



การออกแบบรูปแบบเอกสาร NUthesis ใน L^AT_EX สำหรับปริญญาโท
NUthesis L^AT_EX Class Design for Senior Project Report

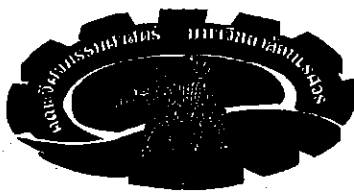


นางสาวนัตตา ฐูปหอม รหัสนิสิต 49364486
นางสาววันัสพร เจริญผล รหัสนิสิต 49364547

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ..... 11/๓.ค. 2555.....
เลขทะเบียน..... 16738350.....
เลขเรียกหนังสือ..... ๗๖.....
มหาวิทยาลัยนเรศวร ๖๓93๗

2562


ปริญญาโทเป็นส่วนหนึ่งของการศึกษาลัทธิปริญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
ปีการศึกษา 2552

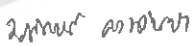


ใบรับรองโครงการวิศวกรรม

หัวข้อโครงการ การออกแบบรูปแบบเอกสาร NUthesis ใน L^AT_EX สำหรับปริญญาโท
 ผู้ดำเนินโครงการ นางสาวนัคตา รุประสิทธิ์ 49364486
 นางสาวนัศพร เจริญผล 49364547
 อาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.ชนิด มาลากร
 สาขาวิชา วิศวกรรมคอมพิวเตอร์
 ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์
 ปีการศึกษา 2552

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงการฉบับนี้เป็นส่วน
 หนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์
 คณะกรรมการสอบโครงการวิศวกรรม


ประธานกรรมการ
 (ผู้ช่วยศาสตราจารย์ ดร.ชนิด มาลากร)


กรรมการ
 (ดร.วรัถย์ คองเด่นฟ้า)

.....กรรมการ
 (อาจารย์ภาณุพงศ์ สอนคม)

หัวข้อโครงการ การออกแบบรูปแบบเอกสาร NUthesis ใน $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ สำหรับปริญญาโท
ผู้ดำเนินโครงการ นางสาวนัศดา ฐูปหอม รหัส 49364486
นางสาววันัสพร เจริญผล รหัส 49364547
อาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.ชนิต มาลากร
สาขาวิชา วิศวกรรมคอมพิวเตอร์
ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา 2552

บทคัดย่อ

การจัดพิมพ์เอกสารด้วย โปรแกรม $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ มีความสะดวกและรวดเร็วกว่าการพิมพ์งานใน Microsoft Word โดยเฉพาะอย่างยิ่งหากงานที่ต้องการจัดพิมพ์นั้นมีสมการและสูตรทางคณิตศาสตร์ร่วมอยู่ด้วย อย่างไรก็ตาม $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ยังคงมีจุดด้อยอยู่บ้าง อาทิ เช่น

1. ผู้ใช้งานต้องเสียเวลาในการเรียนรู้ถึงวิธีใช้งานก่อนที่จะสามารถใช้งานได้โดยมีประสิทธิภาพ แม้ว่า $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ เป็นภาษาที่เขียนได้โดยไม่ยาก
2. ผู้ใช้งานอาจต้องเสียเวลาในการหาจุดบกพร่องในเอกสารต้นฉบับ
3. ผู้ใช้งานอาจต้องอาศัยหนังสือคู่มือในการศึกษาถึงคำสั่งของ $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
4. ผู้ใช้งานอาจจำเป็นต้องใช้โปรแกรมตัดคำ เช่น SWATH หรือ CTTEX ในการพิมพ์งานภาษาไทย เนื่องจากคำในภาษาไทยไม่มีการเว้นช่องว่างระหว่างคำ

วัตถุประสงค์ในการจัดทำโครงการนี้คือการพัฒนารูปแบบเอกสารใน $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ชื่อว่า NUthesis Class รวมทั้งพัฒนาส่วนต่อประสานกราฟฟิคกับผู้ใช้ (GUI) ด้วยโปรแกรม VB.Net โดยมีวัตถุประสงค์เพื่อ

1. ช่วยเหลือนิสิตในการจัดเตรียมปริญญาโทให้ถูกต้องตามรูปแบบที่ภาควิชาฯ กำหนด
2. ช่วยผู้ใช้งานทั่วไปในการจัดเตรียมเอกสารเพื่อการนำเสนอ
3. สร้างช่องว่างระหว่างคำในภาษาไทยโดยการกดเพียงปุ่มเดียวแทนการพิมพ์คำสั่งของ SWATH ในหน้าต่างของ DOS

Project Title NUthesis L^AT_EX Class Design for Senior Project Report
Name Miss Nadda Thuphorn ID 49364486
Miss Wanussaporn Wanussaporn ID 49364547
Project Advisor Assist. Prof. Tanit Malakorn, Ph.D.
Major Computer Engineering
Department Electrical and Computer Engineering
Academic Year 2009

ABSTRACT

Writing documents with L^AT_EX can actually be much easier and faster than with MS Word, especially if mathematical equations and formulas are involved; however, L^AT_EX has some drawbacks, such as

1. users must put a little time into learning it before they can use it effectively, though L^AT_EX is not a difficult language;
2. users may need to spend some time debugging the code;
3. users may need a book to reference the L^AT_EX commands;
4. users may need some Thai Word Segmentation program, such as SWATH or CTTEX, for Thai type-setting since Thai does not have word boundary.

This project primarily concerns with the development of the L^AT_EX document class, namely NUthesis Class, as well as the Graphical User Interface (GUI) program using VB.Net framework for the following purposes:

1. to assist students in the preparation of the senior project reports in the right format,
2. to aid users for their presentation documents,
3. to make word boundaries by clicking at one button rather than writing the SWATH command in DOS window.

กิตติกรรมประกาศ

โครงการวิศวกรรมคอมพิวเตอร์ฉบับนี้สำเร็จลุล่วงมาได้นั้น เนื่องจากคณะผู้จัดทำได้รับความอนุเคราะห์จากผู้ช่วยศาสตราจารย์ ดร.ชนิต มาลากร ที่ท่านกรุณาคอยให้คำปรึกษา แนะนำวิธีการทำงาน พร้อมทั้งแนะแนวทางในการแก้ไขปัญหาต่าง ๆ และคอยกระตุ้นให้คณะผู้จัดทำทำงานอย่างต่อเนื่อง คณะผู้จัดทำจึงรู้สึกเป็นเกียรติอย่างมากที่ได้รับความอนุเคราะห์จากอาจารย์

ทั้งนี้ต้องขอขอบพระคุณอาจารย์สุคชาย บุญโต อาจารย์ประจำภาควิชาวิศวกรรมระบบควบคุมและเครื่องมือวัด มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี และคุณเอกวิทย์ กลิ่นกลิ่นหอม นักพัฒนา (Developer) บริษัท Callvoice Communications ที่เสียสละเวลาอันมีค่ามาให้คำปรึกษาและแนะแนวทางในการแก้ปัญหาต่าง ๆ

สุดท้ายนี้คณะผู้จัดทำต้องขอขอบพระคุณอาจารย์ทุกท่าน บิดา มารดา ที่คอยสั่งสอน ให้ความรู้ จนคณะผู้จัดทำสำเร็จการศึกษา และขอขอบคุณเพื่อน ๆ ที่คอยให้กำลังใจ ช่วยให้คำปรึกษาทั้งในเรื่องเรียน เรื่องส่วนตัวจนสำเร็จลุล่วงมาได้ด้วยดี

คณะผู้จัดทำ



สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	จ
สารบัญรูป	ฉ
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	2
1.3 ขอบเขตของโครงการ	2
1.4 แผนการดำเนินงาน	3
1.5 ผลที่คาดว่าจะได้รับ	3
1.6 งบประมาณ	3
บทที่ 2 การใช้งานโปรแกรม L ^A T _E X ภาษาไทย	4
2.1 การเตรียมเอกสารต้นฉบับ	4
2.2 การพิมพ์เอกสารด้วย L ^A T _E X ภาษาไทย	9
2.3 ตัวอย่างของเอกสารที่จัดพิมพ์ด้วย L ^A T _E X	12
2.4 ความแตกต่างระหว่าง Microsoft Office และ โปรแกรม L ^A T _E X	16
บทที่ 3 การพัฒนา NUthesis Class	18
3.1 เปรียบเทียบลักษณะเอกสารประเภทหนังสือและปริญญาบัตร	18
3.2 ขั้นตอนการพัฒนา NUthesis Class	21
บทที่ 4 โปรแกรม GUI และลำดับการทำงาน	29
4.1 แนวคิดในการออกแบบโปรแกรม GUI	29
4.2 กระบวนการสร้างไฟล์เอกสาร	30
4.3 การสร้างไฟล์เอกสารนำเสนอ	42
4.4 กระบวนการตัดคำ	45
บทที่ 5 สรุปผลการดำเนินงานและแนวทางการพัฒนา	46
5.1 สรุปผลการดำเนินงาน	46
5.2 แนวทางการพัฒนา	46
บรรณานุกรม	47
ประวัติผู้เขียนโครงการ	48

สารบัญตาราง

ตารางที่		หน้า
1.1	ตารางการดำเนินงาน	3
2.1	ความแตกต่างระหว่างการพิมพ์งานด้วย Microsoft Office และ L ^A T _E X	17
3.1	แสดงความแตกต่างระหว่างลักษณะของเอกสารที่ได้จากเอกสารประเภทหนังสือ และรูปแบบเอกสารปริิณยานิพนธ์	21



สารบัญรูป

รูปที่		หน้า
2.1	โครงสร้างมาตรฐานของเอกสารต้นฉบับ	7
2.2	เอกสารต้นฉบับประเภทหนังสือ (Book Class)	8
2.3	เอกสารต้นฉบับประเภทสื่อแนะนำเสนอผ่านเครื่องฉาย (Presentation)	9
2.4	แสดงหลักการทำงานของ L ^A T _E X	10
2.5	แสดงเอกสารต้นฉบับประเภทหนังสือก่อนการตัดคำ	12
2.6	แสดงเอกสารประเภทหนังสือที่ผ่านการประมวลผลด้วย L ^A T _E X	13
2.7	แสดงเอกสารต้นฉบับประเภทหนังสือหลังการตัดคำ	13
2.8	แสดงเอกสารหนังสือที่ได้จากเอกสารต้นฉบับที่มีการตัดคำ	14
2.9	แสดงเอกสารประเภทสื่อแนะนำเสนอก่อนการตัดคำ	14
2.10	แสดงเอกสารประเภทสื่อแนะนำเสนอที่ผ่านการประมวลผลด้วย L ^A T _E X	15
2.11	แสดงเอกสารต้นฉบับประเภทสื่อแนะนำเสนอหลังการตัดคำ	15
2.12	แสดงเอกสารประเภทสื่อแนะนำเสนอที่ได้จากเอกสารต้นฉบับที่มีการตัดคำ	16
3.1	เอกสารต้นฉบับประเภทหนังสือ	18
3.2	แสดงหน้าสารบัญในเอกสารประเภทหนังสือ	19
3.3	แสดงหน้าเนื้อหาในเอกสารประเภทหนังสือ	19
4.1	แผนผังการออกแบบโปรแกรม GUI	29
4.2	แสดงหน้าหลักของโปรแกรม GUI	30
4.3	แสดงหน้าต่างการเลือกไฟล์เคอร์เพื่อเก็บไฟล์	30
4.4	แสดงหน้าต่างการเลือกรูปแบบของเอกสาร	31
4.5	แสดงหน้าต่างการเลือกส่วนต่าง ๆ ในการทำเอกสารปริญูณานิพนธ์	31
4.6	แสดงหน้าภาษาไทย	32
4.7	แสดงหน้าบทคัดย่อ	33
4.8	แสดงหน้าภาษาอังกฤษ	33
4.9	แสดงหน้า Abstract	34
4.10	แสดงหน้ากิตติกรรมประกาศ	34
4.11	แสดงหน้ารูปภาพ	35
4.12	แสดงหน้าส่วนเนื้อหาของเอกสารปริญูณานิพนธ์	35
4.13	แสดงเนื้อหาของไฟล์หลัก Main.tex	36
4.14	แสดงหน้าส่วนหลังของเอกสารปริญูณานิพนธ์	37
4.15	แสดงหน้าบทความ	37
4.16	แสดงหน้าบรรณานุกรม	38
4.17	แสดงหน้าบรรณานุกรมประเภทวิทยานิพนธ์	38
4.18	แสดงไฟล์บรรณานุกรม	39
4.19	แสดงหน้าต่างของเอกสารทั่วไป	40

สารบัญรูป(ต่อ)

รูปที่		หน้า
4.20	แสดงหน้าการสร้างเอกสารทั่วไป	40
4.21	แสดงหน้าเอกสารแนะนำเสนอ	42
4.22	แสดงหน้าธีม (Theme) ของเอกสารแนะนำเสนอ	42
4.23	แสดงหน้าสำหรับเลือกสีของรูปแบบ	43
4.24	แสดงหน้าการใส่เนื้อหาของเอกสารแนะนำเสนอ	43
4.25	แสดงตัวอย่างหัวข้อหลักในเอกสารแนะนำเสนอ	44
4.26	แสดงตัวอย่างหัวข้อย่อยในเอกสารแนะนำเสนอ	45
4.27	แสดงหน้าทางการตัดคำ	45



บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

ศาสตราจารย์โดนัลด์ อี. นูธ (Donald E. Knuth) แห่งมหาวิทยาลัยสแตนฟอร์ด (Stanford University) ได้พัฒนาโปรแกรมเรียงพิมพ์ (Typesetting) ที่มีคุณภาพสูง เหมาะสำหรับการจัดทำเอกสารเชิงวิชาการที่อาศัยสูตรและสมการทางคณิตศาสตร์เป็นจำนวนมาก โดยเรียกโปรแกรมดังกล่าวนี้ว่า \TeX ซึ่งมีที่มาจากอักษรกรีก 3 ตัวที่นิยมนำมาใช้ในทางคณิตศาสตร์บ่อยครั้ง นั่นคือ τ , ϵ และ χ แม้ว่าโปรแกรม \TeX นี้จะมีความสามารถสูงในการเรียงพิมพ์เอกสาร แต่การใช้งานมีความยุ่งยากซับซ้อน เนื่องจากการเขียนคำสั่งที่ใช้ใน \TeX จัดเป็นการเขียนคำสั่งระดับล่าง (Low level command) ไม่เหมาะสำหรับผู้ใช้งานทั่วไปที่ไม่มีพื้นฐานเกี่ยวกับการเขียนโปรแกรม ด้วยเหตุนี้ เลสลีย์ แลมพอร์ท (Leslie Lamport) จึงได้พัฒนาโปรแกรม \LaTeX ขึ้น ซึ่งได้เพิ่มขีดความสามารถให้กับ \TeX พื้นฐานได้เป็นอย่างมาก อาทิ เช่น มีคำสั่งในการจัดรูปแบบเอกสารอัตโนมัติ การแบ่งบท การแบ่งหัวข้อ การสร้างสารบัญ การเรียงลำดับหัวข้อ การเรียงลำดับสมการ การอ้างอิง รวมไปถึงการแทรกรูปภาพ เป็นต้น

แม้ว่า \LaTeX ได้เพิ่มขีดความสามารถดังกล่าวข้างต้น แต่ผู้ใช้งานที่ต้องการพิมพ์เอกสารด้วย \LaTeX นั้น จำเป็นต้องจดจำคำสั่งของ \LaTeX ซึ่งต่างจากการพิมพ์เอกสารด้วย MS Word ของบริษัท Microsoft ที่มีเมนูให้เลือก หรือมีปุ่มให้กด ด้วยเหตุนี้ นายชันยา กูว์โรคม [2] จึงได้พัฒนาโปรแกรมส่วนต่อประสานกราฟฟิกกับผู้ใช้ (Graphical User Interface: GUI) ที่ซึ่งมีเมนูให้เลือกลักษณะตัวอักษร เลือกขนาดของตัวอักษร การจัดรูปแบบเอกสาร ระยะเวลาหน้า ระยะเวลาหลัง ระยะเวลาบน ระยะเวลาล่าง เพื่ออำนวยความสะดวกต่อผู้ใช้งานทั่วไป อีกทั้งยังปุ่มให้เลือกเพื่อใช้สร้างเอกสาร ในรูปแบบของปริญญาณิพนธ์อันเป็นประโยชน์สำหรับนิสิตภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ มหาวิทยาลัยนเรศวร ในการเรียงพิมพ์ปริญญาณิพนธ์ให้มีรูปแบบที่ถูกต้อง

อย่างไรก็ตาม ในงานของ [2] นั้นอาศัยเพียงการส่งพารามิเตอร์จากโปรแกรม GUI มาสร้างเป็นคำสั่งในไฟล์หลัก main.tex โดยอาศัย Document class มาตรฐาน กล่าวคือ Book Class จึงทำให้ในไฟล์ main.tex ดังกล่าวปรากฏคำสั่งต่าง ๆ มากมายเพื่อทำการปรับแต่งให้รูปแบบที่ได้จากการเรียงพิมพ์ด้วย Book Class มีรูปแบบถูกต้องตามรูปแบบที่ทางภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ มหาวิทยาลัยนเรศวร ได้กำหนดไว้ ดังนั้น ทางคณะผู้จัดทำโครงการนี้จึงได้พัฒนา Document class ขึ้น เรียกว่า Nuthesis Class เพื่อทำให้อเอกสารที่จัดพิมพ์ขึ้นภายใต้ Document Class นี้มีรูปแบบการเรียงพิมพ์ถูกต้องตามรูปแบบที่ทางภาควิชาฯ กำหนด นอกจากนี้ ยังได้พัฒนา GUI อีกส่วนขึ้นเพื่ออำนวยความสะดวกในการจัดพิมพ์เอกสาร ในรูปแบบของสื่อนำเสนอ (Presentation) อีกด้วย

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อมีความรู้ความเข้าใจในการจัดพิมพ์เอกสารด้วยโปรแกรม L^AT_EX
- 1.2.2 เพื่อพัฒนา NUthesis Class ด้วยโปรแกรม L^AT_EX ในการจัดพิมพ์ปริญญาานิพนธ์ให้ได้รูปแบบที่ถูกต้องตามหลักการเขียนปริญญาานิพนธ์สำหรับนิสิตภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
- 1.2.3 เพื่อศึกษาและพัฒนาส่วนต่อประสานกราฟฟิกกับผู้ใช้ (Graphic User Interface : GUI) ในการจัดพิมพ์ปริญญาานิพนธ์ เอกสารทางวิชาการต่าง ๆ รวมทั้งนำเสนอ (Presentation) เพื่อลดความยุ่งยากในการเขียนคำสั่งโดยตรงของ L^AT_EX

1.3 ขอบเขตของโครงการ

- 1.3.1 ออกแบบและพัฒนา NUthesis Class เพื่ออำนวยความสะดวกในการจัดพิมพ์ปริญญาานิพนธ์สำหรับนิสิตภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
- 1.3.2 ออกแบบโปรแกรมส่วนต่อประสานกราฟฟิกกับผู้ใช้ (GUI) โดยใช้โปรแกรม VB.NET เพื่ออำนวยความสะดวกต่อผู้ใช้งานทั่วไปในการสร้างเอกสารและนำเสนอ (Presentation)

1.4 แผนการดำเนินงาน

กิจกรรม	ปี 2552							ปี 2553	
	ม.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.
1.4.1 ศึกษาข้อมูลเกี่ยวกับโปรแกรม \LaTeX ให้ใช้งานภาษาไทยได้									
1.4.2 ออกแบบรูปแบบการเขียน ปรินตยูนิพจน์ด้วย \LaTeX									
1.4.3 พัฒนา NUthesis Class									
1.4.4 ออกแบบและพัฒนา GUI สำหรับการสร้างเอกสารและการทำ ลื่อนำเสนอด้วย VB.Net									
1.4.5 ทดสอบและวิเคราะห์การ ทดสอบ									
1.4.6 ปรับปรุงและแก้ไข โปรแกรม \LaTeX และจัดทำแผ่นซีดี (CD)									
1.4.7 จัดทำรูปเล่มรายงาน									

