



โปรแกรมช่วยพิมพ์ภาษาญี่ปุ่นโดยใช้ตัวอักษรภาษาไทย

A Japanese character typing assistant using Thai alphabets



นางสาวกอบพร	เชื้อบุญมี	รหัส	47361803
นางสาวมณฑนา	เนียมคำ	รหัส	47362090
นางสาวสลักใจ	ธรรมรักษา	รหัส	47362215

ห้องสมุดคณะวิศวกรรมศาสตร์
25 พ.ค. 2553
วันที่รับ.....
เลขทะเบียน..... 15009340
เลขเรียกหนังสือ..... 0565.1
มหาวิทยาลัยนเรศวร 2550

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
ปีการศึกษา 2550



ใบรับรองโครงการวิศวกรรม

หัวข้อโครงการ โปรแกรมช่วยพิมพ์ภาษาญี่ปุ่นโดยใช้ตัวอักษรภาษาไทย
ผู้ดำเนินโครงการ นางสาวกอบพร เชื้อบุญมี รหัส 47361803
นางสาวมันทนา เนียมคำ รหัส 47362090
นางสาวสลักใจ ธรรมรักษา รหัส 47362215
อาจารย์ที่ปรึกษา คร. สุรเดช จิตประไพกุลศาล
สาขาวิชา วิศวกรรมคอมพิวเตอร์
ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา 2550

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสุรินทร์ อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของ
การศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์
คณะกรรมการสอบ โครงการวิศวกรรม

.....ประธานกรรมการ
(ดร.สุรเดช จิตประไพกุลศาล)

.....กรรมการ
(ดร.อัครพันธ์ วงศ์กั้งแห)

.....กรรมการ
(อาจารย์กาญจนาพงศ์ สอนคม)

หัวข้อโครงการ	โปรแกรมช่วยพิมพ์ภาษาญี่ปุ่นโดยใช้ตัวอักษรภาษาไทย
ผู้ดำเนินโครงการ	นางสาวกอบพร เชื้อบุญมี รหัส 47361803
	นางสาวมณฑนา เนียมคำ รหัส 47362090
	นางสาวสลักใจ ธรรมรักษา รหัส 47362215
อาจารย์ที่ปรึกษา	ดร.สุรเดช จิตประไพกุลศาล
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2550

บทคัดย่อ

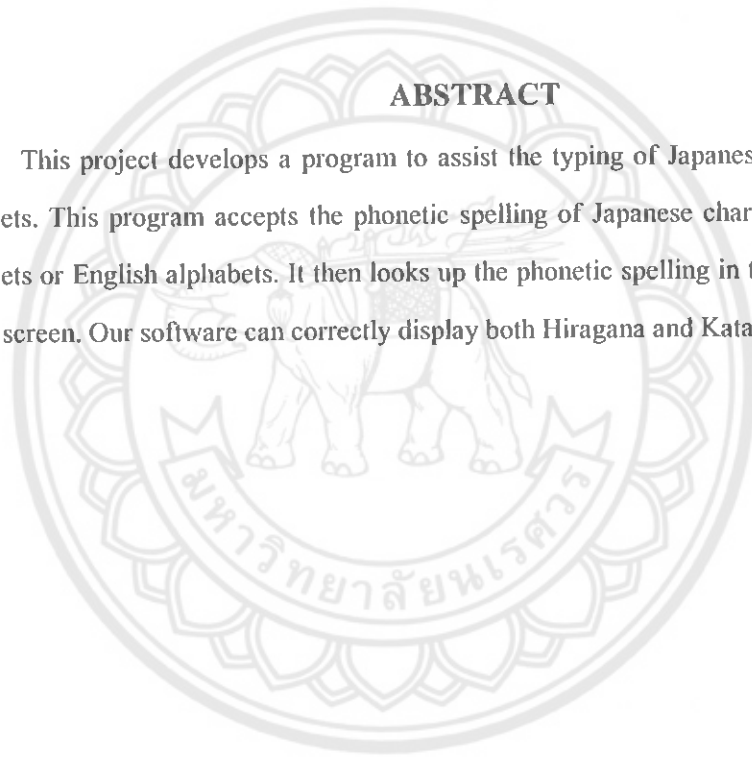
โครงการนี้พัฒนาต้นแบบสำหรับการพิมพ์ภาษาญี่ปุ่นจากคำอ่านที่สะกดเป็นภาษาไทยหรือภาษาอังกฤษได้ ในขั้นต้นจะรับคำตัวอักษรภาษาไทยหรือภาษาอังกฤษจากคีย์บอร์ด จากนั้นจะนำไปเปรียบเทียบกับฐานข้อมูลซึ่งเก็บตัวอักษรภาษาญี่ปุ่นไว้ แล้วนำตัวอักษรภาษาญี่ปุ่นมาแสดงผลที่หน่วยแสดงผล Graphic User Interface (GUI) ของโปรแกรม ไม่ว่าจะเป็นการรับตัวสะกดในภาษาอังกฤษหรือในภาษาไทย โปรแกรมสามารถแสดงผลตัวอักษรภาษาญี่ปุ่นได้ทั้งแบบฮิรางานะและคาตากานะ และยังสามารถพิมพ์ภาษาไทยหรือภาษาอังกฤษแบบไม่มีการแปลงเป็นภาษาญี่ปุ่นได้

Project Title A Japanese character typing assistant using Thai alphabets
Name Miss Kobporn Chuaboonmee ID. 47361803
 Miss Mantana NeamKam ID. 47362090
 Miss Salakjai Thammaraksa ID. 47362215
Project Advisor Dr. Suradet Jitprapaikulsarn
Major Computer Engineering
Department Electrical and Computer Engineering
Academic Year 2007

.....

ABSTRACT

This project develops a program to assist the typing of Japanese characters using Thai alphabets. This program accepts the phonetic spelling of Japanese characters using either Thai alphabets or English alphabets. It then looks up the phonetic spelling in the database and display on the screen. Our software can correctly display both Hiragana and Katakana characters.



กิตติกรรมประกาศ

ขอขอบพระคุณ ดร.สุรเดช จิตประไพกุลศาสตราจารย์ที่ปรึกษาโครงการนี้ ที่คอยให้คำปรึกษา ความช่วยเหลือตลอดจนคำแนะนำต่างๆ ในการทำโครงการนี้

ขอขอบพระคุณ ดร.อักรพันธ์ วงศ์กังแห และอาจารย์ภาณุพงศ์ สอนคม กรรมการคุมสอบโครงการ ที่กรุณาให้คำปรึกษาแนะนำ และช่วยตรวจสอบแก้ไขข้อบกพร่องของโครงการนี้

สุดท้ายนี้ขอขอบพระคุณอาจารย์ทุกท่านและทุกๆ คนที่ยังไม่ได้เอ่ยนามที่คอยให้การสนับสนุนผู้ดำเนินโครงการ ให้สามารถทำโครงการนี้จนสำเร็จลุล่วงไปได้ด้วยดี

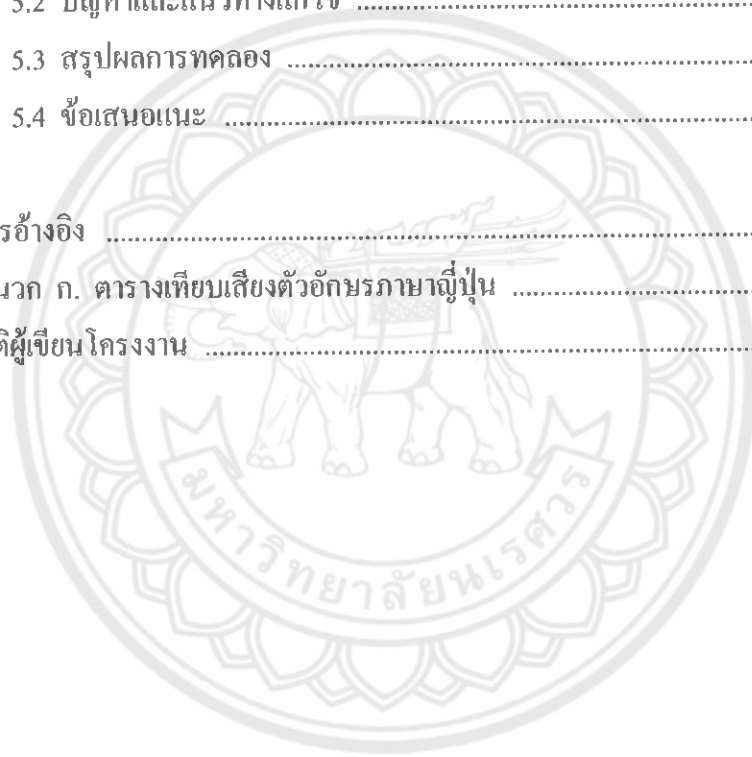


สารบัญ

	หน้า
บทคัดย่อ	ก
Abstract	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ฉ
สารบัญรูปภาพ	ช
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	2
1.3 ขอบข่ายของโครงการ	2
1.4 ขั้นตอนการดำเนินงาน	2
1.5 แผนการดำเนินงาน	3
1.6 ผลที่คาดว่าจะได้รับ	4
1.7 งบประมาณที่ใช้	4
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง	5
2.1 ภาษาญี่ปุ่น	5
2.2 Key Events	11
บทที่ 3 วิธีการดำเนินงาน	13
3.1 Requirements	13
3.2 การออกแบบโปรแกรม	14
3.3 System Architecture	17
3.4 Input	20
3.5 Search	21
3.6 Output	23

สารบัญ (ต่อ)

บทที่ 4 ผลการทดลอง	24
4.1 ทดสอบโปรแกรมส่วนประมวลผล	24
4.2 ทดสอบโปรแกรมส่วนฟังก์ชันการใช้งาน	28
บทที่ 5 บทสรุป	48
5.1 วิเคราะห์ผลการทดลอง	48
5.2 ปัญหาและแนวทางแก้ไข	49
5.3 สรุปผลการทดลอง	50
5.4 ข้อเสนอแนะ	50
เอกสารอ้างอิง	51
ภาคผนวก ก. ตารางเทียบเสียงตัวอักษรภาษาไทยปุ่น	52
ประวัติผู้เขียนโครงการ	55



สารบัญตาราง

ตารางที่	หน้า
3.1 Business Event	14
3.2 แสดง Requirements Traceability Matrix (RTM)	16
4.1 ผลการทดสอบโปรแกรมส่วนประมวลผล	46
4.2 ผลการทดสอบโปรแกรมส่วนฟังก์ชันการใช้งาน	47
5.1 สรุปผลการทดสอบโปรแกรม	48
5.2 การตอบสนอง Requirements	49



สารบัญรูปภาพ

รูปที่	หน้า
1.1 ตัวอย่างการพิมพ์ภาษาญี่ปุ่นใน MS Word.....	1
2.1 ตารางสระและพยัญชนะของตัวอักษรฮิรางานะ	6
2.2 ตารางเทียบเสียงควบกล้ำของตัวอักษรฮิรางานะ.....	7
2.3 ตารางสระและพยัญชนะของตัวอักษรคาตากานะ.....	9
2.4 ตารางเทียบเสียงควบกล้ำของตัวอักษรคาตากานะ	10
3.1 Context Diagram	14
3.2 Sequence Diagram.....	16
3.3 แสดงขั้นตอนการสร้างโปรแกรม	17
3.4 Class Diagram	18
3.5 Sequence Diagram.....	19
3.6 Flow Chart ของส่วน Input	20
3.7 แสดงตัวอย่างการทำงานของส่วน Input.....	21
3.8 Flow Chart ของส่วน Search.....	22
3.9 แสดงตัวอย่างการทำงานของส่วน Search.....	23
3.10 Flow Chart ของส่วน Output.....	23
3.11 แสดงตัวอย่างการทำงานของส่วน Output.....	23
4.1 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาไทยเลือก โหมดไม่มีการแปลง	24
4.2 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาไทยเลือก โหมดอักษรญี่ปุ่นแบบฮิรางานะ	25
4.3 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาไทยเลือก โหมดอักษรญี่ปุ่นแบบคาตากานะ.....	25
4.4 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาอังกฤษเลือก โหมดไม่มีการแปลง	26
4.5 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาอังกฤษเลือก โหมดอักษรญี่ปุ่นแบบฮิรางานะ.....	26
4.6 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาอังกฤษเลือก โหมดอักษรญี่ปุ่นแบบคาตากานะ...	27
4.7 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาไทยและภาษาอังกฤษเลือก ทุก โหมด.....	27
4.8 เลือกฟังก์ชัน New จาก Toolbar.....	28
4.9 เลือกฟังก์ชัน New จาก MenuFile.....	28

สารบัญรูปภาพ (ต่อ)

รูปที่	หน้า
4.10 ผลการทดสอบฟังก์ชัน New	29
4.11 เลือกฟังก์ชัน Open จาก Toolbar	29
4.12 เลือกฟังก์ชัน Open จาก MenuFile	30
4.13 เลือก ไฟล์ที่จะทำการ Open	30
4.14 ผลการทดสอบฟังก์ชัน Open	31
4.15 เลือกฟังก์ชัน Save จาก Toolbar	31
4.16 เลือกฟังก์ชัน Save จาก MenuFile	32
4.17 ตั้งชื่อไฟล์ก่อน Save	32
4.18 ผลการทดสอบฟังก์ชัน Save เปิด โดยใช้โปรแกรม Notepad	33
4.19 เลือกฟังก์ชัน Save As จาก Toolbar	33
4.20 เลือกฟังก์ชัน Save As จาก MenuFile	34
4.21 ตั้งชื่อไฟล์ก่อน Save As	34
4.22 เลือกฟังก์ชัน Cut จาก Toolbar	35
4.23 เลือกฟังก์ชัน Cut จาก MenuEdit	35
4.24 ผลการทดสอบฟังก์ชัน Cut	35
4.25 เลือกฟังก์ชัน Copy จาก Toolbar	36
4.26 เลือกฟังก์ชัน Copy จาก MenuEdit	36
4.27 ผลการทดสอบฟังก์ชัน Copy	36
4.28 เลือกฟังก์ชัน Paste จาก Toolbar	37
4.29 เลือกฟังก์ชัน Paste จาก MenuEdit	37
4.30 ผลการทดสอบฟังก์ชัน Paste	37
4.31 เลือกฟังก์ชัน Undo จาก Toolbar	38
4.32 เลือกฟังก์ชัน Undo จาก MenuEdit	38
4.33 ผลการทดสอบฟังก์ชัน Undo	38
4.34 เลือกฟังก์ชัน Redo จาก Toolbar	39
4.35 เลือกฟังก์ชัน Redo จาก MenuEdit	39
4.36 ผลการทดสอบฟังก์ชัน Redo	39
4.37 เลือกฟังก์ชัน No Converting จาก Toolbar	40

สารบัญรูปร่างภาพ (ต่อ)

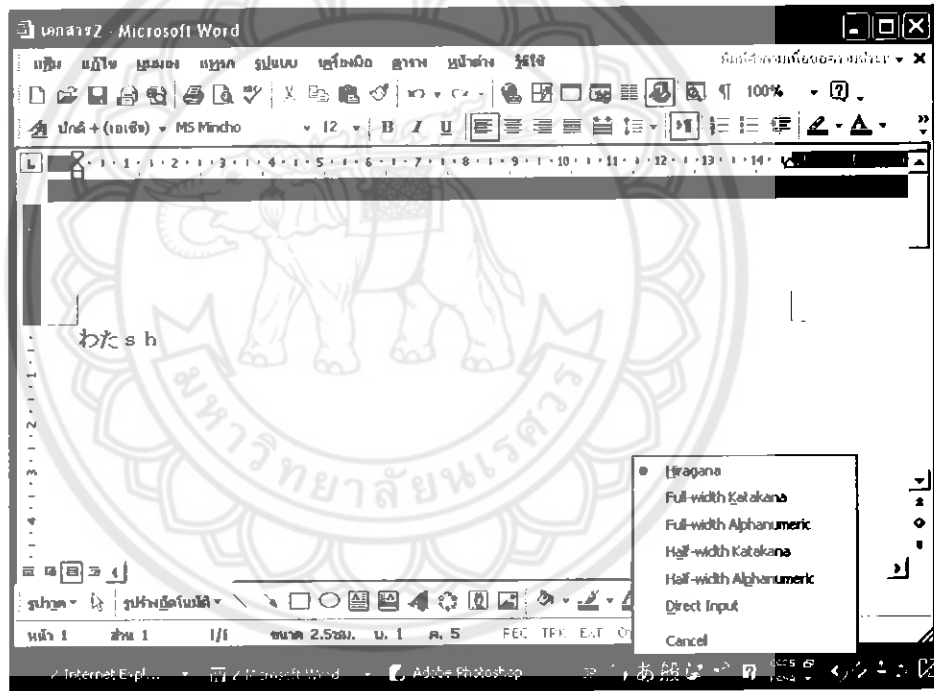
รูปที่	หน้า
4.38 เลือกฟังก์ชัน No Converting จาก MenuMode.....	40
4.39 ผลการทดสอบฟังก์ชัน No Converting.....	40
4.40 เลือกฟังก์ชัน Thai to Japanese(Hiragana) จาก Toolbar.....	41
4.41 เลือกฟังก์ชัน Thai to Japanese(Hiragana) จาก MenuMode	41
4.42 ผลการทดสอบฟังก์ชัน Thai to Japanese(Hiragana).....	41
4.43 เลือกฟังก์ชัน English to Japanese(Hiragana) จาก Toolbar.....	42
4.44 เลือกฟังก์ชัน English to Japanese(Hiragana) จาก MenuMode.....	42
4.45 ผลการทดสอบฟังก์ชัน English to Japanese(Hiragana).....	42
4.46 เลือกฟังก์ชัน Thai to Japanese(Katakana) จาก Toolbar	43
4.47 เลือกฟังก์ชัน Thai to Japanese(Katakana) จาก MenuMode	43
4.48 ผลการทดสอบฟังก์ชัน Thai to Japanese(Katakana)	43
4.49 เลือกฟังก์ชัน English to Japanese(Katakana) จาก Toolbar	44
4.50 เลือกฟังก์ชัน English to Japanese(Katakana) จาก MenuMode	44
4.51 ผลการทดสอบฟังก์ชัน English to Japanese(Katakana)	44
4.52 เลือกฟังก์ชัน Help จาก MenuHelp	45
4.53 ผลการทดสอบฟังก์ชัน Help.....	45

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

เนื่องจากสมาชิกในกลุ่มมีโอกาสได้เรียนภาษาญี่ปุ่นเบื้องต้น ทำให้เกิดความสนใจ และอยากทำโครงการเกี่ยวกับภาษาญี่ปุ่น ในช่วงที่ได้เรียนภาษาญี่ปุ่นนั้น ได้ใช้โปรแกรมที่สามารถพิมพ์ภาษาญี่ปุ่นจากคำอ่าน โดยใช้ตัวอักษรภาษาอังกฤษ จึงอยากที่จะทำโปรแกรมที่สามารถพิมพ์ภาษาญี่ปุ่น โดยใช้คำอ่านเป็นตัวอักษรภาษาไทย ซึ่งโปรแกรมที่สามารถพิมพ์ภาษาญี่ปุ่นโดยใช้การออกเสียงเป็นภาษาอังกฤษ มีลักษณะดังรูป



รูปที่ 1.1 ตัวอย่างการพิมพ์ภาษาญี่ปุ่นใน MS Word

คือ สมมติว่าเราจะพิมพ์คำว่า わたし อ่านว่าวตาชิ เราก็จะพิมพ์คำอ่านเป็นตัวอักษรภาษาอังกฤษ คือ watashi เมื่อพิมพ์ว่า wa ก็จะได้ わ พิมพ์ว่า ta ก็จะได้ た เป็นต้น

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อสร้างโปรแกรมช่วยพิมพ์ภาษาญี่ปุ่นจากคำอ่านโดยใช้ตัวอักษรภาษาไทย
2. เพื่อนำความรู้ที่ได้ศึกษาจากการทำโครงการนี้ไปใช้ได้

1.3 ขอบข่ายของโครงการ

พัฒนาโปรแกรมให้สามารถพิมพ์ภาษาญี่ปุ่นจากคำอ่านโดยใช้ตัวอักษรภาษาไทยในการพิมพ์ได้อย่างถูกต้อง

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาทฤษฎีและหลักการที่เกี่ยวข้องกับโครงการ
 - รวบรวมข้อมูลและบทความที่เกี่ยวข้องกับโครงการ
 - เลือก Tool เพื่อนำมาใช้พัฒนาโปรแกรม และศึกษาวิธีการใช้งาน
 - ศึกษารูปแบบและวิธีการเขียนรายงานที่ถูกต้อง
2. เขียนโปรแกรม
3. ทดสอบและปรับปรุงโปรแกรมให้สมบูรณ์และเหมาะสม
4. เขียนรายงานและจัดทำรูปเล่ม

