



การแสดงผลตำแหน่งของนักศึกษาภายในห้องเรียนโดยใช้ RFID ผ่านทาง Website
(WEBSITE FOR DISPLAY STUDENT'S LOCATION IN CLASSROOMS USING RFID)



นายสันฐิติ พรหมากร

B5011742


นางสาวกมลทิพย์ ณรงค์ฤทธิ์

B5015894

รายงานนี้เป็นส่วนหนึ่งของการศึกษาวิชา 427499 โครงการวิศวกรรมโทรคมนาคม
หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม หลักสูตรปรับปรุง พ.ศ. 2545
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี
ประจำภาคการศึกษาที่ 3 ปีการศึกษา 2553

การแสดงผลตำแหน่งของนักศึกษาภายในห้องเรียนโดยใช้RFID บน WEBSITE

คณะกรรมการสอบโครงการ




(ผู้ช่วยศาสตราจารย์ ดร. รุ่งสรรค์ ทองทา)

กรรมการ/อาจารย์ที่ปรึกษาโครงการ



(ผู้ช่วยศาสตราจารย์ เรืออากาศเอก ดร. ประโยชน์ คำสวัสดิ์)

กรรมการ



(ผู้ช่วยศาสตราจารย์ ดร. ชานชัย ทองโสภ)

กรรมการ

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นำรายงานโครงการฉบับนี้ เป็นส่วนหนึ่งของการศึกษาระดับปริญญาตรี สาขาวิชาวิศวกรรมโทรคมนาคม วิชา 427499 โครงการวิศวกรรมโทรคมนาคม ประจำปีการศึกษา 2553

โครงการ การแสดงผลตำแหน่งของนักศึกษาภายในห้องเรียน โดยใช้ RFID ผ่านทางWebsite
(WEBSITE FOR DISPLAY STUDENT'S LOCATION IN CLASSROOMS USING RFID)

ผู้เสนอโครงการ	1. นายสันฐิติ พรหมากร	B5011742
	2. นางสาวกมลทิพย์ ณรงค์ฤทธิ์	B5015894
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. รังสรรค์ ทองทา	

บทคัดย่อ

ปัจจุบันเทคโนโลยีการสื่อสารที่สำคัญมากอีกอย่างหนึ่งที่ใช้ในชีวิตประจำวันคงจะหนีไม่พ้นอินเทอร์เน็ต ซึ่งเป็นระบบเครือข่ายที่ใช้อย่างกว้างขวางไปทั่วโลก ประโยชน์ของอินเทอร์เน็ตจะมีค่ามากหรือน้อยขึ้นอยู่กับลักษณะการใช้งานของแต่ละบุคคล เราจะนำประโยชน์ของมันมาประยุกต์ใช้งานกับเทคโนโลยี RFID เพื่อแสดงผลและระบุตำแหน่งของวัตถุหรือบุคคล ผ่านทาง Web Server ของระบบปฏิบัติการ Linux ซึ่งสามารถดูหรือติดตามตำแหน่งของวัตถุได้จากหน้าจอคอมพิวเตอร์ ทำให้ง่ายและสะดวกในการติดตาม โดยผู้จัดทำได้ใช้ภาษา PHP ในการเขียนโปรแกรมเพื่อติดต่อกับ RFID โดยจะมีการประมวลผลโดยรับค่ามาเก็บไว้ที่ MySQL Database Server ซึ่งเป็นฐานข้อมูลที่คอยเก็บค่าต่างๆ ก่อนที่จะนำมาแสดงทางหน้าเว็บไซต์ จากผลการทดสอบโครงการสามารถแสดงตำแหน่ง Tag บนเว็บไซต์ได้

กิตติกรรมประกาศ

โครงการ การแสดงผลตำแหน่งของนักศึกษาภายในห้องเรียนโดยใช้ RFID ผ่านทาง Website นี้สำเร็จด้วยดี ผู้จัดทำขอกราบขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร. รังสรรค์ ทองทา ซึ่งเป็นอาจารย์ที่ปรึกษาและผู้ควบคุมโครงการที่ให้แนวความคิด และคำปรึกษาแนะนำ ตลอดจนช่วยเหลือแก้ไขข้อบกพร่องและปัญหาต่าง ๆ ที่เกิดขึ้นอันเป็นประโยชน์ต่อโครงการนี้

ผู้จัดทำขอกราบขอบพระคุณต่อ คณาจารย์ในสาขาวิชาวิศวกรรมโทรคมนาคม ซึ่งกรุณาเป็นกรรมการสอบโครงการและให้คำปรึกษาแนะนำ รวมทั้งพี่ปัญญา นักศึกษาปริญญาโท และเพื่อนๆ นักศึกษาสาขาวิชาวิศวกรรมโทรคมนาคม ชั้นปีที่ 4 ที่ให้คำแนะนำต่าง ๆ และให้กำลังใจตลอดมา

ขอบคุณพี่ๆ บุคลากรฝ่ายตารางสอนตารางสอบ ที่อำนวยความสะดวกในการขอใช้ห้องเรียน อาคารเรียนรวม 1 ในการทำการทดลอง

ขอบคุณ Google ที่คอยตอบคำถาม แม้จะถามบ่อยแค่ไหนก็ไม่บ่น และท้ายสุดนี้ ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และญาติๆ ทุกท่าน ที่ให้ความช่วยเหลือด้านการเงินและเป็นกำลังใจให้การสนับสนุนทุกๆ ด้าน รวมทั้งเป็นกำลังใจที่สำคัญอย่างยิ่งให้เกิดโครงการชิ้นนี้ขึ้นมา

มหาวิทยาลัยเทคโนโลยีสุรนารี

ณ

ฐิติ พรหมากร

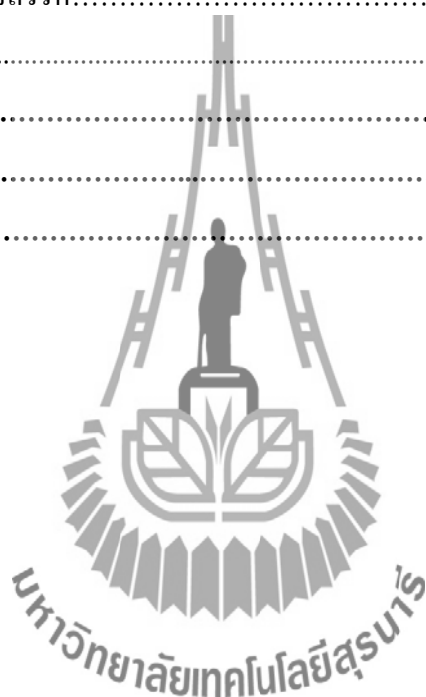
กมลทิพย์ ณรงค์ฤทธิ์

สารบัญ

เรื่อง	หน้า
บทคัดย่อ.....	ก
กิตติกรรมประกาศ.....	ข
สารบัญ.....	ค
สารบัญรูปภาพ.....	ง
บทที่ 1 บทนำ.....	1
1.1 หลักการและเหตุผล.....	1
1.2 วัตถุประสงค์.....	2
1.3 ขอบเขตงาน.....	2
1.4 ขั้นตอนการดำเนินงาน.....	3
1.5 ผลที่คาดว่าจะได้รับ.....	4
1.6 อุปกรณ์และซอฟต์แวร์ที่คาดว่าจะใช้.....	4
บทที่ 2 ทฤษฎีและโปรแกรมที่เกี่ยวข้อง.....	5
2.1 ความหมายของระบบ RFID.....	5
2.2 โครงสร้างของ RFID.....	6
2.3 ตัวอย่างการใช้งาน RFID.....	7
2.4 โปรแกรมการจัดการอาร์เอฟไอดี (Middleware หรือ Savant Software).....	8
2.5 ตัวอย่างการนำเทคโนโลยี RFID มาใช้ในต่างประเทศ.....	12
2.6 ตัวอย่างการนำเทคโนโลยี RFID มาใช้ในประเทศไทย.....	14
2.7 ทฤษฎีซอฟต์แวร์ที่เกี่ยวข้อง.....	16
2.8 phpMyAdmin.....	19
2.9 โปรแกรม File Zilla.....	40
2.10 รายละเอียดการจัดสรรพื้นที่เว็บไซต์.....	44
2.11 คุณสมบัติของอุปกรณ์ฮาร์ดแวร์ในโรงงาน.....	45
2.11.1 คุณสมบัติของบอร์ด RF2315R Active RFID Dual Receiver Module with RSSI USB Version.....	45

	2.11.2 คุณสมบัติของบอร์ด RF8315T Active RFID 8 Meters	
	Transmitting Module.....	47
เรื่อง		หน้า
บทที่ 3 การออกแบบโครงงาน		49
3.1 บทนำ.....		49
3.2 องค์ประกอบโดยรวมของระบบ		50
3.3 การออกแบบ Software.....		51
3.4 การทำงานบนฐานข้อมูล.....		52
3.5 การออกแบบโปรแกรม.....		53
3.5.1 การสร้างฐานข้อมูล โดยใช้ phpMyAdmin.....		53
3.5.2 สร้างตาราง ชื่อว่า table_name.....		54
3.5.3 สร้างตาราง ชื่อว่า table_location.....		56
3.5.4 การเขียนหน้าเว็บเพจ และไฟล์ติดต่อฐานข้อมูล.....		58
3.5.5 การ up load ไฟล์ขึ้น server และการ update ข้อมูล.....		59
3.5.6 การลงทะเบียนหรือการเก็บข้อมูลของผู้ใช้งานใหม่.....		61
3.5.7 ส่งชื่อ tag และตำแหน่ง (location) ไปยัง Server		63
3.5.8 แสดงตำแหน่งบนเว็บไซต์.....		64
บทที่ 4 การทดลอง.....		65
4.1 การใช้งานเครื่อง.....		65
4.1.1 การติดตั้งและใช้งานอุปกรณ์.....		65
4.1.1.1) RF8315T Active RFID 8 Meters Transmitting Module		65
4.1.1.2) RF2315R Active RFID Dual Receiver Module with RSSI....		67
4.1.2 การติดตั้ง RFID Dual Receiver ไว้ในห้องเรียน.....		69
4.2 การเลือก port.....		70
4.3 การตั้งค่า port ในโปรแกรม.....		72
4.4 การทดสอบใช้โครงงาน.....		73
4.5 ผลการทดสอบโครงงาน.....		74
4.6 สรุปผลการทดสอบ.....		81

เรื่อง	หน้า
บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ.....	82
5.1 สรุปผลการทดลอง.....	82
5.2 สิ่งที่ได้รับจากการทำโครงการ.....	82
5.3 ปัญหาและอุปสรรค.....	83
5.4 ข้อเสนอแนะ.....	83
ภาคผนวก	84
เอกสารอ้างอิง.....	96
ประวัติผู้เขียน.....	97



สารบัญภาพ

เรื่อง	หน้า
บทที่ 1 บทนำ.....	1
รูปที่ 1.1 ระบบ RFID.....	1
บทที่ 2 ทฤษฎีและโปรแกรมที่เกี่ยวข้อง.....	5
รูปที่ 2.1 แสดงโครงสร้างของอาร์เอฟไอดี (C.M. Roberts, 2006).....	6
รูปที่ 2.2 โครงสร้างและภาพรวมของอาร์เอฟไอดี.....	7
รูปที่ 2.3 แสดงภาพประยุกต์ใช้งาน RFID ในงานต่างๆ.....	8
รูปที่ 2.4 (ซ้าย) ผังวงจรทางกายภาพ (Layout) ของวงจรไมโครชิป RFID ตัวแรกของ ประเทศไทยที่เทคโนโลยี 0.8 um (ขวา) ภาพการ์ดที่นำไมโครชิปต่อกับ ขดลวดเพื่อไปใช้งานจริง.....	15
รูปที่ 2.5 รูปตัวอย่าง GUI ของโปรแกรมที่นำ RFID ไปประยุกต์ใช้งานต่างๆ.....	15
รูปที่ 2.6 บอร์ดเครื่องอ่าน RFID ยานความถี่ต่ำ.....	16
รูปที่ 2.7 โครงสร้างของ .NET Framework.....	18
รูปที่ 2.8 หน้าจอแรกของโปรแกรม phpMyAdmin.....	20
รูปที่ 2.9 สร้างฐานข้อมูลใหม่ชื่อ product.....	21
รูปที่ 2.10 การสร้างตารางใหม่.....	22
รูปที่ 2.11 การกำหนดรายละเอียดของฟิลด์ต่างๆภายในตาราง.....	27
รูปที่ 2.12 หน้าจอภายหลังสร้างตารางใหม่.....	28
รูปที่ 2.13 รายชื่อตารางภายในฐานข้อมูล คลิกเพื่อจัดการตารางนั้นๆ.....	28
รูปที่ 2.14 หน้าจอการจัดการตาราง ในส่วนการแสดงโครงสร้างตาราง.....	29
รูปที่ 2.15 แสดงเมนู เปิดดู.....	30
รูปที่ 2.16 ข้อมูลภายในตาราง ภายหลังคลิกเมนู เปิดดู.....	30
รูปที่ 2.17 การใช้คำสั่ง SQL เพื่อเปิดดูข้อมูลภายในตาราง.....	31
รูปที่ 2.18 หน้าจอการค้นหา.....	31
รูปที่ 2.19 การเพิ่มข้อมูลลงยังตาราง.....	32
รูปที่ 2.20 การลบข้อมูลภายในตาราง.....	32
รูปที่ 2.21 การแก้ไขข้อมูลในตาราง.....	33
รูปที่ 2.22 การแก้ไขข้อมูลในตาราง (ต่อ).....	33

เรื่อง	รูปที่ 2.23 การลบตารางออกจากฐานข้อมูล.....	34	หน้า
	รูปที่ 2.24 ยืนยันการลบตาราง.....	34	
	รูปที่ 2.25 เมนูส่งออก.....	35	
	รูปที่ 2.26 หน้าจอสำหรับตั้งค่าต่างๆ ของการ Export.....	35	
	รูปที่ 2.27 หน้าต่างสำหรับเลือกบันทึกไฟล์.....	36	
	รูปที่ 2.28 เลือก location ในเครื่องที่ต้องการเซฟไฟล์ลงไป.....	37	
	รูปที่ 2.29 เลือกไฟล์ Export ที่ต้องการ.....	37	
	รูปที่ 2.30 การ Import ไฟล์ข้อมูลลงยังฐานข้อมูล (ต่อ).....	38	
	รูปที่ 2.31 ขั้นตอนการเลือกเพื่อเพิ่มฟิลด์ใหม่.....	38	
	รูปที่ 2.32 การใส่รายละเอียดฟิลด์ใหม่.....	39	
	รูปที่ 2.33 ขั้นตอนการเลือกเพื่อลบฐานข้อมูล.....	39	
	รูปที่ 2.34 ยืนยันการลบฐานข้อมูล.....	39	
	รูปที่ 2.35 โปรแกรม มีไอคอนของ FileZilla ที่ Desktop.....	40	
รูปที่	2.36 การเปิดโปรแกรม FileZilla.....	41	
รูปที่	2.37 การสร้าง New Site.....	42	
	รูปที่ 2.38 การใส่ Host.....	42	
รูปที่	2.39 แสดงไฟล์.....	43	
รูปที่	2.40 การ Upload ไฟล์.....	43	
รูปที่	2.41 คลิกที่ปุ่ม Disconnect.....	44	
	รูปที่ 2.42 ภาพบอร์ด RF2315R Active RFID Dual Receiver Module with RSSI USB Version.....	45	
	รูปที่ 2.43 ภาพบอร์ด RF8315T Active RFID 8 Meters Transmitting Module.....	47	
	บทที่ 3 การออกแบบ การระบุตำแหน่งของ RFID บน Website.....	49	
รูปที่	3.1 การออกแบบฮาร์ดแวร์.....	50	
รูปที่	3.2 ฐานข้อมูล b5011742db.....	53	
	รูปที่ 3.3 โครงสร้างของตาราง table_name.....	54	
	รูปที่ 3.4 ตัวอย่างข้อมูลในตาราง table_name.....	55	
	รูปที่ 3.5 โครงสร้างของตาราง table_name.....	56	
	รูปที่ 3.6 ตัวอย่างข้อมูลในตาราง table_location.....	57	

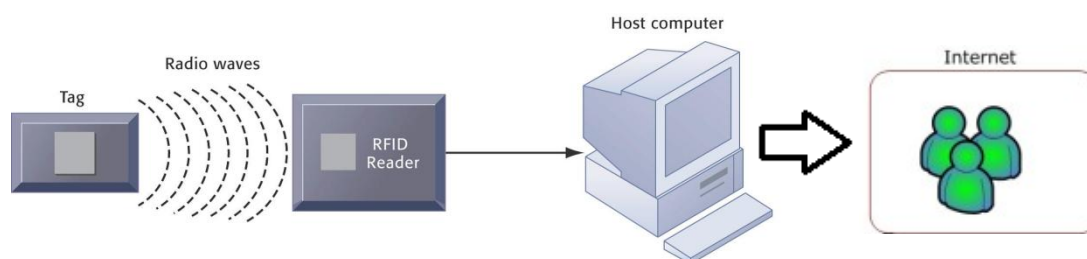
เรื่อง	รูปที่ 3.7 ไฟล์ที่มีการเขียนโปรแกรมไว้.....	58	หน้า
	รูปที่ 3.8 ระบบปฏิบัติการ Linux	59	
	รูปที่ 3.9 การตั้งค่าก่อนการเชื่อมต่อ.....	60	
รูปที่	3.10 การเชื่อมต่อ server ได้สำเร็จ	60	
รูปที่	3.11 การ upload.....	61	
รูปที่	3.12 ตัวอย่างข้อมูลในตาราง table_name.....	62	
รูปที่	3.13 การลงทะเบียนผ่านอินเทอร์เน็ต	62	
	รูปที่ 3.14 ตัวอย่างหน้าเว็บไซต์.....	64	
บทที่ 4	การใช้งานโครงงาน	65	
	รูปที่ 4.1 tag ก่อนการใช้งาน.....	65	
	รูปที่ 4.2 tag เมื่อมีการใส่ แบตเตอรี่.....	65	
	รูปที่ 4.3 tag พร้อมนำไปใช้งาน.....	66	
	รูปที่ 4.4 ตัวอย่างการวาง tag บนโต๊ะเรียน.....	67	
	รูปที่ 4.5 RFID Dual Receiver ก่อนการใช้งาน.....	67	
	รูปที่ 4.6 RFID Dual Receiver เมื่อมีการเชตค่าพร้อมใช้งาน.....	68	
	รูปที่ 4.7 USB HUB	68	
	รูปที่ 4.8 ตำแหน่งที่มีการติดตั้ง RFID Dual Receiver.....	69	
	รูปที่ 4.9 ตัวอย่างการติดตั้ง RFID Dual Receiver ในห้องเรียน.....	69	
	รูปที่ 4.10 ขั้นตอนการเลือก port (1).....	70	
	รูปที่ 4.11 ขั้นตอนการเลือก port (2).....	70	
	รูปที่ 4.12 ขั้นตอนการเลือก port (3).....	71	
	รูปที่ 4.13 ขั้นตอนการเลือก port (4).....	71	
	รูปที่ 4.14 ตำแหน่งที่มีการตั้งค่า port.....	72	
	รูปที่ 4.15 การอ่านค่าในโปรแกรม.....	75	
	รูปที่ 4.16 ข้อมูลที่เก็บอยู่ใน table_location.....	76	
	รูปที่ 4.17 ตำแหน่งที่แสดงบน Website.....	76	

บทที่ 1

บทนำ

1.1 หลักการและเหตุผล

การพัฒนาของเทคโนโลยี RFID เป็นเทคโนโลยีที่ติดต่อระหว่าง RFID กับ Tag ที่สามารถตอบสนองความต้องการของประชาชนในสังคมได้ครบถ้วน นอกจากการใช้ระบุข้อมูลของวัตถุแล้วยังสามารถนำมาประยุกต์ใช้งานด้านต่างๆ เพื่อให้สอดคล้องกับการดำเนินชีวิตในปัจจุบันที่ต้องการความถูกต้อง รวดเร็วและสะดวก เทคโนโลยี RFID ที่นำมาใช้เช็คอินในห้องเรียนนั้น ปัจจุบันมีเพียงแต่การเช็คอินแล้วมีการเข้าห้องหรือไม่ ทำให้มีข้อบกพร่องอยู่ในกรณีที่นักศึกษาไม่ได้อยู่ในห้องเรียน แต่ระบบยังแจ้งว่ามีอยู่ในห้องโดยนักศึกษาฝากบัตรมาเช็คอินแทนกัน จึงนำเอาเทคโนโลยีนี้มาพัฒนาโดยการ จับหาว่ามีวัตถุนั้นอยู่จริง โดยสิ่งที่เราได้ ทำต่อมานั้นจะเป็นการบอกถึงตำแหน่งของวัตถุนั้นๆ ว่าอยู่ที่จุดใด โดยอาศัยความแรงของสัญญาณที่ส่งไป แล้วนำมาประมวลผลเพื่อเก็บในส่วนของฐานข้อมูล ซึ่งเราจะดึงข้อมูลเหล่านี้ไปแสดงบนเว็บไซต์ เพื่อเป็นการระบุตำแหน่งของวัตถุ โดยในส่วนของเนื้อหาผู้จัดทำได้ทำเกี่ยวกับการเช็คอินเข้าห้องเรียน โดยที่นักศึกษามีเครื่องส่งซึ่งต้องพกติดตัว แล้วในห้องเรียนจะมีตัวรับสัญญาณที่สามารถอ่านความแรงของสัญญาณได้ แล้วทำการประมวลผลของตำแหน่งว่าอยู่ ณ จุดใดในห้อง ซึ่งสามารถตรวจสอบได้จากเว็บไซต์ผ่านระบบออนไลน์ ดังนั้นทางผู้จัดทำโครงการได้เล็งเห็นถึงประโยชน์ของ RFID ในด้านของการประยุกต์ใช้กับการบอกตำแหน่งของนักศึกษาในห้องเรียน และแสดงผลบนเว็บไซต์ ทำให้เกิดความสะดวกรวดเร็ว และง่าย ในการตรวจสอบ จึงได้จัดทำโครงการชิ้นนี้ขึ้น



รูปที่ 1.1 ระบบ RFID

1.2 วัตถุประสงค์

1. เพื่อประยุกต์ใช้งานเทคโนโลยี RFID ให้เป็นประโยชน์ต่อสาธารณชนมากขึ้น
2. เพื่อเป็นการศึกษาและพัฒนาทักษะในการใช้ เขียนโปรแกรมโดยภาษา PHP, HTML และใช้โปรแกรม phpMyadmin ในการควบคุมจัดการ MySQL Server
3. เพื่อศึกษาและนำระบบปฏิบัติ Linux ไปใช้งาน
4. เพื่ออำนวยความสะดวกต่ออาจารย์และผู้ปกครองในการเช็คชื่อเข้าห้องเรียนของนักศึกษา
5. เพื่อเป็นแนวทางสำหรับบุคคลที่สนใจในการศึกษาต่อไปในอนาคต

1.3 ขอบเขตงาน

1. ใช้งานในระบบปฏิบัติการ Linux
2. ใช้ภาษา PHP เพื่อเขียนโปรแกรมติดต่อฐานข้อมูลบน Web Server และใช้โปรแกรม phpMyadmin ในการควบคุมจัดการ MySQL Server
3. ทำการทดสอบระบบฐานข้อมูลเพื่อติดต่อกับอุปกรณ์ RFID เพื่อให้ได้ตามวัตถุประสงค์ที่วางไว้ และให้แสดงผลบนเว็บไซต์ได้



1.5 ผลที่คาดว่าจะได้รับ

1. สามารถนำมาเทคโนโลยี RFID มาประยุกต์ให้สามารถแสดงผลบนเว็บไซต์ได้
2. ได้พัฒนาทักษะในการเขียน โปรแกรมบนระบบปฏิบัติการ Linux เพื่อเขียน Web Server
3. ได้ศึกษาเพิ่มเติมเกี่ยวกับภาษาที่ใช้เขียนคือ PHP และ โปรแกรม MySQL Database Server

1.6 อุปกรณ์และซอฟต์แวร์ที่คาดว่าจะใช้

เครื่องมือและอุปกรณ์ที่ใช้ในการจัดทำประกอบด้วย

- 1) เครื่องคอมพิวเตอร์
- 2) ระบบปฏิบัติการ Linux (SUT)
- 3) Hosting (Web Server)
- 4) โปรแกรม File Zilla
- 5) RF8315T Active RFID 8 Meters Transmitting Module (Dimensions, 4 cm X 5 cm X 1.8 cm)
- 6) RF2315R Active RFID Dual Receiver Module with RSSI (Dimensions, 8cm X 6.cm)



บทที่ 2

ทฤษฎีและโปรแกรมที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงความหมายของระบบ RFID การประยุกต์ใช้งาน ตัวอย่างการทำเทคโนโลยี RFID มาใช้งาน และโปรแกรมที่ใช้ในการทำโครงการในครั้งนี้

2.1 ความหมายของระบบ RFID

RFID ย่อมาจากคำว่า Radio Frequency Identification เป็นระบบผลากที่ได้ถูกพัฒนามาตั้งแต่ปี ค.ศ. 1980 โดยที่อุปกรณ์ RFID ที่มีการประดิษฐ์ขึ้นใช้งานเป็นครั้งแรกนั้น เป็นผลงานของ Leon Theremin ซึ่งสร้างให้กับรัฐบาลของประเทศรัสเซียในปี ค.ศ. 1945 ซึ่งอุปกรณ์ที่สร้างขึ้นมาในเวลานั้นทำหน้าที่เป็นเครื่องมือตรวจจับสัญญาณ ไม่ได้ทำหน้าที่เป็นตัวระบุเอกลักษณ์อย่างใช้งานกันอยู่ในปัจจุบัน

RFID ในปัจจุบันมีลักษณะเป็นป้ายอิเล็กทรอนิกส์ (RFID Tag) ที่สามารถอ่านค่าได้โดยผ่านคลื่นวิทยุจากระยะห่าง เพื่อตรวจ ติดตามและบันทึกข้อมูลที่ติดอยู่กับป้าย ซึ่งนำไปฝังไว้หรือติดอยู่กับวัตถุต่างๆ เช่น ผลิตภัณฑ์ กล้อง หรือสิ่งของใดๆ สามารถติดตามข้อมูลของวัตถุ 1 ชิ้นว่าคืออะไร ผลิตที่ไหน ใครเป็นผู้ผลิต ผลิตอย่างไร ผลิตวันใด และเมื่อไร ประกอบไปด้วยชิ้นส่วนกี่ชิ้น และแต่ละชิ้นมาจากที่ไหน รวมทั้งตำแหน่งที่ตั้งของวัตถุนั้นๆ ว่าอยู่ส่วนใดของโลก โดยไม่จำเป็นต้องอาศัยการสัมผัส (Contact-Less) หรือต้องเห็นวัตถุนั้นๆก่อน ทำงานโดยใช้เครื่องอ่านที่สื่อสารกับป้ายด้วยคลื่นวิทยุในการอ่านและเขียนข้อมูล

RFID เป็นระบบระบุลักษณะของวัตถุด้วยคลื่นความถี่วิทยุที่ได้ถูกพัฒนามามีวัตถุประสงค์หลักเพื่อนำไปใช้งานแทนระบบบาร์โค้ด (Barcode) โดยจุดเด่นของ RFID อยู่ที่การอ่านข้อมูลจากแท็ก (Tag) ได้หลายๆ แท็กแบบไร้สัมผัสและสามารถอ่านค่าได้แม้ในสภาพที่ทัศนวิสัยไม่ดี ทนต่อความเปียกชื้น และแรงสั่นสะเทือน การกระทบกระแทก สามารถอ่านข้อมูลได้ด้วยความเร็วสูง โดยข้อมูลจะถูกเก็บไว้ในไมโครชิปที่อยู่ในแท็ก ในปัจจุบันได้มีการนำ RFID ไปประยุกต์ใช้งานในด้านอื่นๆ นอกเหนือจากนำมาใช้แทนระบบบาร์โค้ดแบบเดิม เช่น ใช้ในบัตรชนิดต่างๆ เช่น บัตรสำหรับใช้ผ่านเข้าออกสถานที่ บัตรจอดรถตามศูนย์การค้าต่างๆ ที่เราอาจพบเห็นอยู่ในรูปของแท็กสินค้ามีขนาดเล็กจนสามารถแทรกลงระหว่างชั้นของเนื้อกระดาษได้ หรือเป็นแคปซูลขนาดเล็กฝังเอาไว้ในตัวสัตว์ เพื่อบันทึกประวัติ เป็นต้น

อาร์เอฟไอดี (Radio Frequency Identification หรือ RFID) เป็นระบบที่อาศัยคลื่นแม่เหล็กไฟฟ้าในช่วงความถี่วิทยุในการติดต่อสื่อสาร เป็นระบบที่มีความใกล้เคียงกับระบบบาร์โค้ด แต่แตกต่างกันเช่น ในเรื่องของการอ่านข้อมูล ระบบบาร์โค้ดจะใช้แสงในการอ่านแถบ

บาร์โค้ดซึ่งติดอยู่ที่สินค้า โดยแสงจะต้องอ่านแถบที่ไม่มีอะไรปกปิด หรือต้องอยู่ในเส้นตรง เดียวกับลำแสงที่ยิงจากเครื่องสแกน และอ่านได้ที่ละแถบในระยะใกล้ๆ แต่ระบบอาร์เอฟไอดี สามารถอ่านแท็กซึ่งทำหน้าที่คล้ายแถบบาร์โค้ดได้โดยไม่ต้องเห็นแท็ก หรือแท็กนั้นสามารถอยู่ใน บริเวณที่สามารถรับคลื่นวิทยุได้ก็สามารถอ่านข้อมูลได้ และการอ่านแท็กในระบบอาร์เอฟไอดียังสามารถอ่านได้หลายๆแท็กในเวลาเดียวกัน โดยระยะในการอ่านข้อมูลก็ไกลกว่าระบบบาร์โค้ดด้วย ระบบอาร์เอฟไอดีถือเป็นเทคโนโลยีที่มีแนวโน้มที่จะนำมาใช้มากในอนาคตอันใกล้

เนื่องจากในปัจจุบันมีการแข่งขันที่รุนแรงในหลายอุตสาหกรรม ซึ่งนำไปสู่การที่บริษัทต่างๆ พยายามที่จะหากลยุทธ์หรือวิธีการใหม่ๆมาใช้เพื่อความได้เปรียบเชิงการแข่งขันเพื่อนำไปสู่ ตำแหน่งผู้นำในอุตสาหกรรม การนำเทคโนโลยีใหม่ๆเข้ามาใช้ในอุตสาหกรรมเพื่อปรับปรุง กระบวนการ หรือผลิตภัณฑ์ต่างๆ ให้มีประสิทธิภาพหรือคุณภาพสูงขึ้น ก็เป็นอีกวิธีหนึ่งซึ่ง สามารถสร้างความได้เปรียบเชิงการแข่งขันได้

2.2 โครงสร้างของ RFID

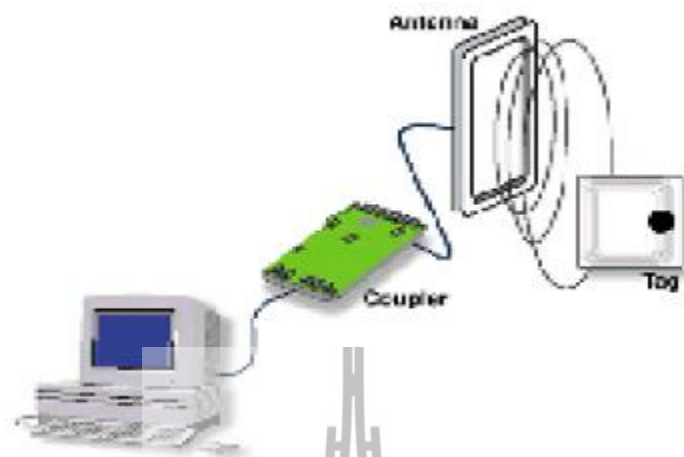
โครงสร้างอาร์เอฟไอดี ประกอบด้วยส่วนหลักๆ 3 ส่วนด้วยกัน คือ

- 1) แท็กอาร์เอฟไอดี (RFID Tag/Transponder)
- 2) เครื่องอ่านแท็กอาร์เอฟไอดี (RFID Tag Reader/Emitter/Writer)
- 3) โปรแกรมจัดการอาร์เอฟไอดี (Middleware หรือ Savant Software)



รูปที่ 2.1 แสดงโครงสร้างของอาร์เอฟไอดี (C.M. Roberts, 2006)

ภาพแสดงการทำงาน



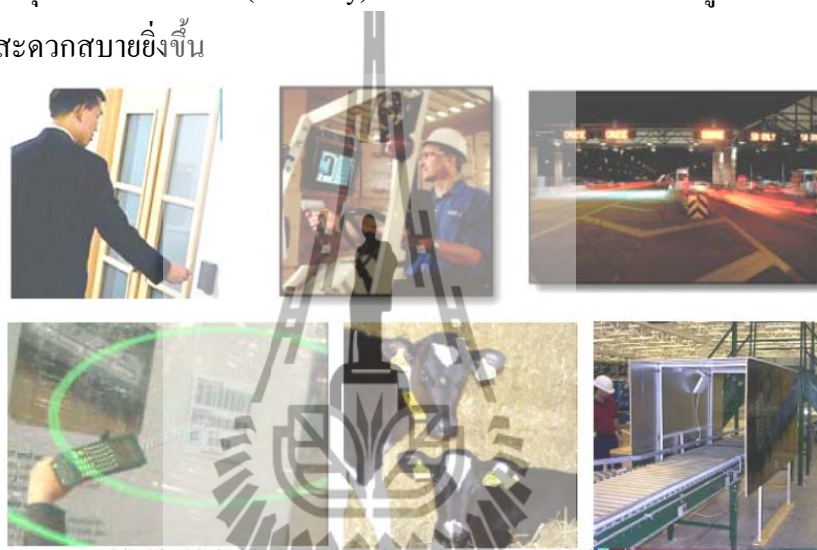
รูปที่ 2.2 โครงสร้างและภาพรวมของอาร์เอฟไอดี

