

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

๓ ๘

# การนำระบบ RFID ประยุกต์ใช้กับระบบดูแลรักษาสัตว์

## VETERINARY MEDICAL SYSTEM DEVELOPMENT BY USING RFID

โดย

ชัยยศ วินัยกุล

CHAIYOS WINIYAKOON

อาจารย์ที่ปรึกษา

รศ.ดร. โชติพัทธ์ ภรณ์วลัย



\*H003333\*

วัน เดือน ปี.....	22 พ.ค. 2550
เลขทะเบียน.....	03333
เลขเรียกหนังสือ.....	ฉ.พ. 446ก 2549
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

6-1752571  
112725123

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน  
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ  
คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่ห้องสมุดเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ภาคเรียนที่ 1 ปีการศึกษา 2549  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**VETERINARY MEDICAL SYSTEM DEVELOPMENT BY USING  
RFID**



**A SYSTEM DEVELOPMENT PROJECT OF THE REQUIREMENT FOR  
THE DEGREE OF  
MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY  
FACULTY OF INFORMATION TECNOLOGY  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**COPYRIGHT 2006**

**FACULTY OF INFORMATION TECHNOLOGY**

เอกสารนี้เป็นทรัพย์สินทางปัญญาของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง การค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการ	การนำระบบ RFID ประยุกต์ใช้กับระบบดูแลรักษาสัตว์
นักศึกษา	นายชัยยศ วินิบาต
รหัสนักศึกษา	47066147
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2549
อาจารย์ที่ปรึกษา	รศ.ดร.โชติพัชร ภรณ์วลัย

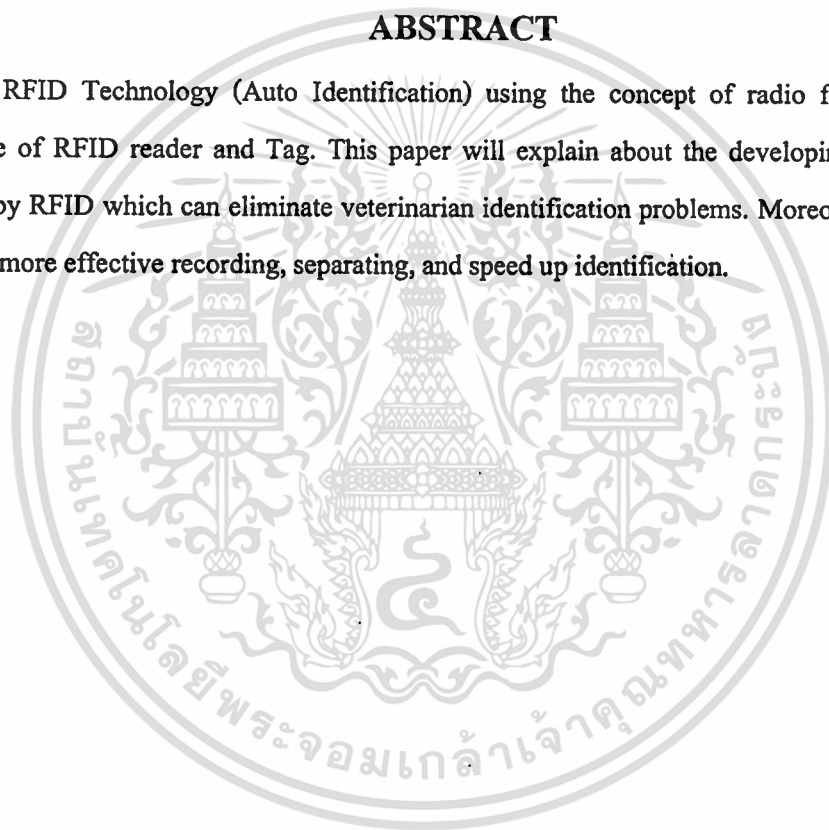
### บทคัดย่อ

เทคโนโลยี RFID(Radio Frequency Identification) เป็นเทคโนโลยีที่ระบุ วัตถุประสงค์ของวัตถุ โดยใช้หลักการของคลื่นวิทยุ ซึ่งประกอบด้วยตัวอ่านและป้ายติดวัตถุ โดยบทความนี้จะกล่าวถึงการนำเทคโนโลยี RFID ไปประยุกต์ใช้กับระบบดูแลรักษาสัตว์ในโรงพยาบาล โดยการนำเทคโนโลยี RFID มาช่วยขจัดปัญหาของสัตว์แพทย์ในการเก็บและแยกแยะลักษณะข้อมูลของสัตว์ให้เป็นระบบเพื่อระบุลักษณะและข้อมูลที่จำเป็นของสัตว์ ซึ่ง สะดวก และรวดเร็วยิ่งขึ้น

<b>Project Title</b>	Veterinary Medical System Development by Using RFID
<b>Student</b>	Mr. Chaiyos Winiyakoon
<b>Student ID</b>	47066147
<b>Degree</b>	Master of Science
<b>Programme</b>	Information Science
<b>Academic Year</b>	2006
<b>Advisor</b>	Assoc. Prof. Dr. Chotipat Pornavalai

## ABSTRACT

RFID Technology (Auto Identification) using the concept of radio frequency which compose of RFID reader and Tag. This paper will explain about the developing of veterinary system by RFID which can eliminate veterinarian identification problems. Moreover, System can provide more effective recording, separating, and speed up identification.



## กิตติกรรมประกาศ

ในการพัฒนาโครงการพัฒนาระบบอาร์เอฟไอดีไปประยุกต์ร่วมกับระบบดูแลรักษาสัตว์สามารถสำเร็จลุล่วงไปด้วยดี ข้าพเจ้าขอขอบคุณพระคุณ รศ.ดร. โขติพัทธ์ ภรณ์วลัย อาจารย์ที่ปรึกษา ที่กรุณาให้คำแนะนำ ข้อเสนอแนะ และเป็นที่ปรึกษาในการแก้ปัญหาต่างๆ รวมทั้งเป็นผู้ตรวจสอบความถูกต้องของโครงการนี้ ซึ่งรายงานเล่มนี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา อาจารย์ที่ให้ความรู้ในการเรียนการสอน กรรมการผู้คุมสอบ และเจ้าหน้าที่ทุกฝ่ายที่ช่วยตรวจการออกเล่มรายงาน ได้ชี้แนะ ช่วยแก้ปัญหา ตลอดจนให้ความรู้และประสบการณ์ที่ดีแก่ข้าพเจ้า

ขอขอบพระคุณ บิดา มารดา ที่ได้ให้การสนับสนุนทางด้านกำลังใจและทุนทรัพย์ตลอดจนการทำโครงการนี้สำเร็จด้วยดีมาตลอด

ขอขอบคุณ พี่เน็ก โป้ง แอม นอว นื่องอร และพี่ เพื่อน นื่อง ทุกคน ที่มีส่วนร่วมช่วยเหลือในด้านต่างๆที่เกี่ยวกับโครงการไว้ ณ ที่นี้ด้วย

ชัยศ วินัยกุล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	V
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 จุดมุ่งหมายและวัตถุประสงค์ของโครงการ.....	2
1.3 ขอบเขตของการพัฒนาระบบ.....	2
1.4 ขั้นตอนการดำเนินโครงการ.....	3
1.5 ประโยชน์ที่คาดว่าจะได้รับจากการพัฒนาโครงการ.....	3
บทที่ 2 กระบวนการเตรียมข้อมูลและทฤษฎีที่เกี่ยวข้อง.....	5
2.1 RFID(Radio Frequency Identification).....	5
2.1.1 ความเป็นมาปัญหาของเทคโนโลยี RFID.....	5
2.1.2 องค์ประกอบของ RFID.....	6
2.1.2.1 ส่วนประกอบของเทคโนโลยี RFID.....	6
2.1.2.1.1 แท็ก(Tag).....	6
2.1.2.1.2 เครื่องอ่าน(Reader).....	7
2.1.3 ขั้นตอนการทำงานระหว่างเครื่องอ่านกับแท็ก.....	8
2.1.4 การอ่านข้อมูลจากแท็กหลายๆตัว.....	9
2.1.4.1 Switch Off.....	9
2.1.4.2 Slow Down.....	9
2.1.4.3 Carrier Sense.....	10
2.1.5 ประเภทของคลื่นความถี่.....	10
2.1.6 ข้อเสียของเทคโนโลยี RFID.....	10
2.1.7 ปัญหาที่เกิดขึ้นในการเขียน โปรแกรมติดต่อกับ RFID.....	11
2.2 เว็บเซอร์วิส (Web Service).....	11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการเชิงงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ในวงกว้าง

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)

	หน้า
2.2.1 องค์ประกอบของเว็บเซอร์วิส.....	11
2.2.1.1 แอปพลิเคชัน(Application).....	12
2.2.1.2 SOAP(Simple Object Access Protocol).....	12
2.2.1.3 WSDL.....	12
2.2.1.4 UDDI.....	12
2.3 โครงสร้างการทำงานของการติดต่อฐานข้อมูล.....	12
2.3.1 การติดต่อฐานข้อมูลระหว่าง SQL Server กับ VB.Net.....	13
2.3.1.1 โครงสร้างการทำงานใน ADO.NET.....	13
2.3.1.2 เปรียบเทียบ DataSet กับ RecordSet ใน ADO.....	14
2.3.2 การใช้พอกเก็ตพีซี ติดต่อฐานข้อมูล.....	15
2.4 การใช้งานเว็บเซอร์วิส.....	17
2.4.1 ASP.NET ในการสร้างเว็บเซอร์วิส.....	17
2.4.2 การเข้าถึงข้อมูลของเว็บเซอร์วิส.....	18
2.4.3 ปัญหาของโมบายแอปพลิเคชันในการต่อฐานข้อมูลและแนวทางการแก้ไข...	19
บทที่ 3 การวิเคราะห์ระบบและออกแบบระบบใหม่.....	20
3.1 การวิเคราะห์ระบบ.....	20
3.1.1 ปัญหาของระบบดูแลรักษาสัตว์.....	20
3.1.2 แนวทางแก้ไขโดยนำระบบ RFID มาประยุกต์ใช้.....	21
3.1.3 แนวทางแก้ไขโดยนำระบบพอกเก็ตพีซีมาประยุกต์ใช้.....	22
3.2 การออกแบบระบบ.....	23
3.2.1 Use Case Diagram.....	24
3.2.2 Class Diagram.....	28
3.2.3 Sequence Diagram.....	30
3.2.4 ตารางฐานข้อมูลต่างๆที่ใช้ในระบบ.....	38
บทที่ 4 การพัฒนาระบบงาน.....	42
4.1 เครื่องมือที่ใช้ในการพัฒนาระบบ.....	42
4.1.1 ภาษาที่ใช้พัฒนาระบบ.....	42
4.1.2 ระบบปฏิบัติการที่ใช้พัฒนาระบบ.....	42
4.1.3 ฐานข้อมูลที่ใช้พัฒนาระบบ.....	42

## สารบัญ (ต่อ)

	หน้า
4.1.4 เครื่องอ่าน RFID ที่ใช้พัฒนาระบบ.....	42
4.2 หน้าจอของระบบ.....	43
4.2.1 ส่วนการเข้าระบบ.....	43
4.2.2 ส่วนของการเข้าถึงข้อมูลส่วนตัวเชิงแบบปกติ.....	44
4.2.3 ส่วนของการเข้าถึงข้อมูลส่วนตัวเชิงแบบใช้เครื่องอ่าน.....	53
4.2.4 ส่วนของการเพิ่มข้อมูลการรักษาส่วนตัวเชิง.....	54
4.2.5 ส่วนของการเปลี่ยนแปลงข้อมูลส่วนตัว.....	58
4.2.6 ส่วนของการเพิ่มเติมเปลี่ยนแปลงองค์ประกอบ.....	60
4.2.7 ส่วนของการเพิ่มเติมพนักงาน หรือ เปลี่ยนแปลงหน้าที่พนักงาน.....	62
<b>บทที่ 5 สรุปผลและข้อเสนอแนะ.....</b>	<b>64</b>
5.1 สรุปผลการดำเนินงาน.....	64
5.2 ปัญหาที่พบในการพัฒนาระบบ.....	64
5.3 ข้อเสนอแนะ.....	64
5.3.1 เทคโนโลยีอื่นๆที่สามารถนำเข้าใช้ร่วมกับระบบดูแลรักษาตัว.....	64
5.3.1 การประยุกต์เทคโนโลยีร่วมกับระบบดูแลรักษาตัว.....	65
5.3.1 การเพิ่มประสิทธิภาพของระบบ.....	65
บรรณานุกรม.....	66
ประวัติผู้เขียน.....	67

# สารบัญตาราง

ตารางที่	หน้า
3.1 แสดงข้อมูลทั้งหมดของระบบ.....	42
3.2 ตารางข้อมูลอธิบายอาการป่วย(Cure Detail).....	43
3.3 ตารางข้อมูลลูกค้า (Customer) .....	43
3.4 ตารางข้อมูลโรคสัตว์เลี้ยง (DiseaseType) .....	43
3.5 ตารางข้อมูลลูกค้า (Customer) .....	43
3.6 ตารางข้อมูลพนักงาน (Employee).....	43
3.7 ตารางข้อมูลหน้าที่ของพนักงาน (EmployeeRole).....	43
3.8 ตารางข้อมูลสัตว์เลี้ยง (Pet).....	43
3.9 ตารางข้อมูลรายละเอียดการรักษาสัตว์เลี้ยง (PetCureFeed).....	43
3.10 ตารางข้อมูลการให้อาหารและยา (PetFeed).....	43
3.11 ตารางข้อมูลชนิดและพันธุ์สัตว์เลี้ยง (PetType).....	43
3.12 ตารางข้อมูลหน้าที่ (Role).....	43

# สารบัญรูป

รูปที่	หน้า
2.1 องค์ประกอบหลักของเทคโนโลยี RFID.....	6
2.2 เครื่องอ่าน RFID Tagในรูปแบบมือถือ.....	7
2.3 เครื่องอ่าน RFID Tagในรูปแบบช่องประตู.....	8
2.4 การป้องกันการชนกันของข้อมูลด้วยวิธี Switch Off .....	9
2.5 การป้องกันการชนกันของข้อมูลด้วยวิธี Slow Down.....	10
2.6 การป้องกันการชนกันของข้อมูลด้วยวิธี Carrier Sense .....	10
2.7 ระดับการทำงานของ Connected Stage กับ Disconnected Stage .....	14
2.8 โครงสร้าง DataSet .....	15
2.9 โครงสร้าง RecordSet .....	15
2.10 DataTable ใน DataSet .....	16
2.11 การทำงานของโปรโตคอล SOAP .....	18
2.12 การเข้าถึงข้อมูลของ Web Services .....	19
3.1 รูปจำลองระหว่างเครื่อง Pocket PC ที่มี RFID และ Tag ที่ติดไว้บนตัวสัตว์.....	14
3.2 การสื่อสารอุปกรณ์ต่างๆ ภายในที่ทำการรักษาพยาบาลสัตว์.....	23
3.3 แสดงภาพ Sequence Diagram การทำงานรวมของระบบ.....	24
3.4 Class Diagram ของระบบ.....	28
3.5 Sequence Diagram ในกรณีใช้การอ่านจาก RFID Tag.....	30
3.6 Sequence Diagram ในกรณีเลือก Customer และ Pet.....	31
3.7 Sequence Diagram ในกรณี เพิ่มการตรวจสัตว์เลี้ยง.....	32
3.8 Sequence Diagram ในกรณี เพิ่มข้อมูลลูกค้า.....	33
3.9 Sequence Diagram ในการจัดการ โรคของสัตว์เลี้ยง.....	34
3.10 Sequence Diagram การจัดการ การให้อาหารและยาแก่สัตว์เลี้ยง.....	35
3.11 Sequence Diagram การจัดการชนิดของสัตว์เลี้ยง.....	36
3.12 Sequence Diagram การจัดการประวัติพนักงาน.....	37
4.1 แสดงการล็อกอินเข้าสู่ระบบ.....	43
4.2 แสดงหน้าเมนูหลักเพื่อเลือกรายการเข้าใช้งาน.....	44
4.3 แสดงรายชื่อลูกค้าที่บันทึกอยู่ในฐานข้อมูล.....	45
4.4 แสดงหน้าจอการเพิ่มรายชื่อลูกค้าใหม่.....	45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.4.1 แสดงหน้าจอการเพิ่มรายละเอียดลูกค้าใหม่.....	46
4.4.2 แสดงหน้าจอการเพิ่มรายละเอียดที่อยู่ลูกค้าใหม่.....	47
4.5 แสดงหน้าจอรายชื่อสัตว์เลี้ยง.....	47
4.6 แสดงหน้าจอการเพิ่มชื่อสัตว์เลี้ยงใหม่.....	48
4.6.1 แสดงหน้าจอการเพิ่มประเภทและพันธุ์สัตว์เลี้ยงใหม่.....	49
4.6.2 แสดงหน้าจอการเพิ่มรายละเอียดสัตว์เลี้ยงใหม่.....	49
4.6.3 แสดงหน้าจอการเพิ่มรายละเอียด Identify สัตว์เลี้ยงลูกค้าใหม่.....	49
4.7 แสดงหน้าจอประวัติการรักษาของสัตว์เลี้ยง.....	50
4.8 แสดงหน้าจอเครื่องอ่าน RFID .....	53
4.8.1 แสดงหน้าจอรายละเอียดสัตว์เลี้ยง.....	53
4.9 แสดงหน้าจอสร้างประวัติการรักษาใหม่.....	54
4.9.1 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (2).....	55
4.9.2 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (3).....	55
4.9.3 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (4).....	56
4.9.4 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (5).....	57
4.9.5 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (6).....	57
4.9.6 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (7).....	57
4.10 หน้าจอแสดงรายละเอียดผู้พัฒนาโปรแกรม.....	57
4.10.1 หน้าจอแสดงการเพิ่มรายละเอียดผู้พัฒนาโปรแกรม.....	59
4.10.2 แสดงหน้าจอการเปลี่ยนรหัสของผู้พัฒนาโปรแกรม.....	59
4.11.1 แสดงหน้าจอรายชื่อ Nutrient .....	60
4.11.2 แสดงหน้าจอรายชื่อ โรค (Diseases) .....	61
4.11.3 แสดงหน้าจอรายชื่อประเภทสัตว์เลี้ยง .....	61
4.11.4 แสดงหน้าจอรายชื่อหน้าที่ของพนักงาน .....	62
4.11.5 แสดงหน้าจอรายชื่อพนักงาน.....	62
4.12 แสดงหน้าจอเพิ่มรายชื่อของพนักงานใหม่.....	63

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันนี้ระบบบ่งชี้อัตโนมัตินามิ(Automatic Identification) หรือ ออโต้ไอดี ถูกนำมาใช้งานและพัฒนาไปอย่างมากทั้งในภาคอุตสาหกรรม ลอจิสติกส์ กระบวนการผลิตการขนถ่ายวัตถุดิบ ฯลฯ ซึ่งระบบ ออโต้ไอดี มีหลากหลายมาก เช่น Optical Character (OCR) ระบบบาร์โค้ด ระบบไบโอเมตริก และ สมาร์ทการ์ด เป็นต้น แต่เทคโนโลยีที่มาแรงและเป็นระบบมีประสิทธิภาพมากที่สุดตอนนี้คือ เทคโนโลยี RFID โดยเทคโนโลยี RFID จะมีการทำงานใกล้เคียงกับสมาร์ทการ์ดแต่มีความแตกต่างคือ การอ่านและการเขียนข้อมูล ไม่จำเป็นต้องสัมผัสกับเครื่องอ่านโดยตรง (Contactless) ดังนั้นจึงได้มีการนำเทคโนโลยี RFID ไปใช้พัฒนาในด้านต่างๆอย่างแพร่หลาย ซึ่งจะเน้นไปในทางกระบวนการผลิต การขนถ่ายวัตถุดิบ แต่ในเอกสารฉบับนี้ จะนำเสนอการนำ RFID ไปใช้ร่วมกับ ระบบการระบุอัตลักษณ์ของสัตว์ ซึ่งนำไปใช้ร่วมกับระบบส่วนหน้าของโรงพยาบาลสัตว์ เพื่อให้บ่งชี้จำเพาะของสัตว์มีความสะดวกและรวดเร็วมากยิ่งขึ้น

#### ปัญหาของระบบการรักษาสัตว์ (ระบบเก่า) โดยมีขั้นตอนการนำเข้าระบบดังนี้

1. เมื่อเจ้าของนำสัตว์เลี้ยงเข้ามาใน โรงพยาบาล พนักงานต้อนรับจะสอบถามอาการเบื้องต้น และสอบถามว่าเจ้าของเคยนำสัตว์เลี้ยงเข้ามาใช้บริการหรือไม่ แบ่งได้เป็น 2 กรณี

1.1 กรณีลูกค้าเก่าเคยใช้บริการมาพบสัตวแพทย์พนักงานนำบัตรประจำตัว เพื่อนำมาค้นหาบัตรตรวจรักษา

1.2 กรณีเจ้าของเคยมาครั้งแรก พนักงานจะให้เจ้าของกรอกประวัติ พร้อมทั้งพนักงานออกบัตรประจำตัวใหม่ให้กับเจ้าของสัตว์เลี้ยง

2. พนักงานนำบัตรตรวจมาเรียงลำดับตามการเข้ารับการรักษา

3. พนักงานเรียกเจ้าของและสัตว์เลี้ยงเพื่อเข้าพบสัตวแพทย์ตามลำดับการเข้ารับการรักษา

4. เมื่อตรวจรักษาเสร็จ แพทย์จะลงบันทึกประวัติการรักษาของสัตว์เลี้ยง โดยแพทย์จะวินิจฉัย แบ่งเป็น 2 กรณี

4.1 กรณีที่สัตว์เลี้ยงมีความจำเป็นที่จะต้องพักรักษาอยู่ที่โรงพยาบาล

4.1.1 พนักงานจำเป็นจะต้องบันทึกข้อมูลที่เกี่ยวข้องกับการรักษาสัตว์

4.1.2 สัตวแพทย์เข้ามาตรวจเช็คเป็นระยะๆเพื่อรักษาและเปลี่ยนแปลงข้อมูลของ

สัตว์

4.2 กรณีที่สัตว์เลี้ยงไม่มีความจำเป็นที่จะต้องพักรักษาอยู่ที่โรงพยาบาล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เมื่อทำการรักษาเสร็จสิ้นเจ้าหน้าที่จะติดต่อเจ้าของสัตว์เลี้ยงเพื่อมารับสัตว์กลับคืน
6. พนักงานเรียกเจ้าของสัตว์เลี้ยงเพื่อชำระเงิน

## 1.2 จุดมุ่งหมายและวัตถุประสงค์ของโครงการ

โครงการพัฒนาระบบงานในหัวข้อเรื่อง การพัฒนาระบบดูแลรักษาสัตว์ด้วย RFID (Radio Frequency Identification) มีวัตถุประสงค์หลักดังนี้

1. ศึกษาถึงประโยชน์และปัญหาของระบบ RFID ซึ่งมีผลต่อการระบุอัตลักษณ์ของวัตถุชนิดที่ต่างกัน
2. ศึกษาและเปรียบเทียบระหว่าง บาร์โค้ด 2 มิติ และ RFID ที่มีข้อดีและข้อเสียแตกต่างกัน โดยขึ้นอยู่กับวัตถุประสงค์ในการนำไปใช้งานและความคุ้มค่าที่จะได้รับ
3. ศึกษาแนวทางการพัฒนา ข้อจำกัด ปัญหา ระหว่างในการพัฒนาระบบดูแลรักษาสัตว์ โดยใช้ RFID ให้ระบบมีการพัฒนาในด้านประสิทธิภาพ
4. การศึกษาและออกแบบระบบดูแลรักษาสัตว์ โดยนำมาประยุกต์ใช้กับ RFID โดยใช้ Use Case Model เป็นเครื่องมือออกแบบเชิงแนวคิด และรวมไปถึงการออกแบบ Interface ของ Pocket PC ให้เหมาะสมกับการใช้งานเพื่อได้ประโยชน์สูงสุด