1.5 แผนการดำเนินงาน

Project Leader: นางสาวสลวิดา เหมหัตถยา		Project: A Japanese character typing assistant using Thai alphabets										Date: 26 Feb 07					
Project Objective: เพื่อพัฒนาเครื่องมือสำหรับงานนี้		Project to Complete By: 31 May 07										Owner / Priority					
Objectives	Major Tasks																
○	1 ศึกษาภาษาญี่ปุ่นเบื้องต้น	○													B	B	A
○	2 เลือกใช้ Tool ที่เหมาะสม	○													B	A	B
○	3 ศึกษาภาษา Java	○	○	○	○	○									B	A	B
○	4 ศึกษา unicode	○	○												A	B	B
○	5 ศึกษา structure ทวีระชัย	○	○	○	○										B	B	A
○	6 ศึกษาวิธีการเขียนรายงาน	○	○												A	B	B
○	7 เขียนโปรแกรมสำหรับเป็นหมวดหมู่		○	○	○	○	○	○							B	B	A
○	8 เขียนกับ unicode			○	○	○	○	○	○						A	B	B
○	9 เก็บเข้าฐานข้อมูล						○	○	○	○					B	B	A
○	10 เขียนโปรแกรมคัดลอกข้อมูลและ unicode						○	○	○	○	○	○	○	○	B	A	B
○	11 ทดสอบโปรแกรม							○	○	○	○	○	○	○	A	B	B
○	12 ปรับปรุงแก้ไข								○	○	○	○	○	○	B	A	B
○	13 เขียนรายงาน	○	○	○	○	○	○	○	○	○	○	○	○	○	B	A	B
○	14 จัดทำรูปเล่มรายงาน												○	○	B	B	A
○	15 เตรียมส่งมอบโครงการไปแรก												○	○	A	B	B
○	A ศึกษาหลักดูแลจัดการที่ต่างกับโครงการ														B	B	A
○	B เขียนโปรแกรม														B	A	B
○	C ทดสอบและปรับปรุงโปรแกรมให้สมบูรณ์และเหมาะสม														A	B	B
○	D เขียนรายงานและจัดทำรูปเล่ม														B	A	B
People working on the project																	
เพื่อสร้างโปรแกรมสำหรับจัดการภาษาญี่ปุ่นเข้าเครื่องคอมพิวเตอร์ เพื่ออำนวยความสะดวกในการจัดการทางด้านวิชาการ		5 Mar 07	12 Apr 07	7 May 07	8 Jun 07	24 Jun 07	8 Jul 07	20 Jul 07	3 Aug 07	26 Aug 07	4 Sep 07	20 Sep 07	2 Oct 07	Kobporm	Salakjai	Mantana	
กำหนดการทำงานไปแรก A Japanese character typing assistant using Thai alphabets จัดกำหนดให้ในวันที่ 5 March 2007 ถึงสุดท้ายวันที่ 31 May 2007																	

1.6 ผลที่คาดว่าจะได้รับ

1. สามารถนำความรู้ที่ได้จากการศึกษา trie structure ไปประยุกต์ใช้กับการพัฒนาในด้านที่เกี่ยวข้องกับฐานข้อมูลได้อย่างเหมาะสม
2. ได้โปรแกรมที่สามารถพิมพ์ภาษาญี่ปุ่นจากคำอ่าน โดยใช้ตัวอักษรภาษาไทยในการพิมพ์ได้
3. ทำให้คนที่ไม่ถนัดหรือไม่มีความรู้ภาษาอังกฤษ แต่มีความรู้ภาษาไทยสามารถพิมพ์ภาษาญี่ปุ่นได้

1.7 งบประมาณที่ใช้

1. ค่าวัสดุสำนักงาน	เป็นเงิน	200	บาท
2. ค่าจัดทำเอกสาร	เป็นเงิน	500	บาท
3. ค่าวัสดุอื่นๆ	เป็นเงิน	800	บาท
4. ค่าหนังสือ	เป็นเงิน	1,500	บาท
	รวมเป็นเงิน	3,000	บาท (สามพันบาทถ้วน)



บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงหลักการและทฤษฎีต่างๆ ที่นำมาประยุกต์ใช้เพื่อทำโครงการนี้ ซึ่งทฤษฎีที่ใช้ ได้แก่ หลักการอ่านภาษาญี่ปุ่น และหลักการรับค่าจากคีย์บอร์ด

2.1 ภาษาญี่ปุ่น

ภาษาญี่ปุ่นเป็นภาษาราชการของประเทศญี่ปุ่น ซึ่งเป็นประเทศเดียวในโลกที่มีภาษาญี่ปุ่นเป็นภาษาราชการเต็มตัว (ไม่มีการใช้ภาษาต่างประเทศในวงราชการ) ภาษาญี่ปุ่นมีแบบภาษาที่เรียกว่า มาตรฐาน 2 แบบ คือ เฮียวจุงโงะ (標準語, hyōjungo, ภาษามาตรฐาน) ซึ่งเป็นภาษาที่ใช้กัน โทรศัพท์ และ เกียวซื่อโงะ (共通語, kyōtsūgo, ภาษาร่วม) ซึ่งเป็นภาษาที่ใช้กันอย่างไม่เป็นทางการ ปัจจุบันมีผู้ใช้ภาษาญี่ปุ่นทั่วโลกราว 130 ล้านคน

ตัวอักษรภาษาญี่ปุ่น

ตัวอักษรในภาษาญี่ปุ่นแบ่งเป็น 2 ประเภท คือ อักษรคันจิ และอักษรคาตะคาเนะ ซึ่งแบ่งย่อยเป็น ฮิรางานะ และคาตากานะ การที่ญี่ปุ่นใช้ตัวอักษรถึง 3 แบบ เนื่องจากว่าแต่เดิมนั้นภาษาญี่ปุ่นมีเพียงภาษาพูดไม่มีตัวอักษรที่ใช้เขียน จนกระทั่งประมาณศตวรรษที่ 5 ญี่ปุ่นได้นำตัวอักษรจีนมาใช้ และได้มีการดัดแปลงให้เหมาะสมกับภาษาญี่ปุ่น ตัวอักษรจีนที่นำมาใช้ในภาษาญี่ปุ่นนี้เรียกว่า อักษรคันจิ แต่เนื่องจากภาษาญี่ปุ่นและภาษาจีนมีความแตกต่างกันหลายประการ เช่น โครงสร้างทางไวยากรณ์ การออกเสียง เป็นต้น ตัวอักษรคันจิไม่สามารถแทนเสียงทุกเสียงในภาษาญี่ปุ่นได้ ดังนั้น ญี่ปุ่นจึงประดิษฐ์ตัวอักษรฮิรางานะ และอักษรคาตากานะ เพื่อใช้แทนเสียงที่ไม่สามารถเขียนด้วยตัวอักษรคันจิได้

2.1.1 ตัวอักษร ฮิรางานะ ひらがな

ตัวอักษรแบบฮิรางานะนี้มักจะใช้เป็นตัวอักษรที่ไว้เขียนคำญี่ปุ่นแท้ๆ จะมีลักษณะโค้งๆ โดยส่วนใหญ่ตัวอักษรชนิดนี้มักจะใช้กันมากที่สุด

ที่มา

ตัวอักษรฮิรางานะพัฒนามาจากอักษรจีน เริ่มแรกเรียก onnade หรือมือของผู้หญิง เพราะใช้เขียนโดยผู้หญิงเป็นส่วนใหญ่ ส่วนผู้ชายจะเขียนโดยใช้คันจิและคาตากานะ ประมาณ พ.ศ. 1500 ฮิรางานะจึงถูกใช้กันอย่างแพร่หลาย คำว่า ฮิรางานะ หมายถึง อักษรพยางค์สามัญ รูปแบบแรกๆ ของฮิรางานะ มีสัญลักษณ์หลายตัวที่ออกเสียงเหมือนกัน ระบบการเขียนมีความแตกต่างกันขึ้นอยู่กับ

กับผู้เขียนแต่ละคน ซึ่งรัฐบาลญี่ปุ่นได้เข้ามาจัดรูปแบบเมื่อ พ.ศ. 2489 จึงกลายเป็นอักษรที่ใช้ในปัจจุบัน

ลักษณะและการใช้งานของอักษรฮิรางานะ

ใช้ผันคำกริยาและคำคุณศัพท์ ตลอดจนคำช่วยทางไวยากรณ์และคำกริยาช่วย ตัวอักษรประเภทนี้เป็นตัวอักษรของญี่ปุ่นเองจริงๆ เพื่อที่จะกำหนดว่า สิ่งของ คำต่างๆ จะมีชื่ออะไร โดยที่สิ่งของหนึ่งอย่างจะมีเสียงได้ 1 เสียง แต่จะมีรูปเขียนได้ 2 รูปเขียนคือ ฮิรางานะ และคันจิ ส่วนเสียง 1 เสียงที่ออกเหมือนกันทุกอย่าง อาจจะมีคามหมายไม่เหมือนกัน ซึ่งจะมีตัวคันจิกำกับ ตัวคันจินี้จะมีรูปแบบไม่เหมือนกัน แม้ว่าเสียงที่ออกจะเหมือนกันแต่ความหมายก็จะไม่เหมือนกัน โดยจะมีรูปร่างของตัวคันจิที่แตกต่างกัน

สระ และ พยัญชนะ				
あ อะ	い อี	う อุ	え เอะ	お โอะ
か คะ	き คี	く คุ	け เคะ	こ โคะ
さ สะ	し ชิ	す สุ	せ เซะ	そ โซะ
た ทะ	ち ชี	つ ทสึ	て เทะ	と โตะ
な นะ	に นี	ぬ นุ	ね เนะ	の โนะ
は ฮะ	ひ ฮี	ふ ฟุ(ฮ)	へ เฮะ	ほ โฮะ
ま มะ	み มึ	む มุ	め เมะ	も โมะ
ยา มะ		ゆ ยู		よ โยะ
ら ระ	り ริ	る รุ	れ เระ	ろ โระ
わ วะ				を โวะ
				ん-น
が กะ	ぎ คี	ぐ กุ	げ เกะ	ご โคะ
ざ ซะ	じ จี	ず ซุ	ぜ เซะ	ぞ โซะ
だ ดะ	ぢ จี	づ ทสึ	で เดะ	ど โตะ
ば มะ	び มึ	ぶ มุ	べ เมะ	ぼ โมะ
ぱ เปะ	ปิ มึ	ปู มุ	เป เปะ	โป โปะ

รูปที่ 2.1 ตารางสระและพยัญชนะของตัวอักษรฮิรางานะ

เสียงความกล้า

(บะ)	(บ)	(โอะ)
きゃ คิ บะ	きゅ คิ บ	きょ คิ โอะ
しゃ ชิ บะ	しゅ ชิ บ	しょ ชิ โอะ
ちゃ ชิ บะ	ちゅ ชิ บ	ちょ ชิ โอะ
こゃ คิ บะ	こゅ คิ บ	こょ คิ โอะ
ひゃ ฮิ บะ	ひゅ ฮิ บ	ひょ ฮิ โอะ
みゃ มิ บะ	みゅ มิ บ	みょ มิ โอะ

りゃ ริ บะ りゅ ริ บ りょ ริ โอะ

きゃ คิ บะ	きゅ คิ บ	きょ คิ โอะ
じゃ จิ บะ	じゅ จิ บ	じょ จิ โอะ
ちゃ (จิ บะ)	ちゅ (จิ บ)	ちょ (จิ โอะ)
ひゃ มิ บะ	ひゅ มิ บ	ひょ มิ โอะ
ひゃ มิ บะ	ひゅ มิ บ	ひょ มิ โอะ

รูปที่ 2.2 ตารางเสียงความกล้าของตัวอักษรฮิรางานะ

หลักการอ่านตัวอักษรฮิรางานะ

ตัวอักษรฮิรางานะ มีอยู่ทั้งหมด 46 ตัว แต่จะมีเพียงแค่ 5 เสียง คือเสียง a i u e o (อะ อิ อุ เอะ โอะ) ตัวอักษรภาษาญี่ปุ่นสามารถออกเสียงได้ในตัวของมันเอง ไม่ต้องไปผสมกับตัวอักษรตัวอื่น ไม่ต้องประสมกับสระและวรรณยุกต์ แบ่งเป็นวรรคได้ดังนี้

- วรรค a (อะ) คือ a i u e o (อะ อิ อุ เอะ โอะ)
- วรรค ka (คะ) คือ ka ki ku ke ko (คะ คิ คุ เคะ โคะ)
- วรรค sa (สะ) คือ sa shi su se so (สะ ชิ สุ เสะ โสะ)
- วรรค ta (ทะ) คือ ta chi tsu te to (ทะ จิ ทชิ เทะ โทะ)
- วรรค na (นะ) คือ na ni nu ne no (นะ นิ นุ เนะ โนะ)
- วรรค ha (ฮะ) คือ ha hi hu he ho (ฮะ ฮิ ฮุ (ฟู) เฮะ โหะ)
- วรรค ma (มะ) คือ ma mi mu me mo (มะ มิ มุ เมะ โมะ)

วรรณ ya (ยะ) คือ ya - yu - yo (ยะ ยู โยะ)

วรรณ ra (ระ) คือ ra ri ru re ro (ระ ริ รุ เระ โระ)

วรรณ wa (วะ) คือ wa - - - wo(o) (วะ โวะ(โอะ))

ตัวอื่น (-n) เป็นตัวสะกดตัวเดียวในภาษานี้ บางทีก็เป็น น บางทีก็เป็น ม บางทีก็เป็น ง ขึ้นกับความง่ายในการออกเสียงของคำถัดมา เช่น minna (มินนะ) , sanpo (ซัมโปะ) , ringo (ริงโงะ) นอกจากนี้ยังมีการนำสัญลักษณ์พิเศษ ° (tenten) มาเพิ่มในวรรณ ka sa ta ha ซึ่งจะช่วยให้มีการเปลี่ยนเสียงไปเล็กน้อย คือเสียงที่ออกมาจะเป็นเสียงที่ก้อง ดังนี้

วรรณ ka เติม tenten จะเป็น ga gi gu ge go (กะ กิ กุ เกะ โกะ หรือ งะ จิ งู เงะ โงะ)

วรรณ sa เติม tenten จะเป็น za ji zu ze zo (ซะ จิ ชู เซะ โซะ)

วรรณ ta เติม tenten จะเป็น da ji du de do (ดะ จิ ดู เดะ โดะ)

วรรณ ha เติม tenten จะเป็น ba bi bu be bo (บะ บี บู เบะ โบะ) แต่ในวรรณ ha นี้สามารถเปลี่ยนได้อีก 1 แบบ คือ การเติมสัญลักษณ์ ° (maru) จะกลายเป็น pa pi pu pe po (ปะ ปี ปู่ เปะ โปะ หรือ พะ พิ พู เพะ โปะ)

เสียงควบกล้ำ : เกิดจากเสียงแล้ว i คือแล้วที่มีตัว i ต่อท้าย เช่น ki gi shi ji chi ni hi bi pi mi ri นำมาผสมกับตัว ya yu yo จะได้ออกมาเป็น

ki + ya = kya (ตัดตัว i ออกไป) จะได้เสียง “คียะ”

ki + yu = kyū (ตัดตัว i ออกไป) จะได้เสียง “คิยู”

ki + yo = kyo (ตัดตัว i ออกไป) จะได้เสียง “คิโอะ”

ในส่วนของวรรณอื่นๆนั้นก็ทำแบบนี้เช่นเดียวกัน

2.1.2 ตัวอักษร คาทาคานะ カタカナ

ตัวอักษรชนิดนี้เกิดจากการเอาขีดบางขีดที่อยู่ในตัวอักษรจีนบางตัวแบ่งออกมาให้เป็นตัวแทนของเสียงของตัวอักษรนั้น มักจะใช้ในการเขียนศัพท์ภาษาต่างประเทศ หรือว่าอาจจะใช้ในกรณีที่ต้องการเน้นคำ ตัวอักษรแบบคาทาคานะนี้มักจะมีลักษณะเป็นเหลี่ยมๆมีจำนวนของตัวอักษรเท่ากันกับฮิรางานะ

ที่มา

ตัวอักษรคาทาคานะนั้นถูกสร้างขึ้นในยุคเฮอัน (平安時代) โดยนำมาจากส่วนหนึ่งของตัวคันจิ ซึ่งพัฒนามาจากอักษรจีนที่ใช้โดยพระภิกษุเพื่อแสดงการออกเสียงอักษรจีนที่ถูกต้องเมื่อประมาณ ปี พ.ศ. 1400 ในช่วงแรกมีสัญลักษณ์หลายตัวที่แสดงเสียงเดียวกัน แต่เมื่อเวลาผ่านไป อักษรถูกปรับปรุงให้ดีขึ้นราว พ.ศ. 1900 มีสัญลักษณ์ 1 ตัว ต่อ 1 พยางค์เท่านั้น คำว่า คาทาคานะ หมายถึง อักษรพยางค์ที่เป็นส่วน (ของคันจิ)

ลักษณะและการใช้งานของอักษรคาตากานะ

อักษรคาตากานะ รูปนี้ก็เป็นตัวอักษรที่เป็นของญี่ปุ่นจริงๆ มีการออกเสียงเหมือนกับตัวฮิรางานะทุกประการ เพียงแต่ว่าการเขียนจะไม่เหมือนกันเสียง 1 เสียงจะมีรูปเขียนทั้ง 2 แบบคือฮิรางานะ และคาตากานะ ซึ่งญี่ปุ่นได้สร้างตัวคาตากานะขึ้นมาเพื่อเขียนออกเสียงให้กับคำที่มาจากต่างชาติ นั่นคือคำที่อยู่ในภาษาญี่ปุ่นไม่มีนั่นเอง ซึ่งก็อาจดัดแปลงมาจากภาษาอังกฤษ โปรตุเกส ฝรั่งเศส เป็นต้น เช่น รถบัสในภาษาญี่ปุ่นไม่มีคำว่ารถบัสจึงประยุกต์ตามภาษาอังกฤษ เรียกว่า บัสสุ ซึ่งจะเขียนด้วยตัวคาตากานะเท่านั้น

วิธีใช้อักษรคาตากานะ

- ใช้สะกดคำศัพท์ หรือชื่อที่มาจากต่างประเทศ
- ใช้เขียนชื่อสัตว์ พืช หรือสิ่งของเครื่องใช้ในกรณีที่ตัวอักษรคันจิของคำศัพท์เหล่านั้นยากเกินไป
- ใช้เขียนคำเลียนเสียงธรรมชาติ
- ใช้ในการเน้นคำหรือวลีในประโยค
- ใช้แสดงถึงคำพูดที่ไม่เป็นทางการ เช่น บทสนทนาในหนังสือการ์ตูน

สระ และ พยัญชนะ

ア อะ	イ อิ	ウ อุ	エ เอะ	オ โอะ
カ คะ	キ คิ	ク คุ	ケ เคะ	コ โคะ
サ สะ	シ ชิ	ス สุ	セ เซะ	ソ โซะ
タ ทะ	チ จิ	ツ ทชิ	テ เทะ	ト โทะ
ナ นะ	ニ นิ	ヌ นุ	ネ เนะ	ノ โนะ
ハ ฮะ	ヒ ฮิ	フ ฟุ(ฮ)	ヘ เฮะ	ホ โฮะ
マ มะ	ミ มิ	ム มุ	メ เมะ	モ โมะ
ヤ ยะ		ユ ยุ		ヨ โยะ
ラ ระ	リ ริ	ル รุ	レ เระ	ロ โระ
ワ วะ				ヲ โวะ
				ン -ก
ガ กะ	ギ กิ	グ กุ	ゲ เกะ	ゴ โกะ
ザ ซะ	ジ จิ	ズ จุ	ゼ เซะ	ゾ โซะ
ダ ดะ	チ จิ	ツ ทชิ	デ เดะ	ド โดะ
バ บะ	ビ บิ	ブ บุ	ベ เบะ	ボ โบะ
パ ปะ	ピ บิ	プ บุ	ペ เปะ	ポ โปะ

รูปที่ 2.3 ตารางสระและพยัญชนะของตัวอักษรคาตากานะ

เสียงควบกล้ำ

ヤ ยะ	ユ ยู	ヨ โยะ
キヤ คิ ยะ	キユ คิ ยู	キヨ คิ โยะ
シヤ ชิ ยะ	シユ ชิ ยู	シヨ ชิ โยะ
チャ จิ ยะ	チュ จิ ยู	チョ จิ โยะ
ニヤ นิ ยะ	ニユ นิ ยู	ニヨ นิ โยะ
ヒヤ ฮิ ยะ	ヒユ ฮิ ยู	ヒヨ ฮิ โยะ
ミヤ มิ ยะ	ミュ มิ ยู	ミヨ มิ โยะ

リヤ ริ ยะ リユ ริ ยู リヨ ริ โยะ

ギヤ กิ ยะ	ギユ กิ ยู	ギヨ กิ โยะ
ジヤ จิ ยะ	ジュ จิ ยู	ジヨ จิ โยะ
チャ จิ ยะ	チュ จิ ยู	チョ จิ โยะ
ピヤ ปิ ยะ	ピユ ปิ ยู	ピヨ ปิ โยะ
ヒヤ ฮิ ยะ	ヒユ ฮิ ยู	ヒヨ ฮิ โยะ

