

โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่งเพื่อระบบโลจิสติกส์

Product Arrangement and Distributing Sequence Program for Logistics System



โครงการพิเศษเป็นส่วนหนึ่งของภาควิชาเทคโนโลยีสารสนเทศ วิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2555

โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่งเพื่อระบบโลจิสติกส์

Product Arrangement and Distributing Sequence Program for Logistics System



โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูผู้สอนเพื่อใช้ในการสอนเท่านั้น ไม่ควรเอาไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปีการศึกษา 2555

**PRODUCT ARRANGEMENT AND DISTRIBUTING SEQUENCE PROGRAM
FOR LOGISTICS SYSTEM**



**A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE
IN COMPUTER SCIENCE**

**FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

ACADEMIC YEAR 2012

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงชื่อเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการพิเศษ โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่งเพื่อระบบโลจิสติกส์
Product Arrangement and Distributing Sequence Program for Logistics System

ชื่อนักศึกษา นายเนกย์ ตันทวารักษ์ 52050731
นายพรพล มั่นธรรม 52050757

ปริญญา วิทยาศาสตรบัณฑิต
สาขาวิชา วิทยาการคอมพิวเตอร์
อาจารย์ที่ปรึกษา ดร.วรางคณา กิมปาน

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ประจำปีการศึกษา 2555

คณะกรรมการสอบ	ลายมือชื่อ
ผศ.ดร.นันทิกา เบญจเทพานันท์ ประธานกรรมการ	
รศ.ดร.จิรพร วีระพันธุ์ กรรมการ	
ดร.วรางคณา กิมปาน กรรมการและอาจารย์ที่ปรึกษา	

ลิขสิทธิ์ของคณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการพิเศษ	โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่งเพื่อระบบโลจิสติกส์	
รายชื่อนักศึกษา	นายเนกษ์ ดันทวรัักษ์	52050731
	นายพรพล มั่นธรรม	52050757
ปริญญา	วิทยาศาสตร์บัณฑิต	
สาขาวิชา	วิทยาการคอมพิวเตอร์	
ปีการศึกษา	2555	
อาจารย์ที่ปรึกษา	ดร.วรางคณา กิมปาน	

บทคัดย่อ

โครงการพิเศษนี้นำเสนอการจัดเรียงกล่องสินค้าด้วยโปรแกรมคอมพิวเตอร์ โดยจุดมุ่งหมายของโปรแกรมนี้เพื่อลดเวลาที่ใช้ในการจัดเรียงสินค้า รวมถึงลดเวลาและค่าใช้จ่ายในการจัดส่ง ซึ่งโปรแกรมสามารถจัดเรียงสินค้าโดยเลือกขนาดของกล่องและขนาดของกระบะได้อย่างอิสระ โดยจะจัดเรียงสินค้าตามลำดับการจัดส่ง ด้วยอัลกอริทึม Best Fit เพื่อให้ใช้พื้นที่ได้อย่างมีประสิทธิภาพ ซึ่งผู้ใช้สามารถเพิ่มแก้ไข และลบข้อมูลต่างๆ เช่น การจัดส่ง กล่อง หรือรถ เป็นต้น โดยผู้ใช้สามารถค้นหากล่องที่ถูกจัดเรียงไปแล้ว เพื่อระบุตำแหน่งของกล่องที่อยู่ในกระบะ ซึ่งโปรแกรมจะออกรายงานการจัดส่งเพื่อให้ผู้ใช้ได้ตรวจสอบรายละเอียดการจัดส่งสินค้า โดยโปรแกรมจะแสดงผลการจัดเรียงสินค้าในรูปแบบ 3 มิติ ด้วย OpenTK ซึ่งมีการเคลื่อนไหวของมุมมองได้อย่างอิสระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Product Arrangement and Distributing Sequence Program for Logistics System		
Students	Mr.Nake	Tantawarak	52050731
	Mr.Pornphol	Mantham	52050757
Degree	Bachelor of Science		
Major Program	Computer Science		
Academic Year	2012		
Advisor	Dr.Warangkhana Kimpan		

ABSTRACT

This project proposed the product arrangement using computer program for logistics system. The aim is to reduce logistics costs, arrangement time and delivery time. The program can arrange different sizes of cubical boxes into the bin of trucks, vans, or wagons considering their distributing sequence by using Best Fit algorithm to optimize the space. Users can use this program to insert, update, and delete arrangement, boxes and vehicles information. Different types of boxes and vehicles can be adjusted and arranged freely. Users can also search the arranged boxes in order to check them. The program can create the report for users to check the delivery details. The experimental results of the program are shown in 3D using OpenTK with freely movement view.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำโครงการพิเศษหัวข้อ โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่งเพื่อระบบโลจิสติกส์ สำเร็จลุล่วงไปได้ด้วยดี เนื่องจากได้รับความอนุเคราะห์ของบุคคลหลายท่านดังนี้

ขอขอบพระคุณ ดร.วรางคณา กิมปาน อาจารย์ที่ปรึกษาในการทำหัวข้อโครงการพิเศษนี้ ที่ได้ให้ความรู้ คำแนะนำ การแก้ไขและตรวจทานข้อบกพร่องต่างๆ อย่างละเอียด ด้วยความเอาใจใส่ทุกขั้นตอน เพื่อให้การทำหัวข้อโครงการพิเศษนี้สมบูรณ์ที่สุด และที่สำคัญที่สุดอาจารย์เป็นผู้ให้กำลังใจให้มีแรงใจในการจัดทำและแก้ไขปัญหาต่างๆ ในหัวข้อโครงการพิเศษนี้อย่างเต็มใจ

ขอขอบพระคุณ อาจารย์สาขาวิทยาการคอมพิวเตอร์ทุกท่าน ที่อบรมสั่งสอน ให้ความรู้ต่างๆ แก่ทางคณะผู้จัดทำมาตลอดระยะเวลา 4 ปี จนกระทั่งโครงการพิเศษนี้สำเร็จลุล่วงไปได้ด้วยดี

ขอขอบพระคุณ บิดา มารดา และสมาชิกในครอบครัวทุกคน ผู้ซึ่งอยู่เบื้องหลัง คอยอบรมสั่งสอน ให้กำลังใจและความอบอุ่นเสมอมา

สุดท้ายนี้ต้องขอบคุณ เพื่อนๆ ทุกคน ที่คอยให้กำลังใจและให้คำปรึกษาในเรื่องต่างๆ เสมอมา

นายเนกษ์ ตันทวรัักษ์

นายพรพล มั่นธรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูป	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์ของงานวิจัย	1
1.3 ขอบเขตของงานวิจัย	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 ขั้นตอนในการดำเนินการ	3
1.6 อุปกรณ์ที่ใช้ในการทำโครงงานพิเศษ	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	4
2.1 ระบบโลจิสติกส์	4
2.1.1 บทบาทสำคัญของโลจิสติกส์	5
2.1.2 กิจกรรมที่สำคัญของโลจิสติกส์	6
2.1.3 เป้าหมายที่สำคัญของโลจิสติกส์	6
2.2 Bin Packing	6
2.2.1 การกำหนดปัญหา	7
2.2.2 อัลกอริทึม Best Fit (Best Fit Algorithm)	8
2.3 OpenGL	9
2.3.1 OpenGL คืออะไร	9
2.3.2 องค์ประกอบของ OpenGL	9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.3.3 ฟังก์ชันใน OpenGL	11
2.3.3.1 ฟังก์ชันทำงาน	11
2.3.3.2 ฟังก์ชันเรียกกลับ (Callback function)	11
2.3.3.3 ฟังก์ชันกำหนดการวนรอบ	12
2.3.4 ตัวอย่าง OpenGL Code	12
2.4 OpenTK	18
2.4.1 ตัวอย่าง OpenTK Code	18
2.5 Visual Basic	21
2.5.1 Visual Basic คืออะไร	21
2.5.2 แนวคิดของ Visual Basic	21
2.5.3 วัตถุประสงค์ของ Visual Basic	21
2.5.4 Visual Basic in Visual Studio 2010	22
บทที่ 3 วิธีการดำเนินงาน	24
3.1 Use Case Diagram	24
3.2 Activity Diagram	25
3.2.1 Activity Diagram สำหรับการเพิ่มข้อมูล	25
3.2.2 Activity Diagram สำหรับการแก้ไขข้อมูล	26
3.2.3 Activity Diagram สำหรับการลบข้อมูล	27
3.2.4 Activity Diagram สำหรับดูการจัดเรียงสินค้า	28
3.2.5 Activity Diagram สำหรับการเข้าสู่ระบบ	28
3.3 ER Diagram	29
3.4 การประยุกต์ใช้ Best Fit Algorithm	32
3.5 Flow Chart	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.5.1 Flow Chart แสดงการทำงานของ Best Fit Algorithm	33
3.5.2 Flow Chart แสดงการทำงานของระบบ	34
3.6 Graphical User Interface	35
3.6.1 หน้าหลักของโปรแกรม	35
3.6.2 หน้ารายละเอียดกล่อง	36
3.6.3 หน้าการจัดส่งสินค้า	36
บทที่ 4 ผลการดำเนินงาน	37
4.1 การแสดงผลของโปรแกรมจัดเรียงสินค้า	37
4.2 ผลการจัดเรียงสินค้า	42
บทที่ 5 สรุปผลและข้อเสนอแนะ	49
5.1 สรุปผลการทดลอง	49
5.2 ปัญหาและอุปสรรคในการทดลอง	49
5.3 ข้อเสนอแนะและแนวทางการพัฒนา	49
เอกสารอ้างอิง	50
ภาคผนวก ก. การติดตั้งโปรแกรมที่จำเป็นต่อการพัฒนาระบบ	51
ก.1 การติดตั้งโปรแกรม Visual Studio 2010	52
ก.2 การติดตั้ง OpenTK	55
ภาคผนวก ข. การติดตั้งโปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่งเพื่อระบบโลจิสติกส์	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 เปรียบเทียบฟังก์ชันของ OpenGL และ OpenTK	18
3.1 truck	30
3.2 arrangement	30
3.3 orderArr	30
3.4 descriptionBox	31
3.5 box	31
3.6 province	31
3.7 district	32
3.8 accountLogin	32
4.1 ข้อมูลของกล่องที่จัดเรียงในจังหวัดที่ 1	42
4.2 ข้อมูลของกล่องที่จัดเรียงในจังหวัดที่ 2	43
4.3 ข้อมูลของกล่องที่จัดเรียงในจังหวัดที่ 3	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 แผนผังกระบวนการทางโลจิสติกส์	4
2.2 การจัดเรียงสิ่งของด้วยอัลกอริทึม Best Fit	8
2.3 โครงสร้างของ OpenGL	10
2.4 หน้าการสร้างฟอร์มด้วยโปรแกรม Visual Studio 2010	22
2.5 หน้าจอการเขียนโค้ดด้วยโปรแกรม Visual Studio 2010	22
3.1 Use Case Diagram	24
3.2 Activity Diagram สำหรับการเพิ่มข้อมูล	25
3.3 Activity Diagram สำหรับการแก้ไขข้อมูล	26
3.4 Activity Diagram สำหรับการลบข้อมูล	27
3.5 Activity Diagram สำหรับการดูการจัดเรียงสินค้า	28
3.6 Activity Diagram สำหรับการเข้าสู่ระบบ	28
3.7 ER Diagram	29
3.8 Flow Chart แสดงการทำงานของ Best Fit Algorithm	33
3.9 Flow Chart แสดงการทำงานของระบบ	34
3.10 หน้าหลักของโปรแกรม	35
3.11 หน้าหน้ารายละเอียดกล่อง	36
3.12 หน้าการจัดส่งสินค้า	36
4.1 หน้า Log in	37
4.2 หน้าแก้ไขข้อมูล	37
4.3 หน้าข้อมูลรถ	38
4.4 หน้าข้อมูลกล่อง	38
4.5 หน้าข้อมูลจังหวัด	39
4.6 หน้าข้อมูลอำเภอที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์อื่น	39
4.7 หน้าข้อมูลผู้ใช้	40

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.8 หน้าออร์เดอร์	40
4.9 หน้าการจัดส่งสินค้า	41
4.10 หน้าจอการจัดเรียงสินค้าในรูปแบบ 3 มิติ	41
4.11 ผลการจัดเรียงสินค้ากรณีที่ 1	42
4.12 ผลการจัดเรียงสินค้ากรณีที่ 2	43
4.13 ผลการจัดเรียงสินค้ากรณีที่ 3	44
4.14 สินค้ามีน้ำหนักรวมเกินกว่าที่รถสามารถบรรทุกได้	45
4.15 สินค้ามีจำนวนเกินกว่าที่รถสามารถบรรทุกได้	45
4.16 แสดงหมายเลขกล่อง	46
4.17 แสดงหมายเลขออร์เดอร์	47
4.18 ค้นหาโดยแสดงผลหมายเลขกล่อง	47
4.19 ค้นหาโดยแสดงผลหมายเลขออร์เดอร์	48
4.20 รายงานการจัดส่งสินค้า	48
ก.1 หน้าMicrosoft Visual Studio 2010 Setup	52
ก.2 หน้า Microsoft Visual Studio 2010 Ultimate	52
ก.3 หน้า Microsoft Visual Studio 2010 Ultimate Setup – Start Page	53
ก.4 หน้า Microsoft Visual Studio 2010 Ultimate Setup – Options Page	53
ก.5 หน้า Microsoft Visual Studio 2010 Ultimate Setup – Install Page	54
ก.6 หน้าMicrosoft Visual Studio 2010 Ultimate Setup – Finish Page	54
ก.7 หน้าลิขสิทธิ์ของ OpenTK	55
ก.8 หน้าเลือก Pathที่ใช้ในการติดตั้งโปรแกรม	55
ก.9 หน้าเลือกส่วนต่างๆ ที่ต้องการติดตั้งของโปรแกรม	56
ก.10 หน้าการติดตั้ง OpenTK สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้	56

สารบัญรูป(ต่อ)

รูปที่	หน้า
ก.11 หน้าสิ้นสุดการติดตั้งโปรแกรม	57
ข.1 ไฟล์ติดตั้งโปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง	59
ข.2 หน้า ProductArrangement Setup	59
ข.3 หน้าเลือก path ที่ใช้ในการติดตั้ง โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง	60
ข.4 หน้ายืนยันการติดตั้ง โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง	60
ข.5 หน้าการติดตั้ง โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง	61
ข.6 หน้าสิ้นสุดการติดตั้งโปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง	61



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

ปัจจุบันการประกอบการธุรกิจได้ขยายตัวอย่างรวดเร็ว ทำให้เกิดการขนส่งสินค้ามากขึ้น การขนส่งถูกจัดความสำคัญไว้เป็นลำดับต้นๆ ที่ช่วยสนับสนุนกิจการด้านต่างๆ ให้ประสบผลสำเร็จ ซึ่งการขนส่งมีบทบาทสำคัญต่อการสนับสนุนการกระจายสินค้าสู่ตลาด ทั้งยังมีบทบาทเป็นดัชนีชี้วัดความเจริญก้าวหน้าของประเทศได้อีกทางหนึ่ง เพราะการขนส่งทำหน้าที่ในการเคลื่อนย้ายปัจจัยการผลิตจากแหล่งผลิตต่างๆ มาสู่โรงงานเพื่อใช้ในการผลิตสินค้า เมื่อผลิตเป็นสินค้าสำเร็จรูปแล้วก็นำมาเก็บไว้ในคลังสินค้า เพื่อจัดส่งผ่านไปยังพ่อค้าคนกลางจนกระทั่งถึงผู้บริโภค ในเวลาที่ผู้บริโภคต้องการและในสถานที่ที่ผู้บริโภคสะดวกที่จะซื้อหา นอกจากนี้การขนส่งยังมีผลต่อต้นทุนรวมในการสนับสนุนการกระจายสินค้าสู่ตลาดอีกด้วย เพราะค่าใช้จ่ายในการขนส่งสินค้าถือเป็นต้นทุนส่วนหนึ่งในการนำมากำหนดราคาสินค้าที่จำหน่ายในตลาด ดังนั้นการขนส่งจึงมิใช่เรื่องของการพัฒนาขนพาหนะ หรือการแข่งขันทางด้านขนตรกรรมเทคโนโลยีเท่านั้น แต่ยังคงคำนึงถึงวิธีการที่เรียกว่า ระบบการขนส่ง หรือกระบวนการบริหารจัดการทางด้านการขนส่งอย่างเป็นระบบ ในแง่การขนส่งสินค้า ก็ยังต้องใช้วิธีการทางโลจิสติกส์เข้ามาบูรณาการอย่างเป็นระบบ

ทางคณะผู้จัดทำจึงได้ทำการศึกษาเรื่องของการขนส่งหรือโลจิสติกส์ เพื่อแก้ปัญหาการบรรทุกและจัดส่งสินค้าไปยังสถานที่ต่างๆ ซึ่งจะทำเป็นโปรแกรมหรือแอปพลิเคชัน ที่ช่วยให้ผู้ใช้สามารถจัดเรียงสิ่งของบนกระเบื้องและการจัดส่งสินค้าไปยังสถานที่ต่างๆ ได้อย่างมีประสิทธิภาพ ซึ่งในการศึกษารั้งนี้ได้นำทฤษฎี Bin-Packing Algorithm, Visual Basic และ OpenGL มาประยุกต์ใช้ในการพัฒนาแอปพลิเคชัน เพื่อช่วยให้องค์กรธุรกิจหรือผู้จัดส่งสินค้าได้มีแนวคิดในการจัดเรียงและจัดส่งสินค้ามากยิ่งขึ้น ซึ่งจะช่วยให้ประหยัดเวลาและลดต้นทุนได้เป็นอย่างดี

1.2 วัตถุประสงค์ของงานวิจัย

- 1) เพื่อศึกษาและพัฒนาแอปพลิเคชันที่เกี่ยวกับการบรรทุกและจัดส่งสินค้าไปยังสถานที่ต่างๆ
 - 2) ช่วยให้จัดเรียงสินค้าบนกระเบื้องก่อนการจัดส่งเป็นไปอย่างมีประสิทธิภาพ ใช้พื้นที่บนกระเบื้องอย่างคุ้มค่า การขนส่งที่มีประสิทธิภาพจะสามารถทำการขนส่งวัตถุดิบในแต่ละครั้งได้เป็นจำนวนมาก
 - 3) ช่วยให้การขนส่งสินค้าไปยังสถานที่ต่างๆ เป็นไปอย่างมีประสิทธิภาพ และลดต้นทุนการขนส่ง
- เนื่องจากใช้วิธีการขนส่งที่จะช่วยให้ได้ระยะทางที่สั้นที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานานับ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ ถูกจัดส่งได้อย่างรวดเร็ว ตรงต่อเวลา

5) สามารถนำแอปพลิเคชันนี้ไปประยุกต์ใช้กับธุรกิจต่างๆ ได้

1.3 ขอบเขตของงานวิจัย

- 1) สร้างโปรแกรมโดยนำอัลกอริทึมมาประยุกต์ใช้ รวมถึงสร้างฐานข้อมูล (Database) เพื่อใช้สำหรับการเชื่อมโยงข้อมูลระหว่างส่วนที่ติดต่อกับผู้ใช้และส่วนกราฟิก
- 2) จัดทำกราฟิก 3D เพื่อใช้สำหรับดูรายละเอียดการจัดเรียงสินค้า ทำให้เข้าใจการจัดเรียงมากยิ่งขึ้น
- 3) โปรแกรมนี้ครอบคลุมเฉพาะการจัดเรียงสินค้าเท่านั้น ไม่ครอบคลุมถึงเส้นทางการจัดส่งสินค้า

1.4 ประโยชน์ที่คาดว่าจะได้รับ

ก. ส่วนผู้พัฒนาโปรแกรม

- 1) ได้เรียนรู้และรับประสบการณ์ใหม่จากการเขียนโปรแกรม
- 2) เข้าใจเกี่ยวกับการขนส่งหรือโลจิสติกส์มากยิ่งขึ้น
- 3) มีความรู้ทางด้าน Bin-Packing Algorithm, Logistics, Visual Basic และ OpenGL
- 4) ได้ฝึกฝนการทำงานร่วมกันเป็นทีม

ข. ส่วนผู้ใช้งาน

- 1) แอปพลิเคชันนี้จะช่วยให้ผู้ใช้งานมีความสะดวกสบายในเรื่องการบรรทุกและจัดส่งของมากยิ่งขึ้น เพียงแค่ป้อนข้อมูลผ่านทางส่วนติดต่อผู้ใช้ ก็จะทำให้ทราบรายละเอียดเกี่ยวกับการบรรทุกและจัดส่งของได้ง่ายขึ้น
- 2) ทำให้ผู้ใช้งานทราบวิธีการจัดเรียงสินค้าได้อย่างเป็นระเบียบ โดยคำนึงถึงสถานที่ในการจัดส่งเพื่อเป็นการประหยัดเวลาและค่าใช้จ่าย
- 3) สามารถนำไปประยุกต์ใช้กับองค์กรธุรกิจต่างๆ ได้ เพราะปัจจุบันการขนส่งสินค้าเป็นสิ่งที่สำคัญและจำเป็นในหลายธุรกิจ
- 4) การติดตามการส่งของสามารถทำได้โดยง่าย เนื่องจากทราบเส้นทางที่แน่นอนในการจัดส่งของ

1.5 ขั้นตอนในการดำเนินการ

เอกสาร 1) นี้ รวบรวมความต้องการของผู้ใช้จากแหล่งต่างๆ การศึกษาเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่า 2) วิเคราะห์หาข้อสรุปจากความต้องการที่ได้รวบรวมมาเพื่อนำไปเป็นแนวทางในการสร้างแอปพลิเคชัน

- 3) ศึกษา Bin-Packing Algorithm, Visual Basic, OpenGL และ OpenTK เพื่อใช้ในการสร้างแอปพลิเคชัน
- 4) ออกแบบระบบ ฐานข้อมูล รวมถึงส่วนต่างๆ ของโปรแกรม
- 5) ทำการเขียน โปรแกรม
- 6) ทดสอบโปรแกรมและแก้ไขข้อผิดพลาดที่เกิดขึ้น
- 7) จัดทำเอกสารและคู่มือการใช้งาน
- 8) ส่งมอบงาน

1.6 อุปกรณ์ที่ใช้ในการทำปัญหาพิเศษ

- 1) ฮาร์ดแวร์ (Hardware)
เครื่องคอมพิวเตอร์แบบตั้งโต๊ะ (PC) และคอมพิวเตอร์แบบพกพา (Notebook)
- 2) ซอฟต์แวร์ (Software)
โปรแกรมที่ใช้ในการสร้างแอปพลิเคชัน ได้แก่ Visual Studio เป็นต้น
- 3) ระบบปฏิบัติการ (Operating System)
ระบบปฏิบัติการ Windows 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

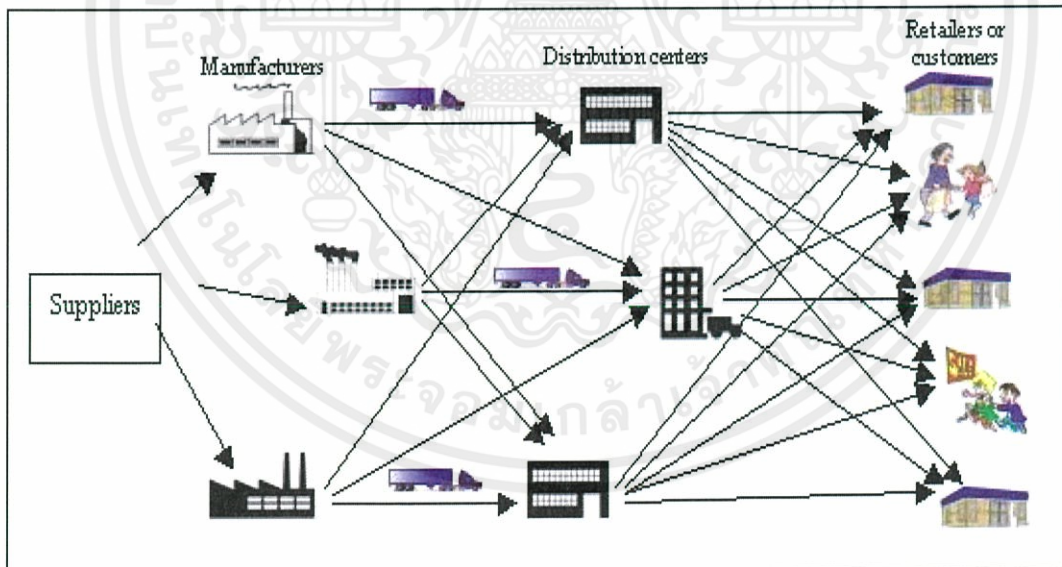
บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

