จะสำเร็จลุล่วงมิได้ หากไม่มี ดร.วรลักษณ์ คงเด่นฟ้า ที่ได้ให้คำแนะนำในการวางแผนการออกแบบและพัฒนาโปรแกรมอย่างเป็นขั้นตอน ช่วยให้มีความรอบคอบและมีระบบระเบียบในการเขียนโปรแกรม ให้คำปรึกษาและคำแนะนำในทุกขั้นตอนของการจัดทำโครงงานด้วยดีเสมอมา รวมถึงเสียสละเวลาเพื่อช่วยตรวจสอบและเสนอแนะข้อบกพร่องที่ควรแก้ไขของโปรแกรมในขั้นตอนต่าง ๆ เสมอ

รวมถึง ดร.ไพศาล มุณีสว่าง และ อ.ภาณุพงศ์ สอนคม ที่ช่วยตรวจสอบความถูกต้องและให้คำแนะนำในการปรับปรุงโครงงานให้ดียิ่งขึ้น จนทำให้การจัดทำโครงงานนี้สำเร็จลุล่วงไปได้ด้วยดี

ทั้งนี้ ผู้พัฒนาโครงงานขอขอบพระคุณคุณพ่อ คุณแม่และครอบครัว ที่คอยให้กำลังใจตลอดเวลา เป็นหนึ่งในเบื้องหลังของความสำเร็จในการพัฒนาโครงงานนี้ เพื่อนๆ ที่ช่วยเป็นกำลังใจและให้ความช่วยเหลือเสมอ รวมถึงคณาจารย์ทุกท่านที่ได้ให้ความรู้ต่าง ๆ ตลอดมา

ผู้พัฒนาโครงงานจึงขอขอบพระคุณทุกท่านไว้ ณ โอกาสนี้

นายกิตติกร หวานใจ

นายธีระพล สันติสำราญวิไล

สารบัญ

หน้า

บทคัดย่อภาษาไทย .....ก

บทคัดย่อภาษาอังกฤษ .....ข

กิตติกรรมประกาศ .....ค

สารบัญ .....ง

สารบัญตาราง .....ช

สารบัญรูป .....ฌ

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญ 1

1.2 วัตถุประสงค์ของโครงการ 2

1.3 ขอบข่ายของโครงงาน 2

1.4 ขั้นตอนการดำเนินงาน 2

1.5 ผลที่คาดจะได้รับ.....4

1.6 รายละเอียดงบประมาณของโครงงาน 4

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 ฐานข้อมูล (Database) 5

2.1.1 ไฟล์ Text (Text Files) 5

2.1.2 ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) 6

2.1.3 ไฟล์ XML (XML File) 6

2.2 ภาษาสอบถาม (Query Languages) 8

2.2.1 ภาษาสอบถามเชิงโครงสร้าง (Structure Query Language : SQL) 8

2.2.2 เอกซ์พาท (XPATH) 9

2.2.3 เอกซ์คิวรี (XQuery) 9

2.2.4 Language Integrated Query (LINQ).....10

2.2.4.1 LINQ to Object 11

2.2.4.2 LINQ to SQL ,LINQ to Datasets ,LINQ to Entities 11

2.2.4.3 LINQ to XML 13

สารบัญ (ต่อ)

หน้า

2.3 เทคโนโลยีในการพัฒนา Web Applications.....13

2.3.1 เอเอสพี (Active Server Pages : ASP) 14

2.3.1.1 การทำงานของ ASP 14

2.3.1.2 ข้อดีของ ASP 14

2.3.1.3 เอกสาร ASP 15

2.3.1.4 ความต้องการของระบบในการใช้งาน ASP.....16

2.3.2 พีเอชพี (Professional Home Page : PHP) 16

2.3.2.1 การทำงานของ PHP 16

2.3.2.2 ข้อดีของ PHP 16

2.3.2.2 เอกสาร PHP 17

2.3.2.4 ความต้องการของระบบในการใช้งาน PHP 17

2.3.3 เจเอสพี (Java Server Page : JSP) 17

2.3.3.1 การทำงานของ JSP 17

2.3.3.2 ข้อดีของ JSP 17

2.3.3.3 เอกสาร JSP 18

2.3.3.4 ความต้องการของระบบในการใช้งาน JSP 19

2.4 โมเดล-วิว-คอนโทลเลอร์ (Model-view-controller : MVC) 19

2.4.1 ความรู้พื้นฐานเกี่ยวกับ MVC 19

2.4.2 การทำงานของ MVC 20

2.4.3 ประโยชน์ของ MVC 21

2.5 การค้นหา (Search) 22

2.5.1 เทคนิคการสืบค้นข้อมูล 22

2.5.2 การค้นหาโดยใช้เมตาดาตา 23

2.6 การนำทางโดยฐานข้อมูล (Database navigation) 23

2.6.1 ออกแบบระบบนำทาง (Designing Web Navigation) 24

2.7 Ontology 26

2.8 เว็บค้นหา (Web Search Engine) 30

2.8.1 ประเภทของ Search Engine 31

สารบัญ (ต่อ)

หน้า

2.8.2 ครอว์เลอร์ (Crawler) 32
2.8.2 .1 ชนิดของครอว์เลอร์ (Crawler) 33
2.8.2.2 การประยุกต์ใช้ Crawler กับงาน Search Engine 33
2.8.3 การทำงานของเว็บค้นหา (Search Engine) 34

บทที่ 3 ขั้นตอนและวิธีการดำเนินงาน

3.1 การออกแบบระบบ (System Design) 40
3.2 แหล่งข้อมูล (Data Sources) 41
3.2.1 ดีบีแอลพี (DBLP) 41
3.2.2 เอกสาร (Publication) 45
3.2.3 Ontology 45
3.2.4 แทกซ์ (Tags) 47
3.2.5 ข้อมูลจากผู้ใช้ (User-Contributed Content) 48
3.3 การเข้าถึงข้อมูล (Data Access) 48
3.4 แบบจำลองไฮเปอร์เทกซ์ (Hypertext Model) 49
3.5 การค้นหา (Search) 51
3.5.1 เมตาดาตา (Meta Data) 51
3:5.2 เอกสาร (Publications) 51
3.5.3 Ontology 51
3.5.4 การค้นหาโดยใช้ข้อมูลจาก Metadata, Content และ Ontology 52
3.6 ระบบนำทาง (Navigation) 54
3.4.1 Gird view 54
3.4.2 Repeater view 54
3.4.3 Master detail view 54
3.4.4 Tree view 54
3.7 การจัดการข้อมูลผู้ใช้ (User - Contributed Content Management) 56
3.7.1 ประวัติผู้ใช้ (User Profile) 56

สารบัญ (ต่อ)

หน้า

3.7.2 การสนทนา (Conversation) 57
3.7.3 กลุ่ม (Group) 58
3.7.4 การแชร์ไฟล์ (File sharing) 58

บทที่ 4 ผลการทดลอง

4.1 การค้นหา (Search) 60
4.1.1 วิธีการทดสอบ 60
4.1.2 ผลการทดสอบ 61
4.1.3 สรุปผลการทดสอบ 63
4.2 ระบบนำทาง (Navigation) 63
4.2.1 วิธีการทดสอบ 63
4.2.2 ผลการทดสอบ 64
4.2.3 สรุปผลการทดสอบ 64
4.3 การจัดการข้อมูลผู้ใช้ (User - Contributed Content Management) 65
4.3.1 วิธีการทดสอบ 65
4.3.2 ผลการทดสอบ 65
4.3.3 สรุปผลการทดสอบ 66

บทที่ 5 บทสรุป

5.1 สรุปผลการดำเนินการ 67
5.2 ปัญหาและอุปสรรคในการพัฒนา 67
5.3 ข้อเสนอแนะ 67
5.4 แนวทางในการพัฒนาต่อในอนาคต 68

เอกสารอ้างอิง 69

ประวัติผู้เขียนโครงงาน 71

สารบัญตาราง

ตารางที่ หน้า

1.1 แผนการดำเนินงาน 3

4.1 ตารางทดสอบเพื่อวัดประสิทธิภาพของระบบโดยการค้นหาแบบ Metadata 61

4.2 ตารางทดสอบเพื่อวัดประสิทธิภาพของระบบโดยการค้นหาแบบ Content 61

4.3 ตารางทดสอบเพื่อวัดประสิทธิภาพของระบบโดยการค้นหาแบบ Ontology 62

4.4 ตารางทดสอบเพื่อวัดประสิทธิภาพของระบบโดยการค้นหาแบบ การ Search ที่ใช้ข้อมูลจากMetadata, Content และ Ontology 62

4.5 ตารางแบบสอบถามความคิดเห็นที่มีต่อการ Navigation 64

4.6 ตารางแบบสอบถามความคิดเห็นที่มีต่อการ User - Contributed Content Management...65

สารบัญรูป

รูปที่		หน้า
2.1	แสดงความสัมพันธ์ Virtual object database และ Linq to SQL File (.dbml)	10
2.2	แสดงถึงส่วนประกอบ LINQ	11
2.3	แสดงการทำ Mapping ของ LINQ to SQL	12
2.4	แสดงการทำทำงานของ LINQ to SQL	13
2.5	แสดงส่วนประกอบของ MVC	19
2.6	แสดงการทำงานร่วมกัน MVC	20
2.7	แสดงการทำงานร่วมกัน MVC ลูกค้าสั่งซื้อสินค้าทางเว็บไซต์	21
2.8	แสดงรูปแบบ RDF Triple	28
2.9	แสดงรูปแบบ RDF Triples แบบอื่นๆ	28
2.10	ลักษณะการทำงานของ Crawler	32
2.11	แสดงการเดินทางของ Crawler	33
2.12	การประยุกต์ใช้ Crawler กับงาน Search Engine	33
2.13	กระบวนการของ Search Engine	34
2.14	กระบวนการตรวจสอบการรับค่าจาก User Interface โดยใช้ N-Gram	35
2.15	กระบวนการประมวลผลข้อความเพื่อนาคำไปสร้าง Index	35
2.16	แสดงการตัดคำเพื่อนาไปสร้าง Index	36
2.17	แสดงการตัดคำแบบ N-Gram	37
2.18	การสร้างดัชนีแบบ Inverted Index	37
2.19	ฐานข้อมูลจาก Indexer	38
2.20	สูตรคำนวณหาค่า Similarity	38
3.1	แสดงระบบการทำงานของ Web Application โดยใช้ MVC Pattern	40
3.2	รายละเอียดของ Book	42
3.3	รายละเอียดของ Proceeding	42
3.4	รายละเอียดของ Article	43
3.5	รายละเอียดของ Inproceeding	43
3.6	รายละเอียดของ Incollection	44
3.7	รายละเอียดของ User Content	44
3.8	แสดงโครงสร้าง ER – DIAGRAM ของ DBLP	44

สารบัญรูป(ต่อ)

รูปที่		หน้า
3.9	ตัวอย่างเอกสาร Publication	45
3.10	แสดงรายละเอียด Ontology AI&DB	46
3.11	แสดงรายละเอียด Tags	48
3.12	แสดงรายละเอียดเอกสารที่ User อัพโหลดไว้	48
3.13	แสดงโครงสร้าง Set ของหน้าเว็บเพจ การสืบค้นเอกสารอ้างอิงโดยใช้ Ontology	49
3.14	โครงสร้างระบบการสืบค้นเอกสารอ้างอิงโดยใช้ Ontology	52
3.15	แสดงการ Navigation นำเสนอข้อมูลในรูปแบบต่างๆ	54
3.16	รูปแบบแสดงรายการข้อมูลแบบ Gird view	55
3.17	รูปแบบแสดงรายการข้อมูลแบบ Repeater view	55
3.18	รูปแบบแสดงรายการข้อมูลแบบ Master detail view	56
3.19	รูปแบบแสดงรายการข้อมูลแบบ Tree view	56
3.20	แสดง Information ของ User	57
3.21	ระบบ Conversations	57
3.22	แสดง Group ของ User	58
3.23	แสดงการอัพโหลดไฟล์ ของ User	58
4.1	แสดงเซตเอกสารทั้งหมด	60
4.2	สมการการหา Precision	60
4.3	สมการการหา Recall	60
4.4	สมการการหา F-measure	61
4.5	ผลการวัดประสิทธิภาพของระบบการ Search	63
4.6	แบบสอบถามความคิดเห็นระบบการสืบค้นเอกสารอ้างอิงโดยใช้ Ontology	64
4.7	แบบสอบถามความคิดเห็นระบบการ User - Contributed Content Management	65

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญ

ในโลกยุคปัจจุบันเทคโนโลยีมีการพัฒนาเป็นอย่างมาก ซึ่งคอมพิวเตอร์ถือเป็นส่วนสำคัญอย่างหนึ่งของเทคโนโลยี การเข้ามาของ ICT ทำให้การทำงานต่างๆมีความสะดวกรวดเร็วมากยิ่งขึ้น และจากการพัฒนาความสามารถของคอมพิวเตอร์ให้สามารถเชื่อมโยงเป็นเครือข่ายคอมพิวเตอร์ ทำให้การติดต่อสื่อสารเป็นไปอย่างรวดเร็วและมีความสะดวกมากยิ่งขึ้นและที่สำคัญยังเป็นการลดค่าใช้จ่ายในการเดินทางเพื่อติดต่อสื่อสารระหว่างกัน นอกจากนี้การที่คอมพิวเตอร์มีการประมวลผลที่แม่นยำทำให้การทำงานต่างๆในองค์การมีประสิทธิภาพมากยิ่งขึ้น จากเดิมการทำงานในองค์กรต่างๆจำเป็นต้องใช้เอกสารจำนวนมาก แต่การเข้ามาแทนที่ของคอมพิวเตอร์ทำให้ลดการใช้เอกสารได้เป็นจำนวนมาก ซึ่งเป็นการลดการใช้ทรัพยากรและรักษาสิ่งแวดล้อมอีกด้วย

ในช่วง 5 ถึง 10 ปีที่ผ่านมาการพัฒนาของเทคโนโลยีเว็บแอปพลิเคชั่น ทำให้การค้นหาข้อมูลเป็นไปอย่างสะดวกมากยิ่งขึ้น และการค้นคว้าแหล่งข้อมูลที่อยู่ในที่ไกลๆเป็นไปได้สะดวกรวดเร็วยิ่งขึ้นและมีค่าใช้จ่ายที่น้อย อีกทั้งในขณะนี้มีผู้ให้บริการเว็บแอปพลิเคชั่นที่ให้ผู้คนต่างๆทั่วโลกสามารถที่จะแชร์ข้อมูลต่างๆ ที่เป็นประโยชน์แก่บุคคลอื่นได้ ทำให้ความรู้มีการกระจายสู่ผู้ที่อยู่ห่างไกลได้ ซึ่งทำให้การค้นคว้าข้อมูลต่างๆมีความหลากหลายและตรงตามความต้องการมากยิ่งขึ้น

จากความก้าวหน้าของการค้นหาข้อมูลบนโลกอินเตอร์เน็ต ซึ่งทำให้มีการแพร่หลายของข้อมูลจำนวนมากไว้บนอินเตอร์เน็ต เพื่อให้สามารถที่จะแลกเปลี่ยนข้อมูลกันได้สะดวกและรวดเร็วมากขึ้น ทำให้ผู้จัดทำโครงการนี้เล็งเห็นความสำคัญของการค้นหาข้อมูลบนโลกอินเตอร์เน็ต จึงมีความต้องการที่จะพัฒนาเว็บแอปพลิเคชั่นที่สามารถช่วยเหลือการค้นหาและสืบค้นข้อมูลเอกสารผลงานวิชาการสำหรับผู้ที่สนใจในสาขา Computer Science หรือ Computer Engineering โดยสิ่งสำคัญคือที่ต้องการเสนอก็จะเป็นในเรื่องของ Search และ Navigation ซึ่งระบบการ Search ทำให้ผู้ใช้สามารถที่จะค้นหาข้อมูลที่ต้องการจากเว็บไซต์ได้อย่างครอบคลุม และตรงตามความต้องการมากยิ่งขึ้น และ Navigation จะเป็นส่วนที่ช่วยให้ผู้ใช้สามารถที่เข้าถึงส่วนต่างๆของเว็บไซต์ได้ง่ายและตรงตามความต้องการ อีกทั้งยังสามารถทราบถึงข้อมูลที่มีความสัมพันธ์กันของสิ่งที่ต้องการค้นคว้าอีกด้วย

จากข้างต้นการพัฒนาเว็บแอปพลิเคชั่นจะทำให้สามารถช่วยเหลือการสืบค้นข้อมูลของผู้ที่สนใจในเรื่องของ Computer Science หรือ Computer Engineering สามารถค้นหาข้อมูลได้ง่ายและมีความสะดวกรวดเร็ว โดยสามารถที่จะตอบสนองความต้องการของผู้ใช้ได้เป็นอย่างดีและสามารถนำข้อมูลนั้นๆไปใช้ได้อย่างมีประสิทธิภาพ

1.2 วัตถุประสงค์ของโครงการ

ออกแบบและพัฒนา Web Application สำหรับผู้ที่สนใจในงานวิชาการทางด้าน Computer Since and Engincering ในงานดังต่อไปนี้

1.2.1 พัฒนาการสืบค้นข้อมูลโดยการ Navigation

1.2.2 พัฒนาการสืบค้นข้อมูลโดยการ Search

1.2.3 เปรียบเทียบว่าการใช้ Ontology มาช่วยในการ Navigation กับ Search จะช่วยให้ผู้ใช้งานสามารถสืบค้นผลงานวิชาการได้ดีขึ้นกว่าการใช้ Keyword based approach หรือไม่

1.3 ขอบข่ายของโครงงาน

1.3.1 ในโครงงานนี้สนใจเฉพาะข้อมูลเอกสารอ้างอิงที่เกี่ยวกับ Computer Since and Engineering เท่านั้นโดยนำข้อมูลมาจาก DBLP (DataBase systems and Logic Programming)

1.3.2 ใน Web Application ที่เราสร้างจะครอบคลุมเฉพาะ Navigation กับ Search

1.3.3 Ontology ที่นำมาใช้สนใจเฉพาะ DataBases และ Artificial Intelligence เบื้องต้น

1.4 ขั้นตอนการดำเนินงาน

1.4.1 ทำการศึกษาค้นคว้าข้อมูลเกี่ยวกับระบบฐานข้อมูลประเภทต่างๆ

1.4.2 ทำการศึกษาค้นคว้าข้อมูลเกี่ยวกับการกระทำการกับระบบฐานข้อมูลโดยใช้ LINQ

1.4.3 ทำการค้นคว้าหลักการพัฒนาเว็บแอปพลิเคชั่นโดย ASP.NET

1.4.4 ทำการพัฒนาโครงสร้างความสัมพันธ์ของเว็บแอปพลิเคชั่น

1.4.5 ทำการศึกษาค้นคว้าข้อมูลเกี่ยวกับแหล่งเอกสารอ้างอิง DBLP

1.4.6 ทำการศึกษาค้นคว้าข้อมูลเกี่ยวกับหลักการทำงานของ Navigation และ Search

1.4.7 ทำการศึกษาค้นคว้าข้อมูลเกี่ยวกับหลักการสำคัญของ Ontology

1.4.8 ทำการพัฒนาเว็บแอปพลิเคชั่นโดยการแสดงผลจากฐานข้อมูล DBLP

1.4.9 พัฒนาระบบ Navigation

1.4.10 พัฒนาระบบ Search

1.4.11 ทำการพัฒนาเว็บแอปพลิเคชั่นโดยการการนำหลักการของ Ontology เพื่อช่วยในการค้นหาข้อมูล

1.4.12 ทดลองนำแอปพลิเคชั่น ไปใช้งานจริงแล้ววิเคราะห์จุดผิดพลาดและปรับปรุงแก้ไข

1.4.13 สรุปผลการทดลองและจัดทำรูปเล่มโครงงาน

แผนการดำเนินงาน

ตารางที่ 1.1 แผนการดำเนินงาน

กิจกรรม	พ.ศ. 2552						พ.ศ. 2553	
	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.
1. ทำการศึกษาค้นคว้าข้อมูลเกี่ยวกับระบบฐานข้อมูลประเภทต่างๆ								
2. ทำการศึกษาค้นคว้าข้อมูลเกี่ยวกับการกระทำการกับระบบฐานข้อมูลโดยใช้ LINQ								
3. ทำการค้นคว้าหลักการพัฒนาเว็บแอปพลิเคชั่นโดย ASP.NET								
4. ทำการพัฒนาโครงสร้างความสัมพันธ์ของเว็บแอปพลิเคชั่น								
5. ทำการศึกษาค้นคว้าข้อมูลเกี่ยวกับแหล่งเอกสารอ้างอิง DBLP								
6. ทำการศึกษาค้นคว้าข้อมูลเกี่ยวกับหลักการทำงานของ Navigation และ Search								
7. ทำการศึกษาค้นคว้าข้อมูลเกี่ยวกับหลักการสำคัญของ Ontology								
8. ทำการพัฒนาเว็บแอปพลิเคชั่นโดยการแสดงผลจากฐานข้อมูล DBLP								
9. พัฒนาระบบ Navigation								
10. พัฒนาระบบ Search								
11. ทำการพัฒนาเว็บแอปพลิเคชั่นโดยการการนำหลักการของ Ontology เพื่อช่วยในการค้นหาข้อมูล								
12. ทดลองนำแอปพลิเคชั่น ไปใช้งานจริงแล้ววิเคราะห์จุดผิดพลาดและปรับปรุงแก้ไข								

กิจกรรม	พ.ศ. 2552						พ.ศ. 2553	
	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.
13. สรุปผลการทดลองและจัดทำรูปเล่มโครงงาน								

1.5 ผลที่คาดว่าจะได้รับ

Web Application ที่พัฒนาขึ้นสามารถตอบสนองความต้องการของผู้ที่สนใจในงานวิชาการทางด้าน Computer Since and Engineering โดยมีระบบ Navigation และ Search ช่วยในการสืบค้นหาข้อมูลตามที่ผู้ใช้งานต้องการ และระบบสามารถที่จะนำความรู้ทาง Ontology มาช่วยเพิ่มประสิทธิภาพให้กับ Navigation และ Search ในการค้นหาข้อมูลได้ดีกว่าการค้นหาข้อมูลแบบ Keyword based approach

1.6 รายละเอียดงบประมาณของโครงงาน

1.6.1 ค่าใช้จ่ายระหว่างจัดทำโครงงาน	1,300	บาท
1.6.2 ค่าใช้จ่ายในการจัดทำรูปเล่มรายงาน	700	บาท
รวม	2,000	บาท

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

จากบทที่ผ่านมาทำให้มีความจำเป็นที่จะต้องศึกษาเกี่ยวกับหลักการและทฤษฎีในการพัฒนาเว็บแอปพลิเคชั่น ซึ่งสิ่งที่จะศึกษาในบทนี้ได้แก่ เรื่องของระบบฐานข้อมูลซึ่งมีหลากหลายชนิดและมีความแตกต่างกัน ต่อมาจะศึกษาในเรื่องของการกระทำการกับข้อมูล เพื่อใช้ในการแสดงผล เพิ่ม ลบ แก้ไขข้อมูล และ การศึกษาค้นคว้าเกี่ยวกับเทคโนโลยีที่เกี่ยวกับการพัฒนาเว็บแอปพลิเคชั่น รวมทั้งการศึกษาความสัมพันธ์ระหว่าง Model View และ Controller

2.1 ฐานข้อมูล (Database)

ในปัจจุบันคอมพิวเตอร์เป็นสิ่งที่มีความสำคัญ และเป็นที่นิยมใช้กันอย่างแพร่หลาย เพราะคอมพิวเตอร์สามารถที่จะประมวลผลได้อย่างรวดเร็วมีหน่วยความจำที่สูง และสามารถที่จะเชื่อต่อเข้ากับเครือข่าย ทำให้สามารถติดต่อข้อมูลกันได้อย่างทันท่วงที และประกอบกับในขณะนี้ธุรกิจต่างๆมีการขยายตัวเป็นอย่างมากจึงทำให้มีความต้องการที่จะนำคอมพิวเตอร์มาเพื่อเก็บข้อมูลต่างๆ ที่มีปริมาณมาก ประกอบกับความความซับซ้อนและยากต่อการทำงานด้านเอกสาร จึงได้มีการนำคอมพิวเตอร์เข้ามาจัดเก็บและประมวลผล ซึ่งทำให้การประมวลผลเป็นไปอย่างรวดเร็วมีความสะดวกต่อการเรียกใช้งาน อย่างไรก็ตามการออกแบบก็ต้องคำนึงถึงความถูกต้องและมีประสิทธิภาพด้วย

ระบบฐานข้อมูลเป็นระบบที่ทำหน้าที่ในการจัดเก็บข้อมูลต่างๆอย่างเป็นระบบ โดยสามารถแยกความแตกต่างสำหรับสิ่งที่กำลังพิจารณาได้ และลดความซ้ำซ้อนของข้อมูลและให้ข้อมูลเป็นไปในแนวทางเดียวกันสามารถหาความสัมพันธ์ของข้อมูลที่มีอยู่ได้ ซึ่งจะทำให้ข้อมูลน่าเชื่อถือเป็นมาตรฐาน และมีความปลอดภัย รูปแบบฐานข้อมูลอาจมีลักษณะที่แตกต่างกัน ซึ่งมีความสามารถในการใช้งานที่แตกต่างกันดังนั้นเราจึงควรปรับใช้ชนิดของฐานข้อมูลให้เหมาะสมกับลักษณะงานและความต้องการของระบบ ซึ่งระบบฐานข้อมูลแบ่งออกเป็นประเภทต่างๆ ดังนี้

2.1.1 ไฟล์ Text (Text Files)

เป็นลักษณะของการจัดเก็บข้อมูลลงใน Text File ซึ่งรูปแบบการจัดเก็บข้อมูลนั้นไม่มีรูปแบบตายตัว จะขึ้นอยู่กับลักษณะของการเรียกใช้งานที่ผู้ใช้สร้างขึ้น ซึ่งข้อดีของการใช้ระบบฐานข้อมูลแบบ Text File คือ จะมีความรวดเร็วในการเรียกอ่านข้อมูล

2.1.2 ฐานข้อมูลเชิงสัมพันธ์ (Relational Database)

Relational Database หมายถึงกลุ่มของข้อมูลที่ถูกเขียนอยู่ในรูปแบบของตาราง ซึ่งจะแบ่งออกเป็นข้อมูลที่อยู่ในลักษณะแนวนอน (Row) เรียกว่า Record และ ข้อมูลที่อยู่ในลักษณะแนวตั้ง (Column) เรียกว่า Attribute ซึ่งแต่ละตารางจะมีความสัมพันธ์ซึ่งกันและกัน ซึ่งมีส่วนประกอบดังนี้

1 .Entity เป็นสิ่งที่อ้างอิงถึงสิ่งต่างๆ ที่เราสนใจไม่ว่าจะเป็น บุคคล สถานที่ สิ่งของ ซึ่งแต่ละความสนใจจะถูกแยกออกเป็น Attribute ต่างๆที่เป็นรายละเอียดของสิ่งนั้นๆ

2. Attribute เป็นข้อมูลที่แสดงรายละเอียดของ Entity เช่น Attribute ของลูกค้าที่ประกอบไปด้วย ชื่อ ที่อยู่ เพศ เบอร์โทร และอื่นๆ ซึ่งเป็นการบอกรายละเอียดของลูกค้าคนนั้นๆ

3. Record เป็นการนำ Attribute หลายๆ Attribute มารวมกันเพื่อเป็นข้อมูลที่เจาะจงถึงสิ่งใดสิ่งหนึ่ง เช่น ข้อมูลลูกค้าก็ประกอบไปด้วย Attribute ชื่อ ที่อยู่ เพศ เบอร์โทร ซึ่งจะเป็นสิ่งที่เฉพาะของลูกค้าคนนั้น

4. Table เป็นการนำ Record หลายๆ Record มารวมกันเป็น Table เช่น การที่มีลูกค้าหลายๆคนก็จะนำมารวามกันเป็น Table เดียวกัน

5. Relationships เป็นความสัมพันธ์ของ Entity ซึ่งอยู่ในลักษณะต่างๆ เช่น ลูกค้าสามารถสั่งซื้อสินค้าได้หลายชนิด พนักงานหลายคนสามารถเพิ่มสินค้าได้หลายชนิด ซึ่งความสัมพันธ์จะสามารถแบ่งออกได้ 3 ชนิด ดังนี้

One To One Relationships เป็นความสัมพันธ์ที่หนึ่ง Record ในตารางสามารถมีความสัมพันธ์กับอีกหนึ่ง Record ในอีกตารางเท่านั้น ตารางสินค้ากับตารางจำนวนสินค้า

One To Many Relationships เป็นความสัมพันธ์ที่หนึ่ง Record ในตารางสามารถมีความสัมพันธ์กับอีกหนึ่งหลาย Record ในอีกตาราง เช่น ลูกค้าสามารถที่จะมีใบสั่งซื้อสินค้าได้หลายใบ แต่ใบสั่งซื้อสินค้าสามารถที่จะมีลูกค้าได้แค่เพียงคนเดียว

Many To Many Relationships เป็นความสัมพันธ์หลาย Record ในตารางสามารถมีความสัมพันธ์กับอีกหนึ่งหลาย Record ในอีกตาราง เช่น ใบสั่งซื้อสินค้ามีสินค้าได้หลายชนิด และสินค้าชนิดนั้นสามารถมีได้ในหลายใบสั่งซื้อ

2.1.3 ไฟล์ XML (XML File)

