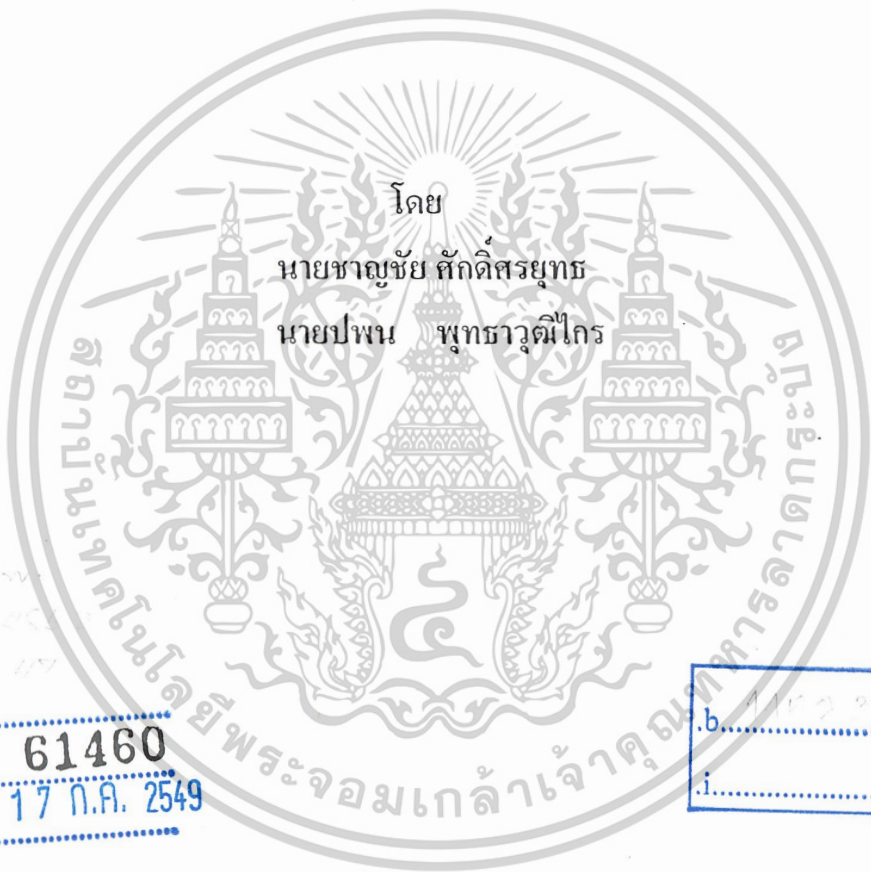


สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบจัดการสารสนเทศร้านอาหารด้วยพีดีเอ

RESTAURANT INFORMATION SYSTEM WITH PDA



โดย  
นายชาญชัย ตักดีสรยุทธ  
นายปพน พุทธาวุฒิไกร

เลขหมู่.....  
เลขทะเบียน..... 61460  
วัน,เดือน,ปี..... 17 ก.ค. 2549

b.....  
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**RESTAURANT INFORMATION SYSTEM WITH PDA**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENT FOR THE DEGREE OF  
BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING  
FACULTY OF ENGINEERING  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

**2004**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบจัดการสารสนเทศร้านอาหารด้วยพีดีเอ	
ชื่อนักศึกษา	นายชาญชัย ศักดิ์ศรยุทธ	รหัสประจำตัว 44010110
	นายปพน พุทธาวุฒิไกร	รหัสประจำตัว 44010279
อาจารย์ที่ปรึกษา	อ.ภูซงค์ หงษ์สุวรรณ	
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต	
	สาขาวิศวกรรมสารสนเทศ	
ภาควิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2547	

ปริญญานิพนธ์ฉบับนี้ได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเป็นที่เรียบร้อยแล้ว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญาบัตร	ระบบจัดการสารสนเทศร้านอาหารด้วยพีดีเอ	
ชื่อนักศึกษา	นายชาญชัย ศักดิ์ศรยุทธ	รหัสประจำตัว 44010110
	นายปพน พุทธาวุฒิไกร	รหัสประจำตัว 44010279
อาจารย์ที่ปรึกษา	อ.อุซงค์ หงษ์สุวรรณ	
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต	
	สาขาวิศวกรรมสารสนเทศ	
ภาควิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2547	

### บทคัดย่อ

ในปัจจุบันระบบการสื่อสารเครือข่ายไร้สายมีความต้องการมากขึ้นกว่าในอดีต อีกทั้งได้เข้ามามีบทบาทในชีวิตประจำวันมากขึ้น จึงทำให้เกิดแนวคิดในการพัฒนาโครงการระบบจัดการสารสนเทศร้านอาหารด้วยพีดีเอ นอกจากนี้ระบบนี้ได้มีการนำอุปกรณ์ไร้สายมาร่วมใช้ในการพัฒนาซึ่งทำให้เกิดความสะดวกและรวดเร็วในการใช้งาน

สำหรับระบบที่พัฒนาขึ้นได้มีการนำเทคโนโลยีคอทเน็ตมาใช้ในการพัฒนาซึ่งเป็นเทคโนโลยีใหม่ที่มีความสามารถรองรับการพัฒนาแอปพลิเคชันเชิงคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<b>Thesis Title</b>	Restaurant Information System With PDA
<b>Student</b>	MR. CHANCHAI SAKSORNYUTH ID. 44010110 MR. PAPON PUTTHAWUTIKRAI ID. 44010279
<b>Advisor</b>	Mr.Puchong Hongsuwan
<b>Graduate Level</b>	Bachelor Degree of Information Engineering
<b>Department</b>	Information Engineering
<b>Academic Year</b>	2004

## ABSTRACT

Nowadays, Wireless Network Communication is required much more than in the past and it's also participate in our modern life style. Therefore ,Restaurant Information Service System with PDA is established. In addition wireless device is used in this project for more convinces and can access in any time.

For this system is developed by Dot Net Technology that new technology able to support component application development.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ปริญญาบัตรฉบับนี้คงไม่อาจสำเร็จไปได้เลย หากไม่ได้รับความช่วยเหลือและความร่วมมือจากหลายๆ ฝ่ายด้วยกัน เริ่มจากอาจารย์ภูษงค์ หงษ์สุวรรณ อาจารย์ที่ปรึกษาที่คอยให้คำปรึกษา แนะนำ ตรวจสอบแก้ไข และเอาใจใส่ตลอดระยะเวลาทั้งหมด รวมถึงเพื่อนๆ และพี่ๆ ที่ให้คำแนะนำและคอยเป็นกำลังใจเสมอมา

สุดท้ายนี้ทางคณะผู้จัดทำ ขอขอบพระคุณบิดา มารดาที่สนับสนุนและคอยช่วยเหลือคณะผู้จัดทำตลอดเรื่อยมา จนทำให้ปริญญาบัตรฉบับนี้สำเร็จลงได้ด้วยดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

เรื่อง	หน้า
บทคัดย่อไทย	ก
บทคัดย่ออังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	ช
สารบัญตาราง	ญ
<b>บทที่ 1 บทนำ</b>	1
1.1 แนวคิดและที่มาของปัญหา	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	2
1.4 ขั้นตอนการทำงาน	2
1.5 ผลที่คาดว่าจะได้รับ	2
1.6 ขั้นตอนการดำเนินงาน	3
1.7 อุปกรณ์ที่ใช้ในการพัฒนาโครงการ	4
1.7.1 ฮาร์ดแวร์	4
1.7.2 ซอฟต์แวร์	4
<b>บทที่ 2 ทฤษฎีและหลักการที่ใช้ในโครงการ</b>	5
2.1 การพัฒนาระบบเชิงวัตถุ	5
2.1.1 หลักการของการพัฒนาระบบเชิงวัตถุ	6
2.1.2 ขั้นตอนของกระบวนการพัฒนาซอฟต์แวร์เชิงวัตถุ	8
2.2 ภาษายูเอ็มแอล	12
2.2.1 ความเป็นมาของภาษายูเอ็มแอล	12
2.2.2 ข้อดีของยูเอ็มแอล	13
2.2.3 องค์ประกอบของยูเอ็มแอล	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
2.3 เทคโนโลยีคือทเน็ตเฟรมเวิร์ค	17
2.3.1 สถาปัตยกรรมคือทเน็ต	17
2.3.2 เลเยอร์ คอมมอนแลงเกจจรันไทม์	17
2.3.3 เลเยอร์เบสคาสไลบรารีแลงเกจ	22
2.3.4 เลเยอร์คอมมอนแลงเกจสเปคซิฟิเคชัน	24
2.4 สถาปัตยกรรมของคือทเน็ตคอมแพคเฟรมเวิร์ค	25
2.5 เทคโนโลยีเอดีโอคือทเน็ต	27
2.6 เอสคิวแอล เซิร์ฟเวอร์ 2000	30
2.6.1 ส่วนของผู้ให้บริการ	30
2.6.2 ส่วนของผู้ใช้บริการ	30
2.6.3 ข่ายการสื่อสารหรือเน็ตเวิร์ค)	30
2.6.4 สถาปัตยกรรมของเอสคิวแอล เซิร์ฟเวอร์	31
<b>บทที่ 3 การออกแบบ</b>	33
3.1 ข้อมูลเบื้องต้น	33
3.1.1 วิเคราะห์ข้อมูลของร้านอาหาร	33
3.1.2 การรับรายการอาหารและเครื่องดื่ม	33
3.1.3 การชำระเงิน	34
3.1.4 โครงสร้างโดยทั่วไปของระบบ	35
3.2 การออกแบบระบบ	36
3.2.1 ไฮเลเวลยูสเคสไดอะแกรมของระบบ	36
3.2.2 ยูสเคสไดอะแกรม	37
3.2.3 แอคทิวิตี้ไดอะแกรม	45
3.2.4 การออกแบบฐานข้อมูล	59
3.2.5 คาด้าดิกชันนารี	60
<b>บทที่ 4 ผลการทดลอง</b>	64
4.1 ส่วนของพนักงานคิดเงิน	64
4.2 ส่วนของบริกร	81
4.3 ส่วนของห้องครัว	89

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
4.4 ส่วนของบาร์เครื่องคั้ม	90
<b>บทที่ 5</b> รูปผลการทดลอง	91
5.1 รูปผลการทดลอง	91
5.2 ปัญหาที่พบและข้อจำกัดในการพัฒนาโครงการ	91
5.3 แนวทางในการพัฒนาต่อ	91
บรรณานุกรม	92



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

ภาพ	หน้า
รูปที่ 1.1 ขั้นตอนการทำงานของโครงการ	2
รูปที่ 1.2 ขั้นตอนการดำเนินงานในการทำโครงการ	3
รูปที่ 2.1 คลาสไคอะแกรมของระบบ Washing machine	14
รูปที่ 2.2 ออบเจกต์ไคอะแกรมของระบบ Washing machine	14
รูปที่ 2.3 ยูสเคสไคอะแกรม	15
รูปที่ 2.4 แสดงตัวอย่างของแอ็กทิวิตีไคอะแกรม	16
รูปที่ 2.5 รูปแสดงโครงสร้างการพัฒนาแอปพลิเคชันคือทเน็ต	17
รูปที่ 2.6 โครงสร้างแอปพลิเคชันแรกๆ และการติดต่อกันระหว่างแอปพลิเคชันเป็นเรื่องยาก	18
รูปที่ 2.7 การพัฒนาแอปพลิเคชันบนเทคโนโลยีคือทเน็ตคลาสภายในแต่ละแอปพลิเคชัน	18
รูปที่ 2.8 โครงสร้างการคอมไพล์จากวิซวลสตูดิโอคือทเน็ตและเมื่อนำไปใช้งานจริง	19
รูปที่ 2.9 แสดงสถาปัตยกรรมของคอมมอนแลงแวกจรันไทม์	20
รูปที่ 2.10 แสดงความสามารถในการติดต่อข้ามแอปพลิเคชัน เมื่อเขียนด้วยภาษาที่ต่างกัน	21
รูปที่ 2.11 เบสคลาสไลบรารีของสถาปัตยกรรมคือทเน็ต	23
รูปที่ 2.12 คอมมอนแลงแวกจสเป็คทรีไคออนของสถาปัตยกรรมคือทเน็ต	24
รูปที่ 2.13 แสดงสถาปัตยกรรมของคือทเน็ตคอมแพคเฟรมเวิร์ค	25
รูปที่ 2.14 แสดงคลาสไลบรารีชื่อ คือทเน็ตคอมแพคเฟรมเวิร์ค	26
รูปที่ 2.15 สถาปัตยกรรมการเชื่อมต่อข้อมูลผ่านเอดีโอ	27
รูปที่ 2.16 แสดงเส้นทางในการติดต่อแลกเปลี่ยนข้อมูลและคำสั่งระหว่างเครื่องเซิร์ฟเวอร์และเครื่องไคลเอนต์	31
รูปที่ 2.17 แสดงเส้นทางในการติดต่อแลกเปลี่ยนข้อมูลและคำสั่ง ระหว่างเครื่องค้ำเบสเซิร์ฟเวอร์และเครื่องไคลเอนต์	31
รูปที่ 3.1 โดเมนโมเดล	35
รูปที่ 3.2 แสดงไฮเลเวลยูสเคส	36
รูปที่ 3.3 ยูสเคสไคอะแกรมสำหรับการจัดการเกี่ยวกับรายการที่สั่งสินค้า	37
รูปที่ 3.4 ยูสเคสไคอะแกรมการจัดการเกี่ยวกับการสร้างรายการที่สั่งสินค้า	37
รูปที่ 3.5 ยูสเคสไคอะแกรมเกี่ยวกับการยกเลิกรายการสั่งสินค้า	38
รูปที่ 3.6 ยูสเคสไคอะแกรมสำหรับการเรียกเก็บเงิน	38
รูปที่ 3.7 ยูสเคสไคอะแกรมสำหรับการจ่ายเงิน	39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

ภาพ	หน้า
รูปที่ 3.8 ยูสเคสไดอะแกรมสำหรับการตรวจสอบสต็อกสินค้า	39
รูปที่ 3.9 ยูสเคสไดอะแกรมเกี่ยวกับการแจ้งการทำอาหารเสร็จ	40
รูปที่ 3.10 ยูสเคสไดอะแกรมการออกใบรายการจ่ายเงินค่าอาหาร	40
รูปที่ 3.11 ยูสเคสไดอะแกรมการปรับปรุงข้อมูลสต็อกสินค้า	41
รูปที่ 3.12 ยูสเคสไดอะแกรมการออกใบเรียกเก็บเงิน	41
รูปที่ 3.13 ยูสเคสไดอะแกรมสำหรับการสร้างรายงานการขาย	42
รูปที่ 3.14 ยูสเคสไดอะแกรมเกี่ยวกับการสร้างใบเสร็จรับเงิน	42
รูปที่ 3.15 ยูสเคสไดอะแกรมเกี่ยวกับการสร้างสรุปรายงานการขาย	43
รูปที่ 3.16 ยูสเคสไดอะแกรมสำหรับการตรวจสอบสถานะของโต๊ะ	43
รูปที่ 3.17 ยูสเคสไดอะแกรมเกี่ยวกับการส่งรายการอาหารจากบริกร ไปห้องครัว	44
รูปที่ 3.18 ยูสเคสไดอะแกรมเกี่ยวกับการส่งรายการอาหารจากบริกร ไปยังพนักงาน	44
รูปที่ 3.19 แอคทิวิตี้ไดอะแกรมสำหรับการสั่งสินค้า	45
รูปที่ 3.20 แอคทิวิตี้ไดอะแกรมสำหรับการเรียกเก็บเงิน	46
รูปที่ 3.21 แอคทิวิตี้ไดอะแกรมสำหรับการจ่ายเงิน	47
รูปที่ 3.22 แอคทิวิตี้ไดอะแกรมสำหรับการแก้ไขใบเสร็จเรียกเก็บเงิน	48
รูปที่ 3.23 แอคทิวิตี้ไดอะแกรมสำหรับการตรวจสอบโต๊ะว่าง	49
รูปที่ 3.24 แอคทิวิตี้ไดอะแกรมสำหรับการยกเลิกรายการอาหาร	50
รูปที่ 3.25 แอคทิวิตี้ไดอะแกรมสำหรับการล็อกอินของพนักงานงานคิดเงิน	51
รูปที่ 3.26 แอคทิวิตี้ไดอะแกรมสำหรับการล็อกอินของบริกร	52
รูปที่ 3.27 แอคทิวิตี้ไดอะแกรมสำหรับการตรวจสอบวัตถุดิบ	53
รูปที่ 3.28 แอคทิวิตี้ไดอะแกรมสำหรับการดูข้อมูลทางสถิติของร้าน	54
รูปที่ 3.29 แอคทิวิตี้ไดอะแกรมสำหรับการจองโต๊ะ	55
รูปที่ 3.30 แอคทิวิตี้ไดอะแกรมสำหรับการสร้างใบรายการสั่งซื้อ	56
รูปที่ 3.31 แอคทิวิตี้ไดอะแกรมสำหรับการแก้ไขรายการสั่งอาหาร	57
รูปที่ 3.32 แอคทิวิตี้ไดอะแกรมสำหรับการสั่งซื้อความ	58
รูปที่ 3.33 อี-อาร์ โมเดล (E-R Model) ของระบบฐานข้อมูล	59
รูปที่ 4.1 หน้าจอสำหรับการล็อกอินเข้าใช้ระบบ	64
รูปที่ 4.2 หน้าจอหลักของพนักงานคิดเงิน	65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

ภาพ	หน้า
รูปที่ 4.3 หน้าจอหลักสำหรับการส่งข้อความ	66
รูปที่ 4.4 หน้าจอแสดงรายการอาหาร	67
รูปที่ 4.5 หน้าจอการชำระเงิน	68
รูปที่ 4.6 หน้าจอสำหรับพิมพ์ใบรายการชำระเงิน	69
รูปที่ 4.7 หน้าจอแสดงการรวมโต๊ะ	70
รูปที่ 4.8 หน้าจอสำหรับการจองโต๊ะอาหาร	70
รูปที่ 4.9 หน้าจอบันทึกรายการอาหารและเครื่องดื่ม	71
รูปที่ 4.10 หน้าจอบันทึกรายละเอียดของโต๊ะอาหาร	72
รูปที่ 4.11 หน้าจอแสดงรายการพนักงานภายในร้าน	73
รูปที่ 4.12 หน้าจอสำหรับสั่งซื้อวัตถุดิบ	74
รูปที่ 4.13 หน้าจอแสดงรายการวัตถุดิบ	75
รูปที่ 4.14 หน้าจอสำหรับพิมพ์ใบสั่งซื้อวัตถุดิบ	76
รูปที่ 4.15 หน้าจอแสดงการรับวัตถุดิบ	77
รูปที่ 4.16 หน้าจอสำหรับพิมพ์รายการวัตถุดิบทั้งหมด	78
รูปที่ 4.17 หน้าจอแสดงรายงานสถิติของอาหารยอดนิยม	79
รูปที่ 4.18 หน้าจอแสดงรายงานสถิติเครื่องดื่มยอดนิยม	80
รูปที่ 4.19 หน้าจอสำหรับการล็อกอินเข้าใช้ระบบ	81
รูปที่ 4.20 หน้าจอหลักสำหรับบริการ	82
รูปที่ 4.21 หน้าจอสำหรับบริการทำการสั่งอาหาร	83
รูปที่ 4.22 หน้าจอสำหรับตรวจสอบรายการอาหาร	84
รูปที่ 4.23 หน้าจอแสดงราคาอาหาร	85
รูปที่ 4.24 หน้าจอแสดงรายละเอียดโต๊ะอาหาร	86
รูปที่ 4.25 หน้าจอแสดงรูปภาพและรายละเอียดอาหาร	87
รูปที่ 4.26 หน้าจอสำหรับการติดต่อกับพนักงานคิดเงิน	88
รูปที่ 4.27 หน้าจอหลักสำหรับห้องครัว	89
รูปที่ 4.28 หน้าจอหลักสำหรับบาร์เครื่องดื่ม	90

## สารบัญตาราง

ตาราง	หน้า
ตารางที่ 2.1 เปรียบเทียบการเชื่อมต่อฐานข้อมูลระหว่างเทคโนโลยี ADO และ ADO.NET	29
ตารางที่ 3.1 ตารางข้อมูลพนักงาน	60
ตารางที่ 3.2 ตารางข้อมูลสินค้า	60
ตารางที่ 3.3 ตารางข้อมูลโต๊ะ	60
ตารางที่ 3.4 ตารางข้อมูลวัตถุดิบ	61
ตารางที่ 3.5 ตารางข้อมูลรายการที่สั่งซื้อวัตถุดิบ	61
ตารางที่ 3.6 ตารางข้อมูลใบสั่งซื้อ	61
ตารางที่ 3.7 ตารางข้อมูลใบสั่งอาหาร	61
ตารางที่ 3.8 ตารางข้อมูลรายการที่ใช้วัตถุดิบ	62
ตารางที่ 3.9 ตารางข้อมูลรายการที่ส่งสินค้า	62
ตารางที่ 3.10 ตารางข้อมูลการจอง	62
ตารางที่ 3.11 ตารางข้อมูลใบเสร็จเก็บเงิน	63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 แนวคิดและที่มาของปัญหา

ในยุคปัจจุบันความเจริญก้าวหน้าทางด้านการสื่อสารได้พัฒนาขึ้นอย่างรวดเร็ว ดังจะเห็นได้ว่า การติดต่อสื่อสารแบบไร้สายได้ก้าวเข้ามามีบทบาทและมีอิทธิพลต่อชีวิตประจำวันมากขึ้น โดยเฉพาะในทางธุรกิจ โดยมีการนำเทคโนโลยีการสื่อสารแบบไร้สายมาเป็นสื่อกลางในการติดต่อและให้บริการแก่ลูกค้า เพื่อให้การดำเนินธุรกิจมีความสะดวกและรวดเร็ว รวมถึงเป็นการลดค่าใช้จ่ายในระยะยาวของการดำเนินธุรกิจนั้น ซึ่งเราจะเห็นได้ว่า ราคาของอุปกรณ์ไร้สายมีราคาที่ถูกลง จึงเป็นการดีที่จะนำเทคโนโลยีไร้สายมาประยุกต์ใช้กับการดำเนินงานต่างๆ

โครงการนี้ผู้จัดทำได้เล็งเห็นถึงแนวทางและประโยชน์อย่างมากในการพัฒนาการใช้ประโยชน์จากการผสมผสานของความสามารถในระบบซอฟต์แวร์ (Software) และอุปกรณ์ไร้สาย (Wireless Device) เพื่อเป็นการพัฒนาและสร้างสรรค์เพื่อก่อให้เกิดประโยชน์สูงสุด

### 1.2 วัตถุประสงค์ของโครงการ

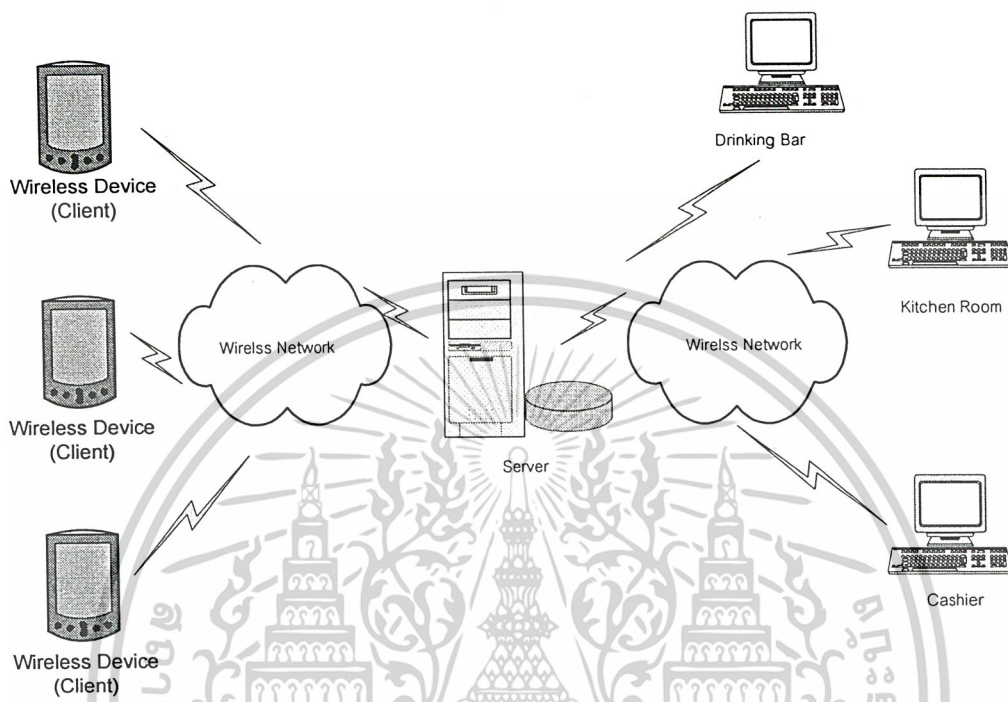
- 1.2.1 เพื่อพัฒนาระบบจัดการสารสนเทศร้านอาหาร ให้มีมาตรฐาน
- 1.2.2 เพื่ออำนวยความสะดวกและเพิ่มความรวดเร็วในการให้บริการ
- 1.2.3 เพื่อเป็นการศึกษาระบบงานธุรกิจร้านอาหาร
- 1.2.4 เพื่อพัฒนาและนำเทคโนโลยีไร้สาย มาใช้งานกับระบบต่าง ๆ ให้มากขึ้น

### 1.3 ขอบเขตของโครงการ

- 1.3.1 สามารถที่จะทำการสั่งอาหารผ่านอุปกรณ์ไร้สายได้
- 1.3.2 สามารถรองรับการทำงานแบบการใช้ข้อมูลพร้อมกัน (Multi-User) ได้
- 1.3.3 สามารถทำการพิมพ์รายการชำระเงินได้
- 1.3.4 ระบบฐานข้อมูลถูกออกแบบในลักษณะแบบจำลองข้อมูลเชิงสัมพันธ์
- 1.3.5 สามารถดูโต๊ะที่นั่งว่างภายในร้านอาหารได้
- 1.3.6 สามารถดูส่วนประกอบของอาหารและภาพอาหารได้
- 1.3.7 สามารถส่งคำสั่งรายการอาหารไปยังห้องครัวและบาร์เครื่องดื่มได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.4 ขั้นตอนการทำงาน



รูปที่ 1.1 ขั้นตอนการทำงานของโครงการ

## 1.5 ผลที่คาดว่าจะได้รับ

- 1.5.1 สามารถนำระบบที่พัฒนาขึ้นมาประยุกต์ใช้กับร้านอาหารได้
- 1.5.2 สามารถเพิ่มประสิทธิภาพในการให้บริการได้
- 1.5.3 สามารถนำธุรกิจมาประยุกต์ใช้ร่วมกับเทคโนโลยีใหม่ๆ ในปัจจุบันได้
- 1.5.4 มีการพัฒนาและนำเทคโนโลยีไร้สายมาประยุกต์ใช้มากขึ้นในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.6 ขั้นตอนการดำเนินงาน

ID	Task Name	2004						2005		
		AOA	LS	ES	LA	MS	FA	FS	AOA	LS
1	Problem Definition & Get Requirement	[Redacted]								
2	Analysis & Design	[Redacted]								
3	Software Design	[Redacted]								
4	Database Design	[Redacted]								
5	User Interface Design	[Redacted]								
6	Implementation	[Redacted]								
7	พัฒนาโมดูลในส่วนของการสั่งอาหาร	[Redacted]								
8	พัฒนาโมดูลในส่วน Database	[Redacted]								
9	พัฒนาโมดูลในส่วนของที่นั่งภายในร้าน	[Redacted]								
10	พัฒนาโมดูลในส่วนของการชำระเงิน	[Redacted]								
11	Test & Debug	[Redacted]								
12	Documentation	[Redacted]								

รูปที่ 1.2 ขั้นตอนการดำเนินงานในการทำโครงการ

## 1.7 อุปกรณ์ที่ใช้ในการพัฒนาโครงการงาน

### 1.7.1 ฮาร์ดแวร์

- เครื่องคอมพิวเตอร์สำหรับพัฒนาโปรแกรม 2 เครื่อง
- เครื่องคอมพิวเตอร์สำหรับเป็นเซิร์ฟเวอร์
- Pocket PC
- Wireless Card
- Wireless access point

### 1.7.2 ซอฟต์แวร์

- Microsoft Visual Studio .NET 2003
- Microsoft SQL Server 2000
- Microsoft Pocket PC 2003 SDK



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีและหลักการที่ใช้ในโครงการ

#### 2.1 การพัฒนาระบบเชิงวัตถุ (Object Oriented)

การพัฒนาระบบซอฟต์แวร์หรือระบบสารสนเทศในอดีตนั้นจะใช้วิธีการพัฒนาแบบโครงสร้าง (Structural) ซึ่งมีข้อดีเหนือกว่าการพัฒนาแบบโดยไม่ใช้โมเดล (Model) แต่ในปัจจุบันพบว่า การพัฒนาแบบโครงสร้างนั้นมีปัญหาคือ โมเดลในการวิเคราะห์และออกแบบระบบจะไม่เชื่อมต่อไปถึงรายละเอียดในการพัฒนาระบบ ตัวอย่างเช่น เมื่อผู้พัฒนาระบบวิเคราะห์และออกแบบระบบได้เป็นแผนภาพคอนเท็กซ์ (Context Diagram) หรือแผนภาพ DFD ระดับสูงสุด, แผนภาพกระแสข้อมูล (Data Flow) หรือแม้แต่แผนภาพที่แสดงความสัมพันธ์ขององค์ประกอบต่างๆ เพื่อสร้างฐานข้อมูลเชิงสัมพันธ์ (Entity Relationship) เองแผนภาพเหล่านี้เพียงบอกแต่ว่าระบบจะมีลักษณะการทำงานอย่างไร แต่ไม่ได้บอกวิธีในการพัฒนาระบบหรือรูปแบบในการเขียนโปรแกรม ทำให้ระบบที่พัฒนาขึ้นมีปัญหา เนื่องจากว่าการตรวจสอบความถูกต้องของระบบ (Validation) ในเชิงพฤติกรรม (Behavior) นั้นทำยากและไม่มีโมเดลใดๆ ที่รองรับ จึงได้มีผู้คิดค้นและพัฒนาวิธีการหรือโมเดลในการพัฒนาระบบเชิงวัตถุ ขึ้นมา

การพัฒนาระบบเชิงวัตถุ เป็นแนวความคิดในการออกแบบระบบต่างๆ โดยมองระบบต่างๆ ว่าเกิดจากการประกอบกันของวัตถุ (Object) ซึ่งหมายถึงสิ่งต่างๆ ที่อยู่รอบๆ ตัวเรา ออบเจกต์หนึ่งอาจจะประกอบด้วยออบเจกต์อื่นๆ หลายๆ ออบเจกต์ ขนาดที่เล็กที่สุดหรือใหญ่ที่สุดของออบเจกต์จะขึ้นอยู่กับขอบเขตของระบบที่เรากำลังสนใจอยู่ ว่าต้องการรายละเอียดในระดับใด

ในการกำหนดออบเจกต์จะต้องกำหนดคลาสขึ้นมาก่อน คลาสคือรูปแบบทั่วไปของออบเจกต์หรือเป็นพิมพ์เขียวของออบเจกต์ และออบเจกต์คือ อินสแตนซ์ (Instant) ของคลาส เช่น Mr. John เป็นอินสแตนซ์ ของคลาส Person

ออบเจกต์จะมีโครงสร้างที่ประกอบด้วยแอททริบิวต์ และพฤติกรรม เช่นในคลาส Person จะมีแอททริบิวต์ คือ height, weight และ age และมีพฤติกรรมคือ eat, sleep, read, write เป็นต้น ซึ่งแอททริบิวต์และพฤติกรรมดังกล่าวจะมีอะไรบางอย่าง ขึ้นอยู่กับระบบที่ต้องการออกแบบ เช่น Person ในระบบ Human Resource ก็จะมีรายละเอียดอื่นๆ ที่เกี่ยวข้องเพิ่มขึ้น

นอกจากนั้นการวิเคราะห์และออกแบบระบบเชิงวัตถุจะมีโมเดลที่รองรับ ซึ่งโมเดลที่เป็นมาตรฐานอยู่ในขณะนี้ชื่อว่า UML (Unified Modeling Language) โดยเป็นกระบวนการความคิด

(Methodology) และโมเดลที่สนับสนุนการวิเคราะห์และออกแบบเชิงวัตถุที่สามารถเข้าใจได้ง่าย และสามารถประกอบในขั้นตอนการลงมือเขียนโปรแกรมได้ด้วย

### 2.1.1 หลักการของการพัฒนาระบบเชิงวัตถุ

#### Abstraction

หมายถึงการกำหนดขอบเจ็ทต์ โดยไม่ได้ระบุรายละเอียดของแอททริบิวต์และพฤติกรรม หรือระบุเป็นส่วนน้อยเท่าที่สามารถทำได้ ซึ่งทำให้การออกแบบมีความยืดหยุ่นมากขึ้น และไม่เกิดการกำหนดขอบเจ็ทต์เดียวกันมากกว่า 1 ออบเจ็ทต์ ผู้ใช้ในแต่ละฟังก์ชันสามารถกำหนดแอททริบิวต์ และ พฤติกรรมเพิ่มเติมให้กับออบเจ็ทต์ได้ ตามความจำเป็นที่ต้องมีในฟังก์ชันงานนั้นๆ

#### Inheritance

การสืบทอดคุณสมบัติ คือ เป็นวิธีการในการสร้างคลาสใหม่จากคลาสเดิมที่มีอยู่ ทั้งนี้คลาสที่สร้างขึ้นใหม่จะมีวัตถุประสงค์ในการทำงานที่เฉพาะเจาะจงมากยิ่งขึ้น โดยการที่ออบเจ็ทต์มีคุณสมบัติทั้งหมดของคลาสที่ออบเจ็ทต์นั้นเป็นอินสแตนซ์อยู่ คลาสที่สืบทอดจะเรียกว่า ซับคลาส (subclass) ส่วนคลาสที่ถูกสืบทอดจะเรียกว่า ซุปเปอร์คลาส (superclass) ซึ่งซับคลาสจะมีคุณสมบัติเหมือนกับซุปเปอร์คลาสทุกประการ โดยไม่ต้องกำหนดซ้ำ

#### Polymorphism

คุณสมบัติการเปลี่ยนรูป (Polymorphism) หมายถึงการเปลี่ยนรูปร่างของออบเจ็ทต์หนึ่งๆ ซึ่งในเชิงโปรแกรมมิ่งจะเป็นการที่ตัวแปรออบเจ็ทต์ของคลาสใดคลาสหนึ่งสามารถเปลี่ยนรูปแบบไปจากคลาสเดิมได้ โดยคลาสที่มีการสืบทอดแต่โอเพอร์เรชันในซับคลาส มีการทำงานที่แตกต่างกับกับโอเพอร์เรชันเดียวกันในซุปเปอร์คลาส เช่น ถ้าคลาส Polygon มีโอเพอร์เรชัน draw และ คลาส Rectangle สืบทอดจากคลาส Polygon ซึ่งจะทำให้มีโอเพอร์เรชัน draw เช่นเดียวกัน แต่การ draw ของ Polygon กับของ Rectangle มีการทำงานคนละอย่างกัน ดังนั้นจึงต้องมีการกำหนดโอเพอร์เรชัน draw ของคลาส Rectangle ใหม่