รูปที่ 2.4 ตารางเสียงควบกล้ำของตัวอักษรคาตากานะ

หลักการอ่านตัวอักษร คาตากานะ

ตัวอักษรคาตากานะมีอยู่ทั้งหมด 46 ตัว แต่จะมีเพียงแค่ 5 เสียง คือเสียง a i u e o (อะ อี อุ เอะ โอะ) ซึ่งอักษรคาตากานะจะเหมือนกับอักษรฮิรางานะ คือการออกเสียงจะเหมือนกัน มีแต่หน้าตาเท่านั้นที่ต่างกัน

2.1.3 ตัวอักษร คันจิ 事安倉做

ตัวอักษรคันจินี้ก็เป็นตัวอักษรที่มาจากจีน จะเขียนเหมือนกับภาษาจีนเลย ในตัวอักษรคันจิแต่ละตัว ก็จะสามารถสื่อความหมายได้ในตัวของมันเอง อย่างเช่นเราจะเขียนคำๆหนึ่งด้วยตัวอักษรฮิรางานะ อาจจะมีหลายความหมายก็ได้ เช่น คำว่า "ฮานะ" อาจจะไม่แปลว่า "ดอกไม้" หรือว่า "จุมุก" ก็ได้ แต่ถ้าหากเราเขียนคำๆนี้ด้วยคันจิ จะไม่มีการสับสนเพราะคำว่า "ฮานะ" ที่แปลว่า "ดอกไม้" กับ "ฮานะ" ที่แปลว่า "จุมุก" จะเขียนไม่เหมือนกันในอักษรคันจิ แต่ว่าอักษรคันจินาคคนญี่ปุ่นเองยังถือว่ายากมาก เพราะมีจำนวนมากเหลือเกิน ถึงหลายแสนตัวเลย

ในปัจจุบันคนต่างชาติเรียนภาษาญี่ปุ่นกันมากขึ้น และเพื่อให้เทียบกับภาษาอังกฤษได้ จึงมีการตั้งตัวอักษรโรมัน (a-z) หรือที่เรียกว่า โรมันจิ เพื่อให้คนต่างชาติสามารถอ่านภาษาญี่ปุ่นได้ง่ายขึ้น

ตัวอย่าง

ตัวอักษรฮิรางานะ わたしは です

ตัวอักษรคาตากานะ マンタナ

เมื่อนำมารวมกัน จะได้ประโยคว่า わたしはマンタナです。

และเมื่อนำมาเขียนเป็นแบบตัวอักษรโรมันก็จะได้ว่า watashi wa mantana desu. แปลว่า ฉันชื่อ มันทานาคะ

2.2 Key Events

เมื่อใดก็ตามที่ user กดแป้นพิมพ์ใดจะเกิดเหตุการณ์ที่เรียกว่า KEY_PRESSED และเมื่อ user ปล่อยแป้นพิมพ์นั้น เหตุการณ์ที่เกิดขึ้นคือ KEY_RELEASE ซึ่งเราสามารถตรวจสอบเหตุการณ์ดังกล่าวได้โดยใช้ method keyPressed() และ keyReleased() ของ class ใดก็ได้ที่ implements interface ชื่อ KeyListener อีก method หนึ่งที่มีประโยชน์มากก็คือ keyTyped() ซึ่งใช้ในการส่งคืนค่าของตัวอักษรที่ user กดแป้นพิมพ์ลงไป

ตัวอักษรในภาษาจาวาแบ่งออกเป็น 2 ชนิดคือ characters และ virtual key code โดย virtual key code จะขึ้นต้นด้วย VK_ เช่น VK_A หรือ VK_SHIFT ทั้งนี้ virtual key code ก็คือแป้นพิมพ์ที่ใช้งาน แต่ character จะหมายถึงแป้นพิมพ์ที่ user เห็น

ตัวอย่างเช่น เมื่อ user พิมพ์ตัว A จาวาจะรายงานผลเห็นเหตุการณ์จำนวน 5 เหตุการณ์ดังนี้

- กดปุ่ม SHIFT (keyPressed เรียกว่า VK_SHIFT)
- กดปุ่ม A (keyPressed เรียกว่า VK_A)
- พิมพ์ตัว A (keyTyped เรียกว่า A เป็น character ที่มองเห็นได้)
- ปล่อยปุ่มตัวอักษร A (keyReleased เรียกว่า VK_A)
- ปล่อยปุ่ม SHIFT (keyReleased เรียกว่า VK_SHIFT)

ในทางกลับกัน ถ้าพิมพ์ตัวอักษร a จะเกิดเหตุการณ์เพียง 3 เหตุการณ์ได้แก่

- กดปุ่ม A (keyPressed เรียกว่า VK_A)
- พิมพ์ตัว a (keyTyped เรียกว่า a เป็น character ที่มองเห็นได้)
- ปล่อยปุ่มตัวอักษร A (keyReleased เรียกว่า VK_A)

ดังนั้น keyTyped() จึงใช้ในการรายงานค่าของตัวอักษร (character) ที่พิมพ์เข้าไป (เช่น A หรือ a) ในขณะที่ keyPressed() และ keyReleased() จะใช้ในการรายงานปุ่มจริงที่ user กดหรือปล่อย ซึ่งส่วนใหญ่จะมีได้มากกว่า 1 ปุ่ม

ในการทำงานกับ method keyPressed() หรือ keyReleased() จะต้องทำการตรวจสอบค่าของ keycode ก่อนเสมอ(ชนิดเป็น integer) โดยใช้ method keyPressed() หรือ keyReleased() และตรวจสอบด้วย Propertyget KeyCode ดังนี้

```
public void keyPressed(KeyEvent e) {
    int keyCode = e.getKeyCode();
    .....
}
```

โดย keyCode จะมีค่าเท่ากับค่าคงที่ค่าใดค่าหนึ่ง ดังนี้

```
VK_A...VK_Z
VK_0...VK_9
VK_COMMA , VK_PERIOD , VK_SLASH , VK_SEMICOLON , VK_EQUALS , VK_OPEN_BRACKET ,
VK_BACK_SLASH , VK_CLOSE_BRACKET , VK_BACK_QUOTE , VK_QUOTE , VK_GREATER ,
VK_LESS , VK_UNDERSCORE , VK_MINUS , VK_AMPERSAND , VK_ASTERISK , VK_AT ,
VK_BRACELEFT , VK_BRACERIGHT , VK_LEFT_PARENTHESIS , VK_CIRCUMFLEX ,
VK_COLON , VK_NUMBER_SIGN , VK_QUOTEDBL , VK_EXCLAMATION_MARK ,
VK_INVERTED_EXCLAMATION_MARK , VK_DEAD_ABOVEDOT , VK_DEAD_ABOVEERING ,
VK_DEAD_ACUTE , VK_DEAD_BREVE , VK_DEAD_CAROM , VK_DEAD_CEDILLA ,
VK_DEAD_CIRCUMFLEX , VK_DEAD_DIAERESIS , VK_DEAD_DOUBLEACUTE ,
VK_DEAD_GRAVE , VK_DEAD_TOTA , VK_DEAD
```

และในการตรวจสอบการกดปุ่ม SHIFT , CONTROL , ALT , และ META นั้นจะใช้ method ต่อไปนี้คือ isShiftDown() , isControlDown() , isAltDown() , isMetaDown()

แต่สำหรับการเขียน method keyTyped() นั้นจะต้องเรียกใช้ method getKeyChar() แทน method getKeiCode() เพื่อตรวจสอบ character ที่พิมพ์ลงไป

บทที่ 3

วิธีการดำเนินงาน

ในบทนี้จะกล่าวถึงขั้นตอนการทำงาน เพื่อให้ได้มาซึ่งโปรแกรมที่สามารถพิมพ์ตัวอักษรญี่ปุ่น โดยรับอินพุตจากคีย์บอร์ดภาษาไทยหรือภาษาอังกฤษได้ ซึ่งสามารถแบ่งขั้นตอนการดำเนินงานออกเป็น 2 ขั้นตอนหลัก ๆ คือ ขั้นตอนการสำรวจความต้องการของผู้ใช้ว่า ผู้ใช้ต้องการให้โปรแกรมสามารถทำงานได้อย่างไรบ้าง และอีกขั้นตอนหนึ่งก็คือ การสร้างโปรแกรมตามความต้องการของผู้ใช้ โดยขั้นตอนการสร้างโปรแกรมนี้นี้สามารถแบ่งย่อยได้เป็น 3 ขั้นตอนดังนี้ ขั้นแรกเป็นการรับอินพุตจากคีย์บอร์ด ขั้นที่สองนำอินพุตที่รับมาจากคีย์บอร์ดไปเปรียบเทียบกับฐานข้อมูล ขั้นสุดท้ายเป็นการนำผลลัพธ์ที่ได้จากการเปรียบเทียบกับฐานข้อมูลมาแสดงผล

3.1 Requirements

3.1.1 Functional

1. รับอินพุตจากคีย์บอร์ดตัวอักษรไทยแล้วแปลงเป็นตัวอักษรญี่ปุ่น ได้อย่างถูกต้อง
2. รับอินพุตจากคีย์บอร์ดตัวอักษรอังกฤษแล้วแปลงเป็นตัวอักษรญี่ปุ่น ได้อย่างถูกต้อง
3. สามารถพิมพ์แบบธรรมดาได้คือพิมพ์แล้วไม่ต้องมีการแปลงเป็นภาษาญี่ปุ่น
4. มีแถบเครื่องมือที่ผู้ใช้โปรแกรมเข้าใจได้และช่วยอำนวยความสะดวกในการใช้โปรแกรม
5. แสดงผลได้ทั้งสีรายงานและคาตากานะ

3.1.2 Non – Functional

1. ตอบสนองทันที(ภายใน 1 วินาที)

3.1.3 Constraints

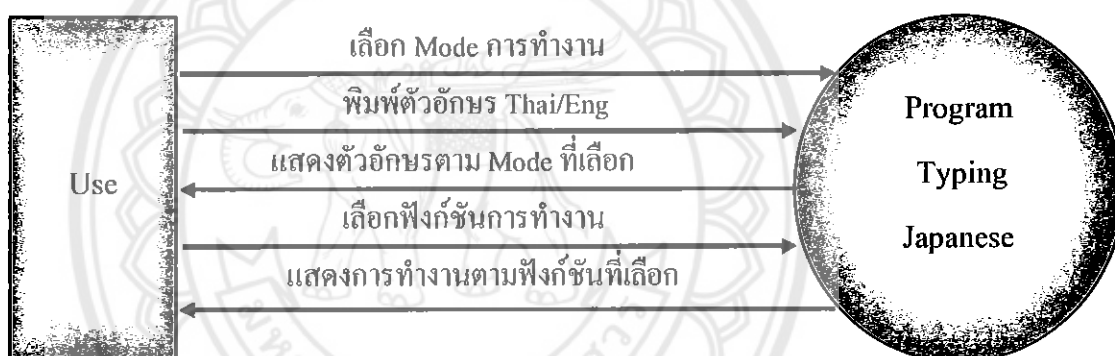
1. ผู้ใช้ควรมีความรู้พื้นฐานเกี่ยวกับภาษาญี่ปุ่นเบื้องต้น
2. ผู้ใช้ต้องศึกษาตารางเทียบเสียงจากคู่มือของโปรแกรมนี้ให้เข้าใจก่อนใช้งาน
3. โปรแกรมนี้พิมพ์ตัวอักษรภาษาญี่ปุ่นได้ 2 แบบคือ ตัวอักษรแบบสีรายงานะ และตัวอักษรแบบคาตากานะเท่านั้น
4. การบันทึกข้อมูลสามารถบันทึกได้เป็นไฟล์ .txt เท่านั้น
5. โปรแกรมนี้จะต้องใช้งานร่วมกับ The Java SE Development Kit (JDK) Version 1.6.0 ขึ้นไป
6. ก่อนจะใช้โปรแกรมนี้ เครื่องคอมพิวเตอร์ต้องแสดงผลภาษาญี่ปุ่นได้

3.1.4 Specification

1. โปรแกรมสามารถพิมพ์ตัวอักษรไทยแล้วแปลงเป็นตัวอักษรญี่ปุ่นได้อย่างถูกต้อง
2. โปรแกรมสามารถพิมพ์ตัวอักษรอังกฤษแล้วแปลงเป็นตัวอักษรญี่ปุ่นได้อย่างถูกต้อง
3. โปรแกรมสามารถพิมพ์แบบธรรมดาได้คือพิมพ์แล้วไม่ต้องมีการแปลงเป็นภาษาญี่ปุ่น
4. โปรแกรมมีแถบเครื่องมือที่ผู้ใช้โปรแกรมเข้าใจได้และช่วยอำนวยความสะดวกในการใช้โปรแกรม

3.2 การออกแบบโปรแกรม

3.2.1 Context Diagram



รูปที่ 3.1 Context Diagram

3.2.2 Business Event

เหตุการณ์ (Event)	ผล (Effect)
1. User เลือก mode การทำงาน	mode (in)
2. User พิมพ์ตัวอักษรภาษาไทยหรือภาษาอังกฤษ	ตัวอักษร (in)
3. โปรแกรมแสดงผลตัวอักษรตาม mode ที่ user เลือก	ตัวอักษร (out)
4. User เลือก function การทำงาน	Function (in)
5. โปรแกรมแสดงการทำงานตาม function ที่ user เลือก	การทำงาน (out)

ตารางที่ 3.1 Business Event

3.2.3 Business Use Case

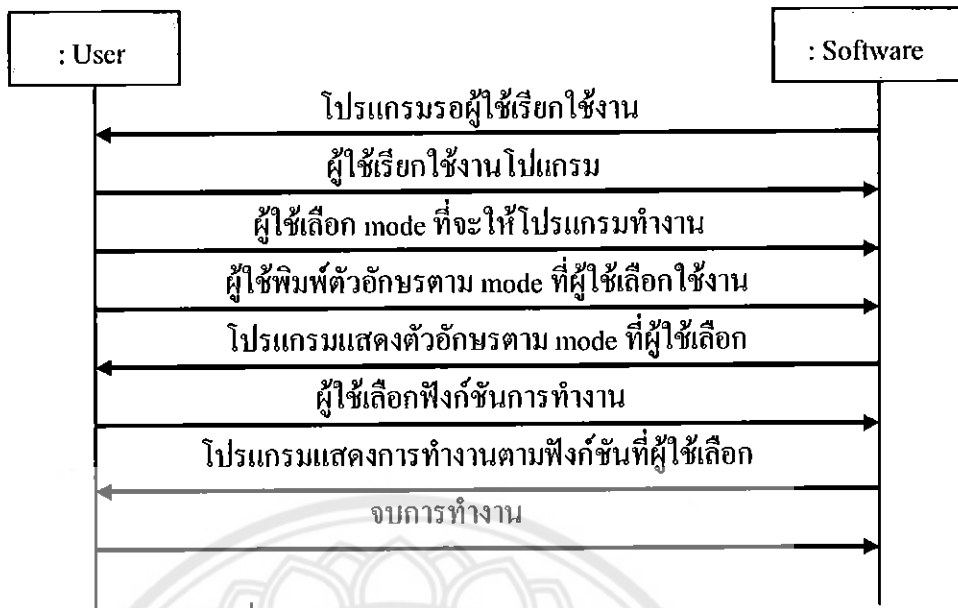
Name: โปรแกรมพิมพ์ภาษาญี่ปุ่น

Description: ผู้ใช้ใช้โปรแกรมพิมพ์ภาษาญี่ปุ่น โดยการใช้คีย์บอร์ดภาษาไทยหรือภาษาอังกฤษ

Preconditions: ผู้ใช้เลือก mode การทำงาน

Basic Course of Action:

1. โปรแกรมรอผู้ใช้เรียกใช้งาน
2. ผู้ใช้เรียกใช้งานโปรแกรม
3. ผู้ใช้เลือก mode ที่จะให้โปรแกรมทำงาน
4. ผู้ใช้พิมพ์ตัวอักษรตาม mode ที่ผู้ใช้เลือกใช้งาน
5. โปรแกรมแสดงผลตัวอักษรตาม mode ที่ผู้ใช้เลือกใช้งาน
6. ผู้ใช้ต้องการเลือก mode อื่น ๆ
7. ผู้ใช้กลับไปทำข้อที่ 3
8. ผู้ใช้เลือก function ที่จะให้โปรแกรมทำงาน
9. โปรแกรมแสดงการทำงานตาม function ที่ผู้ใช้เลือกใช้งาน
10. ผู้ใช้ต้องการเลือก function อื่น ๆ
11. ผู้ใช้กลับไปทำข้อที่ 8
12. จบการทำงาน

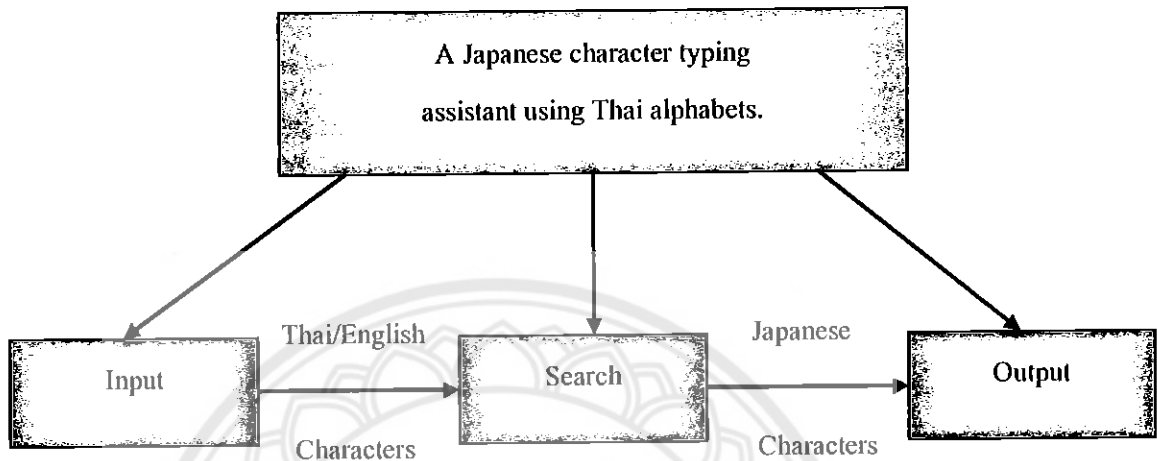


รูปที่ 3.2 Sequence diagram จาก Business Use Case

Requirement	Specification	Class	Method	Business Event
F-Req 1	1	MainMain	ThaiSearch,TKSearch	User เลือก mode User กดคีย์บอร์ด
F-Req 2	2	MainMain	EngSearch,EKSearch	User เลือก mode User กดคีย์บอร์ด
F-Req 3	3	MainMain	TextArea	User เลือก mode User กดคีย์บอร์ด
F-Req 4	4	MainMain	Toolbar	User เลือก function

ตารางที่ 3.2 แสดง Requirements Traceability Matrix (RTM)