2.3 ตัวอย่างการใช้งาน RFID

- ปัจจุบันการนำระบบ RFID มาประยุกต์ใช้งานหลากหลายประเภท เช่น
- ทดแทนระบบบาร์โค้ด (Barcode) รุ่นเก่า
 - Access Control/ Personal Identification หรือการเข้า-ออกอาคาร แทนการใช้บัตรแม่เหล็ก เมื่อใช้งานมากๆ ก็จะเสื่อมเร็ว แต่บัตรแบบ RFID (Proximity Card 1) ใช้เพียงแตะหรือแสดงผ่านหน้าเครื่องอ่านเท่านั้น รวมทั้งยังสามารถใช้กับการเช็คเวลาเข้า-ออกงานของพนักงานด้วย
 - ห่วงโซ่อุปทานและระบบลอจิสติก ภาพที่จะเห็นในโรงงานอนาคต คือ สามารถติด Tag ไว้กับชิ้นงาน เมื่อชิ้นงานผ่านสายพานขนส่งในโรงงานแต่ละแผนกจะรู้ว่าต้องทำอะไร ตัดอะไรบ้าง และต้องส่งไปที่ไหนต่อ รวมถึงการจัดการสินค้าในคลังสินค้าว่ารับสินค้ามาเมื่อใด จะต้องเก็บไว้ที่ไหน จะส่งไปที่ไหนอย่างไร ใครจะมารับ ส่วนภาพที่ผู้บริโภคจะเห็นคือ การซื้อสินค้าในซูเปอร์มาร์เก็ต เวลาซื้อก็หยิบใส่ตะกร้า คิดเงินผ่านเครื่องอ่าน RFID ครั้งเดียวคิดเงินได้ทันที ไม่ต้องหยิบมายิงบาร์โค้ดทีละชิ้นให้เสียเวลาและเดือนผู้ซื้อได้หากสินค้าที่ซื้อหมดอายุ
 - ระบบ Animal Tracking มาใช้เหมาะกับเกษตรกรไทยในการพัฒนาด้านปศุสัตว์ให้เป็นระบบฟาร์มอัตโนมัติด้วยชิป RFID ติดตัวสัตว์เลี้ยงทำให้สามารถทราบเจ้าของ ตรวจสอบสายพันธุ์ การให้อาหารและการควบคุมโรคติดต่อในสัตว์ รวมถึงการสร้าง Food Traceability สำหรับ

ต่อสู้กับข้อกีดกันทางการค้าของสหรัฐอเมริกาและกลุ่มสหภาพยุโรปที่อยู่ระหว่างตัดสินใจว่าผู้ส่งออกสินค้าเนื้อสัตว์ชำแหละ

- ระบบตั๋วอิเล็กทรอนิกส์ (e-ticket) เช่น บัตรทางด่วน บัตรรถไฟฟ้าใต้ดิน
- ระบบหนังสือเดินทางอิเล็กทรอนิกส์ (e-Passport) ที่ทางประเทศสหรัฐอเมริกากำหนดมาตรฐานการเข้าออกของประเทศ เพื่อป้องกันผู้ก่อการร้าย รวมไปถึง e-Citizen ด้วย
- ระบบกุญแจอิเล็กทรอนิกส์ (immobilizer) ในรถยนต์เพื่อป้องกันกุญแจผิดในการขโมยรถยนต์ หรือ พวง Keyless ในรถยนต์ราคาแพงบางรุ่นก็เริ่มนำมาใช้งานแล้ว
- ระบบห้องสมุดดิจิทัล (e-Library) ในการยืมคืนอัตโนมัติทำให้ผู้ใช้บริการได้รับความรวดเร็วและสะดวกสบายยิ่งขึ้น



รูปที่ 2.3 แสดงภาพประยุกต์ใช้งาน RFID ในงานต่างๆ

2.4 โปรแกรมการจัดการอาร์เอฟไอดี (Middleware หรือ Savant Software)

เครื่องอ่านแท็ก RFID จะส่งสัญญาณภาษาเดียวกันไปยังแท็ก RFID ถ้าเป็นแท็กที่สามารถสื่อสารกับเครื่องอ่านได้ แท็กจะส่งข้อมูลกลับไปยังเครื่องอ่าน โดยเครื่องอ่านจะเก็บข้อมูลแล้วแปลงสัญญาณที่ได้เป็นภาษาที่ใช้ในซอฟต์แวร์ด้วย Middleware หรือ Savant Software ไปยังระบบซอฟต์แวร์ที่ใช้ เช่น ซอฟต์แวร์ทางด้านการจัดการคลังสินค้าคงคลัง

ธนศ (2547) อธิบายความหมายของอาร์เอฟไอดีว่า เป็นเทคโนโลยีในการติดตามข้อมูลของวัตถุ 1 ชิ้น คืออะไร ผลิตที่ใด และใครเป็นผู้ผลิต ผลิตอย่างไร ผลิตวันไหน และเวลาอะไร ประกอบไปด้วยชิ้นส่วนกี่ชิ้น แต่ละชิ้นมาจากไหน รวมทั้งตำแหน่งที่ตั้งในปัจจุบันว่าอยู่ที่ส่วนใดในโลก

เมธินี (2547) อธิบายความหมายของอาร์เอฟไอดีว่า เป็นเทคโนโลยีการใช้ในการระบุคุณลักษณะและ/หรือติดตามที่ตั้งของสิ่งของ สิ่งของนี้จะถูกติดด้วยแท็กชนิดพิเศษ ซึ่งสามารถส่งคลื่นวิทยุไปยังเครื่องอ่าน เมื่อแท็กสื่อสารกับเครื่องอ่านข้อมูลที่ถูเก็บอยู่ในแท็กจะถูกส่งไปแสดงผลยังเครื่องอ่านทำให้ทราบคุณลักษณะและ/หรือตำแหน่งที่ตั้งของสิ่งของนั้นได้

ดร.กานาย (2547) อธิบายความหมายของอาร์เอฟไอดีว่า เป็นป้ายที่ฝังไมโครชิปชนิดที่ไม่จำเป็นต้องใช้พลังงาน โดยเชื่อมสัญญาณด้วยสายอากาศโดยไม่ต้องใช้สาย เมื่อแปลสัญญาณจากป้ายโดยผ่านเครื่องอ่านที่หน่วยงานเป็นสารสนเทศที่ต้องการ ซึ่งทำได้รวดเร็วผ่านคลื่นวิทยุ (RF) หลังจากนั้นจะทำการส่งต่อเข้าสู่ระบบประมวลผลต่อไป ป้าย RFID เรียกว่าป้ายบ่งชี้อัตโนมัติ (Auto-ID) ซึ่งป้ายนี้ถูกแปลรหัสเป็นรหัสอิเล็กทรอนิกส์ของสินค้า ป้าย Auto-ID สามารถบรรจุข้อมูลสูงถึง 96 บิต เพื่อให้มองภาพเข้าใจมากขึ้น 23 บิต สามารถบรรจุข้อมูลระดับโลก 33 บิต สามารถใช้สำหรับคนได้ทั่วโลก 54 บิต สามารถเก็บได้ทุกเม็ดของข้าวสาร

สมวงศ์ (2547) และเว็บไซต์ www.members.surfbest.net ซึ่งระบุว่า RFID เป็นเทคโนโลยีที่มีการค้นพบและนำมาใช้งานกันมานานกว่า 50 ปี ตั้งแต่สมัยสงครามโลกครั้งที่ 2 ที่ประเทศอังกฤษ โดยเอามาใช้เพื่อติดตามข้อมูลของเครื่องบินฝ่ายสัมพันธมิตรในสงครามเพราะเกรงว่าจะยิงพวกเดียวกัน แต่ก็ยังมีการใช้งานในวงแคบเนื่องจากมีขนาดเทอะทะและราคาแพง

ในช่วงศตวรรษที่ 60 ถึงต้นศตวรรษที่ 70 มีความจำเป็นในระบบของการรักษาความปลอดภัย เนื่องจากมีการใช้นิวเคลียร์ ซึ่งส่งผลให้มีการพัฒนา RFID เพื่อใช้ในการติดอุปกรณ์ ต่อมาในปี 1977 เทคโนโลยีนี้ถูกพัฒนาในห้องแล็บ Los Alamos Scientific Laboratories ซึ่งเป็นห้องแล็บของรัฐบาลอังกฤษเพื่อการใช้งานในภาครัฐบาล

ต่อมาเทคโนโลยีของชิปที่ใช้งานในเครื่องคอมพิวเตอร์มีการพัฒนาการไปอย่างรวดเร็ว ทำให้ชิปมีขนาดเล็กลงแต่ให้ประสิทธิภาพสูงมากขึ้นแต่มีราคาถูกลง ส่งผลให้ RFID ซึ่งมีชิปเป็นองค์ประกอบมีพัฒนาการไปในทิศทางเดียวกันตามไปด้วย ปัจจุบัน RFID สามารถย่อส่วนให้มีขนาดเล็กลงเหลือเท่ากับเมล็ดข้าวสารแล้วถูกบรรจุอยู่ในผลึกแก้วจนแทบมองไม่เห็น จนถึงปัจจุบันมีการจดทะเบียนสิทธิบัตรต่างๆ มากกว่า 350 ฉบับ เกี่ยวกับการนำระบบ RFID เข้ามาใช้ในประเทศสหรัฐอเมริกา

เว็บไซต์ของศูนย์ออโต้ไอดี www.autoidcenter.org ให้คำนิยามของอาร์เอฟไอดี (Radio Frequency Identification หรือ RFID) ว่าเป็นเทคโนโลยีซึ่งใช้คลื่นวิทยุเพื่อบ่งชี้ตัวตนของวัตถุแต่ละชิ้น ซึ่งการบ่งชี้วัตถุมีหลายวิธีการ แต่โดยส่วนมากจะใช้วิธีการเก็บชุดตัวเลขในไมโครชิป (Microchip) ซึ่งติดอยู่กับสายอากาศ (Antenna) โดยสองส่วนนี้รวมกันเรียกว่าแท็ก (Tag) โดย

สายอากาศจะเป็นตัวทำให้ชิปส่งข้อมูลที่บ่งชี้ไปยังเครื่องอ่าน (Reader) และเครื่องอ่านจะแปลงคลื่นวิทยุที่ได้รับจากแท็กไปอยู่ในรูปแบบที่สามารถส่งไปยังคอมพิวเตอร์และสามารถใช้งานได้ เว็บไซต์ของ Laran RFID www.laranrfid.com (2547) ให้คำนิยามของอาร์เอฟไอดีว่าเป็น เทคโนโลยีในการบ่งชี้คน หรือวัตถุ โดยใช้การส่งคลื่นความถี่ และสามารถใช้ในการติดตามแยกประเภท หรือการสืบค้นวัตถุ โดยการสื่อสารเกิดขึ้นระหว่างเครื่องอ่านและแท็ก ซึ่งแท็ก ประกอบด้วยสายอากาศและชิป



การนำเสนอ Case Study ที่น่าสนใจของบริษัทที่มีลักษณะเดียวกันกับ บริษัท ไปรษณีย์ไทย จำกัด และประสบความสำเร็จในการประยุกต์ระบบ RFID เข้ามาใช้

- Steven Deare ; ZDNet Australia 18 October 2006 : RFID in Australia Post Source : <http://www.zdnet.com.au/news/communications/soa/Aussie-Post-keeps-track-with-RFID/0,13006179,339271738,00.htm>

ได้กล่าวถึงการเปลี่ยนแปลงของระบบไปรษณีย์ภัณฑ์ของประเทศออสเตรเลีย ที่เล็งเห็นความสำคัญของระบบการตรวจสอบติดตามจดหมายด้วย RFID ที่ชาวโลกให้ความยอมรับ ซึ่งหลังจากที่ได้นำระบบ RFID เข้ามาใช้กับไปรษณีย์ออสเตรเลียแล้ว จากรายงานผลกำไรปีที่ผ่านมามีกำไรก่อนหักภาษีถึง 515.6 ล้านดอลลาร์ออสเตรเลีย

- Dr.Peter , March 2007 Harrop : RFID for Postal and Courier Service 2007-2017 Source : http://www.electronics.ca/reports/rfid/postal_courier.html กล่าวว่าการเจริญเติบโตของระบบ RFID จะมีการเจริญเติบโตอย่างมากระหว่างปี 2007-2017 ในตลาดของผู้ให้บริการไปรษณีย์ภัณฑ์ จะมีการเจริญเติบโตของการลงทุนในการใช้อุปกรณ์สำหรับ RFID รวมทั้ง Tag ประมาณ 3 ล้านเหรียญสหรัฐในปี 2016 คาดว่าการเจริญเติบโตของวงการไปรษณีย์และผู้ให้บริการด้านไปรษณีย์ภัณฑ์ในด้านการนำระบบ RFID เข้ามาพัฒนาเพื่อเพิ่มศักยภาพให้กับหน่วยงานหรือองค์กร จะเป็นการเจริญเติบโตของเทคโนโลยี RFID อันดับที่สองรองมาจากธุรกิจ Retail Supply Chain ซึ่งได้มีการนำระบบ RFID มาใช้ก่อนหน้านี้แล้ว

- วอล-มาร์ท (Wall-mart) เป็นผู้ค้าปลีกรายแรกที่ประกาศใช้ RFID ของโลกปัจจุบัน วอล-มาร์ทมีศูนย์กระจายสินค้า (Distribution Center) จำนวน 108 แห่ง มียอดขายปีละ 256,000 ล้านดอลลาร์สหรัฐ รัชสิรินทร์ (2547) ระบุว่าความแม่นยำในการบริหารสินค้าคงคลังของวอล-มาร์ทสูงถึง 99 เปอร์เซ็นต์ แต่อีก 1 เปอร์เซ็นต์ที่ขาดไปคิดเป็นจำนวนเงินถึงสองพันกว่าล้านเหรียญสหรัฐ ซึ่งจากข้อดีต่างๆ ของเทคโนโลยี RFID ทำให้วอล-มาร์ทพยายามที่จะนำ RFID มาใช้ในธุรกิจการค้าปลีก จากวารสาร Retail Technology Quarterly (2547) ระบุว่าวอล-มาร์ทประกาศว่าภายในเดือนมกราคม ปีพ.ศ. 2549 ซัพพลายเออร์ที่มียอดขายสูงสุด 100 อันดับแรก เช่น ฮิวเลตต์ แพ็คการ์ด (Hewlett-Packard) ยิลเลตต์ (Gilliette) จอห์นสัน (John & Johnson) คิมเบอร์ลี-คลา (Kimberly-Clark) คราฟท์ฟู้ด (Krafts Foods) พร็อกเตอร์ แอนด์ แกมเบิล (Procter & Gamble) และ ยูนิลีเวอร์ (Unilever) เป็นต้น จะต้องใช้ RFID ติดแท็ก RFID ลงบนพาเลทและกล่องบรรจุสินค้าที่จะส่งมาให้ทางห้าง ส่วนซัพพลายเออร์รายอื่นๆ จะต้องติดแท็ก RFID ในรถส่งสินค้าให้เสร็จสิ้นภายในปี 2549 และวอล-มาร์ทยังประกาศว่าจะใช้เทคโนโลยี RFID กับศูนย์กระจายสินค้าของวอล-มาร์ท 13 แห่ง ร้านค้าของวอล-มาร์ท 600 แห่ง รวมถึงแซมคลับ (Sam's Club) ภายในเดือนตุลาคมปี พ.ศ. 2549

- Johnchul Song , Carl T. Haas , Carlos Caldas , Esin Ergen and Burcu Akinci (2006) ได้ทำการศึกษาการติดตามกระบวนการขนส่งและรับสินค้า ของบริษัทผลิตท่อสำหรับใช้ในอุตสาหกรรมต่างๆ ในการศึกษาแบ่งเป็น 2 ระยะ คือ 1) ระยะที่ทำการติดตั้งระบบ RFID โดยใช้ Handheld Reader สำหรับกระบวนการ Load สินค้า และ Fixed Reader (4 Antenna) สำหรับติดตั้งที่ประตูทางเข้า-ออก 2) ทำการติดตั้ง Fixed Reader (4 Antenna) เพิ่มที่ประตูทางเข้าอื่นๆ เพื่อติดตามการส่งและรับท่อ ทั้งนี้เพื่อขยายผลการใช้งานจากระยะแรกอย่างเต็มรูปแบบด้วย ผลที่ได้จากการศึกษาในครั้งนี้แสดงให้เห็นว่า RFID สามารถช่วยให้กระบวนการส่งและรับสินค้ามีความถูกต้องแม่นยำถึง 100 % โดยมีข้อจำกัดว่าพนักงานขับรถต้องขับโดยผ่าน Fixed Reader และสายอากาศ 4 ต้นด้วยความเร็วไม่เกิน 2 mph

- Ning Wang, Naiquian and Maohua Wang (2006) ได้ศึกษาการนำระบบ Wireless Sensors มาใช้ในอุตสาหกรรมเกษตรและอาหาร โดยศึกษาการพัฒนาในปัจจุบันและมุมมองในอนาคต และเนื่องจากในปัจจุบันอุตสาหกรรมอาหารได้มีความต้องการในเรื่องของความปลอดภัยในการผลิต ความครบถ้วนของข้อมูลในการผลิตอาหารตั้งแต่จากแหล่งเพาะปลูกจนถึงมือลูกค้า เป็นต้น จึงได้มีการทดลองนำระบบ RFID มาใช้เพื่อการติดตามในกระบวนการต่างๆ เช่น 1) การระบุตัวสัตว์และการตรวจคุณภาพของสัตว์แต่ละตัว 2) การบรรจุอาหาร 3) การขนส่ง 4) การตรวจสอบคุณภาพอาหาร โดย RFID สามารถทำหน้าที่ในการติดตามในกระบวนการต่างๆ ได้อย่าง

มีประสิทธิภาพ และหากนำมาเปรียบเทียบกับ Barcode แล้ว RFID สามารถติด Tag ได้ทุกตำแหน่ง และสามารถอ่านข้อมูลได้โดยไม่ต้องสัมผัสกับเครื่องอ่านโดยตรง นอกจากนั้น Tag ยังสามารถ Update ข้อมูลได้ตลอดทั้งกระบวนการของ Supply Chain เช่น ข้อมูลของการเจริญเติบโตของตัวสัตว์ การผลิต การบรรจุ การขนส่ง การกระจายสินค้า การจัดเก็บ การจัดขึ้นชั้นวางสินค้า และการนำกลับมาใช้อีก ดังนั้นจึงสรุปได้ว่า เมื่อนำ RFID มาทำงานร่วมกับ Wireless Sensors ทำให้สามารถเก็บและรายงานข้อมูลในกระบวนการต่างๆ รวมถึงตรวจสอบคุณภาพและความปลอดภัยของการผลิตอาหารได้ตลอดทั้งกระบวนการ

- The RFID Tag Project : Trace of Rolling Cages Post Denmark A/S
- RFID in Korea Post
- RFID in Telematics Cooperative User Group Advanced Electronics Service
- Improvement of Postal Delivery Service, in case Korea Post
- New Participant For Japan-China-Korea Supply Chain RFID Pilot Project

(<http://www.chinarfidnews.com/index.php/2006/02/08/new-participant-for-japan-china-korea-supply-chain-rfid-pilot-project/>)

การประยุกต์ใช้งาน RFID

การนำเทคโนโลยี RFID เข้ามาประยุกต์ใช้ใน โลจิสติกส์ สามารถทำได้มากมาย แต่ตัวอย่างที่ชัดเจนและมีการนำไปใช้กันอย่างแพร่หลายที่สุดก็คงหนีไม่พ้น ห่วงโซ่อุปทานและระบบลอจิสติกส์ด้วยเทคโนโลยี RFID ที่คิดไว้ในผลิตภัณฑ์ จะช่วยทำให้ผลิตภัณฑ์นั้นๆ สื่อสารระหว่างกันได้ และยังสามารถสื่อสารไปยังหน่วยธุรกิจและผู้บริโภค ได้เช่นกัน ซึ่งจะเป็นการเปลี่ยนแปลงครั้งใหญ่ในกระบวนการผลิต การขาย และการจับจ่ายซื้อสินค้า

2.5 ตัวอย่างการนำเทคโนโลยี RFID มาใช้ในต่างประเทศ

- Wall Mart ร้านค้าปลีกชื่อดังของสหรัฐฯ ซึ่งมียอดขายปีละกว่า 250,000 ล้านดอลลาร์สหรัฐฯ ได้ออกระเบียบกำหนดให้ Suppliers รายใหญ่ 100 ราย เช่น Gillette, Nestle, Johnsons & Johnsons และ Kimberly Clark ติด RFID Chip บนหีบห่อ และกล่องบรรจุสินค้าให้เรียบร้อยก่อนส่งมาถึงห้าง ส่วน Suppliers รายเล็กๆ จะต้องติดชิปในรถส่งสินค้าให้แล้วเสร็จภายในสิ้นปี 2549 Wall Mart มองว่า เมื่อระบบดังกล่าวเสร็จสิ้นอย่างสมบูรณ์จะช่วยให้บริษัททราบถึงการเดินทางของสินค้าได้ทุกระยะ ตั้งแต่โรงงานของ Suppliers จนถึงศูนย์กระจายสินค้าของห้าง และเมื่อใดที่สินค้าถูกหยิบออกจากชั้นไป RFID ก็ส่งสัญญาณเตือนไปยังพนักงานให้นำสินค้ามาเติมใหม่ ทำ

ให้ Wall Mart ไม่จำเป็นต้องเก็บสต็อกสินค้า แต่สามารถสั่งให้ Suppliers มาส่งของได้ทันทีรวมทั้งช่วย Guarantee ว่าสินค้ามีวางจำหน่ายตลอดเวลา และประโยชน์ที่สำคัญอีกประการหนึ่ง ก็คือ จะช่วยลดปัญหาการโจรกรรมสินค้า และปลอมแปลงสินค้าได้อีกด้วย

- Extra Future Store ซึ่งเป็น Supermarket ในเยอรมนี ก็ได้นำเทคโนโลยี RFID มาใช้งานแล้ว หากลูกค้าต้องการซื้อชีส ลูกค้าก็เพียงป้อนคำสั่งลงในหน้าจอรระบบสัมผัสที่อยู่หน้ารถเข็น จากนั้นหน้าจอก็จะปรากฏแผนที่บอกทางไปสู่ชั้นวางชีส ทั้งนี้ที่ลูกค้าหยิบชีสจากชั้นวาง ชิปที่ติดอยู่บนห่อชีสก็จะส่งสัญญาณข้อมูลไปยังแผ่นเก็บข้อมูลหนา 2 มิลลิเมตรที่อยู่ใต้ชั้นวาง และอุปกรณ์ตรวจจับที่อยู่บนแผ่นดังกล่าวก็จะส่งสัญญาณแจ้งไปยังฐานข้อมูลของคลังสินค้าว่า ชีสห่อนั้นถูกหยิบออกจากชั้น ไปแล้ว ขณะเดียวกันข้อมูลดังกล่าวก็จะถูกส่งต่อไปยังบริษัทผู้ผลิตชีสด้วย และเมื่อข้อมูลพฤติกรรมของผู้บริโภคถูกเก็บรวบรวมไว้มากพอสมควรจนสามารถกำหนดเป็นพฤติกรรมผู้บริโภคได้แล้ว บริษัทผู้ผลิตและร้านค้าก็สามารถนำข้อมูลเหล่านั้นไปใช้ในการวางแผนการตลาดที่เหมาะสมและสามารถตอบสนองความต้องการของผู้บริโภคได้มากขึ้น

- ห้าง PRADA ที่อยู่กลางกรุงนิวยอร์ก ก็ได้ทดลองนำชิปไปติดไว้กับเสื้อผ้า เมื่อใดที่ลูกค้าหยิบชุดขึ้นมา และถือไว้ใกล้ๆ กับ RFID Reader จอภาพก็จะปรากฏภาพนางแบบที่สวมชุดนั้นอยู่ เพื่อให้ลูกค้าดูเป็นตัวอย่างอีกด้วย

- ห้างสรรพสินค้าชื่อดังในประเทศญี่ปุ่น “มัตสึโกชิ” ได้นำเทคโนโลยี VoIP กับ RFID มาทดลองให้บริการแก่ลูกค้าเพื่อใช้สื่อสารระหว่างลูกค้ากับพนักงานห้างในห้องทดลองเสื้อผ้าเพื่อตอบข้อซักถามของลูกค้า อาทิ สต็อก สินค้า ขนาด แบบ และสีสินค้าที่วางจำหน่ายโดยไม่ต้องเสียเวลาเดินเข้าเดินออกในห้องทดลองใส่เสื้อผ้า

- ห้าง TESCO ได้เริ่มนำ RFID Tag มาใช้กับสินค้าประเภทที่มีโชอาหาร ณ ศูนย์กระจายสินค้าในสหราชอาณาจักรแล้ว

- Metro Group ซึ่งเป็นผู้ค้าส่งขนาดใหญ่ที่ให้บริการกว่า 2,300 แห่ง กำหนดให้ Suppliers รายใหญ่ๆ กว่า 300 ราย ต้องติด RFID Tag

- Mark & Spencer ร้านค้าชั้นนำของอังกฤษกำลังทดลองติดตั้งชิปลงในชุดสูทผู้ชาย เมื่อลูกค้าซื้อสูทตัวใดขนาดใด สัญญาณข้อมูลจะถูกส่งไปยังห้องเก็บสต็อกสินค้าให้นำสูทตัวใหม่เข้ามาเติม

- บริษัทบริหารจัดการขยะในญี่ปุ่นแห่งหนึ่ง กำลังศึกษาความเป็นไปได้ในการนำเทคโนโลยี RFID มาใช้ในการจัดการขยะทางการแพทย์ ซึ่งเป็นขยะอันตรายจึงต้องระมัดระวังในเรื่องการกำจัด รวมถึงการร่วมมือกับโรงพยาบาลและบริษัทรับขนส่ง เพื่อพัฒนาระบบการติดตามขยะทางการแพทย์

- โรงเรียนแห่งหนึ่งในเมืองโอซากา ประเทศญี่ปุ่น นำชิปติดไว้กับกระเป๋านักเรียนป้ายชื่อ หรือเสื้อนักเรียนเด็กชั้นประถมศึกษา เพื่อป้องกันเด็กหาย หรือถูกลักพาตัว โดยโรงเรียนจะติดตั้ง RFID Reader ไว้ที่ประตูทางเข้าโรงเรียน และสถานที่สำคัญต่างๆ
- The United States of Food and Drug Administration (USFDA) พบว่าปัจจุบัน โรงพยาบาลบางแห่งในสหรัฐฯ ได้ฝัง RFID Tag ไว้ใต้ผิวหนังบริเวณท่อนแขน ตรงส่วนกล้ามเนื้อ Triceps ของคนไข้ เพื่อความสะดวกในการตรวจรักษาและติดตามข้อมูลการรักษาของผู้ป่วย เมื่ออวัยวะที่ได้รับการฝังชิปไว้ภายในถูกสแกนด้วย RFID Reader ระบบจะแสดงข้อมูลการรักษาของคนไข้รายนั้นออกมา ทำให้แพทย์ที่ถูกเปลี่ยนให้ดูแลรักษาคนไข้รายดังกล่าวได้รับทราบประวัติการรักษาโดยแพทย์คนก่อนหน้านั้น ได้อย่างถูกต้อง การฝังชิปลงไปใต้ผิวหนังก็ไม่ได้ยุ่งยากมากนัก เพียงแค่บรรจชิปลงในหยอดนึคยา แล้วฉีดลงไปซึ่งชิปจะถูกเคลือบด้วยสารที่ชื่อว่า Biobond ช่วยในการยึดเกาะกับเนื้อเยื่อภายในร่างกาย และช่วยป้องกันไม่ให้ชิปเสียหายด้วย

2.6 ตัวอย่างการนำเทคโนโลยี RFID มาใช้ในประเทศไทย

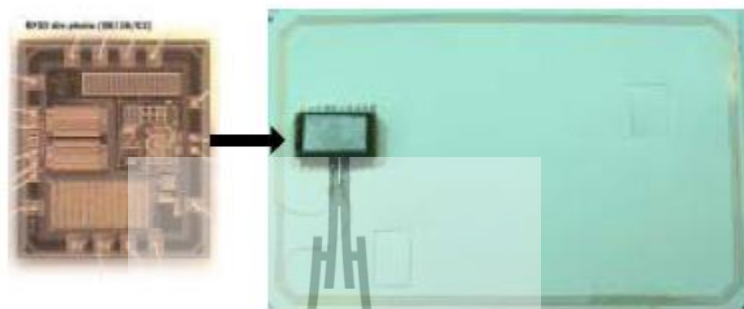
- การขนส่ง (บัตรทางด่วน, บัตรโดยสารรถไฟฟ้า)
- ห้องสมุดฉลาด (Intelligent check-in and check out)
 - การควบคุมการเข้าออกสถานที่ (บัตรพนักงาน, บัตรจอดรถ)
- Supply chain management และ Logistics เพื่อเพิ่มขีดความสามารถในการแข่งขันของ ไทย (e-Seal, e-Port)

e-Seal คือ การใช้สติ๊กเกอร์อิเล็กทรอนิกส์ RFID ปิดล็อกตู้คอนเทนเนอร์เพื่อสามารถตรวจสอบติดตาม และบริหารจัดการการขนส่ง ในประเทศสหรัฐอเมริกา ออกกฎหมายว่าด้วย ช่องทางการค้าที่รักษาความปลอดภัย (Secure Trade Lane Lea) กำหนดให้ผู้ส่งสินค้าต้องติด e-Seal

e-Port เป็นโครงการนำร่องระดับท่าเรือแหลมฉบังให้เป็น e-Port โดยความร่วมมือ 3 กระทรวง (คลัง คมนาคม วิทยาศาสตร์)
- ปศุสัตว์ (การให้อาหาร, การติดตามโรค)
- เอกสารราชการ (หนังสือเดินทางอิเล็กทรอนิกส์, บัตรประชาชน)

การพัฒนาเทคโนโลยี RFID ในประเทศไทยศูนย์อิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ หรือเนคเทค (NECTEC) เป็นองค์การของรัฐภายใต้กำกับดูแลของสำนักพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ กระทรวงวิทยาศาสตร์และเทคโนโลยี ได้มีโครงการวิจัยและพัฒนา RFID ตั้งแต่ พ.ศ. 2544 เริ่มพัฒนาตัวโมโครชิป RFID ด้วยการดึงผู้เชี่ยวชาญคนไทยในต่างประเทศมาช่วย

ในการออกแบบวงจรร่วมกับอาจารย์ของมหาวิทยาลัยมหานคร และทีมวิจัย ออกแบบวงจรของศูนย์พัฒนาธุรกิจออกแบบวงจรรวม (Thailand IC Design Incubator : TIDI) ในการออกแบบตัวไมโครชิป RFID ชิปแรกของประเทศไทย โดยคุณสมบัติของไมโครชิป สร้างขึ้นด้วยซีมอสเทคโนโลยีขนาด 0.8 ไมครอนทำงานที่ย่านความถี่ 13.56 MHz มีหน่วยความจำภายในแบบ WORM ขนาด 64 บิต ดังแสดงในรูปที่ 2.5



รูปที่ 2.4 (ซ้าย) ผังวงจรทางกายภาพ (Layout) ของวงจรไมโครชิป RFID ตัวแรกของประเทศไทยที่เทคโนโลยี 0.8 um (ขวา) ภาพการ์ดที่นำไมโครชิปต่อกับขดลวดเพื่อไปใช้งานจริง

ในปีต่อมาทางศูนย์ฯ ก็ได้พัฒนาเครื่องอ่านที่ย่านความถี่ 13.56 MHz เพื่อนำมาใช้งานเพื่อความสมบูรณ์ของระบบ จากนั้นในปี พ.ศ. 2546 ผู้เชี่ยวชาญและทีมงานวิจัยบางส่วนได้ออกไปก่อตั้งบริษัท ซิลิคอน คราฟท์ เทคโนโลยี จำกัด ซึ่งถือว่าเป็นบริษัทแรกของคนไทยในการดำเนินธุรกิจเกี่ยวกับการออกแบบวงจรรวม ที่มุ่งเน้นการออกแบบไมโครชิปด้วย RFID โดยเริ่มต้นได้รับการสนับสนุนจากทางศูนย์ฯ และทาง สวทช. ในการพัฒนาระบบ RFID ที่ใช้ในปศุสัตว์ (Animal Identification) ปัจจุบันทางศูนย์ฯ ได้วิจัยและพัฒนาเครื่องอ่าน RFID อีกประเภท โดยมุ่งเน้นไปยังการพัฒนาเครื่องอ่าน RFID ย่านความถี่ต่ำ โดยพัฒนาตามมาตรฐาน ISO 11784/85 เพื่อประยุกต์ใช้งานจำพวก Animal Tracking, Access Control



รูปที่ 2.5 รูปตัวอย่าง GUI ของโปรแกรมที่นำ RFID ไปประยุกต์ใช้งานต่างๆ



รูปที่ 2.6 บอร์ดเครื่องอ่าน RFID ย่านความถี่ต่ำ

บทสรุป

จะเห็นได้ว่าเทคโนโลยี RFID สามารถนำไปประยุกต์ใช้งานได้หลากหลายประเภท แต่การนำไปใช้งานจะต้องพิจารณาในเรื่องต่างๆ ของระบบที่ต้องการนำไปใช้ เช่น ย่านความถี่ที่ใช้งาน ระยะที่ต้องการอ่านข้อมูล และเรื่องของความปลอดภัยต่างๆ เพื่อจะได้นำเอาเทคโนโลยี RFID ไปใช้งานได้อย่างมีประสิทธิภาพ

2.7 ทฤษฎีซอฟต์แวร์ที่เกี่ยวข้อง

1) ทฤษฎี .NET Framework

เป็นเทคโนโลยีที่กำลังร้อนแรงในขณะนี้ และกำลังจะทำให้รูปแบบและแนวคิดในการเขียนโปรแกรมแบบเดิมๆ เปลี่ยนไป

.NET เป็นกรอบการทำงาน (Framework) ใหม่ของไมโครซอฟต์ที่ครอบคลุมทั้งส่วนของผู้ใช้ ธุรกิจ และนักพัฒนา โดยมีการขยายกรอบการทำงานจากจุดเดิมคือวินโดวส์ที่เน้นเพียงเครื่องคอมพิวเตอร์ (PC ทั้งที่เป็นไคลเอนต์และเซิร์ฟเวอร์) มาเป็นการทำโซลูชัน (Solution) ที่สามารถใช้งานที่ไหนก็ได้ เวลาใดก็ได้ และบนอุปกรณ์ใดๆ ก็ได้ (Anywhere, Anytime and on Any Devices) สิ่งที่สำคัญของ .NET คือ เครือข่ายคอมพิวเตอร์ซึ่งอินเทอร์เน็ตก็เป็นหนึ่งในเครือข่ายหลักที่ใช้ใน .NET และมีการขยายขอบเขตการมองระบบปฏิบัติการ และซอฟต์แวร์ให้กว้างยิ่งขึ้น โดยถือว่าอินเทอร์เน็ตนั้นคือระบบปฏิบัติการ (Operating System) และเว็บไซต์ (ซึ่งให้บริการหนึ่งๆ) เป็นซอฟต์แวร์ของระบบปฏิบัติการ (อินเทอร์เน็ต) แต่แทนที่จะเรียกว่าซอฟต์แวร์อย่างเดิม ก็เรียกใหม่ว่าเป็นเซอร์วิส (Services) หรือเว็บเซอร์วิส (Web Services) แทน

