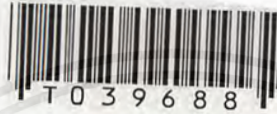


ตัวแปลงภาษารูปแบบ XSL

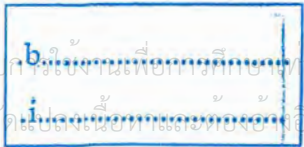
XSL TRANSLATOR



ธนา ถ้องพิพัฒน์
ศศิธร เมธาวี
อภิชัย เจียรกุล

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2543

เลขหมู่.....
เลขทะเบียน 39688
วัน, เดือน, ปี 19 สิงหาคม 2544



ไว้สำหรับ..... ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
..... ห้ามมิให้คัดลอกหรือเผยแพร่โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XSL TRANSLATOR



THANA THONGPIPAT
SASITORN MATHAWEE
APICHAJ CHIARAKUL

A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE
DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCES
FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2000

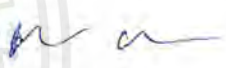


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

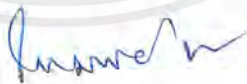
หัวข้อปัญหาพิเศษ ตัวแปลงภาษารูปแบบ XSL
 XSL TRANSLATOR

ชื่อนักศึกษา นายธนา ถ้องพิพัฒน์ 40056035
 นางสาวศศิธร เมธาวิ 40056084
 นายอภิชัย เจียรกุล 40056109

ภาควิชา คณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
 สาขาวิชา วิทยาการคอมพิวเตอร์
 อาจารย์ที่ปรึกษา อาจารย์วิสันต์ ตั้งวงษ์เจริญ

ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นำปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิทยาศาสตร์บัณฑิต สาขาวิทยาการคอมพิวเตอร์ ประจำปีการศึกษา 2543

คณะกรรมการสอบ	ลายมือชื่อ
ประธานกรรมการ ผู้ช่วยศาสตราจารย์ธีรวัฒน์ ประกอบผล	
กรรมการ ผู้ช่วยศาสตราจารย์ไพโรบลย์ พันธรัักษ์พงษ์	
กรรมการและอาจารย์ที่ปรึกษา อาจารย์วิสันต์ ตั้งวงษ์เจริญ	



(ผู้ช่วยศาสตราจารย์ไพโรบลย์ พันธรัักษ์พงษ์)

หัวหน้าภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

ลิขสิทธิ์ของภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	ตัวแปลงภาษารูปแบบ XSL	
ชื่อนักศึกษา	นายธนา ถ้องพิพัฒน์	40056035
	นางสาวศศิธร เมธาวี	40056084
	นายอภิชัย เจียรกุล	40056109
ปริญญา	วิทยาศาสตรบัณฑิต	
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์	
สาขาวิชา	วิทยาการคอมพิวเตอร์	
ปีการศึกษา	2543	
อาจารย์ที่ปรึกษา	อาจารย์วิสันต์ ตั้งวงษ์เจริญ	

บทคัดย่อ

ในปัจจุบัน เอกสารสำหรับนำเสนอข้อมูลบนเครือข่ายอินเทอร์เน็ตมีมากมายแตกต่างกันไปตามอุปกรณ์ที่ใช้แสดงผล ดังเช่น เอกสาร HTML สำหรับ Web browser และเอกสาร WML สำหรับโทรศัพท์มือถือ การจัดทำเอกสารเพื่อให้สามารถแสดงผลได้ทั้งบน Web browser และบนโทรศัพท์มือถือ จึงทำให้เกิดแนวทางการพัฒนาเอกสาร HTML และเอกสาร WML โดยสร้างเอกสารเพียงชุดเดียว

ตัวแปลงภาษารูปแบบ XSL สร้างขึ้นด้วยภาษา Java เป็นโปรแกรมที่ช่วยประหยัดเวลาในการสร้างเอกสารเพื่อนำเสนอข้อมูล ทำหน้าที่สร้างเอกสาร WML จากเอกสาร HTML และสร้างเอกสาร HTML จากเอกสาร WML โดยการนำเอกสารอินพุตมาเปรียบเทียบกับกฎการปรับเปลี่ยนโครงสร้างเอกสารที่เขียนขึ้นในเอกสาร XSL เพื่อสร้างเอกสารเอาต์พุต ซึ่งการดำเนินการของตัวแปลงภาษารูปแบบ XSL เป็นไปโดยอัตโนมัติ กล่าวคือ ถ้าเอกสารเอกสารอินพุตเป็นเอกสาร HTML ผลลัพธ์ที่ได้คือ เอกสาร WML ในทางกลับกัน ถ้าเอกสารอินพุตเป็นเอกสาร WML ผลลัพธ์คือ เอกสาร HTML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Project Title	XSL TRANSLATOR		
Students	Mister Thana Thongpipat	40056035	
	Miss Sasitorn Mathawee	40056084	
	Mister Apichai Chiarakul	40056109	
Degree	Bachelor's Degree of Science		
Department	Mathematics and Computer Sciences, Faculty of Science		
Programme	Computer Sciences		
Academic Year	2000		
Special Project Advisor	Lecturer Wisan Tangwongcharoen		

ABSTRACT

Nowadays, there are several online documents via internet depending on whatever device it is supposed to be displayed with, such as HTML Document for Web browser and WML for mobile phone. After all, it becomes the new trace for developing both HTML and WML Documents using just one stand alone document.

XSL Translator is a programme which enhanced by JAVA Programming Language. It's not only been used for translating HTML-based document to WML-based document, but in reverse also. With XSL-base document, we use its structure to be the standard for mapping input through output. Automatically, if there's a HTML input, the target output would be WML. In the contrary, WML input would have HTML output as well.

กิตติกรรมประกาศ

ในการทำปัญหาพิเศษเรื่อง ตัวแปลงภาษารูปแบบ XSL สามารถสำเร็จลุล่วงไปได้ด้วยดี ทางคณะผู้จัดทำต้องขอขอบพระคุณ ท่านอาจารย์วิสันต์ ตั้งวงษ์เจริญ ซึ่งเป็นอาจารย์ผู้รับผิดชอบ ปัญหาพิเศษฉบับนี้ ที่กรุณาให้คำแนะนำและเป็นทีปรึกษาในการแก้ปัญหาต่าง ๆ รวมทั้งเป็นผู้ ตรวจสอบความถูกต้องของปัญหาพิเศษฉบับนี้

ท้ายที่สุด ทางคณะผู้จัดทำต้องขอขอบพระคุณ บิดา มารดา ที่ได้ให้ความสนับสนุนทางด้านกำลังใจและทุนทรัพย์ในการทำปัญหาพิเศษ รวมถึง เพื่อน พี่และน้อง ผู้คอยให้กำลังใจแก่ คณะผู้จัดทำ จนการทำปัญหาพิเศษนี้สำเร็จด้วยดีไว้ ณ ที่นี้

คณะผู้จัดทำ

มีนาคม 2544



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญภาพ.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	1
1.3 ทฤษฎีหรือแนวคิดที่ใช้ในการศึกษา.....	1
1.4 ขอบเขตของการศึกษา.....	1
1.5 ขั้นตอนการศึกษา.....	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 ภาษา XML (eXtensible Markup Language)	3
2.1.1 โครงสร้างประโยคพื้นฐานในภาษา XML.....	3
2.1.1.1 ตัวอักษร (Characters).....	3
2.1.1.2 ชื่อ (Names).....	4
2.1.1.3 อีลิเมนต์ (Element).....	5
2.1.1.4 แอตทริบิวต์ (Attributes).....	6
2.1.1.5 พื้นที่ว่าง (White Space).....	7
2.1.1.6 หมายเหตุ (Comment).....	7
2.1.2 โครงสร้างของเอกสาร XML.....	8
2.1.3 ตัวอย่างเอกสาร XML.....	9
2.2 ไวยากรณ์ DTD (Document Type Definition)	9
2.2.1 โครงสร้างเอกสาร DTD.....	9
2.2.1.1 อีลิเมนต์ (Element).....	9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.2.1.2 แอตทริบิวต์ (Attribute).....	10
2.2.1.3 เอนตีตี้ (Entity).....	11
2.2.1.4 โนเตชัน (Notation).....	12
2.2.2 ตัวอย่างเอกสาร DTD.....	13
2.3 ภาษา XSL (eXtensible Stylesheet Language).....	14
2.3.1 โครงสร้างเอกสาร XSL.....	15
2.3.2 ตัวอย่างการใช้งานเอกสาร XSL.....	18
บทที่ 3 วิธีดำเนินการวิจัย.....	20
3.1 ขั้นตอนการศึกษาโครงสร้างภาษา HTML และ WML.....	20
3.2 ขั้นตอนการสร้างเอกสาร XSL สำหรับใช้ในการแปลงเอกสาร.....	31
3.3 ขั้นตอนการออกแบบและสร้างตัวแปลงภาษารูปแบบ XSL.....	34
บทที่ 4 ผลจากการดำเนินงานของตัวแปลงภาษารูปแบบ XSL.....	44
4.1 ผลจากการดำเนินการแปลงเอกสาร HTML ให้เป็นเอกสาร WML.....	44
4.2 ผลจากการดำเนินการแปลงเอกสาร WML ให้เป็นเอกสาร HTML.....	45
บทที่ 5 การอภิปรายผลการดำเนินงานของตัวแปลงภาษารูปแบบ XSL.....	47
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ.....	48
6.1 สรุปผลการวิจัย.....	48
6.2 ข้อเสนอแนะ.....	48
ภาคผนวก.....	49
ภาคผนวก ก. Document Type Definition ของภาษา XSL ในส่วนของ XSLT.....	50
ภาคผนวก ข. Document Type Definition ของภาษา WML.....	58
ภาคผนวก ค. คู่มือการใช้โปรแกรมตัวแปลงภาษารูปแบบ XSL.....	68
บรรณานุกรม.....	80

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงค่าตัวอักษร Unicode ในรูปแบบเลขฐานสิบหก.....	4
2.2 แสดงค่าตัวอักษร 4 ตัว ที่ใช้เป็นพื้นที่ว่างในรูปแบบเลขฐานสิบหก.....	7
2.3 แสดงตัวบ่งปริมาณและความหมายของตัวบ่งปริมาณ.....	10
2.4 แสดงค่าที่เป็นไปได้ของค่าโดยปริยายสำหรับใช้ในการแจ้งแอตทริบิวต์.....	11
3.1 แสดงการเปรียบเทียบแท็กและแอตทริบิวต์ระหว่าง WML และ HTML.....	20



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่	หน้า
2.1 แสดงโครงสร้างของเอกสาร XML.....	8
2.2 แสดงโครงสร้างทางตรรกะของเอกสาร XML ที่นิยามตามตัวอย่างที่ 2.4.....	10
2.3 แสดงผลจากการเรียกดูเอกสาร XML.....	14
2.4 แสดงผลจากการเรียกดูเอกสาร XML ที่กำหนดรูปแบบการแสดงผลแล้ว.....	15
2.5 แสดงโครงสร้างของเอกสาร XSL.....	15
2.6 แสดงผลจากการเรียกดูเอกสาร XML จากตัวอย่าง	19
2.7 แสดงผลจากการเรียกดูเอกสาร XML ซึ่งถูกกำหนดรูปแบบการแสดงผล.....	19
3.1 แสดงผลจากการเรียกดูเอกสาร HTML ที่ใช้ทดสอบการทำงานของเอกสาร XSL.....	32
3.2 แสดงเอกสาร WML ที่ได้จากการกำหนดของเอกสาร XSL และผลการเรียกดูเอกสาร.....	33
3.3 แสดงการรับเอกสารอินพุตและเอาต์พุตของตัวแปลงภาษารูปแบบ XSL.....	34
3.4 แผนภาพ Flow Chart แสดงขั้นตอนการแปลงเอกสาร	34
3.5 แผนภาพ Flow Chart แสดงขั้นตอนการการทำงานของฟังก์ชันย่อยอ่านเอกสารอินพุต.....	35
3.6 แผนภาพ Flow Chart แสดงขั้นตอนการการทำงานของฟังก์ชันย่อยแปลงเอกสารอินพุตให้เป็นเอกสารเอาต์พุต.....	36
3.7 แผนภาพ Flow Chart แสดงขั้นตอนการการทำงานของฟังก์ชันย่อยแปลงเอกสาร HTML ให้เป็นเอกสาร WML.....	37
3.8 แผนภาพ Flow Chart แสดงขั้นตอนการการทำงานของฟังก์ชันย่อยแปลงเอกสาร WML ให้เป็นเอกสาร HTML.....	38
3.9 แผนภาพ Flow Chart แสดงขั้นตอนการการทำงานของฟังก์ชันย่อยแปลงเอกสารว่าง.....	39
3.10 แผนภาพ Flow Chart แสดงขั้นตอนการแปลงเอกสารอินพุตให้เป็นเอกสารเอาต์พุต.....	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันอุปกรณ์ที่ใช้เข้าถึงข้อมูลบนระบบเครือข่ายอินเทอร์เน็ตมีรูปแบบหลากหลาย ไม่จำกัดอยู่ที่คอมพิวเตอร์เพียงรูปแบบเดียว ด้วยความต่างกันในด้านต่าง ๆ ของอุปกรณ์เหล่านั้น อันเนื่องมาจากเทคโนโลยีที่ต่างกัน ทำให้เกิดภาษามาตรฐานสำหรับกำหนดรูปแบบการแสดงผล ข้อมูลขึ้นมากมาย

จากพัฒนาการดังกล่าว ทำให้เกิดความจำเป็นในการสร้างเอกสารหลาย ๆ รูปแบบเพื่อตอบสนองความต้องการในข้อมูลชุดเดียวกัน ดังนั้นเพื่อความลดความสับสนที่ในการจัดเก็บเอกสารและลดเวลาในการจัดสร้างเอกสารจึงจำเป็นต้องมีตัวแปลงเอกสาร เพื่อแปลงเอกสารชุดที่ถูกร้องขอให้เป็นชุดข้อมูลที่เหมาะสมกับการแสดงผลสำหรับอุปกรณ์นั้น ๆ

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

- 1.2.1 เพื่อศึกษาโครงสร้างของรูปแบบเอกสารที่ใช้ในการให้บริการบนเครือข่าย
- 1.2.2 เพื่อสร้างตัวแปลงเอกสารโดยใช้ Style Sheet เป็นหลักในการสร้าง
- 1.2.3 เพื่อนำตัวแปลงเอกสารที่สร้างขึ้นไปประยุกต์ใช้ในระบบงานจริง

1.3 ทฤษฎีหรือแนวคิดที่ใช้ในการศึกษา

ภาษาที่ใช้จัดสร้างเอกสารอิเล็กทรอนิกส์สำหรับให้บริการบนเครือข่ายอินเทอร์เน็ตส่วนใหญ่มีพื้นฐานมาจากภาษา XML แต่อาจจะมีโครงสร้างเอกสารแตกต่างกันไป ตัวอย่างเช่น ภาษา WML, ภาษา XHTML เป็นต้น ด้วยความสามารถในการปรับเปลี่ยนโครงสร้างเอกสาร XML ของภาษา XSL จึงเป็นภาษาที่เหมาะสมสำหรับสร้างตัวแปลงเอกสาร

1.4 ขอบเขตของการศึกษา

- 1.4.1 จัดสร้างตัวแปลงภาษารูปแบบ XSL เพื่อทำการแปลงเอกสาร WML ให้แสดงผลแบบเอกสาร HTML และ ทำการแปลงเอกสาร HTML ให้แสดงผลเป็นเอกสารแบบ WML
- 1.4.2 ใช้เอกสาร XSL เพื่อกำหนดรูปแบบแสดงผล
- 1.4.3 ใช้ภาษา JAVA เป็นภาษาในการสร้างตัวแปลงเอกสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ขั้นตอนการศึกษา

- 1.5.1 ศึกษาโครงสร้างของเอกสารรูปแบบต่าง ๆ ที่มีให้บริการบนเครือข่ายอินเทอร์เน็ต รวมทั้งโครงสร้างเอกสาร XSL ซึ่งเป็นตัวกำหนดรูปแบบการแสดงผล
- 1.5.2 ศึกษาการทำงานของภาษา JAVA ในส่วนที่เกี่ยวข้องกับการทำงานบนระบบเครือข่ายอินเทอร์เน็ต
- 1.5.3 สร้างตัวแปลงเอกสารระหว่างเอกสาร WML กับเอกสาร HTML โดยใช้ภาษา JAVA
- 1.5.4 ทดสอบการทำงานของตัวแปลงเอกสารที่ได้จากการดำเนินการในข้อ 1.5.3 พร้อมทั้งแก้ไขข้อผิดพลาด
- 1.5.5 รวบรวมข้อมูลในการทำปัญหาพิเศษทั้งหมด มาดำเนินการจัดทำเป็นเอกสาร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 ภาษา XML (eXtensible Markup Language)

XML เกิดขึ้นในปี 1996 โดย W3C¹ เป็นภาษา Markup² ที่รวมเอาความยืดหยุ่นและความมีประสิทธิภาพของภาษา SGML³ เข้ากับการเป็นที่ยอมรับอย่างกว้างขวางของภาษา HTML

ภาษา XML สร้างขึ้นจากพื้นฐานของภาษา SGML เพื่อให้อธิบายโครงสร้างของเอกสาร โดยมีการนำเสนอเป็นส่วนแยกอิสระออกมา ทำให้สามารถใช้ข้อมูลชุดเดิมซ้ำได้หลาย ๆ ครั้งและแสดงออกมาในรูปแบบที่แตกต่างกันได้ แล้วแต่การกำหนดรูปแบบการนำเสนอหรือรูปแบบการแสดงผล

2.1.1 โครงสร้างประโยคพื้นฐานในภาษา XML

โครงสร้างของภาษา XML มีลักษณะคล้ายกับภาษา HTML กล่าวคือ จะใช้เครื่องหมาย "<" และ ">" เพื่อกำหนดแท็ก (tag) โดยเริ่มต้นด้วย "<" และจบด้วยเครื่องหมาย ">" นอกจากนี้ ภาษา XML ยังมีคุณสมบัติ Case-sensitive ที่จะพิจารณาว่า คำเดียวกันแต่เขียนตัวเล็กตัวใหญ่ ต่างกัน ถือว่าไม่ใช่คำเดียวกัน เช่น

Book ≠ BOOK ≠ book ≠ book

คุณสมบัติ Case-Sensitive มีเป้าหมายเพื่อสามารถใช้ภาษาได้อย่างกว้างขวาง และทำให้กระบวนการเป็นไปโดยง่าย

2.1.1.1 ตัวอักษร (Characters)

เนื่องจากภาษา XML มีจุดมุ่งหมายเพื่อใช้ได้ในการขยายที่กว้างขวางขึ้น ตัวอักษรจึงไม่ได้จำกัดอยู่แค่เพียง 7-bit ASCII เท่านั้น ภาษา XML กำหนดใช้ตัวอักษรแบบ 16-bit+UNICODE

¹ W3C (World Wide Web Consortium) คือ สมาคมที่ก่อตั้งขึ้นเพื่อพัฒนาโปรโตคอลและวิธีการสื่อสารข้อมูลให้เป็นมาตรฐานกลาง เพื่อเปิดโอกาสให้ผู้พัฒนาซอฟต์แวร์ใช้อ้างอิง

² ภาษา Markup หรือ Markup language เป็นภาษาที่ใช้แท็กหรือลำดับของตัวอักษรเพื่อกำหนดขอบเขตและอธิบายข้อมูล