#### Encapsulation

คุณสมบัติเอ็นแคปซูลชัน (Encapsulation) หรือเรียกอีกอย่างหนึ่งคือว่า การซ่อนข้อมูล (Information hiding) คือการจัดกลุ่มองค์ความคิดที่คล้ายคลึงกันเข้าเป็นหน่วยเดียวกันเพื่อทำการอ้างอิงด้วยชื่อเดียวกัน เป็นการซ่อนข้อมูลหรือกระบวนการในการทำโอเพอร์เรชันจากผู้ใช้คลาส ในการออกแบบเชิงออบเจ็ทต์สามารถทำได้ใน 3 ระดับ คือ

- private เป็นการกำหนดให้ข้อมูลหรือโอเปอเรชันนั้น เข้าถึงได้จากภายในคลาสเดียวกันเท่านั้น โดยสามารถทำได้ทั้งการอ่านและเขียน

- protected เป็นการกำหนดข้อมูลหรือโอเปอเรชันนั้นเข้าถึงได้จากภายในคลาสเดียวกันและคลาสอื่น โดยถ้าเป็นคลาสเดียวกัน จะสามารถทำได้ทั้งการอ่านและเขียน แต่ถ้าเป็นคลาสอื่นจะทำได้เพียงการอ่านอย่างเดียว

- public เป็นการกำหนดให้ข้อมูลหรือโอเปอเรชันนั้น เข้าถึงได้จากภายในคลาสเดียวกันและคลาสอื่น โดยสามารถทำได้ทั้งการอ่านและเขียน

### Message Sending

การส่งแมสเสจทำให้ออบเจกต์ในระบบสามารถทำงานร่วมกันได้ โดยออบเจกต์หนึ่งจะส่งแมสเสจให้กับอีกออบเจกต์หนึ่งเพื่อให้ออบเจกต์นั้นทำโอเปอเรชันอย่างใดอย่างหนึ่ง ออบเจกต์ที่ได้รับแมสเสจก็จะทำโอเปอเรชันตามแมสเสจที่ได้รับ แต่ในบางครั้งจะพบว่าการกำหนดให้ออบเจกต์ทำโอเปอเรชันอาจจะสามารถทำได้โดยการเรียกใช้อินเทอร์เฟซของออบเจกต์นั้นโดยตรง เช่น การเรียกใช้ฟังก์ชันของออบเจกต์ที่ประกาศเป็น Public ซึ่งก็ถือเป็น message sending อย่างไม่หนึ่ง

### Associations

ออบเจกต์ที่อยู่ในระบบเดียวกัน จะมีความสัมพันธ์กับออบเจกต์อื่นอย่างน้อย 1 ออบเจกต์เสมอในบางครั้งอาจมีความสัมพันธ์ระหว่างออบเจกต์ คือการกำหนดจำนวน (multiplicity) ซึ่งเป็นการกำหนดจำนวนออบเจกต์ในคลาสหนึ่ง ที่มีความสัมพันธ์กับออบเจกต์หนึ่งในอีกคลาสหนึ่ง

### Aggregation

เป็นรูปแบบความสัมพันธ์อีกแบบหนึ่ง โดยออบเจกต์หนึ่งประกอบขึ้นจากออบเจกต์อื่นๆ ตั้งแต่ 1 ออบเจกต์ขึ้นไป เช่น ในออบเจกต์ computer A มีออบเจกต์ Hard drive 2 ออบเจกต์เป็นต้น ออบเจกต์ Computer เรียกว่า Aggregate Object และออบเจกต์ที่เป็นองค์ประกอบจะเรียกว่า Component Object

อีกรูปแบบหนึ่งของ Aggregation ที่เป็นความสัมพันธ์แบบความสัมพันธ์แบบแข็ง (Strong relationship) คือ Composition โดย Component Object จะต้องมีใน Composite Object เช่น ออบเจกต์คอมพิวเตอร์ จะต้องมีส่วนซีพียู (CPU) จึงเป็น Composition แต่ออบเจกต์คอมพิวเตอร์อาจจะไม่มีเครื่องพิมพ์ก็ได้ ทำให้ความสัมพันธ์ระหว่าง คอมพิวเตอร์ กับ เครื่องพิมพ์ เป็นความสัมพันธ์แบบ Aggregation แต่ไม่เป็น Composition

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กล่าวโดยสรุป ข้อดีของการประยุกต์ใช้แนวคิดเชิงวัตถุในการพัฒนาระบบซอฟต์แวร์มีดังต่อไปนี้

- สนับสนุนการพัฒนาที่ซับซ้อน (Problem Decomposition)

ในการพัฒนาโปรแกรมรูปแบบเดิม นักพัฒนาโปรแกรมจะทำงานกันที่ระดับฟังก์ชัน กล่าวคือตั้งแต่เริ่มการพัฒนาจนจบ จะเป็นการสร้างปรับปรุงแก้ไขรายละเอียดของฟังก์ชันต่างๆ ภายในตัวโปรแกรม แต่สำหรับแนวคิดเชิงวัตถุ นักพัฒนาจะทำการวิเคราะห์ออกแบบระบบกันในระดับของออบเจกต์ซึ่งประกอบด้วยทั้งข้อมูลและฟังก์ชันภายในแต่ละออบเจกต์ ดังนั้นจึงกล่าวได้ว่าการอาศัยแนวคิดเชิงวัตถุจะช่วยให้ นักพัฒนาสามารถสร้างโปรแกรมได้ง่ายขึ้นมาก

- สนับสนุนการนำกลับมาใช้งานซ้ำอีกครั้ง (Promotion of Reusability)

เนื่องจากแต่ละคลาสหรือออบเจกต์ที่กำหนดขึ้นนั้นจะมีความสมบูรณ์อยู่ในตัวมันเองบนพื้นฐานแนวคิดของแต่ละออบเจกต์เอง รวมทั้งยังเป็นอิสระจากสภาพแวดล้อมอื่น ดังนั้นแต่ละคลาสจึงง่ายต่อการนำกลับมาใช้งานปรับปรุงเพิ่มเติม การนำกลับมาใช้งานอาจอยู่ในรูปแบบของการสืบทอดคุณสมบัติระหว่างออบเจกต์จากคลาสพ่อแม่ไปใส่ไว้ในโปรแกรมต่างๆ ได้ตามต้องการ นั่นคือคลาสพ่อแม่สามารถถูกนำกลับมาใช้งานได้

- ปรับปรุงแก้ไขและบำรุงรักษาง่าย (Easy for Change and Extensibility)

เนื่องจากข้อมูลและฟังก์ชันการทำงานที่เกี่ยวข้องกับออบเจกต์หนึ่งๆ จะถูกรวบรวมอยู่ที่เดียวกัน การทำงานภายในของแต่ละออบเจกต์จะไม่เกี่ยวข้องฟังก์ชันกับโค้ดที่อยู่ภายนอกออบเจกต์ ดังนั้น นักพัฒนาสามารถทำการแก้ไขปรับปรุงรายละเอียดภายในของแต่ละคลาสได้โดยไม่กระทบต่อส่วนที่เรียกใช้งานภายนอกแต่อย่างใด นอกจากนี้ในการขยายระบบก็สามารถทำได้โดยง่าย โดยการสร้างออบเจกต์หรือคลาสเพิ่มเติมลงไปในตัวโปรแกรมนั่นเอง

### 2.1.2 ขั้นตอนของกระบวนการพัฒนาซอฟต์แวร์เชิงวัตถุ

วัตถุประสงค์หลักของทุกกระบวนการพัฒนาซอฟต์แวร์คือ การแปลงความต้องการของผู้ใช้ให้เป็นระบบที่มีคุณภาพและสามารถใช้งานได้จริง อีกทั้งยังช่วยลดระยะเวลาที่ต้องทำการเขียนโปรแกรมและแก้ไขข้อผิดพลาดให้น้อยลงอีกด้วย ซึ่งในระหว่างการพัฒนาจะมีการใช้เครื่องมือช่วยต่างๆ หลายอย่างในที่นี้ได้แก่ภาษามาตรฐานยูเอ็มแอล

กระบวนการที่จะอธิบายต่อไปนี้เป็นวิธีการทั่วไปในการพัฒนาซอฟต์แวร์เชิงวัตถุที่ประกอบไปด้วย 5 ขั้นตอน โดยในแต่ละขั้นตอนจะถูกเรียกว่า เฟส (Phase) ซึ่งผลลัพธ์ที่ได้จากการพัฒนาในแต่ละเฟสจะต้องถูกนำไปใช้ในการพัฒนาต่อในเฟสถัดไป และมีเพียงเฟสแรกที่จะถูกดำเนินการในช่วงเริ่มต้นของการพัฒนาครั้งเดียวและในสี่เฟสที่เหลือจะถูกทำซ้ำ (Iteration) เพื่อทำการขยายต่อเติมฟังก์ชันของระบบจนกว่าจะได้ระบบที่สมบูรณ์ นั่นคือในแต่ละรอบของการวนซ้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สิ่งที่ได้จะเป็นเพียงต้นแบบ (Prototype) อันประกอบไปด้วยฟังก์ชันบางอย่างสอดคล้องกับความต้องการของผู้ใช้ และสิ่งนี้เองที่ฝ่ายพัฒนาควรทำการส่งมอบให้ผู้พิจารณาเป็นระยะๆ ในช่วงของการพัฒนา และในความเป็นจริงสิ่งที่เกิดขึ้นตามมาซึ่งมักหลีกเลี่ยงไม่ได้คือความจำเป็นในการดำเนินการเฟสแรกมากกว่าหนึ่งครั้งอันเนื่องมาจากการเปลี่ยนแปลงความต้องการของผู้ใช้นั่นเอง

### 2.1.2.1 การวิเคราะห์ความต้องการของระบบหรือผู้ใช้ (Requirement Analysis)

โครงการที่ถือว่าประสบความสำเร็จจะต้องตอบสนองต่อความต้องการและความคาดหวังของผู้ใช้ได้อย่างครบถ้วนหรือมากกว่าอันเป็นจุดสำคัญของทุกกระบวนการพัฒนาซอฟต์แวร์ การวิเคราะห์ความต้องการของผู้ใช้เป็นการค้นหาขอบข่ายของระบบและเป็นการเตรียมข้อมูลด้านความสามารถของระบบ (System Functions) จากมุมมองภายนอกที่จะต้องถูกทำการพัฒนาโดยไม่คำนึงถึงรายละเอียดหรือกรรมวิธีทางเทคนิคต่างๆ

ในความเป็นจริงการเริ่มต้นเฟสแรกนี้จะต้องทำในรูปของการกำหนดเป็นข้อตกลง (Agreement) ด้วยการเจรจกันระหว่างผู้ใช้งานกับผู้พัฒนา ซึ่งฝ่ายผู้พัฒนาจะต้องบันทึกความต้องการของผู้ใช้อย่างละเอียดที่สุดเท่าที่จะเป็นไปได้ เรียกขั้นตอนย่อยนี้ว่า User Requirement Elicitation อันหมายถึงการสัมภาษณ์ผู้ใช้งานระบบถึงความต้องการต่างๆ เฟสนี้จำเป็นอย่างมากโดยเฉพาะในการพัฒนาระบบด้านธุรกิจ แต่มักไม่จำเป็นสำหรับการสร้างระบบเชิงวิจัยพัฒนาซึ่งมีเป้าหมายที่ชัดเจนอยู่แล้ว สำหรับในส่วนที่ไม่สามารถลงรายละเอียดของความต้องการได้ก็ควรกำหนดเป้าหมายให้ชัดเจนในส่วนนั้นๆ เพื่อป้องกันความเข้าใจที่ไม่ตรงกันระหว่างผู้ใช้และผู้พัฒนา

นอกจากการค้นหาความสามารถที่ต้องมีอยู่ในตัวระบบแล้ว ผู้พัฒนายังจำเป็นต้องทำการสำรวจถึงความต้องการในแง่อื่นๆ ด้วย (Nonfunctional Requirements) ตัวอย่างเช่น ประสิทธิภาพช่วงเวลาในการตอบสนอง ส่วนอินเทอร์เน็ตเฟส การใช้งานร่วมกับระบบเดิม เป็นต้น อย่างไรก็ตาม หากมองในภาพที่กว้างขึ้น จะพบว่า การดำเนินการพัฒนาระบบจะต้องเกี่ยวข้องกับงบประมาณระยะเวลาในการส่งมอบผลงานแต่ละช่วง การทำข้อสัญญา และการบริหาร โครงการ เป็นต้น

### 2.1.2.2 การวิเคราะห์ระบบ (Domain Analysis)

การวิเคราะห์ระบบ (Domain Analysis) เป็นการวิเคราะห์โครงสร้าง (Structure) และพฤติกรรม (Behavior) ของระบบที่จะทำการพัฒนาซึ่งจะถูกนำไปกำหนดรายละเอียดเชิงเทคนิคในเฟสการออกแบบและจะถูกสร้างจริงเป็นลำดับต่อไปในอิมพลีเมนต์ระบบ กิจกรรมในเฟสนี้จะเป็นการทำความเข้าใจกับระบบกำลังพัฒนาในสองด้าน กล่าวคือ

- การวิเคราะห์โครงสร้าง เป็นการทำความเข้าใจกับระบบที่กำลังพัฒนาเพื่อค้นหาคลาสและความสัมพันธ์ระหว่างคลาสในระบบ ซึ่งผลที่ได้เหล่านี้จะถูกนำไปอิมพลีเมนต์สร้างเป็นระบบขึ้นจริงในเฟสถัดๆไป

- การวิเคราะห์พฤติกรรม เนื่องจากออบเจกต์ที่เกิดจากคลาสต่างๆในโครงสร้างของระบบจะทำงานร่วมกันเพื่อให้ระบบโดยรวมสามารถตอบสนองต่อการใช้งานของผู้ใช้ได้ในทุกฟังก์ชัน โดยกิจกรรมในระหว่างการปฏิบัติงานร่วมกันของแต่ละออบเจกต์จำเป็นต้องถูกทำความเข้าใจเพื่อนำไปอิมพลีเมนต์เป็นขั้นตอนการทำงานของแต่ละฟังก์ชันของระบบได้

### 2.1.2.3 การออกแบบ (Design)

ซึ่งเป็นกระบวนการคิดค้นวิธีแก้ไขปัญหาหรือพิจารณารายละเอียดเชิงเทคนิค เพื่อเตรียมที่จะอิมพลีเมนต์ระบบขึ้นจริง ซึ่งจะเป็นการนำผลการวิเคราะห์จากเฟสที่สองมาทำการแก้ไขเพิ่มเติมรายละเอียดเชิงเทคนิค เพื่อสามารถที่จะถูกนำไปสร้างเป็นระบบซอฟต์แวร์จริงได้อย่างสมบูรณ์ ตัวอย่างของกิจกรรมที่เกิดขึ้นในเฟสนี้คือ

- การเลือกใช้ภาษาโปรแกรมคอมพิวเตอร์ที่จะใช้ในการพัฒนาระบบ
- การออกแบบฐานข้อมูล โดยเฉพาะในส่วนที่เรียกว่า ระเบียบแนวคิด ซึ่งนิยมใช้ อีอาร์โมเดลเข้ามาช่วย
- การออกแบบสถาปัตยกรรมระบบ จำนวนเครื่องคอมพิวเตอร์
- การออกแบบลักษณะการเชื่อมต่อของเครือข่าย
- การนำกลับมาใช้งานของคลาสไลบรารีหรือซอฟต์แวร์คอมโพเนนต์ที่มีอยู่
- การออกแบบหน้าต่างของตัวโปรแกรม (User Interface)
- รวมถึงการจัดการกับข้อผิดพลาดที่อาจเกิดขึ้นในการใช้งานระบบ เช่น ระบบควรมีการจัดการอย่างไรหากผู้ใช้ป้อนข้อมูลผิดประเภท เป็นต้น

ดังนั้นเฟสนี้จำเป็นต้องอาศัยความรู้ทางเทคนิคและเทคโนโลยีต่างๆ ที่มีอยู่เพื่อสามารถเลือกได้อย่างเหมาะสม

### 2.1.2.4 การสร้างโปรแกรมระบบ (Construction, Coding, Implementation)

หลังจากที่ระบบถูกออกแบบไว้อย่างสมบูรณ์พร้อมที่จะถูกนำไปอิมพลีเมนต์จริงโดยโปรแกรมเมอร์ ในเฟสนี้สิ่งที่เกิดขึ้นจากการออกแบบทั้งหมดจะถูกแปลงไปสู่ระบบจริง โดยส่วนใหญ่กิจกรรมในเฟสนี้จะเป็นการสร้างโปรแกรมหรือการอิมพลีเมนต์ระบบอันเป็นขั้นตอนของ OOP (Object Oriented Programming) ที่คุ้นเคยกันดี ซึ่งจะต้องอาศัยความรู้ในตัวภาษาโปรแกรมที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้ในการสร้างโค้ด ในขั้นตอนนี้จะถูกดำเนินการ โดยทีมโปรแกรมเมอร์ภายหลังจากได้รับข้อมูลการออกแบบที่เพียงพอจากฝ่ายวิเคราะห์ออกแบบ

#### 2.1.2.5 การทดสอบระบบ (Testing)

เป็นการทดสอบความถูกต้องของระบบที่พัฒนาเพื่อค้นหาข้อผิดพลาดเชิงเทคนิค และการตรวจสอบความสอดคล้องกับความต้องการที่ถูกระบุอยู่ในความต้องการของผู้ใช้งาน ทั้งนี้การค้นพบข้อผิดพลาดจะถือว่าเป็นความสำเร็จของการดำเนินงานในเฟสนี้มีใช้ความล้มเหลวแต่อย่างใด นอกจากนี้ยังเป็นการประเมินความสมบูรณ์ของระบบว่าจำเป็นต้องทำการวิเคราะห์ออกแบบเพิ่มเติมอีกครั้งหรือไม่ โดยปกติในเฟสนี้จะต้องมีการจัดเตรียมข้อมูลที่จะใช้สำหรับการทดสอบและการประเมินผลลัพธ์เรียกว่า Test Cases จะประกอบไปด้วยกิจกรรมในการทดสอบการป้อนข้อมูลและการทดสอบผลลัพธ์รวมถึงความคาดหวังจากการทำงานของระบบว่าน่าจะเป็นที่พึงพอใจแก่ผู้ใช้งานจริงหรือไม่ ผลของการทดสอบจะถูกบันทึกลงรายงานการทดสอบซึ่งรวมถึงการบรรยายรายละเอียดข้อผิดพลาดที่ปรากฏเพื่อทำการแก้ไขต่อไปโดยโปรแกรมเมอร์

ในเฟสนี้สิ่งที่ทำให้นักพัฒนาปวดหัวอยู่เสมอคือข้อผิดพลาด (Error) ซึ่งแบ่งได้เป็นสองด้าน ได้แก่

-Functional เป็นความผิดพลาดในระดับฟังก์ชันการทำงานของระบบ ซึ่งมักเกิดขึ้นบ่อยครั้ง

-Nonfunctional เป็นความผิดพลาดที่ไม่เกี่ยวข้องกับระดับฟังก์ชันการทำงานของระบบที่พัฒนาโดยตรง เช่น การที่ประสิทธิภาพการทำงานของระบบต่ำเกินไป ช่วงเวลาในการตอบสนองนานเกินไป เป็นต้น

โดยปกติความต้องการของผู้ใช้งานมักไม่ถูกค้นพบทั้งหมดในช่วงเริ่มต้นของการสัมภาษณ์ผู้ใช้ จึงเป็นเหตุผลว่าทำไมผู้ใช้งานจึงมีร้องขอการเปลี่ยนแปลง ซึ่งเป็นปัญหาที่เกิดขึ้นกับนักพัฒนาทั้งหลายโดยตลอดมา ข้อเสนอแนะต่อไปนี้อาจช่วยป้องกันปัญหาดังกล่าวได้บ้างไม่มากนัก

-กำหนดรายละเอียดความต้องการให้ชัดเจนถึงสิ่งที่ผู้ต้องการและไม่ต้องการทั้งนี้ห้ามสมมติเอาเอง เพื่อใช้ในการป้องกันหรือชะลอความเสียหายจากผู้ที่มีการเปลี่ยนแปลงความต้องการบ่อยๆ

-สิ่งที่ผู้ใช้เกริ่นว่าจะต้องการแต่ไม่ใช่ตอนนี้อาจถูกออกแบบพัฒนาเพื่อไว้โดยปกติ 10% เพื่อลดผลกระทบที่อาจเกิดขึ้นเมื่อเวลาที่ต้องพัฒนาส่วนนั้นจริงๆมาถึง

-กำหนดข้อตกลงให้ชัดเจนในการส่งมอบงานหรือระบบตัวอย่างแต่ละช่วงเวลาในการพัฒนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-การร้องขอให้แก้ไขข้อผิดพลาดในระบบที่เพิ่งส่งมอบและตรวจรับเรียบร้อยแล้วอาจถูกคิดเป็นค่าบริการพิเศษ ทั้งนี้ต้องกำหนดไว้ในสัญญาอย่างชัดเจน

เราจำเป็นต้องมองการเปลี่ยนแปลงความต้องการของผู้ใช้งานเป็นเรื่องปกติธรรมดาที่ผู้พัฒนาจะต้องเผชิญ หากแต่สิ่งที่ควรจะทำคือการหาวิธีการรับที่จะทำให้ได้รับผลกระทบน้อยที่สุดเมื่อความต้องการการเปลี่ยนแปลงดังกล่าวเกิดขึ้น

## 2.2 ภาษายูเอ็มแอล

UML (Unified Modeling Language) หมายถึง ภาษาสัญลักษณ์รูปภาพมาตรฐาน (Standard Modeling Language) ที่ใช้กำหนดลักษณะคลาส การสร้างคลาสและเป็นเอกสารที่บอกถึงรายละเอียดของระบบ โครงสร้าง โดยที่ยูเอ็มแอลจะสามารถแสดงโครงสร้างของระบบเชิงวัตถุในรายละเอียดต่างๆ ได้ดี ในรูปแบบของแผนภาพไดอะแกรม แผนภาพเหล่านี้จะทำให้เกิดความเข้าใจที่ตรงกันระหว่างผู้ออกแบบระบบและโปรแกรมเมอร์ ทำให้การปรับปรุงแก้ไขโปรแกรมทำได้ง่ายขึ้น

### 2.2.1 ความเป็นมาของภาษายูเอ็มแอล

ยูเอ็มแอลถูกเริ่มครั้งแรกในปี 1994 ที่บริษัท Rational Software โดย Grady Booch และ James Rumbaugh วัตถุประสงค์เบื้องต้นในการร่วมงานกันระหว่างทั้งสองคนเป็นการพัฒนากระบวนการพัฒนาซอฟต์แวร์เชิงวัตถุที่เป็นหนึ่งเดียวกัน (Unified Method) โดยนำเอาวิธีของแต่ละคนคือวิธีของ Booch และวิธี OMT มารวมกันและปรับปรุงใหม่ ต่อมาในปี 1995 Ivar Jacobson ผู้พัฒนากระบวนการ OOSE หรือ Objectory ได้เข้าร่วมกับโครงการดังกล่าวซึ่งในครั้งนี้เป็น การสร้างภาษาโมเดลขึ้นใหม่เรียกว่า Unified Modeling Language (UML) และทั้งสามขนานนามตัวเองว่า Three Amigos

เนื่องจากวิธีการพัฒนาซอฟต์แวร์เชิงวัตถุของทั้งสามมีชื่อเสียงอยู่แล้วในช่วงเวลานั้น ดังนั้น ภาษายูเอ็มแอลที่ถูกพัฒนาขึ้นใหม่นี้จึงกลายมาเป็นที่นิยมใช้อย่างแพร่หลายเช่นกัน ในปี 1996 มีหลายบริษัทที่ขอเข้าร่วมในการพัฒนาภาษายูเอ็มแอล ซึ่งได้แก่ บริษัทดิจิตอลลิควิเมนต์ บริษัทเอชพี บริษัทไอบีเอ็ม รวมทั้งบริษัทออราเคิล บริษัทไมโครซอฟต์ และบริษัทอื่นๆอีกหลายบริษัท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2.2 ข้อดีของยูเอ็มแอล

ยูเอ็มแอลมีข้อดีหลายประการดังต่อไปนี้

- เป็นภาษารูปภาพมาตรฐานหรือภาษาสากลที่ใช้ในการพัฒนาซอฟต์แวร์เชิงวัตถุ และสามารถใช้ในการแลกเปลี่ยนโมเดลได้อย่างสื่อความหมาย รวมถึงการจัดสร้างเอกสารการวิเคราะห์ ออกแบบระบบ

- สามารถนำเสนอและสนับสนุนหลักการเชิงวัตถุได้อย่างครบถ้วนชัดเจน ทำให้นักพัฒนาระบบสามารถทำความเข้าใจกับปัญหาและค้นพบวิธีแก้ไขได้อย่างรวดเร็วและง่ายยิ่งขึ้น

- ไม่ผูกติดกับภาษาโปรแกรมใดภาษาหนึ่ง กล่าวคือโมเดลที่ถูกสร้างขึ้นจากภาษามาตรฐานซึ่งสามารถแปลงไปเป็นระบบจริงที่ถูกสร้างขึ้นด้วยภาษาโปรแกรมเชิงวัตถุใดๆก็ได้

- เป็นภาษาที่ง่ายต่อการทำความเข้าใจ ผู้ที่ทำการศึกษาหรือนำไปใช้งานไม่จำเป็นต้องมีความรู้อื่นใดนอกจากแนวคิดเชิงวัตถุ

- สามารถถูกแปลงเป็นภาษาที่ใช้ในการสร้างระบบขึ้นได้จริงได้อย่างอัตโนมัติ จึงเป็นการช่วยลดภาระ เวลา และค่าใช้จ่ายการพัฒนาระบบได้อย่างมาก

- ยูเอ็มแอลใช้ในการบันทึกความคิดของนักพัฒนาในลักษณะของเอกสารที่พร้อมจะถูกนำมาทำความเข้าใจหรือสานต่ออีกครั้งได้อย่างรวดเร็ว

## 2.2.3 องค์ประกอบของยูเอ็มแอล

ยูเอ็มแอลเป็นภาษาการสร้างโมเดลในรูปแบบกราฟิก ซึ่งใช้ในการประกอบกันเป็นไดอะแกรม โดยมีกฎในการประกอบกันของอีลิเมนต์ต่างๆ ไดอะแกรมจะแสดงถึงมุมมองต่างๆ (multiple view) ของระบบ ซึ่งจะเรียกว่าโมเดล โมเดลยูเอ็มแอลจะบ่งบอกถึงรายละเอียดของระบบ แต่จะไม่ระบุถึงวิธีการในการพัฒนาระบบ

### 1. คลาสไดอะแกรม (Class Diagram)

คลาสไดอะแกรมเป็นแผนภาพที่ใช้แสดงคลาส และความสัมพันธ์ในแง่ต่างๆระหว่างคลาสเหล่านั้น ซึ่งความสัมพันธ์ที่กล่าวถึงในคลาสไดอะแกรมนี้ ถือเป็นความสัมพันธ์เชิงสถิตย์ (Static Relationship) หมายถึงความสัมพันธ์ที่มีอยู่แล้วในคลาสต่างๆ ซึ่งเรียกว่าความสัมพันธ์เชิงกิจกรรม (Dynamic Relationship) สิ่งปรากฏในคลาสไดอะแกรมนั้น ประกอบไปด้วยกลุ่มของคลาส และกลุ่มของความสัมพันธ์ โดยใช้สัญลักษณ์ที่ใช้ในการแสดงคลาสนั้น แทนด้วยสี่เหลี่ยมที่แบ่งออกเป็น 3 ส่วนโดยส่วนบนสุดจะเป็นชื่อคลาส ส่วนกลางจะแสดงแอททริบิวต์ ของคลาส และส่วนล่างจะแสดงโอเปอเรชันของคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเช่น คลาส Washing Machine มีแอททริบิวต์คือ brand name, model, serial number และ capacity มีโอเปอเรชันคือ add clothes, add detergent, turn on และ remove clothes สามารถเขียนให้อยู่ในรูปแบบของยูเอ็มแอล คลาสไดอะแกรมได้ดังรูป

Washing Machine
- brand name
- model name
- serial number
- capacity
- add clothes()
- add detergent()
- remove clothes()
- turn on()

รูปที่ 2.1 คลาสไดอะแกรมของระบบ Washing Machine

## 2. ออบเจกต์ไดอะแกรม (Object Diagram)

ออบเจกต์ คือ อินสแตนซ์ของคลาส การกำหนดรายละเอียดของออบเจกต์ คือการกำหนดค่าให้กับแอททริบิวต์ ออบเจกต์ไดอะแกรมของยูเอ็มแอล จะแสดงเป็นรูปสี่เหลี่ยม โดยชื่อจะประกอบด้วยชื่อออบเจกต์และชื่อคลาส อันระหว่างชื่อด้วยเครื่องหมาย “:” และขีดเส้นใต้ที่ชื่อด้วย ตัวอย่างเช่น ออบเจกต์ Washing Machine ชื่อ My Washer มี brand name คือ Laundatorium, model คือ Washmeister, serial number คือ GL57774 และ capacity คือ 16 ปอนด์ สามารถแสดงได้ดังรูป

<u>My Washer : Washing Machine</u>
brand name = Laundatorium
model name = Washmeister
serial number = GL57774
capacity = 16 pounds

รูปที่ 2.2 ออบเจกต์ไดอะแกรมของระบบ washing machine

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. ยูสเคสไดอะแกรม (Use Case Diagram)

เป็นแผนภูมิที่ถูกรวบรวมมาเพื่ออธิบายถึงกิจกรรมของระบบจากมุมมองของผู้ใช้ซึ่งจะทำให้ได้ระบบที่ตรงกับความต้องการและสามารถนำไปใช้งานได้จริงยูสเคสไดอะแกรม

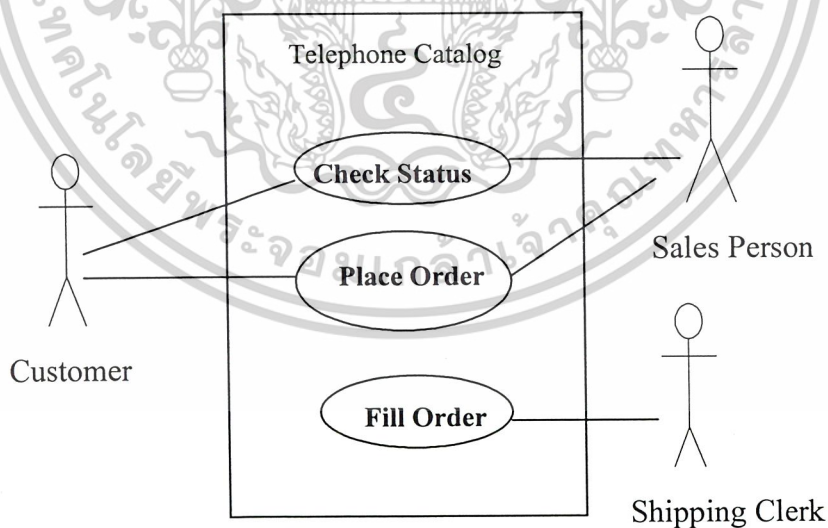
จะประกอบด้วย แอ็กเตอร์ (Actor), ยูสเคส (Use Case), เส้นแสดงความสัมพันธ์ (Relationship)

- แอ็กเตอร์ คือผู้ที่ทำกระทำกับระบบ (เป็นผู้ที่ใช้ยูสเคส) โดยแอ็กเตอร์ที่เริ่มต้นเหตุการณ์จะอยู่ด้านซ้ายของยูสเคส และแอ็กเตอร์ ที่ได้รับผลจากเหตุการณ์นั้นจะอยู่ทางด้านขวาของยูสเคส ชื่อของแอ็กเตอร์จะอยู่ด้านล่างของรูปที่ใช้แทนแอ็กเตอร์นั้น

- ยูสเคส เป็นกิจกรรมที่ทำโดย แอ็กเตอร์ซึ่งเป็นฟังก์ชันต่างๆ ที่ซอฟต์แวร์ต้องทำได้ทั้งหมดจึงจะถือว่าซอฟต์แวร์ไม่มีข้อผิดพลาดหรืออีกนัยหนึ่งยูสเคส คือสิ่งที่บอกว่าระบบหรือซอฟต์แวร์ทำอะไรได้บ้าง โดย ยูสเคสจะใช้เครื่องหมายวงรีในการแสดง และชื่อของยูสเคสจะอยู่ภายในวงรีหรืออยู่ใต้วงรี

Use Case Dependency คือความสัมพันธ์ระหว่าง ยูสเคส ซึ่งมี 2 ลักษณะคือ แบบ inclusion และแบบ extension แบบ include เป็นการมีการทำขึ้นตอนนี้ทุกครั้ง ส่วนแบบ extend จะเป็นการทำงานนี้เป็นบางครั้ง

- เส้นแสดงความสัมพันธ์คือเส้นที่บอกความสัมพันธ์ระหว่างกันทั้งของคลาส หรือ แอ็กเตอร์



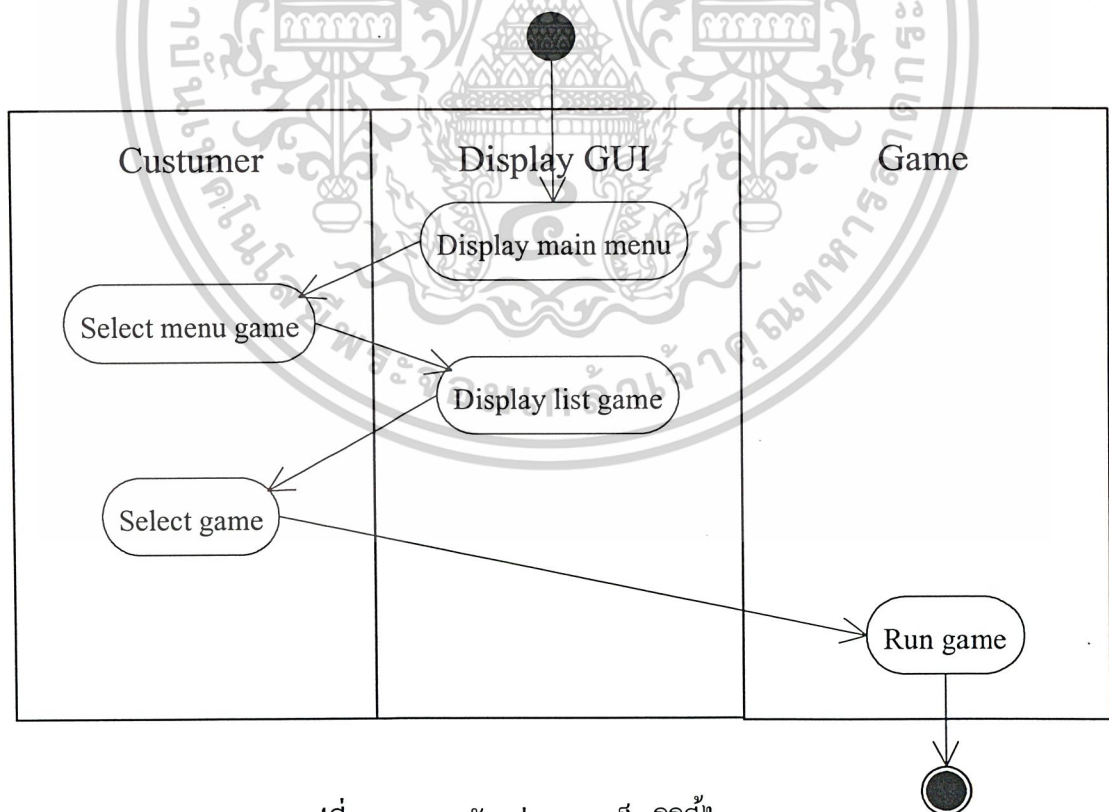
รูปที่ 2.3 ยูสเคสไดอะแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4. แอ็กทิวิตีไดอะแกรม (Activity Diagram)