## 1.3 ขอบเขตของการพัฒนาระบบ

โครงการพัฒนาระบบงานในหัวข้อเรื่อง การพัฒนาระบบดูแลรักษาสัตว์ด้วย RFID นี้ จะเป็นการนำเทคโนโลยี พ็อกเก็ตพีซี, RFID และ เว็บเซอร์วิส มาใช้ร่วมกัน โดยขอบเขตของการพัฒนาระบบจะประกอบไปด้วย

1. การพัฒนาโปรแกรมให้อยู่ในรูปแบบของ โมบายแอปพลิเคชัน
2. พัฒนาระบบเพื่อรองรับการทำงานของฐานข้อมูลผ่าน เว็บเซอร์วิส
3. พัฒนาระบบในส่วนติดต่อระหว่าง ฮาร์ดแวร์ RFID กับ โมบายแอปพลิเคชัน
4. พัฒนาระบบในส่วนของการอ่านและการเขียนด้วย RFID กับ แท็ก ผ่าน โมบายแอปพลิเคชัน
5. Login แสดงถึงกระบวนการพิสูจน์ตัวตนก่อนจะเข้าทำการใช้ระบบ
6. จัดการข้อมูลสัตว์ แสดงถึงกระบวนการจัดการข้อมูลสัตว์ เช่น ชื่อสัตว์ ประเภทสัตว์ เป็นต้น
7. บันทึกอาการป่วยของสัตว์ แสดงถึงกระบวนการบันทึกอาการป่วยของสัตว์ในแต่ละครั้ง
8. ดูแลการให้อาหารกับสัตว์ แสดงถึงกระบวนการกำหนดการให้อาหารและวัคซีนกับสัตว์
9. จัดการข้อมูล RFID แสดงถึงกระบวนการกำหนดรหัสข้อมูลเพื่อบ่งบอกถึงสัตว์แต่ละตัวที่เข้าทำการรักษา ซึ่งจะเกิดขึ้นในกรณีที่สัตว์มีความจำเป็นจะต้องรักษาอยู่ในโรงพยาบาล

## 1.4 ขั้นตอนการดำเนินโครงการ

การดำเนินโครงการพัฒนาระบบงานในหัวข้อเรื่อง การพัฒนาระบบดูแลรักษาสัตว์ด้วย RFID นั้น มีขั้นตอนในการดำเนินโครงการ ดังต่อไปนี้

1. กำหนดวัตถุประสงค์และขอบเขตของโครงการที่จะพัฒนา
2. ศึกษาขั้นตอนและหลักการในการติดต่อระหว่าง Mobile Application กับ ฮาร์ดแวร์ RFID
3. ศึกษาขั้นตอนและหลักการของการติดต่อฐานข้อมูลของ Mobile Application กับ ฐานข้อมูล โดยผ่าน Web Service
4. ทำการออกแบบระบบเพื่อให้ครอบคลุมวัตถุประสงค์ของการพัฒนาระบบ
5. จัดทำขั้นตอนและคู่มือในการใช้งานระบบที่พัฒนาขึ้นมา
6. ทำการประเมินผลและวิเคราะห์สิ่งที่ได้รับจากการพัฒนาโครงการนี้ขึ้นมา

## 1.5 ประโยชน์ที่คาดว่าจะได้รับการพัฒนาโครงการ

1. ความแม่นยำของการอ่านข้อมูล เนื่องจากความเสถียรของตัวแท็ก ซึ่งมีมากกว่า บาร์โค้ด (สามารถถูกทำให้ภาพบาร์โค้ดเสียหายได้ง่ายทำให้การอ่านข้อมูลคลาดเคลื่อนหรือไม่สามารถอ่านได้)
2. ลดความเสี่ยงในการปฏิบัติงานของสัตวแพทย์ โดยเมื่อสัตวแพทย์จะทำการอ่านข้อมูลจากสัตว์ จะไม่จำเป็นต้องเข้าไปใกล้ตัวสัตว์มากจนอาจก่อให้เกิดอันตรายต่อสัตวแพทย์ได้
3. หากเกิดในกรณีที่ข้อมูลสูญหาย จะยังมีข้อมูลสำรองไว้ใน แท็ก อยู่บางส่วนเพื่อให้การกู้ข้อมูลทำได้สะดวกมากยิ่งขึ้น
4. เพิ่มความรวดเร็วในการทำงานของพนักงานต่อสัตวแพทย์
5. เพิ่มความรวดเร็วในการทำงานของพนักงานต่อสัตวแพทย์
6. เพิ่มความรวดเร็วในการทำงานเข้าถึงข้อมูลของสัตว์ต่อสัตวแพทย์
7. ช่วยลดต้นทุนค่าใช้จ่ายเกี่ยวกับเอกสารต่างๆ ไม่ว่าจะเป็นสาเหตุที่มาจาก การสูญหายของเอกสาร หรือ จะเป็นการเปลี่ยนแปลงข้อมูลใหม่ต่างๆ
8. ช่วยลดต้นทุนค่าใช้จ่ายในกรณีหากใช้เทคโนโลยีทดแทนเช่น หากใช้บาร์โค้ดในการประยุกต์ใช้ จะมีข้อเสีย โดยในกรณีแรก หากต้องการความแข็งแรงของบาร์โค้ดก็จำเป็นต้องพิมพ์บาร์โค้ดไว้ในวัตถุที่มีความทนทานและบาร์โค้ดชนิดนี้จะเป็นประเภทที่เปลี่ยนแปลงไม่บ่อย กรณีที่สอง หากต้องการบาร์โค้ดที่ยืดหยุ่นก็จะใช้การพิมพ์บาร์โค้ด ไปลงบนแผ่นกระดาษเพื่อลดต้นทุนเมื่อเกิดการเปลี่ยนแปลง หากเป็นในกรณีของสมาร์ทการ์ด ซึ่งจะมีคุณสมบัติของ Contactfull (การระบุตัวตนแบบสัมผัส) นั่นคือหากจำเป็นต้องการที่จะค้นข้อมูลจาก สมาร์ทการ์ด จำเป็นจะต้องนำเครื่องอ่านเข้าไปใกล้ตัวสัตว์ โดยหากสัตวแพทย์ไม่มีความจำเป็นที่จะต้องเข้าหาสัตว์ (ในเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่สัตว์มีความดุร้าย หรือ คิดเชื่อร้ายแรง) ซึ่งในทุกๆกรณี ที่กล่าวมาด้วยคุณสมบัติของเทคโนโลยี RFID สามารถแก้ไขปัญหาดังกล่าวได้อย่างสมบูรณ์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# กระบวนการเตรียมข้อมูลและทฤษฎีที่เกี่ยวข้อง

### 2.1 RFID (Radio Frequency Identification)

#### 2.1.1 ความเป็นมาปัญหาของเทคโนโลยี RFID

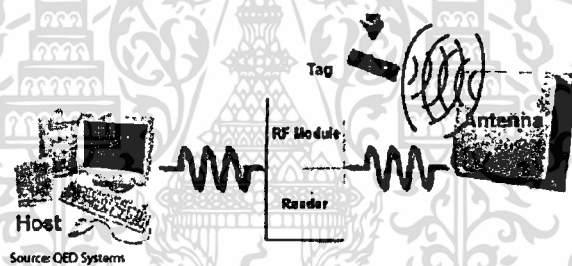
ในอุตสาหกรรมมอติคจนถึงปัจจุบัน ได้มีการใช้เทคโนโลยีเข้าเพื่อเพิ่มประสิทธิภาพ ทั้งในด้านความเร็วและประสิทธิภาพในเชิงคุณภาพ โดยการระบุตัวตนของวัตถุเป็นปัญหาสำคัญอย่างหนึ่งที่สำคัญ ซึ่งส่งผลกระทบต่อเวลาที่ล่าช้าในการผลิต อีกทั้งความคลาดเคลื่อนและไม่แน่นอนที่อาจเกิดขึ้นได้สูง อุตสาหกรรมหลายๆ ด้านจึงได้นำเทคโนโลยีบาร์โค้ดเข้ามาเพื่อแก้ปัญหการระบุตัวตนของวัตถุ เพื่อให้เข้าถึงข้อมูลในตัววัตถุนั้นง่ายขึ้น โดยใช้หลักการนำตัวเลขมาเพื่อไว้อ้างอิงกับฐานข้อมูล แต่รูปแบบของตัวเลขนั้นได้เปลี่ยนมาใช้แท่งรหัสซึ่งเรียกว่า บาร์โค้ดนั่นเอง บาร์โค้ดเข้ามามีบทบาทต่ออุตสาหกรรมหลายๆด้านอยู่ และทำให้การระบุตัวตนผ่านเครื่องอ่านบาร์โค้ดมีประสิทธิภาพมากกว่าระบบเก่ามากขึ้น โดยเฉพาะการจะใช้บาร์โค้ดส่วนมากจำเป็นที่จะต้องมีการมีฐานข้อมูลที่ใหญ่เพื่อรองรับบาร์โค้ดจำนวนมาก ซึ่งปัญหาจากการใช้บาร์โค้ดก็เกิดขึ้น เนื่องจากเมื่อจะทำการระบุตัวตนของวัตถุแต่ละชิ้นจะต้องการแปลงรหัสแท่งเป็นตัวเลขแล้วทำการอ้างอิงจากฐานข้อมูลซึ่งเท่ากับว่าในตัวบาร์โค้ดเองแทบจะไม่มีข้อมูลใดๆอยู่เลยนอกจากตัวเลขที่ไว้สำหรับอ้างอิง แต่ข้อมูลที่แท้จริงอยู่ในส่วนของฐานข้อมูล ซึ่งหากจะทำการเปลี่ยนข้อมูลใดๆ จะต้องทำการอ้างอิงกับฐานข้อมูลตลอด ซึ่งหากเป็นวัตถุหรือสิ่งมีชีวิตที่จะต้องทำการเปลี่ยนแปลงข้อมูลบ่อยครั้ง การใช้แท่งรหัสหรือบาร์โค้ด จึงเป็นการเสียเวลาในกระบวนการมาก และปัญหาอีกด้านหนึ่งของบาร์โค้ดคือ เมื่อจะทำการอ่านบาร์โค้ดจะต้องทำการอ่านจากบาร์โค้ดโดยตรง โดยห้ามมีสิ่งกีดขวางกั้นหน้าบาร์โค้ด ดังนั้นบาร์โค้ดถูกพิมพ์ไว้อยู่ข้างหน้าวัตถุตลอดซึ่งมีโอกาสสูงที่จะถูกทำให้เสียหายได้ และการที่พิมพ์บาร์โค้ดไว้อยู่ข้างหน้าวัตถุจึงมีโอกาสเสี่ยงสูงหากมีการคัดลอกเพื่อนำไปศึกษาเพื่อไว้ใช้ประโยชน์ด้านอื่นๆ ได้เช่น การขโมย การเปลี่ยนแปลงบาร์โค้ด การลอกรหัสบาร์โค้ด เป็นต้น ต่อมาจึงได้มีการนำเทคโนโลยี อาร์เอฟไอดี (RFID) หรือ เรียกว่า Radio Frequency Identification ซึ่งการระบุตัวตนโดยใช้คลื่นวิทยุ ซึ่งสามารถขจัดปัญหาการอ้างอิงข้อมูลจากฐานข้อมูลเพียงอย่างเดียว มาเป็นการอ้างอิงข้อมูลในระดับหนึ่งจากตัวรับสัญญาณ ซึ่งทำให้ความเร็วในการเปลี่ยนแปลงข้อมูลสูงขึ้นมากทีเดียว อีกทั้ง แท้ก็ ไม่จำเป็นจะต้องวางอยู่ข้างหน้าวัตถุ แต่สามารถวางอยู่ที่ใดก็ได้ในวัตถุนั้นๆ

## 2.1.2 องค์ประกอบของ RFID

ระบบระบุลักษณะของวัตถุด้วยคลื่นความถี่วิทยุโดยมีวัตถุประสงค์หลักเพื่อนำไปใช้งานแทนที่ระบบบาร์โค้ดโดยจุดเด่นของ RFID อยู่ที่การอ่านข้อมูลจากแท็ก (แท็ก) ได้หลายๆ แท็กแบบไร้สัมผัสและสามารถอ่านค่าได้แม้ในสภาพที่แวดล้อมรอบข้างไม่กระทบต่อความเปียกชื้นและแรงสั่นสะเทือน การกระทบกระแทกยิ่งไปกว่านั้นยังสามารถอ่านข้อมูลได้ด้วยความเร็วสูงโดยข้อมูลจะถูกเก็บไว้ใน ไมโครชิปที่อยู่ในแท็ก ซึ่งมีหลายชนิดขึ้นอยู่กับความต้องการใช้

### 2.1.2.1 ส่วนประกอบของเทคโนโลยี RFID

ในระบบ RFID จะมีองค์ประกอบหลักๆ อยู่ 2 ส่วน 1. Transponder/แท็ก เป็นตัวที่ใช้ติดกับวัตถุต่างๆ โดยแท็กจะบันทึกข้อมูลที่เกี่ยวข้องกับวัตถุนั้นไว้ 2. Interrogator/Reader เครื่องสำหรับอ่านและเขียนข้อมูลภายในแท็กด้วยคลื่นความถี่วิทยุ โดยสามารถอ่านได้ที่ละหลายๆแท็กและสามารถอ่านข้อมูลได้ครบเท่าที่ระยะของคลื่นของเครื่องอ่านไปถึงได้ โดยไม่จำเป็นจะต้องเห็นแท็กที่ติดกับวัตถุนั้นๆ



รูปที่ 2.1 องค์ประกอบหลักของเทคโนโลยี RFID

#### 2.1.2.1.1 แท็ก (แท็ก)

โครงสร้างภายในของแท็กจะประกอบด้วย 2 ส่วนใหญ่ๆ คือ 1. ขดลวดขนาดเล็กซึ่งทำหน้าที่เป็นสายอากาศ (Antenna) สำหรับรับส่งสัญญาณคลื่นความถี่วิทยุและสร้างพลังงานป้อนให้ส่วนของไมโครชิป (Microchip) ที่ทำหน้าที่เก็บข้อมูลของวัตถุเช่นรหัสสินค้าโดยทั่วไปตัวแท็กอาจอยู่ในชนิดทั้งเป็นกระดาษ แผ่นฟิล์ม พลาสติก มีขนาดและรูปร่างกันไป ทั้งนี้ขึ้นอยู่กับวัสดุที่จะนำมาไปติดซึ่งมีหลายรูปแบบโดยอาจจะเป็นขนาดเท่าบัตรเครดิต เหยียบู ฉลากสินค้า เป็นต้น โดยแท็กจะสามารถแบ่งได้เป็น 2 ชนิดใหญ่ และแต่ละชนิดมีหลักการทำงานและราคาที่แตกต่างกัน

- Passive แท็ก

แท็กชนิดนี้ไม่จำเป็นจะต้องอาศัยแหล่งจ่ายไฟฟ้าภายนอกใดๆ เพราะภายในแท็กจะมีวงจรกำเนิดไฟฟ้าเหนี่ยวนำขนาดเล็กเป็นแหล่งจ่ายไฟในตัวอยู่ทำให้การอ่าน ข้อมูลทำได้ไม่ไกลมากนัก ระยะอ่านสูงสุดอยู่ที่ประมาณ 1 เมตร ขึ้นอยู่กับความแรงของเครื่องส่งและคลื่นความถี่วิทยุที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้โดยแท็กชนิดนี้จะมีหน่วยความจำอยู่ที่ประมาณ 16-1,024 ไบต์ ซึ่งตัวแท็กมีขนาดเล็กและน้ำหนักเบาอีกทั้งราคาต่อหน่วยที่ต่ำ

- Active แท็ก

แท็กชนิดนี้จำเป็นจะต้องอาศัยแหล่งจ่ายไฟฟ้าจากภายนอกเพื่อเข้ามากระตุ้นวงจรภายในให้สามารถทำงานได้ แท็กชนิดนี้จะมีหน่วยความจำภายในขนาดใหญ่ได้ถึง 1 เมกะไบต์ และสามารถอ่านได้ในระยะไกลสูงสุดประมาณ 10 เมตร ถึงแม้ว่าแท็กจะมีคุณสมบัติข้อดีหลายอย่างแต่ในทางกลับกันกลับมีข้อเสียด้านปัจจัยภายนอก เช่น ราคาต่อหน่วยที่แพง และขนาดที่ใหญ่ อีกทั้งยังมีระยะเวลาทำงานที่จำกัดอีกด้วย

นอกจากนี้แท็กยังถูกแบ่งเป็นอีก 3 ชนิดเพื่อการใช้งานที่แตกต่างกัน คือ

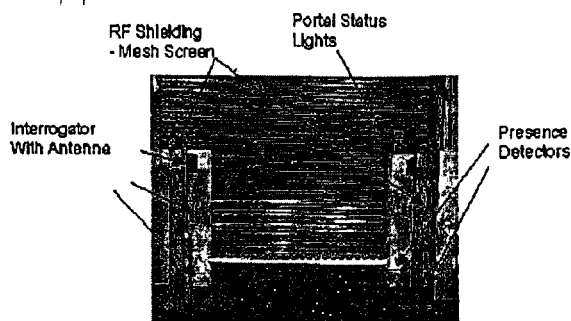
1. แบบที่สามารถถูกและเขียนข้อมูลได้อย่างอิสระ (Read-Write)
2. แบบที่สามารถเขียนได้เพียงครั้งเดียวเท่านั้นแต่สามารถอ่านได้อย่างอิสระ (Write-Once Read Many)
3. แบบอ่านได้เพียงอย่างเดียว (Read Only)

#### 2.1.2.1.2 เครื่องอ่าน (Reader)

โครงสร้างภายในของแท็กจะประกอบด้วย 2 ส่วนใหญ่ๆ คือ 1. ขดลวดขนาดเล็กซึ่งทำหน้าที่เป็นสายอากาศ (Antenna) สำหรับรับส่งสัญญาณคลื่นความถี่วิทยุและสร้างพลังงานป้อนให้ส่วนของไมโครชิป (Microchip) ที่ทำหน้าที่เก็บข้อมูลของวัตถุเช่นรหัสสินค้า โดยทั่วไปตัวแท็กอาจอยู่ในชนิดทั้งเป็นกระดาษ แผ่นฟิล์ม พลาสติก มีขนาดและรูปร่างต่างๆกันไป ทั้งนี้ขึ้นอยู่กับวัสดุที่จะนำมาไปติดซึ่งมีหลายรูปแบบโดยอาจจะเป็นขนาดเท่าบัตรเครดิต เหยียดยูนี ลากสินค้า เป็นต้น โดยแท็กจะสามารถแบ่งได้เป็น 2 ชนิดใหญ่ และแต่ละชนิดมีหลักการทำงานและราคาที่แตกต่างกัน



รูปที่ 2.2 เครื่องอ่าน RFID แท็กในรูปแบบมือถือ



รูปที่ 2.3 เครื่องอ่าน RFID แท้กในรูปแบบช่องประตู

โดยหน้าที่ของเครื่องอ่านก็คือการเชื่อมต่อเพื่อเขียนหรืออ่านข้อมูลลงในแท็กด้วยสัญญาณความถี่วิทยุโดยภายในเครื่องอ่านจะประกอบไปด้วยองค์ประกอบหลักดังนี้

1. ภาครับและส่งสัญญาณวิทยุ
2. ภาคสร้างสัญญาณพาหะ
3. ขดลวดที่ทำหน้าที่เป็นสายอากาศ
4. วงจรจูนสัญญาณ
5. หน่วยประมวลผลข้อมูลและภาคติดต่อกับคอมพิวเตอร์

โดยหน่วยประมวลผลข้อมูลที่อยู่ภายในเครื่องอ่านจะเป็นไมโครคอนโทรลเลอร์ซึ่งโครงสร้างที่อยู่ภายในโปรแกรมจะทำหน้าที่ถอดรหัสข้อมูล (Decoding) ที่ได้รับโดยตัวของมันเอง อีกทั้ง ลักษณะ ขนาด และรูปร่างของเครื่องอ่านจะแตกต่างกันไปตามประเภทของการใช้งาน

### 2.1.3 ขั้นตอนการทำงานระหว่างเครื่องอ่านกับแท็ก

1. ตัวเครื่องอ่านจะทำการส่งสัญญาณวิทยุอย่างต่อเนื่องหรือเป็นจังหวะและรอคอยสัญญาณตอบจากตัวแท็ก
2. เมื่อแท็กได้รับสัญญาณคลื่นวิทยุที่ส่งมาจากเครื่องอ่านในระดับที่เพียงพอ ก็จะทำการเหนี่ยวนำเพื่อสร้างพลังงานป้อนให้แท็กทำงาน โดยแท็กจะสร้างสัญญาณนาฬิกาเพื่อกระตุ้นให้วงจรภาคดิจิทัลในแท็กทำงาน
3. วงจรภาคดิจิทัลจะไปอ่านข้อมูลจากหน่วยความจำภายในและทำการเข้ารหัสข้อมูลแล้วส่งไปยัง ภาคแอนะล็อกที่ทำหน้าที่มอดูเลตข้อมูล
4. ข้อมูลที่ถูกมอดูเลตจะถูกส่งไปยังขดลวดที่ทำหน้าที่เป็นสายอากาศ เพื่อส่งไปยังเครื่องอ่าน
5. เครื่องอ่านจะสามารถตรวจจับสัญญาณการเปลี่ยนแปลงของแอมพลิจูด (Envelope Detector) และใช้ พีค ดีเทกเตอร์ (Peak Detector) ในการแปลงสัญญาณข้อมูลที่มีมอดูเลตแล้วจากแท็ก
6. เครื่องอ่านจะถอดรหัสข้อมูลและส่งไปยังคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมต่อไป

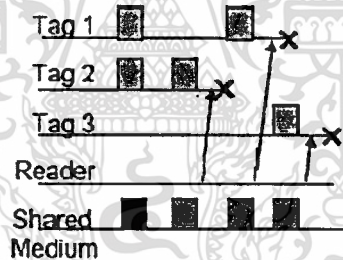
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.4 การอ่านข้อมูลจากแท็กหลายๆตัว

เทคโนโลยี RFID ในอะโลฮาโปรโตคอล (Aloha Protocol) ซึ่งใช้ TDMA (Time Division Multiple Access) ในการอ่านแท็กหลายๆตัวในเวลาเดียวกันซึ่งอะโลฮา โปรโตคอลได้ถูกเพิ่มศักยภาพในการป้องกันการชนกันของข้อมูลด้วยวิธีต่างๆกัน 3 วิธี

#### 2.1.4.1 Switch Off

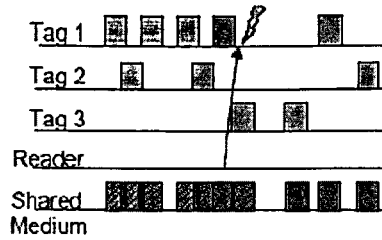
เป็นหลักการในการอ่านข้อมูลที่ส่งจากแท็กมาเพื่อเกิด การชนกันแบบ Completed Collision (ชนกันแบบสมบูรณ์) โปรโตคอล Aloha ของเครื่องอ่านแท็ก จะทำการ หยุดอ่านข้อมูลที่ออกจากแท็กทั้งหมดแล้วเริ่มอ่านใหม่อีกครั้งหนึ่งซึ่งเป็นการป้องกันการชนกันแบบ Completed Collision เช่น เมื่อเครื่องอ่าน RFID ทำการอ่านข้อมูลเกิดการชนกันระหว่างข้อมูลของ แท็ก 1 และ แท็ก 2 ทำให้เครื่องอ่านทำการยกเลิกการอ่านทั้งหมดและกำหนดค่าในแท็กทำการชนกันนั้นเป็น quiet state ซึ่งอยู่ในสถานะไม่สามารถรับส่งสัญญาณได้ชั่วขณะหนึ่งและจะสามารถอ่านได้ต่อเมื่อเครื่องอ่านทำการส่งคำสั่ง wake up แต่มีในกรณีที่แท็กไม่สามารถรับคำสั่ง wake up ที่ถูกส่งมาได้ แท็กที่อยู่ในสถานะ quiet state นั้นจะรอจนกว่า quiet state จะหมดเวลาจึงเปลี่ยนสถานะได้