ตัวอย่างของกรณีดังกล่าวคือ เราในฐานะผู้พัฒนาเว็บไซต์ไปดึงเอาเซอร์วิสต่างๆ เช่น เกล็นเตอร์ ฐานข้อมูล และห้องสนทนา จากเว็บไซต์อื่นๆ จากนั้นเพิ่มเติมรายละเอียดอีกนิดหน่อย สำหรับเว็บไซต์เรา เพียงเท่านั้นก็มีบริการใหม่ให้ผู้ใช้เข้ามาเยี่ยมชมได้แล้ว

ในอนาคตของ .NET จะมีบริการให้นักพัฒนาเลือกใช้มากมาย นักพัฒนาที่เพียงเรียกใช้ และประกอบโซลูชันออกมาให้ดีและเหมาะสมเท่านั้น ฉะนั้นในมุมมองของนักพัฒนา (ระดับ ทั่วๆ ไป) การสร้างโซลูชันก็จะง่ายขึ้น อีกทั้งสามารถให้บริการโซลูชันของตนกับผู้ใช้โดยผ่าน เครือข่ายคอมพิวเตอร์หรืออินเทอร์เน็ตได้ ในขณะที่ไม่ต้องไปกังวลว่าเซอร์วิสต่างๆ ที่นำมา ประกอบนั้นอยู่ที่ไหนหรือต้องไปนั่งเขียนโปรแกรมเพื่อจัดการเรื่องเน็ตเวิร์กอีก นักพัฒนาเพียงรู้ว่า เซอร์วิสมันทำอะไรให้ได้ดีบ้าง มีข้อจำกัดอย่างไร จากนั้นก็ประกอบและปรับ แต่ง บางอย่างให้ เหมาะสมเท่านั้น

สำหรับ .NET ในมุมมองของผู้ใช้ก็เพียงแค่เรียกใช้บริการเท่านั้น บริการต่างๆ ก็จะเข้ามา ให้บริการถึงที่ โดยที่อุปกรณ์ที่ผู้ใช้ๆ นั้นก็ไม่ได้จำกัดอยู่เฉพาะคอมพิวเตอร์ แต่จะมีการขยายไปถึง โทรศัพท์มือถือ คอมพิวเตอร์พกพาและอุปกรณ์อื่นๆ ที่สามารถติดต่อกับเครือข่ายคอมพิวเตอร์ได้ บริการต่างๆ ที่นำเสนอมาให้นั้นก็จะนำเสนอในรูปแบบที่เหมาะสมกับอุปกรณ์ของผู้ใช้ ยกตัวอย่าง เช่น ต้องการจะซื้อดอกไม้จากร้านค้าบนอินเทอร์เน็ต ถ้าใช้คอมพิวเตอร์ก็อาจมีรูปภาพซับซ้อน สวยงาม แต่ถ้าใช้โทรศัพท์มือถือซึ่งมีหน้าจอเล็กกว่ามาก ก็จะมีขนาด รูปแบบและรายละเอียดของ การนำเสนอที่แตกต่างกันไป (ในขณะที่ทางฝั่งร้านค้านั้นอาจใช้ข้อมูลจากฐานข้อมูลเพียงอย่าง เดียว)

ต่อไปสำหรับผู้ใช้งาน เวลาจะใช้ซอฟต์แวร์ก็ไม่ต้องซื้อซอฟต์แวร์ทุกตัวมาใช้ อาจมีการ เช่าเซอร์วิสแทน โดยเมื่อจะใช้ก็เรียกใช้บริการก็จะมีค่าใช้จ่ายไปดึงบริการมาจากศูนย์ต่างๆ เมื่อใช้งานเสร็จ ก็จะถูกรับเงินค่าบริการจากการใช้งานนั้น ตัวอย่างหนึ่งที่เรอาจใช้งานกันเป็นประจำ แต่ไม่รู้สึกละเลย ใดๆ เพราะยังใช้งานโดยไม่ต้องเสียค่าบริการ (ในความเป็นจริงนั้นมีการส่งโฆษณาให้คุณเสมอ) คือ บริการเว็บเบสอีเมลล์ เช่น Hotmail เป็นต้น เราเพียงเรียกใช้โดยไม่จำเป็นต้องรู้ว่าระบบพาสปอร์ต (ซึ่งใช้ตรวจสอบผู้ใช้) นั้นอยู่ที่ไหน เขาเก็บข้อมูลของเราอย่างไร ใช้ฐานข้อมูลอะไร อยู่บน ปฏิบัติการอะไร เรากำลังใช้บริการจากเครื่องใดในโลก ในฐานะผู้ใช้เราไม่ต้องรับภาระเรื่องการ อัปเดตซอฟต์แวร์ ทางศูนย์จัดการให้เราเสร็จสรรพ (นี่เป็นสาเหตุหนึ่งที่ทำให้ Hotmail บางช่วง ทำงานได้ไม่เร็วเพราะมีการอัปเดตบริการใหม่ๆ อยู่เป็นประจำ)

บริการ .NET ที่ทางไมโครซอฟท์ให้ลองใช้ในช่วงแรกนี้มี 3 ส่วนคือ เซอร์วิสพื้นที่เก็บ ข้อมูล (Storage) เซอร์วิสการตรวจสอบความเป็นส่วนบุคคล (Authentication) และเซอร์วิสเตือน ต่างๆ (Notification) ตัวอย่างเช่น ให้พื้นที่เก็บเมลล์และไฟล์ มีบริการตรวจสอบโดยใช้พาสเวิร์ดและ

มีบริการเตือนต่างๆ โดยผ่าน MSN Message (ตอนนี้ยังสามารถเตือนผ่านอุปกรณ์ประเภท Mobile ได้อีกด้วย)

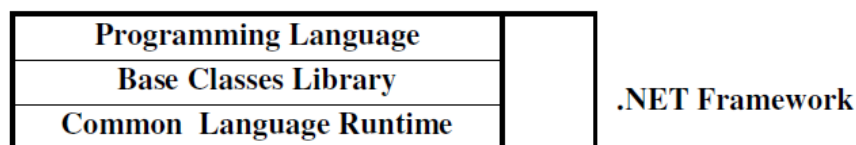
ดังที่ได้กล่าวไปแล้วในตอนต้นว่าไมโครซอฟท์ต้องการที่จะสร้างอะไรที่เป็นมาตรฐานขึ้นมา เพื่อให้ทุกสิ่งทุกอย่าง เพื่อให้สามารถติดต่อสื่อสารกันได้หมด โดยคิดค้นระบบซึ่งให้เป็นระบบมาตรฐาน ระบบนี้คือ .NET Framework ซึ่งระบบนี้ไม่ใช่ระบบปฏิบัติการ (OS) แต่เปรียบเสมือนโปรแกรมหนึ่งที่จะสามารถสร้างสภาวะแวดล้อมหนึ่งซึ่งสามารถทำงานในระบบ .NET นี้ได้

1) Programming Language : เป็นรูปแบบของภาษาที่ออกแบบมาเพื่อให้สามารถทำงานในสภาวะที่เป็น .NET ได้ โดยทางไมโครซอฟท์ได้เปิดตัวภาษาหลักๆ ที่จะใช้พัฒนาบน .NET นี้ 3 ภาษา

- C# เป็นภาษาใหม่ที่ไม่โครซอฟท์พัฒนามาจาก C++ กับ JAVA เป็นหลัก
- VB.NET เป็นภาษาที่พัฒนามาจาก Visual Basic ในเวอร์ชัน 6.0
- Jscript.net เป็นภาษาที่พัฒนามาจาก Jscript ซึ่งเป็น JavaScript ในเวอร์ชันของไมโครซอฟท์

2) Base Classes Library : Library นั้นเปรียบเสมือนชุดคำสั่งสำเร็จรูปย่อยๆ ที่เพิ่มเข้ามาซึ่งส่วนใหญ่จะเป็นชุดคำสั่งที่ต้องใช้งานอยู่เป็นประจำ ดังนั้นจึงมีผู้คิดค้นเพื่ออำนวยความสะดวกในการเขียนโปรแกรม ซึ่ง Library ในภาษาต่างๆ ส่วนใหญ่จะอยู่ในรูปแบบไฟล์ include แต่ถ้าเป็น ASP สิ่งที่เป็น Library ก็คือ คอมโพเนนท์ต่างๆ นั่นเองซึ่งภายในระบบ .NET จะสร้างสิ่งที่เรียกว่าเป็น Library พื้นฐานขึ้น ทำให้ไม่ว่าจะใช้ภาษาใดในการพัฒนาโปรแกรมก็สามารถที่จะเรียกใช้ Library ที่เป็นตัวเดียวกันได้หมด

3) Common Language Runtime (CLR) : นับเป็นสิ่งสำคัญแทบจะที่สุดของระบบ .NET นี้ก็ว่าได้ เพราะ CLR ที่ว่านี้มีหน้าที่ทำให้โปรแกรมที่เขียนขึ้นมาด้วยภาษาต่างๆ กัน กลายเป็นภาษารูปแบบมาตรฐานเดียวกันทั้งหมด เราเรียกภาษานี้ว่า Intermediate Language (IL) ซึ่งเมื่อต้องการที่จะรันโปรแกรมใด CLR ที่ว่านี้จะตรวจสอบเครื่องที่รันว่ามีสภาวะแวดล้อมการทำงานเช่นใด หลังจากนั้นก็จะคอมไพล์เป็นโปรแกรมที่เหมาะสมต่อการทำงานของเครื่องนั้น ทำให้เราสามารถใช้งานโปรแกรมต่างๆ ได้อย่างมีประสิทธิภาพสูงสุดในแต่ละเครื่อง



รูปที่ 2.7 โครงสร้างของ .NET Framework

ประโยชน์และข้อดีของ .NET Framework นั้นพอจะสรุปออกมาได้เป็นข้อๆ ดังนี้

1. เป็นระบบที่มีไลบรารีที่เป็นมาตรฐานเดียวกัน ทำให้เราไม่ต้องกังวลว่าภาษาที่ใช้เขียนนั้นมีไลบรารีหรือไม่ รวมไปถึงการใช้ไลบรารีของภาษาหนึ่งแล้วอีกภาษาหนึ่งจะไม่มีไลบรารีตัวนั้น
2. ไม่ขึ้นกับระบบปฏิบัติการ (OS) เนื่องจากระบบปฏิบัติการที่แต่ละบุคคลหรือองค์กรนั้นย่อมไม่เหมือนกัน แต่ภายใน .NET Framework จะไม่มีปัญหานี้ ขอเพียงแค่มีระบบ .NET Framework ก็จะทำให้สามารถใช้งาน โปรแกรมต่างๆ ได้ ซึ่งเป็นข้อดีตรงที่เราจะสามารถใช้โปรแกรมต่างๆ ได้ทุกระบบปฏิบัติการ
3. ใช้ในการพัฒนาได้ทุกภาษา ทำให้ไม่ต้องศึกษาภาษาใหม่ๆ ในการสร้างโปรแกรมแต่ละครั้ง นอกจากนั้นเรายังสามารถเลือกใช้ภาษาที่เราถนัดที่สุดในการพัฒนาโปรแกรมต่างๆ ได้ด้วย
4. มีการควบคุมสิ่งแวดล้อมในการทำงานเป็นอย่างดี เนื่องจากเป็นระบบที่เป็นมาตรฐานทำให้การควบคุมจัดสรรระบบต่างๆ ทำได้ง่ายขึ้น ไม่ว่าจะเป็นการจัดสรรหน่วยความจำ ด้านการใช้งานเครื่องก็มีความรวดเร็วมากขึ้นลดโอกาสที่เครื่องจะแฮกก็ได้เป็นอย่างดี
5. ความปลอดภัยที่มีมากขึ้น .NET Framework สามารถกำหนดสิทธิ์การใช้งานหรือ Permission ของผู้ใช้งานได้มากขึ้นทำให้สามารถกำหนดว่า จะให้โปรแกรมในส่วนใดใช้งานได้หรือไม่ได้แล้วแต่เฉพาะบุคคล ทั้งหมดนี้เป็นเพียงประโยชน์ส่วนหนึ่งในแนวคิดของไมโครซอฟท์ที่กำลังจะพัฒนาให้สำเร็จเท่านั้น บางข้อทำได้สำเร็จแล้วแต่บางข้อก็ยังไม่สำเร็จดังนั้นจึงต้องคอยรอดูว่า ฝันของไมโครซอฟท์จะเป็นจริงและสำเร็จได้เมื่อไร

2) ทฤษฎี MsComm ที่ใช้ใน Component Visual Basic .NET 2008

ในการติดต่อสื่อสารกับอุปกรณ์ RFID System จะต้องใช้ Component พิเศษในการรับค่าข้อมูลจาก RS232 Port หรือรับจาก Bluetooth ซึ่งจะต้องศึกษารูปแบบคำสั่งและการใช้งาน เราจึงสามารถเขียนโปรแกรมรับค่าข้อมูลจาก RFID System ได้ถูกต้อง

2.8 phpMyAdmin

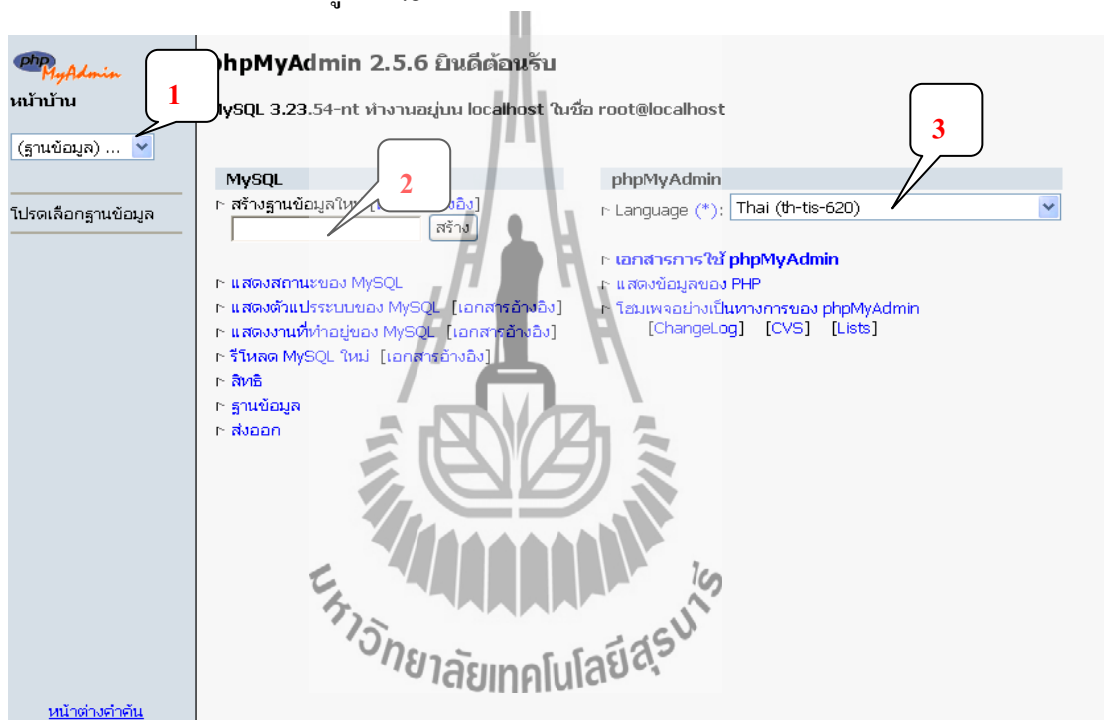
phpMyAdmin เป็นโปรแกรมประเภท MySQL Client ตัวหนึ่งที่ใช้ในการจัดการข้อมูล MySQL ผ่าน Web Browser ได้โดยตรง phpMyAdmin ตัวนี้จะทำงานบน Web Server เป็น PHP Application ที่ใช้ควบคุมจัดการ MySQL Server ความสามารถของ phpMyAdmin คือ

1. สร้างและลบ Database
2. สร้างและจัดการ Table เช่น แทรก record, ลบ record, แก้ไข record หรือลบ Table, เพิ่มหรือแก้ไข field ในตาราง

3. โหลดเท็กซ์ไฟล์เข้าไปเก็บเป็นข้อมูลในตารางได้
4. ทาสรูป (Query) ด้วยคำสั่ง SQL
และอีกหลายๆ ความสามารถที่ phpMyAdmin ทำได้

การเริ่มต้นใช้งาน phpMyAdmin

การเข้าใช้งานโปรแกรม phpMyAdmin จะใช้งานผ่านโปรแกรม Internet Explorer โดยพิมพ์ URL ไปยังไต่แรกทอริของ phpMyAdmin เช่น <http://localhost/phpMyAdmin> ซึ่งจะปรากฏหน้าจอรแรกในการใช้งาน ดังรูปที่ 2.8



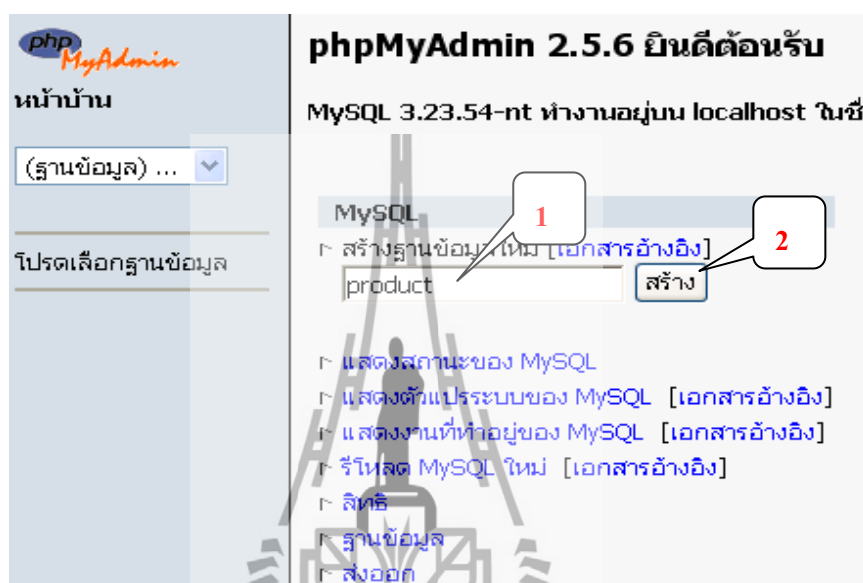
รูปที่ 2.8 หน้าจอรแรกของโปรแกรม phpMyAdmin

จากรูปที่ 2.8 จะเห็นได้ว่าโปรแกรมได้แบ่งส่วนของการแสดงผลออกเป็น 2 ส่วน คือ ส่วนซ้าย (เลข 1) และ ส่วนขวา (เลข 2, 3) ซึ่งจะอธิบายได้ดังนี้

1. ใช้สำหรับ เลือก/เปลี่ยน จัดการกับฐานข้อมูลเดิมที่มีอยู่แล้ว
2. สำหรับใส่ชื่อฐานข้อมูลใหม่ที่ต้องการสร้าง
3. สำหรับเลือกภาษาที่ต้องการให้แสดงภายในโปรแกรม phpMyAdmin

การสร้างฐานข้อมูลใหม่

ก่อนที่จะเก็บข้อมูลลงใน MySQL จะต้องเลือกเสียก่อนว่าจะสร้างฐานข้อมูลชื่ออะไร จากนั้นจึงทำการสร้างตาราง และ ฟิลด์ เพื่อบันทึกข้อมูล ในอันดับแรกจะเป็นการทดลองสร้างฐานข้อมูลขึ้นมาใหม่ โดยจะยกตัวอย่างฐานข้อมูลเกี่ยวกับสินค้า ซึ่งมีชื่อฐานข้อมูลว่า "product" ดังรูปที่ 2.9



รูปที่ 2.9 สร้างฐานข้อมูลใหม่ชื่อ product

จากรูปที่ 2.9 อธิบายขั้นตอนการสร้างฐานข้อมูลใหม่ตามลำดับ ได้ดังนี้

1. ใส่ชื่อของฐานข้อมูลที่ต้องการสร้าง ในที่นี้ให้ชื่อว่า product
2. คลิกที่ปุ่ม สร้าง เพื่อสร้างฐานข้อมูล

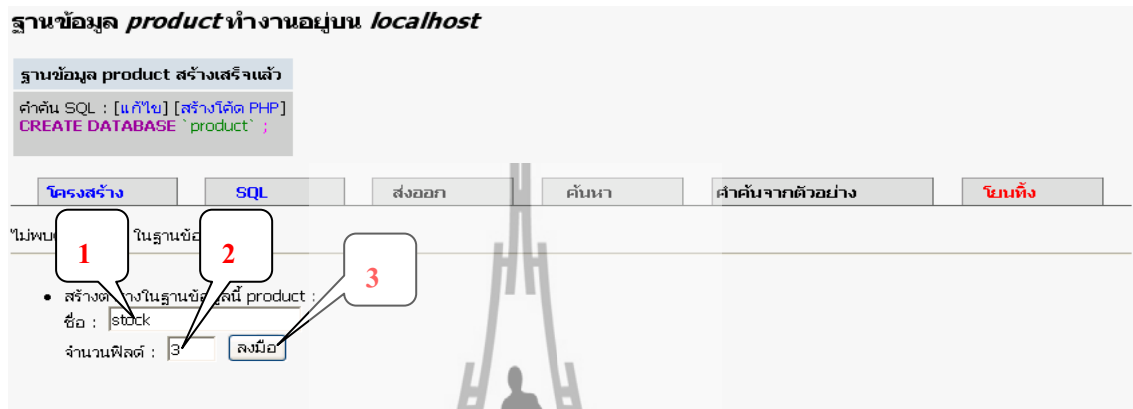
การสร้างตารางใหม่

ภายหลังจากที่สร้างฐานข้อมูลเรียบร้อยแล้ว phpMyAdmin จะแจ้งให้เราทราบว่า ได้ทำการสร้างฐานข้อมูล product เรียบร้อยแล้ว และท่านใช้งานฐานข้อมูล product อยู่ ยังไม่มีตารางใดๆ อยู่เลย ในที่นี้จะสร้างตาราง stock สำหรับเก็บข้อมูลสินค้า โดยรายละเอียดในตารางจะมีโครงสร้างดังนี้

ที่	ชื่อ	ประเภท	ขนาด	หมายเหตุ
1.	name	VARCHAR	30	PK
2.	cost	INT	20	
3.	quan	INT	50	

ให้ทำการสร้างตารางใหม่ โดยมีขั้นตอนดังนี้

1. ใส่ชื่อตารางที่ต้องการสร้าง
2. ใส่จำนวนฟิลด์ของตาราง สำหรับตาราง stock จะมีจำนวน 3 ฟิลด์
3. คลิกปุ่ม ลงมือ เพื่อดำเนินการขั้นต่อไป



รูปที่ 2.10 การสร้างตารางใหม่

ขั้นตอนต่อไป ภายหลังจากคลิกปุ่ม ลงมือในขั้นตอนก่อนหน้า จะเป็นการกำหนด ชื่อฟิลด์ ชนิดของข้อมูลที่เก็บ และรายละเอียดอื่นๆ ซึ่งจะอธิบายในส่วนต่างๆ ได้ดังนี้

- ฟิลด์ สำหรับใส่ชื่อฟิลด์
- ชนิด สำหรับเลือกชนิดของข้อมูลที่ต้องการเก็บในฟิลด์นั้น
- ความยาว/เขต สำหรับกำหนดขนาดของข้อมูล
- แอตทริบิวต์ สำหรับเลือกลักษณะเฉพาะของข้อมูลที่จะเก็บ เช่น ตัวเลขแบบคิดเครื่องหมาย บวกหรือลบ เป็นต้น
- ค่าว่างเปล่า (null) สำหรับเลือกว่า ฟิลด์นั้นสามารถใส่ค่าว่างได้หรือไม่
- ค่าปริยาย สำหรับกำหนดค่าเริ่มต้นของฟิลด์ (ค่า Default)
- เพิ่มเติม สำหรับกำหนดค่าเพิ่มเติม เช่น กรณีที่ฟิลด์เก็บข้อมูลตัวเลขจำนวนเต็ม (Integer) จะสามารถเลือกให้มีการเพิ่มค่าอัตโนมัติ (auto_increment) ได้ เป็นต้น
- ไพรมารี เลือกเมื่อต้องการกำหนดให้ฟิลด์นั้นๆ เป็นไพรมารีคีย์ (Primary Key)
- ดัชนี เลือกเมื่อต้องการสร้างฟิลด์นั้นๆ เป็นดัชนี (Index)
- เอกลักษณะ เลือกเมื่อต้องการให้ฟิลด์นั้นเป็น Unique

ต่อไปนี้เป็นคำอธิบายของชนิดข้อมูลแต่ละชนิดที่สามารถเลือกได้

VARCHAR : สำหรับเก็บข้อมูลประเภทตัวอักษร ทุกครั้งที่เลือกชนิดของฟิลด์เป็นประเภทนี้ จะต้องมีการกำหนดความยาวของข้อมูลลงไปด้วย ซึ่งสามารถกำหนดค่าได้ตั้งแต่ 1 - 255 ฟิลด์ชนิดนี้เหมาะสำหรับการเก็บข้อมูลสั้นๆ เช่น ชื่อ นามสกุล หรือหัวข้อต่างๆ เป็นต้น. ในส่วนฟิลด์ประเภทนี้ จะสามารถเลือก "แอดทริบิวต์" เป็น BINARY ได้ โดยปกติแล้วการจัดเรียงข้อมูลเวลาสืบค้น (query) สำหรับ VARCHAR จะเป็นแบบ case-sensitive (ตัวอักษรใหญ่ และเล็กมีความหมายแตกต่างกัน) แต่หากระบุ "แอดทริบิวต์" เป็น BINARY การสืบค้นจะไม่คำนึงตัวอักษรว่าจะเป็นตัวใหญ่ หรือตัวเล็ก

CHAR : สำหรับเก็บข้อมูลประเภท ตัวอักษร แบบที่ถูกจำกัดความกว้างเอาไว้คือ 255 ตัวอักษร ไม่สามารถปรับเปลี่ยนได้เหมือนกับ VARCHAR หากทำการสืบค้นโดยเรียงตามลำดับ ก็ จะเรียงข้อมูลแบบ case-sensitive เว้นแต่จะกำหนดแอดทริบิวต์เป็น BINARY ที่จะทำให้การเรียงข้อมูลเป็นแบบ non case-sensitive เช่นเดียวกับ VARCHAR

TINYTEXT : ในกรณีที่ข้อความยาวๆ หรือต้องการที่จะค้นหาข้อความ โดยอาศัยฟีเจอร์ FULL TEXT SEARCH ของ MySQL เราอาจจะเลือกที่จะไม่เก็บข้อมูลลงในฟิลด์ประเภท VARCHAR ที่มีข้อจำกัดแค่ 256 ตัวอักษร แต่เราจะเก็บลงฟิลด์ประเภท TEXT แทน โดย TINYTEXT นี้ จะสามารถเก็บข้อมูลได้ 256 ตัวอักษร ซึ่งมองเผินๆ ก็ไม่ต่างกับเก็บลงฟิลด์ประเภท CHAR หรือ VARCHAR(255) เลย แต่จริงๆ มันต่างกันตรงที่มันทำ FULL TEXT SEARCH ได้

TEXT : สำหรับเก็บข้อมูลประเภทตัวอักษร เช่นเดียวกับ TINYTEXT แต่สามารถเก็บได้มากขึ้น โดยสูงสุดคือ 65,535 ตัวอักษร หรือ 64KB เหมาะสำหรับการเก็บข้อมูลพวกเนื้อหาต่างๆ ที่ยาวๆ

MEDIUMTEXT : เก็บข้อมูลประเภทตัวอักษร เช่นเดียวกับ TINYTEXT แต่เก็บข้อมูลได้ 16,777,215 ตัวอักษร

LONGTEXT : เก็บข้อมูลประเภทตัวอักษร เช่นเดียวกับ TINYTEXT แต่เก็บข้อมูลได้ 4,294,967,295 ตัวอักษร

TINYINT : สำหรับเก็บข้อมูลชนิดตัวเลขที่มีขนาด 8 บิต ข้อมูลประเภทนี้เราสามารถกำหนดเพิ่มเติมในส่วนของ "แอตทริบิวต์" ได้ว่าจะเลือกเป็น UNSIGNED หรือ UNSIGNED ZEROFILL โดยจะมีความแตกต่างดังนี้

- UNSIGNED : จะหมายถึงเก็บค่าตัวเลขแบบไม่มีเครื่องหมาย แบบนี้จะทำให้สามารถเก็บค่าได้ตั้งแต่ 0 - 255
- UNSIGNED ZEROFILL : เหมือนข้างต้น แต่ว่าหากข้อมูลที่กรอกเข้ามาไม่ครบตามจำนวนหลักที่เรากำหนด ตัว MySQL จะทำการเติม 0 ให้ครบหลักเอง เช่น ถ้ากำหนดให้ใส่ได้ 3 หลัก แล้วทำการเก็บข้อมูล 25 เข้าไป เวลาที่สืบค้นดู เราจะได้ค่าออกมาเป็น 025

หากไม่เลือก "แอตทริบิวต์" สิ่งที่เราจะได้ก็คือ SIGNED นั่นก็จะต้องเสียบิตหนึ่งไปเก็บเครื่องหมาย บวก/ลบ ทำให้สามารถเก็บข้อมูลได้อยู่ในช่วง -128 ถึง 127 เท่านั้น

SMALLINT : สำหรับเก็บข้อมูลประเภทตัวเลขที่มีขนาด 16 บิต จึงสามารถเก็บค่าได้ตั้งแต่ -32768 ถึง 32767 (ในกรณีแบบคิดเครื่องหมาย) หรือ 0 ถึง 65535 (ในกรณี UNSIGNED หรือ ไม่คิดเครื่องหมาย) ซึ่งสามารถเลือก Attribute เป็น UNSIGNED และ UNSIGNED ZEROFILL ได้เช่นเดียวกับ TINYINT

MEDIUMINT : สำหรับเก็บข้อมูลประเภทตัวเลขที่มีขนาด 24 บิต นั่นก็หมายความว่าสามารถเก็บข้อมูลตัวเลขได้ตั้งแต่ -8388608 ไปจนถึง 8388607 (ในกรณีแบบคิดเครื่องหมาย) หรือ 0 ถึง 16777215 (ในกรณีที่เป็น UNSIGNED หรือ ไม่คิดเครื่องหมาย) ซึ่งสามารถเลือก Attribute เป็น UNSIGNED และ UNSIGNED ZEROFILL ได้เช่นเดียวกับ TINYINT

INT : สำหรับเก็บข้อมูลประเภทตัวเลขที่มีขนาด 32 บิต หรือสามารถเก็บข้อมูลได้ตั้งแต่ -2147483648 ไปจนถึง 2147483647 (ในกรณีแบบคิดเครื่องหมาย) หรือ 0 ถึง 4294967295 (ในกรณีที่เป็น UNSIGNED หรือ ไม่คิดเครื่องหมาย) ซึ่งสามารถเลือก Attribute เป็น UNSIGNED และ UNSIGNED ZEROFILL ได้เช่นเดียวกับ TINYINT

BIGINT : สำหรับเก็บข้อมูลประเภทตัวเลขที่มีขนาด 64 บิต สามารถเก็บข้อมูลได้ตั้งแต่ -9223372036854775808 ไปจนถึง 9223372036854775807 เลขที่เดียว (แบบคิดเครื่องหมาย) หรือ 0

ถึง 18446744073709551615 (ในกรณีที่เป็น UNSIGNED หรือไม่คิดเครื่องหมาย) ซึ่งสามารถเลือก Attribute เป็น UNSIGNED และ UNSIGNED ZEROFILL ได้เช่นเดียวกับ TINYINT

FLOAT(M,D) : ที่กล่าวถึงไปทั้งหมด ในตระกูล INT นั้นจะเป็นเลขจำนวนเต็ม หากเราบันทึกข้อมูลที่มีเศษทศนิยม มันจะถูกปัดทันทันที ดังนั้นหากต้องการจะเก็บค่าที่เป็นเลขทศนิยม ต้องเลือกชนิดของฟิลด์เป็น FLOAT โดยจะเก็บข้อมูลแบบ 32 บิต คือมีค่าตั้งแต่ $-3.402823466E+38$ ไปจนถึง $-1.175494351E-38$, 0 และ $1.175494351E-38$ ถึง $3.402823466E+38$

DOUBLE(M,D) : สำหรับเก็บข้อมูลประเภทตัวเลขทศนิยม เช่นเดียวกับ FLOAT แต่มีขนาดเป็น 64 บิต สามารถเก็บได้ตั้งแต่ $-1.7976931348623157E+308$ ถึง $-2.2250738585072014E-308$, 0 และ $2.2250738585072014E-308$ ถึง $1.7976931348623157E+308$

DECIMAL(M,D) : สำหรับเก็บข้อมูลประเภทตัวเลขทศนิยม เช่นเดียวกับ FLOAT แต่ใช้กับข้อมูลที่ต้องการความละเอียดและถูกต้องของข้อมูลสูง

ข้อสังเกต เกี่ยวกับข้อมูลประเภท FLOAT, DOUBLE และ DECIMAL ก็คือ เวลากำหนดความยาวของข้อมูลในฟิลด์ จะถูกกำหนดอยู่ในรูปแบบ (M,D) ซึ่งหมายความว่า ต้องมีการระบุว่าจะให้มีตัวเลขส่วนที่เป็นจำนวนเต็มกี่หลัก และมีเลขทศนิยมกี่หลัก เช่น ถ้าเรากำหนดว่า FLOAT(5,2) จะหมายความว่า เราจะเก็บข้อมูลเป็นตัวเลขจำนวนเต็ม 5 หลัก และทศนิยม 2 หลัก ดังนั้นหากทำการใส่ข้อมูล 12345.6789 เข้าไป สิ่งที่จะเข้าไปอยู่ในข้อมูลจริงๆ ก็คือ 12345.68 (ปัดเศษให้มีจำนวนหลักตามที่กำหนดไว้)