เนื่องจากทุกวันนี้โลกของเรามีการนำเทคโนโลยีสารสนเทศเข้ามาใช้ในชีวิตประจำวันมากขึ้น และได้เข้ามาเป็นส่วนหนึ่งของการทำงาน ที่เห็นได้ชัดคือ เทคโนโลยีบนอินเทอร์เน็ต ไม่ว่าจะเป็นการเข้าไปเยี่ยมชมเว็บไซต์ต่าง ๆ หรือการส่งจดหมายอีเล็กโทนิกส์ที่เรียกว่า E-mail ซึ่งกำลังเป็นมาตรฐานของการติดต่อสื่อสารสำหรับอนาคต ทำให้ต้องมีการคิดเพื่อพัฒนาให้มีความก้าวหน้ามากขึ้น การเขียนเว็บไซต์ในปัจจุบันนี้ โดยปกติแล้วสิ่งที่จะใช้สร้างเว็บได้คือ ภาษา HTML ที่เรารู้จักกันดีเท่านั้น แต่ว่า

วันนี้โลกได้มีการพัฒนาเทคโนโลยีใหม่ที่ใช้สำหรับการเขียนเว็บ นั่นคือ XML ซึ่งเป็นสิ่งที่หลายผลิตภัณฑ์ให้การสนับสนุน

XML ย่อมาจากคำว่า Extensible Markup Language เป็นภาษาที่ใช้กำหนดรูปแบบของคำสั่งที่เรียกว่า Meta Data ซึ่งจะใช้สำหรับกำหนดรูปแบบของคำสั่ง Markup ต่าง ๆ แต่มีข้อแตกต่างกับ HTML ที่เป็น Markup Language ซึ่ง XML ได้รับการพัฒนามาจาก SGML (Standard Generalized Markup Language) ที่เป็นข้อกำหนดในการสร้างหรือจัดทำเอกสารในรูปแบบอิเล็กทรอนิกส์ที่กำหนดโดย W3C หรือ World Wide Web Consortium ซึ่งเป็นภาษาที่นิยมใช้และได้รับการพัฒนาให้มีประสิทธิภาพสูงสุดในการทำงานบนเว็บ โดย XML จะประกอบด้วย 3 ส่วนพื้นฐานด้วยกัน คือ เอกสารข้อมูล (Data document) เอกสารนิยามความหมาย (definition document) และ นิยามภาษา(definition language) [1]

การใช้งาน XML จำเป็นต้องใช้ร่วมกับ Style Sheet หรือมาตรฐานอื่น ๆ เพราะ XML เพียงแต่กำหนดรูปแบบของ Tag เท่านั้น ไม่ได้กำหนดว่า Tag จะแสดงผลแบบใด ดังนั้น หากเอาข้อมูลในรูปแบบ XML ไปแสดงผลในอุปกรณ์ชนิดใดก็ตาม จะต้องกำหนดวิธีแสดงผลของอุปกรณ์นั้นด้วย นอกจากนี้ XML ยังสนับสนุนตัวอักษรภาษานานาชาติ โดยใช้มาตรฐาน ISO 10646 ส่วนประกอบของ XML มีดังนี้ [17]

1. อิลิเมนต์
2. tag
3. แอตทริบิวต์
4. Entity

อิลิเมนต์ (Element) และ tag

อิลิเมนต์ใน XML จะหมายถึงส่วนของข้อมูลที่ประกอบไปด้วย tag เปิดและ tag ปิดรวมกัน ยกตัวอย่างเช่น อิลิเมนต์ note จะมี tag ดังนี้คือ

<note> </note>

สังเกตว่าการใช้งาน tag ของ XML จะมีลักษณะเหมือนกับการใช้งาน tag ของ HTML แต่ว่า XML จะสามารถสร้างอิลิเมนต์ใหม่ได้เองตามความต้องการของผู้ใช้

Attribute

นอกจากแท็กแล้วยังมี สิ่งที่เรียกว่า attribute ด้วยโดยที่มีรูปแบบดังนี้

<student name="example_name"></student>

<student name='example_name'></student>

จะเห็นว่าทั้งสองแบบมีความเหมือนกันแตกต่างกันเล็กน้อยคือใช้เครื่องหมาย " กับ ' ซึ่งสามารถใช้ได้ทั้งคู่

ตัวอย่าง Tag ของ XML จะมีลักษณะดังนี้

<book mdate="2002-01-03" key="books/aw/Stroustrup91">

<author>Bjarne Stroustrup</author>

<title>The C++ Programming Language, Second Edition</title>

<publisher>Addison-Wesley</publisher>

<year>1991</year>

<isbn>0-201-53992-6</isbn>

</book>

จากตัวอย่าง XML ข้างต้นจะเป็นลักษณะของ Tag เกี่ยวกับหนังสือหนึ่งเล่มที่แสดงรายละเอียดเกี่ยวกับหนังสือ 1 เล่ม อธิบายโดยละเอียดดังนี้

<book mdate="2002-01-03" key="books/aw/Stroustrup91">....</book>

ในส่วนนี้จะเป็นส่วนที่แยกความแตกต่างของหนังสือแต่ละเล่ม ซึ่งจะมี Attribute ของ วันที่ และ key ของหนังสือเล่มนั้นๆ

<author>Bjarne Stroustrup</author>

<title>The C++ Programming Language, Second Edition</title>

<publisher>Addison-Wesley</publisher>

<year>1991</year>

<isbn>0-201-53992-6</isbn>

ส่วนนี้จะเป็นส่วนที่แสดงรายละเอียดของหนังสือซึ่งประกอบไปด้วย ผู้แต่ง , ชื่อหนังสือ , สำนักพิมพ์ , และ รหัสของหนังสือเล่มนั้นๆ จะเห็นได้ว่าการใส่ Tag แต่ละ Tag จะเริ่มต้นด้วย <Tag> และปิดท้ายด้วย </Tag> เสมอ

2.2 ภาษาสอบถาม (Query Languages)

จากหัวข้อที่ 2.1 ทำให้ทราบว่าการจัดเก็บข้อมูลสามารถจัดเก็บได้หลายรูปแบบ ซึ่งแต่ละรูปแบบมีภาษาที่ใช้ในการสืบค้นหาข้อมูลที่แตกต่างกัน ดังต่อไปนี้

2.2.1 ภาษาสอบถามเชิงโครงสร้าง (Structure Query Language : SQL)

SQL หรือ Structure Query Language เป็นชุดคำสั่งที่ใช้จัดการข้อมูลในฐานข้อมูล ชุดคำสั่ง SQL นิยมใช้มากในระบบฐานข้อมูลซึ่งอยู่ในรูปแบบของตารางที่มีความสัมพันธ์กัน หรือที่เรียกว่า Relational Database ดังที่กล่าวในหัวข้อ 2.1.2 ชุดคำสั่งที่ใช้ใน การจัดการกับฐานข้อมูลที่สำคัญๆ มี 4 คำสั่ง คือ

1. Select เป็นคำสั่งสำหรับการดึงข้อมูลจากฐานข้อมูล SQL มาแสดง โดยสามารถที่จะกำหนดเงื่อนไขสำหรับการแสดง โดยใช้คำสั่ง WHERE เพื่อดึงข้อมูลเฉพาะที่ต้องการใช้งาน ซึ่งมีรูปแบบดังนี้

Select *คอลัมน์ที่เลือก* From *ตารางที่ต้องการอ้างถึง* Where *เงื่อนไขสำหรับการเลือกแสดงผล*

2. Update เป็นคำสั่งสำหรับการแก้ไขข้อมูลจากฐานข้อมูล ซึ่งจะเป็นการระบุเป็นคอลัมน์ที่ต้องการแก้ไข ซึ่งอาจจะมีเงื่อนไขสำหรับข้อมูลที่ต้องการจะแก้ไขโดยใช้คำสั่ง WHERE ซึ่งมีรูปแบบดังนี้

Update *ตารางที่อ้างถึง* Set [*ชื่อคอลัมน์=ข้อมูลที่ต้องการแก้ไข*] Where *เงื่อนไขสำหรับการเลือกแก้ไข*

3. Delete เป็นคำสั่งสำหรับการลบข้อมูลซึ่งเป็นการลบออกทีละแถว (Row) ซึ่งจะมีการระบุเงื่อนไขในการลบโดยคำสั่ง WHERE ซึ่งมีรูปแบบดังนี้

Delete From *ชื่อตารางที่อ้างถึง* Where *เงื่อนไขสำหรับการลบ*

4. Insert เป็นคำสั่งสำหรับการเพิ่งข้อมูลลงตารางโดยการเพิ่มทีละแถว (Row) ซึ่งสามารถที่จะระบุคอลัมน์ที่ต้องการที่จะเพื่อข้อมูลในแถวๆนั้นได้ ซึ่งมีรูปแบบดังนี้

Insert Into *ชื่อตารางที่อ้างถึง [ชื่อคอลัมน์ที่ต้องการเพิ่มข้อมูลในแถวนั้น]* Values [*ค่าที่ต้องการใส่ให้คอลัมน์นั้น*]

2.2.2 เอกซ์พาท (XPATH)

XPATH เป็นภาษาที่ใช้กระทำการกับข้อมูล XML เพื่อทำการดึงข้อมูลโดยการระบุ path ที่ต้องการแสดงและสามารถระบุเงื่อนไขตามที่ต้องการได้ ดังนี้

"Book" เป็นการเข้าถึง node ของ Book โดยจะทำการแสดงทุกๆ child node ของ Book

"/Book" เป็นการเข้าถึง node ของ Book ที่อยู่ถัดจาก root node

"//Book" เป็นการเข้าถึง node ของ Book ทุกๆตัวในไฟล์ XML

"@cost" จะเป็นการเข้าถึง node ต่างๆที่มี attribute ที่ชื่อว่า cost

การใช้เงื่อนไขในการระบุ path

"//Book[@cost>10]" เป็นการเข้าถึง node Book ที่มี attribute ของ cost มากกว่า 10

2.2.3 เอกซ์คิวรี (XQuery)

XQuery นั้นถูกพัฒนาโดยมีพื้นฐานมาจากภาษา SQL ซึ่งใช้งานได้ง่ายและเป็นที่ยอมรับจึงทำให้ XQuery สามารถทำความเข้าใจและใช้งานได้ง่ายเช่นเดียวกัน โดย XQuery เป็นภาษาสืบค้นสำหรับ XML ในรุ่นหลัง ซึ่งได้รับอิทธิพลมาจากภาษาสืบค้นสำหรับ XML ในรุ่นแรกๆ และมาตรฐานอื่นๆ ที่เกี่ยวข้อง ซึ่ง XQuery จะใช้แนวคิดของ path expression ในการระบุไปยังข้อมูลตำแหน่งต่างๆ ใน

เอกสารที่กำลังสนใจ โดย path expression นั้นจะอ้างอิงจากมาตรฐานของ XPath โดยลักษณะของการ Query เป็นดังนี้

FOR เป็นการระบุไปยังสิ่งที่สนใจ โดยใช้ path expression

LET เป็นการระบุค่าให้กับตัวแปลเพื่อช่วยการเสริมการทำงานของ FOR

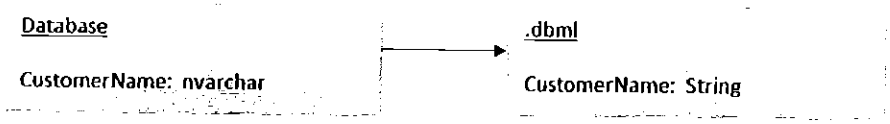
WHERE เป็นการตรวจสอบเงื่อนไขจากตัวแปรที่ระบุมาจาก FOR

2.2.4 Language Integrated Query (LINQ)

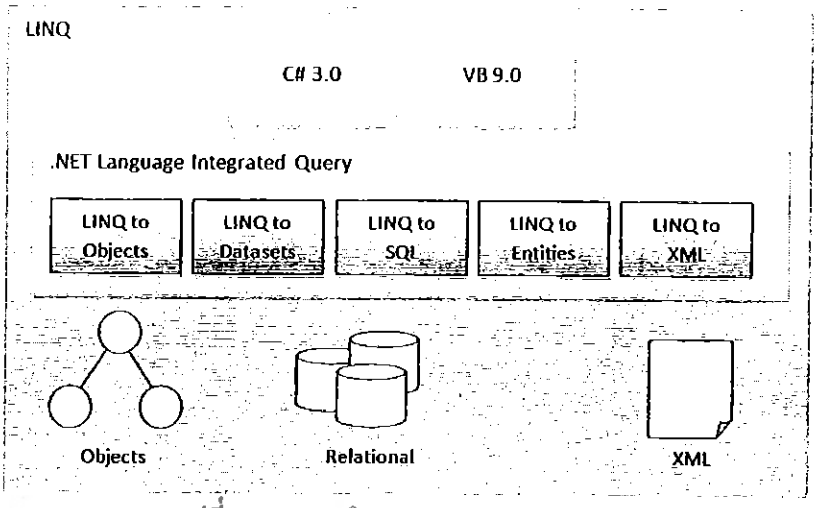
LINQ หรือ Language Integrated Query เป็นภาษาที่ไมโครซอฟต์สร้างขึ้น เพื่อช่วยให้การ "สอบถามข้อมูล" ซึ่งอยู่ในรูปแบบที่หลากหลาย [2]

จากหัวข้อที่ 2.1 เราทราบได้ว่าข้อมูลสามารถถูกจัดเก็บอยู่ในรูปแบบที่หลากหลายไม่ว่าจะเป็น Text File, XML File หรือข้อมูลที่อยู่ในรูปแบบของตารางจากที่กล่าวมาข้างต้น การที่เราจะต้องแยกภาษาสำหรับการดึงข้อมูลนั้น เป็นสิ่งที่ไม่สะดวกเพราะทำให้ผู้ใช้งานจะต้องเรียนรูปในหลายๆภาษา ทำให้ไมโครซอฟต์เล็งเห็นความสำคัญของสิ่งนี้จึงได้ พัฒนา LINQ ขึ้นเพื่อทำให้การสอบถามข้อมูลนั้นสามารถทำได้กับข้อมูลในหลายชนิดแต่ให้เพียงภาษาเดียว LINQ ไม่ได้เข้ามาทำหน้าที่แทนภาษา SQL แต่ LINQ จะทำหน้าที่เข้ามาเสริมเติมเต็มส่วนที่ SQL ไม่สามารถทำได้ [2]

LINQ เป็น O/RM (Object-relational mapping) ที่จะImplement ลงใน Visual Studio Orcas(VS2008) เป็นเทคนิคในการเขียนโปรแกรมเพื่อทำการแปลงโครงสร้างจาก Database ให้มาอยู่ในรูปของ Object-Oriented และในการทำ Mapping ออกมาจะต้องทำให้ Type ทางฝั่ง Database และทางฝั่งของ Object-Oriented มีความสอดคล้องกันโดยสร้าง "Virtual object database" ขึ้นมาซึ่งจะต้องใช้เครื่องมือที่เป็น ORM Tools ในการสร้าง Virtual object database นอกจากนั้นตัว ORM Tools จะสร้าง Class ออกมาแทน Object ต่างๆ ใน Database เช่น Table, View และสร้าง Function, Method ต่างๆ เพื่อใช้ในการ Select, Insert, Delete, Update ข้อมูลต่างๆ เพื่อจัดการกับข้อมูลใน Database ใน LINQ To SQL สิ่งที่เป็น "virtual object database" ก็คือ Linq to SQL File (.dbml) ดังรูปที่ 2.1 และการ Select ข้อมูลก็จะใช้ Query Syntax ของ LINQ เข้ามาทำในส่วนนี้ ส่วนการทำ Data Manipulation จะใช้ Method Submit Changes ในการส่งค่าต่างๆ คืนลงใน Database [3]



รูปที่ 2.1 แสดงความสัมพันธ์ Virtual object database และ Linq to SQL File (.dbml) [3]



รูปที่ 2.2 แสดงถึงส่วนประกอบ LINQ [3]

จารูปที่ 2.2 จะเห็นว่า LINQ แบ่งออกเป็น 5 ส่วนประกอบหลักๆ คือ

1. LINQ to Objects เพื่อใช้ LINQ ติดต่อกับ Object อื่นๆ
2. LINQ to Datasets
3. LINQ to SQL
4. LINQ to Entities ก็ใช้ในการติดต่อกับ Relational Database
5. LINQ to XML ใช้ในการติดต่อกับ XML Document โดยเฉพาะ

ในที่นี้จะแบ่ง LINQ เป็นประเภทการสอบถามเป็น 3 กลุ่มใหญ่ๆ คือ

2.2.4.1 LINQ to Object

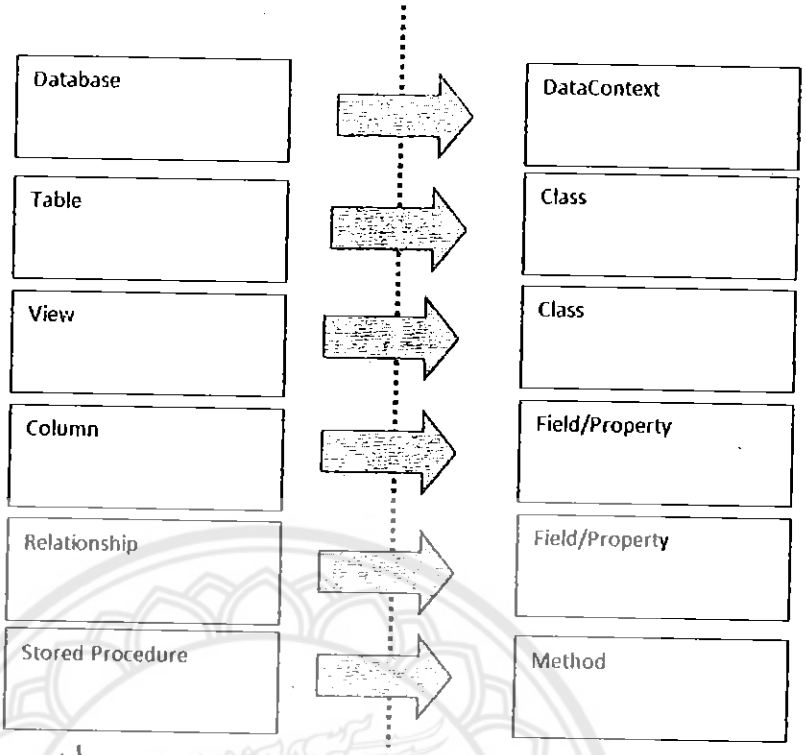
ทำหน้าที่สอบถามข้อมูลที่เก็บอยู่ในรูปแบบของ Object ต่างๆ เช่น เก็บใน Array, ใน Class ต่างๆ เป็นต้น ไม่มีความซับซ้อน

2.2.4.2 LINQ to SQL ,LINQ to Datasets ,LINQ to Entities

ทำหน้าที่สอบถามข้อมูลที่เก็บในรูปแบบด้านฐานข้อมูล ใช้ในการติดต่อกับ Relational Databaseที่กล่าวมาในหัวข้อ 2.1.2 [4]

การทำ Mapping ของ LINQ to SQL

จากรูปที่ 2.3 การทำ Mapping ของ LINQ to SQL ฝั่งซ้ายจะแทนโครงสร้างของ Database และทางฝั่งขวาคือ Object-Oriented



รูปที่2.3 แสดงการทำ Mapping ของ LINQ to SQL [5]

โดยการทำ Mapping จาก Database จะมีส่วนที่เพิ่มเติมคือ Data Context ที่จะเป็นส่วนที่ใช้ในการดึงข้อมูลจาก Database และทำการเปลี่ยนแปลงแก้ไขหรือกระทำการต่างๆไปยัง Database โดย Data Context ได้เหมือนกับ ADO.NET Connection ทั่วไป

หน้าที่หลักของ Data Context คือ

1. การสร้างConnectionไปยัง Database
2. ทำการแปลง Query Syntax ให้เป็นคำสั่ง SQL เพื่อ Execute ไปยัง Table
3. ทำการแปลง Object ที่ต่างๆ ที่ถูก Query ขึ้นมา ให้กลับลงไปยัง Database ในรูปแบบที่ Database Modeling (.dbml) ได้ทำการ Mapping เข้ามา

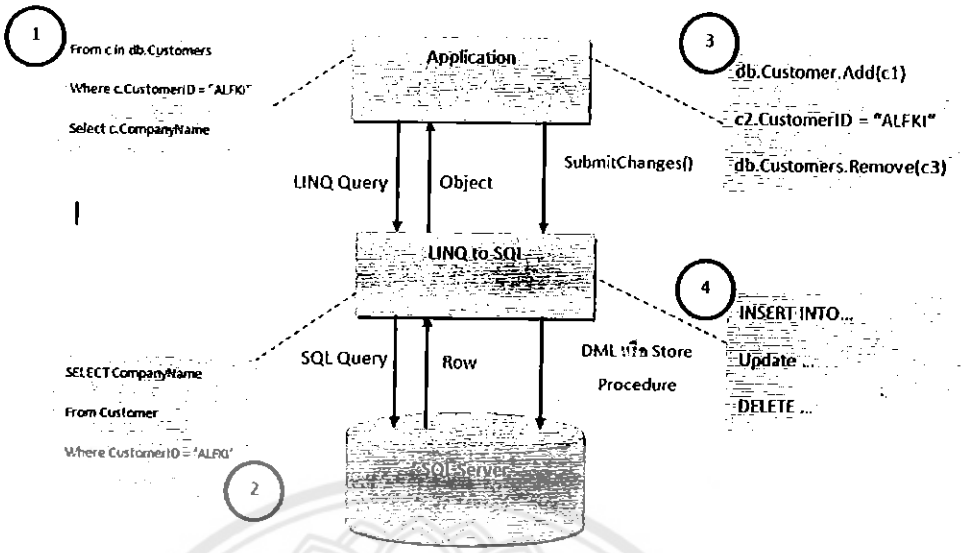
จากรูปที่ 2.4 เป็นการทำงานของ LINQ to SQL

1.ทำการเขียนโปรแกรมด้วย Query Syntax ซึ่งเป็นภาษาที่ใช้ในการ Query สำหรับ LINQ โดยเฉพาะ แล้วทำการ Select ข้อมูล

2. LINQ จะทำการแปลง Query Syntax ให้อยู่ในรูปของ SQL ซึ่งจะทำการ Execute ข้อมูลจากตารางขึ้นมา ซึ่งข้อมูลที่ถูกเลือกขึ้นมานั้นจะถูกแปลงเป็น Object ให้กับ Application

3. ถ้าทางฝั่ง Application ก็จะมีการปรับเปลี่ยน Property หรือเพิ่ม Entity Object ต่างๆ เข้าไปที่ Data Context เสร็จแล้วทำการเรียก Method Submit Changes ()

4. LINQ to SQL ก็จะทำการแปลงกลุ่ม Entity เป็นคำสั่ง SQL ที่เป็น DML หรือ Stored Procedure กลับไปยัง SQL Server



รูปที่2.4 แสดงการทำงานของ LINQ to SQL [3]

2.2.4.3 LINQ to XML

ใช้ในการติดต่อกับ XML Document โดยเฉพาะ ทำหน้าที่สอบถามข้อมูลที่เก็บอยู่ในรูปแบบของแหล่งข้อมูล XML โดยปกติแล้วในการ คิวรีข้อมูลจากไฟล์ XMLจะต้องใช้ภาษามาตรฐานที่เรียกว่า ซึ่งจะเป็นการระบุ Path และเงื่อนไขข้อมูลที่ต้องการ แต่สำหรับ LINQ to XML สามารถที่จะคิวรีข้อมูลหรือทำการแก้ไขข้อมูลได้โดยไม่จำเป็นต้องใช้ XPATH นิพจน์ XPATH ใช้ในการเชื่อมต่อ ตัวควบคุมกับเขตข้อมูลและกลุ่มใน แหล่งข้อมูล ของแม่แบบฟอร์ม นิพจน์ XPATH ยังใช้สำหรับการตรวจสอบความถูกต้องของข้อมูลและนิพจน์การจัดรูปแบบตามเงื่อนไข [4]

2.3 เทคโนโลยีในการพัฒนา Web Applications

อินเตอร์เน็ต คือ เครือข่ายคอมพิวเตอร์ขนาดใหญ่ที่มีการเชื่อมโยงกันอยู่ทั่วโลก มีการใช้อินเตอร์เน็ตเพื่อการสื่อสารกันอย่างแพร่หลาย และมีการขยายตัวเพิ่มขึ้นอย่างรวดเร็วบริการที่มีอยู่ในอินเตอร์เน็ตมีอยู่หลากหลาย แต่ที่ได้ใช้กันบ่อย ๆ คือ World Wide Web (www) เป็นบริการแสดง Web Page สามารถแสดงได้ทั้งภาพนิ่ง, ตัวอักษร, เสียง และภาพเคลื่อนไหว นอกจากนี้ยังมีการบริการอื่น ๆ อีกบนอินเตอร์เน็ต เช่น อีเมล์, บริการ FTP (Files Transfer Protocol) เป็นต้น

ระบบเครือข่ายอินเตอร์เน็ต ปัจจุบันมีจุดเชื่อมต่ออยู่ทั่วโลก ซึ่งจะเป็นประโยชน์ต่อการติดต่อสื่อสารกันอย่างกว้างขวาง ซึ่งความสามารถของอินเตอร์เน็ตนั้นจะช่วยอำนวยความสะดวกด้านต่างๆหลากหลายด้านไม่ว่าจะเป็น ภาพนิ่ง , ตัวอักษร, เสียงและภาพเคลื่อนไหว และนอกจากนี้ยังมีการบริการอื่นๆที่ช่วยอำนวยความสะดวกในการทำกิจกรรมต่างๆ ให้สะดวกรวดเร็วอีกด้วย

จากข้างต้นจะเห็นว่าประโยชน์ของอินเตอร์เน็ตนั้นมีความหลากหลาย ซึ่งทำให้การพัฒนาเว็บไซต์มีความสำคัญ ไม่ว่าจะเป็นเรื่องของการโฆษณา การประชาสัมพันธ์ ซึ่งเป็นสิ่งที่น่าสนใจเป็นอย่างยิ่ง เพราะเมื่อระบบเครือข่ายอินเตอร์เน็ตมีการถูกเชื่อมโยงมากขึ้น ก็จะส่งผลดีต่อธุรกิจในหลายๆแขนง

คำจำกัดความของ Web Application คือ Application ที่เข้าถึงด้วย Web Browser ผ่านเครือข่ายคอมพิวเตอร์อย่าง อินเทอร์เน็ต หรือ อินทราเน็ต" ซึ่งเทคโนโลยีในการพัฒนา Web Application แบ่งออกเป็นประเภทต่างๆ ดังนี้

2.3.1 เอเอสพี (Active Server Pages : ASP)

ASP คือเทคโนโลยีที่ใช้ในการพัฒนา Web Programming หรือ Web Application ซึ่งเป็นเทคโนโลยีที่ถูกพัฒนาโดย บริษัทไมโครซอฟต์ ซึ่งความสำคัญของการใช้ ASP คือ การที่เว็บไซต์ต่างๆ มีความจำเป็น ที่จะต้องโต้ตอบการทำงานกับผู้ใช้เป็นหลัก ไมโครซอฟท์จึงพัฒนาให้ ASP มีคุณสมบัติให้สามารถเขียนเว็บไซต์ที่มีการเคลื่อนไหวหรือเปลี่ยนแปลงได้ตลอดเวลา หรือที่เรียกว่า Dynamic Web Page ซึ่งเป็นจุดประสงค์หลักของ Web Programming ซึ่งจะทำให้การพัฒนามีความยืดหยุ่นสูง ซึ่งไม่จำกัดว่าผู้เขียนโปรแกรมจะต้องเขียนสคริปต์ภาษาใดภาษาหนึ่งเป็นเพียงอย่างเดียว [6]

ASP มีความสามารถในการพัฒนาและจัดการแอพพลิเคชั่นบนฝั่งเซิร์ฟเวอร์ (Web Server) ซึ่งการทำงานของ ASP นั้นจะเป็นโปรแกรมที่แปลงภาษา สำหรับการตีความ Web Page ซึ่งถูกเขียนโดยภาษาสคริปต์ ซึ่งการทำงานบน Web Browser เช่น Internet Explorer นั้นไม่สามารถที่แสดงผลได้โดยตรง จึงจำเป็นที่จะต้องใช้ ASP ในการแปลงข้อมูลให้อยู่ในรูปแบบของ HTML เพื่อที่จะสามารถทำให้ Web Browser สามารถที่จะนำไปแสดงผลได้ต่อไป [6]

2.3.1.1 การทำงานของ ASP

เมื่อมีการเรียกดู Web Page จะมีการร้องขอจาก Web Browser ไปยัง Web Server ที่เก็บ Web Page นั้นหลังจากนั้น Web Server จะแยกความแตกต่างระหว่างไฟล์ที่ถูกร้องขอว่าเป็นไฟล์ชนิดใด ถ้าเป็นไฟล์ทีอยู่ในลักษณะของ HTML ซึ่งเป็นไฟล์สกุล .html ก็จะถูกส่ง ไฟล์ไปให้โดยตรง แต่ถ้าเป็นการร้องขอ Web Page ที่มีลักษณะของ ASP ซึ่งเป็นไฟล์สกุล .asp ทางด้าน Server จะต้องแปลงไฟล์โดย ASP Interpreter ก่อนเพื่อทำให้ไฟล์นั้นสามารถที่จะถูกอ่านโดย Web Browser ฝั่ง Client ได้โดยสคริปต์จะไม่ถูกส่งมาด้วย แต่จะถูกส่งมาแสดงเพียงแค่ HTML เท่านั้น