รูปที่ 2.4 การป้องกันการชนกันของข้อมูลด้วยวิธี Switch Off

#### 2.1.4.2 Slow down

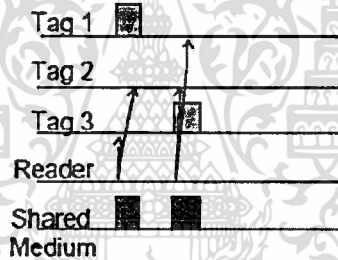
เป็นหลักการในการอ่านข้อมูลที่ส่งจากแท็กมาเพื่อเกิด การชนกันแบบ Partial Collision (ชนกันแบบข้อมูลเหลื่อมล้ำกัน) โปรโตคอล Aloha ของเครื่องอ่านแท็ก จะทำการยกเลิกการอ่านข้อมูลที่ได้อ่านไปแล้ว แล้วชะลอการอ่านข้อมูลลงเพื่อป้องกันความเหลื่อมล้ำของข้อมูล เช่น เมื่อเครื่องอ่าน RFID แท็กทำการอ่านข้อมูลแล้วข้อมูล แท็ก 1, แท็ก 2 และ แท็ก 3 เกิดการเหลื่อมล้ำกันของข้อมูลเครื่องอ่าน RFID แท็ก จึงทำการลดความเร็วในการอ่านของข้อมูลลง



รูปที่ 2.5 การป้องกันการชนกันของข้อมูลด้วยวิธี Slow Down

#### 2.1.4.3 Carrier Sense

เมื่อเครื่องอ่าน RFID จะทำการอ่านข้อมูลจากแท็ก ว่ามีการส่งข้อมูลมาหรือไม่ ก่อนที่จะทำการอ่านข้อมูลที่ละแท็กเกิด เช่น เครื่องอ่านจะทำการอ่านข้อมูล เมื่อมีข้อมูลจาก แท็ก 1 มาเครื่องอ่านจะทำการตรวจสอบข้อมูลจากแท็ก 2 และ 3 ไม่ให้มีการส่งข้อมูลเพื่อป้องกันการชนกันของข้อมูล



รูปที่ 2.6 การป้องกันการชนกันของข้อมูลด้วยวิธี Carrier Sense

#### 2.1.5 ประเภทของคลื่นความถี่

- ความถี่ต่ำ 125 - 134 KHz ระยะ < 0.5 เมตร
- ความถี่สูง 13.553 – 13.567 MHz ระยะ 1 เมตร
- ความถี่สูงมาก 400 - 1000 MHz ระยะ 3 เมตร
- คลื่นไมโครเวฟ 2.45 GHz ระยะ >10 เมตร

#### 2.1.6 ข้อเสียของเทคโนโลยี RFID

- ต้นทุนสูง

ถึงแม้ว่าแท็กของ RFID มีราคาถูก แต่ด้วยความสามารถในการอ่านสัญญาณของเครื่องอ่านมีระยะสั้นเมื่อนำระบบ RFID ไปใช้ในอุตสาหกรรมที่มีพื้นที่ขนาดใหญ่จะต้องติดตั้งเครื่องอ่านไว้ทุกกระยะหรือทุกที่ที่ต้องมีการอ่านข้อมูลจากแท็กซึ่งจะทำให้สิ้นเปลืองทรัพยากรในการอ่านอย่างไม่คุ้มค่า

- ปัญหาในการอ่านสัญญาณจากวัตถุ

เนื่องจากปัญหาของตัวแท็กซึ่งมีปฏิกิริยากับวัตถุที่เป็นโลหะเมื่อนำ RFID แท็กไปติดไว้กับวัตถุที่เป็นโลหะ เครื่องอ่านจะไม่สามารถอ่านได้วิธีทางแก้คือการติด RFID แท็กโดยห่างจากวัตถุที่เป็นโลหะอย่างน้อย 3 มิลลิเมตร

### 2.1.7 ปัญหาที่เกิดขึ้นในการเขียนโปรแกรมต่อกับ RFID

เนื่องจากระบบที่พัฒนาจะเป็นการพัฒนาบน โมบายแอปพลิเคชัน โดยการนำเสนอจะอยู่ในรูปแบบของ อิมูเลเตอร์ (พอกเก็ตพีซี 2003 อิมูเลเตอร์) ซึ่งมีข้อจำกัดหลายๆประการ เช่นในการจะเปิด Serial Port ของ พอกเก็ตพีซี หากใช้ Visual Basic .Net 2003 หรือ เวอร์ชันต่ำกว่าลงไป จะไม่มี Components ในส่วนของ Serial Port รองรับ ซึ่งหากไม่มี Components ดังกล่าวรองรับ จะทำให้การเขียนโปรแกรมมีความยากลำบากมาก ดังนั้นการพัฒนาระบบจึงต้องขึ้นอยู่กับ โมบายแอปพลิเคชันที่มี Serial Port Component รองรับด้วย เช่น Visual Basic .Net 2005 เป็นต้น

ในส่วนของฮาร์ดแวร์ RFID หากทำการติดต่อด้วย แอปพลิเคชัน ปกติ จะสามารถติดต่อดีตามปกติโดยใช้หลักการ ส่งคำสั่งตัวเลขไปเพื่อสั่งตัวฮาร์ดแวร์ให้ทำการอ่านหรือเขียน แท็ก แต่หากเป็น โมบายแอปพลิเคชัน การทำงานจะแตกต่างกันเล็กน้อย โดยหลังจากส่งคำสั่งตัวเลขไปเพื่อสั่งให้ตัวฮาร์ดแวร์ทำการอ่านหรือเขียน แท็ก จะต้องตามด้วย คำสั่งที่บังคับให้ตัวฮาร์ดแวร์ RFID ส่ง ACK (Acknowledge) ซึ่งเป็นการบอกกับตัว โมบายแอปพลิเคชัน ว่าข้อมูลส่งกลับไปครบถ้วนเพื่อหลีกเลี่ยงปัญหาการอ่านข้อมูลซ้ำซ้อนหรือไม่ครบถ้วน ซึ่งจะเกิดขึ้นได้บ่อยครั้ง

## 2.2 เว็บเซอร์วิส

เว็บเซอร์วิส คือสิ่งทำให้แอปพลิเคชัน กลายเป็นเว็บแอปพลิเคชัน กล่าวคือเว็บเซอร์วิสสามารถทำให้แอปพลิเคชันอื่นบนคอมพิวเตอร์เครื่องอื่น เรียกใช้งาน แอปพลิเคชันของตนเองได้ แม้ว่าจะอยู่บนเครื่องคอมพิวเตอร์ คนละแพลตฟอร์ม หรือใช้ภาษาที่ใช้พัฒนาแอปพลิเคชันต่างกันก็ตาม

### 2.2.1 องค์ประกอบของ เว็บเซอร์วิส

เว็บเซอร์วิส ประกอบด้วยส่วนต่าง ๆ ดังนี้

- แอปพลิเคชัน - โปรแกรมที่ทำหน้าที่ให้บริการ
- SOAP - โพรโตคอลที่ใช้ในการสื่อสารระหว่างแอปพลิเคชัน
- WSDL - ไฟล์ที่เก็บวิธีการเรียกใช้งาน เว็บเซอร์วิส
- UDDI - ไคลเอนท์ที่รวบรวม WSDL จำนวนมากเข้าไว้ด้วยกัน

### 2.2.1.1 แอปพลิเคชัน

แอปพลิเคชันที่จะให้บริการเว็บเซอร์วิส ควรอยู่บนเครื่องเว็บเซิร์ฟเวอร์ที่เปิดให้บริการตลอดเวลา ที่สามารถติดต่อด้วยโพรโทคอล HTTP ได้ และพัฒนาด้วยภาษาที่มีความสามารถจัดการกับ SOAP โดยอาจเป็น โมดูลเสริม หรือมีคลาสให้เรียกใช้งานได้

### 2.2.1.2 SOAP(Simple Object Access Protocol)

SOAP คือ โพรโทคอลหรือระเบียบวิธีในการสื่อสารระหว่างเว็บเซอร์วิส โดยใช้ข้อมูลที่กำหนดรูปแบบด้วยภาษา XML ทำให้เว็บเซอร์วิสสามารถสื่อสารกันได้แม้ว่า จะอยู่บนเครื่องคอมพิวเตอร์คนละแพลตฟอร์ม หรือพัฒนาด้วยภาษาโปรแกรมที่ต่างกัน

เมื่อผู้พัฒนาแอปพลิเคชัน ต้องการใช้งาน เว็บเซอร์วิส ผู้พัฒนาก็เพียงแค่เขียนโปรแกรมเพื่อติดต่อกับโมดูล SOAP ในภาษาที่ตนใช้ จากนั้น SOAP ก็สร้าง SOAP message เพื่อติดต่อกับแอปพลิเคชันปลายทางให้โดยอัตโนมัติ

### 2.2.1.3 WSDL

เว็บ Services Description Language (WSDL) คือ เอกสาร XML ที่อธิบายรายละเอียดในการติดต่อกับเว็บเซอร์วิส เพื่อให้ แอปพลิเคชันที่ต้องการเรียกใช้เว็บเซอร์วิสรู้ว่าเซิร์ฟเวอร์นั้น ให้บริการอะไรบ้าง และจะติดต่อได้อย่างไร

### 2.2.1.4 UDDI

UDDI (Universal Description, Discovery and Integration) เป็น ไคเรกทอรี ที่เก็บรวบรวม เว็บเซอร์วิส ที่มีการลงทะเบียนไว้ ซึ่งอาจรวมไปถึงบริการอื่นๆที่เป็นอิเล็กทรอนิกส์ และไม่เป็นอิเล็กทรอนิกส์ด้วย UDDI จะเก็บรวบรวมข้อมูลของ เว็บ เซอร์วิส ต่างๆไว้ในรูปแบบ WSDL

หน้าที่ของ UDDI จะคล้ายกับ เว็บไคเรกทอรี กล่าวคือ UDDI ช่วยให้ผู้พัฒนา เว็บเซอร์วิส ได้ประกาศหรือประชาสัมพันธ์บริการของตนเองสู่สาธารณะ และช่วยให้ผู้ใช้ เว็บ เซอร์วิส ค้นพบ เว็บเซอร์วิส ที่ต้องการใช้งาน

## 2.3 โครงสร้างการทำงานของการติดต่อฐานข้อมูล

ในการพัฒนานี้ใช้โปรแกรม Visual Basic .NET เป็นตัวออกแบบหน้าจอและเขียนโค้ด ของพอกเก็ตพีซี เขียนด้วย Visual Studio.Net ซึ่งจะนำไปรันบน พอกเก็ตพีซี โดยที่จะเชื่อมต่อกับเซิร์ฟเวอร์ฐานข้อมูล (MS SQL Server 2000) ที่รัน อยู่บนเครื่อง พีซี

การติดต่อสื่อสารระหว่างพีซี กับ SQL Server 2000 จะติดต่อผ่าน ADO ส่วน พอกเก็ตพีซี กับ SQL Server 2000 ต้องติดต่อผ่าน ADO.Net แล้วจึงติดต่อฐานข้อมูลได้ โดยจะส่ง SQL Query (String Query)

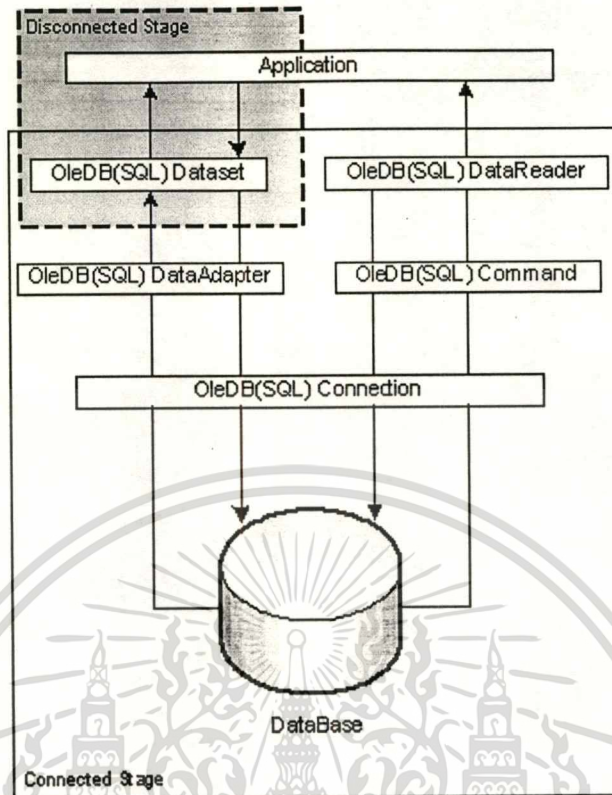
### 2.3.1 การติดต่อฐานข้อมูลระหว่าง SQL Server กับ VB .Net

#### 2.3.1.1 โครงสร้างการทำงานใน ADO.Net

เทคโนโลยี ADO ใน Visual Basic 6 นั้นเป็นเครื่องมือในการติดต่อและจัดการข้อมูลในแบบ Connected Database ก็จะต้องเปิดการเชื่อมต่อกับฐานข้อมูลไว้ตลอดเวลาในการทำงานใดๆ เกี่ยวกับฐานข้อมูล ซึ่งอาจทำให้สิ้นเปลืองทรัพยากรบน เซิร์ฟเวอร์ สูง ดังนั้นในเทคโนโลยี ADO.Net จึงหันมาใช้ในการจัดการข้อมูลแบบ Disconnected Database ก็สามารจัดการข้อมูลในฐานข้อมูลโดยไม่ต้องเปิดการเชื่อมต่อไว้ตลอด โดยข้อมูลจะเก็บไว้ในหน่วยความจำก่อน จากนั้นจะตัดการเชื่อมต่อกับฐานข้อมูล ทำให้การกระทำใดๆ หลังจากนี้จะเป็นการกระทำกับหน่วยความจำเท่านั้น เมื่อเสร็จกระบวนการแล้วจะบันทึกลงฐานข้อมูล ซึ่งจะช่วยให้ประหยัดทรัพยากรของระบบและเพิ่มประสิทธิภาพการทำงานของ เซิร์ฟเวอร์

กลุ่มคลาสใน DataSet

1. Connection Class ทำหน้าที่ติดต่อฐานข้อมูล SQL Server
2. Command Class จะใช้ร่วมกับ Parameter Object ในการประมวลผลคำสั่ง SQL, Query หรือ Stored Procedure ในแบบที่ส่งผลลัพธ์คืนกลับมาหรือไม่กลับมาก็ได้
3. Data Reader Class จะใช้สำหรับการดึงข้อมูลแบบทีละเรคอร์ดจากฐานข้อมูลบนเซิร์ฟเวอร์ โดย เซิร์ฟเวอร์ Cursor จะเป็นแบบ Forward-Only คือเดินหน้าอย่างเดียวถอยหลังไม่ได้และไม่สามารถแก้ไขข้อมูลได้ด้วย ทำให้ความเร็วในการเข้าถึงข้อมูลสูง
4. Data Adapter Class ประกอบด้วยออบเจ็ค 4 ตัว คือ SelectCommand, InsertCommand, UpdateCommand และ DeleteCommand โดยออบเจ็ค SelectCommand จะทำหน้าที่เลือกข้อมูลจากฐานข้อมูลมาสร้างชุดข้อมูลมาเก็บไว้ในหน่วยความจำของเครื่อง Client หลังจากนั้นจะตัดการเชื่อมต่อกับฐานข้อมูลจริงบนเครื่องเซิร์ฟเวอร์
5. DataSet Class ใช้แทนชุดข้อมูลในหน่วยความจำ การกระทำใดๆ หลังตัดการเชื่อมต่อกับฐานข้อมูลเช่น Update, Delete หรือ Insert จะเป็นการกระทำกับข้อมูลในฐานข้อมูล เท่านั้น ไม่ได้กระทำกับฐานข้อมูลจริง หลังจากปรับปรุงแก้ไขใน DataSet แล้ว จะใช้ออบเจ็ค InsertCommand, UpdateCommand และ DeleteCommand บันทึกข้อมูลลงในฐานข้อมูลอีกครั้งหนึ่ง

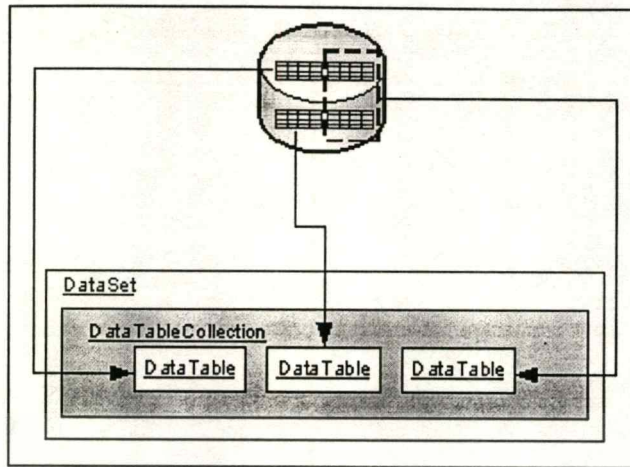


รูปที่ 2.7 ระดับการทำงานของ Connected Stage กับ Disconnected Stage

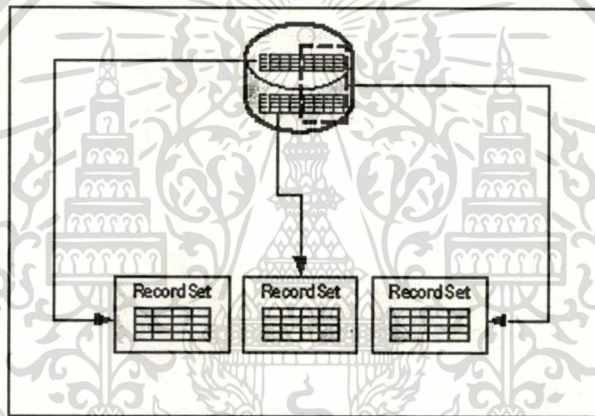
### 2.3.1.2 เปรียบเทียบ DataSet กับ RecordSet ใน ADO

สำหรับ DataSet Class นี้ถูกนำมาใช้แทนออบเจ็กต์ RecordSet ใน ADO โดยมีคุณสมบัติที่แตกต่างจาก RecordSet ดังนี้

- ใน 1 DataSet จะประกอบด้วย DataTable ตั้งแต่ 1 ตัวขึ้นไป โดยกลุ่ม DataTable ทั้งหมดจะเก็บอยู่ใน DataTableCollection จึงสามารถแสดงผลลัพธ์ ได้มากกว่า 1 ผลลัพธ์ใน DataSet เดียวกัน ถ้าเปรียบเทียบกับ ADO ออบเจ็กต์ Record 1 ตัว จะแทนข้อมูลจาก 1 ตาราง หรือ 1 Query เท่านั้น ถ้าต้องการนำข้อมูลจากหลายๆ ตารางมาสร้าง Record จะต้องนำตารางเหล่านั้นมาจับคู่เพื่อสร้างข้อมูล RecordSet เท่ากับจำนวนตารางนั้น จากที่กล่าวมาแล้ว จึงสามารถสร้างความสัมพันธ์ระหว่าง DataTable เหล่านี้ได้ เช่นเดียวกับ Data Relation Class



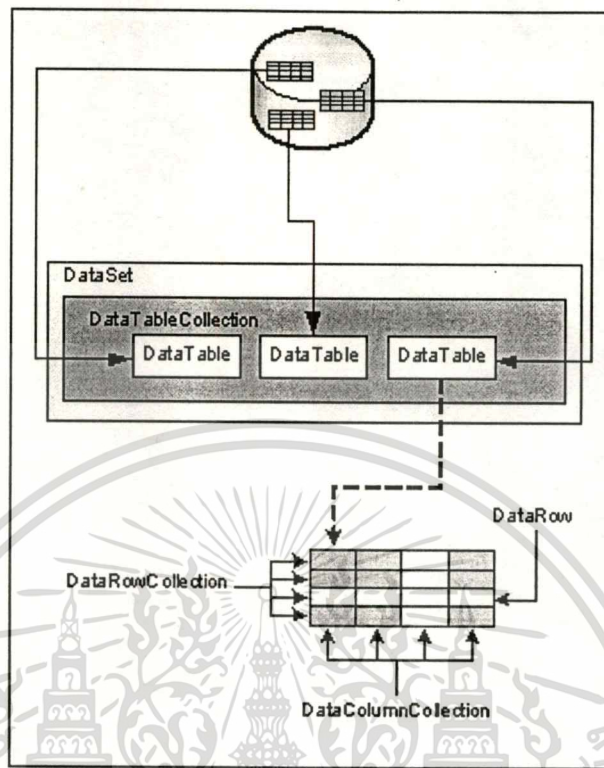
รูปที่ 2.8 โครงสร้าง DataSet



รูปที่ 2.9 โครงสร้าง RecordSet

- DataTable แต่ละตัวใน DataSet จะประกอบด้วยกลุ่มของ DataRow และกลุ่มของ DataColumn โดย 1 DataRow จะหมายถึงข้อมูล 1 Record ส่วน 1 DataColumn จะหมายถึง 1 필ด์ของ DataTable นั้น นอกจากนี้ในแต่ละ DataTable ยังเก็บข้อมูลโครงสร้างเช่น ColumnName, DataType, AutoIncrement, AllowDBNull เป็นต้น รวมทั้งส่วนของ ConstraintCollection ที่เก็บกฎเกณฑ์และเงื่อนไขต่างๆ ที่ใช้ควบคุมการคงสภาพความถูกต้องของข้อมูล

- การทำงานร่วมกับ DataSet จะเป็นอิสระจากการเชื่อมต่อกับฐานข้อมูลบนเครื่องเซิร์ฟเวอร์ โดยจะทำหน้าที่เป็น Memory-Resident Database คือเก็บข้อมูลจากรางไว้ในหน่วยความจำของเครื่อง Client โดยจำลองโครงสร้างและรูปแบบการเก็บข้อมูลจริงในฐานข้อมูลบนฐานข้อมูลเซิร์ฟเวอร์



รูปที่ 2.10 DataTable ใน DataSet

- เนื่องจากข้อมูลใน DataSet จะถูกเก็บบนหน่วยความจำของเครื่อง Client จึงไม่มีการนำเซิร์ฟเวอร์เคอร์เซอร์ มาใช้เหมือนกับการใช้ออบเจ็กต์ RecordSet ใน ADO

- การปรับปรุงแก้ไขข้อมูลอะไรก็ตามทำได้เฉพาะในหน่วยความจำ ซึ่งไม่มีผลต่อฐานข้อมูลในฐานข้อมูลจริง ถ้าต้องการบันทึกข้อมูลที่แก้ไขหรือเพิ่มใหม่ลงฐานข้อมูลจริงจะต้องทำผ่าน Method Update() ของ DataAdapter เพื่อประมวลผลคำสั่ง InsertCommand, UpdateCommand และ DeleteCommand

- สำหรับการพัฒนาในครั้งนี้ ผู้พัฒนาได้เลือกใช้ DataReader Class และ DataCommand Class แทนการใช้ DataSet Class ในการติดต่อฐานข้อมูล เนื่องจากการใช้ DataSet Class นั้นจะใช้หน่วยความจำของเครื่องเซิร์ฟเวอร์ในการเก็บข้อมูล ซึ่งการใช้งานใน พอกเก็ตพีซี ไม่ได้มี Transaction สูงจึงไม่มีความจำเป็นจะใช้ DataSet Class