DATE : สำหรับเก็บข้อมูลประเภทวันที่ โดยเก็บได้จาก 1 มกราคม ค.ศ. 1000 ถึง 31 ธันวาคม ค.ศ. 9999 โดยจะแสดงผลในรูปแบบ YYYY-MM-DD

DATETIME : สำหรับเก็บข้อมูลประเภท วันที่ และเวลา โดยจะเก็บได้ตั้งแต่ 1 มกราคม ค.ศ. 1000 เวลา 00:00:00 ไปจนถึง 31 ธันวาคม ค.ศ. 9999 เวลา 23:59:59 โดยรูปแบบการแสดงผลเวลาที่ทำการสืบค้น (query) ออกมา จะเป็น YYYY-MM-DD HH:MM:SS

TIMESTAMP(M) : สำหรับเก็บข้อมูลประเภท วันที่ และเวลาเช่นกัน แต่จะเก็บในรูปแบบของ YYYYMMDDHHMMSS หรือ YMMDDHHMMSS หรือ YYYYMMDD หรือ

YYMMDD แล้วแต่ว่าจะระบุค่า M เป็น 14, 12, 8 หรือ 6 ตามลำดับ สามารถเก็บได้ตั้งแต่วันที่ 1 มกราคม ค.ศ. 1000 ไปจนถึงประมาณปี ค.ศ. 2037

TIME : สำหรับเก็บข้อมูลประเภทเวลา มีค่าได้ตั้งแต่ -838:59:59 ไปจนถึง 838:59:59 โดยจะแสดงผลออกมาในรูปแบบ HH:MM:SS

YEAR[(2/4)] : สำหรับเก็บข้อมูลประเภทปี ในรูปแบบ YYYY หรือ YY แล้วแต่ว่าจะเลือก 2 หรือ 4 (หากไม่ระบุ จะถือว่าเป็น 4 หลัก) โดยหากเลือกเป็น 4 หลัก จะเก็บค่าได้ตั้งแต่ ค.ศ. 1901 ถึง 2155 แต่หากเป็น 2 หลัก จะเก็บตั้งแต่ ค.ศ. 1970 ถึง 2069

ข้อสังเกต ค่าที่เก็บในข้อมูลประเภท **TIMESTAMP** และ **YEAR** นั้นจะมีความสามารถพอๆ กับการเก็บข้อมูลวันเดือนปี และเวลา ด้วยฟิลด์ชนิด **VARCHAR** แต่ต่างกันตรงที่ จะใช้เนื้อที่เก็บข้อมูลน้อยกว่า... ทว่า ฟิลด์ประเภท **TIMESTAMP** นั้นจะมีข้อจำกัดในเรื่องของเวลาที่สามารถเก็บได้ คือจะต้องอยู่ในระหว่าง 1 มกราคม ค.ศ. 1000 ไปจนถึง ค.ศ. 2037 แต่หากเก็บเป็น **VARCHAR** นั้นจะไม่ติดข้อจำกัดนี้

- ฟิลด์ชนิด **YEAR** ใช้เนื้อที่แค่ 1 ไบต์เท่านั้นในการเก็บข้อมูล แต่ข้อจำกัดจะอยู่ที่ปี ค.ศ. 1901 ถึง 2155 เท่านั้น (หรือ ค.ศ. 1970 ถึง 2069 ในกรณี 2 หลัก) แต่หากเก็บเป็น **VARCHAR** จะได้ตั้งแต่ 0000 ถึง 9999

TINYBLOB : สำหรับเก็บข้อมูลประเภท ไบนารี ได้แก่ ไฟล์ข้อมูลต่างๆ , ไฟล์รูปภาพ, ไฟล์มัลติมีเดีย เป็นต้น คือไฟล์อะไรก็ตามที่อัปโหลดผ่านฟอร์มอัปโหลดไฟล์ในภาษา HTML โดย **TINYBLOB** นั้นจะมีเนื้อที่ให้เก็บข้อมูลได้ 256 ไบต์

BLOB : สำหรับเก็บข้อมูลประเภท ไบนารี เช่นเดียวกับ **TINYBLOB** แต่สามารถเก็บข้อมูลได้ 64KB

MEDIUMBLOB : สำหรับเก็บข้อมูลประเภท ไบนารี เช่นเดียวกับ **TINYBLOB** แต่เก็บข้อมูลได้ 16MB

LONGBLOB : สำหรับเก็บข้อมูลประเภท ไบนารี เช่นเดียวกับ **TINYBLOB** แต่เก็บข้อมูลได้ 4GB

ข้อสังเกต ข้อมูลประเภท BLOB นั้น แม้จะมีประโยชน์ในเรื่องของการเก็บข้อมูลประเภท BINARY ให้อยู่กับตัวฐานข้อมูล ทำให้สะดวกเวลาสืบค้นก็ตาม แต่มันก็ทำให้ฐานข้อมูลมีขนาดใหญ่เกินความจำเป็นด้วย ทำให้เกิดความไม่สะดวกในการสำรองฐานข้อมูลในกรณีที่ มีข้อมูลอัปเดตไปเก็บมากๆ โดยปกติแล้ว จะใช้วิธีการอัปเดตไปเก็บไว้ในไฟล์เคอร์ แล้วเก็บลิงก์ไปยังไฟล์เหล่านั้น เป็นฟิลด์ชนิด VARCHAR มากกว่า

SET : สำหรับเก็บข้อมูลที่เป็นกลุ่มของข้อมูลที่ยอมให้เลือกได้ 1 ค่าหรือหลายๆ ค่า ซึ่งสามารถกำหนดได้ถึง 64 ค่า

ให้กรอกรายละเอียดต่างๆ ให้ครบ เช่น สร้างฟิลด์ 9 ฟิลด์ เราต้องระบุชื่อฟิลด์ ชนิดของข้อมูลที่จัดเก็บ หรือรายละเอียดอื่นๆ ให้ครบทั้ง 9 ฟิลด์ก่อน เมื่อเรียบร้อยแล้วคลิกที่ปุ่ม บันทึก เพื่อสร้างตารางใหม่ตามรายละเอียดที่ได้กำหนดไว้ ดังรูปที่ 2.11

ฐานข้อมูล *product* - ตาราง *stock* ทำงานอยู่บน *localhost*

ฟิลด์	ชนิด [เอกสารอ้างอิง]	ความยาว/ เขต*	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย**	เพิ่มเติม	โพร มารี	ดัชนี	เอกลักษณ์	---
name	VARCHAR	30		not null			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
cost	INT			not null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
quan	INT			not null			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

หมายเหตุของตาราง :

ชนิดตาราง : ค่าปริยาย

รูปที่ 2.11 การกำหนดรายละเอียดของฟิลด์ต่างๆ ภายในตาราง

หลังจากคลิกที่ปุ่ม บันทึก แล้ว โปรแกรมจะแจ้งให้ทราบว่า ได้สร้างตาราง *stock* เรียบร้อยแล้ว และจะเข้าสู่หน้าจอสำหรับการจัดการตาราง ภายในหน้าจอจะมีเมนูต่างๆ หลายเมนู ซึ่งจะอธิบายในลำดับถัดไป และ มีการแจ้งให้ผู้ใช้ทราบรายละเอียดอื่นๆ ที่เกี่ยวข้องกับตารางนั้นๆ เช่น คณิตที่ใช้งาน ค่าสถิติต่างๆ เป็นต้น ดังรูปที่ 2.12

ฐานข้อมูล **product** - ตาราง **stock** ทำงานอยู่บน **localhost**

ตาราง stock สร้างเสร็จแล้ว

คำสั่ง SQL : [แก้ไข] [สร้างโค้ด PHP]
CREATE TABLE `stock` (
 `name` VARCHAR(30) NOT NULL ,
 `cost` INT NOT NULL ,
 `quan` INT NOT NULL ,
 PRIMARY KEY (`name`)
);

โครงสร้าง เปิดดู SQL ค้นหา แทรก ส่งออก กระบวนการ ลบข้อมูล โยนทิ้ง

ฟิลด์	ชนิด	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	การจัดการ
<input type="checkbox"/> name	varchar(30)		ไม่			
<input type="checkbox"/> cost	int(11)		ไม่	0		
<input type="checkbox"/> quan	int(11)		ไม่	0		

เลือกทั้งหมด / ไม่เลือกเลย ทำกับที่เลือก:

ดัชนี : [เอกสารอ้างอิง]

ชื่อคีย์	ชนิด	Cardinality	การจัดการ	ฟิลด์	เนื้อที่ที่ใช้:	สถิติของแถว:
PRIMARY PRIMARY		0		name	ชนิด ใช้งาน	คำสั่ง ค่า
					ข้อมูล 0 ไบต์	รูปแบบ ไม่คงที่
					ดัชนี 1,024 ไบต์	แถว 0
					รวม 1,024 ไบต์	สร้างเมื่อ 13 ส.ค. 2004 น.
						ปรับปรุงครั้งสุดท้ายเมื่อ 13 ส.ค. 2004 น.

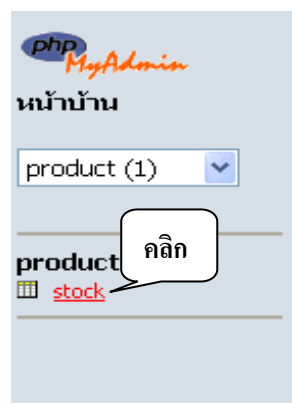
สร้างดัชนีโดยคอลัมน์ 1

รูปที่ 2.12 หน้าจอภายหลังสร้างตารางใหม่

การจัดการกับฟิลด์และข้อมูลของตาราง






1. การเรียกดูโครงสร้างของตาราง

การเรียกดูโครงสร้างของตาราง เราจะต้องอยู่ในหน้าจอการจัดการตารางก่อน (รูปที่ 2.12) ซึ่งการเข้ามายังหน้าจอนี้ได้ มี 2 วิธี คือ คลิกจากชื่อตารางซึ่งแสดงอยู่ส่วนซ้ายของโปรแกรม แสดงดังรูปที่ 2.13 หรือ ภายหลังจากสร้างตารางใหม่ก็จะเข้ามายังหน้าจอจัดการตารางเช่นเดียวกัน



รูปที่ 2.13 รายชื่อตารางภายในฐานข้อมูล คลิกเพื่อจัดการตารางนั้นๆ

ภายในหน้าจอดีการตาราง โปรแกรมจะแสดงโครงสร้างของตารางเป็นหน้าจอแรก หรือ เราสามารถคลิกที่เมนู โครงสร้าง เพื่อแสดงโครงสร้างของข้อมูลได้เช่นเดียวกัน ดังรูปที่ 2.14 โปรแกรมจะแสดงโครงสร้างของตารางนั้นๆ อันประกอบไปด้วย ชื่อฟิลด์ ชนิดของข้อมูล แดททริบิวต์ การใส่ค่าว่าง และไอคอนสำหรับจัดการ ดังนี้

-  สำหรับแก้ไข ชื่อฟิลด์ ชนิดของข้อมูลที่เก็บ ไพรมารีคีย์ หรือค่าเริ่มต้นอื่นๆ
-  สำหรับลบฟิลด์นั้นๆ ออกจากตาราง
-  สำหรับกำหนดให้ฟิลด์นั้นๆ เป็นไพรมารีคีย์ (Primary Key)
-  สำหรับกำหนดให้ฟิลด์นั้นๆ เป็นดัชนี (Index)
-  สำหรับกำหนดให้ฟิลด์นั้นๆ เป็นเอกลักษณ์ (Unique)

ฐานข้อมูล **คลิก** - ตาราง *stock* ทำงานอยู่บน *localhost*

โครงสร้าง | เปิดดู | SQL | ค้นหา | เดรก | ส่งออก | กระบวนการ | ลบข้อมูล | โคมัง

ฟิลด์	ชนิด	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> name	varchar(30)		ไม่			
<input type="checkbox"/> cost	int(11)		ไม่	0		
<input type="checkbox"/> quan	int(11)		ไม่	0		

เลือกทั้งหมด / ไม่เลือกเลย ทำกับที่เลือก:

ดัชนี : [เอกสารอ้างอิง]

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์	เนื้อที่ที่ใช้ :	สถิติของแถว :
PRIMARY	PRIMARY	0		name	ชนิด : 0 ไบต์	ค่าสิ่ง : ไม่คงที่
					ดัชนี : 1,024 ไบต์	แถว : 0
					รวม : 1,024 ไบต์	สร้างเมื่อ : 13 ส.ค. 2004 น.
						ปรับปรุงครั้งสุดท้ายเมื่อ : 13 ส.ค. 2004 น.

สร้างดัชนีโดยอัตโนมัติ 1

รูปที่ 2.14 หน้าจอดีการตาราง ในส่วนการแสดงผลโครงสร้างตาราง

2. การเปิดดูข้อมูลภายในตาราง

การเปิดดูข้อมูลภายในตาราง ทำได้โดยคลิกที่เมนู **เปิดดู** ซึ่งจะคลิกได้ก็ต่อเมื่อมีข้อมูลภายในตารางแล้วเท่านั้น สำหรับเมนู **เปิดดู** แสดงดังรูปที่ 2.15 และ 2.16

ฐานข้อมูล *product* - ตาราง *stock* ทำงานอยู่บน *localhost*

โครงสร้าง **เปิดดู** SQL ค้นหา เพชร ส่งออก กระบวนการ ลบข้อมูล

ฟิลด์	ชนิด	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ				
<input type="checkbox"/> name	varchar(30)		ไม่							
<input type="checkbox"/> cost	int(11)		ไม่	0						
<input type="checkbox"/> quan	int(11)		ไม่	0						

เลือกทั้งหมด / *ไม่เลือกเลย ทำกับที่เลือก:

ดัชนี : [เอกสารอ้างอิง]

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	1		name

สร้างดัชนีโดยคอลัมน์

เนื้อที่ที่ใช้ :

ชนิด	ใช้งาน
ข้อมูล	40 ไบต์
ดัชนี	2,048 ไบต์
รวม	2,088 ไบต์

สถิติของแถว :

ค่าส่ง	ค่า
รูปแบบ	ไม่คงที่
แถว	1
ความยาวแถว	40
ขนาดแถว	2,088 ไบต์
สร้างเมื่อ	13 ส.ค. 2004 น.
ปรับปรุงครั้งสุดท้ายเมื่อ	13 ส.ค. 2004 น.

รูปที่ 2.15 แสดงเมนู เปิดดู

ฐานข้อมูล *product* - ตาราง *stock* ทำงานอยู่บน *localhost*

โครงสร้าง **เปิดดู** SQL ค้นหา เพชร ส่งออก กระบวนการ ลบข้อมูล

แสดงระเบียนที่ 0 - 0 (1 ทั้งหมด, ค่าค้นใช้เวลา 0.0005 วินาที)

คำค้น SQL : [แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP]
SELECT *
FROM `stock` LIMIT 0 , 30

แสดง : แถว เริ่มจากแถวที่

อยู่ใน และซ้ำหัวแถวทุกๆ เซลล์

←T→	name	cost	quan
<input type="checkbox"/>	ยาแก้แสบ	100	5

ทำกับที่เลือก:

แสดง : แถว เริ่มจากแถวที่

อยู่ใน และซ้ำหัวแถวทุกๆ เซลล์

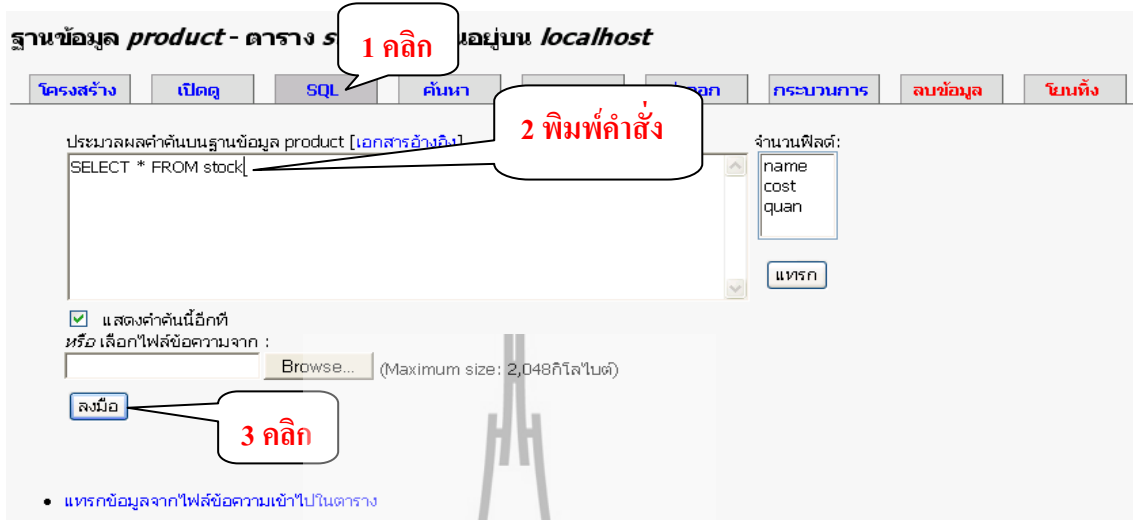
แทรกแถวใหม่
 แสดง
 Print view (with full texts)

รูปที่ 2.16 ข้อมูลภายในตาราง ภายหลังจากคลิกเมนู เปิดดู

3. การเปิดดูข้อมูลภายในตารางด้วยคำสั่ง SQL

ในหัวข้อที่ผ่านมา (2) เป็นการเปิดดูข้อมูลโดยดูข้อมูลทั้งหมด ในการใช้งานจริงจะมีการเลือกดูข้อมูลแบบมีเงื่อนไข ซึ่งสามารถเลือกดูได้ในโปรแกรม โดยคลิกที่เมนู SQL จากนั้นจะ

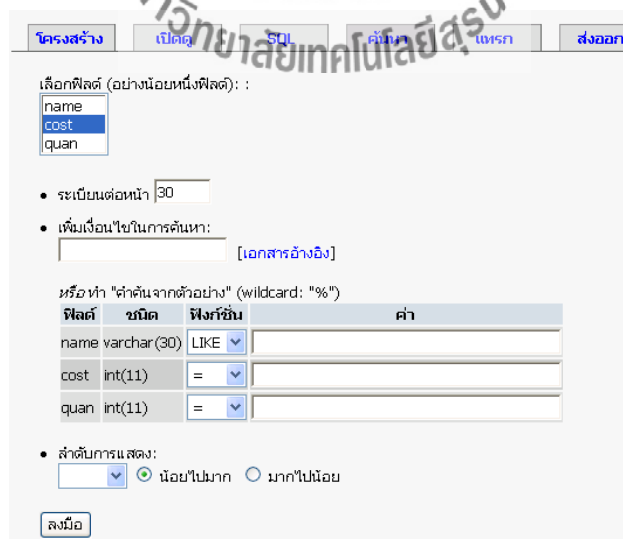
ปรากฏ text area สำหรับพิมพ์คำสั่ง SQL ลงไป เมื่อเรียบร้อยแล้วคลิกที่ปุ่ม ลงมือ เพื่อรันคำสั่ง SQL ดังแสดงในรูปที่ 2.17



รูปที่ 2.17 การใช้คำสั่ง SQL เพื่อเปิดดูข้อมูลภายในตาราง

4. การค้นหา

การค้นหาภายในเมนูค้นหา มีความหมายเดียวกับการใช้คำสั่ง SQL สำหรับเปิดดูข้อมูล แต่ในเมนูการค้นหานี้ จะเป็น user interface ให้ผู้ใช้เลือก ทำให้ง่ายต่อผู้ใช้ที่ต้องการความสะดวก และง่ายต่อการใช้งาน สำหรับหน้าจอของเมนู การค้นหา แสดงดังรูปที่ 2.18



รูปที่ 2.18 หน้าจอการค้นหา

จากรูปที่ 2.18 นำมาอธิบายการใช้งานการค้นหา ได้ดังนี้

1. เลือกฟิลด์ที่ต้องการแสดงข้อมูล (เลือกอย่างน้อยหนึ่งฟิลด์)
2. กำหนดจำนวนระเบียบที่ต้องการแสดงใน 1 หน้า
3. กรอกเงื่อนไขในการค้นหา (ถ้ามี)
4. ระบุเงื่อนไขของฟิลด์ต่างๆ

5. การเพิ่มข้อมูลลงยังตาราง

การเพิ่มข้อมูลลงยังตาราง สามารถทำได้โดยคลิกที่เมนู แทรก โปรแกรมจะแสดงหน้าจอสำหรับเพิ่มข้อมูลดังรูปที่ 2.19 จากนั้นให้กรอกข้อมูลที่ต้องการลงไปคอลัมน์ ค่า (Value) เมื่อกรอกข้อมูลเรียบร้อยแล้ว หากต้องการกลับมาเพิ่มข้อมูลในแถวต่อไปเลย ให้คลิกที่ แทรกระเบียบใหม่ จากนั้นคลิกที่ ลงมือ

ฟิลด์	ชนิด	ฟังก์ชัน	วางเปล่า (null)	ค่า
name	varchar(30)			
cost	int(11)		0	
quan	int(11)		0	

แทรกเป็นแถวใหม่ -- และ --

ส่งกลับ หรือ แทรกระเบียบใหม่

รูปที่ 2.19 การเพิ่มข้อมูลลงยังตาราง

6. การลบข้อมูลในตาราง

การลบข้อมูลภายในตาราง คลิกที่เมนู เปิดดู โปรแกรมจะแสดงรายการข้อมูล จากนั้นคลิกที่รูป เพื่อลบข้อมูล ดังรูปที่ 2.20 (สามารถให้คำสั่ง SQL ในการลบข้อมูลได้อีกวิธีหนึ่ง)

	cost	quan
	100	5

ทำกับที่เลือก:

รูปที่ 2.20 การลบข้อมูลภายในตาราง

7. การแก้ไขข้อมูลในตาราง

การแก้ไขข้อมูลในตาราง คลิกที่เมนูเปิดดู โปรแกรมจะแสดงรายการข้อมูลจากนั้นคลิกที่รูป เพื่อแก้ไขข้อมูล ดังรูปที่ 2.21 (สามารถให้คำสั่ง SQL ในการแก้ไขได้อีกวิธีหนึ่ง) เมื่อคลิกที่รูปแก้ไขแล้ว จะปรากฏหน้าจอสำหรับแก้ไขข้อมูลดังรูปที่ 2.22 เป็นช่องสำหรับแก้ไข โดยจะแสดงทุกๆ ฟิวด์ภายในตาราง และหน้าจอแก้ไขนี้สามารถบันทึก หรือ แทรกข้อมูลเป็นแถวใหม่ได้ ด้วย




รูปที่ 2.21 การแก้ไขข้อมูลในตาราง

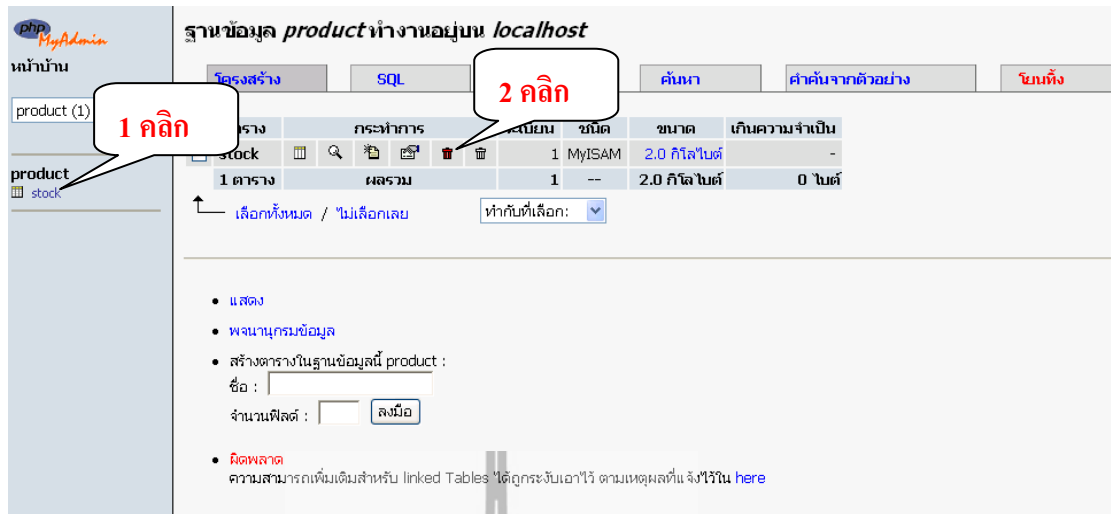
ฟิลด์	ชนิด	ฟังก์ชัน	ว่างเปล่า (null)	ค่า
name	varchar(30)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	ยาเกินมุง
cost	int(11)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	100
quan	int(11)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	5

บันทึก หรือ แทรกเป็นแถวใหม่
 ส่งกลับ หรือ แทรกระเบียบใหม่

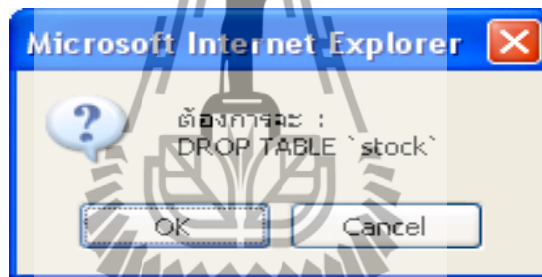
รูปที่ 2.22 การแก้ไขข้อมูลในตาราง (ต่อ)

8. การลบตาราง

การลบตารางออกจากฐานข้อมูล ทำได้โดยคลิกที่ชื่อฐานข้อมูลทางด้านซ้ายของโปรแกรม เพื่อแสดงตารางทั้งหมดขึ้นมา จากนั้นเลือกคลิกที่รูป  เพื่อลบตารางที่ต้องการ แสดงการลบตาราง ดังรูปที่ 2.23



รูปที่ 2.23 การลบตารางออกจากฐานข้อมูล



รูปที่ 2.24 ยืนยันการลบตาราง

จากรูปที่ 2.24 เป็นหน้าจอยืนยันการลบตาราง ถ้าต้องการลบตารางให้คลิกที่ OK ถ้าไม่ต้องการลบตารางให้คลิกที่ Cancel

9. การ Export ข้อมูล (ส่งออก)

การ Export ข้อมูล หรือ ฐานข้อมูล ทำได้โดย คลิกที่เมนู ส่งออก ดังรูปที่ 2.25 จากนั้น โปรแกรมจะแสดงหน้าจอสำหรับตั้งค่าการ Export ต่างๆ ดังรูปที่ 2.26

ฐานข้อมูล *product* ทำงานบน *localhost*

หน้าบ้าน

product

product

stock

1. คลิก

2. คลิก

ตาราง	ประเภทการ	ระบบ	ชนิด	ขนาด	เก็บความจำเป็น
stock		1	MyISAM	2.0 กิโลไบต์	-
1 ตาราง	ผลรวม	1	--	2.0 กิโลไบต์	0 ไบต์

เลือกทั้งหมด / ไม่เลือกเลย

ทำกับที่เลือก: ▼

- แสดง
- พจนานุกรมข้อมูล
- สร้างตารางในฐานข้อมูลนี้ *product* :
 - ชื่อ :
 - จำนวนฟิลด์ :
- ผิดพลาด

ความสามารถเพิ่มเติมสำหรับ linked Tables *ได้ถูกรับเอาไป ตามเหตุผลที่แจ้งไว้ใน [here](#)

รูปที่ 2.25 เมนูส่งออก

ดูโครงสร้างของฐานข้อมูล

ส่งออก

ตัวเลือก SQL (เอกสารอ้างอิง)

SQL

LaTeX

ข้อมูล CSV สำหรับไมโครซอฟต์เอ็กเซล

ข้อมูล CSV (ต้นตอเครื่องหมายลูกน้ำ ",")

XML

คลิก

กำหนดชื่อไฟล์

ส่งออก

ตัวเลือก SQL (เอกสารอ้างอิง)

โครงสร้าง

เพิ่มคำสั่ง 'drop table'

Add AUTO_INCREMENT value

ใส่ 'backquote' ให้กับชื่อตารางและฟิลด์

Add into comments

Creation/Update/Check dates

ข้อมูล

คำสั่ง INSERT เดิมรูปแบบ

แทรกหลายระบบในคราวเดียว

Use delayed inserts

Export type: INSERT ▼

ส่งมาเป็นไฟล์

รูปแบบของชื่อไฟล์ : (จำรูปแบบ)*

บันทึกข้อมูล

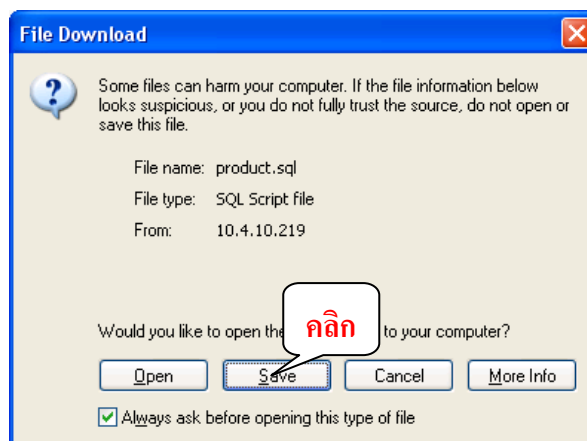
ไม่มี

รูปที่ 2.26 หน้าจอสำหรับตั้งค่าต่างๆ ของการ Export

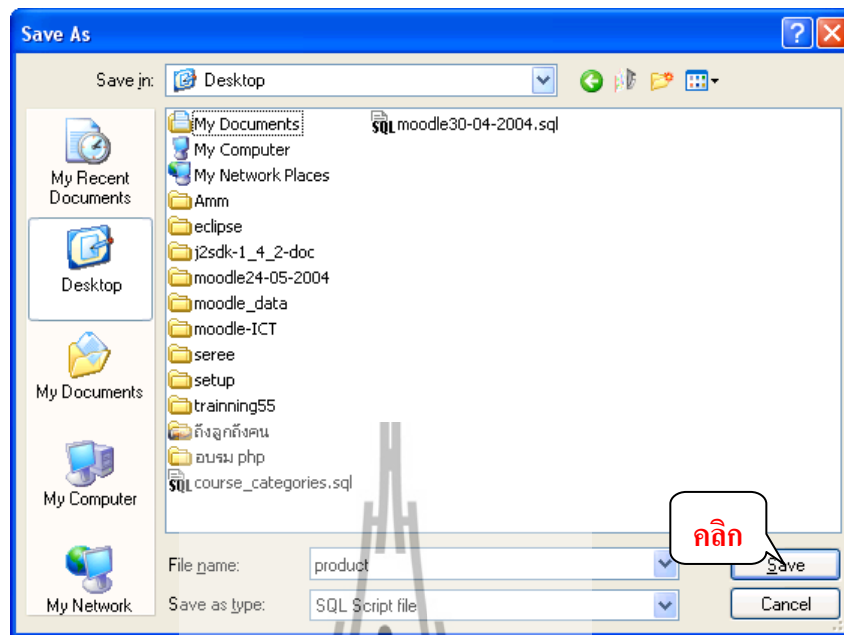
จากรูปที่ 2.26 เป็นหน้าจอสำหรับการตั้งค่าต่างๆ ของการ Export ซึ่งประกอบไปด้วย ส่วนต่างๆ ดังนี้

- ส่งออก ใช้สำหรับเลือกชนิดของข้อมูลที่จะส่งออก เช่น SQL, XML เป็นต้น
- โครงสร้าง
 - เพิ่มคำสั่ง drop table ใช้ในกรณีที่ต้องการให้ลบตารางเก่าก่อนสร้างตารางใหม่ ในขณะที่นำไฟล์มา Import
 - Add AUTO_INCREMENT value ใช้เพิ่มข้อมูลประเภท auto_increment ในขณะที่สร้างตาราง ในกรณีที่น่าไฟล์มา Import
 - ใส่ backquote ให้กับชื่อตารางและฟิลด์
- ข้อมูล
 - คำสั่ง INSERT เต็มรูปแบบ
 - แทรกหลายระเบียนในคราวเดียว
 - Export Type เป็นการเลือกเฉพาะชนิดของคำสั่งที่ต้องการ ได้แก่ INSERT UPDATE และ REPLACE
- ส่งมาเป็นไฟล์ ให้คลิกที่ checkbox ด้านหน้าด้วย
- ใส่รูปแบบของชื่อไฟล์ เป็นการใส่ชื่อไฟล์ที่ต้องการ

เมื่อกำหนดค่าการ Export ต่างๆ เรียบร้อยแล้ว ให้คลิกที่ ลงมือ จะปรากฏหน้าต่างสำหรับเลือกที่จะบันทึกไฟล์ หรือเปิดไฟล์ ให้คลิกที่ SAVE ดังรูปที่ 2.27 หลังจากนั้นเลือก location ภายในเครื่องเพื่อแจ้งให้โปรแกรมทราบว่าจะเซฟไฟล์ไว้ที่ใด เมื่อเรียบร้อยแล้วคลิกที่ SAVE เพื่อบันทึกไฟล์ข้อมูลลงเครื่องดังรูปที่ 2.28



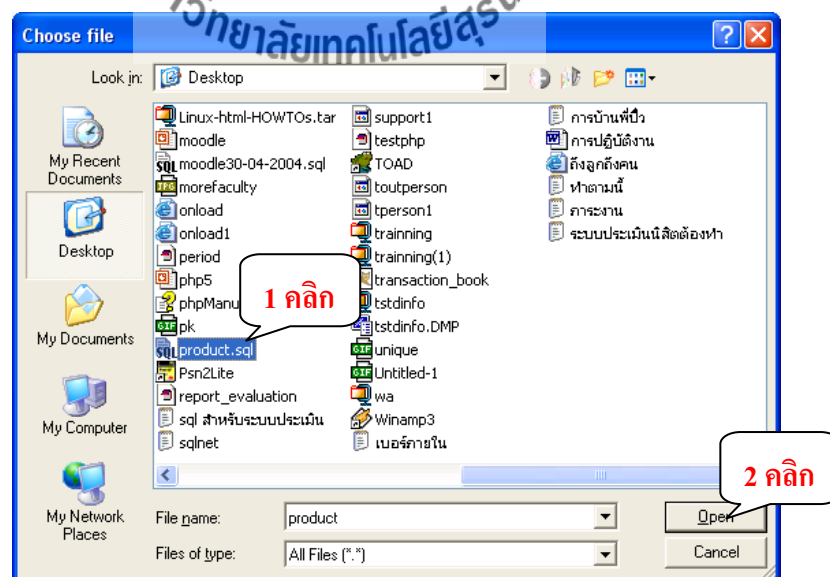
รูปที่ 2.27 หน้าต่างสำหรับเลือกบันทึกไฟล์