2.3.1.2 ข้อดีของ ASP

1.ASP สามารถที่จะรองรับการทำงานแบบ Dynamic ซึ่งทำให้ Web Page นั้นสามารถเปลี่ยนแปลงการแสดงข้อมูลตามความต้องการต้องการของผู้ใช้ได้ ซึ่งต่างการทำงานของเว็บไซต์ทั่วไปที่จะแสดงข้อมูลแบบตายตัว

2. ทำให้การทำงานรวดเร็วขึ้น กล่าวคือการทำงานจะมีการประมวลผลข้อมูลทางฝั่ง Server แล้วส่งเพียงผลลัพธ์เพียงอย่างเดียวมาแสดงผล ซึ่งจะทำเสียเวลาในการรอข้อมูลน้อยลง จึงการใช้งานมีประสิทธิภาพขึ้น

3. ทำให้การแสดงข้อมูลมีความปลอดภัย เพราะการเขียนโปรแกรมในการติดต่อกับฐานข้อมูลโดยมากจะต้องมีการติดต่อกับ Directory ต่างที่เก็บฐานข้อมูล ซึ่ง ASP จะช่วยขจัดปัญหาที่จะเกิดขึ้นในการเข้าถึงฐานข้อมูลโดยไม่ได้รับอนุญาต โดยการแสดงผลส่วนที่เป็น Directory จะไม่ถูกแสดงในฝั่ง Client แต่จะแสดงเพียงแค่ข้อมูลที่อยู่ในฐานข้อมูลเท่านั้น

4. ขจัดปัญหาของเครื่อง Client ที่ต้องการใช้บริการโดยการทำงานของ ASP. จะส่งเพียงแค่ข้อมูลที่หรือผลลัพธ์ในรูปแบบของ HTML เท่านั้น ทำให้เครื่องไม่จำเป็นต้องมี speck ที่แตกต่างจากการดูเว็บไซต์ธรรมดาทั่วไป

2.3.1.3 เอกสาร ASP

การเขียนโปรแกรม เนื่องจากในไฟล์ของ ASP นั้นจะมีโค้ดในส่วนของ HTML และส่วนของ ASP รวมอยู่ด้วยกัน การที่เราจะแยกความแตกต่างระหว่าง โค้ดทั้งสองส่วนนั้นเราจะใช้ ASP Delimiters โดยจะมีอยู่ด้วยกัน 3 แบบ ดังนี้

1. เครื่องหมายที่ใช้กำหนดขอบเขตของโค้ด ASP ซึ่งเป็นเครื่องหมาย <% และ %> ที่แทรกอยู่ตามโค้ด HTML

<% <โค้ด คำสั่งการทำงานของ ASP> %>

2. เครื่องหมายที่ใช้แสดงค่าของนิพจน์หรือตัวแปร โดยการใส่ค่าตัวแปรหรือสิ่งที่ต้องการแสดงในส่วนของ <%= และ %>

<%=<สิ่งที่ต้องการแสดง>%>

3. การกำหนดขอบเขตของ ASP โดยการระบุภาษาที่จะใช้ โดยการใช้ Attribute คือ RUNAT=SERVER แทรกใน <Script> เพื่อเป็นการบ่งบอกว่า Script ที่อยู่ระหว่าง <Script> และ </Script> ต่อไปนี้จะทำงานที่ฝั่ง Server เท่านั้น ซึ่งจะไม่แสดงโค้ดให้เห็นทางฝั่ง Client ด้วย

แต่เนื่องจาก ASP นั้นสามารถที่จะรองรับการเขียนใน สคริปต์หลายๆภาษาทำให้เราจำเป็นที่จะต้องระบุว่าในแต่ละส่วนของสคริปต์นั้นใช้ภาษาอะไรในการพัฒนา โดยสามารถที่จะแทรกชนิดของสคริปต์ได้โดยการ Attribute คือ LANGUAGE = ภาษาที่ใช้ภายในสคริปต์ ซึ่งจะได้รูปแบบดังนี้

```
<Script Language=<ภาษาที่ต้องการจะใช้เขียน> RUNAT=SERVER>
<โค้ดคำสั่งการทำงาน>
</Script>
```

2.3.1.4 ความต้องการของระบบในการใช้งาน ASP

ในการที่จะพัฒนาและใช้งานแอพพลิเคชั่น ASP นั้น มีความต้องการทางด้านซอฟท์แวร์และฮาร์ดแวร์ ดังนี้

1. หน่วยประมวลผล 486 เป็นอย่างต่ำ (ถ้าจะให้ดีควรเป็นรุ่น Pentium ขึ้นไป)
2. แรมอย่างน้อย 32 MB (ถ้าจะให้ดีควรเป็น 48 MB ขึ้นไป)
3. พื้นที่ว่างในฮาร์ดดิสก์อย่างน้อย 150 MB
4. ระบบปฏิบัติการ Windows NT Server 4.0, Windows NT Workstation 4.0 หรือWindows 95/98/ME/2000/XP และต้องมีการติดตั้งโปรโตคอล TCP/IP ด้วย
5. โปรแกรม Web Server ที่สนับสนุน ASP เช่น Internet Information Service3.0/4.0/5.0/6.0 สำหรับ Windows NT และ Windows 2000/XP หรือ Personal Web Server (PWS) สำหรับ Windows 9X/ME
6. ถ้ามีการพัฒนาระบบฐานข้อมูลขึ้นใช้งาน เราต้องมีโปรแกรมฐานข้อมูลที่รองรับกับ ODBC ด้วย เช่น Microsoft Access, Microsoft SQL Server, My SQL เป็นต้น

2.3.2 พีเอชพี (Professional Home Page : PHP)

PHP (Professional Home Page) เป็นภาษาสคริปต์ซึ่งจะถูกเก็บอยู่ในไฟล์ .php เวลาใช้งานจะต้องอาศัยตัวแปลชุดคำสั่ง ภาษา PHP เป็นภาษาที่ถูกสร้างขึ้นเพื่อใช้งานร่วมกับเอกสาร HTML ซึ่งเป็นสคริปต์ที่จะช่วยให้ HTML อยู่ในลักษณะของ Dynamic มากขึ้นทำให้ HTML ทำงานได้อย่างมีประสิทธิภาพและมีความหลากหลาย

2.3.2.1 การทำงานของ PHP

PHP จัดว่าเป็นโปรแกรมฝั่ง Server (Server-Side Language) เนื่องจาโค้ด PHP จะถูกประมวลผลที่ฝั่ง Server ซึ่งต่างจากภาษาหรือเทคโนโลยีอย่างเช่น Java Script , Flash หรือ ActiveX ที่จะถูกประมวลผลโดยโปรแกรม Web Browser ที่ฝั่งผู้ใช้ ดังนั้นผู้ใช้จึงไม่มีทางเห็นโค้ด PHP ที่เราเขียนไว้ใน PHP page เพราะว่าโค้ดเหล่านี้จะถูกประมวลผลไม่จนหมดที่ฝั่ง Web Server แล้วให้ผลลัพธ์ออกมาเป็นข้อความธรรมดาและ Tag ในภาษา HTML หรือ Java script หรืออื่นๆ ที่ถูกส่งไปประมวลผลที่ฝั่ง Web Browser ด้วย [7]

2.3.2.2 ข้อดีของ PHP

มีผู้นิยมมาก เป็นภาษาที่ฟรี เขียนง่าย มีผู้ให้บริการ Server อยู่มาก Compiler มีขนาดเล็ก มี lib สำเร็จมากมาย มีความเร็วสูง มีการพัฒนาอย่างต่อเนื่อง [8]

2.3.2.2 เอกสาร PHP

การเขียน PHP มีลักษณะรูปแบบการเขียนเป็น HTML ผสมกับ tag ของ PHP แทรกกัน เช่น *ตัวอย่างการเขียน PHP*

File test.php

<html><body>

<% php

PHP Command1;

PHP Command2;

PHP Command3;

%>

</body>

</html>

2.3.2.4 ความต้องการของระบบในการใช้งาน PHP

1. เครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรม Web Browser เช่น Internet Explorer
2. โปรแกรม Web Server เช่น Apache Web Server
3. โปรแกรม PHP
4. โปรแกรมฐานข้อมูล เช่น My SQL

2.3.3 เจเอสพี (Java Server Page : JSP)

JSP เป็นสคริปต์อีกภาษาหนึ่ง ซึ่งเป็นทางเลือกที่น่าสนใจสำหรับการพัฒนาเว็บแอปพลิเคชั่น จุดเด่นที่สำคัญของ JSP อยู่ที่การใช้ภาษา Java ซึ่งเป็นเชิง OOP ที่ช่วยสามารถพัฒนาแอปพลิเคชั่นขนาดใหญ่และซับซ่อน ได้อย่างสะดวกและรวดเร็ว

2.3.3.1 การทำงานของ JSP

JSP (Java Server Page) เป็นเทคโนโลยีที่อยู่ในลักษณะของ Java เป็นการทำงานที่อยู่บนฝั่ง Server หรือการทำงานแบบ Server Side ซึ่งการทำงานจะเป็นไปในลักษณะการร้องขอของ Browser บนฝั่ง Client จากนั้น Server ก็จำการแปลง JSP เป็น Servlet ก่อนแล้วส่งกลับไปให้ Client ในรูปแบบที่ Browser ประมวลผลได้นั่นคือ HTML [9]

2.3.3.2 ข้อดีของ JSP

การพัฒนาเว็บแอปพลิเคชั่นด้วย JSP มีข้อดีต่างๆ ดังนี้ [8]

1. ทำงานโดยไม่ยึดติดแพลตฟอร์มใดๆ JSP ได้สืบทอดคุณสมบัติ ของ Java มาอย่างเต็มที่ โดยสามารถทำงานได้ในทุกแพลตฟอร์ม ไม่ว่าจะเป็น windows, linux , mac os
2. ใช้งาน Java API ได้หลากหลาย ซึ่ง Java API คือกลุ่มของคลาสที่ช่วยอำนวยความสะดวกในการใช้งาน ไม่ว่าจะเป็น การติดต่อกับ database, การรับส่ง อีเมล์ เป็นต้น
3. นำ Component กลับมาใช้ได้อีก ไม่ต้องเสียเวลาสร้างใหม่ ดังนั้นเราจึงไม่ต้องเสียเวลาเขียนสคริปต์ ใหม่เพื่อทำงานครั้งต่อไป จึงช่วยให้นักพัฒนาสามารถสร้างสรรค์ผลงานได้เร็วขึ้น
4. มีความยืดหยุ่นในการใช้งาน คือเราสามรถกำหลด Tag ใหม่ขึ้นมาใช้งานได้ รวมทั้งยังสามารถนำไปใช้งานร่วมกับ XML ได้เป็นอย่างดี
5. ความปลอดภัยสูง JSP มีระบบจัดการข้อผิดพลาดต่างๆ ไม่ว่าจะเป็นข้อผิดพลาดที่เกิดขึ้นระหว่างการเขียน program หรือข้อผิดพลาดที่เกิดขึ้นระหว่าง runtime ล้วนสามารถตรวจสอบข้อผิดพลาดได้ทันที่และตรงจุด

2.3.3.3 เอกสาร JSP

การเขียน JSP มีลักษณะรูปแบบการเขียนเป็น HTML ผสมกับ tag ของ Java แทรกกันเช่น

ตัวอย่างการเขียน JSP

File test.jsp

```
<html><body>
<%
JSP Command1;
JSP Command2;
JSP Command3;
%>
</body>
</html>
```

ไฟล์ที่เขียนด้วยเทคโนโลยีนี้ จะต้องกำหนดให้ไฟล์มีนามสกุลเป็น *.jsp เสมอ เพื่อบอกให้ server ทราบว่าจะต้องทำการประมวลผลกับไฟล์ที่ถูกร้องขอย่างไร การทำงานเมื่อทำการประมวลผล jsp ด้วย browser ๆ จะยังไม่สามารถทำการแสดงผลได้ในทันที แต่ jsp จะถูกทำการประมวลผลเป็น servlet ก่อน จากนั้น server จะทำการส่งผลลัพธ์ออกมาให้ browser เพื่อทำการแสดงผลอีกทีหนึ่ง (จะเห็นว่าการทำงานจะแตกต่างกับ HTML คือ JSP จะไม่สามารถทำการแสดงผลได้ทันที แต่จะต้องมีการประมวลผลบน server ก่อน) [9]

2.3.3.4 ความต้องการของระบบในการใช้งาน JSP

การเตรียมเครื่องการทำงานกับ jsp จะเหมือนกับการทำงานร่วมกับ java คือ ผู้พัฒนาระบบจะต้องเตรียม J2S DK , Web Server ตัวใดตัวหนึ่งเพื่อทำการจำลองเครื่องเป็น server และประมวลผล jsp

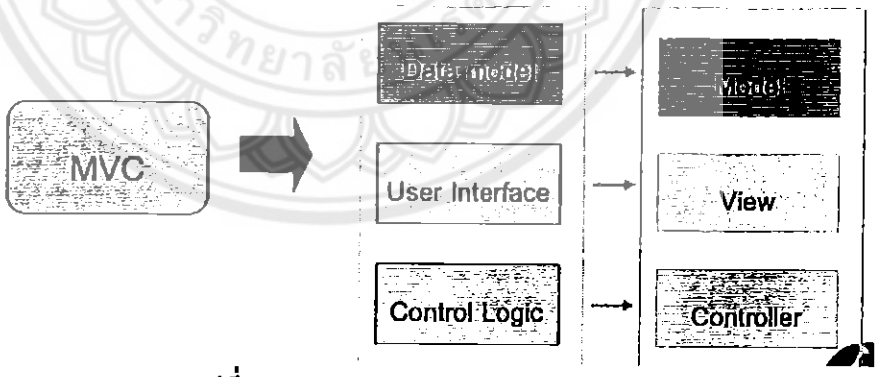
2.4 Model-view-controller (MVC)

ในการพัฒนาเว็บแอปพลิเคชั่นที่ดีนั้น เว็บแอปพลิเคชั่นจะต้องสามารถโต้ตอบกับผู้ใช้ได้ตรงตามความต้องการและมีความถูกต้อง อีกทั้งระบบจะต้องมีความสามารถที่จะจัดการกับ ฐานข้อมูลได้อย่างมีประสิทธิภาพ โดยหลักการของ MVC จะเป็นส่วนช่วยให้การพัฒนาเว็บแอปพลิเคชั่นนั้นสามารถตอบโต้กับผู้ใช้และกระทำการกับข้อมูลต่างๆได้อย่างเป็นระบบและมีความชัดเจนยิ่งขึ้น

2.4.1 ความรู้พื้นฐานเกี่ยวกับ MVC

MVC เป็นระบบการทำงานที่มีมาแล้วกว่า 20 ปี ซึ่งเป็น Framework ที่มีการแยกระบบการทำงานออกเป็น ส่วนสำหรับเก็บข้อมูล (model), ส่วนที่แสดงผลข้อมูล (view) และ ส่วนที่ใช้ติดต่อกับผู้ใช้ (controller) ซึ่งการทำงานทั้ง 3 ส่วนจะแยกออกจากกันอย่างชัดเจน

MVC คือสถาปัตยกรรมซอฟแวร์ (software architecture) ที่มีการแบ่งแยกระบบออก เป็น 3 ส่วนหลักๆ ได้แก่ Data model, User interface, and Control logic ดังรูปที่ 2.5



รูปที่2.5 แสดงส่วนประกอบของ MVC [10]

1. Model เป็นส่วนที่ทำหน้าที่ในติดต่อกับระบบฐานข้อมูลซึ่งเป็นการดำเนินการต่างๆ ไม่ว่าจะเป็นการเลือกข้อมูลมาแสดงผล การแก้ไข การเพิ่ม หรือ การลบ ซึ่งการดำเนินการต่างๆเหล่านี้จะเกิดขึ้นได้ก็ต่อเมื่อมีการเรียกใช้งานจากในส่วน Controller หรือ View [10] ในการทำงานของ Model นั้นจะเป็นการทำหน้าที่ของการติดต่อกับระบบฐานข้อมูลโดยตรง ซึ่งข้อมูลที่จะถูกเก็บนั้นจะสามารถ

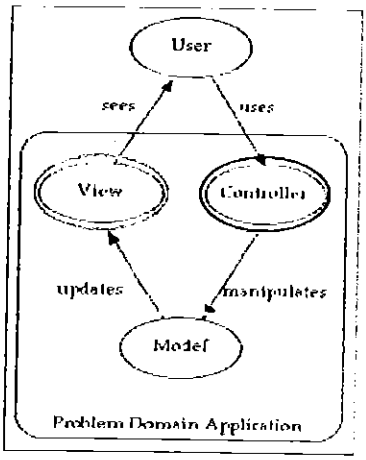
ถูกเก็บได้หลายรูปแบบดังที่กล่าวไปแล้วข้างต้น ยกตัวอย่างเช่นการคิวรีของภาษา SQL ซึ่งจะสามารถทำการ แสดงผล เพิ่ม หรือ ลบข้อมูลได้

2 . *View* เป็นส่วนที่ทำหน้าที่ในการแสดงผล ซึ่งเป็นส่วนที่ทำการเรียกข้อมูลจาก Model มาแสดงผล และยังทำหน้าที่ในการติดต่อกับผู้ใช้ ซึ่งทำให้ผู้ใช้สามารถที่จะตอบโต้กับระบบได้เป็นอย่างดี อย่างไรก็ตาม View ไม่ได้ทำหน้าที่ในตอบโต้กับผู้ใช้อย่างแท้จริง แต่จะต้องส่งต่อให้กับ Controller เพื่อดำเนินการต่อไป [10] สำหรับการพัฒนาในส่วนของ View จะมีส่วนภาษาที่ใช้ในการพัฒนายกตัวอย่างเช่น ASP ซึ่งจะทำหน้าที่ในการสร้างฟอร์มสำหรับตอบโต้การใช้งานกับผู้ใช้เพื่อที่จะทำให้ผู้ใช้สามารถใช้งานได้ง่าย ซึ่งจะเป็นการเรียกฟังก์ชั่นการทำงานต่างๆจาก View ไปยัง Controller เพื่อไปประมวลผลต่อไป

3. *Controller* เป็นส่วนที่ทำหน้าที่ควบคุมระบบการทำงาน ซึ่งได้รับการสั่งการมาจากผู้ใช้และทำการประมวลผลหรือการดำเนินการต่างๆเกี่ยวกับระบบฐานข้อมูล โดย Controller จะทำหน้าที่เป็นตัวกลางคอยควบคุมให้ Model และ View ดำเนินไปในทิศทางเดียวกัน [10] การทำงานของ Controller นั้นจะเป็นส่วนของ Logic ต่างที่ขับเคลื่อนการทำงานให้สามารถดำเนินไปสู่ผลลัพธ์ได้ โดยในส่วนของการทำงานนั้นจะอาศัยสคริปต์เพื่อที่จะเป็นตัวควบคุมการทำงาน ยกตัวอย่างเช่น C# , JSP ,PHP นอกจากนี้แล้วยังมีภาษาที่ถูกคิดค้นขึ้นมาใหญ่เพื่อใช้สำหรับการควบคุมการดำเนินงานของระบบฐานข้อมูลอีกด้วยนั่นก็คือ LINQ ที่จะทำหน้าที่ในการคิวรีข้อมูล แต่ไม่ได้ทำหน้าที่แทนคำสั่งสำหรับคิวรีฐานข้อมูลโดยตรง แต่ LINQ ทำหน้าที่เป็นตัวกลางระหว่าง Controller กับ Model ซึ่งจะถูกแปลงให้เป็นคำสั่งที่ถูกใช้สำหรับการกระทำการในขั้นตอนของ Model อีกทีหนึ่ง ซึ่งจะเห็นได้ว่าในการใช้ LINQ จะทำให้สามารถขจัดปัญหาการกระทำการสำหรับฐานข้อมูลที่มีหลากหลายชนิดได้

2.4.2 การทำงานของ MVC

จากรูปที่ 2.6 เป็นการทำงานร่วมกันของ Model View และ Controller โดยจะทำหน้าที่ต่างๆไปในทิศทางเดียวกัน เพื่อตอบสนองความต้องการของผู้ใช้ให้มากที่สุด



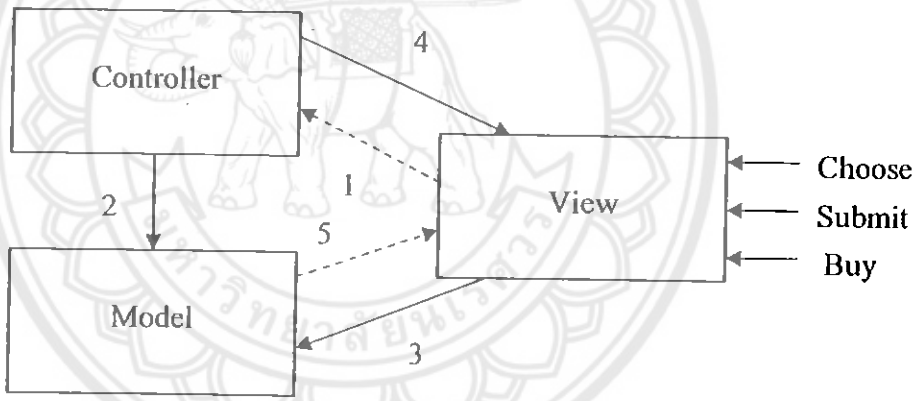
รูปที่2. 6 แสดงการทำงานร่วมกัน MVC [10]

เหตุการณ์สมมติ "ลูกค้าต้องการซื้อสินค้าผ่านทางเว็บไซต์" ดังรูปที่ 2.7 การทำงานแบบ MVC จะมีลักษณะดังนี้

1.เมื่อเมื่อลูกค้ากดปุ่ม Submit เพื่อซื้อสินค้า ซึ่งอยู่ที่ View จะส่งคำร้องนี้ไปยัง Controller **(1)** จะทำการรับการโต้ตอบจากปุ่ม Submit และสร้าง handler หรือ callback ขึ้นมา เพื่อใช้ติดต่อระหว่าง Layer

2.Controller ทำการเลือก Model ที่ตรงกับข้อมูลของลูกค้าที่ส่งข้อมูลเข้ามา **(2)** แล้ว Model ทำการตรวจสอบความถูกต้องของข้อมูลตามข้อกำหนดที่ตั้งไว้ ถ้ามีข้อผิดพลาดจะส่งคำร้องไปยัง View **(5)** ให้ทำการสร้างหน้าแจ้งข้อผิดพลาดออกมาและจบการทำงานทันที แต่ถ้าไม่มีข้อผิดพลาดจะใส่ข้อมูลนั้นลงฐานข้อมูล โดยที่ Model จะเป็นคนจัดการข้อมูลในฐานข้อมูลเองทั้งหมด **(3)** แล้ว Controller จะบอกให้ View **(4)** จะทำการสร้างส่วนติดต่อกับผู้ใช้ใหม่ขึ้นมาโดยไปดึงชื่อสินค้าและราคา Model **(5)** ออกมาแสดงที่ส่วนติดต่อผู้ใช้งาน (User Interface)

3. การทำงานจะเป็นแบบข้อที่ 1 และข้อที่ 2 ไปเรื่อย ๆ จนกว่าโปรแกรมจะจบการทำงาน



รูปที่2. 7 แสดงการทำงานร่วมกัน MVC ลูกค้าสั่งซื้อสินค้าทางเว็บไซต์

2.4.3 ประโยชน์ของ MVC

1. การแบ่งส่วนกันอย่างชัดเจนของMVC จะทำให้การทำงานถูกแยกออกจากกันอย่างชัดเจน

2. สามารถโต้ตอบกับผู้ใช้ด้วยการแสดงผลที่หน้าจอหลากหลายของข้อมูลเดียวกัน หรือสามารถเปลี่ยนแปลง รูปแบบหรือโครงสร้างข้อมูล ได้โดยที่ View ยังเหมือนเดิม

3. สนับสนุนการนำ Source code มาใช้ใหม่และช่วยให้นักพัฒนามุ่งสนใจกับหน้าตา Application (View) เดียว โดยไม่สนหน้าตา Application ส่วนอื่น

4. การเปลี่ยนแปลงส่วนใดส่วนหนึ่งของส่วนการเรียกใช้งานจะไม่ส่งผลกระทบต่อส่วนการเรียกใช้งานอื่น

2.5 การค้นหา (Search)

โลกในยุคปัจจุบันเต็มไปด้วยข้อมูลข่าวสาร โดยเฉพาะอินเทอร์เน็ตนั้นถือได้ว่าเป็นแหล่งรวบรวมข้อมูลขนาดใหญ่ จนได้รับสมญานามว่า "ห้องสมุดโลก" ซึ่งมีข้อมูลหลากหลายประเภทและมีแนวโน้มจะเพิ่มขึ้นอย่างรวดเร็ว ดังนั้นในการที่เราจะค้นหา ข้อมูลที่ต้องการได้อย่างรวดเร็วนั้น ไม่ใช่เรื่องง่าย ๆ สำหรับผู้ที่ไม่คุ้นเคยกับแหล่งข้อมูลชนิดนี้ ซึ่งส่วนมากมักประสบปัญหาไม่ทราบว่าข้อมูลที่ต้องการนั้นอยู่ในเว็บไซต์ใด ดังนั้นการ Search จึงทำให้ความสะดวกและมีความรวดเร็วในการค้นหาข้อมูล

2.5.1 เทคนิคการสืบค้นข้อมูล

1. การค้นหาแบบพื้นฐาน (Basic Search) เป็นการค้นหาสารสนเทศอย่างง่ายๆ ไม่ซับซ้อน โดยใช้คำโดดๆ หรือผสมเพียง 1 คำ ในการสืบค้นข้อมูล โดยส่วนใหญ่การค้นหาแบบง่ายจะมีทางเลือกในการค้นหา ได้แก่

1.1 ชื่อผู้แต่ง (Author) เป็น การค้นหาโดยใช้ชื่อของบุคคล กลุ่มบุคคล นามปากกา หรือชื่อหน่วยงาน/องค์กร ที่เป็นผู้แต่งหรือเขียนหนังสือ บทความ งานวิจัย วิทยานิพนธ์ หรือทรัพยากรสารสนเทศนั้นๆ ซึ่งมีหลักการค้นหาง่ายๆ

1.2 ชื่อเรื่อง (Title) เป็น การค้นหาข้อมูล ด้วยชื่อเรื่อง เช่น ชื่อหนังสือ ชื่อบทความ ชื่อเรื่องสั้น นวนิยาย ชื่องานวิจัย หรือวิทยานิพนธ์ การค้นโดยใช้ชื่อเรื่องนี้ เป็นการค้นหาแบบเจาะจง ดังนั้นผู้ค้นต้องรู้จักชื่อเรื่อง หลักการค้นหาด้วยชื่อเรื่องทั้งภาษาไทยและภาษาต่างประเทศ ใช้หลักการเดียวกัน คือ ค้นหาตามชื่อนั้นๆ ได้เลย โดยระบบจะทำการค้นหาจากชื่อเรื่อง เริ่มจากอักษรตัวแรกและตัวถัดไปตามลำดับ

1.3 หัวเรื่อง (Subject Heading) คือ คำหรือวลีที่กำหนดขึ้นมา เพื่อใช้แทนเนื้อหาของหนังสือ บทความ งานวิจัย วิทยานิพนธ์หรือทรัพยากรสารสนเทศนั้นๆ

1.4 คำสำคัญ (Keywords) คือ การ ค้นหาด้วยคำหรือวลีที่กำหนดขึ้นมา เพื่อใช้แทนเรื่องที่ต้องการค้นหา โดยทั่วไปคำสำคัญจะมีลักษณะที่สั้น กะทัดรัด ได้ใจความ มีความหมาย เป็นคำนามหรือเป็นศัพท์เฉพาะในแต่ละสาขาวิชา

2.การค้นหาแบบขั้นสูง (Advanced Search) เป็น การค้นหาที่ซับซ้อนมากกว่าแบบพื้นฐาน ซึ่งมีเทคนิคหรือรูปแบบการค้นที่จะช่วยให้ผู้ค้นสามารถจำกัดขอบเขตการค้นหา หรือค้นแบบเจาะจงได้มากขึ้น เพื่อให้สามารถค้นหาข้อมูลได้ที่ตรงกับความต้องการมากที่สุด

การสืบค้นข้อมูล โดยใช้ตรรกะบูลีน (Boolean Logic) หรือการค้นหาโดยใช้ Operator เป็นการค้นหา โดยใช้คำเชื่อม 3 ตัว คือ AND, OR, NOT ดังนี้

- AND ใช้เชื่อมคำค้น เพื่อจำกัดขอบเขตการค้นหาให้แคบลง เช่นต้องการค้นหาคำว่าส้มตำที่เป็นอาหาร มีรูปแบบการค้นดังนี้ คือ ส้มตำ AND อาหาร หมายถึง ต้องการค้นหาคำว่า ส้มตำ และคำว่า อาหาร

- OR ใช้เชื่อมคำค้น เพื่อขยายขอบเขตให้กว้างขึ้น เช่น ส้มตำไทย OR ส้มตำปูปลาร้าหมายถึง ต้องการค้นหาคำว่าส้มตำไทย และส้มตำปูปลาร้า หรือค้นหาคำใดคำหนึ่งก็ได้

- NOT ใช้เชื่อมคำค้น เพื่อจำกัดขอบเขตให้แคบลง เช่น ต้องการค้นหาคำว่า ส้มตำ AND อาหาร NOT เพลง หมายถึง ต้องการค้นหา คำว่า ส้มตำ ที่เป็นอาหาร ไม่เอาส้มตำที่เป็นเพลง เป็นต้น

2.5.2 การค้นหาโดยใช้เมตาดาตา

เป็นการค้นหาข้อมูลในระดับเมตาดาตา (metadata) ซึ่ง metadata คือคำอธิบายข้อมูลหรือหมายถึงข้อมูลที่บอกรายละเอียดของข้อมูล(data about data) เช่นการค้นหาเบอร์โทรศัพท์จากสมุดโทรศัพท์ ซึ่งภายในเล่มจะเป็นแหล่งรวบรวม metadata ของแต่ละบุคคลเพื่อบอกรายละเอียดของบุคคลที่เราต้องการค้นหา ได้แก่ ชื่อ ที่อยู่ เบอร์โทรศัพท์ หรือป้าย/ฉลากที่ติดบนภาชนะบรรจุ อาหาร เครื่องดื่ม สิ่งของเครื่องใช้ต่างๆ จะมี metadata เพื่ออธิบายข้อมูลแต่ละรายการว่าคืออะไร เช่น ฉลากยาจะประกอบด้วย metadata อธิบาย สรรพคุณ วิธีใช้ ส่วนประกอบ ขนาดบรรจุ วันผลิต วันหมดอายุ ของยาชนิดนั้นๆ เป็นต้น

จุดมุ่งหมายของการจัดทำ metadata การให้ความสำคัญใน การจัดทำ metadata เริ่มแพร่หลายมากในในยุคต์ของการใช้อินเตอร์เน็ตเพื่อการให้บริการสารสนเทศผ่าน World Wide Web ทำให้วงการสิ่งพิมพ์และสื่อสารมวลชนวิธีการสร้างสรรค์งานการเผยแพร่ การบันทึกและสืบค้นสารสนเทศตลอดจนการนำเสนอสารสนเทศเปลี่ยนไป จากที่เคยเผยแพร่ในรูปสิ่งพิมพ์ กลายเป็นเผยแพร่ด้วยสื่ออิเล็กทรอนิกส์สารสนเทศที่เผยแพร่ผ่าน World Wide Web ได้รับความนิยมและเพิ่มปริมาณมากขึ้นอย่างรวดเร็วเป็นผลให้เกิดความยุ่งยากในการสืบค้นเพื่อให้ได้สารสนเทศตามที่ต้องการ จึงมีความจำเป็นในการจัดทำ metadata เพื่อบอกถึงคุณลักษณะและรายละเอียด ของสารสนเทศเหล่านั้นเพื่อให้ได้สารสนเทศตามลักษณะเฉพาะที่แท้จริงในแต่ละ รายการเป็นการนำไปสู่การสืบค้นที่มีประสิทธิภาพและรวดเร็ว สามารถกำหนดว่าต้องการสารสนเทศเกี่ยวกับเรื่องอะไรก็จะได้สารสนเทศในเรื่อง นั้นๆ เราจะพบว่าหัวเรื่องหรือคำสำคัญ(Keyword)ต่างๆ ที่ผู้เขียนเว็บเพจใส่ไว้ที่ Meta Tag ในส่วนบน(Head) ของเอกสาร HTML เป็น metadata ของสารสนเทศในเรื่องนั้นๆ เพื่อเป็นดัชนี ในการค้นหาข้อมูลจากเว็บไซต์ อย่างมีประสิทธิภาพเนื่องจากข้อความหรือคำสำคัญต่างๆ ที่ใส่ไว้ใน Meta Tag จะไปปรากฏตามหน่วยบริการค้นหาข้อมูล(Search Engines)

2.6 การนำทางโดยฐานข้อมูล (Database navigation)

โดยทั่วไปโครงสร้างข้อมูลแบบฐานข้อมูลจะเป็นที่นิยมนำมาใช้กับเว็บขนาดใหญ่ โดยจะมีผู้รับผิดชอบเรื่องระบบฐานข้อมูลโดยเฉพาะ ฐานข้อมูลเป็นระบบข้อมูลที่เป็นที่นิยมมาก โดยข้อมูลจะถูกจัดให้อยู่ในรูปแบบแถวและคอลัมน์ ด้วยกฎเกณฑ์บางอย่างที่มีการกำหนดไว้เฉพาะฐานข้อมูลนั้น ๆ การนำระบบฐานข้อมูลเข้ามาใช้จะช่วยเพิ่มความสามารถในการค้นหาข้อมูลได้อย่าง ถูกต้องรวดเร็ว นอกจากนี้ยังช่วยเพิ่มความสะดวกในการดูแลและปรับปรุงเนื้อหาอย่างมี ประสิทธิภาพอีกด้วย

2.6.1 ออกแบบระบบนำทาง (Designing Web Navigation)

ระบบเว็บ navigation เปรียบเสมือนป้ายบอกทาง การขับรถไปตามถนนโดยไม่มีป้ายบอกทาอาจทำให้หลงทางได้ แต่ถ้ามีป้ายบอกทางก็สามารถเดินทางไปยังเป้าหมายที่ตั้งเอาไว้ได้ หรือทำให้ทราบตำแหน่งที่อยู่ในปัจจุบันได้ เช่นเดียวกับการเข้าไปในอินเตอร์เน็ต อาจทำให้หลงทางได้ เพราะเว็บไซต์บางแห่งขาดระบบนำทางที่ดี โดยปกติแล้วการออกแบบโครงสร้างของเว็บที่ดี ก็มีส่วนให้ผู้ใช้เข้าถึงข้อมูลได้ง่ายขึ้นส่วนระบบ navigation นั้น เป็นส่วนเสริมในการสร้างสิ่งแวดล้อมที่สื่อความหมายเพื่อให้ผู้ใช้สามารถท่องเว็บได้โดยไม่หลงทาง ผู้ใช้จะไม่มีความกังวลใจว่า ขณะนี้ตนเองกำลังอยู่ที่ใดได้ผ่านที่ไหนมาบ้างและกำลังจะไปไหนต่อ [11]

หลักการสำคัญของระบบ navigation คือ การเข้าถึงข้อมูลอย่างสะดวก ไม่ว่าเนื้อหาในเว็บไซต์จะดีเพียงใดก็จะไม่มีประโยชน์ ถ้าหากผู้ใช้ไม่สามารถเข้าถึงเนื้อหานั้นได้ โดยปกติแล้วในการทำงานของเว็บบราวเซอร์ ก็มีระบบ navigation อยู่แล้ว ตัวอย่างเช่น ปุ่ม Back Forward หรือ History เป็นต้น หรือแม้กระทั่งลิงค์ของเว็บไซต์ ที่ใช้สีในการสื่อความหมายว่าข้อความส่วนใดเป็นส่วนที่ลิงค์ได้และหลังจากที่เคยลิงค์ไปแล้วเว็บบราวเซอร์ก็ยังสามารถบอกได้ว่าส่วนใดเคยลิงค์ไปบ้างแล้ว ในการสร้างระบบเว็บ navigation ต้องทราบลักษณะnavigation พื้นฐานของเว็บบราวเซอร์แล้วเป็นอย่างดี เพื่อจะได้หลีกเลี่ยงการออกแบบที่จะไปลบล้างหรือทำซ้ำกับลักษณะนั้นให้สับสน [11]

1. รูปแบบของระบบ navigation

ใน เว็บไซต์ขนาดใหญ่จะมีระบบ navigation หลายรูปแบบ เพื่อเพิ่มช่องทางการเข้าถึงข้อมูลให้มากขึ้น ซึ่งผู้ออกแบบควรมีความเข้าใจและเลือกใช้อย่างเหมาะสม ระบบ navigation แบบออกเป็น 4 รูปแบบดังต่อไปนี้

- *ระบบ navigation แบบลำดับชั้น (Hierarchical)* เป็นระบบ navigation พื้นฐานที่ผู้คนมักใช้กันอยู่ในเว็บไซต์อยู่แล้วคือการที่มีลิงค์ไปยังหน้าอื่น ๆ ในเว็บไซต์นั้น

- ระบบ *navigation* แบบ Global หรือระบบ *navigation* แบบตลอดทั่วทั้งเว็บไซต์ เป็นระบบที่ทำให้สามารถเคลื่อนได้ทั้งแนวตั้งและแนวนอนได้อย่างมีประสิทธิภาพตลอดทั้งเว็บไซต์ โดยปกติแล้วจะทำหน้าที่ลิงค์ไปยังส่วนหลัก ๆ ของเว็บไซต์

- *ระบบ navigation แบบ Local* เป็นระบบที่เหมาะสำหรับเว็บที่มีความซับซ้อนมากๆ เพราะการใช้ระบบ navigation แบบ Global อาจยังไม่เพียงพอ ซึ่งระบบ navigation แบบ Local จะช่วยในกรณี

ที่บางส่วนของเว็บต้องการระบบ navigation ที่มีลักษณะเฉพาะตัว เช่น หัวข้อย่อยของเนื้อหาที่อยู่ภายในส่วนหลัก ๆ ของเว็บไซต์ เพื่อเพิ่มความสะดวกและสื่อความหมายให้ดียิ่งขึ้น

- *ระบบ navigation เฉพาะที่ (Ad Hoc)* เป็นระบบ navigation แบบเฉพาะที่เกิดขึ้นตามความจำเป็นของเนื้อหา ซึ่งก็คือลิงค์ของคำหรือข้อความที่น่าสนใจซึ่งฝังอยู่ในประโยค (Embedded link) ที่เชื่อมโยงรายละเอียดเกี่ยวกับคำนั้นๆ เพิ่มเติม

2. *ประเภทของระบบ navigation หลัก*

ระบบ navigation หลัก (Main Navigation Elements) เป็นระบบ navigation ที่สำคัญและพบได้มากที่สุด โดยจะอยู่ในหน้าเดียวกับเนื้อหา ไม่ใช่ navigation ที่อยู่ในหน้าแรก ช่วยให้ผู้ใช้ที่ผ่านหน้าแรกเข้าไป ไม่ต้องกลับมาเริ่มต้นใหม่ที่หน้าแรกเพื่อเข้าเนื้อหาในส่วนอื่น ๆ ต่อไป navigation หลักทั้งส่วน Global และ Local จะช่วยให้ผู้ใช้สามารถย้ายจากหน้าใด ๆ ไปสู่ส่วนอื่นในเว็บไซต์ได้อย่างคล่องตัว ระบบ navigation หลักมีหลายรูปแบบ ได้แก่

- *Navigation Bar* เป็นระบบพื้นฐานที่ใช้ได้หลายรูปแบบทั้งแบบลำดับชั้น แบบ Global และแบบ Local โดยทั่วไป Navigation Bar จะประกอบด้วยกลุ่มของลิงค์ต่าง ๆ ที่รวมกันอยู่ในบริเวณหนึ่งของหน้าเว็บ โดยอาจจะเป็นตัวหนังสือหรือกราฟิกก็ได้ถือเป็นรูปแบบของระบบ navigation ที่ ได้รับความนิยมมากที่สุด

- *Navigation Bar ระบบเฟรม (Frame-Based)* เป็นระบบ navigation อีกระบบหนึ่งที่ผู้ใช้เข้าถึงได้ง่ายและสม่ำเสมอ คุณสมบัติของเฟรมจะทำให้สามารถแสดงเว็บเพจได้หลายๆ หน้าในหน้าต่างเว็บบราวเซอร์เดียวกัน โดยที่แต่ละหน้ายังเป็นอิสระต่อกัน การลิงค์จากเฟรมที่เป็น Navigation Bar สามารถควบคุมการแสดงผลของข้อมูลในอีก เฟรมหนึ่งได้ ดังนั้น ส่วนที่เป็น Navigation Bar จะปรากฏคงอยู่เสมอ ในขณะที่ผู้ใช้เลื่อนดูข้อมูลในอีกเฟรมหนึ่ง

- *Pull – Down Menu* ซึ่งเป็นส่วนประกอบของฟอร์ม แต่ก็สามารถนำมาใช้เป็นระบบ navigation ได้เช่นเดียวกันคุณสมบัติที่โดดเด่นคือ มีรายการให้เลือกมากมายแต่ใช้พื้นที่น้อยมาก ซึ่งเหมาะสำหรับข้อมูลประเภทเดียวกันที่มีจำนวนมาก เช่น รายชื่อประเทศ จังหวัด เป็นต้น ไม่เหมาะที่จะใช้กับข้อมูลจำนวนน้อยและไม่ควรมีระบบ navigation หลายๆแห่งในหนึ่งหน้า

- *Pop – Up Menu* จะมีลักษณะคล้ายกับ Pull – Down menu แต่รายการย่อยจะปรากฏ เมื่อนำเมาส์ไปวางไว้เหนือตำแหน่งของรายการในเมนูหลัก จากนั้นก็สามารถเลื่อนเมาส์ไปเลือกรายการต่าง ๆ ที่ปรากฏขึ้นได้ ระบบ navigation แบบนี้จะไม่ทำให้เว็บเพจ ดูรกและช่วยประหยัดพื้นที่แสดงรายการย่อยของเมนูได้

- *Image Map* เป็นการใช้กราฟิก เป็นลิงค์ในแบบ Image map ได้รับความนิยมนำมาใช้กับระบบ navigation มากขึ้นเรื่อย ๆ โดยจะกำหนดให้บางบริเวณของกราฟิกนั้นสามารถลิงค์ไปยังส่วนต่าง ๆ ตามที่ต้องการ

I5729646
ปร.
ก16735
2552

- *Search Box* การที่เราเตรียมระบบการสืบค้นข้อมูล (Search) ภายในเว็บไซต์เป็นระบบ navigation แบบหนึ่งที่มีประโยชน์สำหรับเว็บที่มี ข้อมูลปริมาณมาก ๆ ทำให้ผู้ใช้ค้นหาสิ่งที่ต้องการได้อย่างรวดเร็ว

3. ประเภทของระบบ navigation เสริม

ระบบ navigation เสริม (Supplement Navigation Element) หรืออาจจะเรียกอีกอย่างว่า Remote Navigation เป็น navigation อีกแบบที่ช่วยเพิ่มแนวทางการท่องเว็บของผู้ใช้ และยังช่วยให้ผู้ใช้มองเห็นภาพรวมของเว็บไซต์ได้ดียิ่งขึ้น แต่ไม่สามารถนำมาใช้แทนระบบ navigation หลักได้ เพราะผู้ใช้จะเลือกระบบ navigation เสริมนี้ ก็ต่อเมื่อระบบ navigation ไม่สามารถตอบสนองความต้องการได้ ตัวอย่างของระบบ navigation เสริม ได้แก่ ระบบสารบัญ (Table of Contents) ระบบดัชนี (Index System) แผนที่เว็บไซต์ (Site Map) ไกด์ทัวร์ (Guided Tour) เป็นต้น

4.คุณสมบัติของระบบ navigation

แม้ว่าในปัจจุบันจะ มีโปรแกรมช่วยสร้างเว็บไซต์มากมาย แต่ไม่มีโปรแกรมไหนที่จะช่วยสร้างระบบ navigation ได้อย่างมีประสิทธิภาพหลักในการสร้างระบบ navigation ที่เหมาะสม เพื่อสื่อถึงเนื้อหาได้อย่างชัดเจนและน่าสนใจ คุณสมบัติทั้ง 10 ประการต่อไปนี้ ถือเป็นหลักที่จะช่วยในกระบวนการออกแบบระบบ navigation ได้เป็นอย่างดี

- เข้าใจง่าย
- มีความสม่ำเสมอ
- มีการตอบสนองต่อผู้ใช้
- มีความพร้อมและเหมาะสมต่อการใช้งาน
- นำเสนอหลายทางเลือก
- มีขั้นตอนสั้นและประหยัดเวลา
- มีรูปแบบที่สื่อความหมาย
- มีคำอธิบายชัดเจนและเข้าใจได้ง่าย
- เหมาะสมกับวัตถุประสงค์ของเว็บไซต์
- สนับสนุนเป้าหมายและพฤติกรรมของผู้ใช้

2.7 Ontology

Ontology เป็นลักษณะภาษาที่นำมาใช้บรรยายโครงสร้างและความสัมพันธ์ของระบบผ่านโหนดแบบลำดับชั้น (Hierarchies) ในปัจจุบันได้กำหนดภาษามาตรฐานที่ใช้จำลองและออกแบบโครงสร้างของเอกสารเอ็กซ์เอ็มแอล(XML) โดยใช้นิยามแนวคิดให้อยู่ในรูปของกฎ (Role) คลาส (Class) ความสัมพันธ์ระหว่างคลาส (Relation) และคุณสมบัติของคลาส (Properties) แล้วนำเสนอ

ออกมาในรูปของโหนด และความสัมพันธ์แบบลำดับชั้น [12] Ontology เป็นการกำหนดวิธีนำเสนอข้อมูล และแสดงความสัมพันธ์ระหว่างโดเมน โดยมีการใช้อย่างแพร่หลายในวิชา AI, Software Engineering และวิชาที่เกี่ยวกับการจัดการข้อมูล ซึ่ง Ontology มักจะอธิบายถึง

- Individual มีการจัดระดับชั้น object อย่างไร เช่น คน, สัตว์, สิ่งของ
- Class มีการจัดกลุ่ม object อย่างไร เช่น คนหมายถึงคนทั้งหมด, ผู้บริหาร หมายถึงบางส่วนของคน
- Attribute คุณสมบัติของ object เช่น คนมี 2 ขา
- Relation ความสัมพันธ์ระหว่าง object เช่น คนหายใจได้ คุยกันได้
- Event สิ่งที่ทำให้คุณสมบัติหรือความสัมพันธ์ของ object เปลี่ยนไป

อย่างเช่น ontology การเกษตร

ผลิตผลการเกษตร

|- พืช

| |- พืชยืนต้น

| | |- มะม่วง

| | |- มะม่วงแก้ว

| |- พืชล้มลุก

| |- ถั่ว

| |- ข้าว

|- สัตว์

ใช้ช่วยในการทำ Classification (จำแนกกลุ่ม) ในงานวิจัยด้านต่างๆ [13]

a) Ontology representation Languages

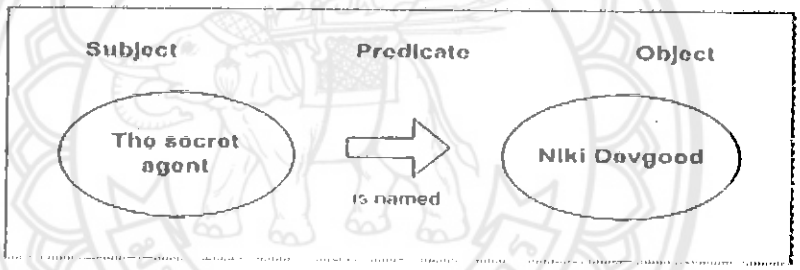
การจัดการข้อมูลในลักษณะของการเชื่อมโยงความสัมพันธ์ของข้อมูล ในระดับเมตะเดต้า (Metadata) โดยทำการอ่านข้อมูลแบบออนไลน์ทำให้เครื่องคอมพิวเตอร์สามารถเข้าใจ ความหมายของข้อมูลต่างๆ ได้ว่าเป็นอะไรมาจากข้อมูลส่วนไหนของชุดข้อมูล ทำให้คอมพิวเตอร์ สามารถนำข้อมูลที่ได้นั้นไปประมวลผลได้โดยอัตโนมัติ ยกตัวอย่างเช่น ถ้าคอมพิวเตอร์พบข้อมูลว่า X เป็นน้องของ Y และ Z เป็นลูกของ Y คอมพิวเตอร์จะสามารถรับรู้ได้เองว่า Z เป็นหลานของ X เป็นต้น โดยจะทำทุกสิ่งทุกอย่าง โดยออนไลน์หมดและข้อมูลทุกอย่างที่ ออนไลน์นั้นจะเชื่อมโยงกันด้วยความสัมพันธ์กันทั่วทั้งระบบทำให้เราสามารถลดขนาดและระยะเวลาในการทำงานให้น้อยลงได้โดยมาตรฐานเทคโนโลยีของ Resource Description Framework หรือ "RDF" Web Ontology Language หรือ "OWL" และ Extensible Markup Language หรือ "XML" มารวมกันเพื่อที่จะพัฒนาให้มีประสิทธิภาพมากยิ่งขึ้น[14]

b)RDF

Resource Description Framework หรือ "RDF"เป็นมาตรฐานที่อิงมาจากภาษา XML แต่มีโครงสร้างรูปแบบหลากหลายกว่า XML แบบจำลองของ RDF เมตะเดต้าอยู่บนพื้นฐานของความคิดในการสร้างข้อความสั่งหรือ"Statement" ของทรัพยากร หรือ "Resource" ในรูปแบบประธาน ภาคแสดง และกรรม หรือ "Subject-Predicate-Object" ซึ่งเรียกว่า A Triple in RDF ซึ่ง Subjectจะหมายถึง Resource Predicate หมายถึงคุณลักษณะ หรือ "Property" และ Object ก็คือ ค่าของคุณลักษณะ หรือ "Property Value"ดังตัวอย่างที่อยู่ในรูปภาษาอังกฤษ [14]

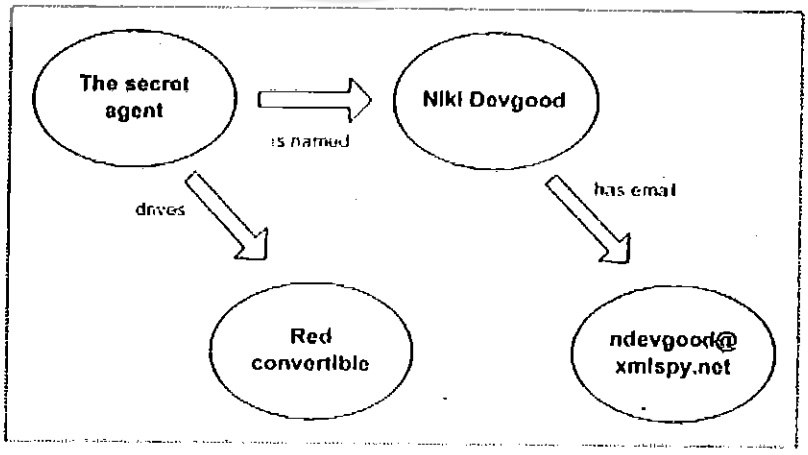
[Resource]	[Property]	[Value]
The secret agent	is	Niki Devgood
[Subject]	[Predicate]	[Object]

RDF Triples สามารถเขียนโดยใช้ XML Tags ได้ดังรูปที่ 2.8



รูปที่2.8 แสดงรูปแบบ RDF Triple [14]

หลังจากสร้างเป็นรูปแบบ RDF Triples แล้วก็สามารถสร้าง Triples อื่นๆ ที่เกี่ยวเนื่องได้ ดังรูปที่ 2.9



รูปที่2.9 แสดงรูปแบบ RDF Triples แบบอื่นๆ [14]

หลังจากนี้เราสามารถสร้างโค้ดทั้งในรูปแบบของ RDF/ XML หรือ n-Triples ได้ตัวอย่างเช่น บทความในเว็บของ Wikipedia ที่เกี่ยวกับ Tony Benn กำหนดให้ "http://en.wikipedia.org/wiki/Tony_Benn" เป็น Resource ที่เฉพาะเจาะจงซึ่งจะกล่าวได้ว่าหัวข้อ หรือ "Title" ของ Resourceนี้ก็คือ"Tony Benn" และ เจ้าของบทความ หรือ "Publisher" คือ "Wikipedia"ซึ่งทั้งสองสามารถแสดงออกมาเป็นเครื่องหมายแบบ RDF Statements ได้ ใน RDF แบบ N-Triples จะมี Statements คล้ายๆ ดังนี้

<http://en.wikipedia.org/wiki/Tony_Benn> <http://purl.org/dc/elements/1.1/title> "Tony Benn" .
<http://en.wikipedia.org/wiki/Tony_Benn> <http://purl.org/dc/elements/1.1/publisher> "Wikipedia" .

และ Statements ทั้งหมดอาจเขียนในรูปแบบ RDF/XML ได้ดังนี้

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:dc="http://purl.org/dc/elements/1.1/">
    <rdf:Description rdf:about="http://en.wikipedia.org/wiki/Tony_Benn">
        <dc:title>Tony Benn</dc:title>
        <dc:publisher>Wikipedia</dc:publisher>
    </rdf:Description>
</rdf:RDF>
```

ซึ่งจะทำให้เข้าใจได้อย่างแน่นอนว่า หัวข้อของ Resource นี้คือ "Tony Benn" จุดประสงค์ของ RDF คือสามารถทำให้เครื่องตีความและเข้าใจในความต้องการด้านบนได้ แต่ทั้งสองรูปแบบด้านบนยังมีการใช้คำมากเกินไปกว่าความต้องการของRDF เพราะว่า RDF ต้องการ Subject Resource ที่เป็นหนึ่งเดียว หรือ "Unique" เพื่อให้ได้ความหมายที่แท้จริง และ Predicate ก็ต้องการความเป็นหนึ่งเดียวเพื่อที่จะลดโอกาสที่ Title หรือ Publisher จะมีความหมายที่คลุมเครือกำกวมในการสร้าง Triples ที่ประกอบด้วย Subjects Predicates และ Objects นั้น RDF จะอนุญาตให้ใช้เครื่องมือในการสร้างการตัดสินใจอย่างมีเหตุผลซึ่งต้อง อยู่บนพื้นฐานของความสัมพันธ์ร่วมกันระหว่าง Subjects และ Objects อย่างไรก็ตามขณะที่ RDF สนับสนุนตัวแบบ หรือ "Model" และไวยากรณ์หรือ "Syntax" ของภาษาสำหรับการระบุถึง Resource แต่มันก็ยังคงไม่ระบุรายละเอียดในเชิงความหมายของ Resource ได้ จึงต้องการ RDFS และ OWL [14]

c) RDFS

RDF Schema หรือ "RDFS" ใช้ในการสร้างคำศัพท์ซึ่งกล่าวถึงความ เกี่ยวข้องกัน ของกลุ่มของ RDF Resources และความสัมพันธ์ของ Resources เหล่านั้น คำศัพท์ของ RDFS จะอธิบายคุณสมบัติซึ่ง

สามารถระบุต่อ RDF Resources ภายในโดเมนที่กำหนดให้ ซึ่ง RDFS จะอนุญาตให้สร้างคลาส หรือ "Classes" ของ Resources ซึ่งสามารถแชร์คุณสมบัติร่วมกัน [14]

การใช้แบบ Triples แบบเดียวกันโดยใช้ RDFS Triples จะประกอบด้วย Classesคุณสมบัติของ Classes และ Values ซึ่งกำหนดความสัมพันธ์ระหว่าง Resources ภายในโดเมนที่เจาะจง Resources จะถูกกำหนดเป็นอินสแตนท์ หรือ "Instances" ของ Classes และ Classes ก็จะเป็น Resource ด้วยและ Classes ใดๆ ก็จะสามารถเป็นซับคลาสหรือ "Subclasses" ของ Classes อื่นๆได้ด้วยโดยลำดับขั้นที่เกี่ยวข้องกันของข้อมูลจะถูกเครื่องตีความบนความหมายของ Resources ที่อยู่บนพื้นฐานของคุณสมบัติและ Classes ของพวกมันโดยรวมแล้ว RDFS เป็นภาษาคำศัพท์พื้นฐานสำหรับแสดงความสัมพันธ์ ระหว่าง Resources [14]

d) OWL

Web Ontology Language หรือ "OWL" เป็นภาษาที่รวมกันระหว่างข้อความหรือ "Text" และข้อความพิเศษ หรือ "Extra Information" ที่เพิ่มเติมเข้ามาเกี่ยวกับข้อความ ที่มีแบบแผนอย่างเป็นทางการที่อธิบายลำดับชั้นและความสัมพันธ์ระหว่าง Resources ที่แตกต่างกัน สร้างอยู่บน RDF และ RDFS ซึ่งประกอบด้วยอนุกรมวิธาน หรือ "Taxonomy" และเซทของกฎที่ได้จากเครื่องที่สามารถสร้างข้อสรุปแบบเชิงตรรกะ หรือ "Logical" [14]

Taxonomy เป็นระบบของการแบ่ง Class เช่น ในทางวิทยาศาสตร์จะมี Kingdom Phylum Class Order ฯลฯ เป็นระบบในการแบ่ง Classในพืชและสัตว์ซึ่งกลุ่ม Resources ภายใน Classes และ Subclasses จะขึ้นอยู่กับความสัมพันธ์และการมีคุณสมบัติร่วมกันของพวกมันเพราะว่า Taxonomy เป็นการแสดงระดับชั้นความสัมพันธ์ระหว่าง Resource ดังนั้นจึงได้ใช้ OWL เพื่อ กำหนดคุณสมบัติของ Class ของ Resource และอนุญาตให้มี Subclass ที่สืบทอด หรือ "Inherit" มาจากคุณสมบัติเดียว กัน ทำให้ OWL มีประโยชน์ต่อรูปแบบข้อมูลของ XML Schema ด้วย โดยที่ OWL ถูกพัฒนาเป็นหลักเพราะมันแสดงออกถึงความหมายและความจริงได้มากกว่า XML RDF และ RDF/S ดังนั้น OWLจึงมีความสามารถมากกว่าภาษาเหล่านั้น [14]

2.8 เว็บค้นหา (Web Search Engine)

เป็นเครื่องมือการค้นหาข้อมูลผ่านอินเตอร์เน็ต มีการเชื่อมโยงไปยังเอกสารในรูปแบบไฮเปอร์เท็กซ์ โดยผลของการสืบค้นให้รายชื่อของเอกสารที่มีเนื้อหาเข้ากันได้ และเมื่อคลิกตรง Web Site ที่ต้องการรายละเอียด Search Engines จะนำไปสู่รายละเอียดของโฮมเพจหน้านั้นๆ ฐานข้อมูลที่เกิดจากเครื่องช่วยค้น [15] ซึ่งการค้นหาข้อมูลมีด้วยกัน 2 วิธีด้วยกันดังนี้

1) การค้นหาในรูปแบบ Index Directory

วิธีการค้นหาข้อมูลแบบ Index จะสามารถคัดแยกข้อมูลออกมาเป็นหมวดหมู่ และจัดแบ่งแยก Site ต่างๆ ออก เป็นประเภท สำหรับการใช้งานสามารถที่จะ Click เลือกข้อมูลที่ต้องการจะดูได้เลยใน Web Browser จากนั้นที่หน้าจอก็จะแสดงรายละเอียดของหัวข้อปลีกย่อย ลึกลงมาอีกระดับหนึ่ง ปรากฏขึ้นมาให้เราเลือกอีก ส่วนจะแสดงออกมาให้เลือกมากเท่าไรก็ขึ้นอยู่กับขนาดของฐานข้อมูลใน Index ว่าในแต่ละประเภท จัดรวบรวมเก็บเอาไว้มากน้อยเพียงใด เมื่อได้ข้อมูลตรงกับความต้องการแล้ว สามารถ Click ลงไปยัง Link เพื่อขอเชื่อมต่อทางเว็บไซต์ก็จะนำเอาผลของข้อมูลดังกล่าวออกมาแสดงผลทันที [15]

2) การค้นหาในรูปแบบ Search Engine

วิธีการอีกอย่างที่นิยมใช้การค้นหาข้อมูลคือการใช้ Search Engine ซึ่งผู้ใช้ส่วนใหญ่กว่า 70% จะใช้วิธีการค้นหาแบบนี้ หลักการทำงานของ Search Engine จะแตกต่างจากการใช้ Index จะมีลักษณะเป็นฐานข้อมูลขนาดใหญ่มหาศาล ที่กระจัดกระจายอยู่ทั่วไปบน Internet ไม่มีการแสดงข้อมูลออกมาเป็นลำดับขั้นของความสำคัญ การใช้งานจะเหมือนการสืบค้นฐานข้อมูลอื่นๆ คือ ต้องพิมพ์คำสำคัญ (Keyword) ซึ่งเป็นการอธิบายถึงข้อมูลที่ต้องการค้นหานั้นๆ เข้าไปจากนั้น Search Engine ก็จะแสดงข้อมูลและ Website ต่างๆที่เกี่ยวข้องออกมา [15]

ข้อแตกต่างระหว่าง Index Directory กับ ใช้ Search Engine คือ วิธีในการค้นหาข้อมูลแบบ Index จะใช้คนเป็นผู้จัดรวบรวมและทำระบบฐานข้อมูลขึ้นมา ส่วนแบบ Search Engine นั้น ระบบฐานข้อมูลของจะได้รับการจัดสร้างโดยใช้ Software ที่เรียกว่า Spider ซึ่งการทำงานจะใช้วิธีการเชื่อมโยงไปตามเครือข่ายต่างๆ ที่เชื่อมโยงถึงกันอยู่เต็มไปหมดใน Internet เพื่อค้นหา Website ที่เกิดขึ้นมาใหม่ๆ รวมทั้งยังสามารถตรวจสอบหาความเปลี่ยนแปลงของ ข้อมูลใน Website เดิมที่มีอยู่ ว่าที่ใดถูกอัพเดตแล้วบ้าง จากนั้นก็จะนำเอาข้อมูลทั้งหมดที่สำรวจเข้ามา ได้เก็บใส่เข้าไปในฐานข้อมูลของตนโดยอัตโนมัติ [15]

2.8.1 ประเภทของ Search Engine

1) แบบอาศัยการจัดเก็บข้อมูลเป็นหลัก (Crawler-Based Search Engine) คือ เครื่องมือค้นหาบนอินเตอร์เน็ต แบบอาศัยการบันทึกและจัดเก็บข้อมูลด้วยตัวเองเป็นหลัก เป็นประเภทของ Search Engine ที่ได้รับความนิยมในปัจจุบันมากที่สุด เช่น Google ด้วยประสิทธิภาพในการค้นหาให้ความแม่นยาสูง และการประมวลผลการค้นหาภายในระยะเวลารวดเร็ว โดยมีองค์ประกอบหลักเพียง 2 ส่วนด้วยกัน คือ

1 ฐานข้อมูล โดยส่วนใหญ่แล้ว Crawler Based Search Engine เหล่านี้จะมีฐานข้อมูลเป็นของตัวเอง ที่มีระบบการประมวลผล และ การจัดอันดับที่เฉพาะ เป็นเอกลักษณ์ของตนเองอย่างมาก [15]

2 ซอฟแวร์ คือ เครื่องมือหลักสำคัญที่สุดอีกส่วนหนึ่งสาหรับ Search Engine ประเภทนี้ เนื่องจากต้องอาศัยโปรแกรมเล็กๆ ทำหน้าที่ในการตรวจหา และทำการจัดเก็บข้อมูล หน้าเพจ หรือ

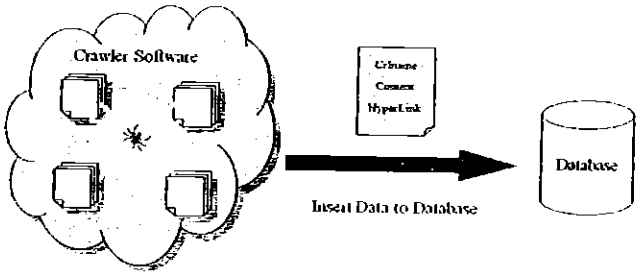
เว็บไซต์ต่าง ๆ ในรูปแบบของการทำสำเนาข้อมูล เหมือนกับต้นฉบับทุกอย่าง ซึ่งเราจะรู้จักกันในนาม Spider หรือ Web Crawler หรือ Search Engine Robots [15]

2) แบบอ้างอิงใน"ชุดคำสั่งเมตะ" (Meta Search Engine) มักจะมีอยู่ใน Code HTML ตามหน้าเพจต่างๆ เรียกว่า "Meta Tag" คือ ชุดแท็กคำสั่งที่ใช้สำหรับประกาศข้อมูลสำคัญต่างๆ ของเอกสารหน้าเพจนั้นๆ เช่น ชื่อผู้พัฒนา, เจ้าของเว็บไซต์, คำค้นหา(Keyword), คำอธิบายย่อของเว็บไซต์ โดยจะปรากฏในส่วนหัวของเอกสารเว็บ(Head Section) มีหน้าที่แจ้งข้อมูลสำคัญให้กับ Search Engine Robot ได้รับทราบ Search Engine ประเภทนี้จะไม่มีระบบฐานข้อมูลของตนเอง แต่อาศัยข้อมูลจาก Search Engine Index Server จากที่อื่นๆ นำมาประมวลผลร่วมกัน ทำให้เกิดผลการค้นหาที่หลากหลาย แต่ในทางตรงข้าม ผลการค้นหาที่หลากหลายนั้นมักได้ผลลัพธ์ไม่ตรงเท่าที่ควร เนื่องจากอาศัยข้อมูลจาก Search Engine Index Server หลายๆแห่งนำมารวมกัน [15]

3) แบบสารบัญเว็บไซต์ (Web Directory) มีหลักการทำงานเหมือนวิธีการใช้สมุดหน้าเหลือง คือมีเนื้อหาที่ถูกแยกหมวด และจัดทำดัชนีไว้แล้วอย่างเป็นระเบียบ ข้อดี คือ สามารถเข้าไปดูข้อมูลของเว็บไซต์ทั้งหมดที่มีเนื้อหาเกี่ยวข้องกัน และตรงตามที่ต้องการของหมวดหมู่นั้นๆ ในปริมาณมาก และสามารถนำเอาข้อมูลเหล่านั้นมาเปรียบเทียบกับเว็บไซต์อื่นๆ กันได้เลย ODP หรือ Dmoz ที่หลาย ๆ คนรู้จัก ซึ่งเป็น Web Directory ที่ใหญ่ที่สุดในโลก Search Engine หลาย ๆ แห่ง ก็ใช้ข้อมูลจากที่แห่งนี้เกือบทั้งสิ้น [15]

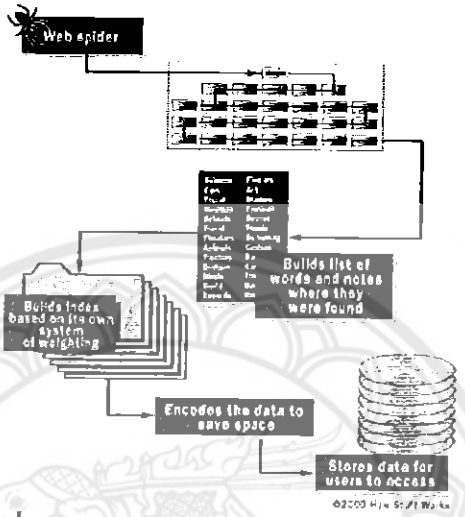
2.8.2 ครอว์เลอร์ (Crawler)

Crawler หรือ Spider, Robot เป็นซอฟต์แวร์ที่วิ่งไปบนอินเตอร์เน็ต ทำหน้าที่การเก็บรวบรวมเอกสารอิเล็กทรอนิกส์บนเว็บ ทั้งหมด เช่น แฟ้มเอกสารประเภท HTML, PHP, PDF, DOC และอื่นๆ บนเว็บ ซึ่งจะเรียกสั้นๆว่า เอกสารบนเว็บ หรือ เอกสาร (Web Documents) ซึ่งมีลักษณะในการทำงานดังรูปที่ 2.10



รูปที่2. 10 ลักษณะการทำงานของ Crawler [16]

เพื่อที่จะบำรุงรักษาให้รายการที่สร้างขึ้นมาใช้ได้อย่างมีประโยชน์ Crawler จะเริ่มต้นปฏิบัติการจากไซต์ที่ได้รับความนิยมก่อน การแสดงครรชนีของข้อมูลที่ค้นหาได้บนเพจ และลิงค์ที่เกี่ยวข้อง ซึ่งค้นหาจากภายใน ด้วยวิธีการนี้ Crawler จึงเริ่มต้นและค้นหาได้อย่างรวดเร็ว โดยการแพร่กระจายไปยังส่วนต่างๆ ของเว็บนั่นเองดังภาพที่ 2.11



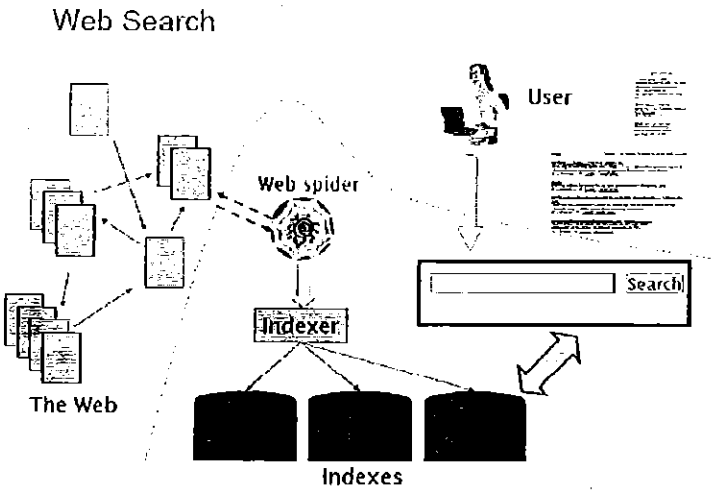
รูปที่2. 11 แสดงการเดินทางของ Crawler [16]

2.8.2.1 ชนิดของ ครอว์เลอร์ (Crawler)

1) Single Crawler มีลักษณะการทำงานของตัว Crawler1 ตัว ที่ใช้วิ่งไปเก็บข้อมูลบนเว็บไซค์มาไว้ที่เครื่อง Server โดยการทำงานของ Single Crawler จะเหมือนกับการทำงานของ 1 Process ตามอัลกอรึทึมพื้นฐานของ Crawler

2) Multi Crawler หรือ Parallel Crawler จะมีลักษณะการทำงานของตัว Crawler หลายๆตัวทำงานพร้อมกัน ใช้วิ่งไปเก็บข้อมูลบนเว็บไซค์มาไว้ที่เครื่อง Server โดยแต่ละตัวมีอิสระต่อกัน บางทีอาจเรียกว่า Distributed Crawler เพื่อให้สามารถทำงานได้อย่างรวดเร็ว

2.8.2.2 การประยุกต์ใช้ Crawler กับงาน Search Engine



รูปที่ 2.12 การประยุกต์ใช้ Crawler กับงาน Search Engine [16]

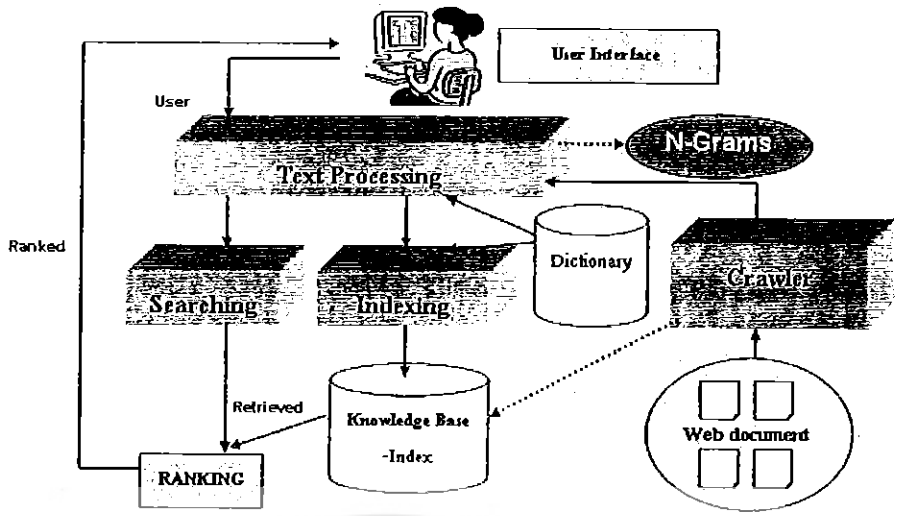
จากรูปที่ 2.12 ในกระบวนการทำงานของ Web Spider หรือ Crawler จะเหมือนกับตัวที่เก็บข้อมูลต่างๆ จากเว็บไซต์ เพื่อมาเก็บไว้ใช้งาน จะเห็นได้ว่ากระบวนการของ Crawler ถือว่าเป็นสิ่งตั้งต้น หรือ สิ่งที่ใช้สร้างฐานข้อมูลความรู้ สำหรับนำมาสร้าง Search Engine นั่นเอง ประกอบไปด้วยกันทั้งหมด 3 ส่วน คือ

1 ส่วนที่เป็น Spider หรือ Crawler จะทำหน้าที่ท่องไปในเว็บต่างๆ แล้วอ่านข้อมูลในเพจ และเข้าไปยังเว็บเพจอื่นๆ ที่ถูกเชื่อมโยงไว้ในเพจนั้นๆ และด้วยวิธีการเข้าถึงข้อมูลแบบนี้ จึงถูกเรียกว่า Crawler *หรือ Spider* จะถูกกำหนดให้เวียนกลับมาอ่านข้อมูลในเว็บไซต์ที่เคยเข้าไปทุกๆเดือน หรือขึ้นอยู่กับการกำหนดในโปรแกรม

2 ส่วนที่เป็นฐานข้อมูลเก็บข้อมูลได้จาก Crawler ค้นหามาได้ ส่วนนี้เรียกว่า Index หรือ Catalog และถ้าข้อมูลของเว็บเพจมีการเปลี่ยนแปลง ข้อมูลในส่วนนี้ก็จะเปลี่ยนแปลงไปด้วย

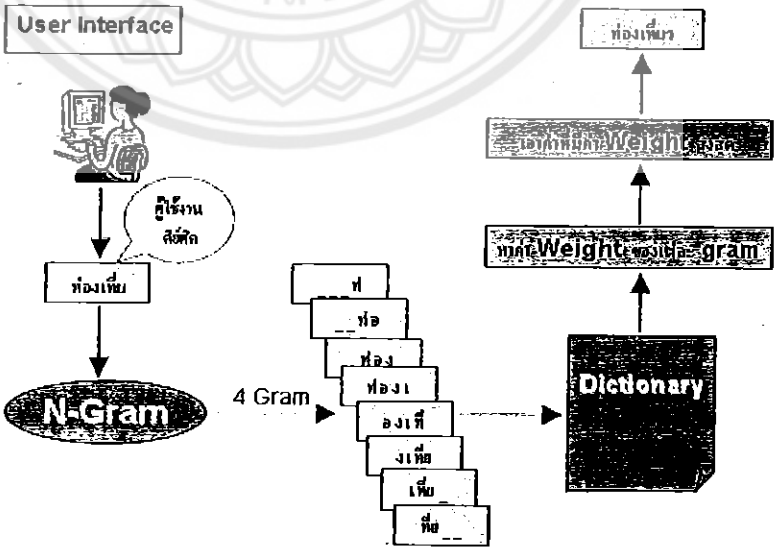
3 โปรแกรมที่ทำหน้าที่เปรียบเทียบข้อมูล จาก Index กับข้อมูลที่ผู้ใช้ค้นหา ว่าเว็บเพจไหนเกี่ยวข้อง หรือมีข้อมูลตรงกับที่ผู้ใช้ค้นหามากที่สุดแล้วจัดลำดับให้

2.8.3 การทำงานของเว็บค้นหา (Search Engine)



รูปที่ 2.13 กระบวนการของ Search Engine [16]

1User Interface ในส่วนของ User Interface มีหน้าที่ในการรับ Query จากผู้ใช้ โดยผู้ใช้สามารถป้อนคำที่ต้องการค้นหา เพื่อนำ Query ไปสู่กระบวนการค้นหาต่อไป และยังเป็นส่วนที่แสดงผลลัพธ์จากการค้นคืนให้กับผู้ใช้ ซึ่งในขั้นตอนการรับค่าจากผู้ใช้ได้นำการตัดคำ ซึ่งอยู่ในส่วนประมวลผลข้อความ โดยใช้ N-Gram มาตัดคำ เพื่อตรวจสอบคำที่ผู้ใช้ป้อนเข้ามาว่าเป็นคำที่ถูกต้องหรือไม่ ซึ่งเป็นการเดา (Suggestion) คำที่คาดว่าน่าจะเป็น ที่ผู้ใช้อาจป้อนเข้ามาผิด ดังภาพ โดยมีกระบวนการ ดังนี้



รูปที่ 2.14 กระบวนการตรวจสอบการรับค่าจาก User Interface โดยใช้ N-Gram [16]

จากรูปที่ 2.14 มีขั้นตอนดังนี้

1) รับค่า Query ที่ได้จากผู้ใช้

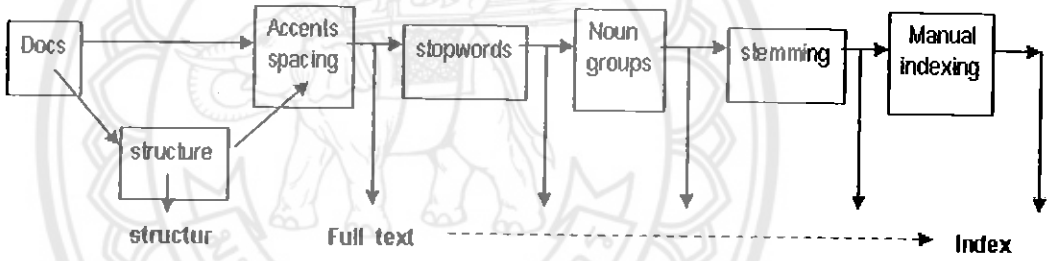
2) นำ N-Gram มาตัดคำของค่า Query ที่รับจากผู้ใช้ โดยใช้ 4 N-Gram ในการตัดคำ ตัวอย่างเช่น รับค่า Query คือ ท่องเที่ย เมื่อทำการตัดคำโดยใช้ N-Gram แล้ว จะได้ผลลัพธ์ ดังนี้ _ท่อง ท่องเ องเที่ งเที่ย เที่ย_ ที่ย_ _ ย_ _ _

3) แล้วนำมาเทียบคำกับพจนานุกรม (Dictionary) จากนั้นจึงให้น้ำหนักกับคำที่ตัดได้

4) เมื่อคำต่าง ๆ ผ่านกระบวนการให้น้ำหนักแล้ว จะดูคำที่มีค่าน้ำหนักสูงสุด จึงจะนำคำนั้น ๆ ไปทำการค้นหา แทนคำที่คีย์เข้ามาผิด

2 Text Processing การประมวลผลข้อความ (Text processing) บนพื้นฐานของระบบการค้นคืนสารสนเทศ (Information Retrieval) สิ่งที่เป็นพื้นฐานที่จำเป็นอย่างยิ่งคือ "หน่วยคำ" ดังนั้นการหาขอบเขตของแต่ละคำจึงเป็นสิ่งแรกที่ต้องคำนึงถึง เพราะหากเลือกการหาขอบเขตคำไม่เหมาะสม อาจนำมาสู่ระบบการประมวลผลข้อความที่ไม่ถูกต้อง แบ่งเป็น 2 ส่วน ดังนี้

1) ส่วนประมวลผลข้อความสำหรับนำไปสร้างดัชนีคำ (Index Terms) ซึ่งใช้ Full Text Search ในการสร้าง Index ดังภาพที่ 2.15



รูปที่ 2.15 กระบวนการประมวลผลข้อความเพื่อนำคำไปสร้าง Index [16]

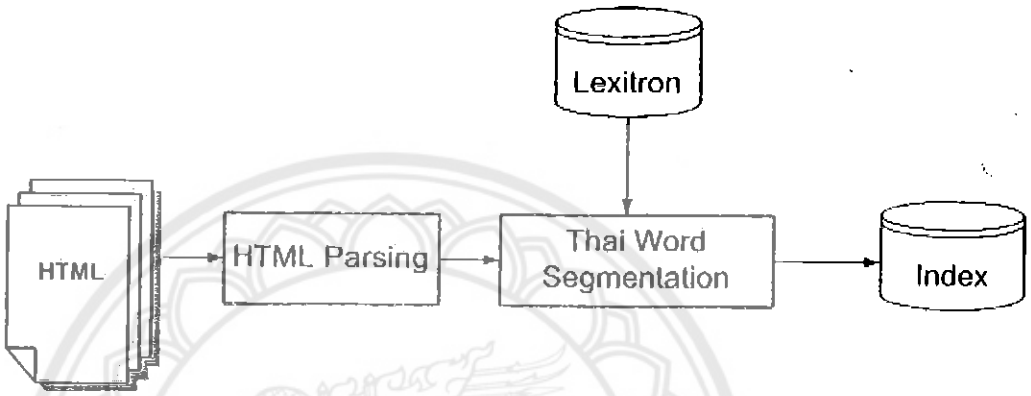
ซึ่งโครงสร้างของการประมวลผลข้อความมี ดังนี้

1.1) จากภาพที่ 2.11 เมื่อ Crawler ไต่ไปตามเว็บต่าง ๆ แล้ว จะได้เอกสารบนเว็บ (Html) ดังรูปที่ 2.16 เอกสารจะมีรูปแบบที่เป็นไร้โครงสร้าง จึงเริ่มด้วยการตัดช่องว่าง (Space) สำหรับภาษาอังกฤษ ส่วนภาษาไทยจะใช้กระบวนการตัดคำ (Word Segmentation) ซึ่งเป็นการแบ่งตัวอักษรจากข้อความ (String) เพื่อหาขอบเขตของแต่ละหน่วยคำ(Morpheme) เนื่องจากส่วนใหญ่ภาษาไทยมีการเขียนในลักษณะที่ติดกันโดยไม่มีการใช้เครื่องหมายวรรคตอนคั่นระหว่างคำ เหมือนภาษาอังกฤษ ซึ่งใช้ช่องว่าง (Space) คั่นระหว่างคำ แต่ภาษาไทยจะมีการเว้นวรรคเป็นระยะเพื่อให้ผู้อ่านทำความเข้าใจกับความหมายของคำ โดยในที่นี้ใช้การตัดคำ ส่วนภาษาไทย โดยอ่านตัวอักษรทีละอักขระ แล้วนำไปเทียบกับพจนานุกรมที่ชื่อว่า Lexitron จึงจะทำการตัดคำตามพจนานุกรม จะได้คำตามพจนานุกรมทั้งเอกสาร

1.2) เมื่อทำการตัดคำแล้ว ขั้นตอนต่อไปจะทำการตัดคำหยุด (Stop word) ซึ่งเป็นคำที่ไม่จำเป็นหรือคำที่ใช้บ่อย ๆ เช่น "การ" "ความ" "ควร" "ที่" "จะ" "ไป" "ใช้" "ได้" "and" "the" "a" "on" "from" เป็นต้น

1.3) แปลงคำให้อยู่ในรากศัพท์ (Stemming) โดยทำให้เป็นคำ ๆ เดียว ซึ่งส่วนนี้จะใช้กับภาษาอังกฤษ เช่น "run" "ran" "running" จะถูกตัดให้เป็นคำว่า "run" คำเดียว เป็นต้น

1.4) นำคำที่ได้ทั้งหมด มาเก็บไว้ในฐานข้อมูล เพื่อนำไปทำดัชนีคำ (Index)



รูปที่ 2.16 แสดงการตัดคำเพื่อนำไปสร้าง Index [16]

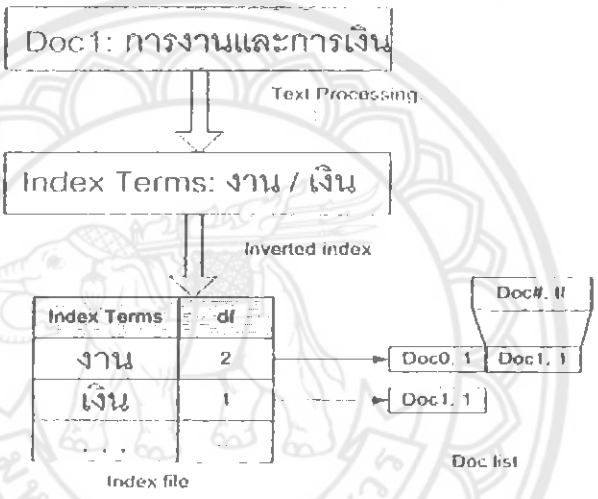
2) ส่วนประมวลผลข้อความสำหรับการรับค่า Query จากผู้ใช้ ในส่วนของ User Interface ได้นำ N-gram มาประยุกต์ใช้ในการตัดคำ N-Gram จะทำการนำบางส่วนของข้อความนั้นออกมาเป็นข้อความตามค่า N ซึ่งเรียกวิธีนี้ว่า *N-Gram* เข้ามาใช้ในการตัดคำแทน และเราจะทำการเพิ่มช่องว่างข้างหน้า และท้ายของข้อความ (ใช้ _ ในการแสดงช่องว่าง) เพื่อเพิ่มโอกาสในการจับคู่ของคำที่อยู่ข้างต้น และท้ายของข้อความ แต่ในภาษาไทยนั้นเราไม่สามารถกำหนดได้ว่า 1 ตัวอักษรคือ 1 Gram เนื่องจากภาษาไทยมีสระและวรรณยุกต์ ดังนั้นคำที่ประกอบไปด้วยสระ และวรรณยุกต์จะถือว่าเป็น 1 Gram เช่น มีข้อความ "สวัสดี" ทำการตัดคำแบบ N-Gram ได้ ดังรูปที่ 2.17

bi-grams : _ส, สว, วั, สด, ดี_
tri-grams : _สว, สวั, วัส, สดี, ดี_ _
quad-grams : _สวั, สวัส, วัสด, สดี_, ดี_ _ _

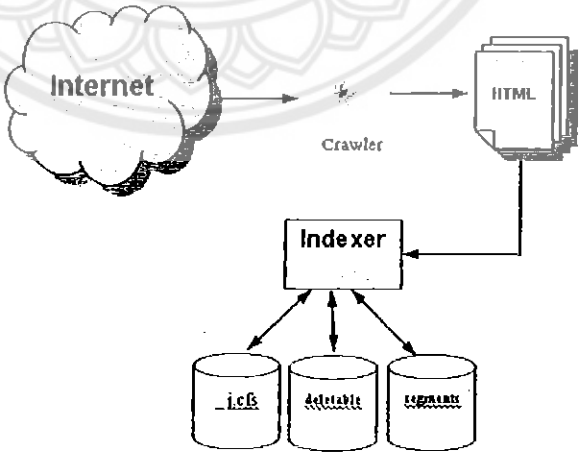
รูปที่ 2.17 แสดงการตัดคำแบบ N-Gram [16]

3) Indexing ในขั้นตอนนี้ระบบจะทำการสร้างดัชนีจากเอกสารบนเว็บ โดย Crawler ทำการดูดเอกสารมาจากเว็บต่าง ๆ โดยที่เอกสารต่าง ๆ ต้องผ่านการประมวลผลข้อความ (Text Processing) มาก่อน เพื่อสกัดเอาคำที่สำคัญ ๆ มาไปสร้างดัชนีคำ (Index Term) ซึ่งขั้นตอนการประมวลผลข้อความจะ

ทำการตัดช่องว่าง คำหยุด (Stop Words) และแปลงคำให้เป็นรากศัพท์ (Streaming) เมื่อมีการเปลี่ยนแปลงเอกสารหรือการเพิ่มเอกสาร ระบบจะต้องการสร้างดัชนีใหม่เพื่อให้สอดคล้องกับเอกสารใหม่ ในกรณีที่มีการเปลี่ยนแปลงของเอกสารเพียงเล็กน้อย ใช้การสร้างดัชนีแบบ Inverted File Index โดยเมื่อทำการประมวลผลข้อความตามที่ได้กล่าวไปข้างต้นแล้ว จากนั้นจึงนำมาเก็บเป็นไฟล์ดัชนีซึ่งมีคำต่าง ๆ พร้อมทั้งจำนวนเอกสารที่คำนั้น ๆ ปรากฏอยู่ และแต่ละคำก็จะมีข้อมูลซึ่งระบุรายการหมายเลขของเอกสารพร้อมทั้งจำนวนค่าที่ ปรากฏอยู่ในเอกสารนั้น ดังภาพ และเก็บรายละเอียดของเอกสารบนเว็บ เช่น URL เพื่ออ้างถึงที่อยู่ของเว็บเพจนั้นๆ ซึ่งกระบวนการสร้างดัชนีคำมีรายละเอียด ดังรูปที่ 2.18 และ 2.19



รูปที่2.18 การสร้างดัชนีแบบ Inverted Index [16]



รูปที่ 2.19 ฐานข้อมูลจาก Indexer [16]

4) Searching เป็นส่วนที่นำคำที่ต้องการค้นหาจากผู้ใช้ผ่านทาง User Interface มาค้นหาในฐานข้อมูลและส่งผลการค้นหากลับไปแสดงบนหน้าจอ โดยนำคำศัพท์แต่ละคำที่ได้มาเปิดหาใน

ฐานข้อมูลดัชนีคำและนำผลลัพธ์ที่ได้จากการค้นหาของแต่ครั้งมาหาเอกสารที่มีคำศัพท์ครบทุกคำ แล้วจึงแสดงชื่อ URL และคำอธิบายของเว็บเพจนั้น ๆ เพื่อกลับไปแสดงทางหน้าจอของผู้ใช้งานซึ่ง Model ที่นำมาใช้คือ Vector Space Model (VSM) โดยการค้นคืนแบบ VSM นั้น จะทำการค้นคืนได้เอกสารที่มีแม้แต่ความเหมือนแค่เพียงบางส่วน โดยที่ไม่ต้องเป็นคำที่ตรงกันทั้งหมด ซึ่งค่าความเหมือนระหว่างเอกสารกับคำค้น โดยคุณสมบัติของ Vector ทำให้สามารถคำนวณค่าความคล้ายคลึง (Similarity) จากการคำนวณด้วยค่าของมุม Cosine ซึ่งพิจารณาจากความถี่ของคำและเอกสารที่ค้นคืนได้สามารถนำไปจัดอันดับ (Ranking) ได้ โดยมีสูตรคำนวณหาค่า Similarity ดังรูปที่ 2.20

$$\mathrm{sim}(D_i, D_j) = D_i \bullet D_j = \sum_{k=1}^{t} (w_{i,k} \times w_{j,k})$$

รูปที่ 2.20 สูตรคำนวณหาค่า Similarity [16]

โดยเริ่มจากการให้น้ำหนักของคำที่ใช้กันอย่างมากในการสืบค้นข้อมูลคือ *tf* x *idf* (TermFrequency x Inverse Document Frequency) โดยค่า *idf* คำนวณจากค่า log(*N*/*df*) ซึ่ง *N* คือจำนวนเอกสารในชุดเอกสารทั้งหมด และ *df* คือ จำนวนเอกสารที่มีคำ ๆ นั้นปรากฏอยู่โดยที่ค่า *tf* คือความถี่ของคำk จากสูตรหาค่า Similarity จึงหาค่าความคล้ายคลึงของคู่เอกสารได้

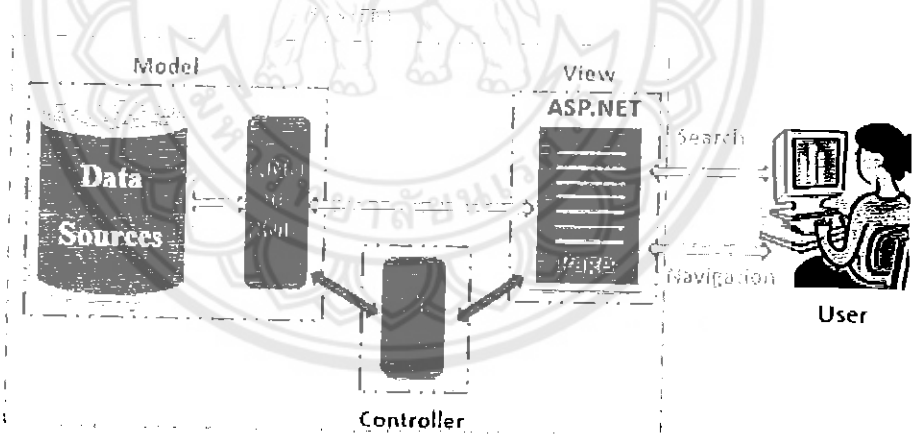
5) Ranking เมื่อทำการค้นหาข้อมูล (Scarching) เพื่อนำมาแสดงให้กับผู้ใช้จะมีการจัดทำลำดับของเอกสาร (Ranking) ตามลำดับความเหมือนของเอกสารกับข้อความที่ต้องการค้นหา (Similarity) ซึ่งเอกสารที่เกี่ยวข้องกับผู้ใช้มากที่สุด (High Relevance) จะถูกแสดงอยู่ในลำดับต้น ๆ ซึ่ง การ จัด ลำดับของเอกสารจะดูจากความถี่ของคำในเอกสาร

บทที่3
ขั้นตอนและวิธีการดำเนินงาน

จากการศึกษาค้นคว้าข้อมูลและทฤษฎีข้างต้น ทำให้สามารถเข้าใจถึงหลักการและวิธีการในการดำเนินงาน ในบทนี้เราจะนำความรู้ที่ได้มาประยุกต์ใช้กับการจัดการระบบฐานข้อมูลในเรื่องของ DBLP โดยการนำ MVC มาใช้ในการออกแบบระบบให้สามารถทำงานได้อย่างมีประสิทธิภาพโดยยึดหลักการของ Ontology เพื่อช่วยในการ Search กับ Navigation

3.1 การออกแบบระบบ (System Design)

เราใช้ MVC ในบทที่ 2 หัวข้อ 2.4 ในการ Design System โดย MVC Pattern เป็นระบบการทำงานที่แยกระบบการทำงานออกเป็น ส่วนสำหรับเก็บข้อมูล (model), ส่วนที่แสดงผลข้อมูล (view) และ ส่วนที่ใช้ติดต่อกับผู้ใช้ (controller) ซึ่งการทำงานทั้ง 3 ส่วนจะแยกออกจากกันอย่างชัดเจนเราใช้ MVC ในการออกแบบระบบของเราได้โครงสร้างดังรูปที่ 3.1



รูปที่ 3.1 แสดงระบบการทำงานของ Web Application โดยใช้ MVC Pattern ในการออกแบบระบบ

ความสำคัญในการเลือกใช้ MVC Pattern คือ MVC เป็น Design Pattern ในการพัฒนา Web Application ขนาดใหญ่ ลักษณะของ Source Code ที่ได้จะมีการแบ่งแยกกันอย่างชัดเจน ยืดหยุ่น สามารถปรับปรุงแก้ไขได้ง่ายโดยไม่กระทบถึงส่วนอื่นของโปรแกรม ในโครงการนี้ออกแบบระบบโดยใช้ MVC Pattern ซึ่ง MVC Patter จะแยกระบบการทำงานออกเป็น ส่วนสำหรับเก็บข้อมูล (model), ส่วนที่แสดงผลข้อมูล (view) และ ส่วนที่ใช้ติดต่อกับผู้ใช้ (controller) ดังต่อไปนี้

Model: เป็นส่วนที่ทำหน้าที่ในติดต่อกับระบบฐานข้อมูลซึ่งเป็นการดำเนินการต่างๆ ไม่ว่าจะเป็นการเลือกข้อมูลมาแสดงผล การแก้ไข การเพิ่ม หรือ การลบ ซึ่งการดำเนินการต่างๆเหล่านี้จะเกิดขึ้นได้ก็ต่อเมื่อมีการเรียกใช้งานจากในส่วน Controller หรือ View

ในระบบของเรานั้น Model จะประกอบด้วยแหล่งข้อมูลดังต่อไปนี้: DBLP, Ontology, Publication, Tags และ User Content ซึ่งแหล่งข้อมูลเหล่านี้ถูกจัดเก็บในรูปแบบของ XML Data Model (ซึ่งจะอธิบายในหัวข้อ 3.2) จากนั้นเราจะใช้คำสั่งในภาษา LINQ to XML เพื่อเรียกใช้ข้อมูล XML เหล่านั้น (ซึ่งจะอธิบายในหัวข้อ 3.3)

View: ทำหน้าที่ทำหน้าที่ในการติดต่อกับผู้ใช้ และการแสดงผลในหน้าเว็บเพจ ซึ่งเป็นส่วนที่ทำการเรียกข้อมูลจาก Model มาแสดงผล ซึ่งทำให้ผู้ใช้สามารถที่จะตอบโต้กับระบบได้เป็นอย่างดี อย่างไรก็ตาม View ไม่ได้ทำหน้าที่ในตอบโต้กับผู้ใช้อย่างแท้จริง แต่จะต้องส่งต่อให้กับ Controller เพื่อดำเนินการต่อไป

สำหรับการพัฒนาในส่วนของ View ภาษาที่ใช้ในการพัฒนาคือ ASP.NET ซึ่งจะทำหน้าที่ในการสร้างฟอร์มสำหรับตอบโต้การใช้งานกับผู้ใช้เพื่อที่จะทำให้ผู้ใช้สามารถใช้งานได้ง่าย ซึ่งจะเป็นการเรียกฟังก์ชั่นการทำงานต่างๆจาก View ไปยัง Controller เพื่อไปประมวลผลต่อไป การออกแบบ View ของระบบจะใช้ Hypertext Model (ซึ่งจะอธิบายในหัวข้อ 3.4) ในการออกแบบ

Controller: เป็นส่วนที่ทำหน้าที่ควบคุมระบบการทำงาน ซึ่งได้รับการสั่งการมาจากผู้ใช้และทำการประมวลผลหรือการดำเนินการต่างๆเกี่ยวกับระบบฐานข้อมูล โดย Controller จะทำหน้าที่เป็นตัวกลางคอยควบคุมให้ Model และ View ดำเนินไปในทิศทางเดียวกัน

การทำงานของ Controller นั้นจะเป็นส่วนของ Search (ซึ่งจะอธิบายในหัวข้อ 3.5) , Navigation (ซึ่งจะอธิบายในหัวข้อ 3.6) , User - Contributed Content Management (ซึ่งจะอธิบายในหัวข้อ 3.7) ซึ่งจะเป็น Logic ต่างๆที่ขับเคลื่อนการทำงานให้สามารถดำเนินไปสู่ผลลัพธ์ได้ โดยในส่วนของการทำงานนั้นจะอาศัยสคริปต์เพื่อที่จะเป็นตัวควบคุมการทำงานโดยใช้ C#

3.2 แหล่งข้อมูล (Data Sources)

แหล่งข้อมูลที่ใช้ในระบบการสืบค้นเอกสารอ้างโดยใช้ Ontology ในโครงการนี้ประกอบไปแหล่งข้อมูลเหล่านี้ถูกจัดเก็บในรูปแบบของ XML Data Model ดังต่อไปนี้

3.2.1 ดีบีแอลพี (DBLP)

ในการทำงานของการให้ข้อมูลทางด้านเอกสารอ้างอิงซึ่งเป็น journal และ proceedings ของ Computer science DBLP มีข้อมูลกว่า 1 ล้านหัวข้อและสิ่งที่เกี่ยวข้องกว่า 1 หมื่น link ที่จะเชื่อมไปยัง home page ของข้อมูลทางด้าน Computer page โดยเราจะสนใจในเรื่องของ Database system และ Logic programming ซึ่งในขณะนี้กำลังเติบโตขึ้นเรื่อยๆในสาขา Computer science ซึ่งสามารถเรียก

DBLP ว่า "Digital Bibliography & Library Project"[14] เป็นDataBase ที่เก็บข้อมูลเอกสารอ้างอิงทางวิชาการของ Computer scienceระบบการค้นหาเอกสารอ้างอิงโดยใช้ Ontology ใช้ DBLP เป็นระบบฐานข้อมูล

a) Code ของ DBLP Data Sources

Book เป็นรายละเอียดของหนังสือที่ประกอบไปด้วย รหัสหนังสือ สำนักพิมพ์ ซีรี่ ชื่อเรื่อง ภาค ปี ผู้แต่งหรือผู้แก้ไข แหล่งอ้างอิง ดังรูปที่ 3.2

```
<book mdate="2004-03-08" key="books/acm/Kim95">
 <editor>Won Kim</editor>
 <title>Modern Database Systems: The Object Model, Interoperability, and Beyond.</title>
 <booktitle>Modern Database Systems</booktitle>
 <publisher>ACM Press and Addison-Wesley</publisher>
 <year>1995</year>
 <isbn>0-201-59098-0</isbn>
 <url>db/books/collections/kim95.html</url>
</book>
```

รูปที่ 3.2 รายละเอียดของ Book

Proceeding เป็นรายละเอียดของ รวมเล่มการประชุมวิชาการซึ่งประกอบไปด้วย รหัส ชื่อการประชุม ซีรี่ ชื่อเรื่อง สำนักพิมพ์ ปี ผู้แต่ง ดังรูปที่ 3.3

```
<proceedings key="conf/aadebug/2005" mdate="2005-09-26">
 <editor>Clinton Jeffery</editor>
 <editor>Jong-Deok Choi</editor>
 <editor>Raimondas Lencevicius</editor>
 <title>Proceedings of the Sixth International Workshop on Automated Debugging,
       AADEBUG 2005, Monterey, California, USA, September 19-21, 2005</title>
 <booktitle>AADEBUG</booktitle>
 <year>2005</year>
 <url>db/conf/aadebug/aadebug2005.html</url>
 <publisher>ACM</publisher>
 <isbn>1-59593-050-7</isbn>
</proceedings>
```

รูปที่ 3.3 รายละเอียดของ Proceeding

Article เป็นส่วนแสดงรายละเอียดของ Article ซึ่งเป็นส่วนที่ย่อยที่สุดสำหรับ book journal และ proceeding ดังรูปที่ 3.4

```
<article mdate="2008-09-12" key="journals/winet/TsaiPLL05">
 <author>Hsien-Ming Tsai</author>
 <author>Ai-Chun Pang</author>
 <author>Yung-Chun Lin</author>
 <author>Yi-Bing Lin</author>
 <title>Repacking on Demand for Hierarchical Cellular Networks.</title>
 <pages>719-728</pages>
 <year>2005</year>
 <volume>11</volume>
 <journal>Wireless Networks</journal>
 <number>6</number>
 <ee>http://dx.doi.org/10.1007/s11276-005-3526-5</ee>
 <url>db/journals/winet/winet11.html#TsaiPLL05</url>
</article>
```

รูปที่ 3.4 รายละเอียดของ Article

Inproceedings เป็นส่วนแสดงรายละเอียดของ Content ของ Proceedings ดังรูปที่ 3.5

```
<inproceedings key="conf/aadebug/Pugh05" mdate="2006-02-13">
 <author>William Pugh</author>
 <title>Keynote talk: Advice to bug hunters.</title>
 <pages>1-2</pages>
 <year>2005</year>
 <crossref>conf/aadebug/2005</crossref>
 <booktitle>AADEBUG</booktitle>
 <ee>http://doi.acm.org/10.1145/1085130.1085131</ee>
 <url>db/conf/aadebug/aadebug2005.html#Pugh05</url>
</inproceedings>
```

รูปที่ 3.5 รายละเอียดของ Inproceeding

Incollection เป็นส่วนแสดงรายละเอียดของ Content ของ book ดังรูปที่ 3.6

```
<incollection mdate="2002-01-03" key="books/acm/kim95/Garcia-MolinaH95">
 <author>Hector Garcia-Molina</author>
 <author>Meichun Hsu</author>
 <title>Distributed Databases.</title>
 <pages>477-493</pages>
 <year>1995</year>
 <booktitle>Modern Database Systems</booktitle>
 <url>db/books/collections/kim95.html#Garcia-MolinaH95</url>
</incollection>
```

รูปที่ 3.6 รายละเอียดของ Incollection

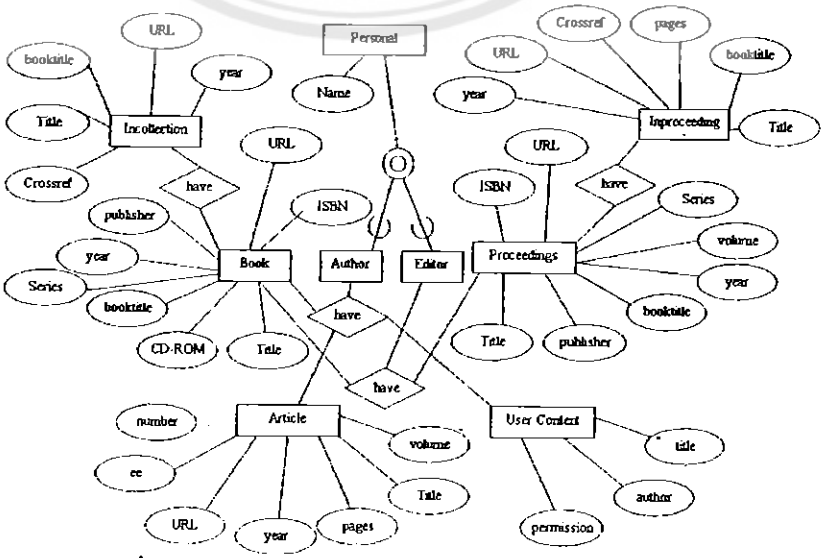
User Content เป็นส่วนเก็บรายละเอียดที่User ส่ง Content ของตนเองขึ้นสู่เว็บไซต์ ดังรูปที่ 3.7

```
<userContent key="jearbcpe/database1.txt">
 <username>jearbcpe</username>
 <title>database 1</title>
 <author>jearbcpe</author>
 <permission>public</permission>
</userContent>
```

รูปที่ 3.7 รายละเอียดของ User Content

b) ER – DIAGRAM ของ DBLP Data Sources

จากหัวข้อ *a) DBLP XML* แต่ละ Entiry จะมีสัมพันธ์ระว่าง Entiry หรือกลุ่มข้อมูลของ DBLP สามารถที่จะเขียนเป็น Entity Relationship Diagram ได้ดังรูปที่ 3.8

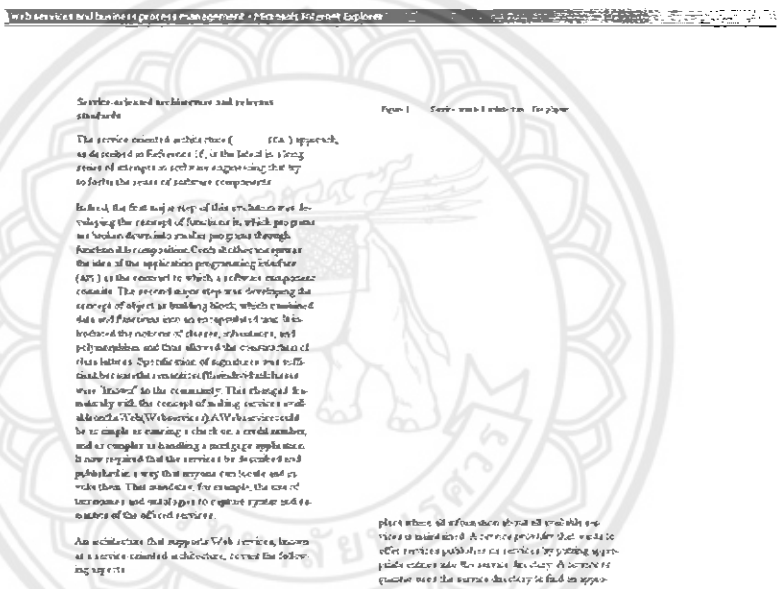


รูปที่ 3.8 แสดงโครงสร้าง ER – DIAGRAM ของ DBLP

3.2.2 เอกสาร (Publication)

เอกสารจะประกอบด้วย Content เป็นเนื้อหาภายในเอกสารใช้เป็น Database เอกสารที่เก็บไว้ในฐานข้อมูลทั้งที่ได้มาจากการโหลดมาและที่มีอยู่แล้วในระบบฐานข้อมูล

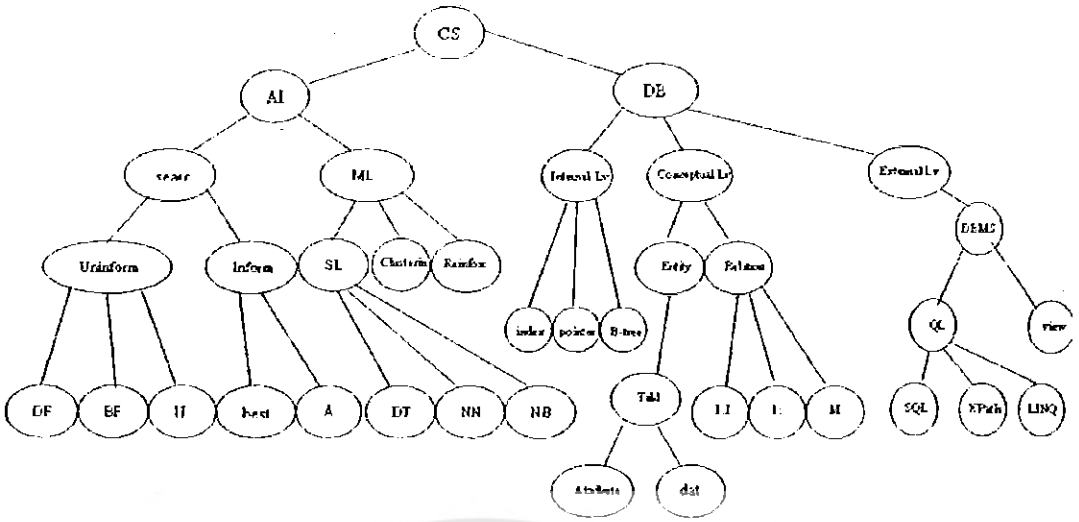
การ Download เอกสารจากเข้าสู่ฐานข้อมูลของเว็บนั้น กระทำการโดยการ Request ค่าผ่าน URL ของ Google Scholar โดยระบุค่าตัวแปรของสิ่งที่ต้องการ เช่นการกำหนดค่า Title เพื่อค้นหาเอกสารที่เกี่ยวข้อง เช่น http://scholar.google.com/scholar?hl=en&q=" + title + "&btnG=Search หลังจากนั้น Google Scholar จะส่งค่ากลับมาในลักษณะของ HTML โปรแกรมจะทำการเข้าไปจับลิงค์ที่สามารถ download เอกสารได้ หลังจานั้นจะทำการโหลดเอกสารจากลิงค์นั้น เพื่อเก็บเข้าสู่ฐานข้อมูลของเว็บไซต์ ดังรูปที่ 3.9



รูปที่ 3.9 ตัวอย่างเอกสาร Publication

3.2.3 Ontology

ในโครงงานนี้จะออกแบบและพัฒนา Web Application ที่มีข้อมูลเอกสารอ้างอิงที่เกี่ยวกับ Computer Since and Engineering เท่านั้นจากรูปที่ 3.10 โดยนำข้อมูลมาจาก DBLP (DataBase systems and Logic Programming) โดย Ontology ที่นำมาใช้สนใจเฉพาะ DataBase และ Artificial Intelligence เบื้องต้น



รูปที่ 3.10 แสดงรายละเอียด Ontology AI&DB

Ontology เป็นลักษณะภาษาที่นำมาใช้บรรยายโครงสร้างและความสัมพันธ์ของระบบผ่านโหนดแบบลำดับชั้น ซึ่งในที่นี้ให้ความสนใจในด้าน Computer science ในเรื่องของ DataBase และ Artificial Intelligence เมื่อมาบรรยายแบบ Ontology แล้วจะได้ดังรูปที่ 3.18 ซึ่งนำมาเขียนเป็น XML ในดังนี้

```
<RDF>
 <!--Lv1-->
 <concept name="Computer Since" id="1">
  <synoname name="cs"/>
  < hasChild id="2"/>
  < hasChild id="3"/>
 </concept>

 <!--Lv2-->
 <concept name="artificial intelligence" id="2">
  <synoname name="ai"/>
  <subclassof id="1"/>
  < hasChild id="4"/>
  < hasChild id="5"/>
 </concept>
 <concept name="database" id="3">
```

```
<synoname name="db"/>
<subclassof id="1"/>
< hasChild id="6"/>
< hasChild id="7"/>
< hasChild id="8"/>
</concept>
```

โดยการเขียนจะอธิบายโครงสร้างและความสัมพันธ์ของระบบผ่านโหนดแบบลำดับชั้นโดยมีรายละเอียดดังนี้

concept name	คือ ชื่อโหนดนั้น
synoname name	คือ ชื่อที่ความหมายเหมือนกันของชื่อโหนดนั้น
subclassof	คือ ลำดับที่ของโหนดพ่อแม่(Parent Node) ของโหนดนั้น
hasChild	คือ ลำดับที่ของโหนดลูก (Child Node) ทั้งหมดที่เป็นของโหนดนั้น

เราสามารถอธิบาย Ontology ที่สร้างขึ้นดูได้จาก Lv2 โหนด AI ดังต่อไปนี้

```
<!--Lv2-->                                  :
<concept name="artificial intelligence" id="2">  : ชื่อโหนด artificial intelligence , id =2
 <synoname name="ai"/>                      : ชื่อ ai
 <subclassof id="1"/>                       : โหนดพ่อแม่ของ ai คือ cs โดย cs มี id =1
 < hasChild id="4"/>                        : โหนดลูกของ ai คือ Search โดย Search มี id = 4
 < hasChild id="5"/>                        : โหนดลูกของ ai คือ ML โดย ML มี id = 5
</concept>
```

3.2.4 แทกซ์ (Tags)

คำสั้นๆ ที่User ใช้อธิบาย ความหมายของ สิ่งใดสิ่งหนึ่งอย่างชัดเจนเป็นคำๆ ซึ่ง Tags คือ คำสำคัญของบทความ เพื่อช่วยในการค้นหา ดังรูปที่ 3.11

```
<?xml version="1.0" encoding="utf-8"?>
<userTags>
 <userTag user="jearbcpe">
  <tags key="jearbcpe/test.txt">
   <tag>computer</tag>
   <note>test</note>
  </tags>
  <tags key="jearbcpe/delElement.txt">
   <tag>test3</tag>
   <note>test3</note>
  </tags>
  <tags key="books/acm/kim95/DittrichD95">
   <tag>test2</tag>
   <note>test32</note>
  </tags>
```

รูปที่ 3.11 แสดงรายละเอียด Tags

3.2.5 ข้อมูลจากผู้ใช้ (User-Contributed Content)

User Contributed Content มาจากเอกสารที่ User อัพโหลดเก็บไว้ในฐานข้อมูลจะมีรายละเอียดดังรูปที่ 3.12