แอ็กทิวิตีไดอะแกรมจะแสดงขั้นตอนการทำงานของยูสเคส โดยจะเน้นที่งานย่อยของวัตถุ ซึ่งการเจาะจงไปที่งานๆ หนึ่งของวัตถุนั้นจะรู้สึกเหมือนกับการแสดงสตอรี่ไดอะแกรมที่แสดงสถานะของวัตถุ แอ็กทิวิตีไดอะแกรมสามารถเปลี่ยนสถานะได้โดยไม่ต้องมีเหตุการณ์ที่กำหนดไว้ในไดอะแกรมมาก่อน แต่จะเปลี่ยนสถานะเองตามกระบวนการทำงานคล้ายกับโฟลชาร์ต สัญลักษณ์ที่ใช้ใน แอ็กทิวิตีไดอะแกรม มีดังนี้

สัญลักษณ์	ความหมาย
●	เป็นจุดเริ่มต้นเข้าสู่แอ็กทิวิตีไดอะแกรม
◐	เป็นจุดสำหรับออกจากแอ็กทิวิตีไดอะแกรม
◇	เป็นจุดตัดสินใจเงื่อนไขว่าเป็นจริงหรือเท็จ
○ (Action)	เป็นกิจกรรมที่ทำ
→	แสดงขั้นตอนในการทำงานถัดไป



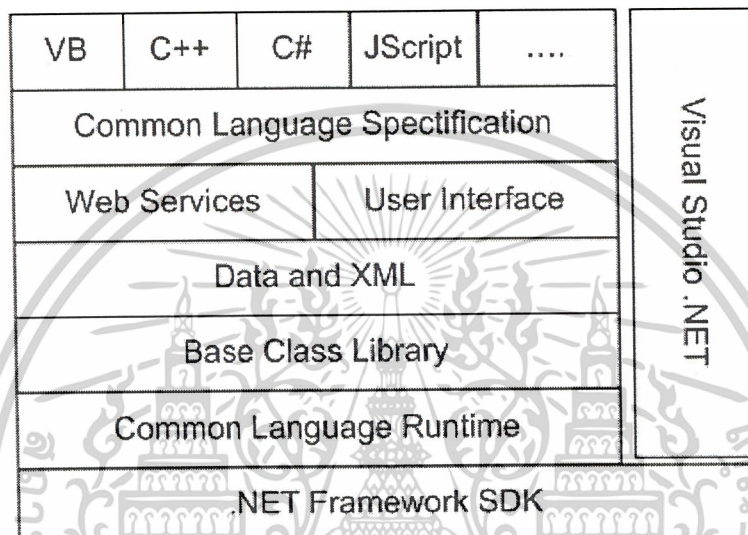
รูปที่ 2.4 แสดงตัวอย่างของแอ็กทิวิตีไดอะแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 เทคโนโลยีค็อตเน็ตเฟรมเวิร์ค (.NET Framework)

### 2.3.1 สถาปัตยกรรม .NET

โครงสร้างโดยรวมทั้งหมดของการสร้างแอปพลิเคชันค็อตเน็ตซึ่งแสดง ได้ดัง โครงสร้าง  
ด้านล่าง



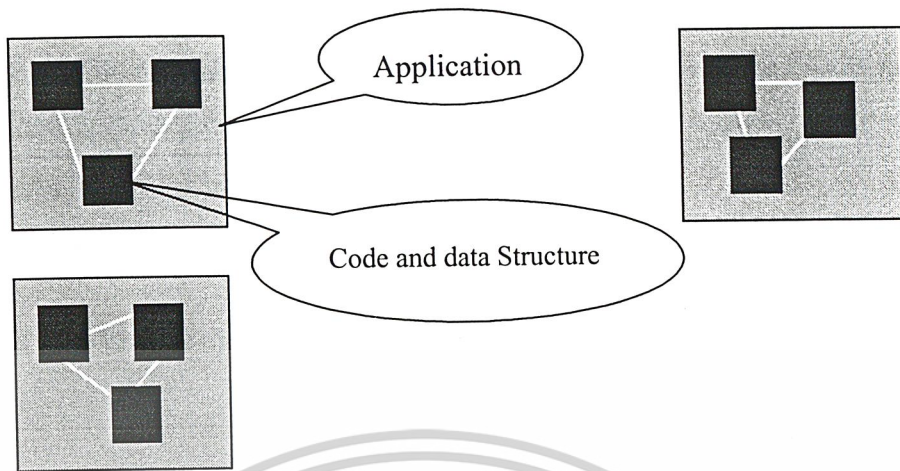
รูปที่ 2.5 รูปแสดงโครงสร้างการพัฒนาแอปพลิเคชันค็อตเน็ต

จากรูป เป็นการแสดงถึงสถาปัตยกรรมของแอปพลิเคชันค็อตเน็ต ที่พัฒนาด้วยวิซวลสตูดิโอค็อตเน็ตทั้งแพลตฟอร์มที่ใช้ในการพัฒนาแอปพลิเคชัน และสถาปัตยกรรมที่ใช้โดยมีเลเยอร์ล่างสุด คือ .NET Framework SDK เปรียบเสมือน Runtime Library ที่จะรันอยู่คอยสนับสนุนการทำงานของแอปพลิเคชันจากนั้นจะเป็นเลเยอร์ของ Common Language Runtime ซึ่งเป็นผลลัพธ์ของการคอมไพล์แอปพลิเคชันค็อตเน็ต เลเยอร์ถัดขึ้นมาเป็นเครื่องมือ (Tools) และเทคนิคต่างๆที่เราสามารถใช้พัฒนาแอปพลิเคชันได้ทั้งในเรื่องของ Webservice, ADO.NET, ASP.NET จนกระทั่งถึงเลเยอร์บนสุดคือ ภาษาที่ใช้ในการพัฒนาแอปพลิเคชันด้วยวิซวลสตูดิโอค็อตเน็ต

### 2.3.2 เลเยอร์ คอมมอนแลงแวกจรันไทม์

ก่อนที่จะมีการพัฒนาโปรแกรมเป็นเชิงวัตถุ (Object Oriented) นั้น แอปพลิเคชันแต่ละตัวเปรียบเสมือนกล่อง ภายในแอปพลิเคชันก็จะมี โค้ด (Code) และมีโครงสร้างข้อมูล (data structure) ต่างๆของตัวเอง มีฟังก์ชันต่างๆของตัวเองแอปพลิเคชันนั้นๆ

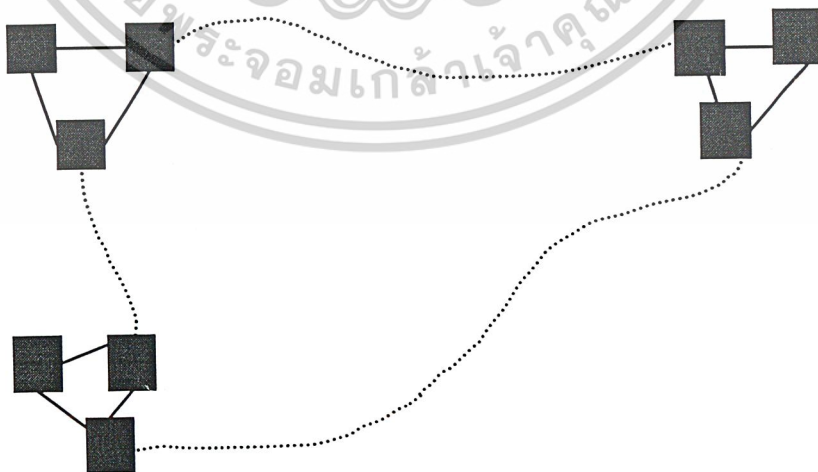
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงหรือทำซ้ำอย่างอื่นของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 โครงสร้างแอปพลิเคชันแรกๆ และการติดต่อกันระหว่างแอปพลิเคชันเป็นเรื่องยาก

การที่แอปพลิเคชันต่างๆ จะมีการเรียกใช้การทำงานของกันและกัน หรือมีการส่งผ่านข้อมูลถึงกันและกัน เป็นสิ่งที่ทำได้ยาก ซึ่งอาจจะต้องมีการกำหนดว่าอะไรขึ้นมาจากระหว่าง 2 แอปพลิเคชันนั้นๆ จนกระทั่งในยุคถัดมา ไมโครซอฟท์ได้คิดค้นเทคโนโลยี COM (Component Object Model) เป็นวิธีที่ทำให้เราเขียนโปรแกรมเป็นแบบเชิงวัตถุ และเรียกใช้การทำงานที่มาจากต่างแอปพลิเคชันได้

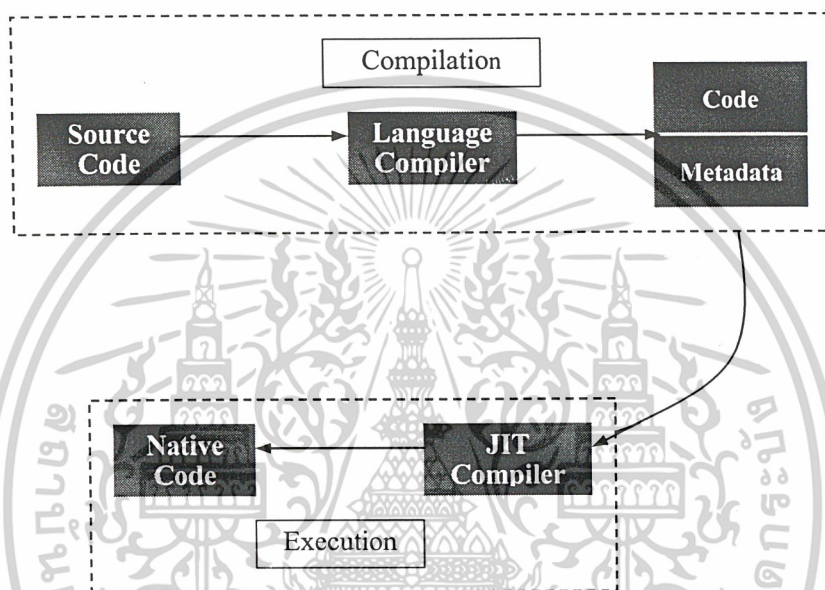
หากเราจะอธิบายให้ง่ายขึ้น ก็คือเปรียบเทียบเหมือนเราเอาแพ็คเกจอันหนึ่งห่อแอปพลิเคชันของเราไว้ โดยการพูดคุยหรือสื่อสารกันของแอปพลิเคชันก็พูดคุยผ่านแพ็คเกจที่เราห่อเอาไว้ จนกระทั่งมาถึงตัววิซวลสตูดิโอคือทเน็ต ที่ได้รับการออกแบบใหม่ จะเห็นว่าจากรูปเดิม กล่องแพ็คเกจหายไป ถ้าเราพัฒนาด้วยรูปแบบเทคโนโลยีคือทเน็ตนั้น คลาสต่างๆ สามารถติดต่อกันได้โดยตรง



รูปที่ 2.7 การพัฒนาแอปพลิเคชันบนเทคโนโลยีคือทเน็ตคลาสภายในแต่ละแอปพลิเคชันสามารถติดต่อกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาแอปพลิเคชันด้วย วิชาลสตูดิโอคือทเน็ดนั้นเมื่อเรากอมไฟล์สิ่งที่เราจะได้ จะไม่ใช่โค้ดไบนารี (binary code) เลขที่เดียว แต่จะได้เป็นภาษากลางอันหนึ่งเรียกว่า MSE (Microsoft Intermediate Language) ซึ่งเป็นภาษาในระดับเลขอร์ต่างๆ โครงสร้างภาษาจะเหมือนภาษา Assembly ภายในสิ่งที่เกิดจากการคอมไพล์ก็จะเป็น Microsoft Intermediate Language ตัวนี้ ภายในตัวมันจะประกอบด้วย 2 ส่วนคือ โค้ดกับตัวแอตทริบิวต์หรือพรีอเพอร์ต์ต่างๆที่ใช้อธิบายตัวโค้ดนั้นซึ่งเรียกว่าเมต้าดาต้า (Meta Data)



รูปที่ 2.8 โครงสร้างการคอมไพล์จากวิชาลสตูดิโอคือทเน็ดและเมื่อนำไปใช้งานจริง

จากนั้น เมื่อโค้ดซึ่งเป็นภาษากลาง (Intermediate Language) ถูกเรียกใช้งานจริงๆ จะมีตัวแปรภาษาตัวหนึ่งมาทำการคอมไพล์โค้ดตัวนั้นให้เป็นโค้ดไบนารี ซึ่งตัวแปรภาษาตัวนั้นจะเรียกว่า JIT Compiler (Just In Time Compiler) เพราะว่าจะมีการคอมไพล์เมื่อมีการใช้งาน ฉะนั้นคลาสหรือโค้ดต่างๆที่เราพัฒนาขึ้นแล้วจะถูกคอมไพล์มาเป็นภาษากลาง ที่มีโครงสร้างภาษาแบบเดียวกัน เพราะฉะนั้นคลาสต่างๆ ในแอปพลิเคชันจึงจะสามารถทำงานได้อย่างกลมกลืนกันและไม่มีข้อติดขัดอะไร

### 2.3.2.1 การทำงานของ คอมมอนแลงเกจรันไทม์ (Common Language Runtime)

ภายในตัวคอมมอนแลงเกจรันไทม์ จะมีโมดูล (Module) ย่อยๆ ซึ่งเป็นสถาปัตยกรรมภายในดังรูป

#### Common Language Runtime

Base Class Library Support		
Thread Support	COM Marshaler	
Type Checker	Exception Manager	
Security Engine	Debug Engine	
IL to Native Compilers	Code Manager	Garbage Collector
Class Loader		

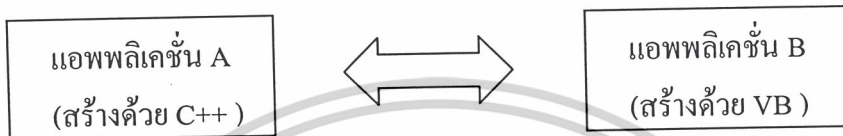
รูปที่ 2.9 แสดงสถาปัตยกรรมของคอมมอนแลงเกจรันไทม์

ด้านล่างสุดจะมี Class Loader ซึ่งเอาไว้โหลดโปรแกรมของเราขึ้นมาทำงาน นอกจากนี้ก็จะมีคอมไพเลอร์ ซึ่งจะทำการคอมไพล์ภาษากลาง ให้เป็นภาษาไบনারีโดยจะมีตัวโค้ดแมนเนเจอร์ (Code Manager) และ การ์เบจคอลเลคเตอร์ (Garbage Collector) คอยจัดการกับหน่วยความจำที่เราจองเวลาเรียกใช้งาน นอกจากนี้ก็จะมีเรื่องความปลอดภัย (Security) ในการทำงาน รวมทั้ง Debug Engine ในการดักจับรันไทม์เออเรอร์ (Runtime error) และ เอ็กซ์เซ็ปชันแมนเนเจอร์ การตรวจเช็คชนิดของตัวแปรต่างๆ และด้านบนสุดจะเป็นการใช้งานร่วมกับไลบรารีคลาสต่างๆซึ่งจัดเตรียมมาให้ เพราะฉะนั้นส่วนของเลเยอร์ของคอมมอนแลงเกจรันไทม์จะสามารถคอมไพล์แอปพลิเคชันใดก็ตาม โดยไม่ว่าจะเป็น ASP.NET หรือเขียนแอปพลิเคชันบนวินโดวส์ธรรมดา หรือจะเป็นการเขียนเว็บเซอร์วิสก็ตาม สิ่งที่ได้จากการคอมไพล์จะเป็น คอมมอนแลงเกจรันไทม์ ตามแผนภาพนี้

นั่นคือ จากการออกแบบเพื่อสนับสนุนการทำงานร่วมกันใน คอมมอนแลงเกจรันไทม์ จึงจัดข้อเสียของ COM ไปได้ เนื่องจากข้อจำกัดของ COM คือเป็นเพียงการเอาอะไรบางอย่างมาห่อคลาสไว้เท่านั้น ดังนั้นเวลาที่เอา COM ไปใช้งานจึงค่อนข้างยุ่งยาก รวมทั้งถ้ามีการเปลี่ยนแปลง COM นั้นๆ ก็จะทำให้เกิดปัญหาเรื่องการเข้ากันได้ (compatibility) ระหว่างเวอร์ชันเดิมกับเวอร์ชันปัจจุบัน แต่ถ้าเราพัฒนาด้วยวิซวลสตูดิโอคือทเน็ต ข้อเสียของ COM ทั้งหมดก็จะถูกขจัดเสีย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความจริงคอมมอนแลงแวกจันไทม์ เป็นหลักการงานที่มีวิวัฒนาการมาจาก COM อีกทีหนึ่ง เป็นเชิงวัตถุที่แกนของภาษาเลข โดยวิชาวสตุคิโอคือทเน็ต นั้น ถูกออกแบบเพื่อสนับสนุนเชิงวัตถุโดยเฉพาะคลาสต่างๆที่อยู่ในแต่ละแอฟฟลิเคชันสามารถติดต่อกันได้โดยตรงในแอฟฟลิเคชัน A เราอาจจะเขียนคลาสด้วย ภาษา C++ และแอฟฟลิเคชัน A อาจจะติดต่อกับแอฟฟลิเคชัน B ซึ่งเขียนด้วยภาษา Visual Basic ได้ คือการสืบทอดข้ามภาษานั้นสามารถทำได้



รูปที่ 2.10 แสดงความสามารถในการติดต่อข้ามแอฟฟลิเคชัน เมื่อเขียนด้วยภาษาที่ต่างกัน

ใน วิชาวสตุคิโอคือทเน็ตจะคอมไพล์เป็นภาษาเดียวกันคือ ภาษากลาง ตามรูป นอกจากนี้ยังสามารถทำงานด้วยกันกับ COM แบบเดิมที่เราเคยเขียนมาแล้วได้ด้วยซึ่งใช้ Visual Studio เดิม โค้ดเหล่านี้ก็ไม่จำเป็นต้องโยนทิ้ง ในวิชาวสตุคิโอคือทเน็ต เราสามารถเรียกใช้งาน คอมโพเนนต์ที่เขียนด้วยวิชาวสตุคิโอคือทเน็ตได้เช่นกัน คือเป็นการเข้ากันได้ทั้งสองทาง (Backward Forward Compatibility) นี่คือข้อดีมากๆ ของ คีอทเน็ต ทำให้เราไม่ต้องมาพัฒนาโค้ดใหม่

### 2.3.2.2 การจัดการเกี่ยวกับหน่วยความจำเมื่อทำการประมวลผล

เนื่องจากการทำงานทั้งหมดของตัวคีโอทเน็ตจะมีการดูแลในเรื่องของคอมมอนแลงแวกจันไทม์ ปัญหาหนึ่งที่มีจะพบ เมื่อเราพัฒนาแอฟฟลิเคชัน คือเรื่องของการ์เบจคอลเลกชันในการจัดการกับหน่วยความจำ โดยเฉพาะบางภาษาอย่างเช่น ภาษาซีพลัสพลัสที่ต้องมีการเรียกใช้งานพอยน์เตอร์ (Pointer) ค่อนข้างมาก ในการใช้งานพอยน์เตอร์ นั้นถ้าเราทำผิดไปนิดเดียวก็อาจจะทำให้แอฟฟลิเคชันของเราแฮงค์ในระหว่างรันหรือประมวลผลก็ได้ ทำให้เมื่อแอฟฟลิเคชันของเรารันไปเรื่อยๆ หน่วยความจำส่วนนี้ก็จะขยายไปเรื่อยๆ และสักพักหนึ่งก็จะเกิดการหยุดไป

การแก้ไขข้อผิดพลาด (Debug) ตรงนี้ทำได้ยากมาก เพราะกว่าจะรู้ว่าเกิดความผิดพลาดขึ้นเราก็เสียเวลาในการรันไปแล้ว ซึ่งการจัดการกับหน่วยความจำนี้ ถ้าเราพัฒนาด้วยตัวแพลตฟอร์มของคีโอทเน็ตจะมี เครื่องมือหนึ่งเรียกว่า การ์เบจคอลเลคเตอร์ เป็นตัวที่คอยจัดการเกี่ยวกับหน่วยความจำให้เรา ความจริงตัวการ์เบจคอลเลคเตอร์ จะเป็นตัวที่ทำการกำหนดค่าหน่วยความจำให้ว่างให้เราเองในส่วนที่เราไม่ใช้งานโดยอัตโนมัติ

นี่คือข้อดีอันหนึ่งในการพัฒนาด้วย แพลตฟอร์มของคีโอทเน็ต

### 2.3.2.3 ระบบการตรวจจับความผิดพลาด

นอกจากนี้ในเรื่องของเอ็กซ์เซ็ปชันแมนเนเจอร์ (Exception Manager) หรือการดักจับไทม์เออเรอร์ (Runtime error) ในการสร้างแอปพลิเคชันที่มีการรันได้อย่างดีนั้น การดักจับไทม์เออเรอร์ ที่เกิดขึ้นในระหว่างการทำงาน ก็เป็นปัจจัยหนึ่งที่เราจะต้องคำนึงถึง ซึ่งในตัวโค้ดที่เขียนจะมีโครงสร้างภาษาในการดักจับไทม์เออเรอร์ ที่มีลักษณะเป็นโครงสร้างทำให้หาได้ง่ายขึ้น เราเรียกว่า สตรักเจอร์เอ็กซ์เซ็ปชันแฮนดิง (Structure Exception Handling)

### 2.3.2.4 รูปแบบการคอมไพล์แอปพลิเคชัน

นอกจากนี้การคอมไพล์แอปพลิเคชันคือทเน็ตมีให้เลือกหลายแบบ บางคนอาจจะคอมไพล์ให้เป็น โค้ดไบนารีเลย ไม่ต้องคอมไพล์ให้เป็นอินเทอร์พรีเตอร์ก็เลยสามารถ จะพัฒนาได้โดยใช้ Visual C++ .NET เรียกว่าแมนเนจซีพลัสพลัส (Manage C++) หรือถ้าเราเลือกใช้เครื่องมือภาษาอื่น ๆ ที่มีความง่ายในการพัฒนายิ่งกว่า คือ Visual Basic หรือซีชาร์ป ก็ สามารถคอมไพล์เป็นอินเทอร์พรีเตอร์ก็เลยสามารถ

ในการคอมไพล์จะไม่มีตัวอินเทอร์พรีรันไทม์ไลบรารี (Interpreter Runtime Library) ต่างๆ เช่น Visual Basic ที่ต้องมีรันไทม์ไลบรารีของตัวเองเวลาพัฒนาด้วย Visual Studio .NET นี้ก็จะไม่จำเป็นต้องมีอีกต่อไป เราใช้ตัวรันไทม์อันเดียวกัน ก็คือตัว คีโตนเน็ตเฟรมเวิร์ค เป็นตัวรันไทม์ไลบรารี ที่กล่าวมาข้างต้น

### 2.3.2.5 รูปแบบการทำงานร่วมกับภาษาอื่น ๆ

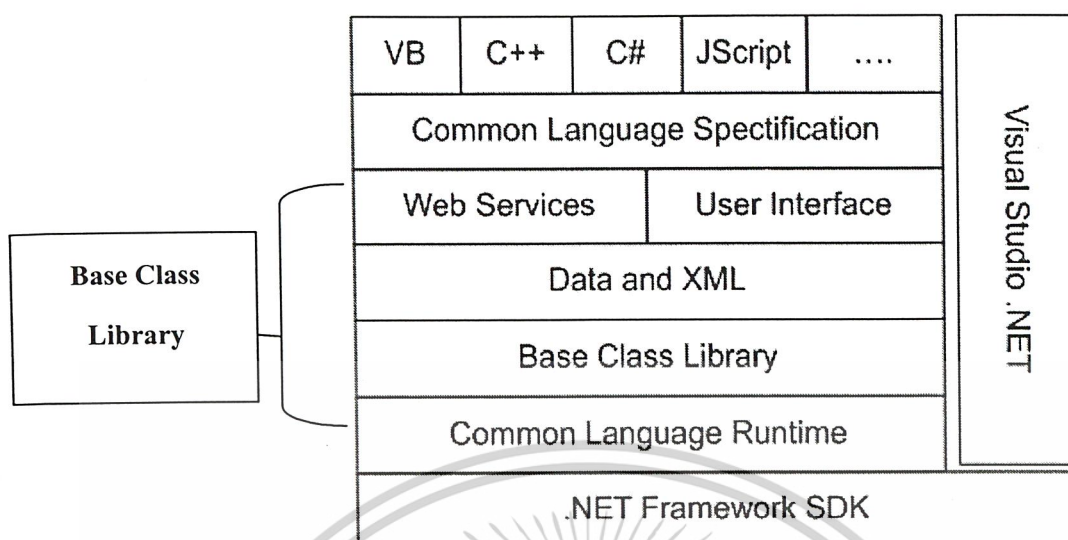
การพัฒนาแอปพลิเคชันในคีโตนเน็ต รวมไปถึงเทคนิคต่างๆ ที่กล่าวมาแล้วนั้น ไม่จำเป็นจะใช้เฉพาะภาษาตระกูลใน Visual Studio .NET ไม่ว่าจะเป็น Visual Basic คีโตนเน็ต (VB.NET), ซีชาร์ป (C#) หรือซีพลัสพลัส (C++) ถ้าภาษาใดก็ตามสามารถคอมไพล์มาเป็นคอมมอนแลงเกจรันไทม์ได้ภาษานั้นก็จะสามารถใช้คุณลักษณะทั้งหมดของคีโตนเน็ตได้ซึ่งตอนนี้มีภาษาอื่น ๆ ที่สามารถทำงานร่วมกับแอปพลิเคชันคีโตนเน็ต มากกว่า 20 ภาษาที่พัฒนาตัวคอมไพล์ขึ้นมาเพื่อคอมไพล์ภาษาต่างๆ ให้อยู่ในรูปของแพลตฟอร์ม คีโตนเน็ต ไม่ว่าจะเป็นภาษาโคบอล (COBOL), ปาสคาล (PASCAL) และอื่นๆ

ในตัว Visual Studio คีโตนเน็ต ก็มีเครื่องมือระดับสูง (High Level Tools) ที่จะทำให้เราพัฒนาแอปพลิเคชันได้สะดวกและง่ายมากขึ้น นี่เป็นเรื่องของคอมมอนแลงเกจรันไทม์ ซึ่งถือได้ว่าเป็นเลเยอร์ล่างสุดของการคอมไพล์แอปพลิเคชันที่เขียนด้วย Visual Studio คีโตนเน็ต

## 2.3.3 เลเยอร์เบสคลาสไลบรารีแลงเกจ (Base Class Library Language)

เลเยอร์ถัดมาเป็นของโครงสร้างการพัฒนาแอปพลิเคชันคีโตนเน็ตต่อจากคอมมอนรันไทม์แลงเกจ ก็คือเบสคลาสไลบรารี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 เบสคลาสไลบรารีของสถาปัตยกรรมคือทเน็ต

เทคนิคต่างๆที่วิศวกรรมคือทเน็ตจัดเตรียมให้เราในการใช้งานนั้นเบสคลาสไลบรารีเปรียบเสมือนเป็นการรวบรวมเอาฟังก์ชันของเอพีไอ (API: Application Programming Interface) ทั้งหมดสมมติว่าตอนที่เราพัฒนาแอปพลิเคชันด้วยวิศวกรรมคือทเน็ต เวลาเราเรียกใช้งานฟังก์ชันบางอย่างในระดับสูง หรือต้องการทำงานแบบลึกๆกับระบบเรามักจะเรียกใช้เอพีไอ ภาษาในวิศวกรรมคือทเน็ตมีความสามารถในการเรียกฟังก์ชันเอพีไอได้ต่างกัน

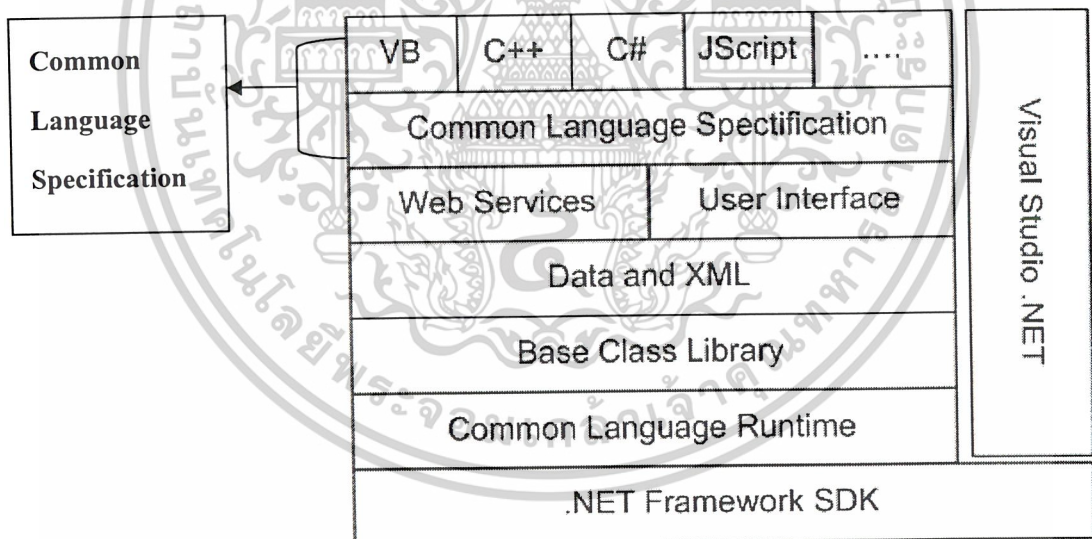
ตัวเบสคลาสไลบรารี ก็คือ การที่เรารวบรวมฟังก์ชันเอพีไอซึ่งกระจัดกระจายอยู่เวลาจะเรียกใช้เราต้องไปค้นหาในเมนูช่วยเหลือ (Help) นั่นคือเบสคลาสไลบรารี พยายามที่จะรวบรวมเอพีไอและฟังก์ชันทั้งหมดเกี่ยวกับระบบเข้ามาไว้ในลักษณะของเชิงวัตถุทำเป็นคลาสอันหนึ่งซึ่งเป็นมาตรฐาน เป็นคลาสที่สร้างมาในตัวระบบเรียบร้อย ซึ่งคลาสทั้งหมดจะอยู่ภายใต้คลาสหลักอันหนึ่งที่เรียกว่า ซิสเต็ม (System) ทุกอย่างที่พัฒนาด้วยภาษาในวิศวกรรมคือทเน็ต จะเป็นเชิงวัตถุทั้งหมด โดยมีคลาสที่ใหญ่ที่สุดโดยเรียกว่าคลาสซิสเต็ม ซึ่งภายในคลาสซิสเต็ม จะมีคลาสย่อยๆมากมาย ซึ่งแต่ละอันจะสนับสนุนฟังก์ชันเอพีไอหรือสนับสนุนการทำงานที่เราต้องการได้ไม่ว่าจะเป็นเรื่องของการทำกราฟิก การทำเกี่ยวกับโครงสร้างข้อมูลการทำเกี่ยวกับเรื่องเครือข่าย ฟังก์ชันเอพีไอเหล่านี้จะถูกจัดกลุ่มให้เป็นเชิงวัตถุ อยู่ภายใต้คลาสซิสเต็ม การเรียกใช้งานคลาสซิสเต็ม ถ้าเป็นวิศวกรเบสกับซัพพอร์ตก็ใช้งานได้ทั้ง 2 อย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.4 เลเยอร์คอมมอนแลงเกจสเปคซิฟิเคชัน (Common Language Specification

#### Layer)

เลเยอร์สุดท้ายในสถาปัตยกรรมคือท่อนเน็ตที่เราจะพูดถึงก็คือเลเยอร์คอมมอนแลงเกจสเปคซิฟิเคชัน เรื่องของมาตรฐานบนพื้นฐานคือท่อนเน็ต ซึ่งคอมไพเลอร์จะต้องทำงานตามมาตรฐานดังกล่าว เพื่อให้สามารถทำงานร่วมกับภาษาบนพื้นฐานคือท่อนเน็ต และภาษาอื่นๆได้ ไมโครซอฟท์ได้ทำการปรับภาษาต่างๆ เช่น ซีชาร์ป, วิซวลเบสิก, ให้เข้ามาตรฐานคือท่อนเน็ต นอกจากนั้น ผู้ผลิตรายอื่นสามารถพัฒนาตามข้อกำหนดนี้เพื่อให้สามารถทำงานบนพื้นฐานคือท่อนเน็ตได้ เลเยอร์ที่ 3 ของสถาปัตยกรรมคือท่อนเน็ต ก็คือเครื่องมือที่ใช้ในการสร้างแอปพลิเคชัน แต่สิ่งที่อยู่เหนือกว่าทุกๆ อย่างก็คือภาษาที่เราใช้งานคือท่อนเน็ต นั่นมีข้อดีคือ ต้องสนับสนุนมาตรฐานเดียวกัน เรียกว่า คอมมอนแลงเกจสเปคซิฟิเคชัน ซึ่งไมโครซอฟท์ได้จดทะเบียนมาตรฐานนี้เข้ากับองค์กร ECMA แล้ว ซึ่งเป็นองค์กรที่ดูแลเรื่องโครงสร้างของภาษาต่างๆ และก็เป็นแบบเปิดด้วยเพราะฉะนั้นเจ้าของภาษาอื่นก็สามารถสร้างตัวแปลภาษา หรือคอมไพเลอร์ เพื่อคอมไพล์ภาษาของเขาให้เข้ามาเป็นคอมมอนแลงเกจสเปคซิฟิเคชัน อันนี้

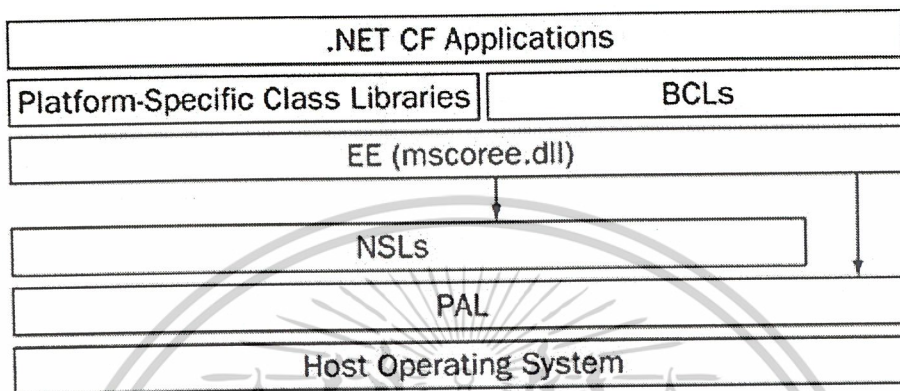


รูปที่ 2.12 คอมมอนแลงเกจสเปคซิฟิเคชันของสถาปัตยกรรมคือท่อนเน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4 สถาปัตยกรรมของคีย์บอร์ดคอมแพคเฟรมเวิร์ค (.NET Compact Framework)

ส่วนหนึ่งของคีย์บอร์ดคอมแพคเฟรมเวิร์ค ถูกสร้างขึ้น โดยเป็นส่วนย่อยของ เดสทอป (Desktop Framework) โดยมีเป้าหมายหลักคือ สามารถเข้ากันได้กับทุกแพลตฟอร์ม (Platform)