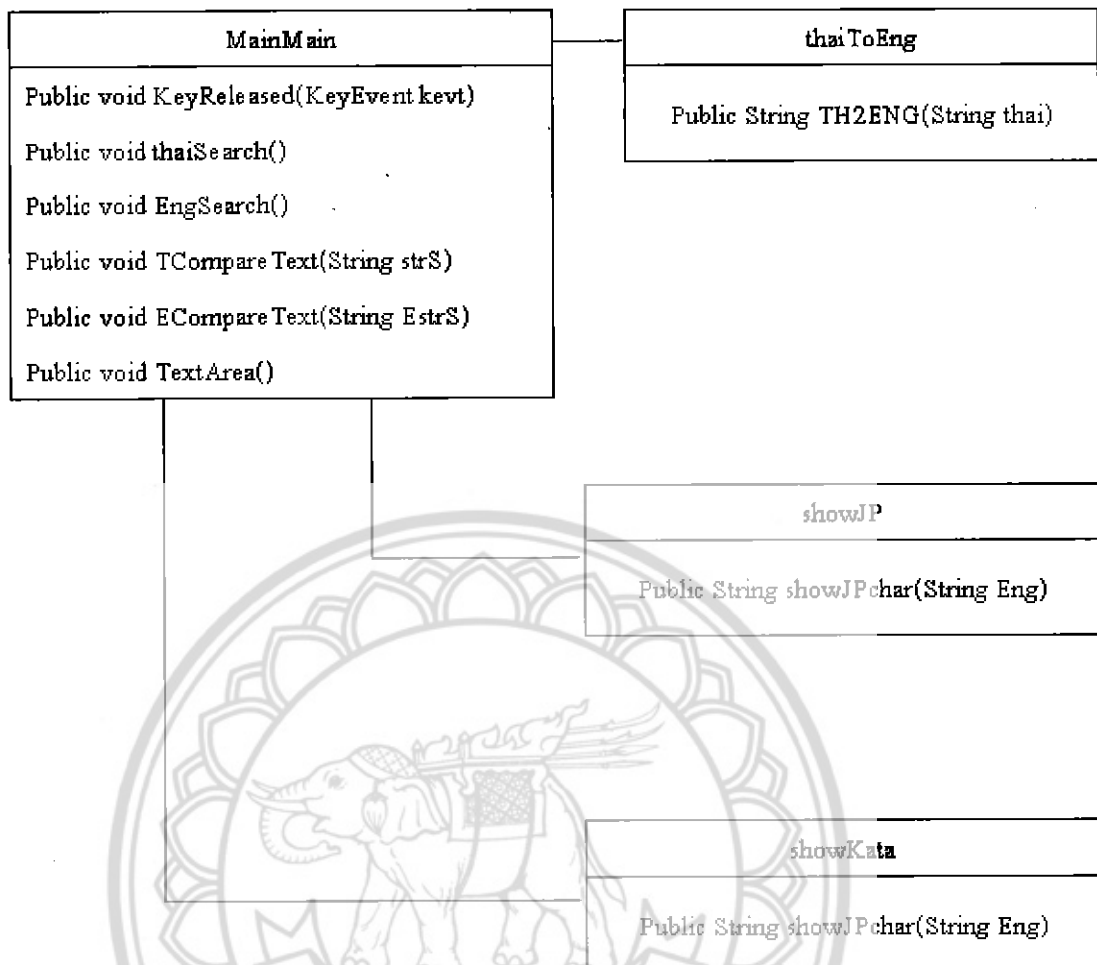
3.3 System Architecture



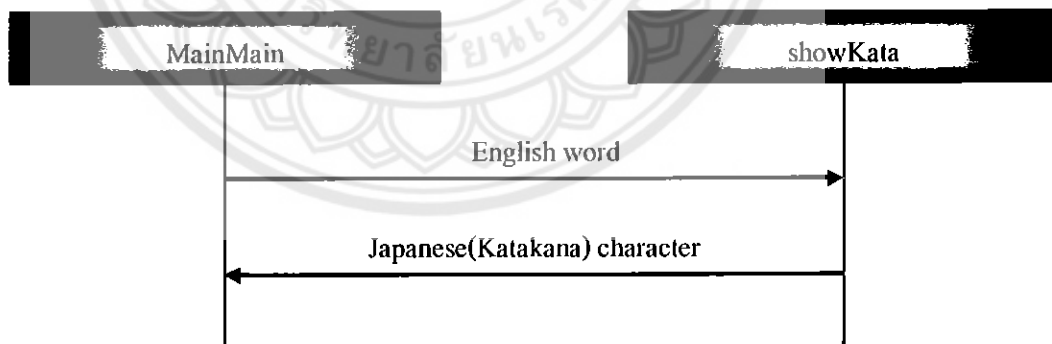
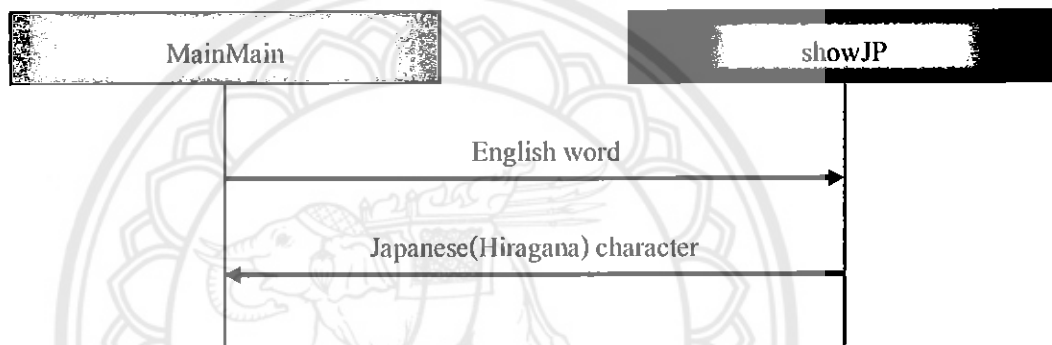
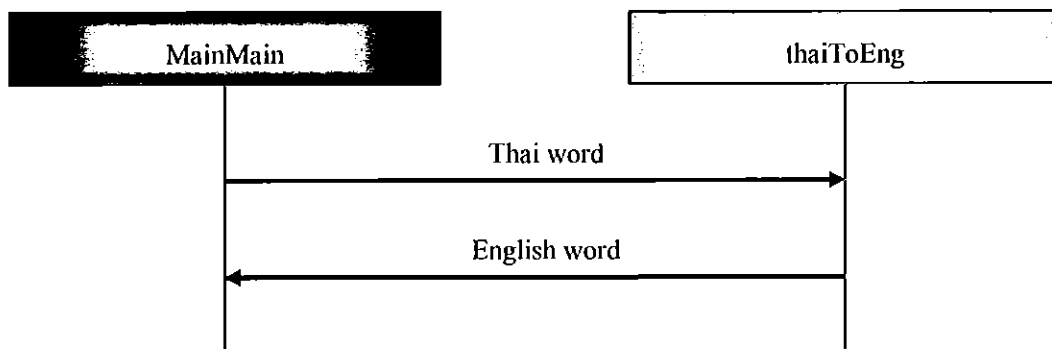
รูปที่ 3.3 แสดงขั้นตอนการสร้างโปรแกรม

โปรแกรมนี้มีขั้นตอนการสร้างโปรแกรมซึ่งแบ่งออกเป็น 3 ขั้นตอนหลัก ๆ ดังนี้

1. **Input** เป็นการรับอินพุตที่จากคีย์บอร์ด โดยรับข้อมูลเข้ามาเป็นตัวอักษรภาษาไทยหรือตัวอักษรภาษาอังกฤษ
2. **Search** เป็นการนำข้อมูลตัวอักษรที่รับเข้ามาไปเปรียบเทียบกับฐานข้อมูล แล้วเก็บผลลัพธ์ที่ได้จากการเปรียบเทียบไว้
3. **Output** เป็นการนำผลลัพธ์ที่เก็บไว้ไปแสดงผล



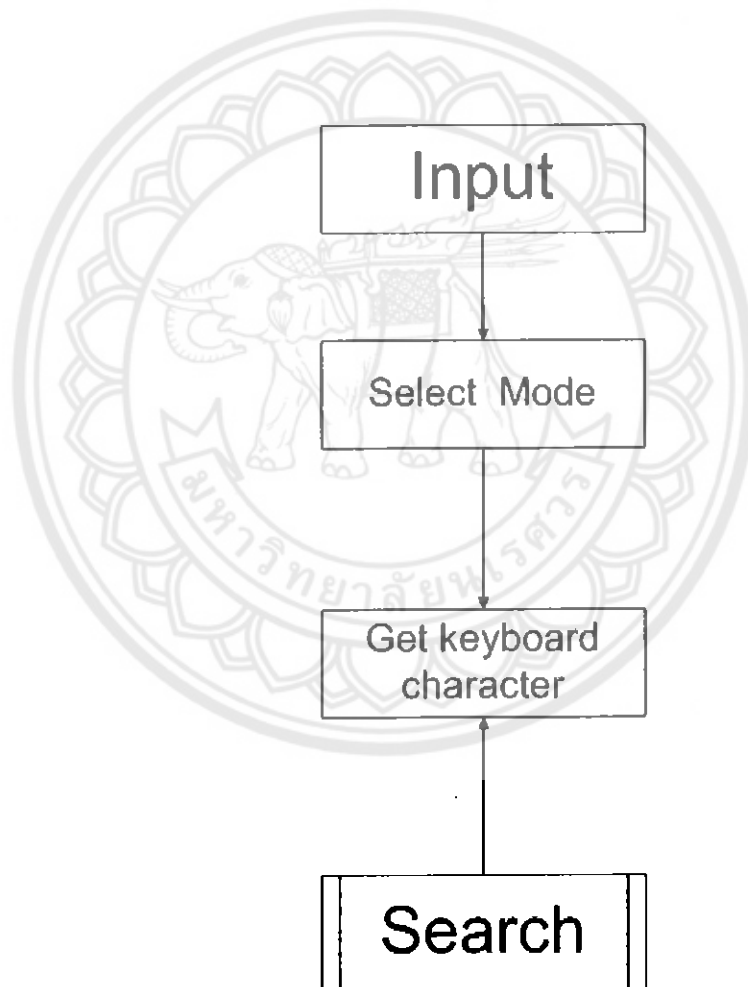
รูปที่ 3.4 Class Diagram



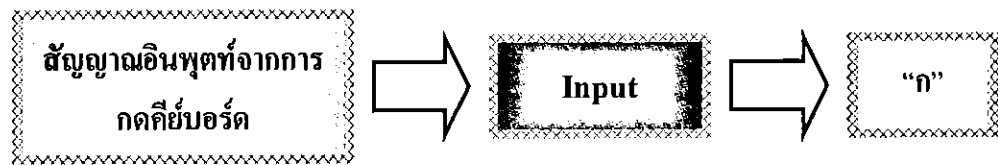
รูปที่ 3.5 Sequence Diagram

3.4 Input

ขั้นตอนนี้เป็นารรับข้อมูลจากการกดคีย์บอร์ดของผู้ใช้ ซึ่งจะวนรับข้อมูลที่ละตัวอักษร โดยการใช้การจับ KeyEvent จากการกดคีย์บอร์ด และเลือกใช้ method keyReleased() เนื่องจากมีการใช้งานร่วมกับ textarea ซึ่งถ้าใช้ method keyType() จะทำให้มีค่าค้างอยู่ที่ textarea เพราะ textarea ทำงานช้ากว่า KeyEvent เมื่อรับข้อมูลมาจากการกดคีย์บอร์ดแล้ว ก็จะทำการตรวจสอบโหมดการทำงานว่าอยู่ในโหมดอะไร จากนั้นจะทำการตรวจสอบว่าข้อมูลที่รับมานั้นเป็นข้อมูลตัวอักษรหรือเป็น tab, space bar หรือ enter ถ้าเป็น tab, space bar หรือ enter ก็จะทำการแสดงผลของการกดคีย์บอร์ดนั้น ถ้าเป็นข้อมูลตัวอักษรก็จะทำการเก็บข้อมูลตัวอักษรนั้นไว้แล้วเรียกใช้ฟังก์ชัน search ต่อไป



รูปที่ 3.6 Flow Chart ของส่วน Input

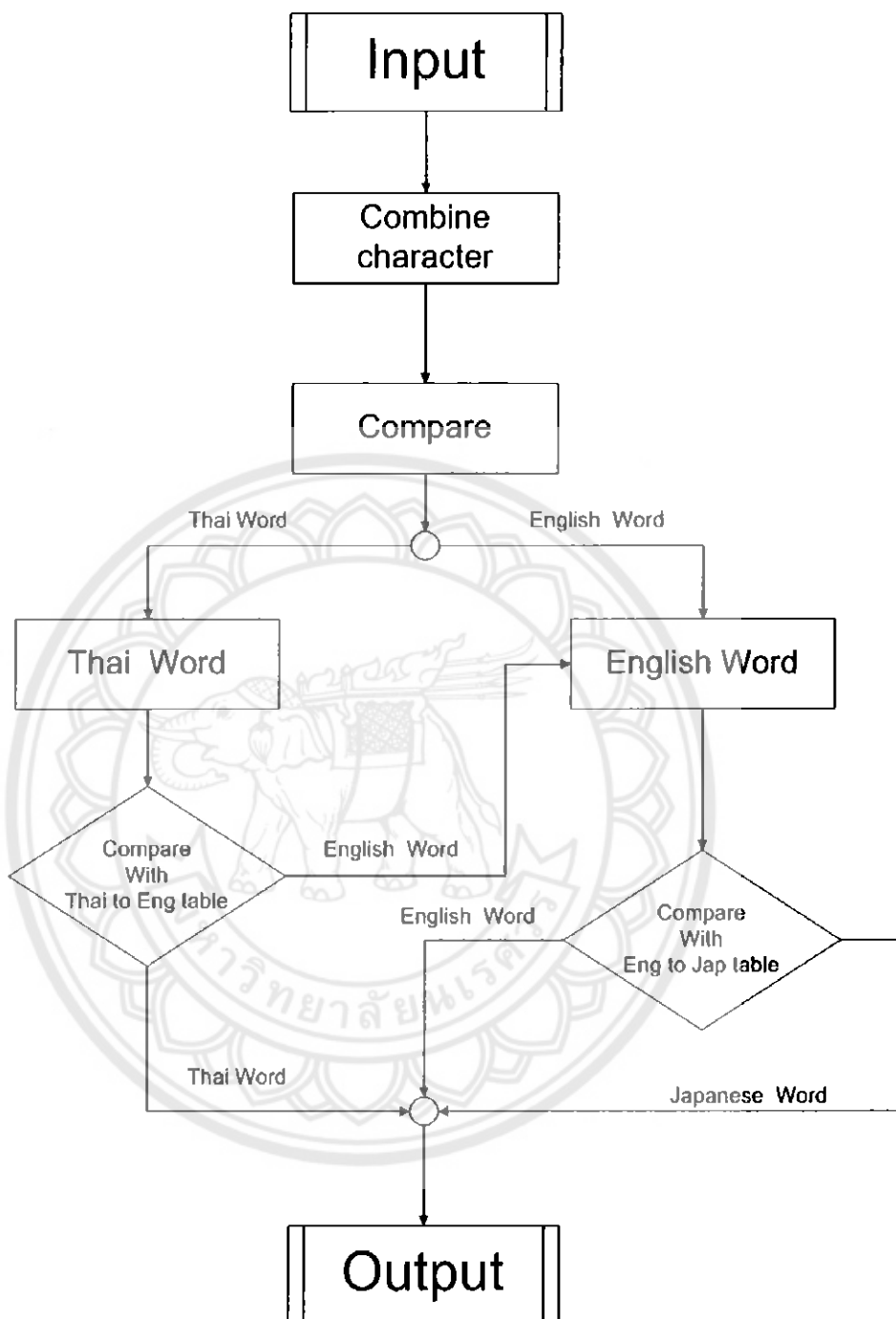


รูปที่ 3.7 แสดงตัวอย่างการทำงานของส่วน Input

3.5 Search

ขั้นตอนนี้จะเป็นการนำตัวอักษรที่ได้จากขั้นตอนที่แล้วมาจัดเป็นคำและนำไปเปรียบเทียบ ถ้าข้อมูลที่ได้รับเข้ามาเป็นภาษาไทยก็จะนำคำนั้นไปเปรียบเทียบกับตารางสำหรับแปลงคำภาษาไทยเป็นภาษาอังกฤษ ถ้าเทียบแล้วไม่ตรงกับคำที่มีอยู่ในตารางก็จะแสดงผลคำนั้น ถ้าเทียบแล้วตรงกับคำที่มีอยู่ในตารางก็จะแปลงคำนั้นเป็นภาษาอังกฤษ จากนั้นจะนำคำภาษาอังกฤษไปเปรียบเทียบกับตารางสำหรับแปลงคำจากภาษาอังกฤษเป็นภาษาญี่ปุ่น แล้วแสดงผลตัวอักษรภาษาญี่ปุ่น

ถ้าข้อมูลที่ได้รับเข้ามาเป็นภาษาอังกฤษก็จะนำคำนั้นไปเปรียบเทียบกับตารางสำหรับแปลงคำจากภาษาอังกฤษเป็นภาษาญี่ปุ่น ถ้าเทียบแล้วไม่ตรงกับคำที่มีในตารางก็จะแสดงผลคำนั้น ถ้าเทียบแล้วตรงกับคำที่มีในตารางก็จะแปลงคำนั้นเป็นภาษาญี่ปุ่น แล้วแสดงผลตัวอักษรภาษาญี่ปุ่น



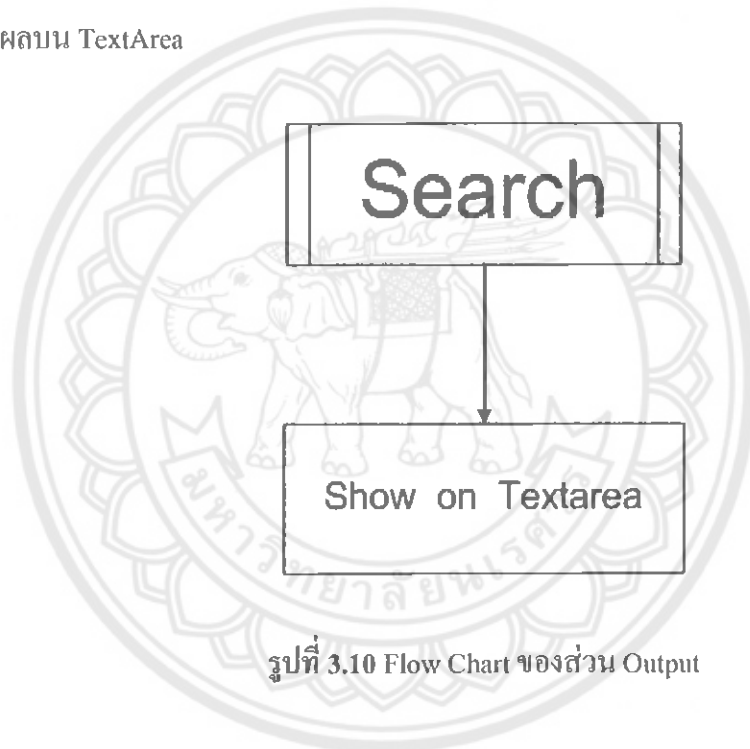
รูปที่ 3.8 Flow Chart ของส่วน Search



รูปที่ 3.9 แสดงตัวอย่างการทำงานของส่วน Search

3.6 Output

ขั้นตอนนี้เป็นกรนำเอาตัวอักษรที่ได้จากการขั้นตอนที่แล้วมาทำการแสดงผล โดยให้แสดงผลบน TextArea