### 2.3.2 การใช้ พอกเก็ตพีซี ติดต่อฐานข้อมูล

สำหรับการใช้ พอกเก็ตพีซี ติดต่อฐานข้อมูลนั้น ผู้พัฒนาได้ใช้เทคนิคของ เว็บเซอร์วิส เป็นตัวเรียกใช้งานในการดึงข้อมูลมาจากฐานข้อมูลบนเซิร์ฟเวอร์ และส่งผลของการดึงข้อมูลมาเป็นไฟล์ XML แล้วจึงเรียกอ่านข้อมูลจากไฟล์ XML อีกทีหนึ่ง เนื่องจากเทคโนโลยีในการพัฒนาแอปพลิเคชัน บน พอกเก็ตพีซี นั้นไม่สนับสนุน ADO.NET จะมีก็แต่ ADOCE.NET ซึ่งจะเป็น ADO ที่ใช้ติดต่อกับฐานข้อมูล (SQLCE) สำหรับ พอกเก็ตพีซี โดยเฉพาะ ซึ่งการทำงานไม่เกี่ยวกับฐานข้อมูลบนเซิร์ฟเวอร์ แต่อย่างใด

## 2.4 การใช้งาน เว็บเซอร์วิส

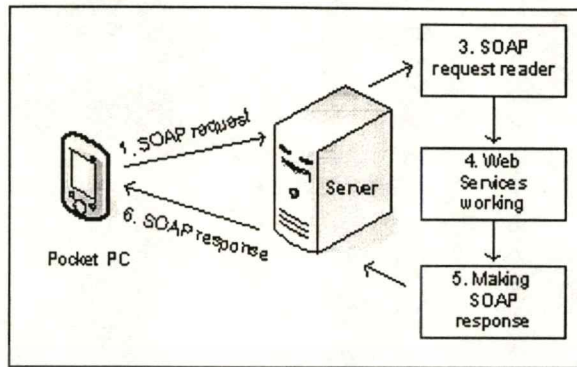
### 2.4.1 ASP.NET ในการสร้าง เว็บเซอร์วิส

เว็บเซอร์วิส เป็นเทคโนโลยีที่สามารถเรียกใช้ฟังก์ชันจากเซิร์ฟเวอร์ ได้เหมือนมีการเรียกใช้งานในเครื่องนั้นปกติ เว็บเซอร์วิส อาศัย HTTP, SOAP เป็นโปรโตคอลมาตรฐาน และ XML ซึ่งเป็นภาษามาตรฐานเพื่อช่วยในการเรียกใช้ฟังก์ชัน (Function Call) และรับข้อมูลที่ส่งกลับมา ผ่านเครือข่ายได้

SOAP (Simple Object Access Protocol) เป็นโปรโตคอลที่ทำให้เรียกใช้ฟังก์ชันที่อยู่ คนละเครื่องได้ (Remote Procedure Call : RPC) โดยอาศัย HTTP และ XML เรียกผ่านเครือข่าย SOAP จะนำข้อมูลที่ต้องการส่งมาเปลี่ยนเป็นเอกสาร XML แล้วส่งไปในเครือข่ายด้วยโปรโตคอล HTTP

ข้อมูลที่ถูกส่งด้วย SOAP จะแบ่งเป็น 2 ส่วนคือ ข้อมูลที่เรียกใช้งานฟังก์ชันและข้อมูลซึ่งเป็นค่าส่งกลับจากฟังก์ชัน โดยเรียกว่า "SOAP request" และ "SOAP response"

SOAP request จะเก็บข้อมูลที่มีการเรียกใช้ฟังก์ชันและเก็บพารามิเตอร์ที่ส่งไปให้ฟังก์ชันบนเซิร์ฟเวอร์ โดยใช้แท็กเป็นตัวกำหนดชื่อฟังก์ชันและชื่อพารามิเตอร์ เมื่อเซิร์ฟเวอร์ ให้บริการเว็บเซอร์วิส ได้รับ SOAP request ก็จะไปเรียกฟังก์ชันที่ระบุมาใน SOAP request ให้ทำงานแล้วนำค่าที่ได้ส่งกลับไปให้ผู้ใช้ด้วย SOAP response ซึ่งใช้แท็กบอกว่าส่วนใดคือผลลัพธ์จากฟังก์ชันเช่นกัน



รูปที่ 2.11 การทำงานของโปรโตคอล SOAP

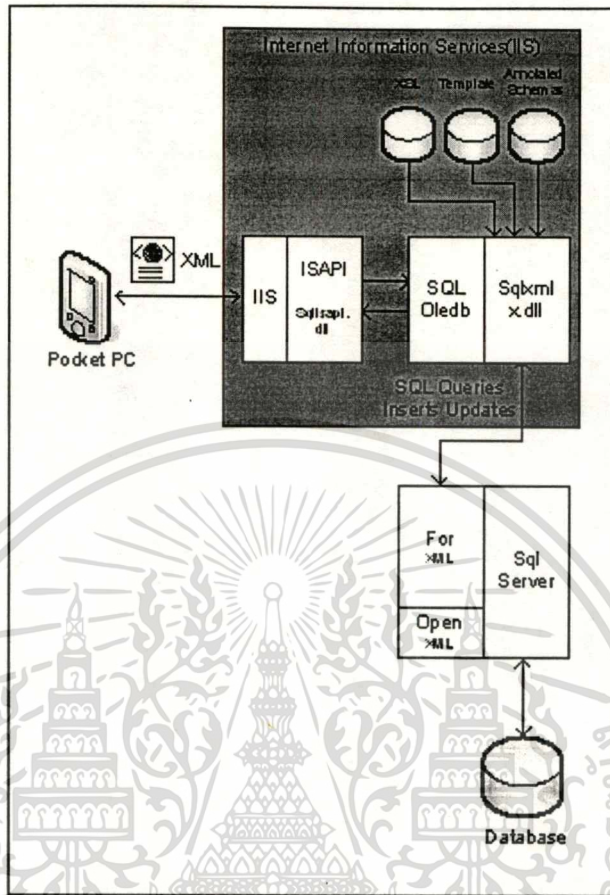
#### 2.4.2 การเข้าถึงข้อมูลของ เว็บเซอร์วิส

การเข้าถึงข้อมูลของเซิร์ฟเวอร์ โดยการอาศัย เว็บเซอร์วิส ที่สนับสนุนไฟล์ XML นั้น ทางฝั่งเซิร์ฟเวอร์จะแบ่งการทำงานออกเป็น 2 ส่วนคือ ส่วนที่ทำหน้าที่ตอบรับการทำงานของ SOAP request คือ เว็บเซิร์ฟเวอร์อย่าง IIS (Internet Information Services) และส่วนที่ทำหน้าที่เป็นฐานข้อมูลคือ เซิร์ฟเวอร์ฐานข้อมูล ทั้งสองส่วนนี้อาจทำงานอยู่บนเครื่องเดียวกันหรือไม่ก็ได้

ที่ IIS จะต้องมีการสร้าง Virtual Directory ให้สนับสนุนการทำงานร่วมกับ XML โดยใช้วิธีการของ SOAP และเมื่อระบุ URL มายังเซิร์ฟเวอร์ IIS จะตรวจสอบไปยัง Virtual Directory เพื่อโหลด ISPAI DLL Extension ที่ชื่อ sqlisapi.dll ขึ้นมาทำงาน พร้อมทั้งส่งผ่าน URL ต่อไปยัง DLL นี้

สำหรับ sqlisapi.dll ทำหน้าที่ติดต่อกับ เซิร์ฟเวอร์ฐานข้อมูล โดยผ่าน OLEDB ซึ่งต้องกำหนดให้คำสั่งที่เกี่ยวข้องกับ XML ปกติ รวมถึงการใช้ Template, XML Data Reduced(XDR) Schema, Extensible Stylesheet Language(XSL), XPath Queries และ XDR Schemas ถูกส่งผ่านการทำงานไปยัง sqlxml.dll เพื่อทำการเอ็ชคิวรี่ และส่งผลลัพธ์ที่ได้กลับมายัง SQLOLEDB จากนั้นจึงส่งข้อมูลนี้กลับไปแสดงผลในรูปแบบของ XML/HTML โดยผ่าน Server ในที่สุด

การทำงานที่เกิดขึ้นในส่วนของเซิร์ฟเวอร์ ทั้งหมด ระบบจะจัดการให้เองโดยอัตโนมัติ เพียงแต่กำหนด Virtual Directory ที่ต้องการให้ เว็บเซิร์ฟเวอร์ สนับสนุนการใช้งาน XML เพื่อเข้าถึงข้อมูลที่ต้องการได้



รูปที่ 2.12 การเข้าถึงข้อมูลของ เว็บเซอร์วิส

#### 2.4.3 ปัญหาของ Mobile แอปพลิเคชัน ในการต่อฐานข้อมูลและแนวทางการแก้ไข

เนื่องจากระบบที่พัฒนาจะเป็นการพัฒนาบน โมบายแอปพลิเคชัน โดยคำแนะนำจะอยู่ในรูปแบบของ อิมูเลเตอร์ (พอกเก็ตพีซี 2003 อิมูเลเตอร์) ซึ่งมีข้อจำกัดในการติดต่อกับฐานข้อมูล โดย เว็บแอปพลิเคชัน และ แอปพลิเคชัน จะใช้การติดต่อกับฐานข้อมูลโดยตรงโดยผ่าน Engine ต่างๆที่มีมา Support แต่หากเป็น โมบายแอปพลิเคชัน การติดต่อกับฐานข้อมูลจะต้องติดต่อโดยผ่าน เว็บเซอร์วิส ซึ่งจะทำหน้าที่เป็น Interface ในการติดต่อเรียกใช้ฐานข้อมูลอีกชั้นหนึ่ง

## บทที่ 3

# การวิเคราะห์ระบบและออกแบบระบบใหม่

### 3.1 การวิเคราะห์ระบบ

#### 3.1.1 ปัญหาของระบบดูแลรักษาสัตว์

ระบบดูแลรักษาสัตว์โดยทั่วไปมีดังนี้

1. เมื่อเจ้าของนำสัตว์เลี้ยงเข้ามาในโรงพยาบาล พนักงานต้อนรับจะสอบถามอาการเบื้องต้น และสอบถามว่าเจ้าของเคยนำสัตว์เลี้ยงเข้ามาใช้บริการหรือไม่ แบ่งได้เป็น 2 กรณี

1.1 กรณีลูกค้าเก่าเคยใช้บริการมาพบสัตวแพทย์พนักงานนำบัตรประจำตัว เพื่อนำมาค้นหาบัตรตรวจรักษา

1.2 กรณีเจ้าของเคยมาครั้งแรก พนักงานจะให้เจ้าของกรอกประวัติ พร้อมทั้งพนักงานออกบัตรประจำตัวใหม่ให้กับเจ้าของสัตว์เลี้ยง

2. พนักงานนำบัตรตรวจมาเรียงลำดับตามการเข้ารับการรักษา

3. พนักงานเรียกเจ้าของและสัตว์เลี้ยงเพื่อเข้าพบสัตวแพทย์ตามลำดับการเข้ารับการรักษา

4. เมื่อตรวจรักษาเสร็จ แพทย์จะลงบันทึกประวัติการรักษาของสัตว์เลี้ยง โดยแพทย์จะวินิจฉัยโดยแบ่งเป็น 2 กรณี

4.1 กรณีที่สัตว์เลี้ยงมีความจำเป็นที่จะต้องพักรักษาอยู่ที่โรงพยาบาล

4.1.1 พนักงานจำเป็นจะต้องบันทึกข้อมูลที่เกี่ยวข้องกับการรักษาสัตว์

4.1.2 สัตวแพทย์เข้ามาตรวจเช็คเป็นระยะๆเพื่อรักษาและเปลี่ยนแปลงข้อมูลของสัตว์

4.2 กรณีที่สัตว์เลี้ยงไม่มีความจำเป็นที่จะต้องพักรักษาอยู่ที่โรงพยาบาล

5. เมื่อทำการรักษาเสร็จสิ้นเจ้าหน้าที่จะติดต่อเจ้าของสัตว์เลี้ยงเพื่อมารับสัตว์กลับบ้าน

6. พนักงานเรียกเจ้าของสัตว์เลี้ยงเพื่อชำระเงิน

จากขั้นตอนที่กล่าวมาข้างต้น ขั้นตอนที่มีปัญหาในการระบุตัวตนจะเป็นในขั้นตอน 4.1 (กรณีที่สัตว์เลี้ยงมีความจำเป็นที่จะต้องพักรักษาอยู่ที่โรงพยาบาล) เพราะเป็นการยากที่จะทำการระบุอัตลักษณ์ของสัตว์ โดยเมื่อจะทำการระบุ สัตวแพทย์จะต้องขอข้อมูลจากพนักงานเพื่อมาตรวจสอบข้อมูลตามหมายเลขกรง ซึ่งหากสัตว์ถูกขังอยู่ในกรงรวมจะทำให้การแยกแยะสัตว์มีความยากลำบากมากยิ่งขึ้น

### 3.1.2 แนวทางแก้ไขโดยนำระบบ RFID มาประยุกต์ใช้

ปัญหาที่เกิดจากการระบุอัตลักษณ์ของสัตว์ทำให้เกิดความยุ่งยากในกระบวนการสืบค้น ข้อมูล เอกสารฉบับนี้จึงนำเทคโนโลยี RFID (การระบุอัตลักษณ์โดยใช้คลื่นวิทยุ) โดยได้นำประโยชน์จากเทคโนโลยี RFID ที่ใช้คุณสมบัติ Contactless (การอ่านข้อมูลแบบไร้สัมผัส) และคุณสมบัติของตัว แท็ก ซึ่งมีความสามารถในการเก็บข้อมูลได้ในระดับหนึ่งทำให้ไม่จำเป็นจะต้องพึ่งพาฐานข้อมูลเพียงอย่างเดียว

#### การทำงานของระบบการบริการในโรงพยาบาลสัตว์โดยนำ RFID มาประยุกต์ใช้

1. เมื่อเจ้าของนำสัตว์เลี้ยงเข้ามาในโรงพยาบาล พนักงานต้อนรับจะสอบถามอาการเบื้องต้นและสอบถามว่าเจ้าของเคยนำสัตว์เลี้ยงเข้ามาใช้บริการหรือไม่ แบ่งได้เป็น 2 กรณี
  - 1.1 กรณีลูกค้าเก่าเคยใช้บริการมาพบสัตวแพทย์พนักงานนำบัตรประจำตัว เพื่อนำมาคืนหาบัตร ตรวจรักษา
  - 1.2 กรณีเจ้าของเคยมาครั้งแรก พนักงานจะให้เจ้าของกรอกประวัติ พร้อมทั้งพนักงานออกบัตรประจำตัวใหม่ให้กับเจ้าของสัตว์เลี้ยง
2. พนักงานนำบัตรตรวจมาเรียงลำดับตามการเข้ารับการรักษา
3. พนักงานเรียกเจ้าของและสัตว์เลี้ยงเพื่อเข้าพบสัตวแพทย์ตามลำดับการเข้ารับการรักษา
4. เมื่อตรวจรักษาเสร็จ แพทย์จะลงบันทึกประวัติการรักษาของสัตว์เลี้ยง โดยแพทย์จะวินิจฉัยแบ่งเป็น 2 กรณี
  - 4.1 กรณีที่สัตว์เลี้ยงมีความจำเป็นที่จะต้องพักรักษาอยู่ที่โรงพยาบาล เจ้าหน้าที่จะทำการจัดการข้อมูลใส่ใน RFID แท็ก เพื่อทำการระบุลักษณะของสัตว์เลี้ยงและนำสัตว์เข้าสู่ขั้นตอนการพักรักษา
  - 4.2 กรณีที่สัตว์เลี้ยง ไม่มีความจำเป็นที่จะต้องพักรักษาอยู่ที่โรงพยาบาล
5. เจ้าหน้าที่จะทำการจัดหาให้เจ้าของสัตว์เลี้ยง
6. พนักงานเรียกเจ้าของสัตว์เลี้ยงเพื่อชำระเงิน

ดังรายละเอียดลำดับของระบบที่กล่าวมาในข้างต้นทำให้สามารถแก้ปัญหากระบวนการสืบค้นข้อมูลสัตว์ได้อย่างมีประสิทธิภาพ อีกทั้งยังมีข้อมูลสำรองอยู่ในตัว แท็ก อีกด้วย และด้วยแนวทางการแก้ไขโดยนำเทคโนโลยี RFID ผสมผสานกับ พอกเก็ตพีซี เพื่อความสะดวกสบายในการตรวจดูรักษาของแพทย์ จะทำให้ความคล่องตัวในการสืบค้นเป็นไปได้อย่างมีประสิทธิภาพมากที่สุด

### 3.1.3 แนวทางแก้ไขโดยนำระบบ พกเก็ตพีซี มาประยุกต์ใช้

ปัญหาที่ของการดูแลรักษาตัวหลักๆคือ การระบุตัวตน ซึ่งปัญหาดังกล่าวได้ใช้เทคโนโลยี RFID ในการแก้ไข แต่หากเป็นการแก้ปัญหาโดยใช้ พีซี แอปพลิเคชัน ปกติ จะทำให้การแก้ปัญหาไม่ราบรื่น เนื่องจากหากเป็นการใช้งานจริงเมื่อจะทำการตรวจสอบหรือจะทำการระบุตัวตน จะต้องนำตัวเล็ยงนั้นๆมาอยู่ ณ เครื่อง พีซี (Personal Computer) ซึ่งจะดูไม่สะดวกนักจะได้คิดแนวทางในการแก้ปัญหาโดยการพัฒนาระบบนั้นจะต้องพัฒนาบนโมบาย แอปพลิเคชันหรือ บน พกเก็ตพีซี นั่นเอง เพื่อการประ โยชน์ใช้สอยของผู้ใช้งานได้อย่างเต็มประสิทธิภาพ



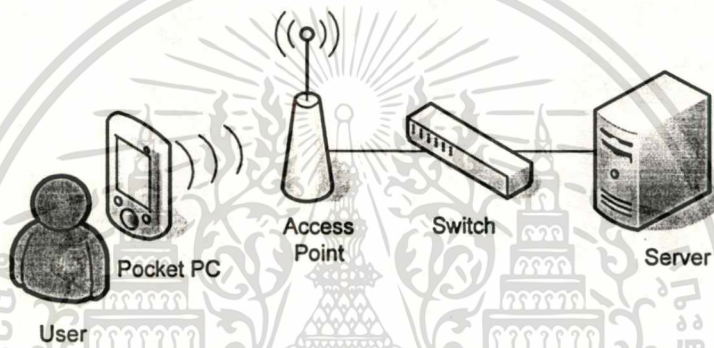
รูปที่ 3.1 รูปจำลองระหว่างเครื่อง พกเก็ตพีซี ที่มี RFID(รูปค้ำ้นซ้าย) และ แท็ก ที่ติดไว้บนตัวสัตว์ (รูปค้ำ้นขวา)

ในอุตสาหกรรมอดีตจนถึงปัจจุบันได้มีการใช้เทคโนโลยีเข้าเพื่อเพิ่มประสิทธิภาพ ทั้งในด้านความเร็วและประสิทธิภาพในเชิงคุณภาพ โดยการระบุตัวตนของวัตถุเป็นปัญหาสำคัญอย่างหนึ่งที่สำคัญ ซึ่งส่งผลกระทบต่อเวลาที่ล่าช้าในการผลิตอีกทั้งความคลาดเคลื่อนและไม่แน่นอนที่อาจเกิดขึ้นได้สูง อุตสาหกรรมหลายๆ ค้ำ้นจึงได้นำเทคโนโลยีบาร์โค้ดเข้ามาเพื่อแก้ปัญหการระบุตัวตนของวัตถุ เพื่อให้เข้าถึงข้อมูลในตัววัตถุได้ง่ายขึ้น โดยใช้หลักการนำตัวเลขมาเพื่อไว้อ้างอิงกับฐานข้อมูล แต่รูปแบบของตัวเลขนั้นได้เปลี่ยนมาใช้แท่งรหัสซึ่งเรียกว่า บาร์โค้ดนั่นเอง บาร์โค้ดเข้ามามีบทบาทต่ออุตสาหกรรมหลายๆด้านอยู่ยาวนาน และทำให้การระบุตัวตนผ่านเครื่องอ่านบาร์โค้ดมีประสิทธิภาพมากกว่าระบบเก่ามากขึ้น โดยเฉพาะการจะใช้บาร์โค้ดส่วนมากจำเป็นที่จะต้องมิฐานข้อมูลที่ใหญ่เพื่อรองรับบาร์โค้ดจำนวนมาก ซึ่งปัญหาจากการใช้บาร์โค้ดก็เกิดขึ้น เนื่องจากเมื่อจะทำการระบุตัวตนของวัตถุแต่ละชิ้นจะต้องการแปลงรหัสแท่งเป็นตัวเลขแล้วทำการอ้างอิงจากฐานข้อมูลซึ่งเท่ากับว่าในตัวบาร์โค้ดเองแทบจะไม่มีข้อมูลใดๆอยู่เลยนอกจากตัวเลขที่ไว้สำหรับอ้างอิง แต่ข้อมูลที่แท้จริงอยู่ในส่วนของฐานข้อมูล ซึ่งหากจะทำการเปลี่ยนข้อมูลใดๆจะต้องทำการอ้างอิงกับฐานข้อมูลตลอด ซึ่งหากเป็นวัตถุหรือสิ่งมีชีวิตที่จะต้องทำการเปลี่ยนแปลงข้อมูลบ่อยครั้ง การใช้แท่งรหัสหรือบาร์โค้ด จึงเป็นการเสียเวลาในกระบวนการมาก และปัญหาอีกด้านหนึ่งของบาร์โค้ดคือ เมื่อจะทำการอ่านบาร์โค้ดจะต้องทำการอ่านจากบาร์โค้ดโดยตรง โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ห้ามมีสิ่งกีดขวางกอนหน้าบาร์โค้ด ดังนั้นบาร์โค้ดถูกพิมพ์ไว้อยู่ข้างหน้าวัตถุตลอดซึ่งมีโอกาสสูงที่จะถูกทำให้เสียหายได้ และการที่พิมพ์บาร์โค้ดไว้อยู่ข้างหน้าวัตถุจึงมีโอกาสเสี่ยงสูงหากมีการคัดลอกเพื่อนำไปศึกษาเพื่อไว้ใช้ประโยชน์ด้านอื่นๆ ได้เช่น การขโมย การเปลี่ยนแปลงบาร์โค้ด การลอกรหัสบาร์โค้ด เป็นต้น ต่อมาจึงได้มีการนำเทคโนโลยี อาร์เอฟไอดี (RFID) หรือ เรียกว่า Radio Frequency Identification ซึ่งการระบุตัวตนโดยใช้คลื่นวิทยุ ซึ่งสามารถจัดปัญหาการอ้างอิงข้อมูลจากฐานข้อมูลเพียงอย่างเดียว มาเป็นการอ้างอิงข้อมูลในระดับหนึ่งจากตัวรับสัญญาณ ซึ่งทำให้ความเร็วในการเปลี่ยนแปลงข้อมูลสูงขึ้นมากทีเดียว อีกทั้ง อาร์เอฟไอดี แท็ก ไม่จำเป็นจะต้องวางอยู่ข้างหน้าวัตถุ แต่สามารถวางอยู่ที่ใดก็ได้ในวัตถุนั้นๆ