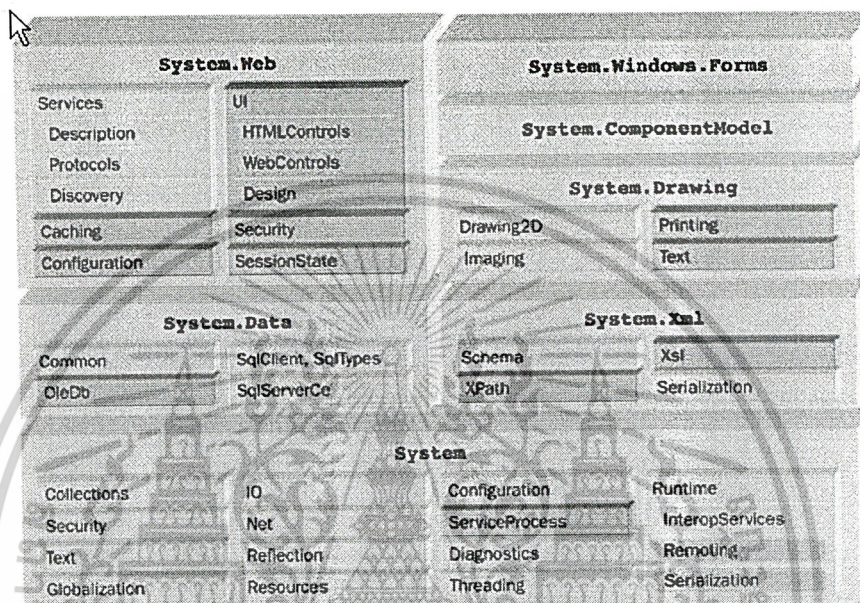
รูปที่ 2.13 แสดงสถาปัตยกรรมของคีย์บอร์ดคอมแพคเฟรมเวิร์ค

ขนาดของคอมแพคเฟรมเวิร์ค จะถูกติดตั้งบนอุปกรณ์ต่างๆ แต่ละอุปกรณ์ โดยปกติจะมีขนาด 1.7 MB – 2.6 MB และสามารถติดตั้งใน RAM, ROM, หรือ FlashROM โดยในอนาคต คาดว่า ถ้าเราเริ่มด้วย RAM เราจะสามารถให้ใช้งาน ได้ทุกอุปกรณ์ทันที OEMs จะทำให้ FlashROM พัฒนาขึ้น และสามารถรวมอยู่ในคอมแพคเฟรมเวิร์ค สำหรับอุปกรณ์พวก Pocket PC

- Host Operating System เป็นชั้นล่างสุดของสถาปัตยกรรมคอมแพคเฟรมเวิร์ค โดยส่วนที่เป็น ไลค์ ที่เขียนสำหรับคอมแพคเฟรมเวิร์ค จะถูกประมวลผลบน Host Operating System เช่น Windows CE โดย คอมแพคเฟรมเวิร์คจะถูกรันบน Windows CE 3.0 และ Windows CE.NET 4.1
- PAL คือส่วนประกอบที่สำคัญ ที่จะทำให้พอร์ตของแพลตฟอร์มสามารถทำงาน ได้ ซึ่งจะบรรจุระบบย่อยๆ ที่เป็นฟังก์ชันการทำงานของระบบปฏิบัติการ (OS) และระบบฮาร์ดแวร์ (Hardware) โดยที่ฮาร์ดแวร์จะอยู่ในส่วนของ API, NSL, EE ตัวอย่างเช่น PAL จะประกอบด้วยอินเทอร์เฟซ (Interface) สำหรับอุปกรณ์ต่างๆ, ระบบจัดการหน่วยความจำและพอร์ตติดต่ออุปกรณ์ภายนอก (I/O Port)
- Native Support Librarys (NSLs) อุปกรณ์แต่ละอุปกรณ์ไม่สามารถรองรับบริการในรูปแบบเดียวกันได้ โดยที่คอมแพคเฟรมเวิร์ค ก็จะใส่กลุ่มของ NSLs เข้าไปและจะทำให้คอมแพคเฟรมเวิร์ค มีลักษณะที่ต้องการ เช่น Heap management, Cryptography, Graphical User Interface (GUI) โดยบริการเหล่านี้จะถูกเรียกใช้ใน PAL และถูกเรียกกลับคืน โดย EE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-EE จะมีกลุ่มของบริการที่สำคัญ ซึ่งแต่ละส่วนจะทำได้เหมือน CLR โดยให้บริการที่เหมือนกันสำหรับเดสทอปและเซิร์ฟเวอร์แอปพลิเคชัน ซึ่งจะต้องมีการประมวลผลบนค็อทเน็ตแอปพลิเคชัน แต่อาจจะใช้ทรัพยากรมาก แต่ในความเป็นจริงแล้ว ตัวหน่วยความจำมันจะมีข้อจำกัด ซึ่งในตัวของ EE จะถูกสร้างมาโดยคำนึงถึงข้อจำกัดนี้



รูปที่ 2.14 แสดงคลาสไลบรารีของค็อทเน็ตคอมแพคเฟรมเวิร์ค

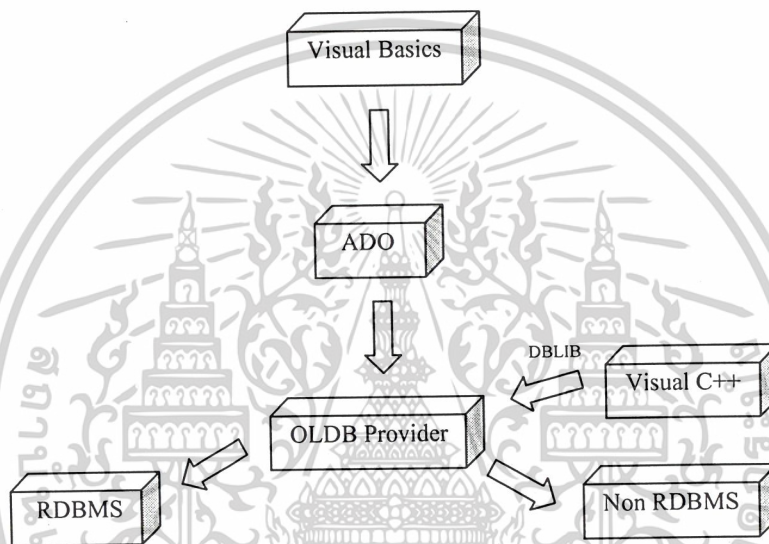
- Class Library เพื่อที่จะสร้างโรบัสโปรแกรมคอมแพคเฟรมเวิร์ค ก็มีกลุ่มของ Class Library เป็นภาษาในการสร้างไฮราคิคอลเนมสเปซ (Hierachical namespace) ซึ่งก็จะเหมือนกับที่เราเจอในเดสทอปเฟรมเวิร์ค แต่ก็จะมีส่วนที่ต่างกันอยู่ 4 อย่าง
  - อาจจะมองได้ว่า คอมแพคเฟรมเวิร์ค เป็นซับเซตของ Desktop Framework
  - ส่วนที่สองที่ทำให้ คอมแพคเฟรมเวิร์ค ต่างจาก Desktop Class Library โดยที่จะมีการพัฒนาที่ต่างจากเดิม
  - คอมแพคเฟรมเวิร์คจะสามารถรองรับ Name space ขึ้นอีก 2 ชนิด โดยแต่ละชนิดจะมีวิธีการที่จะแอสเซมบลีของตัวเอง ซึ่งจะทำให้มีการประมวลผลที่ดีขึ้น คล้ายกับว่าทำให้อุปกรณ์ฉลาดขึ้น
  - คอมแพคเฟรมเวิร์คจะสามารถรองรับการทำงานได้น้อยกว่าของ เดสทอปเฟรมเวิร์ค ซึ่งส่วนที่ คอมแพคเฟรมเวิร์ค สามารถรองรับได้มักจะถูกอ้างถึงในนามของ Common Type System (CTS) เพราะว่าเป็นส่วนสำคัญที่จะสามารถรองรับได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 เทคโนโลยีเอดีโอดีออตเน็ต (ADO.NET)

เทคโนโลยีเอดีโอดีออตเน็ต (ActiveX Data Object Dot Net) เป็น เอพีไอ ที่ใช้ในการติดต่อกับฐานข้อมูลต่างๆ ซึ่งตัวเอดีโอดีออตเน็ต นี้ถูกสร้างมาเพื่อทดแทนอินเทอร์เฟซแบบเก่าที่มีความจำกัดในหลายๆอย่างคือ DAO (Data Access Objects) และ RDO (Remote Data Objects:R)

ตัวเอดีโอดีออตเน็ตเป็นอีกเทคโนโลยีใหม่อันหนึ่งที่จะช่วยในเรื่องการติดต่อกับแหล่งข้อมูลต่างๆ และเอดีโอดีออตเน็ต ยังมีโครงสร้างเหมือนเอดีโอดีออต เพียงแต่ว่าการออกแบบนั้นมีความแตกต่างกันบ้าง



รูปที่ 2.15 สถาปัตยกรรมการเชื่อมต่อข้อมูลผ่านเอดีโอดีออต

การเก็บข้อมูลที่เก็บในเอดีโอดีออต เมื่อมีการเลือกข้อมูลนั้น ผลลัพธ์ที่ออกมาจะเก็บข้อมูลในโครงสร้างของ COM (Common Object Model) แต่ตัวเอดีโอดีออตเน็ต เมื่อเราเลือกข้อมูลออกมาแล้วต่างๆที่เลือกออกมาจะเก็บไว้ในรูปของ เอ็กซ์เอ็มแอล (XML) เพราะฉะนั้น เอดีโอดีออตเน็ตจึงเหมาะกับการทำงานที่มี เป็นแอปพลิเคชันบนอินเทอร์เนต ถ้าเป็นเอดีโอดีออตเน็ต จะเหมาะกับการทำงานที่เป็นไคลเอนต์/เซิร์ฟเวอร์ ที่อยู่ในเครือข่ายแลนหรือเครือข่ายอินทราเน็ต (Intranet) ในองค์กรของเรามากกว่า

การเชื่อมต่อฐานข้อมูลบนเครือข่ายภายในองค์กรนั้น เราสามารถกำหนดการเชื่อมต่อได้คือ เมื่อเปิดแอปพลิเคชัน เราก็ค่อยมาเปิดการเชื่อมต่อฐานข้อมูล ซึ่งเรียกว่าเป็น Connection Oriented แต่เมื่อเอาไปใช้กับการทำงานที่เป็นแอปพลิเคชันบนอินเทอร์เนต ซึ่งเป็นระบบที่จำนวนผู้ใช้ไม่แน่นอน และสูงกว่าระบบเครือข่ายในบริษัทแน่นอน เพราะฉะนั้นการที่เราจะมาเชื่อมต่อกับ

ฐานข้อมูลทั้งไว้จึงดูไม่เหมาะกับการทำงานในเครือข่ายอินเทอร์เน็ตนัก เอดีโอคือทเน็ตจึงมีคุณสมบัติที่สนับสนุน Disconnected Connection ด้วย

นอกจากนั้นยังสามารถใช้กับแหล่งข้อมูลหลายๆแบบเหมือนเดิม เช่น Access กับ OLE DB Provider ของหลายๆแบบ หรือกระทั่งข้อมูลใน Outlook Exchange เราก็สามารถใช้ เอดีโอคือทเน็ตไปเลือกออกมาได้เช่นเดียวกัน

### 2.5.1 ออบเจกต์ที่สำคัญในเอดีโอคือทเน็ต

ออบเจกต์ที่ใช้บ่อยๆใน เอดีโอคือทเน็ตซึ่งจะกล่าวถึงในส่วนนี้ก็คือ Connection, Dataset และ Data Adapter

#### 2.5.1.1 ออบเจกต์คอนเน็คชัน (Connection Object)

ออบเจกต์ตัวแรกก็คือ Connection เป็นออบเจกต์ที่อยู่ใน เอดีโอคือทเน็ตหน้าที่คือ การเปิดการเชื่อมต่อ(Connection)ระหว่างไคลเอ็นต์ของเรากับแหล่งข้อมูล โดยที่เราสามารถกำหนดรูปแบบการเชื่อมต่อไปยังฐานข้อมูลที่ต้องการ พร้อมค่ารายละเอียดต่างๆ เช่น รหัสผ่านเมื่อจะเชื่อมต่อกับฐานข้อมูล ซึ่งยังคงคล้ายกับ เอดีโอแบบเดิมนั่นเอง

#### 2.5.1.2 ออบเจกต์คอมมานด์ (Command Object)

ออบเจกต์ตัวนี้มีหน้าที่เอาไว้เขียนคำสั่ง เอสคิวแอล (SQL) จากไคลเอ็นต์ไปทำงานบนแหล่งข้อมูล เช่นเดียวกับออบเจกต์ Connection ก็ใน เอดีโอปกติก็ยังมีออบเจกต์นี้อยู่

การใช้งานออบเจกต์ Command ในเอดีโอคือทเน็ตก็เหมือนกันกับการทำงานใน เอดีโอเดิมคือเวลาที่เราจะส่งคำสั่งเอสคิวแอลที่ไม่ใช่การเลือก (Select) เช่น Insert, Update หรือ Delete ที่ไม่มีการคืน (Return) ค่ากลับมานั้น ซึ่งส่วนใหญ่ในทุกฐานข้อมูลจะมีโครงสร้างภาษาเอสคิวแอล ให้เราสามารถสร้างฟังก์ชันฝังไว้ในตัวฐานข้อมูลได้

#### 2.5.1.3 ออบเจกต์ดาต้าเซต (Dataset Object)

ออบเจกต์ Dataset นี้มีเฉพาะในเอดีโอคือทเน็ตเท่านั้น ดาต้าเซตเปรียบเสมือนฐานข้อมูลทั้งก้อนเลย คือเมื่อเรา เลือก(Select)ออกมา เราไม่ได้เฉพาะแถวของข้อมูลออกมา แต่ได้ข้อมูลอันใหม่อันหนึ่งโดยเป็นฐานข้อมูลที่อยู่ในหน่วยความจำของเรา

ภายในดาต้าเซตนั้น จะมองว่าเป็นเหมือนฐานข้อมูลเหมือนกัน ในนั้นมีตาราง (tables) , คอลัมน์(columns) และแถว (rows) ต่างๆอยู่ภายใน มีความสัมพันธ์ระหว่างตารางได้มีการกำหนดข้อบังคับความถูกต้อง (constrain), ครอบค้ำ (index key), คีย์ร่วม (public key) ได้ โดยการเก็บข้อมูลของดาต้าเซตจะเก็บโครงสร้าง (schema) และข้อมูลของตัวมันด้วย เราสามารถเลือกข้อมูลออกมาเก็บไว้ในดาต้าเซต จากนั้นก็สามารถออกจากการเชื่อมต่อได้โดยเราก็จะมีดาต้าเซตเอาไว้ใช้งาน จากนั้นภายในโปรแกรมเราก็สามารถใช้คำสั่ง Insert, Update, Delete ลงไปใน Dataset ได้ จนเสร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียบร้อยแล้วเราสามารถกลับมาเชื่อมต่อกับฐานข้อมูลอีกทีหนึ่ง แล้วทำการซิงค์ (Synch) ข้อมูลกลับขึ้นไปได้ โดยตัว เอดีไอคือทนี่จะคอยจัดการความแตกต่างระหว่างข้อมูลด้วยตัวเอง

การซิงค์ข้อมูลกลับขึ้นไปในลักษณะนี้ คือการเขียน โปรแกรมแบบ Disconnected Programming โดยไม่จำเป็นต้องเชื่อมต่อฐานข้อมูลตลอดเวลา เพราะแอปพลิเคชันบนอินเทอร์เน็ต นั้น ถ้าเราเชื่อมต่อฐานข้อมูลไว้ตลอดเวลาจะเป็นการสิ้นเปลืองทรัพยากรของเซิร์ฟเวอร์มาก

#### 2.5.1.4 ออบเจกต์ดาต้าอะแดปเตอร์ (Data Adapter Object)

ออบเจกต์ ดาต้าอะแดปเตอร์ มีเฉพาะในเอดีไอคือทนี่เช่นกัน ออบเจกต์ดาต้าอะแดปเตอร์ ก็เหมาะสำหรับการเลือก(Select) ทั่วๆ ไป ซึ่งเปรียบเสมือนเป็นสะพานเชื่อมระหว่างแหล่งข้อมูลที่มีข้อมูลกับตัวดาต้าเซต ของเราโดยดึงข้อมูลจากแหล่งข้อมูลเอามาเก็บในดาต้าเซต จากนั้นจึงสามารถใช้คำสั่งเอสคิวแอลในดาต้าเซต ทำการเปลี่ยนแปลงข้อมูลในแถวต่างๆ ได้ และหากต้องการซิงค์ข้อมูลจากดาต้าเซต กลับไปในแหล่งข้อมูลใหม่ เราก็ใช้ดาต้าอะแดปเตอร์ ทำการอัปเดตข้อมูลกลับขึ้นไปในแหล่งข้อมูล

หากเปรียบเทียบภาพของดาต้าอะแดปเตอร์ ก็เหมือนสะพานเชื่อม โยงระหว่างฐานข้อมูลของเรากับดาต้าเซต นั่นเอง การที่จะเอาข้อมูลมาเก็บไว้ในดาต้าเซต เราต้องสร้างออบเจกต์ขึ้นมา 2 ตัว ตัวแรกเป็นดาต้าเซต ตัวที่ 2 เป็นดาต้าอะแดปเตอร์ หลังสร้างเสร็จแล้วให้ใช้คำสั่งเอสคิวแอล เพื่อดึงข้อมูลมาจากนั้นใช้คำสั่ง Fill เพื่อนำข้อมูลที่ได้จากการอ่านเข้ามาใส่ในดาต้าเซต

คุณลักษณะ	ADO	ADO.NET
สถาปัตยกรรมที่ใช้	บนพื้นฐานคอม (COM) การเชื่อมต่อแบบ Connection Oriented	XML กับ HTTP การเชื่อมต่อแบบ Connectionless
การเข้าถึงระบบป้องกัน Firewall	ไม่ผ่านระบบ Firewall	ใช้ HTTP และสามารถผ่าน Firewall ได้เป็นอย่างดี
ออบเจกต์ที่ใช้	Connection, Command, Recordset	Connection, Command, Dataset, DataSetview, Dataview

ตารางที่ 2.1 เปรียบเทียบการเชื่อมต่อฐานข้อมูลระหว่างเทคโนโลยี ADO และ ADO.NET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6 เอสคิวแอล เซิร์ฟเวอร์ 2000

เอสคิวแอล เซิร์ฟเวอร์ 2000 (SQL Server 2000 : Sequential Query Language Server 2000) จัดเป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS : Relational Database Management System) ตัวหนึ่ง ซึ่งเป็นที่นิยมกันมากในปัจจุบัน สามารถจัดการระบบฐานข้อมูลอย่างมีประสิทธิภาพ และสนับสนุนการใช้งานบนระบบปฏิบัติการวินโดวส์ (Windows) ได้เป็นอย่างดีโดยเอสคิวแอล เซิร์ฟเวอร์ 2000 (SQL Server 2000 : Sequential Query Language Server 2000) ถูกออกแบบมาให้ทำงานในลักษณะที่เป็นระบบฐานข้อมูลระหว่างเครื่องเซิร์ฟเวอร์ กับเครื่องลูกข่าย (Client Server Database) ในระบบปฏิบัติการวินโดวส์ เอ็นที (Windows NT) และระบบปฏิบัติการวินโดวส์ 2000 (Windows 2000) นอกจากนี้ เอสคิวแอล เซิร์ฟเวอร์ 2000 ยังสามารถใช้งานร่วมกับเว็บดีเวลอปเมนต์แพลตฟอร์ม (Web Development Platform) ทั้งหมด ไม่ว่าจะเป็นภาษาซี (C) ซีพลัสพลัส (C++) จาวา (Java) เพิร์ล (Perl) ทีซีแอล (Tcl) หรือ เอเอสพี (ASP) ก็ตาม

เอสคิวแอล เซิร์ฟเวอร์ 2000 เป็นระบบจัดการฐานข้อมูลแบบ Client Server Relational Database ทำให้ช่วยเพิ่มประสิทธิภาพในการทำงาน และยังมีระบบจัดการเป็นแบบควบคุมจากศูนย์กลาง (Centralized Management) โดยระบบรักษาความปลอดภัย ระบบจัดสรรการทำงาน และใช้ข้อมูลจากหลายๆ งานพร้อมกันได้ ซึ่งประกอบด้วยส่วนหลักๆ 3 ส่วน คือ ส่วนของผู้ให้บริการ (Server) ส่วนของผู้ใช้บริการ (Client) และข่ายการสื่อสารหรือเน็ตเวิร์ค โดยในแต่ละส่วนก็จะมีโปรแกรมสำหรับการทำงานตามหน้าที่ของตน

### 2.6.1 ส่วนของผู้ให้บริการ (Server)

เป็นเครื่องคอมพิวเตอร์เครื่องหลักในระบบเครือข่าย (Network) ที่ทำการติดตั้ง โปรแกรมการทำงานเอสคิวแอลเซิร์ฟเวอร์ โดยทำหน้าที่จัดเก็บรวบรวม ค้นหา เรียกดู จัดการข้อมูลในรูปของฐานข้อมูล และจัดเก็บข้อมูลที่เกิดจากการที่ผู้ใช้แต่ละคนสร้างขึ้นมา เพื่อที่คอมพิวเตอร์ในเครือข่ายจะมาเรียกไปใช้ได้

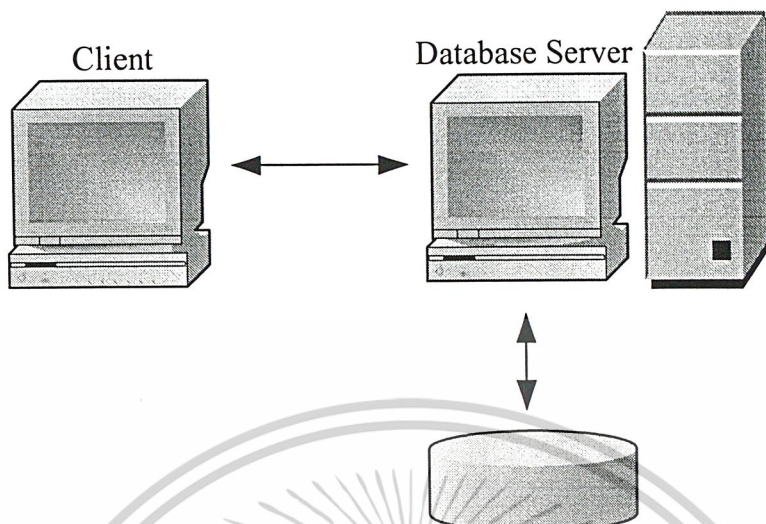
### 2.6.2 ส่วนของผู้ใช้บริการ (Client)

เป็นเครื่องคอมพิวเตอร์ที่ใช้ทำการติดตั้ง เว็บดีเวลอปเมนต์แพลตฟอร์มต่างๆ เช่น จาวา เพิร์ล พีเอชพี เอเอสพี เป็นต้น และ เชื่อมต่อกับเอสคิวแอลเซิร์ฟเวอร์ ฝั่งเซิร์ฟเวอร์ โดยโปรแกรมฝั่งไคลเอนต์ จะทำหน้าที่ส่งและรับข้อมูลจากดาต้าเบส และมีโปรแกรมสำหรับเรียกดู และจัดการข้อมูล โดยมีสิทธิ์เรียกดูข้อมูลบนเครื่องเซิร์ฟเวอร์ได้

### 2.6.3 ข่ายการสื่อสารหรือเน็ตเวิร์ค (Communication Network)

เป็นเส้นทางในการติดต่อแลกเปลี่ยนข้อมูลและคำสั่ง ระหว่างเครื่องเซิร์ฟเวอร์และเครื่องไคลเอนต์ ดังนี้

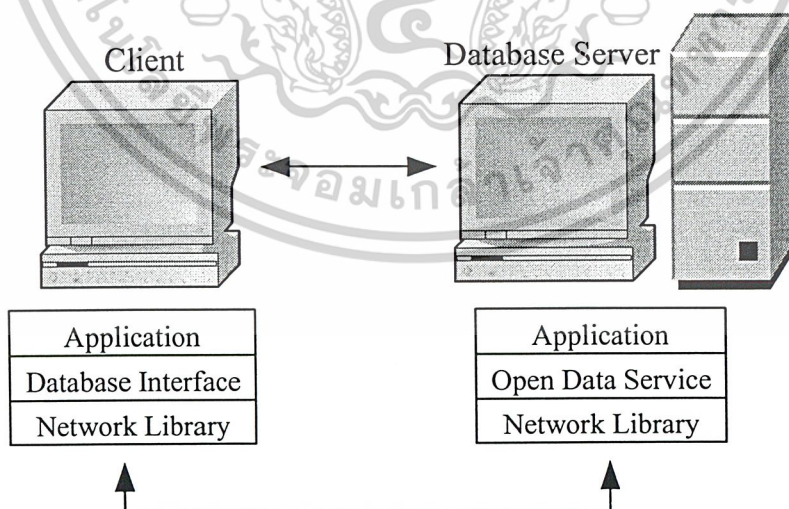
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 แสดงเส้นทางการติดต่อแลกเปลี่ยนข้อมูลและคำสั่ง ระหว่างเครื่องเซิร์ฟเวอร์และเครื่องไคลเอนต์

#### 2.6.4 สถาปัตยกรรมของเอสคิวแอล เซิร์ฟเวอร์ (SQL Server)

สถาปัตยกรรมหรือโครงสร้างภายในของเอสคิวแอลเซิร์ฟเวอร์ จะมีส่วนที่ทำหน้าที่ติดต่อสื่อสารกับทางฝั่งไคลเอนต์อยู่ด้วย แต่ก็แยกส่วนจัดการเน็ตเวิร์คและโปรโตคอลออกจากส่วนที่เป็นแอปพลิเคชัน ทำให้แอปพลิเคชันสามารถทำงานอยู่บนเน็ตเวิร์คแบบใดก็ได้



รูปที่ 2.17 แสดงเส้นทางการติดต่อแลกเปลี่ยนข้อมูลและคำสั่ง ระหว่างเครื่องดาต้าเบสเซิร์ฟเวอร์และเครื่องไคลเอนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยอธิบายความหมายจากรูปได้ดังนี้

- แอปพลิเคชัน (Application) หมายถึง โปรแกรมที่พัฒนาขึ้นเพื่อใช้ข้อมูลจากดาต้าเบสผ่าน อินเทอร์เน็ตของ โปรแกรมที่เรียกว่าเอพีไอ (API : Application Program Interface) เพื่อใช้งานกับดี วิตอปเมนต์แพลตฟอร์มต่างๆมากมาย ไม่ว่าจะเป็นภาษาซี ซีพลัสพลัส จาวา ไอเฟล เพิร์ล หรือ ทีซี แอล เป็นต้น และนอกจากนี้ยังสามารถใช้งานร่วมกับ โอดีบีซี (ODBC : Open Database Connectivity) ซึ่งทำให้เราสามารถใช้งานได้กับเครื่องมืออื่นๆ บนวินโดวส์แพลตฟอร์ม เช่น Access เป็นต้น รวมทั้ง สามารถนำมาประยุกต์เพื่อใช้งานร่วมกับ ASP (Active Server Page)

- ดาต้าเบสอินเทอร์เฟซ (Database Interface) หมายถึง อินเทอร์เน็ตที่ใช้โดยแอปพลิเคชัน เพื่อติดต่อไปยังเอสคิวแอลเซิร์ฟเวอร์ เช่น ODBC โดยสร้างการติดต่อระหว่าง แอปพลิเคชัน และ OLE DB (Object Linking and Embedding Database)

- เน็ตเวิร์คไลบรารี (Network Library) หมายถึง โปรแกรมที่ทำหน้าที่จัดส่งข้อมูลผ่านทางเน็ตเวิร์ค โดยเอสคิวแอล เซิร์ฟเวอร์ ซึ่งเซิร์ฟเวอร์ มีเน็ตเวิร์คไลบรารี ที่ทำงานร่วมกับ โปรโตคอลได้หลายประเภท เช่น ทีซีพีไอพี (TCP/IP) เนมไปปี้ (Named Pipe หรือ NT) เป็นต้น

- โอเพนดาต้าเซอร์วิส (Open data Service) เป็นอินเทอร์เน็ตระหว่างเน็ตเวิร์คไลบรารีกับ แอปพลิเคชันฝั่งเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การออกแบบ

#### 3.1 ข้อมูลเบื้องต้น

เนื่องจากโครงการที่ทำขึ้นเกี่ยวข้องกับระบบร้านอาหาร ดังนั้นก่อนที่จะทำโครงการนี้ทางคณะผู้จัดทำจึงต้องทำการศึกษาค้นคว้าและเก็บรวบรวมความรู้เรื่องร้านอาหาร รวมถึงวิเคราะห์ถึงปัญหาต่างๆที่เกิดขึ้น และนอกจากนี้โครงการได้มีการนำเครื่องพีดีเอ (Personal Digital Assistants) มารวมประยุกต์ใช้กับโครงการด้วย ซึ่งมีรายละเอียดดังนี้

##### 3.1.1 วิเคราะห์ข้อมูลของร้านอาหาร

ในปัจจุบันธุรกิจร้านอาหารนับได้ว่าเป็นเป็นธุรกิจที่น่าสนใจอีกธุรกิจหนึ่ง เนื่องจากมีความเกี่ยวข้องกับชีวิตประจำวันของมนุษย์ ซึ่งเมื่อวิเคราะห์ถึงปัญหาที่พบในธุรกิจนี้ แล้วเราจะพบว่า ปัญหาหลักจะเกี่ยวข้องกับการบริการ ซึ่งเป็นจุดสำคัญของร้านอาหาร การบริการของร้านอาหารส่วนที่สำคัญมี 2 ส่วน

##### 3.1.1.1 การรับรายการอาหารและเครื่องดื่ม

สำหรับรูปแบบการรับรายการอาหารที่เกิดขึ้น จะมีบริการทำการรับรายการอาหาร หลังจากนั้นบริกรจะทำการจกรายการอาหารตามที่ถูกคำสั่งด้วยมือลงในใบรายการสั่งอาหาร แล้วบริกรก็จะนำส่งต่อไปยังครัว และบาร์เครื่องดื่ม ถ้าร้านอาหารนั้นมีการแบ่งแยกครัว หัวหน้าพ่อครัวก็จะเป็นคนจำเนกงานต่าง ๆ ใ้ในแต่ละครัว ซึ่งการรับคำสั่งด้วยวิธีนี้อาจเกิดปัญหาต่างๆ ดังนี้

1. หัวหน้าพ่อครัวอ่านลายมือของบริกรไม่ออก จุดนี้อาจทำให้เกิดการทำรายการอาหารที่ผิดพลาดจกรายการที่สั่งไว้ ซึ่งทำให้เกิดความล่าช้าในการบริการอันเนื่องมาจากการทำอาหารไม่ตรงกับความต้องการของลูกค้า

2. เมื่อลูกค้าต้องการสั่งรายการอาหารเพิ่มเติม อาจจะต้องเสีเวลารอเป็นเวลานานกว่าที่จะได้สั่งรายการอาหารเพิ่มเติม ทำให้การบริการไม่เป็นที่น่าพอใจต่อลูกค้า

3. เมื่อลูกค้าต้องการยกเลิกรายการอาหารที่สั่งไป บริกรจะต้องเดินไปที่ห้องครัว เพื่อดูว่ารายการอาหารที่สั่งดังกล่าวได้ทำไปแล้วหรือยัง และเดินทางกลับมาบอกให้ลูกค้าทราบว่าสามารถยกเลิกรายการดังกล่าวได้หรือไม่ แทนที่ลูกค้าจะสามารถทราบผลได้ทันที

4. เมื่อร้านมีปริมาณลูกค้าจำนวนมากทำให้การบริการนั้นไม่ทั่วถึงและไม่รวดเร็ว เกิดความล่าช้า แต่ถ้าหากจ้างพนักงานเพิ่มก็จะทำให้เกิดค่าใช้จ่ายที่เพิ่มขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.เมื่อลูกค้าต้องการทราบส่วนประกอบอาหารของรายการอาหารนั้น บริกรอาจจะไม่ทราบถึงส่วนประกอบของรายการอาหารนั้น

6.การให้บริการต่างๆ ของบริกร จะเกี่ยวข้องกับครัว และพนักงานคิดเงิน (Cashier) ถ้าหากหาวิธีที่บริกร ไม่ต้องเดินไปที่ครัว ก็จะทำให้การบริการรวดเร็ว

7.ในกรณีที่รายการอาหารที่สั่งไม่ครบถ้วนตามที่ลูกค้าสั่ง บริกรจะไม่สามารถทราบได้เลยถ้าไม่มีการทักท้วงจากลูกค้า

### 3.1.1.2 การชำระเงิน

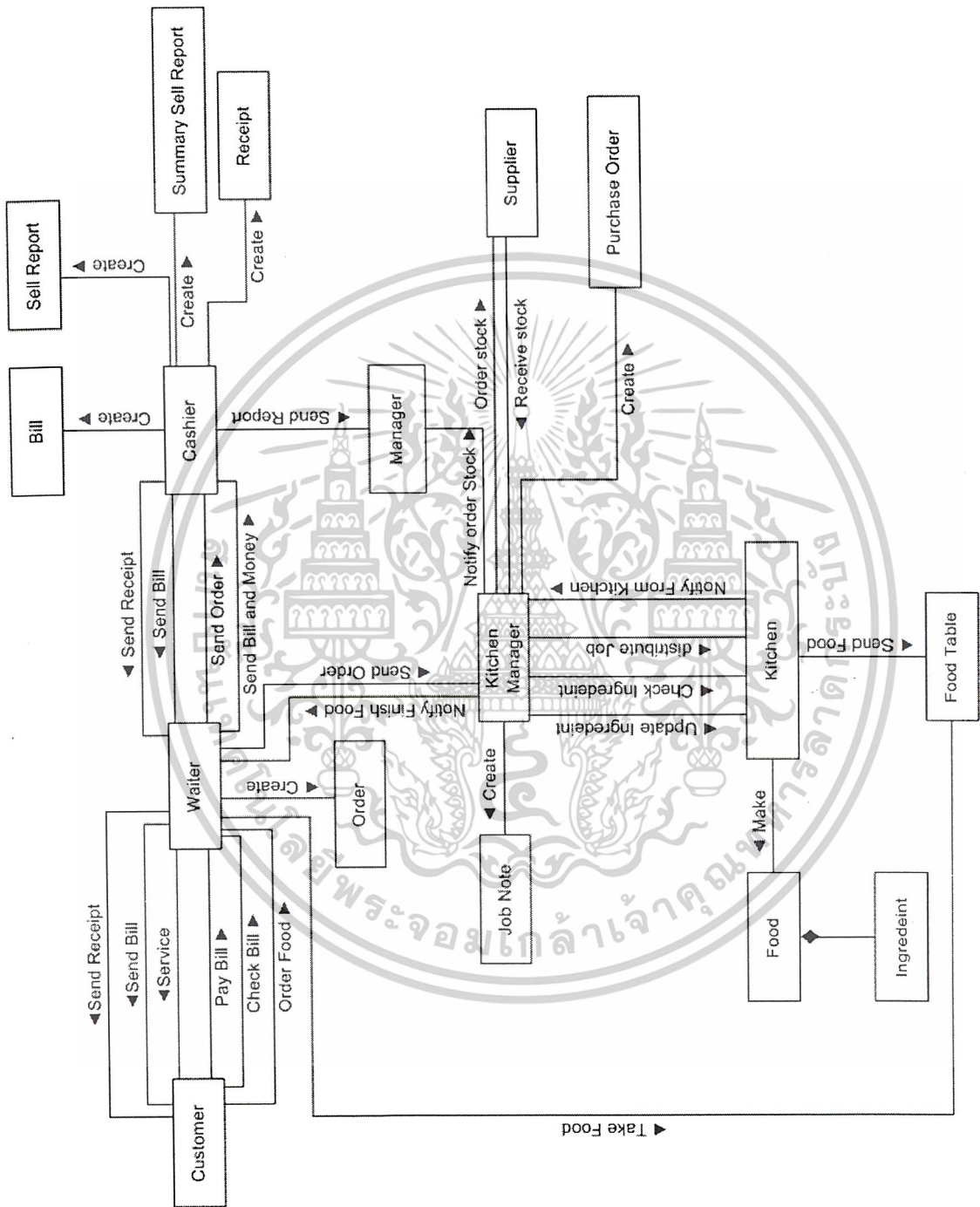
ในส่วนของการชำระเงินนั้น มีลำดับขั้นตอน ดังนี้

- 1.ลูกค้าจะทำการเรียกบริกรเพื่อชำระเงิน
- 2.บริกรทำการแจ้งแก่พนักงานคิดเงินให้ทราบว่าลูกค้าโต๊ะใดต้องการชำระเงิน
- 3.ลูกค้าตรวจสอบรายการอาหารว่าครบถ้วนหรือไม่ ถ้าไม่ครบถ้วนก็ทักท้วง หลังจากนั้นก็ทำการชำระเงิน
- 4.บริกรนำส่งเงินมาให้แก่พนักงานคิดเงิน
- 5.บริกรนำเงินทอนมาให้แก่ลูกค้า

จากขั้นตอนที่เกิดขึ้นต่างๆ จะทำให้เกิดความล่าช้า และมีเรื่องการเดินทางไปกลับของบริกร ดังนั้น เมื่อลูกค้าต้องการชำระเงิน พนักงานคิดเงินก็ควรที่จะทราบทันที พร้อมกับแจ้งใบรายการทั้งหมดให้แก่ลูกค้าได้ทันที จะทำให้บริการเป็นไปอย่างรวดเร็วและสะดวกมากขึ้น นอกจากนี้สำหรับในส่วนเรื่องการบันทึกรายการชำระเงินนั้น โดยมากแล้วจะเป็นเพียงการจดบันทึก ซึ่งการทำลักษณะนี้อาจจะทำให้เกิดความผิดพลาดทางด้านข้อมูลขึ้น

ดังนั้นหากพัฒนา โปรแกรมและอุปกรณ์ซึ่งช่วยลดปัญหาดังกล่าวได้ จึงคาดว่า น่าจะได้รับความสนใจจากผู้ประกอบการร้านอาหาร

### 3.1.4 โครงสร้างโดยทั่วไปของระบบ

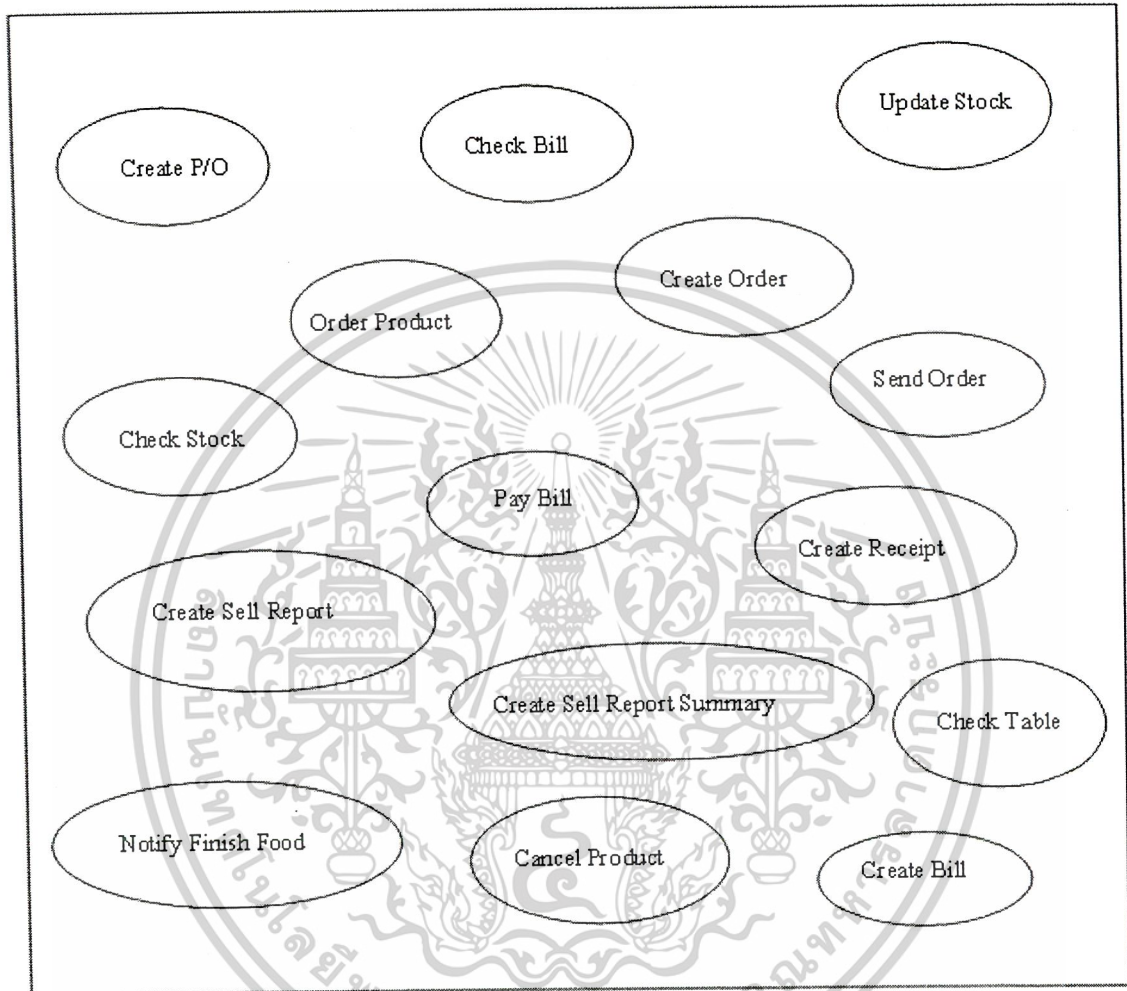


รูปที่ 3.1 โดเมนโมเดล (Domain Model)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2 การออกแบบระบบ

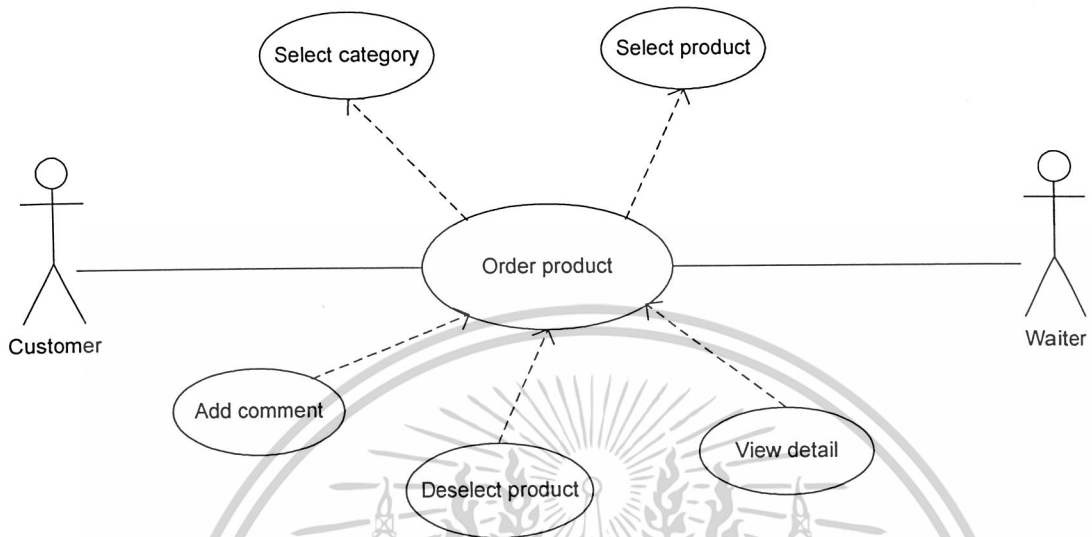
#### 3.2.1 ไส้เลเวลยูสเคสไดอะแกรมของระบบ (High level Use Case Diagram)



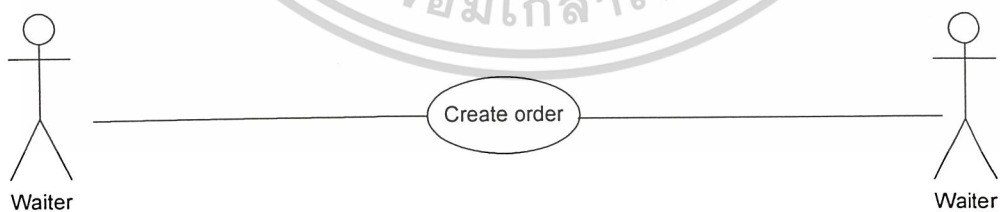
รูปที่ 3.2 แสดงไส้เลเวลยูสเคส (High level Use Case)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.2 ยูสเคสไดอะแกรม (Use Case Diagram)

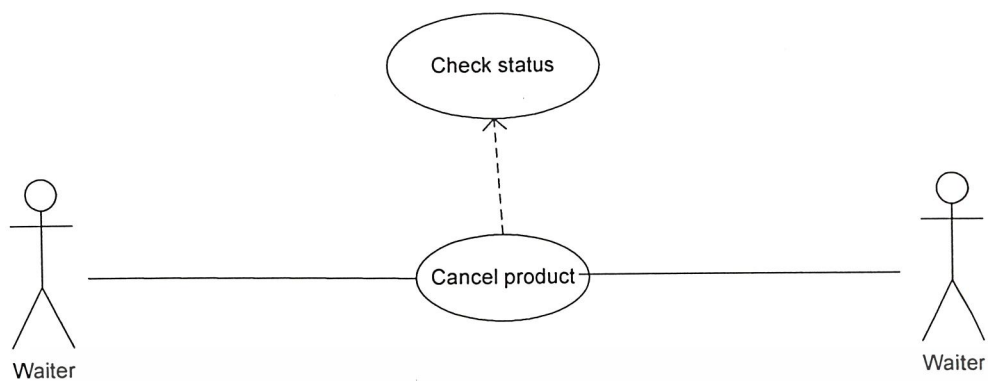


รูปที่ 3.3 ยูสเคสไดอะแกรมสำหรับการจัดการเกี่ยวรายการที่สั่งสินค้า(Order product)

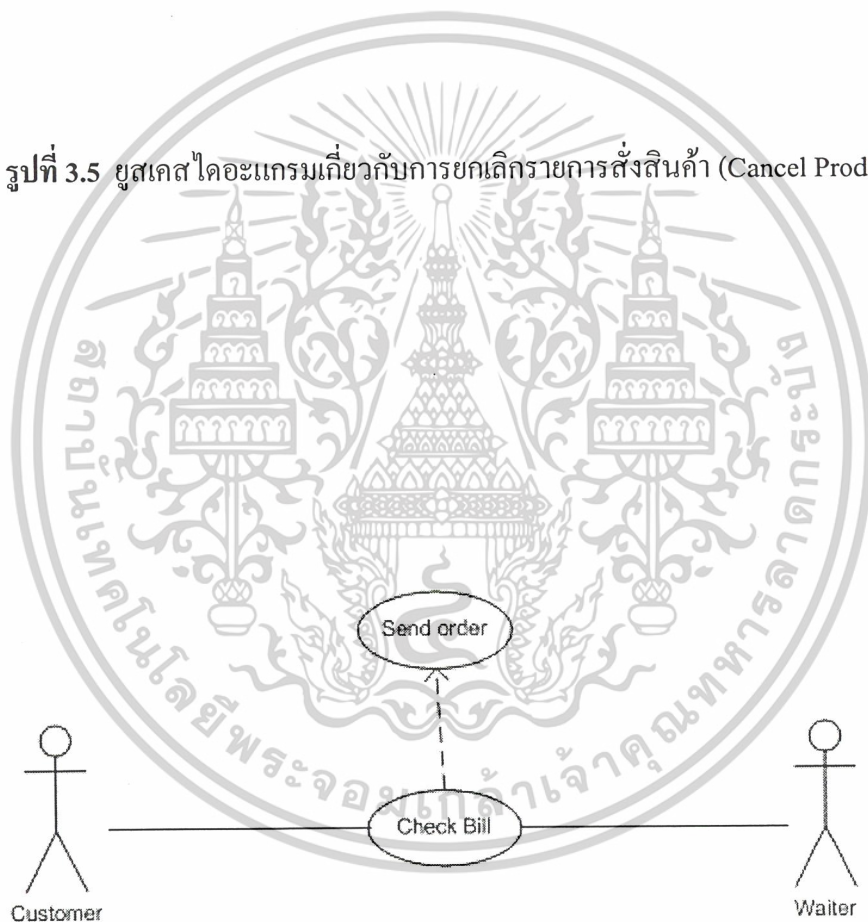


รูปที่ 3.4 ยูสเคสไดอะแกรมการจัดการเกี่ยวกับการสร้างรายการที่สั่งสินค้า (Create Order)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

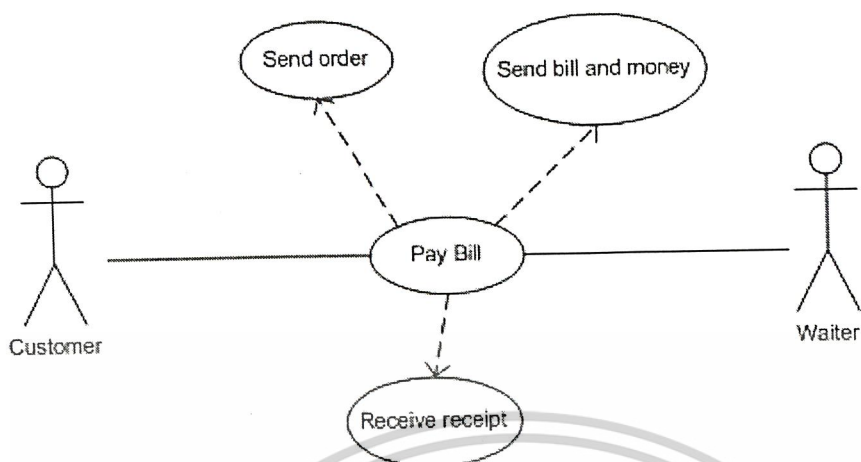


รูปที่ 3.5 ยูสเคสไดอะแกรมเกี่ยวกับการยกเลิกรายการสั่งสินค้า (Cancel Product)

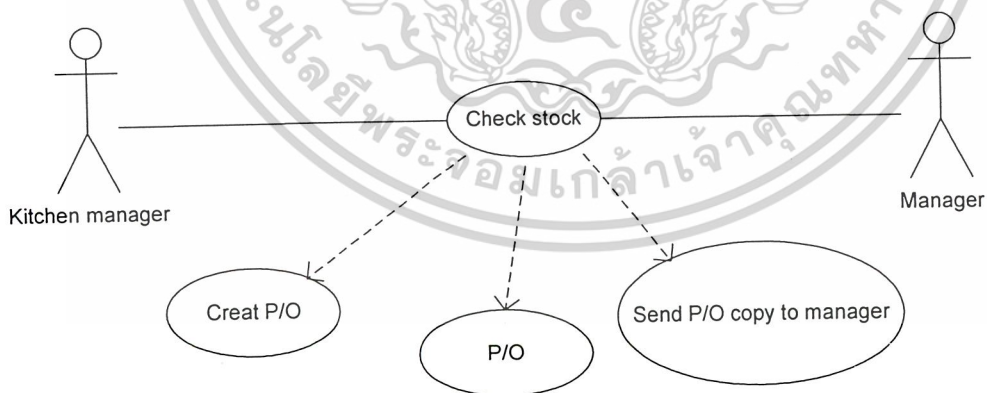


รูปที่ 3.6 ยูสเคสไดอะแกรมสำหรับการเรียกเก็บเงิน (Check Bill)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

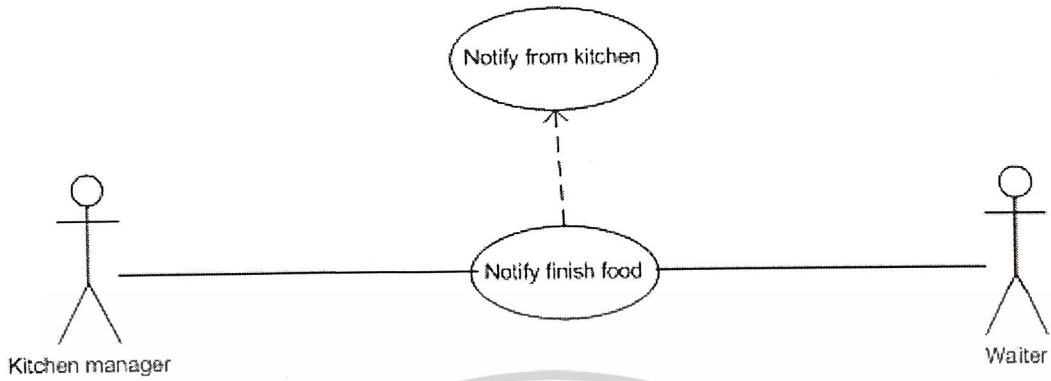


รูปที่ 3.7 ยูสเคสไดอะแกรมสำหรับการจ่ายเงิน (Pay Bill)

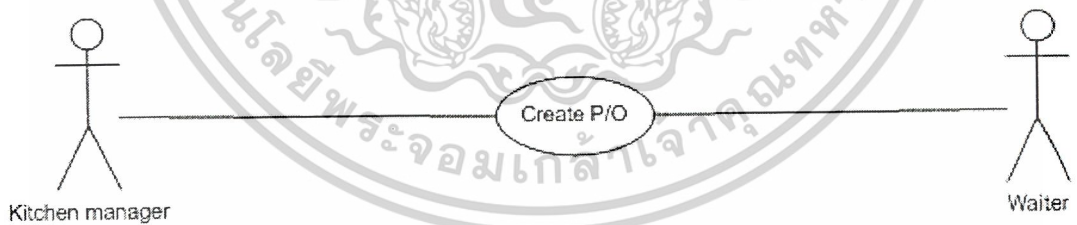


รูปที่ 3.8 ยูสเคสไดอะแกรมสำหรับการตรวจสอบสต็อกสินค้า (Check Stock)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

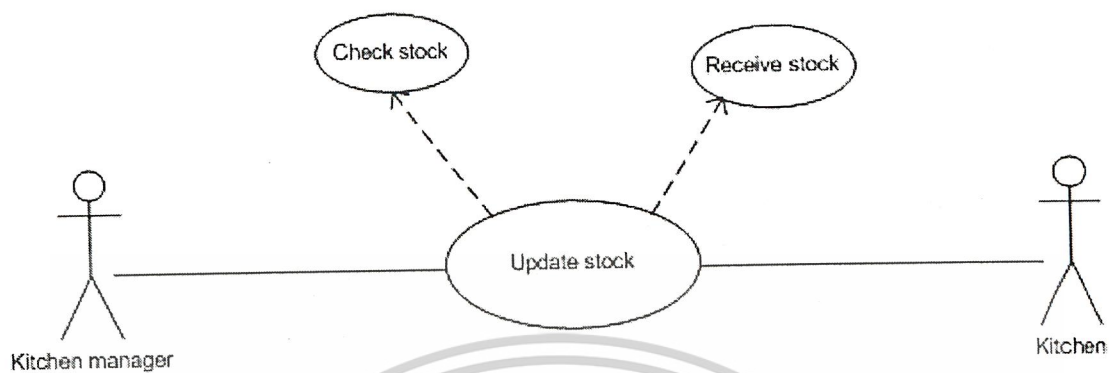


รูปที่ 3.9 ยูสเคสไดอะแกรมเกี่ยวกับการแจ้งการทำอาหารเสร็จ (Notify Finish Food)

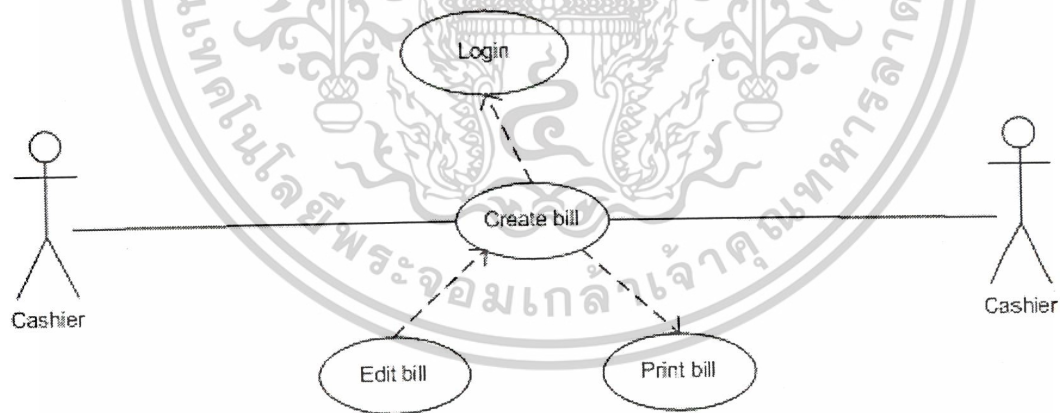


รูปที่ 3.10 ยูสเคสไดอะแกรมการออกใบรายการจ่ายเงินค่าอาหาร (Create P/O)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 ยูสเคสไดอะแกรมการปรับปรุงข้อมูลสต็อกสินค้า (Update Stock)

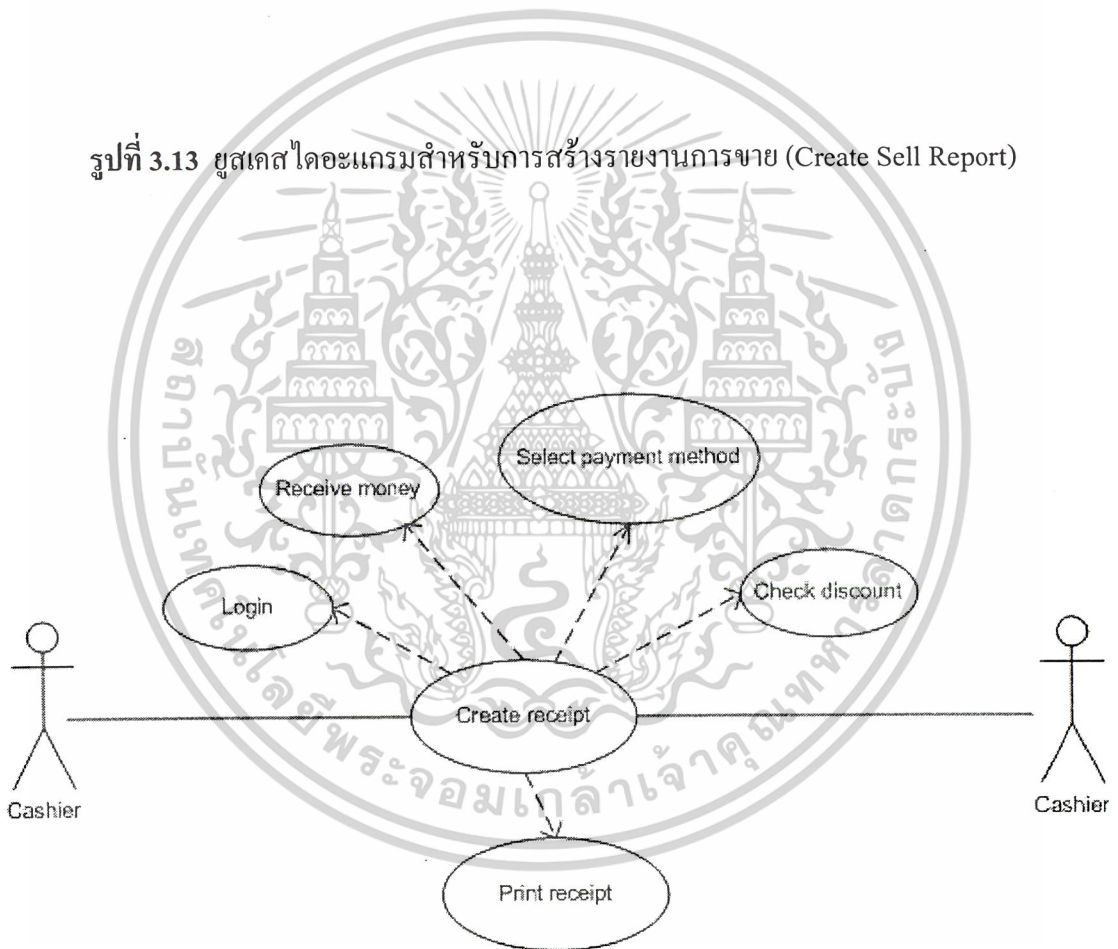


รูปที่ 3.12 ยูสเคสไดอะแกรมการออกใบเรียกเก็บเงิน (Create Bill)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

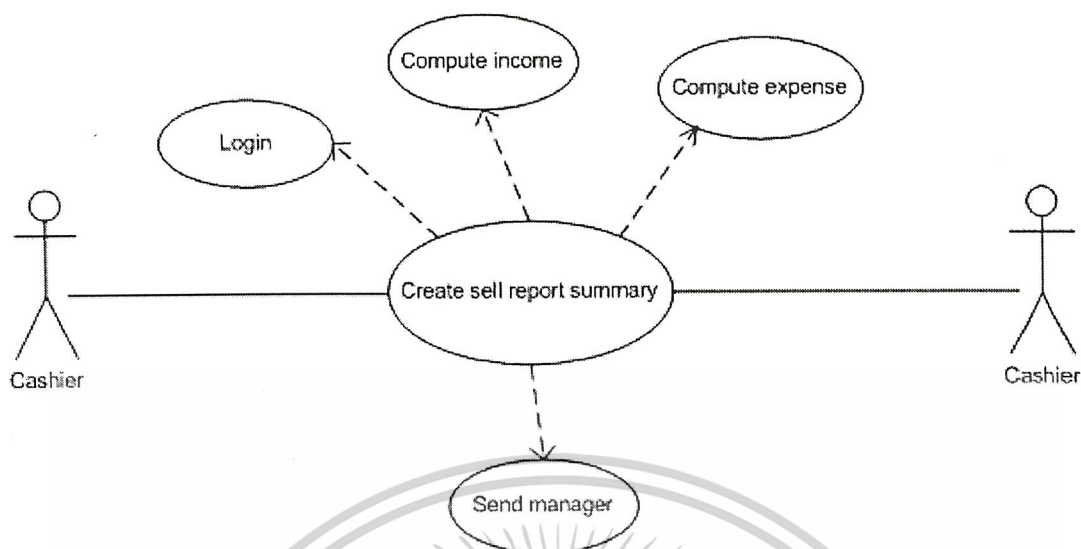


รูปที่ 3.13 ยูสเคสไดอะแกรมสำหรับการสร้างรายงานการขาย (Create Sell Report)

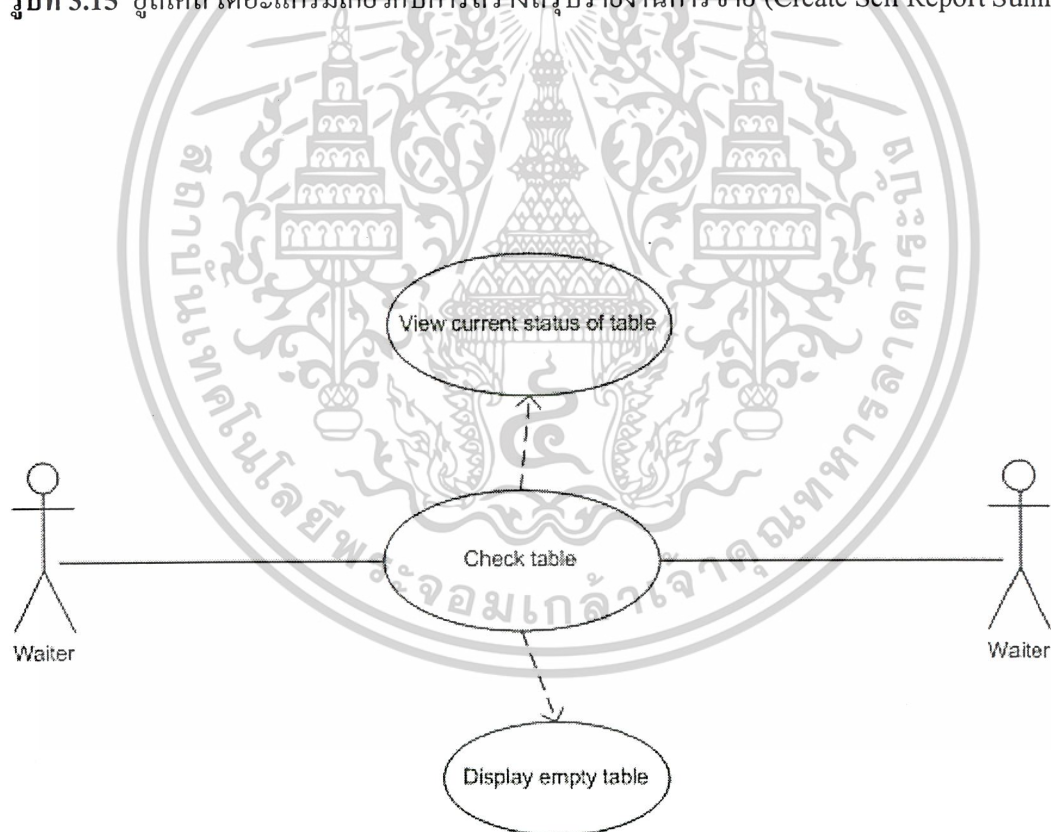


รูปที่ 3.14 ยูสเคสไดอะแกรมเกี่ยวกับการสร้างใบเสร็จรับเงิน (Create Receipt)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

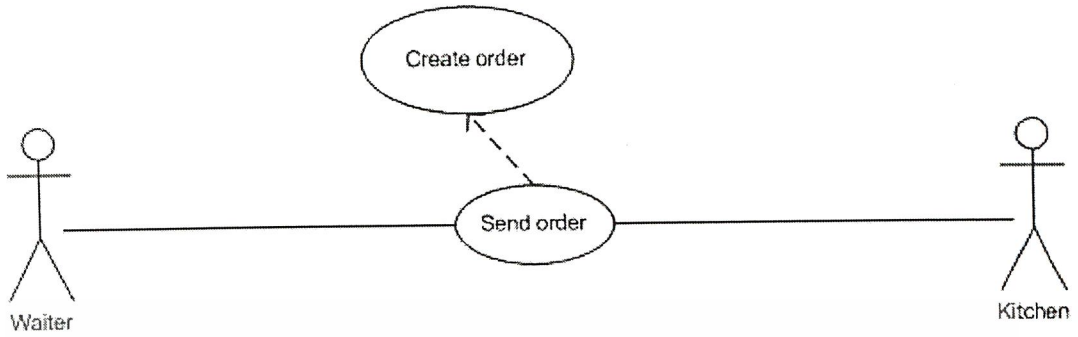


รูปที่ 3.15 ยูสเคสไดอะแกรมเกี่ยวกับการสร้างสรุปรายงานการขาย (Create Sell Report Summary)

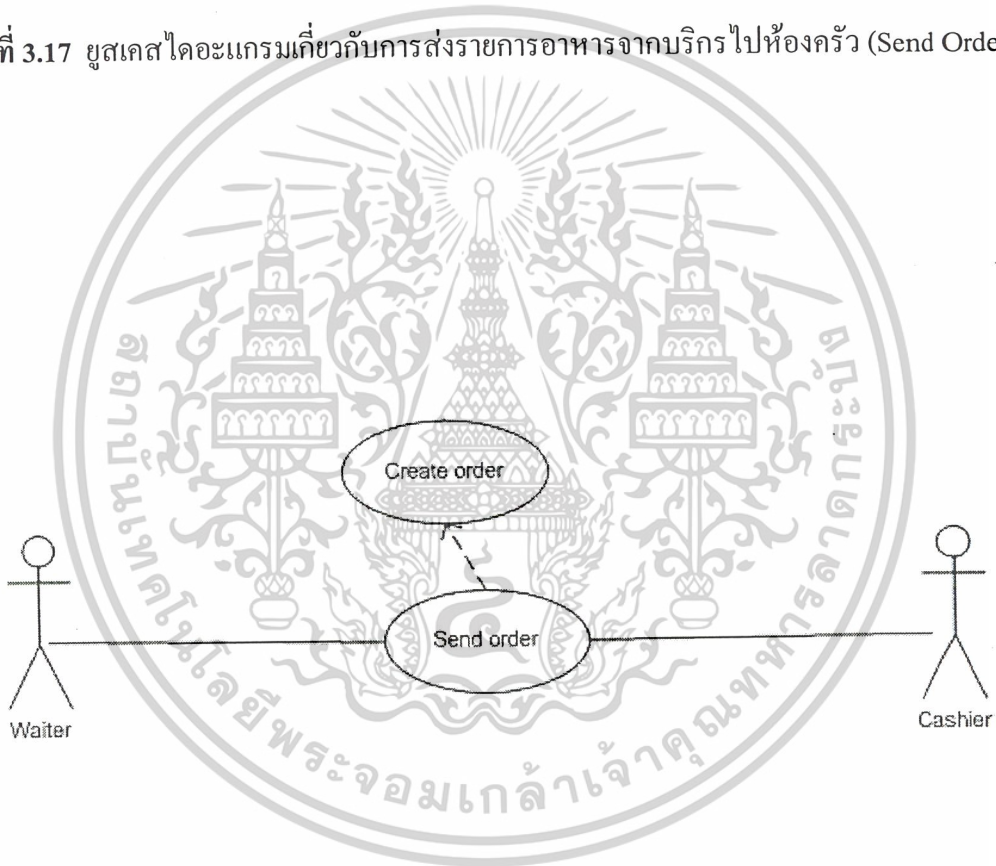


รูปที่ 3.16 ยูสเคสไดอะแกรมสำหรับการตรวจสอบสถานะของโต๊ะ (Check Table)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



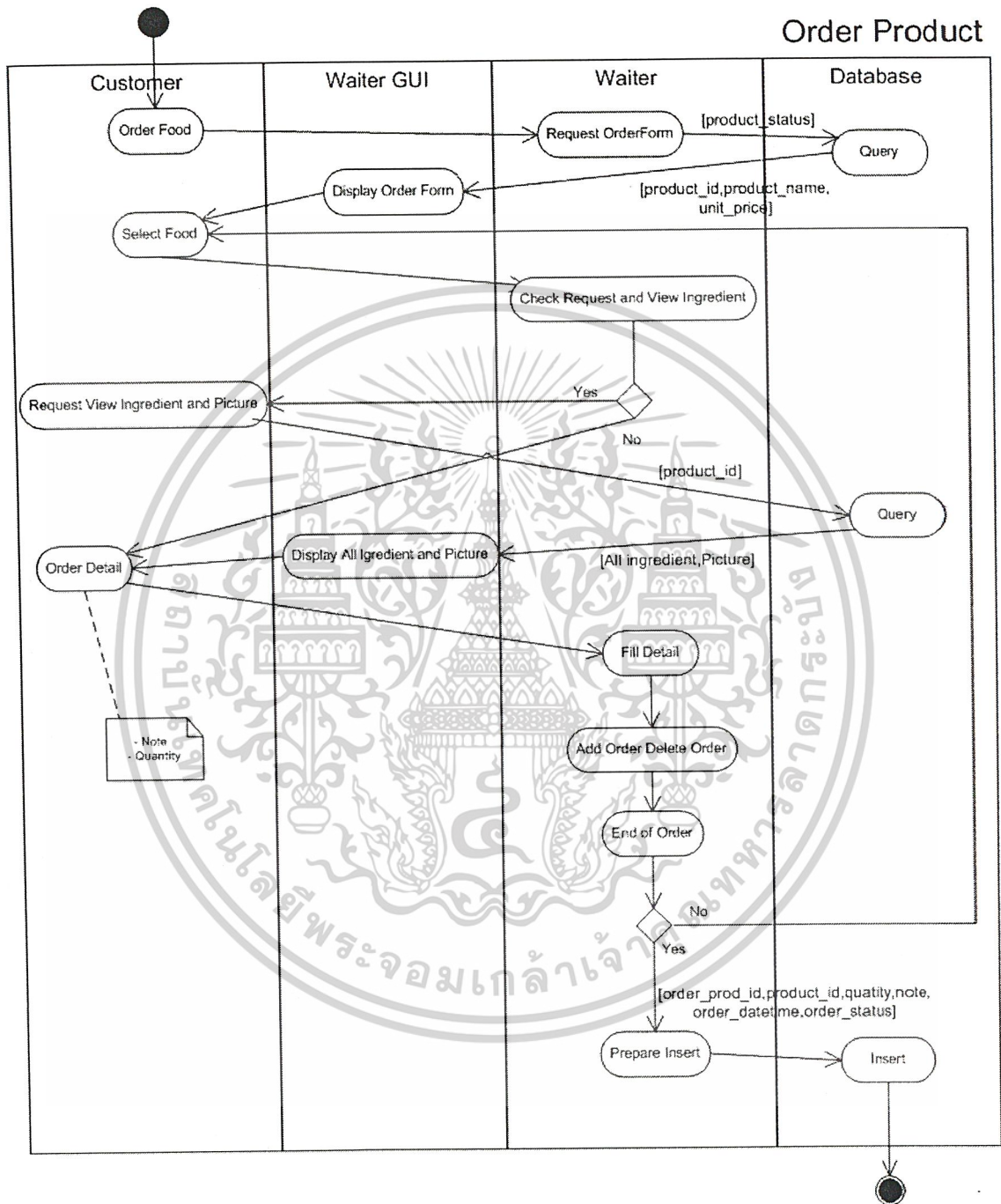
รูปที่ 3.17 ยูสเคสไดอะแกรมเกี่ยวกับการส่งรายการอาหารจากบริกร ไปห้องครัว (Send Order)



รูปที่ 3.18 ยูสเคสไดอะแกรมเกี่ยวกับการส่งรายการอาหารจากบริกรไปยังพนักงานคิดเงิน (Send Order)

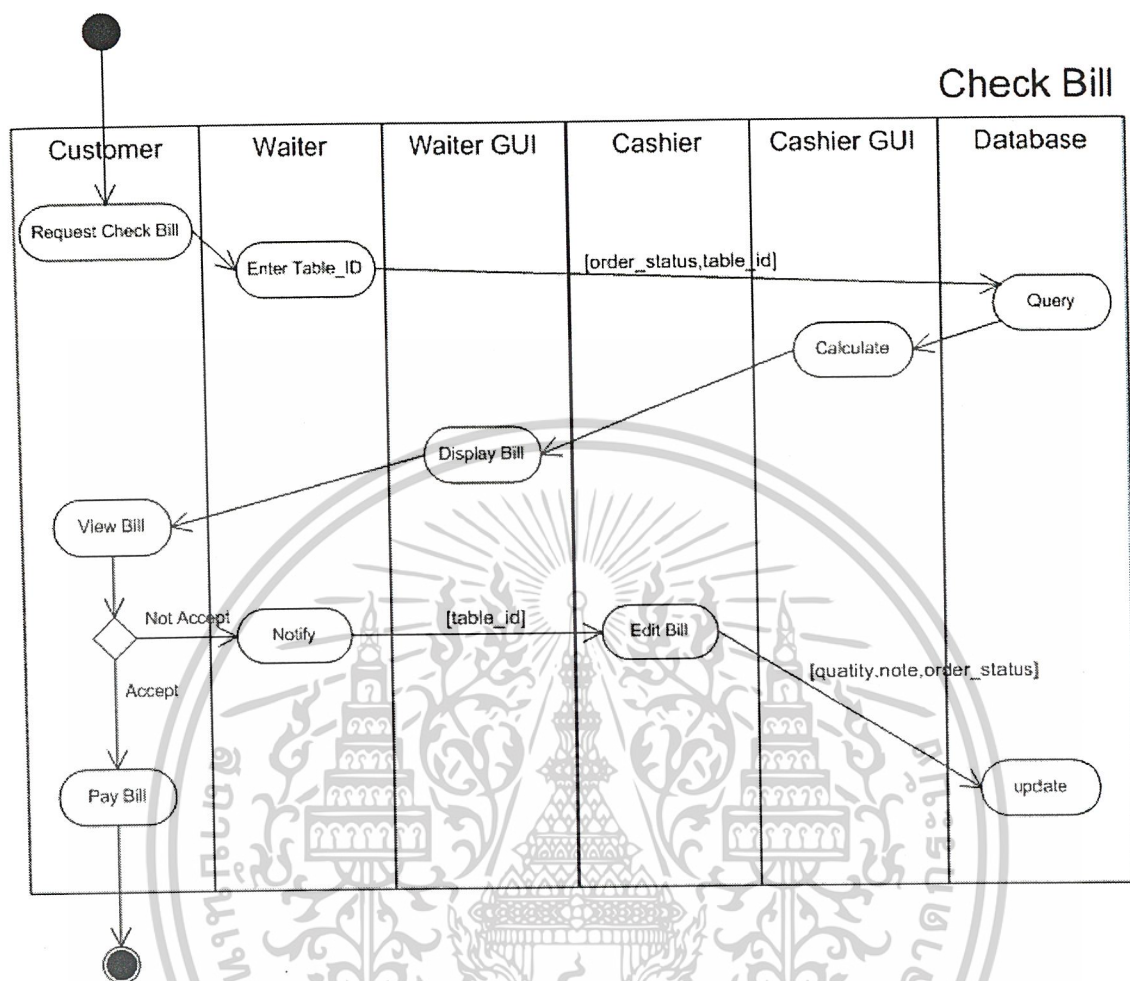
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.3 แอกทิวิตีไดอะแกรม (Activity Diagram)



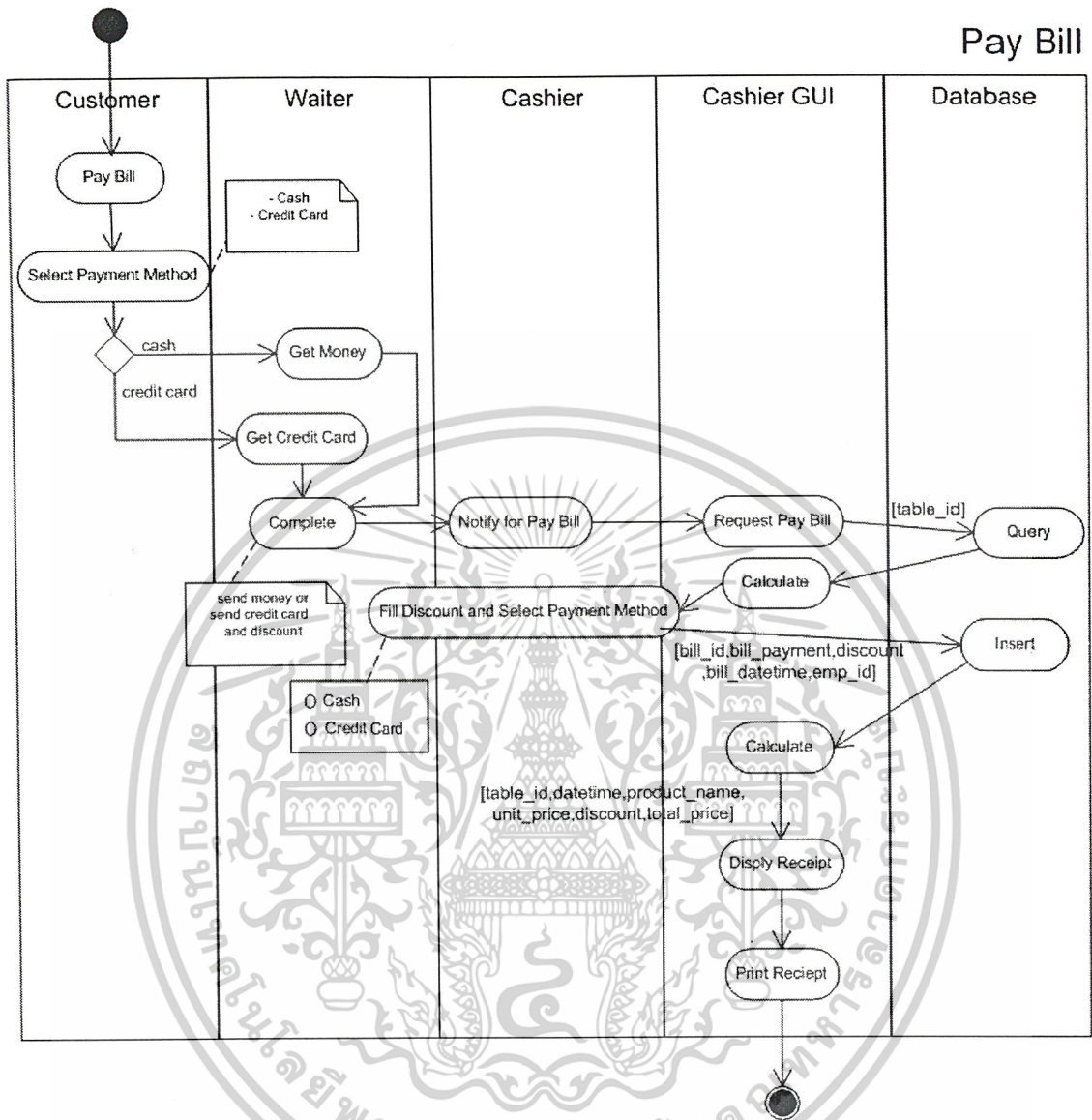
รูปที่ 3.19 แอกทิวิตีไดอะแกรมสำหรับการสั่งสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



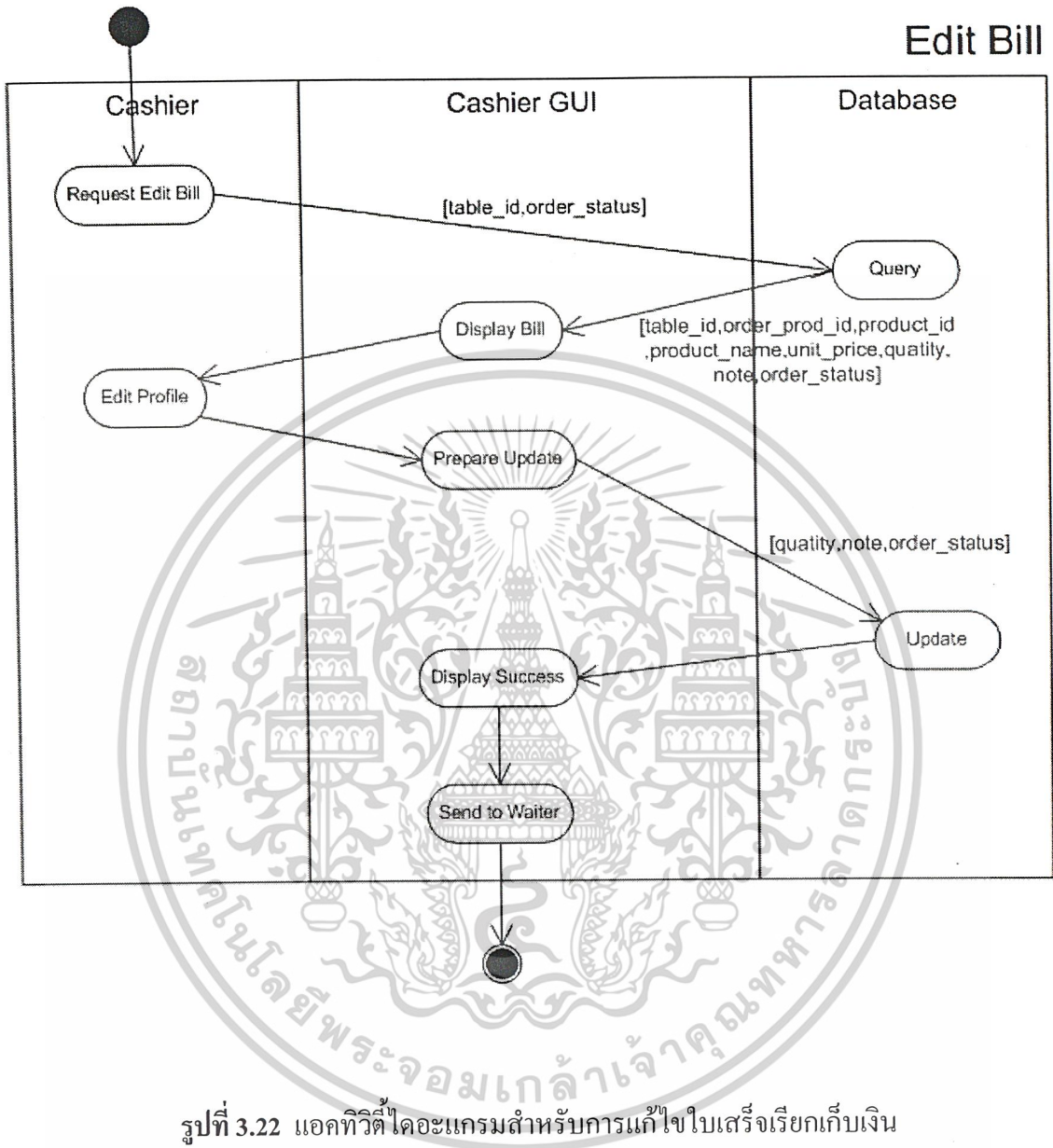
รูปที่ 3.20 แลคทิวทัศน์โคอะแกรมสำหรับการเรียกเก็บเงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

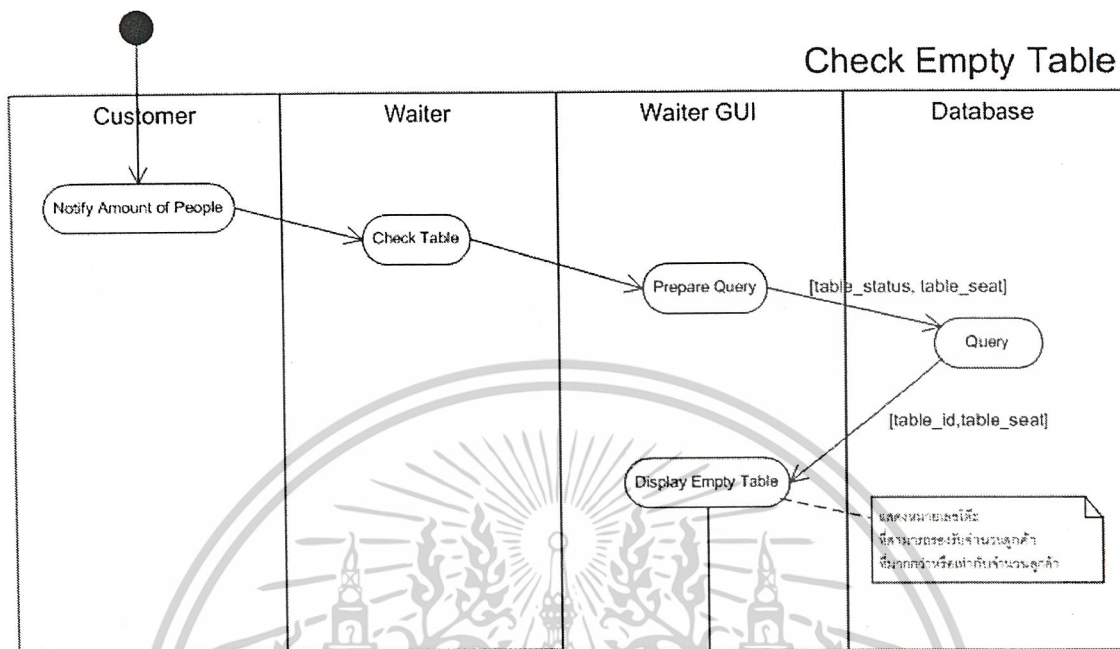


รูปที่ 3.21 แอคทีวิตีไดอะแกรมสำหรับการจ่ายเงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

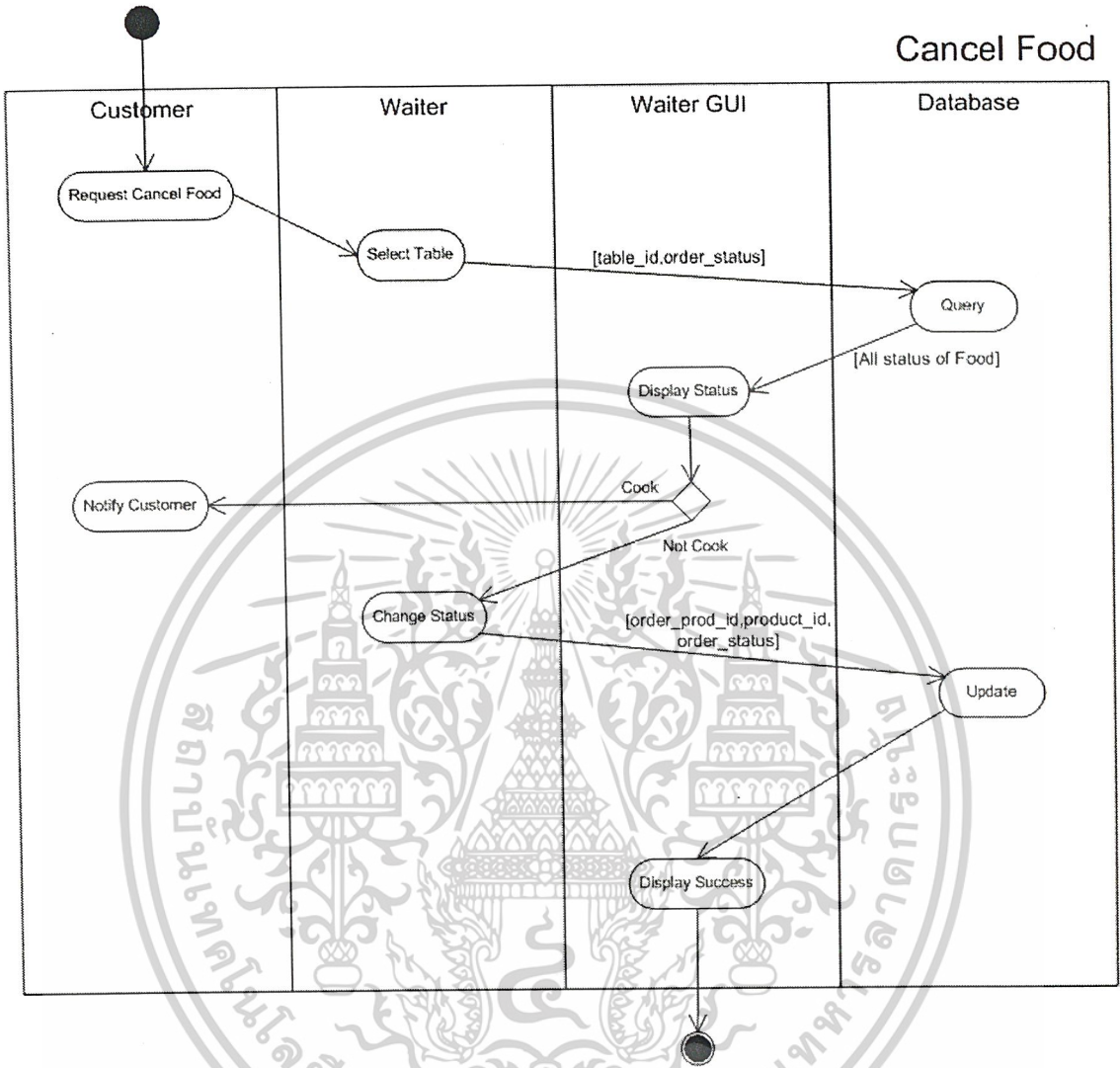


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



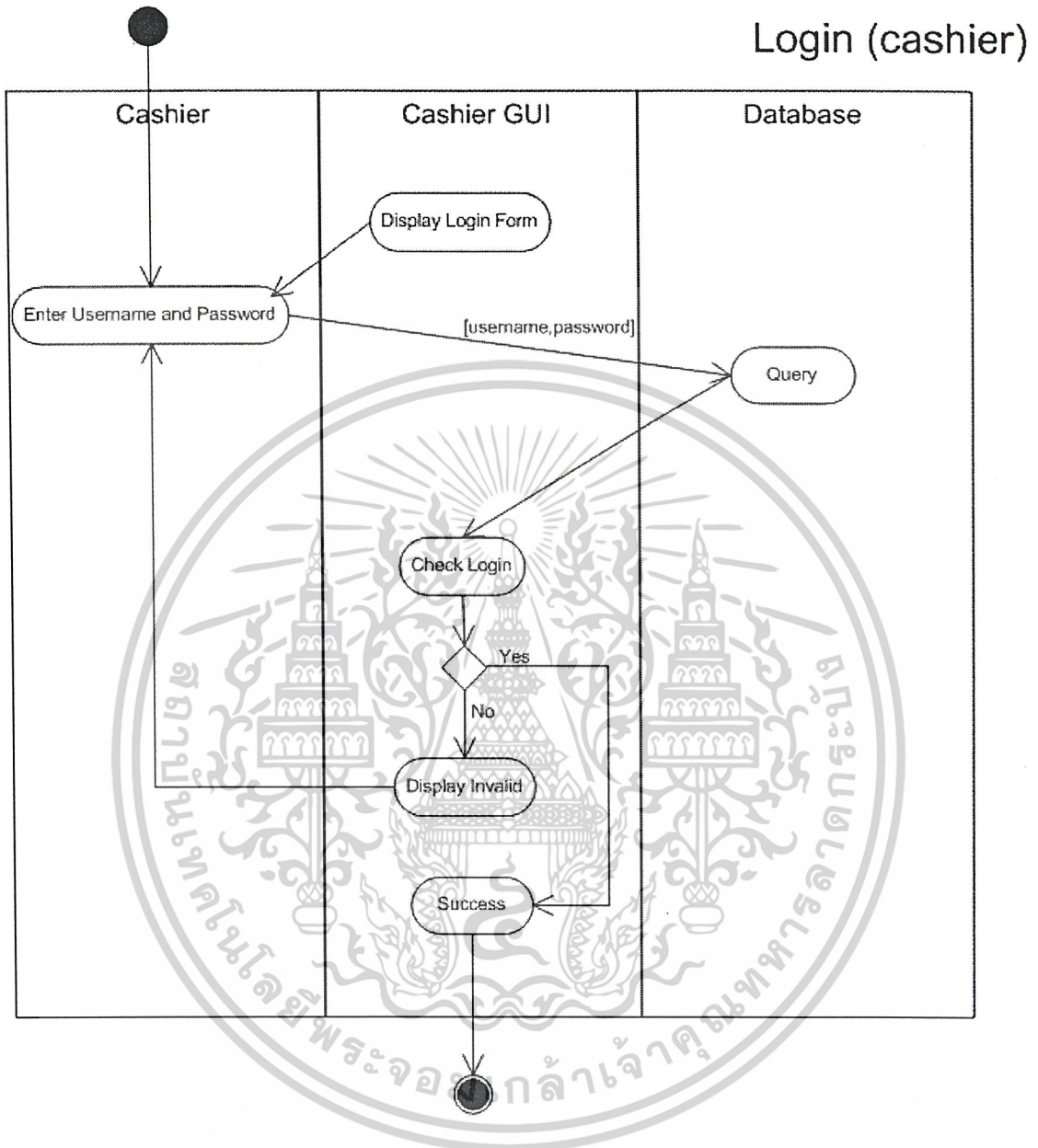
รูปที่ 3.23 แอคทีวิตี้ไดอะแกรมสำหรับการตรวจสอบโต๊ะว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



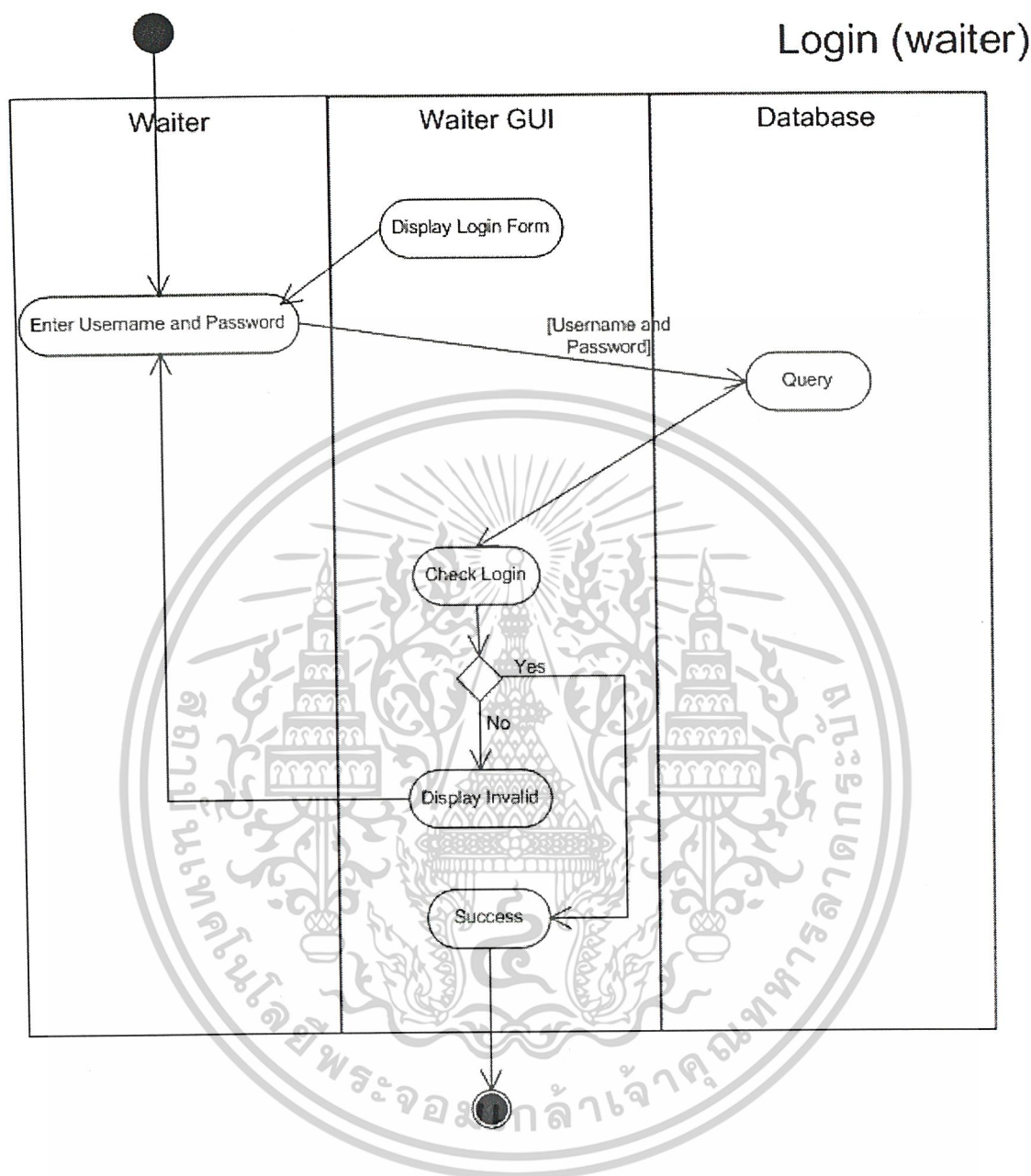
รูปที่ 3.24 แอคทิวิตีไดอะแกรมสำหรับการยกเลิกรายการอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



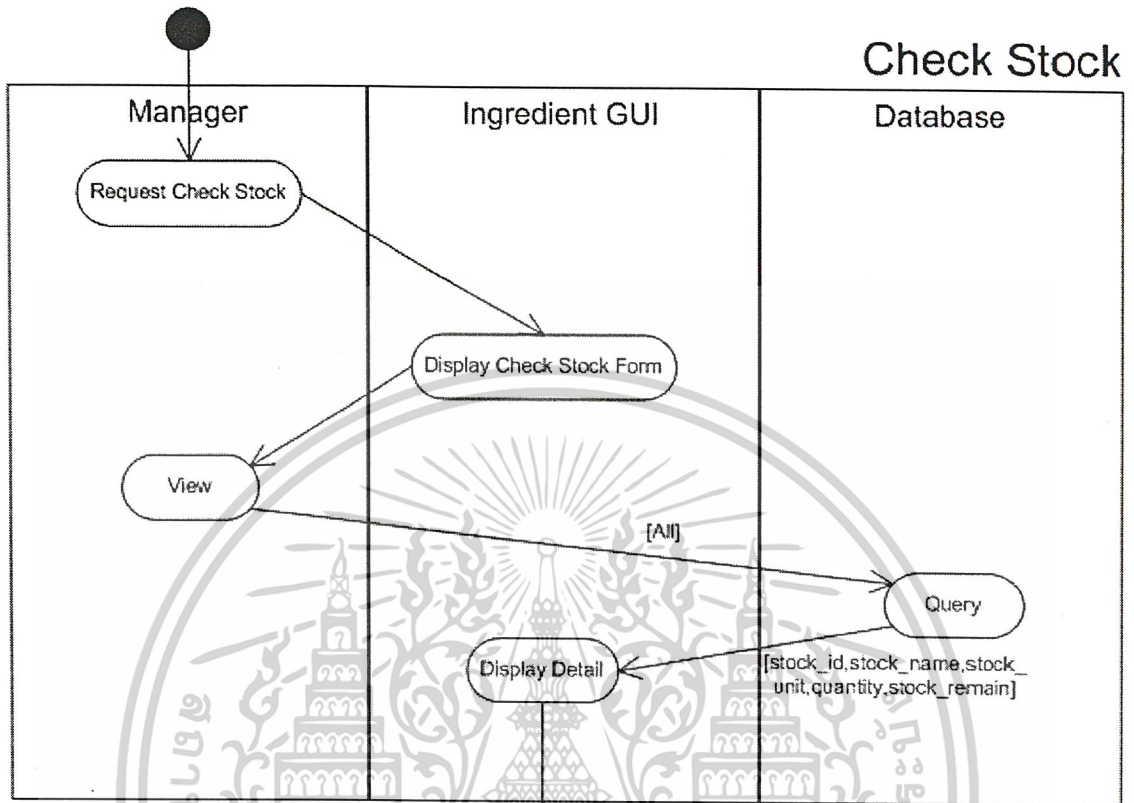
รูปที่ 3.25 แอคทิวิตีไดอะแกรมสำหรับการล็อกอินของพนักงานคิดเงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



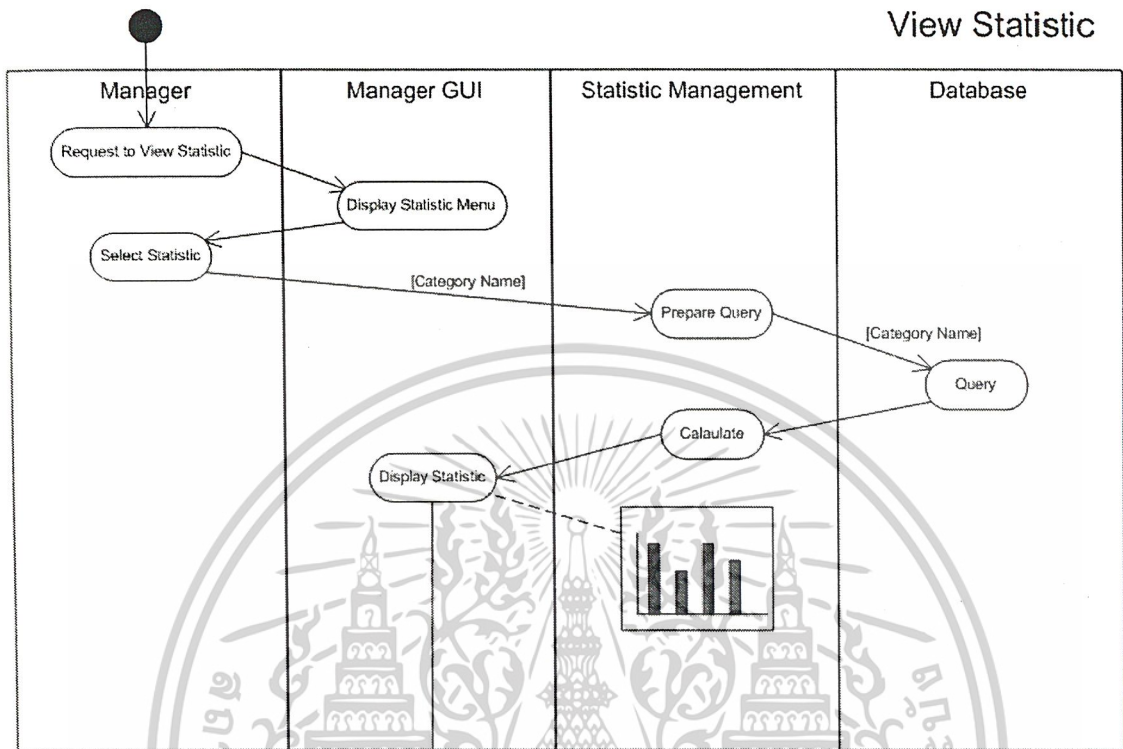
รูปที่ 3.26 แอกทिवิตีไดอะแกรมสำหรับการล็อกอินของบริกร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



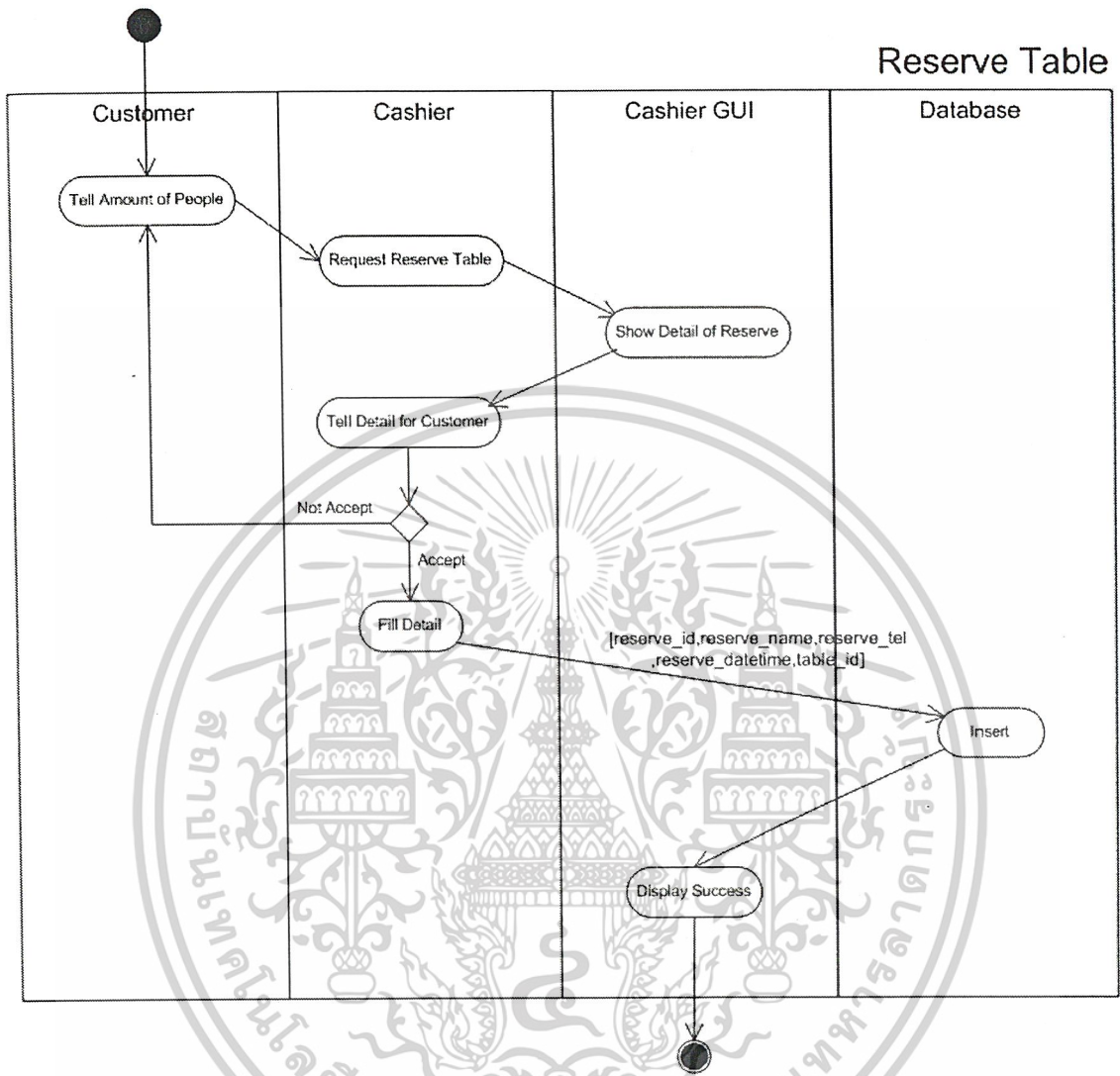
รูปที่ 3.27 แอลทิวิตี้ไดอะแกรมสำหรับการตรวจสอบวัตถุดิบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



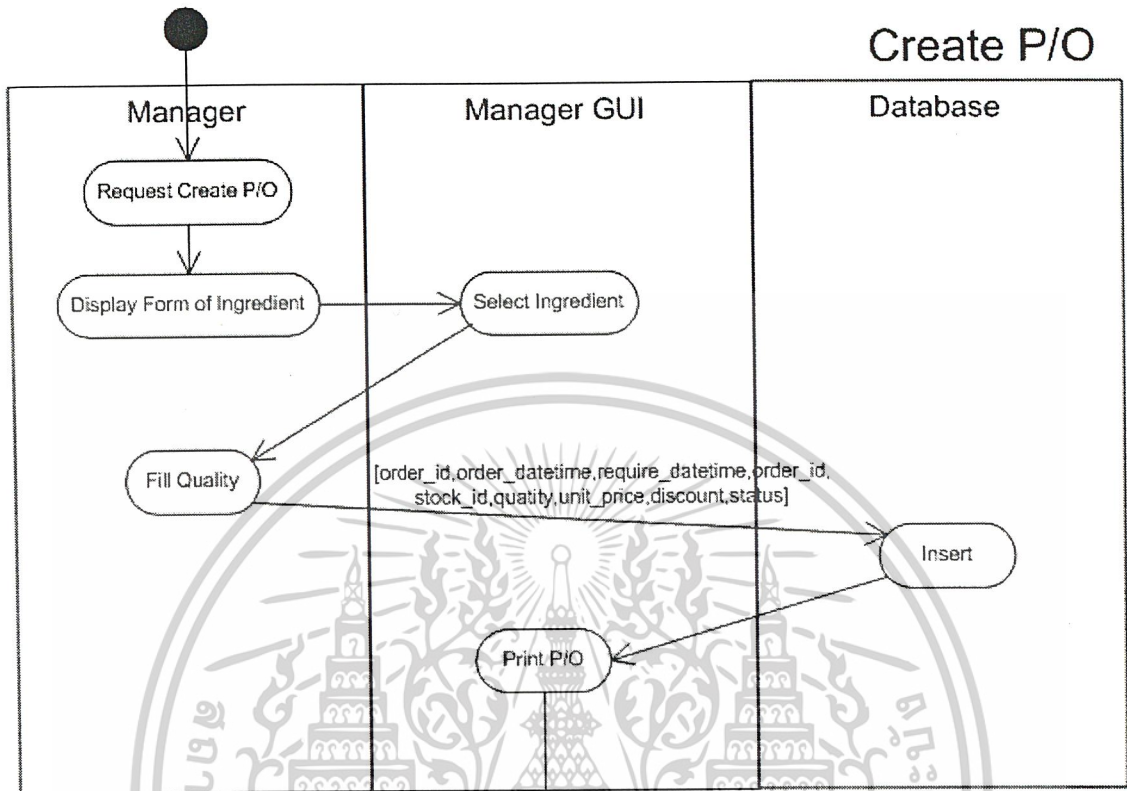
รูปที่ 3.28 แอกทิวิตีไดอะแกรมสำหรับการดูข้อมูลทางสถิติของร้าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



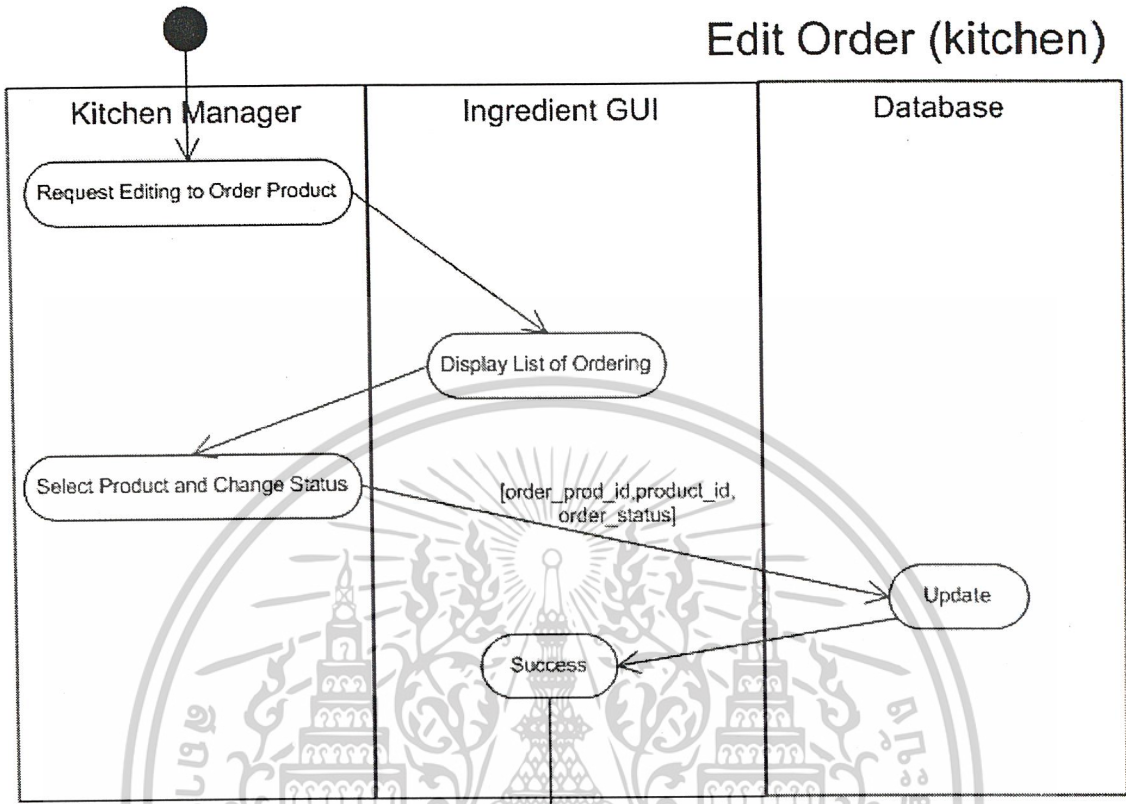
รูปที่ 3.29 แอคทิวิตีไดอะแกรมสำหรับการจองโต๊ะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



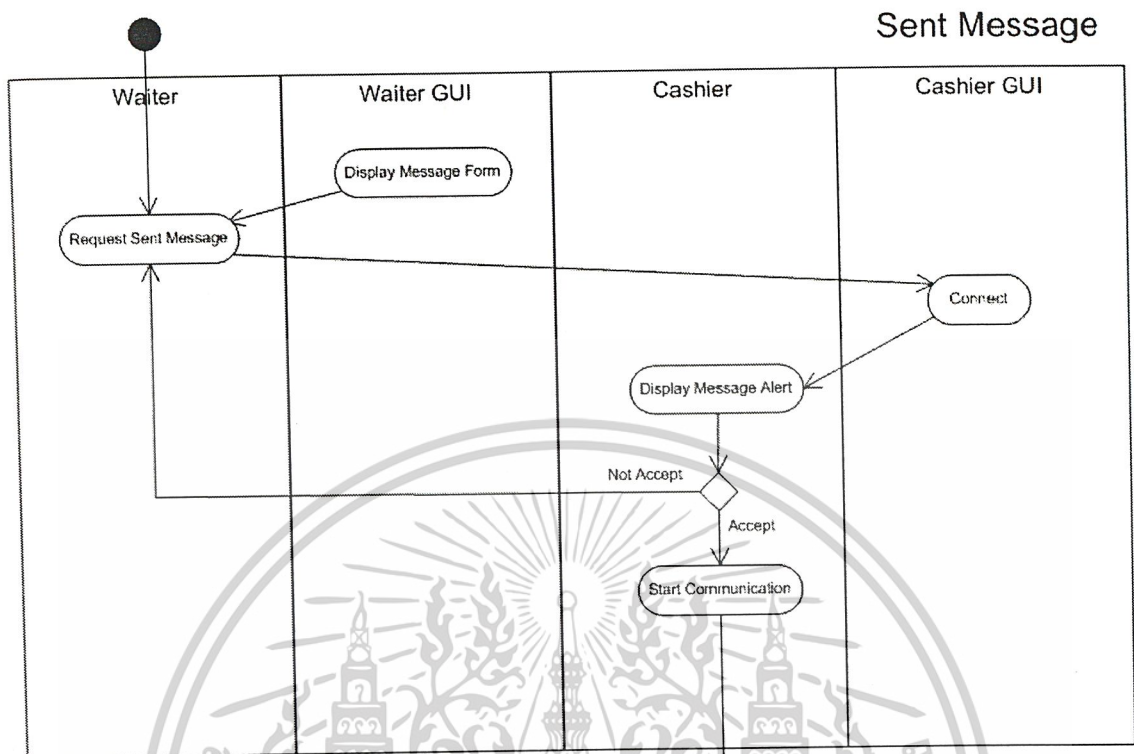
รูปที่ 3.30 แอกทวิตไดอะแกรมสำหรับการสร้างใบรายการสั่งซื้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.31 แอคทีวิตีไดอะแกรมสำหรับการแก้ไขรายการสั่งอาหาร

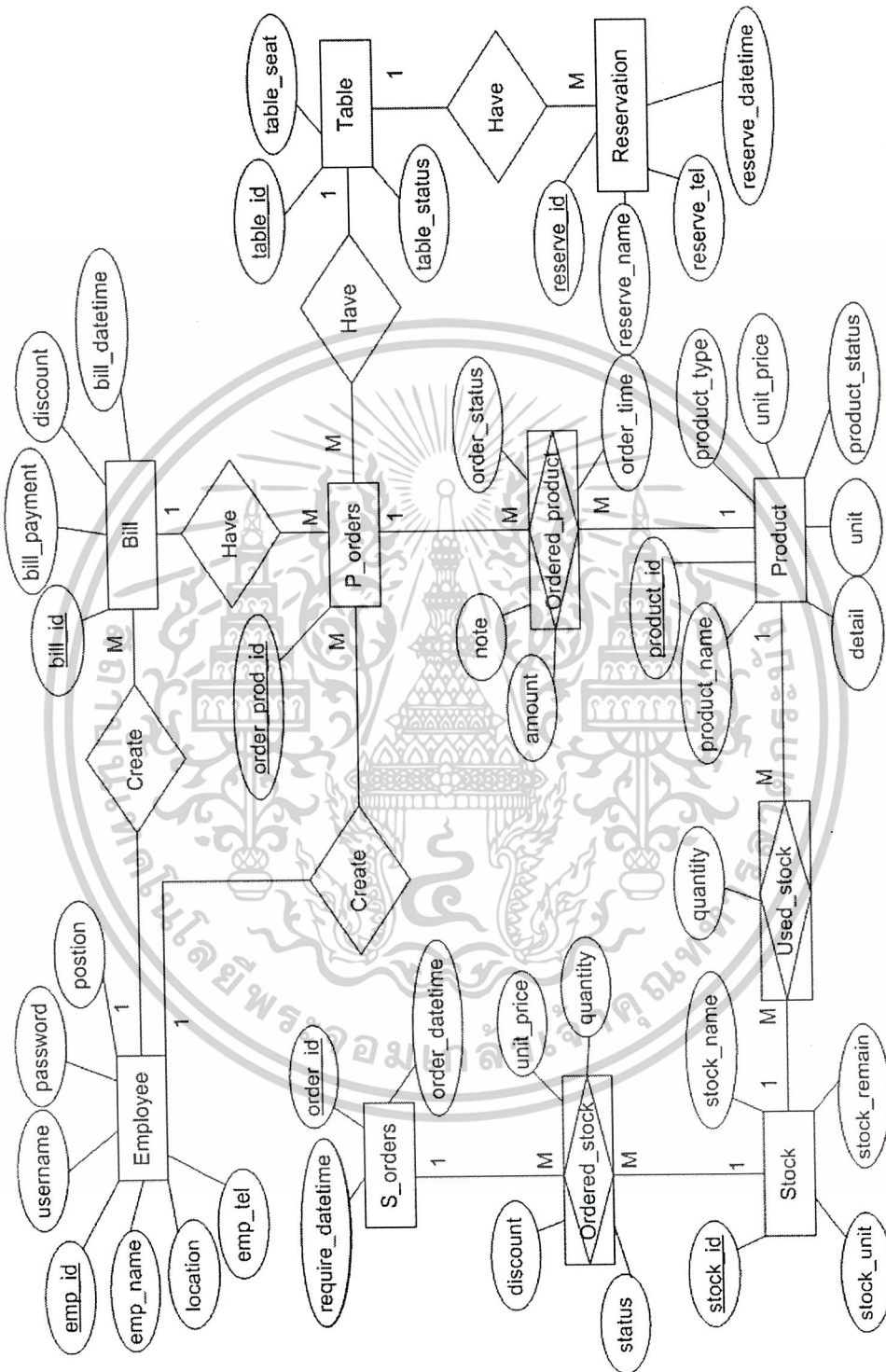
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.32 แอคทีวิตี้ไดอะแกรมสำหรับการส่งข้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.4 การออกแบบฐานข้อมูล (Database Design)



รูปที่ 3.33 อี-อาร์ โมเดล (E-R Model) ของระบบฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.5 ดาต้าดิคชันนารี (Data Dictionary)

ตารางที่ 3.1 ตารางข้อมูลพนักงาน (Employee)

Name	Type	Key	Null	Description
emp_id	nvarchar(3)	PK	NO	รหัสพนักงาน
emp_name	nvarchar(100)		NO	ชื่อพนักงาน
location	nvarchar(100)		NO	ที่อยู่
emp_tel	nvarchar(100)		YES	เบอร์โทรศัพท์
username	nvarchar(100)		YES	ชื่อที่ใช้ในการเข้าระบบ
password	nvarchar(8)		YES	รหัสผ่าน
post	nvarchar(100)		NO	ตำแหน่ง

ตารางที่ 3.2 ตารางข้อมูลสินค้า (Product)

Name	Type	Key	Null	Description
product_id	nvarchar(4)	PK	NO	รหัสสินค้า
product_name	nvarchar(250)		NO	ชื่อสินค้า
product_type	nvarchar(100)		NO	ประเภทสินค้า
unit_price	float(8)		NO	ราคาต่อหน่วย
unit	nvarchar(20)		NO	หน่วยนับ
detail	nvarchar(300)		YES	รายละเอียด
product_status	nvarchar(20)		NO	สถานะของสินค้า

ตารางที่ 3.3 ตารางข้อมูลโต๊ะ (Table)

Name	Type	Key	Null	Description
table_id	nvarchar(3)	PK	NO	รหัสโต๊ะ
table_seat	int(4)		NO	จำนวนที่นั่งในโต๊ะ
table_status	nvarchar(1)		NO	สถานะของโต๊ะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.4 ตารางข้อมูลวัตถุดิบ (Stock)

Name	Type	Key	Null	Description
stock_id	nvarchar(4)	PK	NO	รหัสวัตถุดิบ
stock_name	nvarchar(250)		NO	ชื่อวัตถุดิบ
stock_unit	nvarchar(20)		NO	หน่วยนับ
stock_remain	int(4)		YES	จำนวนคงเหลือ

ตารางที่ 3.5 ตารางข้อมูลรายการที่สั่งซื้อวัตถุดิบ (Ordered\_stock)

Name	Type	Key	Null	Description
order_id	nvarchar(8)	PK	NO	รหัสใบสั่งซื้อวัตถุดิบ
stock_id	nvarchar(4)	PK	NO	รหัสวัตถุดิบ
unit_price	float(8)		NO	ราคาต่อหน่วย
quantity	int(4)		YES	ปริมาณที่สั่งซื้อ
discount	float(8)		YES	ส่วนลด
status	nvarchar(10)		YES	สถานะของสินค้า

ตารางที่ 3.6 ตารางข้อมูลใบสั่งซื้อ (S\_orders)

Name	Type	Key	Null	Description
order_id	nvarchar(8)	PK	NO	รหัสใบสั่งซื้อ
require_datetime	datetime		NO	วันที่ต้องการวัตถุดิบ
order_datetime	datetime		NO	วันที่สั่งวัตถุดิบ

ตารางที่ 3.7 ตารางข้อมูลใบสั่งอาหาร (P\_orders)

Name	Type	Key	Null	Description
order_prod_id	nvarchar(8)	PK	NO	รหัสใบสั่งสินค้า
table_id	nvarchar(3)	FK	NO	รหัสโต๊ะ
bill_id	nvarchar(16)	FK	NO	รหัสใบเสร็จเก็บเงิน
emp_id	nvarchar(3)	FK	NO	รหัสพนักงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.8 ตารางข้อมูลรายการที่ใช้วัตถุดิบ (Used\_stock)

Name	Type	Key	Null	Description
stock_id	nvarchar(4)	PK	NO	รหัสวัตถุดิบ
product_id	nvarchar(4)	PK	NO	รหัสสินค้า
quantity	int(4)		NO	ปริมาณที่ใช้วัตถุดิบ

ตารางที่ 3.9 ตารางข้อมูลรายการที่สั่งซื้อ (Orderd\_product)

Name	Type	Key	Null	Description
order_prod_id	nvarchar(8)	PK	NO	รหัสใบสั่งซื้อ
product_id	nvarchar(4)	PK	NO	รหัสสินค้า
quantity	int(4)		NO	จำนวนที่สั่งซื้อ
note	nvarchar(300)		YES	ส่วนเพิ่มเติม
order_datetime	datetime		NO	วันเวลาที่สั่งซื้อ
order_status	nvarchar(10)		NO	สถานะการสั่งซื้อ

ตารางที่ 3.10 ตารางข้อมูลการจอง (Reservation)

Name	Type	Key	Null	Description
reserve_id	nvarchar(4)	PK	NO	รหัสการจอง
reserve_name	nvarchar(100)		NO	ชื่อผู้จอง
reserve_tel	nvarchar(100)		YES	เบอร์โทรศัพท์ผู้จอง
reserve_datetime	datetime		NO	วันเวลาการจอง
table_id	nvarchar(3)	FK	NO	โต๊ะที่จอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.11 ตารางข้อมูลใบเสร็จเก็บเงิน (Bill)

Name	Type	Key	Null	Description
bill_id	nvarchar(16)	PK	NO	รหัสใบเสร็จเก็บเงิน
bill_payment	nvarchar(16)		NO	วิธีการจ่ายเงิน
discount	float(8)		YES	ส่วนลด
bill_datetime	datetime		NO	วันที่ออกใบเสร็จเก็บเงิน
emp_id	nvarchar(3)	FK	NO	รหัสพนักงาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

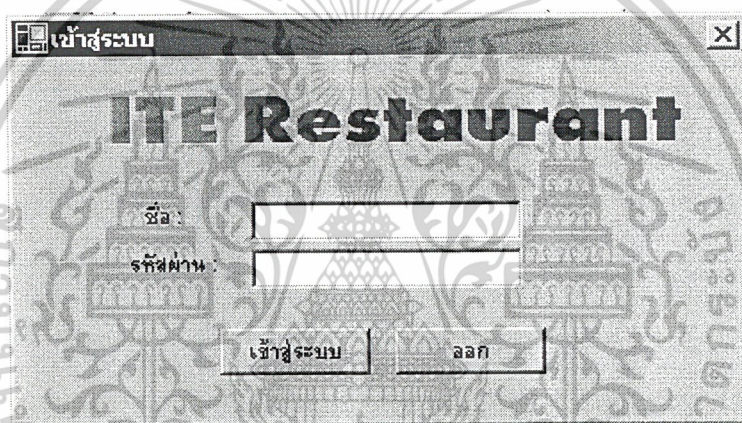
## บทที่ 4

### ผลการทดลอง

จากการดำเนินงานที่ผ่านมาจะได้ระบบร้านอาหาร ซึ่งแบ่งการทำงานออกเป็น 4 ส่วน คือ ส่วนของพนักงานคิดเงิน(Cashier), ส่วนของห้องครัว, ส่วนของบาร์เครื่องดื่ม และส่วนของบริการ สามารถแสดงยูสเซอร์อินเทอร์เฟซ (user interface) และอธิบายการใช้งานได้ ดังนี้

#### 4.1 ส่วนของพนักงานคิดเงิน

##### 4.1.1 ส่วนของหน้าจอสำหรับการเข้าใช้ระบบ (Login)



รูปที่ 4.1 หน้าจอสำหรับการล็อกอินเข้าใช้ระบบ

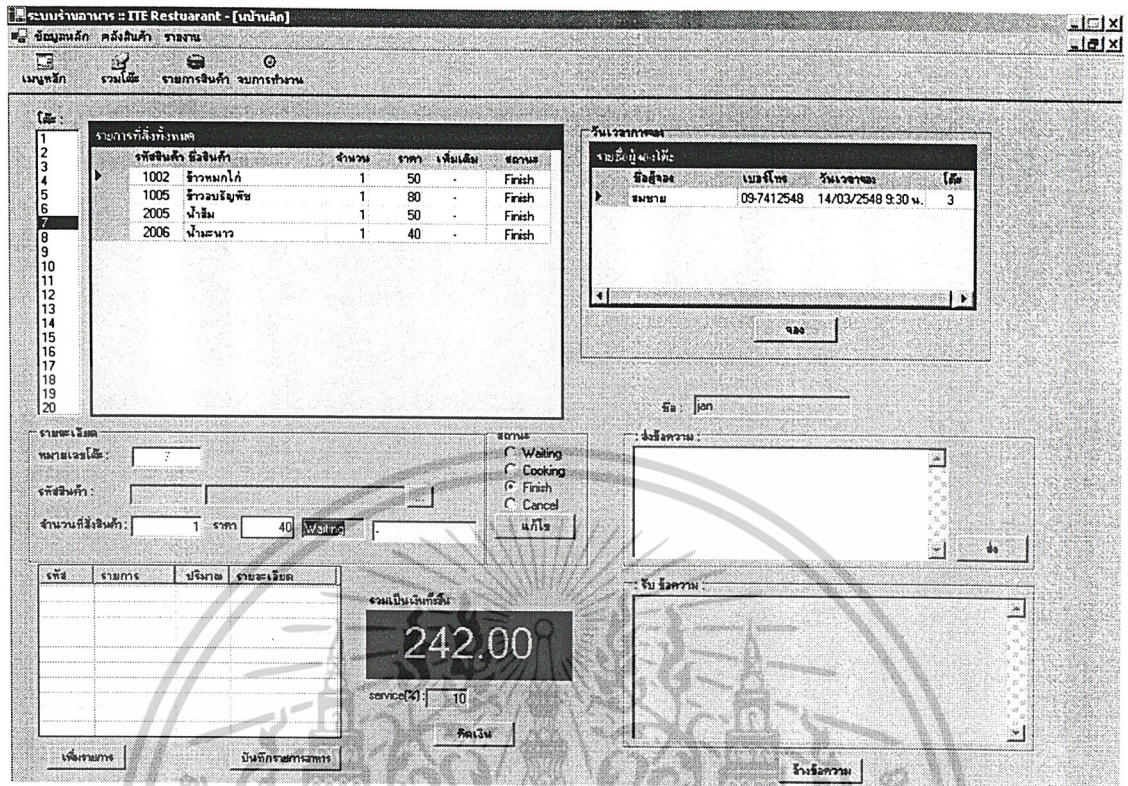
พนักงานคิดเงินสามารถเข้าใช้ระบบ โดยทำการล็อกอินผ่านหน้าจอสำหรับล็อกอิน โดยพนักงานคิดเงินจะต้องใส่ยูสเซอร์เนม (user name) และรหัสผ่าน (password) ให้ถูกต้อง

##### 4.1.2 ส่วนของหน้าจอหลักสำหรับพนักงานคิดเงิน

เมื่อเข้าสู่ระบบได้แล้ว พนักงานคิดเงินสามารถทำการสั่งอาหาร, ชำระเงิน, จองโต๊ะอาหาร, รวมโต๊ะอาหาร, ติดต่อบริกรภายในร้านผ่านพ็อกเก็ตพีซี (Pocket PC), เช็ครายการวัตถุดิบ, รวมทั้งสามารถบันทึกรายการอาหารและรายการวัตถุดิบได้

ส่วนของการสั่งอาหาร พนักงานคิดเงินสามารถเลือกโต๊ะอาหารและรายการอาหารที่ลูกค้าสั่งได้ รวมทั้งสามารถแก้ไขสถานะของอาหารที่สั่งได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 หน้าจอหลักของพนักงานคิดเงิน

ส่วนของการจองโต๊ะอาหาร พนักงานคิดเงินสามารถจองโต๊ะอาหาร จากการที่ถูกค้า โทรศัพท์เข้ามาจองโต๊ะ โดยจะมีตารางแสดงข้อมูลการจองของลูกค้า

ส่วนของการคิดราคาอาหาร พนักงานคิดเงินสามารถดูรายการอาหารและราคาอาหารทั้งหมดที่ถูกค้าสั่ง รวมทั้งสามารถคิดราคาอาหารทั้งหมด โดยแยกเป็นแต่ละโต๊ะอาหาร

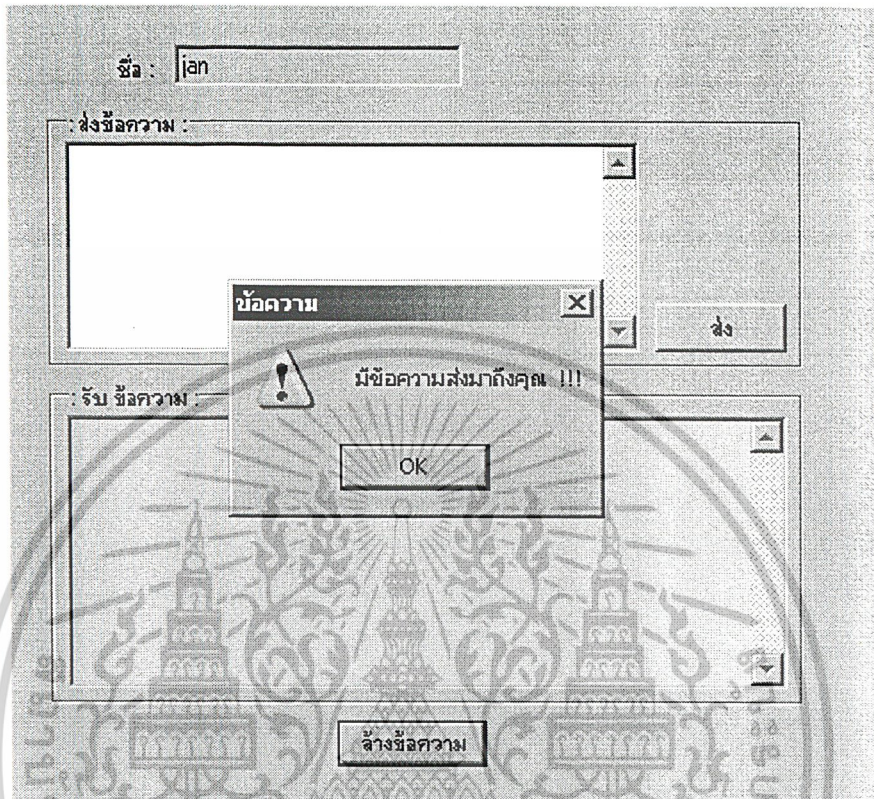
ส่วนของการติดต่อบริกรภายในร้าน พนักงานคิดเงินสามารถติดต่อบริกรภายในร้านได้ทั้งหมด โดยจะทำการสนทนาผ่านหน้าจอสำหรับการสนทนา ซึ่งจะมีกรอบสำหรับรับและส่งข้อความถึงกัน

ส่วนของการรวมโต๊ะอาหาร พนักงานคิดเงินสามารถทำการรวมราคาอาหารจากโต๊ะหนึ่งไปอีกโต๊ะหนึ่งได้ ตามความต้องการของลูกค้าในกรณีที่คิดราคาอาหารรวมกัน

นอกจากนี้ยังมีส่วนสำหรับบันทึกรายละเอียดของโต๊ะอาหารทั้งหมดภายในร้าน, ส่วนของหน้าจอแสดงรายการพนักงานทั้งหมดภายในร้าน, ส่วนของการสั่งและรับวัตถุดิบเข้าร้าน รวมทั้งส่วนของการแสดงรายงานและกราฟ ซึ่งส่วนต่างๆ จะแสดงบนเมนูบาร์ของหน้าจอหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.3 ส่วนของหน้าจอสำหรับการส่งข้อความ



รูปที่ 4.3 หน้าจอหลักสำหรับการส่งข้อความ

หน้าจอนี้เป็นส่วนของการติดต่อกับบริการภายในร้าน โดยพนักงานคิดเงินสามารถติดต่อกับบริการภายในร้าน ได้ทั้งหมด โดยจะทำการสนทนาผ่านหน้าจอสำหรับการสนทนา ซึ่งจะมีกรอบสำหรับรับและส่งข้อความถึงกัน และจะมีกล่องข้อความเข้ามาในกรณีที่ข้อความส่งเข้ามา

#### 4.1.4 ส่วนของหน้าจอสำหรับเลือกรายการอาหาร

รายชื่อสินค้า

ประเภทสินค้า **อาหาร**

รายการ : อาหาร

รหัสสินค้า	ชื่อสินค้า	ราคาต่อหน่วย	หน่วยนับ
1001	ข้าวราดหน้าปลาผงกะหรี่	50	จาน
1002	ข้าวต้มไก่	50	จาน
1003	ข้าวหลอไบบัว	70	จาน
1004	ข้าวอบสับปะรด-ทุบกรอบ	60	จาน
1005	ข้าวอบธัญพืช	80	จาน
1006	สปาเก็ตตี้ไก่อบ	60	จาน
1007	พาสต้าอบสับปะรด	70	จาน
1008	สปาเก็ตตี้หมูย่าง	60	จาน
1009	พอร์คชอว์ก๊วย	150	จาน
1010	นกน้อยแสนกลม	180	จาน
1011	มะกะโรนีผัดซีอิ้ว	85	จาน

รูปที่ 4.4 หน้าจอแสดงรายการอาหาร

พนักงานคิดเงินสามารถเลือกรายการอาหารที่ลูกค้าสั่งจากหน้าจอนี้ โดยสามารถเลือกประเภทได้ ทั้งเครื่องดื่มหรืออาหารตามแต่ลูกค้าต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.5 ส่วนของหน้าจอสำหรับชำระเงิน

The screenshot shows a payment window with the following fields and options:

- รวมเป็นเงิน (Total amount): 242.00
- รับเงินทั้งหมด (Total received): 250
- ส่วนลด (Discount): 0.00
- การจ่ายเงิน (Payment method):
  - เงินสด (Cash)
  - เครดิตการ์ด (Credit Card)
- Buttons: ตกลง (OK), ยกเลิก (Cancel)
- เงินทอน (Change): 8.00

รูปที่ 4.5 หน้าจอการชำระเงิน

พนักงานคิดเงินสามารถราคาอาหารได้จากหน้าจอการชำระเงิน โดยจะแสดงราคารวมของอาหารทั้งหมดที่ถูกคำสั่ง, ช่องสำหรับกรอกจำนวนเงินที่ถูกชำระค่าอาหารและแสดงเงินทอนจากการชำระค่าอาหาร รวมทั้งสามารถสั่งพิมพ์ใบรายการชำระเงินให้แก่ลูกค้าอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.6 ส่วนของหน้าจอสำหรับพิมพ์ใบรายการชำระเงิน

MainReport

YOI

**ITE Restaurant**

BILL 00000057  
TABLE 7

Qty.	Description	Total
1	ข้าวหมกไก่	50.00
1	ข้าวอบธัญพืช	80.00
1	น้ำส้ม	50.00
1	น้ำมะนาว	40.00
		220.00
	service 10 %	22.00
	discount	0.00
	<b>TOTAL</b>	242.00
	Cash	250.00
	<b>Change</b>	8.00

Payment: cash  
\*\*\*\*\* 15/3/2548 18:50:01 \*\*\*\*\*

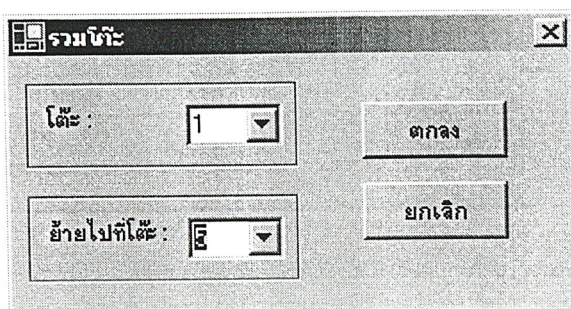
Current Page No: 1 | Total Page No: 1 | Zoom Factor: 100%

รูปที่ 4.6 หน้าจอสำหรับพิมพ์ใบรายการชำระเงิน

พนักงานคิดเงินสามารถพิมพ์ใบรายการชำระเงินได้จากหน้าจอนี้ โดยใบรายการชำระเงินจะแสดงราคาของอาหารทั้งหมดที่ลูกค้าได้สั่ง, เงินที่ลูกค้าชำระค่าอาหาร, ค่าบริการและเงินทอนที่ลูกค้าจะได้รับคืน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.7 ส่วนของหน้าจอแสดงการรวมโต๊ะ



รูปที่ 4.7 หน้าจอแสดงการรวมโต๊ะ

พนักงานคิดเงินสามารถทำการรวมโต๊ะอาหารได้ โดยในกรณีที่ลูกค้าต้องการชำระเงินรวมกัน ซึ่งสามารถรวมจากโต๊ะหนึ่งไปยังอีกโต๊ะหนึ่งได้

#### 4.1.8 ส่วนของหน้าจอแสดงการจองโต๊ะอาหาร

รหัสการจอง : 0001      หมายเลขโต๊ะ : 3        
 ชื่อผู้จอง : สมชาย        
 เบอร์โทรศัพท์ : 09-7412548        
 วันเวลาจอง : 14/03/2548 9:30 น.     

รหัสการจอง	ชื่อผู้จอง	เบอร์โทรศัพท์	วันเวลาจอง	โต๊ะที่จอง
▶ 0001	สมชาย	09-7412548	14/03/2548 9:30 น.	3

รูปที่ 4.8 หน้าจอสำหรับการจองโต๊ะอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พนักงานคิดเงินสามารถทำการจองโต๊ะอาหารในกรณีที่ลูกค้าโทรศัพท์มาจองโต๊ะอาหาร อีกทั้งพนักงานคิดเงินยังสามารถทำการแก้ไขหรือยกเลิกการจองโต๊ะอาหารตามแต่กรณีที่ลูกค้าต้องการ

#### 4.1.9 ส่วนของหน้าจอสำหรับบันทึกรายการอาหารและเครื่องดื่มภายในร้าน

รายการสินค้า

รหัสสินค้า : 1001      ชื่อสินค้า : ข้าวราดหน้าปลาผงกระหรี่      เพิ่มข้อมูล

รายละเอียด : เนื้อปลากระพง, ข้าวสาร, กระเทียมสับ, หอมใหญ่      ประเภท : อาหาร      แก้ไขข้อมูล

ราคาขาย : 50      หน่วยนับ : จาน      สถานะ : available      ล้างข้อมูล

รหัสสินค้า	ชื่อสินค้า	รายละเอียด	ประเภท	ราคาขาย	หน่วยนับ	สถานะ
1001	ข้าวราดหน้าปลาผงกระหรี่	เนื้อปลากระพง, ข้าวสาร, กระเทียมสับ, หอมใหญ่	อาหาร	50	จาน	available
1002	ข้าวหมกไก่	ข้าวสารกลิ้ง, ผงขมิ้น, ลูกผักชี	อาหาร	50	จาน	available
1003	ข้าวห่อใบมะพร้าว	กุ้งกุลาดำ, หนุ่ย, เห็ดขี้เหล็ก	อาหาร	70	จาน	available
1004	ข้าวอบสับปะรดทุเรียน	ข้าวสุก, เนื้อสันในหมู, หนุ่ย, สด	อาหาร	60	จาน	available
1005	ข้าวอบธัญพืช	ข้าวสวย, ข้าวกล้อง, ไข่แดง, กล้วย	อาหาร	80	จาน	available
1006	สปาเก็ตตี้ไก่อบ	เส้นสปาเก็ตตี้, อกไก่, หอมใหญ่	อาหาร	60	จาน	available
1007	พาสต้าอบชีสปะรด	พาสต้าต้มสุก, สับปะรด, หนุ่ย, สด	อาหาร	70	จาน	available
1008	สปาเก็ตตี้หมูผัด	สปาเก็ตตี้, สับปะรด, หนุ่ย, สด	อาหาร	60	จาน	available
1009	แซนด์วิชหมู	แซนด์วิช, หมู, ผักสด	อาหาร	150	จาน	available

รูปที่ 4.9 หน้าจอบันทึกรายการอาหารและเครื่องดื่ม

พนักงานสามารถบันทึกและแก้ไขข้อมูลรายการอาหารและเครื่องดื่มจากหน้าจอนี้ โดยจะมีตารางแสดงรายการอาหารและเครื่องดื่มทั้งหมดพร้อมทั้งมีราคาขายและสถานะของอาหารและเครื่องดื่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.10 ส่วนของหน้าจอบันทึกรายละเอียดของโต๊ะ

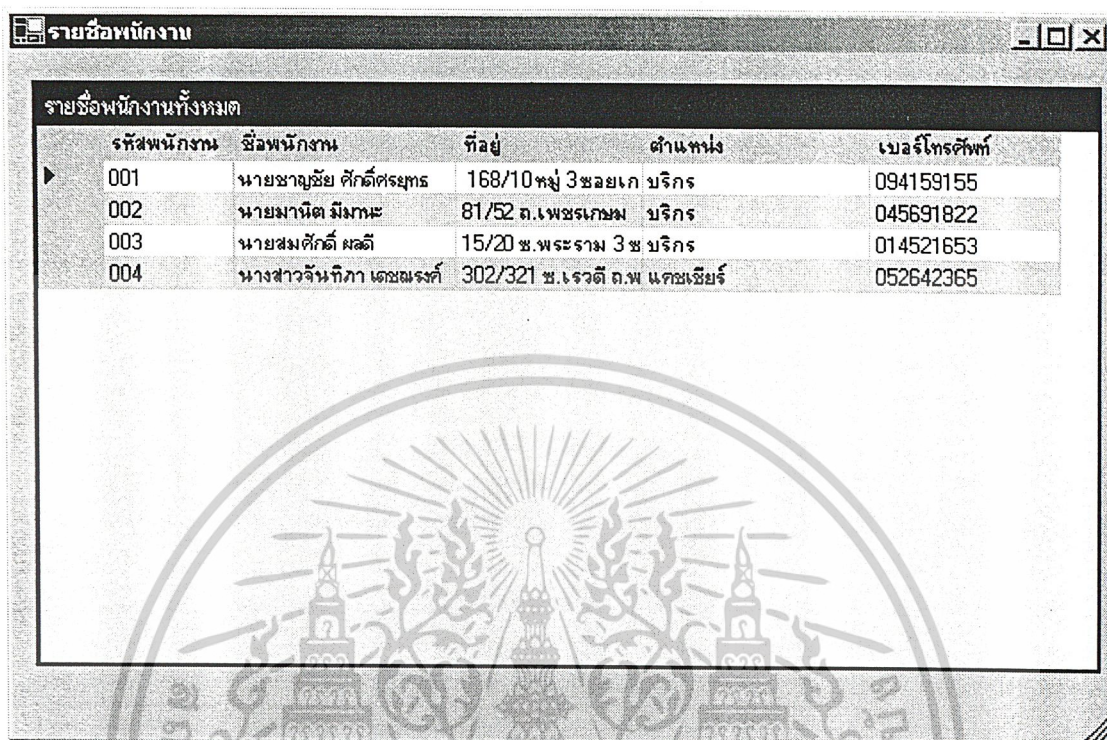
หมายเลขโต๊ะ	จำนวนที่นั่ง
4	2
5	2
3	3
10	3
6	3
1	4
11	4
15	4
16	4
17	4

รูปที่ 4.10 หน้าจอบันทึกรายละเอียดของโต๊ะอาหาร

พนักงานสามารถทำการเพิ่มจำนวน, แก้ไข, ลบรายละเอียดต่างๆ ของโต๊ะอาหารได้จากหน้าจอนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.11 ส่วนของหน้าจอแสดงรายการพนักงานทั้งหมด



The screenshot shows a window titled "รายชื่อพนักงาน" (Employee List) with a table containing the following data:

รหัสพนักงาน	ชื่อพนักงาน	ที่อยู่	ตำแหน่ง	เบอร์โทรศัพท์
001	นายชานูชัย ศักดิ์ศรียุทธ	168/10 หมู่ 3 ซอยเอก บรกร		094159155
002	นายมานิต มีมณะ	81/52 ต.เพชรเกษม บรกร		045691822
003	นายสมศักดิ์ ผลดี	15/20 ซ.พระราม 3 ซ บรกร		014521653
004	นางสาวจันทิกา เชนณรงค์	302/321 ซ.เรวัต ต.พ แคมเขยริ		052642365

รูปที่ 4.11 หน้าจอแสดงรายการพนักงานภายในร้าน

หน้าจอนี้สามารถดูรายการพนักงานทั้งหมดภายในร้านได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.12 ส่วนของหน้าจอสำหรับสั่งซื้อวัตถุดิบเข้าร้าน

ข้อมูลการสั่งซื้อ

วันที่สั่งซื้อ: 14 มีนาคม 2548      วันรับสินค้า: 16 มีนาคม 2548      1,464.00

ข้อมูลวัตถุดิบ

รหัสวัตถุดิบ:      ชื่อวัตถุดิบ:      ต้นทุน:      จำนวน:      ส่วนลด (%):

เพิ่มรายการ      ล้างรายการ

บันทึก

รหัสวัตถุดิบ	ชื่อวัตถุดิบ	ต้นทุน	จำนวน	ส่วนลด	รวมเป็นเงิน
0010	น้ำดื่มตราสิงห์	8	48	0.00	384.00
0001	เบียร์โล	40	12	0.00	480.00
0006	เบียร์ไฮเนเกน	50	12	0.00	600.00

ค่าสินค้า ส่วนลด:

รายการของที่ต้องสั่งซื้อ

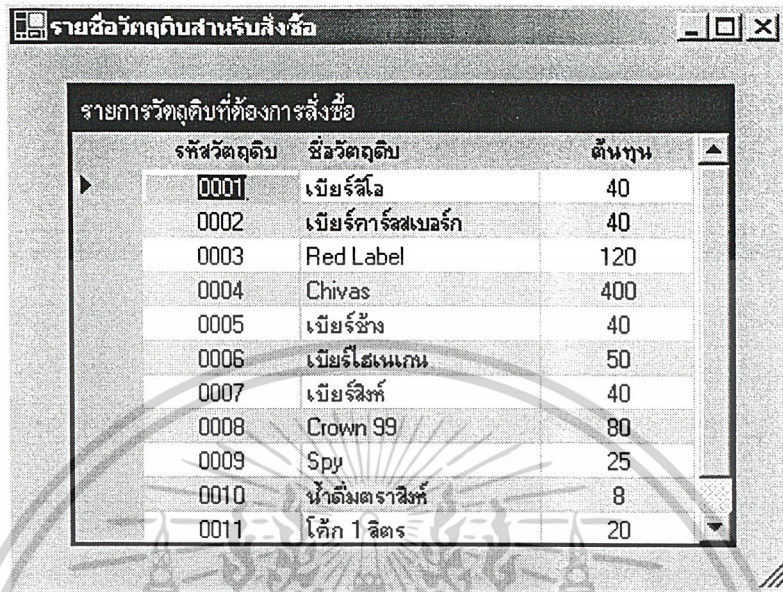
รหัสวัตถุดิบ	ชื่อวัตถุดิบ	หน่วยนับ	จำนวนคงเหลือ
0005	เบียร์ช้าง	ขวด	16
0011	โค้ก 1 ลิตร	ขวด	17

รูปที่ 4.12 หน้าจอสำหรับสั่งซื้อวัตถุดิบ

พนักงานสามารถทำการสั่งซื้อวัตถุดิบเข้าร้านได้จากหน้าจอนี้ โดยจะแสดงรายการวัตถุดิบต่างๆ พร้อมทั้งมีการคิดราคาวัตถุดิบที่ต้องการสั่งซื้อ และสามารถทำการสั่งพิมพ์ใบสั่งซื้อวัตถุดิบ นอกจากนี้ยังมีหน้าต่างแสดงรายการวัตถุดิบที่ต้องการสั่งซื้ออย่างเร่งด่วนเนื่องจากมีปริมาณคงเหลือน้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.13 ส่วนของหน้าจอสำหรับเลือกรายการวัตถุดิบที่ต้องการสั่งซื้อ



รหัสวัตถุดิบ	ชื่อวัตถุดิบ	ต้นทุน
0001	เบียร์ลีไอ	40
0002	เบียร์คาร์ลสเบอร์ก	40
0003	Red Label	120
0004	Chivas	400
0005	เบียร์ช้าง	40
0006	เบียร์ไฮเนเกน	50
0007	เบียร์สิงห์	40
0008	Crown 99	80
0009	Spy	25
0010	น้ำดื่มตราสิงห์	8
0011	โค้ก 1 ลิตร	20

รูปที่ 4.13 หน้าจอแสดงรายการวัตถุดิบ

หน้าจอนี้สามารถแสดงราคาต้นทุนของวัตถุดิบ และพนักงานสามารถเลือกรายการวัตถุดิบที่ต้องการสั่งซื้อจากหน้าจอนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.14 ส่วนของหน้าจอสำหรับพิมพ์ใบสั่งซื้อวัตถุดิบเข้าร้าน

รหัส	ชื่อสินค้า	ปริมาณ	หน่วย	ราคาต่อหน่วย	ส่วนลด	จำนวนเงินรวม
0002	เบียร์คาร์ลสเบอร์ก	12	ขวด	40.00	0	480.00
0006	เบียร์ไฮเนเกน	12	ขวด	50.00	0	600.00
0010	น้ำดื่มดาวสิงห์	48	ขวด	8.00	0	384.00
รวมทั้งหมด						1,464.00

Current Page No: 1      Total Page No: 1      Zoom Factor: 100%

รูปที่ 4.14 หน้าจอสำหรับพิมพ์ใบสั่งซื้อวัตถุดิบ

หน้าจอนี้ แสดงรายละเอียดของรายการวัตถุดิบที่ต้องการสั่งซื้อ โดยแสดงปริมาณและราคาต่อหน่วยของวัตถุดิบ รวมทั้งราคารวมทั้งหมดของวัตถุดิบที่ต้องการสั่งซื้อ พร้อมทั้งสามารถสั่งพิมพ์ใบสั่งซื้อวัตถุดิบได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.15 ส่วนของหน้าจอสำหรับ รับวัตถุดิบที่สั่ง

ค้นหา  
หมายเลขใบสั่งซื้อ: 00000007    ค้นหา

ข้อมูลวัตถุดิบ

รหัสวัตถุดิบ:    ชื่อวัตถุดิบ:    จำนวน: หน่วยนับ

บันทึกที่ละรายการ

รายการวัตถุดิบที่สั่งซื้อ

รหัสวัตถุดิบ	ชื่อวัตถุดิบ	หน่วยนับ	จำนวน	สถานะ
0002	เมียร์คาร์สเบอร์ก	ขวด	12	Order
0006	เมียร์ไฮเนเกน	ขวด	12	Order
0010	น้ำดื่มตราสิงห์	ขวด	48	Order

บันทึกทั้งหมด

รูปที่ 4.15 หน้าจอแสดงการรับวัตถุดิบ

พนักงานสามารถบันทึกรายการวัตถุดิบที่ได้รับหลังจากที่ได้ทำการสั่งไป โดยที่สามารถบันทึกได้ที่ละรายการ ในกรณีที่วัตถุดิบที่สั่งไปได้เพียงแค่บางรายการ หรือจะบันทึกทั้งหมด ในกรณีที่วัตถุดิบที่สั่งไปได้ครบตามจำนวนที่สั่ง นอกจากนี้ยังสามารถทำการค้นหาใบสั่งซื้อวัตถุดิบที่ต้องการ โดยกรอกหมายเลขใบสั่งซื้อวัตถุดิบที่ต้องการค้นหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.16 ส่วนของหน้าจอสำหรับแสดงรายการวัตถุดิบทั้งหมดภายในร้าน

วัตถุดิบ	ชื่อวัตถุดิบ	จำนวนคงเหลือ	หน่วยนับ	ราคาต่อหน่วย
0001	เบียร์สิงห์	36	ขวด	40.00
0002	เบียร์การ์ลสเบอร์ก	38	ขวด	40.00
0003	Red Label	40	ขวด	120.00
0004	Chivas	32	ขวด	400.00
0005	เบียร์ช้าง	16	ขวด	40.00
0006	เบียร์ไฮเนเกน	35	ขวด	50.00
0007	เบียร์สิงห์	20	ขวด	40.00
0008	Crown 99	61	ขวด	80.00
0009	Spy	40	ขวด	25.00
0010	น้ำดื่มตราสิงห์	24	ขวด	8.00
0011	โค้ก 1 ลิตร	17	ขวด	20.00

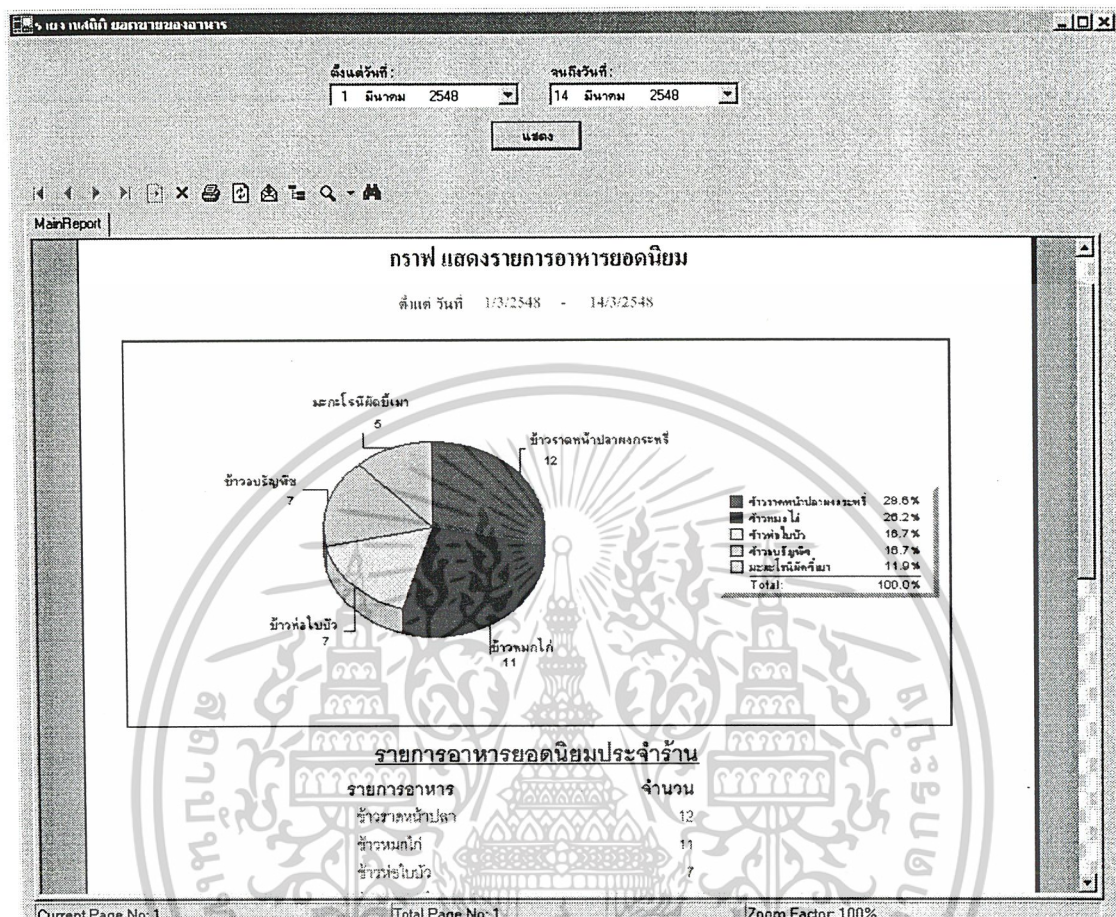
Current Page No: 1      Total Page No: 1      Zoom Factor: 100%

รูปที่ 4.16 หน้าจอสำหรับพิมพ์รายการวัตถุดิบทั้งหมด

หน้าจอนี้ แสดงรายละเอียดของรายการวัตถุดิบทั้งหมดภายในร้าน ซึ่งมีทั้งจำนวนคงเหลือ และราคาต่อหน่วยของวัตถุดิบแต่ละชนิด พร้อมทั้งสามารถสั่งพิมพ์ใบรายงานแสดงรายการวัตถุดิบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.17 ส่วนของหน้าจอแสดงรายงานสถิติอาหารยอดนิยม

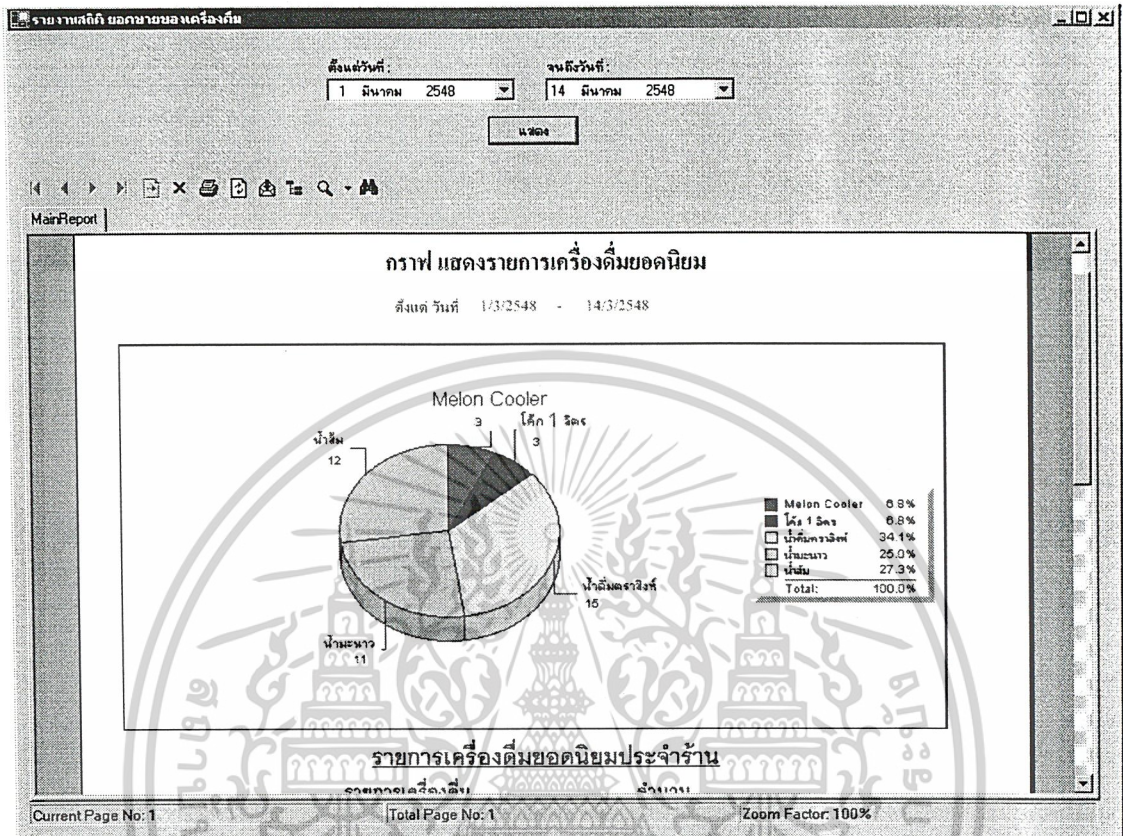


รูปที่ 4.17 หน้าจอแสดงรายงานสถิติอาหารยอดนิยม

หน้าจอนี้ แสดงรายละเอียดของอาหารยอดนิยม โดยจะแสดงข้อมูลอยู่ในรูปของกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.18 ส่วนของหน้าจอแสดงรายงานสถิติเครื่องดื่มยอดนิยม



รูปที่ 4.18 หน้าจอแสดงรายงานสถิติเครื่องดื่มยอดนิยม

หน้าจอนี้ แสดงรายละเอียดของเครื่องดื่มยอดนิยม โดยจะแสดงข้อมูลอยู่ในรูปของกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 ส่วนของบริการ (pocket pc)

### 4.2.1 ส่วนของหน้าจอสำหรับการเข้าใช้ระบบ (Login)



รูปที่ 4.19 หน้าจอสำหรับการล็อกอินเข้าใช้ระบบ

บริการสามารถเข้าใช้ระบบโดยทำการล็อกอินผ่านหน้าจอสำหรับล็อกอิน โดยบริการจะต้องใส่ยูสเซอร์เนม (username) และรหัสผ่าน (password)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.2 ส่วนของหน้าจอหลักสำหรับบริการ



รูปที่ 4.20 หน้าจอหลักสำหรับบริการ

เมื่อเข้าสู่ระบบแล้ว บริการสามารถสั่งอาหารและตรวจสอบรายการอาหารที่ลูกค้าสั่งได้ โดยเลือกหมายเลขโต๊ะที่ต้องการจะสั่งอาหารหรือตรวจสอบรายการอาหาร นอกจากนี้ยังสามารถติดต่อกับพนักงานคิดเงินจากหน้าจอนี้ได้เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.3 ส่วนของหน้าจอสำหรับสั่งอาหาร



รูปที่ 4.21 หน้าจอสำหรับบริการทำการสั่งอาหาร

ส่วนของการสั่งอาหาร หลังจากที่บริการเลือกโต๊ะอาหาร บริการสามารถทำการสั่งอาหารได้ โดยเลือกจากรายการอาหารหรือเครื่องดื่มที่มี อีกทั้งยังสามารถเพิ่มรายละเอียดของอาหารผ่านทางหน้าจอนี้ได้อีกด้วย ก่อนที่รายการอาหารทั้งหมดจะถูกส่งไปยังห้องครัวเพื่อจัดทำต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.4 ส่วนของหน้าจอสำหรับตรวจสอบรายการอาหาร



รูปที่ 4.22 หน้าจอสำหรับตรวจสอบรายการอาหาร

บริการสามารถตรวจสอบรายการอาหารที่โต๊ะนั้นๆ ได้สั่ง โดยสามารถดูสถานะของอาหารที่สั่ง รวมทั้งยังสามารถยกเลิกรายการอาหารที่สั่งไปได้ ในกรณีที่ลูกค้าต้องการยกเลิกการสั่งนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.5 ส่วนของหน้าจอสำหรับตรวจสอบราคาอาหาร



รูปที่ 4.23 หน้าจอแสดงราคาอาหาร

บริกรสามารถตรวจสอบราคาอาหารที่โต๊ะนั้นๆ สั่ง โดยจะแสดงราคาอาหารแต่ละรายการ รวมทั้งยังแสดงราคาอาหารทั้งหมดที่ลูกค้าได้สั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.6 ส่วนของหน้าจอสำหรับตรวจสอบรายละเอียดของโต๊ะอาหาร



รูปที่ 4.24 หน้าจอแสดงรายละเอียดโต๊ะอาหาร

บริการสามารถตรวจสอบรายละเอียดของโต๊ะอาหาร โดยหน้าจอสามารถแสดงจำนวนที่นั่งของแต่ละโต๊ะ รวมทั้งยังสามารถดูรายละเอียดของโต๊ะที่ได้ถูกจองไว้แล้ว ซึ่งจะแสดงวันเวลาการจองและชื่อของผู้จอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.7 ส่วนของหน้าจอสำหรับแสดงรูปภาพและส่วนประกอบของอาหาร



รูปที่ 4.25 หน้าจอแสดงรูปภาพและรายละเอียดอาหาร

ในกรณีที่ลูกค้าต้องการทราบว่าหน้าตาอาหารเป็นอย่างไร บริการสามารถค้นหาและแสดงรูปภาพของอาหาร พร้อมทั้งรายละเอียดของส่วนผสมของอาหารที่ลูกค้าต้องการได้จากหน้าจอนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.8 ส่วนของหน้าจอสำหรับติดต่อกับพนักงานภายในร้าน



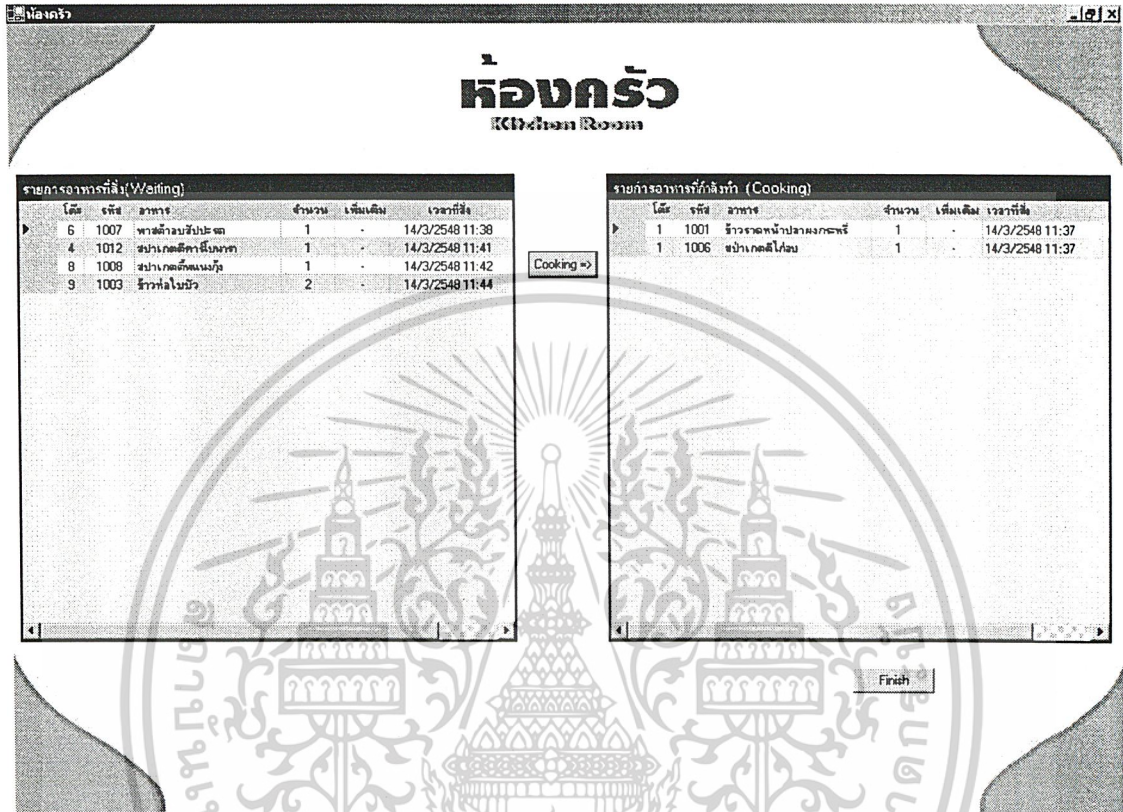
รูปที่ 4.26 หน้าจอสำหรับการติดต่อกับพนักงานคิดเงิน

หน้าจอนี้ใช้ติดต่อระหว่างบริกรกับพนักงานคิดเงินซึ่งทำให้สะดวกรวดเร็วในการพูดคุยกัน โดยบริกรสามารถส่งคำสั่งแจ้งไปยังพนักงานคิดเงินเพื่อทำการคิดราคาอาหารแต่ละโต๊ะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.3 ส่วนของห้องครัว

### 4.3.1 ส่วนของหน้าจอหลักสำหรับห้องครัว



รูปที่ 4.27 หน้าจอหลักสำหรับห้องครัว

ในส่วนนี้จะมีหน้าต่างแสดงข้อมูล 2 หน้าต่างคือ หน้าต่างของรายการอาหารที่สั่งเข้ามา และหน้าต่างของรายการอาหารที่กำลังทำ โดยหน้าต่างของรายการอาหารที่สั่งเข้ามาจะเรียงตามลำดับเวลาก่อนหลัง เพื่อที่พ่อครัวจะได้ทำอาหารตามลำดับที่ลูกค้าสั่งได้ถูกต้อง ส่วนตารางของอาหารที่กำลังทำจะแสดงรายละเอียดของอาหารที่พ่อครัวกำลังลงมือทำ หลังจากทำเสร็จพ่อครัวจะทำการกดปุ่ม Finish เพื่อบ่งบอกว่าอาหารที่ทำนั้นเสร็จสิ้นเรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.4 ส่วนของบาร์เครื่องดื่ม

### 4.4.1 ส่วนของหน้าจอหลักสำหรับบาร์เครื่องดื่ม



รูปที่ 4.28 หน้าจอหลักสำหรับบาร์เครื่องดื่ม

ในส่วนนี้จะมีหน้าต่างแสดงรายการเครื่องดื่มที่ลูกค้าสั่ง โดยจะแสดงรายละเอียดของเวลาที่ลูกค้าสั่งเข้ามาเรียงลำดับก่อนหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปผลการทดลอง

#### 5.1 สรุปผลการทดลอง

ระบบที่ได้สามารถรองรับการสั่งอาหารผ่านอุปกรณ์ไร้สายได้ โดยมีการดำเนินการนั้น จะประกอบด้วย 4 ส่วน คือ ส่วนหน้าจอของพนักงานคิดเงิน ส่วนหน้าจอของบริกร ส่วนหน้าจอห้องครัว ส่วนหน้าจอบาร์เครื่องดื่ม ซึ่งจากการทดสอบ สามารถช่วยเพิ่มความสะดวกในการให้บริการ รวมถึงหน้าจอในส่วนต่างๆ ได้ถูกออกแบบให้สามารถใช้งานได้ง่าย นอกจากนี้ ระบบที่ได้พัฒนาขึ้นนี้ มีความถูกต้องและแม่นยำในการนำไปใช้งาน

#### 5.2 ปัญหาที่พบและข้อจำกัดในการพัฒนาโครงการ

1. เนื่องจากการพัฒนาซอฟต์แวร์บนอุปกรณ์ไร้สาย (PDA) เป็นเรื่องใหม่ และยังมีนักพัฒนามากนัก จึงทำให้ยากลำบากในการหาข้อมูลในการพัฒนา รวมถึงเอกสารหรือหนังสือยังมีจำนวนอยู่น้อยมาก ทำให้เสียเวลาในการศึกษาข้อมูลค่อนข้างมาก และเกิดความเข้าใจผิดในเนื้อหาบางส่วนที่ได้ศึกษา
2. ข้อจำกัดบางประการของตัวภาษาในการพัฒนาบนอุปกรณ์ไร้สาย ทำให้ต้องศึกษารวมถึงเลือกใช้แนวทางและวิธีการพัฒนาให้เหมาะสม ทำให้เสียเวลาในจุดนี้
3. ในส่วนของโครงการ ตัวระบบเองมีความซับซ้อนและรายละเอียดปลีกย่อยมาก ทำให้เสียเวลาในการออกแบบฐานข้อมูล

#### 5.3 แนวทางในการพัฒนาต่อ

1. ทำการพัฒนาในส่วนของ E-coupon เพื่อเป็นการเพิ่มการบริการให้แก่ลูกค้า รวมถึงส่งเสริมการขายด้วย
2. ทำการพัฒนาระบบการส่งข้อความผ่านทางเสียงพูด เพื่อเป็นการเพิ่มความยืดหยุ่นแก่ระบบมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

1. ชาลี วรกุลพิพัฒน์, เทพฤทธิ์ บัณฑิตวัฒนาวงศ์, “UML ภาษามาตรฐานเพื่อผู้พัฒนาซอฟต์แวร์” บริษัท ซีเอ็ดดูเคชั่น จำกัด(มหาชน), 2544
2. ศุภชัย สมพานิช, “สร้างระบบฐานข้อมูลด้วย Visual Basic .NET ” ไอดีซี อินโฟ คิสทริบิวเตอร์ เซ็นเตอร์ จำกัด, 2546
3. อำไพ สินลิขิตกุล, “การประยุกต์และออกแบบฐานข้อมูล ด้วย Oracle และ SQL Server ” บริษัท ซีเอ็ดดูเคชั่น, 2546
4. เอกสารการสอน ชุดวิชา การจัดการระบบฐานข้อมูล โดย มหาวิทยาลัยสุโขทัยธรรมาธิราช โรงพิมพ์ มหาวิทยาลัยสุโขทัยธรรมาธิราช, 2544
5. ดร.ศุภมิตร จิตตะยโสธร, “เอกสารประกอบการสัมมนา Database Design and SQL” โดย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
6. Dan Fox, Jon Box, “Building Solutions with the Microsoft .NET Compact Framework: Architecture and Best Practices for Mobile Development” Addison Wesley, 2003
7. Andy Wigley, Stephen Wheelwright, “MICROSOFT .NET COMPACT FRAMEWORK” Microsoft Press, 2003
8. สุรพรรณย์ เพ็ญจำรัส, “เรียนลัด C# และการเขียน โปรแกรม.NET” โปรวิชั่น, 2546
9. เรวัตกร ธรรมมาภิรมย์, “เจาะลึกเทคโนโลยีใหม่ Microsoft .NET Framework” เอส.พี.ซี.บุ๊กส์, 2544
10. [www.microsoft.com](http://www.microsoft.com)
11. <http://www.twoguru.com/playground/benefit.htm>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้