³ SGML(Standard Generalized Markup Language) เป็นภาษามาตรฐานหนึ่งของ ISO / IEC8879 ใช้อธิบายเอกสารที่มีโครงสร้างชัดเจน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 ซึ่งถือเป็นมาตรฐานใหม่ ถึงแม้ตัวอักษรส่วนใหญ่ จะไม่ได้เป็น Unicode แต่ Unicode ก็ถูกออกแบบมาให้สามารถทำงานได้กับตัวอักษรที่มีการเข้ารหัสทุกชนิดที่มีอยู่ปัจจุบัน ในการแปลจาก ASCII เป็น Unicode จะทำการแปลงรหัสตัวอักษรจาก 7 - bit เป็น 16 - bit โดยที่ยังคงเก็บค่าของตัวอักษร 7 - bit อยู่ ตัวอักษรในภาษา XML ที่ถูกต้องจะรวมถึง ASCII C0 ที่ใช้ควบคุมตัวอักษรแบบ ASCII ธรรมดาและตัวอักษร Unicode

ตารางที่ 2.1 แสดงค่าตัวอักษร Unicode ในรูปแบบเลขฐานสิบหก

Character Value (hexadecimal)	Description
09	Horizontal tab(HT)
0A	Line-feed (LF)
0D	Carriage-Return (CR)
20..7E	ASCII display characters
80..D7FF	Unicode Characters
E000..F8FF	"Private use area"
F900..FFFD	CJK (Chinese-Japanese-Korean) Compatibility

Unicode ได้เพิ่มตัวอักษรขึ้นมากกว่า 137,000 ตัว ที่เรียกว่า "Private Use Area" สำหรับ application ที่ใช้ตัวอักษรเฉพาะ การแลกเปลี่ยนข้อมูลที่จะใช้ตัวอักษรพิเศษเหล่านี้ ต้องมีการตกลงกันในความหมายให้ตรงกันก่อนทำการแลกเปลี่ยนข้อมูล

2.1.1.2 ชื่อ (Names)

โครงสร้างที่ใช้ในภาษา XML ส่วนใหญ่มักจะถูกตั้งชื่อ โดยชื่อในภาษา XML ต้องขึ้นต้นด้วย ตัวอักษร, เครื่องหมายขีดล่าง (underscore, _), หรือเครื่องหมายโคลอน (:) และต่อด้วยตัวอักษรที่ถูกต้องประกอบไปด้วย ตัวเลข (digit), ตัวอักษร, เครื่องหมายยัติภังค์ (hyphen, -), หรือจุด (.) ในกรณีของ เครื่องหมายโคลอน(:) จะใช้เป็นตัวขึ้นระหว่างช่องว่างของชื่อเฉพาะเท่านั้น เนื่องจากตัวอักษรไม่ถูกจำกัดอยู่เพียงตัวอักษร ASCII ดังนั้นผู้ใช้สามารถใช้ภาษาของตนเองเพื่อทำงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Book
 BOOK
 Wrox:Book_Catalog
 สมาคมยุโรปเพื่อเอเชีย(CERN)

ตัวอย่างที่ 2.1 แสดงชื่อที่ถูกต้องในภาษา XML

จากตัวอย่างที่ 2.1 สองบรรทัดแรกชื่อทั้งสองถือว่าเป็นคนละตัวกัน เพราะภาษา XML มีลักษณะเป็น Case-sensitive ส่วนบรรทัดที่สามเป็นการแสดงถึงการใช้เครื่องหมายโคลอน (:) เพื่อแบ่งคั่นช่องว่างตัวอักษร

-Book
 42book
 AmountIn\$
 $E=mc^2$
 XmlData
 XML_on_NeXt_machines

ตัวอย่างที่ 2.2 แสดงชื่อที่ไม่ถูกต้องในภาษา XML

จากตัวอย่างที่ 2.2 สองบรรทัดแรก ใช้ตัวเริ่มต้นที่ผิด ถึงแม้ว่าจะใช้ตัวอักษรที่ถูกต้องตามกฎคือ "-" และ "4" แต่ไม่มีการอนุญาตให้ใช้ "-" และตัวเลข ขึ้นต้นชื่อ บรรทัดที่สามและสี่ผิด เพราะไม่มีการอนุญาตให้ใช้ "\$" หรือตัวยกในการตั้งชื่อ ส่วนสองบรรทัดสุดท้ายผิดเพราะคำว่า "xml" เป็นคำสงวน ตามกฎการสร้าง name ใน XML

2.1.1.3 อีลิเมนต์ (Element)

อีลิเมนต์เป็นขอบเขตพิเศษในภาษา XML ซึ่งอาจจะประกอบด้วยอีลิเมนต์อื่น ๆ, character data, character references, entity reference, PIs, comments และหรือ CDATA ซึ่งทั้งหมดนี้ถือเป็นรายละเอียดของอีลิเมนต์ ซึ่งข้อมูลในภาษา XML ทั้งหมด ต้องถูกบรรจุอยู่ในอีลิเมนต์ ยกเว้น comments, PIs และ White space

อีลิเมนต์จะแบ่งโดยใช้แท็กที่ประกอบไปด้วยชื่อของอีลิเมนต์และปิดด้วยเครื่องหมาย "<>" ทุก ๆ อีลิเมนต์ต้องเริ่มต้นด้วยแท็กเริ่มต้นและปิดด้วยแท็กปิดท้าย อาจมีข้อยกเว้นสำหรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กฎข้อนี้ อย่างหนึ่งคือการใช้อีลิเมนต์ว่าง ซึ่งก็คือ อีลิเมนต์ที่เริ่มแท็กเริ่มต้น แล้วต่อด้วยแท็กปิดท้ายเลย

- แท็กเริ่มต้น (Star-Tags) ประกอบไปด้วย เครื่องหมาย "<" ตามด้วยชื่ออีลิเมนต์ แล้วปิดด้วย ">"
- แท็กปิดท้าย (End-Tags) ประกอบด้วย เครื่องหมาย "<" ตามด้วย "/" ชื่ออีลิเมนต์ และปิดด้วย ">" ทุก ๆ แท็กปิดท้ายต้องมีชื่ออีลิเมนต์ที่ตรงกันกับแท็กเริ่มต้น เช่น
<some_tag> content goes here </some_tag>
- แท็กอีลิเมนต์ว่าง (Empty-Element Tags) จะเริ่มต้นด้วยแท็กเริ่มต้นแล้วต่อด้วยแท็กปิดท้ายเลย จะมีลักษณะดังนี้ <point1></point1> โดยสามารถใช้ย่อได้เป็น <point1/>

2.1.1.4 แอตทริบิวต์ (Attributes)

ถ้าเปรียบอีลิเมนต์เป็นคำนามในภาษา XML แล้ว แอตทริบิวต์ก็สามารถเปรียบได้เป็นคำคุณศัพท์ที่บ่งบอกถึงคุณสมบัติของคำนามนั่นเอง แอตทริบิวต์จะถูกใช้เมื่อต้องการให้ข้อมูลต่าง ๆ เกี่ยวกับอีลิเมนต์นั้น ๆ โดยจะประกอบไปด้วย ชื่อและค่าของแอตทริบิวต์ ซึ่งจะสามารถใช้ใน 2 รูปแบบ คือ

ชื่อแอตทริบิวต์="ค่าแอตทริบิวต์"

ชื่อแอตทริบิวต์='ค่าแอตทริบิวต์'

ค่าของแอตทริบิวต์ อาจประกอบไปด้วย entity references, character references และ/หรือตัวอักษร จะไม่สามารถมีแอตทริบิวต์ที่มีชื่อเหมือนกันอยู่ในแท็กเริ่มต้นหรือแท็กอีลิเมนต์ว่างเดียวกัน

```
<carton number="0-666-42-1">
```

```
...
```

```
</carton>
```

```
<carton number="0-666-42-2">
```

```
...
```

```
</carton>
```

```
<EOF char="1A"/>
```

ตัวอย่างที่ 2.3 แสดงการใช้แอตทริบิวต์ที่ถูกต้อง

จากตัวอย่างที่ 2.3 ชื่อของแอตทริบิวต์ คือ "number" มีค่าเป็น 0-666-42-1 และ 0-666-42-2 ตามลำดับ เช่นเดียวกับในอิลีเมนต์ว่างชื่อ "EOF" ที่มีแอตทริบิวต์ชื่อ "char" ซึ่งมีค่าเป็น 1A

2.1.1.5 พื้นที่ว่าง (White Space)

คืออักขระที่ถูกใช้โดยจะไม่มีข้อกำหนดความหมาย แต่มีประโยชน์สำหรับผู้ใช้ในแง่ของการเขียนโปรแกรม มี 4 ตัวอักษรที่ใช้เป็นพื้นที่ว่างในภาษา XML

ตารางที่ 2.2 แสดงค่าตัวอักษร 4 ตัวที่ใช้เป็นพื้นที่ว่างในรูปแบบเลขฐานสิบหก

Character Values (in hexadecimal)	Description
09	Horizontal tab(HT)
0A	Line-feed(LF)
0D	Carriage-return(CR)
20	ASCII space character

2.1.1.6 หมายเหตุ (Comment)

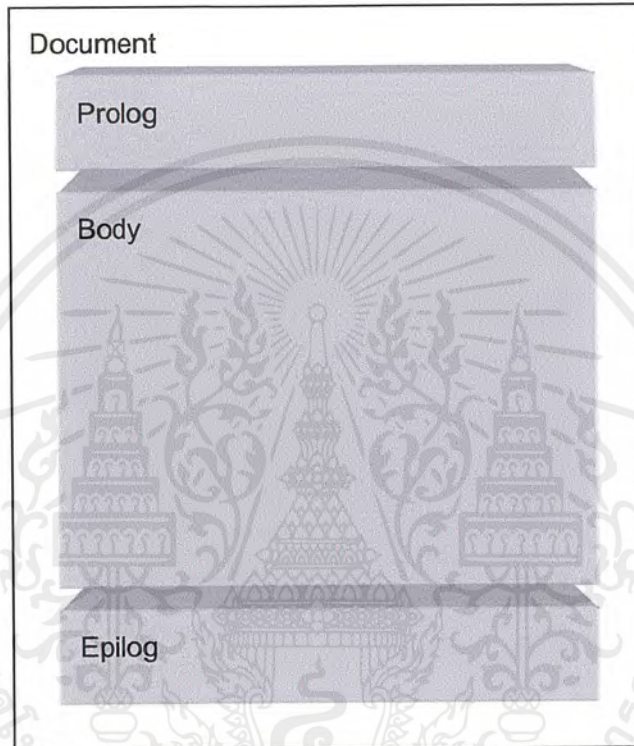
ในการสร้างหมายเหตุในเอกสาร จะแสดงถึงข้อความหมายเหตุพิเศษ เพื่อให้ผู้สร้างสามารถทำความเข้าใจ และง่ายต่อการสืบค้นภายหลัง จะปรากฏในตัวเอกสารหรือไม่ก็ได้ ซึ่งจะไม่ได้รวมเป็นส่วนหนึ่งของเอกสารจริง และสามารถใส่ลงที่ส่วนใดของเอกสารก็ได้ โดยข้อความหมายเหตุต้องอยู่ภายใต้เครื่องหมาย <!-- และ -->

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 โครงสร้างของเอกสาร XML

เอกสาร XML ที่มีรูปแบบถูกต้อง ประกอบด้วย 3 ส่วน ดังนี้

1. Prolog
2. Body
3. Epilog



รูปที่ 2.1 แสดงโครงสร้างของเอกสาร XML

1. Prolog มีหรือไม่มีก็ได้ เป็นส่วนเริ่มของเอกสาร XML ที่ใช้แสดงว่าเป็นข้อมูล XML จะอธิบายถึงวิธีการเข้ารหัสตัวอักษรของเอกสารนั้น และแสดงการกำหนดค่าต่างๆที่จำเป็นต้องใช้ในการตรวจสอบเอกสารก่อนการส่งต่อไปให้กับ Application โดยจะประกอบไปด้วย ส่วนประกาศ XML แล้วตามด้วย comment, PIs และ white space
2. Body ซึ่งจะประกอบไปด้วยอีลิเมนต์อย่างน้อยหนึ่งอีลิเมนต์ และอาจจะมี character data ด้วย
3. Epilog มีหรือไม่มีก็ได้ ประกอบด้วย comment, processing instructions (PI), และ/หรือ white space ที่ตามท้ายอีลิเมนต์ที่รี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 ตัวอย่างเอกสาร XML

```
<? xml version="1.0" encoding="utf-8" standalone="yes"?>
<! - - ===== The Wrox Press Book Catalog Application ===== - - >
<Book>
  <Title>Professional XML</Title>
  <Author>
    Mark Birbeck, Steven Livingstone, Didier Matin, Stephen Mohr, Nicola Ozu, et al.
  </Author>
  <Publisher>Wrox Press, Ltd.</Publisher>
  <PubDate>November 1999</PubDate>
  <Abstract>XML 0-500kmh in 3 seconds</Abstract>
  <Pages>750</Pages>
  <ISBN>1-861003-11-0</ISBN>
  <RecSubjCategories>
    <Category>Internet</Category>
    <Category>Web Publisher</Category>
    <Category>XML</Category>
  </RecSubjCategories>
</Book>
```

2.2 ไวยากรณ์ DTD (Document Type Definition)

DTD เป็นไวยากรณ์ประเภทไม่ขึ้นกับเนื้อหา ใช้เพื่อแจ้งรายการโครงสร้างทางตรรกะที่ใช้ในเอกสาร XML

2.2.1 โครงสร้างเอกสาร DTD

เอกสาร DTD ประกอบด้วย อีลิเมนต์, แอตทริบิวต์, เอนตีตี้, โนเตชัน โดยภายในเอกสารหนึ่ง ๆ จะประกอบด้วยองค์ประกอบทั้ง 4 อย่างหรือน้อยกว่า เรียงต่อกัน

2.2.1.1 อีลิเมนต์ (Element)

ใช้นิยามองค์ประกอบของเอกสารและความสัมพันธ์ระหว่างองค์ประกอบเหล่านั้น การแจ้งอีลิเมนต์ ทำโดยใช้คำสั่ง <! ELEMENT (ชื่อ element,แบบจำลองเนื้อหา) ตัวบ่งปริมาณ> (ตัวบ่งปริมาณและความหมายของตัวบ่งปริมาณ ดูเพิ่มเติมได้ในตารางที่ 2.3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 แสดงตัวบ่งปริมาณและความหมายของตัวบ่งปริมาณ

ตัวบ่งปริมาณ	ชื่อ	ความหมาย
?	Question Mark	มีหรือไม่มีก็ได้(ถ้ามีมีเพียงหนึ่ง)
*	Asterisk	มีหรือไม่มีก็ได้ ถ้ามีสามารถมีได้มากกว่าหนึ่ง
+	Plus Sign	มีได้หนึ่งหรือมากกว่าหนึ่ง

<?xml encoding="UCS-2"?>

<!ELEMENT chapter (title, (paragraph | section))>

<!ELEMENT title (#PCDATA)>

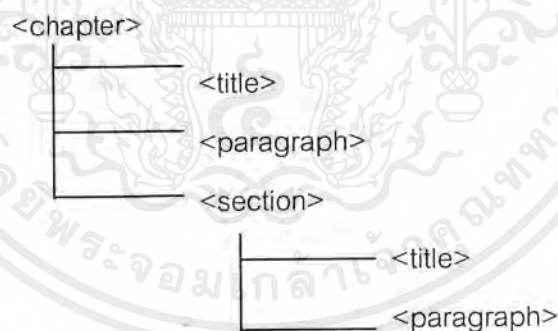
<!ELEMENT paragraph (#PCDATA)>

<!ELEMENT section (title,paragraph)>

ตัวอย่างที่ 2.4 แสดงการแจ้งอิลิเมนต์

เอกสาร XML ที่มีการนิยามตามตัวอย่างที่ 2.4 จะมีโครงสร้างทางตรรกะ เป็นดังรูปที่

2.2



รูปที่ 2.2 แสดงโครงสร้างทางตรรกะของเอกสาร XML ที่นิยามตามตัวอย่างที่ 2.4

2.2.1.2 แอตทริบิวต์ (Attribute)

ใช้นิยามแอตทริบิวต์ต่าง ๆ ของอิลิเมนต์ประกอบด้วยชื่อแอตทริบิวต์, ประเภทแอตทริบิวต์ และค่าโดยปริยาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<!ATTLIST chapter
  author CDATA #IMPLIED
>
```

ตัวอย่างที่ 2.5 แสดงการแจ้งรายการแอตทริบิวต์

ตัวอย่างที่ 2.5 เป็นการแจ้งแอตทริบิวต์ชื่อ `author` ซึ่งเป็นแอตทริบิวต์ของอีลีเมนต์ชื่อ `chapter` แอตทริบิวต์นี้เป็นประเภท `CDATA` (ข้อมูลตัวอักษร) และค่าโดยปริยาย (ดูรายละเอียดเพิ่มเติมได้ในตารางที่ 2.4) คือ `#IMPLIED` (หมายถึง ไม่ได้มีการกำหนดค่าอะไรเป็นพิเศษ) ในกรณีที่อีลีเมนต์มีหลายแอตทริบิวต์ ก็ให้ทำเช่นเดียวกับแถวที่ 2 ของตัวอย่างที่ 2.5 คือแจ้งชื่อแอตทริบิวต์, ประเภท และค่าโดยปริยายของแต่ละแอตทริบิวต์เป็นแถว ๆ ไป แล้วปิดท้ายด้วยเครื่องหมาย `>`

ตารางที่ 2.4 แสดงค่าที่เป็นไปได้ของค่าโดยปริยายสำหรับใช้ในการแจ้งแอตทริบิวต์

ค่าโดยปริยาย	ความหมาย
<code>#REQUIRED</code>	แอตทริบิวต์ต้องถูกประกาศในเอกสาร
<code>#IMPLIED</code>	สามารถประกาศหรือไม่ก็ได้ ถ้าไม่ประกาศจะมีค่าเท่ากับค่าที่แอปพลิเคชันกำหนดไว้ให้
<code>"defaultValue"</code>	มีค่าที่กำหนดให้เป็นไปตามค่าภายใต้เครื่องหมาย " " ในกรณีที่ไม่ได้มีการประกาศค่าไว้
<code>#FIXED "fixedValue"</code>	กำหนดค่าแอตทริบิวต์นั้นมีค่าคงที่เท่ากับค่าภายใต้เครื่องหมาย " "

2.2.1.3 เอนตีตี้ (Entity)

ใช้เพื่อกำหนดชื่อให้แก่สารสนเทศ อันจะทำให้การอ้างอิงสารสนเทศในเอกสารเป็นไปโดยสะดวก โดยสามารถใช้ชื่อที่ให้นั้นในการอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<ENTITY image1
  SYSTEM "filename.gif"
  NDATA gif
>
```

ตัวอย่างที่ 2.6 แสดงการแจ้งเอนิตีที่อยู่ภายนอกเอกสาร

จากตัวอย่างที่ 2.6 “ENTITY” เป็นคำสั่งสำหรับการแจ้งเอนิตี โดยที่ image1 เป็นชื่อที่กำหนดให้แก่สารสนเทศซึ่งเป็นภาพที่อยู่ในแฟ้มชื่อ “filename.gif” “SYSTEM” เป็นคำสั่งที่ใช้บอกว่าสารสนเทศนั้นอยู่ในระบบภายนอกและไม่ได้เป็นส่วนหนึ่งของเอกสาร XML นี้ “NDATA” เป็นคำสั่งใช้บอกว่าข้อมูลเป็นแบบไบนารีและ gif เป็นชื่อประเภทข้อมูล หลังจากการประกาศตามตัวอย่างที่ 2.6 แล้ว เมื่อมีการอ้างอิงชื่อ image1 ในเอกสาร สารสนเทศที่ปรากฏขึ้นมาจริงจะเป็นแฟ้มกราฟฟิก filename.gif

สารสนเทศที่อ้างอิงอาจอยู่ในเอกสารเองก็ได้ ไม่จำเป็นต้องเป็นสิ่งที่อยู่นอกเอกสารเสมอไป

```
<ENTITY chap2 “Chapter 2: ทำไม XML จึงจำเป็น ? ”>
```

ตัวอย่างที่ 2.7 แสดงการแจ้งเอนิตีที่อยู่ภายในเอกสาร

2.2.1.4 โนเตชัน (Notation)

ใช้นิยามความเกี่ยวข้องระหว่างสัญลักษณ์ชื่อกับโปรแกรม หรือการประมวลผลภายนอกที่ไม่เป็นส่วนของการประมวลผล XML ซึ่งสามารถถูกเรียกมาใช้งานได้ โดยสัญลักษณ์ชื่อนั้นต้องถูกแจ้งไว้ก่อน

```
<!NOTATION gif SYSTEM “/programs/viewgif.exe”>
```

ตัวอย่างที่ 2.8 แสดงการแจ้งโนเตชัน

เป็นการอ้างอิงชื่อ gif กับโปรแกรมการดูภาพกราฟิกชื่อ viewgif.exe จากตัวอย่างที่ 2.7 เมื่อใช้คู่กับตัวอย่างที่ 2.8 จะทำให้ระบบประมวลผล XML เรียกโปรแกรม viewgif.exe มาแสดงภาพ image1 ได้อย่างถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 ตัวอย่างเอกสาร DTD

```

<!ELEMENT Author (FirstName , MI? , LastName , Biographical , Portrait ) >
<!ATTLIST Author authorCiteID ID #REQUIRED>
<!ELEMENT FirstName (#PCDATA) >
<!ELEMENT MI (#PCDATA) >
<!ELEMENT LastName (#PCDATA) >
<!ELEMENT Biographical (#PCDATA) >
<!ELEMENT Portrait EMPTY >
<!ATTLIST Portrait piclink CDATA #IMPLIED >

```

เอกสาร XML ที่ประกาศโครงสร้าง DTD ตามตัวอย่างข้างบน จะมีโครงสร้างทางตรรกะดังแผนภาพด้านล่าง



อิลิเมนต์ชื่อ "MI" สามารถมีหรือไม่ก็ได้ ถ้ามีสามารถมีได้เพียงหนึ่ง ในส่วนของแอตทริบิวต์ อิลิเมนต์ที่ชื่อ Author มีแอตทริบิวต์ชื่อ "authorCiteID" ประเภทแอตทริบิวต์เป็น ID (ข้อมูลที่เป็นหนึ่งเดียวในเอกสาร) มีค่าโดยปริยายเป็น #REQUIRED (หมายถึงต้องมีแอตทริบิวต์นี้ในอิลิเมนต์ที่มีการประกาศ) ส่วนอิลิเมนต์ชื่อ "Portrait" มีแอตทริบิวต์ชื่อ "piclink" ประเภทแอตทริบิวต์เป็น CDATA มีค่าโดยปริยายคือ #IMPLIED (หมายถึง สามารถมีแอตทริบิวต์ตัวนี้หรือไม่ก็ได้)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ภาษา XSL (eXtensible Stylesheet Language)

XSL เป็นภาษาที่มีพื้นฐานมาจากภาษา XML ถูกออกแบบขึ้นโดย W3C เพื่อให้ในการกำหนดรูปแบบการแสดงผลหรือรูปแบบการนำเสนอข้อมูลให้กับเอกสาร XML ประกอบด้วย 2 ส่วน คือ

- XSLT (XSL Transformation) คือ ส่วนที่ใช้สำหรับการแปลงรูปเอกสาร XML ให้เป็นเอกสารที่มีโครงสร้างแตกต่างออกไป
- XSLF (XSL Formatting object) คือ ส่วนที่ใช้กำหนดรูปแบบการแสดงผลให้กับเอกสาร XML เช่น การกำหนดขนาดของตัวอักษร, การวางหน้ากระดาษ เป็นต้น

การเรียกดูเอกสาร XML ที่ไม่ได้มีการกำหนดรูปแบบการแสดงผล ผ่านทางบราวเซอร์ ผลจากการเรียกดูเอกสารเป็นดังรูปที่ 2.3 ซึ่งแสดงรายละเอียดของเอกสาร XML ออกมาทั้งหมด

เพื่อให้บราวเซอร์แสดงแต่ข้อมูลที่อยู่ในเอกสารเมื่อมีการเรียกดู เราสามารถใช้ภาษา XSL กำหนดรูปแบบการแสดงผลให้กับเอกสาร XML ได้ โดยที่รูปแบบการแสดงผลสามารถอยู่ในรูปแบบต่าง ๆ มากมาย เช่นอยู่ในรูปตารางดังในรูปที่ 2.4 หรือ อยู่ในรูปเอกสารที่สื่อซึ่งใช้เข้าถึงข้อมูลต่าง ๆ สามารถเรียกดูเอกสารได้ ทั้งนี้ขึ้นอยู่กับวิธีการเขียนเอกสารที่ใช้กำหนดรูปแบบการแสดงผลของผู้ให้บริการข้อมูล

```
<?xml version="1.0" ?>
- <StudentDB>
  <DB_Name>Information of KMITL's Student.</DB_Name>
  <StudentInfo>
    <FirstName>Thana</FirstName>
    <LastName>Thongpipat</LastName>
    <Faculty>Science.</Faculty>
  </StudentInfo>
  <StudentInfo>
    <FirstName>Narayut</FirstName>
    <LastName>Jaidee</LastName>
    <Faculty>Engineering.</Faculty>
  </StudentInfo>
  <StudentInfo>
    <FirstName>Akekachai</FirstName>
    <LastName>Chaipijit</LastName>
    <Faculty>Science.</Faculty>
  </StudentInfo>
</StudentDB>
```

รูปที่ 2.3 แสดงผลจากการเรียกดูเอกสาร XML

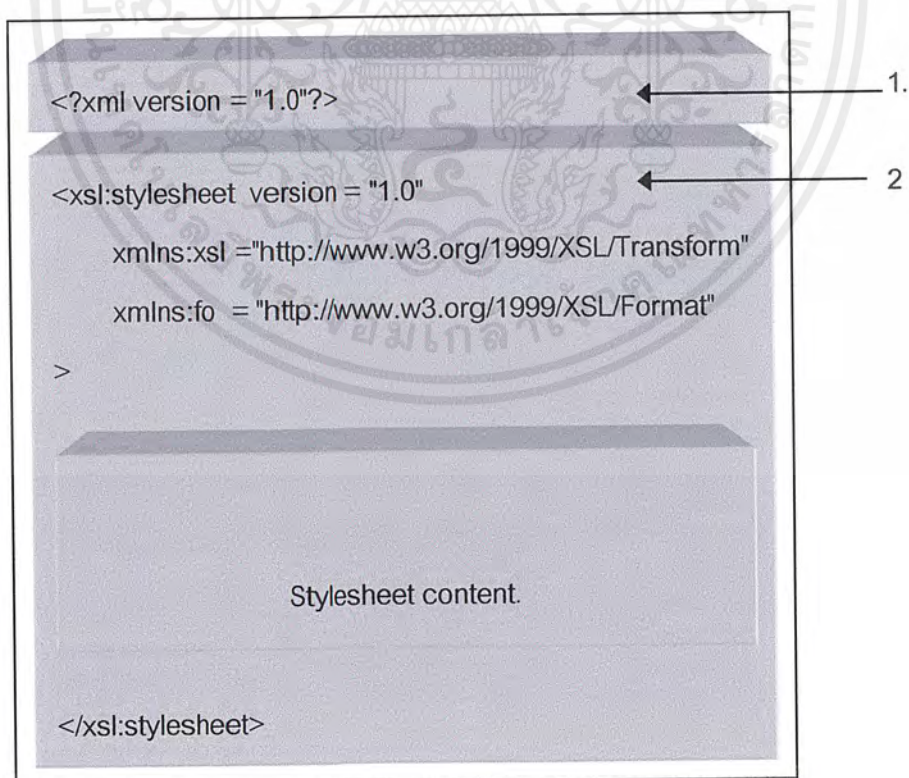
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FirstName	LastName	Faculty
Thana	Thongpipat	Science.
Narayut	Jaidee	Engineering.
Akekachai	Chaipijit	Science.

รูปที่ 2.4 แสดงผลจากการเรียกดูเอกสาร XML ที่กำหนดรูปแบบการแสดงผลแล้ว

2.3.1 โครงสร้างเอกสาร XSL

เนื่องจาก XSL เป็นภาษาที่มีพื้นฐานมาจากภาษา XML ดังนั้นโครงสร้างของเอกสาร XSL จึงมีโครงสร้างที่เหมือนกับเอกสาร XML ซึ่งมีส่วนประกอบหลัก 2 ส่วน ไม่พิจารณาในส่วนของ Epilog ดังแสดงในรูปที่ 2.5



รูปที่ 2.5 แสดงโครงสร้างของเอกสาร XSL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.5 จะขอล่าวถึงรายละเอียดในโครงสร้างแต่ละส่วนดังนี้

ส่วนที่ 1. คือ ส่วนที่ประกาศเพื่อบอกข้อมูลต่างๆ ของภาษา XML ที่เอกสารนี้ใช้ในการอ้างอิง

ส่วนที่ 2. คือ ส่วนที่ประกาศว่าเป็นเอกสารที่เขียนด้วยภาษา XSL บอกข้อมูลที่เกี่ยวข้องกับภาษา XSL ที่ใช้ในการเขียนเอกสาร ได้แก่ version, XML namespaces¹ ของส่วน XSLT และส่วน XSLF ตามลำดับ XML namespaces ในส่วนของ XSLF นั้นให้ประกาศเมื่อต้องการใช้แท็กคำสั่งในส่วนของ XSLF เท่านั้น

ส่วน Stylesheet content. เป็นรายละเอียดในการกำหนดรูปแบบการแสดงผลรายละเอียดของ Element และความสัมพันธ์ระหว่าง Element สามารถดูได้จาก DTD ของเอกสาร XSL ในภาคผนวก ก.



¹ ใช้ในภาษา XML เพื่อแก้ปัญหาการกรอกรใช้แท็กที่มีชื่อเดียวกัน ดูรายละเอียดเพิ่มเติมที่

<http://www.w3.org/TR/1999/REC-xml-names-19990114>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เกิดความเข้าใจมากขึ้น ขอยกตัวอย่างพร้อมคำอธิบายในแต่ละส่วนดังนี้

<?xml version = "1.0"?>	----- 1.
<xsl:stylesheet xmlns:xsl = "http://www.w3.org/1999/XSL/Transform">	----- 2.
<xsl:output method = "html" doctype-public = "-//W3C//DTD HTML 4.0 Transitional//EN"/>	----- 3.
<xsl:template match="/">	----- 4.
<html>	----- 5.
<xsl:apply-templates />	----- 6.
</html>	----- 7.
</xsl:template>	----- 8.
<xsl:template match = "body" >	----- 9.
<body>	----- 10.
<xsl:apply-templates />	----- 11.
</body>	----- 12.
</xsl:template>	----- 13.
<xsl:template match = "b" >	----- 14.
	----- 15.
<xsl:apply-templates />	----- 16.
	----- 17.
</xsl:template>	----- 18.
</xsl:stylesheet>	----- 19.

ตัวอย่างที่ 2.9 แสดงตัวอย่างเอกสาร XSL

จากตัวอย่างที่ 2.9 บรรทัดที่ 1, 2 เป็นการประกาศในส่วนที่ 1. และ 2. ตามลำดับ บรรทัดที่ 3. เป็นแท็กคำสั่ง <xsl:output> ซึ่งเป็นแท็กคำสั่งในภาษา XSL เพื่อใช้ระบุว่า เอกสารที่ได้จากการกำหนดรูปแบบการแสดงผลนั้น เป็นเอกสารที่มีโครงสร้างภาษาเป็นอย่างไร จากตัวอย่างนี้ ระบุว่าเอกสารที่ได้เป็นเอกสารที่มีโครงสร้างภาษาเป็นภาษา HTML บรรทัดที่ 4. เป็นแท็กคำสั่ง <xsl:template match = "/"> ที่กำหนดให้ทำตามคำสั่งที่อยู่ระหว่าง <xsl:template match = "/"> กับ </xsl:template> เมื่อพบแท็ก ซึ่งอยู่ภายใต้เครื่องหมาย " " ในเอกสารที่เขียนขึ้นด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษา XML ในตัวอย่างนี้กำหนดให้ทำตามคำสั่งเมื่อเริ่มทำการกำหนดรูปแบบการแสดงผล ส่วนในบรรทัดที่ 6. เป็นแท็กคำสั่ง `<xsl:apply-templates />` ที่กำหนดให้ทำการค้นหาแท็กต่อไป เพื่อทำการจับคู่กับแท็กที่กำหนดไว้ในเอกสารที่เขียนขึ้นด้วยภาษา XSL

2.3.2 ตัวอย่างการใช้งานเอกสาร XSL

กำหนดเอกสาร XML ให้มีรายละเอียดดังนี้

```
<?xml version = "1.0" ?>
```

```
<Document>
```

```
  <body> <b>XSL</b> is a language for expressing stylesheet.</body>
```

```
</Document>
```

กำหนดรูปแบบการแสดงผลให้เป็นเอกสาร HTML ข้อความที่อยู่ระหว่าง `` กับ `` เป็นตัวหนา สามารถเขียนเอกสารเพื่อกำหนดรูปแบบการแสดงผลได้เป็น

```
<?xml version = "1.0" ?>
```

```
<xsl:stylesheet xmlns:xsl = "http://www.w3.org/1999/XSL/Transform">
```

```
<xsl:output method="html"
```

```
  doctype-public="-//W3C//DTD HTML 4.0 Transitional//EN"
```

```
/>
```

```
<xsl:template match="/">
```

```
  <html> <xsl:apply-templates /></html>
```

```
</xsl:template>
```

```
<xsl:template match = "body" >
```

```
  <body><xsl:apply-templates /></body>
```

```
</xsl:template>
```

```
<xsl:template match = "b" >
```

```
  <b> <xsl:apply-templates /></b>
```

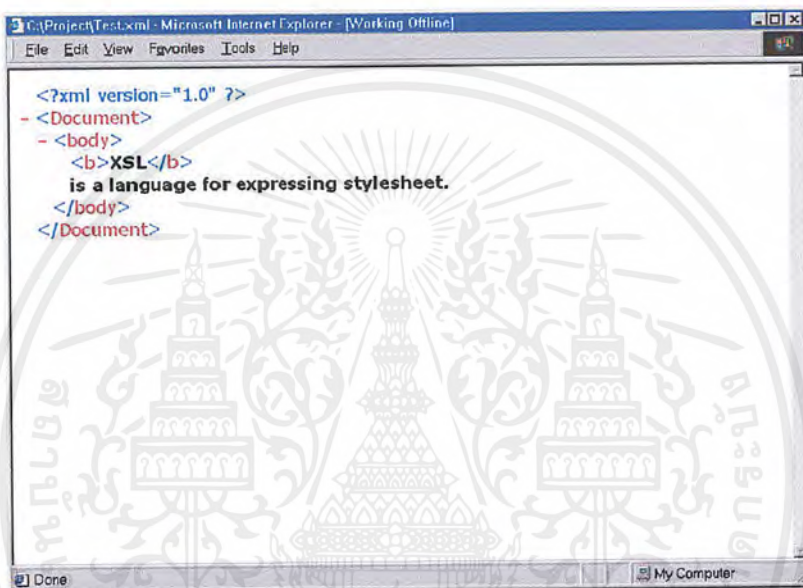
```
</xsl:template>
```

```
</xsl:stylesheet>
```

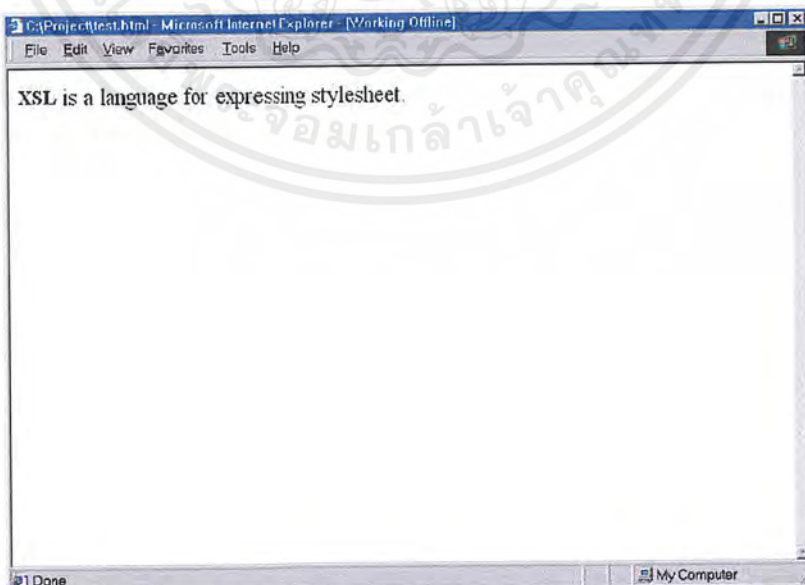
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลที่ได้จากการใช้งานเอกสาร XSL เป็นดังนี้

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
  <body> <b>XSL</b> is a language for expressing stylesheet.
</body>
</html>
```



รูปที่ 2.6 แสดงผลจากการเรียกดูเอกสาร XML จากตัวอย่าง



รูปที่ 2.7 แสดงผลจากการเรียกดูเอกสาร XML ซึ่งถูกกำหนดรูปแบบการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีดำเนินการวิจัย

วิธีดำเนินการวิจัยของปัญหาพิเศษเรื่องตัวแปลงภาษารูปแบบ XSL มีขั้นตอนการดำเนินการซึ่งสามารถแบ่งออกเป็น 3 ส่วน ดังนี้

3.1 ขั้นตอนการศึกษาโครงสร้างภาษา HTML และภาษา WML เพื่อหาความสัมพันธ์ระหว่าง 2 ภาษา ว่ามีแท็กใดบ้างที่เหมือนกัน เพื่อใช้เป็นหลักในการสร้างเอกสาร XSL สำหรับใช้ในการแปลงเอกสาร

3.2 ขั้นตอนการสร้างเอกสาร XSL สำหรับใช้ในการแปลงเอกสาร HTML ให้เป็นเอกสาร WML และเอกสาร XSL สำหรับแปลงเอกสาร WML ให้เป็นเอกสาร HTML เพื่อใช้สำหรับเป็นอินพุตให้กับตัวแปลงภาษารูปแบบ XSL สำหรับการแปลงเอกสารให้ได้ตามความต้องการ

3.3 ขั้นตอนการออกแบบสร้างตัวแปลงภาษารูปแบบ XSL

3.1 ขั้นตอนการศึกษาโครงสร้างภาษา HTML และ WML

จากการศึกษาโครงสร้างของภาษา HTML และ WML พบว่าทั้งสองภาษามีการใช้แท็กที่ใกล้เคียงกัน แต่แอตทริบิวต์ที่ใช้ภายในแท็กบางแท็กไม่เหมือนกัน ดังแสดงในตารางที่ 3.1 ในคอลัมน์ TAG จะแสดงแท็กที่มีใน WML ทั้งหมด คอลัมน์ WML จะแสดงแท็กและแอตทริบิวต์ในภาษา WML และคอลัมน์ HTML แสดงแท็กและแอตทริบิวต์ในภาษา HTML ที่มีความใกล้เคียงกับภาษา WML

ตารางที่ 3.1 แสดงการเปรียบเทียบแท็ก และแอตทริบิวต์ระหว่าง WML และ HTML

TAG	WML	HTML
1. wml	<wml> </wml> xml:lang = nmtoken	<html> </html>
2. card	<card> </card> title = vdata newcontext = boolean ordered = boolean onenterforward = href onenterbackward = href ontimer = href	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นำไปเผยแพร่ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TAG	WML	HTML
	xml:lang id class	
3. template	<template> </template> onenterforward = href onenterbackward = href ontimer = href id class	
4. head	<head> </head>	<head> </head>
5. access	<access> domain = cdata path = cdata id class	
6. meta	<meta/> content = cdata name = cdata http-equiv = cdata forua = boolean scheme = cdata id class	<meta> content = cdata name = name http-equiv = name scheme = cdata
7. do	<do> </do> type = cdata label = vdata name = nmtoken optional = boolean xml:lang id , class	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TAG	WML	HTML
8. onevent	<pre><onevent> </onevent> type = cdata id class</pre>	
9. postfield	<pre><postfield/> name = vdata value = vdata Id class</pre>	
10. go	<pre><go> </go> href = href sendreferer = boolean method = (post get) accept-charset = cdata class</pre>	
11. prev	<pre><prev/> id class</pre>	
12. refresh	<pre><refresh> </refresh> id class</pre>	
13. noop	<pre><noop/> id class</pre>	
14. setvar	<pre><setvar/> name = vdata value = vdata id class</pre>	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TAG	WML	HTML
15. input	<pre><input/> name = nmtoken value = vdata type = (text password) format = cdata emptyok = boolean size = number maxlength = number title = vdata tabindex = number xml:lang id class</pre>	<pre><input> name = cdata value = cdata type = (text password checkbox radio submit reset file hidden image button) size = cdata maxlength = number title tabindex lang id class checked src = uri style alt align accept readonly disabled accesskey usemap onfocus onblur onselect onchange onclick ondblclick</pre>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TAG	WML	HTML
16. select	<code><select> </select></code> multiple = boolean name = nmtoken value = vdata iname = nmtoken ivalue = vdata title = vdata tabindex = number xml:lang id class	<code><select> </select></code> multiple name = cdata title tabindex lang id class size = number style disabled onclick ondblclick onmousedown onmouseup onmouseover onmousemove onmouseout onkeypress onkeydown onkeyup
17. option	<code><option> </option></code> value = vdata title = vdata onpick = href xml:lang id class	<code><option> </option></code> value = cdata title lang id class label = text name = cdata size = number

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TAG	WML	HTML
		multiple style disabled tabindex onclick ondblclick onmousedown onmouseup onmouseover
18. optgroup	<pre><optgroup> </optgroup> title = vdata xml:lang id class</pre>	<pre><optgroup> </optgroup> title lang id class label = text</pre>
19. fieldset	<pre><fieldset> </fieldset> title = vdata xml:lang id class</pre>	
20. anchor	<pre><anchor> </anchor> title = vdata xml:lang id class</pre>	
21. a	<pre><a> title = vdata href xml:lang id</pre>	<pre><a> href name rel rev</pre>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TAG	WML	HTML
	class	tabindex target accesskey
22. img	<code></code> alt = vdata src = href localsrc = vdata vspace = length hspace = length align = (top middle bottom) height = length width = length xml:lang id class	<code></code> alt src = uri longdesc = uri border vspace hspace align height width lang id class title style ismap usemap onclick ondblclick onmousedown onmouseup onmouseover onmousemove onmouseout onkeypress onkeydown onkeyup

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TAG	WML	HTML
23. timer	<timer/> name = nmtoken value = vdata id class	
24. em	 	
25. strong	 	
26. i	<i> </i>	<i> </i>
27. b	 	
28. u	<u> </u>	<u> </u>
29. big	<big> </big>	<big> </big>
30. small	<small> </small>	<small> </small>
31. br	 xml:lang id class	 id class clear title style
32. p	<p> </p> align = (left right center) mode = (wrap nowrap) xml:lang id class	<p> </p> align lang id class title style onclick ondblclick onmousedown onmouseup onmouseover

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TAG	WML	HTML
		onmousemove onmouseout onkeypress onkeydown onkeyup
33. table	<table> </table> title = vdata align = cdata columns = number xml:lang id class	<table> </table> title align = left center right lang id class summary = text width = length style onclick ondblclick onmousedown onmouseup onmouseover onmousemove onmouseout onkeypress onkeydown onkeyup bgcolor frame rules border cellpadding cellspacing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TAG	WML	HTML
34. tr	<tr> </tr> id class	<tr> </tr> id class lang title style onclick ondblclick onmousedown onmouseup onmouseover onmousemove onmouseout onkeypress onkeydown onkeyup align char charoff valign
35. td	<td> </td> xml:lang id class	<td> </td> lang id class headers = idrefs scope = scope-name abbr = text

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TAG	WML	HTML
		axis = cdata rowspan = number colspan = number nowrap width = pixels height = pixels title style onclick ondblclick onmousedown onmouseup onmouseover onmousemove onmouseout onkeypress onkeydown onkeyup bgcolor align char charoff valign

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ขั้นตอนการสร้างเอกสาร XSL สำหรับใช้ในการแปลงเอกสาร

การสร้างเอกสาร XSL สำหรับใช้ในการแปลงเอกสาร อาศัยเท็กพื้นฐานที่เหมือนกันในทั้ง 2 ภาษา เป็นหลักในการสร้าง ตัวอย่างที่ 3.1

```
<xsl:stylesheet xmlns:xsl = "http://www.w3.org/1999/XSL/Transform" >
  <xsl:output method = "xml" doctype-public = "-//WAPFORUM//DTD WML 1.1//EN"
    doctype-system = "http://www.wapforum.org/DTD/wml_1.1.xml" />
  <!-- match root element -->
  <xsl:template match="/">
    <wml>
      <card title="{//title}">
        <xsl:apply-templates select = "../body"/>
      </card>
    </wml>
  </xsl:template>
  <!-- meta part -->
  <!-- body part -->
  <xsl:template match="body">
    <p><xsl:apply-templates /></p>
  </xsl:template>
  <xsl:template match = "br"> <br/>
  </xsl:template>
  <xsl:template match = "b|strong">
    <b><xsl:apply-templates /></b>
  </xsl:template>
  <xsl:template match="img">
    <img src = "{@src}" alt = "{@alt}" width = "{@width}" height = "{@height}"/>
  </xsl:template>
</xsl:stylesheet>
```

ตัวอย่างที่ 3.1 แสดงเอกสาร XSL อย่างง่าย สำหรับการแปลงเอกสาร HTML เป็น WML

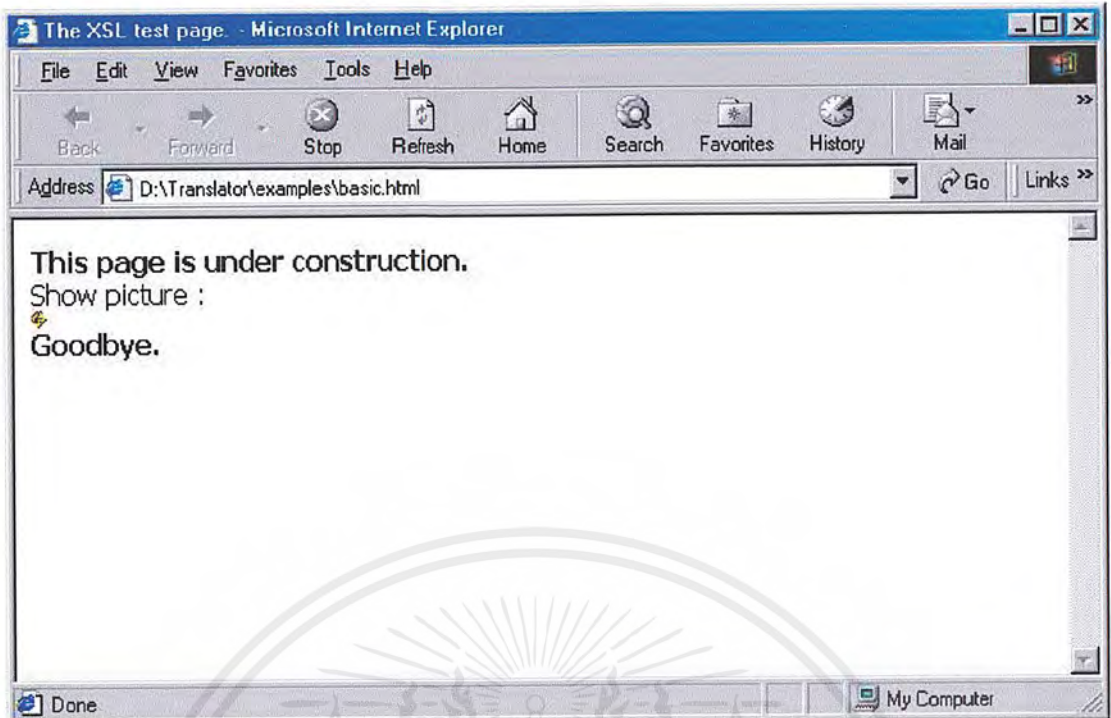
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถใช้ออกสาร XSL ที่ได้จากตัวอย่างที่ 3.1 มาใช้งานโดยการแปลงเอกสาร HTML ที่มีรายละเอียดดังตัวอย่างที่ 3.2 ให้เป็นเอกสาร WML และการแสดงผลเป็นไปดังรูปที่ 3.2

```
<html>
<head>
  <title>The XSL test page.</title>
</head>
<body>
  <b>This page is under construction.</b><br/>
  Show picture : <br/>
<br/>
  <strong>Goodbye.</strong>
</body>
</html>
```

ตัวอย่างที่ 3.2 แสดงรายละเอียดของเอกสาร HTML สำหรับทดสอบการทำงานของเอกสาร XSL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แสดงผลการเรียกดูเอกสาร HTML ที่ใช้ทดสอบการทำงานของเอกสาร XSL



```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE wml PUBLIC "-//WAPFORM//DTD WML
1.1/EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml><card title="The XSL test page."><p>
<b>This page is under construction.</b><br>
Show picture : <br/>
<b>Goodbye.</b>
</p></card></wml>
```

รูปที่ 3.2 แสดงเอกสาร WML ที่ได้จากการกำหนดของเอกสาร XSL และผลการเรียกดูเอกสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ขั้นตอนการออกแบบและสร้างตัวแปลงภาษารูปแบบ XSL

ตัวแปลงภาษารูปแบบ XSL มีอินพุต คือ เอกสารที่ต้องการทำการแปลง และเอาต์พุตที่ได้จากตัวแปลงภาษารูปแบบ XSL คือ เอกสารที่ผ่านการดำเนินการแปลงแล้ว ดังแสดงในรูปที่ 3.3



รูปที่ 3.3 แสดงการรับเอกสารอินพุตและเอาต์พุตของตัวแปลงภาษารูปแบบ XSL

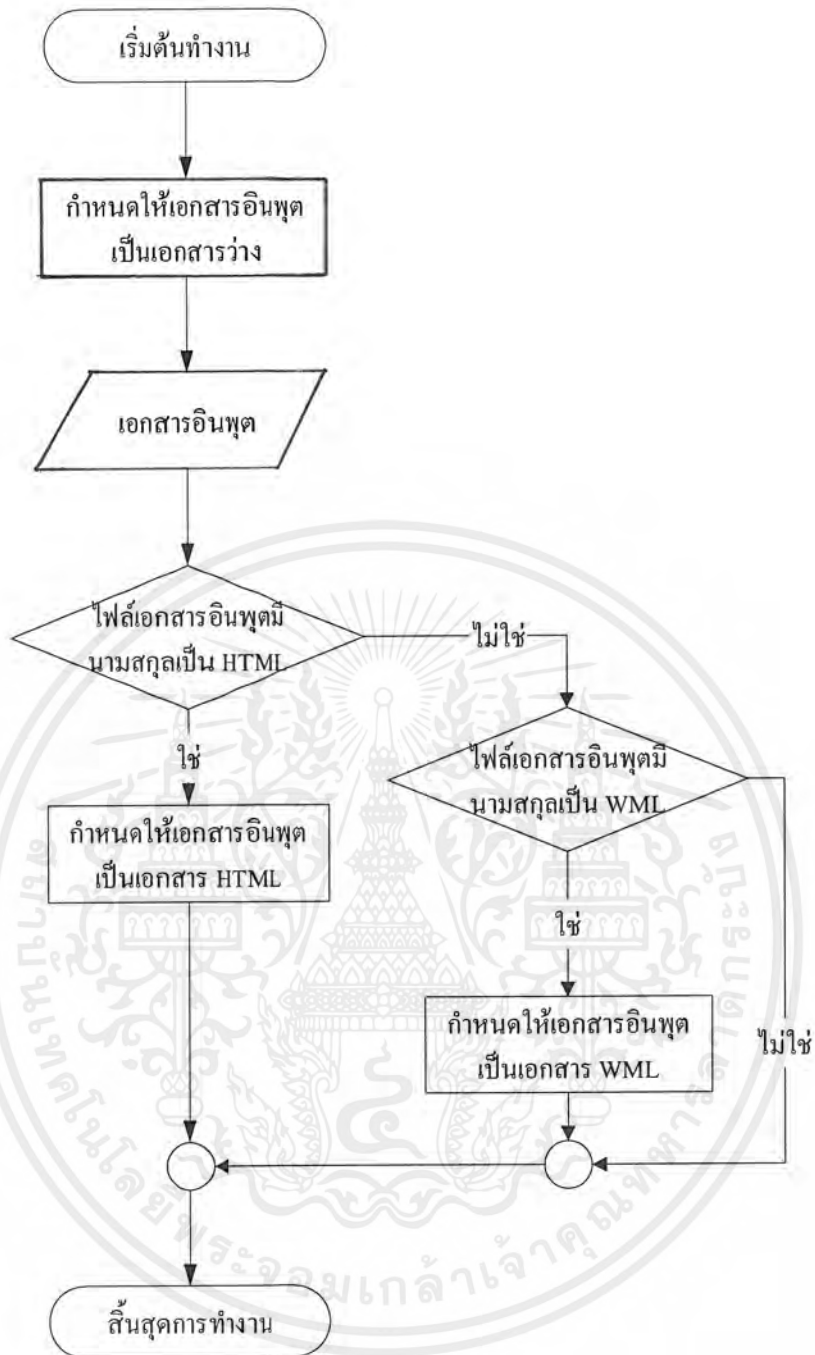
ถ้าเอกสารอินพุตเป็นเอกสาร HTML แล้ว เอกสารเอาต์พุตที่ได้ภายหลังการดำเนินการ คือ เอกสาร WML แต่ถ้าเอกสารอินพุตเป็นเอกสาร WML เอกสารเอาต์พุตที่ได้ภายหลังการดำเนินการ คือ เอกสาร HTML

ในการดำเนินการของตัวแปลงภาษารูปแบบ XSL อาศัยเอกสารอินพุตร่วมกับเอกสาร XSL สำหรับกำหนดกฎในการแปลงเอกสาร ซึ่งอยู่ภายในตัวแปลงภาษารูปแบบ XSL (เป็นเอกสารอยู่ภายใน มี 2 เอกสาร คือเอกสาร XSL สำหรับการแปลงเอกสาร HTML เป็น WML และเอกสาร XSL สำหรับการแปลงเอกสาร WML ให้เป็น HTML) ซึ่งการดำเนินการแปลงเอกสารมีขั้นตอนดังแสดงในรูปที่ 3.4



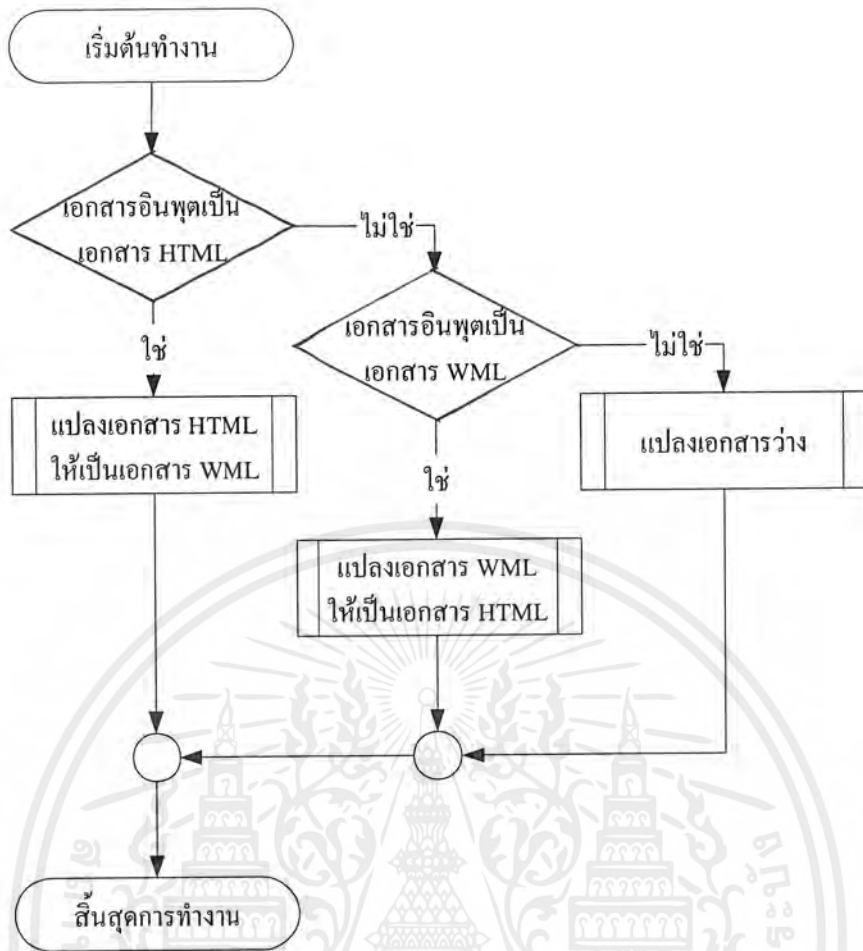
รูปที่ 3.4 แผนภาพ Flow Chart แสดงขั้นตอนการแปลงเอกสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



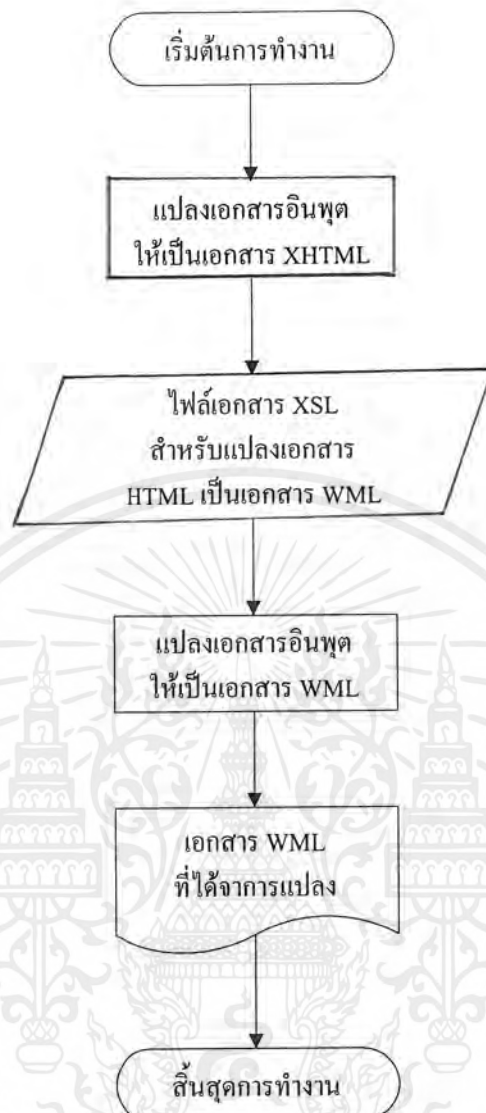
รูปที่ 3.5 แผนภาพ Flow Chart แสดงขั้นตอนการการทำงานของฟังก์ชันย่อยอ่านเอกสารอินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แผนภาพ Flow Chart แสดงขั้นตอนการทำงานของฟังก์ชันย่อยแปลงเอกสารอินพุตให้เป็นเอกสารเอชทีทูด

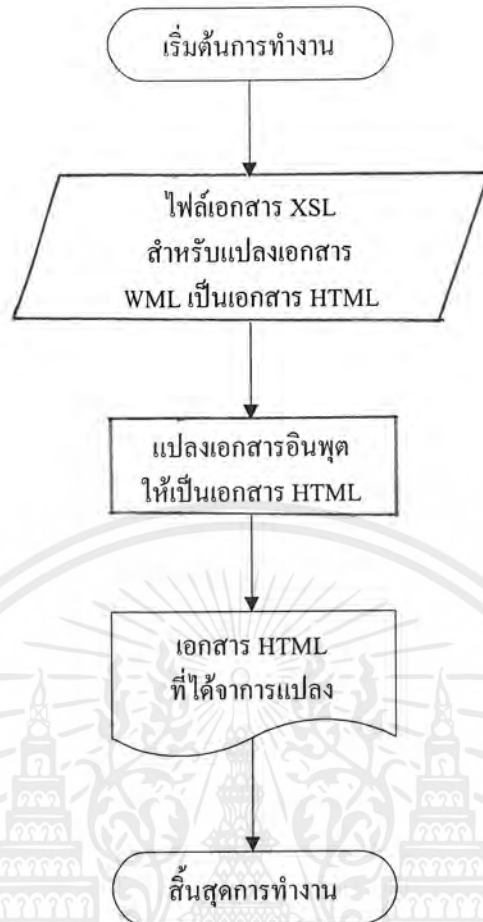
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 แผนภาพ Flow Chart แสดงขั้นตอนการการทำงานของฟังก์ชันย่อยแปลงเอกสาร HTML ให้เป็นเอกสาร WML

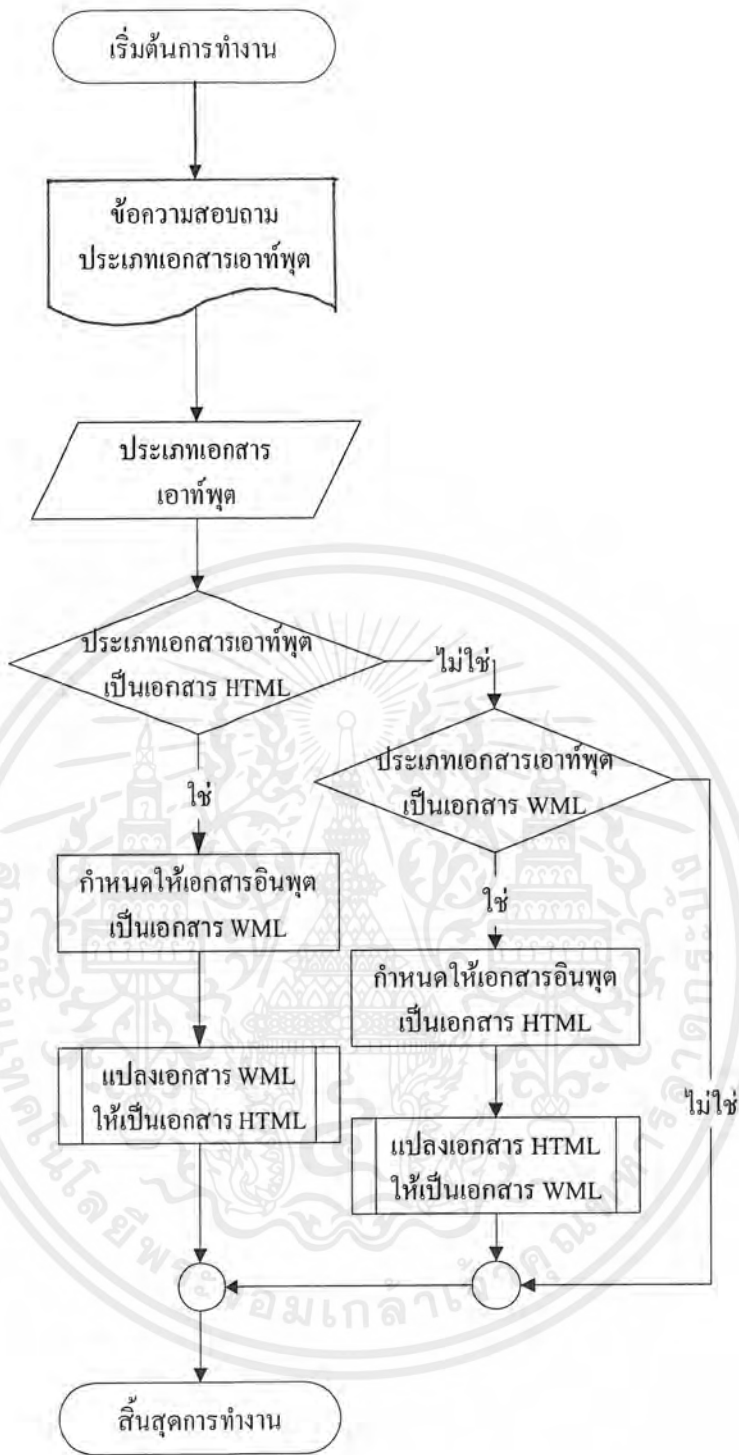
ขั้นตอนการแปลงเอกสารอินพุตให้เป็นเอกสาร XHTML ต้องดำเนินการเนื่องจาก XSL เป็นภาษาสำหรับกำหนดรูปแบบการแสดงผลของเอกสาร XML ดังนั้นจึงต้องแปลงเอกสาร HTML ให้เป็นเอกสาร XHTML ซึ่งมีคุณสมบัติเหมือนกับเอกสาร XML เพื่อให้สามารถดำเนินการแปลงเอกสารได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แผนภาพ Flow Chart แสดงขั้นตอนการทำงานของฟังก์ชันย่อยแปลงเอกสาร WML ให้เป็นเอกสาร HTML

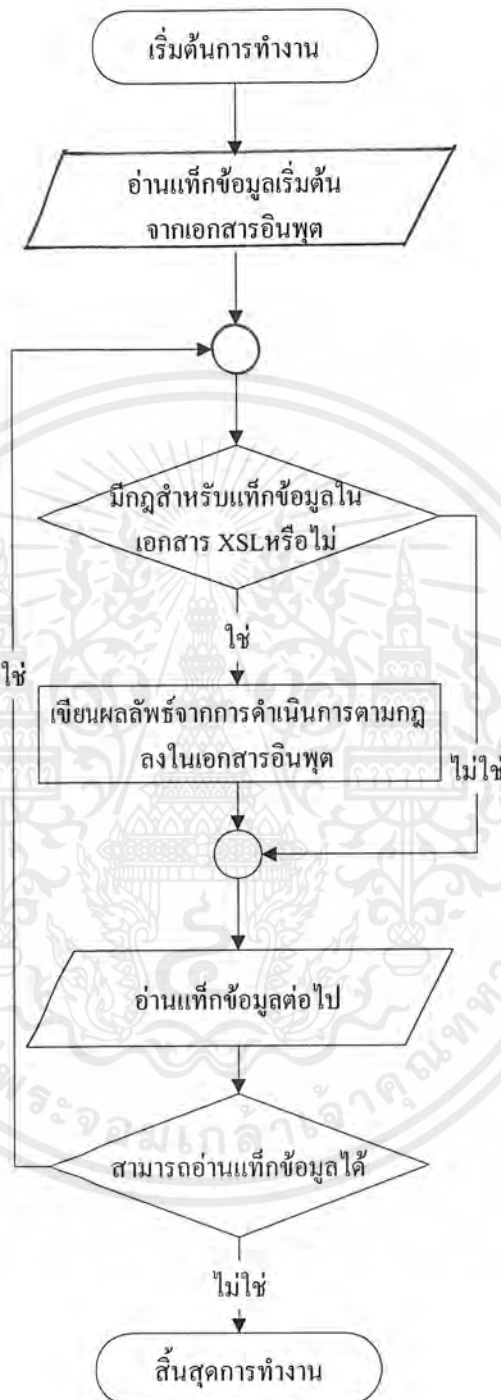
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แผนภาพ Flow Chart แสดงขั้นตอนการการทำงานของฟังก์ชันย่อยแปลงเอกสารว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.7. และรูปที่ 3.8. ขั้นตอนการแปลงเอกสารอินพุตให้เป็นเอกสาร WML และขั้นตอนการแปลงเอกสารอินพุตให้เป็นเอกสาร HTML สามารถดำเนินการได้ตามแผนผังในรูปที่ 3.10.



รูปที่ 3.10 แผนภาพ Flow Chart แสดงขั้นตอนการแปลงเอกสารอินพุตให้เป็นเอกสารเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เข้าใจในขั้นตอนการแปลงเอกสารอินพุตให้เป็นเอกสารเอาต์พุตมากขึ้น จึงขอยกตัวอย่างประกอบ แล้วแสดงการสร้างเอกสารเอาต์พุตตามขั้นตอนทีละขั้น โดยมีเอกสารอินพุตดังตัวอย่างที่ 3.3. และมีเอกสาร XSL ดังตัวอย่างที่ 3.4.

<html>	-----1
<body>	-----2
Hello World! this is me. 	-----3
<u>underline.</u>	-----4
</body>	-----5
</html>	-----6

ตัวอย่างที่ 3.3. เอกสารอินพุตสำหรับแสดงขั้นตอนการแปลงเอกสารเอกสารอินพุตให้เป็นเอกสารเอาต์พุต



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<xsl:stylesheet xmlns:xsl = "http://www.w3.org/1999/XSL/Transform" > -----1
  <xsl:template match="/"> -----2
    <wml> -----3
      <card title = "Untitle."> -----4
        <xsl:apply-templates select = "../body"/> -----5
      </card> -----6
    </wml> -----7
  </xsl:template> -----8
  <xsl:template match="body"> -----9
    <p><xsl:apply-templates /></p> -----10
  </xsl:template> -----11
  <xsl:template match = "br"> <br/> -----12
  </xsl:template> -----13
  <xsl:template match = "b|strong"> -----14
    <b><xsl:apply-templates /></b> -----15
  </xsl:template> -----16
</xsl:stylesheet> -----17

```

ตัวอย่างที่ 3.4 แสดงเอกสาร XSL สำหรับแสดงขั้นตอนการแปลงเอกสารอินพุตให้เป็นเอกสารเอาต์พุต

ขั้นตอนการแปลงเอกสารอินพุตให้เป็นเอกสารเอาต์พุตเริ่มจากการอ่านแท็กข้อมูลเริ่มต้นในเอกสารอินพุต จากตัวอย่างแท็กข้อมูลเริ่มต้นของเอกสารอินพุตคือ แท็ก <html> จากนั้นจึงทำค้นหากฎที่กำหนดไว้ในเอกสาร XSL พบว่ามีกฎสำหรับการปรับเปลี่ยนโครงสร้างตรงกับบรรทัดที่ 2 ในตัวอย่างที่ 3.4. จึงดำเนินการตามกฎที่กำหนดไว้ คือ เขียนข้อมูล <wml> <card title = "Untitle."> ลงเอกสารเอาต์พุต ทำการอ่านแท็กข้อมูลต่อไป ได้แท็ก <body> ทำการค้นหากฎและดำเนินการตามกฎที่เขียนในเอกสาร XSL เมื่อดำเนินการกับแท็ก <body> เสร็จจึงดำเนินการเขียนข้อมูล </wml> ลงในเอกสารเอาต์พุต

เมื่อทำการค้นหากฎสำหรับการแปลงเอกสารในเอกสาร XSL พบว่า มีกฎการแปลงเอกสารสำหรับแท็ก <body> ในบรรทัดที่ 9 ในตัวอย่างที่ 3.4. ทำการเขียนข้อมูล <p>และทำการอ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แท็กข้อมูลต่อไปและดำเนินการค้นหาและทำตามกฎไปจนไม่มีแท็กข้อมูล เมื่อดำเนินการเสร็จแล้ว จึงทำการเขียนข้อมูล </p> ลงในเอกสารเอาพุต

ผลจากการดำเนินการได้เอกสารเอาท์พุต ดังตัวอย่างที่ 3.5.

```
<wml>
  <card title ="Untitle.">
    <p>
      Hello World! <b>this is me.</b><br/>
    </p>
  </card>
</wml>
```

ตัวอย่างที่ 3.5 แสดงเอกสาร XSL สำหรับแสดงขั้นตอนการแปลงเอกสารอินพุตให้เป็นเอกสารเอาท์พุต



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลจากการดำเนินการของตัวแปลงภาษารูปแบบ XSL

ในการศึกษาผลจากการดำเนินการของตัวแปลงภาษารูปแบบ XSL ต้องทำการติดตั้งตัวแปลงภาษารูปแบบ XSL ก่อน โดยทำตามขั้นตอนการติดตั้งโปรแกรมตัวแปลงภาษารูปแบบ XSL ที่ระบุไว้ในภาคผนวก ค. เมื่อติดตั้งตัวแปลงภาษารูปแบบ XSL แล้วจึงทำการศึกษผลจากการดำเนินการของตัวแปลงภาษารูปแบบ XSL ซึ่งผลจากการดำเนินการของตัวแปลงภาษารูปแบบ XSL แบ่งออกเป็น 2 หัวข้อ ดังนี้

4.1 ผลจากการดำเนินการแปลงเอกสาร HTML ให้เป็นเอกสาร WML

4.2 ผลจากการดำเนินการแปลงเอกสาร WML ให้เป็นเอกสาร HTML

การดูผลการแปลงเอกสารของตัวแปลงภาษารูปแบบ XSL สามารถทำได้ตามขั้นตอนที่ระบุว่าคู่มือการใช้โปรแกรมตัวแปลงภาษารูปแบบ XSL ซึ่งอยู่ในภาคผนวก ค.

4.1 ผลจากการดำเนินการแปลงเอกสาร HTML ให้เป็นเอกสาร WML

ทดสอบผลการดำเนินการของตัวแปลงภาษารูปแบบ XSL ด้วยการกำหนดเอกสารที่ต้องการแปลงเป็นเอกสาร HTML ดังตัวอย่างที่ 4.1

```
<html>
  <head> <title>XSLTranslator example.</title></head>
  <body> <font size="5"><b>XSLTranslator</b></font><br>
    Project logo : <br>
    <i>Students</i><br>
    <table> <tr> <td>StdID</td> <td>Name</td></tr>
      <tr> <td>40056035</td><td>Thana</td></tr>
      <tr><td>40056084</td><td>Sasitorn</td></tr>
      <tr><td>40056109</td><td>Apichai</td></tr> </table>
  </body>
</html>
```

ตัวอย่างที่ 4.1 แสดงเอกสาร HTML ที่ใช้ในการทดสอบการดำเนินการของตัวแปลงภาษารูปแบบ XSL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้วดูผลลัพธ์ที่เป็นเอกสาร WML จากการแปลงของตัวแปลงภาษารูปแบบ XSL ผลลัพธ์ที่ได้เป็นดังต่อไปนี้

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml><card title="XSLTranslator."><p>
<big><b>XSLTranslator</b></big><br/>
Project logo : <br/>
<i>Students</i><br/>
<table columns="2">
<tr><td>StdID</td><td>Name</td></tr>
<tr><td>40056035</td><td>Thana</td></tr>
<tr><td>40056084</td><td>Sasitorn</td></tr>
<tr><td>40056109</td><td>Apichai</td></tr></table>
</p></card></wml>
```

4.2 ผลจากการดำเนินการแปลงเอกสาร WML ให้เป็นเอกสาร HTML

ทดสอบผลการดำเนินการของตัวแปลงภาษารูปแบบ XSL ด้วยการกำหนดเอกสารที่ต้องการแปลงเป็นเอกสาร WML ที่เป็นผลลัพธ์ของเอกสาร ในตัวอย่างที่ 4.1 ซึ่งมีรายละเอียดดังตัวอย่างที่ 4.2

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml><card title="XSLTranslator."><p>
<big><b>XSLTranslator</b></big><br/>
Project logo : <br/>
<i>Students</i><br/>
<table columns="2">
<tr><td>StdID</td><td>Name</td></tr>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<tr><td>40056035</td><td>Thana</td></tr>
<tr><td>40056084</td><td>Sasitorn</td></tr>
<tr><td>40056109</td><td>Apichai</td></tr></table>
</p></card></wml>

```

ตัวอย่างที่ 4.2 แสดงเอกสาร WML ที่ใช้ในการทดสอบการดำเนินการของ ตัวแปลงภาษารูปแบบ XSL

แล้วดูผลลัพธ์ที่เป็นเอกสาร HTML จากการแปลงของตัวแปลงภาษารูปแบบ XSL ผลลัพธ์ที่ได้เป็นดังต่อไปนี้

```

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
  <body>
    <p align="left">
      <big><b>XSLTranslator</b></big><br>
      Project logo : <br>
      <i>Students</i>
      <br>
      <table>
        <tr><td>StdID</td><td>Name</td></tr>
        <tr><td>40056035</td><td>Thana</td></tr>
        <tr><td>40056084</td><td>Sasitorn</td></tr>
        <tr><td>40056109</td><td>Apichai</td></tr>
      </table>
    </p>
  </body>
</html>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การอภิปรายผลการดำเนินการของตัวแปลงภาษารูปแบบ XSL

จากการดำเนินการของตัวแปลงภาษารูปแบบ XSL จะเห็นว่า ตัวแปลงภาษารูปแบบ XSL สามารถแปลงเอกสาร HTML เป็น WML และแปลงเอกสาร WML เป็น HTML ได้ โดยพิจารณาแปลงเฉพาะแท็กที่มีเหมือนกันในทั้งสองภาษา ส่วนในแท็กที่ไม่เหมือนกัน ตัวแปลงภาษารูปแบบ XSL จะทำการพิจารณาแปลง ให้เป็นแท็กที่มีการแสดงผลหรือมีความหมายใกล้เคียงกัน เช่น แท็ก font ในเอกสาร HTML ที่มีแอตทริบิวต์ชื่อ size สำหรับกำหนดขนาดตัวอักษร ถ้าค่าของแอตทริบิวต์มากกว่า 3 หมายความว่า ตัวอักษรภายในแท็กดังกล่าวเป็นตัวใหญ่กว่าปกติ ตัวแปลงภาษารูปแบบ XSL จะดำเนินการแปลงให้เป็นแท็ก big ในภาษา WML แทน หรือแอตทริบิวต์ xml:lang ในเอกสาร WML จะถูกแปลงเป็นแอตทริบิวต์ชื่อ lang ในเอกสาร HTML แทน ทั้งนี้เพื่อให้เอกสารหลังจากการดำเนินการมีความใกล้เคียงกันมากที่สุด

การดำเนินการของตัวแปลงภาษารูปแบบ XSL สามารถดำเนินการได้เป็นอย่างดี ในกรณีที่เอกสารมีความซับซ้อนไม่มาก ทั้งนี้เนื่องมาจากความแตกต่างกันของโครงสร้างเอกสาร HTML และเอกสาร WML ที่ถูกออกแบบมาเพื่อแสดงผลในอุปกรณ์ที่ต่างกัน สามารถพิจารณาได้จาก Document Type Definition ของแต่ละภาษา การที่เอกสารมีความซับซ้อนมาก ส่งผลให้เอกสารหลังจากการดำเนินการไม่สามารถแสดงผลในอุปกรณ์แสดงผลได้จริง หรือแสดงผลผิดเพี้ยนไปจากเอกสารต้นแบบ (ส่วนมากเกิดขึ้นในกรณีการแปลงเอกสาร HTML เป็นเอกสาร WML เนื่องจากเอกสาร HTML มีโครงสร้างภาษาที่ซับซ้อนกว่า) เช่น ถ้ามีแท็ก p ซ้อนแท็ก p ในเอกสาร HTML ผลจากการดำเนินการของตัวแปลงภาษารูปแบบ XSL จะได้เอกสาร WML ที่มีแท็ก p ซ้อนแท็ก p เมื่อนำเอกสารผลลัพธ์ที่ได้ไปแสดงผลในโปรแกรมจำลองการแสดงผลเอกสาร WML จะพบว่าบางโปรแกรมสามารถแสดงผลได้ แต่เอกสารผิดเพี้ยนไปจากเดิมและบางโปรแกรมไม่สามารถแสดงผลได้ โปรแกรมที่สามารถแสดงผลได้ ทำทดสอบโดยใช้โปรแกรม WapIDE 3.0 ส่วนโปรแกรมที่ไม่สามารถแสดงผลได้ ทดสอบโดยใช้โปรแกรม NOKIA WAP TOOLKIT Version 1.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

6.1 สรุปผลการวิจัย

ตัวแปลงภาษารูปแบบ XSL สร้างขึ้นด้วยภาษา Java เป็นโปรแกรมที่ทำหน้าที่แปลงเอกสาร HTML ให้เป็นเอกสาร WML และแปลงเอกสาร WML ให้เป็นเอกสาร HTML โดยการนำเอกสารอินพุตมาเปรียบเทียบกับกฎการปรับเปลี่ยนโครงสร้างเอกสารที่เขียนขึ้นในเอกสาร XSL เพื่อสร้างเอกสารเอาต์พุต ซึ่งการดำเนินการของตัวแปลงภาษารูปแบบ XSL เป็นไปโดยอัตโนมัติ กล่าวคือ ถ้าเอกสารเอกสารอินพุตเป็นเอกสาร HTML ผลลัพธ์ที่ได้คือ เอกสาร WML ในทางกลับกัน ถ้าเอกสารอินพุตเป็นเอกสาร WML ผลลัพธ์คือ เอกสาร HTML

ผลการดำเนินการของตัวแปลงภาษารูปแบบ XSL มีประสิทธิภาพสูง ถ้าเอกสารที่ต้องการให้ตัวแปลงภาษารูปแบบ XSL ดำเนินการแปลงมีความซับซ้อนไม่มาก แต่ถ้าเอกสารมีความซับซ้อนมาก อาจทำให้เอกสารหลังจากการดำเนินการไม่สามารถแสดงผลในอุปกรณ์แสดงผลได้จริง หรือแสดงผลผิดเพี้ยนไปมาจากเอกสารต้นแบบ ทั้งนี้เนื่องจากความแตกต่างกันของโครงสร้างภาษา และข้อจำกัดในด้านต่าง ๆ จึงทำให้เอกสาร XSL ที่ใช้สำหรับกำหนดกฎการแปลงเอกสารมีประสิทธิภาพไม่เต็มที่

6.2 ข้อเสนอแนะ

ปัญหาพิเศษเรื่อง ตัวแปลงภาษารูปแบบ XSL สามารถนำไปพัฒนาต่อเพื่อให้โปรแกรมมีประสิทธิภาพมากขึ้น โดยเพิ่มเติมความสามารถจากเดิมที่มีเอกสาร XSL สำหรับแปลง HTML เป็น WML และเอกสาร XSL สำหรับแปลงเอกสาร WML เป็น HTML เพียง 2 เอกสาร ให้มีเอกสาร XSL ที่เป็นกฎสำหรับการแปลงเอกสารมากขึ้น เพื่อรองรับภาษาที่เกิดขึ้นเพิ่มเติมในอนาคต เช่น เอกสาร XSL สำหรับแปลงเอกสาร HTML เป็นเอกสาร cHTML ซึ่งเป็นภาษาที่ใช้สร้างเอกสารสำหรับการแสดงผลในระบบ i-Mode เป็นต้น หรือนำส่วนประกอบของตัวแปลงภาษารูปแบบ XSL ที่ทำหน้าที่ในการแปลงเอกสารไปรวมกับส่วนที่เป็น browser เพื่อให้ browser สามารถแสดงผลเอกสารได้หลายรูปแบบ เช่น สามารถแสดงผลเอกสาร WML ได้ โดยส่วนที่หน้าที่ในการแปลงเอกสารจะแปลงเอกสารที่ browser รับมาให้เป็นเอกสาร HTML เพื่อให้ browser แสดงผลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

Document Type Definition แสดงโครงสร้างภาษา XSLT

```
<!ENTITY % result-elements "
```

```
  | fo:inline-sequence
```

```
  | fo:block
```

```
">
```

```
<!ENTITY % instructions "
```

```
  | xsl:apply-templates
```

```
  | xsl:apply-imports
```

```
  | xsl:for-each
```

```
  | xsl:value-of
```

```
  | xsl:number
```

```
  | xsl:counter
```

```
  | xsl:counters
```

```
  | xsl:counter-increment
```

```
  | xsl:counter-reset
```

```
  | xsl:counter-scope
```

```
  | xsl:choose
```

```
  | xsl:if
```

```
  | xsl:contents
```

```
  | xsl:invoke
```

```
  | xsl:text
```

```
  | xsl:pi
```

```
  | xsl:comment
```

```
  | xsl:element
```

```
  | xsl:attribute
```

```
  | xsl:use
```

```
  | xsl:copy
```

```
">
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!ENTITY % template "
    (#PCDATA
    %instructions;
    %result-elements;)*
">
<!ENTITY % space-att "xml:space (default|preserve) #IMPLIED">
<!ELEMENT xsl:stylesheet
    (xsl:import*,
    (xsl:include
    | xsl:id
    | xsl:strip-space
    | xsl:preserve-space
    | xsl:macro
    | xsl:attribute-set
    | xsl:constant
    | xsl:template)*)
>
<!ATTLIST xsl:stylesheet
    result-ns NMTOKEN #IMPLIED
    default-space (preserve|strip) "preserve"
    indent-result (yes|no) "no"
    id ID #IMPLIED
    xmlns:xsl CDATA #FIXED "http://www.w3.org/TR/W3C-xsl"
    %space-att;
>
<!-- Used for attribute values that are URIs.-->
<!ENTITY % URI "CDATA">
<!-- Used for attribute values that are patterns.-->
<!ENTITY % pattern "CDATA">
<!-- Used for attribute values that are a priority. -->
<!ENTITY % priority "NMTOKEN">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!ELEMENT xsl:import EMPTY>
<!ATTLIST xsl:import href %URI; #REQUIRED>
<!ELEMENT xsl:include EMPTY>
<!ATTLIST xsl:include href %URI; #REQUIRED>
<!ELEMENT xsl:id EMPTY>
<!ATTLIST xsl:id
    attribute NMTOKEN #REQUIRED
    element NMTOKEN #IMPLIED
>
<!ELEMENT xsl:strip-space EMPTY>
<!ATTLIST xsl:strip-space element NMTOKEN #REQUIRED>
<!ELEMENT xsl:preserve-space EMPTY>
<!ATTLIST xsl:preserve-space element NMTOKEN #REQUIRED>
<!ELEMENT xsl:template %template;>
<!ATTLIST xsl:template
    match %pattern; #REQUIRED
    priority %priority; #IMPLIED
    mode NMTOKEN #IMPLIED
    %space-att;
>
<!ELEMENT xsl:value-of EMPTY>
<!ATTLIST xsl:value-of select CDATA #IMPLIED>
<!ENTITY % conversion-atts '
    format CDATA "1"
    xml:lang NMTOKEN #IMPLIED
    letter-value (alphabetic|other) #IMPLIED
    digit-group-sep CDATA #IMPLIED
    n-digits-per-group NMTOKEN #IMPLIED
    sequence-src %URI; #IMPLIED
'>
<!ELEMENT xsl:number EMPTY>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!ATTLIST xsl:number
    level (single|multi|any) "single"
    count CDATA #IMPLIED
    from CDATA #IMPLIED
    %conversion-atts;
>
<!ELEMENT xsl:counter EMPTY>
<!ATTLIST xsl:counter
    name NMTOKEN #REQUIRED
    %conversion-atts;
>
<!ELEMENT xsl:counters EMPTY>
<!ATTLIST xsl:counters
    name NMTOKEN #REQUIRED
    %conversion-atts;
>
<!ELEMENT xsl:counter-increment EMPTY>
<!ATTLIST xsl:counter-increment
    name NMTOKEN #REQUIRED
    amount NMTOKEN #IMPLIED
>
<!ELEMENT xsl:counter-reset EMPTY>
<!ATTLIST xsl:counter-reset
    name NMTOKEN #REQUIRED
    value NMTOKEN #IMPLIED
>
<!ELEMENT xsl:counter-scope %template;>
<!ATTLIST xsl:counter-scope %space-att;>
<!ELEMENT xsl:apply-templates (xsl:sort*)>
<!ATTLIST xsl:apply-templates
    select %pattern; #IMPLIED

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mode NMTOKEN #IMPLIED
>
<!ELEMENT xsl:apply-imports EMPTY>
<!-- xsl:sort cannot occur after any other elements or
any non-whitespace character -->
<!ELEMENT xsl:for-each
    (#PCDATA
    %instructions;
    %result-elements;
    | xsl:sort)*
>
<!ATTLIST xsl:for-each
    select %pattern; #REQUIRED
    %space-att;
>
<!ELEMENT xsl:sort EMPTY>
<!ATTLIST xsl:sort
    select %pattern; "."
    lang CDATA #IMPLIED
    data-type (text|number) "text"
    order (ascending|descending) "ascending"
    case-order (upper-first|lower-first) #IMPLIED
>
<!ELEMENT xsl:if %template;>
<!ATTLIST xsl:if
    test %pattern; #REQUIRED
    %space-att;
>
<!ELEMENT xsl:choose (xsl:when+ , xsl:otherwise?)>
<!ATTLIST xsl:choose %space-att;>
<!ELEMENT xsl:when %template;>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!ATTLIST xml:when
    test %pattern; #REQUIRED
    %space-att;
>
<!ELEMENT xml:otherwise %template;>
<!ATTLIST xml:otherwise %space-att;>
<!ELEMENT xml:attribute-set (xml:attribute|xml:use)*>
<!ATTLIST xml:attribute-set
    name NMTOKEN #REQUIRED
>
<!ELEMENT xml:constant EMPTY>
<!ATTLIST xml:constant
    name NMTOKEN #REQUIRED
    value CDATA #REQUIRED
>
<!-- xml:macro-arg cannot occur after any other elements or
any non-whitespace character -->
<!ELEMENT xml:macro
    (#PCDATA
    %instructions;
    %result-elements;
    | xml:macro-arg)*
>
<!ATTLIST xml:macro
    name NMTOKEN #REQUIRED
    %space-att;
>
<!ELEMENT xml:macro-arg EMPTY>
<!ATTLIST xml:macro-arg
    name NMTOKEN #REQUIRED
    default CDATA #IMPLIED

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

>
<!-- This is allowed only within xsl:macro -->
<!ELEMENT xsl:contents EMPTY>
<!-- xsl:arg cannot occur after any other elements or
any non-whitespace character -->
<!ELEMENT xsl:invoke
    (#PCDATA
    %instructions;
    %result-elements;
    | xsl:arg)*
>
<!ATTLIST xsl:invoke
    macro NMTOKEN #REQUIRED
    %space-att;
>
<!ELEMENT xsl:arg EMPTY>
<!ATTLIST xsl:arg
    name NMTOKEN #REQUIRED
    value CDATA #REQUIRED
>
<!ELEMENT xsl:text (#PCDATA)>
<!ATTLIST xsl:text %space-att;>
<!ELEMENT xsl:pi %template;>
<!ATTLIST xsl:pi
    name CDATA #REQUIRED
    %space-att;
>
<!ELEMENT xsl:element %template;>
<!ATTLIST xsl:element
    name CDATA #REQUIRED
    %space-att;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

>
<!ELEMENT xsl:attribute %template;>
<!ATTLIST xsl:attribute
    name CDATA #REQUIRED
    %space-att;
>
<!ELEMENT xsl:use EMPTY>
<!ATTRIBUTE xsl:use attribute-set NMTOKEN #REQUIRED>
<!ELEMENT xsl:comment %template;>
<!ATTLIST xsl:comment %space-att;>
<!ELEMENT xsl:copy %template;>
<!ATTLIST xsl:copy %space-att;>

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

Document Type Definition แสดงโครงสร้างภาษา WML

<!--

Wireless Markup Language (WML) Document Type Definition.

WML is an XML language. Typical usage:

<?xml version="1.0"?>

<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"

"http://www.wapforum.org/DTD/wml_1.1.xml">

<wml>

...

</wml>

-->

<!ENTITY % length "CDATA"> <!-- [0-9]+ for pixels or [0-9]+%" for percentage length -->

>

<!ENTITY % vdata "CDATA"> <!-- attribute value possibly containing variable references -->

-->

<!ENTITY % HREF "%vdata;"> <!-- URI, URL or URN designating a hypertext node. May contain variable references -->

<!ENTITY % boolean "(true|false)">

<!ENTITY % number "NMTOKEN"> <!-- a number, with format [0-9]+ -->

<!ENTITY % coreattrs "id ID #IMPLIED class CDATA #IMPLIED">

<!ENTITY % emph "em | strong | b | i | u | big | small">

<!ENTITY % layout "br">

<!ENTITY % text "#PCDATA | %emph;">

<!-- flow covers "card-level" elements, such as text and images -->

<!ENTITY % flow "%text; | %layout; | img | anchor | a | table">

<!-- Task types -->

<!ENTITY % task "go | prev | noop | refresh">

<!-- Navigation and event elements -->

<!ENTITY % navelmts "do | onevent">

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!--===== Decks and Cards =====>
<!ELEMENT wml ( head?, template?, card+ )>
<!ATTLIST wml
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!-- card intrinsic events -->
<!ENTITY % cardev
    "onenterforward %HREF; #IMPLIED
    onenterbackward %HREF; #IMPLIED
    ontimer %HREF; #IMPLIED"
>
<!-- card field types -->
<!ENTITY % fields "%flow; | input | select | fieldset">
<!ELEMENT card (onevent*, timer?, (do | p)*)>
<!ATTLIST card
    title %vdata; #IMPLIED
    newcontext %boolean; "false"
    ordered %boolean; "true"
    xml:lang NMTOKEN #IMPLIED
    %cardev;
    %coreattrs;
>
<!--===== Event Bindings =====>
<!ELEMENT do (%task;)>
<!ATTLIST do
    type CDATA #REQUIRED
    label %vdata; #IMPLIED
    name NMTOKEN #IMPLIED
    optional %boolean; "false"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT onevent (%task;)>
<!ATTLIST onevent
    type CDATA #REQUIRED
    %coreattrs;
>
<!--===== Deck-level declarations =====-->
<!ELEMENT head ( access | meta )+>
<!ATTLIST head
    %coreattrs;
>
<!ELEMENT template (%navelmts;)*>
<!ATTLIST template
    %cardev;
    %coreattrs;
>
<!ELEMENT access EMPTY>
<!ATTLIST access
    domain CDATA #IMPLIED
    path CDATA #IMPLIED
    %coreattrs;
>
<!ELEMENT meta EMPTY>
<!ATTLIST meta
    http-equiv CDATA #IMPLIED
    name CDATA #IMPLIED
    forua %boolean; #IMPLIED
    content CDATA #REQUIRED
    scheme CDATA #IMPLIED

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    %coreattrs;
>
<!--===== Tasks =====>
<!ELEMENT go (postfield | setvar)*>
<!ATTLIST go
    href %HREF; #REQUIRED
    sendreferer %boolean; "false"
    method (post|get) "get"
    accept-charset CDATA #IMPLIED
    %coreattrs;
>
<!ELEMENT prev (setvar)*>
<!ATTLIST prev
    %coreattrs;
>
<!ELEMENT refresh (setvar)*>
<!ATTLIST refresh
    %coreattrs;
>
<!ELEMENT noop EMPTY>
<!ATTLIST noop
    %coreattrs;
>
<!--===== postfield =====>
<!ELEMENT postfield EMPTY>
<!ATTLIST postfield
    name %vdata; #REQUIRED
    value %vdata; #REQUIRED
    %coreattrs;
>
<!--===== variables =====>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!ELEMENT setvar EMPTY>
<!ATTLIST setvar
    name %vdata; #REQUIRED
    value %vdata; #REQUIRED
    %coreattrs;
>
<!--===== Card Fields =====>
<!ELEMENT select (optgroup|option)+>
<!ATTLIST select
    title %vdata; #IMPLIED
    name NMTOKEN #IMPLIED
    value %vdata; #IMPLIED
    iname NMTOKEN #IMPLIED
    ivalue %vdata; #IMPLIED
    multiple %boolean; "false"
    tabindex %number; #IMPLIED
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT optgroup (optgroup|option)+ >
<!ATTLIST optgroup
    title %vdata; #IMPLIED
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT option (#PCDATA | onevent)*>
<!ATTLIST option
    value %vdata; #IMPLIED
    title %vdata; #IMPLIED
    onpick %HREF; #IMPLIED
    xml:lang NMTOKEN #IMPLIED

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    %coreattrs;
>
<!ELEMENT input EMPTY>
<!ATTLIST input
    name NMTOKEN #REQUIRED
    type (text|password) "text"
    value %vdata; #IMPLIED
    format CDATA #IMPLIED
    emptyok %boolean; "false"
    size %number; #IMPLIED
    maxlength %number; #IMPLIED
    tabindex %number; #IMPLIED
    title %vdata; #IMPLIED
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT fieldset (%fields; | do)* >
<!ATTLIST fieldset
    title %vdata; #IMPLIED
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT timer EMPTY>
<!ATTLIST timer
    name NMTOKEN #IMPLIED
    value %vdata; #REQUIRED
    %coreattrs;
>
<!--===== Images =====>
<!ENTITY % IAlign "(top|middle|bottom)" >
<!ELEMENT img EMPTY>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!ATTLIST img
    alt %vdata; #REQUIRED
    src %HREF; #REQUIRED
    localsrc %vdata; #IMPLIED
    vspace %length; "0"
    hspace %length; "0"
    align %lAlign; "bottom"
    height %length; #IMPLIED
    width %length; #IMPLIED
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!--===== Anchor =====>
<!ELEMENT anchor ( #PCDATA | br | img | go | prev | refresh )*>
<!ATTLIST anchor
    title %vdata; #IMPLIED
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT a ( #PCDATA | br | img )*>
<!ATTLIST a
    href %HREF; #REQUIRED
    title %vdata; #IMPLIED
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!--===== Tables =====>
<!ELEMENT table (tr)+>
<!ATTLIST table
    title %vdata; #IMPLIED
    align CDATA #IMPLIED

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        columns %number; #REQUIRED
        xml:lang NMTOKEN #IMPLIED
        %coreattrs;
    >
<!ELEMENT tr (td)+>
<!ATTLIST tr
    %coreattrs;
>
<!ELEMENT td ( %text; | %layout; | img | anchor | a )*>
<!ATTLIST td
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!--===== Text layout and line breaks =====>
<!ELEMENT em (%flow;)*>
<!ATTLIST em
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT strong (%flow;)*>
<!ATTLIST strong
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT b (%flow;)*>
<!ATTLIST b
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT i (%flow;)*>
<!ATTLIST i

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT u (%flow;)*>
<!ATTLIST u
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT big (%flow;)*>
<!ATTLIST big
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT small (%flow;)*>
<!ATTLIST small
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ENTITY % TAlign "(left|right|center)">
<!ENTITY % WrapMode "(wrap|nowrap)" >
<!ELEMENT p (%fields; | do)*>
<!ATTLIST p
    align %TAlign; "left"
    mode %WrapMode; #IMPLIED
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;
>
<!ELEMENT br EMPTY>
<!ATTLIST br
    xml:lang NMTOKEN #IMPLIED
    %coreattrs;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

>

<!ENTITY quot "&#34;"> <!-- quotation mark -->

<!ENTITY amp "&#38;#38;"> <!-- ampersand -->

<!ENTITY apos "&#39;"> <!-- apostrophe -->

<!ENTITY lt "&#38;#60;"> <!-- less than -->

<!ENTITY gt "&#62;"> <!-- greater than -->

<!ENTITY nbsp "&#160;"> <!-- non-breaking space -->

<!ENTITY shy "&#173;"> <!-- soft hyphen (discretionary hyphen) -->



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

คู่มือการใช้โปรแกรม

คู่มือการใช้โปรแกรมตัวแปลงภาษารูปแบบ XSL มีหัวข้อต่าง ๆ ดังนี้

1. การติดตั้งโปรแกรมตัวแปลงภาษารูปแบบ XSL
2. ส่วนต่าง ๆ ของโปรแกรมตัวแปลงภาษารูปแบบ XSL
3. ตัวอย่างการใช้งานโปรแกรมตัวแปลงภาษารูปแบบ XSL

1. การติดตั้งโปรแกรมตัวแปลงภาษารูปแบบ XSL

ในการติดตั้งโปรแกรมตัวแปลงภาษารูปแบบ XSL เพื่อใช้งาน จำเป็นต้องติดตั้งโปรแกรม Java SDK¹ ตั้งแต่เวอร์ชัน 1.2.2 ขึ้นไปก่อน เพื่อใช้ในการรันโปรแกรมตัวแปลงภาษารูปแบบ XSL

หลังจากทำการติดตั้งโปรแกรม Java SDK แล้ว ให้ทำการเพิ่ม path ไปสู่โฟลเดอร์ bin ซึ่งเป็นโฟลเดอร์ย่อยของโฟลเดอร์ที่ทำการติดตั้งโปรแกรม Java SDK ไว้ เข้าไปในตัวแปร PATH สมมติว่าโฟลเดอร์ที่ติดตั้งโปรแกรม Java SDK คือ C:/jdk1.2.2 เราสามารถทำการเพิ่ม path ได้ดังนี้

สำหรับระบบปฏิบัติการ Windows NT:

เลือก Start → Control Panel เลือก System แล้วเลือก Environment ทำการเพิ่ม path โดยพิมพ์คำว่า Path ลงใน ช่อง Variables และพิมพ์ %Path%;C:/jdk1.2.2/bin ลงในช่อง Value แล้วกดปุ่ม Set จากนั้นให้กดปุ่ม OK

สำหรับระบบปฏิบัติการ Windows 95/98:

ทำการเปิดและแก้ไขไฟล์ชื่อ AUTOEXEC.BAT ทำการเพิ่มหรือแก้ไขตัวแปร PATH โดยทำการเพิ่มดังนี้

```
set path=C:/jdk1.2.2/bin
```

¹ Java SDK สามารถดาวน์โหลดได้ที่ <http://www.javasoft.com/jdk>

เมื่อดำเนินการเพิ่ม path เสร็จแล้ว ขั้นตอนต่อไปคือการติดตั้งโปรแกรมตัวแปลงภาษารูปแบบ XSL โดย

- ทำการกระจายไฟล์ชื่อ translator.jar ด้วยการรันคำสั่ง `jar xvfm translator.jar` ใน command prompt ของระบบปฏิบัติการ โดยต้องทำการรันในโฟลเดอร์ที่มีไฟล์ translator.jar อยู่
- เมื่อกระจายไฟล์เสร็จ จะได้โฟลเดอร์ที่ชื่อ translator ซึ่งเป็นไดเรกทอรีของตัวแปลงภาษารูปแบบ XSL ให้เข้าไปในโฟลเดอร์ translator แล้วทำการแก้ไขไฟล์ชื่อ start.bat โดยพิมพ์โฟลเดอร์ที่ติดตั้งโปรแกรม Java SDK เข้าไปในตัวแปลชื่อ JAVA_HOME ตัวอย่างเช่น

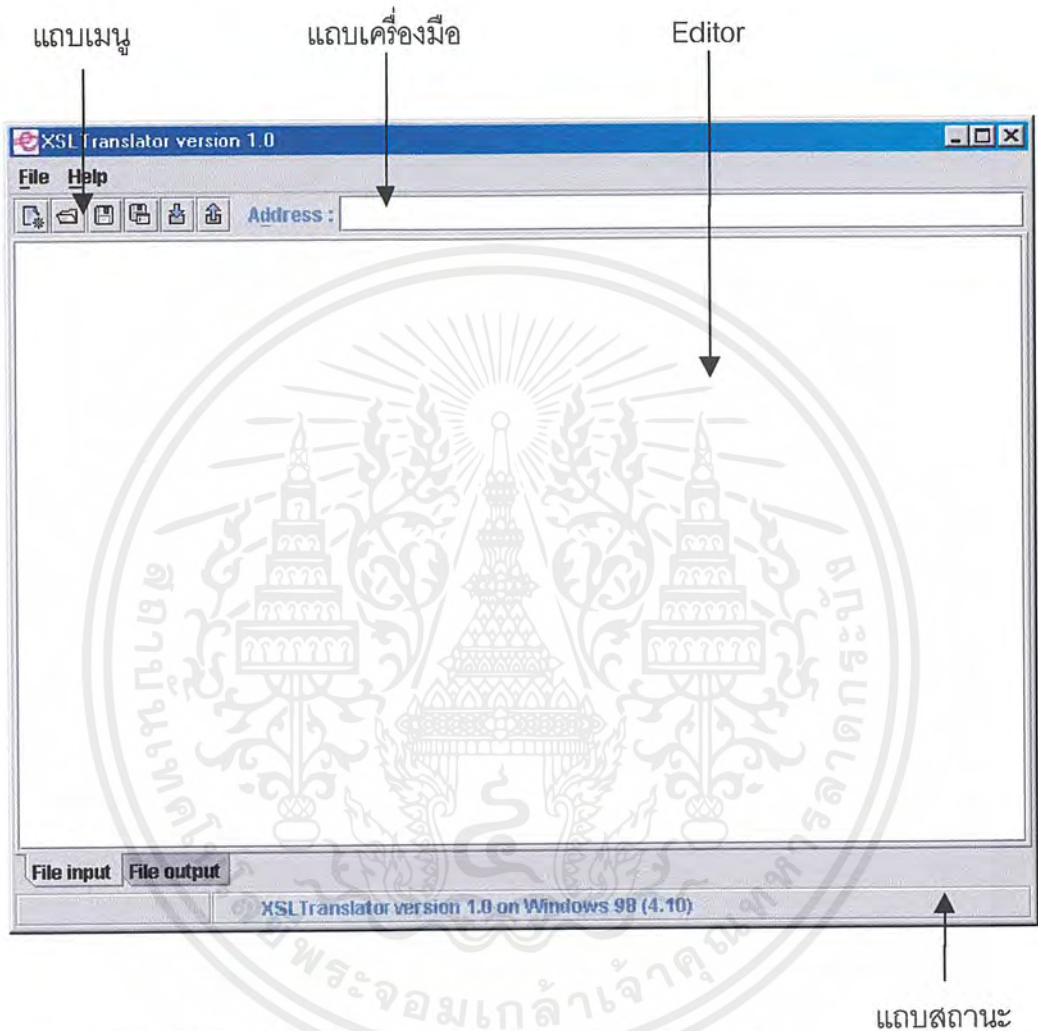
```
set JAVA_HOME=C:/jdk1.2.2
```

โดยที่ C:/jdk1.2.2 เป็นโฟลเดอร์ที่ติดตั้งโปรแกรม Java SDK



2. ส่วนต่าง ๆ ของโปรแกรมตัวแปลงภาษารูปแบบ XSL

การใช้งานโปรแกรมตัวแปลงภาษารูปแบบ XSL ทำได้โดย การรันไฟล์ชื่อ start.bat ที่อยู่ในโฟลเดอร์ของโปรแกรมตัวแปลงภาษารูปแบบ XSL เมื่อโปรแกรมเริ่มทำงาน จะพบกับหน้าจอหลักของโปรแกรม ดังรูปที่ ค-1



รูปที่ ค-1 แสดงหน้าจอหลักของโปรแกรมตัวแปลงภาษารูปแบบ XSL

หน้าจอของโปรแกรมตัวแปลงภาษารูปแบบ XSL แบ่งออกเป็นส่วนต่าง ๆ ดังนี้

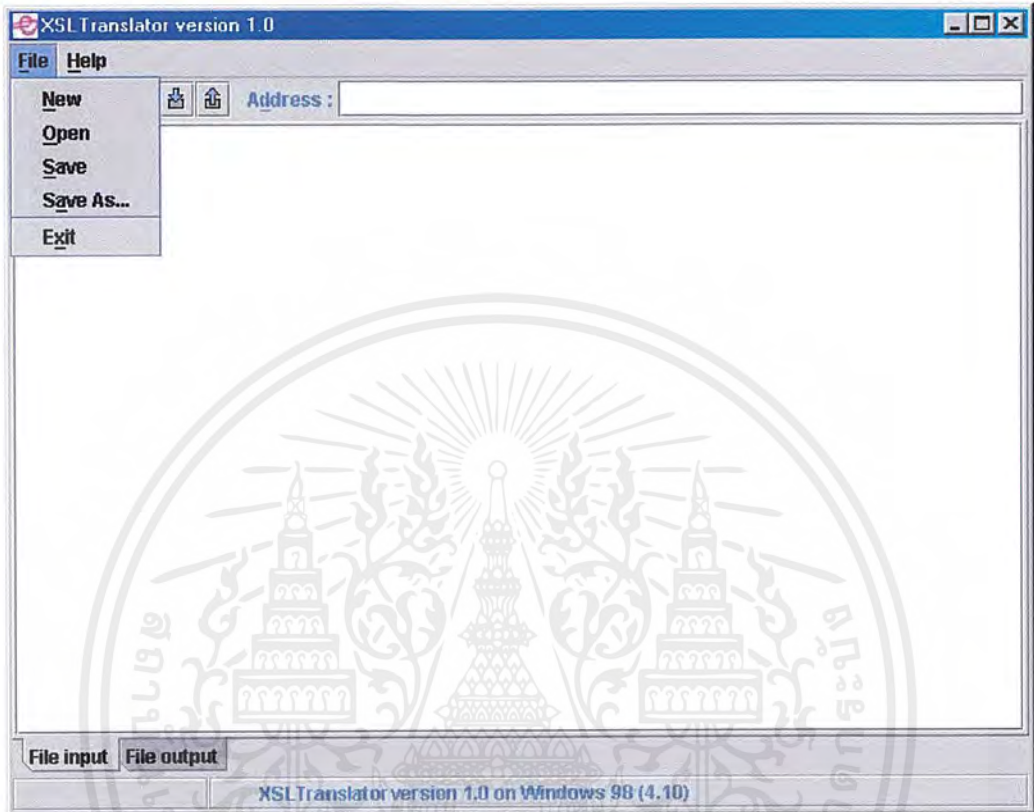
- 2.1 แถบเมนู
- 2.2 แถบเครื่องมือ
- 2.3 editor
- 2.4 แถบสถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 แถบเมนู

ตัวแปลภาษาแบบ XSL มีเมนูหลักอยู่ 2 เมนู คือ

2.1.1 เมนูหลัก File เป็นเมนูเกี่ยวกับการทำงานกับไฟล์ข้อมูล



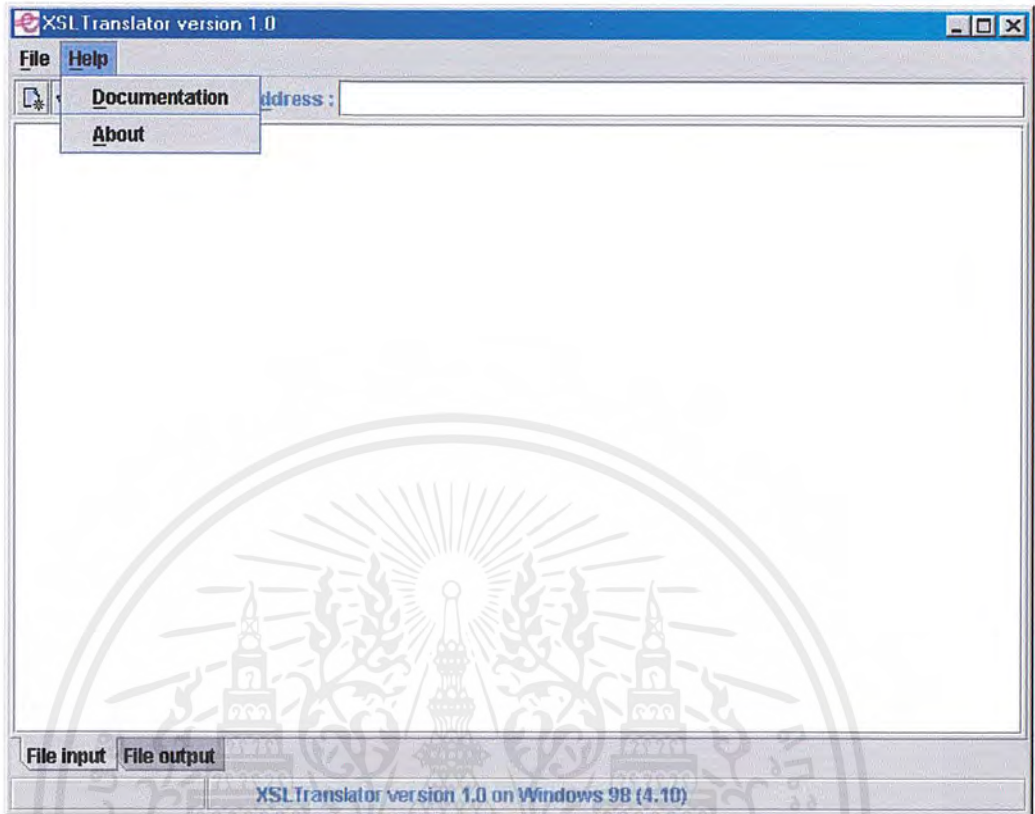
รูปที่ ค-2 แสดงหน้าจอหลักของโปรแกรมเมื่อเลือกเมนู File

เมนูหลัก File ประกอบด้วยเมนูย่อยต่าง ๆ ดังนี้

- New เป็นเมนูสำหรับสร้างเอกสารอินพุตขึ้นใหม่
- Open เป็นเมนูสำหรับอ่านเอกสารอินพุตที่สร้างไว้แล้วมาใช้งาน
- Save เป็นเมนูสำหรับบันทึกเอกสารอินพุตและเอกสารเอาต์พุต
- Save As เป็นเมนูสำหรับบันทึกเอกสารอินพุตและเอกสารเอาต์พุต เป็นชื่อใหม่
- Exit เป็นเมนูสำหรับออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 เมนูหลัก Help เป็นเมนูที่ให้ข้อมูลเกี่ยวกับตัวแปลงภาษารูปแบบ XSL



รูปที่ ค-3 แสดงหน้าจอของโปรแกรมเมื่อเลือกเมนู Help

เมนูหลัก Help ประกอบด้วยเมนูย่อยต่าง ๆ ดังนี้

- Documentation เป็นเมนูสำหรับดูเอกสารปัญหาพิเศษตัวแปลงภาษา XSL
- About เป็นเมนูสำหรับดูรายละเอียดโดยย่อของโปรแกรม







เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ค-4 แสดงหน้าจอของโปรแกรมเมื่อเลือกเมนู Help → About

2.2 แถบเครื่องมือ

แถบเครื่องมือของตัวแปลงภาษารูปแบบ XSL ประกอบด้วยไอคอนและส่วนต่าง ๆ ดังนี้

-  เป็นไอคอนสำหรับสร้างเอกสารอินพุตชิ้นใหม่
-  เป็นไอคอนสำหรับอ่านเอกสารอินพุตที่สร้างไว้แล้วมาใช้งาน
-  เป็นไอคอนสำหรับบันทึกเอกสารอินพุตและเอกสารเอาต์พุต
-  เป็นไอคอนสำหรับบันทึกเอกสารอินพุตและเอกสารเอาต์พุต เป็นชื่อใหม่
-  เป็นไอคอนสำหรับดูเอกสารอินพุตใน Browser
-  เป็นไอคอนสำหรับดูเอกสารเอาต์พุตใน Browser

ช่องข้อความ Address เป็นช่องข้อความที่แสดง path ของเอกสารอินพุต

เมื่อต้องการดูเอกสารอินพุตหรือเอาต์พุตใน Browser ถ้าเป็นการเรียกดูครั้งแรก โปรแกรมจะให้ระบุ path ไปสู่โปรแกรม Browser สำหรับให้โปรแกรมเรียกเพื่อแสดงเอกสารในโปรแกรม Browser ภายนอกแทน และถ้าเอกสารอินพุตเป็นเอกสาร HTML การดูเรียกเอกสารต้องระบุ path ไปยัง web browser แต่ถ้าเอกสารอินพุตเป็นเอกสาร WML การเรียกดูเอกสารต้องระบุ path ไปยัง wap browser

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 editor



รูปที่ ค-5 แสดงหน้าจอของโปรแกรมในส่วน editor

เป็นส่วนที่มีไว้สำหรับแก้ไขข้อมูลเพิ่มเติมในเอกสารอินพุต และส่วนที่แสดงข้อมูลเอกสารเอาต์พุตซ้อนกันอยู่ สามารถสลับดูข้อมูลโดยการกด tab สำหรับดูเอกสารอินพุตและเอกสารเอาต์พุต โดย tab **File input** เป็น tab สำหรับดูข้อมูลเอกสารอินพุตและ tab **File output** สำหรับดูข้อมูลเอกสารเอาต์พุต เมื่อกด tab นี้ ตัวแปลงภาษารูปแบบ XSL จะดำเนินการนำเอกสารอินพุตไปแปลงโครงสร้าง แล้วนำไปแสดงในช่องเอาต์พุต

2.4 แถบสถานะ

แถบสถานะของโปรแกรม มีสถานะของโปรแกรมว่าตอนนี้โปรแกรมตัวแปลงภาษารูปแบบ XSL กำลังทำงานอยู่บนระบบปฏิบัติการใด



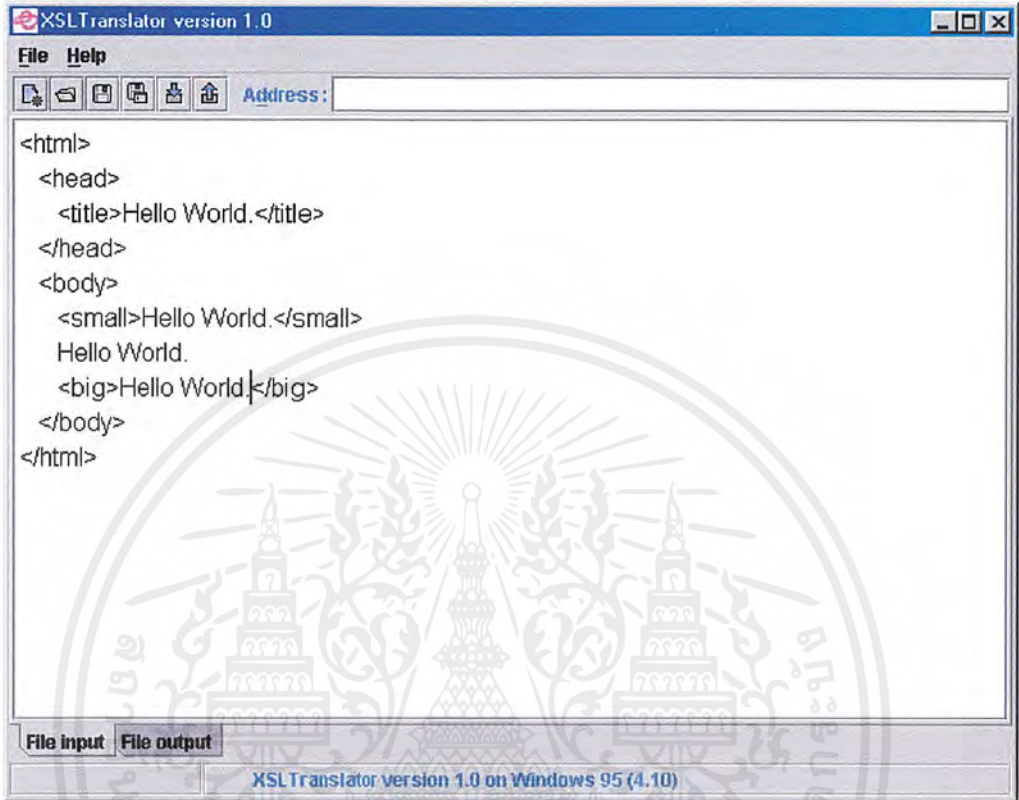
รูปที่ ค-6 แสดงแถบสถานะของโปรแกรมตัวแปลงภาษารูปแบบ XSL

จากรูปที่ ค-6 แถบสถานะแสดงให้เห็นทราบว่า โปรแกรมตัวแปลงภาษารูปแบบ XSL กำลังทำงานอยู่บนระบบปฏิบัติการ Windows98

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ตัวอย่างการใช้งานโปรแกรมตัวแปลงภาษารูปแบบ XSL

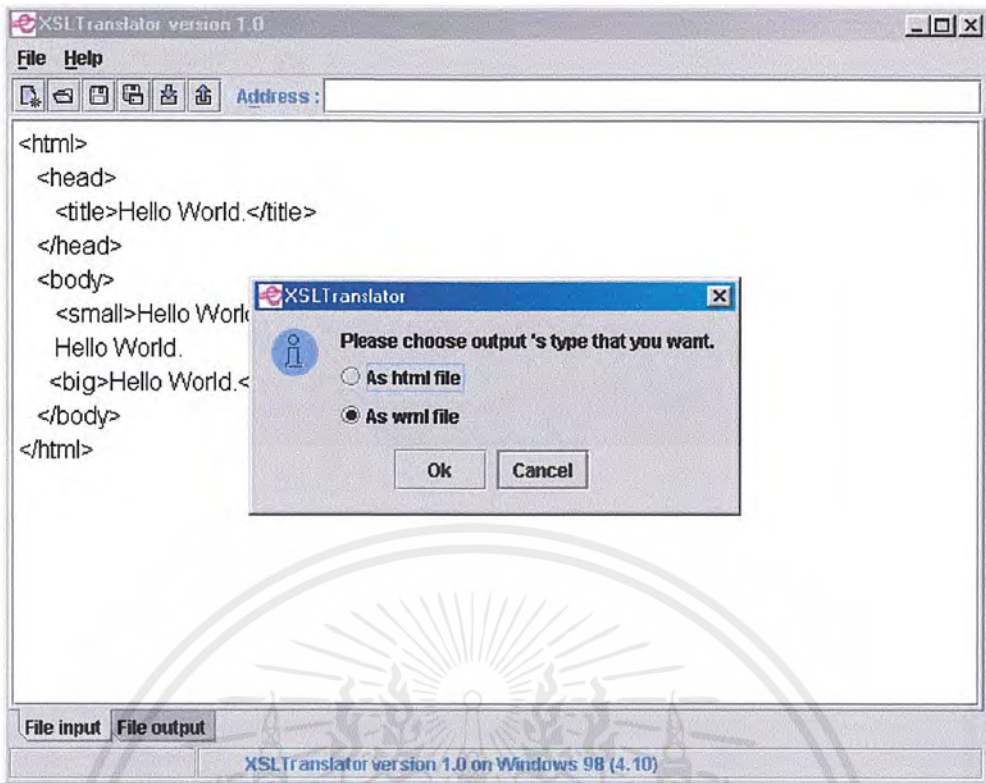
ทำการสร้างเอกสารอินพุตขึ้นมาใหม่แล้วทำการพิมพ์ข้อมูลเอกสารอินพุตลงในส่วน editor ของเอกสารอินพุตดังรูปที่ ค-7



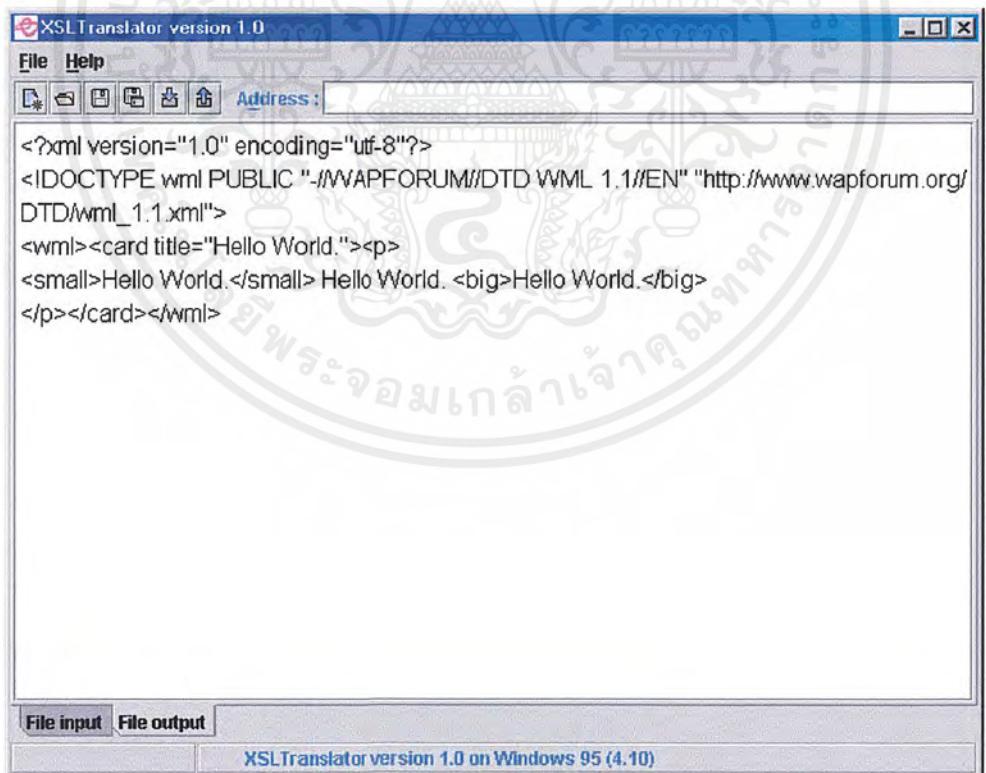
รูปที่ ค-7 แสดงตัวอย่างการแสดงผลเอกสารอินพุตของโปรแกรม

ทำการแปลงเอกสาร โดยการกด tab **File output** แล้วโปรแกรมจะถามว่า ต้องการแปลงเอกสารอินพุตให้เป็นเอกสารเอาต์พุตแบบใด ดังรูปที่ ค-8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




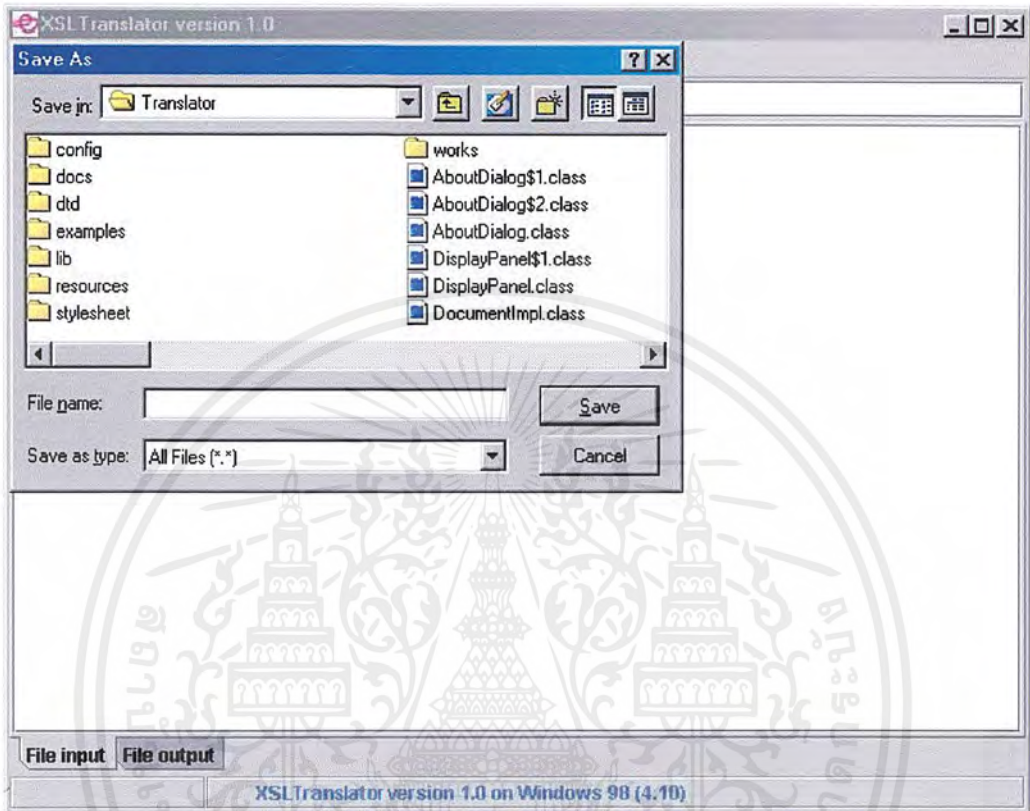
รูปที่ ค-8 แสดงหน้าจอสอบถามประเภทเอกสารเอชทีพีทูที



รูปที่ ค-9 แสดงตัวอย่างการแสดงผลเอกสารเอชทีพีทูทีหลังจากทำการดำเนินการของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

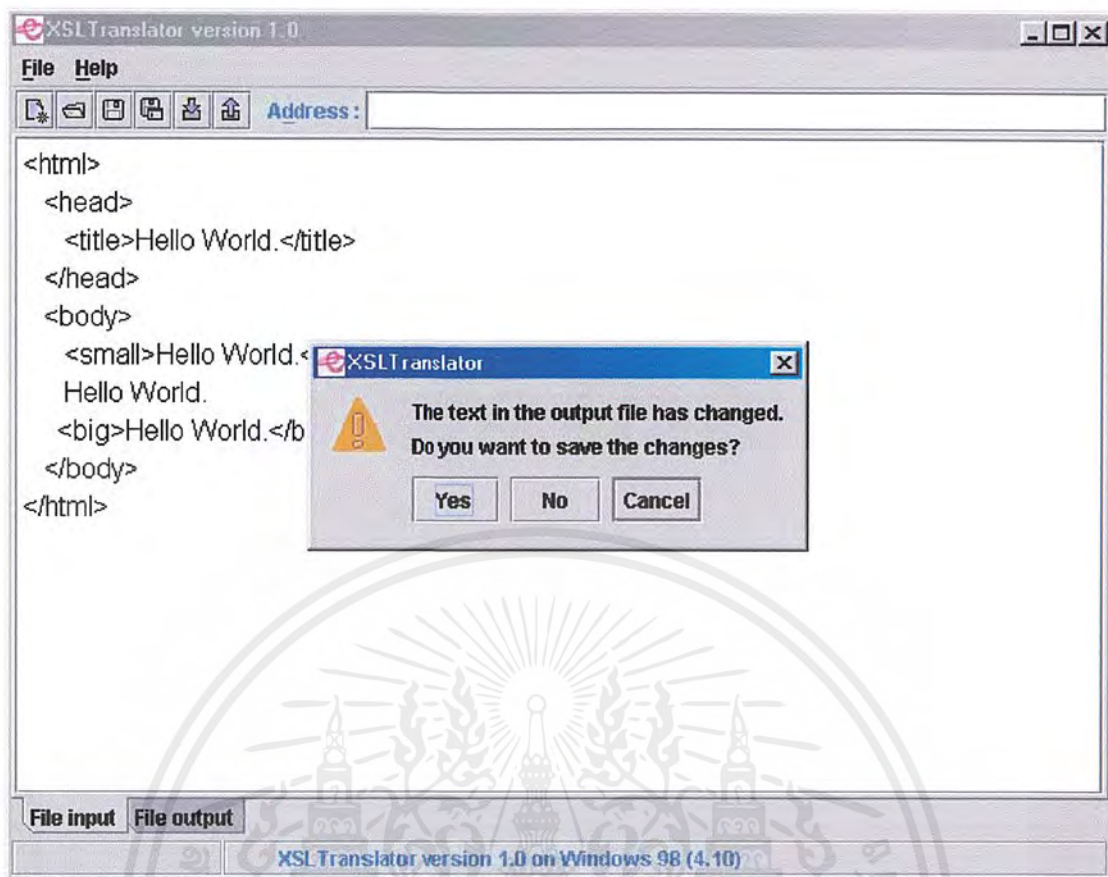
ทำการบันทึกเอกสารอินพุตและเอกสารเอาต์พุตโดยเลือกเมนู File>Save หรือกดที่ไอคอน  บนแถบเครื่องมือ โปรแกรมจะแสดงหน้าจอเพื่อให้ตั้งชื่อเอกสารอินพุต ดังรูปที่ ค-10 ให้ตั้งชื่อแล้ว กด Save เพื่อบันทึกไฟล์เอกสารอินพุต



รูปที่ ค-10 แสดงหน้าจอของโปรแกรมเพื่อให้ตั้งชื่อไฟล์เอกสาร

เมื่อบันทึกเอกสารอินพุตแล้ว โปรแกรมจะถามว่า ต้องการบันทึกเอกสารเอาต์พุตหรือไม่ โดยแสดงหน้าจอ ดังรูปที่ ค-11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

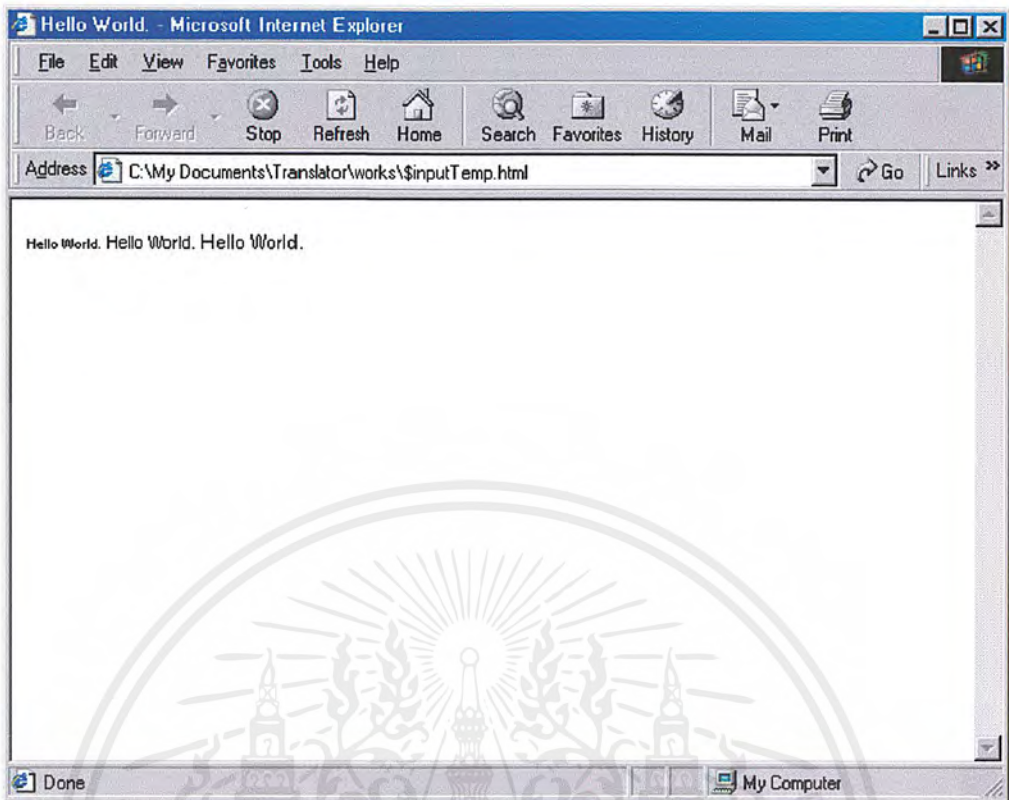


รูปที่ ค-11 แสดงหน้าจอของโปรแกรมเพื่อสอบถามความต้องการบันทึกเอกสารเอาต์พุต

ถ้าเลือก Yes โปรแกรมจะแสดงหน้าจอดังรูปที่ ค-10 เพื่อให้ตั้งชื่อไฟล์เอกสารเอาต์พุต ให้ทำการตั้งชื่อแล้วกด Save

เมื่อทำการบันทึกไฟล์เอกสารอินพุตและเอกสารเอาต์พุตเรียบร้อยแล้ว ทดลองดูผลที่ได้โดยทำการเปิดเอกสารอินพุตใน web browser และเปิดเอกสารเอาต์พุตใน wap browser ซึ่งได้ผลดังรูปที่ ค-12 และรูปที่ ค-13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ค-12 แสดงผลการเปิดเอกสารอินเทอร์เน็ตใน web browser



รูปที่ ค-13 แสดงผลการเปิดเอกสารเอชทีทีทีใน wap browser

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- วิลาศ วุวงศ์. 1999. "แนะนำ XML." Internet Magazine. 49-60.
- Anderson, R. et. al. Professional XML. Birmingham : Wrox. 2000.
- Eric Armstrong. " A Quick Introduction to XML. " [Online]. Available:
http://java.sun.com/xml/jaxp-docs-1.0.1/docs/tutorial/overview/1_xml.html. 2000
- Eric Armstrong. " Creating a Document Type Definition (DTD). " [Online]. Available:
http://java.sun.com/xml/jaxp-docs-1.0.1/docs/tutorial/sax/5a_dtd.html. 2000
- Eric Armstrong. " Defining Attributes and Entities in DTD. " [Online]. Available:
http://java.sun.com/xml/jaxp-docs-1.0.1/docs/tutorial/sax/5c_dtd.html. 2000
- Sun Microsystems, Inc. " Java™ 2 SDK, Standard Edition Documentation version 1.2.2 " [Online]. Available: <http://java.sun.com/products/jdk/1.2/docs/index.html>.
- Wireless Application Protocol Forum. " Wireless Application Protocol Wireless Markup Language Specification Version 1.1 " [Online]. Available:
<http://www1.wapforum.org/tech/documents/SPEC-WML-19990616.pdf>. 1999
- World Wide Web Consortium. " Extensible Stylesheet Language (XSL) Version 1.0 W3C Working Draft." [Online]. Available: <http://www.w3.org/TR/2000/WD-xsl-20000327>. 2000.
- World Wide Web Consortium. " XSL Transformations (XSLT) Version 1.0 W3C Recommendation." [Online]. Available: <http://www.w3.org/TR/1999/REC-xslt-19991116>. 1999.
- World Wide Web Consortium. " HTML 4.0 Specification W3C Recommendation, revised on 24-Apr-1998, World Wide Web Consortium. " [Online]. Available:
<http://www.w3.org/TR/1998/REC-html40-19980424>. 1998