รูปที่ 3.2 การสื่อสารอุปกรณ์ต่างๆ ภายในที่ทำการรักษาพยาบาลสัตว์

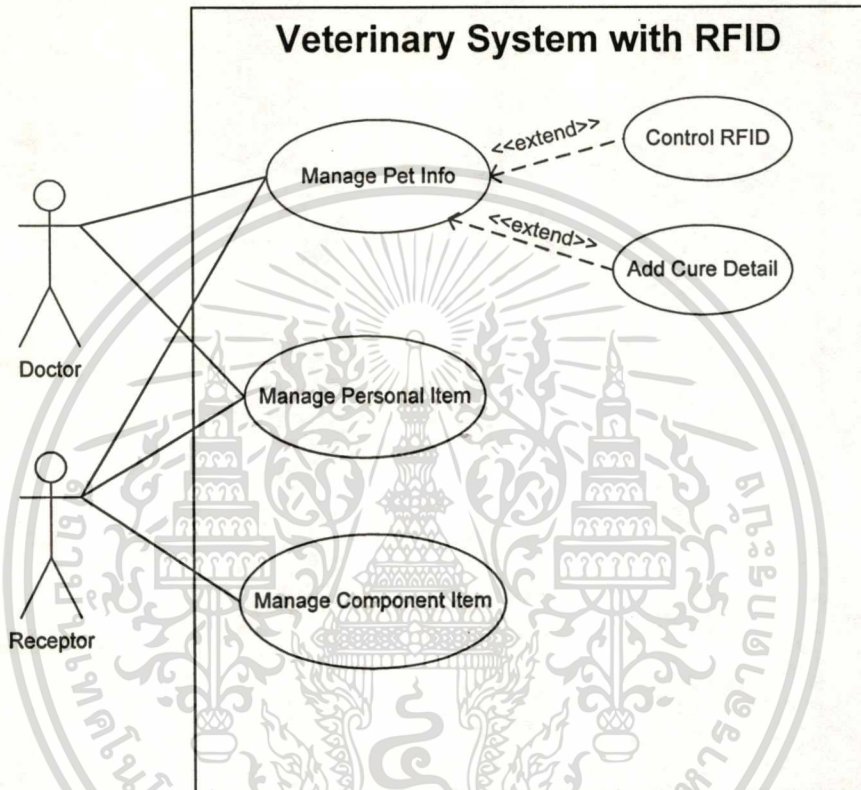
### 3.2 การออกแบบระบบ

จากความเป็นมาและปัญหาที่กล่าวไว้ในบทที่ 1 นั้น ผู้เขียนได้วิเคราะห์และออกแบบระบบจากระบบเดิมให้เป็นระบบใหม่ โดยการวิเคราะห์และออกแบบระบบระบบรักษาสัตว์นี้ใช้ UML (Unified Modeling Language) ในการวิเคราะห์และออกแบบการออกแบบระบบ ซึ่งจะอธิบายขั้นตอนการทำงานต่างๆ ซึ่งช่วยให้สามารถเข้าใจกระบวนการทำงานของแต่ละหน้าที่ ซึ่งทราบถึงการรับรับส่งข้อมูล การประสานงานระหว่างกิจกรรมต่าง ๆ ในการดำเนินงาน ซึ่งเป็นแบบจำลองของระบบ

โดยเริ่มจาก Use Case Diagram, Class Diagram, Sequence Diagram และออกแบบหน้าจอตามลำดับ

### 3.2.1 Use Case Diagram

อธิบายถึงขอบเขตโดยรวมของระบบ ซึ่งจะอธิบายว่ามีเมฆอดต่างๆภายในระบบอย่างไรบ้าง เช่น การจัดการข้อมูลสัตว์เลี้ยง การจัดการองค์ประกอบของรายการ การจัดการองค์ประกอบของรายการที่ได้รับสิทธิแล้ว เป็นต้น



รูปที่ 3.3 แสดงภาพ Sequence Diagram การทำงานรวมของระบบ

#### Use Case Specification

อธิบายรายละเอียดของแต่ละ Use case ว่ามีหน้าที่การทำงาน และ ความสัมพันธ์กับ Use case อื่นอย่างไรบ้าง

#### Use case 1: Manage Pet Info (Manage Pet Information)

**Description:** การจัดการรายละเอียดในการรักษา โรค การให้ยา การให้อาหาร

**Active Actor:** Doctor, Receptor

**Precondition:** -

**Basic Flow:**

1. หลังจากผ่านเข้ามาในระบบ หากเป็น Doctor จะสามารถเลือกการทำงานได้ 3

เอกสารนี้ ส่วน หากเป็น Receptor สามารถเลือกการทำงานได้ 2 ส่วน ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ผู้ใช้เลือกการทำงานในส่วน Manage Pet Info

3. ผู้ใช้สามารถเลือกการทำงานได้ 2 ส่วนย่อย คือ

3.1 ส่วน Enroll System จะเป็นการดูข้อมูลสัตว์เลี้ยงโดยการเลือกลูกค้ำ แล้วจึงเลือก สัตว์เลี้ยงของลูกค้ำ

3.2 ส่วน Identification จะเป็นการดูข้อมูลสัตว์เลี้ยงโดยการใช้ระบบ RFID ซึ่งจะอยู่ในส่วนของ Use Case 4 (Control RFID)

4. Doctor เพิ่มรายการการรักษาของสัตว์เลี้ยง โดยการเพิ่มข้อมูลโรค หรือ ข้อมูลการให้อาหาร หรือ ทั้งข้อมูลโรค และ ข้อมูลการให้อาหาร

**Alternate Flow:-**

**Postcondition:** ระบบเก็บข้อมูลที่ Update ลงใน Server หรือ ในกรณีที่ใช้ส่วน Identification จะต้องถูกส่งไปที่ Use Case 4 (Control RFID)

**Use case 2: Manage Personal Item**

**Description:** การจัดการองค์ประกอบ ประวัติส่วนตัวผู้ใช้, ประวัติการดูแลรักษาสัตว์เลี้ยงของผู้ใช้

**Active Actor:** Doctor, Receptor

**Precondition: -**

**Basic Flow:**

1. หลังจากผ่านเข้ามาในระบบ หากเป็น Doctor จะสามารถเลือกการทำงานได้ 3 ส่วน หากเป็น Receptor สามารถเลือกการทำงานได้ 2 ส่วน

2. ผู้ใช้เลือกการทำงานในส่วน Manage Pet Info

3. ผู้ใช้สามารถเลือกการทำงานได้ 2 ส่วนย่อย คือ

3.1 ส่วน Edit Profile จะเป็นการเข้าไปเปลี่ยนแปลงประวัติของผู้ใช้ ตลอดจนการเปลี่ยนแปลง Password

3.2 ส่วน Personal Case History จะเป็นการดูข้อมูลประวัติสัตว์เลี้ยงที่ Doctor ได้ทำการดูแล หรือ รักษา

**Alternate Flow:-**

**Postcondition:** ระบบเก็บข้อมูลที่ Update ลงใน Server

**Use case 3: Manage Component Item**

**Description:** ส่วนที่ Authorized User มีสิทธิในการจัดการองค์ประกอบในการจัดการ ชนิดโรค ชนิดสัตว์เลี้ยง ชนิดของยา ชนิดของอาหาร ข้อมูลทั่วไปต่างๆ

**Active Actor:** Authorized User

**Precondition:-**

**Basic Flow:**

1. หลังจากผ่านเข้ามาในระบบ
2. Receptor เลือกการทำงานในส่วน Manage Component Item
3. Receptor สามารถเลือกการทำงานได้ 2 ส่วนย่อย คือ
  - 3.1 ส่วน Disease จะเป็นการเข้าไปเปลี่ยนแปลง เพิ่มเติม ข้อมูลโรคของสัตว์เลี้ยง
  - 3.2 ส่วน Nutrient จะเป็นการเข้าไปเปลี่ยนแปลง เพิ่มเติม ข้อมูลการให้อาหารของสัตว์เลี้ยง
  - 3.3 ส่วน Pet Type จะเป็นการเข้าไปเปลี่ยนแปลง เพิ่มเติม ประเภท หรือ พันธุ์ของสัตว์เลี้ยง

**Alternate Flow:-**

**Postcondition:** ระบบเก็บข้อมูลที่ Update ลงใน Server

**Use case 4: Control RFID**

**Description:** ส่วนของการควบคุมเครื่องอ่าน RFID แท็ก (RFID Device) และจัดเรียงข้อมูล แท็ก ให้ถูกต้อง

**Active Actor:** Doctor, Receptor

**Precondition:** Manage Pet Info (Manage Pet Information)

**Basic Flow:**

1. หลังจากผ่านเข้ามาในระบบ ผู้ใช้สามารถเลือกการทำงานได้ 3 ส่วน
2. ผู้ใช้เลือกการทำงานในส่วน Manage Pet Info
3. ผู้ใช้เลือกการทำงานในส่วนของ Identification
4. เครื่องอ่าน RFID ทำการอ่านข้อมูลจาก แท็ก ของสัตว์เลี้ยง
5. แสดงข้อมูลสัตว์เลี้ยงผ่าน แท็ก ซึ่งมีทั้ง แท็ก Information และ General Information

**Alternate Flow:-**

**Postcondition:** Manage Pet Info, ระบบเก็บข้อมูลที่ Update ลงใน Server

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Use case 5: Add Cure Detail**

**Description:** ส่วนที่ให้ Doctor สามารถเพิ่มรายการการรักษาของสัตว์เลี้ยง โดยการเพิ่มข้อมูลโรค หรือ ข้อมูลการให้อาหาร หรือทั้งสองอย่าง

**Active Actor:** Doctor

**Precondition:** Manage Pet Info (Manage Pet Information).

**Basic Flow:**

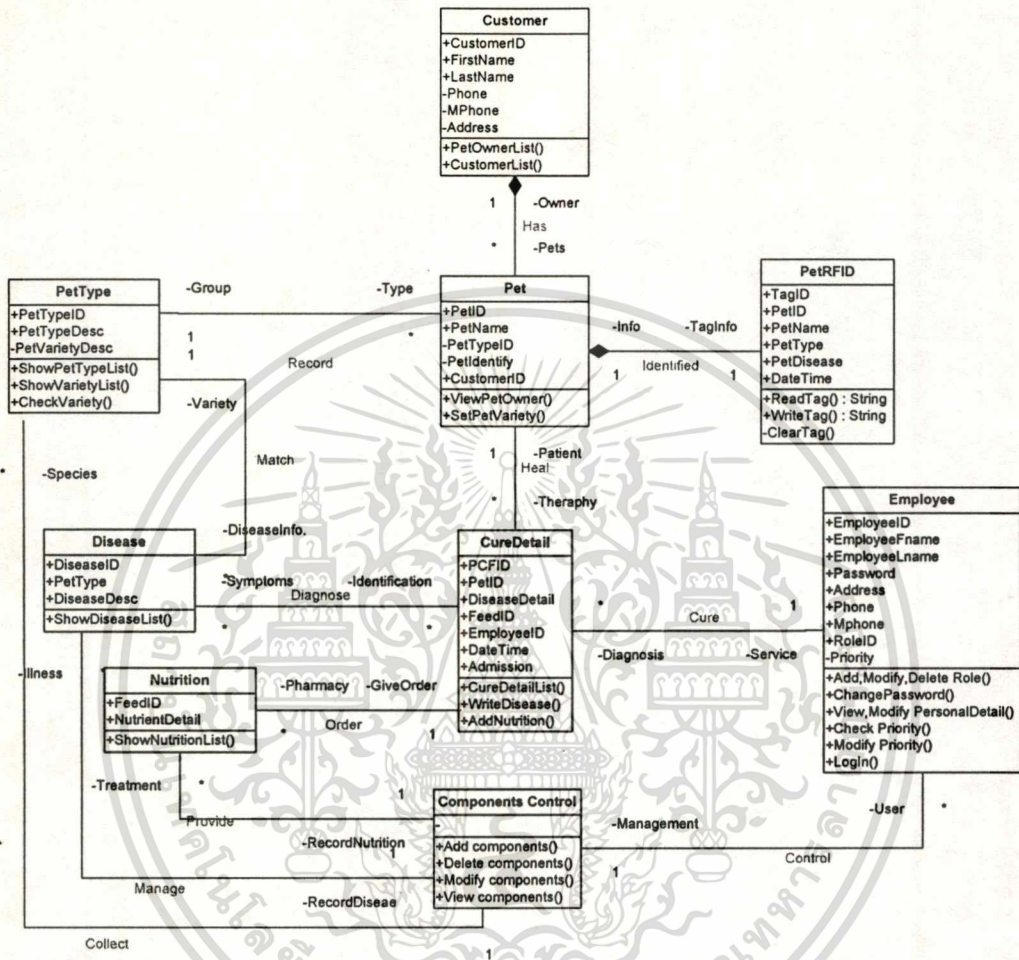
1. หลังจากผ่านเข้ามาในระบบ Doctor สามารถเลือกการทำงานได้ 3 ส่วน
2. Doctor เลือกการทำงานในส่วน Manage Pet Info
3. Doctor เลือกการทำงานในส่วนของ Identification หรือ Enroll System
4. Doctor เพิ่มรายการการรักษาของสัตว์เลี้ยง
5. Doctor ทำการเพิ่มข้อมูลโรค หรือ ข้อมูลการให้อาหาร หรือ ทั้งข้อมูลโรค และ ข้อมูลการให้อาหาร

**Alternate Flow:-**

**Postcondition:** ระบบเก็บข้อมูลที่เพิ่มเติมลงใน Server

### 3.2.2 Class Diagram

อธิบายถึงความสัมพันธ์ของคลาสต่างๆ ของระบบ



รูปที่ 3.4 Class Diagram ของระบบ

จากรูปที่ 3.3 มีความสัมพันธ์กันระหว่างคลาสดังนี้

- คลาส Customer มีความสัมพันธ์กับคลาส Pet แบบ One to Many และถ้าไม่มีคลาส Customer ก็จะไม่เกิดคลาส Pet โดยคลาส Pet สามารถมีได้อย่างน้อย 1 ถึงหลายๆรายการ และแต่ละรายการสามารถอยู่ในคลาส Customer ได้เพียง 1 Customer

- คลาส Pet Type มีความสัมพันธ์กับคลาส Pet แบบ One to Many โดยคลาส Pet Type สามารถมีได้ตั้งแต่ 0 Pet เป็นต้นไป

- คลาส Pet Type มีความสัมพันธ์กับคลาส Disease แบบ One to Many โดยคลาส Pet Type สามารถมี Disease ได้ตั้งแต่ 0 โรคขึ้นไป

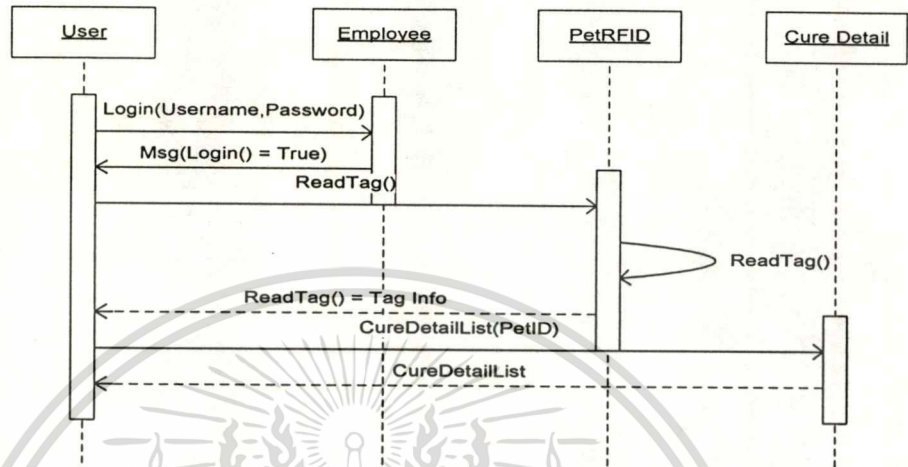
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- คลาส Pet RFID มีความสัมพันธ์กับ คลาส Pet เป็น One to One โดยทุกคลาส Pet RFID จะมีความสัมพันธ์กับคลาส Pet แต่ไม่ทุกคลาส Pet จะมีความสัมพันธ์กับคลาส Pet RFID
- คลาส Pet มีความสัมพันธ์กับ คลาส Cure Detail แบบ One to Many โดยถ้าไม่มีคลาส Pet ก็จะไม่เกิดคลาส Cure Detail โดยคลาส Pet สามารถมีคลาส Cure Detail ได้อย่างน้อย 1 ถึงหลายๆ รายการ และแต่ละคลาส Cure Detail สามารถอยู่ในคลาส Pet ได้เพียง 1 Pet
- คลาส Cure Detail มีความสัมพันธ์กับ คลาส Disease แบบ Many to Many โดย 1 Cure Detail สามารถมี Disease ได้ตั้งแต่ 0 ถึงหลายรายการ
- คลาส Cure Detail มีความสัมพันธ์กับ คลาส Nutrition แบบ One to Many โดย 1 Cure Detail สามารถมี Nutrition ได้ตั้งแต่ 0 ถึงหลายรายการ
- คลาส Employee มีความสัมพันธ์กับ คลาส Cure Detail แบบ One to Many โดย 1 Employee สามารถมี Cure Detail ได้ตั้งแต่ 0 ถึง หลายรายการ
- คลาส Employee มีความสัมพันธ์กับ คลาส Components Control แบบ One to One โดย 1 Employee สามารถควบคุม Components Control ได้ 1 รายการ
- คลาส Components Control มีความสัมพันธ์กับ คลาส Disease แบบ One to Many โดย 1 Components Control สามารถจัดการ Disease ได้หลายรายการ
- คลาส Components Control มีความสัมพันธ์กับ คลาส Nutrition แบบ One to Many โดย Components Control สามารถจัดการ Nutrition ได้หลายรายการ
- คลาส Components Control มีความสัมพันธ์กับ คลาส Pet Type แบบ One to Many โดย Components Control สามารถจัดการ Pet Type ได้หลายรายการ

### 3.2.1 Sequence Diagram

อธิบายถึงการทำงานของเหตุการณ์ต่างๆ และการสื่อสารกันระหว่าง อีอบเจกต์ต่างๆ

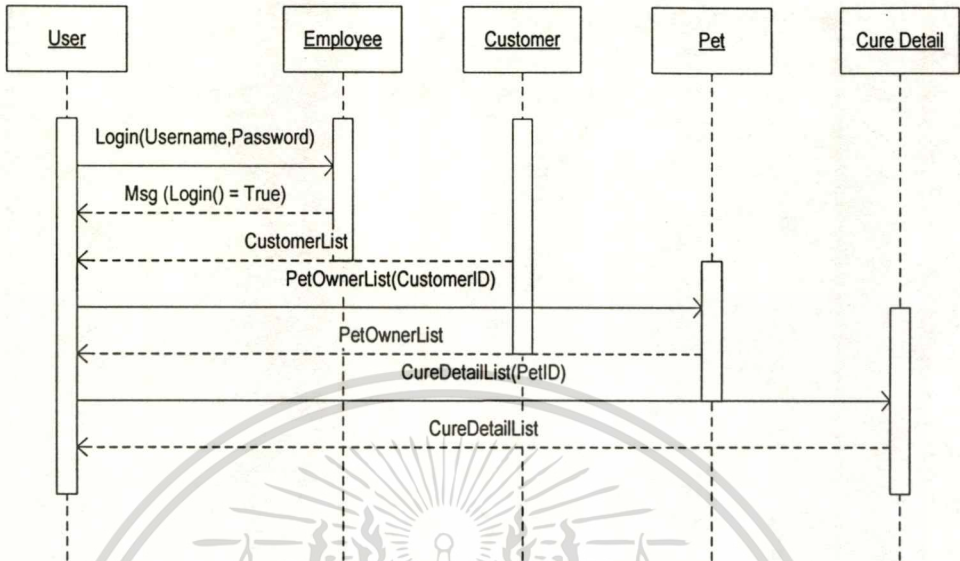
Case: Identification



รูปที่ 3.5 Sequence Diagram ในกรณีใช้การอ่านจาก RFID แท็ก

จากรูปที่ 3.5 ผู้ใช้ทำการ Login เข้าระบบก่อน โดยเรียกเมธอด Login จาก Employee และทำการส่งค่า Username และ Password ไปยัง Employee จากนั้น Employee จะส่ง Msg(Login) ตอบรับกลับมา ซึ่งหากถูกต้องจะมีค่าเป็น True จากนั้น ผู้ใช้เลือกที่จะใช้วิธีการอ่านจาก RFID แท็ก ผู้ใช้จึงเรียกเมธอด Readแท็ก() ของ PetRFID ซึ่ง PetRFID ทำการเรียกเมธอด Readแท็ก จากตัวเอง เพื่ออ่านค่า RFID แท็ก จากนั้นจึงนำผลลัพธ์ในรูปแบบแท็ก Info ส่งกลับไปยังผู้ใช้ ผู้ใช้เรียกดูข้อมูล CureDetailList จาก CureDetail โดยส่งค่า PetID จากนั้น CureDetail จึงส่ง CureDetailList กลับมา

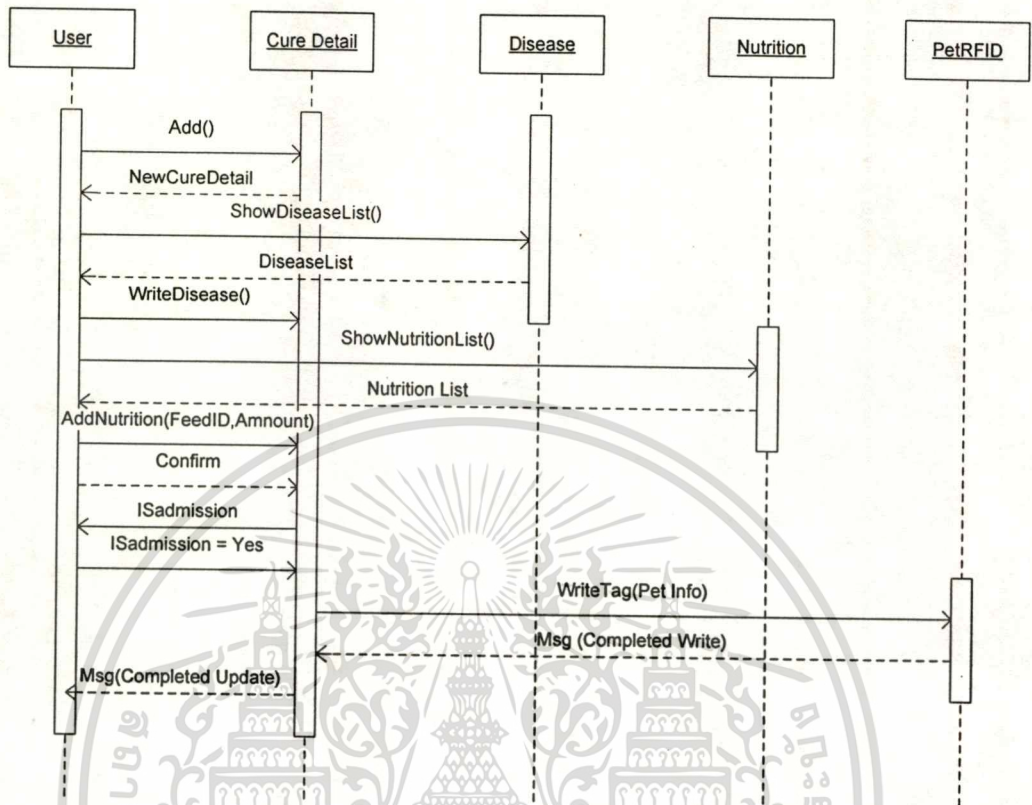
## Case: Enroll System



รูปที่ 3.6 Sequence Diagram ในกรณีเลือก Customer และ Pet

จากรูปที่ 3.6 ผู้ใช้ทำการ Login เข้าสู่ระบบก่อน โดยเรียกเมธอด Login จาก Employee และทำการส่งค่า Username และ Password ไปยัง Employee จากนั้น Employee จะส่ง Msg(Login) ตอบรับกลับมา ซึ่งหากถูกต้องจะมีค่าเป็น True จากนั้น Customer จะส่ง CustomerList มายัง User เมื่อ User เลือก Customer ที่ต้องการแล้วจึงส่งค่า CustomerID ไปยัง Pet เพื่อเรียกใช้เมธอด PetOwnerList จากนั้น Pet จึงส่ง PetOwnerList กลับมายัง User เมื่อ User เลือก PetID ได้ จึงจะส่งไปยัง CureDetail เพื่อเรียกใช้เมธอด CureDetailList เพื่อดูรายละเอียดการรักษาต่อไป