รูปที่ 3.10 Flow Chart ของส่วน Output



รูปที่ 3.11 แสดงตัวอย่างการทำงานของส่วน Output

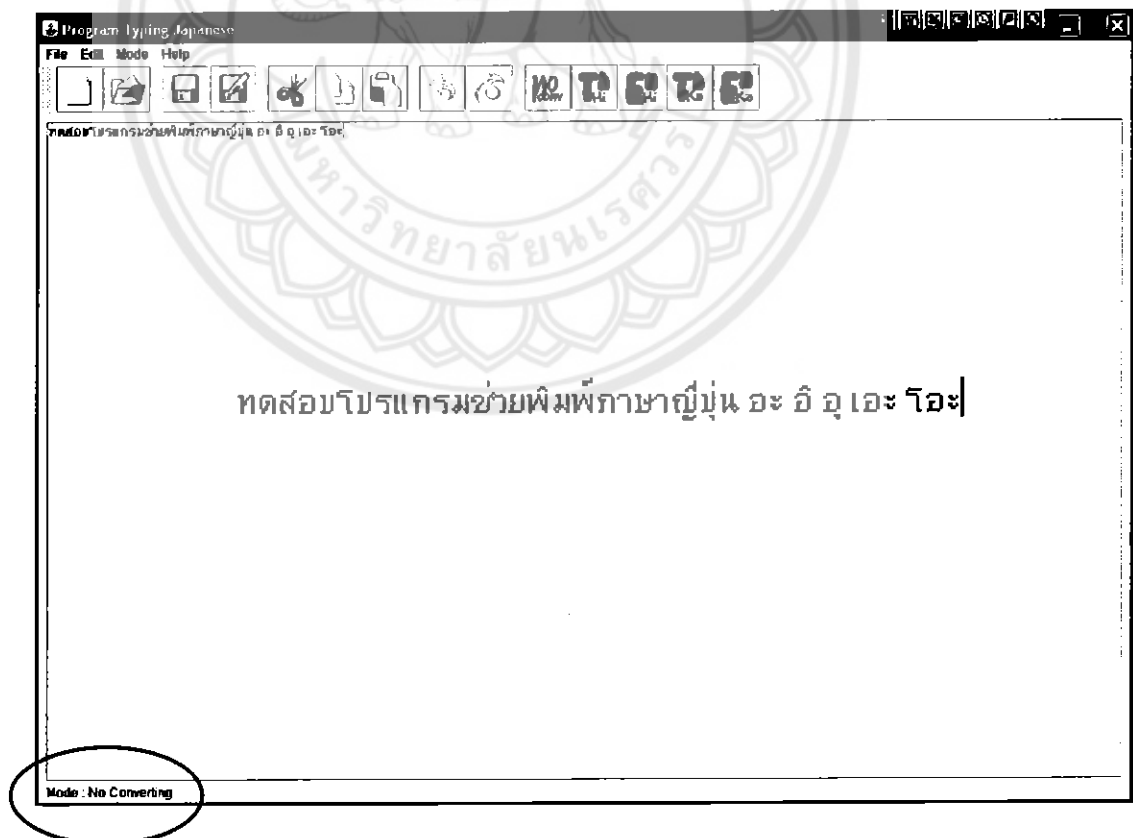
บทที่ 4

ผลการทดลอง

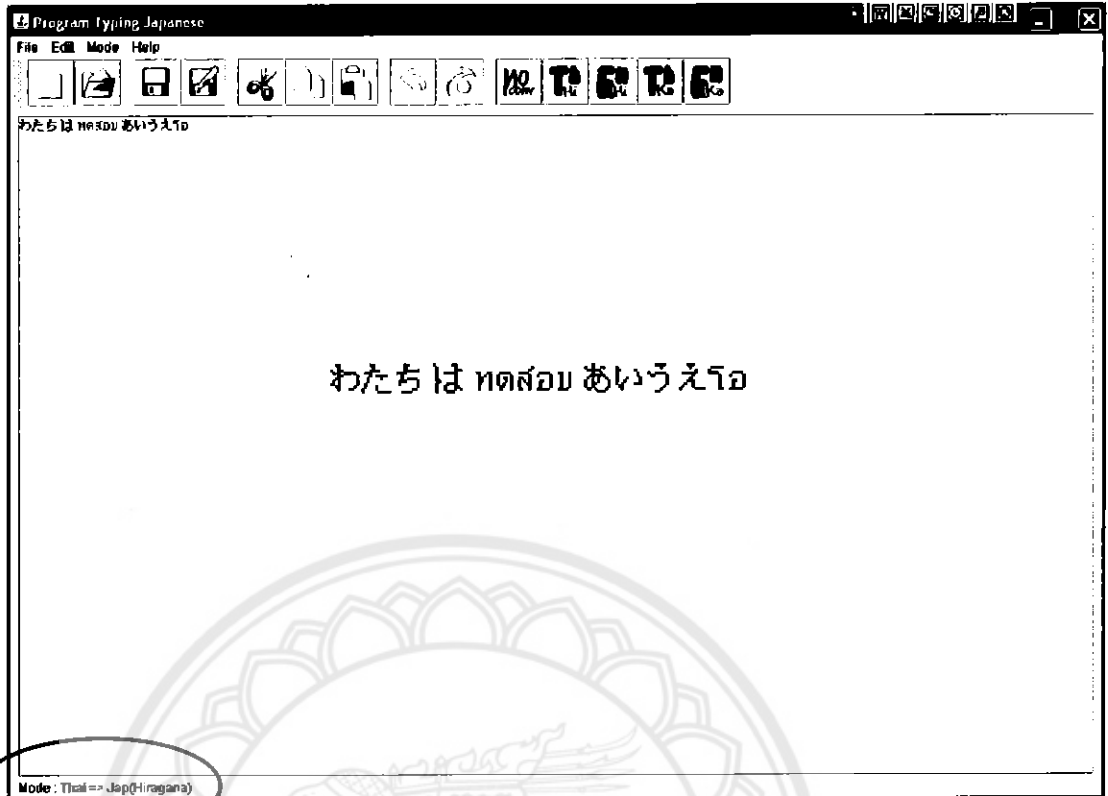
ในบทนี้จะกล่าวถึงการทดสอบโปรแกรมและแสดงผลของการทดสอบโปรแกรม โดยขั้นตอนการทดสอบโปรแกรมจะแบ่งออกเป็น 2 ส่วน เพื่อความสะดวกในการตรวจสอบหาข้อผิดพลาด โดยการทดสอบส่วนที่หนึ่งเป็นการทดสอบโปรแกรมส่วนประมวลผล และการทดสอบส่วนที่สองเป็นการทดสอบโปรแกรมส่วนฟังก์ชันการใช้งานของโปรแกรม

4.1 ทดสอบโปรแกรมส่วนประมวลผล

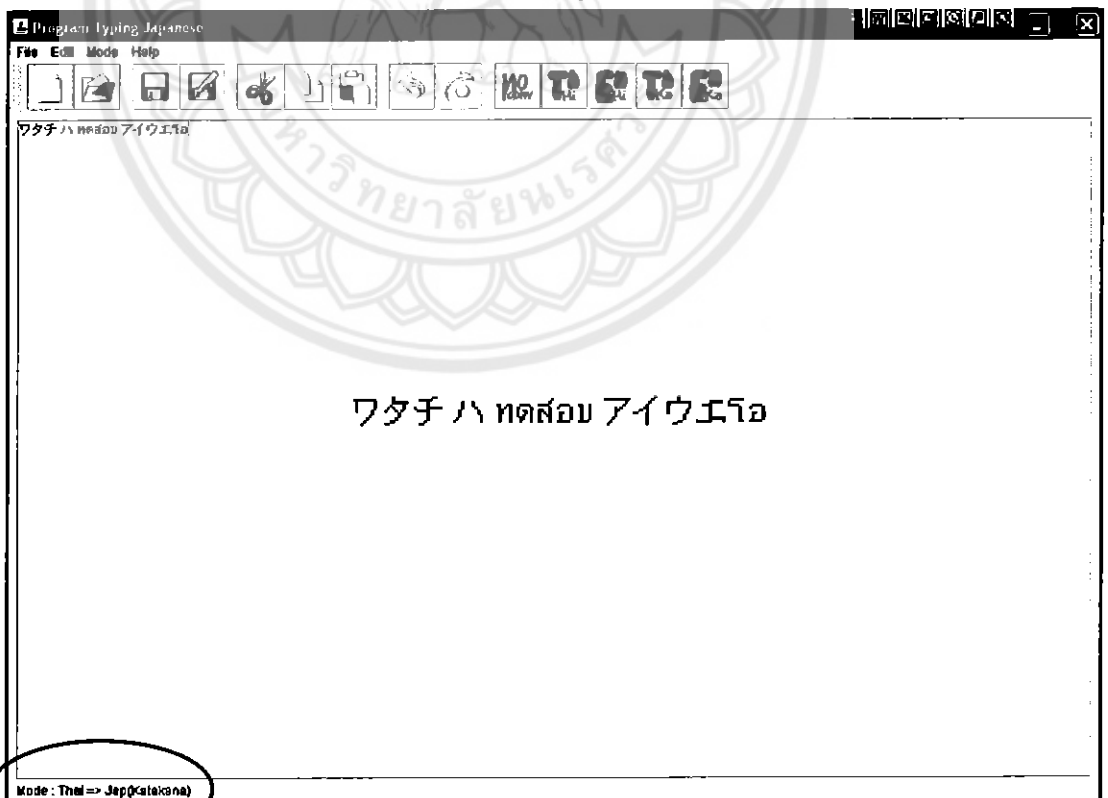
การทดสอบโปรแกรมช่วยพิมพ์ภาษาญี่ปุ่นในส่วนของการประมวลผลนี้เป็นการตรวจสอบว่าโปรแกรมสามารถรับอินพุตที่จากการกดคีย์บอร์ดทั้งภาษาไทยและภาษาอังกฤษ จากนั้นนำค่าที่ได้จากการรับอินพุตไปทำการเปรียบเทียบกับฐานข้อมูลแล้วนำผลลัพธ์ที่ได้จากการเปรียบเทียบกับฐานข้อมูลไปแสดงผลได้ถูกต้อง



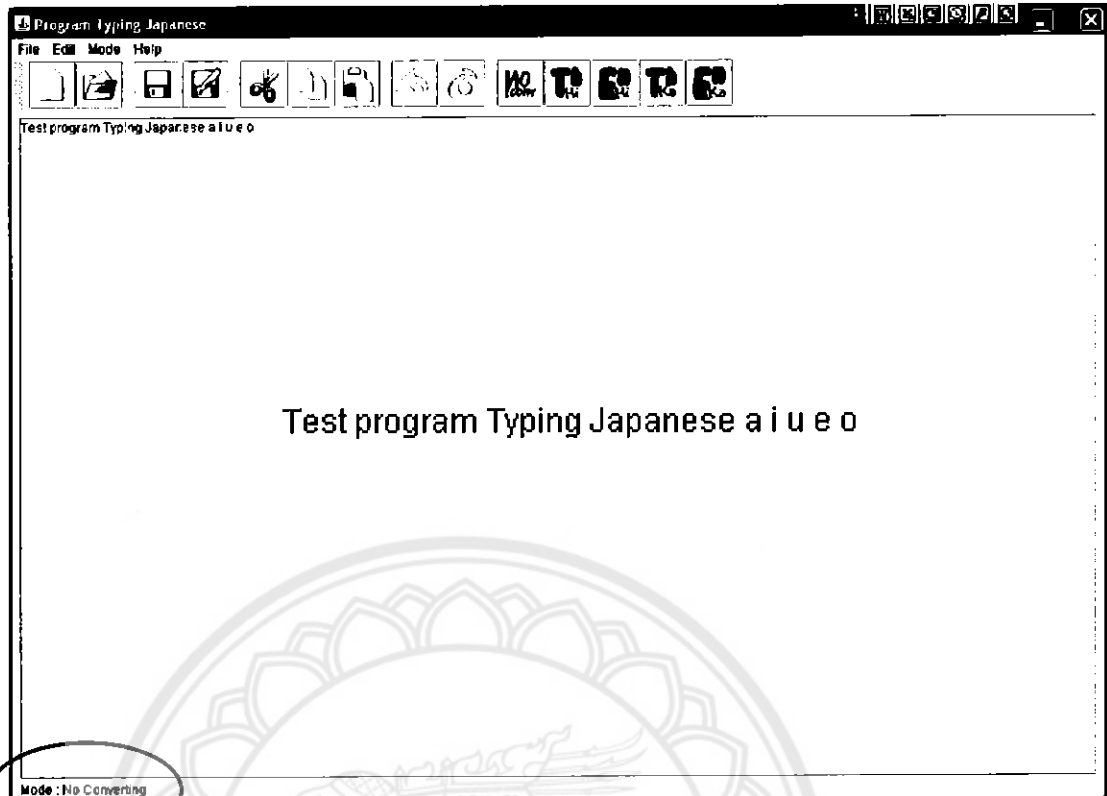
รูปที่ 4.1 แสดงการรับอินพุตที่จากคีย์บอร์ดเป็นภาษาไทย
เลือกโหมดไม่มีการแปลง



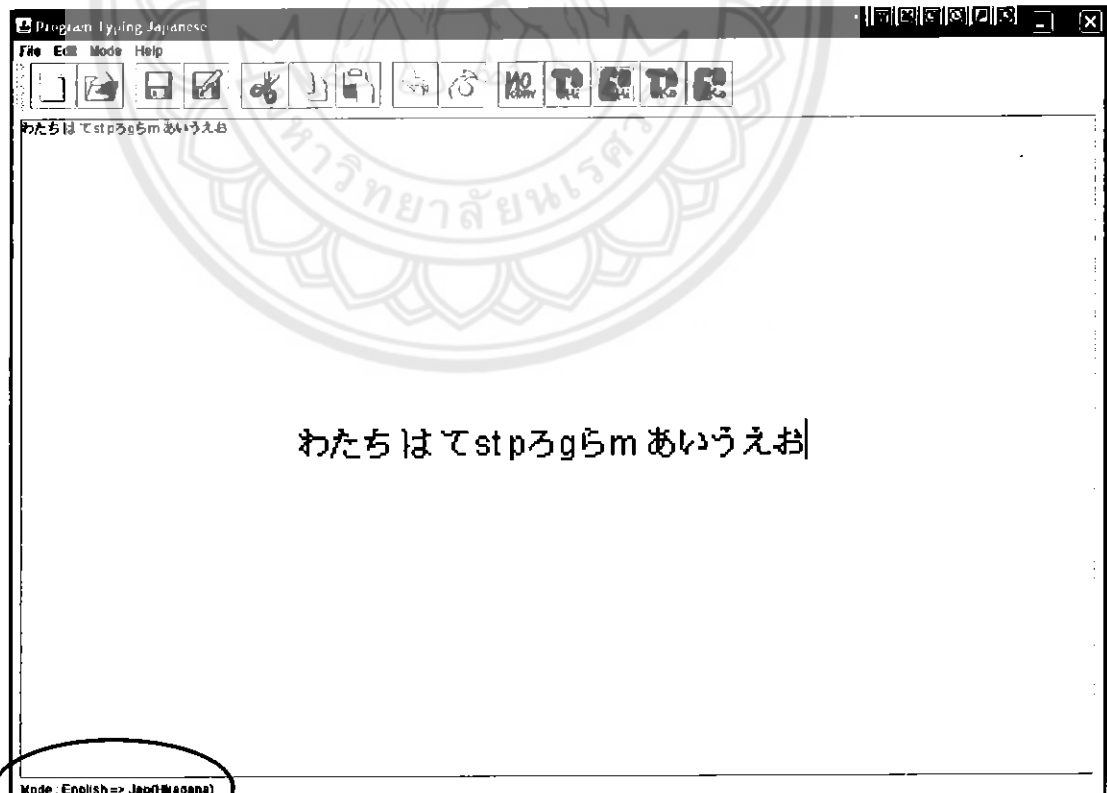
รูปที่ 4.2 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาไทย
เลือกโหมดอักษรญี่ปุ่นแบบฮิรางานะ



รูปที่ 4.3 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาไทย
เลือกโหมดอักษรญี่ปุ่นแบบคาตากานะ



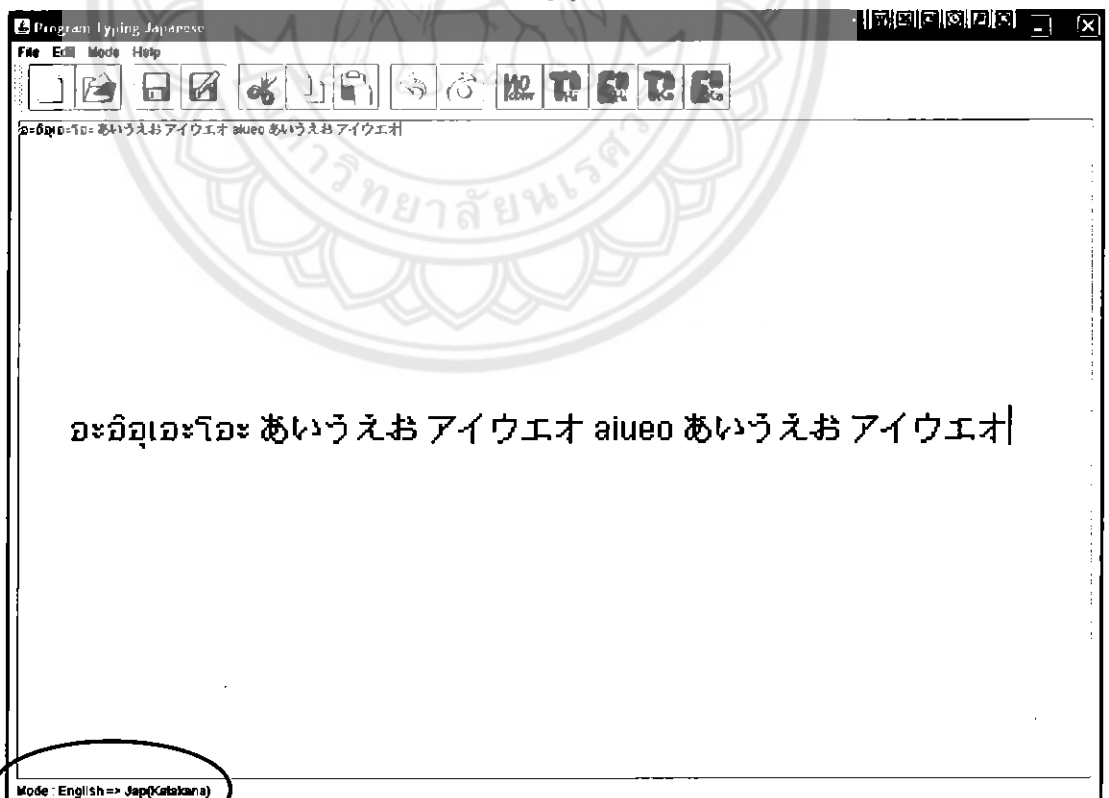
รูปที่ 4.4 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาอังกฤษ
เลือกโหมดไม่มีการแปลง



รูปที่ 4.5 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาอังกฤษ
เลือกโหมดอักษรญี่ปุ่นแบบฮิรางานะ



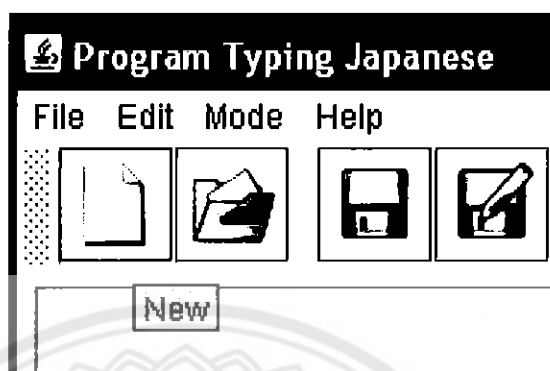
รูปที่ 4.6 แสดงการรับอินพุตจากคีย์บอร์ดเป็นภาษาอังกฤษ
เลือกโหมดอักษรญี่ปุ่นแบบคาตากานะ



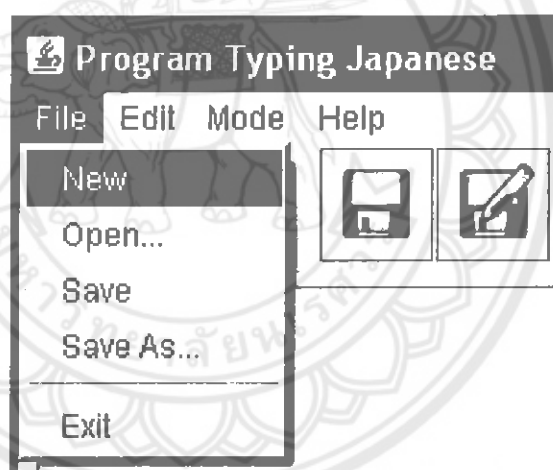
รูปที่ 4.7 แสดงการรับอินพุตจากคีย์บอร์ดเป็น
ภาษาไทยและภาษาอังกฤษเลือกทุกโหมด

4.2 การทดสอบโปรแกรมส่วนฟังก์ชันการใช้งาน

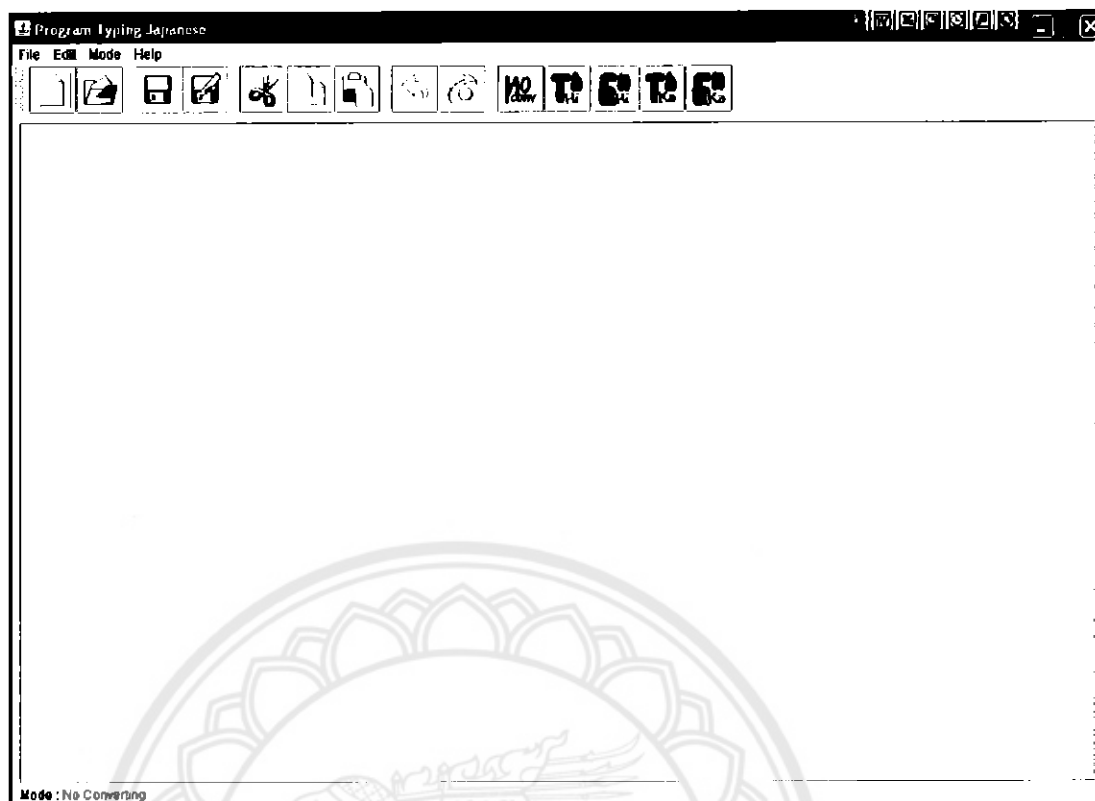
การทดสอบโปรแกรมช่วยพิมพ์ภาษาญี่ปุ่นในส่วนของฟังก์ชันการใช้งานนี้เป็นการตรวจสอบว่าฟังก์ชันต่าง ๆ ของโปรแกรมสามารถทำงานได้ถูกต้อง



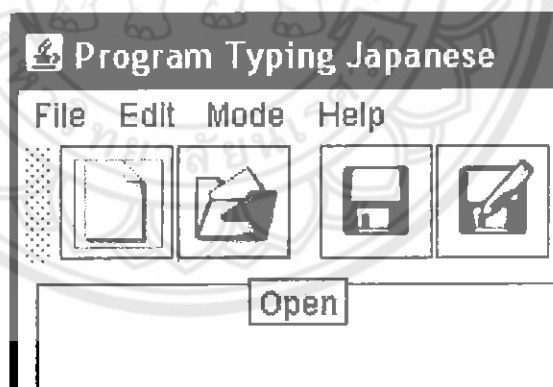
รูปที่ 4.8 เลือกฟังก์ชัน New จาก Toolbar



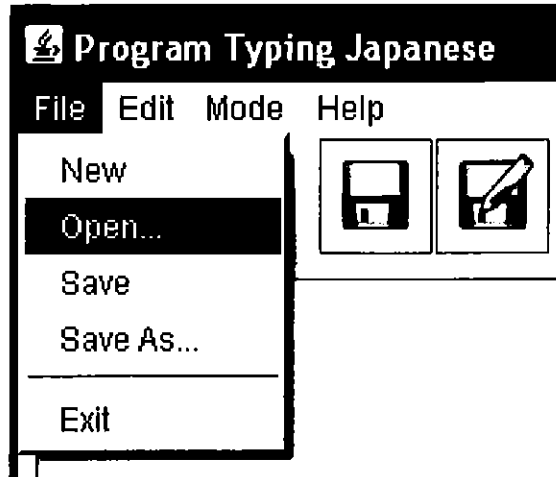
รูปที่ 4.9 เลือกฟังก์ชัน New จาก MenuFile



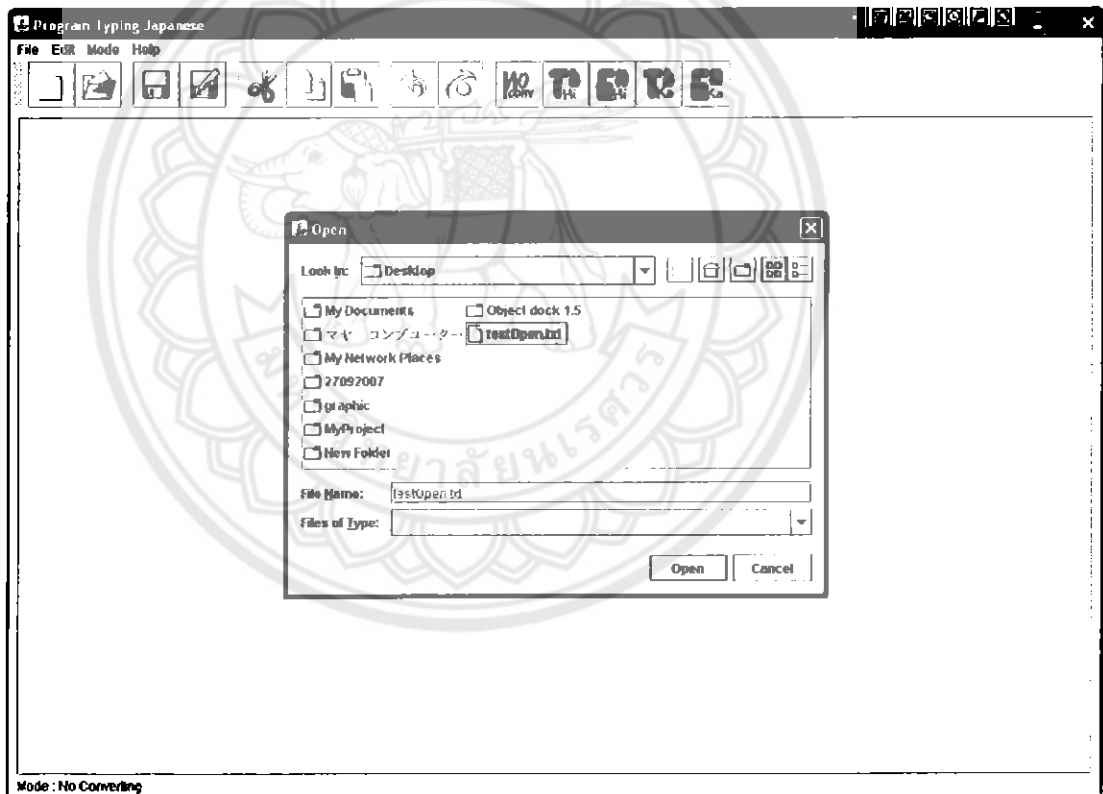
รูปที่ 4.10 ผลการทดสอบฟังก์ชัน New



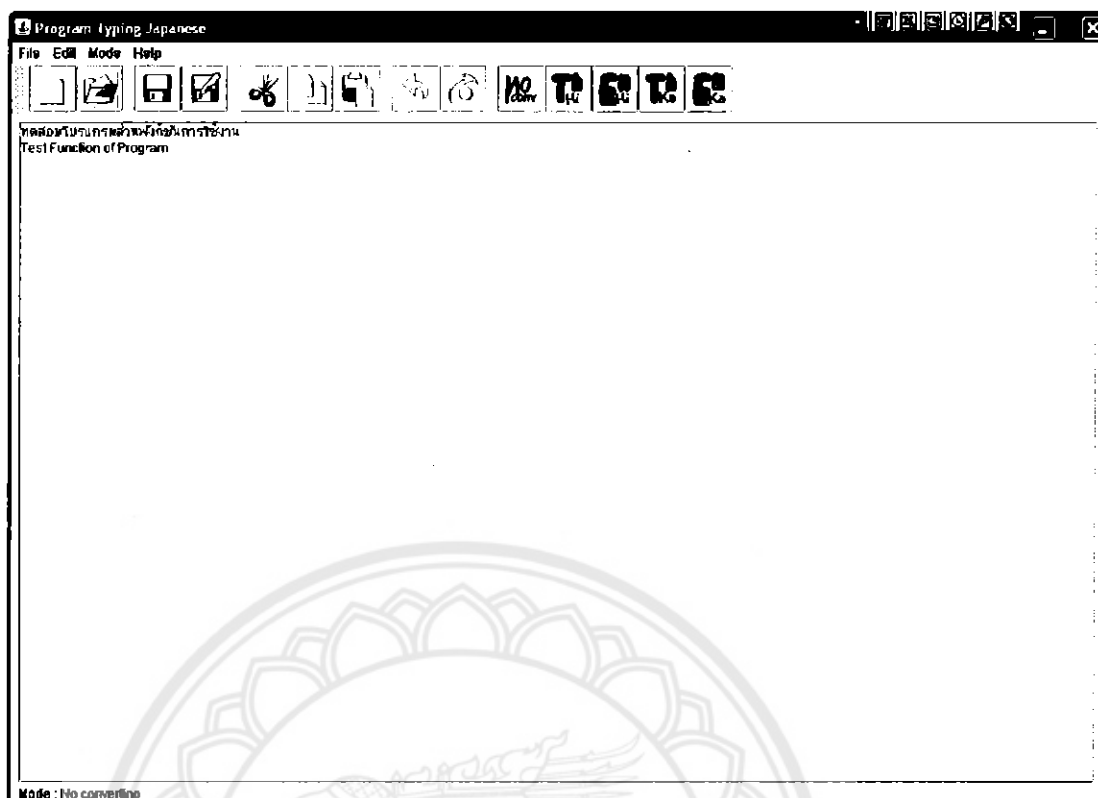
รูปที่ 4.11 เลือกฟังก์ชัน Open จาก Toolbar



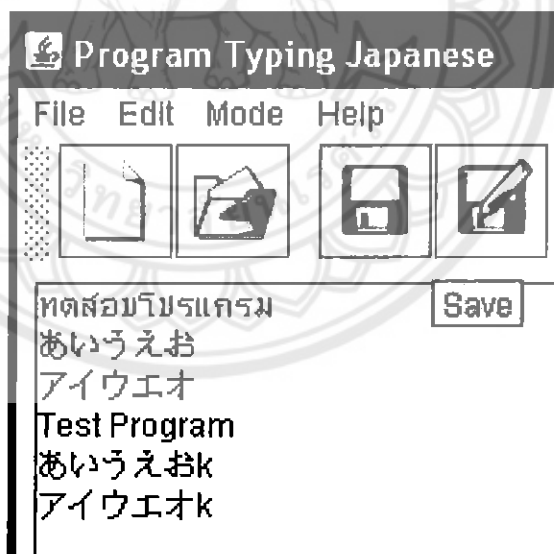
รูปที่ 4.12 เลือกฟังก์ชัน Open จาก MenuFile



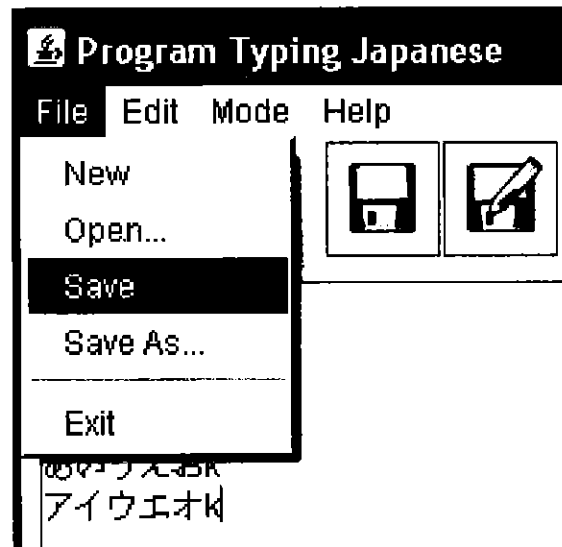
รูปที่ 4.13 เลือกไฟล์ที่จะทำการ Open



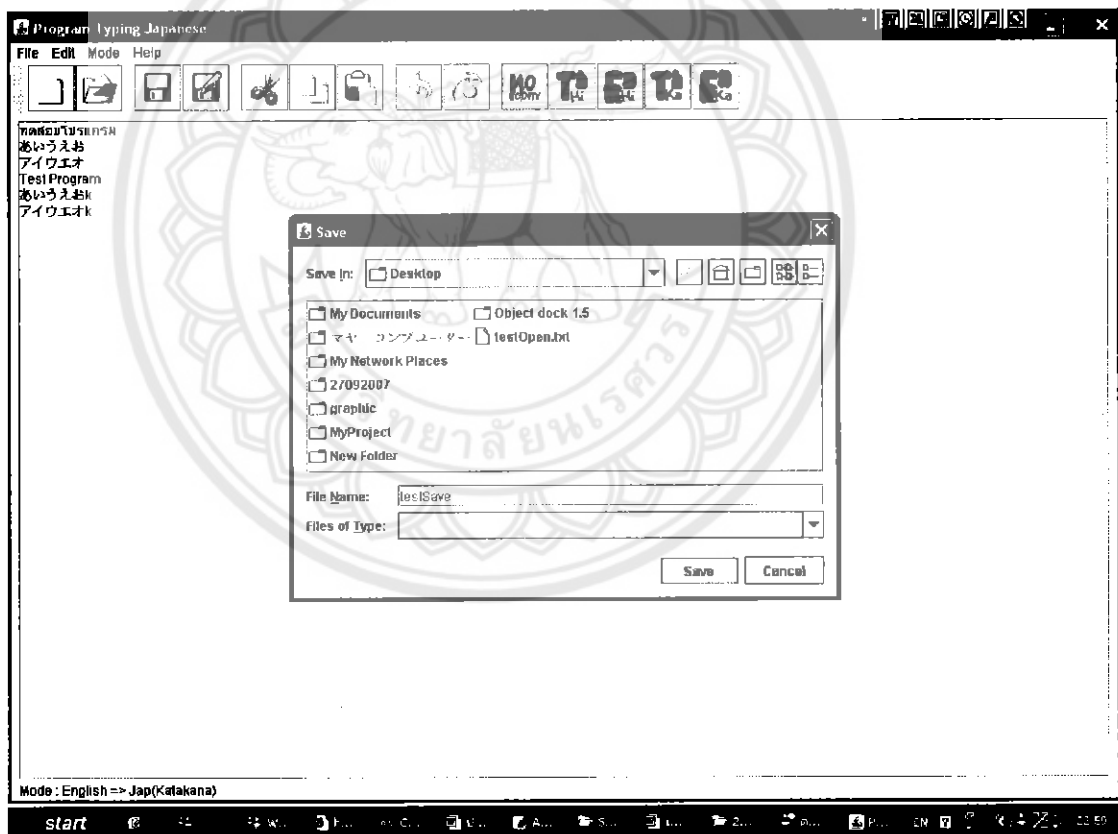
รูปที่ 4.14 ผลการทดสอบฟังก์ชัน Open



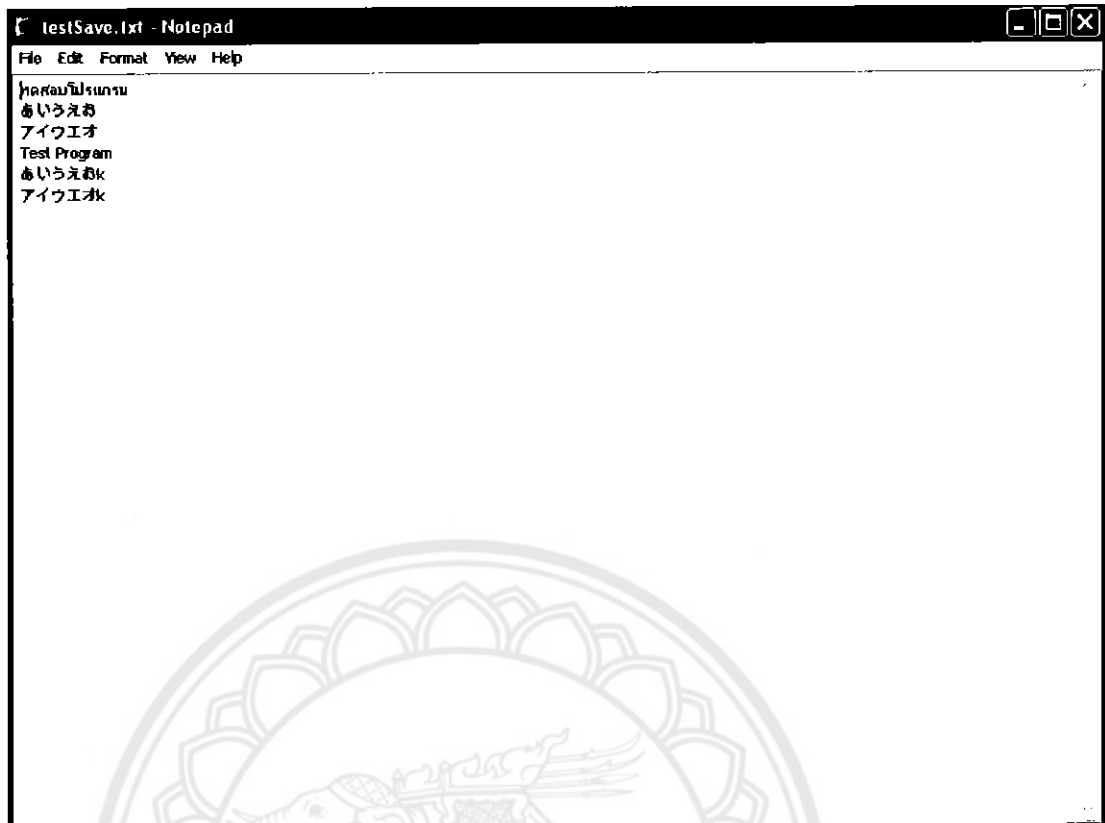
รูปที่ 4.15 เลือกฟังก์ชัน Save จาก Toolbar



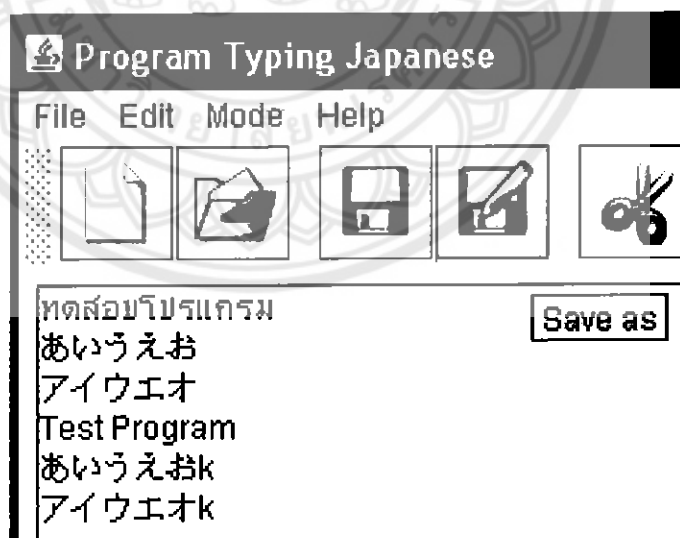
รูปที่ 4.16 เลือกฟังก์ชัน Save จาก MenuFile



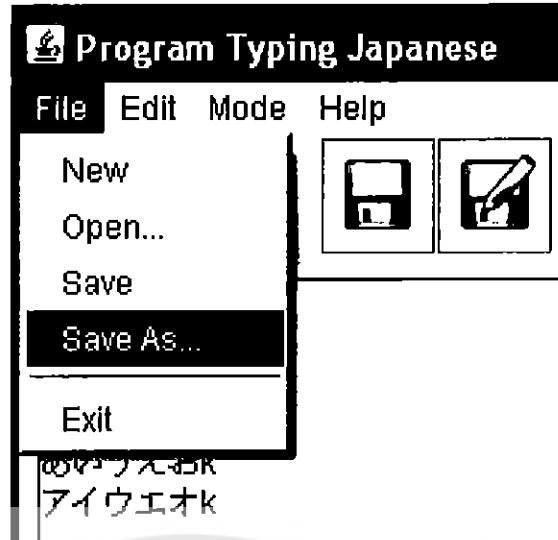
รูปที่ 4.17 ตั้งชื่อไฟล์ก่อน Save



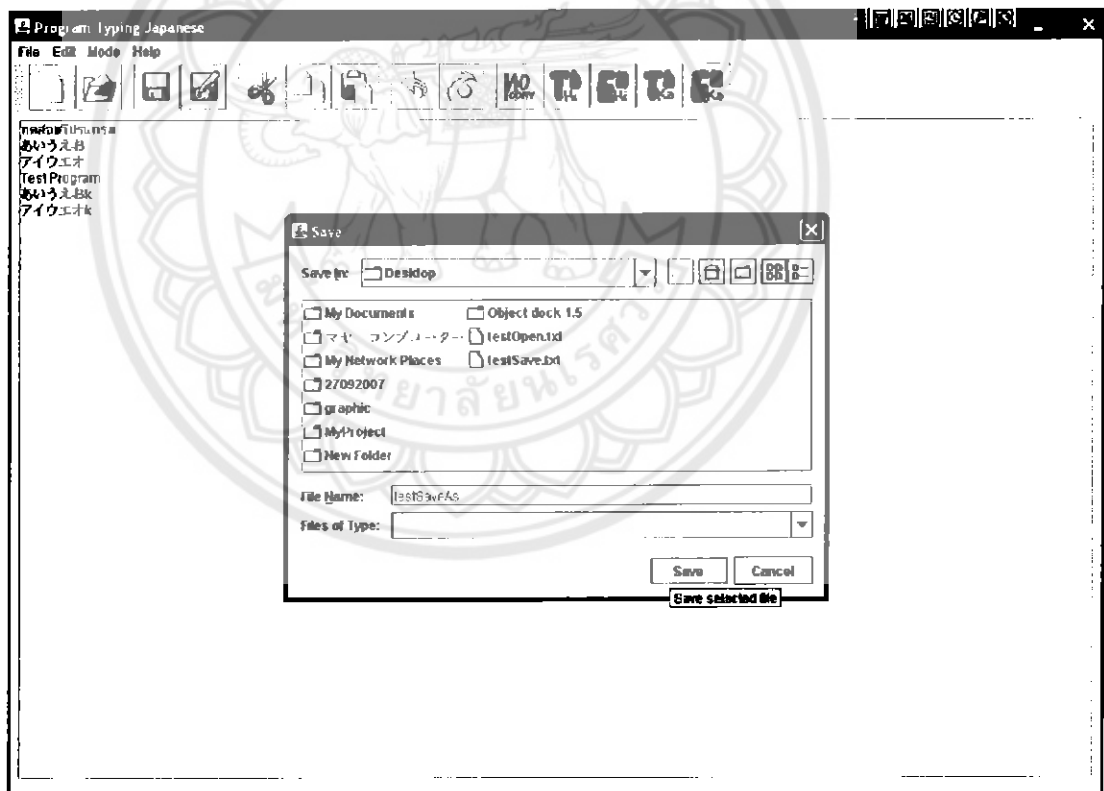
รูปที่ 4.18 ผลการทดสอบฟังก์ชัน Save เปิดโดยใช้โปรแกรม Notepad



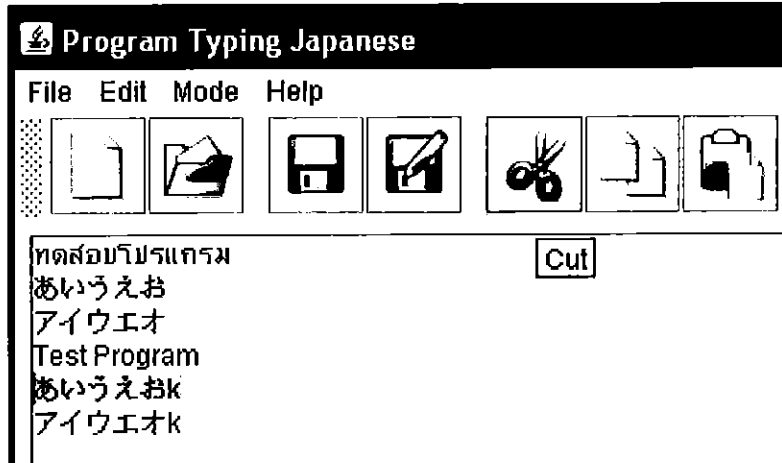
รูปที่ 4.19 เลือกฟังก์ชัน Save as จาก Toolbar



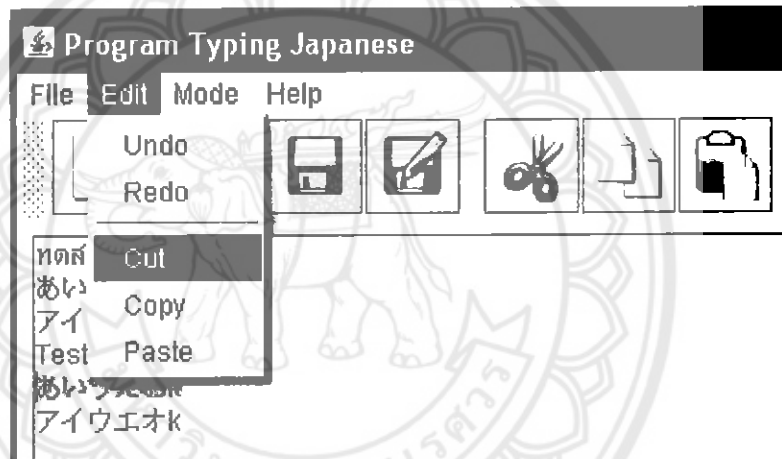
รูปที่ 4.20 เลือกฟังก์ชัน Save as จาก MenuFile



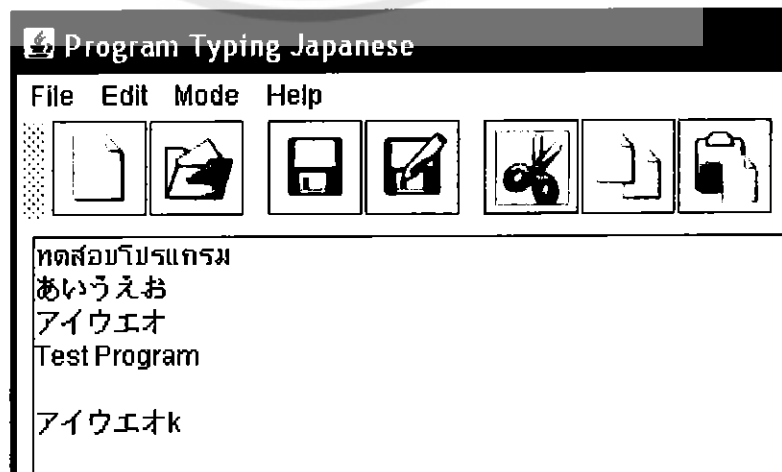
รูปที่ 4.21 ตั้งชื่อไฟล์ก่อน Save as



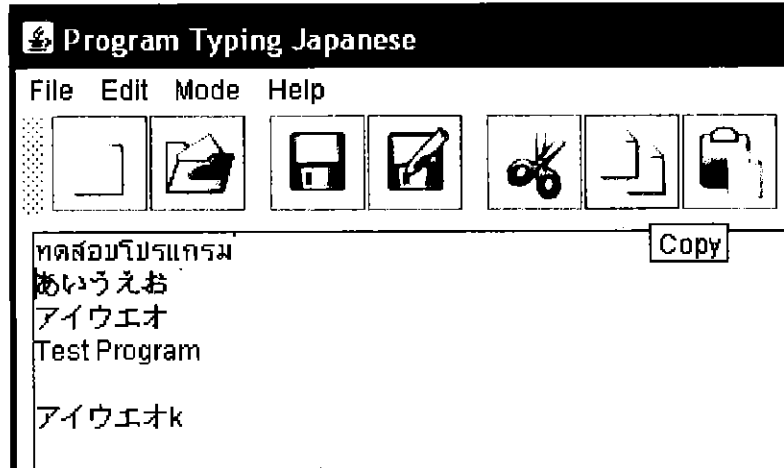
รูปที่ 4.22 เลือกฟังก์ชัน Cut จาก Toolbar



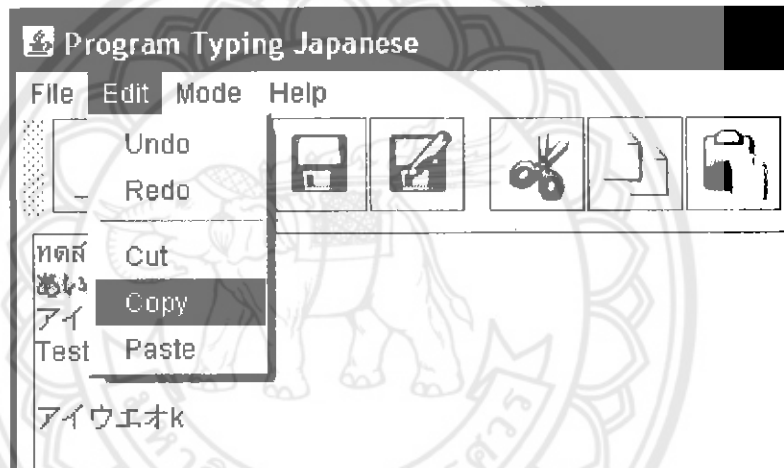
รูปที่ 4.23 เลือกฟังก์ชัน Cut จาก MenuEdit



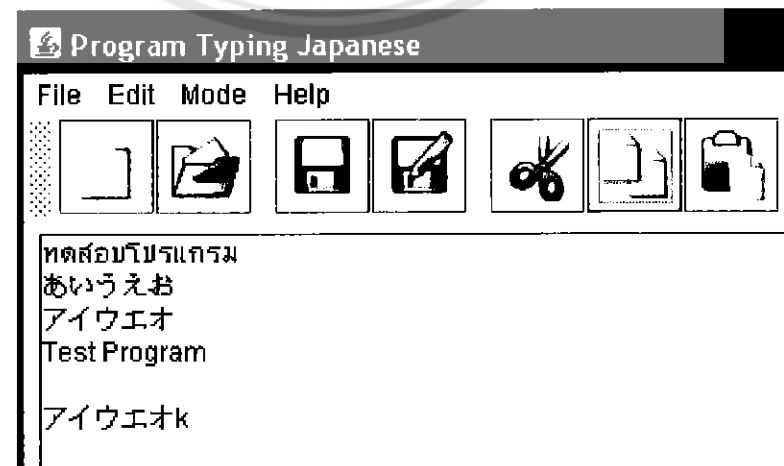
รูปที่ 4.24 ผลการทดสอบฟังก์ชัน Cut



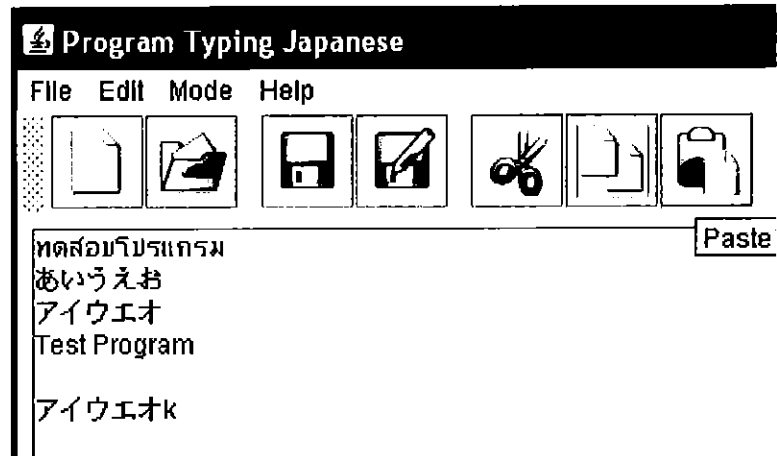
รูปที่ 4.25 เลือกฟังก์ชัน Copy จาก Toolbar



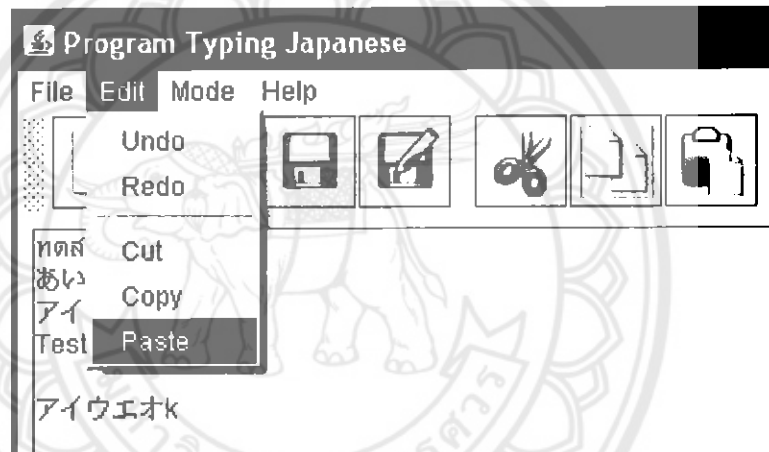
รูปที่ 4.26 เลือกฟังก์ชัน Copy จาก MenuEdit



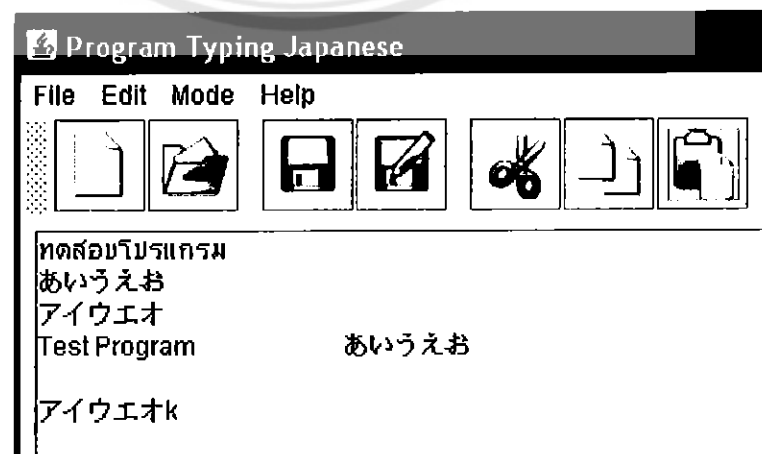
รูปที่ 4.27 ผลการทดสอบฟังก์ชัน Copy



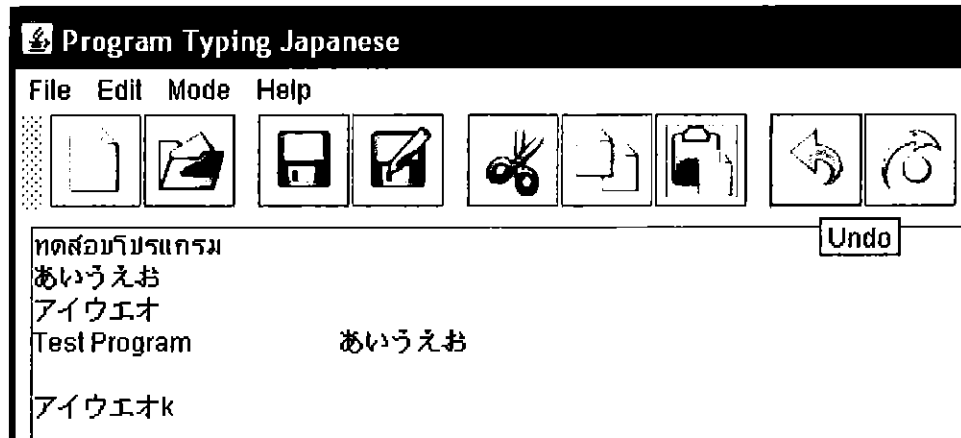
รูปที่ 4.28 เลือกฟังก์ชัน Paste จาก Toolbar



รูปที่ 4.29 เลือกฟังก์ชัน Paste จาก MenuEdit



รูปที่ 4.30 ผลการทดสอบฟังก์ชัน Paste



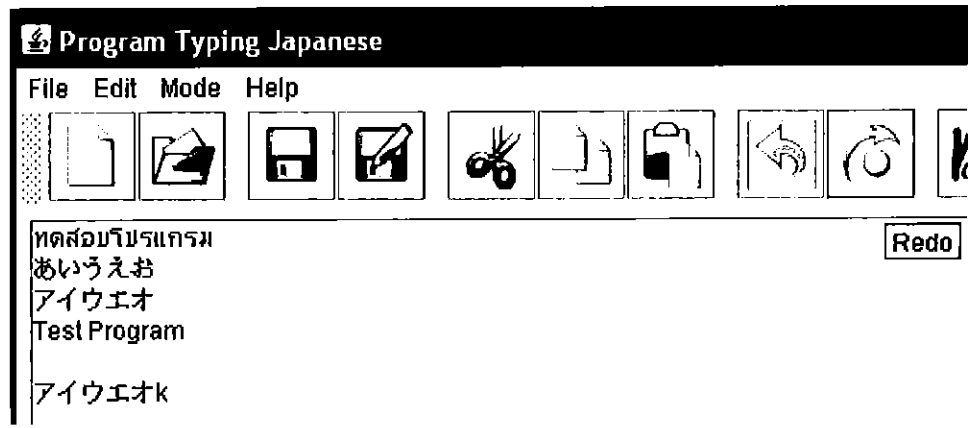
รูปที่ 4.31 เลือกฟังก์ชัน Undo จาก Toolbar



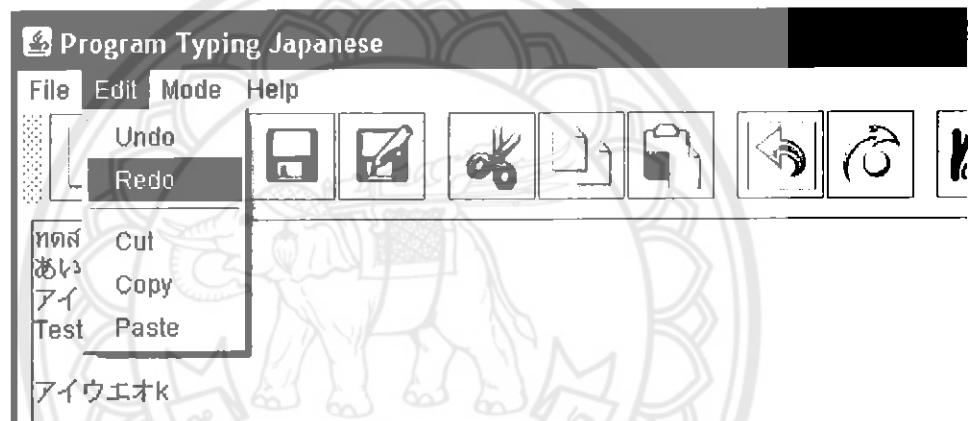
รูปที่ 4.32 เลือกฟังก์ชัน Undo จาก MenuEdit



รูปที่ 4.33 ผลการทดสอบฟังก์ชัน Undo



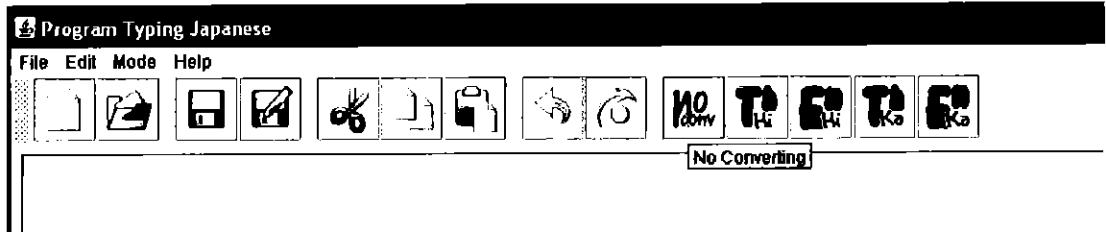
รูปที่ 4.34 เลือกฟังก์ชัน Redo จาก Toolbar



รูปที่ 4.35 เลือกฟังก์ชัน Redo จาก MenuEdit



รูปที่ 4.36 ผลการทดสอบฟังก์ชัน Redo



รูปที่ 4.37 เลือกฟังก์ชัน No converting จาก Toolbar



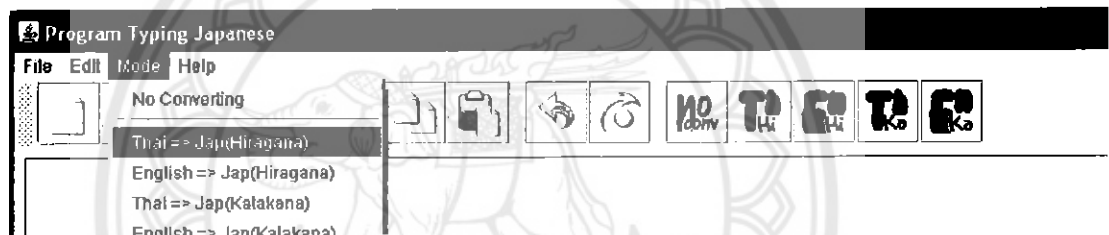
รูปที่ 4.38 เลือกฟังก์ชัน No converting จาก MenuMode



รูปที่ 4.39 ผลการทดสอบฟังก์ชัน No converting



รูปที่ 4.40 เลือกฟังก์ชัน Thai to Japanese(Hiragana) จาก Toolbar



รูปที่ 4.41 เลือกฟังก์ชัน Thai to Japanese(Hiragana) จาก MenuMode



รูปที่ 4.42 ผลการทดสอบฟังก์ชัน Thai to Japanese(Hiragana)



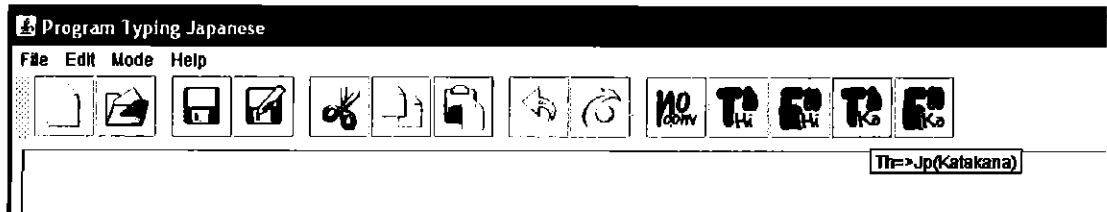
รูปที่ 4.43 เลือกฟังก์ชัน English to Japanese(Hiragana) จาก Toolbar



รูปที่ 4.44 เลือกฟังก์ชัน English to Japanese(Hiragana) จาก MenuMode



รูปที่ 4.45 ผลการทดสอบฟังก์ชัน English to Japanese(Hiragana)



รูปที่ 4.46 เลือกฟังก์ชัน Thai to Japanese(Katakana) จาก Toolbar



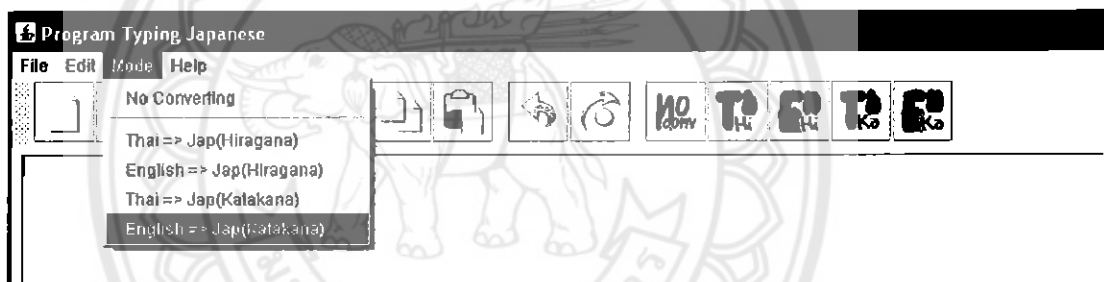
รูปที่ 4.47 เลือกฟังก์ชัน Thai to Japanese(Katakana) จาก MenuMode



รูปที่ 4.8 ผลการทดสอบฟังก์ชัน Thai to Japanese(Katakana)