บทนี้จะครอบคลุมพื้นฐานเกี่ยวกับเรื่อง Logistics, Bin-Packing Algorithm, OpenGL และ Visual Basic ซึ่งเป็นแนวทางในการนำอัลกอริทึม Bin-Packing มาประยุกต์ใช้ให้มีความสัมพันธ์กับระบบโลจิสติกส์

2.1 ระบบโลจิสติกส์

โลจิสติกส์ (Logistics) หมายถึง กระบวนการวางแผนการดำเนินการเพื่อควบคุมประสิทธิภาพการไหลเวียนสินค้า/บริการ หรือกิจกรรมที่มีการเคลื่อนย้าย จัดเก็บสินค้า วัสดุ วัตถุดิบ เอกสาร จากที่หนึ่งไปสู่อีกที่หนึ่ง ในระยะเวลาชั่วคราวหรือระยะเวลาสั้น ไปจนถึงจุดที่มีการใช้งานหรือถึงมือผู้บริโภค โดยมีความพยายามในการจัดการกิจกรรมต่างๆ เหล่านี้ ให้เกิดค่าใช้จ่ายโดยรวมต่ำที่สุด และยังมีกิจกรรมอื่นๆ ที่เป็นอีกส่วนของการจัดการ โลจิสติกส์ เช่น การจัดการคลังสินค้า การกระจายสินค้า การจัดการสินค้าคงคลัง การบรรจุภัณฑ์ การรับ-ส่งสินค้า เป็นต้น [1][2]



รูปที่ 2.1 แผนผังกระบวนการทางโลจิสติกส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เห็นถึงความสัมพันธ์ในกิจกรรมต่างๆ ทางด้านโลจิสติกส์ จะแสดงให้เห็นความสัมพันธ์ของการจัดการโลจิสติกส์ กับอัตราส่วนทางการเงิน เนื่องจากแสดงถึงความเชื่อมโยงของการดำเนินธุรกิจได้ค่อนข้างชัดเจน

อัตราส่วนทางการเงิน หมายถึง การเปรียบเทียบระหว่างรายการในงบการเงินตั้งแต่สองรายการขึ้นไป เพื่อแสดงความสัมพันธ์ระหว่างรายการ ซึ่งอาจจะแสดงได้ในรูปของสัดส่วน หรือในรูปของร้อยละ หรือในรูปของจำนวนครั้ง หรือจำนวนรอบ หรือแม้กระทั่งในรูปของระยะเวลา สำหรับอัตราส่วนทางการเงินที่นิยมใช้วัดประสิทธิภาพของการจัดการโลจิสติกส์ สามารถอธิบายได้ด้วย Strategic Profit Model คือ ROI (Return on Investment) เป็นดัชนีที่ใช้วัดศักยภาพของกิจการในการทำกำไรจากการลงทุนหรือใช้ประเมินประสิทธิภาพของการลงทุนในโครงการต่างๆกัน โดยสามารถเขียนในรูปของสมการได้ว่า[3]

$$ROI = \frac{Profit}{Capital\ Employed} \quad (2.1)$$

2.1.1 บทบาทสำคัญของโลจิสติกส์

- 1) ช่วยให้ธุรกิจเกิดประสิทธิภาพและประสิทธิผลในกระบวนการด้านโลจิสติกส์และการจัดการห่วงโซ่อุปทาน
- 2) ทำให้ธุรกิจปรับตัว รองรับการเปลี่ยนแปลงทางการค้า และสามารถเพิ่มกิจกรรมสำคัญในการสนับสนุนการขายสินค้าและบริการ
- 3) ช่วยเพิ่มอรรถประโยชน์ด้านเวลาและสถานที่สำหรับลูกค้า เมื่อต้องการบริโภคหรือนำไปใช้ผลิตด้วยต้นทุนที่ธุรกิจกำหนดไว้
- 4) สามารถนำเทคโนโลยีสารสนเทศและระบบการสื่อสารมาประยุกต์ใช้ เพื่อช่วยลดระยะเวลาการดำเนินงานในขั้นตอนต่างๆ
- 5) สร้างความพึงพอใจสูงสุดแก่ลูกค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 กิจกรรมที่สำคัญของโลจิสติกส์

การดำเนินกิจกรรมต่างๆ นี้ มีเป้าหมายเพื่อตอบสนองความพึงพอใจของลูกค้าด้วยต้นทุนที่เหมาะสม โดยมีกิจกรรมหลักด้าน โลจิสติกส์ 9 กิจกรรม ดังนี้ คือ[4]

- 1) การให้บริการลูกค้าและกิจกรรมสนับสนุนต่างๆ
- 2) การวางแผนจัดซื้อ จัดหาวัตถุดิบ และอุปกรณ์
- 3) การสื่อสารด้าน โลจิสติกส์และกระบวนการสั่งซื้อ
- 4) การดำเนินการ ผลิต บรรจุ และขนส่ง
- 5) การเลือกสถานที่ตั้งของโรงงานและคลังสินค้า
- 6) การวางแผนกำลังการผลิต และการคาดการณ์ปริมาณความต้องการของลูกค้า
- 7) การบริหารจัดการสินค้าคงคลัง
- 8) การบริหารการจัดเก็บ การรวบรวม การกระจายสินค้า และบรรจุหีบห่อ
- 9) กระบวนการ โลจิสติกส์ย้อนกลับ

2.1.3 เป้าหมายที่สำคัญของโลจิสติกส์

- 1) ความรวดเร็วในการส่งมอบสินค้า
- 2) การไหลลื่นของสินค้า
- 3) การไหลลื่นของข้อมูลข่าวสาร
- 4) การสร้างมูลค่าเพิ่มในกิจกรรมที่เกี่ยวข้องกับความต้องการของตลาด
- 5) ลดต้นทุนในส่วนที่เกี่ยวกับการดำเนินการเกี่ยวกับสินค้า การดูแลและขนส่งสินค้า การเพิ่มศักยภาพ และประสิทธิภาพของการแข่งขัน[5]

2.2 Bin Packing

Bin Packing เป็นปัญหาอย่างหนึ่งทางด้านโลจิสติกส์ คือ จะต้องทำการบรรจุสิ่งของลงในคอนเทนเนอร์ เพื่อให้ใช้พื้นที่ในคอนเทนเนอร์ได้คุ้มค่าที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 การกำหนดปัญหา

เพื่อที่จะหาวิธีการแก้ปัญหาสำหรับคอนเทนเนอร์ b หลักการทั่วไป จะเป็นดังนี้

$$\sum_{i \in b} w_i \leq W \quad (2.2)$$

$$\sum_{i \in b} h_i \leq H \quad (2.3)$$

$$\sum_{i \in b} d_i \leq D \quad (2.4)$$

เมื่อ w_i คือ ความกว้างของสิ่งของชั้นที่ i

h_i คือ ความสูงของสิ่งของชั้นที่ i

d_i คือ ความลึกของสิ่งของชั้นที่ i

W คือ ความกว้างของคอนเทนเนอร์

H คือ ความสูงของคอนเทนเนอร์

D คือ ความลึกของคอนเทนเนอร์

จะสรุปได้ว่า

$$\sum_{i \in b} w_i \cdot h_i \cdot d_i \leq W \cdot H \cdot D \quad (2.5)$$

สำหรับแต่ละคอนเทนเนอร์ b เราตั้งใจที่จะลดปริมาณพื้นที่เสีย โดย

$$W \cdot H \cdot D - \sum_{i \in b} w_i \cdot h_i \cdot d_i \quad (2.6)$$

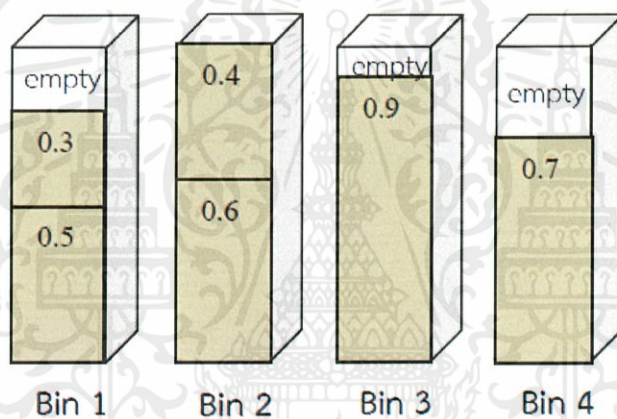
Bin Packing เป็นปัญหา NP-Hard อย่างหนึ่ง ซึ่งปัญหา NP-Hard คือ ปัญหาการตัดสินใจที่มีขนาด
ใหญ่และซับซ้อนมาก (Large-scale and complex decision problem) ยากต่อการหาคำตอบที่ดีที่สุด ในที่นี้จะ
เป็นการทำให้สิ่งของสามารถบรรจุลงในคอนเทนเนอร์ได้ดีที่สุด คือมีพื้นที่เสียน้อยที่สุด วิธีที่นิยมมากที่สุด
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือ วิธีแก้ปัญหาแบบ Heuristic ทางเลือกของสิ่งของที่บรรจุ สามารถทำได้โดยใช้เหตุผลทางตรรกะจาก อัลกอริทึมดังต่อไปนี้

2.2.2 อัลกอริทึม Best Fit (Best Fit Algorithm)

อัลกอริทึม Best Fit จะเป็นอัลกอริทึมที่ใช้บรรจุสิ่งของลงในคอนเทนเนอร์ ซึ่งจะทำให้สิ่งของชิ้นนั้น อยู่ในตำแหน่งที่เหมาะสมและเต็มตู้คอนเทนเนอร์พอดี หลักการ คือ จะบรรจุสิ่งของลงในคอนเทนเนอร์โดย พยายามใส่สิ่งของชิ้นใหม่ลงในคอนเทนเนอร์ให้เหลือพื้นที่น้อยที่สุด

ตัวอย่างเช่น คอนเทนเนอร์มีขนาดคงที่ คือ 1 และมีข้อมูลสิ่งของเป็น $[0.5, 0.3, 0.6, 0.9, 0.4, 0.7]$ จะ บรรจุสิ่งของลงในคอนเทนเนอร์ได้ รูปที่ 2.2



รูปที่ 2.2 การจัดเรียงสิ่งของด้วยอัลกอริทึม Best Fit

สิ่งของขนาด 0.5 , 0.3 จะถูกบรรจุใน Bin 1 แต่ขนาด 0.6 จะบรรจุลงใน Bin 2 เพราะ Bin 1 มีพื้นที่ เหลือไม่พอ จากนั้นสิ่งของขนาด 0.9 จะถูกบรรจุลงใน Bin 3 เพราะ Bin 1 และ Bin 2 มีพื้นที่เหลือไม่พอ เช่นกัน และจะเข้าไปเรื่อยๆ จนกว่าจะบรรจุสิ่งของครบทุกชิ้น

หมายเหตุ: การบรรจุสิ่งของจะเลือกตู้คอนเทนเนอร์ที่มีพื้นที่เหลือน้อยที่สุดและยังสามารถบรรจุสิ่งของนั้นลง ไปได้ ในการบรรจุแต่ละครั้งจึงต้องทำการตรวจสอบคอนเทนเนอร์ทุกตู้ก่อนที่จะบรรจุสิ่งของชิ้นนั้นลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น

Bin คือ การแบ่งที่เก็บสินค้าทั้งหมดออกเป็นส่วนๆ เพื่อทำให้ง่ายต่อการจัดสินค้า และสามารถจัดสินค้าได้ เฉพาะส่วนที่สนใจเท่านั้น

2.3.1 OpenGL คืออะไร

OpenGL (Open Graphics Library) เป็นซอฟต์แวร์ไลบรารี (Software Library) ที่ใช้ติดต่อกับฮาร์ดแวร์เพื่อการแสดงผลกราฟิก โดย OpenGL จะมีคำสั่งสำหรับการวาดภาพพื้นฐาน คือ จุด เส้น และรูปเหลี่ยมต่างๆ และการแสดงผลเรสเตอร์ ซึ่งคำสั่งพื้นฐานมีอยู่ประมาณ 120 คำสั่ง ที่สามารถใช้กำหนดคุณลักษณะและควบคุมการทำงานของแอปพลิเคชัน 3 มิติ ซึ่งผู้พัฒนาโปรแกรมสามารถใช้ไลบรารี OpenGL ได้โดยไม่มีค่าลิขสิทธิ์ ทำให้มีการนำไลบรารีของ OpenGL ไปใช้งานอย่างแพร่หลายในงานกราฟิก

ภาษาที่สนับสนุนสามารถใช้ได้กับ OpenGL มีดังนี้ C/C++ (VC++, Borland C++, C++ Builder, C compiler on UNIX), Delphi, Visual Basic, Java, Perl, Fortran และ Ada เป็นต้น [6]

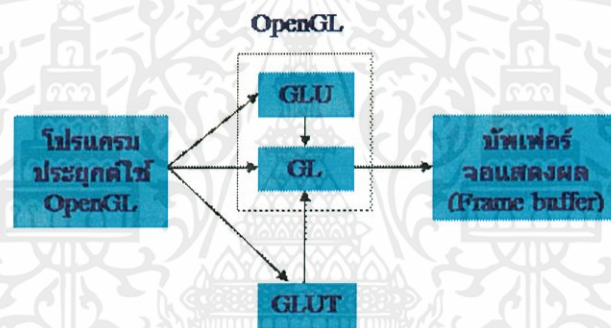
เนื่องจากโครงสร้างของ OpenGL เป็นอินเทอร์เฟซที่เป็นอิสระจากฮาร์ดแวร์ (Hardware-Independent Interface) สามารถใช้ได้กับระบบปฏิบัติการหลายๆ แบบและด้วยเหตุที่ OpenGL ถูกออกแบบให้ทำงานโดยไม่ยึดติดกับระบบปฏิบัติการ สามารถทำงานได้บนทุกๆ แพลตฟอร์ม ทำให้สามารถเคลื่อนย้ายโค้ดโปรแกรมได้โดยง่าย การที่ OpenGL สามารถใช้ได้กับระบบปฏิบัติการที่หลากหลายนี้เอง ทำให้ OpenGL ไม่มีคำสั่งที่จัดการกับระบบปฏิบัติการเลย อีกทั้งยังไม่มีคำสั่งเพื่อรับอินพุตจากผู้ใช้อีกด้วย หน้าที่ทั้งสองอย่างนี้เป็นของผู้เขียนโปรแกรมที่จะต้องออกแบบและเขียนโค้ดเพื่อให้การทำงานเป็นไปอย่างมีประสิทธิภาพ แต่อย่างไรก็ตาม ยังมียูทิลิตี้ที่ช่วยจัดการงานทั้งสองนี้ หากพัฒนาโปรแกรมบนระบบปฏิบัติการแบบ Windows ยูทิลิตี้ดังกล่าวคือ GLUT (OpenGL Utility Toolkit) นอกจากนี้ OpenGL ยังไม่มีคำสั่งระดับสูงที่จะใช้วาดวัตถุ 3 มิติแบบซับซ้อน เช่น รถยนต์ อวัยวะ หรือโมเดลต่างๆ สิ่งที่ OpenGL เตรียมไว้มีเพียงการสร้างรูปจำลอง 3 มิติคือรูปทรงเรขาคณิตอย่างง่าย ซึ่งผู้ใช้งานจะต้องนำรูปทรงเหล่านี้มาประกอบกันเพื่อให้เกิดรูปทรง 3 มิติที่ซับซ้อน

2.3.2 องค์ประกอบของ OpenGL

องค์ประกอบโดยรวมของ OpenGL จะประกอบด้วยไลบรารีกลุ่มใหญ่ๆ อยู่ 3 กลุ่มสำหรับการเรียกใช้งานอันได้แก่ GL, GLU และ GLUT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) Graphics Library (GL) เป็นไลบรารีของ OpenGL ซึ่งฟังก์ชันของ OpenGL จะขึ้นต้นด้วย gl เสมอ
- 2) Graphics Utility Library (GLU) เป็นไลบรารีที่ใช้เฉพาะ GL ฟังก์ชัน แต่ประกอบไปด้วยฟังก์ชันสำหรับการสร้างวัตถุพื้นฐาน เช่น ฟังก์ชันในการสร้างทรงกลม ผู้ใช้ไม่จำเป็นต้องรู้ว่าการสร้างทรงกลมสร้างอย่างไรแต่สามารถเรียกใช้ฟังก์ชันได้เลย GLU เป็นไลบรารีที่มีในทุก OpenGL Implementation
- 3) OpenGL Utility Toolkit (GLUT) คือ ไลบรารีระบบกราฟิก เช่น การติดต่อกับการแสดงผลทางหน้าจอต่างๆ ช่วยให้เข้าใจการทำงานของระบบกราฟิกมากขึ้น GLUT เป็นชุดเครื่องมือที่มีไลบรารีของฟังก์ชันสำหรับการใช้งานกับระบบวินโดวส์ของจอภาพต่างๆไป (ไม่ขึ้นกับระบบใดระบบหนึ่ง) คำสั่งของ GLUT จะขึ้นต้นด้วย glut เสมอ การเรียกใช้จะต้อง Include ไฟล์ Header ที่ชื่อ glut.h ในตอนต้นของโปรแกรม



รูปที่ 2.3 โครงสร้างของ OpenGL

จากรูป โครงสร้างของการใช้งาน OpenGL ผู้ใช้งานจะเป็นผู้เขียนโปรแกรมประยุกต์ซึ่งโดยปกติจะใช้ภาษา C หรือ C++ ภายในโปรแกรมประยุกต์ที่สร้างขึ้นจะต้องรวมเอาไลบรารีของ OpenGL เข้ารวมไปด้วยเพื่อที่จะเรียกใช้งานได้ทางกราฟิก การรวมเอาฟังก์ชันต่างๆ ของ OpenGL เพื่อใช้งานจะกระทำได้โดยเริ่มจากการรวมไฟล์ส่วนหัว (Header file) คือ “GL/glut.h” เข้ากับโปรแกรมที่สร้างขึ้น โปรแกรมที่ได้รวมไฟล์ส่วนหัว glut.h นี้ จะสามารถเรียกใช้ฟังก์ชันต่างๆ ของ OpenGL ได้ ส่วนฟังก์ชันต่างๆ ที่เป็นของ OpenGL จะขึ้นต้นด้วยคำว่า “gl” ฟังก์ชันสำหรับ glu จะขึ้นด้วย “glu” และเช่นเดียวกันสำหรับ glut จากโครงสร้างในรูปที่ 2.3 แสดงให้เห็นว่าฟังก์ชันของ gl จะเป็นแกนหลักและอาจถูกเรียกใช้โดย glu หรือ glut

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 ฟังก์ชันใน OpenGL

ในการเขียนโปรแกรมกราฟิกจะมีลักษณะที่คล้ายๆกันสำหรับ OpenGL ในที่นี้จะใช้วิธีการเขียนโปรแกรมด้วยภาษา C/C++ รูปแบบของโปรแกรมที่ง่ายต่อการเขียนใช้งานจะเป็นในลักษณะของโปรแกรมแบบคอนโซลหรือโปรแกรมที่ยังอิงอยู่กับระบบปฏิบัติการของดอส (DOS) โดยโปรแกรมจะแบ่งออกเป็นชนิดของฟังก์ชันต่างๆ ดังนี้

2.3.3.1 ฟังก์ชันทำงาน

เป็นฟังก์ชันที่ผู้เขียนสร้างขึ้นเพื่อให้ทำงานตามที่ผู้เขียนกำหนด บางฟังก์ชันที่สร้างขึ้นในลักษณะนี้จะเกี่ยวข้องกับกราฟิก ฟังก์ชันการทำงานที่สำคัญมีดังนี้

main : เป็นฟังก์ชันหลักซึ่งเขียนโดยผู้เขียนโปรแกรม มักใช้กำหนดค่าเริ่มต้นต่างๆ และเป็นที่อยู่ของฟังก์ชันสำหรับฟังก์ชันเรียกกลับ (Callback function) และท้ายที่สุดจะต้องตามด้วยฟังก์ชัน `glutMainloop()`; เพื่อที่จะให้โปรแกรมกราฟิกมีการตรวจสอบการทำงานของฟังก์ชันเรียกกลับตลอดเวลา

display : เป็นฟังก์ชันที่ใช้สำหรับการนิยามวัตถุ การแปลงวัตถุและมุมมองของกราฟิก สำหรับการใส่ฟังก์ชัน Display จะต้องประกาศฟังก์ชันเรียกกลับในฟังก์ชัน Main โดยเรียกใช้ฟังก์ชัน `glutDisplayFunc()`; เช่น `glutDisplayFunc(display)`;

reshape : ฟังก์ชันที่เกี่ยวข้องกับการปรับปรุงหรือการที่วินโดวส์มีการเปลี่ยนแปลงขนาด เนื่องจากจากผู้ขยายหรือหน้าต่างต่างของ OpenGL และการฉายภาพจากสามมิติเป็นสองมิติ ในการเรียกใช้ฟังก์ชันจะต้องประกาศฟังก์ชัน `glutReshapeFunc()`; ตัวอย่างเช่น ฟังก์ชันที่จัดการเกี่ยวกับงานดังกล่าว ผู้ใช้ตั้งชื่อว่า reshape การประกาศจะเป็นดังนี้ `glutReshapeFunc(reshape)`; นอกจากนี้แล้วยังมีฟังก์ชันเรียกกลับสำหรับการโต้ตอบกับผู้ใส่เช่น `glutMouseFunc()`, `glutKeyboardFunc()`, `glutIdleFunc()` เป็นต้น

2.3.3.2 ฟังก์ชันเรียกกลับ (Callback function)

เป็นฟังก์ชันที่ใช้สำหรับประกาศฟังก์ชันการทำงาน ฟังก์ชันเรียกกลับเป็นฟังก์ชันของ GLUT และมักจะลงท้ายด้วยคำว่า "Func" เช่น `glutReshapeFunc`, `glutDisplayFunc` ฯลฯ โดยที่พารามิเตอร์ของมันจะเป็นฟังก์ชันการทำงานที่สอดคล้องกับการทำงานของมันที่ควรเป็น โดยฟังก์ชันเรียกกลับนี้ใช้สำหรับตรวจจับ

เหตุการณ์ที่กำหนดใน OpenGL ถ้าเกิดเหตุการณ์ดังกล่าวมุ่งตรงไปยังฟังก์ชันที่สอดคล้องกับเหตุการณ์นั้นๆ เช่น glutMouseFunc(mouse) จะเป็นฟังก์ชันเรียกกลับสำหรับเหตุการณ์การเปลี่ยนแปลงของเมาส์ เช่นถ้าเมาส์มีการกดปุ่ม การควบคุมของโปรแกรมจะตรงไปยังฟังก์ชัน Mouse ซึ่งเป็นชื่อของพารามิเตอร์ของมัน

2.3.3.3 ฟังก์ชันกำหนดการวนรอบ

ได้แก่ฟังก์ชัน glutMainLoop() สำหรับกราฟิกจะเป็นการทำงานที่จะต้องทำการเขียนซ้ำอยู่ตลอดเวลา เพื่อให้ให้วัตถุปรากฏที่หน้าจอ ดังนั้นฟังก์ชัน glutMainLoop() จะทำหน้าที่ในการสร้างรอบวนซ้ำดังกล่าว เหมือนกับว่าฟังก์ชันต่างๆ ถูกกำหนดอยู่ภายใต้กรอบวนของ while(1) ซึ่งมันจะทำงานภายในรอบวนตลอดจนกว่าจะปิดโปรแกรม[7]

2.3.4 ตัวอย่าง OpenGL Code

โปรแกรมตัวอย่างที่ปรากฏด้านล่างนี้เป็นโปรแกรมง่ายๆ ที่ใช้ทั้ง OpenGL ร่วมกับ GLUT ในการแสดง Polygon ให้ปรากฏขึ้นมาใน Window[8]

```

1 #include <GL/glut.h>
2
3 void myDisplay (void)
4 {
5     glClearColor (0.0, 0.0, 0.0, 0.0);
6     glClear (GL_COLOR_BUFFER_BIT);
7     glLoadIdentity ();
8     glOrtho (0.0, 1.0, 0.0, 1.0, -1.0, 1.0);
9     glBegin (GL_POLYGON);
10    glColor3f (1.0, 0.0, 0.0);
11    glVertex3f (0.10, 0.10, 0.0);
12    glColor3f (0.0, 1.0, 0.0);
13    glVertex3f (0.90, 0.10, 0.0);
14    glColor3f (0.0, 0.0, 1.0);
15    glVertex3f (0.90, 0.90, 0.0);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

16  glColor3f(1.0, 0.0, 1.0);
17  glVertex3f(0.10, 0.90, 0.0);
18  glEnd ();
19  glFlush ();
20 }
21
22 main(int argc,char **argv)
23 {
24  glutInit (&argc,argv);
25  glutInitDisplayMode (GLUT_SINGLE | GLUT_RGB);
26  glutCreateWindow ("Simple Program");
27  glutDisplayFunc (myDisplay);
28  glutMainLoop();
29 }

```

#include <GL/glut.h> ที่ปรากฏในตอนต้นของโปรแกรมจะนำ glut.h เข้ามาซึ่งใน glut.h นี้จะ Include ไฟล์ gl.h และ glu.h เข้ามาอีกครั้ง โดยปกติแล้วไฟล์ทั้งสามนี้จะอยู่ภายใต้ /usr/include/GL ดังนั้นในการ Include ไฟล์เหล่านี้ เราต้องใช้ GL นำหน้าชื่อไฟล์ด้วย ซึ่งนั่นก็คือ #include <GL/gl.h> เป็นต้น โปรแกรมส่วนใหญ่ที่ใช้ OpenGL จะต้องมีการ Include ไฟล์ gl.h นี้ ส่วนโปรแกรมที่ใช้ GLU จะมีการ Include ไฟล์ glu.h โปรแกรมที่มีการใช้ GLUT ในการควบคุม Windows จะ Include ไฟล์ glut.h คำสั่งของ OpenGL, GLU และ GLUT จะขึ้นต้นคำสั่งด้วยตัวอักษร gl, glu และ glut ตามลำดับ ตัวอักษรถัดไปในคำสั่งจะตามด้วยตัวอักษร ในลักษณะที่ใช้ตัวอักษรตัวแรกในแต่ละคำเป็นตัวอักษรตัวใหญ่ขึ้นต้น เช่น glColor(), gluOrtho2D() และ glutInitDisplayMode() เป็นต้น

```

main(int argc,char **argv)
{
  glutInit (&argc,argv);
  glutInitDisplayMode (GLUT_SINGLE | GLUT_RGB);
  glutCreateWindow ("Simple Program");
  glutDisplayFunc (myDisplay);

```

```
glutMainLoop();
}
```

คำสั่งใน main() จะทำการสร้าง Window ขึ้นมาบนจอภาพเพื่อให้ OpenGL ทำการ Render วัตถุลงไปบน Window นั้น คำสั่งทั้งหมดใน main() เป็นคำสั่งที่สำคัญในการเริ่มต้นการทำงานของ Window

```
glutInit (&argc,argv);
```

เป็นคำสั่งที่ให้ GLUT เริ่มทำการ Initialize และทำการ Process ค่าที่ได้รับมาจาก Command Line Arguments ซึ่ง glutInit() นี้ควรถูกเรียกก่อน GLUT function อื่นๆ

```
glutInitDisplayMode (GLUT_SINGLE | GLUT_RGB);
```

เป็นการกำหนดรูปแบบการแสดงผลของโปรแกรมว่าโปรแกรมจะใช้ Color model แบบใด เช่น RGB, RGBA หรือ Color index เป็นต้น และยังสามารถใช้คำสั่งนี้ในการกำหนดว่า Window ที่จะสร้างขึ้นเป็นแบบ Single-buffering หรือ Double-buffering ได้อีกด้วย คำสั่งนี้สามารถใช้สร้าง Buffer อื่นๆ เสริมอีกเพื่อใช้ประโยชน์ในด้านอื่นๆ อีก อาทิเช่น depth buffer, stencil buffer และ accumulation buffer เป็นต้น ดังนั้นถ้าหากต้องการสร้าง Window ที่เป็น Double buffering และใช้ RGBA color model พร้อมกับ Depth buffer จะต้องออกคำสั่งดังนี้

```
glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA | GLUT_DEPTH);
glutCreateWindow ("Simple Program");
```

เป็นคำสั่งที่ใช้ในการสร้าง Window ที่ผูกเอา OpenGL context ไว้ซึ่ง OpenGL context เป็นส่วนที่จำเป็นเพื่อให้ OpenGL ใช้ในการ Render ผลภาพให้ปรากฏอยู่บน OpenGL context นี้ คำสั่ง glutCreateWindow() นี้จะ Return ค่า Unique identifier ที่เป็นค่า Integer เพื่อใช้ในการอ้างอิงถึง Window ที่ถูกสร้างขึ้น อย่างไรก็ตาม Window ใหม่นี้จะยังไม่ปรากฏขึ้นจนกว่า glutMainLoop() จะถูกเรียกใช้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
glutDisplayFunc (myDisplay);
```

เป็นคำสั่งที่ใช้ในการกำหนด Display callback function โดยที่ Callback function นี้จะถูก GLUT เรียกใช้เมื่อ Content ภายใน Window จำเป็นต้องมีการ Display ใหม่อีกครั้ง ส่วนของ Function ที่จะมาถูกกำหนดให้ GLUT รับรู้จะต้องเป็น function ไม่มีการส่งกลับค่า หรือชนิดของข้อมูลที่ส่งกลับเป็น void นั่นเอง และ Function นี้ไม่มีการรับ Parameter ใดๆ ดังตัวอย่างเช่น

```
void myDisplay (void);
```

ในบางครั้งต้องการให้มีการ Redisplay content บน Window ได้ตามต้องการ ซึ่งสามารถทำได้โดยการเรียกใช้ glutPostRedisplay() ซึ่งจะส่ง Redisplay event ไปยัง Window ทำให้มีการเรียกใช้ Display callback อีกครั้งหนึ่ง

```
glutMainLoop();
```

เป็นคำสั่งที่ต้องถูกเรียกเป็นคำสั่งท้ายสุด เพื่อให้ระบบ Window ที่ GLUT สร้างขึ้นมาได้เริ่มทำงานเพื่อประมวลผล Window event ต่างๆ ที่เกิดขึ้นมา เมื่อ glutMainLoop() ถูกเรียกใช้ Window ที่ได้ถูกเตรียมสร้างเอาไว้ด้วยคำสั่ง GLUT ต่างๆ ก่อนหน้านี้จะถูกสร้างและปรากฏขึ้นบนหน้าจอ หลังจากนั้น Display callback function จะถูกเรียกใช้งาน และตามด้วยการประมวลผล Window event อื่นๆ ที่รออยู่ใน Queue โดยปกติเมื่อ glutMainLoop() ถูกเรียกเข้าไปแล้ว โปรแกรมจะไม่หลุดออกมาจาก Loop นั้นอีก นอกจากคำสั่งของ GLUT ที่ได้เห็นผ่านมาแล้ว ยังมีคำสั่ง GLUT คำสั่งอื่นๆ อีกเช่น คำสั่งที่สามารถนำมาใช้ควบคุมในการกำหนดตำแหน่งและขนาดของ Window ดังนี้

```
glutInitWindowPosition(x,y);
glutInitWindowSize(width,height);
```

โดยที่ x และ y จะเป็นตัวกำหนดพิกัดของตำแหน่งมุมซ้ายบนของ Window และ Width กับ Height จะใช้กำหนดขนาดของ Window

ส่วน Display callback function ของโปรแกรมนี้อีกคือ myDisplay() ซึ่งคำสั่งต่างๆ ที่ปรากฏใน myDisplay() นี้เป็นคำสั่งของ OpenGL ทั้งหมด โดยสังเกตเห็นได้ว่าทุกคำสั่งจะมีตัวอักษร gl นำหน้าคำสั่ง เช่น glColor3f() เป็นต้น ตัวอย่างเช่น

```
glClearColor (0.0, 0.0, 0.0, 0.0);
```

เป็นคำสั่งที่ใช้กำหนด Background color ซึ่งในตัวอย่างนี้ได้กำหนดให้เป็นสีดำ คำสั่ง glColor3f() ใช้สำหรับกำหนดสีที่จะใช้ในการ Clear window ซึ่งคำสั่ง

```
glClear (GL_COLOR_BUFFER_BIT);
```

เป็นคำสั่งที่ใช้ในการ Clear window อีกครั้งหนึ่ง

```
glOrtho (0.0, 1.0, 0.0, 1.0, -1.0, 1.0);
```

ใช้กำหนด Coordinate system เพื่อให้ OpenGL รับรู้ถึง View volume หรือ View frustum ซึ่งเป็นปริมาตรที่กำหนดบริเวณในการ Render วัตถุออกมา ซึ่งด้วยคำสั่ง glOrtho() นี้จะใช้การ Map ปริมาตรนี้ให้สัมพันธ์กับบริเวณใน Window ในลักษณะของ Orthographic projection โดยที่ Coordinate ที่ถูกส่งไปยัง glOrtho() เป็นดังนี้

```
glOrtho (left, right, bottom, top, near, far);
```

กลุ่มของคำสั่งต่อไปนี้ใช้ร่วมกันเพื่อทำการสร้างรูปสี่เหลี่ยมให้ปรากฏขึ้น

```
glBegin (GL_POLYGON);
```

```
glColor3f (1.0, 0.0, 0.0);
```

```
glVertex3f (0.10, 0.10, 0.0);
```

```
glColor3f (0.0, 1.0, 0.0);
```

```
glVertex3f (0.90, 0.10, 0.0);
```

```
glColor3f (0.0, 0.0, 1.0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูง และต้องอภัยถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
glVertex3f (0.90, 0.90, 0.0);
glColor3f (1.0, 0.0, 1.0);
glVertex3f (0.10, 0.90, 0.0);
glEnd ();
```

ซึ่งจะเห็นได้ว่ากลุ่มของคำสั่ง glColor3f() และ glVertex3f() ถูกล้อมรอบอยู่ด้วยคำสั่ง glBegin() และ glEnd() ซึ่ง glBegin() และ glEnd() ใช้ในการกำหนดรูปแบบของ Primitive ที่จะถูกสร้างขึ้นมา ในที่นี้หากต้องการสร้างรูปหลายเหลี่ยม (Polygon) ที่เป็นรูปสี่เหลี่ยมซึ่งประกอบด้วย 4 Vertices จะต้องใช้ glColor3f() ในการกำหนดค่าสีให้กับจุด Vertex ซึ่งตำแหน่งพิกัดของ Vertex นี้จะถูกกำหนดด้วยคำสั่ง glVertex3f() ที่อยู่ถัดลงมา ลักษณะของโปรแกรมด้านบนนี้ สามารถกำหนดค่าของสีแดง (จากคำสั่ง glColor3f (1.0, 0.0, 0.0)) สำหรับพิกัดมุมแรกของรูปหลายเหลี่ยม และสีเขียว สีน้ำเงิน และสีม่วง สำหรับพิกัดมุมที่เหลือของรูปหลายเหลี่ยม นี้ตามลำดับ

```
glFlush ();
```

ใช้สั่ง OpenGL ให้ทำการ Execute คำสั่ง OpenGL ที่ได้เรียกมาก่อนหน้านี้แล้วแต่บางครั้งคำสั่งนี้ยังถูกเก็บไว้ใน Buffer เพื่อรอการส่งไปยังอุปกรณ์ในภายหลัง คำสั่ง glFlush() จะบังคับให้ OpenGL ทำการ Execute คำสั่งที่อาจยังค้างอยู่ใน Buffer เพื่อที่จะได้ถูกสั่งให้ทำงานให้ปรากฏผลขึ้น ถ้าจะต้องการความมั่นใจว่าคำสั่งที่เคียงอยู่ใน Buffer ได้ทำงานเสร็จสิ้นลงอย่างสมบูรณ์ แล้วสามารถใช้คำสั่ง

```
glFinish ();
```

ซึ่งทำงานคล้ายคำสั่ง glFlush() แต่ glFinish() จะรอค้างอยู่นานกว่าคำสั่ง OpenGL ที่ยังค้างอยู่ได้ถูกทำงานเสร็จสิ้นลงแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 OpenTK

OpenTK หรือ The Open Toolkit เป็นเครื่องมือที่ใช้ ใน C# Library ใน OpenGL, OpenCL, OpenAL เหมาะสำหรับ เกมส์ แอปพลิเคชันทางวิทยาศาสตร์ และโปรเจกต์ต่างๆ ที่ต้องใช้งานด้านกราฟิก 3D เสียข หรือ ด้านคำนวณ โดยรองรับได้หลายระบบปฏิบัติการสามารถรองรับรุ่น 32 และ 64 บิต ของ Windows, Linux และ Mac OS X ใช้ได้ทุกภาษา เช่น .Net language: C#, VB.Net, C++/CLI, F#, Boo [9]

ตารางที่ 2.1 เปรียบเทียบฟังก์ชันของ OpenGL และ OpenTK

OpenGL	OpenTK
<code>glPushMatrix();</code>	<code>GL.PushMatrix()</code>
<code>glPopMatrix();</code>	<code>GL.PopMatrix()</code>
<code>glVertex3f(x,y,z)</code>	<code>GL.Vertex3(x,y,z)</code>
<code>glScaled(x,y,z);</code>	<code>GL.Scale(x,y,z)</code>
<code>glTranslated(x,y,z);</code>	<code>GL.Translate(x,y,z)</code>
<code>glClear(GL_COLOR_BUFFER_BIT GL_DEPTH_BUFFER_BIT);</code>	<code>GL.Clear(ClearBufferMask.ColorBufferBit)</code> <code>GL.Clear(ClearBufferMask.DepthBufferBit)</code>
<code>glLoadIdentity();</code>	<code>GL.LoadIdentity()</code>

2.4.1 ตัวอย่าง OpenTK Code

```
Imports OpenTK
Imports OpenTK.Platform
Imports OpenTK.Graphics.OpenGL
```

```
Public Class aoGLControl
```

```
Inherits GLControl
```

```
Dim _STARTED As Boolean = False
```

```

' Background Color
Private gbackcolor As Color
Public Property glBackColorProp() As Color
    Get
        Return gbackcolor
    End Get
    Set(ByVal value As Color)
        gbackcolor = value
    End Set
End Property

Private Sub aoGLControl_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Me.Load
    _STARTED = True
    GL.ClearColor(gbackcolor)
    SetupViewport()
End Sub

Private Sub aoGLControl_Paint(ByVal sender As Object, ByVal e As
System.Windows.Forms.PaintEventArgs) Handles Me.Paint
    If Not _STARTED Then Return
    render()
End Sub

Private Sub ResizeGL()
    SetupViewport()
    Me.Invalidate()
End Sub

Private Sub SetupViewport()
    Dim w As Integer = Me.Width
    Dim h As Integer = Me.Height

```

ออก Private Sub SetupViewport() สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าจะ Dim w As Integer = Me.Width ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
GL.MatrixMode(MatrixMode.Projection)
```

```
GL.LoadIdentity()
```

```
GL.Ortho(0, w, 0, h, -1, 1)
```

```
GL.Viewport(0, 0, w, h)
```

```
End Sub
```

```
Private Sub render()
```

```
' Clear screen
```

```
GL.Clear(ClearBufferMask.ColorBufferBit)
```

```
GL.Clear(ClearBufferMask.DepthBufferBit)
```

```
' Here Pass the data of what it is to be drawn
```

```
DrawLines()
```

```
' Draw it!!!!
```

```
Me.SwapBuffers()
```

```
End Sub
```

```
Private Sub DrawLines()
```

```
GL.MatrixMode(MatrixMode.Modelview)
```

```
GL.LoadIdentity()
```

```
' Add Geometry with above Translation/Rotation/Color
```

```
GL.Begin(BeginMode.Triangles) 'Depending on type of object
```

```
.
```

```
.
```

```
.
```

```
.
```

```
.
```

```
.
```

```
GL.End()
```

```
End Sub
```

```
End Class
```

เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 Visual Basic

2.5.1 Visual Basic คืออะไร

Visual Basic หรือเรียกว่า VB เป็นภาษาโปรแกรมภาษาหนึ่ง พัฒนาโดยบริษัทไมโครซอฟต์ บรรจู่อยู่ภายใต้ผลิตภัณฑ์ชุดหนึ่งชื่อว่า Microsoft Visual Studio

Visual เป็นส่วนที่หมายถึงเมธอดในการติดต่อแบบ Graphical User Interface (GUI) ซึ่งการสร้างทำได้โดยการเพิ่มออบเจ็กต์ ลงบนฟอร์มที่ทำหน้าที่ติดต่อกับผู้ใช้งานจอภาพ ส่วน Basic เป็นส่วนที่หมายถึงภาษา BASIC (Beginners ALL Purpose Symbolic Instruction Code) โดย Visual Basic ได้เปลี่ยนแปลงจากภาษา BASIC ดั้งเดิม ด้วยการเพิ่มประโยคคำสั่ง ฟังก์ชัน และกีย์เวิร์ด ที่เกี่ยวข้องโดยตรงกับ GUI

2.5.2 แนวคิดของ Visual Basic

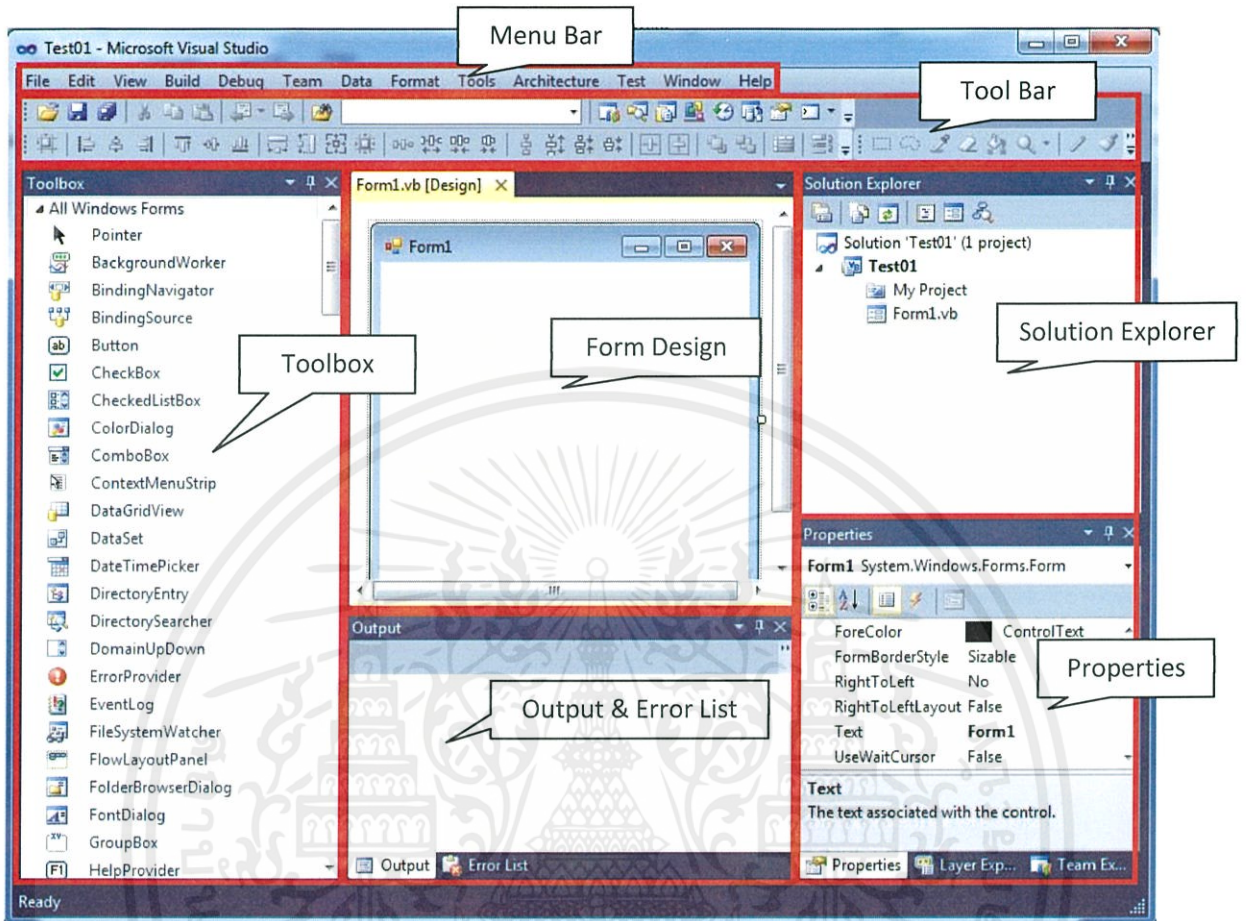
Visual Basic เป็นการพัฒนาในสภาพแวดล้อมของ Windows ซึ่งแนวคิดพื้นฐานในการทำงานของระบบ Windows ที่สำคัญมี 3 ประการ คือ Window, Events และ ข่าวสาร (Message) โปรแกรมประยุกต์ Visual Basic มีการทำงานแบบ Event-Driven ที่เป็นการประมวลผลตามคำสั่งในแต่ละส่วนเพื่อตอบสนองต่อ Event ซึ่ง Event เหล่านี้สามารถเปลี่ยนโดยการทำงานของผู้ใช้ ข่าวสารของระบบหรือโปรแกรมประยุกต์อื่นหรือภายในโปรแกรมเดียวกัน ลำดับการทำงานของ Event จะจัดลำดับโดยจากการประมวลคำสั่ง

2.5.3 วัตถุประสงค์ของ Visual Basic

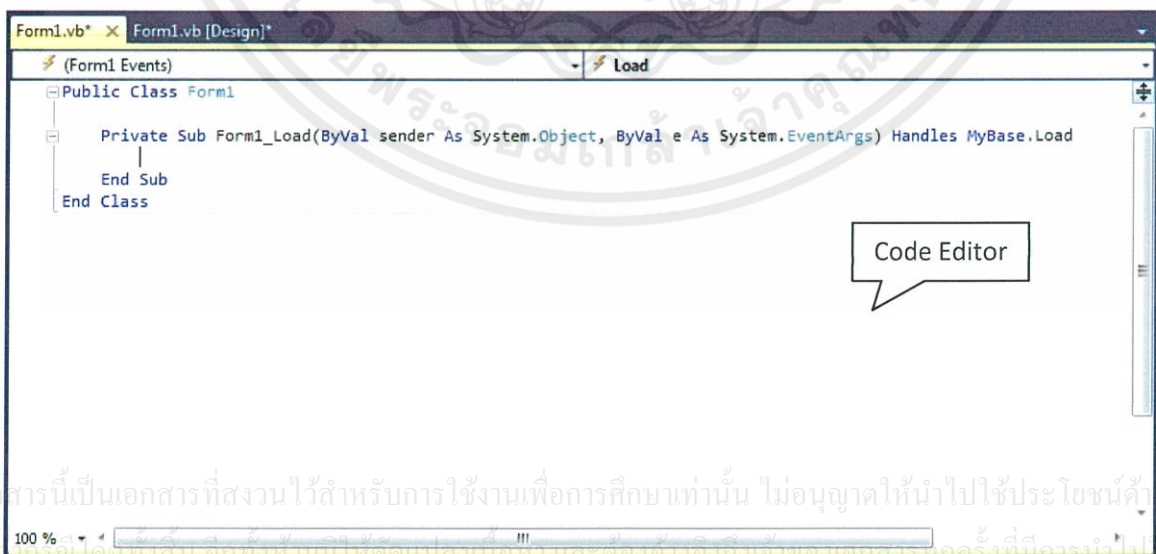
วัตถุประสงค์ คือ ต้องการให้เป็นภาษาที่ใช้สำหรับพัฒนาโปรแกรมหรือแอปพลิเคชัน ที่ทำงานภายใต้ระบบปฏิบัติการ Windows และ Windows NT การใช้งาน Visual Basic นั้นค่อนข้างง่าย โปรแกรมเมอร์สามารถวาดและวางองค์ประกอบต่างๆ บนหน้าจอเพื่อติดต่อกับผู้ใช้ (User Interface) ได้ตามต้องการ เมื่อวาดหน้าจอได้เสร็จก็เขียน โปรแกรมซึ่งเป็นลักษณะแบบมีโครงสร้างทางภาษาค้ายคลึงกับภาษาอังกฤษ เพื่อเป็นการเชื่อมโยงความสัมพันธ์ขององค์ประกอบแต่ละตัวบนหน้าจอเข้าด้วยกัน ให้ทำงานอย่างสัมพันธ์กัน ตามที่โปรแกรมเมอร์ต้องการ ตามหลักการเชิงวัตถุ (Object-Oriented)[10]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4 Visual Basic in Visual Studio 2010



รูปที่ 2.4 หน้าการสร้างฟอร์มด้วยโปรแกรม Visual Studio 2010



รูปที่ 2.5 หน้าจอการเขียนโค้ดด้วยโปรแกรม Visual Studio 2010

จากรูปที่ 2.4 และ 2.5 จะขออธิบายรายละเอียดของแต่ละส่วนดังนี้

- 1) Menu Bar เป็นเมนูที่รวบรวมคำสั่งควบคุมการทำงานของ Visual Studio 2010 โดยจัดเป็นกลุ่มคำสั่งแยกตามประเภทการใช้งานเช่น
 - File คือ กลุ่มคำสั่งเกี่ยวกับการสร้างโปรเจกใหม่ เปิดโปรเจก บันทึกโปรเจก และปิดโปรเจก
 - Edit คือ กลุ่มคำสั่งในการสร้าง/แก้ไขการทำงานของโปรเจก
 - View คือ กลุ่มคำสั่งเกี่ยวกับการแสดงผล
- 2) Tool Bar เป็นเครื่องมือที่รวบรวมคำสั่งที่มีการใช้งานบ่อย เพื่อให้เรียกใช้งานคำสั่งในเมนูบาร์ได้สะดวกมากยิ่งขึ้น เช่น
 - New Project คือ การสร้างโปรเจกใหม่
 - Open File คือ การเปิดไฟล์ที่มีอยู่แล้ว
 - Save คือ บันทึกไฟล์ที่กำลังทำงานอยู่
 - Save All คือ บันทึกไฟล์ทั้งหมดที่มีการเปิดทำงานอยู่
- 3) Toolbox เป็นหน้าต่างที่แสดงคอนโทรลและคอมโพเนนต์ต่างๆ เพื่อสะดวกในการสร้างแอปพลิเคชัน โดยจะมีการแบ่งออกเป็นกลุ่มตามลักษณะการใช้งาน เช่น
 - Common Controls เป็นคอนโทรลที่มีใช้บ่อย จะประกอบด้วย Button, Textbox, Label, Combo Box
- 4) Form Design เป็นหน้าต่างที่ใช้สำหรับออกแบบหน้าจอของแอปพลิเคชัน โดยการลากคอนโทรลต่างๆ จาก Toolbox มาวางบนฟอร์ม Visual Basic จะทำการสร้าง code การออกแบบให้โดยอัตโนมัติ
- 5) Solution Explorer เป็นหน้าต่างแสดงรายการของ Item ที่มีอยู่ในโปรเจกทั้งหมด เช่น ฟอร์ม โมดูล คอมโพเนนต์ และคลาส
- 6) Properties เป็นหน้าต่างแสดงและกำหนดคุณสมบัติ หรือลักษณะเบื้องต้นต่างๆ ของคอนโทรลและออบเจกต์ เช่น ชื่อ สี ขนาด เป็นต้น
- 7) Output & Error List เป็นหน้าต่างที่แสดง Output และข้อผิดพลาดทั้งหมดที่เกิดขึ้นภายในโปรเจก ทั้งข้อผิดพลาดจากการเขียนโปรแกรม ข้อผิดพลาดจากการติดต่อฐานข้อมูล และข้อผิดพลาดทั้งหมด จะแสดงที่หน้าต่างนี้
- 8) Code Editor เป็นหน้าต่างที่ใช้สำหรับเขียนโค้ดควบคุมการทำงานของแอปพลิเคชัน

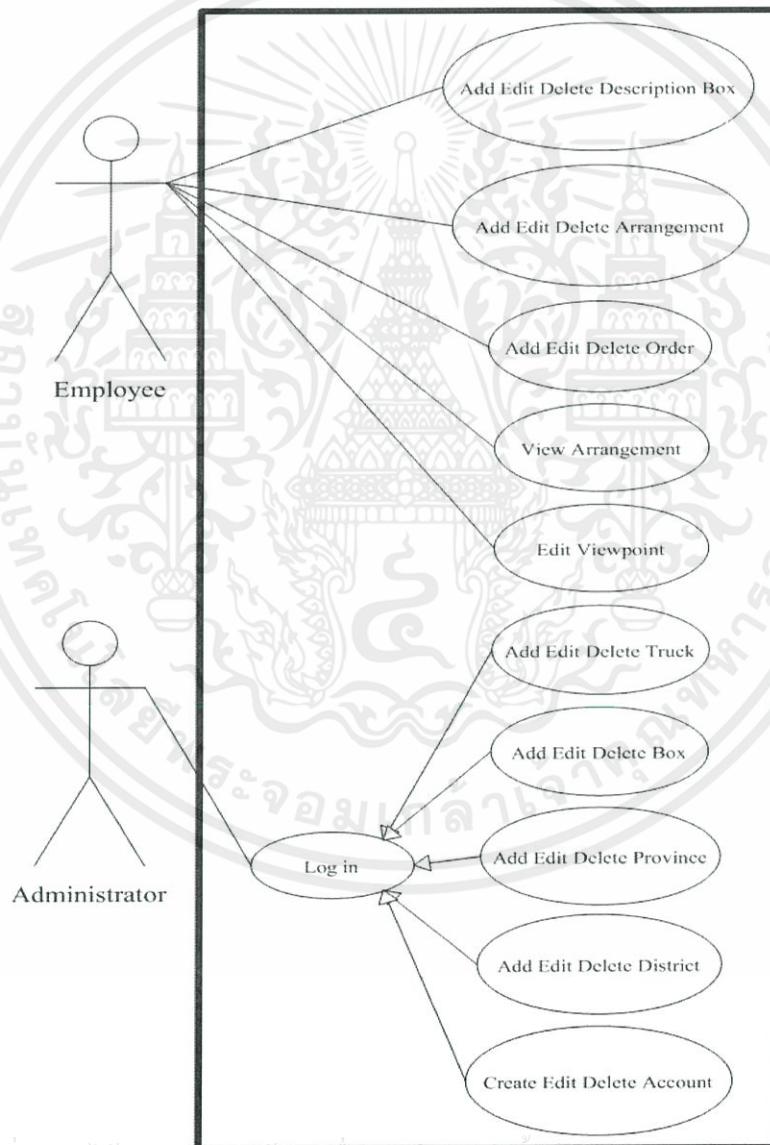
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีการดำเนินงาน

ในบทนี้จะครอบคลุมการออกแบบภายในระบบ โดยมีการออกแบบ Activity Diagram, Use Case Diagram, ER Diagram และ User Interface ซึ่งการออกแบบทั้งหมดนี้จะอธิบายถึงหลักการทำงานของระบบทั้งหมด

3.1 Use Case Diagram



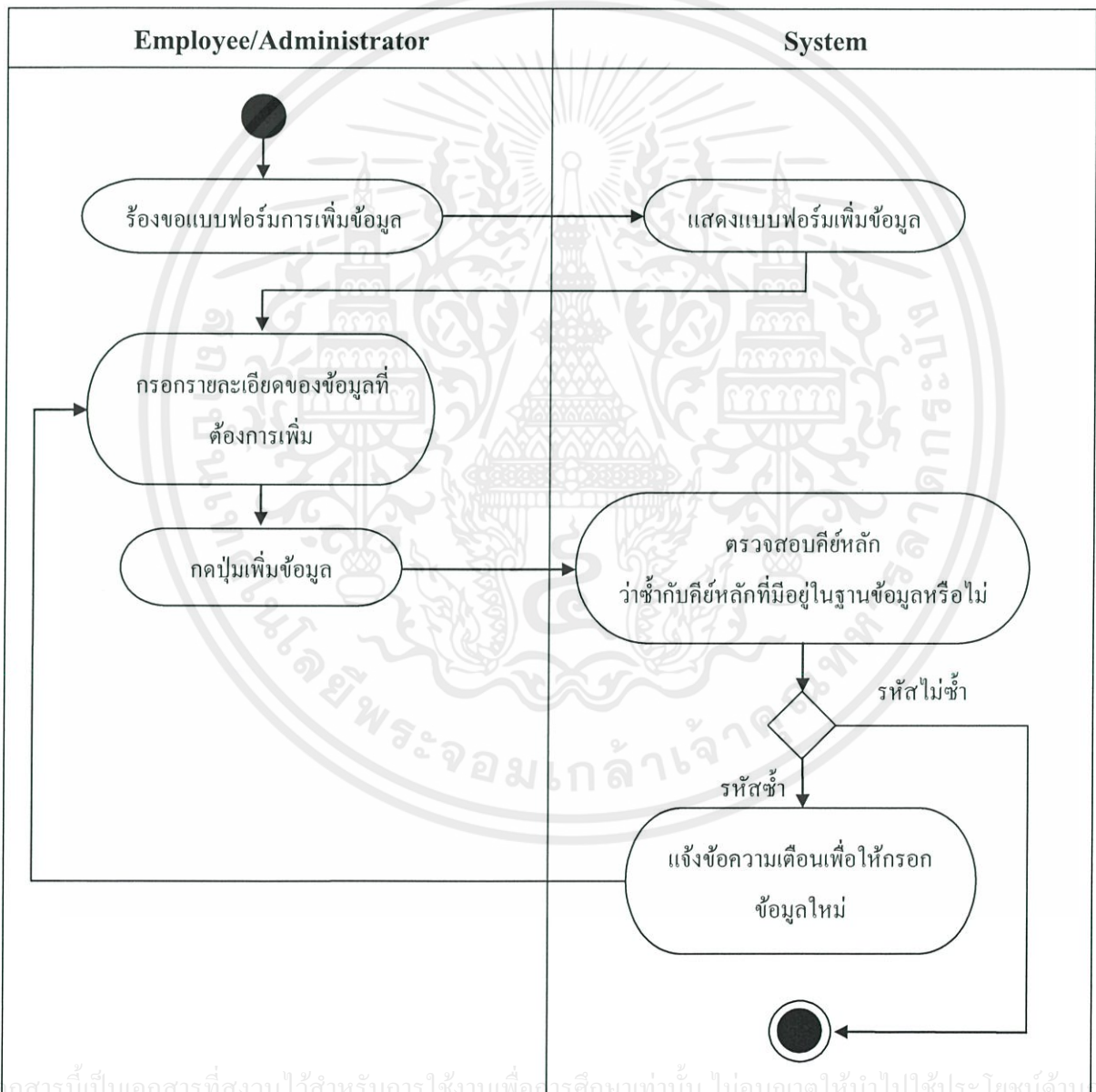
รูปที่ 3.1 Use Case Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.1 แสดงให้เห็นว่าพนักงานทั่วไปสามารถเพิ่ม แก้ไข หรือลบข้อมูลของรายละเอียดสินค้า ออร์เดอร์ และการจัดเรียงสินค้า และสามารถดูวิธีจัดเรียงสินค้าหรือเปลี่ยนมุมมองการดูได้ ส่วนของผู้ดูแล สามารถจัดการระบบได้เหมือนพนักงานทั่วไป และสามารถเพิ่ม แก้ไข หรือลบข้อมูลของรถ จังหวัด อำเภอ คล่อง และสร้างบัญชีผู้ใช้ภายในระบบได้

3.2 Activity Diagram

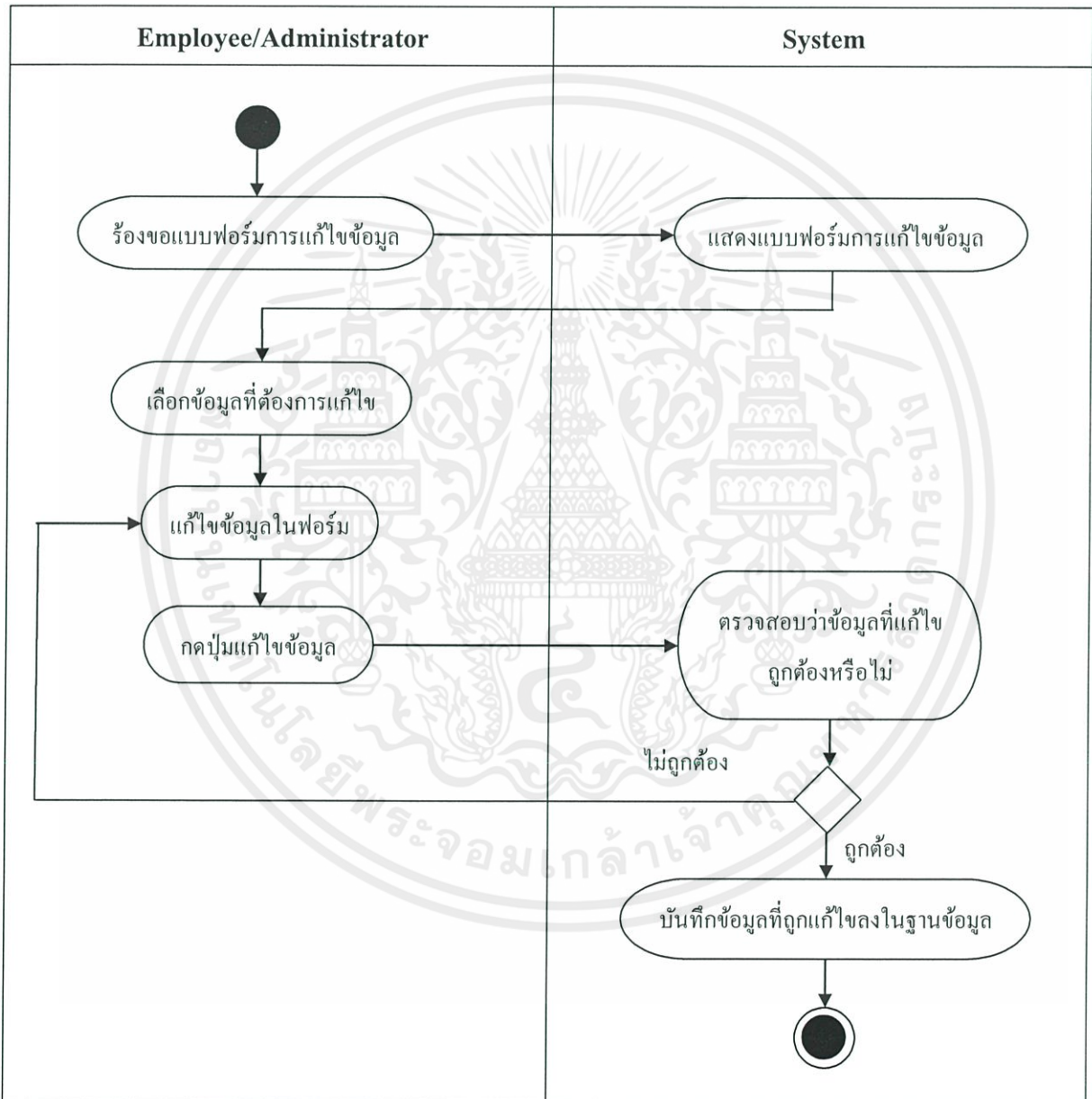
3.2.1 Activity Diagram สำหรับการเพิ่มข้อมูล



รูปที่ 3.2 Activity Diagram สำหรับการเพิ่มข้อมูล

จากรูปที่ 3.2 แสดงถึงการเพิ่มข้อมูลในฟอร์มต่างๆ โดยผู้ใช้จะร้องขอเข้าสู่ฟอร์มทำการเพิ่มข้อมูล โปรแกรมจะแสดงฟอร์มที่ร้องขอจากผู้ใช้ จากนั้นผู้ใช้ทำการป้อนข้อมูลให้ถูกต้องและครบถ้วนพร้อมทั้งกดปุ่มเพิ่มข้อมูล หากข้อมูลถูกต้อง โปรแกรมจะทำการบันทึกข้อมูลลงฐานข้อมูล หากข้อมูลไม่ถูกต้อง โปรแกรมจะแสดงข้อความแจ้งเตือนให้ผู้ใช้รับทราบ

3.2.2 Activity Diagram สำหรับการแก้ไขข้อมูล



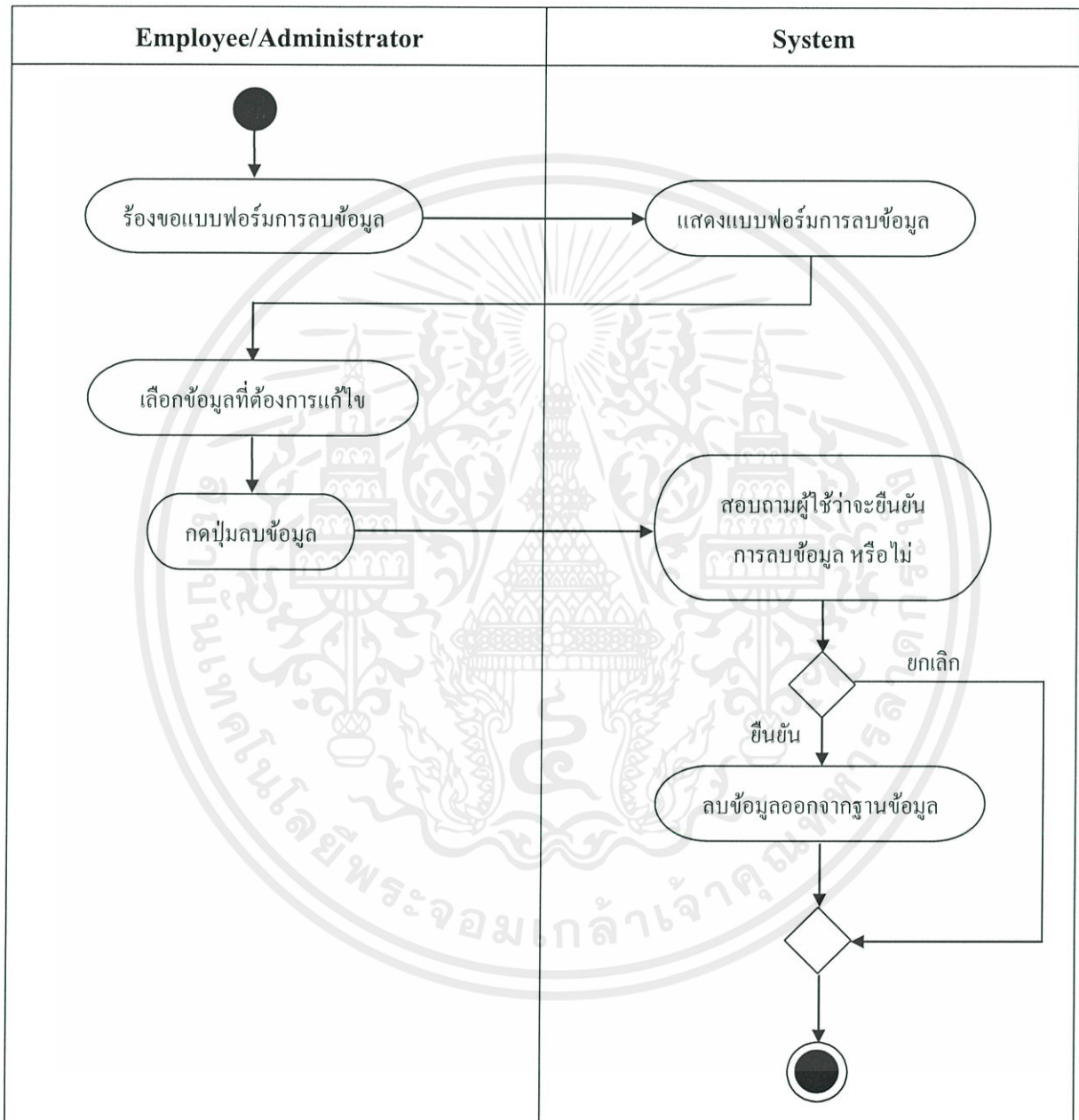
รูปที่ 3.3 Activity Diagram สำหรับการแก้ไขข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

จากรูปที่ 3.3 แสดงถึงการแก้ไขข้อมูลในฟอร์มต่างๆ โดยผู้ใช้จะร้องขอเข้าสู่ฟอร์มทำการแก้ไขข้อมูล โปรแกรมจะแสดงฟอร์มที่ร้องขอจากผู้ใช้ จากนั้นผู้ใช้ทำการแก้ไขข้อมูลที่ต้องการพร้อมทั้งกดปุ่ม

แก้ไขข้อมูล หากข้อมูลถูกต้อง โปรแกรมจะทำการบันทึกข้อมูลลงฐานข้อมูล หากข้อมูลไม่ถูกต้อง โปรแกรมจะแสดงข้อความแจ้งเตือนให้ผู้ใช้รับทราบ

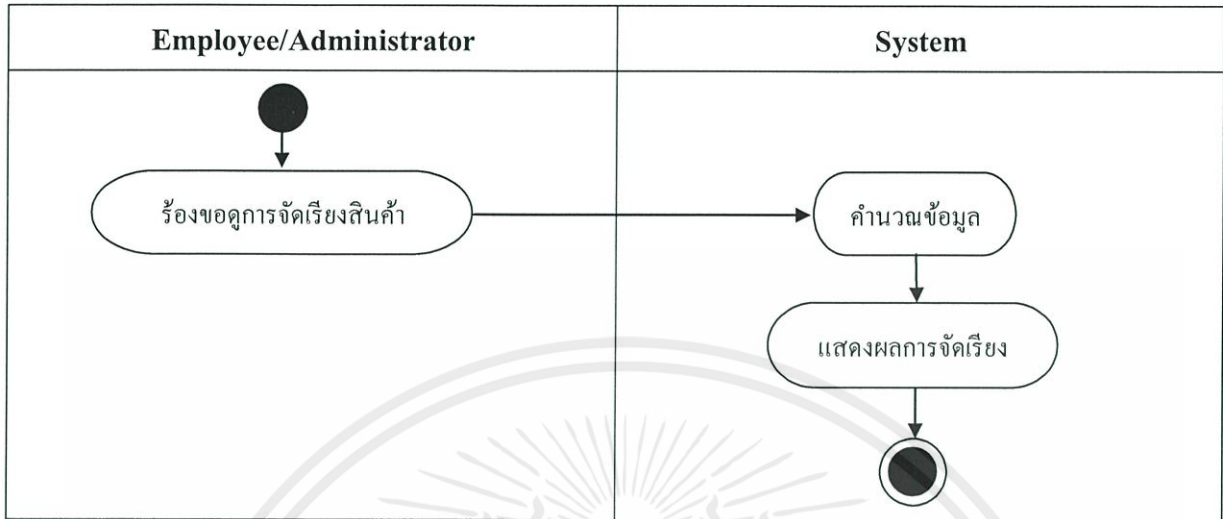
3.2.3 Activity Diagram สำหรับการลบข้อมูล



รูปที่ 3.4 Activity Diagram สำหรับการลบข้อมูล

เอกสารนี้เป็นเอกสารต้นฉบับที่จัดทำขึ้นโดยระบบอัตโนมัติของโครงการ
 จะแสดงฟอร์มที่ร้องขอจากผู้ใช้ จากนั้นผู้ใช้เลือกข้อมูลที่ต้องการลบพร้อมทั้งกดปุ่มลบข้อมูล โปรแกรมจะ
 แสดงข้อความเพื่อยืนยันการลบข้อมูล หากผู้ใช้กดยืนยัน โปรแกรมจะทำการลบข้อมูลออกจากฐานข้อมูล

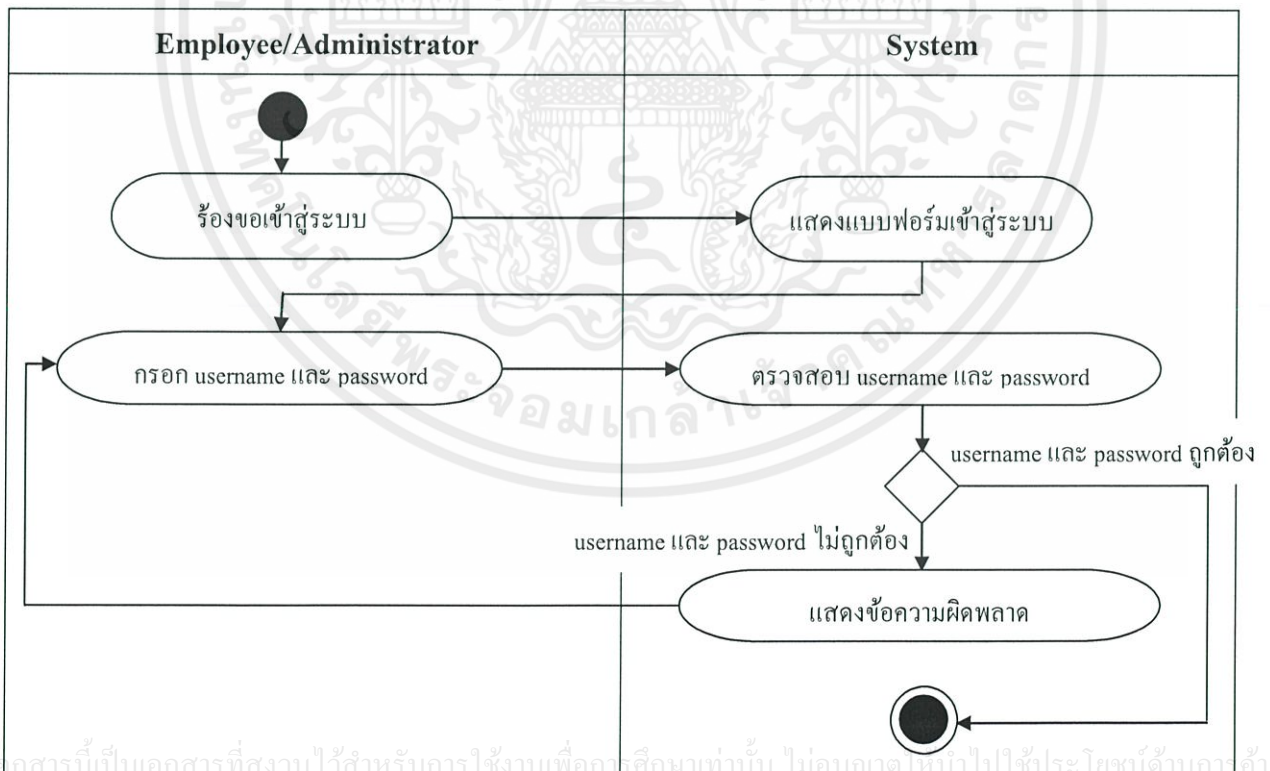
3.2.4 Activity Diagram สำหรับดูการจัดเรียงสินค้า



รูปที่ 3.5 Activity Diagram สำหรับดูการจัดเรียงสินค้า

จากรูปที่ 3.5 เมื่อผู้ใช้ร้องขอการดูการจัดเรียงสินค้า โปรแกรมจะคำนวณข้อมูล แล้วแสดงผลการจัดเรียง

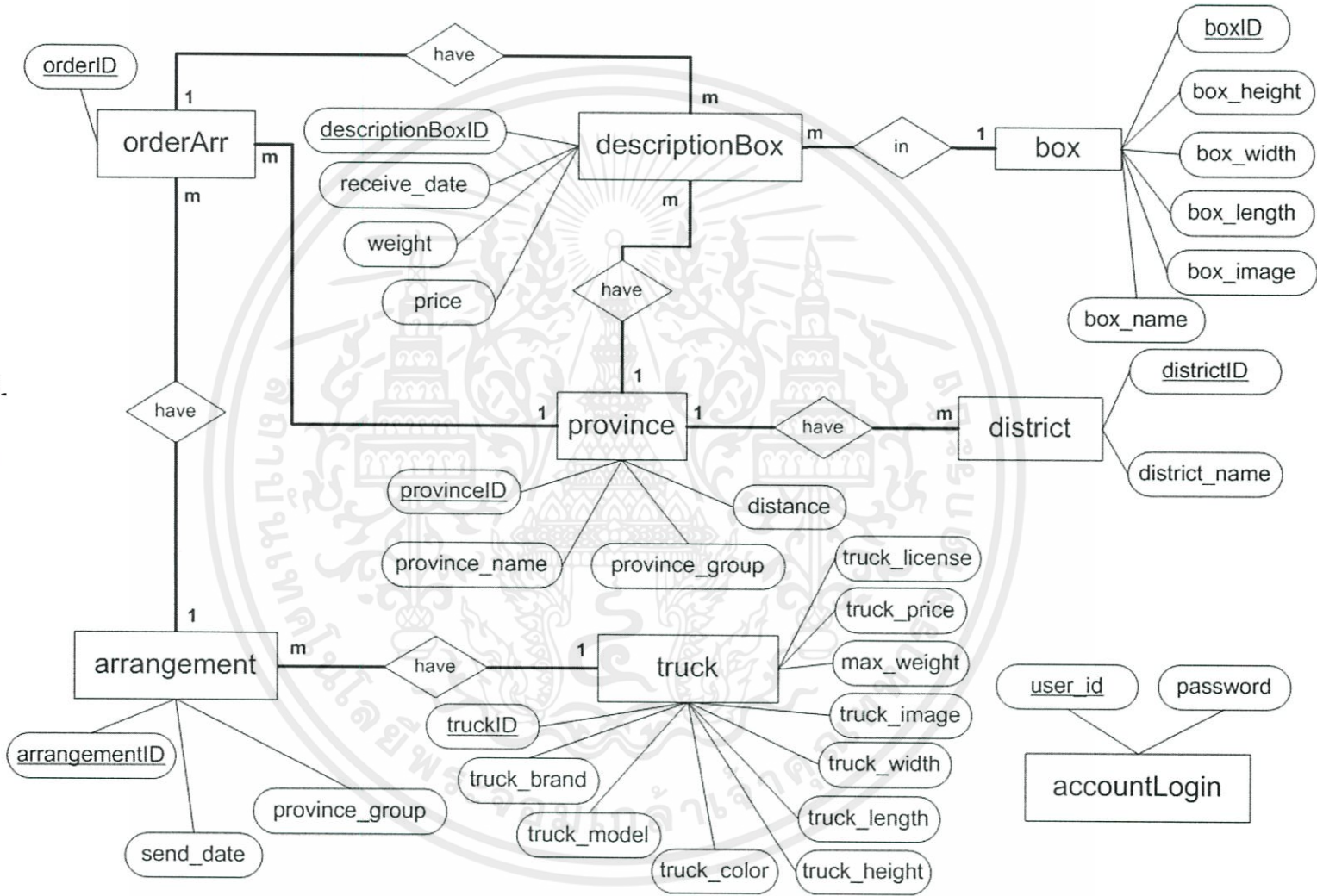
3.2.5 Activity Diagram สำหรับการเข้าสู่ระบบ



รูปที่ 3.6 Activity Diagram สำหรับการเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้วยการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่แบบสงวนเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.7 ER Diagram



3.3 ER Diagram

รูปที่ 3.7 แสดงถึงโครงสร้างของฐานข้อมูล เช่น ความสัมพันธ์ระหว่างข้อมูล แอททริบิวต์ในการเก็บข้อมูลต่างๆ เป็นต้น

ตารางที่ 3.1 truck

Column Name	Data Type	Description	Key
truckID	int	รหัสรถ	PK
truck_brand	varchar(50)	ยี่ห้อรถ	
truck_model	varchar(50)	รุ่นรถ	
truck_color	varchar(50)	สีรถ	
truck_width	float(2,4)	ความกว้างของกระบะรถ	
truck_length	float(2,4)	ความยาวของกระบะรถ	
truck_height	float(2,4)	ความสูงของกระบะรถ	
truck_image	image	รูปภาพรถ	
max_weight	int(6)	น้ำหนักสูงสุดที่สามารถบรรทุกได้	
truck_price	int(7)	ราคาเหมาคันรถ	
truck_license	varchar(50)	หมายเลขทะเบียนรถ	

ตารางที่ 3.2 arrangement

Column Name	Data Type	Description	Key
arrangementID	int	รหัสการจัดส่ง	PK
send_date	datetime	วันที่ส่ง	
truckID	int	รหัสรถ	FK
province_group	int(1)	ภูมิภาคที่จัดส่ง	FK

ตารางที่ 3.3 orderArr

Column Name	Data Type	Description	Key
orderID	int	รหัสการจัดส่งตามจังหวัด	PK
arrangementID	int	รหัสการจัดส่ง	FK
provinceID	int	รหัสจังหวัด	FK

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ห้ามมิให้คัดลอกเนื้อหาและต้องระบุชื่อของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.4 descriptionBox

Column Name	Data Type	Description	Key
descriptionBoxID	int	รหัสรายละเอียดสินค้า	PK
receive_date	datetime	วันที่รับสินค้าเข้าสู่ระบบ	
weight	int(6)	น้ำหนักสินค้า	
price	int(7)	ค่าจัดส่งสินค้า	
orderID	int	รหัสการจัดส่งตามจังหวัด	FK
provinceID	int	รหัสจังหวัด	FK
boxID	int	รหัสขนาดกล่องสินค้า	FK

ตารางที่ 3.5 box

Column Name	Data Type	Description	Key
boxID	int	รหัสขนาดกล่องสินค้า	PK
box_name	varchar(50)	ชื่อขนาดกล่องสินค้า	
box_width	float(2,4)	ความกว้างของกล่องสินค้า	
box_length	float(2,4)	ความยาวของกล่องสินค้า	
box_height	float(2,4)	ความสูงของกล่องสินค้า	
box_image	image	รูปภาพกล่องสินค้า	

ตารางที่ 3.6 province

Column Name	Data Type	Description	Key
provinceID	int	รหัสจังหวัด	PK
province_name	varchar(50)	ชื่อจังหวัด	
province_group	int(1)	ภูมิภาคจังหวัด	
distance	int(4)	ระยะทางการจัดส่ง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 district

Column Name	Data Type	Description	Key
districtID	int	รหัสอำเภอ	PK
district_name	varchar(50)	ชื่ออำเภอ	
provinceID	int	รหัสจังหวัด	FK

ตารางที่ 3.8 accountLogin

Column Name	Data Type	Description	Key
user_id	varchar(20)	ชื่อผู้ใช้งาน	PK
Password	varchar(20)	รหัสผ่านผู้ใช้งาน	

3.4 การประยุกต์ใช้ Best Fit Algorithm

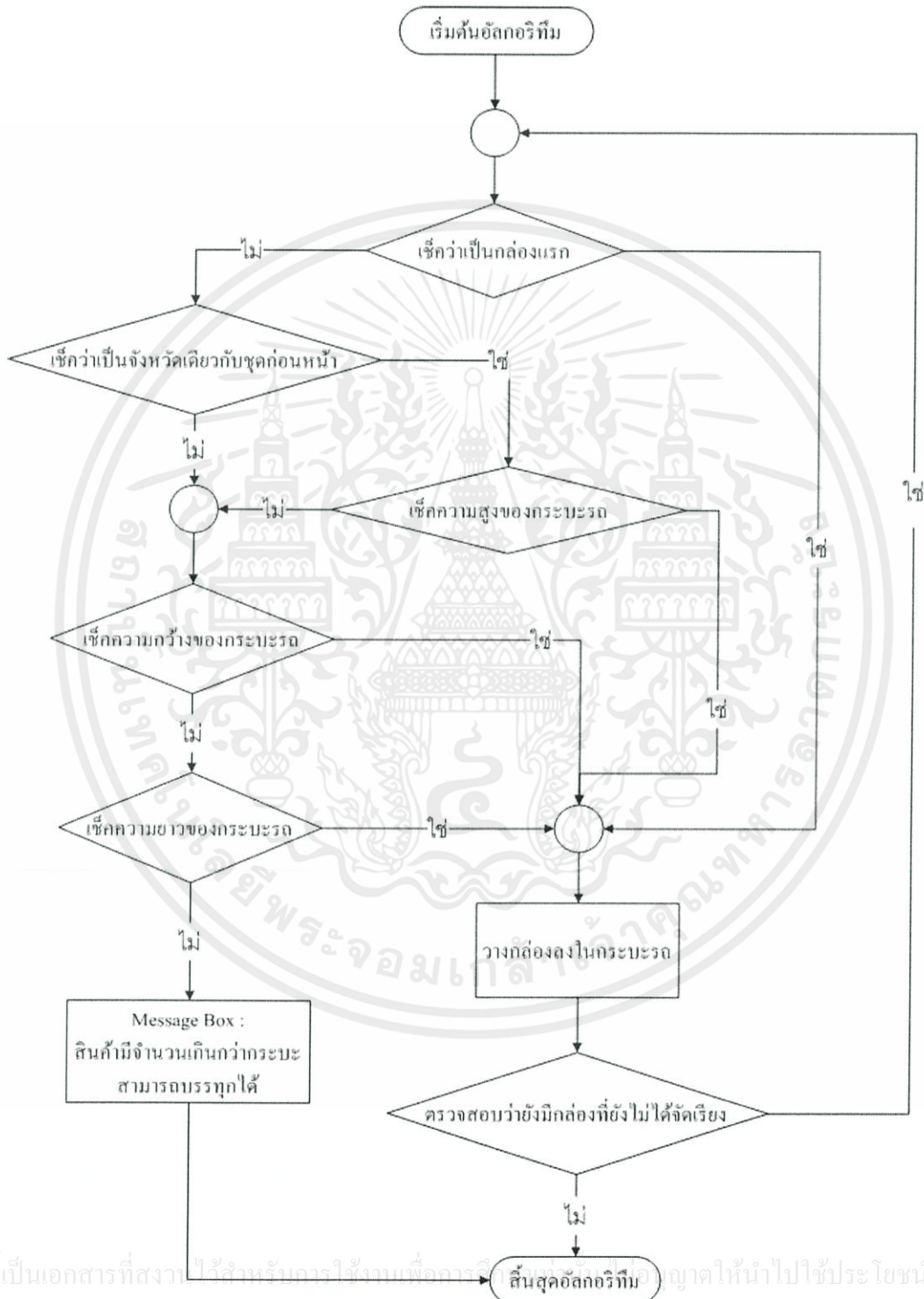
Best Fit Algorithm เป็นอัลกอริทึมที่สามารถประยุกต์ใช้ในการจัดเรียงสิ่งของได้เป็นอย่างดี ซึ่งถ้าใช้อัลกอริทึมนี้จะได้ผลลัพธ์การจัดเรียงได้เหมาะสมมากที่สุด

หลักการการประยุกต์ใช้ Best Fit Algorithm ในโปรแกรม คือ จะแบ่งพาร์ทิชัน(ตู้คอนเทนเนอร์) ออกเป็นส่วนๆ ในที่นี้จะแบ่งตามภาคที่ใช้ในการจัดส่ง ภาคที่จัดส่งก่อนอยู่ด้านนอกและภาคที่จัดส่งทีหลังจะอยู่ถัดไปด้านใน จากนั้นจะจัดเรียงสิ่งของเข้ามาในตู้คอนเทนเนอร์ในส่วนของพาร์ทิชันด้านในสุดก่อน โดยสิ่งของเหล่านั้นจะต้องเรียงลำดับการจัดส่งเอาไว้แล้ว ซึ่ง Best Fit จะทำการเลือกพื้นที่ในคอนเทนเนอร์ที่เหมาะสม คือ พื้นที่ที่มีขนาดใกล้เคียงกับขนาดของสิ่งของเพื่อที่จะบรรจุสิ่งของลงไปแล้วเหลือพื้นที่ว่างน้อยที่สุดก่อนที่จะจัดสิ่งของลงไป ในคอนเทนเนอร์ แต่ในที่นี้จะต้องคำนึงถึงลำดับในการจัดส่งด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 Flow Chart

3.5.1 Flow Chart แสดงการทำงานของ Best Fit Algorithm



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานวิจัยแบบเพื่อการศึกษา (ไม่หวังกำไร) กรุณาให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งยังมีให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
 รูปที่ 3.8 Flow Chart แสดงการทำงานของ Best Fit Algorithm

3.6 Graphical User Interface

3.6.1 หน้าหลักของโปรแกรม

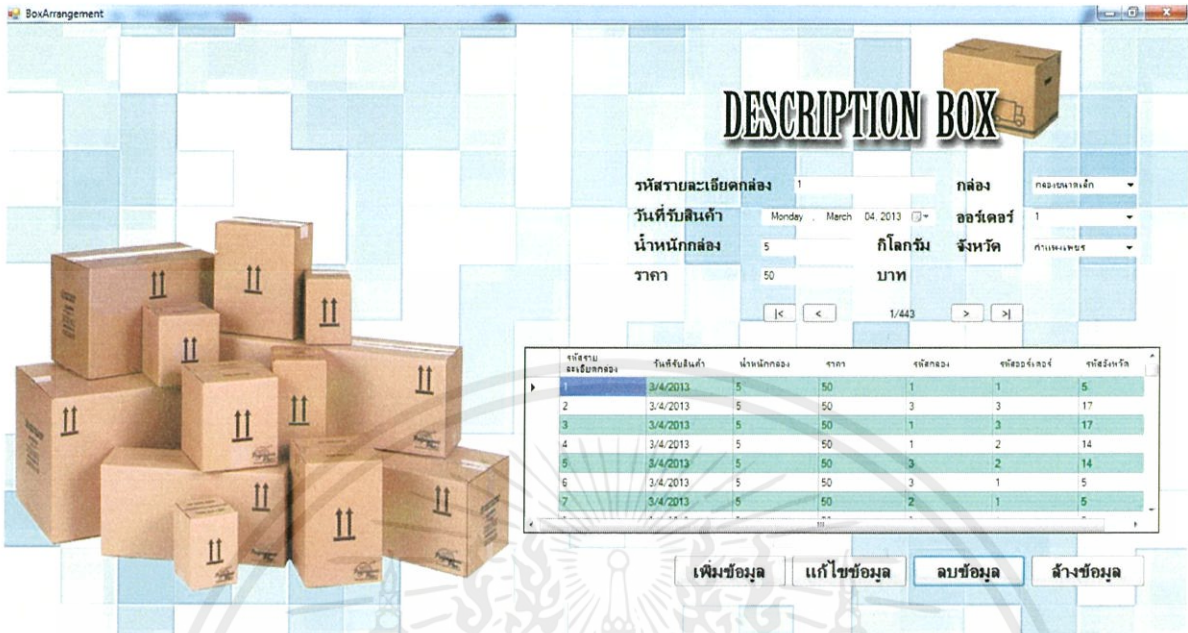


รูปที่ 3.10 หน้าหลักของโปรแกรม

ในหน้าหลักนี้มีทั้งหมด 5 เมนูที่จะนำผู้ใช้เข้าไปสู่ฟอร์มย่อยอื่นๆ ซึ่งมีเมนูรายละเอียดคลัง ออร์เดอร์ การจัดส่งสินค้า และแก้ไขข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

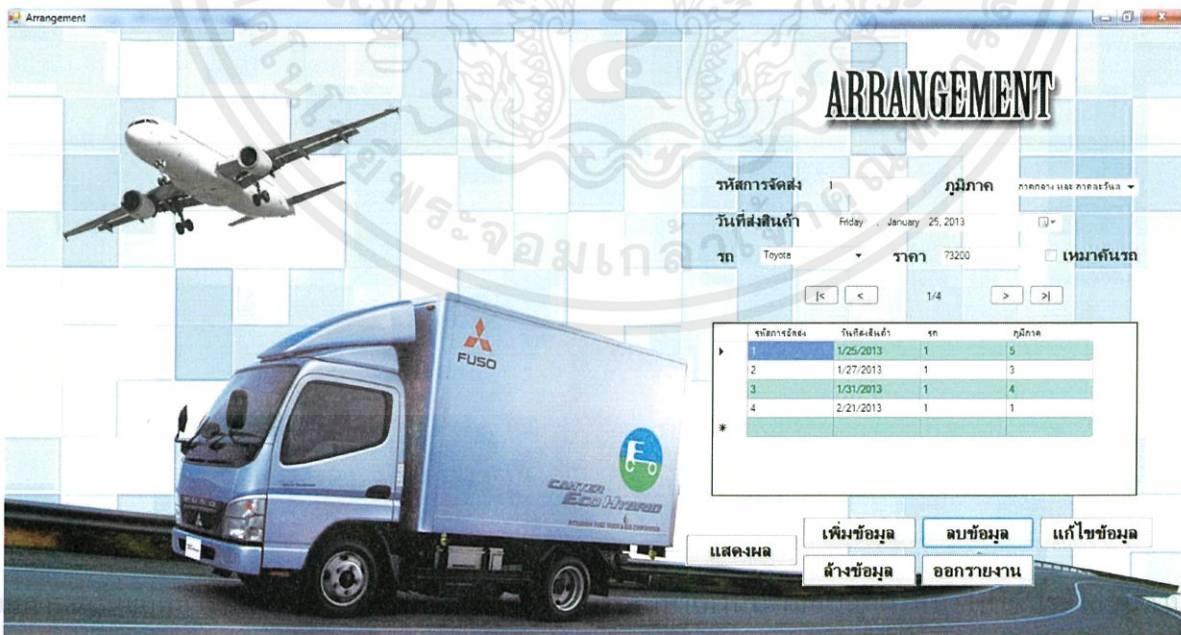
3.6.2 หน้ารายละเอียดกล่อง



รูปที่ 3.11 หน้ารายละเอียดกล่อง

หน้ารายละเอียดกล่อง ใช้สำหรับรับข้อมูลของสินค้าไม่ว่าจะเป็น ขนาดของกล่อง วันที่รับสินค้า จังหวัดและอำเภอที่ต้องการจัดส่ง น้ำหนักกล่อง เป็นต้น

3.6.3 หน้าการจัดส่งสินค้า



รูปที่ 3.12 หน้าการจัดส่งสินค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกไปใช้ของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

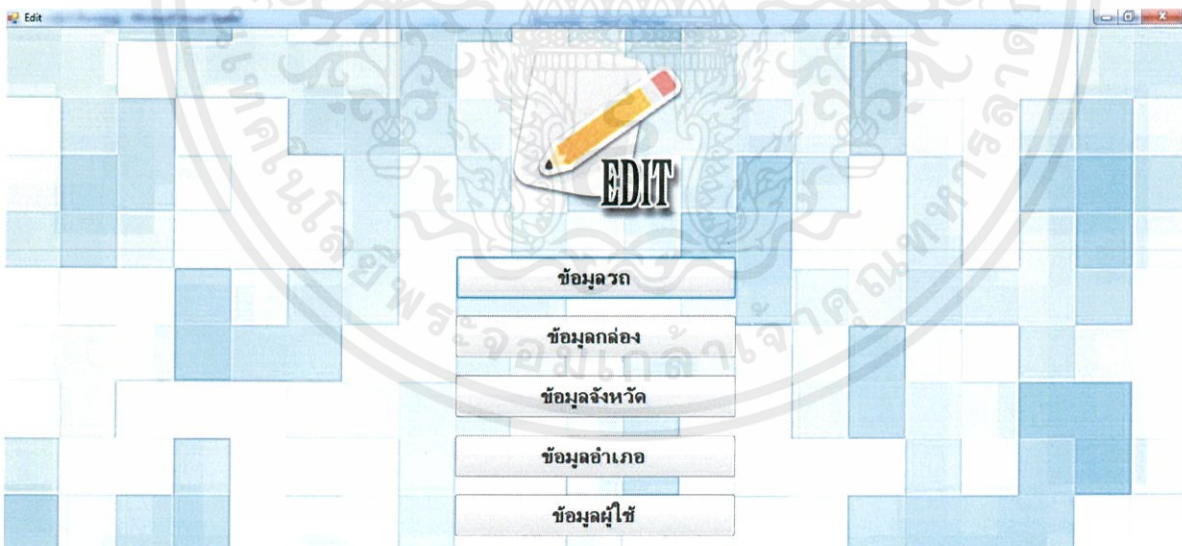
ผลการดำเนินงาน

4.1 การแสดงผลของโปรแกรมจัดเรียงสินค้า



รูปที่ 4.1 หน้า Log in

หน้า Log in ผู้ใช้จะต้องใส่ชื่อผู้ใช้งานและรหัสผ่านเพื่อเข้าไปแก้ไขข้อมูล โดยจะต้องมีบัญชีของระบบเพื่อใช้ในการเข้าถึงข้อมูลส่วนสำคัญ



รูปที่ 4.2 หน้าแก้ไขข้อมูล

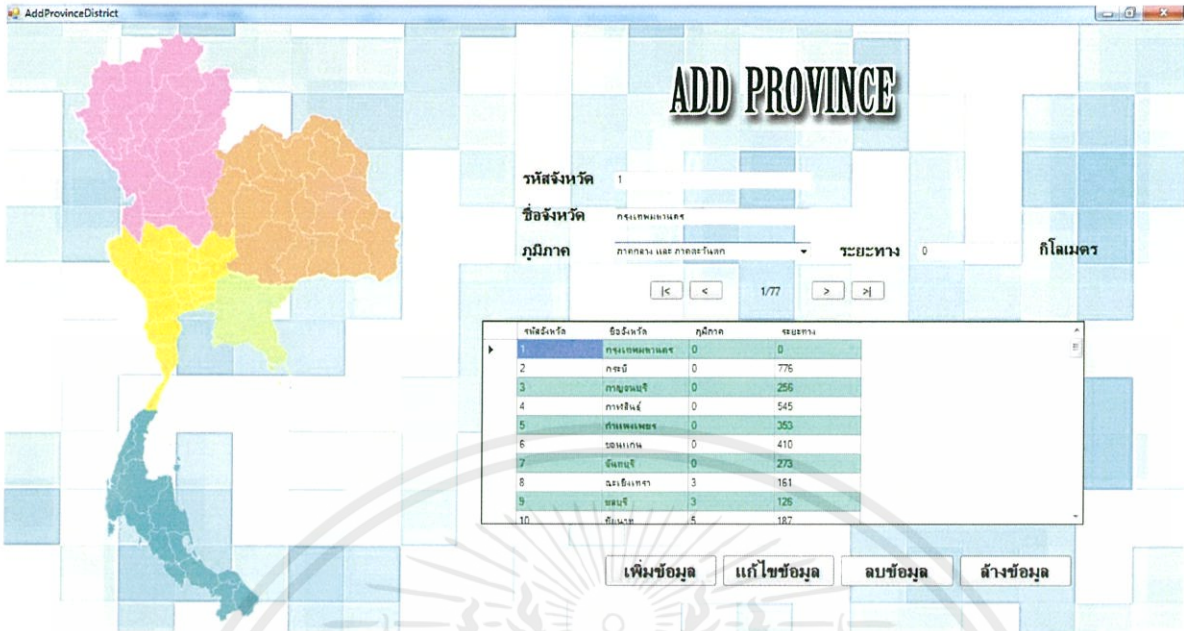
หน้าแก้ไขข้อมูล ผู้ใช้สามารถเลือกได้ว่าแก้ไขข้อมูลส่วนใด ซึ่งจะมีข้อมูล 5 ส่วน คือ ข้อมูลรถ เอกสารนี้เป็นเอกสารหลักฐานไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้า ไม่นอนุญาตให้นำไปใช้ประโยชน์การค้า ข้อมูลกล่อง ข้อมูลจังหวัด ข้อมูลอำเภอ และข้อมูลผู้ใช้ ไม่อนุญาตให้นำไปใช้

รูปที่ 4.3 หน้าข้อมูลรถ

หน้าข้อมูลรถ ผู้ใช้สามารถจัดการข้อมูลรถที่ใช้ในการจัดส่ง ไม่ว่าจะเป็นยี่ห้อรถ รุ่นรถ สีรถ และขนาดของกระบะที่ใช้ในการบรรทุกทุกสินค้า

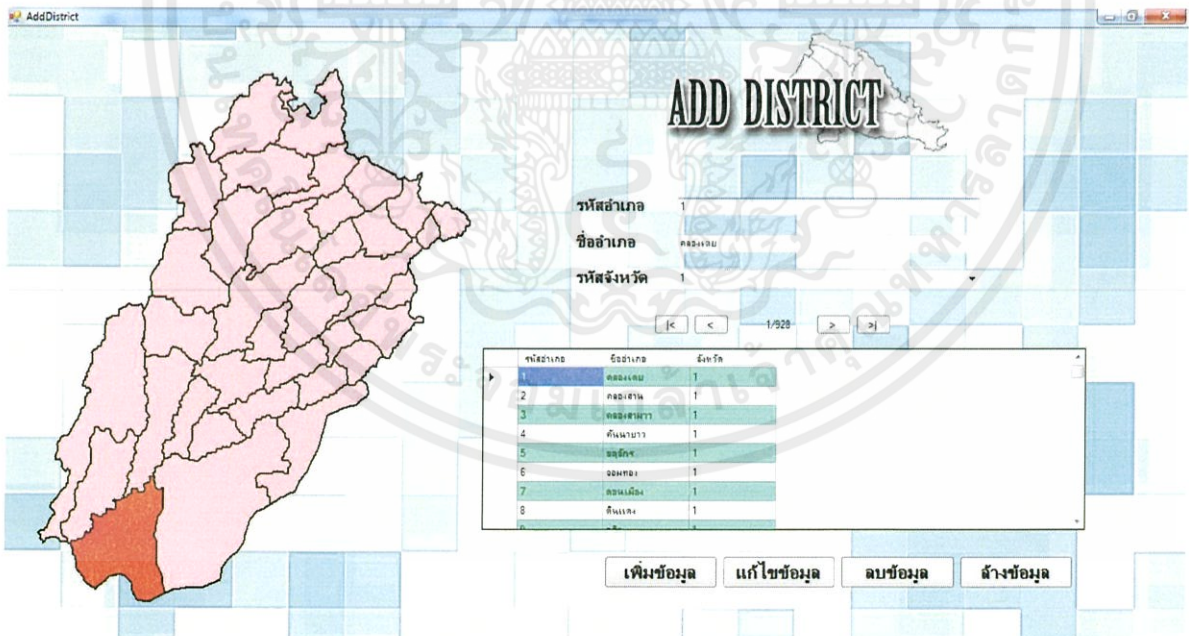
รูปที่ 4.4 หน้าข้อมูลกล่อง

หน้าข้อมูลกล่อง ผู้ใช้สามารถจัดการข้อมูลประเภทของกล่องที่ใช้ในการจัดส่ง โดยจะต้องกำหนดเอกสารเป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ความกว้าง ความยาว และความสูงในแต่ละประเภทของกล่อง ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 หน้าข้อมูลจังหวัด

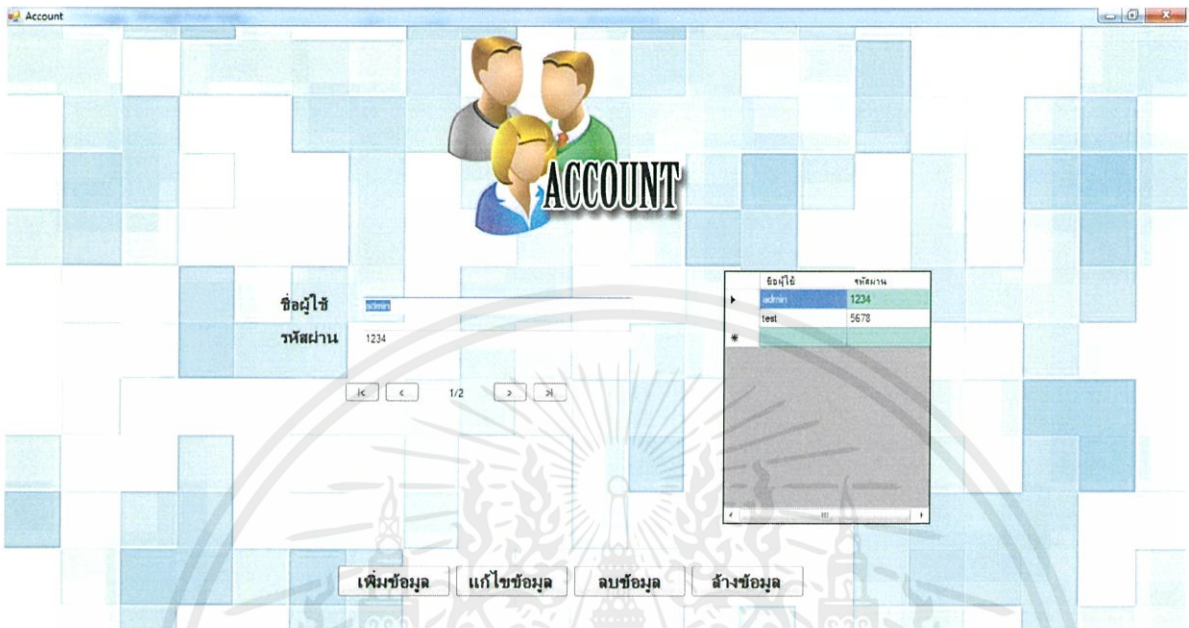
หน้าข้อมูลจังหวัด ผู้ใช้สามารถจัดการข้อมูลจังหวัดที่ใช้ในการจัดส่ง ซึ่งแต่ละจังหวัดจะแบ่งออกไปตามแต่ละภูมิภาคอีกด้วย



รูปที่ 4.6 หน้าข้อมูลอำเภอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าข้อมูลอำเภอ ผู้ใช้สามารถจัดการข้อมูลอำเภอที่ใช้ในการจัดส่ง ซึ่งจะต้องระบุจังหวัดก่อนจึงจะสามารถเพิ่มอำเภอได้



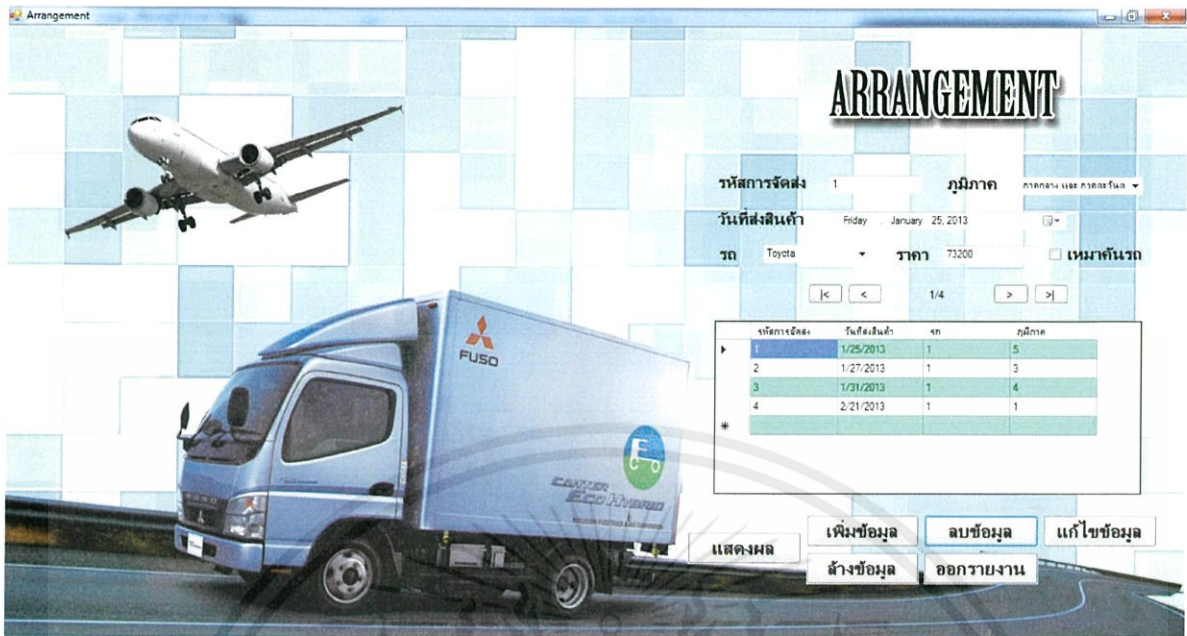
รูปที่ 4.7 หน้าข้อมูลผู้ใช้

หน้าข้อมูลผู้ใช้ ผู้ใช้สามารถจัดการข้อมูลผู้ใช้ ที่ใช้ในการเข้าไปสู่ส่วนการแก้ไขข้อมูล



รูปที่ 4.8 หน้าออเดอร์

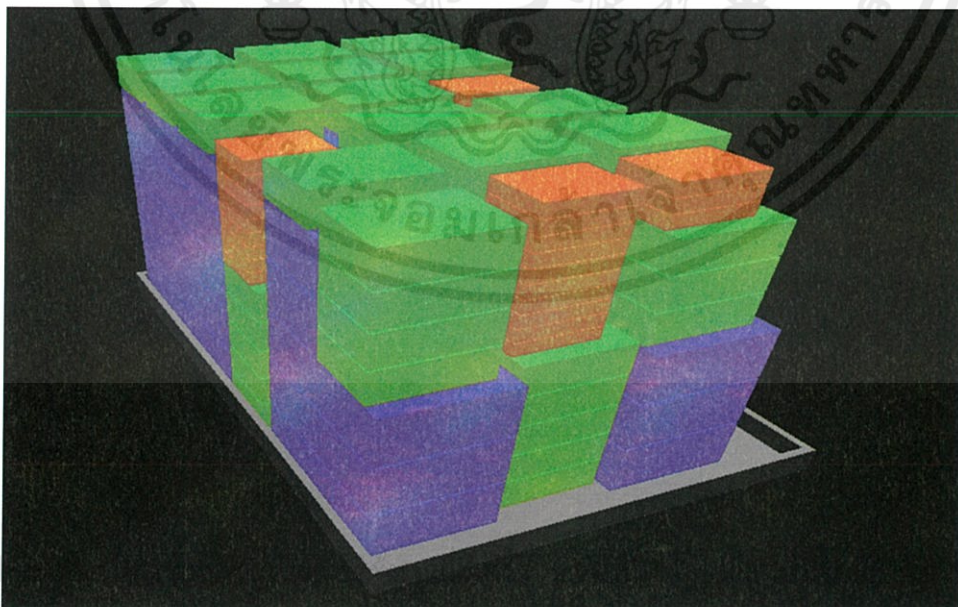
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณี หน้าที่ออเดอร์ ผู้ใช้สามารถจัดการข้อมูลออเดอร์ และจังหวัดที่จะจัดส่งสินค้าได้ ครั้งที่มีการนำไปใช้



รูปที่ 4.9 หน้าการจัดส่งสินค้า

จากรูปที่ 4.9 หน้าการจัดส่งสินค้า ผู้ใช้สามารถเลือกวันที่ส่งสินค้า และรถที่จะใช้ในการจัดส่ง โดยใน ส่วนของราคาที่จะใช้จัดส่งสินค้า—ลูกค้าสามารถเลือกราคาในการจัดส่งเป็นขั้น โดยจะกรอกราคาในหน้าจอรายละเอียดของกล่อง หรือจะเลือกกรอกราคาเหมาทั้งคันรถ โดยเลือก “เหมาคันรถ” ในหน้าจการจัดส่งสินค้า

เมื่อกรอกข้อมูลครบถ้วนแล้วผู้ใช้จะต้องเลือกรหัสการจัดส่งเพื่อแสดงผลการจัดเรียงสินค้าในรูปแบบ 3 มิติ โดยกดปุ่ม “แสดงผล” ซึ่งจะแสดงผลดังรูปที่ 4.10



รูปที่ 4.10 หน้าจอการจัดเรียงสินค้าในรูปแบบ 3 มิติ

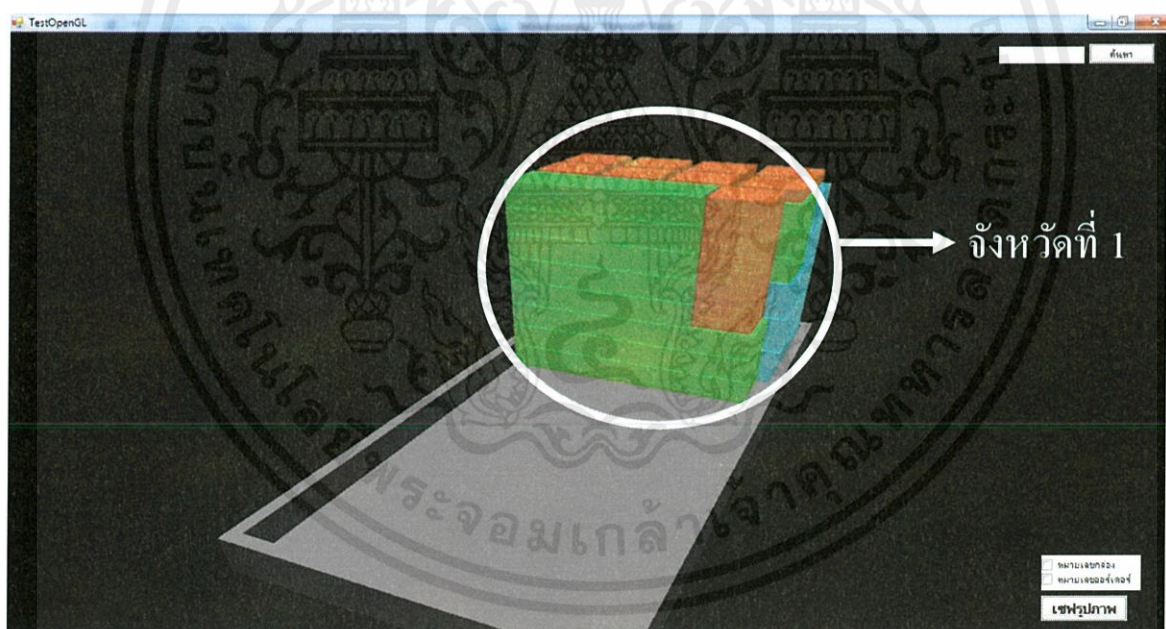
4.2 ผลการจัดเรียงสินค้า

กรณีที่ 1 การจัดเรียงสินค้า 1 จังหวัด

ในการจัดเรียงสินค้า 1 จังหวัด จะมีการจัดเรียงลำดับของกล่องสินค้าใหม่ โดยจะเรียงจากกล่องที่มีขนาดใหญ่ไปหากล่องที่มีขนาดเล็ก จากนั้นจึงจะนำมาจัดเรียงใส่รถตามลำดับ โดยแสดงข้อมูลของกล่องที่จัดเรียงในจังหวัดที่ 1 ดังตารางที่ 4.1 และแสดงรายละเอียดการจัดเรียง ดังรูปที่ 4.11

ตารางที่ 4.1 ข้อมูลของกล่องที่จัดเรียงในจังหวัดที่ 1

ขนาด (cm.)	สี	กว้าง	ยาว	สูง	จำนวน (กล่อง)
เล็ก	แดง	49	52	15	18
กลาง	เขียว	53	56	30	30
ใหญ่	ฟ้า	55	60	45	38



รูปที่ 4.11 ผลการจัดเรียงสินค้ากรณีที่ 1

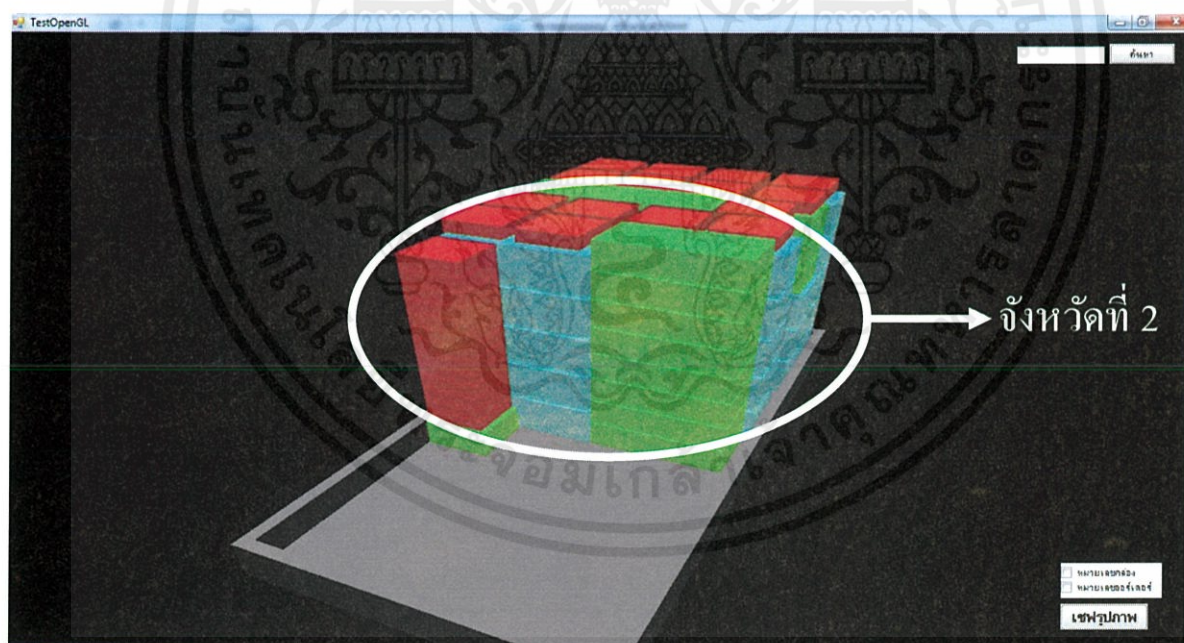
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 2 การจัดเรียงสินค้า 2 จังหวัด

ในการจัดเรียงสินค้า 2 จังหวัดนั้น โปรแกรมจะมีการจัดเรียงลำดับของกล่องสินค้าใหม่ โดยจะมีการจัดเรียง 2 ขั้นตอน โดยขั้นตอนที่ 1 โปรแกรมจะจัดเรียงตามระยะทางของจังหวัด โดยจะเรียงจากจังหวัดที่มีระยะทางไกลไปหาจังหวัดที่มีระยะทางใกล้ จากนั้นจะเข้าสู่การจัดเรียงขั้นตอนที่ 2 โดยจะเรียงเฉพาะในจังหวัดเดียวกันเท่านั้น เริ่มจากเรียงจากกล่องที่มีขนาดใหญ่ไปหากล่องที่มีขนาดเล็ก จากนั้นจึงจะนำมาจัดเรียงใส่รถตามลำดับที่ได้เรียงไว้ โดยแสดงข้อมูลของกล่องที่จัดเรียงในจังหวัดที่ 2 ดังตารางที่ 4.2 และแสดงรายละเอียดการจัดเรียง ดังรูปที่ 4.12

ตารางที่ 4.2 ข้อมูลของกล่องที่จัดเรียงในจังหวัดที่ 2

ขนาด (cm.)	สี	กว้าง	ยาว	สูง	จำนวน (กล่อง)
เล็ก	แดง	49	52	15	19
กลาง	เขียว	53	56	30	17
ใหญ่	ฟ้า	55	60	45	30



รูปที่ 4.12 ผลการจัดเรียงสินค้ากรณีที่ 2

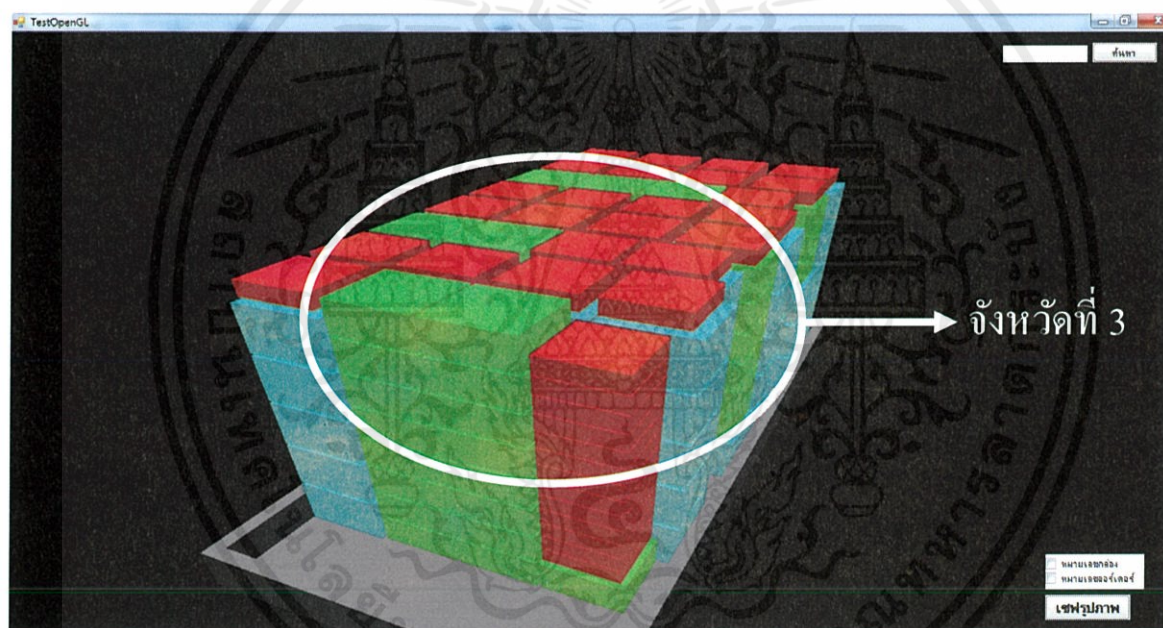
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีศึกษาที่ 3 การจัดเรียงสินค้า 3 จังหวัด

การจัดเรียงสินค้า 3 จังหวัดนั้น จะใช้วิธีการจัดเรียงเช่นเดียวกันกับกรณีศึกษาที่ 2 ดังตารางที่ 4.3 และรูปที่ 4.13

ตารางที่ 4.3 ข้อมูลของกล่องที่จัดเรียงในจังหวัดที่ 3

ขนาด (cm.)	สี	กว้าง	ยาว	สูง	จำนวน (กล่อง)
เล็ก	แดง	49	52	15	23
กลาง	เขียว	53	56	30	22
ใหญ่	ฟ้า	55	60	45	36

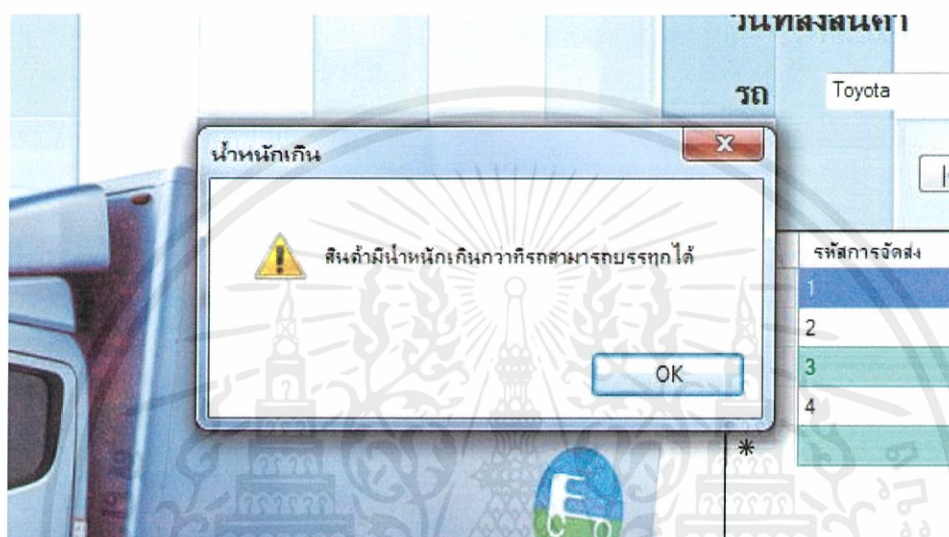


รูปที่ 4.13 ผลการจัดเรียงสินค้ากรณีศึกษาที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 4 สินค้ามีน้ำหนักรวมเกินกว่าที่รถสามารถบรรทุกได้

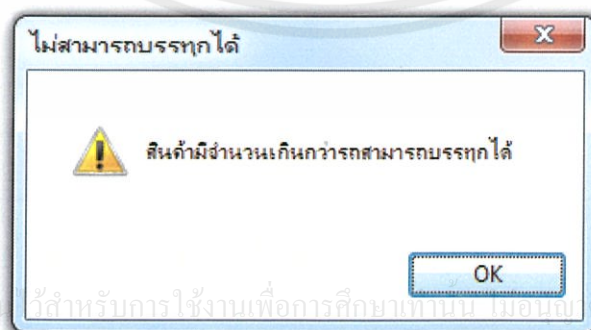
หากผู้ใช้นำกล่องสินค้ามาจัดเรียงแล้ว น้ำหนักรวมของสินค้ามีค่ามากเกินกว่าที่รถจะสามารถบรรทุกได้ โปรแกรมจะแสดงกล่องข้อความแจ้งเตือนว่า สินค้ามีน้ำหนักเกินกว่าที่รถสามารถบรรทุกได้ ซึ่งผู้ใช้จะต้องกลับไปแก้ไขข้อมูลการจัดเรียงใหม่ เพื่อลดน้ำหนักสินค้าลง จนกระทั่งรถสามารถบรรทุกสินค้าได้ ดังรูปที่ 4.14



รูปที่ 4.14 สินค้ามีน้ำหนักรวมเกินกว่าที่รถสามารถบรรทุกได้

กรณีที่ 5 สินค้ามีจำนวนเกินกว่าที่รถสามารถบรรทุกได้

หากผู้ใช้นำกล่องสินค้ามาจัดเรียง โดยที่มีจำนวนสินค้ามากเกินกว่าที่รถจะสามารถบรรทุกได้ โปรแกรมจะแสดงกล่องข้อความแจ้งเตือนว่า สินค้ามีจำนวนเกินกว่ารถสามารถบรรทุกได้ ซึ่งผู้ใช้จะต้องกลับไปแก้ไขข้อมูลการจัดเรียงใหม่ เพื่อลดจำนวนสินค้าลง จนกระทั่งรถสามารถบรรทุกสินค้าได้ ดังรูปที่ 4.15

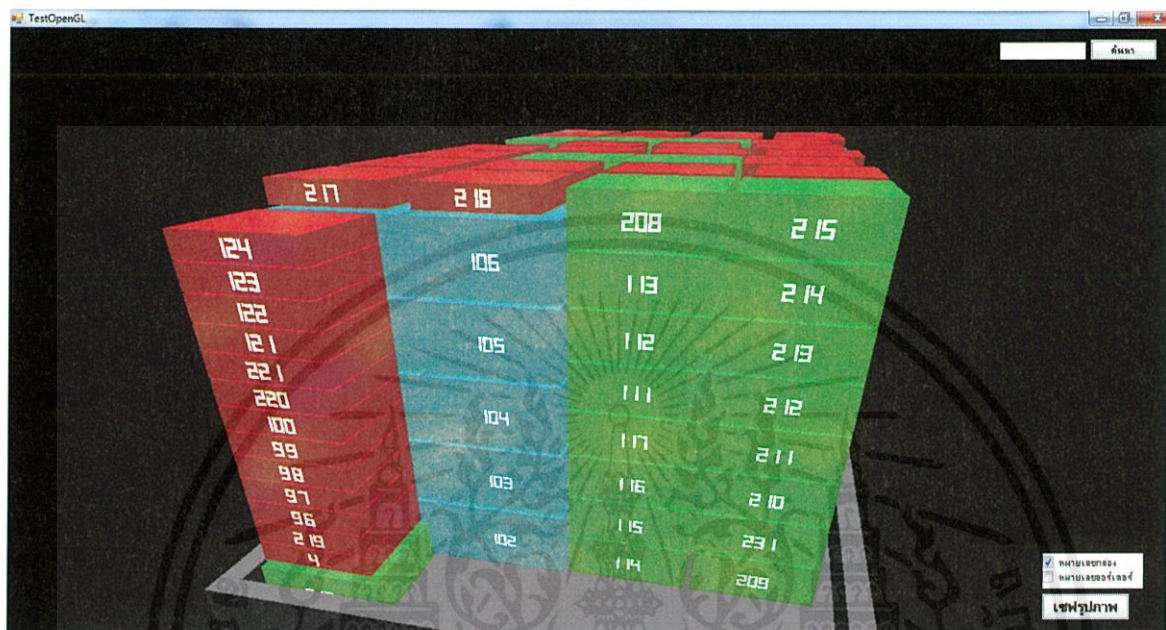


รูปที่ 4.15 สินค้ามีจำนวนเกินกว่าที่รถสามารถบรรทุกได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่ 6 แสดงหมายเลขกล่อง

ผู้ใช้สามารถร้องขอให้โปรแกรมแสดงหมายเลขกล่องได้โดยการเลือก checkbox หมายเลขกล่องทางด้านมุมขวาล่าง โดยโปรแกรมจะแสดงหมายเลขกล่องขึ้นมาทางด้านหน้าของกล่อง ดังรูปที่ 4.16

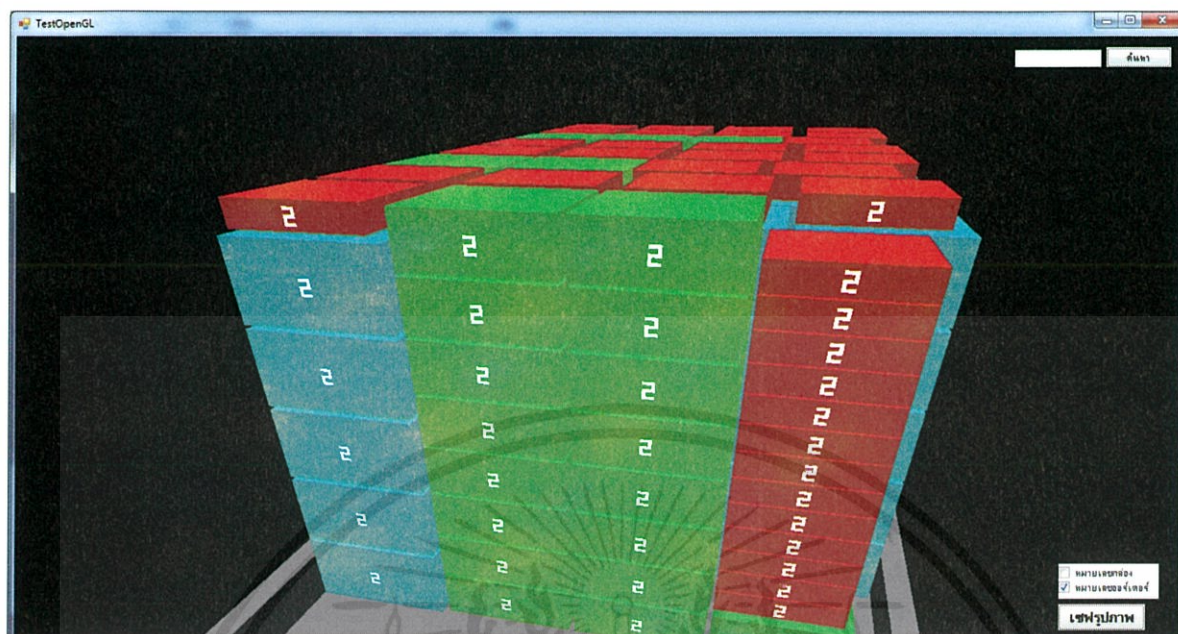


รูปที่ 4.16 แสดงหมายเลขกล่อง

กรณีที่ 7 แสดงหมายเลขออร์เคอร์

ผู้ใช้สามารถร้องขอให้โปรแกรมแสดงหมายเลขออร์เคอร์ได้โดยการเลือก checkbox หมายเลขออร์เคอร์ทางด้านมุมขวาล่าง โดยโปรแกรมจะแสดงหมายเลขออร์เคอร์ขึ้นมาทางด้านหน้าของกล่อง ดังรูปที่ 4.17

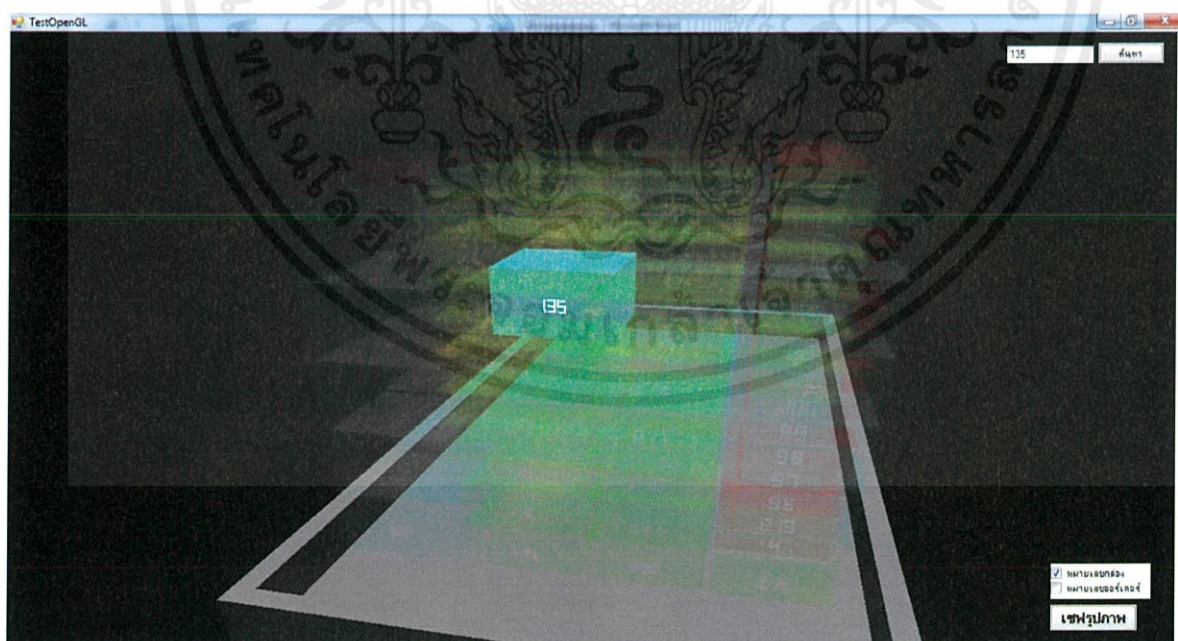
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



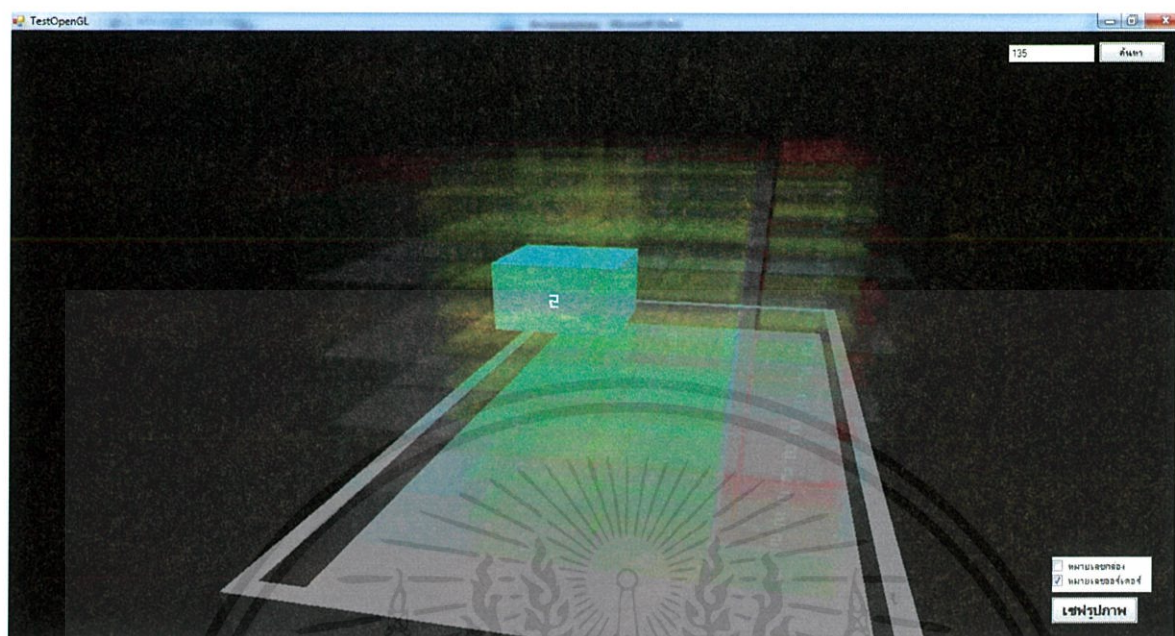
รูปที่ 4.17 แสดงหมายเลขออร์เคอร์

กรณีที่ 8 ค้นหากล่อง

ผู้ใช้สามารถค้นหาหมายเลขกล่องที่ต้องการได้จากช่องค้นหา โดยโปรแกรมจะแสดงผลกล่องอื่นๆ เป็นแบบโปร่งใส และจะแสดงกล่องที่ต้องการแบบทึบแสง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 4.18 ค้นหาโดยแสดงผลกล่อง
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 ค้นหาโดยแสดงหมายเลขออร์เดอร์

จากรูปที่ 4.9 เมื่อคลิกปุ่ม ออกรายงาน โปรแกรมจะทำการออกรายงานของการจัดเรียงที่ใช้เลือก ดังตัวอย่างในรูปที่ 4.20

วันที่ส่ง	การจัดส่ง	ปริมาณ	รถ	ทะเบียนรถ	ออร์เดอร์	จังหวัด	คลัง	จำนวนคลัง
1/7/2013	1	1	Tata Xenon	ฅ9999	1	ลำปางเหนือ	คลังขนาดเล็ก	21
					2	เชียงใหม่	คลังขนาดกลาง	30
							คลังขนาดใหญ่	38
							คลังขนาดเล็ก	19
							คลังขนาดกลาง	17
							คลังขนาดใหญ่	30
					3	ตาก	คลังขนาดเล็ก	18
							คลังขนาดกลาง	25
							คลังขนาดใหญ่	36
							จำนวนคลังทั้งสิ้น	234

รูปที่ 4.20 รายงานการจัดส่งสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผลการทดลอง

ในปัญหาพิเศษนี้จะมุ่งเน้นการจัดเรียงสินค้าด้วยอัลกอริทึม Best - Fit เพื่อแก้ไขปัญหาการจัดเรียงสินค้า เพื่อให้ผู้ใช้สามารถจัดเรียงสินค้าโดยใช้พื้นที่อย่างมีประสิทธิภาพและช่วยลดทรัพยากรในการจัดเรียงไม่ว่าจะเป็น บุคลากร เวลา หรือค่าใช้จ่าย โดยโปรแกรมนี้จะแสดงผลในรูปแบบ 3 มิติ

โดยโปรแกรมมีคุณสมบัติดังนี้

- 1) ผู้ใช้สามารถเพิ่ม แก้ไข หรือลบข้อมูลต่างๆ ภายในตัวโปรแกรมได้
- 2) ผู้ใช้สามารถคำนวณน้ำหนักของสินค้าทั้งหมดได้
- 3) ผู้ใช้สามารถคำนวณราคาทั้งแบบรายชิ้นหรือเหมาคันรถได้
- 4) ผู้ใช้สามารถเก็บข้อมูลทั้งหมดลงฐานข้อมูลได้
- 5) ผู้ใช้สามารถเลือกรถ และกล่องขนาดต่างๆ ได้
- 6) โปรแกรมสามารถแสดงผลการจัดเรียงในรูปแบบ 3 มิติ และบันทึกภาพการจัดเรียงได้
- 7) ผู้ใช้สามารถออกรายงานการจัดส่งสินค้าได้

5.2 ปัญหาและอุปสรรคในการทดลอง

- 1) เนื่องจากผู้พัฒนาได้ใช้ภาษา Visual Basic ในการพัฒนา ซึ่งขาดต่อการแสดงผลในรูปแบบ 3 มิติ จึงทำให้ผู้พัฒนาต้องติดตั้ง OpenTK เพิ่มเติม จึงทำให้ไม่สามารถแสดงผลในรูปแบบที่มีเงาของวัตถุได้

5.3 ข้อเสนอแนะและแนวทางการพัฒนา

- 1) พัฒนาส่วนติดต่อผู้ใช้ให้สามารถใช้งานได้ง่ายและมีประสิทธิภาพมากขึ้น
- 2) เพิ่มกรณีตัวอย่างที่ใช้วางกล่องในรูปแบบต่างๆ เพื่อให้โปรแกรมมีความสมบูรณ์มากขึ้น

เอกสารอ้างอิง

- [1] smooth14. (2009). Logistics. สืบค้นเมื่อ 21 กรกฎาคม 2012, จาก <http://smooth14.exteen.com/20090702/logistics>
- [2] Anonymous. (2009). Logistics. สืบค้นเมื่อ 25 กรกฎาคม 2012, จาก <http://www.logisticafe.com/2009/08/โลจิสติกส์-logistics-คือ>
- [3] อุมาพร มณีเนียม. (2009). ความสำคัญของการจัดการ โลจิสติกส์. สืบค้นเมื่อ 3 มีนาคม 2013, จาก <http://logisticscorner.com/index.php/2009-05-25-00-45-43/logistics/407-2009-07-05-14-54-35.html>
- [4] สุวรรณณี. (2012). โลจิสติกส์คืออะไร. สืบค้นเมื่อ 3 มีนาคม 2013, จาก http://www.tpa.or.th/writer/read_this_book_topic.php?bookID=2377&read=true&count=true
- [5] Anonymous. (ไม่ปรากฏปีพิมพ์). Logistics คือ. สืบค้นเมื่อ 3 มีนาคม 2013, จาก http://www.happyworkplaceclinic.com/index.php?option=com_k2&view=item&layout=item&id=61&Itemid=324
- [6] led0warl. (2010). OpenGL . สืบค้นเมื่อ 23 สิงหาคม 2012, จาก <http://led0warl.exteen.com/20100731/opengl>
- [7] Anonymous. (ไม่ปรากฏปีพิมพ์). OpenGL. สืบค้นเมื่อ 23 สิงหาคม 2012, จาก <http://202.44.14.219/~isumich/ftp/CG/html/OpenGL.htm>
- [8] Chakrit. (2008). OpenGL และ GLUT เบื้องต้น. สืบค้นเมื่อ 23 สิงหาคม 2012, จาก http://access.cs.sci.ku.ac.th/~ckw/wiki/index.php/OpenGL_และ_GLUT_เบื้องต้น
- [9] Anonymous. (ไม่ปรากฏปีพิมพ์). Chapter 0: Learn OpenTK in 15'. สืบค้นเมื่อ 16 กุมภาพันธ์ 2012, จาก <http://www.opentk.com/doc/chapter/0>
- [10] Anonymous (2009). Visual Basic. สืบค้นเมื่อ 14 กันยายน 2012, จาก http://www.oocities.org/vb_training/whatis.html [2012, September 14]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.
การติดตั้งโปรแกรมที่จำเป็นต่อการพัฒนา
โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่งเพื่อระบบโลจิสติกส์

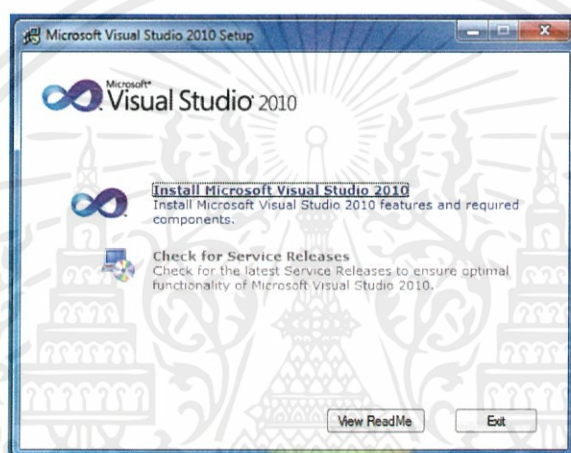


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก.1 การติดตั้งโปรแกรม Visual Studio 2010

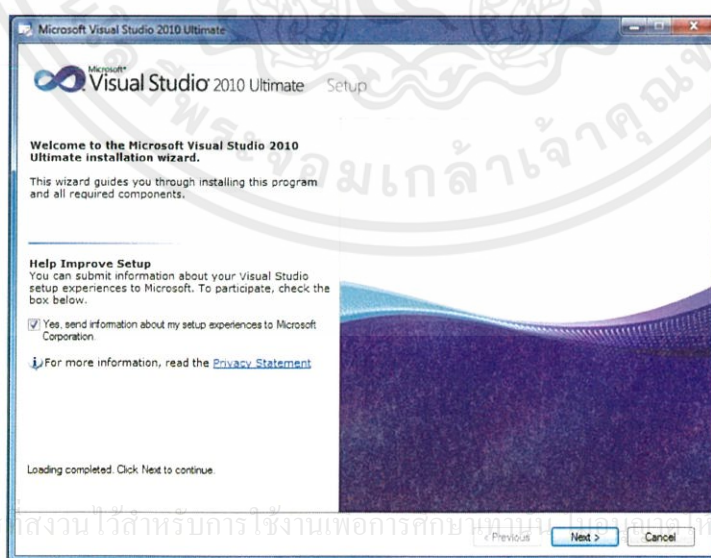
Visual Studio 2010 เป็นซอฟต์แวร์ประเภท IDE (Integrated Development Environment) ช่วยให้ผู้พัฒนาโปรแกรมสามารถเขียนโปรแกรมด้วยความสะดวกสบายขึ้น สามารถแก้ไขข้อผิดพลาดในการเขียนโปรแกรมได้ง่าย รวดเร็ว

1. คับเบิลคลิก ไฟล์ติดตั้งขึ้นมารอสักครู่จะปรากฏหน้าต่าง Microsoft Visual Studio 2010 Setup คลิก Install Microsoft Visual Studio 2010 เพื่อติดตั้งโปรแกรม หรือ คลิกปุ่ม Exit เพื่อออกจากการติดตั้งโปรแกรม



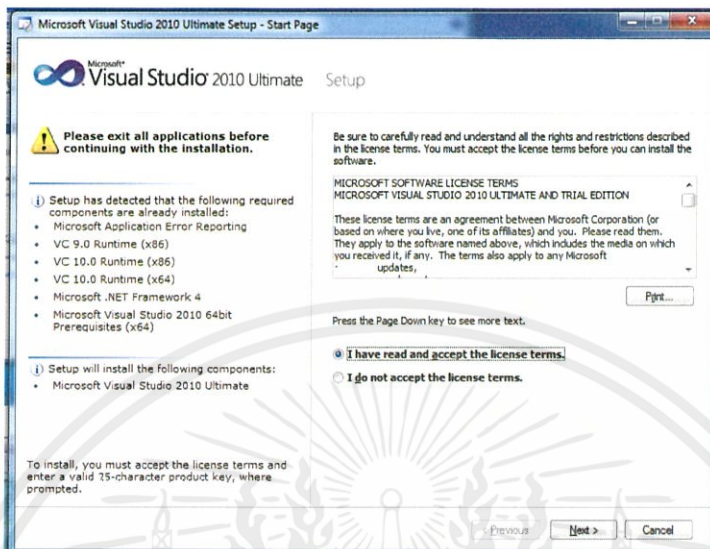
รูปที่ ก.1 หน้า Microsoft Visual Studio 2010 Setup

2. เมื่อเข้าสู่หน้าการติดตั้งโปรแกรม คลิกปุ่ม Next เพื่อเข้าสู่ขั้นตอนต่อไป



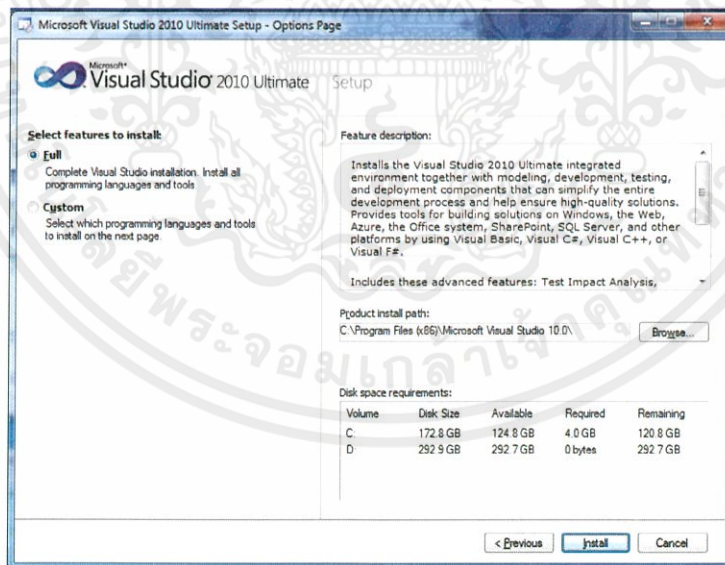
รูปที่ ก.2 หน้า Microsoft Visual Studio 2010 Ultimate

- เลือก I have read and accept the license terms. เพื่อรับทราบเงื่อนไขต่างๆ ของโปรแกรม และคลิกปุ่ม Next เพื่อเข้าสู่ขั้นตอนต่อไป



รูปที่ ก.3 หน้า Microsoft Visual Studio 2010 Ultimate Setup – Start Page

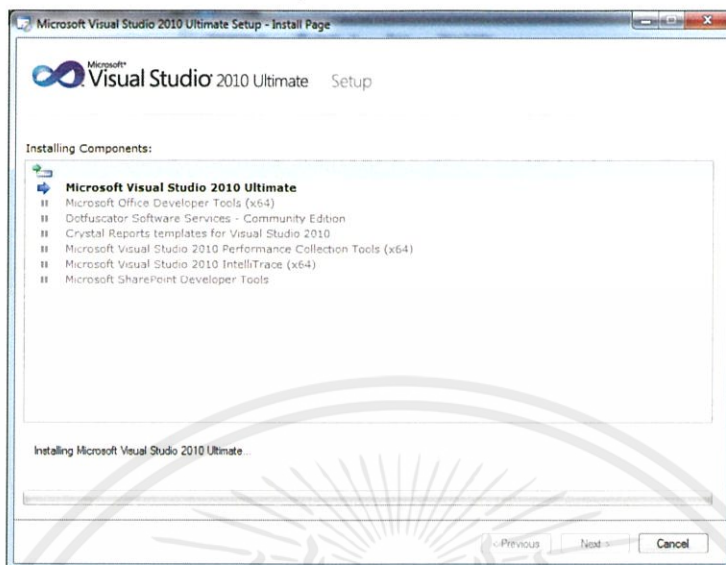
- เลือก Path ที่จะใช้ติดตั้งโปรแกรม โดยโปรแกรมจะแสดงสถานะต่างๆของ Drive ภายในเครื่องคอมพิวเตอร์ จากนั้นกดปุ่ม Install เพื่อติดตั้งโปรแกรม



รูปที่ ก.4 หน้า Microsoft Visual Studio 2010 Ultimate Setup – Options Page

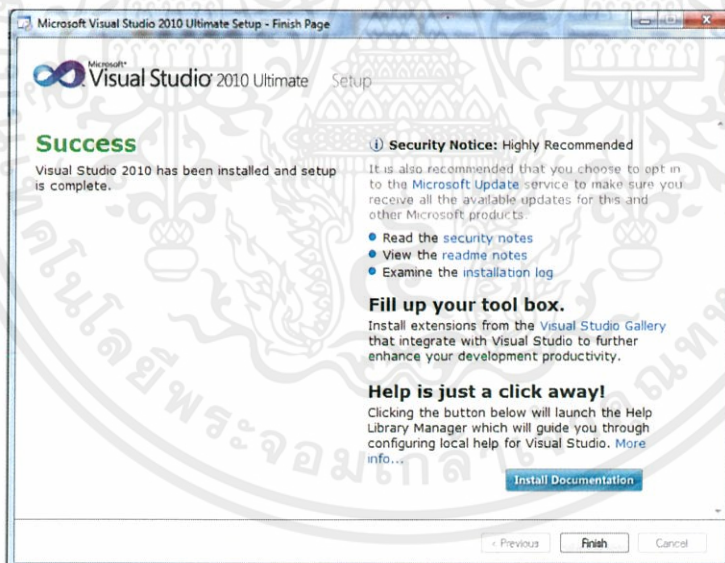
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. รอนกระทั่งโปรแกรมติดตั้งเสร็จสมบูรณ์



รูปที่ ก.5 หน้า Microsoft Visual Studio 2010 Ultimate Setup – Install Page

6. เมื่อติดตั้งโปรแกรมเสร็จจะแสดงหน้าต่างดังรูปที่ ก.6 จากนั้นคลิกปุ่ม Finish เพื่อสิ้นสุดการติดตั้งโปรแกรม

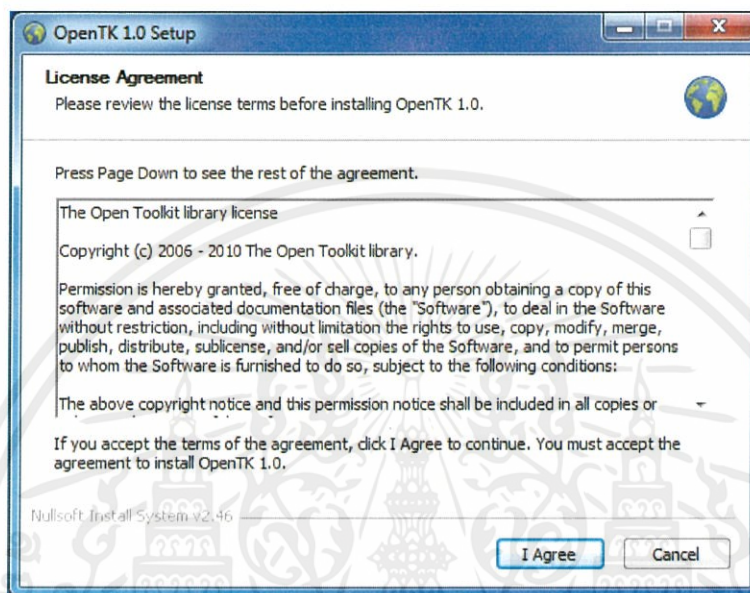


รูปที่ ก.6 หน้า Microsoft Visual Studio 2010 Ultimate Setup – Finish Page

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

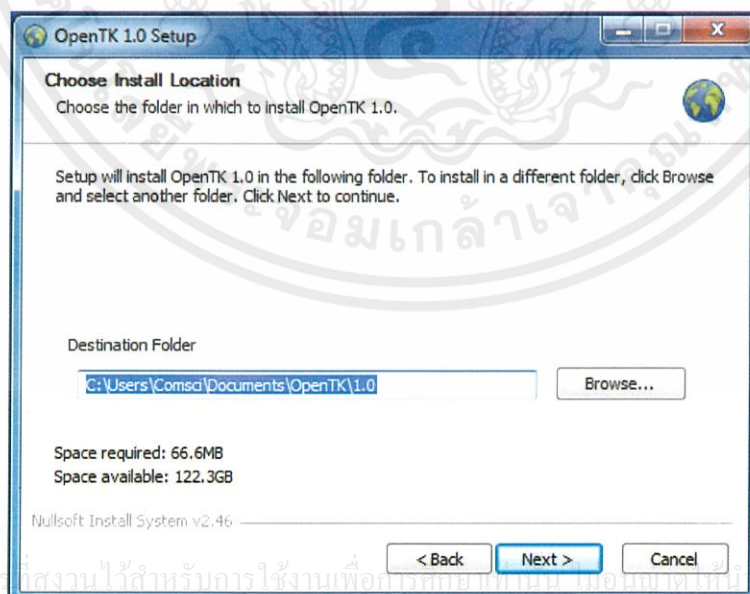
ก.2 การติดตั้ง OpenTK

1. ดับเบิลคลิก ไฟล์ติดตั้งขึ้นมาจะปรากฏหน้าต่าง OpenTK 1.0 Setup คลิก I Agree เพื่อยอมรับลิขสิทธิ์และติดตั้งโปรแกรม หรือ คลิกปุ่ม Cancel เพื่อออกจากการติดตั้งโปรแกรม



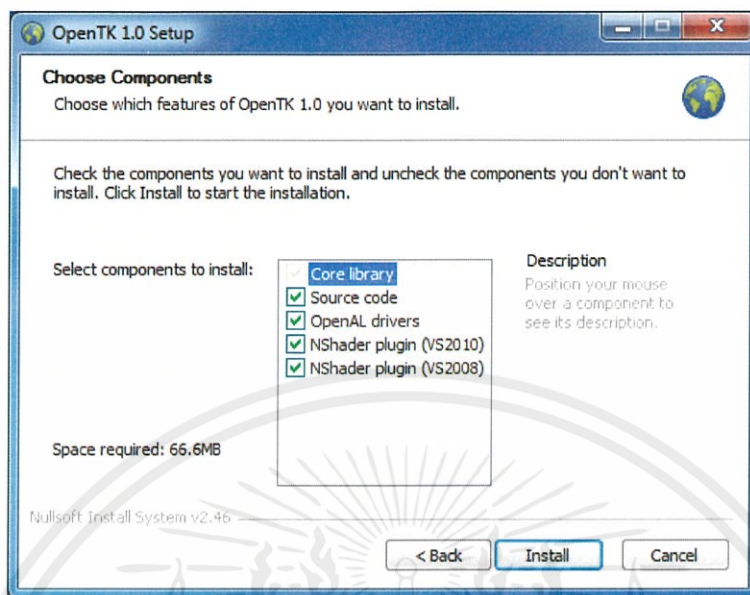
รูปที่ ก.7 หน้าลิขสิทธิ์ของ OpenTK

2. เลือก Path ที่จะใช้ในการติดตั้งโปรแกรม จากนั้นคลิกปุ่ม Next เพื่อเข้าสู่ขั้นตอนต่อไป



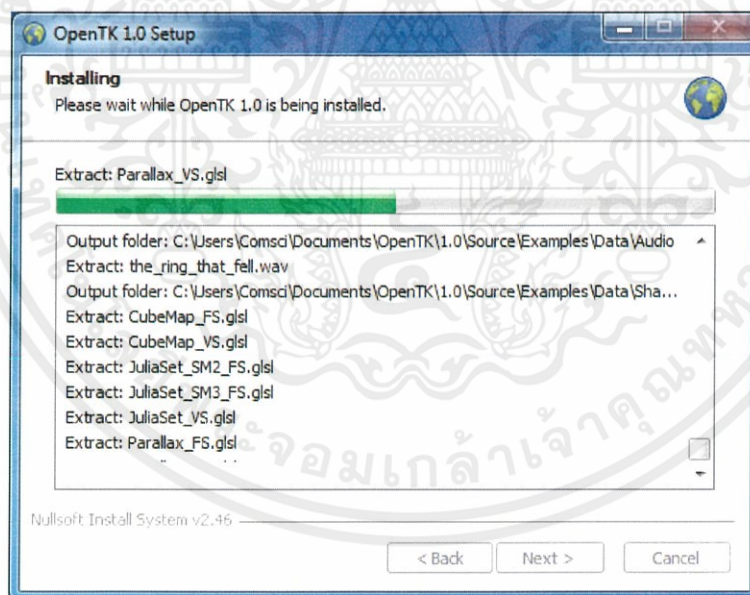
รูปที่ ก.8 หน้าเลือก Path ที่จะใช้ในการติดตั้ง โปรแกรม

3. เลือกส่วนต่างๆ ที่ต้องการติดตั้งของโปรแกรม จากนั้นคลิกปุ่ม Install เพื่อติดตั้งโปรแกรม



รูปที่ ก.9 หน้าเลือกส่วนต่างๆ ที่ต้องการติดตั้งของโปรแกรม

4. รอจนกระทั่งโปรแกรมติดตั้งเสร็จสมบูรณ์



รูปที่ ก.10 หน้าการติดตั้ง OpenTK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เมื่อติดตั้งโปรแกรมเสร็จจะแสดงหน้าต่างดังรูปที่ ก.6 จากนั้นคลิกปุ่ม Finish เพื่อสิ้นสุดการติดตั้งโปรแกรม



รูปที่ ก.11 หน้าสิ้นสุดการติดตั้งโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.
การติดตั้งโปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง
เพื่อระบบโลจิสติกส์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข.1 การติดตั้งโปรแกรม ProductArrangement

1. ดับเบิลคลิก ไฟล์ติดตั้ง ProductArrangement_Setup เพื่อทำการติดตั้งโปรแกรม

Documents library

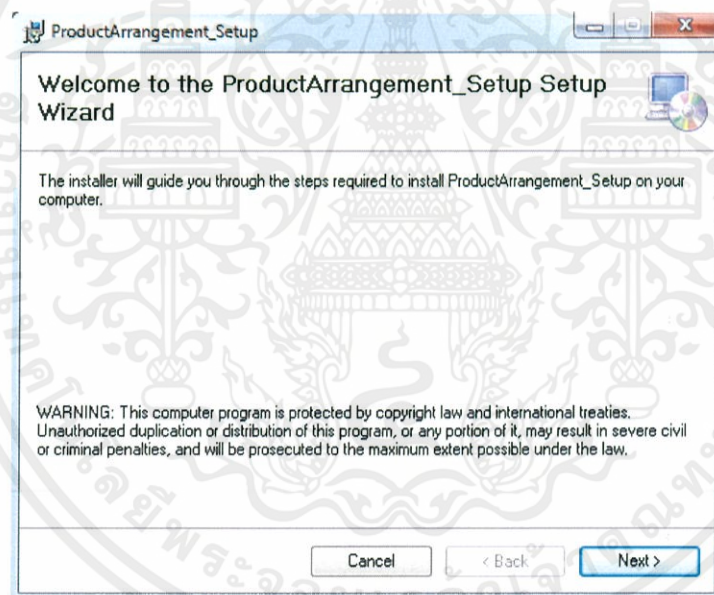
Setup



ProductArrange
ment_Setup

รูปที่ ข.1 ไฟล์ติดตั้ง โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง

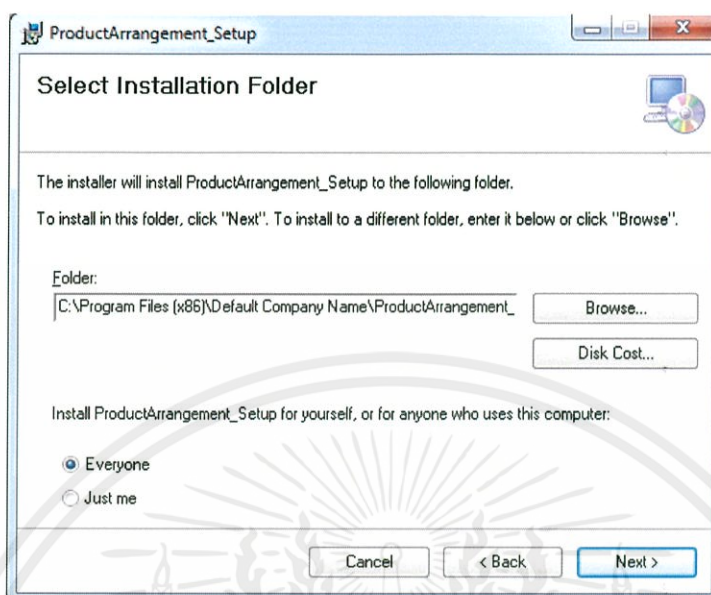
2. เมื่อเข้าสู่หน้าการติดตั้ง โปรแกรม คลิกปุ่ม Next เพื่อ ไปยังขั้นตอนต่อไป



รูปที่ ข.2 หน้า ProductArrangement Setup

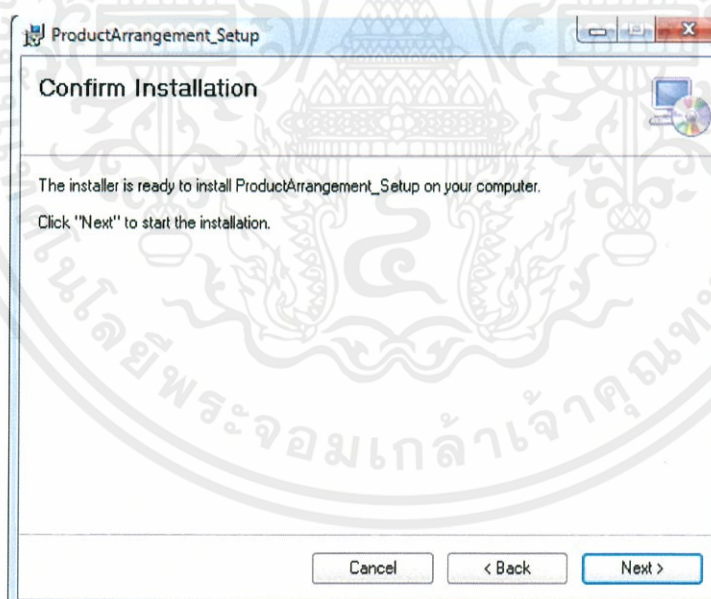
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. กดปุ่ม Browse เพื่อเลือก Path ในการติดตั้งโปรแกรม และคลิกปุ่ม Next เพื่อเข้าสู่ขั้นตอนต่อไป



รูปที่ ข.3 หน้าเลือก path ที่ใช้ในการติดตั้งโปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง

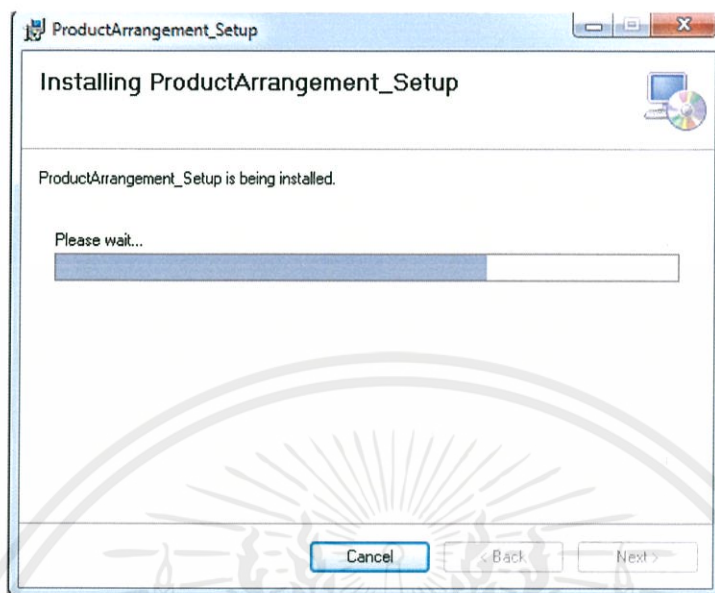
4. เมื่อถึงหน้าจอขั้นตอนการติดตั้ง คลิกปุ่ม Next เพื่อยืนยันการติดตั้งและไปยังขั้นตอนต่อไป



รูปที่ ข.4 หน้ายืนยันการติดตั้งโปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง

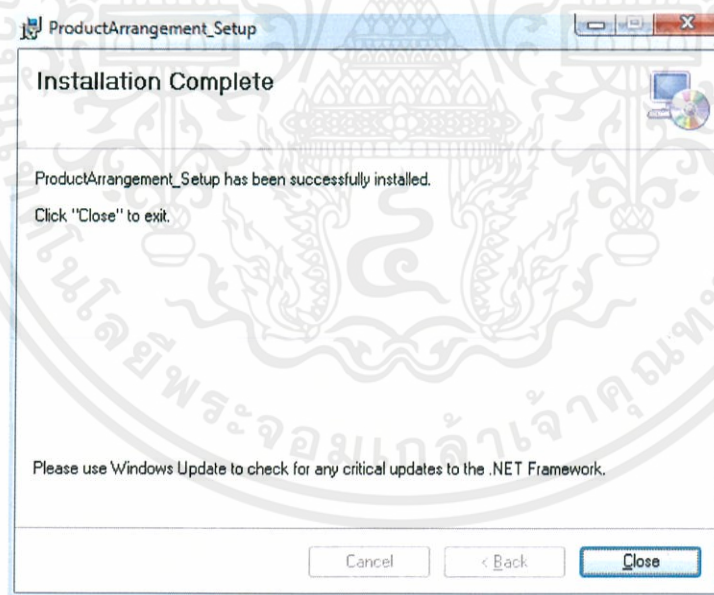
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. รอจนกระทั่งโปรแกรมติดตั้งเสร็จสมบูรณ์



รูปที่ ข.5 หน้าการติดตั้ง โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง

6. เมื่อติดตั้งโปรแกรมเสร็จสมบูรณ์แล้ว คลิกปุ่ม Close เพื่อออกจากหน้าการติดตั้ง



รูปที่ ข.6 หน้าสิ้นสุดการติดตั้ง โปรแกรมจัดเรียงสินค้าตามลำดับการจัดส่ง

หมายเหตุ: หลังจากติดตั้ง โปรแกรมเรียบร้อยแล้วต้องตั้งค่า permission ของไฟล์ ProductArrangement.mdf
 และ ProductArrangement_log.ldf ให้เป็น Full Control ก่อนเริ่มใช้งาน ไม้อนุญาตให้เข้าไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้