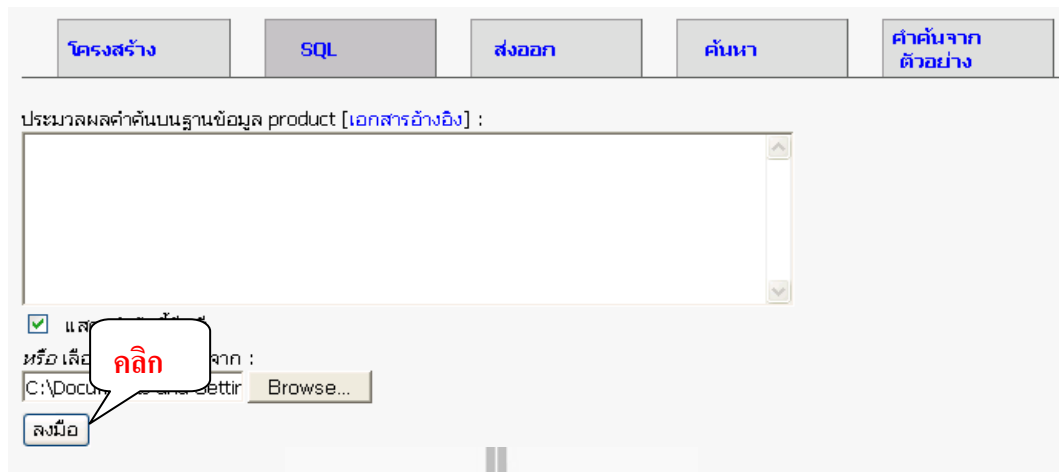
รูปที่ 2.28 เลือก location ในเครื่องที่ต้องการเซฟไฟล์ลงไป

10. การ Import ข้อมูล

การ Import ข้อมูล จะทำงานผ่านเมนู SQL เมื่ออยู่ในหน้าจอ SQL แล้วให้คลิกที่ **Browse** เพื่อเลือกไฟล์ Export ที่ต้องการจากนั้นคลิกที่ ลงมือ เพื่อนำข้อมูลในไฟล์ Import ลงยังฐานข้อมูล การ Import ข้อมูลแสดงดังรูปที่ 2.29 และ 2.30



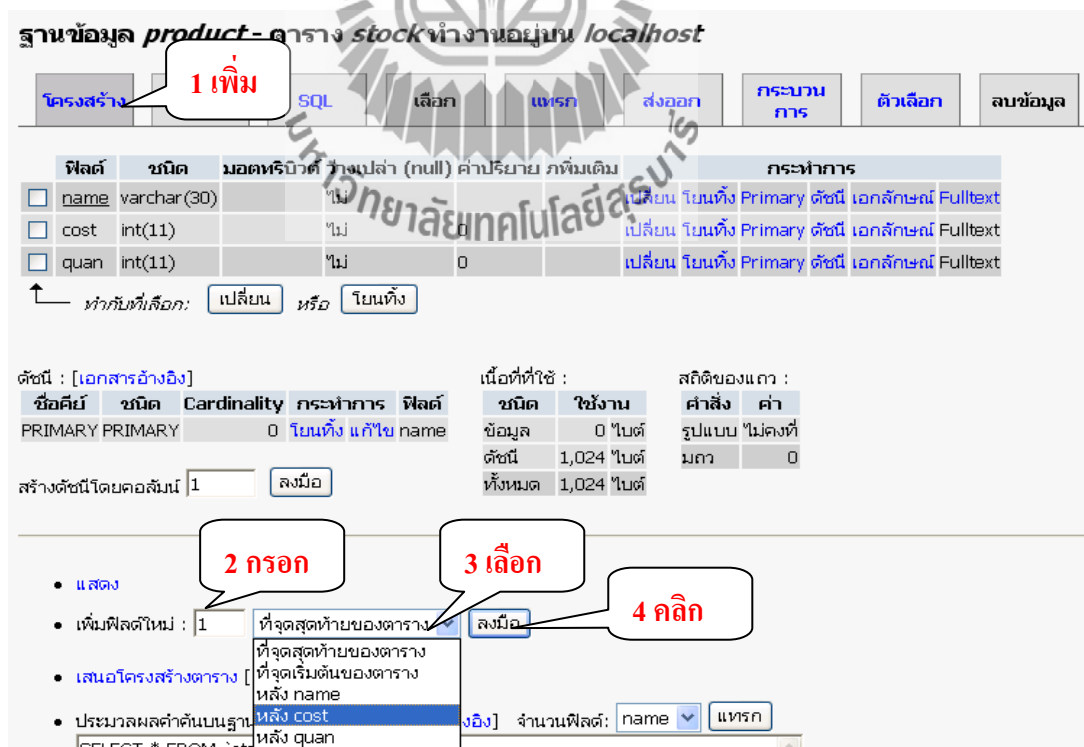
รูปที่ 2.29 เลือกไฟล์ Export ที่ต้องการ



รูปที่ 2.30 การ Import ไฟล์ข้อมูลลงฐานข้อมูล (ต่อ)

11. การเพิ่มฟิลด์ใหม่ลงไปในตารางที่มีอยู่แล้ว

การเพิ่มฟิลด์ใหม่ลงในตารางที่มีอยู่แล้ว จะดำเนินการผ่านเมนู **โครงสร้าง** ชั้นแรกให้คลิกที่เมนูโครงสร้าง จะพบส่วนของการเพิ่มฟิลด์ใหม่ ซึ่งต้องระบุจำนวนฟิลด์ที่ต้องการเพิ่ม และ ระบุว่า จะเพิ่มฟิลด์ใหม่ ไว้หน้าหรือหลังฟิลด์ใดในตาราง การเพิ่มฟิลด์ใหม่แสดงดังรูปที่ 2.31 และ 2.32



รูปที่ 2.31 ขั้นตอนการเลือกเพื่อเพิ่มฟิลด์ใหม่

ฐานข้อมูล *product* - ตาราง *stock* ทำงานอยู่บน *localhost*

ฟิลด์	ชนิด [หรืออ้างอิง]	ความยาว/ เขต*	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย
venderid	VARCHAR	10		not null	
pid	VARCHAR	10		not null	

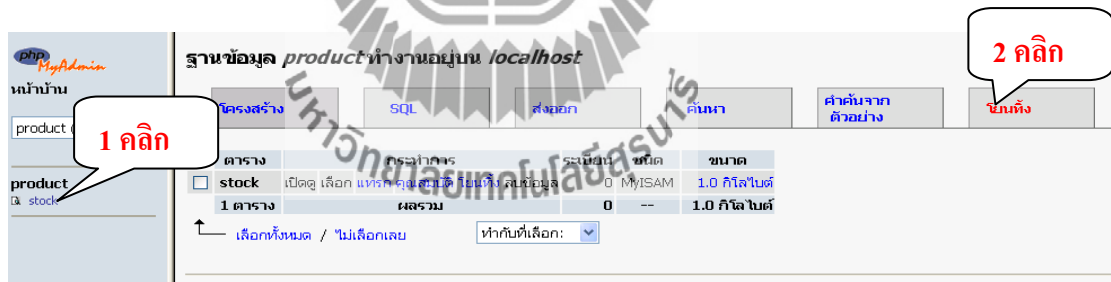
บันทึก 2 คลิก

รูปที่ 2.32 การใส่รายละเอียดฟิลด์ใหม่

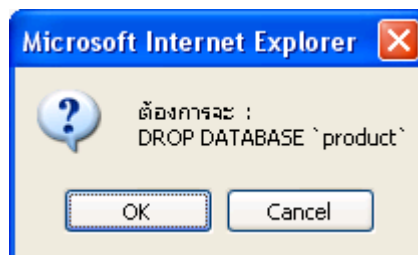
จากรูปที่ 2.32 เป็นหน้าจอสำหรับใส่รายละเอียดของฟิลด์ใหม่ โดยผู้ใช้ต้องกรอกชื่อฟิลด์ ชนิดของข้อมูลที่จะเก็บ ความยาว และค่าอื่นๆ เมื่อกรอกข้อมูลเรียบร้อยแล้วคลิกที่ บันทึก เพื่อเพิ่มฟิลด์ใหม่ลงยังตาราง

12. การลบฐานข้อมูล (Drop Database)

การลบฐานข้อมูลทำได้โดยคลิกเลือกที่ชื่อฐานข้อมูลจากเมนูทางด้านซ้ายของหน้าจอก่อน จากนั้นคลิกเลือกที่ โยนทิ้ง โปรแกรมจะแสดงข้อความยืนยันก่อนลบฐานข้อมูล แสดงดังรูปที่ 2.33 และ 2.34



รูปที่ 2.33 ขั้นตอนการเลือกเพื่อลบฐานข้อมูล

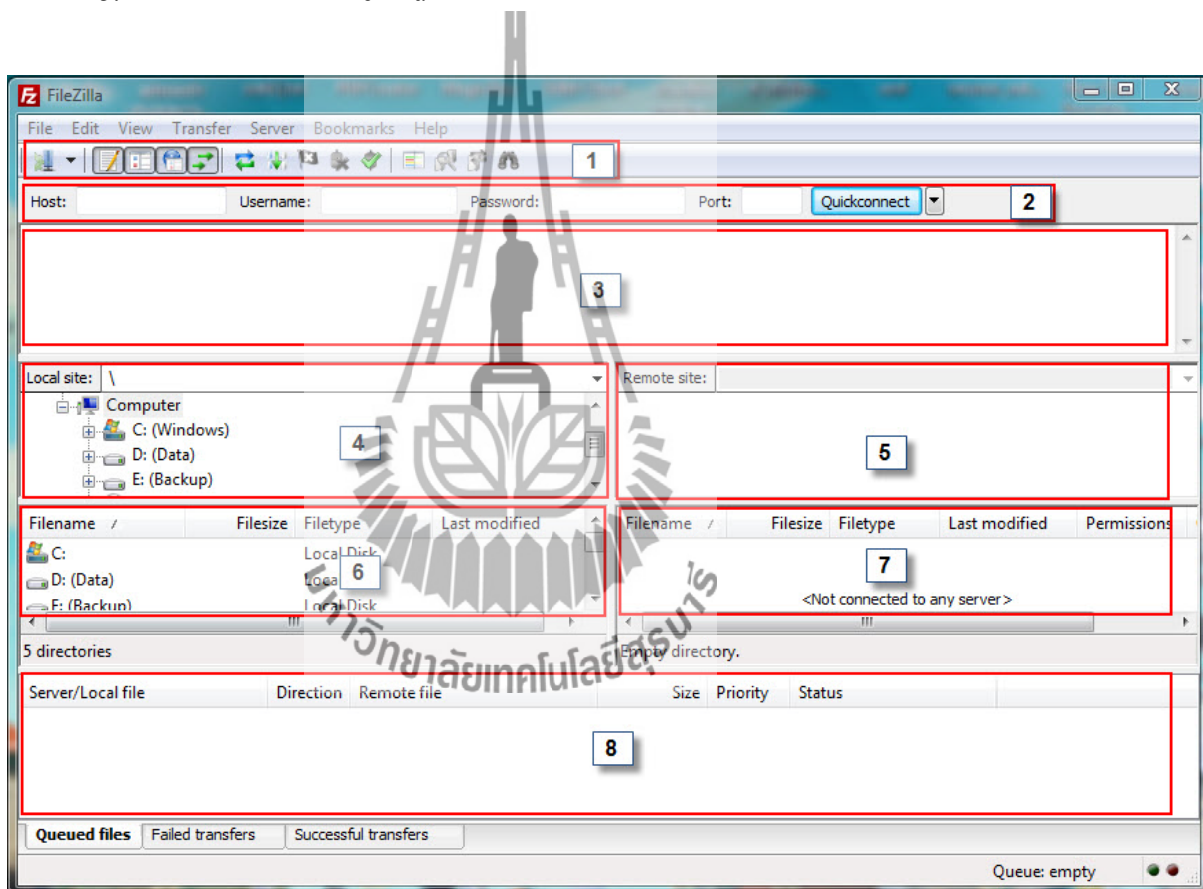


รูปที่ 2.34 ยืนยันการลบฐานข้อมูล

2.9 โปรแกรม FileZilla

โปรแกรม FileZilla เป็นโปรแกรม FTP Client คือโปรแกรมสำหรับรับส่งข้อมูลไปยัง Server ซึ่ง FileZilla เป็นโปรแกรม OpenSource ที่สามารถนำมาใช้งานได้ฟรี

1. สามารถดาวน์โหลดโปรแกรมได้ที่ <http://filezilla-project.org>
2. หน้าเว็บจะมีสองแบบคือ FileZilla Client และ FileZilla Server ให้โหลดตัว Client
3. เมื่อโหลดโปรแกรมเรียบร้อยแล้ว ให้ทำการติดตั้งโปรแกรม
4. หลังจากการติดตั้งโปรแกรม มีไอคอนของ FileZilla ที่ Desktop ดังรูป
5. องค์ประกอบของ FileZilla แสดงดังภาพด้านล่าง

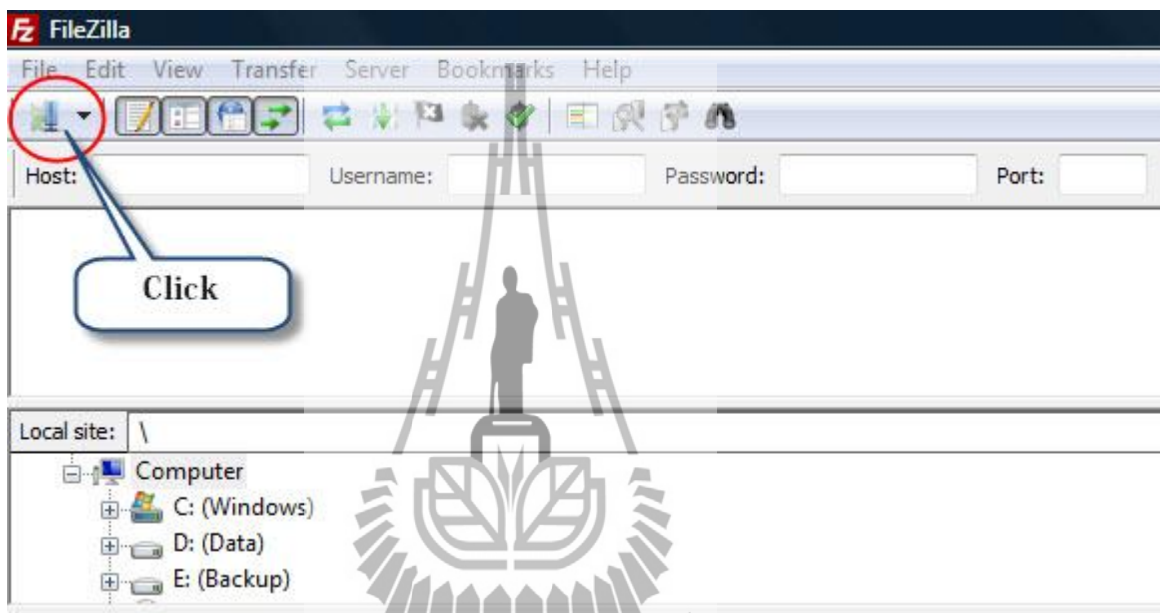


รูปที่ 2.35 โปรแกรม มีไอคอนของ FileZilla ที่ Desktop

- 5.1. General Toolbar -- เมนูทั่วไป
- 5.2. Quick Connect -- ล็อกอินแบบรวดเร็ว
- 5.3. Server Information -- แสดงข้อมูลจาก Server
- 5.4. Local Site Folder Tree -- แสดงโฟลเดอร์ในเครื่อง

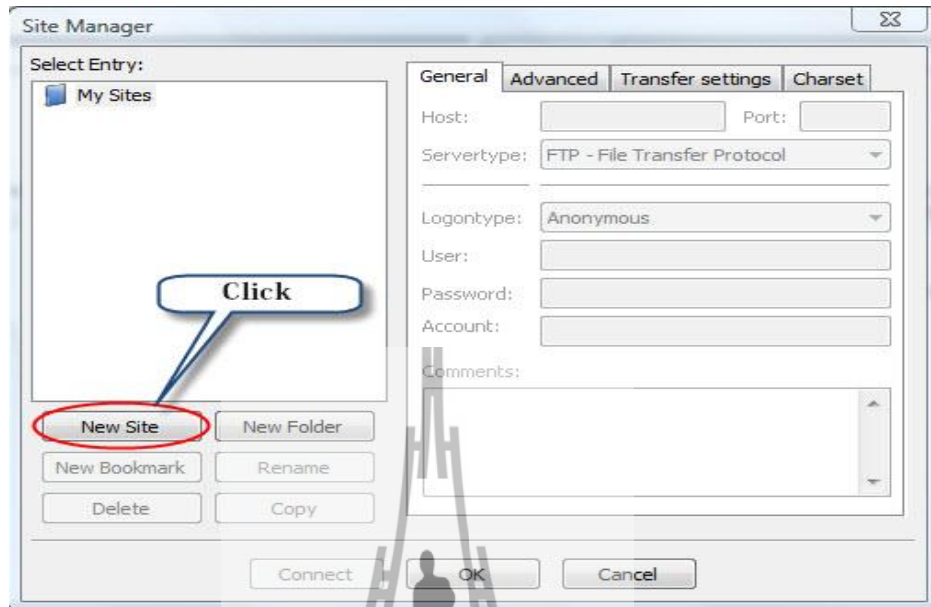
- 5.5. Remote Site Folder Tree -- แสดงโฟลเดอร์ที่อยู่ใน Server
- 5.6. Local Site Files -- แสดงไฟล์ในโฟลเดอร์ที่เลือกไว้
- 5.7. Remote Site Files -- แสดงไฟล์ในโฟลเดอร์ที่อยู่ใน Server ที่เลือกไว้
- 5.8. Queue Files -- แสดงรายชื่อไฟล์ที่จะ Upload / Download

6. ให้ทำการ เปิดโปรแกรม FileZilla จากนั้นคลิก ไอคอนรูปเซิร์ฟเวอร์มุมซ้ายบน



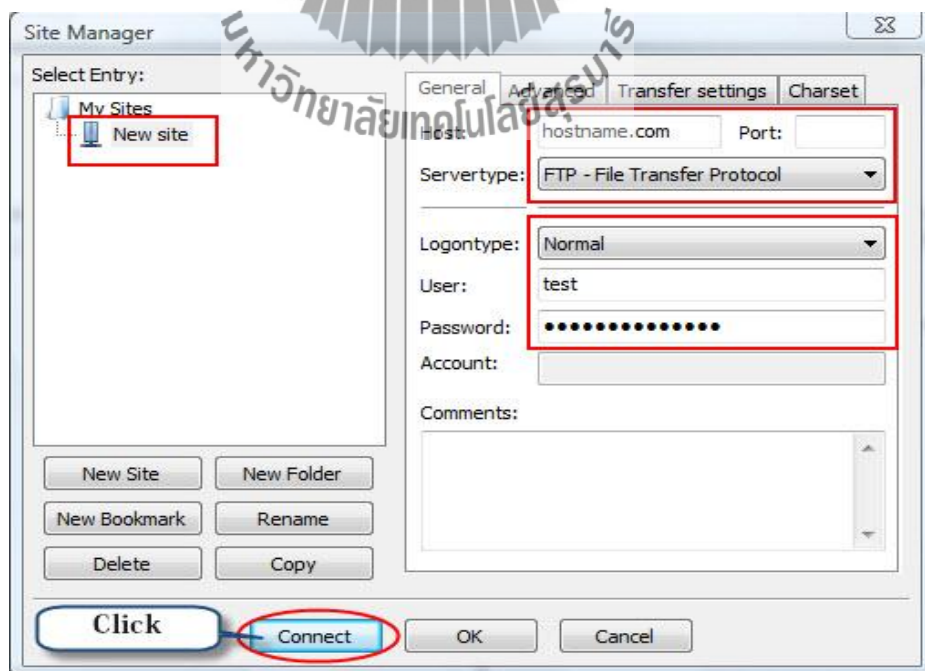
รูปที่ 2.36 การเปิดโปรแกรม FileZilla

7. จากนั้นให้เลือก New Site



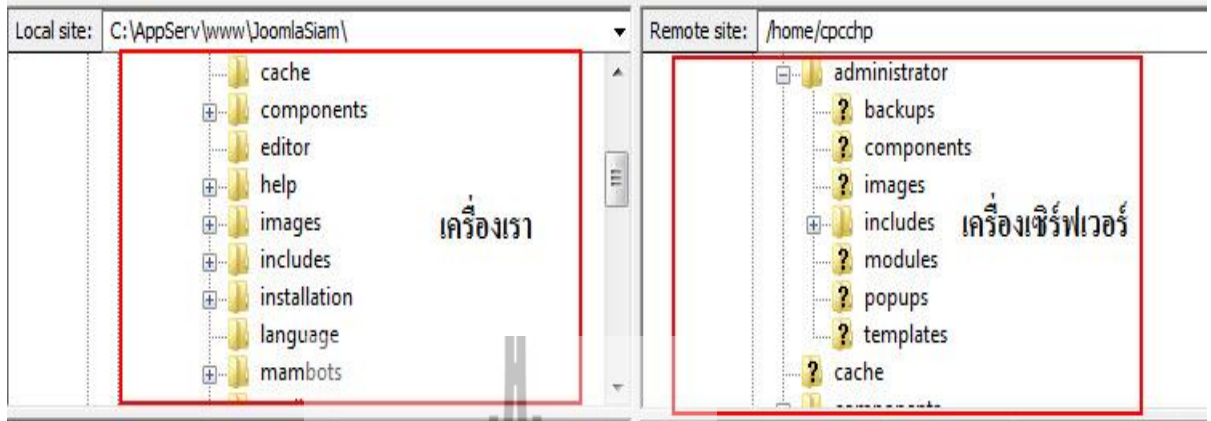
รูปที่ 2.37 การสร้าง New Site

8. ช่อง Host ให้ใส่ ftp.yourname.com (hostname.com คือ ชื่อเว็บไซต์), Port :21



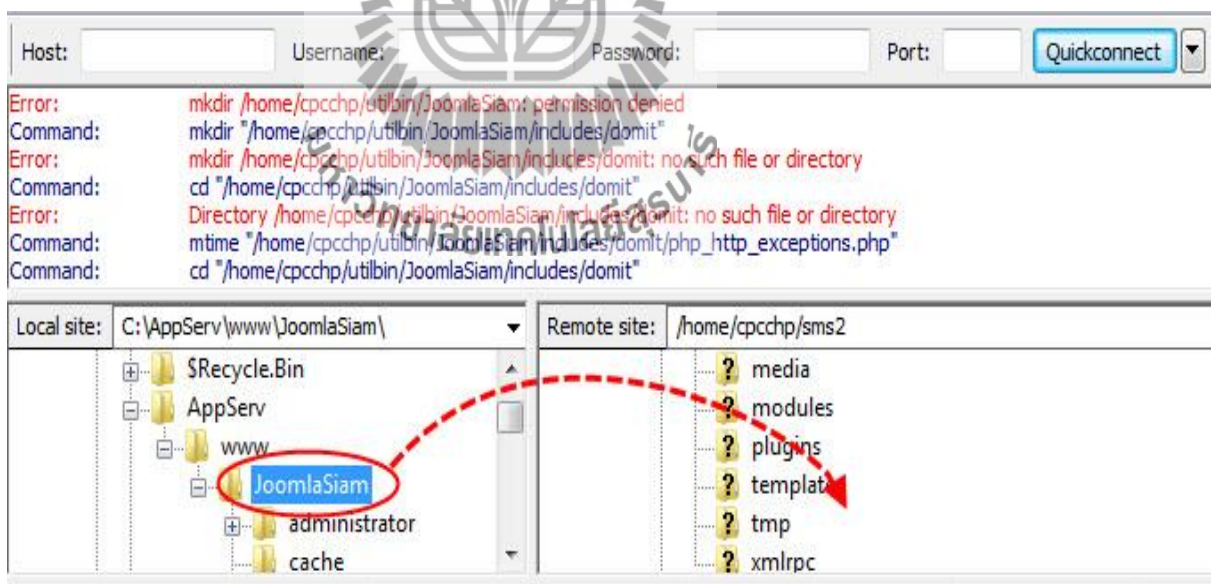
รูปที่ 2.38 การใส่ Host

9. ฟังซ้ายคือ ข้อมูลในเครื่องของเรา ส่วนฝั่งขวา คือ server ที่เราเชื่อมต่อ



รูปที่ 2.39 แสดงไฟล์

10. เริ่มทำการ Upload ไฟล์ โดย ลาก (drag) ไฟล์ที่ต้องการจากหน้าต่างด้านซ้าย (Local Site) ไปวาง (drop) ไว้ในหน้าต่างทางด้านขวา (Remote Site)



รูปที่ 2.40 การ Upload ไฟล์

11. หลังจากการ Upload ไฟล์เสร็จเรียบร้อยแล้วให้คลิกที่ปุ่ม Disconnect เพื่อจบการทำงาน (เพื่อความปลอดภัย กรุณา Disconnect ทุกครั้งหลังใช้งาน)



รูปที่ 2.41 | คลิกที่ปุ่ม Disconnect

2.10 รายละเอียดการจัดการจัดสรรพื้นที่เว็บไซต์ <http://student.sut.ac.th/rfidproject>

- 1) สำหรับการ ftp เพื่อ upload ไฟล์ขึ้น server
 - สามารถเรียกโดยตรงจาก Browser ได้เลย
 - กรณีเรียกผ่าน Browser โดยตรง เรียกดังนี้ <ftp://student.sut.ac.th>
 - พื้นที่ quota สำหรับการใช้งาน 300 Mb

2) การใช้งานฐานข้อมูล (สำหรับสร้างเว็บที่ใช้ฐานข้อมูล)

หากต้องการสร้างฐานข้อมูล จะต้องใช้ tool ช่วย เป็น phpmyadmin (มีให้แล้ว) เข้าไปใช้ได้ที <http://203.158.4.167/phpMyAdmin>

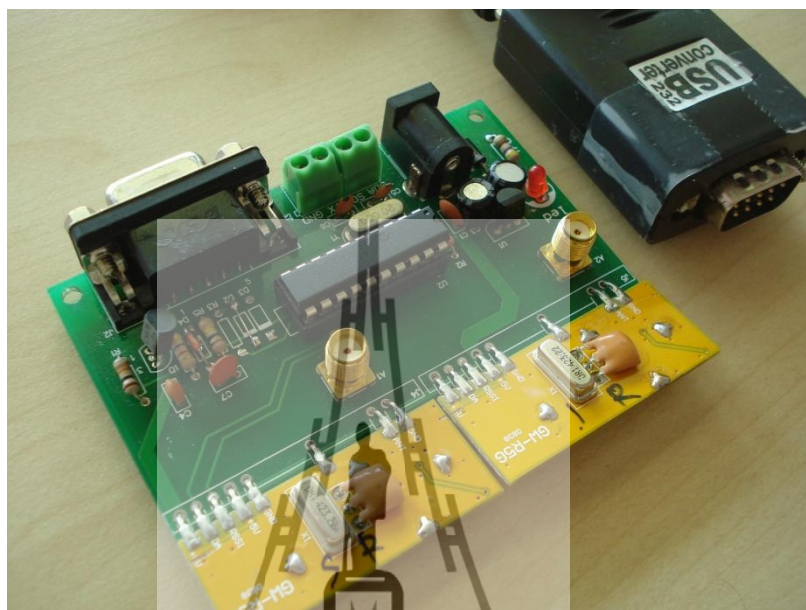
- host : **203.158.4.167** (ไม่ใช่ localhost) ฐานข้อมูลจะอยู่บน server อีกตัวเพื่อรองรับระบบ failover

- Database name : **b5011742db**

ซึ่งสามารถ create / insert / update / delete /drop เฉพาะฐานข้อมูลตัวนี้เท่านั้น ไม่สามารถสร้างฐานข้อมูลอื่น ๆ

2.11 คุณสมบัติของอุปกรณ์ฮาร์ดแวร์ในโรงงาน

2.11.1 คุณสมบัติของบอร์ด RF2315R Active RFID Dual Receiver Module with RSSI USB Version



รูปที่ 2.42 ภาพบอร์ด RF2315R Active RFID Dual Receiver Module with RSSI USB Version

จากภาพเครื่องรับ RFID รุ่น RF2315R เป็นโมดูลที่ใช้งานกับเครื่องรับ ซึ่งจะได้รับข้อมูลที่ส่งมาจากโมดูล RF8315T โดยโมดูล RF2315R จะรายงานค่า RSSI ของบอร์ดโมดูล RF8315T ผ่านทางพอร์ตอนุกรม ซึ่งบอร์ดจะมีขั้ว SMA 2 ขั้ว สำหรับการเชื่อมต่อเป็น สายอากาศ โดยเราสามารถใส่ค่า RSSI เพื่อที่จะกำหนดหรือหาตำแหน่งของวัตถุได้ ซึ่งมีคุณสมบัติดังนี้

- สามารถเสียบสาย USB แล้วใช้งานได้ทันที โดยไม่ต้องติดตั้งโปรแกรมซอฟต์แวร์ของอุปกรณ์
- สามารถอ่าน RFID Active Tag รุ่น RF8315T ได้ทุกๆ Tag
- ข้อมูลของ RSSI ทั้งสองข้อมูล จะรายงานออกมาซึ่งสามารถใช้ในการกำหนดตำแหน่งของวัตถุ
- มี TTL RS232 สำหรับเชื่อมต่อโดยตรงกับ MCU อื่น ๆ หรือวงจรอื่น ๆ ได้
- มี I2C ที่สามารถติดต่อสั่งงานและควบคุมภายใต้สายสัญญาณเพียง 2 เส้น เส้นหนึ่งคือสายข้อมูล อีกเส้นหนึ่งคือ สายสัญญาณนาฬิกา

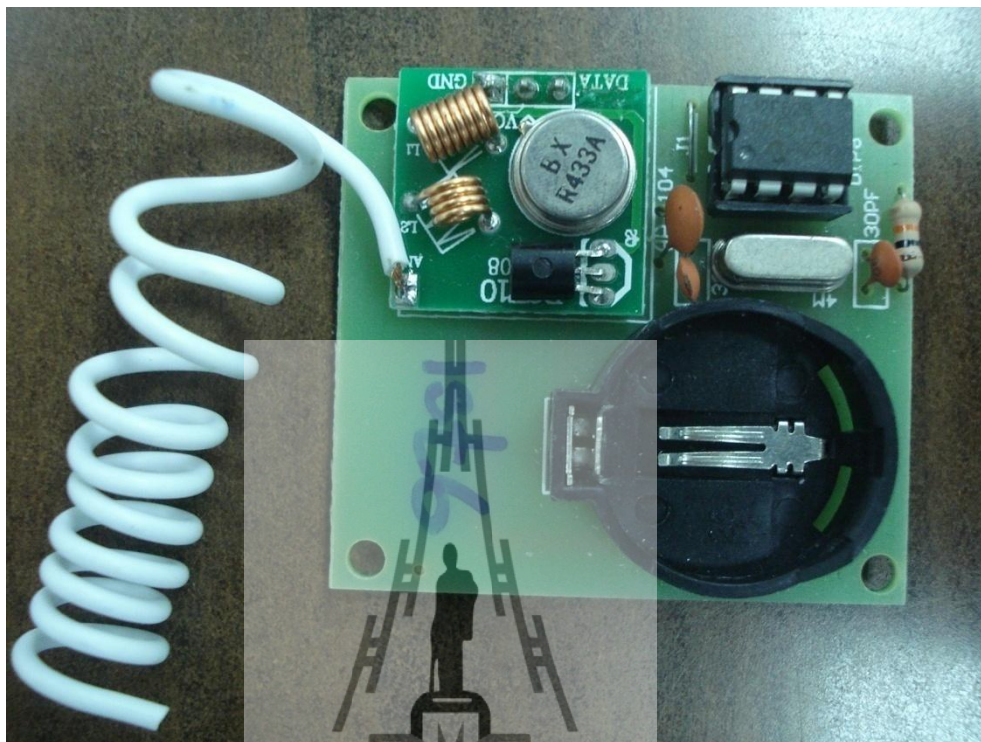
- มีการออกแบบเพื่อให้มีระบบการใช้งานความถี่ มีความเสถียรมากที่สุด
- มีการป้องกันไฟย้อนกลับ

ข้อมูลเฉพาะของเครื่องรับ RF2315R Active RFID Dual Receiver Module with RSSI USB Version

Supply Voltage	9VDC to 12VDC
Supply Current	10mA Typical
อุณหภูมิในการทำงาน	0 – 50 °C
คลื่นความถี่	315 MHz (433 MHz is also available)
Data Output	(4 characters ID, RSSIA,RSSIB)
ความจุ	40 ID ในเวลาเดียวกัน
Build-in Watchdog	2.3 seconds
USB PORT	
Type	RS232, 19200 Baud, 8 bit words,1 stop bit,1 start bit, no parity

ตารางที่ 2.1 ข้อมูลเฉพาะของเครื่องรับ RF2315R Active RFID Dual Receiver Module with RSSI USB Version

2.11.2 คุณสมบัติของบอร์ด RF8315T Active RFID 8 Meters Transmitting Module



รูปที่ 2.43 ภาพบอร์ด RF8315T Active RFID 8 Meters Transmitting Module

จะส่งตัวอักษรที่ไม่ซ้ำกัน 4 ตัวอักษร (AZ, az, 0-9) โดยจะถูกส่งออกในทุก 2.5 วินาทีบวก / ลบ 0.5 วินาที ตัวส่ง RF8315R ที่ตรงกันสามารถรับ ID ได้ภายใน 8 เมตรถึงแม้ว่ากำลังส่ง RF จะต่ำมาก เครื่องส่งสัญญาณจะส่งสัญญาณเมื่อสามารถส่ง ID ได้ (<0.01 วินาที) ซึ่งเป็นผลมาจากตัวมันเอง และจะไม่ทำให้ข้อมูลไปรบกวนยังอุปกรณ์อื่น ๆ ที่ใช้คลื่นความถี่เดียวกัน โดยมีคุณสมบัติดังนี้

- สิ้นเปลืองพลังงานน้อย (5,000 ชั่วโมงสำหรับ CR2025, และ 7,000 ชั่วโมงสำหรับ CR2032)
- สามารถส่งข้อมูลได้ถึง 8 เมตร หาก สายอากาศเป็น 9 เมตร รัศมีจะมีประสิทธิภาพมากขึ้น สามารถส่งข้อมูลได้ถึง 15 เมตร
- มีขนาดเล็ก
- ไม่จำเป็นต้องตั้งค่า
- ป้องกันการชนกัน ของข้อมูลอย่างเป็นขั้นตอน โดย RF8315R สามารถจัดการกับ 160 เครื่องส่งสัญญาณได้ในเวลาเดียวกัน