```
<?xml version="1.0" encoding="utf-8"?>
<userContents>
 <userContent key="jearbcpe/web_application.txt">
  <username>jearbcpe</username>
  <title>web application database database</title>
  <conversation>jearbcpe say : test</conversation>
  <conversation>jearbcpe say : database</conversation>
  <author>jearbcpe</author>
  <permission>contact</permission>
 </userContent>
```

รูปที่ 3.12 แสดงรายละเอียดเอกสารที่ User อัพโหลดไว้

3.3 การเข้าถึงข้อมูล (Data Access)

เป็นการเข้าถึงข้อมูลและการเรียกใช้งานข้อมูลจาก Data Sources โดยโครงงานนี้ใช้ LINQ ซึ่งสามารถที่จะเรียกใช้ Data Sources ได้ดังนี้

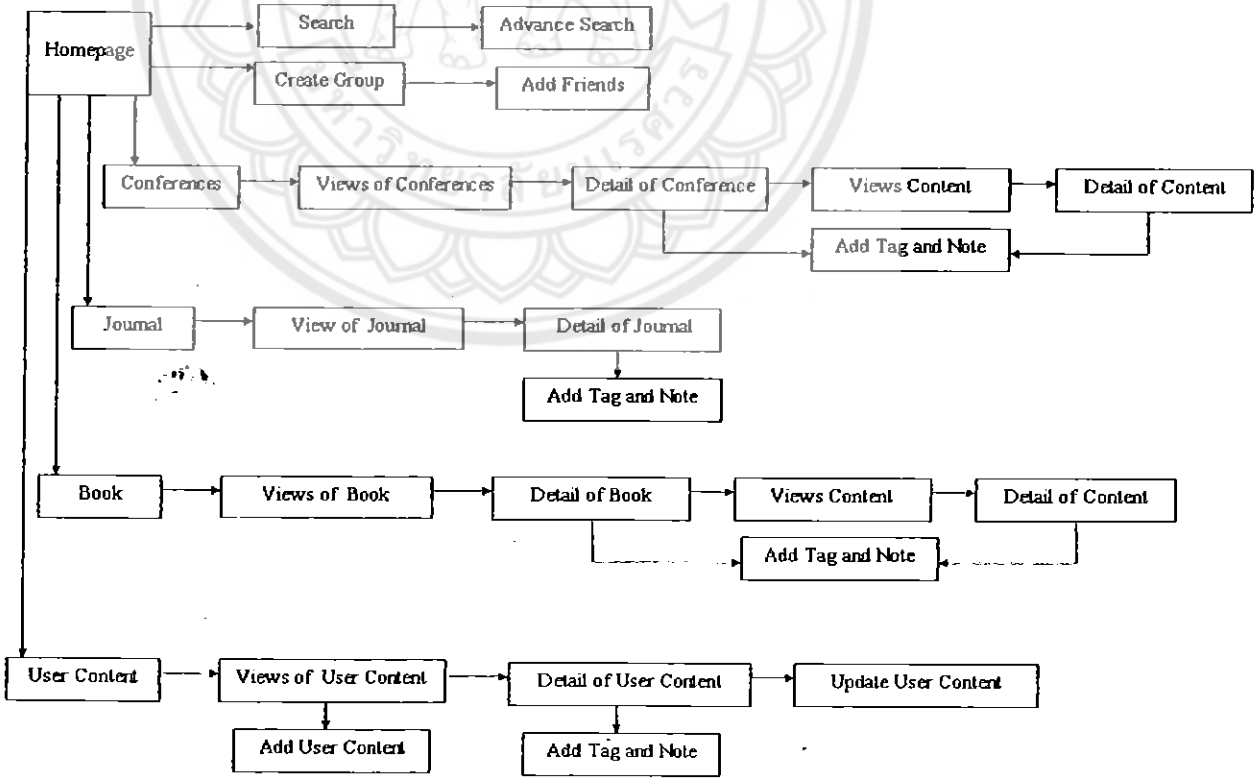
```
var key = Request.QueryString["key"];
var xd = XElement.Load(MapPath("~/App_Data/database.xml"));

var queryTitle = from b in xd.Elements("book")
                 where b.Attribute("key").Value.Equals(key)
                 select (string)b.Element("title");
```

จากโค้ด LINQ ข้างต้นจะเป็นการทำงานในส่วนของการ query ซึ่งเป็นการดึงชื่อ Title ของหนังสือเล่มหนึ่งที่ระบุเงื่อนไข โดยให้มี Attribute ที่ชื่อว่า key มีค่าตรงกับ Query String ซึ่งเป็นค่าของตัวแปร key ที่ส่งมากับ URL ของ page นั้น

3.4 แบบจำลองไฮเปอร์เทกซ์ (Hypertext Model)

Web Application ทั่วไปจะประกอบ Set ของหน้าเว็บเพจต่างๆ ซึ่งเชื่อมโยงกันโดยHyperlinks โดย Web Application ที่เราพัฒนาจะรูปแบบดังรูปที่ 3.13



รูปที่ 3.13 แสดงโครงสร้าง Set ของหน้าเว็บเพจ การสืบค้นเอกสารอ้างอิงโดยใช้ Ontology

1. เริ่มต้นที่หน้าแรกเป็นหน้าของการ Login โดยผู้ใช้จะต้องกรอก username และ password หรือกรณีที่ยังไม่มี username สามารถสมัครได้โดยคลิกลิงค์ register เพื่อสมัครดังหน้า register
2. หลังจาก login เข้ามาจะเป็นหน้า *Homepage* โดยในหน้านี้จะประกอบไปด้วยลิงค์ซึ่งจะเชื่อมต่อไปยังหน้าอื่น รวมทั้งจะแสดงประวัติ และกลุ่ม ของ user ได้เพิ่มเข้ามา
3. หลังจากที่คลิกลิงค์เข้ามาในแต่ละส่วนไม่ว่าจะเป็น *Conference* , *Journal* , *Book* และ *User Content* จะเข้ามาสู่หน้าซึ่งเป็น view แสดงรายการของชุดข้อมูลนั้นโดยจะแสดงเป็น 4 แบบดังนี้
   3.1 List View เป็นส่วนที่แสดงรายการทั้งหมดออกเป็น list ซึ่งจะมีหลายหน้า
   3.2 Grid View เป็นส่วนที่แสดงรายการที่เราสามารถที่จะ sort ข้อมูลต่างๆในตารางได้ เพื่อเพิ่มความสะดวกในการค้นหาข้อมูล
   3.3 Repeater View เป็นส่วนแสดงรายการข้อมูล ที่จะมีรายละเอียดของข้อมูลมาด้วยทำ ให้งานต่อการหาข้อมูลที่ต้องการรายละเอียดสูง
   3.4 Tree View เป็นการแสดงรายข้อมูลในรูปแบบของ Tree ทำให้สามารถค้นหาข้อมูลได้งานและเห็นภาพรวมทั้งหมด
4. หลังจากที่เราทำการคลิกจากหน้า View เข้ามาจะเป็นหน้ารายละเอียดของข้อมูล Detail ต่างๆ
5. จากหน้ารายละเอียดของข้อมูลจะมีลิงค์ที่ชื่อผู้แต่ง ซึ่งเราสามารถคลิกลิงค์เข้าไป จะเห็นรายการของ สิ่งที่ผู้แต่งคนนั้นเคยแต่ง
6. ในหน้ารายละเอียดของข้อมูลจะมีลิงค์ Content ที่สามารถจะเชื่อมโยงไปยัง Content ของข้อมูลชุดนั้นได้ โดยจะแสดงรายละเอียดเป็นView ทั้ง 4 แบบเหมือนกับที่กล่าวในข้างต้น
7. ภายในหน้ารายละเอียดข้อมูลเราสามารถที่ Tag หรือ Note ข้อความได้
8. ภายในหน้ารายละเอียดข้อมูลเราสามารถที่จะทำการสนทนากับผู้ใช้ท่านอื่นได้ โดยผ่านส่วนของ Conversation
9. ในหน้ารายการของ User Content ผู้ใช้สามารถที่จะ Upload Content ของท่านเข้าสู่เว็บไซค์ได้ โดยจะสามารถที่จะกำหนดสิทธิ์สำหรับผู้ที่จะเห็น Content ของท่าน
10. ในหน้าของ รายละเอียดของ User Content ผู้ที่เป็นเจ้าของ Content นั้น สามารถที่จะแก้ document ของท่านได้ แต่จะต้องใช้ชื่อไฟล์เดิมเท่านั้น
11. จากลิงค์ด้านบนของ page จะสามารถเข้าสู้หน้า search ได้ โดนการ search จะเรื่องของ Ontology มาช่วย และหลังจากการ search ท่านจะสามารถคลิกลิงค์เพื่อเข้าสู่หน้ารายละเอียดข้อมูลนั้นๆได้
12. จากหน้า Search จะมีลิงค์ ที่จะเข้ามาสู่หน้า Advance Search ซึ่ง Advance Search จะมีลักษณะคล้ายกับ การ search แต่จะเพิ่มในเรื่องของรายละเอียดที่จำเพาะเจาะจงขึ้น

3.5 การค้นหา (Search)

จากบทที่ 2 หัวข้อที่ 2.5 การ Search แบบ Basic Search และ Advanced Search อาจไม่เพียงพอในการสืบค้นหาข้อมูลและอาจไม่ตรงตามความต้องการผู้ใช้งาน ในโครงการนี้จึงพัฒนาระบบการ Search ดังต่อไปนี้

3.5.1 เมตาดาตา (Meta Data)

เป็นการค้นหาข้อมูลในระดับ Meta data ซึ่ง Meta data คือคำอธิบายข้อมูลหรือหมายถึงข้อมูลที่บอกรายละเอียดของข้อมูล (data about data) ในโครงการนี้ เป็นการ Search จากรายละเอียดของหนังสือเล่มนั้นไม่ว่าจะเป็น ชื่อหนังสือ ชื่อผู้แต่ง สำนักพิมพ์ เป็นต้น โดย Search ฐานข้อมูล DBLP

3.5.2 เอกสาร (Publications)

เป็นการ Search จากภายใน Content ซึ่งเป็นเนื้อหาภายในเอกสาร โดยการค้นหาตาม keyword ที่กับเนื้อหาภายในเอกสารโดยโครงการนี้จะนำ Content มาจากเอกสารที่เก็บไว้ในฐานข้อมูลทั้งที่ได้มาจากการโหลดมาเก็บไว้กับการที่ User อัพโหลดไว้ และนำมาจากการ Request ค่าไปทาง URL แล้วส่งค่ากลับมาแบบเอกสาร ที่มี Content ที่ต้องการจากการ Search

3.5.3 Ontology

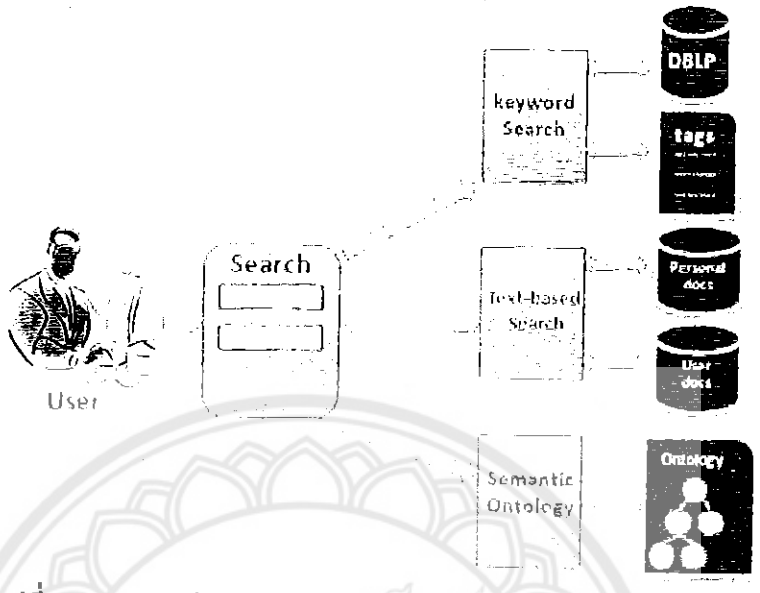
เป็น Search โดยใช้หลักการของ Ontology ซึ่งทำให้ผู้ใช้ค้นหาข้อมูลได้ตรงตามความต้องการยิ่งขึ้น และรับทราบข้อมูลต่างๆที่เกี่ยวข้องอีกด้วย

ตัวอย่างการ Search โดยใช้หลักการของ Ontology ดูได้จากรูปที่ 3.7 เมื่อเราต้องการหาข้อมูลของ Depth-first search แล้วหาไม่เจอ Search โดยใช้หลักการของ Ontology จะสามารถหาข้อมูลที่เกี่ยวข้อง Depth-first search ออกมาให้คือ Uninform Search ซึ่ง Depth-first search จะอยู่ใน Uninform Search ในโครงการนี้จะพัฒนาระบบ Search โดยใช้หลักของ Ontology มาช่วยเพิ่มประสิทธิภาพในการสืบค้นข้อมูลดังต่อไปนี้

1 .Is-a เป็นการสร้างการเชื่อมโยงระหว่างแหล่งข้อมูลซึ่งเป็นรายละเอียดปลีกย่อยของสิ่งที่เรากำลังพิจารณา เช่น การศึกษาในเรื่อง Uniform Search จะแสดงการเชื่อมโยงระหว่างข้อมูลปลีกย่อยที่เกี่ยวข้อง ซึ่งก็คือ Depth-first search , Breath-first search และ Uniform-cost search ซึ่งสามารถดูรูปที่ 3.7 เป็นตัวอย่าง

2. Related-to เป็นการเชื่อมโยงความสัมพันธ์ของข้อมูลแวดล้อมซึ่งอยู่ในระดับเดียวกัน เช่น Uniform search จะสัมพันธ์กับ Inform search เป็นต้น

3.5.4 การ Search ที่ใช้ข้อมูลจาก Metadata, Content และ Ontology



รูปที่ 3.14 โครงสร้างระบบการสืบค้นเอกสารอ้างอิงโดยใช้ Ontology

การค้นหารวมแบบ Meta Data , Content และ Ontology จากรูปที่ 3.14 จะเป็นการค้นหาของโครงงานนี้โดยจะเป็นโครงสร้างระบบการสืบค้นเอกสารอ้างอิงโดยใช้ Ontology มีหลักการทำงานดังนี้

เมื่อระบบรับค่า Query ที่ได้จากผู้ใช้ตัวอย่างเช่น รับค่า Query คือ AI DB นำคำของค่า Query ที่รับจากผู้ใช้มาสลับกันจะได้ AI DB กับ AI แล้วนำคำของค่า Query ที่รับจากผู้ใช้มาตัด เมื่อทำการตัดคำแล้ว จะได้ AI DB , AI , DB และ DB AI, DB,AI แล้วตัดคำที่ซ้ำกันออกจะได้ผลลัพธ์ดังนี้

AI DB , DB AI, AI , DB

จากนั้นนำคำที่ได้จากการตัดคำทั้งหมด เข้าไปทำการ Search ดังต่อไปนี้

1. Search แบบ Ontology โดยการนำแต่ละคำที่ได้มาหาค่า Rate ในไฟล์ Ontology ที่สร้างไว้ โดยมีการคำนวณค่า Rate ดังนี้

- นำคำที่ตัดได้มาทีละคำ และมาเทียบกับแต่ละโหนด ในไฟล์ Ontology เมื่อไม่พบคำๆ นั้นในไฟล์ให้ค่าคำนั้นนำไปค้นหาตามแบบ Content และ Meta data

- ตรงกับโหนดใดโหนดหนึ่งในไฟล์ Ontology เช่น AI จะมีค่าตรงกับโหนดที่ 2 ในไฟล์ Ontology