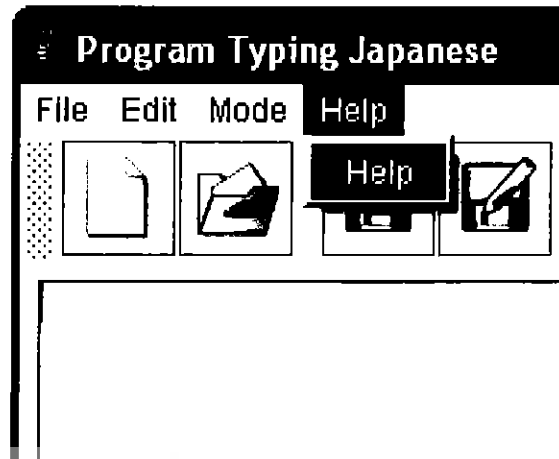
รูปที่ 4.49 เลือกฟังก์ชัน English to Japanese(Katakana) จาก Toolbar



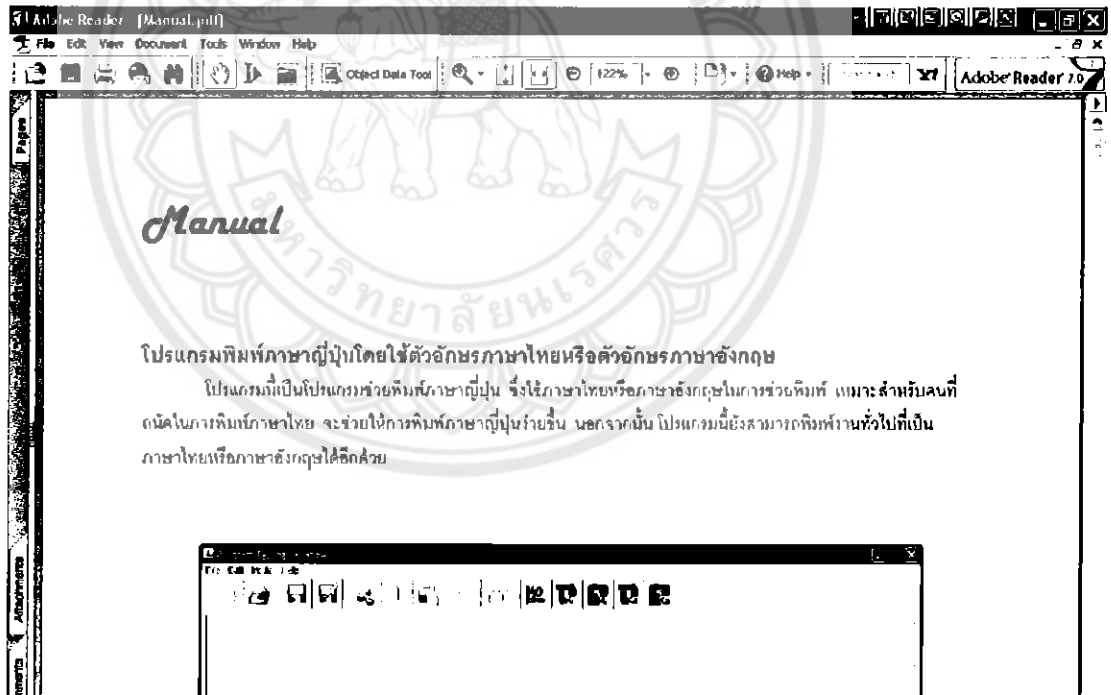
รูปที่ 4.50 เลือกฟังก์ชัน English to Japanese(Katakana) จาก MenuMode



รูปที่ 4.51 ผลการทดสอบฟังก์ชัน English to Japanese(Katakana)



รูปที่ 4.52 เลือกฟังก์ชัน Help จาก MenuHelp



รูปที่ 4.53 ผลการทดสอบฟังก์ชัน Help

รายการ	โปรแกรม ช่วยพิมพ์ ภาษาญี่ปุ่น
Input	
รับตัวอักษรภาษาไทยจากคีย์บอร์ด	✓
รับตัวอักษรภาษาอังกฤษจากคีย์บอร์ด	✓
Search	
จัดตัวอักษรภาษาไทยเป็นคำเพื่อนำไปเทียบกับตารางเทียบเสียงภาษาไทย	✓
จัดตัวอักษรภาษาอังกฤษเป็นคำเพื่อนำไปเทียบกับตารางเทียบเสียงภาษาอังกฤษ	✓
นำคำอ่านภาษาไทยไปเทียบกับตารางเทียบเสียงภาษาไทย	✓
นำคำอ่านภาษาอังกฤษไปเทียบกับตารางเทียบเสียงภาษาอังกฤษ	✓
Output	
แสดงตัวอักษรภาษาญี่ปุ่นแบบฮิรางานะจากคำอ่านที่สะกดเป็นภาษาไทย	✓
แสดงตัวอักษรภาษาญี่ปุ่นแบบฮิรางานะจากคำอ่านที่สะกดเป็นภาษาอังกฤษ	✓
แสดงตัวอักษรภาษาญี่ปุ่นแบบคาตากานะจากคำอ่านที่สะกดเป็นภาษาไทย	✓
แสดงตัวอักษรภาษาญี่ปุ่นแบบคาตากานะจากคำอ่านที่สะกดเป็นภาษาอังกฤษ	✓
แสดงตัวอักษรภาษาไทยแบบที่ไม่มีการแปลง	✓
แสดงตัวอักษรภาษาอังกฤษแบบที่ไม่มีการแปลง	✓

ตารางที่ 4.1 แสดงผลการทดสอบโปรแกรมส่วนประมวลผล

รายการ	โปรแกรม ช่วยพิมพ์ ภาษาญี่ปุ่น
ฟังก์ชันการใช้งาน	
เปิดหน้าต่างกระดาษใหม่ (New)	✓
เปิด ไฟล์ (Open)	✓
บันทึก ไฟล์ (Save)	✓
บันทึกเป็นไฟล์ใหม่ (Save as)	✓
ตัดข้อความ (Cut)	✓
คัดลอกข้อความ (Copy)	✓
วางข้อความ (Paste)	✓
ยกเลิกการทำงานล่าสุด (Undo)	✓
ทำสิ่งที่เคยยกเลิกการทำงานล่าสุด (Redo)	✓
โหมดไม่มีการแปลง (No converting)	✓
โหมดแปลงจากไทยเป็นญี่ปุ่นแบบฮิรางานะ (Thai => Jap(Hiragana))	✓
โหมดแปลงจากไทยเป็นญี่ปุ่นแบบคาตากานะ (Thai => Jap(Katakana))	✓
โหมดแปลงจากอังกฤษเป็นญี่ปุ่นแบบฮิรางานะ (Eng => Jap(Hiragana))	✓
โหมดแปลงจากอังกฤษเป็นญี่ปุ่นแบบคาตากานะ (Eng => Jap(Katakana))	✓

ตารางที่ 4.2 แสดงผลการทดสอบ โปรแกรมส่วนฟังก์ชันการทำงาน

บทที่ 5

บทสรุป

โครงการนี้ได้พัฒนาโปรแกรมเพื่อใช้สำหรับพิมพ์ภาษาญี่ปุ่นแบบฮิรางานะและคาตากานะ จากคำอ่านที่สะกดเป็นภาษาไทยหรือภาษาอังกฤษ และยังสามารพิมพ์ภาษาไทยหรือภาษาอังกฤษ โดยไม่มีการแปลงเป็นภาษาญี่ปุ่น

5.1 วิเคราะห์ผลการทดลอง

จากผลการทดลองบทที่ 4 ซึ่งได้แบ่งการทดสอบโปรแกรมออกเป็น 2 ส่วน คือส่วนประมวลผล และส่วนฟังก์ชันการใช้งานของโปรแกรม การทดสอบโปรแกรมได้ผลการทดลอง ดังตารางที่ 5.1

รายการ	โปรแกรมช่วยพิมพ์ภาษาญี่ปุ่น
แสดงผลภาษาญี่ปุ่นแบบฮิรางานะและคาตากานะจากคำอ่านที่สะกดเป็นภาษาไทย	✓
แสดงผลภาษาญี่ปุ่นแบบฮิรางานะและคาตากานะจากคำอ่านที่สะกดเป็นภาษาอังกฤษ	✓
แสดงตัวอักษรภาษาไทยและภาษาอังกฤษแบบที่ไม่มีการแปลง	✓

ตารางที่ 5.1 สรุปผลการทดสอบโปรแกรม

Requirements	โปรแกรมช่วยพิมพ์ภาษาญี่ปุ่น
รับอินพุตจากคีย์บอร์ดเป็นตัวอักษรภาษาไทยแล้วแปลงเป็นตัวอักษรภาษาญี่ปุ่น	✓
รับอินพุตจากคีย์บอร์ดเป็นตัวอักษรภาษาอังกฤษแล้วแปลงเป็นตัวอักษรภาษาญี่ปุ่น	✓
สามารถพิมพ์ตัวอักษรภาษาไทยแบบไม่มีการแปลง	✓
สามารถพิมพ์ตัวอักษรภาษาอังกฤษแบบไม่มีการแปลง	✓
มีฟังก์ชันพื้นฐานสำหรับโปรแกรมที่ใช้พิมพ์เอกสารทั่วไป เช่น copy, cut, save, save as open เป็นต้น	✓
แสดงผลได้ทั้งฮิรางานะและคาตากานะ	✓
ตอบสนองทันที(ภายใน 1 วินาที)	✓

ตารางที่ 5.2 แสดงการตอบสนอง requirements

จากตารางที่ 5.1 และ 5.2 เครื่องหมายถูก แสดงว่า โปรแกรมสามารถทำงานตามรายการนั้น ๆ ได้ ซึ่งจากตารางที่ 5.1 และ 5.2 พบว่า โปรแกรมช่วยพิมพ์ภาษาญี่ปุ่นสามารถพิมพ์ภาษาญี่ปุ่นแบบฮิรางานะและคาตากานะได้จากคำอ่านที่สะกดเป็นภาษาไทยหรือภาษาอังกฤษ ซึ่งจะตอบสนองทันที(ภายใน 1 วินาที) และสามารถพิมพ์ภาษาไทยหรือภาษาอังกฤษได้โดยไม่ต้องแปลงเป็นภาษาญี่ปุ่นด้วย นอกจากนี้ยังมีฟังก์ชันการทำงานพื้นฐาน เช่น การเปิดไฟล์ การบันทึกไฟล์ การคัดลอกข้อความ เป็นต้น

5.2 ปัญหาและแนวทางแก้ไข

ในระหว่างการทำโครงการนี้ได้ประสบปัญหาทางการเขียน โปรแกรมเป็นส่วนใหญ่ ได้แก่

1. ปัญหาเกี่ยวกับการใช้ textarea ร่วมกับ key event ชนิด key Type เนื่องจาก key Type จะทำงานก่อน textarea เมื่อรับอินพุตจากคีย์บอร์ดทำการประมวลผล แล้วนำผลลัพธ์ไปแสดงผลบน textarea จะทำให้เกิดค่าค้างที่ textarea จึงแก้ไขด้วยการใช้ key Release แทน

2. ปัญหาด้านภาษา คำอ่านภาษาญี่ปุ่นเราใช้การเทียบเสียงมาจากคำอ่านภาษาอังกฤษ ซึ่งถ้าแปลงเป็นคำอ่านภาษาไทยสามารถอ่านได้หลายแบบ เช่น คำว่า “O” อ่านได้ทั้ง โอ หรือ โอะ แต่โปรแกรมนี้เทียบเสียงได้เฉพาะ โอะ เท่านั้น การจะใช้งานโปรแกรมนี้จึงจำเป็นต้องศึกษตารางเทียบเสียงก่อน

3. ปัญหาด้านการแสดงผลตัวอักษรภาษาญี่ปุ่น โปรแกรมนี้จะแสดงผลเป็นภาษาญี่ปุ่นได้ เมื่อเครื่องคอมพิวเตอร์นั้นมีวินโดวส์ที่รองรับภาษาญี่ปุ่นได้

5.3 สรุปผลการทดลอง

โปรแกรมช่วยพิมพ์ภาษาญี่ปุ่นแบบฮิรางานะและคาตากานะจากคำอ่านที่สะกดเป็นภาษาไทยหรือภาษาอังกฤษนี้ ช่วยให้ผู้ที่อาจจะไม่ถนัดพิมพ์ภาษาอังกฤษสามารถพิมพ์ภาษาญี่ปุ่นโดยใช้ตัวอักษรภาษาไทยแทนได้ นอกจากนี้โปรแกรมยังสามารถพิมพ์ตัวอักษรภาษาไทยหรือภาษาอังกฤษโดยไม่ต้องมีการแปลงเป็นภาษาญี่ปุ่นได้ด้วย และยังมีฟังก์ชันต่าง ๆ ที่จะอำนวยความสะดวกแก่ผู้ใช้งาน

5.4 ข้อเสนอแนะ

1. โปรแกรมนี้สามารถนำไปพัฒนาต่อได้ โดยอาจจะใช้ในการช่วยพิมพ์ภาษาอื่น ๆ ได้ เช่น ภาษาจีน ภาษาฝรั่งเศส เป็นต้น
2. ในการพัฒนาต่อไป อาจจะเพิ่มการทำคีย์บอร์ด shortcut ของโหมดแต่ละโหมด เพื่อความสะดวกในการพิมพ์
3. ในการพัฒนาต่อไป อาจจะพัฒนา Algorithm เพื่อให้โปรแกรมสามารถทำงานได้รวดเร็วขึ้น
4. ในการพัฒนาต่อไป อาจจะเพิ่มโหมดที่สามารถอ่านไฟล์เข้ามาแล้วสามารถทำการแปลงเป็นภาษาญี่ปุ่นได้เลย

เอกสารอ้างอิง

- [1] วีระศักดิ์ ชิงฉาวร. Java Programming Volume I(JavaSE 5.0). กรุงเทพมหานคร : ซีเอ็ดดูเคชั่น. 2549.
- [2] วีระศักดิ์ ชิงฉาวร. Java Programming Volume II. กรุงเทพมหานคร : ซีเอ็ดดูเคชั่น. 2548.
- [3] กิตติ ภัคดีวัฒนกุล. คัมภีร์ Java เล่ม 1. กรุงเทพมหานคร : เคทีพี คอมพ์ แอนด์ คอนซัลท์. 2546.
- [4] สุทธิ พงศาสกุลชัย. คัมภีร์ Java เล่ม 2. กรุงเทพมหานคร : เคทีพี คอมพ์ แอนด์ คอนซัลท์. 2548.
- [5] มุลนิธิวิกิมีเดีย. “กะตะกะนะ”. [Online]. Available:
<http://th.wikipedia.org/wiki/%E0%B8%84%E0%B8%B0%E0%B8%95%E0%B8%B0%E0%B8%84%E0%B8%B0%E0%B8%99%E0%B8%B0>
- [6] มุลนิธิวิกิมีเดีย. “อิระงะนะ”. [Online]. Available:
<http://th.wikipedia.org/wiki/%E0%B8%AE%E0%B8%B4%E0%B8%A3%E0%B8%B0%E0%B8%87%E0%B8%B0%E0%B8%99%E0%B8%B0>
- [7] lingvo.exteen.com. “Nihongo”. [Online]. Available:
<http://lingvo.exteen.com/category-Nihongo>
- [8] arukithai.com. “learn_jp”. [Online]. Available:
http://www.arukithai.com/th/learn_jp/default-5.asp

ภาคผนวก ก.

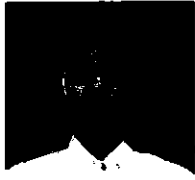
ตารางเทียบเสียงตัวอักษรภาษาญี่ปุ่น

Thai	Eng	Japanese	
		Hiragana	Katakana
อะ, อา	A	あ	ア
อิ	I	い	イ
อุ	U	う	ウ
เอะ	E	え	エ
โอะ	O	お	オ
คะ, คา	KA	か	カ
คิ	KI	き	キ
คุ	KU	く	ク
เคะ	KE	け	ケ
โคะ	KO	こ	コ
สะ, ซา	SA	さ	サ
ชิ, ซึ	SHI	し	シ
สุ	SU	す	ス
เซะ	SE	せ	セ
โซะ	SO	そ	ソ
ทะ, ทา, ตะ, ตา	TA	た	タ
ชิ, ทิ	CHI	ち	チ
ทุ, ซึ, สึ	TSU	つ	ツ
เทะ, เตะ	TE	て	テ
โทะ, โตะ	TO	と	ト
นะ, นา	NA	な	ナ
นิ	NI	に	ニ
นุ	NU	ぬ	ヌ
เนะ	NE	ね	ネ
โนะ	NO	の	ノ
ฮะ, ฮา	HA	は	ハ
ฮิ	HI	ひ	ヒ
ฮุ, ฟุ	HU	ふ	フ
เฮะ	HE	へ	ヘ
โฮะ	HO	ほ	ホ
มะ, มา	MA	ま	マ
มิ	MI	み	ミ
มุ	MU	む	ム

เมะ	ME	め	メ
โมะ	MO	も	モ
ยะ, ยา	YA	や	ヤ
ยุ	YU	ゆ	ユ
โยะ	YO	よ	ヨ
ระ, รา	RA	ら	ラ
ริ	RI	り	リ
รุ	RU	る	ル
เระ	RE	れ	レ
โระ	RO	ろ	ロ
วะ, วา	WA	わ	ワ
โวะ	WO	を	ヲ
กะ, กา	GA	が	ガ
กิ	GI	ぎ	ギ
กุ	GU	ぐ	グ
เกะ	GE	げ	ゲ
โกะ	GO	ご	ゴ
ชะ, ชา	ZA	ざ	ザ
จิ	JI	じ	ジ
จุ	ZU	ず	ズ
เซะ	ZE	ぜ	ゼ
โซะ	ZO	ぞ	ゾ
คะ, คา	DA	だ	ダ
คิ	DI	ぢ	ヂ
คู	DU	づ	ヅ
เดะ	DE	で	デ
โดะ	DO	ど	ド
บะ, บา	BA	ば	バ
บิ	BI	び	ビ
บุ	BU	ぶ	ブ
เบะ	BE	べ	ベ
โบะ	BO	ぼ	ボ
ปะ, ปา	PA	ぱ	パ
ปิ	PI	ぴ	ピ
ปู	PU	ぷ	プ
เฟะ	PE	へ	ヘ

โปะ	PO	ぼ	ボ
คยะ	KYA	きや	キヤ
คยุ	KYU	きゆ	キユ
คโย	KYO	きよ	キヨ
สยะ	SHA	しゃ	シヤ
สยุ	SHU	しゆ	シユ
สโย	SHO	しよ	シヨ
ชยะ	CHA	ちや	チャ
ชยุ	CHU	ちゆ	チュ
ชโย	CHO	ちよ	チョ
นยะ	NYA	にや	ニヤ
นยุ	NYU	にゆ	ニユ
นโย	NYO	によ	ニヨ
ฮยะ	HYA	ひや	ヒヤ
ฮยุ	HYU	ひゆ	ヒユ
ฮโย	HYO	ひよ	ヒヨ
มยะ	MYA	みや	ミヤ
มยุ	MYU	みゆ	ミュ
มโย	MYO	みよ	ミヨ
รยะ	RYA	りや	リヤ
รยุ	RYU	りゆ	リュ
รโย	RYO	りよ	リヨ
กยะ	GYA	ぎや	ギヤ
กยุ	GYU	ぎゆ	ギユ
กโย	GYO	ぎよ	ギヨ
จยะ	JA	じゃ	ジャ
จยุ	JU	じゆ	ジュ
จโย	JO	じよ	ジョ
บยะ	BYA	びや	ビヤ
บยุ	BYU	びゆ	ビユ
บโย	BYO	びよ	ビヨ
ปยะ	PYA	ぴや	ピヤ
ปยุ	PYU	ぴゆ	ピユ
ปโย	PYO	ぴよ	ピヨ
อิน	NN	ん	ン

ประวัติผู้เขียนโครงการ



ชื่อ นางสาวกอบพร เชื้อบุญมี
 ภูมิลำเนา 63/102 หมู่ที่ 7 ตำบลท่าเสา อำเภอเมือง
 จังหวัดอุตรดิตถ์ 53000

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนอุตรดิตถ์ครุณี
- ปัจจุบันกำลังศึกษาในระดับชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร

E-mail k_p_crescent@hotmail.com



ชื่อ นางสาวมณฑนา เนียมคำ
 ภูมิลำเนา 356/4 ถนนสวรรค่วิถี ตำบลปากน้ำโพ อำเภอเมือง
 จังหวัดนครสวรรค์ 60000

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนนครสวรรค์
- ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร

E-mail first0_0@hotmail.com

ประวัติผู้เขียนโครงการงาน (ต่อ)



ชื่อ นางสาวสัททักใจ ชรรรรักษา
 ภูมิลำเนา 48 หมู่ 14 ตำบลค่านซ้าย อำเภอด่านซ้าย
 จังหวัดเลย 42120

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนสตรีราชินูทิศ
- ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร

E-mail mookrata@hotmail.com