ข้อมูลเฉพาะของเครื่องส่ง RF8315T Active RFID 8 Meters Transmitting Module

Power Supply	CR2025 / CR2032
การใช้พลังงาน	ใช้ส่ง 4mA , เมื่อไม่ได้ใช้งาน 19uA
อุณหภูมิในการทำงาน	0 – 50 °C
คลื่นความถี่	315 MHz
Data Output	4 ตัวอักษร (A - Z, a- z, 0-9) เครื่องส่งสัญญาณทั้งหมดจะส่ง ID ที่ไม่ซ้ำกัน
รัศมีที่มีประสิทธิภาพ	8 เมตร โดยมี สายอากาศ (ขนาดเส้นผ่านศูนย์กลาง 8 มม. ยาว 2 ซม.) แต่ถ้ารัศมี 15 เมตรหาก สายอากาศเป็นสาย 9 นิ้ว
RF output power	< 2mW

ตารางที่ 2.2 ข้อมูลเฉพาะของเครื่องส่ง RF8315T Active RFID 8 Meters Transmitting Module



บทที่ 3

การออกแบบ การแสดงผลตำแหน่งของนักศึกษาภายในห้องเรียน โดยใช้ RFID ผ่านทาง Website

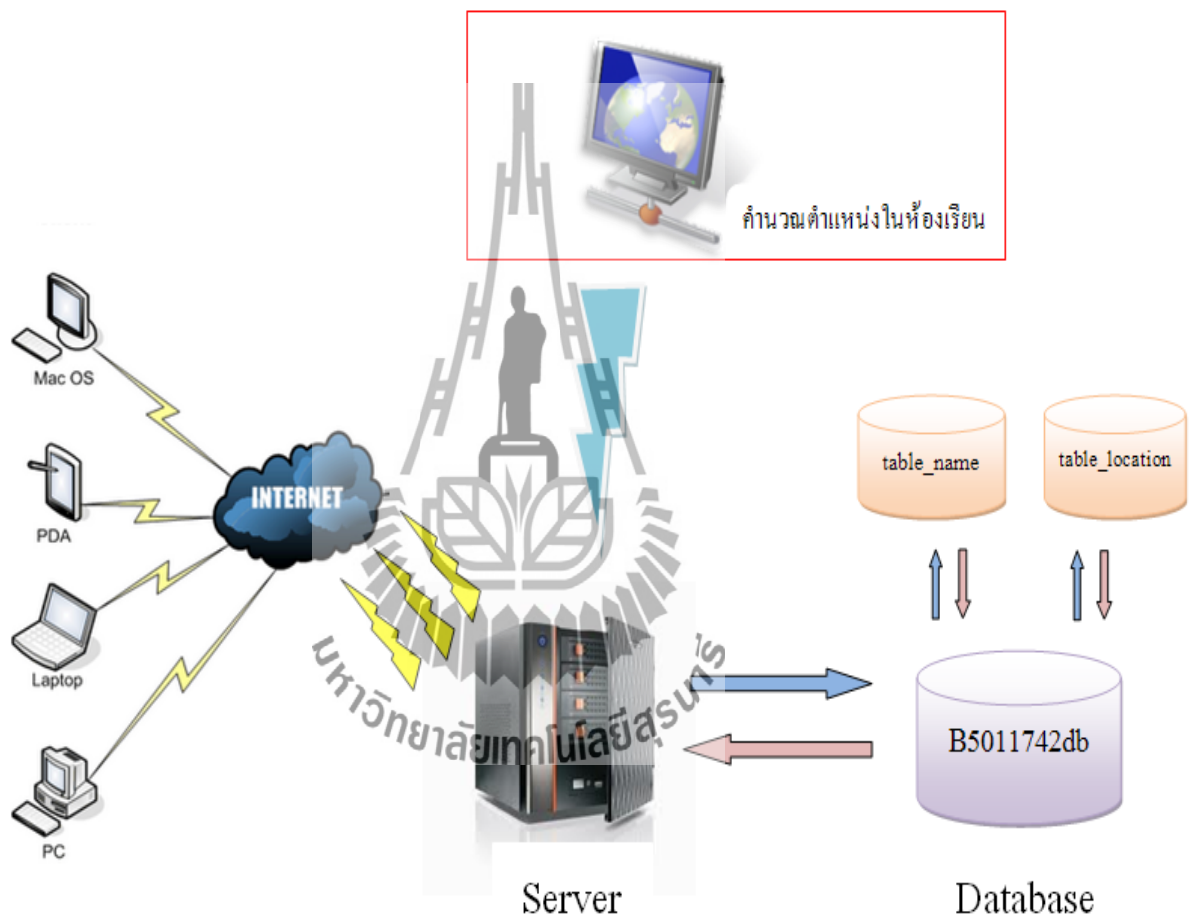
3.1 บทนำ

โครงการ การแสดงผลตำแหน่งของนักศึกษาภายในห้องเรียนโดยใช้ RFID ผ่านทาง Website การทำงานโครงการนี้เริ่มต้นด้วยการส่งค่าตำแหน่งของ tag และชื่อ tag ไปยัง server โดยมีการส่งค่าไปทุกๆ 10 วินาที ก่อนที่จะแสดงบนเว็บไซต์ ชื่อ tag จะต้องเปลี่ยนเป็นรหัส นักศึกษาตามที่ได้มีการบันทึกไว้ เพื่อที่จะแสดงบนเว็บไซต์ โดยมีขั้นตอนการออกแบบดังนี้

1. การสร้างฐานข้อมูล
2. การสร้างตาราง table_name
3. การสร้างตาราง table_location
4. การเขียนหน้าเว็บเพจ และไฟล์ติดต่อฐานข้อมูล
5. การ up load ไฟล์ขึ้น server และการ update ข้อมูล
6. การลงทะเบียนหรือการเก็บข้อมูลของผู้ใช้งานใหม่
7. ส่งชื่อ tag และตำแหน่ง (location) ไปยัง Server
8. แสดงตำแหน่งบนเว็บไซต์

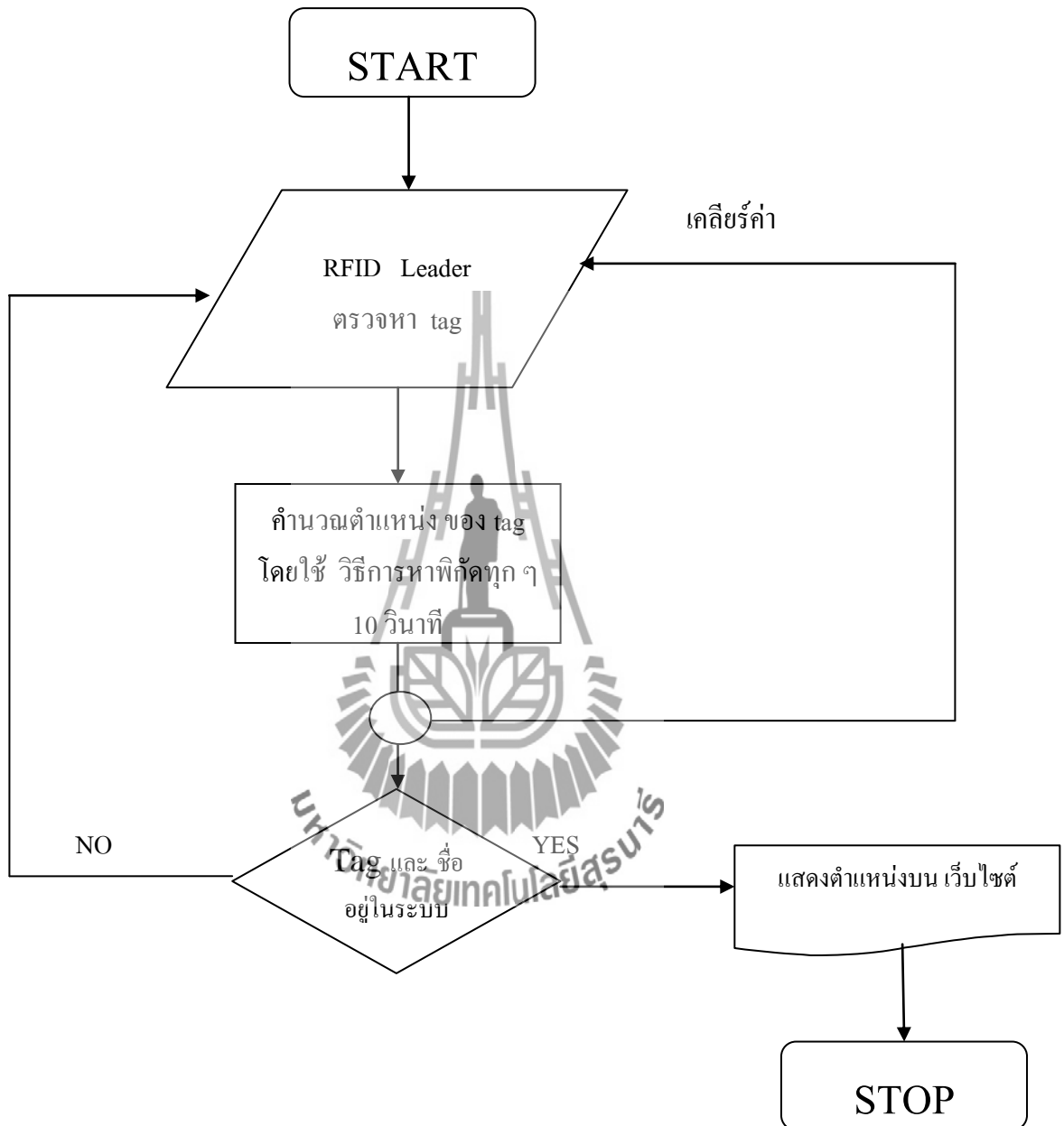
3.2 องค์ประกอบโดยรวมของระบบ

ในโครงการ การแสดงผลตำแหน่งของนักศึกษาภายในห้องเรียนโดยใช้ RFID ผ่านทาง Website จะประกอบไปด้วยส่วนการทำงานที่สำคัญ ดังแผนภาพดังนี้

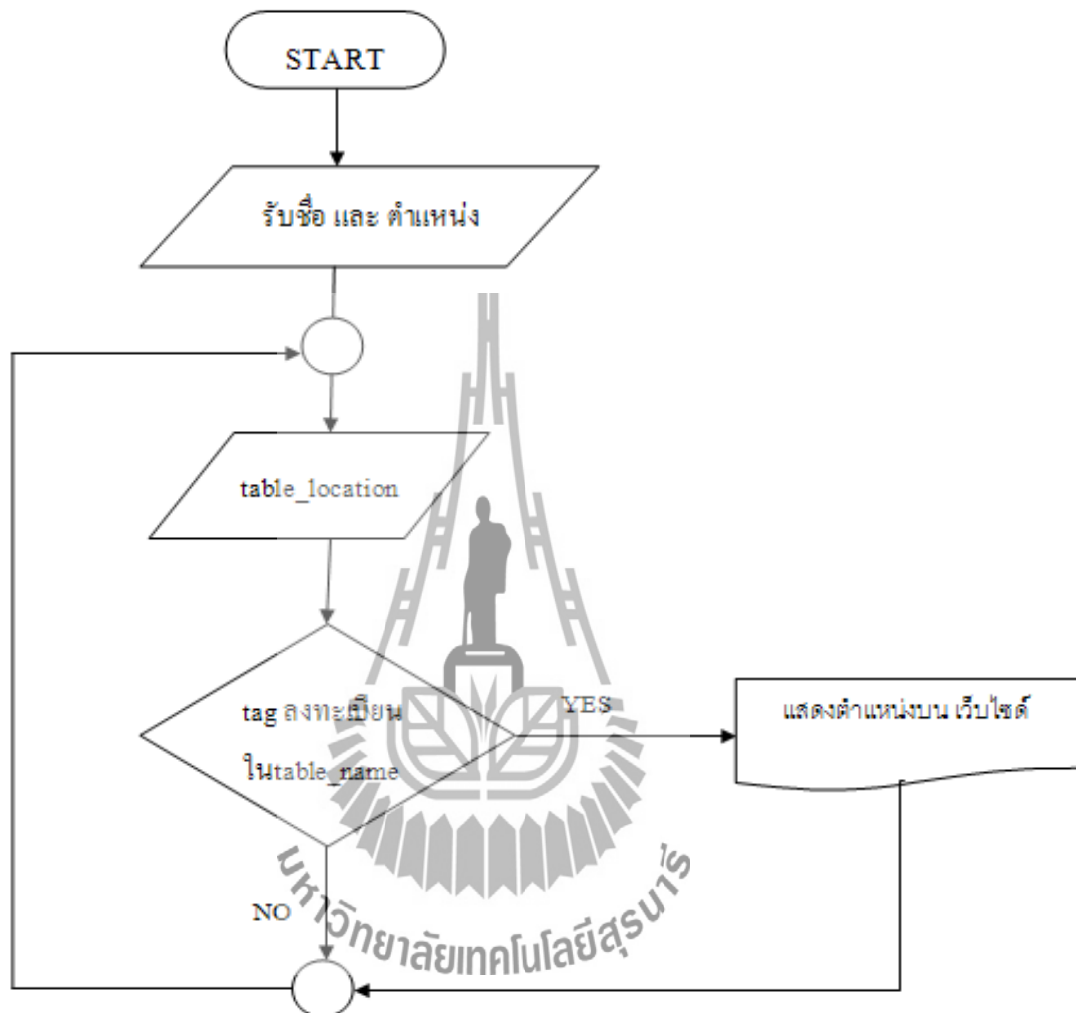


รูปที่ 3.1 องค์ประกอบโดยรวมของระบบ Server

3.3 การออกแบบซอฟต์แวร์



3.4 การทำงานบนฐานข้อมูล

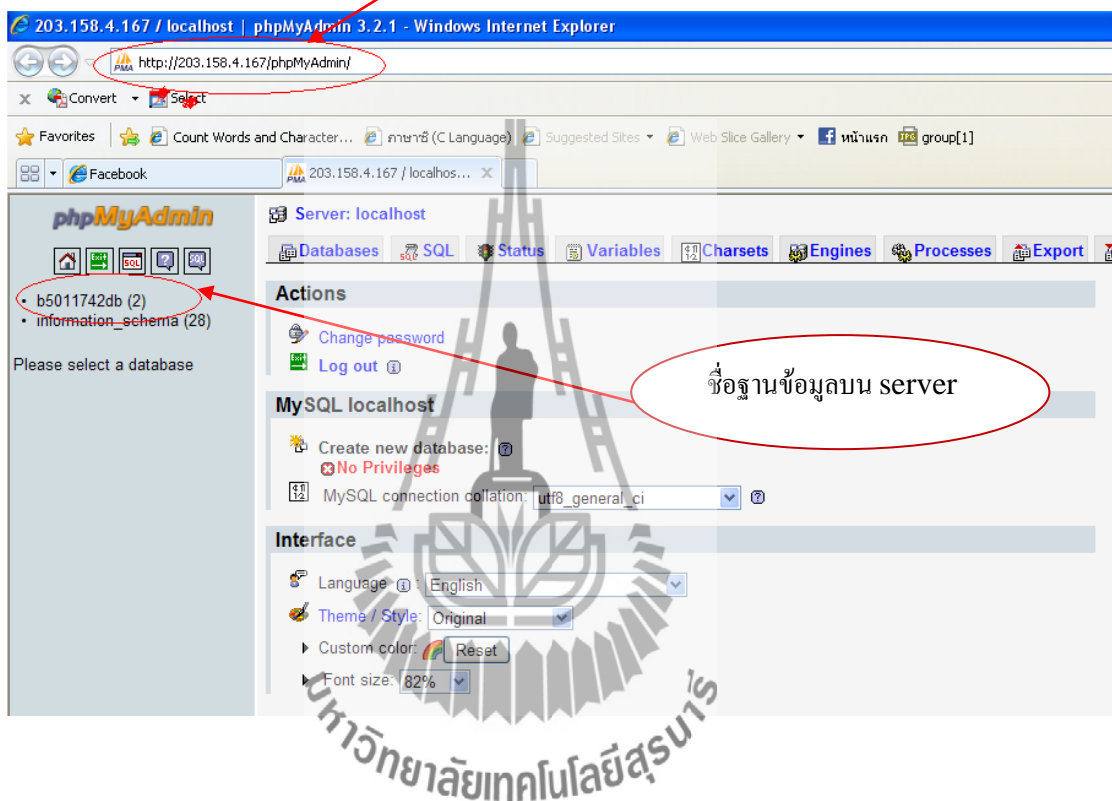


3.5 การออกแบบโปรแกรม

3.5.1 การสร้างฐานข้อมูล โดยใช้ phpMyAdmin

- 1) สร้างฐานข้อมูล ชื่อว่า b5011742db โดยเข้าไปที่

<http://203.158.4.167/phpMyAdmin>

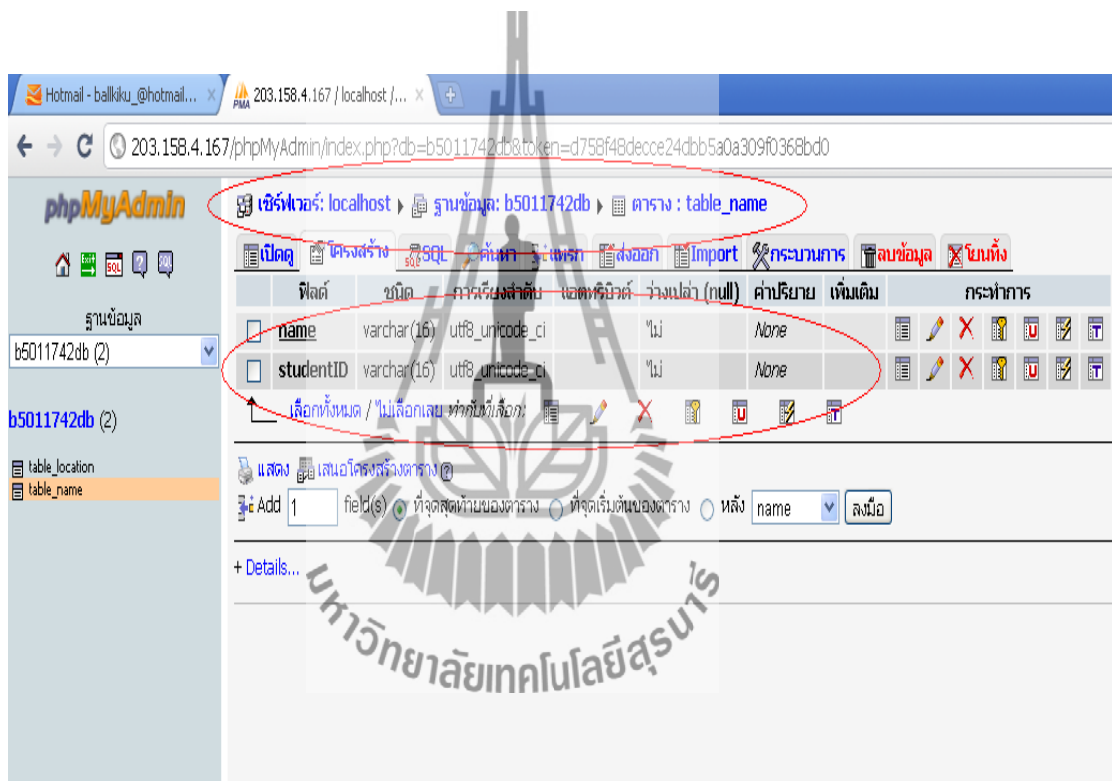


รูปที่ 3.2 ฐานข้อมูล b5011742db

จากรูปที่ 3.2 เมื่อลงชื่อเข้าใช้ใน <http://203.158.4.167/phpMyAdmin> จะเจอกับหน้าฐานข้อมูลนี้ เนื่องจากได้ขอใช้พื้นที่เว็บไซต์ไว้กับทางศูนย์คอมพิวเตอร์ของมหาวิทยาลัย และได้กำหนดให้ใช้ชื่อฐานข้อมูลว่า b5011742db ซึ่งเราไม่สามารถสร้างฐานข้อมูลอื่นๆ ได้อีกนั่นคือการที่เราได้สร้างฐานข้อมูลไว้บน server ซึ่งสามารถเข้าไปใช้งาน create / insert / update / delete / drop เฉพาะฐานข้อมูลตัวนี้เท่านั้น

3.5.2 สร้างตาราง ชื่อว่า table_name

เมื่อนักศึกษาแต่ละคนมี tag ประจำตัว ต้องนำมาเก็บข้อมูลของนักศึกษา และชื่อ tag ไว้ในฐานข้อมูล (จะเหมือนบัตรประจำตัว) โครงสร้างของตาราง table_name ประกอบไปด้วย ตารางไว้เก็บ ชื่อ (name) และตารางไว้เก็บ รหัสนักศึกษา (studentID) ตามรูปที่ 3.3 และในโปรแกรมได้กำหนดให้โปรแกรมรับข้อมูลที่ชื่อ tag ได้ลงทะเบียนไว้ในช่อง name และให้แสดงข้อมูลบนเว็บไซต์โดยชื่อที่แสดงบนเว็บไซต์เป็นข้อมูลในช่อง studentID



รูปที่ 3.3 โครงสร้างของตาราง table_name

Server: localhost Database: b5011742db Table: table_name

Showing rows 0 - 2 (3 total, Query took 0.0002 sec)

```
SELECT *
FROM `table_name`
LIMIT 0 , 30
```

	name	studentID
<input type="checkbox"/>	1st6	B5011742
<input type="checkbox"/>	1suL	B5046010
<input type="checkbox"/>	1szI	B5015894

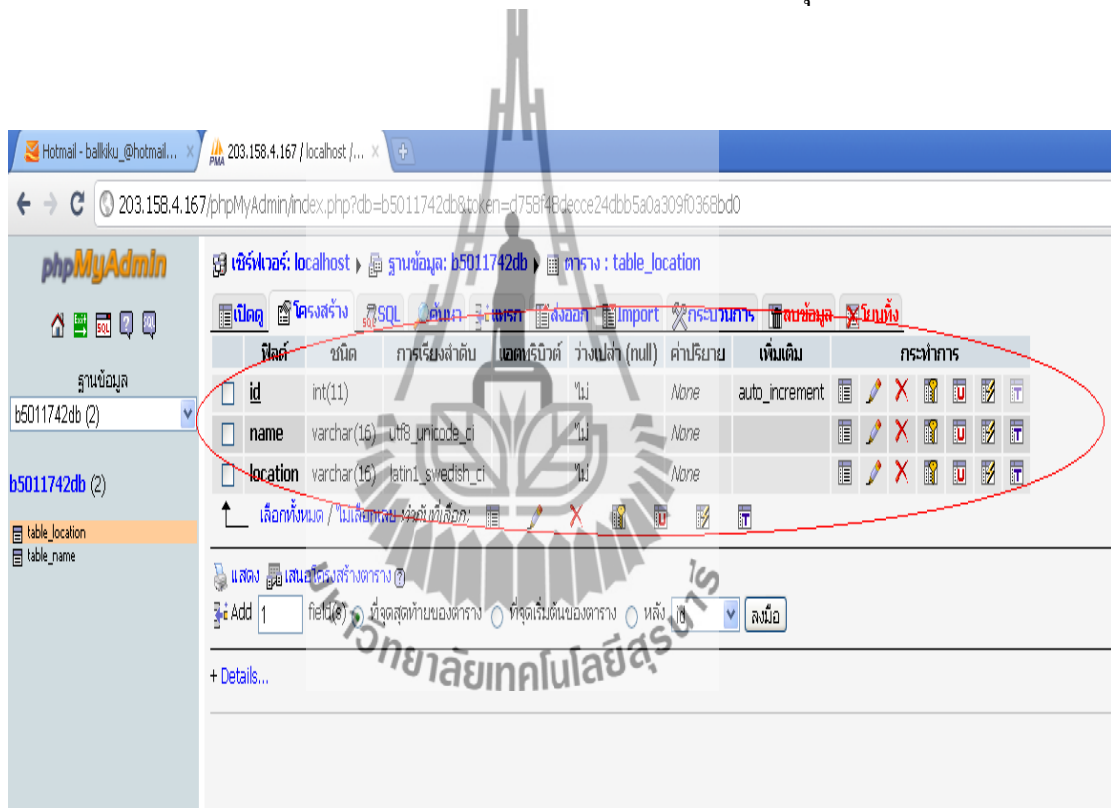
รูปที่ 3.4 ตัวอย่างข้อมูลในตาราง table_name

จากรูปที่ 3.4 มีการลงทะเบียน tag ไว้ 3 ตัว โดยแต่ละตัวจะมีรหัสนักศึกษาแทนชื่อ tag เมื่อโปรแกรมคำนวณตำแหน่งของ 1 st6 บนเว็บไซต์จะแสดงตำแหน่งที่มีการวาง tag 1st6 ด้วย B5011742 หากโปรแกรมคำนวณตำแหน่งของ 1suL บนเว็บไซต์จะแสดงตำแหน่งที่มีการวาง tag 1suL ด้วย B5046010 และถ้าโปรแกรมคำนวณตำแหน่งของ 1szI บนเว็บไซต์จะแสดงตำแหน่งที่มีการวาง tag 1szI ด้วย B5015894

3.5.3 สร้างตาราง ชื่อว่า table_location

ตาราง table_location เป็นตารางที่รอการส่งผลการคำนวณมาจากโปรแกรมว่า tag อยู่ตำแหน่งไหน และนำค่ามาเก็บไว้ในตารางก่อนที่จะแสดงบนเว็บไซต์ โครงสร้างของตาราง table_location ประกอบไปด้วย ตารางไว้เก็บ ลำดับ (id) ตารางไว้เก็บ ชื่อ (name) และ ตารางไว้เก็บ โดเมน (location) ข้อมูลที่อยู่ในตารางนี้จะแสดงผลบนเว็บไซต์ ตามรูปที่ 3.5

ตาราง table_location สามารถเก็บค่าตำแหน่งที่เป็นปัจจุบันได้เนื่องจากจะมีการอัปเดตข้อมูลทุก ๆ ที่มีการส่งค่ามาจากโปรแกรมคำนวณ (10 วินาที) หากมีข้อมูลไปอยู่ที่ตำแหน่งเดียวกันบนเว็บไซต์ เว็บไซต์จะแสดงผลเป็นตำแหน่งที่มีการอัปเดตล่าสุด



รูปที่ 3.5 โครงสร้างของตาราง table_name

The screenshot shows the phpMyAdmin interface for a database named 'b5011742db'. The selected table is 'table_location'. The SQL query executed is: `SELECT * FROM `table_location` LIMIT 0, 30`. The table view displays the following data:

id	name	location
16	1st6	40
17	1suL	30
34	1szI	75

รูปที่ 3.6 ตัวอย่างข้อมูลในตาราง table_location

จากรูปที่ 3.6 ในตารางจะมีการเก็บค่าไว้ 3 ค่าใน table_location แสดงว่าโปรแกรมมีการคำนวณได้ทั้ง 3 ตัว tag สามารถสรุปจากตารางได้ว่า

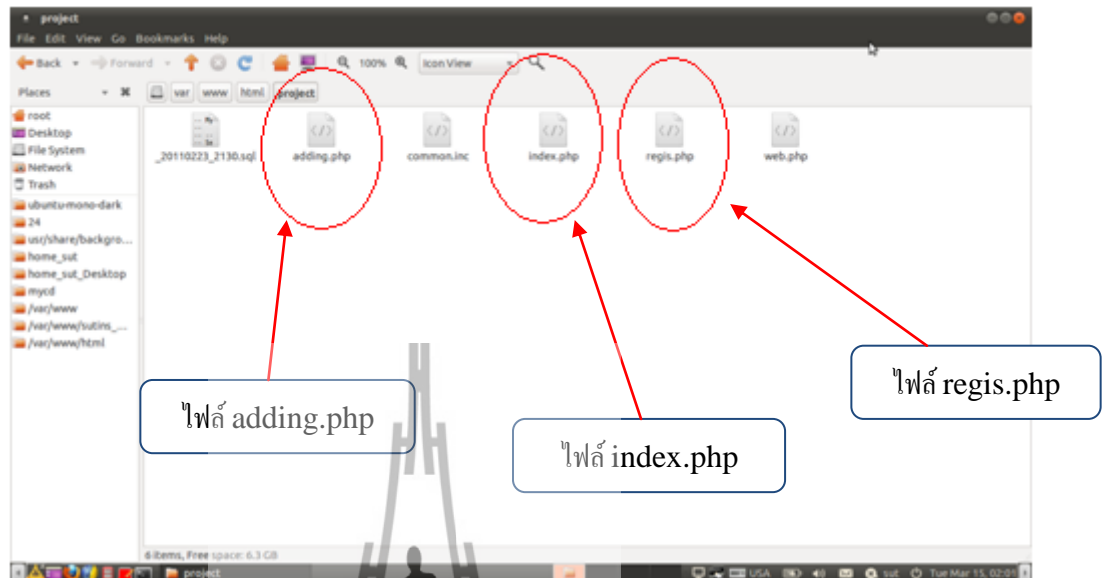
1st6 ตำแหน่งที่โปรแกรมคำนวณ 40 เพราะฉะนั้นตำแหน่งที่แสดงบนเว็บไซต์ก็คือตำแหน่ง 40

1suL ตำแหน่งที่โปรแกรมคำนวณ 30 เพราะฉะนั้นตำแหน่งที่แสดงบนเว็บไซต์ก็คือตำแหน่ง 30

1szI ตำแหน่งที่โปรแกรมคำนวณ 75 เพราะฉะนั้นตำแหน่งที่แสดงบนเว็บไซต์ก็คือตำแหน่ง 75

3.5.4 การเขียนหน้าเว็บเพจ และไฟล์ติดต่อฐานข้อมูล

การเขียน โปรแกรม จะเก็บอยู่ใน var/www/html/project



รูปที่ 3.7 ไฟล์ที่มีการเขียนโปรแกรมไว้

หน้าที่ของแต่ละไฟล์มีดังนี้

1) ไฟล์ regis.php

เป็นไฟล์ที่สร้างไว้ให้การลงทะเบียนผ่านทางเว็บไซต์ โดยข้อมูลที่ต้องการก็คือ ชื่อ tag และ รหัสนักศึกษา ซึ่งข้อมูลทั้งสองจะถูกเก็บไว้ในฐานข้อมูล table_name และ ถ้ามีการบันทึกข้อมูลที่มี ชื่อ tag ซ้ำกัน จะมีการบันทึกข้อมูลลงไปแทนข้อมูลเดิม (รหัส นักศึกษาเดิมจะหายไป)

2) ไฟล์ adding.php

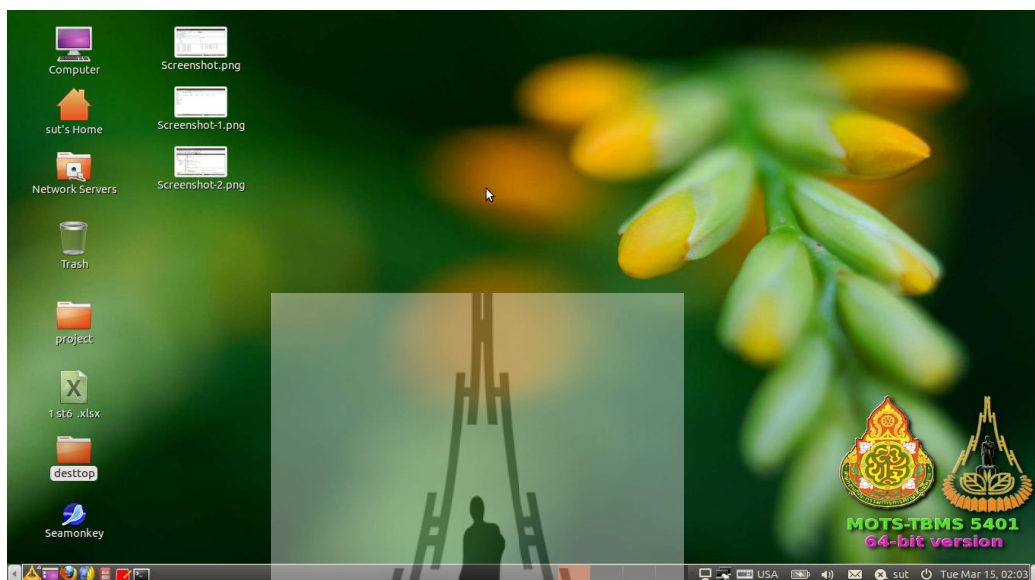
เป็นไฟล์ที่สร้างไว้เพื่อรอรับค่าที่ส่งมาจากโปรแกรมคำนวณ และเก็บไว้ใน ฐานข้อมูล และมีการอัปเดตข้อมูลทุกครั้งที่ได้รับค่ามา

3) ไฟล์ index.php

เป็นไฟล์ที่ใช้เป็นหน้าเว็บเพจของโครงการ มีการสร้างตารางในลักษณะเป็น แผนผังในห้องเรียน และดึงข้อมูลจากฐานข้อมูลมาแสดงบนเว็บ