```
<!--Lv2-->
<concept name="artificial intelligence" id="2">
 <synoname name="ai"/>
 <subclassof id="1"/>
```

< hasChild id="4"/>

< hasChild id="5"/>

</concept>

ซึ่งค่าที่ได้จะเห็นได้ว่า AI เป็น subclassof ของโหนด 1 ใน Ontology นั้นก็คือ Computer Since และ มี hasof คือโหนด 4 และ 5 นั้นก็คือ search และ machine learning ซึ่งเราจะสามารถให้ค่า Rate ของแต่ละโหนดดังนี้

- ให้ค่า Rate ของแต่ละดังนี้

โหนดเริ่มต้น คือ AI = 1

subclassof คือ Computer Since = ค่า rate ของโหนด Level ต่ำสุด – 0.18

hasChild คือ search และ machine learning = ค่า rate ของ parent node -0.12

- จะต้องให้ค่า Rate ทุกโหนดที่เกี่ยวข้องกับโหนด AI โดยการคำนวณแบบเดียวกัน

- จากนั้นทำการค้นหาคำทีละโหนดโดย จำนวนคำที่ตรงกันทั้งหมดในเอกสาร แล้วนำมารวมกัน หารด้วยจำนวนคำในเอกสารทั้งหมดแล้วคูณด้วยค่า Rate ของแต่ละโหนด จากนั้นหารด้วยจำนวนโหนดทั้งหมด จะได้ Score ของ Ontology

2. Search แบบ Content คือ จำนวนคำที่เจอหารด้วยจำนวนคำทั้งหมดในเอกสารนั้น จะได้ Score ของ Content

3. Search แบบ Meta data คือ จำนวนของ tag ที่เจอหารด้วยจำนวน tag ทั้งหมด จะได้ Score ของ Meta data

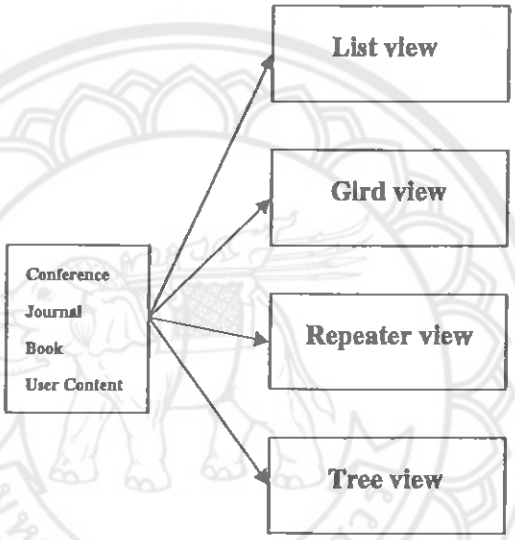
4. นำ Score ของ Ontology , Content และ Meta data มารวมกันแล้วหารด้วย 3 จะได้ Score รวมของคำว่า AI จากนั้นหา Score รวมของคำที่เหลือให้ครบตามขั้นตอนข้างต้น

5. เมื่อได้ Score ของ AI DB ,DB AI ,DB ,AI ครบแล้ว นำ Score ทั้งหมดมารวมกันแล้วหารด้วย 4 จะได้ค่า Score ของเอกสารนั้น

เมื่อทำการค้นหาข้อมูล (Searching) เพื่อนำมาแสดงให้กับผู้ใช้จะมีการจัดทำลำดับของเอกสาร (Ranking) ตามลำดับความเหมือนของเอกสารกับข้อความที่ต้องการค้นหา (Similarity) ซึ่งเอกสารที่เกี่ยวข้องกับผู้ใช้มากที่สุด (High Relevance) จะถูกแสดงอยู่ในลำดับต้น ๆ ซึ่ง การจัด ลำดับของเอกสารจะดูจากค่า Score ของเอกสารนั้น

3.6 ระบบนำทาง (Navigation)

Navigation เป็นส่วนที่ช่วยให้ผู้ใช้สามารถที่เข้าถึงส่วนต่างๆของเว็บไซต์ได้ง่ายและตรงตามความต้องการ อีกทั้งยังสามารถทราบถึงข้อมูลที่มีความสัมพันธ์กันของสิ่งที่ต้องการค้นคว้า ซึ่งในการ Navigation ใน Web Application จะเป็นการเชื่อมต่อกันของหน้าเพจต่างๆเข้าหากันนั้นก็คือ Hyperlink นั้นเอง และรูปแบบของการนำเสนอข้อมูลในหลายรูปแบบ ซึ่งโครงการได้ทำการนำเสนอใน 4 หัวข้อหลัก คือ Conference, Journal, Book, User Content ซึ่งในแต่ละหัวข้อ สามารถที่ Navigation ในการนำเสนอข้อมูลได้เป็น 4 แบบดังรูปที่ 3.15 เพื่อให้ผู้ใช้สามารถที่เข้าถึงส่วนต่างๆของเว็บไซต์ได้ง่ายและตรงตามความต้องการดังต่อไปนี้



รูปที่ 3.15 แสดงการ Navigation นำเสนอข้อมูลในรูปแบบต่างๆ

3.6.1 Gird view เป็นส่วนที่แสดงรายการที่เราสามารถที่จะ sort ข้อมูลต่างๆในตารางได้ เพื่อเพิ่มความสะดวกในการค้นหาข้อมูล ดังรูปที่ 3.16

3.6.2 Repeater view เป็นส่วนแสดงรายการข้อมูล ที่จะมีรายละเอียดของข้อมูลมาด้วยทำ ให้งานต่อการหาข้อมูลที่ต้องการรายละเอียดสูง ดังรูปที่ 3.17

3.6.3 Master detail view เป็นส่วนที่แสดงรายการทั้งหมดออกเป็น list ดังรูปที่ 3.18

3.6.4 Tree view เป็นการแสดงรายข้อมูลในรูปแบบของ Tree ทำให้สามารถค้นหาข้อมูลได้งานและเห็นภาพรวมโครงสร้างของข้อมูลทั้งหมด ดังรูปที่ 3.19

Book	Publisher	Year
A Data Definition Facility for Programming Languages	Garland Publishing, New York	1967
A Guide to DB2, 2nd Edition	Addison-Wesley	1988
A Guide to SQL Standard, 3rd Edition	Addison-Wesley	1993
A Guide to SQL Standard, 4th Edition	Addison-Wesley	1997
A Logical Language for Data and Knowledge Base.	Computer Science Press	1989
A Practical Formal Semantic Definition and Verification System for TYPED LISP	Garland Publishing, New York	1976
A Processor Design for the Efficient Implementation of APL	Garland Publishing, New York	1976
A Retargetable C Compiler: Design and Implementation	Addison Wesley	1995
A Tour on TriGS. Development of an Active System and Application of Rule Patterns for Active Database Design.	Infix Verlag, St. Augustin, Germany	1997
Abstract Interpretation of Declarative Languages	Ellis Horwood	1987
Access Control in Object-Oriented Federated Database Systems.	Infix Verlag, St. Augustin, Germany	1998
Adaptive Neuronale Netze und ihre Anwendung als Modelle der Entwicklung kortikaler Karten	Infix Verlag, St. Augustin, Germany	1993

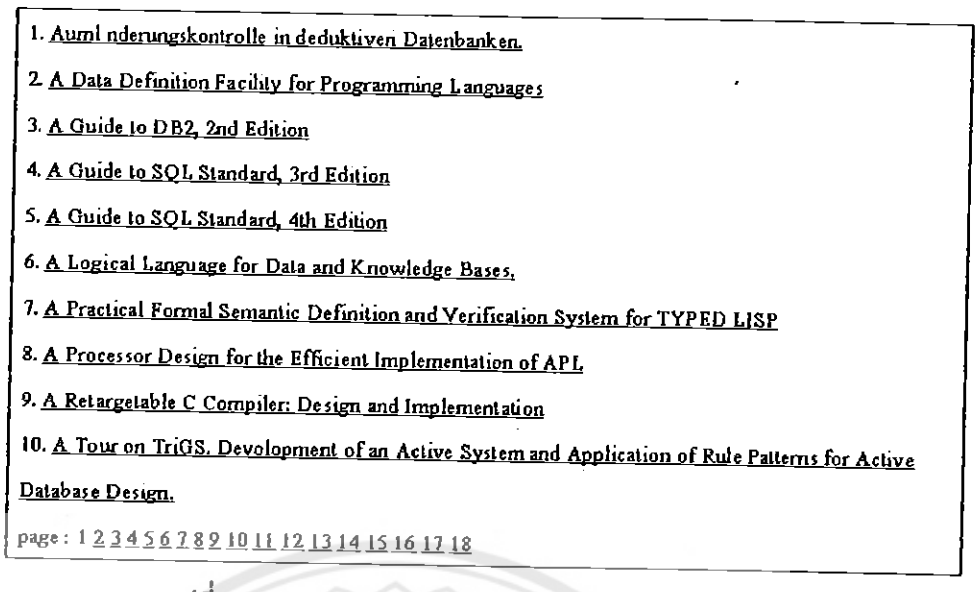
1 2 3 4 5 6 7 8 9 10 ...

รูปที่ 3.16 รูปแบบแสดงรายการข้อมูลแบบ Gird view

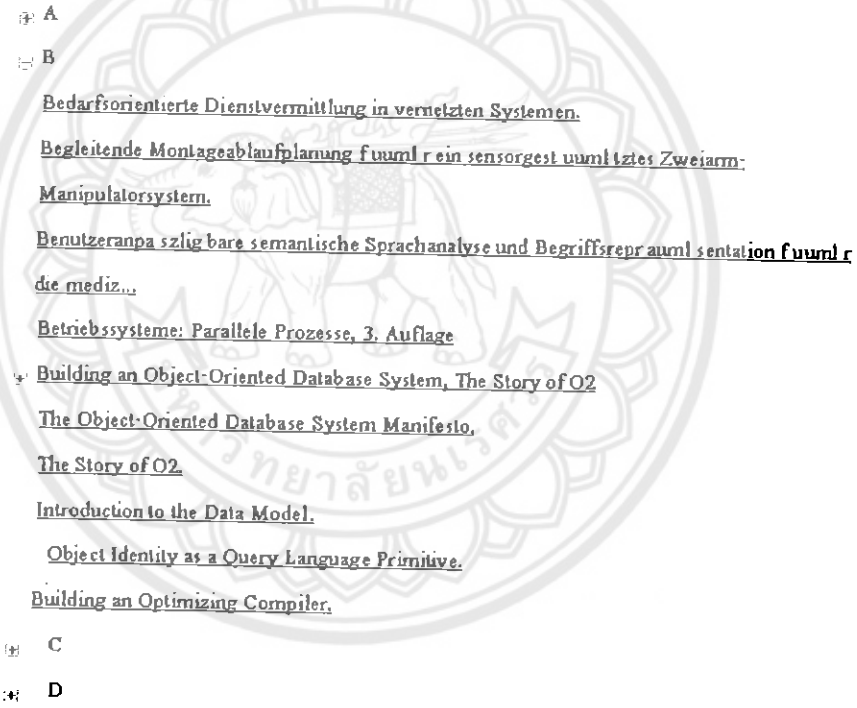
Book
Title : A Data Definition Facility for Programming Languages Author : Thomas A. Standish ISBN : 0-8240-4422-3 Publisher : Garland Publishing, New York Year : 1967
Title : A Guide to DB2, 2nd Edition Author : C. J. Date ISBN : 0-201-09428-2 Publisher : Addison-Wesley Year : 1988
Title : A Guide to SQL Standard, 3rd Edition Author : C. J. Date ISBN : 0-201-55822-X Publisher : Addison-Wesley Year : 1993

Page: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32

รูปที่ 3.17 รูปแบบแสดงรายการข้อมูลแบบ Repeater view



รูปที่ 3.18 รูปแบบแสดงรายการข้อมูลแบบ Master detail view



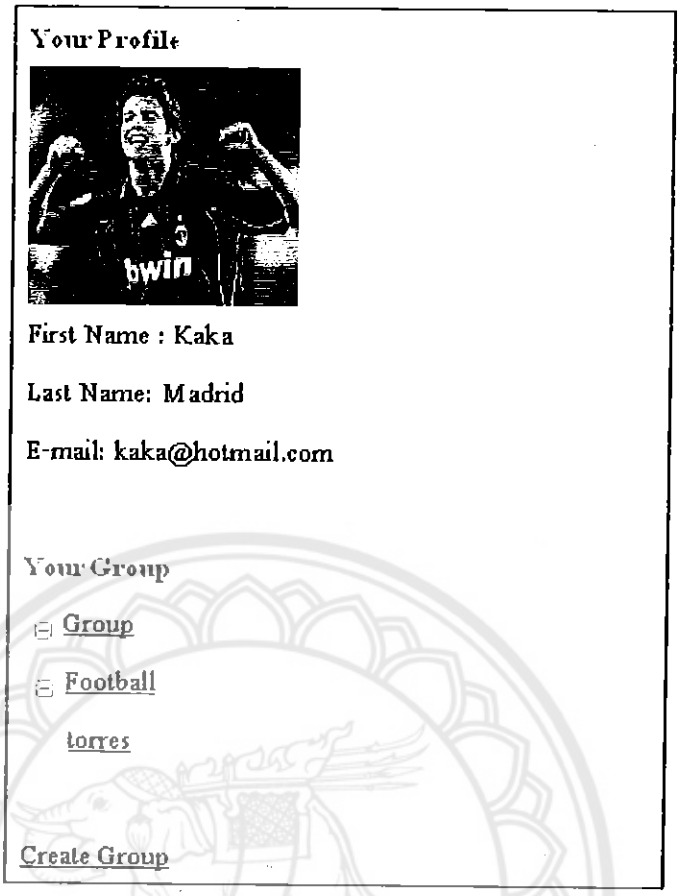
รูปที่ 3.19 รูปแบบแสดงรายการข้อมูลแบบ Tree view

3.7 การจัดการข้อมูลผู้ใช้ (User - Contributed Content Management)

เป็น ระบบที่สามารถให้ User ใช้ระบบระบบการสืบค้นเอกสารอ้างอิงโดยใช้ Ontology ได้ดียิ่งขึ้นโดยการ โดย User สามารถที่จะกระทำได้ดังต่อไปนี้

3.7.1 ประวัติผู้ใช้ (User Profile)

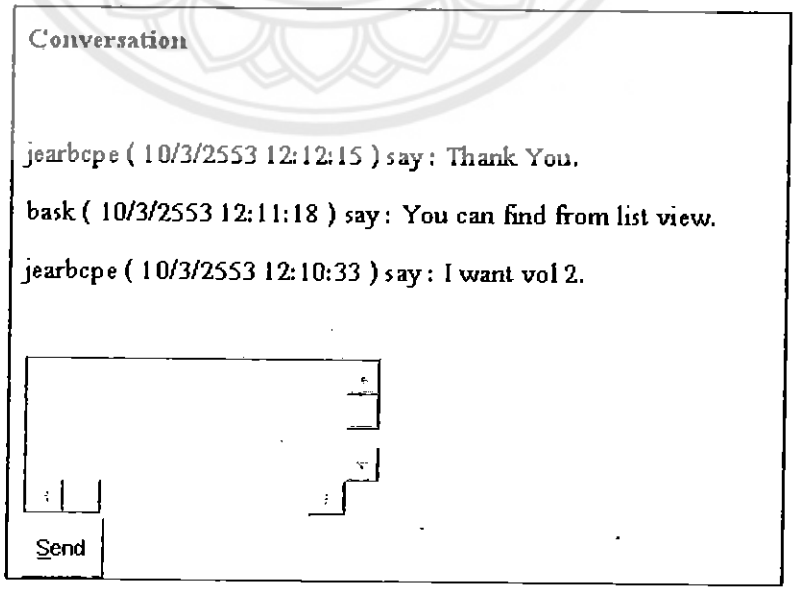
เป็นส่วนแสดงรายละเอียดข้อมูลของ User เมื่อ Login ข้อสู่ระบบ โดยแสดงรายละเอียดดังนี้ รูป ของ User ,First Name, Last Name, E-mail และ Your Group ของ USER ดังรูปที่ 3.20



รูปที่ 3.20 แสดง Information ของ User

3.7.2 การสนทนา (Conversation)

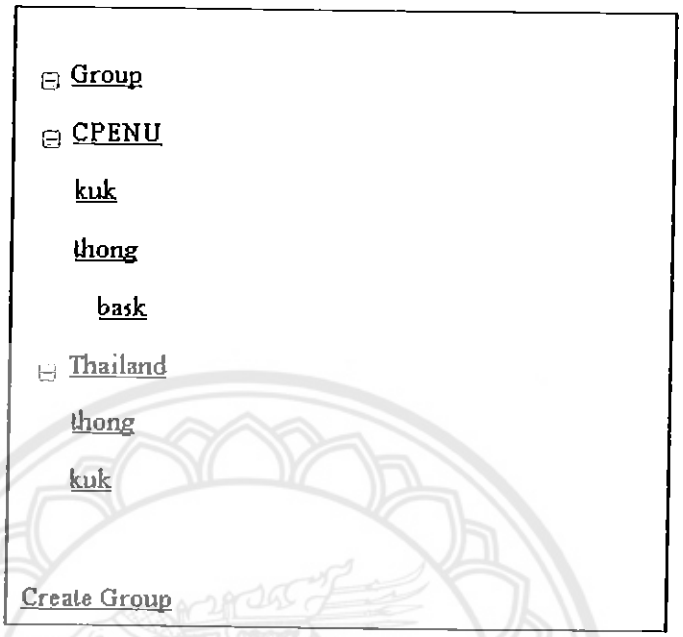
User สามารถที่จะสนทนาระหว่างกันได้ผ่านระบบ Conversation ดังรูปที่ 3.21



รูปที่ 3.21 ระบบ Conversations

3.7.3 กลุ่ม (Group)

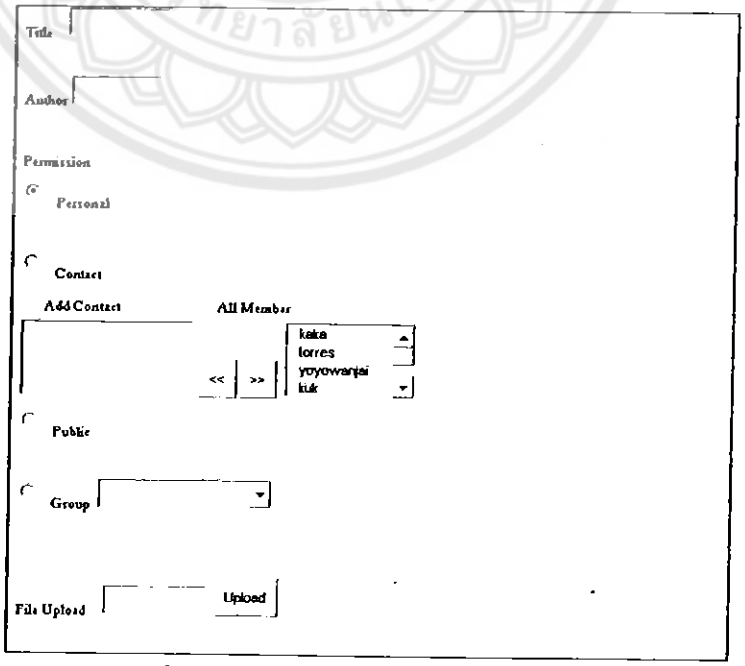
User สามารถที่จะสร้าง Group แล้วนำเพื่อนเข้ามาอยู่ใน Group ที่สร้างขึ้นได้ ดังรูปที่ 3.22



รูปที่ 3.22 แสดง Group ของ User

3.7.4 การแชร์ไฟล์ (File sharing)

User สามารถที่นำไฟล์มาอัพโหลดไว้ในระบบได้ และสามารถที่จะให้สิทธิของการเข้าดูไฟล์ของตนเองกับ User อื่นได้ ซึ่งมีส่วนประกอบดังรูปที่ 3.23



รูปที่ 3.23 แสดงการอัพโหลดไฟล์ ของ User

1) Title ชื่อเรื่อง

2) Author ชื่อผู้เขียน

3) Permission การให้สิทธิในการเข้าดูไฟล์ที่อัพโหลด

- *Personal* ให้สิทธิเฉพาะตัวเอง
- *Contact* ให้สิทธิเฉพาะ User ที่เรา Add เข้ามา
- *Public* ให้สิทธิทุก User
- *Group* ให้สิทธิเฉพาะ User ที่อยู่ใน Group ที่เลือก

4) File Upload เลือกไฟล์ที่ต้องการอัพโหลด

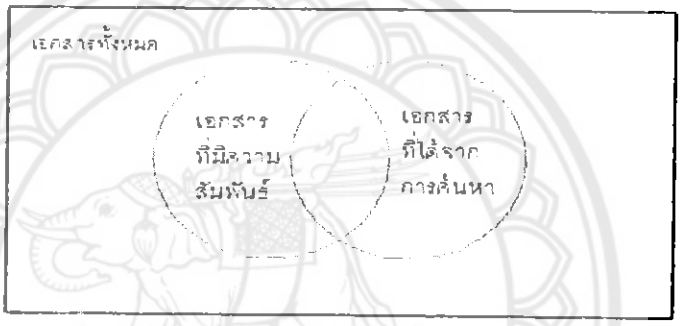
บทที่ 4

ผลการทดลอง

4.1 การค้นหา (Search)

4.1.1 วิธีการทดสอบ

สำหรับการทดสอบประสิทธิภาพของระบบการ Search นั้นจะใช้การทดสอบในรูปแบบ Precision และ Recall จะใช้การวัดความแม่นยำ (Precision) และความถูกต้อง (Recall) จุดประสงค์เพื่อ ใช้ตรวจสอบว่าระบบที่พัฒนาขึ้นสามารถค้นคืนเอกสารได้อย่างมีประสิทธิภาพดังรูปที่ 4.1



รูปที่ 4.1 แสดงเซตเอกสารทั้งหมด [18]

a) Precision คือสัดส่วนของจำนวนของเอกสารที่เกี่ยวข้องที่ค้นคืนได้เมื่อเปรียบเทียบกับจำนวนของเอกสารทั้งหมดที่ค้นออกมา

ความแม่นยำ คืออัตราส่วนระหว่างเซตของเอกสารทั้งได้จากการค้นคืนดังรูปที่ 4.2

Precision = # correct classes found / # correct found

รูปที่ 4.2 สมการการหา Precision

b) Recall คือสัดส่วนของจำนวนของเอกสารที่เกี่ยวข้องและค้นคืนได้เมื่อเปรียบเทียบกับจำนวนเอกสารที่เกี่ยวข้องทั้งหมดในฐานข้อมูล

ความถูกต้องคืออัตราส่วนระหว่างเซตของเอกสารจากการค้นคืนเอกสารดังรูปที่ 4.3

Recall = # correct classes found / # classes found

รูปที่ 4.3 สมการการหา Recall

จากรายละเอียดเกณฑ์ในประเมินประสิทธิภาพของระบบคือค่า F-measure โดยหาได้จากสมการดังรูปที่ 4.4

F-measure = (2 x Precision x Recall)/(Precision + Recall)

รูปที่ 4.4 สมการการหา F-measure

4.1.2 ผลการทดสอบ

1. Metadata

ตารางที่ 4.1 ตารางทดสอบเพื่อวัดประสิทธิภาพของระบบการค้นหาเอกสารอ้างอิงโดยการค้นหาแบบ Metadata

คำที่ใช้ในการค้นหา	เอกสาร			Precision	Recall	F-measure
	correct classes found	correct found	classes found			
Search	23	25	65	0.92	0.38	0.54
dbms	5	5	51	1	0.09	0.17
query	18	24	60	0.75	0.3	0.43
sql	5	5	30	1	0.17	0.29
XPath	0	0	22	0	0	0
สรุป	10.2	11.8	45.6	0.73	0.18	0.29

2. Content

ตารางที่ 4.2 ตารางทดสอบเพื่อวัดประสิทธิภาพของระบบการค้นหาเอกสารอ้างอิงโดยการค้นหาแบบ Content

คำที่ใช้ในการค้นหา	เอกสาร			Precision	Recall	F-measure
	correct classes found	correct found	classes found			
Search	35	56	65	0.63	0.53	0.57
dbms	3	6	51	0.50	0.05	0.09
query	11	21	60	0.52	0.18	0.27
sql	6	10	30	0.60	0.20	0.30
XPath	0	0	22	0	0	0
สรุป	11	18.6	45.6	0.45	0.19	0.25

3. Ontology

ตารางที่ 4.3 ตารางทดสอบเพื่อวัดประสิทธิภาพของ
ระบบการค้นหาเอกสารอ้างอิงโดยการค้นหาแบบ Ontology

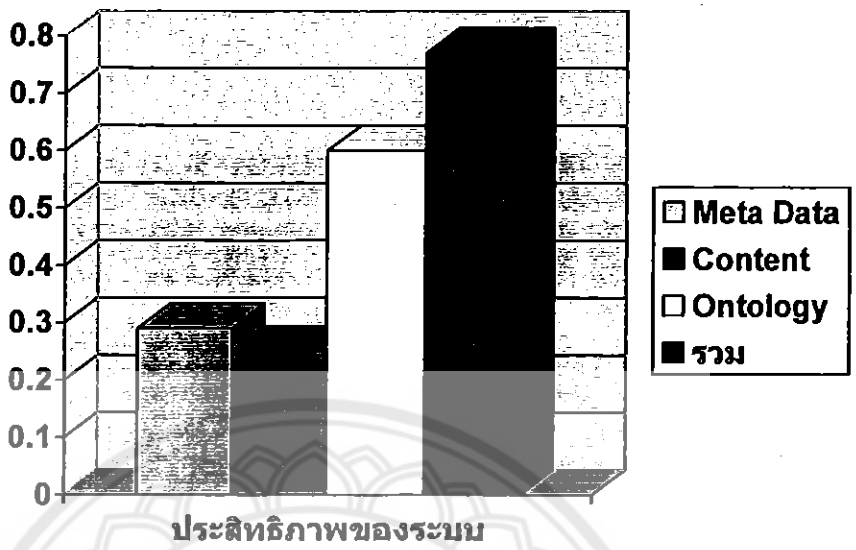
คำที่ใช้ในการค้นหา	เอกสาร			Precision	Recall	F-measure
	correct classes found	correct found	classes found			
Search	41	58	65	0.70	0.63	0.66
dbms	38	52	51	0.73	0.74	0.73
query	32	48	60	0.66	0.53	0.59
sql	19	48	30	0.39	0.63	0.48
XPath	18	48	22	0.37	0.81	0.50
สรุป	29.6	50.8	45.6	0.58	0.64	0.60

4. การ Search ที่ใช้ข้อมูลจาก Metadata, Content และ Ontology

ตารางที่ 4.4 ตารางทดสอบเพื่อวัดประสิทธิภาพของระบบการค้นหา
เอกสารอ้างอิงโดยการค้นหาแบบ การ Search ที่ใช้ข้อมูลจากMetadata, Content และ Ontology

คำที่ใช้ในการค้นหา	เอกสาร			Precision	Recall	F-measure
	correct classes found	correct found	classes found			
Search	63	79	65	0.79	0.96	0.87
dbms	49	55	51	0.89	0.96	0.92
query	56	71	60	0.78	0.93	0.85
sql	28	52	30	0.53	0.93	0.68
XPath	18	48	22	0.37	0.81	0.51
สรุป	42.8	61	45.6	0.70	0.93	0.77

4.1.3 สรุปผลการทดสอบ



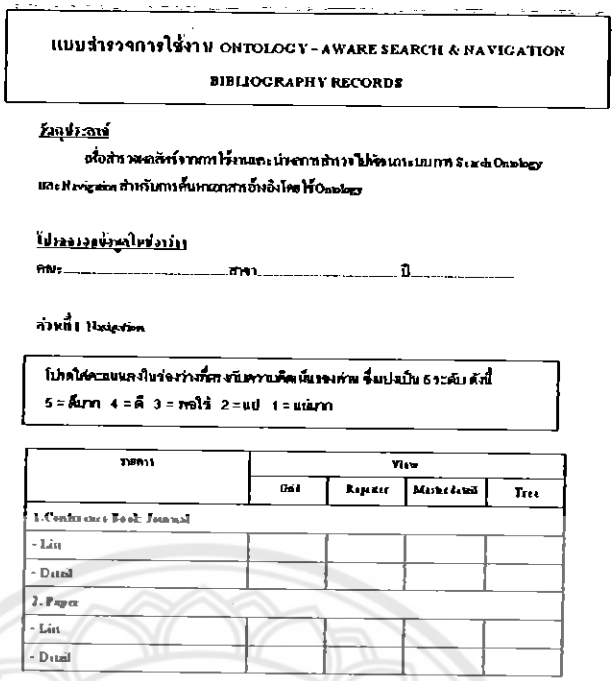
รูปที่4.5 ผลการวัดประสิทธิภาพของระบบการ Search

จากรูปที่ 4.5 จะเห็นได้ว่าประสิทธิภาพของการ Search แบบใช้ metadata , content และ ontology มีความแตกต่างกัน โดยการ search แบบ ontology นั้นมีประสิทธิภาพที่แตกต่างอย่างชัดเจน แต่เมื่อใช้การ search ทั้ง 3 แบบร่วมกัน ก็จะทำให้การค้นหาเอกสารมีประสิทธิภาพมากยิ่งขึ้น ซึ่งค่าประสิทธิภาพของระบบที่พัฒนาขึ้นคือ 77% จึงสรุปได้ว่าประสิทธิภาพในการค้นหาอยู่ในระดับที่ดี

4.2 ระบบนำทาง (Navigation)

4.2.1 วิธีการทดสอบ

สำหรับการทดสอบประสิทธิภาพของระบบการ Navigation นั้นจะใช้การทดสอบในรูปแบบ แบบสอบถามข้อมูลและแสดงความคิดเห็นดังรูปที่ 4.6



แบบสำรวจการใช้งาน ONTOLOGY - AWARE SEARCH & NAVIGATION

BIBLIOGRAPHY RECORDS

วัตถุประสงค์

เพื่อสำรวจผลลัพธ์จากการใช้งานและนำผลการสำรวจไปพัฒนาระบบการ Search Ontology และ Navigation สำหรับการค้นหาเอกสารอ้างอิงโดยใช้Ontology

โปรดลองใช้งานก่อนตอบแบบสำรวจ

คณะ_____สาขา_____ปี_____

ส่วนที่ 1 Navigation

โปรดให้คะแนนลงในช่องว่างที่ตรงกับความคิดเห็นของท่าน ซึ่งแบ่งเป็น 5 ระดับ ดังนี้

5 = ดีมาก 4 = ดี 3 = พอใช้ 2 = แย่ 1 = แย่มาก

รายการ	View			
	Grid	Repeater	Master detail	Tree
1.Conference/Book/Journal				
- List				
- Detail				
2. Paper				
- List				
- Detail				

รูปที่ 4.6 แบบสอบถามความคิดเห็นระบบการสืบค้นเอกสารอ้างอิงโดยใช้ Ontology

4.2.2 ผลการทดสอบ

ตารางที่ 4.5 ตารางแบบสอบถามความคิดเห็นที่มีต่อการ Navigation

รายการ	View			
	Grid	Repeater	Master detail	Tree
1.Conference/Book/Journal				
- List	52%	84%	74%	86%
- Detail	76%	74%	82%	72%
2. Paper				
- List	50%	72%	74%	90%
- Detail	62%	60%	66%	66%

จำนวนผู้โหวตทั้งหมด : 10

ข้อเสนอแนะจากแบบสอบถามความคิดเห็น

1.อยากให้สีตัวหนังสือเปลี่ยน เวลาคลิ๊กลิงค์ไปแล้ว

4.2.3 สรุปผลการทดสอบ

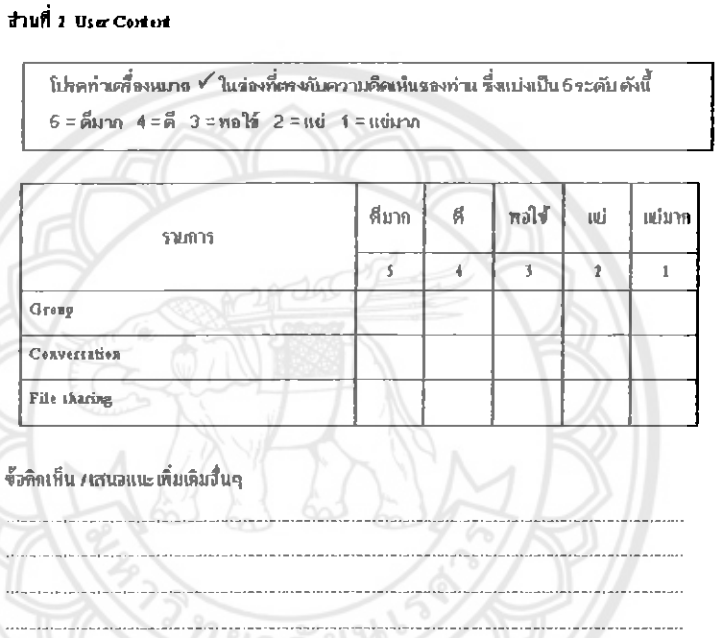
จากผลการทดลองจะเห็นว่าการใช้ระบบ Navigation ในการค้นหานั้น Tree view จะสามารถทำให้ผู้ใช้สามารถที่จะค้นหาเอกสารได้ง่ายขึ้น กล่าวคือผู้ใช้จะเห็นภาพรวมของลำดับ

เอกสาร และสามารถเข้าถึงเอกสารได้สะดวกรวดเร็ว แต่การเข้าถึงรายละเอียดของข้อมูลนั้น จะไม่เห็นความแตกต่างกันมากนัก เนื่องจาก การแสดงผลข้อมูลมีรูปแบบคล้ายคลึงกัน

4.3 การจัดการข้อมูลผู้ใช้ (User - Contributed Content Management)

4.3.1 วิธีการทดสอบ

สำหรับการทดสอบประสิทธิภาพของระบบการ User Content นั้นจะใช้การทดสอบในรูปแบบ แบบสอบถามข้อมูลและแสดงความคิดเห็นดังรูปที่ 4.7



ส่วนที่ 2 User Content

โปรดทำเครื่องหมาย ✓ ในช่องที่ตรงกับความคิดเห็นของท่าน ซึ่งแบ่งเป็น 6 ระดับ ดังนี้
6 = ดีมาก 4 = ดี 3 = พอใช้ 2 = แย่ 1 = แย่มาก

รายการ	ดีมาก	ดี	พอใช้	แย่	แย่มาก
	5	4	3	2	1
Group					
Conversation					
File sharing					

ข้อคิดเห็น /เสนอแนะเพิ่มเติมอื่นๆ

รูปที่ 4.7 แบบสอบถามความคิดเห็นระบบการ User - Contributed Content Management

4.3.2 ผลการทดสอบ

ตารางที่ 4.6 ตารางแบบสอบถามความคิดเห็นที่มีต่อการ User - Contributed Content Management

รายการ	ดีมาก	ดี	พอใช้	แย่	แย่มาก
	5	4	3	2	1
Group	40%	50%	10%	0%	0%
Conversation	60%	40%	0%	0%	0%
File sharing	90%	10%	0%	0%	0%

จำนวนผู้โหวตทั้งหมด : 10

ข้อเสนอแนะจากแบบสอบถามความคิดเห็น

1. อยากให้ user ส่ง message หากันได้

4.3.3 สรุปผลการทดสอบ

จะเห็นว่าระบบการแชร์ไฟล์นั้น ตอบสนองความต้องการของผู้ใช้ได้ดีมาก เนื่องจากการผู้ใช้ได้มีส่วนร่วมกับการนำเสนอผลงาน รวมทั้งยังสามารถที่จะจำกัดสิทธิ์เฉพาะกลุ่มได้ ทำให้ข้อมูลของผู้ใช้มีความปลอดภัย นอกจากนี้ ระบบ Conversation ยังทำให้ผู้ใช้พึงพอใจได้ดีระดับหนึ่ง เพราะผู้ใช้สามารถที่จะสอบถามข้อสงสัยหรือให้คำแนะนำแก่บุคคลอื่นๆได้

บทที่ 5
บทสรุป

5.1 สรุปผลการดำเนินการ

โครงการนี้พัฒนาขึ้นเพื่อศึกษาและทดลองเกี่ยวกับหลักการทำงานของการค้นหาเอกสารอ้างอิงโดยใช้ Ontology ซึ่งมุ่งประเด็นไปที่การค้นหาข้อมูลที่ต้องการแล้วแสดงผลให้ได้ตรงตามความหมายที่ผู้ใช้งานต้องการ โดยการใช้ Ontology เข้ามาช่วยใน Search เพื่อเปรียบเทียบ ความหมายคำที่ต้องการค้นหาจากฐานข้อมูล เพื่อให้ได้เอกสารข้อมูลที่ถูกต้องมากที่สุด แล้วทำการแสดงผลออกมาให้โดยเรียงลำดับเอกสารข้อมูลตามลำดับความสำคัญ แล้วสามารถที่จะเข้าไปหาข้อมูลต่อไปได้ทั้งในรูปเอกสารและการลิงค์ไปเว็บที่เกี่ยวข้อง และระบบ Navigation จะเป็นส่วนที่ช่วยในการนำเสยอข้อมูลในรูปแบบต่างๆ ให้ผู้ใช้สามารถที่เข้าถึงส่วนต่างๆของเว็บไซต์ได้ง่ายและตรงตามความต้องการ อีกทั้งยังสามารถทราบถึงข้อมูลที่มีความสัมพันธ์กันของสิ่งที่ต้องการค้นคว้าอีกด้วย

การพัฒนาระบบการค้นหาเอกสารอ้างอิงโดยใช้ Ontology สามารถสรุปได้ว่าระบบการสืบค้นหาเอกสารอ้างอิงโดยใช้ Ontology สามารถทำได้ดี และตรงกับความต้องการของผู้ใช้งานในการค้นหา และผลของการค้นหาเอกสารอ้างอิงจะขึ้นอยู่กับขนาดของคำที่ใช้ในการค้นหาและขนาดของฐานข้อมูล

โดยโครงงานนี้เป็นโปรแกรมที่พัฒนาขึ้นโดยใช้ Microsoft Visual C# .NET และ ASP.NET

5.2 ปัญหาและอุปสรรคในการพัฒนา

5.2.1 ผลของการทำงานการค้นหาเอกสารอ้างอิงแปรผันกับการคำที่ใช้ในการค้นหา และขนาดของฐานข้อมูลที่ค้นหา ถ้าจำนวนคำในการค้นหาและฐานข้อมูลมีขนาดใหญ่ จะทำให้การค้นหาใช้เวลานานในการค้นหา

5.2.2 องค์ประกอบและสิ่งแวดล้อมของฐานข้อมูลส่งผลต่อคุณสมบัติของการค้นหาที่ได้ เช่น ขนาดของไฟล์ , จำนวนคำในไฟล์ มีการเปลี่ยนแปลง ทำให้ผลการค้นหาที่ได้เปลี่ยนไป

5.3 ข้อเสนอแนะ

เนื่องจากโครงงานนี้ เป็นการเริ่มพัฒนาในขั้นแรก ทำให้อาจจะยังมีข้อผิดพลาดของระบบการทำงานอยู่บ้าง หากผู้ที่สนใจพัฒนาโครงงานในขั้นต่อ ๆ ไป อาจมีข้อเสนอแนะดังนี้

5.3.1 การเพิ่มทฤษฎีส่วนการค้นหาภาษาไทยจะใช้กระบวนการตัดคำ (Word Segmentation) ซึ่งเป็นการแบ่งตัวอักษรจากข้อความ (String) เพื่อหาขอบเขตของแต่ละหน่วยคำ(Morpheme) เนื่องจากส่วน

ใหญ่ภาษาไทยมีการเขียนในลักษณะที่ติดกันโดยไม่มีการใช้เครื่องหมายวรรคตอนคั่นระหว่างคำ เหมือนภาษาอังกฤษ ซึ่งใช้ช่องว่าง (Space) คั่นระหว่างคำ แต่ภาษาไทยจะมีการเว้นวรรคเป็นระยะ เพื่อให้ผู้อ่านทำความเข้าใจกับความหมายของคำ โดยในที่นี้ใช้การตัดคำส่วนภาษาไทย โดยอ่านตัวอักษรทีละอักขระ แล้วนำไปเทียบกับพจนานุกรมที่ชื่อว่า Lexitron จึงจะทำการตัดคำตามพจนานุกรม จะได้คำตามพจนานุกรมทั้งเอกสาร

5.3.2 มีการแปลงคำให้อยู่ในรากศัพท์ (Stemming) โดยทำให้เป็นคำ ๆ เดียว ซึ่งส่วนนี้จะใช้กับภาษาอังกฤษ เช่น "run" "ran" "running" จะถูกตัดให้เป็นคำว่า "run" คำเดียว เป็นต้น

5.3.3 การเพิ่มสรรถนะของการค้นหาให้รองรับขนาดไฟล์ฐานข้อมูลใหญ่ๆ ได้แต่ทั้งนี้ขึ้นกับเครื่องคอมพิวเตอร์ที่ใช้รันโปรแกรมด้วย

5.4 แนวทางในการพัฒนาต่อในอนาคต

5.4.1 สามารถประยุกต์ใช้กับการสืบค้นอื่น ๆ นอกจากเอกสารอ้างอิงได้

5.4.2 สามารถที่จะเพิ่มวิธีการในการค้นหาให้ได้เอกสารที่ตรงตามความต้องการได้มากขึ้น โดยอาศัยคุณสมบัติอื่นๆ เช่น Text Mining,N-Gram

5.4.3 หากต้องการเพิ่มความถูกต้องของโปรแกรม สามารถทดลองทำการค้นหาซ้ำหลาย ๆ ครั้งเพื่อหาค่า Rate ที่เหมาะสมได้ หรือเพิ่มวิธีการในการเปรียบเทียบในการค้นหาแบบ Ontology เพื่อให้มีความแม่นยำมากยิ่งขึ้น

เอกสารอ้างอิง

[1] Sookna. "XML ย่อมาจาก Extensible Markup Language." [Online]. Available : http://www.sookna.com/article_show.php?Category=abbreviation&No=313. 2004.

[2] นาย ศุภชัย สมพานิช. **เรียนรู้ LINQ ฉบับโปรแกรมเมอร์.** พิมพ์ครั้งที่ 1. นนทบุรี : บริษัท ไอดีซี อินโฟ ดิสทริบิวเตอร์ เซ็นเตอร์ จำกัด. 2552.

[3] Banpote_tt. "เบื้องต้นเกี่ยวกับ LINQ To SQL." [Online]. Available : http://greatfriends.biz//webboards/msg.asp?id=50209 2007.

[4] Adminbird. "เมื่อ LINQ มาทำให้ย้ายการ Query มาที่ภาษาโปรแกรมแทนการใช้ SQL." [Online]. Available : http://www.namtarnsci.com/cs12/?p=66. 2009.

[5] Webmaster. "การทำ Mapping ของ LINQ to SQL." [Online]. Available : http://greatfriends.biz/webboards/msg.asp

[6] นาย ศุภชัย สมพานิช. **เริ่มต้นอย่างมืออาชีพ ASP.NET 3.5.** พิมพ์ครั้งที่ 1. นนทบุรี : บริษัท ไอดีซี อินโฟ ดิสทริบิวเตอร์ เซ็นเตอร์ จำกัด. 2552

[7] พร้อมเดิศ หล่อวิจิตร. **PHP และ My SQL สำหรับผู้เริ่มต้น.** พิมพ์ครั้งที่ 1. นนทบุรี : บริษัท ไอดีซี อินโฟ ดิสทริบิวเตอร์ เซ็นเตอร์ จำกัด. 2552.

[8] Webmaster. "ข้อดี ข้อเสีย PHP กับ ASP." [Online]. Available : http://www.narisa.com/forums/index.php?showtopic=10187 2005.

[9] Webmaster. "JSP - Java Server Page." [Online]. Available : http://www.itmelody.com/tu/introj-sp.htm. 2005.

[10] Webmaster. "ความรู้พื้นฐานเกี่ยวกับ MVC." [Online]. Available : http://wiki.nectec.or.th/setec/Knowledge/MVC. 2008.

[11] Webmaster. "โครงสร้างข้อมูลแบบฐานข้อมูล (Database Model)." [Online]. Available : http://www.indyza.com/thread-1676-1-1.html. 2009.

[12] Webmaster. "Taxonomy & Ontology." [Online]. Available : http://www.narisa.com/forums/index.php?showtopic=17578. 2007.

[13] Webmaster. "Ontology คืออะไร." [Online]. Available : http://pi.eng.src.ku.ac.th/mod/forum/discuss.php?d=119. 2005.

[14] Webmaster. "What is the Semantic Web?" [Online]. Available : http://www.altova.com/semantic_web.html. 2009.

เอกสารอ้างอิง(ต่อ)

[15] นางสาวกัตติกา มีพรหม. "**Web Crawler and Search Engine**". วิศวกรรมคอมพิวเตอร์. วิทยาลัยวิศวกรรมศาสตร์ มหาวิทยาลัยรังสิต. 2552.

[16] ธนาทิพ์ ญาณศิลป์. **"Search Engine Pan & James Search."** เทคโนโลยีสารสนเทศ. คณะเทคโนโลยีสารสนเทศ. 2549.

[17] สุวัฒนา สุขสมจินดน์. **คัมภีร์การใช้ XMLฉบับสมบูรณ์.** พิมพ์ครั้งที่ 1. กรุงเทพ : บริษัท ซีเอ็ดยูเคชั่นจำกัด. 2545.

[18] สิทธิโชค ปัญญาฤกษ์ชัย , ศิพาณี นุชิตประสิทธิ์ชัย. "**ระบบการค้นคืนสารสนเทศโดยใช้เทคนิค N-Gram**" วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ. สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ. 2009.

ประวัติผู้เขียนโครงงาน

ชื่อ นายกิตติกร หวานใจ

ภูมิลำเนา 131 ม. 6 ต.บ้านสาง อ.เมือง จ.พะเยา 56000

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนฟากกว๊านวิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยนเรศวร

E-mail : **yoyowanjai@hotmail.com**

ชื่อ นายธีระพล สันติสำราญวิไล

ภูมิลำเนา 129/3 ถ.โกสีย์ ต.ปากน้ำโพ อ.เมืองนครสวรรค์ จ. นครสวรรค์ 60000

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนลาซาลโชติรวีนครสวรรค์
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยนเรศวร

E-mail : **teerapon_jearb@hotmail.com**