ตารางที่ 1.1: ตารางการดำเนินงาน

1.5 ผลที่คาดว่าจะได้รับ

- 1.5.1 ได้แผ่นซีดีสำหรับการลง โปรแกรม \LaTeX รวมทั้งส่วนต่อประสานกราฟฟิคกับผู้ใช้ (GUI) ที่ช่วยอำนวยความสะดวกต่อนิสิตในการจัดพิมพ์งานปรินตยูนิพจน์ให้ได้รูปแบบที่ถูกต้อง ตามหลักการเขียนปรินตยูนิพจน์สำหรับนิสิตภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะ วิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

1.6 งบประมาณ

ค่าหนังสือ	500 บาท
ค่าวัสดุสำนักงาน	500 บาท
ค่าวัสดุคอมพิวเตอร์	500 บาท
ค่าถ่ายเอกสารและเข้าเล่มวิทยานิพนธ์	500 บาท
รวมเป็นเงินทั้งสิ้น	2000 บาท (สองพันบาทถ้วน)

หมายเหตุ ถัวเฉลี่ยทุกรายการ

บทที่ 2

การใช้งานโปรแกรม L^AT_EX ภาษาไทย

ในบทนี้กล่าวถึงการใช้งานทั่วไปของโปรแกรม L^AT_EX ทั้งการเตรียมเอกสารต้นฉบับ การพิมพ์เอกสารด้วย L^AT_EX ภาษาไทย ตัวอย่างเอกสารที่จัดพิมพ์ด้วย L^AT_EX และเปรียบเทียบความแตกต่างระหว่างการใช้งานด้วยโปรแกรมของ Microsoft และ L^AT_EX โดยมีรายละเอียด ดังนี้

2.1 การเตรียมเอกสารต้นฉบับ

ในการเตรียมเอกสารต้นฉบับเพื่อให้โปรแกรม L^AT_EX ประมวลผลนั้น ผู้ใช้งานสามารถจัดพิมพ์เอกสารด้วยโปรแกรมที่ใช้พิมพ์เอกสารทั่วไป เช่น Notepad, Winpad, หรือ T_EXnicCenter แล้วทำการบันทึก (Save) ให้ไฟล์ที่จัดพิมพ์ขึ้นนั้นต้องมีนามสกุล .tex โดยการจัดพิมพ์เนื้อหาในไฟล์ดังกล่าวต้องมีโครงสร้างตามที่ L^AT_EX กำหนด

โดยทั่วไป โครงสร้างของเอกสาร L^AT_EX แบ่งออกเป็น 2 ส่วนใหญ่ คือ ส่วนอาร์มภบท (Preamble) และส่วนเนื้อหา (Body) โดยในส่วนอาร์มภบทจะมีการกำหนดลักษณะโดยรวมของเอกสารที่ต้องการเรียงพิมพ์ ในขณะที่รายละเอียดของเนื้อหา เช่น สารบัญ สารบัญรูป กิตติกรรมประกาศ เนื้อหา ภาคผนวก และบรรณานุกรม จะใส่ไว้ในส่วนของเนื้อหา

ส่วนอาร์มภบท ถือได้ว่าเป็นส่วนที่กำหนดลักษณะของเอกสาร มีการประกาศเลือกใช้ประเภทของเอกสาร (Document Class) การประกาศใช้แพ็คเกจ (Package) ที่จำเป็น การกำหนดขนาด (Font Size) ชนิดของตัวอักษร (Font Style) การกำหนดขนาดกระดาษ การจัดหน้ากระดาษ และการกำหนดคำสั่งพิเศษ เป็นต้น คำพารามิเตอร์ต่าง ๆ ที่กำหนดไว้ในส่วนนี้จะมีค่าเป็นไปตามที่ระบุจนกว่าจะมีการประกาศเพิ่มเติมให้เป็นอย่างอื่น

1. ประเภทของเอกสาร (Document Class)

โดยทั่วไปโปรแกรม L^AT_EX มี Class มาตรฐานอยู่ 5 ประเภท โดยมีชื่อและรายละเอียด ดังนี้

- Article เหมาะสำหรับการจัดทำเอกสารประเภทบทความ, เอกสารประกอบการบรรยาย และรายงานฉบับสั้น
- Book เหมาะสำหรับการจัดทำหนังสือที่มีหลายบทและมีลักษณะการเรียงพิมพ์แบบสองหน้า
- Letter เหมาะสำหรับการเขียนจดหมาย
- Report เหมาะสำหรับการจัดทำรายงานที่มีจำนวนบทไม่มากนักหรือวิทยานิพนธ์ที่มีลักษณะการเรียงพิมพ์แบบหน้าเดียว
- Slide เหมาะสำหรับการจัดทำเอกสารนำเสนอ

2. การกำหนดขนาดตัวอักษร (Font Size)

โปรแกรมมีค่าการกำหนดขนาดของตัวอักษรให้เลือกใช้งานอยู่ 3 ขนาด คือ 10pt, 11pt และ 12pt หากผู้ใช้ไม่ได้กำหนดค่า โปรแกรมจะทำการกำหนดขนาดตัวอักษรเริ่มต้น (Default Value) คือ 10 pt

3. การกำหนดขนาดกระดาษ (Paper Size)

โปรแกรมมีการกำหนดขนาดของกระดาษให้เลือกใช้งานได้อยู่ 6 ขนาด คือ

Letterpaper (11 x 8.5 in)	Legalpaper (14 x 8.5 in)
Executivpaper (10.5 x 7.25 in)	a4paper (29.7 x 21.0 cm)
a5paper (21 x 14.8 cm)	b5paper (25 x 17.6 cm)

หากผู้ใช้ไม่ได้กำหนดค่า โปรแกรมจะทำการกำหนดขนาดกระดาษเริ่มต้น (Default Value) คือ Letterpaper

4. แบบหน้ากระดาษ (Page Style)

LaTeX ได้กำหนดแบบหน้ากระดาษ (Page Style) ที่ประกอบด้วยหัวกระดาษ (Header) และท้ายกระดาษ (Footer) ซึ่งแบบหน้ากระดาษมีตัวเลือกอยู่ 4 ลักษณะ ดังนี้

- หน้ากระดาษแบบ Plain หมายเลขหน้าอยู่ตรงกลางด้านล่างของกระดาษ ถ้าผู้ใช้ไม่กำหนดแบบหน้ากระดาษแก่เอกสาร LaTeX จะกำหนดให้ใช้แบบ Plain นี้เป็นค่าเริ่มต้น
- หน้ากระดาษแบบ Headings มีชื่อทและหมายเลขหน้าอยู่ส่วนหัวกระดาษของทุกหน้า โดยที่ส่วนท้ายกระดาษจะว่างเปล่า
- หน้ากระดาษแบบ Myheadings มีลักษณะคล้ายกับหน้ากระดาษแบบ Heading เพียงแต่ผู้ใช้งานต้องกำหนดตัวเลือก (Option) ด้วยคำสั่ง `\markright` สำหรับเอกสารที่ถูกสร้างขึ้นด้วยตัวเลือก Oneside หรือคำสั่ง `\markboth` สำหรับเอกสารที่ถูกสร้างขึ้นด้วยตัวเลือก Twoside
- หน้ากระดาษแบบ Empty ทั้งส่วนหัวและส่วนท้ายของหน้ากระดาษปล่อยว่าง

5. การจัดหน้ากระดาษ (Page Formats)

การจัดหน้ากระดาษสามารถเลือกได้ 4 กลุ่มใหญ่ โดยแต่ละกลุ่มจะมีค่าเริ่มต้นขึ้นอยู่กับชนิดของเอกสารว่าเป็นบทความ (Article), หนังสือ (Book), จดหมาย (Letter) หรือรายงาน (Report)

- Onecolumn หรือ Twocolumn

ใช้เลือกจัดหน้ากระดาษให้เป็นแบบ 1 หรือ 2 คอลัมน์ หากผู้ใช้ไม่กำหนด ค่าเริ่มต้นที่โปรแกรมจะกำหนดให้คือแบบ 1 คอลัมน์ สำหรับกรณี que เลือกใช้แบบ 2 คอลัมน์ ผู้ใช้สามารถกำหนดความกว้างของระยะห่างระหว่างคอลัมน์และขนาดของเส้นแบ่งคอลัมน์ด้วยการใช้คำสั่ง `\columnsep` และ `\columnseprule`

- Oneside หรือ Twoside

ใช้เลือกลักษณะของเอกสารว่าเป็นแบบพิมพ์หน้าเดียว (Oneside) ซึ่งมีลักษณะ

การจัดพิมพ์เหมือนกันในทุกหน้า หรือแบบพิมพ์ 2 หน้า (Twoside) ซึ่งมีผลให้เอกสารสำหรับพิมพ์กระดาษหน้าซ้าย (หมายเลขหน้าเป็นเลขคู่) และเอกสารสำหรับพิมพ์กระดาษหน้าขวา (หมายเลขหน้าเป็นเลขคี่) มีลักษณะไม่เหมือนกัน กล่าวคือ ในส่วนหัวกระดาษ เลขหน้าสำหรับหน้าซ้ายจะพิมพ์ชิดขอบกระดาษด้านซ้าย และสำหรับหน้าขวาจะพิมพ์ชิดขอบกระดาษด้านขวา ทำให้สะดวกในการพลิกหาหมายเลขหน้าเมื่อเอกสารเสร็จเป็นรูปเล่มแล้ว

- สำหรับเอกสารประเภท Article Class และ Report Class ค่าเริ่มต้น คือ Oneside
- สำหรับเอกสารประเภท Book Class ค่าเริ่มต้น คือ Twoside

- Openright หรือ Openany

ในกรณีที่ใช้ Book Class สำหรับพิมพ์หนังสือ ตัวเลือกนี้จะใช้กำหนดลักษณะของหนังสือว่าเริ่มต้นบทใหม่ที่หน้าด้านขวาเสมอ หรือเริ่มต้นบทใหม่ที่หน้าด้านใดก็ได้ ในกรณีที่เลือกแบบ Openright L^AT_EX อาจพิจารณาเพิ่มเติมหน้ากระดาษเปล่าไว้ท้ายบทก่อนเริ่มต้นบทใหม่ในหน้าด้านขวา (หน้าหมายเลขคี่) ให้ตามความเหมาะสม

- Notitlepage หรือ Titlepage

โดยทั่วไปชื่อ (Title) ของเอกสารประเภทหนังสือและรายงานจะถูกพิมพ์แยกจากส่วนเนื้อหาของเอกสาร ในลักษณะเป็นแผ่นหน้าปก (Title Page) ในขณะที่เอกสารประเภทบทความจะพิมพ์ไว้ที่ส่วนหัวของเอกสาร โดยไม่แยกออกมา ดังนั้นหากต้องการให้เอกสารมีแผ่นหน้าปกหรือไม่มีแผ่นหน้าปก ผู้ใช้สามารถกำหนดได้ด้วยตัวเลือกนี้

6. ตัวเลือกอื่น ๆ

ตัวเลือกอื่น ๆ ที่สามารถเลือกได้อีก ประกอบด้วย

- Leqno โดยปกติหมายเลขลำดับของสมการคณิตศาสตร์จะถูกพิมพ์ไว้ทางด้านขวาของสมการ คำสั่ง Leqno มีผลให้หมายเลขลำดับของสมการถูกย้าย ไปอยู่ด้านซ้ายของสมการแทน
- Fleqn โดยปกติสมการคณิตศาสตร์จะถูกพิมพ์ไว้กึ่งกลางบรรทัด (Center) คำสั่ง Fleqn มีผลให้สมการคณิตศาสตร์ถูกจัดไว้ชิดขอบด้านซ้าย โดยสามารถกำหนดขนาดของการย่อหน้าด้วยการกำหนดขนาดให้กับพารามิเตอร์ `\mathindent`
- Openbib ใช้กำหนดลักษณะการพิมพ์บรรณานุกรม โดยให้พิมพ์แยกข้อมูลส่วนต่าง ๆ ของแต่ละรายการแยกเป็นบรรทัด ในขณะที่ค่าเริ่มต้นจะพิมพ์ข้อมูลทั้งหมดของแต่ละรายการต่อเนื่องกันไป
- Draft ใช้เพื่ออำนวยความสะดวกในการตรวจทานการเรียงพิมพ์ก่อนผลงานสุดท้าย ในกรณีที่การตัดคำเกิดขึ้นอย่างไม่สมบูรณ์ ทำให้มีข้อความบางส่วนถูกพิมพ์ล้นขอบกระดาษด้านขวา (Right Margin) จะมีผลให้ L^AT_EX พิมพ์แถบสีค้ำควดไว้เพื่อเตือนความจำให้แก้ไขให้เรียบร้อยก่อนส่งต้นฉบับไปจัดพิมพ์
- Final ตัวเลือกนี้ทำหน้าที่ตรงกันข้ามกับตัวเลือก Draft กล่าวคือจะไม่แสดงแถบสีค้ำควดให้เห็น

ไม่ว่าจะเกิดความผิดพลาดหรือไม่ก็ตาม

ส่วนเนื้อหา เป็นส่วนที่ใช้สำหรับการพิมพ์เนื้อหาส่วนต่าง ๆ แยกออกเป็นภาค (Part) บท (Chapter) ตอน (Section) ตอนย่อย (Subsection) ย่อหน้า (Paragraph) และย่อหน้าย่อย (Subparagraph) ทั้งนี้การขึ้นย่อหน้าใหม่ การใส่ภาพประกอบ ตาราง จะทำในส่วนนี้โดยรายละเอียดทั้งหมดในส่วนตัวเอกสาร ต้องพิมพ์อยู่ระหว่างคำสั่ง `\begin{document}` และปิดท้ายด้วยคำสั่ง `\end{document}` การจัดรูปแบบเอกสารนั้นประกอบไปด้วย 3 ส่วนหลัก คือ ส่วนหน้า (Front Matter) ส่วนเนื้อหาหลัก (Main Matter) และส่วนหลัง (Back Matter) ซึ่งแต่ละส่วนมีรายละเอียด ดังนี้

1. ส่วนหน้า (Front Matter) โดยทั่วไปส่วนหน้าจะประกอบไปด้วยบทคัดย่อ กิตติกรรมประกาศ สารบัญ สารบัญตารางและสารบัญรูป เป็นต้น
2. ส่วนเนื้อหาหลัก (Main Matter) คือส่วนที่เป็นเนื้อหาทั้งหมด อันประกอบไปด้วยบทหรือหัวข้อต่าง ๆ ในเอกสาร
3. ส่วนหลัง (Back Matter) เป็นส่วนที่อยู่ถัดจากเนื้อหาหลัก ซึ่งประกอบด้วยบรรณานุกรมและภาคผนวก เป็นต้น

โครงสร้างมาตรฐานของเอกสารต้นฉบับที่ใช้โปรแกรม $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ประมวลผลเป็นคังรูปที่ 2.1

```
\documentclass[Option]{Class}
```

```
\usepackage[Option]{Package}
```

```
...
```

```
...
```

```
\begin{document}
```

```
\frontmatter (ส่วนหน้า)
```

```
...
```

```
...
```

```
\mainmatter (ส่วนเนื้อหาหลัก)
```

```
...
```

```
...
```

```
\backmatter (ส่วนหลัง)
```

```
...
```

```
...
```

```
\end{document}
```

ส่วนเจารัมกับท

Preamble

ส่วนเนื้อหา

รูปที่ 2.1: โครงสร้างมาตรฐานของเอกสารต้นฉบับ

2.1.1 เอกสารประเภทสิ่งพิมพ์ทั่วไป (Printed Matter)

เอกสารประเภทนี้ผู้ใช้งานสามารถเลือกประเภทของเอกสารเป็นแบบ Article Class, Book Class หรือ Report Class ทั้งนี้ขึ้นกับลักษณะรูปแบบของสิ่งพิมพ์นั้น ๆ สำหรับตัวอย่างของเอกสารที่ใช้ Book Class แสดงไว้ในรูปที่ 2.2

```

\documentclass{book}
\usepackage[english,thai]{babel}
\usepackage{thswitch}
\begin{document}
\tableofcontents
\chapter{บทนำ}
\section{บทนำและทฤษฎีเบื้องต้น}
\TeX คือ โปรแกรมคอมพิวเตอร์ที่สร้างงานโดยคำสั่งภาษา TeX โดย Donald E. Knuth.
เมื่อใช้ TeX ในการเขียนหนังสือหรือบทความในคอมพิวเตอร์ TeX ใช้ชื่อต้นฉบับโดย Leslie Lamport)
โดย TeX ใช้ชื่อ TeX และชื่อ TeX เป็นชื่อโปรแกรมหลัก TeX
จึงถือว่าเป็นโปรแกรมสำเร็จรูปที่ช่วยอำนวยความสะดวกต่าง ๆ เช่น การใส่คำสั่งเอกสารที่ถูกต้องตาม
มาตรฐานในการนำเอกสารไปพิมพ์ออกด้วยโปรแกรม TeX File ซึ่งเป็นโปรแกรมที่
สามารถนำไปประมวลผลด้วย TeX ในระบบปฏิบัติการหลายประเภท เช่น Linux, Windows, Unix เป็นต้น โปรแกรม
TeX เป็นโปรแกรมที่ Open Source สามารถดาวน์โหลดได้จากเว็บไซต์โดยที่ชื่อคือ "TeX" ได้
เนื่องจากโครงการนี้เป็นที่นิยมในวงกว้างจึงได้มีการออกแบบระบบเอกสารสำหรับโปรแกรม TeX ภาาไทย
ซึ่งผลงานที่ได้เป็นแบบ CD) เมื่อใช้ TeX ลงโปรแกรม TeX ภาษาไทย รวมถึงโปรแกรมต่าง ๆ
ที่จำเป็นต่อการใช้งานในการใช้ TeX แต่ละโปรแกรมส่วนประกอบส่วนประกอบที่ GUI
ที่ใช้ในการสร้างรูปแบบของเอกสารและการผลิตภาษาไทย
\end{document}

```

รูปที่ 2.2: เอกสารต้นฉบับประเภทหนังสือ (Book Class)

2.1.2 เอกสารประเภทสื่อนำเสนอผ่านเครื่องฉาย (Presentation)

ในช่วงแรก LaTeX สามารถเรียงพิมพ์เอกสารประเภทสื่อนำเสนอผ่านทางเครื่องฉายด้วยการเลือกประเภทของเอกสารแบบ Slide Class โดยมุ่งเน้นเอกสารที่ต้องการแสดงสมการทางคณิตศาสตร์เป็นหลัก ต่อมาได้มีผู้พัฒนาประเภทของเอกสารรูปแบบใหม่ คือ Prosper Class และ Beamer Class เพื่อเพิ่มขีดความสามารถในการนำเสนอมากยิ่งขึ้น กล่าวคือสามารถเปลี่ยนธีม (Theme) เปลี่ยนสีพื้นหลังหรือสีของตัวอักษร มีการเปลี่ยนหน้าของเอกสารไปยังหน้าที่ต้องการได้

สำหรับตัวอย่างของเอกสารที่ใช้ Beamer Class แสดงไว้ดังรูปที่ 2.3

```

\documentclass[compress,red,9pt]{beamer}
\usepackage[english,thai]{babel}
\usepackage{thswtch}
\usepackage{beamerthemesplit}
\def\thairmdeFauIt{uangdana}
\usetheme{Berlin}

\begin{document}
\begin{frame}
\titlepage
\end{frame}
\section{main}{}
\frame{\tableofcontents}
\section{Introduction}
\subsection{แนวคิดในการออกแบบ Nuthesis Class}
\frame{\frametitle{แนวคิดในการออกแบบ Nuthesis Class}
เนื้อหาเกี่ยวกับความสำคัญในการจัดการระบบการเรียนการสอนของนักศึกษาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี โดยที่เนื้อหาในหนังสือจะใช้เวลาในการจัดรูปแบบของเอกสาร
และเนื้อหาในโปรแกรมที่ประกอบขึ้นเพื่อเตรียมเอกสารและเนื้อหาในโปรแกรมที่ประกอบขึ้นเพื่อเตรียมเอกสาร
รูปแบบของเอกสารจะปรับการเปลี่ยนแปลงหรือการแก้ไขเนื้อหาได้อย่างใด
}
\end{document}