3.5.5 การ up load ไฟล์ขึ้น server และการ update ข้อมูล

ใช้ระบบปฏิบัติการ Linux ของ มหาวิทยาลัยเทคโนโลยีสุรนารี

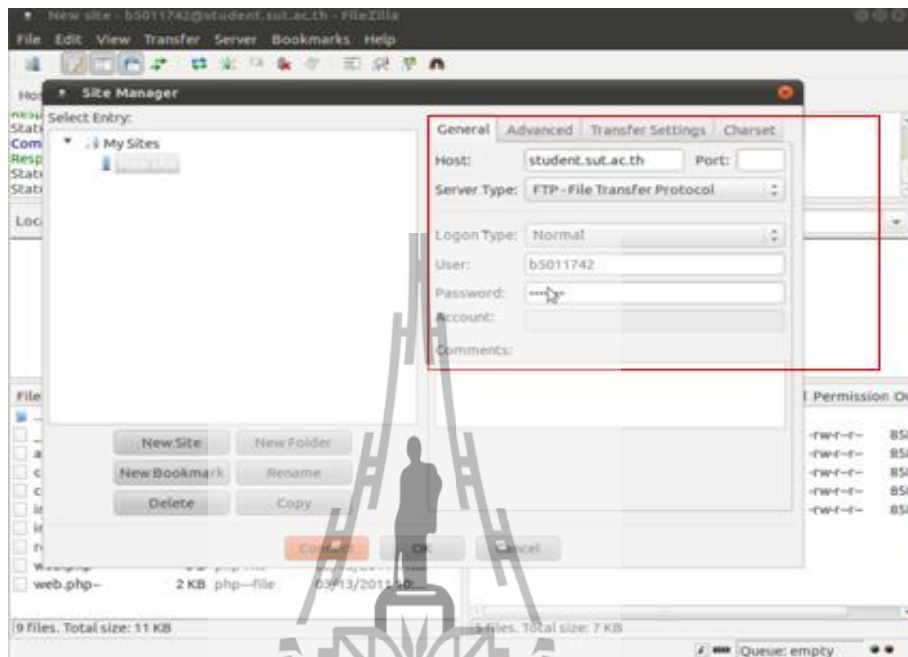


รูปที่ 3.8 ระบบปฏิบัติการ Linux

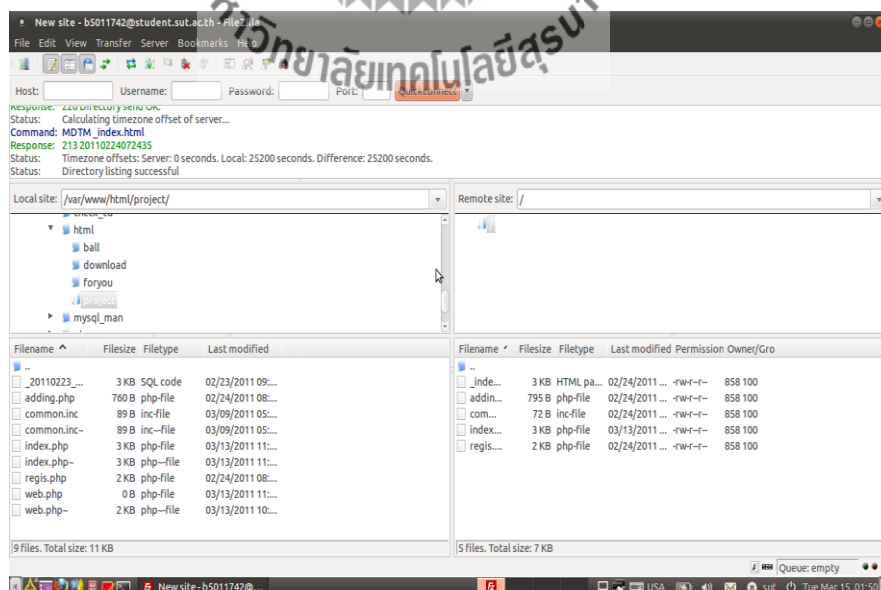


โปรแกรม File Zilla

ใช้งานโดยการกำหนด My Sites ไว้เพื่อสะดวกในการเชื่อมต่อครั้งต่อไป จะต้องมีกรใส่ชื่อ Host ซึ่ง Host ที่เราใช้คือ student.sut.ac.th. ใส่ User และ Password ตามกรอบสีแดง

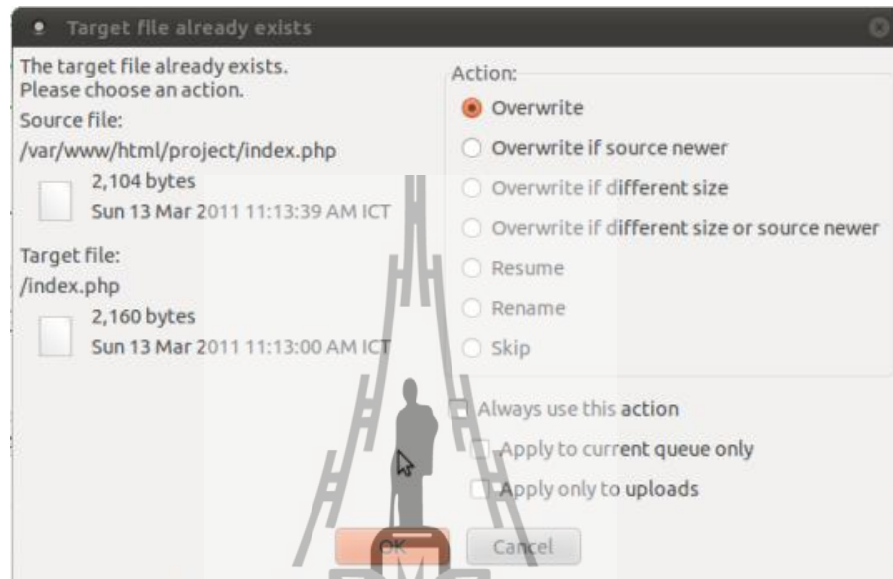


รูปที่ 3.9 การตั้งค่าก่อนการเชื่อมต่อ



รูปที่ 3.10 การเชื่อมต่อ server ได้สำเร็จ

จากรูปจะเห็นว่า ไฟล์ที่มีการ upload ขึ้นไปยัง server แล้ว จะอยู่ในด้านขวามือ ส่วนไฟล์ที่เราใช้ในการแก้ไข หรือสร้างใหม่จะอยู่ด้านซ้ายมือ ซึ่งเราสามารถ upload ไฟล์ขึ้นไปยัง server โดยการคลิกขวาที่ไฟล์ที่เราต้องการและเลือก upload จากนั้นจะขึ้นหน้าจอตามรูปที่ 3.11 และคลิก OK



รูปที่ 3.11 การ upload

3.5.6 การลงทะเบียนหรือการเก็บข้อมูลของผู้ใช้งานใหม่

เป็นการเก็บข้อมูลรหัสนักศึกษาและ tag เพื่อกำหนดค่าใช้จ่ายแต่ละคน โดยมี 2 วิธี คือ

1) การลงทะเบียนผ่านผู้ดูแลระบบลงในตาราง table_name

เข้าไปดูค่าในตารางที่เราเก็บค่าไว้ โดยการ คลิกเข้าไปที่ Browse ซึ่งสามารถแก้ไขฐานข้อมูลได้หรือใช้ลงทะเบียนผู้ใช้ใหม่ ในตารางนี้

phpMyAdmin

เซิร์ฟเวอร์: localhost ▶ ฐานข้อมูล: b5011742db ▶ ตาราง: table_name

แสดงระเบียบที่ 0 - 2 (3 ทั้งหมด, ค่าคั่นใช้เวลา 0.0001 วินาที)

```
SELECT *
FROM `table_name`
LIMIT 0, 30
```

แสดง: 30 แถว เริ่มจากแถวที่ 0

อยู่ใน แถวแรก และซ้ำหัวแถวทุก 100 เซลล์

เรียงโดยคีย์: ไม่มี

+ Options

	name	studentID
<input type="checkbox"/>	1suL	B5046010
<input type="checkbox"/>	1szI	B5015894
<input type="checkbox"/>	1st5	B5011742

เลือกทั้งหมด / *ไม่เลือกเลย หากทับที่เลือก:

แสดง: 30 แถว เริ่มจากแถวที่ 0

อยู่ใน แถวแรก และซ้ำหัวแถวทุก 100 เซลล์

Query results operations

แสดง Print view (with full texts) ส่งออก CREATE VIEW

รูปที่ 3.12 ตัวอย่างข้อมูลในตาราง table_name

- 2) การลงทะเบียนผ่านทาง <http://student.sut.ac.th/rfidproject/regs.php>

หมายเลขการ์ด

รหัสนักศึกษา

Register

รูปที่ 3.13 การลงทะเบียนผ่านอินเทอร์เน็ต

3.5.7 ส่งชื่อ tag และตำแหน่ง (location) ไปยัง Server

ในรูปแบบ `http://student.sut.ac.th/rfidproject/adding.php?location=70&name=1st6` ซึ่งค่าทั้งสองจะถูกส่งไปที่ตาราง `table_location` โดย code ที่ใช้เขียนโปรแกรม มีดังนี้

```

for (int y = 0; y < 3; y++)
{
    int index_ans = index_result[y] + 1;

    string result = null;
    string url = http://student.sut.ac.th/rfidproject/adding.php?location= + Convert.ToString(index_ans) + "&name=" +
name[y];
    WebResponse response = null;
    StreamReader reader = null;

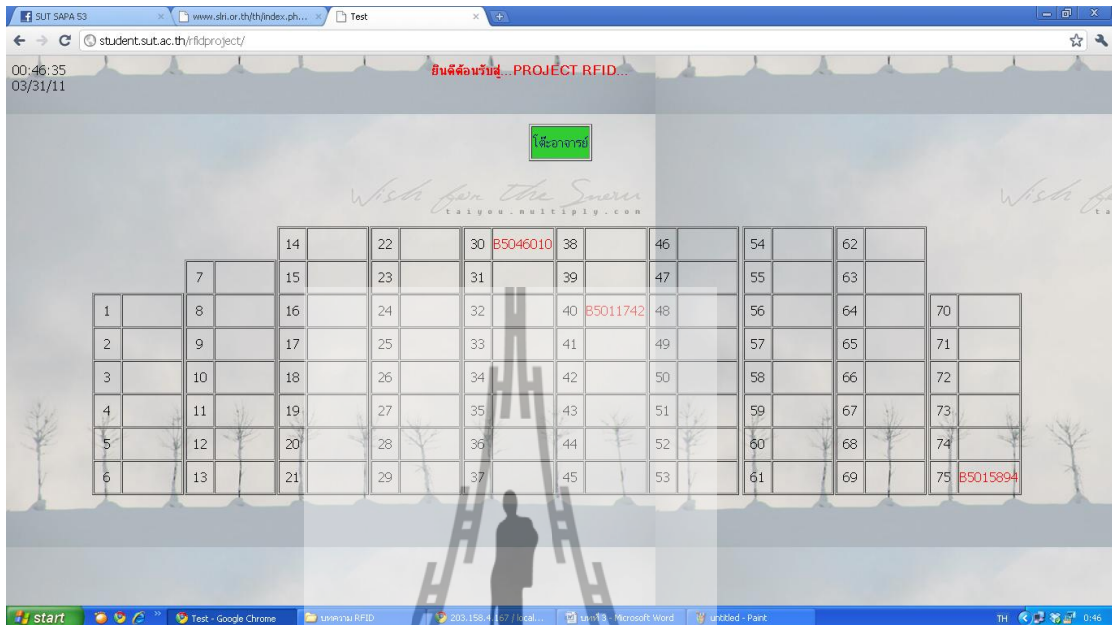
    HttpWebRequest request = (HttpWebRequest)WebRequest.Create(url);
    request.Method = "GET";
    response = request.GetResponse();
    reader = new StreamReader(response.GetResponseStream(), Encoding.UTF8);
    result = reader.ReadToEnd();

    Console.WriteLine("http://student.sut.ac.th/rfidproject/");
}

```

3.5.8 แสดงตำแหน่งบนเว็บไซต์

โดยเข้าไปที่ <http://student.sut.ac.th/rfidproject>



รูปที่ 3.14 ตัวอย่างหน้าเว็บไซต์

จากรูป เป็นแผนผังห้องที่ใช้ในการทดลองโครงการนี้ คือห้อง 150 ที่นั่ง (ห้อง B1208 – B1215)

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 4

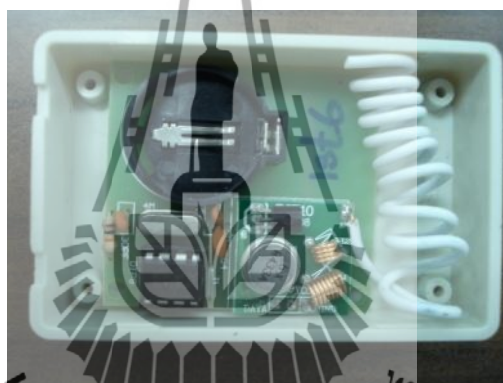
การใช้งานโครงงาน

4.1 การใช้งานเครื่อง

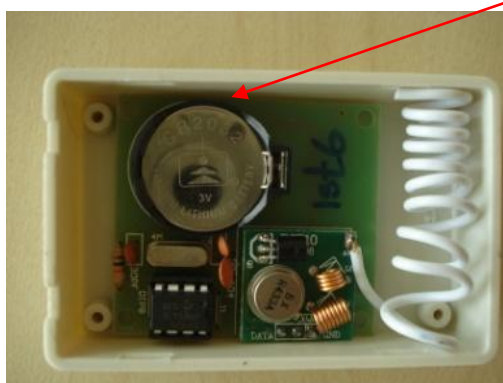
4.1.1 การติดตั้งและใช้งานอุปกรณ์

4.1.1.1) RF8315T Active RFID 8 Meters Transmitting Module (tag)

ในการทดสอบโครงงานนี้ จะใช้ tag 3 ตัว โดยในแต่ละตัวประกอบด้วย แผงวงจรและ ก่อ้งไว้สำหรับใส่แผงวงจร ทำให้สะดวกในการนำไปใช้งาน การใช้งานจะต้องมีการใส่ แบตเตอรี่ CR2025 หรือ CR2032



รูปที่ 4.1 tag ก่อนการใช้งาน



ใส่ แบตเตอรี่

รูปที่ 4.2 tag เมื่อมีการใส่ แบตเตอรี่



รูปที่ 4.3 tag พร้อมนำไปใช้งาน

การวาง tag ไว้บนโต๊ะ

ในการทดสอบโครงการนี้ผู้จัดทำได้เลือกตำแหน่งในการวาง tag ไว้บนโต๊ะต่างๆ โดยวางไว้ที่มุมซ้ายล่างของโต๊ะที่มีหมายเลขโต๊ะ และ tag จะต้องวางในลักษณะเดิม เพราะจะได้กำหนดให้เป็นตำแหน่งเดียวกันในการเก็บข้อมูลและการทดสอบ





รูปที่ 4.4 ตัวอย่างการวาง tag บนโต๊ะเรียน

4.1.1.2) RF2315R Active RFID Dual Receiver Module with RSSI (RFID Dual Receiver)

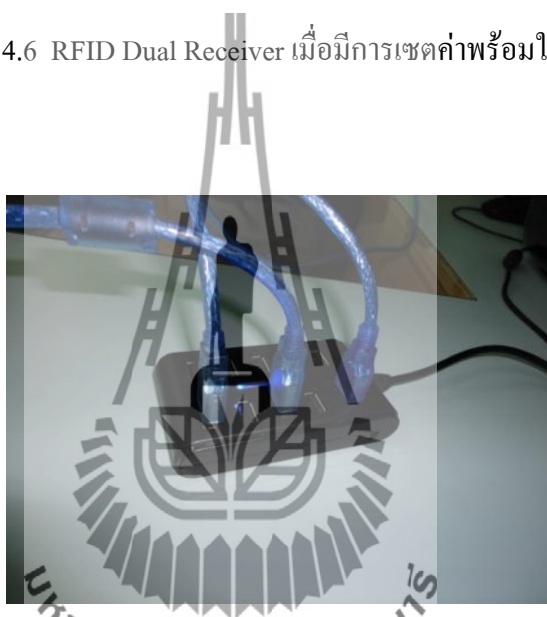
เป็นเครื่องรับสัญญาณที่ติดตั้งไว้ในห้องเรียนจำนวน 3 เครื่อง แต่ละเครื่องมีการต่อสายอากาศเพิ่มด้านขวา (รูปที่ 4.5) การเปิดใช้งานโดยการเชื่อมต่อไปยัง USB HUB (รูปที่ 4.7) ซึ่งทำหน้าที่เชื่อมต่อ Port USB จาก RFID Dual Receiver ทั้ง 3 เครื่อง แล้วต่อ USB HUB เข้ากับคอมพิวเตอร์ เมื่อมีการเชื่อมต่อเรียบร้อย หลอด LED ที่ RFID Dual Receiver จะติด (รูปที่ 4.6)



รูปที่ 4.5 RFID Dual Receiver ก่อนการใช้งาน

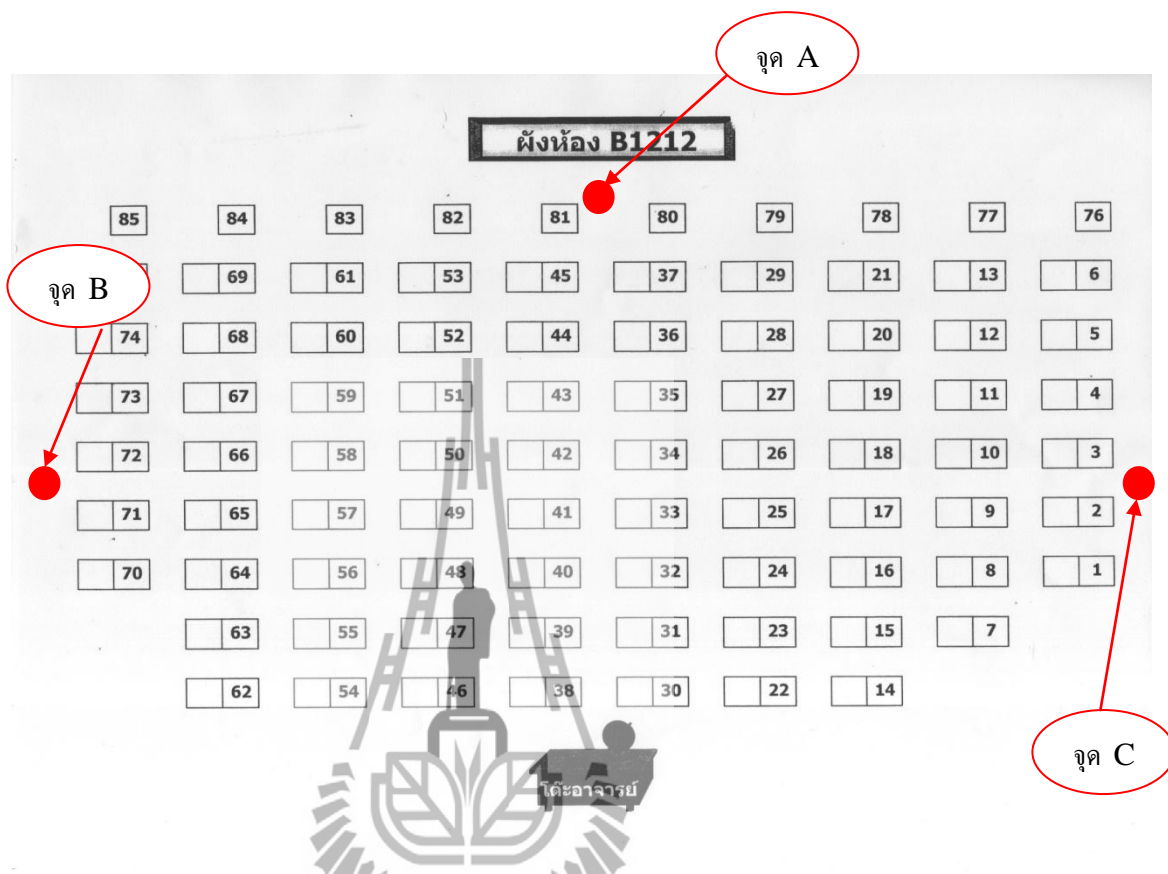


รูปที่ 4.6 RFID Dual Receiver เมื่อมีการเซตค่าพร้อมใช้งาน



รูปที่ 4.7 USB HUB

4.1.2 การติดตั้ง RF2315R Active RFID Dual Receiver Module with RSSI ไว้ในห้องเรียน



● คือ ตำแหน่งที่มีการติดตั้ง RF2315R Active RFID Dual Receiver Module with RSSI

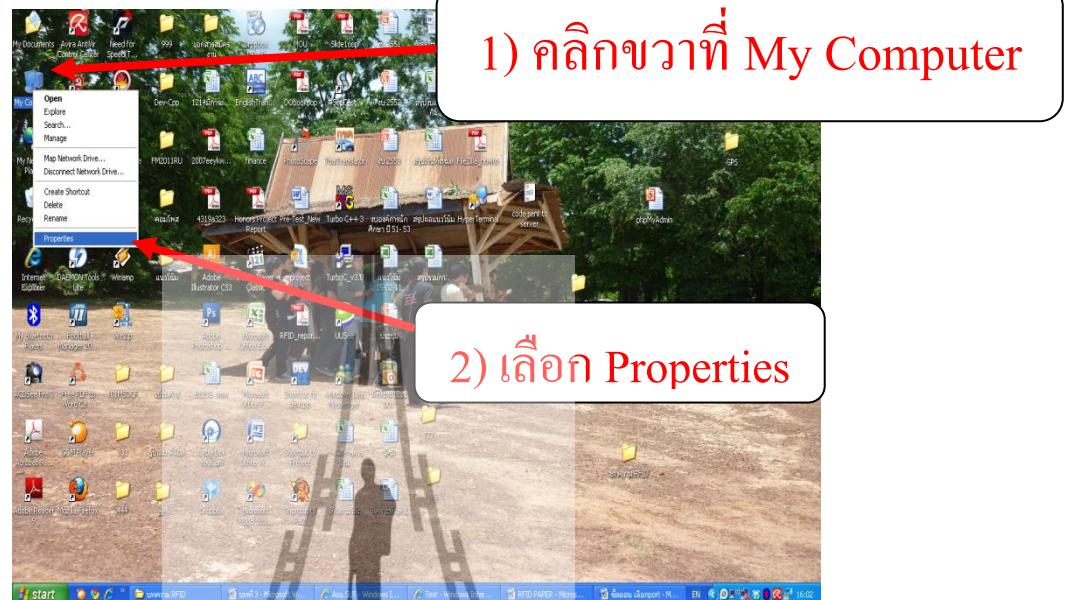
รูปที่ 4.8 ตำแหน่งที่มีการติดตั้ง RFID Dual Receiver



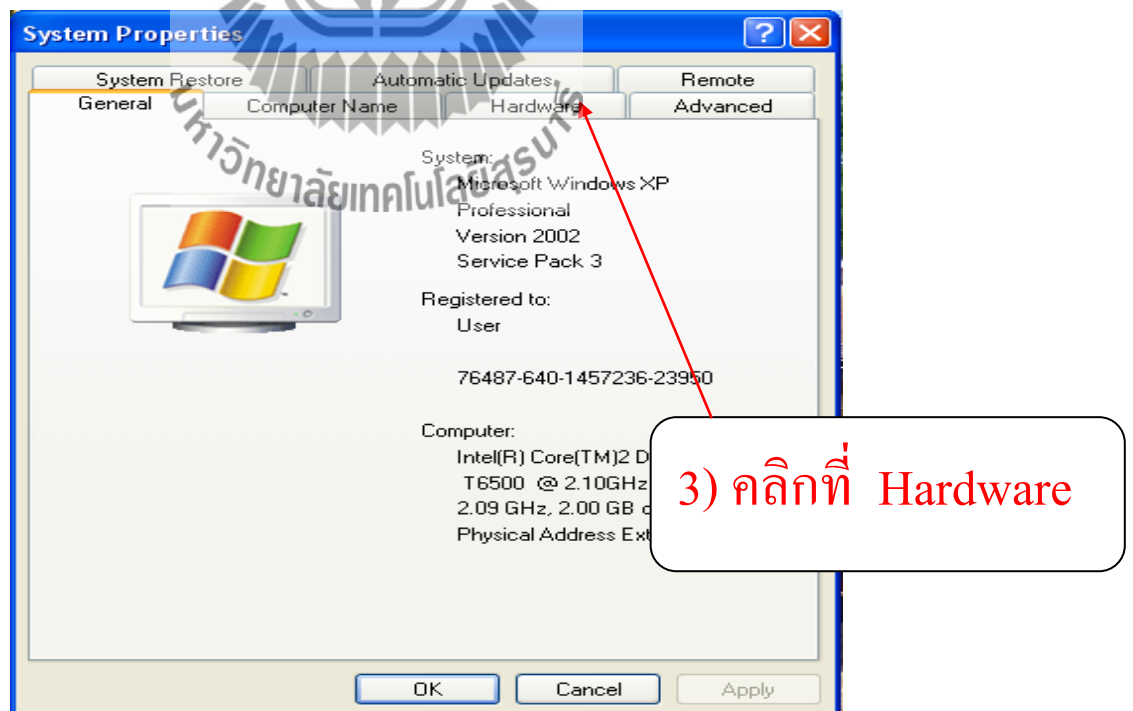
รูปที่ 4.9 ตัวอย่างการติดตั้ง RFID Dual Receiver ในห้องเรียน

4.2 การเลือก port

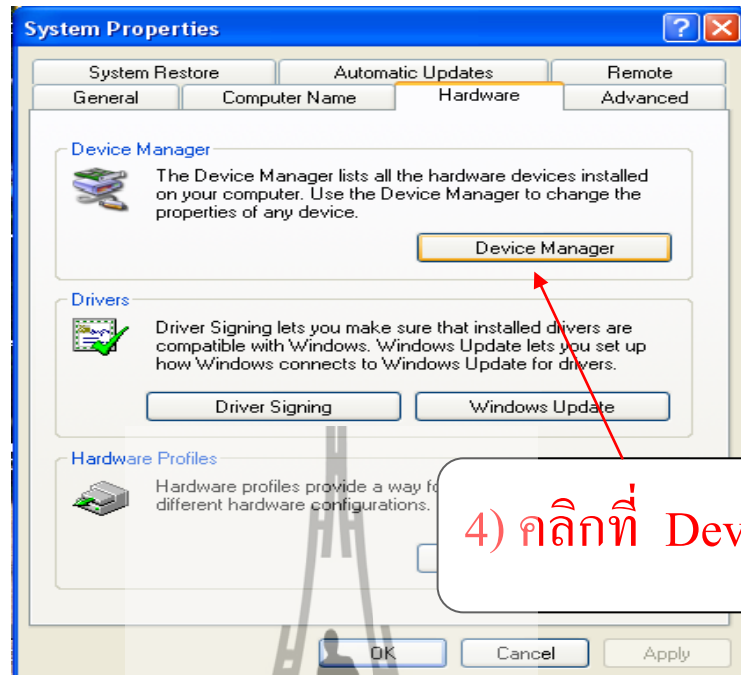
เพื่อตรวจสอบว่ามีการเชื่อมต่อที่ port ไດ จะได้นำไปใช้งานในการกำหนดค่าได้ถูกต้อง โดยมีขั้นตอนการเลือก port ดังนี้



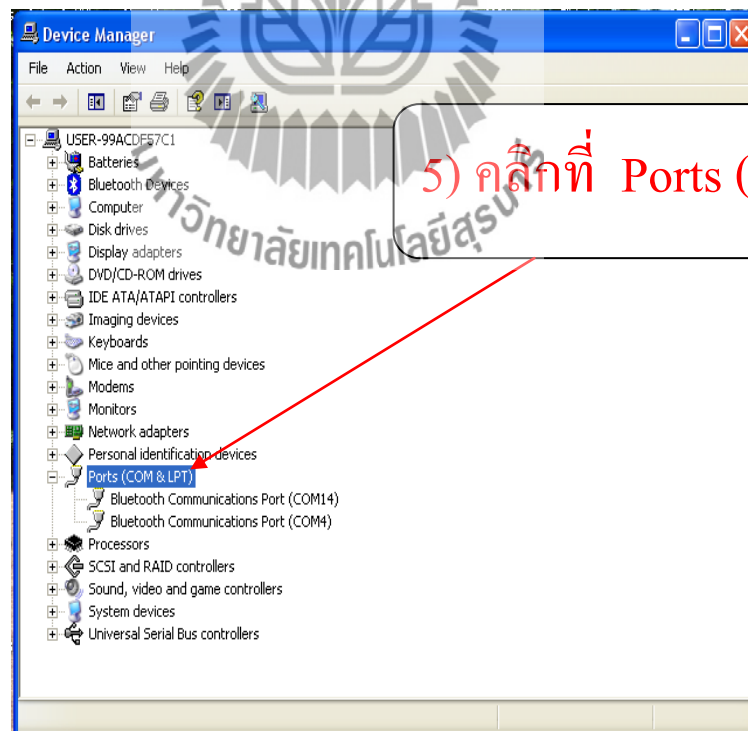
รูปที่ 4.10 ขั้นตอนการเลือก port (1)



รูปที่ 4.11 ขั้นตอนการเลือก port (2)



รูปที่ 4.12 ขั้นตอนการเลือก port (3)



รูปที่ 4.13 ขั้นตอนการเลือก port (4)

4.3 การตั้งค่า port ในโปรแกรม

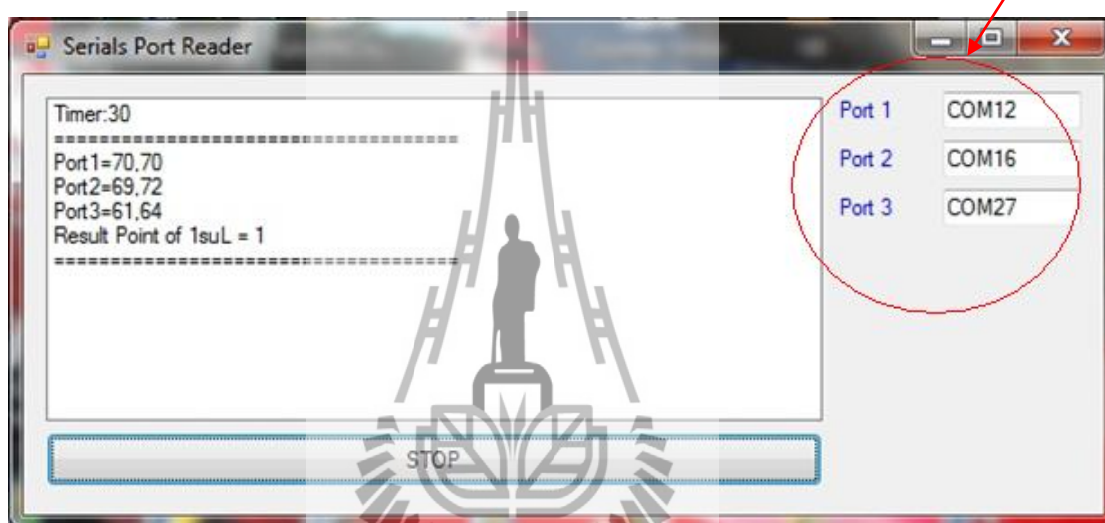
จะต้องตั้งค่า port ให้ตรงกับการเข้าไปดู port ในขั้นตอนก่อนหน้านี้ โดยจะต้องกำหนดดังนี้

Port 1 คือ เครื่อง RFID Dual Receiver ที่ติดตั้งอยู่ที่จุด A

Port 2 คือ เครื่อง RFID Dual Receiver ที่ติดตั้งอยู่ที่จุด B

Port 2 คือ เครื่อง RFID Dual Receiver ที่ติดตั้งอยู่ที่จุด C

ตั้งค่า port



รูปที่ 4.14 ตำแหน่งที่มีการตั้งค่า port

ตัวอย่างในรูปที่ 4.14 มีการกำหนด

- Port 1 มาจาก Port COM ที่ 12
- Port 2 มาจาก Port COM ที่ 16
- Port 3 มาจาก Port COM ที่ 27

ซึ่งในการตั้งค่า Port ในการทดสอบต้องมีการตั้งค่าให้ตรงกับตำแหน่งที่ทำการเก็บข้อมูล เช่น Port 1 ให้เก็บค่าจากจุด A เพราะฉะนั้น ในการทดสอบก็ต้องกำหนดให้ Port 1 ให้เก็บค่าจากจุด A เช่นกัน หากในการเก็บข้อมูลและการทดสอบตั้งค่าไม่ตรงกัน โปรแกรมก็สามารถคำนวณได้แต่ผลการคำนวณจะไม่ถูกต้อง (ระบุตำแหน่งผิดพลาด)

4.4 การทดสอบใช้โครงงาน

การทดลองใช้โครงงาน การแสดงผลตำแหน่งของนักศึกษาภายในห้องเรียนโดยใช้ RFID ผ่านทาง Website ผู้จัดทำได้นำโครงงานไปทดสอบโดยการบันทึกความแรงของ tag ซึ่งวางอยู่บนโต๊ะ โดยค่าความแรงมาจากการที่เครื่องอ่านทั้ง 3 เครื่องที่ติดตั้งอยู่ในห้องเรียนอ่านได้ เก็บไว้เป็นข้อมูลในการคำนวณ และทำการทดสอบโดยการนำ tag ไปวางไว้บนโต๊ะต่างๆ และให้โปรแกรมคำนวณค่าออกมาแสดงบนเว็บไซต์ทุกๆ 10 วินาที และตรวจสอบว่าตำแหน่งที่มี tag วางอยู่กับตำแหน่งที่แสดงบนเว็บไซต์ เป็นตำแหน่งเดียวกันหรือไม่ โดยมีการทดสอบทั้งสิ้น 5 ครั้ง ดังนี้