## Case: New Cure Detail

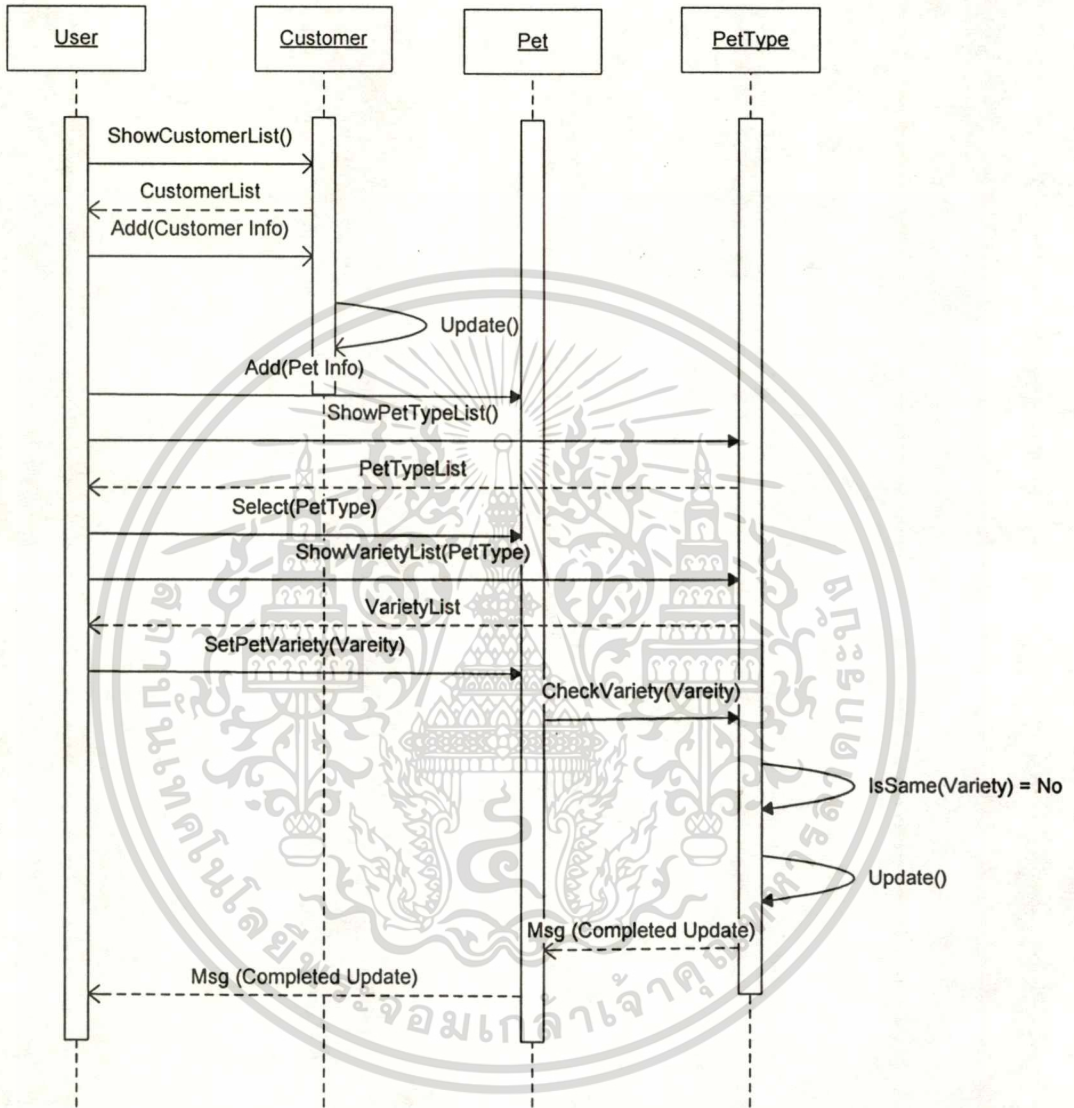


รูปที่ 3.7 Sequence Diagram ในกรณี เพิ่มการตรวจสัตว์เลี้ยง

จากรูปที่ 3.7 ผู้ใช้เรียกใช้เมธอด Add() จาก Cure Detail จากนั้น Cure Detail ส่ง NewCureDetail ในรูปแบบฟอร์มไปให้กับผู้ใช้ หลังจากนั้น ผู้ใช้จะเลือกจะทำการใส่อาการหรือโรคต่างๆ โดยการเรียกใช้ เมธอด ShowDiseaseList() ทั้งหมด จากนั้น Disease จะทำการส่ง Disease List กลับไปยังผู้ใช้ ซึ่ง ผู้ใช้ ทำการเลือก Disease และใส่ลงไปใน NewCureDetail โดยเรียกใช้เมธอด WriteDisease() จาก Cure Detail จากนั้น ผู้ใช้จะทำการเลือก ยาหรืออาหารในการรักษา ผู้ใช้จะเรียกข้อมูล Nutrition List จาก Nutrition โดยใช้เมธอด ShowNutritionList() จากนั้น ผู้ใช้ทำการเพิ่ม Nutrition ใส่ใน Cure Detail โดยใช้เมธอด AddNutrition() โดยใช้ค่าจาก FeedId และ Amount(จำนวนที่ของการให้ยา) เมื่อผู้ใช้ใส่ข้อมูลโรคและการให้อาหารเรียบร้อยแล้ว ผู้ใช้ทำการ Confirm หลังจากนั้น Cure Detail จะทำการถามผู้ใช้ว่าสัตว์เลี้ยงจะต้องพักรักษาที่โรงพยาบาลหรือไม่ โดยการส่ง Msg (Isadmssion) ไปยังผู้ใช้ ในกรณีที่สัตว์เลี้ยงจะต้องพักรักษาที่โรงพยาบาล ผู้ใช้ส่ง Isadmission = True กลับไปยัง Cure Detail จากนั้น Cure Detail จะต้องเขียนข้อมูลรักษาลงใน RFID แท็ก โดยการเรียกเมธอด Writeแท็ก จาก PetRFID โดยส่งข้อมูล Pet Info แนบไปด้วย

เมื่อ PetRFID เขียนข้อมูลเสร็จสิ้น จะทำการส่ง Msg (Completed Write) กลับไปเพื่อให้ Cure Detail ส่ง Msg (Completed update) ไปยังผู้ใช้ เป็นการแสดงถึงการทำงานเสร็จสิ้น

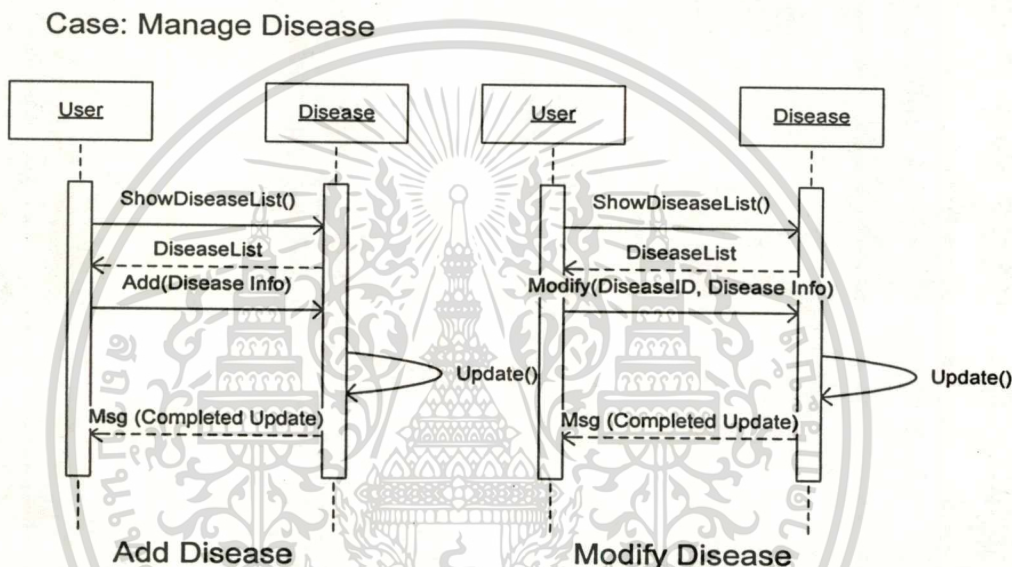
Case: Add New Customer and Customer's Pet



รูปที่ 3.8 Sequence Diagram ในกรณี เพิ่มข้อมูลลูกค้า

จากรูปที่ 3.8 ผู้ใช้เรียกต้องการเรียกดูข้อมูลลูกค้าทั้งหมดจาก Customer โดยใช้เมธอด ShowCustomerList() หลังจากนั้น Customer ส่ง CustomerList กลับมาให้ผู้ใช้ ผู้ใช้ทราบว่าข้อมูลลูกค้านี้เป็นลูกค้าใหม่เนื่องจากไม่มีอยู่ใน CustomerList จึงต้องการเพิ่มข้อมูลลูกค้าโดยเรียกใช้เมธอด Add() จาก Customer แล้วทำการแนบข้อมูล Customer Info ใหม่ลงไปด้วย เมื่อ Customer ได้รับข้อมูลใหม่ จึงทำการ Update หลังจากนั้น ผู้ใช้จะต้องเพิ่มข้อมูลสัตว์เลี้ยงใหม่ลงไปด้วย โดยการเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่จากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกใช้เมธอด Add() จาก Pet และทำการแนบข้อมูล Pet Info และ Pet Type ไปด้วย เมื่อได้รับข้อมูล PetType แล้ว Pet จะเรียกข้อมูลพันธุ์ของสัตว์ชนิดนั้นออกมาโดยเรียกเมธอด ShowVarietyList จาก PetType จากนั้น PetType จะส่ง VarietyList ซึ่งเป็นพันธุ์ของสัตว์ชนิดนั้นกลับมา จากนั้นผู้เลือกพันธุ์สัตว์โดยการเรียกใช้เมธอด SetPetVariety() และแนบ VarietyList ซึ่งเป็นพันธุ์สัตว์ไปด้วย จากนั้น Pet จะเรียกเมธอด CheckVariety จาก PetType เพื่อตรวจสอบว่าพันธุ์ที่ผู้ใช้ส่งมาให้เป็นพันธุ์ที่อยู่ในฐานข้อมูลแล้วหรือไม่ หากไม่มีจะทำการ Update() ข้อมูล และส่ง Msg (Completed Update) แสดงว่าข้อมูลได้เพิ่มเข้าไปสู่ในระบบเรียบร้อยแล้ว



รูปที่ 3.9 Sequence Diagram ในการจัดการ โรคของสัตว์เลี้ยง

จากรูปที่ 3.9 เป็นการแสดงลำดับการทำงานในการจัดการ โรคของสัตว์เลี้ยงในกรณีของ Add Disease (Diagram ด้านซ้าย) ผู้ใช้เรียกใช้เมธอด ShowDiseaseList() จาก Disease จากนั้น Disease ส่ง DiseaseList กลับไป ผู้ใช้จึงเพิ่มข้อมูล Disease Info โดยเรียกใช้ เมธอดจาก Add() จาก Disease และ Disease จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการเพิ่มข้อมูลแล้ว

ในกรณีของ Delete Disease (Diagram ตรงกลาง) ผู้ใช้เรียกใช้เมธอด ShowDiseaseList() จาก Disease จากนั้น Disease ส่ง DiseaseList กลับไป ผู้ใช้จึงลบข้อมูล โดยเรียกใช้ เมธอดจาก Delete() จาก Disease และแนบ DiseaseID ที่ต้องการจะลบ Disease จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการลบข้อมูลแล้ว

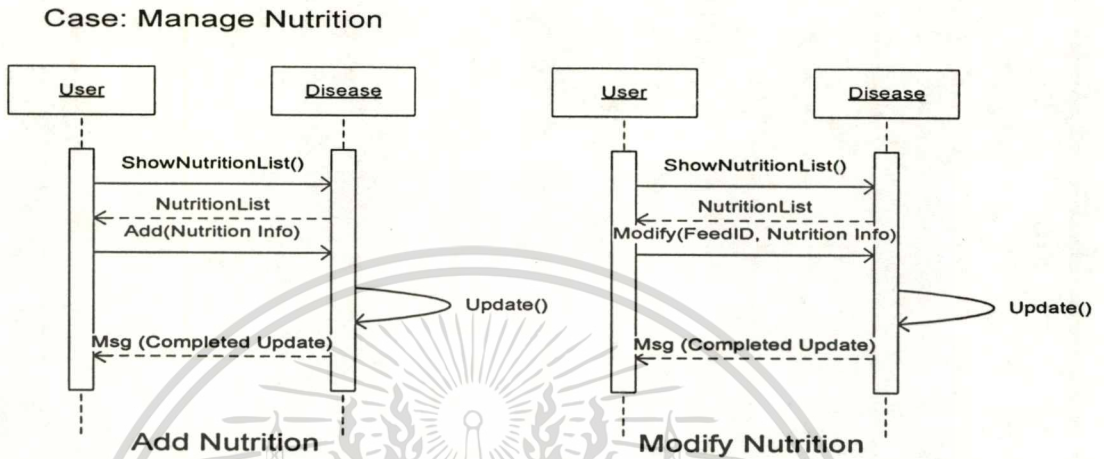
ในกรณีของ Modify Disease (Diagram ด้านขวา) ผู้ใช้เรียกใช้เมธอด ShowDiseaseList()

จาก Disease จากนั้น Disease ส่ง DiseaseList กลับไป ผู้ใช้จึงเปลี่ยนแปลงข้อมูล โดยเรียกใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปเผยแพร่ในเชิงพาณิชย์

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมธอดจาก Modify() จาก Disease และแนบ DiseaseID, Disease Info ที่ต้องการจะเปลี่ยนแปลง ส่งไปด้วย Disease จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการเปลี่ยนแปลงข้อมูลแล้ว



รูปที่ 3.10 Sequence Diagram การจัดการ การให้อาหารและยาแก่สัตว์เลี้ยง

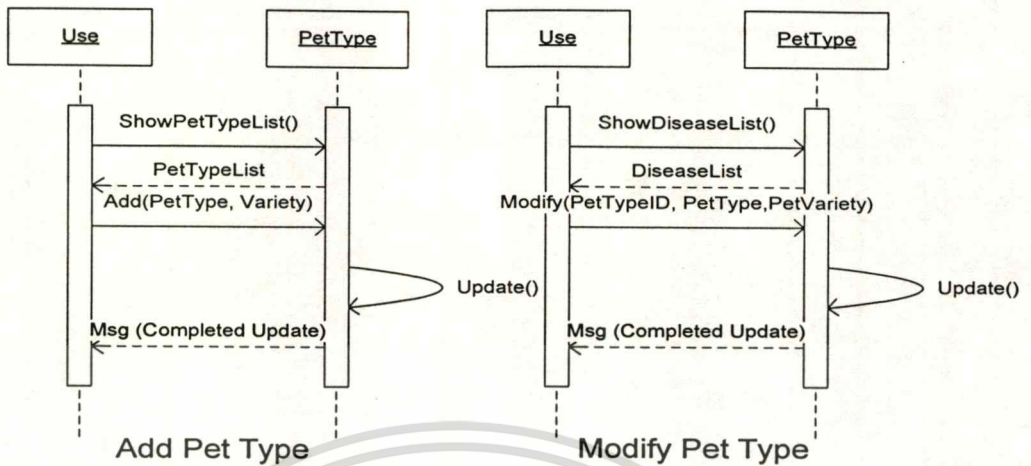
จากรูปที่ 3.10 เป็นการแสดงลำดับการทำงานในการจัดการ การให้อาหารและยาแก่สัตว์เลี้ยง

ในกรณีของ Add Nutrition (Diagram ด้านซ้าย) ผู้ใช้เรียกใช้เมธอด ShowNutritionList() จาก Nutrition จากนั้น Nutrition ส่ง NutritionList กลับไป ผู้ใช้จึงเพิ่มข้อมูล Nutrition Info โดยเรียกใช้ เมธอดจาก Add() จาก Nutrition และ Nutrition จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการเพิ่มข้อมูลแล้ว

ในกรณีของ Delete Nutrition (Diagram ตรงกลาง) ผู้ใช้เรียกใช้เมธอด ShowNutritionList() จาก Nutrition จากนั้น Nutrition ส่ง NutritionList กลับไป ผู้ใช้จึงลบข้อมูล โดยเรียกใช้ เมธอดจาก Delete() จาก Nutrition และแนบ FeedID ที่ต้องการจะลบ Nutrition จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการลบข้อมูลแล้ว

ในกรณีของ Modify Nutrition (Diagram ด้านขวา) ผู้ใช้เรียกใช้เมธอด ShowNutritionList() จาก Nutrition จากนั้น Nutrition ส่ง NutritionList กลับไป ผู้ใช้จึงเปลี่ยนแปลงข้อมูล โดยเรียกใช้ เมธอดจาก Modify() จาก Nutrition และแนบ FeedID, Nutrition Info ที่ต้องการจะเปลี่ยนแปลง ส่งไปด้วย Nutrition จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการเปลี่ยนแปลงข้อมูลแล้ว

## Case: Manage PetType



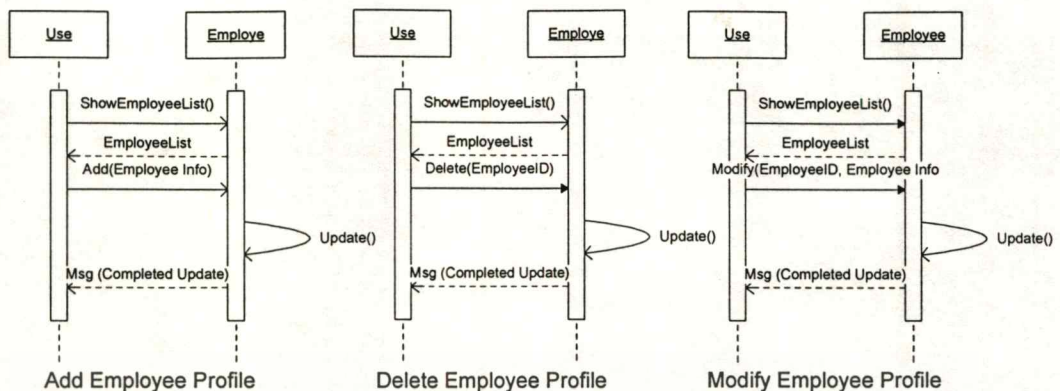
รูปที่ 3.11 Sequence Diagram การจัดการชนิดของสัตว์เลี้ยง

จากรูปที่ 3.11 เป็นการแสดงลำดับการทำงานในการจัดการ การจัดการชนิดของสัตว์เลี้ยง ในกรณีของ Add PetType (Diagram ด้านซ้าย) ผู้ใช้เรียกใช้เมธอด ShowPetTypeList() จาก PetType จากนั้น PetType ส่ง PetTypeList กลับไป ผู้ใช้จึงเพิ่มข้อมูล PetType Info โดยเรียกใช้ เมธอดจาก Add() จาก PetType และ PetType จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการเพิ่มข้อมูลแล้ว

ในกรณีของ Delete PetType (Diagram ตรงกลาง) ผู้ใช้เรียกใช้เมธอด ShowPetTypeList() จาก PetType จากนั้น PetType ส่ง PetTypeList กลับไป ผู้ใช้จึงลบข้อมูล โดยเรียกใช้ เมธอดจาก Delete() จาก PetType และแนบ PetTypeID ที่ต้องการจะลบ PetType จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการลบข้อมูลแล้ว

ในกรณีของ Modify PetType (Diagram ด้านขวา) ผู้ใช้เรียกใช้เมธอด ShowPetTypeList() จาก PetType จากนั้น PetType ส่ง PetTypeList กลับไป ผู้ใช้จึงเปลี่ยนแปลงข้อมูล โดยเรียกใช้ เมธอดจาก Modify() จาก PetType และแนบ PetTypeID, PetType Info ที่ต้องการจะเปลี่ยนแปลงส่งไป ด้วย PetType จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการเปลี่ยนแปลงข้อมูลแล้ว

## Case: Manage Employee Profile



รูปที่ 3.12 Sequence Diagram การจัดการประวัติพนักงาน

จากรูปที่ 3.12 เป็นการแสดงลำดับการทำงานในการจัดการประวัติพนักงาน

ในกรณีของ Add Employee (Diagram ด้านซ้าย) ผู้ใช้เรียกใช้เมธอด ShowEmployeeList() จาก Employee จากนั้น Employee ส่ง EmployeeList กลับไป ผู้ใช้จึงเพิ่มข้อมูล Employee Info โดยเรียกใช้ เมธอดจาก Add() จาก Employee และ Employee จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการเพิ่มข้อมูลแล้ว

ในกรณีของ Delete Employee (Diagram ตรงกลาง) ผู้ใช้เรียกใช้เมธอด ShowEmployeeList() จาก Employee จากนั้น Employee ส่ง EmployeeList กลับไป ผู้ใช้จึงลบข้อมูล โดยเรียกใช้ เมธอดจาก Delete() จาก Employee และแนบ EmployeeID ที่ต้องการจะลบ Employee จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการลบข้อมูลแล้ว

ในกรณีของ Modify Employee (Diagram ด้านขวา) ผู้ใช้เรียกใช้เมธอด ShowEmployeeList() จาก Employee จากนั้น Employee ส่ง EmployeeList กลับไป ผู้ใช้จึงเปลี่ยนแปลงข้อมูล โดยเรียกใช้ เมธอดจาก Modify() จาก Employee และแนบ EmployeeID, Employee Info ที่ต้องการจะเปลี่ยนแปลงส่งไปด้วย Employee จึงทำการ Update() ข้อมูล และส่ง Msg (Completed Update) เป็นการยืนยันการเปลี่ยนแปลงข้อมูลแล้ว

### 3.2.4 ตารางฐานข้อมูลต่างๆ ที่ใช้ในระบบ

มีทั้งหมด 11 ตารางดังตารางที่ 3.1

ตารางที่ 3.1 แสดงข้อมูลทั้งหมดของระบบ

ลำดับที่	ชื่อตาราง	ความหมาย
1.	CureDetail	ตารางข้อมูลอธิบายอาการป่วย
2.	Customer	ตารางข้อมูลลูกค้า
3.	DiseaseType	ตารางข้อมูลโรคสัตว์เลี้ยง
4.	Drug	ตารางข้อมูลยาและอาหารสัตว์เลี้ยง
5.	Employee	ตารางข้อมูลพนักงาน
6.	EmployeeRole	ตารางข้อมูลหน้าที่ของพนักงาน
7.	Pet	ตารางข้อมูลสัตว์เลี้ยง
8.	PetCureFeed	ตารางข้อมูลรายละเอียดการรักษาสัตว์เลี้ยง
9.	PetFeed	ตารางข้อมูลการให้อาหารและอาหาร
10.	PetType	ตารางข้อมูลชนิดและพันธุ์ของสัตว์
11.	Role	ตารางข้อมูลรายละเอียดหน้าที่

ตารางที่ 3.2 ตารางข้อมูลอธิบายอาการป่วย (CureDetail)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	CureID	Character	-	(PK) Not Null	หมายเลขอธิบายอาการป่วย
2.	State	Character	100	Null	อธิบายการป่วย
3.	Admission	Boolean	10	Not Null	สถานะการพักรักษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 ตารางข้อมูลลูกค้า (Customer)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	CustomerID	Character	10	(PK) Not Null	หมายเลขลูกค้า
2.	FirstName	Character	20	Not Null	ชื่อต้นลูกค้า
3.	LastName	Character	20	Not Null	ชื่อสกุลลูกค้า
4.	Phone	Character	15	Not Null	เบอร์โทรศัพท์ลูกค้า
5.	MPhone	Character	15	Null	เบอร์โทรศัพท์มือถือลูกค้า
6.	Address	Character	100	Not Null	ที่อยู่ลูกค้า

ตารางที่ 3.4 ตารางข้อมูลโรคสัตว์เลี้ยง (DiseaseType)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	DiseaseTypeID	Character	10	(PK) Not Null	หมายเลขโรคสัตว์
2.	PetType	Character	10	Not Null	ชนิดสัตว์
3.	Detail	Character	50	Not Null	รายละเอียดโรคสัตว์

ตารางที่ 3.5 ตารางข้อมูลยาสัตว์เลี้ยง (Drug)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	DrugID	Character	10	(PK) Not Null	หมายเลขอาหารและยาสัตว์
2.	Detail	Character	50	Not Null	รายละเอียดอาหารและยาสัตว์

ตารางที่ 3.6 ตารางข้อมูลพนักงาน (Employee)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	EmployeeID	Char	10	(PK) Not Null	หมายเลขพนักงาน
2.	FirstName	Char	20	Not Null	ชื่อต้นพนักงาน
3.	LastName	Character	20	Not Null	ชื่อสกุลพนักงาน
4.	Password	Character	10	Not Null	รหัสผ่านพนักงาน
5.	Phone	Character	15	Not Null	เบอร์โทรศัพท์พนักงาน
6.	MPhone	Character	15	Null	เบอร์โทรศัพท์มือถือพนักงาน
7.	Address	Character	100	Not Null	ที่อยู่พนักงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 ตารางข้อมูลหน้าที่ของพนักงาน (EmployeeRole)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	EmployeeID	Character	10	(PK) Not Null	หมายเลขพนักงาน
2.	RoleID	Character	10	(PK) Not Null	หมายเลขหน้าที่พนักงาน

ตารางที่ 3.8 ตารางข้อมูลสัตว์เลี้ยง (Pet)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	PetID	Character	10	(PK) Not Null	หมายเลขสัตว์เลี้ยง
2.	PetName	Character	30	Not Null	ชื่อสัตว์เลี้ยง
3.	PetTypeID	Character	10	(FK) Not Null	ชื่อพันธุ์สัตว์เลี้ยง
4.	PetIdentify	Character	50	Null	รอยคำหนีสัตว์เลี้ยง
5.	PetSex	Character	10	Null	เพศสัตว์เลี้ยง
6.	Weight	Integer	3	Null	น้ำหนักสัตว์เลี้ยง
7.	Age	Integer	3	Null	อายุสัตว์เลี้ยง
8.	CustomerID	Character	10	(FK)Not Null	หมายเลขลูกค้า

ตารางที่ 3.9 ตารางข้อมูลรายละเอียดการรักษาสัตว์เลี้ยง (PetCureFeed)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	PCFID	Character	10	(PK) Not Null	หมายเลขรายละเอียดการรักษาสัตว์เลี้ยง
2.	PetID	Character	30	(PK) Not Null	หมายเลขสัตว์เลี้ยง
3.	CureID	Character	10	(FK) Null	หมายเลขอธิบายการป่วย
4.	FeedID	Character	50	(FK) Null	หมายเลขการให้ยาและอาหารสัตว์เลี้ยง
5.	DateTime	Character	10	Not Null	วันและเวลารับการตรวจ
6.	EmployeeID	Character	10	(FK) Not Null	หมายเลขพนักงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.10 ตารางข้อมูลการให้อาหารและยา (PetFeed)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	FeedID	Character	10	(PK) Not Null	หมายเลขการให้ยาและอาหารสัตว์
2.	DrugID	Character	10	(PK) Not Null	หมายเลขยาและอาหาร
3.	DrugAmnt	Integer	100	Not Null	จำนวนยาและอาหาร

ตารางที่ 3.11 ตารางข้อมูลชนิดและพันธุ์สัตว์เลี้ยง (PetType)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	PetTypeID	Character	10	(PK) Not Null	หมายเลขการให้ยาและอาหารสัตว์เลี้ยง
2.	PetTypeDesc	Character	20	Not Null	ชื่อชนิดสัตว์
3.	PetVarietyDesc	Character	20	Not Null	ชื่อพันธุ์สัตว์

ตารางที่ 3.12 ตารางข้อมูลหน้าที่ (Role)

ลำดับ	ชื่อคอลัมน์	ประเภท	ขนาด	กำหนดค่า	รายละเอียด
1.	RoleID	Character	10	(PK) Not Null	หมายเลขหน้าที่
2.	RoleDesc	Character	20	Not Null	ชื่อหน้าที่
3.	Priority	Character	1	Not Null	ระดับความสำคัญ

## บทที่ 4

### การพัฒนาระบบงาน

#### 4.1 เครื่องมือที่ใช้ในการพัฒนาระบบ

##### 4.1.1 ภาษาที่ใช้พัฒนาระบบ

ระบบนี้ได้ถูกพัฒนาขึ้นโดยใช้ภาษา Visual Basic ซึ่งมีการใช้ Visual Studio 2005 ซึ่งเป็นซอฟต์แวร์ที่ใช้ในการเขียนโปรแกรมของบริษัท ไมโครซอฟท์ โดยออกวางตลาดเมื่อปลายปี 2005 โดยซอฟต์แวร์ดังกล่าวได้มีการเพิ่มศักยภาพในส่วนของ .NET Framework ซึ่งเป็นตัวหลักในการพัฒนา Pocket PC จากเวอร์ชันเดิมนั้นคือ Visual Studio 2003 ดังนั้นซอฟต์แวร์ Visual Studio 2005 จึงมีความเหมาะสมในการพัฒนาระบบที่ใช้ Pocket PC เป็นหลัก

##### 4.1.2 ระบบปฏิบัติการที่ใช้พัฒนาระบบ

ระบบนี้ได้ถูกพัฒนาขึ้นโดยใช้ระบบปฏิบัติการ Window XP Professional ของทางบริษัท ไมโครซอฟท์ ซึ่งถือว่าเป็นระบบปฏิบัติการที่สอดคล้องกับ ภาษา Visual Basic จึงถือเป็นระบบปฏิบัติที่ทำให้ผลของการรันหรือดำเนินการของระบบเป็นไปได้อย่างมีประสิทธิภาพมากที่สุด

##### 4.1.3 ฐานข้อมูลที่ใช้พัฒนาระบบ

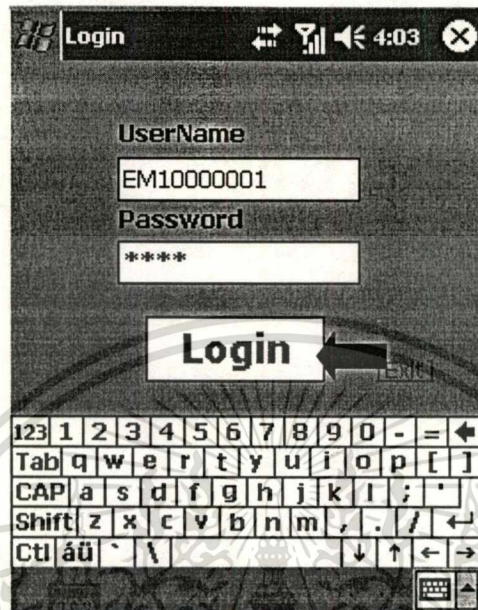
ระบบนี้ได้ถูกพัฒนาขึ้น โดยใช้ฐานข้อมูล SQL Server 2000 ของทางบริษัท ไมโครซอฟท์ ซึ่งถือว่าเป็นระบบฐานข้อมูลที่มีประสิทธิภาพ สามารถรองรับข้อมูลขนาดใหญ่ได้ และอีกทั้งยังสอดคล้องกับ ภาษา Visual Basic จึงทำให้การพัฒนาระบบมีความยืดหยุ่นมากขึ้น

##### 4.1.4 เครื่องอ่าน RFID ที่ใช้พัฒนาระบบ

ระบบนี้ได้ถูกพัฒนาขึ้น โดยใช้เครื่องอ่าน RFID Reader ของ บริษัท OMRON ใช้คลื่นความถี่ในการรับความถี่ต่ำ 125 - 134 KHz ระยะ < 0.5 เมตร (แต่อ่านและเขียนได้จริงในระยะไม่เกิน 3 เซนติเมตร) การอ่านเป็นแบบอ่านที่ละการ์ด โดย RFID ชนิดนี้จะเหมาะกับการใช้การอ่านและเขียน TAG ประเภท Philip I Code ในส่วนของ TAG เป็นประเภท Philip I Code ชนิด Passive TAG จึงมีความเหมาะสมต่อการพัฒนาระบบที่ใช้ในการอ่านและเขียน

## 4.2 หน้าจอของระบบ

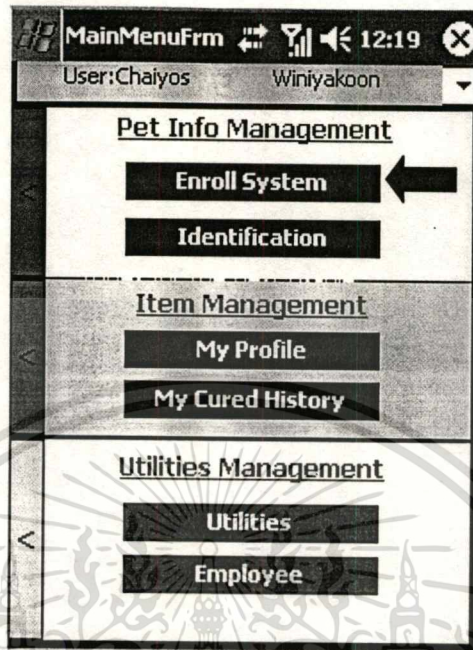
### 4.2.1 ส่วนการเข้าระบบ



รูปที่ 4.1 แสดงการล็อกอินเข้าสู่ระบบ

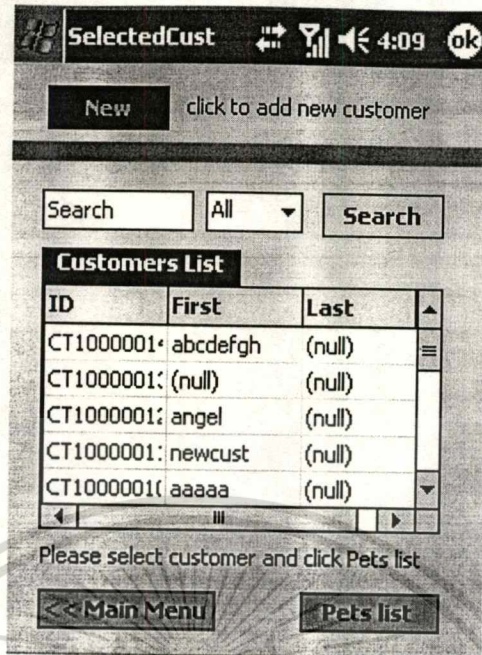
1. User ทำการใส่ข้อมูล UserName และ Password
2. กดปุ่ม Login เพื่อเข้าสู่ระบบ และเป็นการ Authenticated การเข้าถึงระบบของ User แต่ละคน

#### 4.2.2 ส่วนของการเข้าถึงข้อมูลสัตว์เลี้ยงแบบปกติ



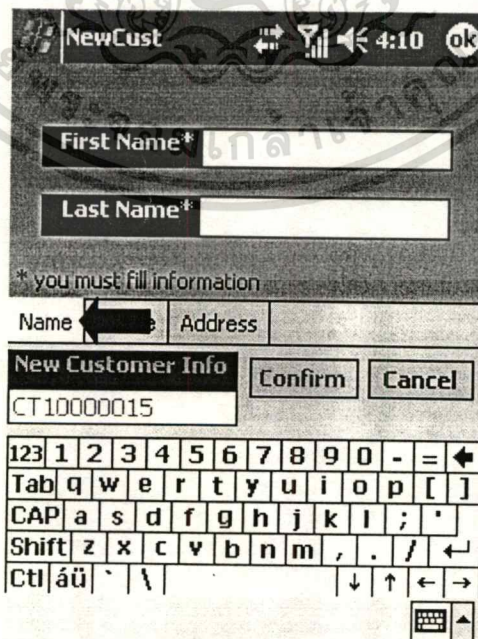
รูปที่ 4.2 แสดงหน้าเมนูหลักเพื่อเลือกรายการเข้าใช้งาน

1. เมื่อผ่านการ Login เข้าสู่ระบบ โดยจะจำกัดการเข้าถึงข้อมูลของ User แต่ละคน โดย Interface นี้จะแสดงถึงการทำงานหลักๆของระบบดังนี้
  - 1.1 Pet Info Management เป็นส่วนของข้อมูลของสัตว์เลี้ยงแต่ละตัวที่เข้ามาทำการรักษาในโรงพยาบาล โดยแบ่งการเข้าถึงข้อมูลสัตว์เป็น 2 ประเภท คือ Identification (ใช้ RFID) หรือ Enroll System
  - 1.2 Item Management เป็นส่วนการจัดการข้อมูลของ User (My Profile) และการตรวจสอบประวัติการรักษาของสัตว์ต่างๆ (My Cured History)
  - 1.3 Utilities Management เป็นส่วนที่ Receptor ใช้ในการจัดการข้อมูลของ Disease, Nutrition, PetType และ Role(บทบาทหน้าที่ของ User แต่ละคน) รวมถึงการเพิ่มจำนวนของพนักงาน และแก้ไขบทบาทของพนักงานแต่ละคน
- 2 โดยสมมติว่า User ทำการเลือก Enroll System ใน Pet Information Management



รูปที่ 4.3 แสดงรายชื่อลูกค้าที่บันทึกอยู่ในฐานข้อมูล

- 1 เมื่อเราทำการเลือก Enroll System จากนั้นจะทำการ เลือก Customer จาก Customer List ที่มีข้อมูลอยู่ในฐานข้อมูล
- 2 เมื่อเลือก Customer เรียบร้อยแล้วจึงกดปุ่ม Pets list แต่หากกดปุ่ม Main Menu จะเป็นการกลับไปยังหน้าก่อนหน้านี้ ซึ่งเป็น Main Menu ของระบบ
- 3 สมมติว่า User กดปุ่ม New ซึ่งจะเป็นการเพิ่ม Customer ใหม่ลงไป ในฐานข้อมูล



รูปที่ 4.4 แสดงหน้าจอการเพิ่มรายชื่อลูกค้าใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1 เมื่อทำการกดปุ่ม New จะทำการ Add Customer ซึ่งช่องที่มีเครื่องหมาย \* แสดงว่าบังคับ ให้ User ต้องใส่ข้อมูลนั้นๆ ของ Customer ให้ครบ โดยเริ่มจากการใส่ชื่อ First Name และ Last Name
- 2 จากนั้นจึงกดปุ่ม Phone จะ ได้ผลดังนี้

รูปที่ 4.4.1 แสดงหน้าจอการเพิ่มรายละเอียดลูกค้าใหม่

- 1 User จะทำการใส่ข้อมูล Phone number ของ Customer โดยแบ่งเป็น Phone number และ Mobile number
- 2 จากนั้นจึงกดปุ่ม Address จะ ได้ผลดังนี้

รูปที่ 4.4.2 แสดงหน้าจอการเพิ่มรายละเอียดที่อยู่ลูกค้าใหม่

- 1 ทำการใส่ข้อมูล Address ของ Customer โดยระบบจะสร้าง Customer Info ให้แก่ลูกค้าใหม่ ซึ่งคือ รหัสของ Customer แต่ละคน ที่ไม่ซ้ำกัน
- 2 เมื่อ User ทำการใส่ข้อมูลของ Customer เสร็จเรียบร้อย ปุ่ม Confirm จะสามารถกดได้ เพื่อเป็นการยืนยันการเพิ่ม Customer ลงในฐานข้อมูลของระบบ

Name	Type	Variety
Jooo	Unknown	Unknown
32132321231	Dog	Unknown
Michael	Tiger	HardPattern
Lueng	Dog	BangKaew

รูปที่ 4.5 แสดงหน้าจอรายชื่อสัตว์เลี้ยง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1 จากนั้นจะเป็นการเลือกสัตว์เลี้ยงของ Customer คนนั้นๆที่ User ทำการเลือกมา โดยกดที่บรรทัดของสัตว์เลี้ยงที่ต้องการนั้น หรือทำการเพิ่มสัตว์เลี้ยง โดยกดปุ่ม New
- 2 ในหน้านี้ User สามารถที่จะดูรายละเอียดของ Customer ที่เลือกมาได้ โดยกดที่ปุ่ม drop down ของ list จะปรากฏรายละเอียดของ Customer นั้นมา
- 3 หากกดปุ่ม Back จะกลับไปยังหน้า SelectedCust อีกครั้ง แต่หากกดปุ่ม Main Menu จะเป็นการกลับไปยังหน้า MainMenuFrm

รูปที่ 4.6 แสดงหน้าจอการเพิ่มชื่อสัตว์เลี้ยงใหม่

- 1 เมื่อ User ทำการกด New เพื่อเป็นการ Add Pet จะขึ้นหน้า AddPet ขึ้นมา โดยให้ User ใส่ชื่อสัตว์เลี้ยง และช่องที่มีเครื่องหมาย \* แสดงว่าบังคับให้ User ต้องใส่ข้อมูลนั้นๆ ของ Pet ให้ครบ
- 2 จากนั้นหาก User กด Type จะปรากฏหน้าจอดังนี้

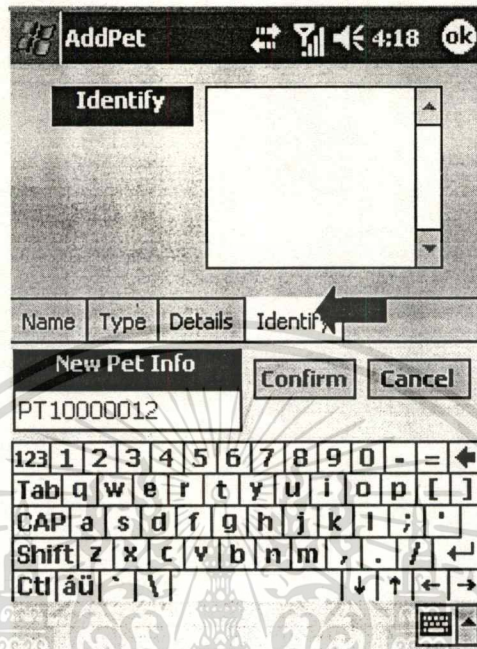
รูปที่4.6.1 แสดงหน้าจอการเพิ่มประเภทและพันธุ์สัตว์เลี้ยงใหม่

- 1 User ทำการใส่ข้อมูล Type คือ ชนิดของสัตว์เลี้ยง และ Variety คือ พันธุ์ของสัตว์เลี้ยงนั้น โดย User สามารถเลือกข้อมูลที่มีอยู่ในฐานข้อมูลของระบบ โดยกดที่ drop down ของ list นั้นๆ จะมีตัวเลือกปรากฏขึ้นมา
- 2 จากนั้นหาก User กด Detail จะปรากฏหน้าจอดังนี้

รูปที่4.6.2 แสดงหน้าจอการเพิ่มรายละเอียดสัตว์เลี้ยงใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1 User ทำการใส่ข้อมูล Weight และ Age ของสัตว์เลี้ยง ซึ่งสามารถเลือกหน่วยกำกับได้
- 2 จากนั้นหาก User กด Identify จะปรากฏหน้าจอดังนี้



รูปที่ 4.6.3 แสดงหน้าจอการเพิ่มรายละเอียด Identify สัตว์เลี้ยงลูกค้าใหม่

- 1 User ใส่รายละเอียดเกี่ยวกับสัตว์เลี้ยงที่ Add เข้ามาในระบบ เพื่อแยกแยะข้อแตกต่างของสัตว์เลี้ยงต่างๆ ได้สะดวกขึ้น
- 2 ในช่อง New Pet Info ระบบจะทำการสร้างรหัส Pet เพื่อเป็นรหัสประจำตัวสัตว์เลี้ยงแต่ละตัว
- 3 เมื่อเรารอกข้อมูลครบเรียบร้อย ปุ่ม Confirm จะสามารถกดได้ เพื่อเป็นการยืนยันการ Add Pet นั้นๆ

Cured History

Customer/Pet Name

Cured History 13/08/06 12:09

Disease Detail: Admission : OFF

State

by Chaiyos Winiyakoon

Description	Amount
Paracentamol	30
Tiiffy	40

<< Main Menu Add

<< Back

รูปที่ 4.7 แสดงหน้าจอประวัติการรักษาของสัตว์เลี้ยง

- 1 หน้า Cured History จะแสดง ชื่อของ Customer และ Pet Name โดย User สามารถกด drop down ลงมาเพื่อแสดงรายละเอียดต่างๆ ได้, Disease Detail จะแสดงรายละเอียดของโรคที่ Doctor ได้ใส่ไว้เพื่อประกอบการรักษา, Admission : OFF แสดงว่า สัตว์เลี้ยงตัวนั้น ไม่ได้พักที่โรงพยาบาล แต่หากเป็น IN แสดงว่าสัตว์เลี้ยงตัวนั้นมารับการรักษา โดยต้องอยู่ที่โรงพยาบาล เป็นระยะเวลาหนึ่ง, ชื่อของ Doctor ที่ทำการรักษา และ รายละเอียดของยา ที่ Doctor สั่งจ่ายให้ Pet ต่างตามประวัติการรักษา
- 2 เมื่อ User กดปุ่ม Confirm ระบบจะแสดงหน้าจอ Cured History คือ แสดงประวัติการรักษาสัตว์เลี้ยงตัวนั้นๆที่เราเลือก โดยสามารถเลือกดูประวัติการรักษาได้จากวันที่ใน list ของ Cured History โดยกด drop down ลงมา

Cured History 4:23 ok

Customer/Pet Name

Cured History 13/08/06 12:09

Disease Detail: 13/08/06 12:09

State 13/08/06 12:08

25/04/06 09:20

13/04/06 12:20

by Chaiyos Winiyakoon

Description	Amount
Paracentamol	30
Tiiffy	40

<< Main Menu Add

<< Back

รูปที่ 4.7.1 แสดงหน้าจอประวัติการรักษาของสัตว์เลี้ยง (2)

1 เมื่อ User กด drop down ที่ list ของชื่อ Doctor ที่ทำการรักษา จะมี Role ของ Doctor แต่ละคนปรากฏออกมา เช่นดังตัวอย่างต่อไป

Cured History 4:23 ok

Customer/Pet Name

Cured History 13/08/06 12:09

Disease Detail: Admission : OFF

State

by Chaiyos Winiyakoon

Description	Role
Paracentamol	Board
Tiiffy	Doctor
	Specialist

<< Main Menu Add

<< Back

รูปที่ 4.7.2 แสดงหน้าจอประวัติการรักษาของสัตว์เลี้ยง (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.3 ส่วนของการเข้าถึงข้อมูลสัตว์เลี้ยงแบบใช้เครื่องอ่าน

รูปที่ 4.8 แสดงหน้าจอเครื่องอ่าน RFID

เมื่อผู้ใช้ต้องการจะเข้าถึงข้อมูลสัตว์เลี้ยงด้วยเครื่องอ่าน RFID Reader ผู้ใช้จะสามารถกดปุ่ม Read เพื่อทำการอ่านข้อมูลจาก Tag ของสัตว์เลี้ยงได้ โดยเมื่ออ่านได้แล้ว ผู้ใช้สามารถดูข้อมูล General Info ซึ่งเป็นข้อมูลทั่วไปที่แพทย์ทำการรักษาสัตว์เลี้ยงได้ด้วย

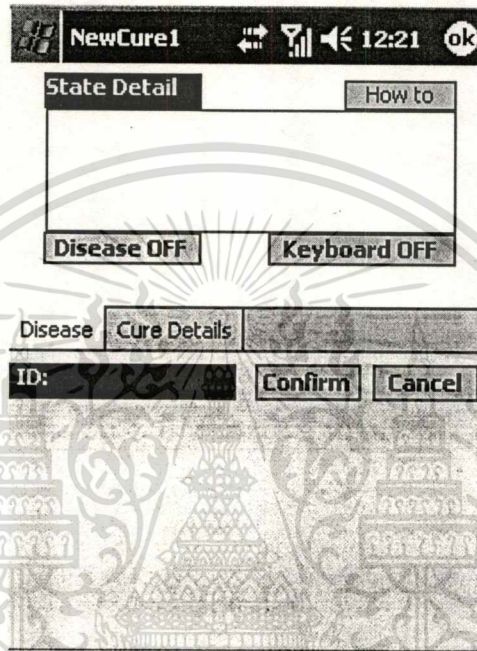
Description	Amount
Paracentamol	30
Tiiffy	40

รูปที่ 4.8.1 แสดงหน้าจอรายละเอียดสัตว์เลี้ยง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ผู้ใดเห็นนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

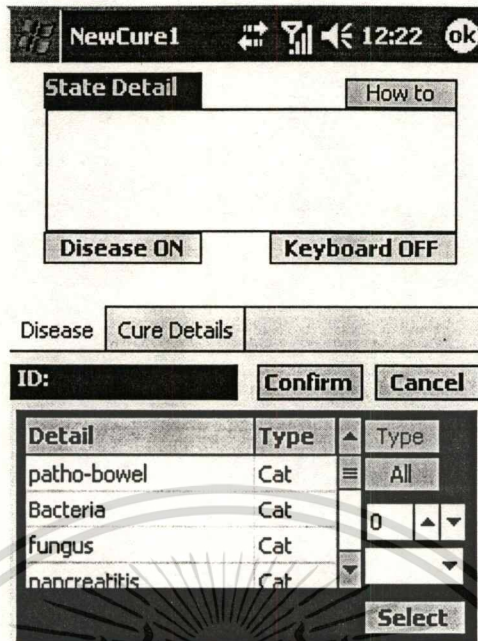
ผู้ใช้สามารถดูข้อมูล General Info ซึ่งเป็นข้อมูลทั่วไปที่แพทย์ทำการรักษาสัตว์เลี้ยงได้ด้วย โดยสามารถเลือกจากวันที่รักษา ข้อมูลการรักษาจะแสดงขึ้นมา ไม่ว่าจะเป็นข้อมูลโรคของสัตว์เลี้ยง หรือ ข้อมูลการให้ยาแก่สัตว์เลี้ยง

#### 4.2.4 ส่วนของการเพิ่มข้อมูลการรักษาสัตว์เลี้ยง



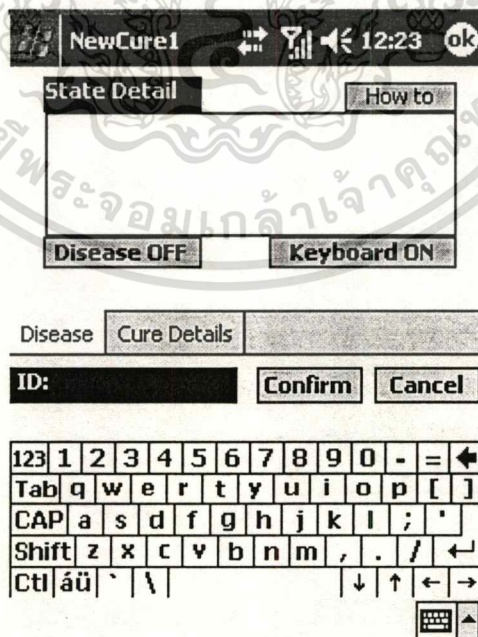
รูปที่ 4.9 แสดงหน้าจอสร้างประวัติการรักษาใหม่

หน้าจอในส่วนของการจดบันทึกอาการ และ เลือกวิธีการรักษา โดยแบ่งเป็น 2 ส่วนหลักคือ Disease และ Cure Detail โดยในส่วนของ Disease ช่องข้อความในการบันทึก โดยผู้ใช้สามารถเลือกการบันทึกได้ 2 แบบ คือ การใช้เครื่องมือช่วย กับ การพิมพ์ด้วยตนเอง หากต้องการใช้เครื่องมือช่วยให้กด Disease OFF แต่หากต้องการพิมพ์ด้วยตนเองให้กด Keyboard OFF



รูปที่4.9.1 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (2)

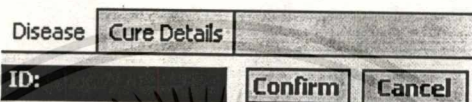
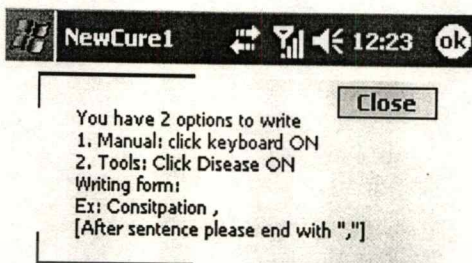
เมื่อกด Disease OFF ปุ่ม Disease OFF จะเปลี่ยนเป็น Disease ON นั้นหมายถึง ส่วนของ เครื่องมือได้เปิดให้ใช้งานแล้ว ซึ่งในส่วนนี้จะเป็นการเลือกชนิดของโรคสัตว์ที่เป็นอยู่ โดยผู้ใช้ สามารถเลือกได้ 2 ทางเลือก คือ 1. Type แสดงรายละเอียดโรคที่สัตว์เลี้ยงเป็นบ่อตามชนิดของ สัตว์เลี้ยง และ 2.All แสดงรายละเอียดโรคที่สัตว์เลี้ยงเป็นบ่อแบบแสดงสัตว์เลี้ยงทุกชนิด



รูปที่4.9.2 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (3)

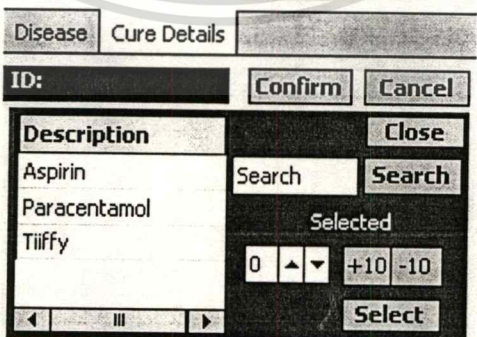
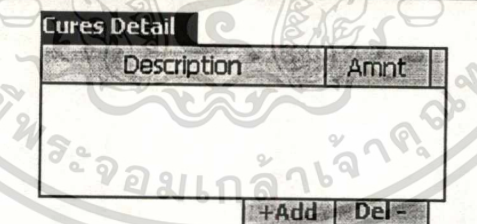
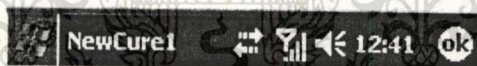
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อกด Keyboard OFF ปุ่ม Keyboard OFF จะเปลี่ยนเป็น Keyboard ON นั้นหมายถึง ส่วนของการพิมพ์เองเปิดให้ใช้งานแล้ว



รูปที่ 4.9.3 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (4)

เมื่อผู้ใช้งานกดปุ่ม How to จะเป็นการบอกช่องทางการเขียนในช่องข้อความ โรคของสัตว์เลี้ยง



รูปที่ 4.9.4 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (5)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อกดปุ่มในส่วนของ Cure Details จะมีช่องว่างให้เพิ่มรายละเอียด ซึ่งผู้ใช้จะสามารถเพิ่มการให้ยารักษาโดยการกดปุ่ม +Add หลังจากนั้น จะมีประเภทยารักษาให้เลือก

Description	Amnt
Aspirin	2
Paracentamol	13

+Add Del -

Disease Cure Details

ID: Confirm Cancel

Description Close

Aspirin Search Search

Paracentamol

Tiiffy

Selected Item

0 +10 -10

Select

รูปที่ 4.9.5 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (6)

เมื่อเลือกยารักษา รายการให้ยารักษาจึงปรากฏในส่วนด้านบนพร้อมกับจำนวนของยารักษา

Description	Amnt
Aspirin	2
Paracentamol	13

Please prepare Tag for recording Information !!

Dis Admission

ID: Confirm

Di

Aspirin Search Search

Paracentamol

Tiiffy

Selected Item

0 +10 -10

Select

รูปที่ 4.9.6 แสดงหน้าจอสร้างประวัติการรักษาใหม่ (7)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้กดปุ่ม Confirm เป็นการยืนยันว่าจะบันทึกข้อมูลการรักษา จะมีข้อความขึ้นมาถามว่า สัตว์เลี้ยงดังกล่าวจะต้องเข้ารับการ Admission หรือไม่ ซึ่งหากเลือก Admission จะมีข้อความขึ้นเตือนให้นำ TAG มาเพื่อบันทึกรายการรักษาต่อไป

#### 4.2.5 ส่วนของการเปลี่ยนแปลงข้อมูลส่วนตัว



รูปที่ 4.10 หน้าจอแสดงรายละเอียดผู้พัฒนาโปรแกรม

เมื่อผู้ใช้กดปุ่ม My Profile ระบบจะแสดงทางเลือกการเปลี่ยนแปลงข้อมูล คือ 1. เปลี่ยนแปลงข้อมูล ชื่อ โทรศัพท์ ที่อยู่ เป็นต้น 2. ส่วนของการเปลี่ยนแปลงรหัสผ่าน

MyProfile 2:26 ok

First Name Chaiyos

Last Name Winiyakoon

Phone 02758-1038

MPhone 081-651-4798

Address 81/14 Tippawan Village.  
Bangmuengmai Mueng

View Confirm Cancel

Edit Profile Change Password

123 1 2 3 4 5 6 7 8 9 0 - = <

Tab q w e r t y u i o p [ ]

CAP a s d f g h j k l ; ' <

Shift z x c v b n m , . / <

Ctl á ü ` \ | / < >

รูปที่ 4.10.1 หน้าจอแสดงการเพิ่มรายละเอียดผู้พัฒนาโปรแกรม

เมื่อผู้ใช้กดปุ่ม Edit Profile ระบบจะแสดงข้อมูลของผู้ใช้ในรูปแบบมุมมอง โดยเมื่อผู้ใช้กดปุ่ม Edit ระบบจะเปลี่ยนเป็นระบบรับการเปลี่ยนแปลงโดยจะมี คีย์บอร์ดแสดงขึ้นมาเพื่อรอรับค่าของผู้ใช้ในการเปลี่ยนแปลง

MyProfile 2:27 ok

Password >> [ ]

Confirm Password >> [ ]

[Not over than 8 characters]

Confirm Cancel

Edit Profile Change Password

รูปที่ 4.10.2 แสดงหน้าจอการเปลี่ยนรหัสของผู้พัฒนาโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

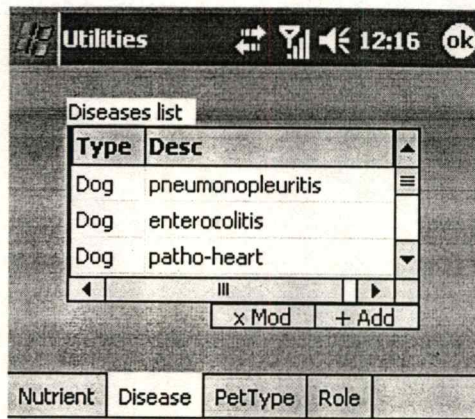
เมื่อผู้ใช้เลือก Change Password นั่นคือ ผู้ใช้ต้องการเปลี่ยนรหัสผ่าน โดยผู้ใช้จะต้องพิมพ์รหัสผ่านไม่เกิน 8 ตัวอักษร และ ยืนยันด้วยรหัสเดิมอีกครั้งเป็นการยืนยันการเปลี่ยนแปลง

4.2.6 ส่วนของการเพิ่มเติมเปลี่ยนแปลงองค์ประกอบ (โรคสัตว์เลี้ยง,ยารักษา,ชนิดสัตว์เลี้ยง,หน้าที่พนักงาน)



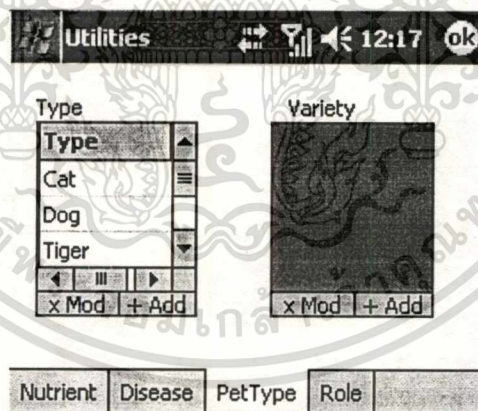
รูปที่ 4.11.1 แสดงหน้าจอรายชื่อ Nutrient

เมื่อเลือกในส่วนของ Nutrient จะเป็นในส่วนของการเพิ่มเติม หรือ เปลี่ยนแปลงยารักษาสัตว์เลี้ยงได้



รูปที่ 4.11.2 แสดงหน้าจอรายชื่อโรค (Diseases)

เมื่อเลือกในส่วนของ Disease จะเป็นในส่วนของการเพิ่มเติม หรือ เปลี่ยนแปลงโรคของสัตว์เลี้ยงได้ โดยผู้ใช้งานจะต้องระบุชนิดของสัตว์เลี้ยงไปพร้อมกับระบุโรคของสัตว์เลี้ยงด้วย



รูปที่ 4.11.3 แสดงหน้าจอรายชื่อประเภทสัตว์เลี้ยง

เมื่อเลือกในส่วนของ PetType จะเป็นในส่วนของการเพิ่มเติม หรือ เปลี่ยนแปลงชนิดและประเภทสัตว์เลี้ยงได้ โดยผู้ใช้งานสามารถเลือกได้ว่า จะเพิ่มชนิดสัตว์เลี้ยงใหม่ หรือ เพิ่มพันธุ์สัตว์เลี้ยงเอกสารใหม่แต่เป็นชนิดสัตว์เลี้ยงเก่าได้ ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Roles list

Desc.	Priorit	
General Doctor	0	
Dog Specialist	0	
Cat Specialist	0	

x Mod + Add - Del

Nutrient	Disease	PetType	Role
----------	---------	---------	------

#### รูปที่ 4.11.4 แสดงหน้าจอรายชื่อหน้าที่ของพนักงาน

เมื่อเลือกในส่วนของ Role จะเป็นในส่วนของการเพิ่มเติม หรือ เปลี่ยนแปลงหน้าที่ของพนักงานได้ โดยจะมีการกำหนด Priority หรือ ลำดับความสำคัญของหน้าที่ได้

#### 4.2.7 ส่วนของการเพิ่มเติมพนักงาน หรือ เปลี่ยนแปลงหน้าที่พนักงาน



Employees List

ID	Fname	Lname
EM10000001	Chaiyos	Winiyakoon

+Add -Del Edit Role

Search All Search

เมื่อเลือกในส่วนของ Employee จะเป็นในส่วนของการเพิ่มเติม หรือ เปลี่ยนแปลงพนักงานได้

Employee 3:09 ok

First Name\*

Last Name\*

\* you must fill information

Name	Phone	Address	Role
EM10000002			

Confirm Cancel

123 1 2 3 4 5 6 7 8 9 0 - = <

Tab q w e r t y u i o p [ ]

CAP a s d f g h j k l ; ' <

Shift z x c v b n m , . / <

Ctl | á ñ | ` | \ | | ↓ ↑ ← →

รูปที่ 4.12 แสดงหน้าจอเพิ่มรายชื่อของพนักงานใหม่

เมื่อกดปุ่ม +Add ระบบจะให้ผู้ใช้กรอกข้อมูลที่จำเป็นเกี่ยวกับพนักงานโดยรวมไปถึง หน้าที่ของพนักงานคนๆนั้นด้วย

## บทที่ 5

# สรุปผลและข้อเสนอแนะ

### 5.1 สรุปผลการดำเนินงาน

จากการทดลองใช้งานระบบ RFID ประยุกต์ใช้กับระบบดูแลรักษาสัตว์ พบว่าระบบนี้สามารถทำงานบนปฏิบัติการได้เฉพาะบน Pocket PC และจะต้องมีการกำหนดค่าพอร์ตของเครื่องคอมพิวเตอร์ให้ถูกต้องก่อนจึงจะสามารถใช้งานได้ (โดยจะต้องกำหนดให้ค่าพอร์ตของเครื่องอยู่ที่พอร์ต 5 ) สามารถลดเวลาการเข้าถึงข้อมูลสัตว์เลี้ยง ได้อย่างมีประสิทธิภาพโดยการนำเทคโนโลยี Pocket PC มาใช้ ตั้งแต่การบันทึกการลูกค้า สัตว์เลี้ยง พนักงาน ตลอดจน การนำเทคโนโลยี RFID มาใช้ในการระบุอัตลักษณ์ของสัตว์เลี้ยง โดยระบบจะได้สร้างจอร์บและแสดงผลให้กับผู้ใช้ ได้อย่างลงตัวในแบบฉบับเฉพาะของ Pocket PC

แต่อย่างไรก็ตามระบบนี้ยังมีการกำหนดพอร์ต ซึ่งเป็นการกำหนดแบบตายตัว ยังไม่มีความยืดหยุ่น และในส่วนของความเร็วเว็บเซอร์วิส ซึ่งมีความช้ากว่าระบบอีเธอร์เนตเป็นอย่างมาก แต่คาดว่าอีกไม่นาน เทคโนโลยี เว็บเซอร์วิส จะมีการถูกพัฒนาให้เร็วยิ่งขึ้น

### 5.2 ปัญหาที่พบในการพัฒนาระบบ

เนื่องจากตัว Pocket PC 2003 Emulator ที่มีมากับ Visual Studio 2005 มีการกำหนดแรมเท่ากับ 64 MB ตายตัว เมื่อขนาดข้อมูลของระบบที่ถูกพัฒนามีขนาดใหญ่ จึงทำให้การดำเนินงานของ Emulator เป็นไปได้ช้ากว่าที่ควรจะเป็น อีกทั้งการแมปพอร์ตของ Emulator กับ พอร์ตของคอมพิวเตอร์ จะต้องมีการเชื่อมต่อด้วยซอฟต์แวร์ Virtual Network ในการช่วยแมปพอร์ต(มีไวรัส Brontoks แบนมากับตัวซอฟต์แวร์ดังกล่าวด้วย)

### 5.3 ข้อเสนอแนะ

#### 5.3.1 เทคโนโลยีอื่น ๆ ที่สามารถนำเข้าใช้ร่วมกับระบบดูแลรักษาสัตว์

เทคโนโลยีอาร์เอฟไอดีเป็นเทคโนโลยีระบุอัตลักษณ์ที่มีประสิทธิภาพสูง แต่ในขณะเดียวกันก็มีค่าใช้จ่ายสูงตามมา ซึ่งการนำเทคโนโลยีมาใช้ในการประยุกต์ใช้ระบบดูแลรักษาสัตว์มีทางเลือกได้หลายทาง เช่น เทคโนโลยี บาร์โค้ด ซึ่งหากขอบเขตของระบบมีอย่างจำกัด การใช้เทคโนโลยี บาร์โค้ดจึงเป็นทางเลือกที่มีประสิทธิภาพทั้งระบุอัตลักษณ์ และ ค่าใช้จ่าย

การนำระบบดูแลรักษาสัตว์มาประยุกต์ร่วมกับเทคโนโลยีอาร์เอฟไอดี ในขอบเขตดังกล่าวทำให้ไม่สามารถนำเทคโนโลยีอาร์เอฟไอดีมาใช้ได้อย่างมีประสิทธิภาพที่สุด แต่หากทำการขยาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขอบเขตของระบบ เช่น การอนุญาตให้คน ใช้นำการคัดอาร์เอฟไอคืนได้ เพื่อจะได้มีความสะดวก ในการระบุอัตลักษณ์ในการเข้าถึงข้อมูลส่วนตัวเสี่ยงในครั้งต่อไปที่จะนำมารักษา หรือ จะเป็นการ ขยายขอบเขตของระบบโดยการนำระบบดังกล่าวนำมาใช้กับโรงพยาบาลที่มีหลากหลายสาขา เพื่อที่จะนำข้อมูลมีอยู่ในตัวแท็กอาร์เอฟไอคืน มาใช้ได้อย่างมีประสิทธิภาพสูงสุด

### 5.3.2 การประยุกต์เทคโนโลยีร่วมกับระบบดูแลรักษาส่วนตัว

การนำเทคโนโลยีพอกเก็ตพีซี มาใช้ร่วมกับ ระบบดูแลรักษาส่วนตัว โดยมีการแยกฐานข้อมูล ต่างหากอยู่ที่ Database Server จะทำให้เกิดข้อเสียในเรื่องความคล่องตัวคือหากจะทำการตรวจสอบ รักษาจะต้องขึ้นกับเครือข่ายภายในที่ทำการรักษา แต่หากนำฐานข้อมูลมาพัฒนาเพื่อใช้อยู่บนพอกเก็ตพีซีแต่ละเครื่องจะทำให้สามารถดึงข้อมูลมาใช้ได้อย่างอิสระโดยไม่ต้องพึ่งพาระบบเครือข่าย เพียงอย่างเดียว

### 5.3.3 การเพิ่มประสิทธิภาพของระบบ

เนื่องพอกเก็ตพีซี มีหน่วยความจำที่จำกัด จึงควรจะต้องพัฒนาการเขียน โปรแกรมโดย คำนึงถึงหน่วยความจำของพอกเก็ตพีซี ด้วย อีกทั้งเครื่องมือที่ช่วยแพทย์ในการเขียนหรือบันทึกการ รักษาส่วนตัวเสี่ยงควรมีความยืดหยุ่นมากกว่าที่เป็นอยู่จะได้รองรับความเร็วในการบันทึกข้อมูลส่วนตัวเสี่ยงในกรณีเร่งด่วน ได้อย่างรวดเร็วยิ่งขึ้น

## บรรณานุกรม

บริษัท ออมรอน อิเล็กทรอนิกส์ จำกัด 2004. เทคโนโลยี RFID. บริษัท ออมรอน อิเล็กทรอนิกส์ จำกัด  
มติโชติ สมานไทย. 2546. การเขียนโค้ด ASP.NET ฉบับสมบูรณ์. อินโฟเพส.

สมพร จิวรสกุล. 2545. คู่มือการติดตั้งและใช้งาน Microsoft SQL Server 2000 ฉบับสมบูรณ์.  
อินโฟเพส.

สุรสิทธิ์ ทิวประสพศักดิ์ และ นันทนี แขวงโสภกา. 2546. อินไซต์ Visual Basic .NET. โปรวิชั่น.

Chompunuch Buaphud. 2003. **Front office Operation System and Inventory System of Veterinary  
Hospital Multiple. Seminar : Bachelor**

Luc André Burdet. 2004. **RFID Multiple Access Methods. Seminar. : ETH Zürich.**

Microsoft. **Smart Client Developer Center. [Online] Available: <http://msdn.microsoft.com/smartclient/understanding/netcf/FAQ/default.aspx>. 2004.**

Microsoft. **Getting Started in .NET. [Online] Available: <http://www.microsoft.com/thailand/net/>. 2004.**

Paul Dickns. et al. 2002. **Professional ADO.NET with VB.NET. Birmingham: Wrox Press.**

Paul Yao and David Durant. 2004. **.NET Compact Framework Programming with Visual Basic NET.**  
Boston: Addison-Wesley.

## ประวัติผู้เขียน

ชื่อ-นามสกุล นายชัยศ วินิชกุล  
 วัน เดือน ปีเกิด 16 เมษายน 2524 ที่กรุงเทพมหานคร  
 ที่อยู่ 81/14 หมู่บ้าน ทิววัล ซอย 21/4 ถ.เทพารักษ์ ต.บางเมืองใหม่ อ.เมือง จ.  
 สมุทรปราการ 10270 โทร. 0-2758-1038

ประวัติการศึกษา

ปริญญาตรี

คณะบริหารธุรกิจการจัดการ สาขาเทคโนโลยีสารสนเทศธุรกิจ  
 มหาวิทยาลัย อัสสัมชัญ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้