```

รูปที่ 2.3: เอกสารต้นฉบับประเภทสื่อนำเสนอผ่านเครื่องฉาย (Presentation)

เนื่องจากรูปแบบของ Beamer ใช้สำหรับการเรียงพิมพ์เอกสารที่ใช้ในการนำเสนอ ดังนั้นจึงมีลักษณะพิเศษเพิ่มเติมที่แตกต่างจากเอกสารประเภทสิ่งพิมพ์ทั่วไป กล่าวคือ มีธีม (Theme) ให้เลือก ยกตัวอย่าง เช่น

```

\usepackage{beamerthemesplit}
\usetheme{Berlin}

```

ตัวอย่างของรายชื่อธีมที่มีอยู่ใน LaTeX

Antibes	Bergen	Berkeley	Berlin
Boadilla	Cambridgeus	Classic	Darmstadt
Goettingen	Hannover	Juanlespins	Lined
Marburg	Shadow	Szeged	Tree

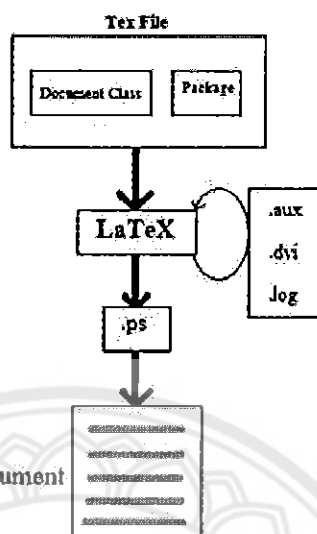
โดยที่ไฟล์ของธีม (Theme) ต้องเป็นไฟล์ที่มีนามสกุล .sty

2.2 การพิมพ์เอกสารด้วย LaTeX ภาษาไทย

เมื่อผู้ใช้ได้จัดทำเอกสารต้นฉบับเป็นที่เรียบร้อยแล้ว สิ่งที่ต้องทำถัดไป คือ การประมวลผล (Compile) ด้วยโปรแกรม LaTeX โดยมีรายละเอียดของขั้นตอนการทำงาน ดังนี้

2.2.1 หลักการทำงานทั่วไปของ LaTeX

ขั้นตอนการทำงานโดยทั่วไปของ LaTeX แสดงไว้ดังรูปที่ 2.4



รูปที่ 2.4: แสดงหลักการทำงานของ IAT_EX

จากรูปที่ 2.4 พบว่าเมื่อนำเอกสารต้นฉบับมาผ่านการประมวลผลด้วย IAT_EX แล้ว จะได้ไฟล์ที่สำคัญออกมา 3 ไฟล์ คือ .aux, .dvi และ .log โดยที่

- ไฟล์ .aux เป็นไฟล์สนับสนุนที่เขียนโดย IAT_EX เพื่อใช้ในการเก็บข้อมูลจำพวกคำสั่งที่มีความสำคัญในการจัดพิมพ์เอกสาร ไฟล์ .aux จะมีเพียงไฟล์เดียวต่อไฟล์ต้นฉบับ 1 ไฟล์ โดยไฟล์ .aux จะถูกสร้างใหม่ขึ้นมาทุก ๆ ครั้งในการประมวลผล
- ไฟล์ .dvi มาเป็นที่เรียบร้อยแล้ว ถ้าต้องการแสดงรูปแบบเอกสารที่ได้ผ่านทางหน้าจอ ให้เรียกใช้โปรแกรม Yap ซึ่งเป็น โปรแกรมที่ติดตั้งมาพร้อมกับ IAT_EX เป็นตัวแสดงผล หลังจากที่เราตรวจทานเอกสารที่ได้สร้างขึ้นมาเป็นที่เรียบร้อยแล้ว ถ้าผู้ใช้ต้องการแปลงไฟล์ .dvi เป็นไฟล์แสดงผลประเภทอื่น เช่น ไฟล์ที่มีส่วนขยายเป็น .ps (Postscript) หรือ .pdf นั้นต้องทำการติดตั้ง โปรแกรมที่มีชื่อว่า GhostView และ Acrobat Reader จาก Adobe เพื่อใช้ในการแสดงผลตามลำดับ
- ไฟล์ .log เป็นไฟล์ที่ถูกสร้างขึ้นอัตโนมัติโดยโปรแกรม ซึ่งใช้ในการเก็บข้อมูลสถานะต่าง ๆ และข้อผิดพลาดที่เกิดขึ้น

นอกจากไฟล์ทั้ง 3 แล้ว หากผู้ใช้มีการเรียกใช้คำสั่งในการสร้างสารบัญ สารบัญตาราง และสารบัญรูป กล่าวคือ \tableofcontents, \listoftables และ \listoffigures ตามลำดับแล้ว โปรแกรมจะสร้างไฟล์ .toc, .lot และ .lof ขึ้นมาเพื่อทำหน้าที่เป็นเหมือน Bookkeeper ในการเก็บข้อมูลรายละเอียดของบท หัวข้อ หัวข้อย่อย คำบรรยายได้รูป คำบรรยายได้ตาราง และหน้า เพื่อนำมาสร้างหน้าสารบัญ สารบัญตาราง และสารบัญรูปในเอกสาร

2.2.2 การใช้งาน $\text{IAT}_{\text{E}}\text{X}$ ภาษาไทย

โปรแกรม $\text{IAT}_{\text{E}}\text{X}$ บนระบบปฏิบัติการวินโดวส์ (Windows) ที่นิยมเรียกใช้กันคือ โปรแกรม $\text{MiK}_{\text{T}}\text{E}_{\text{X}}$ ซึ่งสามารถดาวน์โหลดมาใช้งานได้โดยไม่เสียค่าใช้จ่าย และหากประสงค์ที่จะพิมพ์งานภาษาไทยผู้ใช้ควรดาวน์โหลดแพคเกจที่ชื่อว่า $\text{MiK}_{\text{T}}\text{E}_{\text{X}}$ Thai Extension ซึ่งเป็นไฟล์สนับสนุนการใช้งานภาษาไทยของ $\text{MiK}_{\text{T}}\text{E}_{\text{X}}$

การติดตั้งภาษาไทยให้สามารถใช้งานบน $\text{IAT}_{\text{E}}\text{X}$ ได้นั้น มีขั้นตอนการติดตั้งที่แตกต่างกันออกไปในโปรแกรมแต่ละรุ่น หากผู้ใช้เลือกใช้ $\text{MiK}_{\text{T}}\text{E}_{\text{X}}$ รุ่น 2.5 หรือ 2.6 ซึ่งมีการพัฒนาตัวติดตั้งภาษาไทยแบบใหม่ โดยมีการปรับปรุงให้มีชนิดตัวอักษรมากขึ้นซึ่งรวมไปถึงสามารถใช้ชุดแบบอักษร AngsanaNew และ AngsanaUPC ได้ โดยในการใช้งานภาษาไทยนั้นผู้จำเป็นต้องเพิ่มชุดคำสั่ง `\usepackage[English,thai]{babel}` เข้าไปในส่วนอาร์กิวเมนต์ของเอกสารต้นฉบับ หากผู้ใช้เลือกใช้ $\text{MiK}_{\text{T}}\text{E}_{\text{X}}$ รุ่น 2.7 ซึ่งเป็นรุ่นล่าสุดในการใช้ประมวลผล พบว่ามีการเลือกใช้ชุดแบบอักษรภาษาไทยผ่านทาง XeLaTeX ซึ่งเป็น $\text{T}_{\text{E}}\text{X}$ รุ่นที่ออกแบบมาเพื่อให้สนับสนุนชุดแบบอักษรแบบ OpenType ตั้งแต่ต้น ทำให้มีชุดแบบอักษรที่หลากหลายมากขึ้น แต่ข้อเสียของ XeLaTeX ยังมีอยู่ กล่าวคือ XeLaTeX เป็นโปรแกรมที่เริ่มพัฒนาไม่นานนัก ยังพบข้อบกพร่องอยู่มาก ทำให้ไม่สามารถใช้ชุดคำสั่งบางส่วนได้ โดยเฉพาะชุดคำสั่งที่เกี่ยวกับการจัดการไฟล์ PDF

2.2.3 การตัดคำภาษาไทย

ตามหลักไวยากรณ์ในภาษาไทยนั้น แต่ละคำในประโยคจะเขียนเรียงติดกันไปและจะทำการเว้นวรรคหรือเว้นช่องว่างเมื่อจบประโยคนั้น ดังนั้นหากประโยคที่ต้องการจัดเรียงพิมพ์มีความยาวเกินระยะกั้นหลังของหน้ากระดาษ โปรแกรม $\text{IAT}_{\text{E}}\text{X}$ จะไม่สามารถตัดบางส่วนของประโยคเพื่อนำมาขึ้นบรรทัดใหม่ ซึ่งลักษณะดังกล่าวนี้เป็นข้อแตกต่างจากลักษณะการเขียนประโยคของภาษาตะวันตกที่มีการเว้นวรรคระหว่างคำ

ด้วยเหตุนี้จึงจำเป็นต้องอาศัยโปรแกรมอื่นมาช่วยในการสร้างช่องว่างระหว่างคำ เพื่อให้โปรแกรม $\text{IAT}_{\text{E}}\text{X}$ ทำการตัดคำในการขึ้นบรรทัดใหม่ได้อย่างถูกต้องสำหรับโปรแกรมที่ช่วยสร้างช่องว่างระหว่างคำ ได้แก่

1. โปรแกรม SWATH (Smart Word Analysis for Thai)

โปรแกรม SWATH จะทำการใส่คำสั่ง `\wbr` ด้านท้ายของแต่ละคำในเอกสารต้นฉบับ โดยคำสั่งดังกล่าวมีนิยาม คือ

```
\def\wbr{\hskip0pt plus0.6pt\relax}
```

ซึ่งทำหน้าที่สร้างช่องว่างขนาด 0 pt เพื่อให้โปรแกรม $\text{IAT}_{\text{E}}\text{X}$ รับรู้ว่ามีขอบดังกล่าวสามารถแยกคำออกจากกันในการขึ้นบรรทัดใหม่ได้

ตัวอย่างการใช้งานโปรแกรม SWATH

\$ swath -f latex < file.tex > file_new.tex

ตัวอย่างเช่น ในเอกสารต้นฉบับ มีประโยคที่ว่า “ภาษาไทยวันละคำ” เมื่อผ่านการประมวลผลด้วยโปรแกรมSWATHแล้วจะได้ประโยคใหม่เป็น“ภาษาไทย\wbrวัน\wbrละ\wbrคำ\wbr” เป็นต้น

2. โปรแกรม CTTEX

สำหรับโปรแกรม CTTEX มีลักษณะการทำงานคล้ายกับโปรแกรม SWATH ซึ่งตัวอย่างคำสั่งการตัดคำของโปรแกรม CTTEX เป็นดังนี้

\$ cttx -w < oldfile.tex > newfile.tex

โดยที่โปรแกรมจะแทรกคำสั่ง “\cb” ไว้ระหว่างคำ อย่างไรก็ตามหากเอกสารที่ได้ยังมีบางคำที่อยู่เลยแนวกันหลัง ผู้ใช้งานสามารถแก้ไขไฟล์ต้นฉบับด้วยการใช้เครื่องหมาย “-” หรือเว้นวรรคระหว่างคำได้ [1], [4]

3. Ss (Simple thai word Separator)

Ss เป็น โปรแกรมตัดคำโดยอาศัยคำในพจนานุกรมเป็นตัวเปรียบเทียบ ซึ่งโปรแกรมนี้พัฒนามาจาก โปรแกรม CTTEX อย่างไรก็ตามหากเอกสารที่ได้ยังมีบางคำที่อยู่เลยแนวกันหลัง ผู้ใช้สามารถแก้ไขไฟล์ต้นฉบับด้วยการใส่ <WBR> ซึ่งโปรแกรมนี้พัฒนาโดยคุณธีระ กิตติเจริญพิศ

2.3 ตัวอย่างของเอกสารที่จัดพิมพ์ด้วย L^AT_EX

ในหัวข้อนี้เป็นตัวอย่างการประมวลผลเอกสารที่จัดพิมพ์ด้วย L^AT_EX และการตัดคำภาษาไทย ทั้งในเอกสารประเภทหนังสือและเอกสารประเภทสื่อนำเสนอผ่านเครื่องฉาย

2.3.1 ตัวอย่างเอกสารประเภทหนังสือ

พิจารณาเอกสารต้นฉบับ ดังรูปที่ 2.5

```

\documentclass{book}
\usepackage[english,thai]{babel}
\usepackage{thswatch}
\begin{document}
\tableofcontents
\chapter{บทนำและความสำคัญของโครงการ}
\section{ประวัติของโครงการ}
\TeX คือ โปรแกรมคอมพิวเตอร์ที่สร้างชื่อโดยศาสตราจารย์โดนัลด์ อี. นูธ (Donald E. Knuth)
(หรือในบางครั้งเรียกสั้นๆว่า "นูธ") โปรแกรม \TeX นี้ถูกพัฒนาขึ้นโดยเลสลีย์ แลมพอร์ต (Leslie Lamport)
เพื่อใช้กับระบบ \LaTeX ซึ่ง \TeX เป็นตัวประมวลผล \LaTeX และ \TeX เป็นโปรแกรมที่ทำงานบนเครื่องคอมพิวเตอร์ทุกชนิด
และ \TeX สามารถทำงานได้บนระบบปฏิบัติการที่หลากหลาย เช่น Unix, Windows, Mac OS เป็นต้น โปรแกรม
สามารถนำไปประมวลผลด้วย \LaTeX บนระบบปฏิบัติการที่หลากหลายได้ เช่น Linux, Windows, Unix เป็นต้น โปรแกรม
\LaTeX เป็นโปรแกรมที่ Open Source สามารถดาวน์โหลดได้จากเว็บไซต์ "http://www.latex-project.org"
เป็นโครงการที่สนับสนุนโดยมูลนิธิโครงการ "American Mathematical Society"
ซึ่งมีสำนักงานใน "Washington, D.C." และมีการออกนิตยสาร "Notices of the American Mathematical Society"
ซึ่งมีฉบับภาษาไทยเป็นฉบับพิเศษ (CD) ในชื่อ "โครงการประมวลผล \LaTeX ภาษาไทย"
ซึ่งมีโปรแกรม "thswatch" และโปรแกรม "thswatch" (GUI)
ที่ "http://www.thswatch.org"
\end{document}

```

รูปที่ 2.5: แสดงเอกสารต้นฉบับประเภทหนังสือก่อนการตัดคำ

เมื่อนำเอกสารดังกล่าวไปผ่านการประมวลผล (Compile) ด้วย \LaTeX จะได้เอกสาร
ดั่งรูปที่ 2.6

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

\TeX คือ โปรแกรมคอมพิวเตอร์ที่สร้างขึ้นโดยศาสตราจารย์ดอนัลด์ อี. นูธ (Donald E. Knuth) เพื่อใช้ในการเรียงพิมพ์ข้อความ ในขณะที่โปรแกรม \LaTeX ได้ถูกพัฒนาขึ้นโดยเรสตี แลมพอร์ท (Leslie Lamport) โดยที่โปรแกรม \TeX ต้องใช้ \TeX เป็นตัวประมวลผลหลัก \LaTeX จัดว่าเป็นโปรแกรมสำเร็จที่รวมกลุ่มคำสั่งต่าง ๆ เพื่อให้ผู้ใช้สามารถจัดเรียงเอกสารที่มีคุณภาพ และข้อมูลในการพิมพ์เอกสารยังถูกจัดเก็บอยู่ในรูปของ Text File ซึ่งเป็นไฟล์ที่มีขนาดเล็ก สามารถนำไปประมวลผลด้วย \LaTeX บนระบบปฏิบัติการได้หลายประเภท เช่น Linux, Windows, Unix เป็นต้น โปรแกรม \LaTeX เป็นโปรแกรมตระกูล Open Source สามารถดาวน์โหลดได้โดยไม่เสียค่าใช้จ่าย

เนื่องจากโครงการนี้เป็นการพัฒนาโครงการต่อเนื่องจากนายอานา สุวีโรคม ซึ่งทำโครงการในหัวข้อเรื่องการออกแบบรูป \LaTeX ภาษาไทย จึงผลงาที่ได้เป็นแผ่นซีดี (CD) เพื่อให้การลงโปรแกรม \LaTeX ภาษาไทย รวมถึงโปรแกรมต่าง ๆ ที่จำเป็นคือจึงนำมาใช้ในการใช้งาน \LaTeX และโปรแกรมส่วนต่อประสานกราฟที่ผู้ (GUI) ที่ใช้ในการจัดรูปแบบของเอกสารและการติดตั้งภาษาไทย

รูปที่ 2.6: แสดงเอกสารประเภทหนังสือที่ผ่านการประมวลผลด้วย \LaTeX

จากรูปที่ 2.6 จะเห็นว่าข้อความในเอกสารที่ได้มีข้อความที่สั้นออกมาจากตำแหน่ง
ระยะกั้นหลังที่ได้ตั้งค่าไว้ อันเป็นผลมาจาก \LaTeX ไม่สามารถหาช่องว่างในเอกสารต้นฉบับ
เพื่อนำมาใช้ในการเว้นวรรคเพื่อขึ้นบรรทัดใหม่ได้ ดังนั้นจึงต้องนำเอกสารต้นฉบับ ในรูป
ที่ 2.5 ไปทำการตัดคำภาษาไทย ในที่นี้คณะผู้จัดทำเลือกใช้โปรแกรม SWATH ผู้อ่านสามารถ
สังเกตได้ว่าโปรแกรม SWATH จะแทรกคำสั่ง \wbr ไว้ท้ายของคำแต่ละคำ ดังแสดงในรูป
ที่ 2.7

```
\documentclass{book}
\usepackage[english,thai]{babel}
\usepackage{thswlthch}
\begin{document}
\tableofcontents
\chapter{บทนำ}
\section{ที่มาและความสำคัญของโครงการ}
 $\text{\LaTeX}$  คือ โปรแกรมคอมพิวเตอร์ที่สร้างขึ้นโดยศาสตราจารย์ดอนัลด์ อี. นูธ (Donald E. Knuth) และ  $\text{\LaTeX}$  ได้ถูกพัฒนาขึ้นโดยเรสตี แลมพอร์ท (Leslie Lamport) โดยที่โปรแกรม  $\text{\TeX}$  ต้องใช้  $\text{\TeX}$  เป็นตัวประมวลผลหลัก  $\text{\LaTeX}$  จัดว่าเป็นโปรแกรมสำเร็จที่รวมกลุ่มคำสั่งต่าง ๆ เพื่อให้ผู้ใช้สามารถจัดเรียงเอกสารที่มีคุณภาพ และข้อมูลในการพิมพ์เอกสารยังถูกจัดเก็บอยู่ในรูปของ Text File ซึ่งเป็นไฟล์ที่มีขนาดเล็ก สามารถนำไปประมวลผลด้วย  $\text{\LaTeX}$  บนระบบปฏิบัติการได้หลายประเภท เช่น Linux, Windows, Unix เป็นต้น โปรแกรม  $\text{\LaTeX}$  เป็นโปรแกรมตระกูล Open Source สามารถดาวน์โหลดได้โดยไม่เสียค่าใช้จ่าย
```

รูปที่ 2.7: แสดงเอกสารต้นฉบับประเภทหนังสือหลังการตัดคำ

เมื่อนำเอกสารในรูปที่ 2.7 ไปผ่านการประมวลผลด้วยโปรแกรม L^AT_EX จะได้เอกสาร ดังรูปที่ 2.8

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

L^AT_EX คือ โปรแกรมคอมพิวเตอร์ที่สร้างขึ้นโดยศาสตราจารย์โดนัลด์ อี.นุท (Donald E.Knuth) เพื่อใช้ในการเรียงพิมพ์ข้อความ ในขณะที่โปรแกรม D^AT_EX ได้ถูกพัฒนาขึ้นโดยเรสตีแลมพอร์ต (Leslie Lamport) โดยที่โปรแกรม L^AT_EX ต้องใช้ T_EX เป็นตัวประมวลผลหลัก L^AT_EX จัดว่าเป็นโปรแกรมสำเร็จที่รวบรวมกลุ่มคำสั่งต่าง ๆ เพื่อให้ผู้ใช้งานสามารถจัดเรียงเอกสารที่มีคุณภาพ และข้อมูลในการพิมพ์เอกสารจะถูกจัดเก็บอยู่ในรูปของ Text File ซึ่งเป็นไฟล์ที่มีขนาดเล็ก สามารถนำไปประมวลผลด้วย L^AT_EX บนระบบปฏิบัติการได้หลายประเภท เช่น Linux, Windows, Unix เป็นต้น โปรแกรม D^AT_EX เป็นโปรแกรมตระกูล Open Source สามารถดาวน์โหลดได้โดยไม่เสียค่าใช้จ่ายใด ๆ

เนื่องจากโครงงานนี้เป็นการพัฒนาโครงงานต่อจากนายอำนาจ ผู้วิจัย จึงทำโครงงานในหัวข้อเรื่องการออกแบบรูปแบบเอกสารสำหรับปฏิญญาพันธ์ด้วยโปรแกรมสำเร็จ L^AT_EX ภาษาไทย ซึ่งผลงานที่ได้เป็นแผ่นซีดี (CD) เพื่อใช้ในการลงโปรแกรม L^AT_EX ภาษาไทย รวมถึงโปรแกรมต่าง ๆ ที่จำเป็นต่อการใช้งานในการใช้ L^AT_EX และโปรแกรมส่วนต่อประสานกราฟฟิคกับผู้ใช้ (GUI) ที่ใช้ในการจัดรูปแบบของเอกสารและการตั้งค่าภาษาไทย

รูปที่ 2.8: แสดงเอกสารหนังสือที่ได้จากเอกสารต้นฉบับที่มีการตัดคำ

จากรูปที่ 2.8 จะเห็นว่าเอกสารที่ได้หลังจากตัดคำมีความสวยงามและมีการขึ้นบรรทัดใหม่อย่างถูกต้อง ซึ่งหากผู้ใช้ต้องการจะแก้ไขเนื้อหาในเอกสารต้นฉบับควรทำการแก้ไขในเอกสารก่อนตัดคำ เพราะง่ายและสะดวกต่อการอ่าน

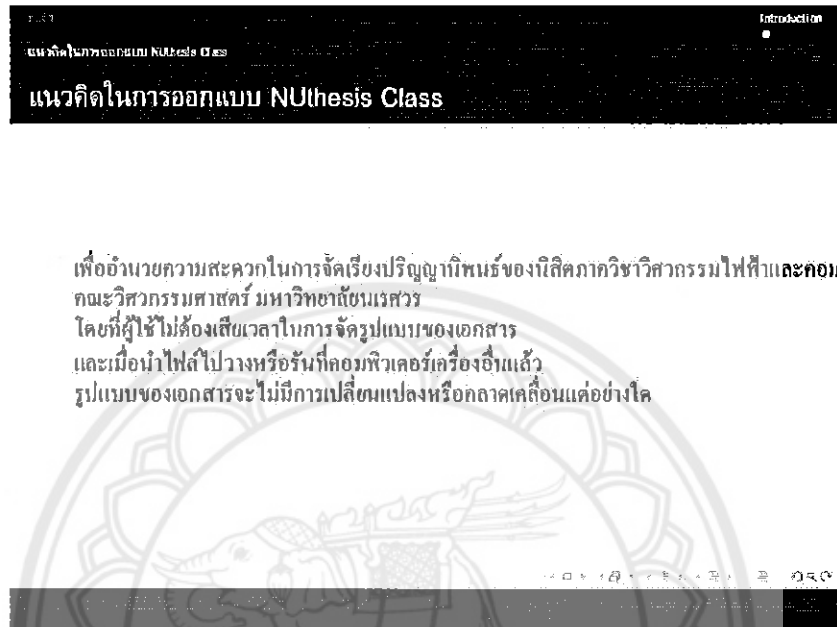
2.3.2 ตัวอย่างเอกสารประเภทเอกสารประเภทสื่อแนะนำเสนอผ่านเครื่องฉาย

พิจารณาเอกสารต้นฉบับประเภทสื่อแนะนำเสนอผ่านเครื่องฉาย ดังแสดงในรูปที่ 2.9

```
\documentclass[compress,red,9pt]{beamer}
\usepackage[english,thai]{babel}
\usepackage{thswitch}
\usepackage{beamerthemsplit}
\def\thai{\rmdefault(uangsana)}
\usetheme{Berlin}
|
\begin{document}
\begin{frame}
\titlepage
\end{frame}
\section{main}{}
\frame{\tableofcontents}
\section{Introduction}
\subsection{แนวทางในการออกแบบ Nuthesis Class}
\frame{\frametitle{แนวทางในการออกแบบ Nuthesis Class}
เป็น "อำนาจความสะอาดในการจัดเรียงปฏิญญาพันธ์ของสภาทนาย" วิชาทนายศาสตร์และกฎหมายเพื่อ
คณะทนายศาสตร์ มหาวิทยาลัยธรรมศาสตร์ โดยมี "ใช้" ใน "ห้องเรียนเวลาในการจัดรูปแบบของเอกสาร
และเน" อย่างไรก็ตาม "ไปทาง" อธิบาย "ก่อน" มิว "เพอร์" "ออง" "น" "น" "น"
รูปแบบของเอกสารจะไม่มีการเปลี่ยนแปลงหรือ "คัดลอก" "คัดลอก" "น" "น" "น" "น" "น"
)
\end{document}
```

รูปที่ 2.9: แสดงเอกสารประเภทสื่อแนะนำเสนอก่อนการตัดคำ

จากนั้นนำเอกสารในรูปแบบที่ 2.9 ไปทำการประมวลผลด้วย $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ จะได้เอกสาร ดังรูปที่ 2.10



รูปที่ 2.10: แสดงเอกสารประเภทสื่อนำเสนอที่ผ่านการประมวลผลด้วย $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$

จากรูปที่ 2.10 จะเห็นว่าข้อความในเอกสารที่ได้มีข้อความสั้นออกมาจากตำแหน่งระยะกั้นหลังที่ได้ตั้งค่าเช่นเดียวกับในกรณีของเอกสารประเภทหนังสือ (ดูรูปที่ 2.6 ประกอบ) จึงต้องนำเอกสารต้นฉบับดังกล่าวไปผ่าน โปรแกรมตัดคำภาษาไทย ซึ่งจะได้เอกสารตามรูปที่ 2.11

```

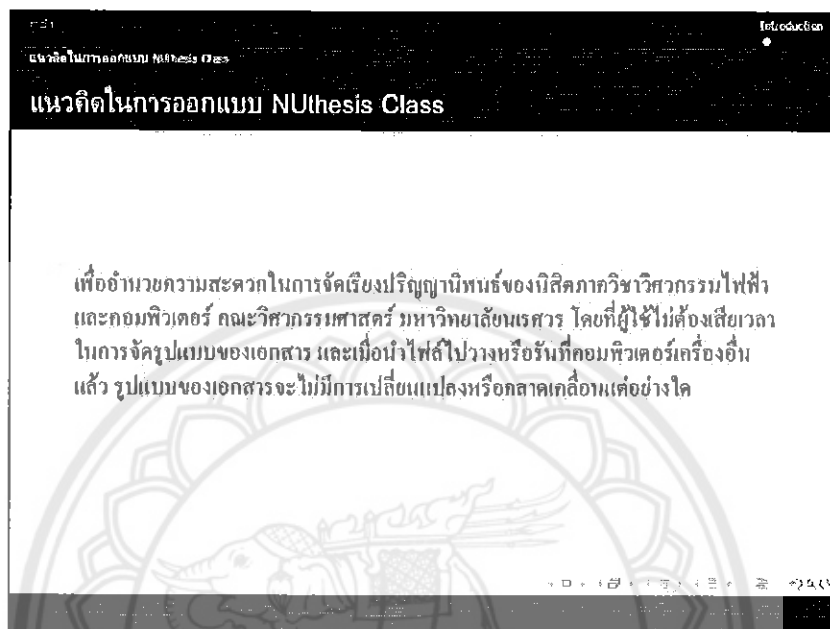
\documentclass[compress,red,9pt](beamer)
\usepackage[english,thai](babel)
\usepackage{thswatch}
\usepackage{beamerthemessplit}
\def\thairndefault(uangsana)
\usetheme(Berlin)

\begin{document}
\begin{frame}
\titlerpage
\end{frame}
\section{main}{}
\frame{\tableofcontents}
\section{Introduction}
\subsection{แนว\br/>ก\br/>ใน\br/>การ\br/>ออกแบบ NUThesis Class}
\frame{\frametitle{แนว\br/>ก\br/>ใน\br/>การ\br/>ออกแบบ NUThesis Class}
ใน\br/>ฉบับ\br/>นี้\br/>จะ\br/>กล่าว\br/>ถึง\br/>แนว\br/>คิด\br/>ในการ\br/>ออกแบบ\br/>เอกสาร\br/>ที่\br/>มี\br/>ข้อความ\br/>สั้น\br/>ออกมา\br/>จาก\br/>ตำแหน่ง\br/>ระยะ\br/>กั้น\br/>หลัง\br/>ที่\br/>ได้\br/>ตั้งค่า\br/>เช่น\br/>เดียวกับ\br/>ใน\br/>กรณี\br/>ของ\br/>เอกสาร\br/>ประเภท\br/>หนังสือ\br/>(ดู\br/>รูป\br/>ที่\br/>2.6\br/>ประกอบ)\br/>จึง\br/>ต้อง\br/>นำ\br/>เอกสาร\br/>ต้น\br/>ฉบับ\br/>ดังกล่าว\br/>ไป\br/>ผ่าน\br/>โปรแกรม\br/>ตัด\br/>คำ\br/>ภาษาไทย\br/>ซึ่ง\br/>จะ\br/>ได้\br/>เอกสาร\br/>ตาม\br/>รูป\br/>ที่\br/>2.11
\end{frame}
\end{document}

```

รูปที่ 2.11: แสดงเอกสารต้นฉบับประเภทสื่อนำเสนอหลังการตัดคำ

เมื่อนำเอกสารที่ผ่านการตัดคำในรูปที่ 2.11 ไปผ่านการประมวลผลด้วย โปรแกรม L^AT_EX จะได้เอกสารประเภทสื่อนำเสนอ ดังรูปที่ 2.12



รูปที่ 2.12: แสดงเอกสารประเภทสื่อนำเสนอที่ได้จากเอกสารต้นฉบับที่มีการตัดคำ

2.4 ความแตกต่างระหว่าง Microsoft Office และโปรแกรม L^AT_EX

ในการจัดพิมพ์เอกสารทั่วไป ผู้ใช้งานส่วนใหญ่เลือกใช้ Microsoft Office เนื่องจากใช้งานง่าย สะดวก และไม่มีปัญหาในเรื่องการตัดคำภาษาไทย รวมทั้งยังมีเครื่องมือ (Tools) เพื่ออำนวยความสะดวกในการใช้งาน ในขณะที่การจัดพิมพ์เอกสารด้วยโปรแกรม L^AT_EX ผู้ใช้งานจำเป็นต้องจดจำคำสั่งของ L^AT_EX และมีความยุ่งยากในการใช้งานสำหรับผู้เริ่มต้น อย่างไรก็ตาม โปรแกรม L^AT_EX เป็นโปรแกรมเรียงพิมพ์ที่มีประสิทธิภาพสูง กล่าวคือ สามารถแสดงผลสมการ รูป และตาราง ได้อย่างสวยงาม โดยไม่ขึ้นกับอุปกรณ์ในการแสดงผล อีกทั้งยังสามารถใช้งานได้กับทุกระบบปฏิบัติการ และยังเหมาะกับองค์กรในกรณีที่มีเอกสารต้นแบบและต้องการให้เอกสารอื่น ๆ ที่ทำขึ้นภายหลังมีลักษณะตรงตามต้นแบบนั้น

จากที่กล่าวมาข้างต้นสามารถสรุปความแตกต่างระหว่าง Microsoft Office และ โปรแกรม $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ดังตารางต่อไปนี้

Microsoft Office	$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
<p>ข้อดี</p> <ol style="list-style-type: none"> 1. ใช้งานง่าย 2. ไม่มีปัญหาเรื่องการตัดคำภาษาไทย 3. มีหนังสือหรือ E-Book ให้ศึกษามาก 4. แลกเปลี่ยนเอกสารกับผู้อื่นได้สะดวก 5. มีเครื่องมือ (Tool) ให้ใช้งานมาก 	<p>ข้อดี</p> <ol style="list-style-type: none"> 1. การเขียนสูตรคณิตศาสตร์เป็นมาตรฐานสากล และยังมีระบบการเรียงลำดับเลขที่กำกับสมการ รูป ตาราง โดยอัตโนมัติ 2. ใช้หน่วยความจำไม่มาก 3. เป็นโปรแกรมโอเพนซอร์ส (Open Source) 4. สามารถสร้างเอกสารที่ซับซ้อนได้ 5. สามารถใช้ได้กับคอมพิวเตอร์ทุกระบบปฏิบัติการ (OS) เช่น Windows, Mac และ UNIX เป็นต้น 6. มีการจัดช่องว่างระหว่างตัวอักษรโดยอัตโนมัติ 7. สามารถทำ Cross-Reference ได้อัตโนมัติ เช่น สารบัญ สารบัญตาราง สารบัญรูป ภาคผนวก และ บรรณานุกรม เป็นต้น
<p>ข้อเสีย</p> <ol style="list-style-type: none"> 1. ระบบการเรียงลำดับเลขที่กำกับสมการ รูป ตาราง มีความยุ่งยากในการใช้งาน 2. การสร้างสารบัญมีความยุ่งยากในการใช้งาน 3. โปรแกรมมีลิขสิทธิ์ 4. ไม่สามารถใช้กับคอมพิวเตอร์ได้ทุกระบบปฏิบัติการ (OS) 	<p>ข้อเสีย</p> <ol style="list-style-type: none"> 1. ใช้งานยุ่งยากสำหรับผู้เริ่มต้น 2. มีปัญหาเรื่องการตัดคำภาษาไทย 3. มีหนังสือภาษาไทยให้ศึกษาน้อย

ตารางที่ 2.1: ความแตกต่างระหว่างการพิมพ์งานด้วย Microsoft Office และ $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$

สารบัญ

1	บทที่ 1	2
1.1	ที่มาและความสำคัญของโครงการงาน	2

1

รูปที่ 3.2: แสดงหน้าสารบัญในเอกสารประเภทหนังสือ

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการงาน

TeX คือ โปรแกรมคอมพิวเตอร์ที่สร้างขึ้นโดยศาสตราจารย์ดอนัลด์ อี คุน (Donald E. Knuth) เพื่อใช้ในการเรียงพิมพ์ข้อความ ในขณะที่โปรแกรม LaTeX ได้ถูกพัฒนาขึ้นโดยเรสตี แลมพอน (Leslie Lamport) โดยที่โปรแกรม LaTeX ต้องใช้ TeX เป็นตัวประมวลผลหลัก LaTeX จัดว่าเป็นโปรแกรมสำเร็จที่รวบรวมชุดคำสั่งต่าง ๆ เพื่อให้ผู้ใช้สามารถจัดเรียงเอกสารที่มีคุณภาพ และข้อมูลในการพิมพ์เอกสารยังคงจัดเก็บอยู่ในรูปของ Text File ซึ่งเป็นไฟล์ที่มีขนาดเล็กลงกว่าโปรแกรมต้นฉบับของ LaTeX บนระบบปฏิบัติการได้หลายประเภท เช่น Linux, Windows, Unix เป็นต้น โปรแกรม LaTeX เป็นโปรแกรมตระกูล Open Source สามารถดาวน์โหลดได้โดยไม่เสียค่าใช้จ่าย

เนื่องจากโครงการนี้เป็นการพัฒนาโครงการจากภายนอก (ผู้ควบคุม) ซึ่งทำโครงการในเครือข่ายของหน่วยงานแบบรูป LaTeX ภายนอก ซึ่งผลงานที่ได้เป็นแผ่นซีดี (CD) เพื่อใช้ในการลงโปรแกรม LaTeX ภายนอก รวมถึงโปรแกรมต่าง ๆ ที่จำเป็นต้องใช้ในการใช้งาน LaTeX และโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ (GUI) ที่ใช้ในการจัดรูปแบบของเอกสารและการตัดคำภาษาไทย

2

รูปที่ 3.3: แสดงหน้าเนื้อหาในเอกสารประเภทหนังสือ

จากรูปที่ 3.2 และรูปที่ 3.3 ผู้อ่านสามารถสังเกตได้ว่า เอกสารประเภทหนังสือมีลักษณะที่สำคัญ ดังนี้

1. คำว่า “สารบัญ” อยู่ชิดซ้ายของหน้ากระดาษ
2. คำว่า “บทที่” ในหน้าสารบัญ ไม่ปรากฏก่อนหน้าชื่อประจำบท
3. คำว่า “หน้า” ในหน้าสารบัญ ไม่ปรากฏเหนือหมายเลขหน้า
4. หมายเลขหน้าในหน้าสารบัญ แสดงที่ตำแหน่งกึ่งกลางด้านล่างของหน้ากระดาษ
5. คำว่า “บทที่” อยู่ชิดซ้ายของหน้ากระดาษ
6. คำว่า “ชื่อบท” อยู่ชิดซ้ายของหน้ากระดาษ

7. หมายเลขหน้า แสดงที่ตำแหน่งกึ่งกลางด้านล่างของหน้ากระดาษและปรากฏในทุกหน้าเนื้อหาของรายงาน

3.1.2 รูปแบบเอกสารปริญญานิพนธ์

จากเอกสารคู่มือการเขียนรายงาน วิศวกรรมศาสตร์ของภาควิชาวิศวกรรมไฟฟ้า และคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร [3] มีรายละเอียดในการเขียนเอกสารปริญญานิพนธ์ ดังนี้

1. ใช้ตัวอักษรแบบ Angsana UPC ตลอดทั้งเอกสาร
2. บทที่ ต้องพิมพ์บรรทัดแรกและจัดตรงกลางหน้ากระดาษ พร้อมหมายเลขบท โดยใช้ขนาดตัวอักษร 20 pt
3. ชื่อเรื่องประจําบท พิมพ์กลางหน้ากระดาษ โดยใช้ขนาดตัวอักษร 24 pt
4. การลำดับหน้าและการพิมพ์เลขหน้า
 - ส่วนหน้า (Front Matter) ตั้งแต่บทคัดย่อถึงสารบัญรูป ให้ใช้ตัวอักษรภาษาไทย ก-ฮ ตามลำดับ โดยพิมพ์ไว้ตรงกลางส่วนล่างของหน้ากระดาษ
 - ส่วนเนื้อหา (Main Matter) ตั้งแต่บทนำจนจบเนื้อหา ให้ใช้ตัวเลขอารบิก 1, 2, 3, ... แสดงเลขหน้า โดยพิมพ์ไว้ด้านบนขวา
 - ส่วนหลัง (Back Matter) การใส่เลขหน้าให้ใช้ตัวเลขอารบิก 1, 2, 3, ... ตามลำดับ ต่อเนื่องจากส่วนเนื้อหา
 - ส่วนที่เป็นบทที่หรือหน้าแรกของแต่ละบท ไม่ต้องใส่เลขหน้าแต่ให้นับหน้านั้นด้วย
5. สารบัญ สารบัญตาราง และสารบัญรูป พิมพ์คำว่า สารบัญ สารบัญตาราง และสารบัญรูป ไว้กึ่งกลางที่บรรทัดแรกของหน้ากระดาษ เว้น 1 บรรทัด และพิมพ์คำว่า “หน้า” ซิด ด้านขวา ในส่วนของสารบัญตารางและสารบัญรูป ให้พิมพ์คำว่า “ตารางที่” หรือ “รูปที่” โดยพิมพ์ชิดขอบซ้ายและอยู่บรรทัดเดียวกับคำว่า “หน้า” และถ้าเนื้อหาในหน้าสารบัญ สารบัญตาราง และสารบัญรูป ไม่สามารถเขียนได้หมดภายใน 1 หน้ากระดาษ จำเป็นต้องขึ้นหน้าใหม่ให้เขียนกลางหน้ากระดาษว่า “สารบัญ(ต่อ)”, “สารบัญตาราง(ต่อ)” และ “สารบัญรูป(ต่อ)” ตามลำดับ

จากที่ได้กล่าวมาข้างต้น สรุปข้อแตกต่างระหว่างลักษณะของเอกสารที่ได้จากเอกสารประเภทหนังสือ และรูปแบบเอกสารปริญญานิพนธ์ ดังนี้

เอกสารจาก Book Class	เอกสารปริญญาโท
1. คำว่า “บทที่” อยู่ชิดซ้ายของหน้ากระดาษ	1. คำว่า “บทที่” อยู่กึ่งกลางหน้ากระดาษ
2. ชื่อบทอยู่ชิดซ้ายของหน้ากระดาษ	2. ชื่อบทอยู่กึ่งกลางหน้ากระดาษ
3. คำว่า สารบัญ สารบัญตาราง และสารบัญ รูปอยู่ชิดซ้ายของหน้ากระดาษ	3. คำว่า สารบัญ สารบัญตาราง และสารบัญรูป อยู่กึ่งกลางหน้ากระดาษ
4. ในหน้าสารบัญไม่ปรากฏคำว่า “บทที่” ที่หน้าชื่อบท	4. ในหน้าสารบัญปรากฏคำว่า “บทที่” ที่ หน้าชื่อบท
5. ในหน้าสารบัญไม่ปรากฏคำว่า “หน้า” เหนือหมายเลขหน้า	5. ในหน้าสารบัญปรากฏคำว่า “หน้า” เหนือ หมายเลขหน้า
6. หมายเลขหน้าแสดงที่ตำแหน่งกึ่งกลาง ด้านล่างของหน้ากระดาษตลอดทั้งเอกสาร	6. หมายเลขหน้าของส่วนหน้าแสดงที่ตำแหน่ง กึ่งกลางด้านล่างของหน้ากระดาษและใช้ตัวอักษร ท-ธ ตามลำดับ ในการแสดงเลขหน้า สำหรับ ส่วนอื่น ๆ ที่เหลือใช้ตัวเลขอารบิก 1, 2, 3, ... แสดงเลขหน้าโดยวางเลขหน้า ที่ด้านขวาบนของหน้ากระดาษ
	7. หน้าแรกของแต่ละบท ไม่มีเลขหน้ากำกับ

ตารางที่ 3.1: แสดงความแตกต่างระหว่างลักษณะของเอกสารที่ได้จากเอกสารประเภทหนังสือและรูปแบบเอกสารปริญญาโท

3.2 ขั้นตอนการพัฒนา NUthesis Class

หัวข้อนี้จะแสดงให้เห็นถึงขั้นตอนในการพัฒนารูปแบบเอกสารประเภทใหม่ที่เรียกว่า NUthesis Class โดยมีรายละเอียด ดังนี้

1. การสร้างชื่อประเภทของเอกสาร

ในโครงการนี้ต้องการพัฒนารูปแบบเอกสารจากรูปแบบเอกสารประเภทหนังสือ (Book Class) และตั้งชื่อรูปแบบเอกสารที่ได้ว่า NUthesis โดยใช้ $\LaTeX 2\epsilon$ เป็นตัวประมวลผล จากความประสงค์ดังกล่าว จึงต้องเรียกใช้คำสั่ง ดังนี้

```
\NeedsTeXFormat{LaTeX2e}[2009/15/06]
```

```
\ProvidesClass{NUthesis}
```

2. การเรียกใช้แพ็คเกจ (Package) ต่าง ๆ

แพ็คเกจ คือ เท็กซ์ไฟล์ (Text File) ที่บรรจุบรรดาคำสั่งต่างๆ ของ \LaTeX ที่ถูกเรียบเรียงขึ้นในลักษณะของสคริปต์สำหรับการทำงานอย่างใดอย่างหนึ่ง โดยที่แพ็คเกจจะเป็นตัวกำหนดรายละเอียดปลีกย่อยนอกเหนือจากการกำหนดลักษณะของเอกสารซึ่งกำหนดโดย Document Class แพ็คเกจมีทั้งชนิดที่เป็นส่วนหนึ่งของชุดติดตั้ง \LaTeX มาตรฐานและชนิดที่มีนักพัฒนารายอื่นผลิตขึ้น ซึ่งไฟล์แพ็คเกจจะมีนามสกุลเป็น .sty

เมื่อต้องการเรียกใช้แพ็คเกจใด จำเป็นที่จะต้องประกาศให้ \LaTeX ทราบในส่วนอาร์มบพของเอกสารต้นฉบับ โดยใช้คำสั่ง `\usepackage` หรือคำสั่ง `\RequirePackage` ในกรณีของ `NUthesis Class` มีการเรียกใช้แพ็คเกจ เช่น

```
\RequirePackage[english,thai]{babel}
\usepackage{thswitch}
\RequirePackage{fancyhdr}
```

โดยคำสั่ง `\RequirePackage[english,thai]{babel}` เป็นการประกาศว่าเอกสารนี้สามารถใช้ได้ทั้งภาษาอังกฤษและภาษาไทย คำสั่ง `\usepackage{thswitch}` ใช้ในการสลับการใช้ภาษาอังกฤษและภาษาไทย และคำสั่ง `\RequirePackage{fancyhdr}` เป็นการกำหนดการจัดหน้ากระดาษหรือการกำหนดลักษณะหัวกระดาษและท้ายกระดาษ

หมายเหตุ : แพ็คเกจแต่ละชุดมีตัวเลือกและรายละเอียดการใช้งานที่แตกต่างกันไป ผู้ใช้ต้องศึกษาข้อมูล วิธีการใช้แต่ละแพ็คเกจเป็นกรณีไป

3. ขนาดหน้ากระดาษและชนิดของตัวอักษร

ข้อกำหนดในการจัดพิมพ์เอกสารปริญญาบัตรนั้น ต้องใช้กระดาษขนาดมาตรฐาน A4 (21.0 ซม. x 29.7 ซม.) โดยมีการเว้นระยะห่างจากขอบกระดาษ ดังนี้

ด้านบนจะเว้นระยะห่างจากขอบกระดาษ 1 นิ้ว (2.54 ซม.)

ด้านล่างจะเว้นระยะห่างจากขอบกระดาษ 1 นิ้ว (2.54 ซม.)

ด้านขวาจะเว้นระยะห่างจากขอบกระดาษ 1 นิ้ว (2.54 ซม.)

ด้านซ้ายจะเว้นระยะห่างจากขอบกระดาษ 1.5 นิ้ว (3.81 ซม.)

กำหนดให้ใช้ตัวอักษร Angsana UPC ตลอดทั้งเอกสาร โดยในแต่ละหน้าจะกำหนดขนาดตัวอักษรและการย่อหน้าต่างกันไปในการพัฒนา `NUthesis Class` จึงมีการกำหนดค่าต่าง ๆ ให้เป็นไปตามข้อกำหนดดังกล่าว โดยมีการใช้คำสั่ง `\setlength` เพื่อใช้กำหนดค่าหน้ากระดาษและการกำหนดค่าเริ่มต้นของรูปแบบตัวอักษร

ในส่วนของการเว้นระยะห่างจากขอบกระดาษนั้น โดยหลักการทำงานพื้นฐานของโปรแกรม \LaTeX ไม่มีการตั้งค่าน้ำกระดาษโดยวัดจากขอบกระดาษ หากแต่ทำการวัดระยะจากจุดกึ่งกลางของหน้ากระดาษและตั้งค่าน้ำกระดาษ อันประกอบด้วยรายละเอียดและตัวอย่างดังนี้

```
\setlength\paperheight{29.7cm}
\setlength\paperwidth{21.0cm}
\def\thairmdefault{uangsana}
\setlength{\parindent}{1.20cm}
```

คำสั่ง `\setlength\paperheight{29.7cm}` คำสั่งนี้เป็นการกำหนดความยาวของหน้ากระดาษ โดยการนำระยะห่างจากขอบกระดาษด้านบนและด้านล่างมาหักออกจากค่าที่กำหนด (29.7 ซม.) จากนั้นนำค่าที่ได้ไปหารครึ่งแล้วทำการตั้งค่าน้ำกระดาษจากจุดกึ่งกลางหน้ากระดาษ

คำสั่ง `\setlength\paperwidth{21.0cm}` คำสั่งนี้เป็นการกำหนดความกว้างของหน้ากระดาษ โดยการนำระยะห่างจากขอบกระดาษด้านซ้ายและด้านขวามาหักออกจากค่าที่กำหนด (21.0 ซม.) จากนั้นนำค่าที่ได้ไปหารครึ่งแล้วทำการตั้งค่าน้ำกระดาษจากจุดกึ่งกลางหน้ากระดาษ

คำสั่ง `\def\hairmdefault{uangsana}` คำสั่งนี้เป็นการกำหนดให้ใช้ตัวอักษรแบบ `angsana`

คำสั่ง `\setlength{\parindent}{1.20cm}` คำสั่งนี้เป็นการกำหนดให้ย่อหน้ามีขนาด 1.20 ซม.

```
\setlength{\topmargin}{0.00cm}
\setlength{\oddsidemargin}{0.00cm}
\setlength{\evensidemargin}{0.00cm}
\setlength{\headheight}{0.00cm}
\setlength{\headsep}{1.00cm}
\setlength{\textwidth}{14.65cm}
\setlength{\textheight}{24.62cm}
\setlength{\footskip}{0.00cm}
\setlength{\parskip}{0.00cm}
```

คำสั่ง `\setlength{\topmargin}{0.0cm}` คำสั่งนี้เป็นการกำหนดระยะห่างจากขอบกระดาษด้านบน

คำสั่ง `\setlength{\oddsidemargin}{0.0cm}` คำสั่งนี้ใช้สำหรับหน้ากระดาษด้านขวา (หน้าหมายเลขคี่) สำหรับกำหนดระยะห่างจากขอบกระดาษด้านซ้าย

คำสั่ง `\setlength{\evensidemargin}{0.0cm}` คำสั่งนี้ใช้สำหรับหน้ากระดาษด้านซ้าย (หน้าหมายเลขคู่) สำหรับกำหนดระยะห่างจากขอบกระดาษด้านขวา

คำสั่ง `\setlength{\headheight}{0.0cm}` คำสั่งนี้ใช้สำหรับกำหนดขนาดของหัวกระดาษ

คำสั่ง `\setlength{\headsep}{1.0cm}` คำสั่งนี้ใช้สำหรับกำหนดระยะห่างของหัวกระดาษกับส่วนเนื้อหาของเอกสาร

คำสั่ง `\setlength{\textwidth}{14.65cm}` คำสั่งนี้เป็นการกำหนดความกว้างของเนื้อหาที่แสดงในเอกสาร ซึ่งเป็นการนำค่าความกว้างของหน้ากระดาษ 21.0 ซม. มาลบออกด้วยระยะกั้นหน้า 3.81 ซม. และระยะกั้นหลัง 2.54 ซม. ดังนั้นจึงได้ค่าความกว้างของเนื้อหาเป็น 14.65 ซม.

คำสั่ง `\setlength{\textheight}{24.62cm}` คำสั่งนี้เป็นการกำหนดความยาวของเนื้อหาที่แสดงในเอกสาร ซึ่งเป็นการนำค่าความยาวของหน้ากระดาษ 29.7 ซม. มาลบออกด้วยระยะขอบบน 2.54 ซม. และระยะขอบล่าง 2.54 ซม. ดังนั้นจึงได้ค่าความยาวของเนื้อหาเป็น 24.62 ซม.

คำสั่ง `\setlength{\footskip}{0.0cm}` คำสั่งนี้เป็นการเว้นระยะห่างจากเนื้อหาบางส่วนท้าย

หน้ากระดาษ

คำสั่ง `\setlength{\parskip}{0.0cm}` คำสั่งนี้เป็นการกำหนดระยะห่างระหว่างย่อหน้า (Distance Between Paragraph)

นอกจากการตั้งหน้ากระดาษ โดยใช้คำสั่งข้างต้นแล้วจะเห็นได้ว่ามีความยุ่งยากพอสมควร หารู้ก็ตามโปรแกรม $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ มีชุดคำสั่งที่ง่ายต่อการจัดหน้ากระดาษ กล่าวคือ การใช้แพ็คเกจ `geometry` ยกตัวอย่างคำสั่ง ดังนี้

```
\usepackage{geometry}
\geometry{top=2.54cm,bottom=2.54cm,left=3.81cm,right=2.54cm}
```

คำสั่ง `\usepackage{geometry}` คำสั่งนี้เป็นคำสั่งในการจัดหน้ากระดาษ คำสั่ง `\geometry{top=2.54cm,bottom=2.54cm,left=3.81cm,right=2.54cm}` คำสั่งนี้เป็นการกำหนดระยะห่างจากขอบกระดาษ คือ กำหนดให้ด้านบนห่าง 2.54 ซม. ด้านล่างห่าง 2.54 ซม. ด้านซ้ายห่าง 3.81 ซม. และด้านขวาห่าง 2.54 ซม.

4. รูปแบบลักษณะของหมายเลขหน้า

จากข้อกำหนดของภาควิชาฯ ระบุว่า หมายเลขแสดงเลขหน้าในส่วนหน้าของเอกสาร ต้องเป็นตัวอักษรภาษาไทย ก, ข, ค, ... ตามลำดับ ในขณะที่หมายเลขแสดงเลขหน้าในส่วนเนื้อหาหลัก (Main Matter) และส่วนหลัง ให้ใช้ตัวเลขอารบิก 1, 2, 3, ... เรียงกันไปตามลำดับ

```
\newif\if@mainmatter\@mainmattertrue
```

```
\newcommand\frontmatter{%
```

```
\cleardoublepage
```

```
\@mainmatterfalse
```

```
\pagestyle{plain}
```

```
\pagenumbering{thaialph}
```

```
}
```

```
\newcommand\mainmatter{%
```

```
\cleardoublepage
```

```
\@mainmattertrue
```

```
\pagestyle{myheadings}
```

```
\pagenumbering{arabic}
```

```
}
```

```
\newcommand\backmatter{%
```

```
\if@openright
```

```
\cleardoublepage
```

```
\else
\clearpage
\fi
\@mainmatterfalse
}
```

5. การจัดวางตำแหน่งของเลขหน้า

ข้อกำหนดของการทำปฏิญานិพนธ์ของภาควิชาฯ ระบุว่า ในส่วนของเนื้อหาหลักและส่วนหลังนั้น หมายเลขแสดงเลขหน้าต้องอยู่ที่ตำแหน่งด้านบนขวาของหน้ากระดาษ นอกจากนี้ในหน้าแรกของแต่ละบทต้องไม่แสดงเลขหน้า จากข้อกำหนดดังกล่าว จึงนำมาเขียนโค้ดได้ ดังนี้

- การทำให้หน้าแรกของแต่ละบทไม่ปรากฏเลขหน้า

```
\pagestyle{myheadings}
\def\ps@headings{%
\let\@oddfoot\@empty
\def\hfill\thepage{(\slshape
\rightmark)\hfill\thepage}%
\let\@mkboth\markboth
\def\chaptermark##1{%
\markright{\MakeUppercase{%
\ifnum\c@secnumdepth>\m@ne
\@chapapp\thechapter.\%
\fi
##1}}}}
```

- การทำให้หมายเลขหน้าในส่วนเนื้อหาไปอยู่ทางด้านบนขวา

```
\def\ps@headings{%
\let\@oddfoot\@empty
\def\@oddhead{\hfill\thepage}%
\let\@mkboth\markboth
\def\chaptermark##1{%
\markright{\MakeUppercase{%
\ifnum\c@secnumdepth>\m@ne
\@chapapp\thechapter.\%
\fi
##1}}}}
```

16738350

ร/ร.

น39317

2552

6. การจัดการเกี่ยวกับหน้าสารบัญ

เนื่องจากค่าเริ่มต้นเกี่ยวกับคำสั่ง \tableofcontent ที่ใช้สร้างสารบัญนั้น ไม่ตรงตามข้อกำหนดของภาควิชาฯ จึงได้มีการปรับปรุงแก้ไขรูปแบบต่าง ๆ ของสารบัญให้เป็นไปตามข้อกำหนด โดยมีรายละเอียดดังนี้

- การทำให้ชื่อบทในหน้าสารบัญไม่เป็นตัวหนา

เนื่องจากเมื่อมีการสร้างสารบัญแล้ว ตัวอักษรในหน้าสารบัญเป็นตัวหนาซึ่งไม่ถูกต้องตามข้อกำหนดของภาควิชาฯ จึงเปลี่ยนแปลงค่าเพื่อให้ตัวอักษรในหน้าสารบัญเป็นตัวอักษรปกติ จึงมีการปรับปรุงแก้ไขคำสั่ง เพื่อเปลี่ยนแปลงค่าให้เป็นไปตามต้องการ ดังตัวอย่างนี้

```
\renewcommand*\l@chapter[2]{%
\ifnum\c@tocdepth>\m@ne
\addpenalty{-\@highpenalty}%
\vskip0em\@plus\p@
\setlength\@tempdima{1.5em}%
\begingroup
\parindent\z@\rightskip\@pnumwidth
\parfillskip-\@pnumwidth
\advance\leftskip\@tempdima
\hskip-\leftskip#1\nobreak\hfill
\nobreak\hb@xt@\@pnumwidth{
\hss#2}\par\penalty\@highpenalty
\endgroup
\fi
}
```

- เพิ่มคำว่า “หน้า” และ “สารบัญ(ต่อ)”

เนื่องจากรูปแบบของเอกสารประเภท Book Class ไม่มีคำว่า “หน้า” ให้ในการสร้างสารบัญ และเมื่อสารบัญมีมากกว่า 1 หน้า ไม่มีการสร้างคำว่า “สารบัญ(ต่อ)” คณะผู้จัดทำจึงทำการปรับปรุงแก้ไขคำสั่ง เพื่อให้การแสดงผลของสารบัญมีคำว่า “หน้า” และคำว่า “สารบัญ(ต่อ)” เพื่อให้ถูกต้องตามข้อกำหนดของภาควิชาฯ ดังตัวอย่างนี้

```
\newcommand\tableofcontentmatter{%
\cleardoublepage\@mainmatterfalse
\pagestyle{empty}
\pagestyle{fancy}
\addtolength{\voffset}{2\baselineskip}
```

```

\thead{\hfill\raisebox(-.1cm)[0cm][0cm]
{\LARGE\bfseries\contentsname(ตอ)}\hfill
\raisebox(-0.8cm)[0em][0cm]{\bfseries
\normalsize(หน้า)}}
\renewcommand{\headrulewidth}{0pt}
\renewcommand{\footrulewidth}{0pt}
}

```

- การจัดการเกี่ยวกับระยะห่างระหว่างคำว่าสารบัญและเนื้อหาของสารบัญให้เหมาะสม โดยให้
เว้น 1 บรรทัด ระหว่างสารบัญและเนื้อหาของสารบัญ

```

\def\@chapter[#1]#2{\ifnum\c@secnumdepth>\m@ne
\if@mainmatter
\refstepcounter{chapter}%
\typeout{\@chapapp\space\thechapter.}%
\addcontentsline{toc}{chapter}%
{\protect\@chapapp}\numberline{\thechapter}#1}%
\else
\addcontentsline{toc}{chapter}{#1}%
\fi
\else
\addcontentsline{toc}{chapter}{#1}%
\fi
\chaptermark{#1}%
\addtocontents{lof}{\protect\addvspace{1\p@}}%
\addtocontents{lot}{\protect\addvspace{1\p@}}%
\if@twocolumn
\@topnewpage[\@makechapterhead{#2}]%
\else
\@makechapterhead{#2}%
\@afterheading
\fi
}

```

7. การจัดการเกี่ยวกับตำแหน่งของชื่อบท

ค่าเริ่มต้นของคำสั่ง `\chapter` ในเอกสารประเภท Book Class จะทำให้คำว่า “บทที่” อยู่ชิดด้านซ้ายของหน้ากระดาษ แต่ความต้องการของภาควิชาฯ ระบุว่า คำว่า “บทที่” ต้องอยู่กึ่งกลางหน้ากระดาษ จึงทำการแก้ไข ดังนี้

```
\def\@makechapterhead#1{%
\vspace*{-60\p@}%
{\parindent\z@\centering
\thispagestyle(empty)\normalfont
\ifnum\c@secnumdepth>\m@ne
\Large\bfseries\@chapapp\space
\thechapter\par\nobreak
\vskip0\p@
\fi
\interlinepenalty\@M
\LARGE\bfseries#1\par\nobreak
\vskip25\p@
}}
```

8. การสร้าง คำสั่ง enum_section

หากต้องการสร้างลำดับของข้อความด้วย environment ประเภท enumerate จะได้หมายเลขของลำดับข้อความในลักษณะของ 1, 2, 3, ... ต่อเนื่องกันไป แต่หากต้องการให้หมายเลขลำดับข้อความ อยู่ในลักษณะของ หมายเลขหัวข้อ.หมายเลขหัวข้อย่อย.ลำดับข้อความ ยกตัวอย่างเช่น 1.1.1, 1.1.2, 1.1.3, ... โดยไม่เลือกใช้แพ็คเกจ enumerate คณะผู้จัดทำจึงได้สร้าง environment ใหม่ขึ้น มีชื่อว่า “enum_section” ดังนี้

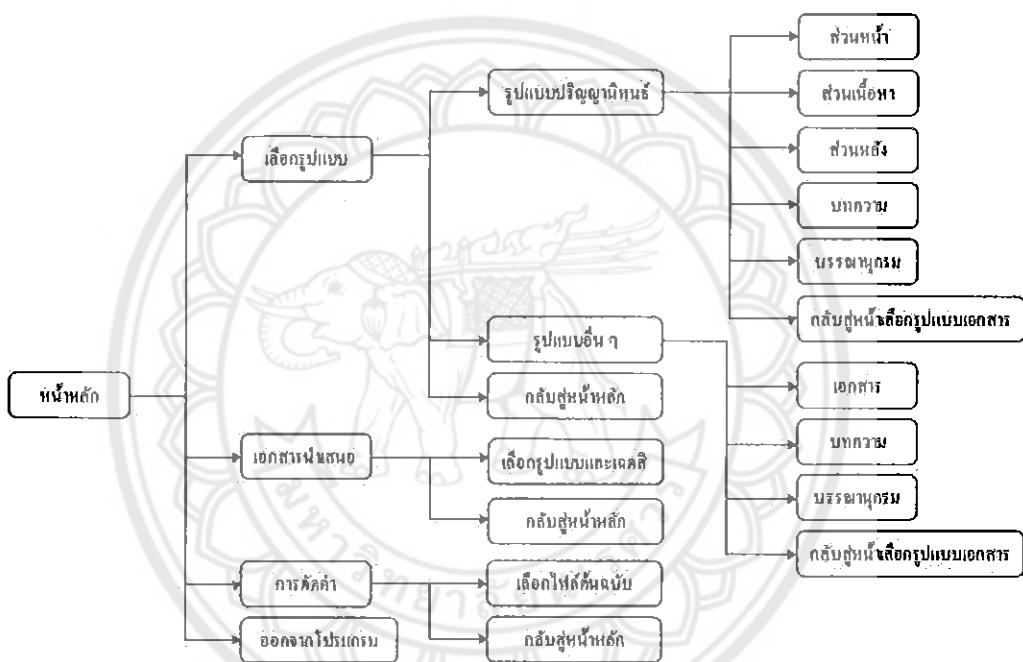
```
\newcounter{lcount}
\newenvironment{enum_section}{
\begin{list}{(\thesection.\arabic{lcount})}
{\usecounter{lcount}
\setlength{\leftmargin}{13ex}}
{\end{list}}
```

บทที่ 4

โปรแกรม GUI และลำดับการทำงาน

ในบทนี้กล่าวถึงขั้นตอนการทำงาน กระบวนการต่าง ๆ ของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ (Graphic User Interface : GUI) เพื่อให้เข้าใจระบบการทำงาน กระบวนการต่าง ๆ ของ GUI ในการสร้างเอกสารปฏิญญาพันธ, เอกสารแนะนำเสนอ, เอกสารในรูปแบบอื่น ๆ และกระบวนการตัดคำ

4.1 แนวคิดในการออกแบบโปรแกรม GUI



รูปที่ 4.1: แผนผังการออกแบบโปรแกรม GUI

ในการออกแบบโปรแกรม GUI แบ่งเป็น 3 ส่วนหลัก คือ การเลือกรูปแบบ การทำเอกสารนำเสนอ และการตัดคำ

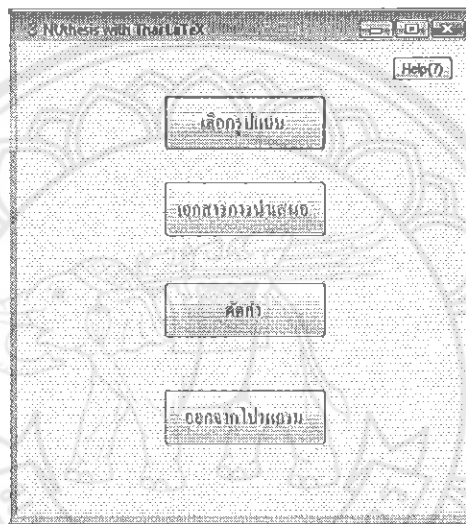
1. การเลือกรูปแบบแบ่งเป็น 2 ส่วน คือ การทำเอกสารปฏิญญาพันธและการทำเอกสารรูปแบบอื่น ๆ ซึ่งมีรายละเอียดดังนี้
 - การออกแบบเอกสารปฏิญญาพันธแบ่งเป็น 5 ส่วนย่อย คือ ส่วนหน้า ส่วนเนื้อหา ส่วนหลัง บทความ และบรรณานุกรม
 - การออกแบบเอกสารรูปแบบอื่น ๆ ในส่วนนี้ได้ออกแบบโปรแกรม GUI ให้ผู้ใช้งานสามารถสร้างเอกสารได้ 3 แบบ คือ เอกสาร บทความ และบรรณานุกรม
2. การสร้างเอกสารการนำเสนอ ในส่วนนี้ได้ออกแบบโปรแกรม GUI ให้ผู้ใช้งานกรอกรายละเอียด เช่น ชื่อหัวข้อภาษาไทย ชื่อหัวข้อภาษาอังกฤษ รูปแบบตัวอักษร ขนาดตัวอักษร ข้อมูลผู้จัดทำ รูปแบบ

เอกสารการนำเสนอ และสีของรูปแบบเอกสารการนำเสนอ เป็นต้น

3. การตัดคำ ในส่วนนี้ได้ออกแบบเพื่ออำนวยความสะดวกต่อผู้ใช้แทนการเขียนคำสั่งใน DOS โดยให้ผู้ใช้งานเลือกไฟล์เอกสารที่ต้องการตัดคำ จากนั้น โปรแกรมจะทำการตัดคำให้ทันที

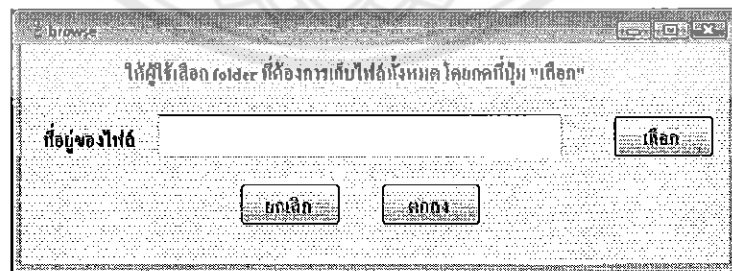
4.2 กระบวนการสร้างไฟล์เอกสาร

เมื่อเปิด โปรแกรมขึ้นมาจะพบหน้าต่าง ดังรูปที่ 4.2 หากผู้ใช้ต้องการสร้างไฟล์เอกสารให้กดที่ปุ่ม “เลือกรูปแบบ”



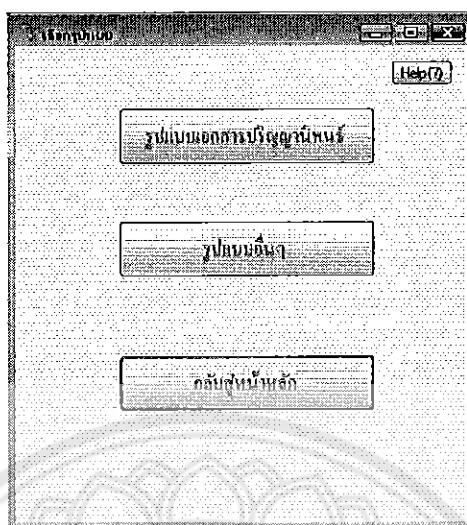
รูปที่ 4.2: แสดงหน้าต่างหลักของโปรแกรม GUI

จากนั้นจะมีหน้าต่างแสดงขึ้นมา ดังรูปที่ 4.3 เพื่อเลือกโฟลเดอร์ที่ใช้เก็บไฟล์ที่ต้องการสร้าง



รูปที่ 4.3: แสดงหน้าต่างการเลือกโฟลเดอร์เพื่อเก็บไฟล์

เมื่อผู้ใช้เลือกโฟลเดอร์ที่ต้องการเป็นที่เรียบร้อยแล้ว ให้ผู้ใช้กดปุ่ม “ตกลง” ซึ่งโปรแกรมจะแสดง หน้า “เลือกรูปแบบ” ดังรูปที่ 4.4

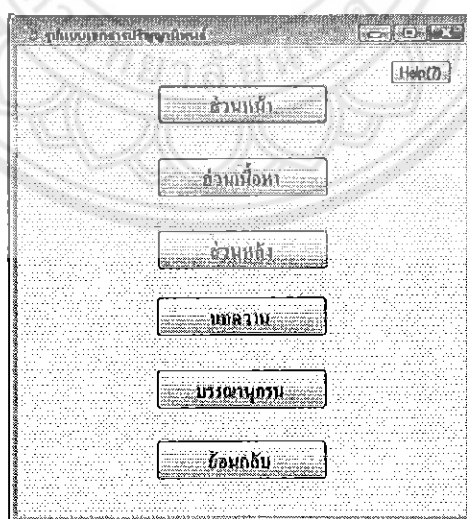


รูปที่ 4.4: แสดงหน้าต่างการเลือกรูปแบบของเอกสาร

โดยหน้านี้จะแบ่งการสร้างเอกสารออกเป็น 2 ประเภท คือ การสร้างรูปแบบเอกสารปริญญาโท และการสร้างเอกสารอื่นๆ

4.2.1 การสร้างเอกสารปริญญาโท

เมื่อผู้ใช้งานต้องการสร้างเอกสารปริญญาโท ให้กดที่ปุ่ม “รูปแบบเอกสารปริญญาโท” ซึ่งจะปรากฏหน้าต่าง ดังรูปที่ 4.5



รูปที่ 4.5: แสดงหน้าต่างการเลือกส่วนต่างๆ ในการทำเอกสารปริญญาโท

โดยแต่ละปุ่มมีรายละเอียด ดังนี้

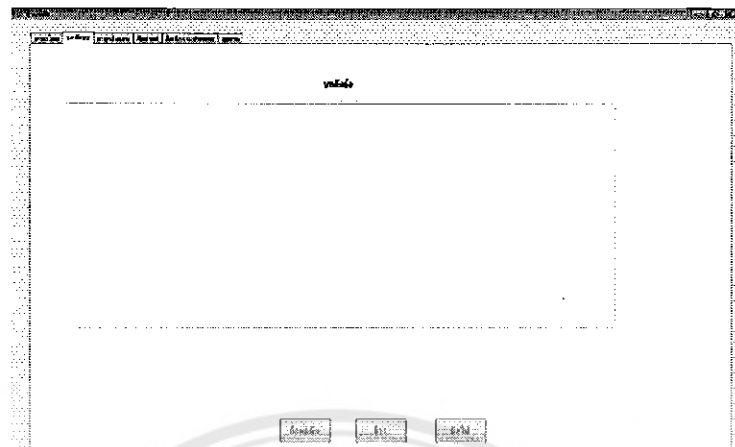
ส่วนหน้า ส่วนนี้ออกแบบเพื่ออำนวยความสะดวกต่อผู้ใช้ในการสร้างเอกสารส่วนหน้าของ
ปริญญาบัตร อันประกอบด้วยปกหน้า ปกใน ใบรับรอง โครงการวิศวกรรม บทคัดย่อ
ภาษาไทย บทคัดย่อภาษาอังกฤษ และกิตติกรรมประกาศ หากผู้ใช้คลิกปุ่ม “ส่วนหน้า”
จะปรากฏหน้าต่าง ดังรูปที่ 4.6

รูปที่ 4.6: แสดงหน้าภาษาไทย

หน้าภาษาไทย ในหน้านี้ กำหนดให้ผู้ใช้กรอกข้อมูลที่จำเป็น อันได้แก่ ชื่อหัวข้อ
โครงการ สาขาวิชา ปีการศึกษาที่ทำโครงการ ชื่อ-สกุล รหัสนิสิต รายชื่ออาจารย์
ที่ปรึกษา และกรรมการ

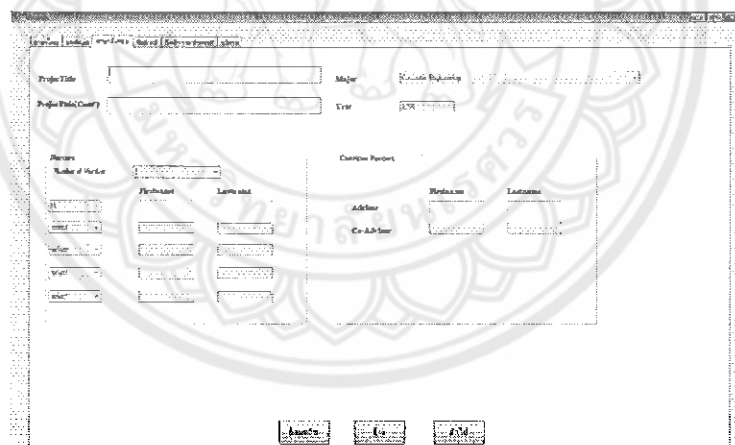
สำหรับในส่วน of ชื่อหัวข้อนั้น หากมีความยาวมากเกินไป ผู้ใช้ควร
แบ่งชื่อหัวข้อออกเป็น 2 บรรทัด โดยบรรทัดแรกให้กรอกในช่อง “ชื่อหัวข้อ”
และบรรทัดที่สองให้กรอกในช่อง “ชื่อหัวข้อ(ต่อ)” ทั้งนี้ผู้ใช้จำเป็นต้องเป็นผู้แบ่งชื่อ
หัวข้อตามความเหมาะสม

เมื่อผู้ใช้กรอกข้อมูลเรียบร้อยแล้ว ให้คลิกปุ่ม “ถัดไป” หรือกด “ล้าง”
หากต้องการลบข้อมูลทั้งหมดที่กรอกในหน้านี้



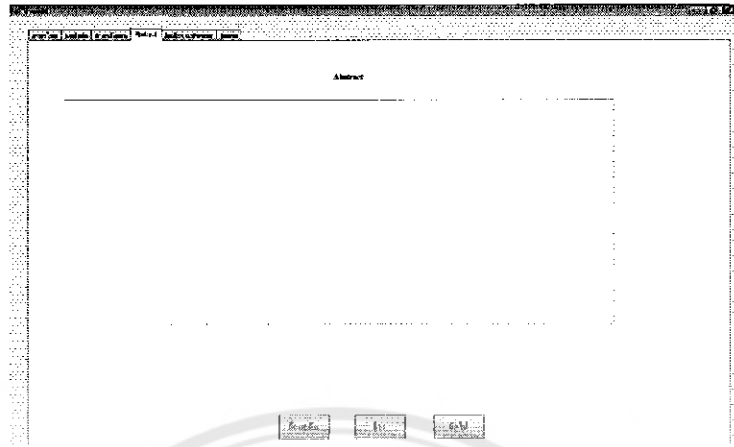
รูปที่ 4.7: แสดงหน้าบทคัดย่อ

บทคัดย่อ ให้ผู้ใช้กรอกเนื้อหาของบทคัดย่อ โดยสามารถใส่สัญลักษณ์หรือคำสั่งของ LaTeX ได้ และเมื่อผู้ใช้กรอกข้อมูลเรียบร้อยแล้ว ให้กดปุ่ม “ถัดไป” หรือกด “ล้าง” หากต้องการลบข้อมูลทั้งหมดที่กรอกในหน้านี้ หากผู้ใช้ต้องการย้อนกลับไปแก้ไขข้อมูลในหน้าก่อนหน้าให้กดปุ่ม “ย้อนกลับ”



รูปที่ 4.8: แสดงหน้าภาษาอังกฤษ

ภาษาอังกฤษ ให้ผู้ใช้กรอกชื่อโครงการ ชื่อ-สกุล ของสมาชิกในกลุ่ม และชื่ออาจารย์ที่ปรึกษาเป็นภาษาอังกฤษ สำหรับข้อมูลอื่น ๆ เช่น ปีการศึกษา คำนำหน้าชื่อและสาขาวิชา โปรแกรมจะนำข้อมูลมาจากหน้า ภาษาไทย โดยตรง เมื่อผู้ใช้กรอกข้อมูลเรียบร้อยแล้ว ให้กดปุ่ม “ถัดไป” หรือกด “ล้าง” หากต้องการลบข้อมูลทั้งหมดที่กรอกในหน้านี้หากผู้ใช้ต้องการย้อนกลับไปแก้ไขข้อมูลในหน้าก่อนหน้าให้กดปุ่ม “ย้อนกลับ”



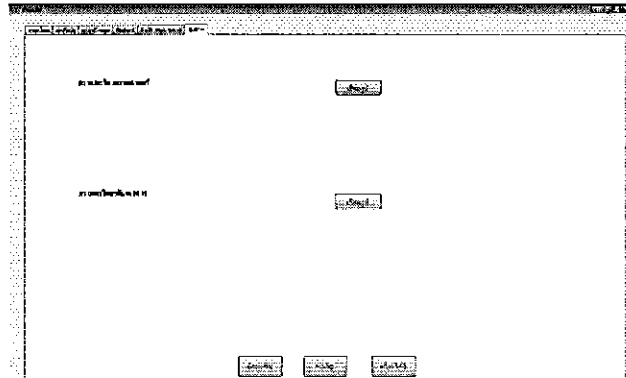
รูปที่ 4.9: แสดงหน้า Abstract

Abstract ให้ผู้ใช้กรอกบทคัดย่อภาษาอังกฤษ โดยสามารถใส่สัญลักษณ์หรือคำสั่งของ LaTeX ได้ และเมื่อผู้ใช้กรอกข้อมูลเรียบร้อยแล้ว ให้กดปุ่ม “ถัดไป” หรือกด “ล้าง” หากต้องการลบข้อมูลทั้งหมดที่กรอกในหน้านี้



รูปที่ 4.10: แสดงหน้ากิตติกรรมประกาศ

กิตติกรรมประกาศ ให้ผู้ใช้กรอกคำคุณูปการหรือคำขอบคุณต่อผู้ให้ความช่วยเหลือและเมื่อผู้ใช้กรอกข้อมูลเรียบร้อยแล้ว ให้กดปุ่ม “ถัดไป” หรือกด “ล้าง” หากต้องการลบข้อมูลทั้งหมดที่กรอกในหน้านี้ หากผู้ใช้ต้องการย้อนกลับไปแก้ไขข้อมูลในหน้าก่อนหน้าให้กดปุ่ม “ย้อนกลับ”

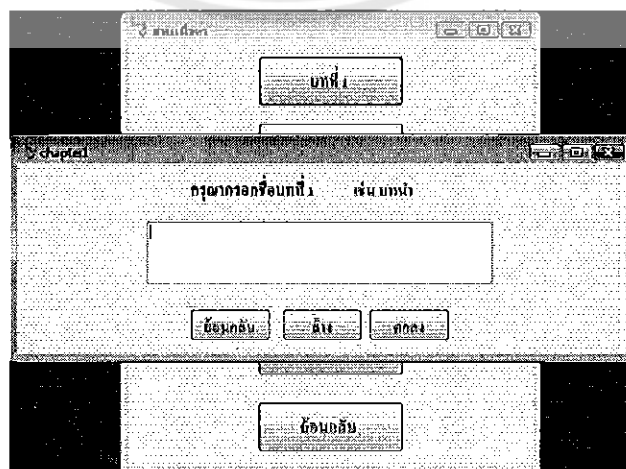


รูปที่ 4.11: แสดงหน้ารูปภาพ

รูปภาพ ให้ผู้ใช้เลือกรูปตราคณะวิศวกรรมศาสตร์และตรามหาวิทยาลัยเพื่อใช้ใส่ที่ปกหน้าและใบรับรอง ครงงานวิศวกรรม ซึ่งมีไฟล์รูปทั้งสองแนบไว้ในแผ่น CD แล้ว

หลังจากที่ผู้ใช้เลือกใส่รูปเป็นที่เรียบร้อยแล้ว ให้คลิกปุ่มสร้างไฟล์ซึ่งโปรแกรม จะทำการสร้างไฟล์ต้นฉบับทั้งหมด 5 ไฟล์ คือ หน้าปก (FontPage.tex), ใบรับรองโครงการ (Approval.tex), บทคัดย่อภาษาไทย (AbstractThai.tex), บทคัดย่อภาษาอังกฤษ (AbstractEng.tex), กิตติกรรมประกาศ (Acknowledgement.tex) และนอกจากนี้ยังมี ไฟล์ที่เกิดจากการตัดคำอีก 5 ไฟล์ คือ FontPage_new.tex, Approval_new.tex, AbstractThai_new.tex, AbstractEng_new.tex, Acknowledgement_new.tex โดยผู้ใช้ สามารถนำไฟล์ที่ผ่านกระบวนการตัดคำแล้วไปใช้งานต่อไป

ส่วนเนื้อหา เมื่อผู้ใช้คลิกปุ่มเลือก "ส่วนเนื้อหา" โปรแกรมจะแสดงหน้าต่างเพื่อให้ผู้ใช้งาน เลือกบทเพื่อใส่ชื่อของแต่ละบท ดังรูปที่ 4.12



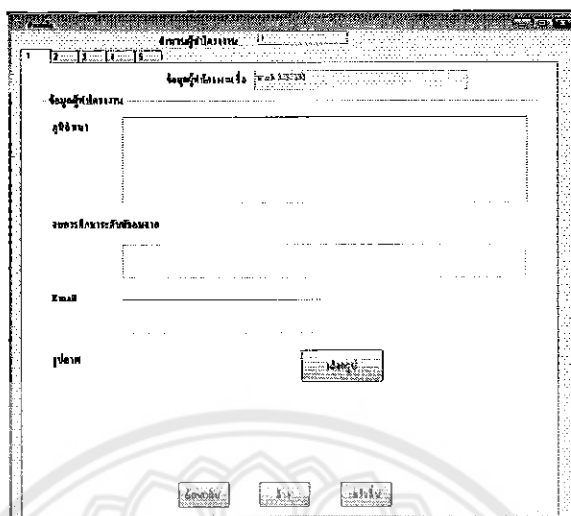
รูปที่ 4.12: แสดงหน้าส่วนเนื้อหาของเอกสารปริญญาบัตร

พร้อมทั้งสร้างไฟล์ Main.tex ซึ่งเป็นไฟล์หลักที่ใช้สำหรับประมวลเอกสารทั้งหมดด้วยวิธีการดึงไฟล์ ที่ได้สร้างขึ้นใน “ส่วนหน้า” อันได้แก่ FrontPage_new.tex, Approval_new.tex, AbstractThai_new.tex, AbstractEng_new.tex, Acknowledgement_new.tex มีการดึงไฟล์ที่สร้างขึ้นจาก “ส่วนหลัง” คือ Antecedents_new รวมทั้งดึงไฟล์เนื้อหา 5 ไฟล์ นั่นคือ chapter1_new.tex, chapter2_new.tex, chapter3_new.tex, chapter4_new.tex และ chapter5_new.tex ไว้ครบถ้วน ตัวอย่างของไฟล์ Main.tex ดังรูปที่ 4.13

```
\documentclass[12pt,oneside,a4paper]{MThesis}
\begin{document}
\thispagestyle{empty}
\input{FrontPage_new}
\newpage
\thispagestyle{empty}
\input{Approval_new}
\newpage
\frontmatter
\addcontentsline{toc}{chapter}{unn"ย"ย"ย"}
\input{AbstractThai_new}
\newpage
\addcontentsline{toc}{chapter}{Abstract}
\input{abstractEnglish_new}
\newpage
\addcontentsline{toc}{chapter}{ก"มท"กสรบประภาด}
\input{Acknowledgement_new}
\newpage
\tableofcontentmatter
\tableofcontents
\listoftablematter
\listoftables
\listoffigurematter
\listoffigures
\mainmatter
\input{chapter1_new}
\input{chapter2_new}
\input{chapter3_new}
\input{chapter4_new}
\input{chapter5_new}
```

รูปที่ 4.13: แสดงเนื้อหาของไฟล์หลัก Main.tex

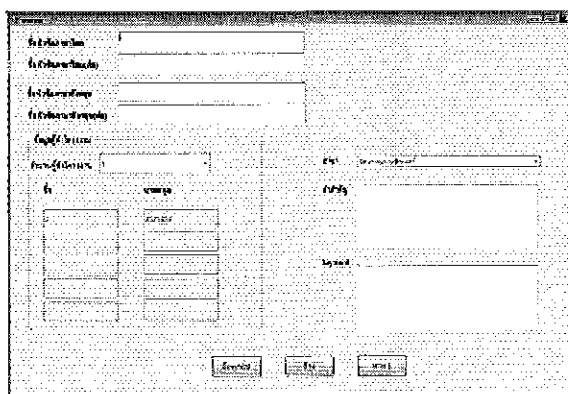
หมายเหตุ : หากผู้ใช้ต้องการเพิ่มหรือลดจำนวนบทในส่วนของเนื้อหาหลัก ให้ผู้ใช้เปิดไฟล์ Main.tex แล้วทำการเพิ่มหรือลบคำสั่ง \input ของบทที่ต้องการเพิ่มหรือลด ส่วนหลัง คือ ส่วนที่เกี่ยวข้องกับประวัติผู้จัดทำโครงการ โดยแสดงลักษณะหน้า GUI ดังรูปที่ 4.14



รูปที่ 4.14: แสดงหน้าส่วนหลังของเอกสารปริญาณิพนธ์

ในส่วนนี้ให้ผู้ใช้กรอกข้อมูลให้ครบถ้วน ประกอบด้วยภูมิลำเนา โรงเรียน ระดับมัธยมศึกษาตอนปลาย จดหมายอิเล็กทรอนิกส์ (E-Mail) และรูปภาพในชุดนิสิต โดยที่จำนวนผู้จัดทำโครงการและรายชื่อของผู้จัดทำโครงการ โปรแกรมจะนำข้อมูลมาจาก “ส่วนหน้า” โดยตรง ดังนั้นผู้ใช้จำเป็นต้องกรอกข้อมูลในส่วนหน้าก่อนถึงจะสามารถเข้ามากรอกข้อมูลในส่วนหลังได้ และเมื่อกรอกข้อมูลเรียบร้อยแล้วให้กดปุ่ม “เสร็จสิ้น” จากนั้นโปรแกรมจะสร้างไฟล์ Antecedents.tex และ Antecedents_new.tex ซึ่งเป็นไฟล์ที่มีการตัดค่าเรียบร้อยแล้ว

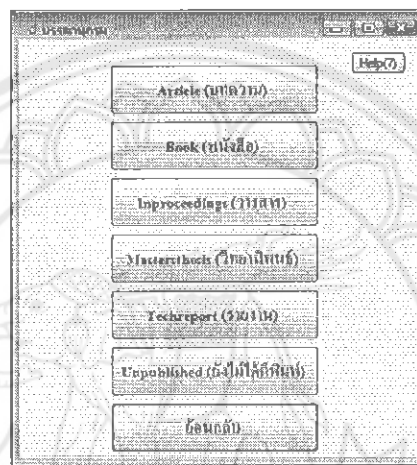
บทความ เมื่อผู้ใช้ต้องการสร้างไฟล์บทความเชิงวิชาการที่ต้องนำส่งพร้อมเอกสารปริญาณิพนธ์ ให้เลือก “บทความ” โดยจะปรากฏหน้าต่าง GUI ดังแสดงในรูปที่ 4.15 ซึ่งข้อมูลบางส่วนอาศัยการส่งผ่านจากข้อมูลที่กรอกใน “ส่วนหน้า” อันได้แก่ จำนวนผู้จัดทำโครงการ รายชื่อของผู้จัดทำ และชื่อสาขาวิชาที่ตนสังกัด



รูปที่ 4.15: แสดงหน้าบทความ

หลังจากที่ผู้ใช้กรอกข้อมูลครบถ้วนแล้ว โปรแกรมจะทำการสร้างไฟล์ Article.tex และ Article_new.tex จากนั้นให้ผู้ใช้พิมพ์เนื้อหาที่ต้องการลงในไฟล์ Article.tex บันทึกแล้วนำไปตัดคำ (ดูหัวข้อ 4.4 หน้า 45) ซึ่งเมื่อประมวลผลด้วย L^AT_EX แล้ว จะได้บทความที่มีรูปแบบตามที่ภาควิชาฯ กำหนด

บรรณานุกรม เมื่อผู้ใช้ต้องการอ้างอิงเอกสารต่าง ๆ ที่ใช้ประกอบการทำวิทยานิพนธ์ ให้ผู้ใช้เลือกบรรณานุกรม ซึ่งจะแสดงหน้าต่าง GUI ดังรูปที่ 4.16 เพื่อให้ผู้ใช้เลือกที่จะทำการอ้างอิงเอกสารประเภทใด



รูปที่ 4.16: แสดงหน้าบรรณานุกรม

หากเอกสารที่นำมาอ้างอิงเป็นวิทยานิพนธ์ ให้ผู้ใช้คลิกปุ่ม “Masterstheis (วิทยานิพนธ์)” ซึ่งจะปรากฏหน้าต่าง ดังรูปที่ 4.17

รูปที่ 4.17: แสดงหน้าบรรณานุกรมประเภทวิทยานิพนธ์

ผู้ใช้งานเป็นค้องกรอกข้อมูลในช่องที่มีเครื่องหมาย * ให้ครบทุกช่อง โดยที่

- ช่อง Reference ไว้ใส่คำอ้างอิงที่ผู้ใช้งานไปอ้างอิงในเนื้อหาของเอกสาร
- ช่อง Author ไว้ใส่ชื่อผู้แต่งวิทยานิพนธ์
- ช่อง Title ไว้ใส่ชื่อเรื่องของวิทยานิพนธ์
- ช่อง School ไว้ใส่ชื่อสถาบันหรือมหาวิทยาลัยที่ผู้แต่งสำเร็จการศึกษา
- ช่อง Year ไว้ใส่ปี พ.ศ. ที่จัดพิมพ์

เมื่อผู้ใช้งานกรอกข้อมูลสำหรับเอกสารอ้างอิงเป็นที่เรียบร้อยแล้ว ให้กดที่ปุ่ม “ตกลง” โปรแกรมจะคงหน้าต่างเดิมเพื่อให้ผู้ใช้งานกรอกข้อมูลสำหรับเอกสารอ้างอิงประเภทเดิม แต่หากผู้ใช้งานต้องการเปลี่ยนประเภทเอกสารอ้างอิง ให้ผู้ใช้งานปุ่ม “ย้อนกลับ”

เมื่อผู้ใช้งานใส่เอกสารอ้างอิงจนครบแล้วให้กดปุ่ม “ย้อนกลับ” ของหน้าต่างรูปที่ 4.16 โดยโปรแกรมจะแสดงหน้าต่างหลัก ดังรูปที่ 4.5 ในขณะที่เดียวกัน โปรแกรมจะทำการสร้างไฟล์ขึ้นมาชื่อว่า bibliography.bib ตัวอย่างของไฟล์ bibliography.bib แสดงดังรูปที่ 4.18

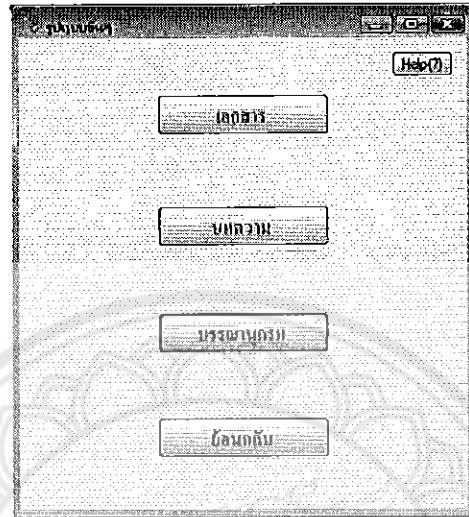
```
@BOOK{1,
author = {กิตตินันท์ พลสวัสดิ์},
title = {เริ่มต้น Visual Basic 2008 ฉบับ
โปรแกรมเมอร์},
publisher = {บริษัท ไรต์ซี อีแอนด์ไอ.ลิสทริบิวเตอร์ เซ็นเตอร์
จำกัด},
year = {2552},
edition = {1},
series = {},
volume = {},
number = {},
address = {},
month = {},
language = {},
note = {},
}
```

รูปที่ 4.18: แสดงไฟล์บรรณานุกรม

หมายเหตุ : ประเภทของเอกสารอ้างอิงที่ปรากฏในหน้าต่างบรรณานุกรม ดังรูปที่ 4.16 เป็นเอกสารที่นิยมใช้ในการอ้างอิง หากผู้ใช้งานต้องการอ้างอิงถึงเอกสารประเภทอื่นนอกเหนือที่กำหนดให้ ให้ผู้ใช้งานเปิดไฟล์ bibliography.bib แล้วทำการแทรกข้อมูลของเอกสารอ้างอิงที่ต้องการลงไปตามรูปแบบของ bibtex

4.2.2 การสร้างไฟล์เอกสารรูปแบบอื่น ๆ

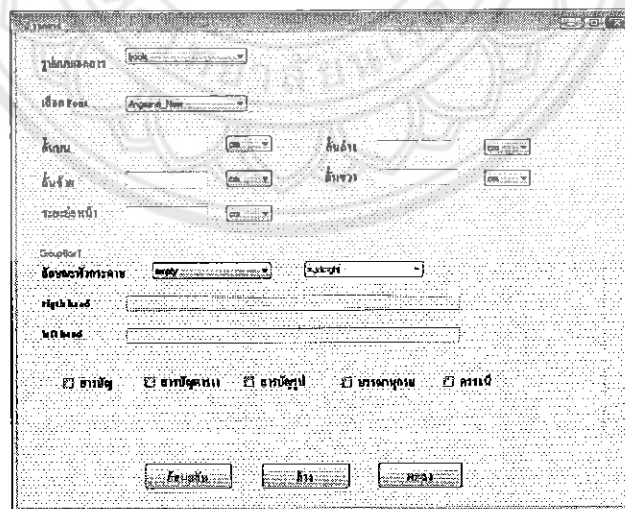
ถ้าผู้ใช้คลิกเลือกปุ่ม “รูปแบบอื่น ๆ” โปรแกรมจะแสดงหน้าต่าง ดังรูปที่ 4.19



รูปที่ 4.19: แสดงหน้าต่างของเอกสารทั่วไป

ในส่วนของรูปแบบอื่น ๆ ผู้ใช้สามารถเลือกปุ่มเอกสาร, บทความ หรือบรรณานุกรม ได้ โดยมีรายละเอียด ดังนี้

เอกสาร โดยลักษณะหน้าเอกสาร แสดงดังรูปที่ 4.20



รูปที่ 4.20: แสดงหน้าการสร้างเอกสารทั่วไป

ส่วนนี้กำหนดให้ผู้ใช้กรอกข้อมูลที่จำเป็นอัน ได้แก่เลือกรูปแบบเอกสารเลือกแบบตัวอักษร ระยะกั้นบน ระยะกั้นล่าง ระยะกั้นซ้าย ระยะกั้นขวาและระยะย่อหน้า หากผู้ใช้ไม่กำหนดขนาดต่าง ๆ ที่กล่าวมาข้างต้น โปรแกรมจะทำการกำหนดระยะให้เป็น 0 เซนติเมตร โดยอัตโนมัติ จากนั้นให้ผู้ใช้เลือกแบบหน้ากระดาษ (Page Style)

และหากผู้ใช้ต้องการให้ปรากฏสารบัญ สารบัญตาราง สารบัญรูป บรรณานุกรมหรือ
ครรรชนี ให้ผู้ใช้ทำเครื่องหมาย ✓ หน้าช่องต่าง ๆ ที่กล่าวมาข้างต้น

เมื่อผู้ใช้กรอกข้อมูลครบถ้วน ให้ผู้ใช้คลิกปุ่ม “ตกลง” จะได้ไฟล์ที่มีชื่อว่า Main-
Other.tex ซึ่งเป็นไฟล์หลักในการประมวลผล

บทความ ส่วนนี้กำหนดให้ผู้ใช้กรอกข้อมูลที่จำเป็นอัน ได้แก่ เลือกรูปแบบเอกสาร เลือก
แบบตัวอักษร ระยะกั้นบน ระยะกั้นล่าง ระยะกั้นซ้าย ระยะกั้นขวาและระยะย่อหน้า
หากผู้ใช้ไม่กำหนดขนาดต่าง ๆ ที่กล่าวมาข้างต้น โปรแกรมจะทำการกำหนดระยะให้
เป็น 0 เซนติเมตร โคยอต์ โนมติ และหากผู้ใช้ต้องการให้ปรากฏบรรณานุกรมให้ผู้ใช้
ทำเครื่องหมาย ✓ หน้าช่องบรรณานุกรม

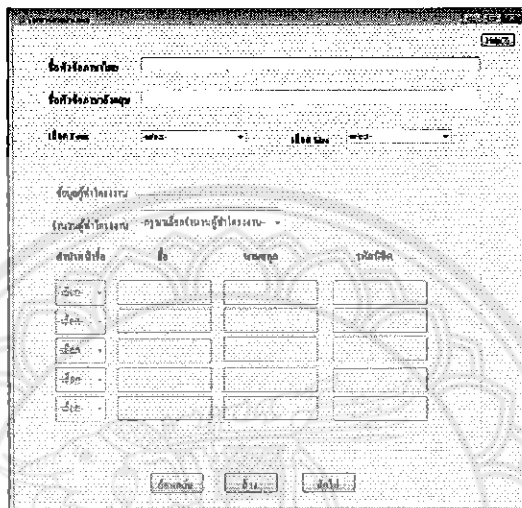
เมื่อผู้ใช้กรอกข้อมูลครบถ้วน ให้ผู้ใช้คลิกปุ่ม “ตกลง” จะได้ไฟล์ที่มีชื่อว่า Main-
Other.tex ซึ่งเป็นไฟล์หลักในการประมวลผล

บรรณานุกรม ลักษณะการทำงานในส่วนนี้มีลักษณะเดียวกับหน้าบรรณานุกรมในส่วนของ
เอกสารปริญญาโท (ดูรูปที่ 4.16 หน้า 38)



4.3 การสร้างไฟล์เอกสารนำเสนอ

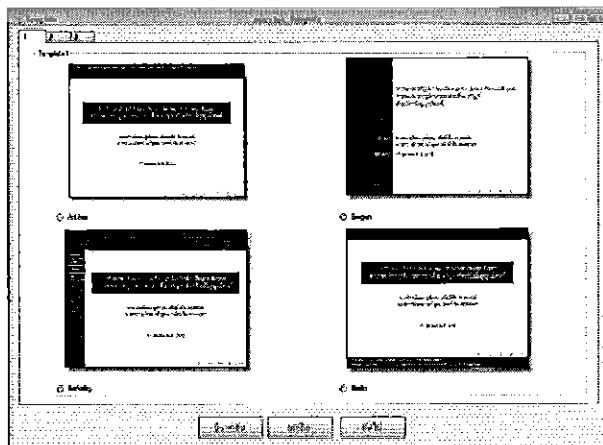
จากหน้าหลักของโปรแกรม GUI (ดูรูป 4.2 หน้า 30) หากผู้ใช้คลิกปุ่มเลือก “เอกสารนำเสนอ” โปรแกรมจะให้ผู้ใช้เลือกไฟล์เดสก์ทอปที่เก็บไฟล์ที่ต้องการสร้างขึ้น ในขั้นตอนนี้มีลักษณะเหมือนกับหลักการสร้างเอกสารปริ้นท์ เมื่อผู้ใช้คลิกปุ่ม “ตกลง” โปรแกรมจะแสดงหน้าต่าง ดังรูปที่ 4.21



รูปที่ 4.21: แสดงหน้าต่างเอกสารนำเสนอ

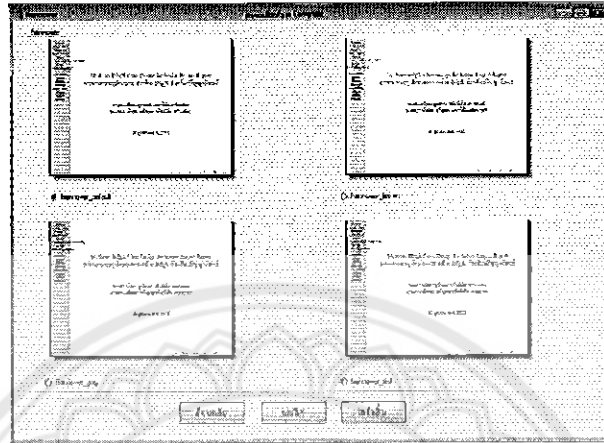
ในหน้านี้ กำหนดให้ผู้ใช้กรอกข้อมูลที่จำเป็น อันได้แก่ ชื่อหัวข้อโครงการภาษาไทย ชื่อหัวข้อโครงการภาษาอังกฤษ ชื่อ-สกุล รหัสชนิด รวมถึงเมื่อผู้ใช้กรอกข้อมูลเรียบร้อยแล้ว ให้คลิกปุ่ม “ถัดไป” หรือกด “ซ้าย” หากต้องการลบเนื้อหาทั้งหมดในหน้านี้ และหากผู้ใช้ต้องการย้อนกลับ ไปหน้าหลักของโปรแกรม GUI ให้ผู้ใช้คลิกปุ่ม “ย้อนกลับ”

หากผู้ใช้คลิกปุ่ม “ถัดไป” โปรแกรมจะแสดงหน้าต่าง เพื่อเลือกรูปแบบของเอกสารนำเสนอ ดังรูปที่ 4.22



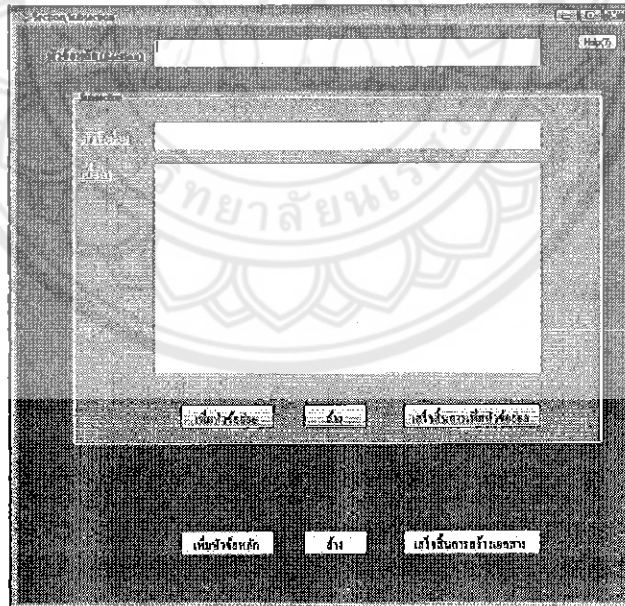
รูปที่ 4.22: แสดงหน้าธีม (Theme) ของเอกสารนำเสนอ

โปรแกรมได้กำหนดหน้าธีม (Theme) ของเอกสารสื่อนำเสนอไว้ทั้งหมด 12 แบบ เมื่อผู้ใช้เลือกรูปแบบที่ต้องการแล้ว ให้คลิกปุ่ม “ถัดไป” เพื่อทำการเลือกสีของรูปแบบ ดังรูปที่ 4.23



รูปที่ 4.23: แสดงหน้าสำหรับเลือกสีของรูปแบบ

เมื่อเลือกสีที่ต้องการแล้ว ให้คลิกปุ่ม “เสร็จสิ้น” โปรแกรมจะแสดงหน้าต่างต่าง ดังรูปที่ 4.24 เพื่อให้ผู้ใช้กรอกเนื้อหาสำหรับเอกสารสื่อนำเสนอ



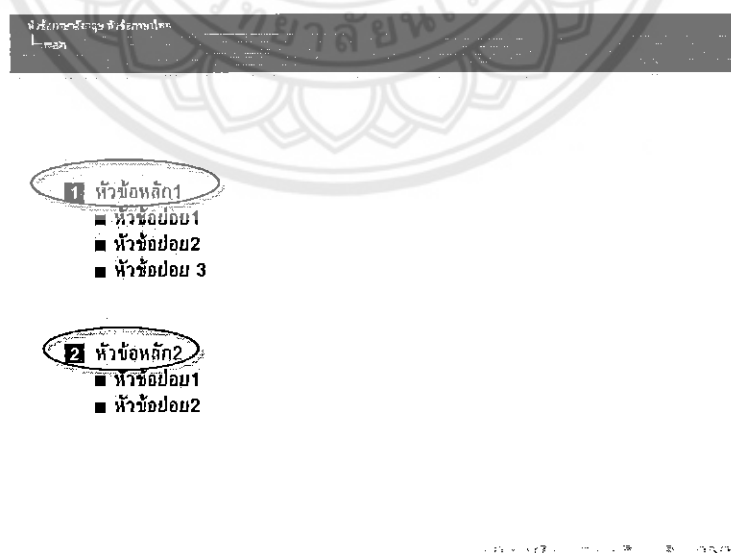
รูปที่ 4.24: แสดงหน้าการใส่เนื้อหาของเอกสารสื่อนำเสนอ

โดยในหน้านี้ ผู้ใช้สามารถใส่เนื้อหาของเอกสารสื่อนำเสนอได้ โดยหน้าต่างจะมีช่องให้ใส่เนื้อหาอยู่ 3 ช่อง คือ

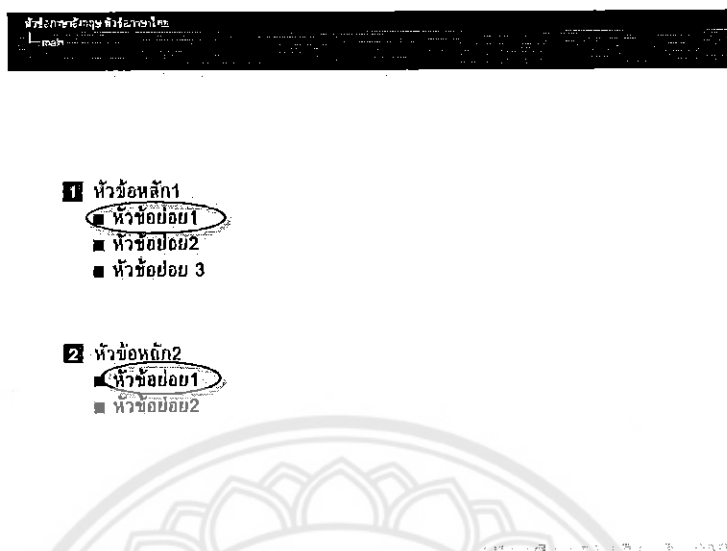
- หัวข้อหลัก (Section) ให้ผู้ใช้ใส่ชื่อหัวข้อหลัก
- หัวข้อย่อย (Subsection) ให้ผู้ใช้ใส่ชื่อหัวข้อย่อย

- เนื้อหา ให้ผู้ใช้ใส่เนื้อหา
- เพิ่มหัวข้อย่อย เมื่อผู้ใช้ต้องการเพิ่มหัวข้อย่อย
- ล้าง หากผู้ใช้ต้องการลบเนื้อหาทั้งหมดในหน้านี้
- เสร็จสิ้นการเพิ่มหัวข้อย่อย เมื่อผู้ใช้กรอกข้อมูลและต้องการเสร็จสิ้นการเพิ่มหัวข้อย่อย
- เพิ่มหัวข้อหลัก เมื่อผู้ใช้ต้องการเพิ่มหัวข้อหลัก
- ล้าง หากผู้ใช้ต้องการลบเนื้อหาทั้งหมดในหน้านี้
- เสร็จสิ้นการเพิ่มหัวข้อหลัก เมื่อผู้ใช้ต้องการเสร็จสิ้นการเพิ่มหัวข้อหลักและสิ้นสุดการสร้างเอกสาร นำเสนอ

โดยปกติในหนึ่งหัวข้อหลักอาจมีหลายหัวข้อย่อย ดังนั้นเมื่อผู้ใช้ใส่หัวข้อหลัก หัวข้อย่อยและเนื้อหา ตามลำดับ ผู้ใช้สามารถเพิ่มหัวข้อย่อยให้กับหัวข้อหลักนี้ได้โดยคลิกปุ่ม “เพิ่มหัวข้อย่อย” ซึ่งจะปรากฏหน้าต่างใหม่ขึ้นมา โดยที่ชื่อหัวข้อหลักจะยังเป็นชื่อเดิมและโปรแกรมจะไม่ให้ผู้ใช้แก้ไขชื่อหัวข้อหลักได้ และเมื่อผู้ใช้ใส่เนื้อหาในส่วนของหัวข้อหลักนี้เป็นที่เรียบร้อยแล้ว ให้คลิกปุ่ม “เสร็จสิ้นการเพิ่มหัวข้อย่อย” จากนั้นจะปรากฏหน้าต่างใหม่ให้ผู้ใช้เพิ่มหัวข้อหลัก หัวข้อย่อย และเนื้อหาได้ต่อไป เมื่อผู้ใช้ทำการใส่เนื้อหาทั้งหมดเป็นที่เรียบร้อยแล้ว ให้ผู้ใช้คลิกปุ่ม “เสร็จสิ้นการสร้างเอกสาร” เพื่อสิ้นสุดการสร้างเอกสาร นำเสนอ และจะได้ไฟล์เอกสารนำเสนอชื่อว่า Presentation.tex และ Presentation_new.tex เพื่อใช้ในการประมวลผลต่อไป ตัวอย่างเอกสารนำเสนอแสดงในรูปที่ 4.25 และ 4.26



รูปที่ 4.25: แสดงตัวอย่างหัวข้อหลักในเอกสารนำเสนอ

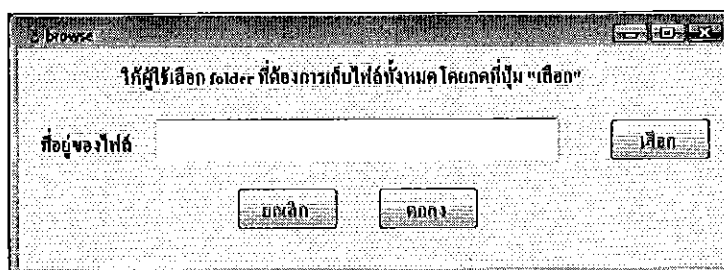


รูปที่ 4.26: แสดงตัวอย่างหัวข้อย่อยในเอกสารแนะนำเสนอ

4.4 กระบวนการตัดคำ

เนื่องจากการเขียนคำสั่งเพื่อให้โปรแกรม SWATH หรือโปรแกรม CTTEX ทำการใส่ \wbr กับเอกสารต้นฉบับนั้น ต้องทำการเขียนคำสั่งใน DOS ซึ่งมีความยุ่งยากซับซ้อนพอสมควร ดังนั้นคณะผู้จัดทำจึงสร้างโปรแกรม GUI เพื่ออำนวยความสะดวกต่อผู้ใช้งานทั่วไป

จากหน้าหลักของโปรแกรม GUI (ดูรูป 4.2 หน้า 30) หากผู้ใช้งานกดปุ่ม “ตัดคำ” โปรแกรมจะแสดงหน้าต่าง ดังรูปที่ 4.27 จากนั้นให้ผู้ใช้กดปุ่ม “เลือก” เพื่อเลือกไฟล์เอกสาร .tex ที่ต้องการตัดคำแล้วกดปุ่ม “ตกลง” ซึ่งโปรแกรมจะทำการการสร้างไฟล์ .bat ที่มีการเขียนคำสั่งลงไปให้เรียกโปรแกรม SWATH ขึ้นมาโดยอัตโนมัติเพื่อทำการแทรกคำสั่ง \wbr ในไฟล์เอกสารที่เลือกไว้ข้างต้น แล้วทำการบันทึก (Save) เป็นไฟล์ใหม่ที่มี _new ต่อท้ายชื่อไฟล์เอกสารเดิม ยกตัวอย่างเช่น หากไฟล์เอกสารต้นฉบับที่ต้องการตัดคำชื่อว่า chapter1.tex หลังจากใช้โปรแกรมนี้ตัดคำแล้วจะได้ไฟล์ใหม่ที่มีชื่อว่า chapter1_new.tex ขึ้นมา



รูปที่ 4.27: แสดงหน้าต่างการตัดคำ

บทที่ 5

สรุปผลการดำเนินงานและแนวทางการพัฒนา

5.1 สรุปผลการดำเนินงาน

ในโครงการนี้ได้จัดทำและพัฒนา โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ (GUI) เพื่อการใช้งาน $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ภาษาไทยและอำนวยความสะดวกต่อผู้ใช้งานทั่วไป ทั้งในส่วนของกรจัดทำเอกสารประเภทสิ่งพิมพ์และประเภทสื่อนำเสนอ นอกจากนี้คณะผู้จัดทำได้พัฒนาประเภทเอกสารที่มีชื่อว่า NUthesis Class เพื่ออำนวยความสะดวกให้กับนิสิตในภาควิชาฯ เพื่อใช้จัดทำปริญญาบัตรให้ถูกต้องตามข้อกำหนดของภาควิชาฯ อีกด้วย

5.2 แนวทางการพัฒนา

- 5.2.1 ในการพัฒนาโครงการควรใช้โปรแกรม XeTeX แทนโปรแกรม MikTeX เพื่อเพิ่มขีดจำกัดความสามารถของโปรแกรมให้สามารถรองรับภาษาไทยได้โดยไม่ต้องทำการติดตั้งโปรแกรมสนับสนุนการใช้ภาษาไทยเพิ่มเติม
- 5.2.2 ควรทำการศึกษาและพัฒนารูปแบบ โครงสร้าง Document Class สำหรับบทความ
- 5.2.3 ในส่วนของการจัดทำเอกสาร สื่อนำเสนอที่ได้จัดทำขึ้นในโครงการนี้มีข้อจำกัด กล่าวคือสามารถจัดทำเอกสารสื่อนำเสนอสำหรับโครงการได้เท่านั้น ดังนั้นผู้สนใจสามารถนำโครงการนี้มาพัฒนาส่วนของเอกสารสื่อนำเสนอสำหรับการนำเสนอแบบทั่วไป เพื่อเพิ่มความหลากหลายในการออกแบบสื่อการนำเสนอ
- 5.2.4 ควรทำการพัฒนาให้ผู้ใช้งานสามารถระบุจำนวนบทตามที่ใช้ต้องการจากนั้นปรากฏหน้าต่างแสดงจำนวนบทตามที่ระบุ และผู้ใช้สามารถกรอกชื่อประจำบทได้ทันที (ศึกษารูปที่ 4.5 หน้า 31 และรูปที่ 4.12 หน้า 35 ประกอบ)
- 5.2.5 เนื่องจาก GUI คณะผู้จัดทำสร้างขึ้นจากโปรแกรม VB.NET เป็นโปรแกรมที่ได้รับการรองรับจากระบบปฏิบัติการวินโดวส์ (Windows) ทำให้ผู้ใช้งานที่ใช้ระบบปฏิบัติการอื่นไม่สามารถใช้งานได้ ดังนั้นจึงควรมีการพัฒนาให้ใช้งาน GUI นี้บน Web Based ได้ซึ่งจะทำให้ผู้ใช้งานบนระบบปฏิบัติการอื่นสามารถใช้งานได้ง่ายและสะดวกมากยิ่งขึ้น
- 5.2.6 ควรทำการพัฒนาในส่วนของกรจัดทำเอกสารสื่อนำเสนอให้ผู้ใช้งานสามารถทำการอัปโหลด (Upload) แม่แบบ (Template) ของผู้ใช้งานเองได้ โดยในหน้าต่างที่ให้ผู้เลือกใช้ Template นั้น ควรเพิ่มเติมช่องให้ผู้ใช้งานสามารถทำการอัปโหลดไฟล์ Template ที่มีนามสกุล .sty จากนั้นโปรแกรมจะทำการส่งไฟล์ดังกล่าวไปยังที่อยู่ที่ใช้เก็บไฟล์ Template ของ MikTeX ซึ่งจะทำให้ผู้ใช้งานสามารถเรียกใช้งาน Template ของตนเองได้

บรรณานุกรม

- [1] จักรกฤษณ์ วิสวกุล. บทแนะนำ $\text{\LaTeX}2\epsilon$ ฉบับไม่ค่อยย่อ. Available : <http://linux.thai.net/projects/thailatex/>, 2545.
- [2] ชันยา ภู่วโรดม. การออกแบบรูปแบบเอกสารสำหรับปริญญาานิพนธ์ด้วย โปรแกรมสำเร็จ \LaTeX ภาษาไทย. มหาวิทยาลัยนเรศวร, 2551.
- [3] มหาวิทยาลัยนเรศวร. คู่มือการเขียนรายงาน โครงการงานวิศวกรรม. ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร, 2547.
- [4] ดร.ประมวล สุธีจารูวัฒน. การผลิตเอกสารวิชาการด้านวิทยาศาสตร์และวิศวกรรมด้วย $\text{\LaTeX}2\epsilon$. ภาควิชาวิศวกรรมอุตสาหกรรม คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2547.



ประวัติผู้เขียนโครงการ



ชื่อ นางสาวนัตตา รุ่ยหอม
 ภูมิลำเนา 37 ม.12 ต.แหลมรัง อ.บึงนาราง จ.พิจิตร
 - จบมัธยมศึกษาจาก โรงเรียนนครสวรรค์
 - ปัจจุบันกำลังศึกษาชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์
 ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
 มหาวิทยาลัยนเรศวร

E-mail : para_2b@hotmail.com



ชื่อ นางสาวนัสพร เจริญผล
 ภูมิลำเนา 239 ม.3 ต.แม่พูล อ.ลับแล จ.อุตรดิตถ์
 - จบมัธยมศึกษาจาก โรงเรียนอุตรดิตถ์ศรีธานี
 - ปัจจุบันกำลังศึกษาชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์
 ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
 มหาวิทยาลัยนเรศวร

E-mail : benz_t8@hotmail.com