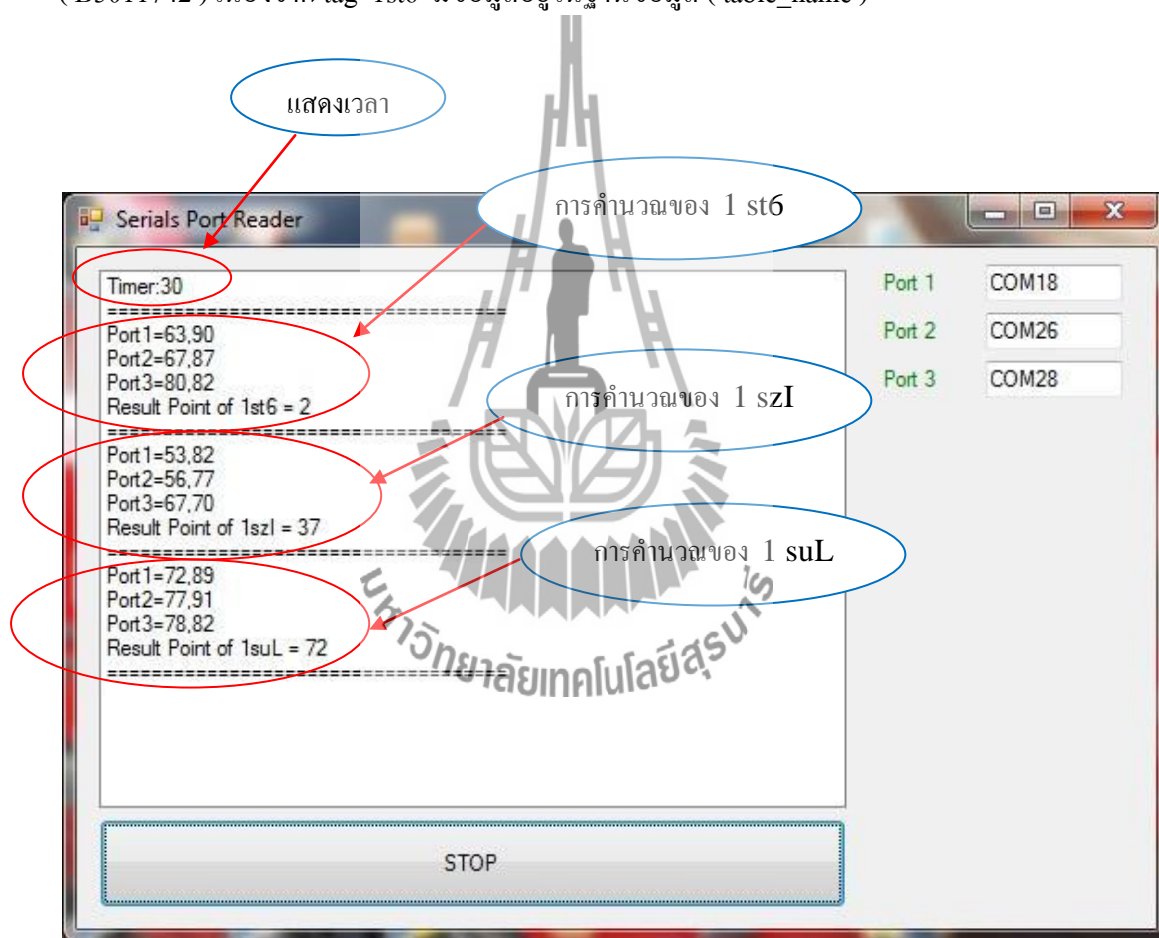
- 1) การแสดงค่าตำแหน่งบนเว็บไซต์ทุกๆ 10 วินาที
- 2) เก็บข้อมูลของ tag ชื่อ 1 st6 จำนวน 75 โต๊ะ (โต๊ะ 1-75)ทดสอบการใช้งานโดยการวางบนโต๊ะโดยใช้ tag ชื่อ 1 st6 ตัวเดียว
- 3) เก็บข้อมูลของ tag ชื่อ 1 st6 จำนวน 75 โต๊ะ (โต๊ะ 1-75)ทดสอบการใช้งานโดยการวางบนโต๊ะโดยใช้ tag ชื่อ 1 suL ตัวเดียว
- 4) เก็บข้อมูลของ tag ชื่อ 1 st6 จำนวน 75 โต๊ะ (โต๊ะ 1-75)ทดสอบการใช้งานโดยการวางบนโต๊ะโดยใช้ tag ชื่อ 1 szI ตัวเดียว
- 5) เก็บข้อมูลของ tag ชื่อ 1 st6 จำนวน 75 โต๊ะ (โต๊ะ 1-75)ทดสอบการใช้งานเมื่อ มี tag 3 ตัว (1 st6, 1 suL, 1 szI) อยู่ในระบบพร้อมๆ กัน

4.5 ผลการทดสอบโครงการ

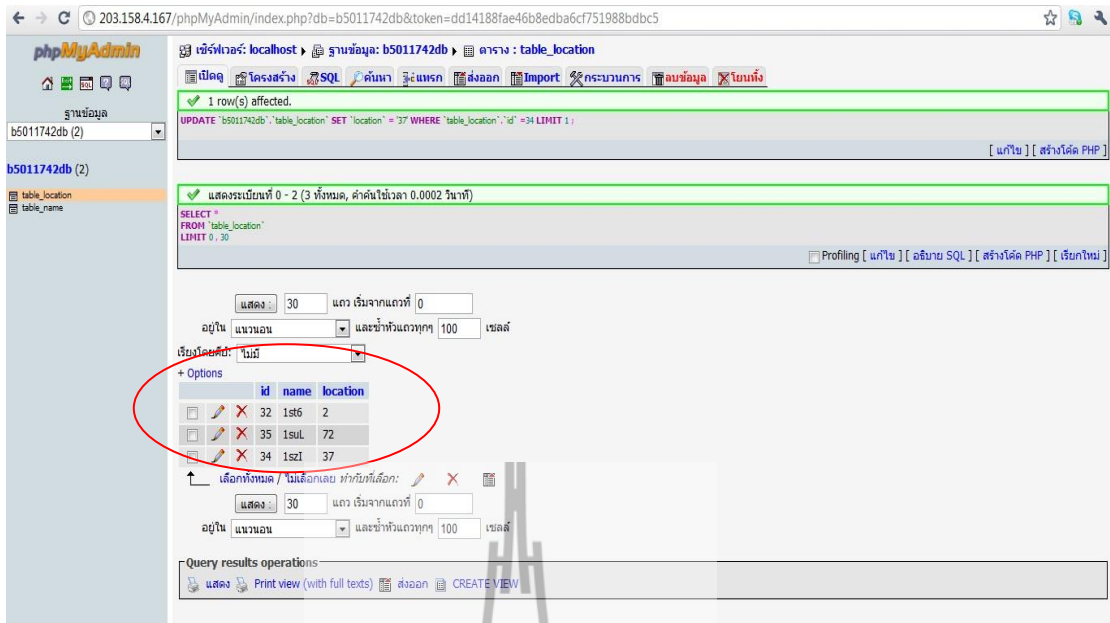
เมื่อนำโครงการมาทดสอบใช้งาน โดยการเข้าไปดูที่ <http://student.sut.ac.th/rfidproject/> และมีการวาง tag ไว้บนโต๊ะ โดยมีผลการทดสอบดังนี้

1. สามารถแสดงตำแหน่ง tag ที่วางไว้บนโต๊ะ บนเว็บไซต์ได้ และสามารถระบุได้ว่ามี tag อยู่ที่ตำแหน่งใดในห้องเรียน เนื่องจากมีการเก็บข้อมูลไว้แล้วว่า tag แต่ละตัว มีชื่อผู้ใช้งาน 1 คน
2. ในฐานะข้อมูลมีการบันทึกตำแหน่งของ tag ไว้ทุกครั้งที่มีการส่งค่ามาจากโปรแกรม ก่อนที่จะนำไปแสดงบนเว็บไซต์
3. การกำหนดให้ มีการ refresh หน้าเว็บ ทุก 10 วินาที พบว่า สามารถแสดงตำแหน่งได้ เป็นปัจจุบัน เนื่องจากมีการกำหนดโปรแกรมให้มีการคำนวณทุก ๆ 10 วินาที โดยการเก็บค่าทุก ๆ 2.3 วินาที รอให้ได้ครบทั้ง 3 port แต่ในเวลา 10 วินาทีแรกอาจมีค่าที่อ่านได้ผิดพลาดหรือค่ายังไม่นิ่ง ค่าที่นิ่งและถูกต้องทุกครั้งทดสอบ คือ 30 วินาที
4. ตำแหน่งที่วาง tag ของแต่ละโต๊ะ ต้องเป็นตำแหน่งเดิมที่มีการเก็บข้อมูลไว้
5. การแสดงตำแหน่งบนเว็บไซต์จะแสดงเฉพาะ tag ที่มีการลงทะเบียนไว้
6. ในระบบที่มี tag 3 ตัวการแสดงผลตำแหน่งบนเว็บไซต์ได้ผลเช่นเดียวกันกับการที่มี tag ในระบบเพียงตัวเดียว

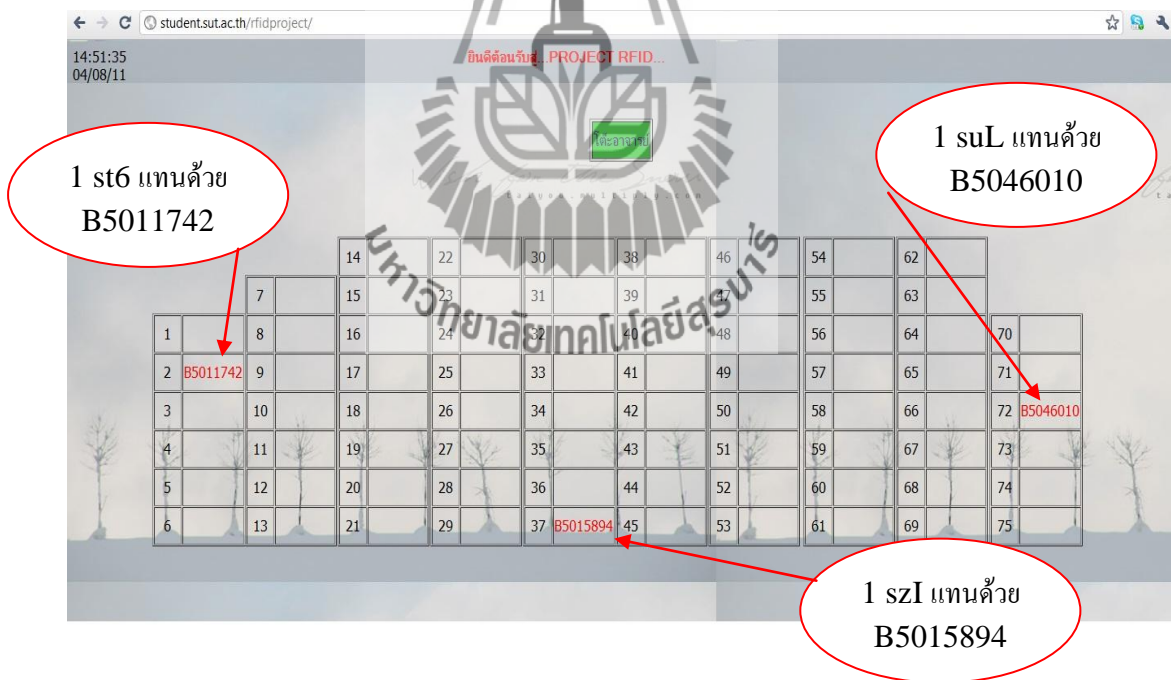
เมื่อมีการเปิดโปรแกรมจะทำการคำนวณหาตำแหน่งของ tag แต่ละตัว และมีการแสดงผลการคำนวณ ไว้ที่หน้าโปรแกรม ตามรูปที่ 4.15 ก่อนที่จะส่งผลการคำนวณไปยัง server จากนั้นข้อมูลจะไปเก็บอยู่ในฐานข้อมูล (table_location) ตามรูปที่ 4.16 และแสดงผลบนเว็บไซต์ ตามรูปที่ 4.17 ซึ่งในการทดลองตามรูปมีการเปิดใช้งาน tag 2 ตัว คือ 1szI ที่แทนด้วยรหัสนักศึกษา B5015894 ซึ่งวางอยู่ที่โต๊ะ 52 และ 1suL ที่แทนด้วยรหัสนักศึกษา B5046010 ซึ่งวางอยู่ที่โต๊ะ 37 ส่วน tag 1st6 ไม่ได้มีการเปิดใช้งานหรือไม่อยู่ในบริเวณที่เครื่อง RFID Dual Receiver สามารถอ่านค่าได้ เนื่องจากทั้งสาม Port มีค่าเป็นศูนย์ แต่ที่สามารถระบุตำแหน่งบนเว็บไซต์ได้ (B5011742) เนื่องจาก tag 1st6 มีข้อมูลอยู่ในฐานข้อมูล (table_name)



รูปที่ 4.15 การอ่านค่าในโปรแกรม



รูปที่ 4.16 ข้อมูลที่เก็บอยู่ใน table_location



รูปที่ 4.17 ตำแหน่งที่แสดงบน Website

ผลการทดสอบโครงการในแต่ละการครั้ง

1) การแสดงค่าตำแหน่งบนเว็บไซต์ทุกๆ 10 วินาที

เวลา (วินาที)	โต๊ะที่วาง tag	ตำแหน่งที่แสดงบน Web
10	6	x
20		✓
30		✓
40		✓
50		✓
60		✓
70		✓
80		✓
90		✓

เวลา (วินาที)	โต๊ะที่วาง tag	ตำแหน่งที่แสดงบน Web
10	17	x
20		x
30		✓
40		✓
50		✓
60		✓
70		✓
80		✓
90		✓

เวลา (วินาที)	โต๊ะที่วาง tag	ตำแหน่งที่แสดงบน Web
10	44	✗
20		✗
30		✓
40		✓
50		✓
60		✓
70		✓
80		✓
90		✓

เวลา (วินาที)	โต๊ะที่วาง tag	ตำแหน่งที่แสดงบน Web
10	67	✗
20		✓
30		✓
40		✓
50		✓
60		✓
70		✓
80		✓
90		✓

- 2) เก็บข้อมูลของ tag ชื่อ 1 st6 จำนวน 75 โຕ้ะ (โຕ้ะ 1-75) และ ทดสอบใช้งานโดยการวางบนโຕ้ะโดยใช้ tag ชื่อ 1 st6 ตัวเดียว

ชื่อ tag	โຕ้ะที่วาง tag	ตำแหน่งที่แสดงบน Web
1 st6	3	✓
1 st6	6	✓
1 st6	12	✓
1 st6	17	✓
1 st6	23	✓
1 st6	44	✓
1 st6	51	✗
1 st6	61	✓
1 st6	67	✓
1 st6	75	✓

- 3) เก็บข้อมูลของ tag ชื่อ 1 suL จำนวน 75 โຕ้ะ (โຕ้ะ 1-75) และ ทดสอบใช้งานโดยการวางบนโຕ้ะโดยใช้ tag ชื่อ 1 suL ตัวเดียว

ชื่อ tag	โຕ้ะที่วาง tag	ตำแหน่งที่แสดงบน Web
1 suL	4	✗
1 suL	6	✓
1 suL	15	✓
1 suL	19	✓
1 suL	29	✓
1 suL	35	✓
1 suL	37	✓
1 suL	60	✓
1 suL	66	✗
1 suL	69	✓

- 4) เก็บข้อมูลของ tag ชื่อ 1 st6 จำนวน 75 โຕ้ะ (โຕ้ะ 1-75) และ ทดสอบใช้งานโดยการวางบนโຕ้ะโดยใช้ tag ชื่อ 1 szI ตัวเดียว

ชื่อ tag	โຕ้ะที่วาง tag	ตำแหน่งที่แสดงบน Web
1 szI	8	✓
1 szI	9	✓
1 szI	16	✓
1 szI	19	✓
1 szI	25	✓
1 szI	32	✗
1 szI	38	✓
1 szI	44	✓
1 szI	65	✗
1 szI	67	✓

- 5) เก็บข้อมูลของ tag ชื่อ 1 st6 จำนวน 75 โຕ้ะ (โຕ้ะ 1-75) ทดสอบการใช้งานเมื่อ มี tag 3 ตัว (1 st6, 1 suL, 1 szI) อยู่ในระบบพร้อมๆกัน

ครั้งที่	ชื่อ tag	โຕ้ะที่วาง tag	ตำแหน่งที่แสดงบน Web
1	1 st6	3	✓
	1 suL	6	✓
	1 szI	8	✓
2	1 st6	6	✓
	1 suL	4	✗
	1 szI	9	✓
3	1 st6	12	✓
	1 suL	15	✓
	1 szI	16	✓

4	1 st6	17	✓
	1 suL	19	✓
	1 szI	25	✓
5	1 st6	23	✓
	1 suL	29	✓
	1 szI	19	✓
6	1 st6	44	✓
	1 suL	35	✓
	1 szI	32	✗
7	1 st6	51	✗
	1 suL	37	✓
	1 szI	38	✓
8	1 st6	61	✓
	1 suL	60	✓
	1 szI	44	✓
9	1 st6	67	✓
	1 suL	66	✗
	1 szI	65	✗
10	1 st6	75	✓
	1 suL	67	✓
	1 szI	69	✓

4.6 สรุปผลการทดสอบ

จากการทดสอบจะเห็นว่าภาระบุตำแหน่งบนเว็บไซต์ จะมีการบันทึกข้อมูลลงในฐานข้อมูลและ สามารถแสดงตำแหน่งบนเว็บไซต์ได้ถูกต้อง 80 % และ เวลาที่เหมาะสมในการแสดงบนเว็บ คือ เมื่อมีการคำนวณครั้งที่ 3 หรือ 30 วินาที ขึ้นไป

บทที่ 5

สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผลการทดลอง

โครงการ การแสดงผลตำแหน่งของนักศึกษาภายในห้องเรียนโดยใช้ RFID ผ่านทาง Website เป็นโครงการที่ใช้งานสำหรับการแสดงตำแหน่งในห้องเรียนบนเว็บไซต์ โดยจะต้องมีการสร้างฐานข้อมูลไว้เพื่อเตรียมรับข้อมูลที่ส่งมาจากการคำนวณได้ตำแหน่งที่วาง tag ก่อนที่จะแสดงบนเว็บไซต์ ซึ่งสามารถสรุปได้ดังนี้

1. สามารถแสดงตำแหน่งของ tag ที่ได้บันทึกข้อมูลลงในตาราง table_name บนเว็บไซต์ได้ (tag ที่มีการลงทะเบียนไว้)
2. สามารถแสดงตำแหน่งของ tag บนเว็บไซต์ได้ถูกต้อง 90 % สำหรับ 1 st6 และ 80 % สำหรับ 1suL และ 1szI เนื่องจากการแสดงบนเว็บไซต์จะแสดงตำแหน่งเดียวกับที่คำนวณได้ในโปรแกรม
3. เวลาที่เหมาะสมในการใช้งานระบบคือ เมื่อมีการวาง tag 30 วินาที

5.2 สิ่งที่ได้รับจากการทำโครงการ

- 1) ได้รับความรู้เกี่ยวกับหลักการการทำงานและการใช้งาน RF8315T Active RFID 8 Meters Transmitting Module และ RF2315R Active RFID Dual Receiver Module with RSSI
- 2) ได้รับความรู้เรื่อง RFID
- 3) ได้รับความรู้เกี่ยวกับการใช้งานระบบปฏิบัติการ Linux
- 4) ได้เรียนรู้การใช้งานภาษาที่ใช้คือ PHP , HTML และโปรแกรม phpMyadmin ที่ใช้ควบคุมจัดการ MySQL Server
- 5) ได้เรียนรู้การเขียนโปรแกรม Visual C# ในการส่งค่าไปยัง server
- 6) ได้เรียนรู้การทำงานร่วมกับผู้อื่น
- 7) สามารถนำความรู้ที่ได้จากทฤษฎีมาปฏิบัติจริง
- 8) สามารถนำความรู้และประสบการณ์ที่ได้จากการทำโครงการไปประยุกต์ใช้ในชีวิตจริง

5.3 ปัญหาและอุปสรรค

- 1) ในการทดสอบในแต่ละวันต้องมีการเก็บข้อมูลใหม่ทั้ง 75 โตะ เนื่องจากมีการติดตั้ง RFID Dual Receiver ใหม่ ทำให้เสียเวลาในการทดสอบทุกครั้ง เพราะเราไม่ได้ติดตั้งอุปกรณ์ไว้ถาวร
- 2) ไม่มีความรู้เกี่ยวกับเรื่อง server , ภาษาที่ใช้ คือ PHP , HTML ต้องใช้เวลาในการศึกษาเป็นอย่างมาก
- 3) เมื่อมีการใช้งานเครื่อง RFID Dual Receiver ไปนานๆ เครื่องจะหยุดทำงาน สัมผัสจากค่าที่อ่านได้เป็นศูนย์หลายๆ ครั้งติดกันใน Port เดิม
- 4) tag ที่มีการลงทะเบียนไว้ แต่ไม่ได้นำมาใช้งาน แต่สามารถแสดงตำแหน่งบนเว็บไซต์
- 5) ห้องเรียนที่ว่างให้เข้าไปทดลองได้จะมีวันอังคารและวันพฤหัสบดี เวลา 13.00 น.-17.00 น. และ เสาร์ – อาทิตย์ ทำให้การทดลองบางครั้งไม่เสร็จ

5.4 ข้อเสนอแนะ

- 1) การใช้งานอุปกรณ์ RFID ควรที่จะระมัดระวังในการใช้เนื่องจากอุปกรณ์มีสภาพที่ไม่ทนทานต่อแรงกระแทกอาจจะทำให้อุปกรณ์แตกหักหรือชำรุดและอุปกรณ์มีราคาแพง ควรศึกษาข้อมูลวิธีการรักษาอุปกรณ์การใช้งานอย่างถูกวิธี
- 2) หากมีการนำไปพัฒนาต่อควรมีการติดตั้ง RFID Dual Receiver ไว้ในห้องตลอดการทำโครงการ ซึ่งจะทำให้ไม่ต้องสิ้นเปลืองเวลาในการเก็บข้อมูล
- 3) ควรพัฒนาโปรแกรมให้มีการเก็บข้อมูลย้อนหลัง เพื่อบันทึกไว้ในฐานข้อมูล

ภาคผนวก

Code PHP ที่ใช้ในโปรแกรม

ไฟล์ regis.php

```

<html>
<head>
<title>Register</title>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html;charset=UTF-8">
</head>
<body>
<form method="get" action="<?=basename($_SERVER['PHP_SELF'])?>">
หมายเลขการ์ด
<input type="text" name="cId">
<br />
รหัสนักศึกษา
<input type="text" name="stuId">
<input type="hidden" name="flag" value="1">
<br />
<input type="submit" value="Register">
</form>

<?php
include("common.inc");
$card=$_GET['cId'];
$student=$_GET['stuId'];
$con=mysql_connect($host,$user,$pass);
if (!$con)
{

```

```
die('ติดต่อบริการข้อมูลไม่ได้ เนื่องจาก: ' . mysql_error());
}
mysql_select_db("b5011742db");

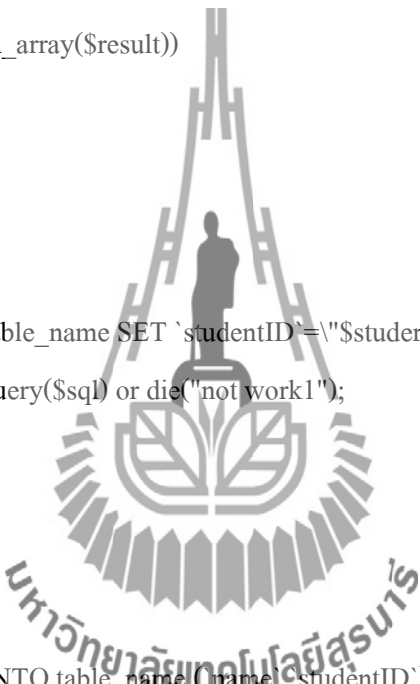
$sql0="select * from table_name";

$result=mysql_query($sql0);

while($row = mysql_fetch_array($result))
{
    if($row[0]==$card)
    {
        $sql="UPDATE table_name SET `studentID`='$student\' WHERE `name`='$card'";
        $status = mysql_query($sql) or die("not work1");
        exit(0);
    }
}

$sql2="INSERT INTO table_name (name, studentID) values ('$card','$student')";
$status=mysql_query($sql2)or die("not work2");

?>
</body>
</html>
```



ไฟล์ adding.php

```
<?php
$locate=$_GET['location'];
$name=$_GET['name'];
include("common.inc");

$con=mysql_connect($host,$user,$pass);
if (!$con)
{
    die('ติดต่อฐานข้อมูลไม่ได้ เนื่องจาก: ' . mysql_error());
}
mysql_select_db("b5011742db");

$sql0="select `name` from table_location";

$result=mysql_query($sql0);

while($row = mysql_fetch_array($result))
{
    if($row[0]==$name)
    {
        $sql="UPDATE table_location SET `location`='".$locate.'" WHERE `name`='".$name.'"";
        mysql_query($sql) or die("not work");
        exit(0);
    }
}

$sql2="INSERT INTO table_location (`name`,`location`) values ('".$name."','".$locate.'"");
mysql_query($sql2);

?>
```


ไฟล์ index.php

```

<html>
<head>
<title>Test</title>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html;charset=UTF-8">
</head>
<body>

<meta http-equiv="refresh" content="10"/>

<div style="position:absolute; left:480px; width:350px; height:10px; z-index:8"><font face="MS
Sans Serif" size="3" color="#FFFFFF"><b>
<marquee align="middle" scrolldelay="150"><font color="red">ยินดีต้อนรับผู้...PROJECT
RFID...</font></marquee>
</b></font> </div>

<style>body {background-image:url(http://www.2how.com/board/picture/0801/05npy957.jpg);
background-repeat: repeat;
background-position: center left;
ba
}</style>

<?php
include("common.inc");
$con=mysql_connect($host,$user,$pass);
if (!$con)
{
die('ติดต่อฐานข้อมูลไม่ได้ เนื่องจาก: ' . mysql_error());
}

```

```

}
mysql_select_db("b5011742db",$con);

$count=1;
print date('H:i:s');echo "<br>";
print date('m/d/y');

echo "<br>";
echo "<br>";
echo "<br>";

// โต้ะอาจารย์
echo "<center><table border=1 ></center>";
echo"<td bgcolor=\"#32CD32\" rowspan=1 align=\"center\" height=40><font color=\"#000080\"
face=\"MS Sans Serif\"> โต้ะอาจารย์ </font>";</td>";

// โต้ะ 1 - 6
echo "<table border=\"1\" style=\"position:absolute; left:106px; top:292px;\" cellspacing=\"1\"
cellpadding=\"1\">";
for($x=0;$x<6;$x++){
    echo "<tr height=\"40px\">";
        echo "<td width=\"30px\" align=\"center\">";

        echo $count;
        echo "</td>";
        echo "<td width=\"70px\" style=\"color:red\">";
        echo "<div id=\"pos\".$count++.\">";
        echo "</div>";
        echo "</td>";

```

```

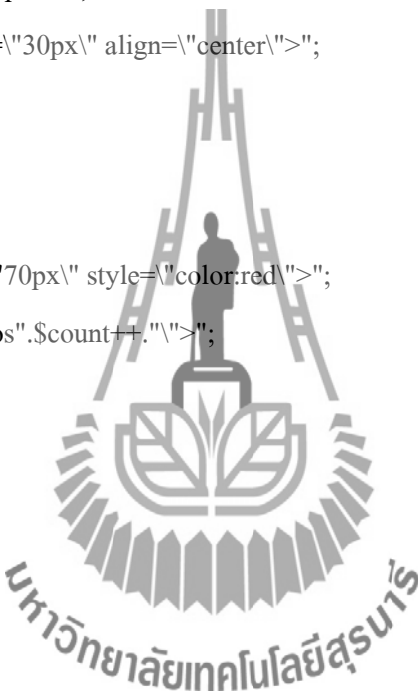
}
echo "</table>";

// ใ้ 7 - 13
echo "<table border=\"1\" style=\"position:absolute; left:220px; top:251px;\" cellspacing=\"1\"
cellpadding=\"1\">";
for($x=0;$x<7;$x++){
echo "<tr height=\"40px\">";
echo "<td width=\"30px\" align=\"center\">";

echo $count;
echo "</td>";
echo "<td width=\"70px\" style=\"color:red\">";
echo "<div id=\"pos\".$count++.\"\">";
echo "</div>";
echo "</td>";

}
echo "</table>";

```



```

// ใ้ 14 - 21
echo "<table border=\"1\" style=\"position:absolute; left:334px; top:210px;\" cellspacing=\"1\"
cellpadding=\"1\">";
for($x=0;$x<8;$x++){
echo "<tr height=\"40px\">";
echo "<td width=\"30px\" align=\"center\">";

echo $count;
echo "</td>";

```

```

echo "<td width=\"70px\" style=\"color:red\">";
echo "<div id=\"pos\". $count++.\">";
echo "</div>";
echo "</td>";
}
echo "</table>";

```

```
// โจทย์ 22 - 29
```

```

echo "<table border=\"1\" style=\"position:absolute; left:448px; top:210px;\" cellpadding=\"1\" cellspacing=\"1\">";
for($x=0;$x<8;$x++){
echo "<tr height=\"40px\">";

echo "<td width=\"30px\" align=\"center\">";

echo $count;
echo "</td>";
echo "<td width=\"70px\" style=\"color:red\">";
echo "<div id=\"pos\". $count++.\">";
echo "</div>";
echo "</td>";
}
echo "</table>";

```

```
// โจทย์ 30 - 37
```

```

echo "<table border=\"1\" style=\"position:absolute; left:562px; top:210px;\" cellpadding=\"1\" cellspacing=\"1\">";

```

```

for($x=0;$x<8;$x++){
    echo "<tr height=\"40px\">";

        echo "<td width=\"30px\" align=\"center\">";

            echo $count;
            echo "</td>";
            echo "<td width=\"70px\" style=\"color:red\">";
            echo "<div id=\"pos\".$count++.\">";
            echo "</div>";
            echo "</td>";
        }
    echo "</table>";

// โ้ต๊ะ 38 - 45

echo "<table border=\"1\" style=\"position:absolute; left:676px; top:210px;\" cellspacing=\"1\"
cellpadding=\"1\">";
for($x=0;$x<8;$x++){
    echo "<tr height=\"40px\">";

        echo "<td width=\"30px\" align=\"center\">";

            echo $count;
            echo "</td>";
            echo "<td width=\"70px\" style=\"color:red\">";
            echo "<div id=\"pos\".$count++.\">";
            echo "</div>";
            echo "</td>";

```

```

}
echo "</table>";

// โ้๓๓ 46 - 53
echo "<table border=\"1\" style=\"position:absolute; left:790px; top:210px;\" cellspacing=\"1\"
cellpadding=\"1\">";
for($x=0;$x<8;$x++){
    echo "<tr height=\"40px\">";

    echo "<td width=\"30px\" align=\"center\">";

    echo $count;
    echo "</td>";
    echo "<td width=\"70px\" style=\"color:red\">";
    echo "<div id=\"pos\".$count++.\"\">";
    echo "</div>";
    echo "</td>";
}
echo "</table>";

```



```

// โ้๓๓ 54 - 61
echo "<table border=\"1\" style=\"position:absolute; left:907px; top:210px;\" cellspacing=\"1\"
cellpadding=\"1\">";
for($x=0;$x<8;$x++){
    echo "<tr height=\"40px\">";

    echo "<td width=\"30px\" align=\"center\">";

    echo $count;
    echo "</td>";

```

```

echo "<td width=\"70px\" style=\"color:red\">";
echo "<div id=\"pos\". $count++.\">";
echo "</div>";
echo "</td>";
}
echo "</table>";

```

```
// ใ้ล้ล้ 62 - 69
```

```

echo "<table border=\"1\" style=\"position:absolute; left:1021px; top:210px;\" cellspacing=\"1\"
cellpadding=\"1\">";
for($x=0;$x<8;$x++){
echo "<tr height=\"40px\">";

echo "<td width=\"30px\" align=\"center\">";

echo $count;
echo "</td>";
echo "<td width=\"70px\" style=\"color:red\">";
echo "<div id=\"pos\". $count++.\">";
echo "</div>";
echo "</td>";
}
echo "</table>";

```

```
// ใ้ล้ล้ 70 - 75
```

```

echo "<table border=\"1\" style=\"position:absolute; left:1136px; top:292px;\" cellspacing=\"1\"
cellpadding=\"1\">";
for($x=0;$x<6;$x++){
echo "<tr height=\"40px\">";

```

```

echo "<td width=\"30px\" align=\"center\">";

echo $count;
echo "</td>";
echo "<td width=\"70px\" style=\"color:red\">";
echo "<div id=\"pos\".$count++.\">";
echo "</div>";
echo "</td>";
}
echo "</table>";

$sql="SELECT tl.name,tl.location,tn.studentID FROM table_location tl, table_name tn WHERE
tl.name=tn.name";
$result=mysql_query($sql);
while($row = mysql_fetch_array($result))
{
    $name=$row[2];
    $locate=$row[1];
    //$row['2'] //location
    echo "<script language=\"javascript\">";
    echo "document.getElementById(\"pos$locate\").innerHTML=\"$name\";";
    echo "</script>";
}

?>

```


ไฟล์ **common.inc**

```
<?php  
$host="203.158.4.167";  
$user="b5011742";  
$pass="ballann";  
$dbname="b5011742db";  
?>
```



เอกสารอ้างอิง

นิรุช อำนวยศิลป์. (2548). เทคนิค PHP เพื่อการต่อยอด . พิมพ์ครั้งที่ 1 . กรุงเทพฯ :
 ด้านสุทธาการพิมพ์

นิรุช อำนวยศิลป์. (2548). เทคนิค PHP เพื่อการต่อยอด 2 . พิมพ์ครั้งที่ 1 . กรุงเทพฯ :
 ด้านสุทธาการพิมพ์

อ.บัณฑิต จามรภูติ. (2552). คัมภีร์ Ubuntu Linux Server เล่ม 1 . พิมพ์ครั้งที่ 1 . กรุงเทพฯ :
 บริษัท เอช เอ็นกรุ๊ป จำกัด

PHP Tutorials: MySQL . สืบค้นจาก :

<http://www.youtube.com/watch?v=0jaJ1rTyTnY&feature=fvwrel>

(13 มกราคม 2554)

มหาวิทยาลัยศรีปทุม . ข้อมูลเกี่ยวกับ RFID. สืบค้นจาก :

http://csits.spu.ac.th/rfid/Sr_Project_Info/RFID_senior_project_info.html (15 มกราคม 2554)

ดร.ทวิศักดิ์ กอนันตกุล . 2548 . เทคโนโลยี RFID ที่ส่งผลกระทบต่อประเทศไทย. สืบค้นจาก :

http://www.ttc.most.go.th/stvolunteer/UploadClinic/RFID/A_ReflexRFID.pdf (21 กุมภาพันธ์ 2554)

How to: Send Data Using the WebRequest Class . สืบค้นจาก :

<http://msdn.microsoft.com/en-us/library/debx8sh9.aspx> (2 มีนาคม 2554)

ประวัติผู้เขียน



นายสันฐิติ พรหมากร

เกิดเมื่อวันที่ 5 สิงหาคม พ.ศ. 2531

ภูมิลำเนาอยู่ที่ ตำบลบางมะเดื่อ อำเภอพุนพิน

จังหวัดสุราษฎร์ธานี

สำเร็จการศึกษาระดับมัธยมปลายจากโรงเรียนสุราษฎร์ธานี

อำเภอเมือง จังหวัดนครราชสีมา เมื่อปี พ.ศ. 2550

ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี



นางสาวกมลทิพย์ ฝืนรงค์ฤทธิ์

เกิดเมื่อวันที่ 24 พฤศจิกายน พ.ศ. 2531

ภูมิลำเนาอยู่ที่ ตำบลโป่งแดง อำเภอขามทะเลสอ

จังหวัดนครราชสีมา

สำเร็จการศึกษาระดับมัธยมปลายจากโรงเรียนราชสีมาวิทยาลัย 2

อำเภอเมือง จังหวัดนครราชสีมา เมื่อปี พ.ศ. 2550

ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี