

**ระบบนำเข้าข้อมูลจากผู้ใช้เทียบกับข้อมูลที่ได้จากเซ็นเซอร์ผ่าน**

**WEB APPLICATION**

**WEB APPLICATION FOR USER INPUT DATA RELATE WITH  
ONLINE SENSOR DATA**

**อรรถกร เทพลีสิทธิ์**

**ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต**

**ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**

**ปีการศึกษา 2558**

ระบบนำเข้าข้อมูลจากผู้ใช้เทียบกับข้อมูลที่ได้จากเซ็นเซอร์ผ่าน

WEB APPLICATION

WEB APPLICATION FOR USER INPUT DATA RELATE WITH

ONLINE SENSOR DATA

อรรถกร เทพสิทธิ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2558

ปริญญานิพนธ์ปีการศึกษา 2558

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบนำเข้าข้อมูลจากผู้ใช้เทียบกับข้อมูลที่ได้จากเซ็นเซอร์ผ่าน WEB APPLICATION

WEB APPLICATION FOR USER INPUT DATA RELATE WITH ONLINE

SENSOR DATA

ผู้จัดทำ

1. นายอรรถกร เทพสิทธิ์

รหัสนักศึกษา 55011432



อาจารย์ที่ปรึกษา

(อาจารย์เกียรติณรงค์ ทองประเสริฐ)

# ระบบนำเข้าข้อมูลจากผู้ใช้เทียบกับข้อมูลที่ได้จากเซ็นเซอร์ผ่าน WEB APPLICATION

นายอรรถกร           เทพสิทธิ์           55011432  
อาจารย์เกียรติณรงค์ ทองประเสริฐ    อาจารย์ที่ปรึกษา  
ปีการศึกษา 2558

## บทคัดย่อ

โครงการนี้จัดทำขึ้นเพื่อให้เกษตรกรสามารถดูข้อมูลเซ็นเซอร์ในสวนเมล่อนและทำนายผลของการทำเกษตรในปีถัดๆไปได้ ในเรื่องของผลผลิต สภาพอากาศ โอกาสการเกิดโรคของพืชที่ปลูกและอื่นๆ แต่เนื่องจากข้อมูลที่มีนั้นเป็นข้อมูลของเซ็นเซอร์ที่ได้จากโครงการของปีการศึกษา 2557 โดยการนำอุปกรณ์เซ็นเซอร์ไปติดตั้งในสวนเมล่อนและเก็บข้อมูลของเซ็นเซอร์ทุกตัวทุกๆ 5 นาทีลงในฐานข้อมูลซึ่งข้อมูลที่ได้จากเซ็นเซอร์คือ อุณหภูมิในดิน ทิศทางของลม ความเร็วลม ความเร็วลมกันโชก ปริมาณน้ำในแปลงผัก อุณหภูมิ ค่าความเป็นกรดเบส และปริมาณแบตเตอรี่ ซึ่งข้อมูลนี้ไม่เพียงพอที่จะนำมาวิเคราะห์ ดังนั้นจึงจัดทำโครงการนี้ขึ้น เป็นเว็บแอปพลิเคชันที่แสดงผลข้อมูลจากเซ็นเซอร์ในรูปแบบของกราฟเส้นตรงและตารางเพื่อให้เกษตรกรสามารถเห็นการเปลี่ยนแปลงได้อย่างชัดเจนและมีระบบแจ้งเตือนผู้ใช้งานเมื่อมีความผิดปกติของข้อมูลเซ็นเซอร์อีกด้วย รวมทั้งเก็บข้อมูลการประเมินผลจากเซ็นเซอร์และแนวโน้มของผลผลิต โดยให้ผู้ใช้งานระบบเป็นผู้ประเมินลงในเว็บแอปพลิเคชันทุกวัน เว็บแอปพลิเคชันทำขึ้นโดยแบ่งออกเป็น 3 ส่วนสำคัญคือส่วนของการแสดงผลบนหน้าเว็บซึ่งในส่วนนี้จะใช้ภาษา HTML, CSS, JavaScript และ PHP เป็นส่วนสำคัญในการพัฒนาโปรแกรม ส่วนที่สองคือส่วนของ Web Service ที่เป็นตัวกลางสำหรับรับส่งข้อมูลระหว่างเว็บแอปพลิเคชันและฐานข้อมูล ในส่วนสุดท้ายเป็นส่วนของฐานข้อมูลที่ทำหน้าที่เก็บข้อมูลที่ได้รับจากเซ็นเซอร์และข้อมูลการประเมินที่ผู้ใช้งานระบบเป็นผู้ประเมิน

เว็บแอปพลิเคชันที่พัฒนาขึ้นสามารถตอบ โจทย์ที่ตั้งไว้ได้โดยแบ่งออกเป็นการทำงานใหญ่ๆ 2 ระบบคือส่วนของการแสดงข้อมูลจากเซ็นเซอร์บนหน้าเว็บในรูปแบบของตารางข้อมูลที่ สามารถเลือกวันที่ที่ต้องการให้แสดงข้อมูลได้และรูปแบบของกราฟเส้นตรงที่แสดงข้อมูลของเซ็นเซอร์ในหนึ่งวันทำให้สามารถเห็นการเปลี่ยนแปลงของข้อมูลได้ ระบบที่สองคือระบบประเมินผลจากผู้ใช้งานระบบในรูปแบบการเลือกระดับผลประเมิน 5 ระดับจาก มากที่สุด มาก ปานกลาง น้อย และน้อยที่สุด เพื่อเก็บข้อมูลทั้งหมดลงในฐานข้อมูล

# WEB APPLICATION FOR USER INPUT DATA RELATE WITH ONLINE SENSOR DATA

Mr. Attakorn      Thepsit                      55011432

Mr. Kiatnarong   Tongprasert              Advisor

Academic Year 2015

## ABSTRACT

This project is intended to allow farmers to see the sensor information in the melon farm, and predict the outcome of the agriculture in subsequent years about the matter of the yield, weather conditions, crop diseases, and other. But since the information available is only sensor data, it derived from the project of the academic year 2014, the sensor is installed in the melon farm and collect data of all sensors every 5 minutes into the database. The sensor data are as follows: soil temperature, wind direction, wind speed, gust speed, the volume of water in vegetable plot, temperature, soil acidification and quantity of the battery, it is not enough to be analyzed. Therefore, this project was created is a Web application that displays information from a sensors in the form of a linear graph and table so that the farmers can see clearly about the change. Moreover, there is also a system for alerting users of sensor data abnormalities. Including collect evaluation data from sensor and the trend of the results produced by allowing users to evaluate system into web application every day. Web applications developed by divided into three main sections: part of the display on a Web page, which in this section will use the language such as HTML, CSS, JavaScript and PHP, which was an important part in the development of the program. The second part is Web Service, which is an intermediary for data transmission between web applications and databases. The last part is the part of the database that stores the information received from the sensor and the evaluation data that the user evaluates.

Web applications can be developed that meet set by divided into 2 major system operation: part of the display of sensor data on a web page in the form of a table that can select the desired date, and the format of the linear graph that displays the sensor data in one day, allowing for visible changes in the data. The second system is the evaluation system from the user in the form of the selection level of the evaluation results 5 levels: most, more, moderate, low, and very low. To store all the data into the database.

## กิตติกรรมประกาศ

รายงานฉบับนี้ อาจไม่เกิดขึ้นได้หากไม่ได้รับคำปรึกษา และข้อชี้แนะจากอาจารย์ที่ปรึกษา  
โครงการ อาจารย์เกียรติณรงค์ ทองประเสริฐ ซึ่งสนับสนุน และช่วยเหลือในเรื่องต่างๆ จนทำให้  
โครงการวิชาคอมพิวเตอร์นี้สามารถดำเนินโครงการจนสำเร็จเสร็จสิ้นไปด้วยดี ข้าพเจ้ารู้สึกซาบซึ้ง  
ในความอนุเคราะห์จากอาจารย์ ข้าพเจ้าจึงขอขอบคุณท่านไว้ ณ ที่นี้ด้วย

ขอขอบคุณสาขาวิชาวิศวกรรมคอมพิวเตอร์ที่ได้จัดเตรียมสิ่งอำนวยความสะดวก เพื่อให้  
การศึกษาโครงการคอมพิวเตอร์เป็นไปอย่างสะดวกและรวดเร็ว

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ สาขาวิชาวิศวกรรมคอมพิวเตอร์ ที่ให้คำปรึกษา ให้ความ  
ช่วยเหลือ เสมอมา

และที่ขาดไม่ได้ ขอขอบคุณบิดา มารดา และครอบครัวอันเป็นที่เคารพรัก คอยเลี้ยงดูและ  
การสนับสนุนการเรียนรู้อย่างเสมอมา จนทำให้ข้าพเจ้ามีวันนี้ได้ ข้าพเจ้าขอขอบพระคุณมา ณ ที่นี้  
ด้วย

อรรถกร เทพสิทธิ์

# สารบัญ

หน้า

บทคัดย่อภาษาไทย .....	I
บทคัดย่อภาษาอังกฤษ .....	II
กิตติกรรมประกาศ.....	III
สารบัญ .....	IV
สารบัญรูป .....	VII
บทที่ 1 บทนำ .....	1
1.1 ความเป็นมาของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ .....	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ .....	1
1.4 ขอบเขตของโครงการ .....	1
1.5 ขั้นตอนการดำเนินงาน .....	1
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 Web application .....	3
2.2 Front-end.....	3
2.3 Back-end .....	4
2.4 HTML .....	5
2.5 CSS .....	6
2.6 JavaScript.....	6
2.7 PHP .....	7
2.8 JQuery.....	8
2.9 Bootstrap.....	9
2.10 JSON.....	10
2.11 XML .....	10
2.12 File get content .....	11
2.13 Session .....	12

## สารบัญ (ต่อ)

	หน้า
2.14 Web Server .....	13
2.15 MySQL .....	15
2.16 Database.....	16
2.17 ER Diagram .....	17
บทที่ 3 วิธีดำเนินการ .....	20
3.1 ภาพรวมของระบบ .....	20
3.2 Front end.....	21
3.3 ส่วนแสดงผลข้อมูลและติดต่อ Web Service.....	22
3.4 Web Service.....	32
3.5 Database.....	35
บทที่ 4 ผลการวิจัย .....	40
4.1 หน้าหลักของระบบ .....	40
4.2 หน้าแสดงข้อมูลเฉลี่ยจาก Sensor.....	41
4.3 หน้าแสดงข้อมูลจาก Sensor.....	42
4.4 หน้าแสดงข้อมูลที่เก็บจากผู้ใช้งาน.....	43
4.5 หน้าแก้ไขข้อมูลจากผู้ใช้งาน.....	44
4.6 เก็บข้อมูลจากผู้ใช้งาน .....	45
4.7 ตั้งค่าการเก็บข้อมูลจากผู้ใช้งาน .....	45
4.8 ตั้งค่าการแจ้งเตือน.....	46
4.9 หน้า Finish Page .....	47
บทที่ 5 บทสรุปและข้อเสนอแนะ.....	48
5.1 สรุป.....	48
5.2 ปัญหา.....	48
5.3 แนวทางการแก้ไข .....	49
5.4 แนวทางในการพัฒนาต่อ.....	49

## สารบัญ (ต่อ)

	หน้า
บรรณานุกรม.....	50
ภาคผนวก .....	53

# สารบัญรูป

รูป	หน้า
2.1 Front-end.....	3
2.2 Back-end .....	4
2.3 ตัวอย่างโครงสร้างของภาษา PHP .....	8
2.4 รูปแบบการใช้งานฟังก์ชัน file_get_contents() .....	12
2.5 คำอธิบายพารามิเตอร์ของ file_get_contents() .....	12
2.6 ตัวอย่างการใช้งาน file_get_contents().....	12
2.7 ตัวอย่างการกำหนดค่าตัวแปร.....	12
2.8 ตัวอย่างการแสดงผลตัวแปร .....	13
2.9 โมเดลการทำงานของเว็บเซอร์วิส .....	13
2.10 มาตรฐานหลักของเว็บเซอร์วิส.....	14
2.11 มาตรฐานต่างๆ ที่เกี่ยวข้องกับเว็บเซอร์วิส .....	15
2.13 ตัวอย่าง ER Diagram.....	18
2.14 ตัวอย่างการแสดงความสัมพันธ์ระหว่าง entity .....	19
3.1 ภาพรวมของระบบ .....	21
3.2 ตัวอย่าง Code การเพิ่ม style sheet ของ Bootstrap.....	21
3.3 เมนูบน Web Application .....	22
3.4 ตัวอย่าง Code แสดงเมนูบน Web Application.....	22
3.5 ตัวอย่างหน้าเว็บของหน้าอ่านข้อมูลจาก Sensor.....	23
3.6 ตัวอย่าง Code หาวันเดือนปีปัจจุบัน .....	23
3.7 ตัวอย่าง Code เลือกวันเดือนปี.....	24
3.8 ตัวอย่าง Code แสดงรายการที่พบทั้งหมด .....	24
3.9 ตัวอย่าง Code เรียกใช้งานไฟล์ nav_page_top.php .....	24
3.10 ตัวอย่าง Code แถบเปลี่ยนหน้าตารางส่วนบน.....	25
3.11 ตัวอย่าง Code ตำแหน่งข้อมูลจาก JavaScript.....	25
3.12 ตัวอย่าง Code เรียกใช้งานไฟล์ nav_page_bottom.php .....	26
3.13 ตัวอย่าง Code แถบเปลี่ยนหน้าตารางส่วนล่าง .....	26
3.14 ตัวอย่าง Code การเรียกใช้ jQuery .....	26
3.15 ตัวอย่าง Code JavaScript ส่วนของฟังก์ชันการทำงานค้นหา .....	28

## สารบัญรูป (ต่อ)

รูป	หน้า
3.16 ตัวอย่าง Code เรียกใช้งานไฟล์ webservice_addr.php.....	28
3.17 ตัวอย่าง Code ของไฟล์ webservice_addr.php .....	29
3.18 ตัวอย่าง Code กำหนดค่าตัวแปร .....	29
3.19 ตัวอย่าง Code ส่วนกำหนดข้อมูลและติดต่อกับฐานข้อมูล.....	30
3.20 ตัวอย่าง Code เชื่อมโยงการทำงาน.....	30
3.21 ตัวอย่าง Code php สร้างตารางแสดงข้อมูล .....	31
3.22 ตัวอย่าง Code นำข้อมูลจากฐานข้อมูลมาแสดงลงในตาราง .....	31
3.23 หน้าแสดงสถานะการนำเข้าหรืออัปเดตข้อมูลลงฐานข้อมูล .....	32
3.24 ตัวอย่าง Code เชื่อมโยงสถานะของหน้า Finish Page .....	32
3.25 ตัวอย่าง Code เรียกใช้งานไฟล์ dbconfig.php.....	33
3.26 ตัวอย่าง Code สร้างการเชื่อมต่อกับฐานข้อมูล .....	33
3.27 ตัวอย่าง Code sql การ Select ข้อมูล .....	34
3.28 ตัวอย่าง Code sql การ Insert ข้อมูล.....	34
3.29 ตัวอย่าง Code sql การ Update ข้อมูล .....	35
3.30 ตัวอย่าง Code กำหนดค่าสถานะข้อมูลและการส่งข้อมูลกลับ.....	35
3.31 ER Diagram .....	36
3.32 แสดงตารางฐานข้อมูล.....	36
3.33 ฐานข้อมูลตาราง alert.....	37
3.34 ฐานข้อมูลตาราง inputdata_detail .....	37
3.35 ฐานข้อมูลตาราง inputdata_group.....	38
3.36 ฐานข้อมูลตาราง inputdata_title.....	38
3.37 ฐานข้อมูลตาราง outputdata .....	38
3.38 ฐานข้อมูลตาราง performance .....	39
4.1 แสดงหน้าหลักของระบบ .....	40
4.2 แสดงปุ่มการทำงาน .....	41
4.3 แสดงหน้าอ่านข้อมูลจาก Sensor .....	41
4.4 ปฏิทินในหน้าอ่านข้อมูลเฉลี่ยจาก Sensor.....	42
4.5 แสดงหน้าอ่านข้อมูลจาก Sensor .....	42

## สารบัญรูป (ต่อ)

รูป	หน้า
4.6 ปฏิทินในหน้าอ่านข้อมูลจาก Sensor.....	43
4.7 แสดงหน้าแสดงข้อมูลการเก็บข้อมูล.....	43
4.8 ปฏิทินในหน้าเก็บข้อมูล.....	44
4.9 แสดงหน้าแก้ไขข้อมูล.....	44
4.10 ปุ่มการทำงานของหน้าแก้ไขข้อมูล.....	45
4.11 แสดงหน้าเก็บข้อมูล.....	45
4.12 ปุ่มการทำงานของหน้าเก็บข้อมูล.....	45
4.13 หน้าตั้งค่าการเก็บข้อมูล.....	46
4.14 ปุ่มการทำงานหน้าตั้งค่าการเก็บข้อมูล.....	46
4.15 แสดงหน้าตั้งค่าการแจ้งเตือน.....	47
4.16 แสดงหน้า Finish Page.....	47

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาของปัญหา

ทุกวันนี้เทคโนโลยีได้เข้ามาเป็นส่วนหนึ่งของชีวิตประจำวันของทุกคน เทคโนโลยีต่าง ๆ นั้น ได้ถูกคิดค้นและพัฒนาขึ้นทุกวัน เนื่องจากมีข้อมูลที่ได้จากการเก็บข้อมูลเซ็นเซอร์ในแปลง การเกษตรจึงมีแนวคิดที่จะใช้เทคโนโลยีมาช่วยพัฒนาให้การเกษตรมีประสิทธิภาพมากขึ้น โดยการนำข้อมูลเซ็นเซอร์ที่มีมาวิเคราะห์ เพื่อหาความสัมพันธ์และแนวโน้มสำหรับใช้เป็นส่วนหนึ่งในการตัดสินใจในการทำการเกษตร แต่เนื่องจากเฉพาะข้อมูลเซ็นเซอร์ไม่เพียงพอต่อการนำมาวิเคราะห์จึงมีการจัดทำเว็บแอปพลิเคชันเพื่อแสดงข้อมูลของเซ็นเซอร์ที่มีอยู่และเก็บรวบรวมข้อมูลอื่นๆที่จำเป็นสำหรับการนำข้อมูลเซ็นเซอร์มาวิเคราะห์

### 1.2 ความมุ่งหมายและวัตถุประสงค์

- 1) เพื่อศึกษาความรู้เกี่ยวกับการเรียนรู้ของเครื่องประเภทต่างๆ
- 2) เพื่อศึกษาความรู้เกี่ยวกับการพัฒนาเว็บแอปพลิเคชัน
- 3) เพื่อพัฒนาเว็บแอปพลิเคชันสำหรับแสดงข้อมูลเซ็นเซอร์และเก็บข้อมูลอื่นๆที่จำเป็นในการวิเคราะห์

### 1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ได้รับความรู้เกี่ยวกับการเรียนรู้ของเครื่องประเภทต่างๆ
- 2) ได้รับความรู้เกี่ยวกับการพัฒนาเว็บแอปพลิเคชัน
- 3) สามารถพัฒนาเว็บแอปพลิเคชันสำหรับแสดงข้อมูลเซ็นเซอร์และเก็บข้อมูลอื่นๆที่จำเป็นในการวิเคราะห์

### 1.4 ขอบเขตของโครงการ

- 1) สามารถจัดทำเว็บแอปพลิเคชันสำหรับให้เกษตรกรใช้งาน

### 1.5 ขั้นตอนการดำเนินงาน

- 1) ศึกษาทฤษฎีการเรียนรู้ของเครื่อง ประเภทของการเรียนรู้
- 2) ศึกษาการพัฒนาเว็บแอปพลิเคชัน
- 3) สร้าง web service สำหรับส่งและรับข้อมูลระหว่างเว็บแอปพลิเคชันและฐานข้อมูล

- 4) สร้างและพัฒนาเว็บแอปพลิเคชันสำหรับแสดงและเก็บข้อมูล

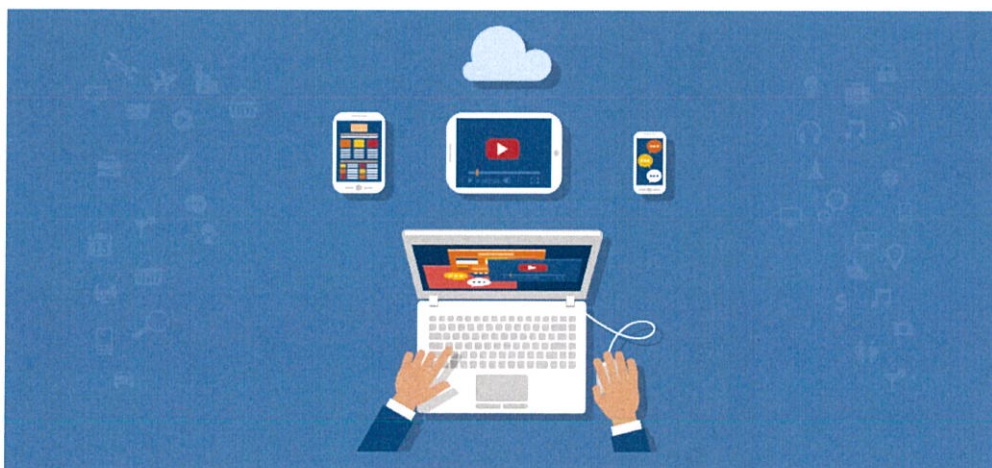
## บทที่ 2

# ทฤษฎีที่เกี่ยวข้อง

### 2.1 Web application

Web application หรือโปรแกรมประยุกต์บนเว็บ คือ โปรแกรมประยุกต์ที่สามารถเข้าถึงได้ด้วยโปรแกรมประเภท Browser ผ่านเครือข่ายคอมพิวเตอร์ (อินเทอร์เน็ต) มีความสามารถในการอัปเดต และดูแล โดยไม่ต้องแจกจ่ายหรือติดตั้งบนเครื่องของผู้ใช้งาน เนื่องจากจะถูกติดตั้งไว้ที่ Server เพื่อคอยให้บริการกับ Client ไม่จำกัดว่าเครื่องที่ใช้จะเป็นระบบปฏิบัติการหรืออุปกรณ์อะไร และลดข้อจำกัดเรื่องสถานที่การใช้งานอีกด้วย เช่น เว็บเมลล์ การพาณิชย์อิเล็กทรอนิกส์ การประมูลออนไลน์ กระดานสนทนา บล็อก วิกี เป็นต้น

### 2.2 Front-end



รูป 2.1 Front-end

#### 2.2.1 ความหมายของ Front-end

สำหรับ developer จะเรียกว่า หน้าบ้าน ซึ่งเป็นส่วนติดต่อกับผู้ใช้ (User interface) คือ ส่วนที่แสดงให้เห็นเป็นรูปลักษณะของเว็บไซต์ ทั้งเนื้อหา ภาพแบนเนอร์ รูปภาพต่างๆ ลิงค์ และเรื่องราวต่าง ๆ เพื่อให้ user ทั่วไปสามารถเห็นและเข้ามาใช้งานส่วนต่างๆของเว็บไซต์ได้

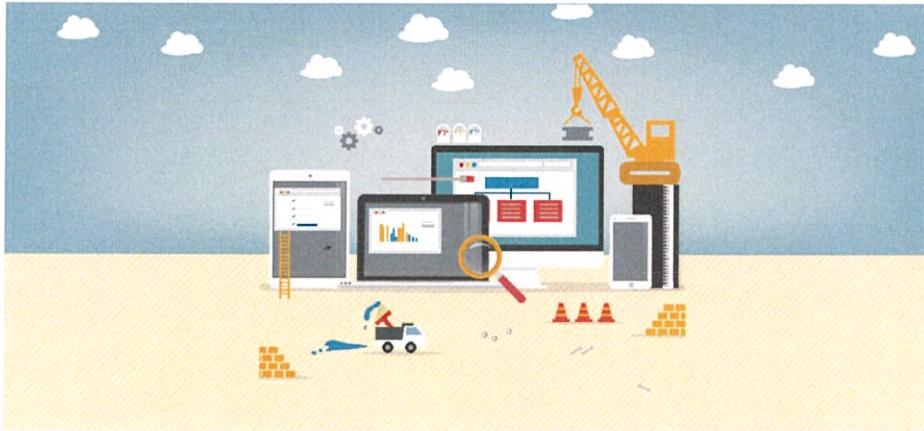
#### 2.2.2 ความสำคัญของ Front-end

ส่วนนี้จะแสดงหน้าตาของเว็บไซต์ให้ผู้เข้าชมเห็น การออกแบบก็เป็นส่วนที่ช่วยดึงดูดและทำให้ผู้สนใจเว็บไซต์ ทั้งความสวยงาม การใช้งานเว็บไซต์ที่เข้าใจง่าย สะดวก และทำให้ผู้เข้าชมเห็นว่าเว็บไซต์มีการพัฒนาอยู่ตลอด

### 2.2.3 ทักษะที่ใช้กับงานด้าน Front-end

- 1) Graphic Designer
- 2) Web Designer
- 3) Web Developer (JavaScript, CSS, HTML5, jQuery)

## 2.3 Back-end



รูป 2.2 Back-end

### 2.3.1 ความหมายของ Back-end

สำหรับ developer จะเรียกว่า หลังบ้าน ซึ่งเป็นส่วนของระบบจัดการเว็บไซต์ เช่น จัดการฐานข้อมูล โครงสร้างเว็บไซต์ การเขียนโค้ดควบคุม XML, text file, JAVA, PHP, C#, C++ เป็นต้น โดยจะมีไว้สำหรับ admin หรือผู้ที่ได้รับอนุญาต เพื่อทำการบริหารจัดการส่วนต่างๆภายในเว็บไซต์ เช่นการเพิ่ม ลบ แก้ไข เปลี่ยนแปลงข้อมูลเว็บไซต์ เป็นต้น

### 2.3.2 ความสำคัญของ Back-end

ส่วนของ Back-end มีความสำคัญเป็นอย่างยิ่ง ไม่ว่าจะเป็นการรักษาความปลอดภัยของข้อมูล ทั้ง username password ข้อมูลเว็บไซต์ต่างๆ หาก Back-end มีการ update ก็ต้องทำการ backup ข้อมูลเดิมเก็บไว้ และตรวจสอบให้ดีก่อนทำการ update รวมไปถึงการทำงานของเว็บไซต์ ความเร็วในการแสดงผล

### 2.3.3 ทักษะที่ใช้กับงานด้าน Back-end

Web Developer (JavaScript, CSS, HTML5, jQuery)

### 2.3.4 ความเกี่ยวข้องระหว่าง Front-end กับ Back-end

Front-end กับ Back-end จะมีความสำคัญต่อกันเป็นอย่างมาก หาก Front-end แสดงผลช้าหรือเกิดข้อผิดพลาด ส่วนของ Back-end ก็ต้องทำการแก้ไขปัญหา หรือถ้าหากว่า Back-end ไม่

ทำการปรับปรุงเว็บไซต์อย่างสม่ำเสมอ จะส่งผลต่อเว็บไซต์เป็นอย่างมาก ไม่ว่าจะเป็น เว็บไซต์ล์่าหลัง ขาดความน่าสนใจ ผู้เยี่ยมชมลดลง ก่อให้เกิดผลกระทบหลาย ๆ ด้าน

## 2.4 HTML

เป็นภาษามาตรฐานที่ใช้ในการพัฒนาเอกสารในรูปแบบของเว็บเพจบนระบบเครือข่าย ซึ่งพัฒนาจากภาษา SGML (Standard Generalized Markup Language) โดย นาย Tim Berners - Lee การเรียกใช้งานเอกสารทำได้โดยการเรียกใช้ผ่านโปรแกรมเว็บเบราว์เซอร์ เช่น Google Chrome, Internet Explorer, Opera เป็นต้น สามารถเรียกใช้ได้ในระบบปฏิบัติการหลายชนิด

ภาษา HTML จะมีการใช้คำสั่งที่เรียกว่า แท็ก (Tag) มาใช้ในการกำหนดลักษณะและรูปแบบของเอกสาร ซึ่งจะประกอบไปด้วยเครื่องหมายน้อยกว่า (<) ตามด้วยชื่อแท็ก และปิดแท็กด้วยเครื่องหมายมากกว่า (>) ชื่อแท็กสามารถเป็นได้ทั้งตัวพิมพ์เล็กและตัวพิมพ์ใหญ่ ซึ่งแท็กในภาษา HTML สามารถแบ่งได้เป็น 2 ชนิด คือ

แท็กที่ประกอบด้วยแท็กเปิดและแท็กปิด เช่น <HTML>...</HTML>

แท็กที่ไม่มีแท็กปิด เช่น <BR> ไม่ต้องปิดท้ายด้วยแท็ก </BR>

### 2.4.1 รูปแบบของการเขียน HTML

ในการเขียน HTML จะต้องจัดวางตำแหน่งแต่ละแท็กดังนี้

```
<HTML>
  <HEAD>
    <TITLE>ชื่อเอกสาร</TITLE>
  </HEAD>
  <BODY>
    &ข้อมูลเอกสาร
  </BODY>
</HTML>
```

### 2.4.2 ความหมายของแต่ละแท็ก

<HTML>...</HTML>	เป็นแท็กกำหนดจุดเริ่มต้นและจุดสิ้นสุดของเอกสาร
<HEAD>...</HEAD>	เป็นแท็กกำหนดจุดเริ่มต้นและจุดสิ้นสุดของส่วนกำหนดค่าเริ่มต้นของเอกสาร HTML เช่น ชื่อของเอกสาร
<TITLE>...</TITLE>	เป็นแท็กกำหนดชื่อของเอกสาร
<BODY>...</BODY>	เป็นแท็กกำหนดจุดเริ่มต้นและจุดสิ้นสุดของส่วนแสดงข้อมูลของเอกสาร

## 2.5 CSS

CSS ย่อมาจาก Cascading Style Sheets เป็นภาษาที่มีรูปแบบการเขียน Syntax ที่เฉพาะ และ ถูกกำหนดมาตรฐานโดย W3C (World Wide Web Consortium) เช่นเดียวกับ HTML และ XHTML ใช้สำหรับตกแต่งเอกสาร HTML/ XHTML ให้มีหน้าตา สี สัน ตัวอักษร เส้นขอบ พื้นหลัง ระยะห่าง ฯลฯ อย่างที่เราต้องการ ด้วยการกำหนดคุณสมบัติให้กับ Element ต่างๆ ของ HTML เช่น `<body>`, `<p>`, `<h1>` เป็นต้น

### 2.5.1 ประโยชน์ของ CSS

- 1) การใช้ CSS ในการจัดรูปแบบการแสดงผล จะช่วยลดการใช้ภาษา HTML ในการตกแต่งเอกสารเว็บเพจ ทำให้ code ภายในเอกสาร HTML เหลือเพียงส่วนเนื้อหา ทำให้เข้าใจง่ายขึ้น การแก้ไขเอกสารทำได้ง่ายและรวดเร็ว
- 2) เมื่อ code ภายในเอกสาร HTML ลดลง ทำให้ขนาดไฟล์เล็กลง จึงดาวน์โหลดได้เร็ว
- 3) สามารถกำหนดรูปแบบการแสดงผลจากคำสั่ง style sheet ชุดเดียวกัน ให้มีผลกับเอกสาร HTML ทั่วหน้า หรือทุกหน้าได้ ทำให้เวลาแก้ไขหรือปรับปรุงทำได้ง่ายขึ้น ไม่ต้องไล่ตามแก้ที่ HTML tag ต่างๆ ทั่วทั้งเอกสาร
- 4) สามารถควบคุมการแสดงผลให้เหมือนกัน หรือใกล้เคียงกัน ได้ในหลาย Web Browser
- 5) สามารถกำหนดการแสดงผลในรูปแบบที่เหมาะสมกับสื่อชนิดต่างๆ ไม่ว่าจะเป็นการแสดงผลบนหน้าจอ, บนกระดาษเมื่อสั่งพิมพ์, บนมือถือ หรือบน PDA โดยที่เป็นเนื้อหาเดียวกัน
- 6) ทำให้เป็นเว็บไซต์ที่มีมาตรฐาน ปัจจุบันการใช้ attribute ของ HTML ตกแต่งเอกสารเว็บเพจ นั้นล้าสมัยแล้ว W3C แนะนำให้เราใช้ CSS แทน ดังนั้นหากเราใช้ CSS กับเอกสาร HTML ของเรา ก็จะช่วยให้เข้ากับเว็บเบราว์เซอร์ในอนาคตได้ดี

## 2.6 JavaScript

JavaScript คือภาษาสำหรับการเขียนโปรแกรมบนระบบเครือข่าย ถูกพัฒนาขึ้นโดย เน็ตสเคป คอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่า Live Script ต่อมาได้ร่วมมือกับ บริษัท ซัน ไมโครซิสเต็มส์ ปรับปรุงระบบของเบราว์เซอร์เพื่อให้สามารถติดต่อใช้งานกับภาษาจาวาได้ และได้ปรับปรุง Live Script ใหม่เมื่อ ปี 2538 แล้วตั้งชื่อใหม่ว่า JavaScript

JavaScript เป็นภาษาสคริปต์เชิงวัตถุ ใช้ในการสร้างและพัฒนาเว็บไซต์ร่วมกับภาษา HTML เพื่อให้เว็บไซต์มีความน่าสนใจมากยิ่งขึ้น มีการตอบสนองต่อผู้ใช้งานมากขึ้น มีลูกเล่นและสามารถเคลื่อนไหวได้

### 2.6.1 ความสามารถของ JavaScript

- 1) ทำให้เว็บไซต์มีปฏิสัมพันธ์กับผู้ใช้งานมากขึ้น เนื่องจากมีคำสั่งที่ใช้ในการตอบสนองต่อผู้ใช้งาน
- 2) สามารถเขียนหรือเปลี่ยนแปลง HTML Element ได้ กล่าวคือ สามารถเปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้ สามารถแสดงหรือซ่อนเนื้อหาได้แบบง่ายๆ
- 3) สามารถใช้ในการตรวจสอบข้อมูลได้ เช่น ในการกรอกข้อมูลตามเว็บไซต์ต่างๆ
- 4) สามารถใช้ในการตรวจสอบผู้ใช้งานได้ เช่น ชนิดของเว็บเบราว์เซอร์ที่ใช้งาน
- 5) มีการสร้าง Cookies ซึ่งเป็นการเก็บข้อมูลของผู้ใช้ในเครื่องคอมพิวเตอร์ของผู้ใช้งาน

### 2.6.2 ข้อดีและข้อเสียของ JavaScript

เนื่องจากการทำงานของ JavaScript เกิดขึ้นบนเบราว์เซอร์ (Client-Side Script) จึงสามารถใช้งาน JavaScript ในเว็บเพจได้ ไม่ว่าจะใช้เซิร์ฟเวอร์อะไรหรือที่ไหนก็ตาม แตกต่างกับภาษาสคริปต์อื่นๆ เช่น Perl PHP หรือ ASP ที่มีการแปลความหมายและทำงานที่เซิร์ฟเวอร์ (Server-Side Script) ซึ่งต้องใช้งานบนเซิร์ฟเวอร์ที่สนับสนุนภาษาเหล่านี้เท่านั้น

แต่ด้วยข้อดีดังกล่าว ก็นำมาซึ่งข้อจำกัด คือการไม่สามารถรับส่งข้อมูลต่างๆกับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเครื่องเซิร์ฟเวอร์เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ใช้งานเพื่อนำไปเก็บที่เครื่องเซิร์ฟเวอร์ การทำงานในลักษณะดังกล่าวจึงยังต้องอาศัยภาษา Server-Side Script อยู่

## 2.7 PHP

PHP เป็นภาษาที่มีการแปลความหมายและทำงานที่เซิร์ฟเวอร์ (Server-Side Script) มีความสามารถในการรับข้อมูลที่ส่งมาจากฟอร์มของ HTML และมีความสามารถในการติดต่อกับฐานข้อมูล ถูกสร้างขึ้นโดย Rasmus Lerdorf ในปี ค.ศ.1994 ต่อมาเมื่อผู้ให้ความสนใจเป็นจำนวนมาก จึงได้ออกเป็นแพ็คเกจ "Personal Home Page" ซึ่งเป็นที่มาของชื่อภาษา PHP

### 2.7.1 โครงสร้างของภาษา PHP

ภาษา PHP มีลักษณะเป็น Embedded Script คือสามารถฝังคำสั่ง PHP ไว้ในเว็บเพจร่วมกับแท็กของภาษา HTML ได้ สคริปต์ของภาษา PHP จะเริ่มต้นด้วย "<?" ตามด้วยคำสั่งเรียก

ฟังก์ชันหรือข้อความ และปิดท้ายด้วย “?”” ดังตัวอย่างสคริปต์ที่ใช้ในการแสดงข้อความ “I’m a PHP script” โดยใช้คำสั่ง echo ซึ่งใช้ในการแสดงผลของภาษา PHP

```

1 <html>
2 <head>
3 <title>Example 1 </title>
4 </head>
5 <body>
6 <?
7     echo"Hi, I'm a PHP script!";
8 ?>
9 </body>
10 </html>

```

รูป 2.3 ตัวอย่างโครงสร้างของภาษา PHP

### 2.7.2 ความสามารถของภาษา PHP

- เป็นภาษาที่มีลักษณะเป็น Open Source ผู้ใช้งานสามารถดาวน์โหลด Code ไปใช้งานได้โดยไม่เสียค่าใช้จ่าย
- เป็นภาษาที่มีการแปลความหมายและทำงานที่เซิร์ฟเวอร์ (Server-Side Script) ซึ่งจะอ่าน โค้ด และทำงานที่เซิร์ฟเวอร์ จากนั้นจึงส่งผลลัพธ์ที่ได้จากการประมวลผลมาที่เครื่องของผู้ใช้ในรูปแบบของ HTML จึงไม่ส่งผลต่อการทำงานของเครื่อง Client
- สามารถทำงานได้อย่างมีประสิทธิภาพในหลายระบบปฏิบัติการ
- สามารถทำงานได้กับเว็บเซิร์ฟเวอร์หลายชนิด เช่น Personal Web Server (PWS) Apache OmniHTTPd และ Internet Information Service (IIS) เป็นต้น
- สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming)
- สามารถทำงานร่วมกับระบบจัดการฐานข้อมูลที่สนับสนุนได้หลากหลาย เช่น Oracle MySQL FilePro Solid FrontBase mSQL และ MS SQL เป็นต้น
- อนุญาตให้ผู้ใช้สร้างเว็บไซต์ซึ่งทำงานผ่านโปรโตคอลชนิดต่างๆ ได้ เช่น LDAP IMAP SNMP POP3 และ HTTP เป็นต้น
- Code ภาษา PHP สามารถเขียนและอ่านในรูปแบบของ XML ได้

## 2.8 JQuery

jQuery คือ JavaScript Library ซึ่งถูกออกแบบมาเพื่อให้การเขียน JavaScript นั้นมีความสะดวกและง่ายขึ้น เพราะว่าการนำ JavaScript เอาไปประยุกต์กับงานจำพวกเว็บ (Client-side

JavaScript) นั้นเป็นสิ่งที่ยุ่งยาก ไม่ว่าจะเป็นเรื่องความไม่เข้ากันของ Web Browser แต่ละค่าย, DOM หรือ API เป็นต้น

ดังนั้น jQuery จึงรวมเอา Object และ Function ต่างๆ ที่จำเป็นมารวบรวมไว้ในรูปแบบของ Library พอเป็นเช่นนี้แล้ว ไม่ว่าจะโค้ดที่คุณเขียนจะใช้ JavaScript หลายบรรทัดขนาดไหน ก็สามารถทำให้สั้นลงได้ อาจทำให้เหลือสั้นเพียงแค่บรรทัดเดียวเท่านั้น

### 2.8.1 ฟังก์ชันต่างๆของ jQuery

- 1) HTML/DOM manipulation
- 2) CSS manipulation
- 3) HTML event methods
- 4) Effects and animations
- 5) AJAX
- 6) Utilities

### 2.8.2 ความสามารถของ jQuery

- 1) ความสามารถในการทำงานแบบ ajax
- 2) การสร้าง animation ได้แบบง่ายๆเลย ไม่ว่าจะเป็นทำรูปให้เคลื่อนที่ หรือ DIV เช่นเอา mouse จับ DIV ลากไปมา
- 3) ความสามารถในการ binds หรือการผูก หรือจับ function ที่เขาเขียนขึ้นให้ทำงานร่วมกับ function อื่นๆ
- 4) สามารถจัดการกับ CSS (style sheet) ของ element นั้นๆได้
- 5) ค้นหา element ที่เราต้องการและจัดการ เพิ่มหรือลบ Attributes ที่เราต้องการได้
- 6) ทำ Effect ต่างๆกับ Element ที่เราต้องการ เช่นการ hide DIV ที่เราต้องการ
- 7) การดักเหตุการณ์ต่างๆ

## 2.9 Bootstrap

Bootstrap คือ Frontend Framework ที่ รวม HTML, CSS และ JavaScript เข้าด้วยกัน พัฒนาขึ้นโดยทีมงานจาก Twitter หรือ Twitter.com เพื่อตอบโจทย์ในด้าน Responsive Web Design มีระบบ Grid เข้ามาช่วยและมีการคำนวณค่าหน้าจอพร้อมกับการปรับขนาดของเว็บให้สามารถแสดงผลได้กับทุกขนาดหน้าจอโดยอัตโนมัติ รองรับการผลิตในทุกลูกข่าย

### 2.9.1 ส่วนประกอบของ Bootstrap

สิ่งที่ Bootstrap ให้มา มี 4 อย่าง ดังนี้

- 1) Scaffolding  
Grid System จำนวน 12 คอลัมน์
- 2) Base CSS  
Style Sheets สำหรับ HTML Elements พื้นฐาน เช่น Typography Tables Forms และ Images
- 3) Components  
Style Sheets สำหรับสิ่งที่เราต้องใช้บ่อยๆ เช่น Navigation Breadcrumbs และ Pagination
- 4) JavaScript  
jQuery Plugins ต่างๆ เช่น Modal Carousel และ Tooltip

## 2.10 JSON

JSON (JavaScript Object Notation) คือ รูปแบบของข้อมูลที่ใช้สำหรับแลกเปลี่ยนข้อมูลที่มีขนาดเล็ก ซึ่งคนสามารถทำความเข้าใจได้ง่าย และสามารถถูกสร้างและอ่าน โดยเครื่องได้ง่าย มันถูกกำหนดภายใต้ภาษา JavaScript

JSON เป็นรูปแบบข้อมูลตัวอักษรที่มีความเป็นอิสระอย่างสมบูรณ์ แต่จะมีหลักการการเขียนที่คุ้นเคยกับนักเขียน โปรแกรมภาษาต่างๆ ได้ ไม่ว่าจะเป็น ภาษา C, C++, C#, Java, JavaScript, Perl, Python และอื่นๆ คุณสมบัติเหล่านี้ทำให้ JSON เป็นภาษาแลกเปลี่ยนข้อมูลที่มีสมบูรณ์แบบ

มาตรฐานของฟอร์แมต JSON คือ RFC 4627 มี Internet media type เป็น application/json และมีนามสกุลของไฟล์เป็น .json

คำสั่งที่สำคัญมี 2 คำสั่งคือ

- 1) json\_encode ใช้ในการจัดรูปแบบตัวแปรของ PHP เป็น JSON String เพื่อใช้ในการส่งไปยัง JavaScript
- 2) json\_decode ใช้ในการจัดรูปแบบข้อมูล JSON String ที่ได้รับมาเป็นตัวแปรของ PHP

## 2.11 XML

XML ย่อมาจาก Extensible Markup Language คือภาษาหนึ่งที่ใช้ในการแสดงผลข้อมูล ซึ่งภาษาที่ใช้กำหนดรูปแบบของคำสั่งภาษา HTML หรือที่เรียกว่า Meta Data ซึ่งจะใช้สำหรับกำหนดรูปแบบของคำสั่ง Markup ต่าง ๆ ถ้าเปรียบเทียบกับภาษา HTML จะแตกต่างกันที่ HTML ถูกออกแบบมาเพื่อการแสดงผลอย่างเดียวเท่านั้น เช่น ให้แสดงผลตัวเล็ก ตัวหนา ตัวเอียง เหมือนที่คุณเคยเห็นในเว็บเพจทั่วไป แต่ภาษา XML นั้นถูกออกแบบมาเพื่อเก็บข้อมูล โดยทั้งข้อมูลและ

โครงสร้างของข้อมูลนั้นๆไว้ด้วยกัน ส่วนการแสดงผลก็จะใช้ภาษาเฉพาะซึ่งก็คือ XSL (Extensible Stylesheet Language)

ภาษา XML มีโครงสร้างที่ประกอบด้วยแท็กเปิด และแท็กปิด เช่นเดียวกับภาษา HTML แต่ภาษา XML คุณสามารถสร้างแท็กรวมทั้งกำหนดโครงสร้างของข้อมูลได้เอง ซึ่งความสามารถตรงนี้ตัวภาษา ทำไม่ได้เพราะภาษา HTML ถูกกำหนดแท็กตายตัวโดย W3C หรือ World Wide Web Consortium อาจกล่าวได้ว่า XML เป็นส่วนเสริมของ HTML เพราะตัว XML ไม่สามารถแสดงผลได้ในตัวของมันเอง หากต้องการแสดงผลที่ถูกต้อง จะต้องมีการใช้ร่วมกับภาษาอื่น เช่น HTML, JSP, PHP, ASP หรือภาษาอื่น ๆ ที่สนับสนุน XML จะมีนามสกุลเป็น .XML สามารถสร้างขึ้นจาก โปรแกรมประเภท Text Editor ใดก็ได้ เช่น Notepad, Editplus, Dreamweaver, MS Word เป็นต้น

### 2.11.1 จุดเด่นของ XML

- 1) ดูเอกสารได้ง่าย สะดวก และได้ผลดีเหมือน HTML
- 2) สนับสนุนการประยุกต์ใช้กับงานต่าง ๆ และสนับสนุน โปรแกรมประยุกต์ต่าง ๆ
- 3) เขียนง่าย
- 4) อ่านได้ด้วยมนุษย์ โดยไม่ต้องอาศัย โปรแกรมหรือเครื่องมือช่วยแปล
- 5) การเขียน XML ทำได้ด้วยการใช้ Text editor ทั่วไป และไม่ต้องการเครื่องมือที่ซับซ้อน
- 6) ใช้เป็นตัวควบคุมข้อมูล (Meta data) จึงเป็นแนวทางในการขนส่งข้อมูล และสร้างการเชื่อมโยงระหว่างแอปพลิเคชันได้ง่าย
- 7) สนับสนุน UNICODE ทำให้ใช้ได้กับหลากหลายภาษา และผสมกันได้หลากหลายภาษา
- 8) ดึงเอกสาร XML มาใช้งานได้ง่าย และใช้ร่วมกับ โปรแกรมประยุกต์อื่นได้ง่าย เช่น โปรแกรม DB2, Oracle, SAP เป็นต้น
- 9) นำไปประยุกต์ใช้ในการดำเนินกิจกรรมบนเครือข่ายได้มาก เช่น E-Business, EDI, E-Commerce, การจัดการ Supply chain / Demand chain management, การดำเนินการแบบ intranet และ Web Base Application

## 2.12 File get content

ฟังก์ชัน `file_get_contents()` ใช้ในการอ่านไฟล์ลงในสตริง โดยฟังก์ชันนี้เป็นวิธีในการอ่านเนื้อหาของไฟล์ที่เป็นสตริง เพราะมันจะใช้เทคนิคการทำแผนที่หน่วยความจำว่านี้ได้รับการสนับสนุนโดยเซิร์ฟเวอร์เพื่อเพิ่มประสิทธิภาพ

```
file_get_contents(path,include_path,context,start,max_length)
```

## รูป 2.4 รูปแบบการใช้งานฟังก์ชัน file\_get\_contents()

พารามิเตอร์	ลักษณะ
path	จำเป็นต่อใช้ ระบุไฟล์ที่จะอ่าน
include_path	ไม่จำเป็น. ตั้งค่าพารามิเตอร์นี้ '1' ถ้าคุณต้องการค้นหาไฟล์ใน include_path (in php.ini) ได้เป็นอย่างดี
context	ไม่จำเป็น. ระบุบริบทของการจัดการไฟล์ บริบทคือชุดของตัวเลือกที่สามารถปรับเปลี่ยนพฤติกรรมของกระแสที่ สามารถข้ามโดยใช้โมฆะ
start	ไม่จำเป็น. ระบุที่อยู่ไบต์ที่เริ่มต้นการอ่าน พารามิเตอร์นี้ถูกบันทึกอยู่ใน PHP 5.1
max_length	ไม่จำเป็น. ระบุวิธีการหลายไบต์ในการอ่าน พารามิเตอร์นี้ถูกบันทึกอยู่ใน PHP 5.1

## รูป 2.5 คำอธิบายพารามิเตอร์ของ file\_get\_contents()

```
<?php
echo file_get_contents("test.txt");
?>
```

## รูป 2.6 ตัวอย่างการใช้งาน file\_get\_contents()

### 2.13 Session

เซสชัน (Session) คือตัวแปรในภาษา PHP ซึ่งมีคุณสมบัติทุกอย่างเหมือนกับตัวแปรปกติทั้งเก็บค่าข้อความหรือตัวเลข ตลอดทั้งตัวแปรที่เป็น Object ต่างๆ ได้ แต่จะมีความพิเศษกว่าตัวแปรปกติคือ ตัวแปรเซสชันและค่าตัวแปรจะยังคงอยู่ไม่ว่าเราจะเปลี่ยนหน้าจากหน้าหนึ่งไป สู่อีกหน้าหนึ่งซึ่งเป็นข้อดีของตัวแปรนี้เช่น การกำหนดค่าตัวแปรเซสชันในหน้า sample1.php แล้วเอาค่าที่ได้มาแสดงผลในหน้า sample2.php โดยจะได้ค่าที่กำหนดจากหน้า sample1.php มาทั้งหมด ตัวแปรเซสชันจะยังมีค่าคงอยู่เสมอจนกระทั่งตัวแปรหมดอายุหรือถูกทำการลบค่า ทั้ง ตัวแปรเซสชันนิยมนำมาใช้ในการรับส่งค่าข้อมูลที่เก็บเป็นความลับที่จะต้อง ใช้ในการระบุตัวตนใช้สิทธิของระบบ เช่น การยืนยันตัวตนด้วย Username และ Password เพื่อให้ได้ค่าตัวแปรนี้มาระบุตัวตน ตัวแปรเซสชันจึงมีความปลอดภัยในการเก็บรักษาข้อมูลได้อย่างมาก

```
1 <?php
2 session_start();
3 ob_start();
4 $_SESSION['abc']="Hello";
5 ?>
```

## รูป 2.7 ตัวอย่างการกำหนดค่าตัวแปร

```

1 <?php
2 session_start();
3 ob_start();
4 echo $_SESSION['abc']; // ผลลัพธ์คือแสดงข้อความ Hello
5 ?>

```

รูป 2.8 ตัวอย่างการแสดงผลตัวแปร

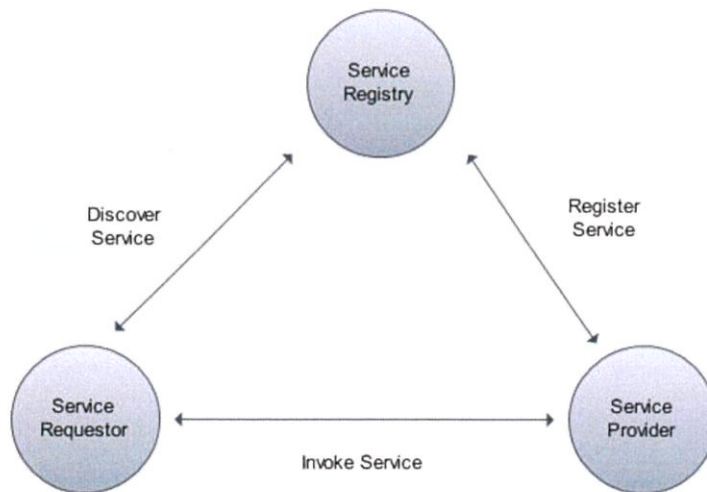
## 2.14 Web Server

เว็บเซอร์วิส (Web Service) เป็นเทคโนโลยีที่ออกแบบมา สำหรับใช้ในการแลกเปลี่ยนข้อมูล ระหว่างเครื่องคอมพิวเตอร์ผ่านทางระบบเครือข่าย โดยอินเทอร์เน็ตเฟสและการสื่อสารจะใช้ ภาษา XML ทำให้สามารถเรียกใช้จากต่าง Platform หรือใช้ภาษาคอมพิวเตอร์ที่ต่างกันได้

เว็บเซอร์วิส ถือเป็นซอฟต์แวร์คอมโพเนนต์ (Software Component) แบบ Loosely Couple คือ แต่ละคอมโพเนนต์จะเป็นอิสระต่อกันและมีฟังก์ชันที่สมบูรณ์ในตัว สามารถรับอินพุตเข้ามา เพื่อประมวลผล และจะส่งผลลัพธ์กลับออกไป หรือสามารถสร้างฟังก์ชันที่เฉพาะการ ประมวลผลซึ่งจะไม่เกี่ยวข้องกัส่วนแสดงผล (Presentation Logic) และยังสามารถเรียกใช้ จากซอฟต์แวร์ประยุกต์อื่นๆ ผ่านทางอินเทอร์เน็ตโพรโทคอล

### 2.14.1 โมเดลการทำงานของเว็บเซอร์วิส

แบ่งการทำงานออกเป็น 3 ส่วนด้วยกัน ดังรูปที่ 2.9



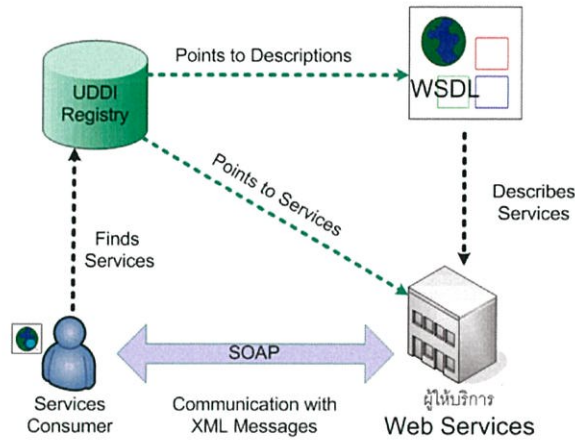
รูป 2.9 โมเดลการทำงานของเว็บเซอร์วิส

ผู้ให้บริการ (Service Provider) คือ ผู้ที่ทำหน้าที่ในการให้บริการ โดยจะต้องทำการ ลงทะเบียนบริการที่ให้บริการไว้กับ Service Registry

ผู้ใช้บริการ (Service Requestor) คือ ผู้ร้องขอใช้บริการ โดยการเข้ามาค้นหาบริการและ รายละเอียดของบริการที่ต้องการ แล้วทำการเรียกใช้บริการ

Service Registry หรือ Service Broker คือ ตัวกลางที่ทำหน้าที่ในการรวบรวมบริการต่างๆที่ผู้ให้บริการได้ทำการลงทะเบียนบริการไว้ และช่วยค้นหาบริการกับรายละเอียดของบริการ

### 2.14.2 มาตรฐานหลักของเว็บเซอร์วิส



รูป 2.10 มาตรฐานหลักของเว็บเซอร์วิส

#### 1) Extensible Markup Language (XML)

XML เป็นภาษา Markup ที่ทาง W3C (World Wide Web Consortium) เป็นกำหนดมาตรฐานไว้ โดย XML สามารถสร้างรูปแบบในการแสดงผลข้อมูล ใสและกำหนด Tag ได้อย่างอิสระ ซึ่งปัจจุบัน XML ถือเป็นมาตรฐานสำคัญที่ใช้ในการกำหนดรูปแบบเอกสารอิเล็กทรอนิกส์อีกด้วย

#### 2) Simple Object Access Protocol (SOAP)

SOAP เป็นเอกสารรูปแบบภาษา XML ที่ใช้เป็นโพรโทคอลสำหรับแลกเปลี่ยนข้อมูลระหว่างเว็บเซอร์วิส (Message Protocol)

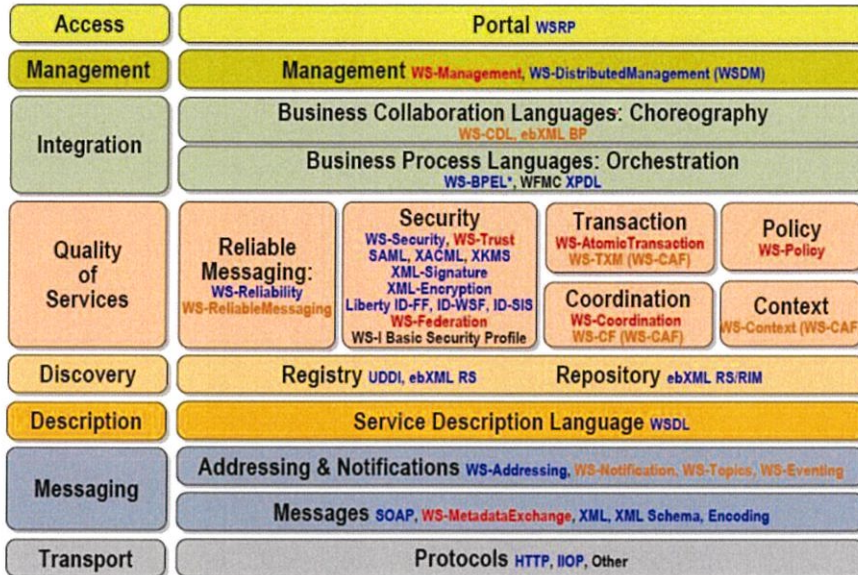
#### 3) Web Services Description Language (WSDL)

WSDL เป็นเอกสารรูปแบบภาษา XML ที่ใช้สำหรับอธิบายข้อมูลเกี่ยวกับเว็บเซอร์วิส โดยแบ่งออกเป็น 2 ส่วน คือ Abstract (Interface) เป็นส่วนที่อธิบายเกี่ยวกับการดำเนินงานต่างๆ (Operation) กับค่าอินพุต-เอาต์พุต และ Concrete (Implementation) เป็นส่วนที่อธิบายเกี่ยวกับโพรโทคอล รูปแบบของข้อมูล และตำแหน่งในรูปแบบของ URL

#### 4) Universal Description, Discovery and Integration (UDDI)

UDDI เป็นตัวกำหนดมาตรฐานของ Service Registry โดยจะเก็บข้อมูลเกี่ยวกับบริการที่ได้ทำการลงทะเบียนไว้ ซึ่งผู้ให้บริการจะสามารถลงทะเบียนเพื่อประกาศเว็บเซอร์วิสได้ แล้วผู้ใช้บริการสามารถเข้ามาค้นหาข้อมูลบริการต่างๆที่ต้องการได้

โครงสร้างของ UDDI ประกอบด้วย ข้อมูลเกี่ยวกับบริษัท (Business Entity) ข้อมูลเกี่ยวกับบริการ (Business Service) ข้อมูลทางเทคนิค (Binding Template) ข้อมูลเกี่ยวกับการติดต่อ (tModel) และข้อมูลเกี่ยวกับความสัมพันธ์ระหว่าง Business Entity (Publisher Assertion)



รูป 2.11 มาตรฐานต่างๆ ที่เกี่ยวข้องกับเว็บเซอร์วิส

## 2.15 MySQL

MySQL คือ โปรแกรมระบบจัดการฐานข้อมูลเพื่อจัดเก็บข้อมูลอย่างเป็นระบบโดยใช้ภาษา SQL ซึ่งพัฒนาโดยบริษัท MySQL AB ประเทศสวีเดน โดยต้องใช้ใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่น เช่น ทำงานร่วมกับเว็บเซิร์ฟเวอร์ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานในฝั่งเครื่องเซิร์ฟเวอร์ (Server-Side Script) เช่น ภาษา PHP, APS.net, JSP เป็นต้นทำงานร่วมกับโปรแกรมประยุกต์ (Application Program) เช่น ภาษา Visual Basic .NET, Java, C++ เป็นต้น MySQL สามารถทำงานได้บนหลายระบบปฏิบัติการ และเป็นระบบฐานข้อมูล Open Source ที่ถูกนำไปใช้งานมากที่สุด

### 2.15.1 ความสามารถและการทำงานของโปรแกรม MySQL

#### 2.15.1.1 เป็นทั้งตัวฐานข้อมูลและระบบจัดการฐานข้อมูล

ระบบจัดการฐานข้อมูล (DBMS : DataBase Management System) จะทำหน้าที่เป็นตัวกลางในการจัดการกับฐานข้อมูล เพื่อความสะดวกในการจัดการกับข้อมูลจำนวนมาก

#### 2.15.1.2 เป็นระบบจัดการฐานข้อมูลแบบ Relational

ฐานข้อมูลแบบ Relational จะเก็บข้อมูลในรูปแบบของตารางแทนการเก็บข้อมูลทั้งหมดลงในไฟล์ไฟล์เดียว สามารถเชื่อมโยง รวมหรือจัดกลุ่มจากแต่ละตารางได้ตาม

ต้องการโดยใช้ภาษา SQL ซึ่งเป็นภาษามาตรฐานในการเข้าถึงข้อมูล จึงทำให้สามารถทำงานได้อย่างรวดเร็วและมีความยืดหยุ่นสูง

### 2.15.1.3 สามารถใช้งานแบบ Open Source

ผู้ใช้งานสามารถดาวน์โหลดโปรแกรม MySQL มาใช้งานได้โดยไม่มีค่าใช้จ่าย และสามารถปรับแต่งตามการใช้งานได้ตามที่ต้องการ

### 2.15.2 ด้านกราฟิก (Graphical)

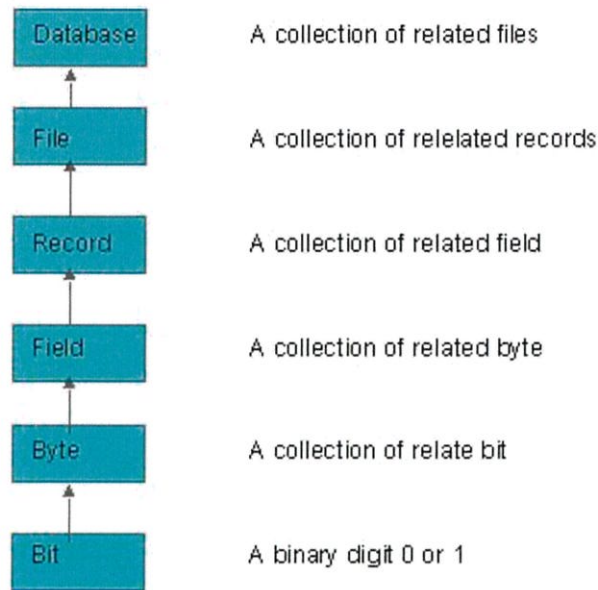
MySQL รองรับการทำงานด้านกราฟิก (GUI) ผ่านโปรแกรมต่างๆมากมาย เช่น phpMyAdmin, Navicat, OpenOffice.org, SQLBuddy, Sequel Pro, SQLYog, Toad for MySQL, Adminer, DaDaBIK เป็นต้น

### 2.15.3 การเขียนโปรแกรม (Programming)

MySQL รองรับและสนับสนุนการทำงานบนหลายๆระบบปฏิบัติการ เช่น AIX, BSDi, FreeBSD, HP-UX, eComStation, i5/OS, IRIX, Linux, Mac OS X, Microsoft Windows, NetBSD, Novell NetWare, OpenBSD, OpenSolaris, OS/2 Warp, QNX, Solaris, Symbian, SunOS เป็นต้น

## 2.16 Database

ฐานข้อมูล (Database) หมายถึง กลุ่มของข้อมูลที่มีความสัมพันธ์กัน นำมาเก็บรวบรวมเข้าไว้ด้วยกันอย่างมีระบบและข้อมูลที่ประกอบกันเป็นฐานข้อมูลนั้น ต้องตรงตามวัตถุประสงค์การใช้งานขององค์กรด้วยเช่นกัน เช่น ในสำนักงานก็รวบรวมข้อมูล ตั้งแต่หมายเลขโทรศัพท์ของผู้ที่มาติดต่อจนถึงการเก็บเอกสารทุกอย่างของสำนักงาน ซึ่งข้อมูลส่วนนี้จะมีส่วนที่สัมพันธ์กันและเป็นที่ต้องการนำออกมาใช้ประโยชน์ต่อไปภายหลัง ข้อมูลนั้นอาจจะเกี่ยวกับบุคคล สิ่งของสถานที่ หรือเหตุการณ์ใด ๆ ก็ได้ที่เราสนใจศึกษา หรืออาจได้มาจากการสังเกต การนับหรือการวัดก็เป็นได้ รวมทั้งข้อมูลที่เป็นตัวเลข ข้อความ และรูปภาพต่าง ๆ ก็สามารถนำมาจัดเก็บเป็นฐานข้อมูลได้ และที่สำคัญข้อมูลทุกอย่างต้องมีความสัมพันธ์กัน เพราะเราต้องการนำมาใช้ประโยชน์ต่อไปในอนาคต



**รูป 2.12 องค์ประกอบของฐานข้อมูล**

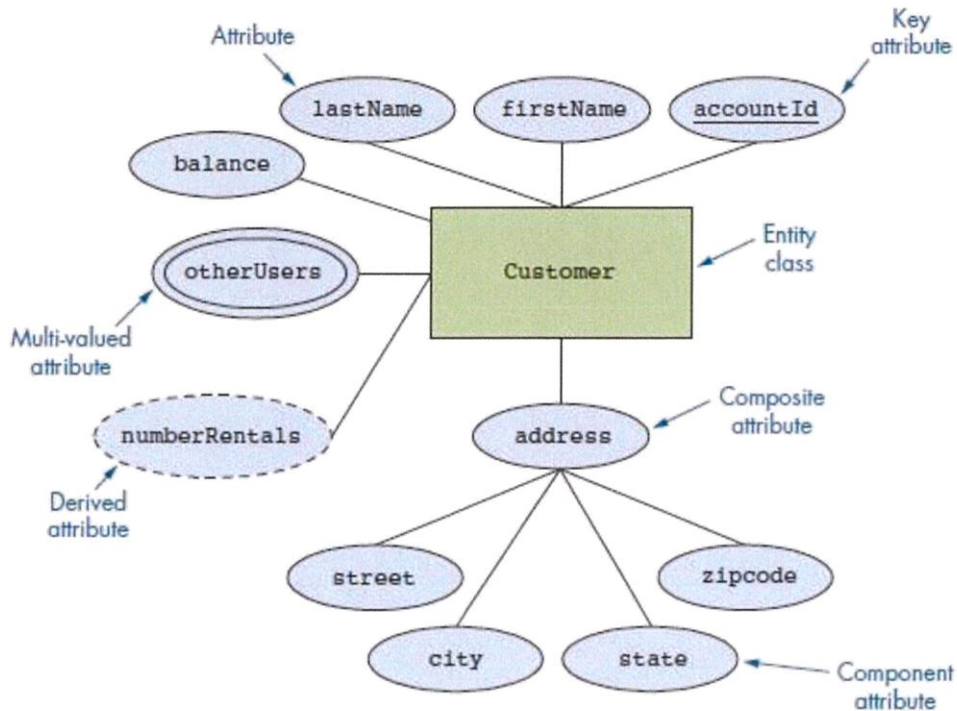
- บิต (bit) ย่อมาจาก Binary Digit คือข้อมูลคอมพิวเตอร์ 1 บิต จะมีแค่ 2 ค่าคือ 0 กับ 1
- ไบต์ (byte) ถ้าเราเอาไบต์หลายๆไบต์มาเรียงต่อกันเราก็จะเรียกว่า เขตข้อมูล (field)
- เขตข้อมูล เรียกว่าอีกอย่างว่า เอกสารก็ได้ (record) เช่น เราสร้างเอกสารมา 1 เอกสารในนั้นเก็บประวัติของนักศึกษาไว้ก็จัดเป็นเอกสาร 1
- แฟ้มข้อมูล (File) ถ้าเรานำเอาเอกสารหลายๆเอกสารมาเก็บไว้ด้วยกันก็จะได้เป็นแฟ้มข้อมูลขึ้นมา 1 แฟ้ม
- และสุดท้าย การจัดเก็บ แฟ้มข้อมูล หลายๆแฟ้มไว้ด้วยกันเราจะเรียกการจัดเก็บแบบนี้ว่า ฐานข้อมูล หรือ Database ที่เราเรียกกัน เช่น การเก็บรายชื่อนักศึกษา

## 2.17 ER Diagram

ER Diagram (Entity-Relationship Diagrams) คือ แบบจำลองที่ใช้อธิบายโครงสร้างของฐานข้อมูลซึ่งเขียนออกมาในลักษณะของรูปภาพ การอธิบายโครงสร้างและความสัมพันธ์ของข้อมูล (Relationship) ประกอบด้วย

- เอนทิตี (Entity) เป็นวัตถุ หรือสิ่งของที่เราสงใจในระบบงานนั้น ๆ
- แอททริบิว (Attribute) เป็นคุณสมบัติของวัตถุที่เราสงใจ
- ความสัมพันธ์ (Relationship) คือ ความสัมพันธ์ระหว่างเอนทิตี

ER Diagram มีความสำคัญต่อการพัฒนาระบบงานฐานข้อมูล Application ต่างๆ ที่ต้องการการเก็บข้อมูลอย่างมีระบบ มีโครงสร้าง ดังนั้น ER Diagram จึงใช้เพื่อเป็นเอกสารในการสื่อสารระหว่าง นักออกแบบระบบ และนักพัฒนาระบบ เพื่อให้สื่อสารอย่างตรงกัน และเป็นสากลอีกด้วย



รูป 2.13 ตัวอย่าง ER Diagram

### 2.17.1 Key attribute

คือ attribute ที่ถูกกำหนดให้เป็น key ของ entity โดยแทนที่ด้วย attribute ที่ถูกขีดเส้นใต้ จากรูป 2.13 account Id ถูกขีดเส้นใต้เพื่อแสดงว่า attribute นี้ถูกใช้เป็น key ของ entity Customer

### 2.17.2 Multi-valued attribute

คือ attribute ที่มีค่าบรรจุอยู่มากกว่าหนึ่งค่า โดยแทนที่ด้วยวงกลมรูปไข่ซ้อนกันสองวง จากรูป 2.13 attribute ที่ชื่อ otherUsers เป็น multi-valued attribute หมายถึง Customer สามารถมีผู้ใช้คนอื่น ๆ ที่ใช้บัญชีของ Customer ได้ (ผู้ใช้คนอื่นอาจจะเป็นญาติกับ Customer เช่น ลูก ภรรยา น้อง)

### 2.17.3 Derived attribute

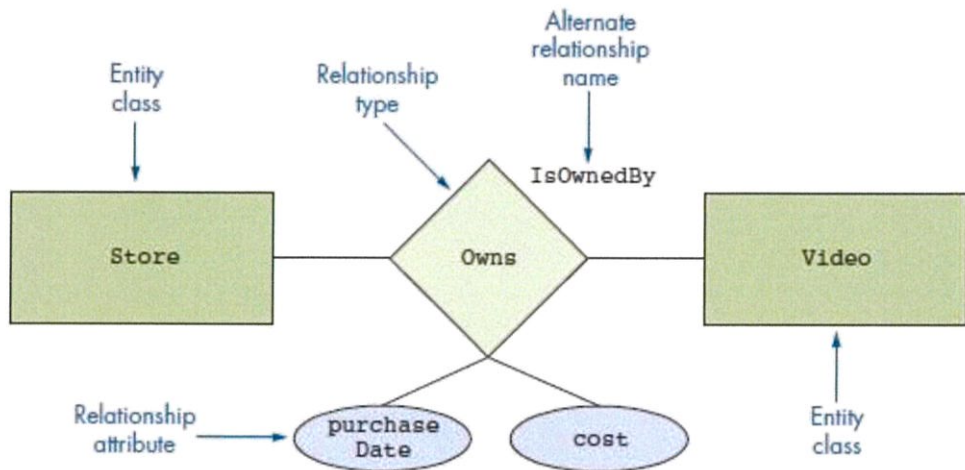
คือ attribute ที่ค่าของมันได้มาจากการคำนวณของ attribute อื่น โดยแทนที่ด้วยวงกลมรูปไข่ที่เป็นเส้นประ จากรูป 2.13 attribute ที่ชื่อ numberRentals หรือจำนวนที่เช่าซึ่งได้มาจากการรวมจำนวนสินค้าที่เช่าทั้งหมดเข้าด้วยกัน

### 2.17.4 Composite attribute

คือ attribute ที่สามารถแยกออกเป็น attribute ย่อย ๆ ได้หลาย attribute แทนที่โดยชื่อ attribute ในวงกลมรูปไข่ที่มีเส้นตรงลากไปเชื่อมโยงกับ attribute หลัก รูป 2.13 attribute ที่ชื่อ address สามารถแยกออกเป็น attribute ย่อยที่ชื่อ street, city, state, zipcode ได้อีก

### 2.17.5 Relationship Types

ใช้แสดงความสัมพันธ์ระหว่าง entity โดยแทนที่ด้วยรูปสี่เหลี่ยมข้าวหลามตัด จากรูป 2.14 Store Owns (เป็นเจ้าของ) Video (ในกรณีที่อ่านจากซ้ายไปขวา) หรือ Video IsOwnedBy (ถูกเป็นเจ้าของโดย) Store (ในกรณีที่อ่านจากขวาไปซ้าย) พึงสังเกตว่าชื่อของ relationship types จะต้องเป็นคำกริยาเสมอ และความสัมพันธ์สามารถมี attribute ของตัวเองได้ เช่นในภาพ ความสัมพันธ์ Owns มี attribute คือ purchase Date และ cost



รูป 2.14 ตัวอย่างการแสดงความสัมพันธ์ระหว่าง entity

### 2.17.6 Cardinality Constraints

ใช้แสดงถึงข้อกำหนดของความสัมพันธ์ระหว่าง entity แบ่งออกเป็นสองแบบคือ

#### 2.17.7 Cardinality ratio

ใช้แสดงถึงอัตราส่วนของความสัมพันธ์ แทนที่ด้วยตัวเลข 1, M และ N

1 : 1 แทนความสัมพันธ์แบบหนึ่งต่อหนึ่ง

1 : N แทนความสัมพันธ์แบบหนึ่งต่อหลาย

M : N แทนความสัมพันธ์แบบหลายต่อหลาย

### 2.17.8 Participation

ใช้แสดงการมีส่วนร่วมในความสัมพันธ์ของสมาชิกใน entity แทนที่ด้วยเส้นตรงหรือเส้นคู่

total (เส้นคู่) ทุก ๆ สมาชิกที่อยู่ใน entity จะต้องอยู่ในความสัมพันธ์ทั้งหมด

partial (เส้นเดี่ยว) บางส่วนของสมาชิกที่อยู่ใน entity เท่านั้นที่อยู่ในความสัมพันธ์

## บทที่ 3

### วิธีดำเนินการ

ระบบนำเข้าข้อมูลจากผู้ใช้เทียบกับข้อมูลที่ได้จากเซ็นเซอร์ผ่าน WEB APPLICATION นี้มีวัตถุประสงค์เพื่อพัฒนาเว็บแอปพลิเคชันสำหรับให้เกษตรกรใช้งานเพื่อดูข้อมูลของเซ็นเซอร์และบันทึกข้อมูลเพิ่มเติม

#### 3.1 ภาพรวมของระบบ

แบ่งการทำงานออกเป็น 3 ส่วน คือ

##### 3.1.1 Web Application

เป็นส่วนสำหรับผู้ใช้งาน ในส่วนนี้จะมีการทำงานทั้งหมด 4 ระบบคือ

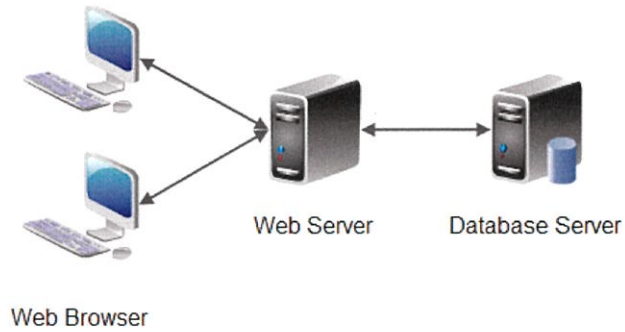
- 1) ระบบอ่านข้อมูลจาก Sensor เป็นส่วนของการแสดงข้อมูลเฉลี่ยของข้อมูลเซ็นเซอร์ ในหนึ่งวันเพื่อให้เกษตรกรสามารถดูข้อมูลจากเซ็นเซอร์ได้ง่าย ในส่วนนี้จะมีการแสดงข้อมูลซึ่งประกอบด้วย อุณหภูมิของพื้นดิน ทิศทางลม ความเร็วลม ความเร็วลมกระโชก ปริมาณน้ำ อุณหภูมิ ความชื้นสัมพัทธ์ จุดน้ำค้าง แบตเตอรี่
- 2) ระบบเก็บข้อมูล เป็นส่วนบันทึกข้อมูลเพื่อเก็บข้อมูลที่ต้องการเพิ่มสำหรับนำมาวิเคราะห์ต่อไป โดยแบ่งออกเป็นระดับต่างๆ เช่นคุณภาพของผลผลิต แบ่งผลประเมินที่ต้องบันทึกเป็น ดีมาก ดี ปานกลาง น้อย น้อยที่สุด
- 3) ระบบแจ้งเตือน เป็นการแสดงข้อความเตือนผู้ใช้งานเมื่อมีค่าที่อ่านได้จากเซ็นเซอร์มากหรือน้อยเกินกว่าที่ได้ตั้งค่าไว้
- 4) ระบบตั้งค่า เป็นระบบสำหรับตั้งค่าข้อมูลเซ็นเซอร์ใช้สำหรับการแจ้งเตือน
- 5) ระบบแสดงกราฟข้อมูลจาก Sensor เพื่อให้สามารถดูแนวโน้มการเปลี่ยนแปลงข้อมูลของแต่ละวันได้ซึ่งแสดงในรูปแบบของกราฟเส้นตรง

##### 3.1.2 Web Service

เป็นส่วนสำหรับประมวลผลข้อมูลคำนวณและวิเคราะห์ข้อมูลต่างๆที่รับมาจากรฐานข้อมูลเพื่อส่งให้ส่วนของ Web Application แสดงผล และรับข้อมูลจากส่วน Web Application เพื่อส่งไปบันทึกข้อมูลที่ฐานข้อมูล

##### 3.1.3 Database

เป็นส่วนซึ่งฐานข้อมูลที่มีการบันทึกข้อมูลของเซ็นเซอร์และใช้สำหรับเก็บข้อมูลที่ต้องบันทึกผลเพิ่มเติมสำหรับใช้ในการวิเคราะห์



รูป 3.1 ภาพรวมของระบบ

## 3.2 Front end

เป็นส่วนแสดงหน้าตาของเว็บไซต์ให้ผู้เข้าชมเห็น การออกแบบก็เป็นส่วนที่ช่วยดึงดูดและทำให้ผู้สนใจเว็บไซต์ ทั้งความสวยงาม การใช้งานเว็บไซต์ที่เข้าใจง่ายและสะดวก

### 3.2.1 Bootstrap

Bootstrap คือ Front-end Framework ตัวหนึ่ง ที่จะช่วยให้การพัฒนาเว็บไซต์ของเราเร็วขึ้น ง่ายขึ้น และเป็นระบบมากขึ้น ซึ่งคำว่า Bootstrap นี้ในภาษาอังกฤษมันมักจะหมายถึง “สิ่งที่ช่วยให้ง่ายขึ้น” หรือ “สิ่งที่ทำได้ด้วยตัวของมันเอง” การใช้งาน Bootstrap สามารถทำได้ดังนี้

- 1) ดาวน์โหลดไฟล์ Bootstrap มาติดตั้งก่อนซึ่งสามารถแตกไฟล์ออกมาได้ 3 folders คือ css img และ js
- 2) สำหรับการเรียกใช้ให้เพิ่ม style sheet ของ Bootstrap เข้าไปที่ <head> ของ html ที่เขียนขึ้นดังรูป

```
<head>
  <link rel="stylesheet" type="text/css" href="../css/bootstrap.min.css">
</head>
```

รูป 3.2 ตัวอย่าง Code การเพิ่ม style sheet ของ Bootstrap

### 3.2.2 Header

ในส่วนนี้จะ เป็นเมนูสำหรับเลือกเปลี่ยนหน้าไปยังหน้าต่างๆ โดยแบ่งออกเป็น Home เฉลี่ยต่อวัน แสดงทั้งหมด เก็บข้อมูล ตั้งค่าการเก็บข้อมูล ตั้งค่าการแจ้งเตือน

### รูป 3.3 เมนูบน Web Application

Code เมนูสำหรับเปลี่ยนหน้าไปยังหน้าต่างๆจะมีการใส่สัญลักษณ์ด้วย `<span class="">` เพื่อเพิ่มรูปสัญลักษณ์หน้าเมนู และใส่ link สำหรับเวลาที่เมนูแล้วจะไปยังหน้านั้นๆ ด้วยคำสั่ง `<a href="">` ในเครื่องหมาย “ ” จะใส่ที่อยู่ของหน้าที่ต้องการเปิดสำหรับเปลี่ยนหน้าดังรูป

```
<div class="navbar-brand ">Project 2</div>
</div>
<div id="navbar" class="navbar-collapse collapse">
  <ul class="nav navbar-nav">

    <li class=""><a href=".."><span class="glyphicon glyphicon-home" ></span>Home</a></li>
    <li><a href=".."><span class="glyphicon glyphicon-floppy-open"></span> เลือยตัววัน</a></li>
    <li><a href=".."><span class="glyphicon glyphicon-floppy-open"></span> แสดงทั้งหมด</a></li>

    <li class=""><a href=".."><span class="glyphicon glyphicon-floppy-save"></span> เก็บข้อมูล</a></li>
    <li class=""><a href=".."><span class="glyphicon glyphicon-tasks"></span> ส่งค่าการเก็บข้อมูล</a></li>
    <li class=""><a href=".."><span class="glyphicon glyphicon-asterisk"></span> ส่งค่าการแจ้งเตือน</a></li>

  </ul>
</div>
```

### รูป 3.4 ตัวอย่าง Code แสดงเมนูบน Web Application

## 3.3 ส่วนแสดงผลข้อมูลและติดต่อ Web Service

ในส่วนนี้จะแบ่งออกเป็น 3 ส่วนสำคัญเพื่อส่งข้อมูลสำคัญไปใช้สำหรับดึงข้อมูลจาก database และนำข้อมูลที่ได้ออกมาแสดงในรูปแบบตารางเพื่อให้ง่ายในการนำเสนอ มีการแบ่งออกเป็น 2 ไฟล์โดยแบ่งเป็นส่วนนำเสนอและส่วนประมวลผลคือ ไฟล์ `indexall.php` ที่เป็นส่วน front end สำหรับแสดงหน้าเว็บออกมาดังรูปด้านล่าง และไฟล์ `indexall_interface.php` สำหรับติดต่อกับ Web Service และประมวลผลต่างๆออกมาส่งให้หน้าแสดงผล

Project 2 Home เติบโตสวน แสดงทั้งหมด เก็บข้อมูล ส่งผลการเก็บข้อมูล ส่งผลการแจ้งเตือน

## อ่านข้อมูลจาก Sensor

ค้นหาจากรันที่ 19/05/2560   1

จำนวนรายการทั้งหมด 4 รายการ  2

หน้า 1 / 1  3

วันที่	เวลา	Soil Temperature (C)	Wind Dir	Wind Speed (m/s)	Gust Speed (m/s)	Water Content Crop 2	Water Content Crop 3	Temp (C)	Relative Humidity	Dew Point (C)	Battery (V)
19/05/2560	06:15:00	30.67	115	3.9	5.3	0.55	1.31	33.16	75.94	25.65	4.31
19/05/2560	06:10:00	34.09	110	0.4	4.4	1.77	0.14	37.36	91.19	29.84	4.31
19/05/2560	06:05:00	34.58	126	0	0.5	0.78	0.98	35.1	92.65	21.21	4.31
19/05/2560	06:00:00	31.18	109	2.3	5.6	1.18	1.28	22.34	82.61	25.48	4.31

หน้า 1 / 1  5

© 2017, King Mongkut's Institute of Technology Ladkrabang

รูป 3.5 ตัวอย่างหน้าเว็บของหน้าอ่านข้อมูลจาก Sensor

### 3.3.1 Html Code

เป็นส่วนของการแสดงผลนำข้อมูลที่ได้จากส่วนของ JavaScript มาแสดงผลและส่งข้อมูลที่ต้องใช้ในการค้นหาให้กับส่วนของ JavaScript

- 1) เป็นส่วนสำหรับเลือกวันที่ที่ต้องการค้นหาข้อมูลจากฐานข้อมูลในหมายเลข 1 จากรูปด้านบน ในส่วนของ php code เป็นการแสดงวันเดือนปีปัจจุบันในรูปแบบมีเครื่อง / ชั้นในแต่ละส่วน วันเดือนปีนี้จะเป็ค่าพื้นฐานแสดงในกรอบหมายเลข 1 จากรูปด้านบนเมื่อกดเข้ามาที่หน้านี้

```
<?php
date_default_timezone_set('Asia/Bangkok');
$dateOfRequest = date("Y-m-d H:i:s");
$day=substr($dateOfRequest,8,2);
$month=substr($dateOfRequest,5,2);
$year=substr($dateOfRequest,0,4);
$show_year = ($year + 543);
$start_date = $day."/". $month."/". $show_year;
?>
```

รูป 3.6 ตัวอย่าง Code หาวันเดือนปีปัจจุบัน

Html code แสดงผลในกรอบหมายเลข 1 จากรูปด้านบน เป็นส่วนในการเลือกวันเดือนปี เพื่อใช้ในส่วนของ JavaScript เพื่อส่งต่อให้กับ Web Service

```
<div class="panel">
  <div class="panel-body" style="height: 65px;">
    <div class="form-inline col-sm-6">
      <label class=""><span></span></label>
      <input type="text" class="form-control mdate" style="width:150px" id="start_date" placeholder="วันเริ่มต้น" value="">
      <button class="btn btn-black" type="button" onclick="search()"><span class="glyphicon glyphicon-search"></span></button>
    </div>
  </div class="col-sm-6"></div>
</div>
```

### รูป 3.7 ตัวอย่าง Code เลือกวันเดือนปี

- 2) ส่วนแสดงจำนวนรายการที่พบทั้งหมดในกรอบหมายเลข 2 จากรูปด้านบน โดยส่วนนี้จำนวนรายการจะมาจากส่วนของ JavaScript ที่ได้ข้อมูลจาก Web Service แล้วนำมาแสดงใน `<div class="form-control" id="count_mem"></div>`

```
<div class="panel" style="background-color:RGBA(255,255,255,0.5);">
  <div class="panel-body">
    <div class="col-sm-2">
      <div id="outline_count_mem" class="input-group input-group-sm">
        <span class="input-group-addon">
          <span class="glyphicon glyphicon-briefcase"></span><span></span>
        </span>
        <div class="form-control" id="count_mem"></div>
        <span class="input-group-addon"><span></span></span>
      </div>
    </div>
  </div>
</div>
```

### รูป 3.8 ตัวอย่าง Code แสดงรายการที่พบทั้งหมด

- 3) ส่วนแสดงแถบเลื่อนเปลี่ยนหน้าส่วนบน ซึ่งจะมีทั้งเปลี่ยนไปหน้าแรก เปลี่ยนไปหน้าก่อนหน้า เปลี่ยนไปหน้าที่กำหนด เปลี่ยนไปหน้าต่อไป และเปลี่ยนไปหน้าสุดท้าย มีการแสดงจำนวนหน้าและหน้าที่กำลังเปิดอยู่ดังในกรอบหมายเลข 3 จากรูปด้านบน ในไฟล์ indexall.php จะใส่ code เรียกใช้แถบเปลี่ยนหน้าส่วนบนจากไฟล์ nav\_page\_top.php ใน folder includes เพื่อให้ง่ายในการใช้งานเวลาทำแบบเดียวกันในหน้าอื่นๆก็สามารถแก้ไขแถบเปลี่ยนหน้าได้ในไฟล์เดียวกัน

```
<?php require ("../includes/nav_page_top.php");?>
```

### รูป 3.9 ตัวอย่าง Code เรียกใช้งานไฟล์ nav\_page\_top.php

Html code ของหน้า nav\_page\_top.php แบ่งเป็น 6 ส่วนคือ ปุ่มหน้าแรก ปุ่มหน้าก่อนหน้า แถบแสดงหน้าที่แสดงอยู่และหน้าทั้งหมด ปุ่มกรอกเลขไปยังหน้าที่ ปุ่มหน้าถัดไป ปุ่มหน้าสุดท้าย

```
<!-- PAGE NAVIGATER TOP -->
<div class="row" style="padding-left:15px;padding-right:15px;">
<div class="panel panel-default" style="border-radius: 50px;">
<div class="panel-body" style="padding:0px;">
<div class="col-sm-12">
<nav>
<ul class="pager" style="margin:10px">
<li class="previous"><!-- previous -->
<a href="#" onclick="to_first_page()" style="margin-top: 5px;">
<span class="glyphicon glyphicon-step-backward"></span>
</a>
</li><!-- END previous -->
<li class="click_page_prev" id="page_previous_1" ><!-- previous -->
<a href="#" id="a_previous_1" onclick="pagedog()" ><span class="glyphicon glyphicon-chevron-left"></span></a>
</li><!-- END previous -->
<li><div class="well well-sm page_number" id="page_display_1" ></div></li><!-- Show page / allpage-->
<li class="go_to"><!-- current -->
<a class="page_input" style="padding-left: 0px;">
<div class="input-group" style="width:150px;z-index:0">
<input type="text" id="go_page_field" class="form-control" placeholder="xxxxxx" style="height:32px">
<span class="input-group-btn">
<button class="btn btn-black" type="button" style="height:32px" id="go_btn" onclick="go_to_page():">
<span class="glyphicon glyphicon-play-circle"></span>
</button>
</span>
</div>
</div>
</a>
</li><!-- END current -->
<li class="click_page_next" id="page_next_1" ><!-- next -->
<a href="#" id="a_next_1" onclick="pagedog()"><span class="glyphicon glyphicon-chevron-right"></span></a>
</li><!-- END next -->
<li class="next"> <!-- next -->
<a href="#" onclick="to_last_page()" style="margin-top: 5px;"><span class="glyphicon glyphicon-step-forward"></span></a>
</li><!-- END next -->
</ul>
</nav>
</div>
</div>
</div>
</div>
<!-- END PAGE NAVIGATER TOP -->
```

### รูป 3.10 ตัวอย่าง Code แถบเปลี่ยนหน้าตารางส่วนบน

- 4) แสดงตารางแสดงข้อมูลจากฐานข้อมูลในกรอบหมายเลข 4 จากรูปด้านบน โดยในส่วนนี้จะมี Html code เพื่อใช้อ้างอิง id = " result" สำหรับแสดงข้อมูลจาก JavaScript มาแสดงในตำแหน่งของ id

```
<div class="row">
<div class="col-sm-12">
<div class="table-responsive">
<div id = "result"></div>
</div>
</div>
</div>
```

### รูป 3.11 ตัวอย่าง Code ตำแหน่งข้อมูลจาก JavaScript

- 5) ส่วนแสดงแถบเลื่อนเปลี่ยนหน้าส่วนล่าง ซึ่งจะมีทั้งเปลี่ยนไปหน้าแรก เปลี่ยนไปหน้าก่อนหน้า เปลี่ยนไปหน้าที่กำหนด เปลี่ยนไปหน้าต่อไป เปลี่ยนไปหน้าสุดท้าย และมีการแสดงจำนวนหน้าดังในกรอบหมายเลข 3 จากรูปด้านบน ในไฟล์ indexall.php จะใส่ code เรียกใช้ แถบเปลี่ยนหน้าส่วนบนจากไฟล์

nav\_page\_bottom.php ใน folder includes เพื่อให้ง่ายในการใช้งานเวลาทำแบบเดียวกัน ในหน้าอื่นๆก็สามารถแก้ไขแถบเปลี่ยนหน้าได้ในไฟล์เดียวกัน

```
<?php require ("../includes/nav_page_bottom.php");?>
```

### รูป 3.12 ตัวอย่าง Code เรียกใช้งานไฟล์ nav\_page\_bottom.php

Html code ของหน้า nav\_page\_bottom.php แบ่งเป็น 5 ส่วนคือ ปุ่มหน้าแรก ปุ่มหน้าก่อนหน้า แถบแสดงหน้าที่แสดงอยู่และหน้าทั้งหมด ปุ่มหน้าถัดไป ปุ่มหน้าสุดท้าย

```
<!-- PAGE NAVIGATER BOTTOM -->
<div class="row" style="padding-left:15px;padding-right:15px;">
<div class="panel panel-default" style="border-radius: 50px;">
<div class="panel-body" style="padding:0px;">
<div class="col-sm-12">
<nav>
<ul class="pager" style="margin:10px">
<li class="previous"><!-- ปุ่มหน้าแรก -->
<a href="#" onclick="to_first_page()" style="margin-bottom: 5px;">
<span class="glyphicon glyphicon-step-backward"></span>
</a>
</li><!-- END ปุ่มหน้าแรก -->
<li class="click_page_prev" id="page_previous_2" ><!-- ปุ่มหน้าก่อนหน้า -->
<a href="#" id="a_previous_2" onclick="page_prev()"><span class="glyphicon glyphicon-chevron-left"></span></a>
</li><!-- END ปุ่มหน้าก่อนหน้า -->
<li><div class="well well-sm page_number" id="page_display_2" ></div></li><!-- Show page / allpage -->
<li class="click_page_next" id="page_next_2" ><!-- ปุ่มหน้าถัดไป -->
<a href="#" id="a_next_2" onclick="page_next()"><span class="glyphicon glyphicon-chevron-right"></span></a>
</li><!-- END ปุ่มหน้าถัดไป -->
<li class="next"> <!-- ปุ่มหน้าสุดท้าย -->
<a href="#" onclick="to_last_page()" style="margin-bottom: 5px;"><span class="glyphicon glyphicon-step-forward"></span></a>
</li><!-- END ปุ่มหน้าสุดท้าย -->
</ul>
</nav>
</div>
</div>
</div>
</div>
<!-- END PAGE NAVIGATER BOTTOM -->
```

### รูป 3.13 ตัวอย่าง Code แถบเปลี่ยนหน้าตารางส่วนล่าง

## 3.3.2 JavaScript Code

เป็นส่วนที่รับค่าตัวแปรที่ใช้ในการค้นหาจากส่วนของ Html นำไปส่งให้กับส่วนของ PHP ประมวลผลและรับค่าที่ประมวลผลเรียบร้อยแล้วกลับมาส่งไปให้ Html

- 1) jQuery เป็น JavaScript Library ที่มีการรวบรวม function ของ JavaScript ต่าง ๆ ให้อยู่ในรูปแบบ Patterns Framework ที่สะดวกและง่ายต่อการใช้งาน มีความยืดหยุ่นรองรับต่อการใช้งาน Cross Browser คือไม่ว่าจะใช้งานบน Web Browser ใด ใน Library ของ jQuery จะมีการเลือกใช้ function ที่สามารถเหมาะสมต่อการทำงานและแสดงผลใน Web Browser ที่กำลังรันอยู่ ซึ่งช่วยลดปัญหาการทำงานที่ผิดพลาดในฝั่งของ Client ได้ การเรียกใช้จะเป็นดังรูปด้านล่าง

```
<script src="../js/jquery-3.1.1.min.js"></script>
```

### รูป 3.14 ตัวอย่าง Code การเรียกใช้ jQuery

- 2) ส่วนในการส่งข้อมูลไปให้หน้า `indexall_interface.php` เพื่อให้ติดต่อกับ Web Service และประมวลผลเพื่อส่งกลับมาให้หน้า `indexall.php` แสดงผลออกมา ดังรูปด้านล่างจะเห็นว่า function การทำงานค้นหาข้อมูลจะแบ่งออกเป็น 4 ส่วนย่อยๆคือ A B C และ D
- A เป็นส่วนประกาศตัวแปรต่างๆและกำหนดค่าให้ตัวแปร เช่น `start_date` เป็นการนำค่าในตำแหน่งของ `id = "start_date"` มาใส่ในตัวแปร `start_date`
  - B เป็นส่วนการส่งข้อมูลรูปแบบ Post ไปให้หน้า `indexall_interface.php` ใช้สำหรับประมวลผลหาจำนวนข้อมูลที่ค้นพบออกมาแสดงในตำแหน่ง `id="count_mem"` ในกรอบหมายเลข 2 จากรูปด้านบน
  - C เป็นส่วนการส่งข้อมูลรูปแบบ Post ไปให้หน้า `indexall_interface.php` ใช้สำหรับหาจำนวนหน้าทั้งหมดของข้อมูลที่จะนำมาแสดง
  - D เป็นส่วนการส่งข้อมูลรูปแบบ Post ไปให้หน้า `indexall_interface.php` ใช้สำหรับเลือกข้อมูลมาแสดงในรูปแบบตารางมาแสดงในตำแหน่ง `id="result"` ในกรอบหมายเลข 4 จากรูปด้านบน

```

<script>
var page_number = 0;
var last_page = 99;
$(document).ready(function ()
{
    search();

});
//ส่วนหน้า interface
function search()
{
    var start_date = "";
    url = "indexall_interface.php";
    start_date = $('#start_date').val();
    //ส่วนหน้า interface
    $.post(url, {
        pj_start_date : start_date ,
        pj_page       : page_number,
        pj_method     : "show"
    }, function (data_ck) {
        $('#count mem').html(data_ck); //จำนวนทั้งหมด
    });
    $.post(url, {
        pj_start_date : start_date ,
        pj_page       : page_number,
        pj_method     : "find_last_page"
    }, function (data_ck) {
        checkpage(data_ck); //เลือกหน้า last page ที่คลิก disabled
    });
    $.post(url, {
        pj_start_date : start_date ,
        pj_page       : page_number ,
        pj_method     : "select"
    }, function (data_ck) {
        $('#result').html(data_ck);
    });
});
});
}
</script>

```

รูป 3.15 ตัวอย่าง Code JavaScript ส่วนของฟังก์ชันการทำงานค้นหา

### 3.3.3 PHP Code

เป็นส่วนของการประมวลผล จะรับค่าตัวแปรมาจาก JavaScript เพื่อประมวลผลจัดโครงสร้างข้อมูลแล้วส่งไปให้กับ Web Service แล้วรับค่าที่ได้จาก Web Service กลับมาเพื่อใช้ประมวลผลข้อมูลเป็นรูปแบบต่างๆแล้วส่งกลับให้ส่วนของ JavaScript

- 1) ส่วนกำหนดค่าที่ใช้เชื่อมต่อกับ Web Service โดยจะประกาศเรียกไว้ในหน้า indexall\_interface.php เพื่อให้สามารถติดต่อกับ Web Service ได้

```
require_once('../includes/config/webservice_addr.php');
```

รูป 3.16 ตัวอย่าง Code เรียกใช้งานไฟล์ webservice\_addr.php

รายละเอียดในไฟล์ `webservice_addr.php` ในส่วนนี้ที่สำคัญคือกำหนดค่าที่อยู่ของ Web Service ในที่นี้คือ `http://localhost/project2/ws/` ซึ่งกำหนดให้เก็บในตัวแปร `wssserver` ในส่วนของ `PAGE_LIMIT` เป็นการกำหนดจำนวนของข้อมูลที่จะแสดงข้อมูลลงในตาราง และส่วนสุดท้ายคือส่วนของ `function post_to_ws` ใช้สำหรับส่งข้อมูลไปยัง url ของ Web Service ด้วยคำสั่ง `file_get_contents`

```

<?php
$wssserver = "http://localhost/project2/ws/";
$webserver = "http://localhost/project2/web/";

$PAGE_LIMIT = "10";

?>

<?php
//subfunction
function post_to_ws($url ,array $data)
{
    $options = array(
        'http' => array(
            'header' => "Content-type: application/x-www-form-urlencoded\r\n",
            'method' => 'POST',
            'content' => http_build_query($data),
        ),
    );
    $context = stream_context_create($options);
    $result = file_get_contents($url, false, $context);

    return $result;
}

?>

```

รูป 3.17 ตัวอย่าง Code ของไฟล์ `webservice_addr.php`

- 2) ส่วนของการกำหนดค่าตัวแปรโดยค่าที่กำหนดให้คือค่าที่ส่งมาจาก JavaScript โดยใช้คำสั่ง `$_REQUEST` ในการรับค่าของตัวแปรมา

```

// init zone -----
$pj_start_date = $_REQUEST['pj_start_date'];
$pj_page       = $_REQUEST['pj_page'];
$pj_method     = $_REQUEST['pj_method'];

// end init zone -----

```

รูป 3.18 ตัวอย่าง Code กำหนดค่าตัวแปร

- 3) ส่วนที่ใช้ติดต่อกับ Web Service โดยการกำหนดค่าของ url ที่ต้องการส่งข้อมูลไป และรอรับข้อมูลกลับมา และกำหนดค่าข้อมูลในรูปแบบ array แล้วส่งข้อมูลไปยัง Web Service ด้วยคำสั่ง `post_to_ws` ซึ่งต้องใช้คำสั่ง `json_decode` เนื่องจากข้อมูลที่ส่งกลับมาจาก Web Service มีการ encode ข้อมูลด้วย

```
//ติดกับ webservice
global $wsserver;
$wsfolder = "/outputdata";
$wsfile = "/select_outputall_list.php";
$url = $wsserver.$wsfolder.$wsfile;
$data = array(
    'start_date' => $start_date,
    'method' => 'select',
);
$data_return = json_decode(post_to_ws($url,$data));
$row=number_format($data_return->{'num_row'});
```

รูป 3.19 ตัวอย่าง Code ส่วนกำหนดข้อมูลและติดต่อกับฐานข้อมูล

- 4) ส่วนนี้คือส่วนของเลือกเงื่อนไขโดยใช้ค่าจากตัวแปร `pj_method` ที่รับมาจาก JavaScript เป็นตัวกำหนดว่าให้ทำเงื่อนไขอะไร โดยแบ่งเป็น 3 รูปแบบคือ `show`, `find_last_page`, `select` รูปแบบแรกคือ `show` เป็นการส่งค่าจำนวนข้อมูลกลับ รูปแบบที่สองคือการหาหน้าสุดท้ายของตาราง และรูปแบบสุดท้ายคือสร้างตาราง แสดงข้อมูลซึ่งจะกล่าวถึงในข้อต่อไป

```
if($pj_method == 'show')
{
    echo $row;
}
else if($pj_method == 'find_last_page')
{
    if(sizeof($data_return->{'results'})==0)
        echo "ไม่มีข้อมูลถึงนะ";
    else
    {
        $total_page = ceil($row/$PAGE_LIMIT) ;
        echo $total_page;
    }
}
```

รูป 3.20 ตัวอย่าง Code เงื่อนไขการทำงาน

- 5) เป็นการสร้างตารางแสดงข้อมูลจากข้อมูลที่รับมาจาก Web Service โดยรูปแบบ จะเป็นการกำหนดหัวตาราง

```

echo'<table class="table table-bordered table-hover">
    <thead>
        <tr>
            <th width="5%">วันที่</th>
            <th width="5%">เวลา</th>
            <th >Soil Temperature (*C)</th>
            <th >Wind Dir</th>
            <th >Wind Speed (m/s)</th>
            <th >Gust Speed (m/s)</th>
            <th >Water Content Crop 2</th>
            <th >Water Content Crop 3</th>
            <th >Temp (*C)</th>
            <th >Relative Humidity</th>
            <th >Dew Point (*C)</th>
            <th >Battery (V)</th>
        </tr>
    </thead>
    <tbody>';

```

รูป 3.21 ตัวอย่าง Code php สร้างตารางแสดงข้อมูล

ในรูปล่างจะเป็นการนำข้อมูลที่ได้จาก Web Service ในรูปแบบ array มาใส่ข้อมูลต่างๆ ลงในตาราง

```

$start_row=$pj_page*$PAGE_LIMIT;
$data_show=array_slice($data_return->{'results'}, $start_row, $PAGE_LIMIT);
foreach($data_show as $data)
{
    $start_row += 1;
    //วันที่
    if($data->{'pj_date'}!="")
    {
        $day = substr($data->{'pj_date'},8,2);
        $month= substr($data->{'pj_date'},5,2);
        $year = substr($data->{'pj_date'},0,4);
        $time = substr($data->{'pj_date'},11,8);
        $date = $day.'/'.$month.'/'.(($year+543));
    }
    else
    {
        $date = "";
    }
    echo '<td class=" text-center">'. $date. '</td>';
    echo '<td class=" text-center">'. $time. '</td>';
    echo '<td class=" text-center">'. $data->{'pj_soil_temp'}. '</td>';
    echo '<td class=" text-center">'. $data->{'pj_wind_dir'}. '</td>';
    echo '<td class=" text-center">'. $data->{'pj_wind_speed'}. '</td>';
    echo '<td class=" text-center">'. $data->{'pj_gust_speed'}. '</td>';
    echo '<td class=" text-center">'. $data->{'pj_water_crop2'}. '</td>';
    echo '<td class=" text-center">'. $data->{'pj_water_crop3'}. '</td>';
    echo '<td class=" text-center">'. $data->{'pj_temp'}. '</td>';
    echo '<td class=" text-center">'. $data->{'pj_rh'}. '</td>';
    echo '<td class=" text-center">'. $data->{'pj_dew'}. '</td>';
    echo '<td class=" text-center">'. $data->{'pj_battery'}. '</td>';
    echo '</tr>';
}
echo'
</tbody>
</table>';

```

รูป 3.22 ตัวอย่าง Code นำข้อมูลจากรฐานข้อมูลมาแสดงลงในตาราง

### 3.3.4 Finish Page

หน้าแสดงสถานะการนำเข้าหรืออัปเดตข้อมูลจะแบ่งออกเป็น สำเร็จสีเขียว และไม่สำเร็จสีแดง



รูป 3.23 หน้าแสดงสถานะการนำเข้าหรืออัปเดตข้อมูลลงฐานข้อมูล

ในส่วนของหน้าแสดงสถานะใช้ Session ในการเก็บค่า mode เพื่อแสดงหน้าสำเร็จหรือไม่สำเร็จ

```
<?php
session_start();
$_SESSION["this_page"] = "finish_page";

// Set Auto redirect time (Seconds)
$redirect_time = 5 ;
// Set Text redirect
$arr_result["txt_autoredirect"] = '(ไปสูหน้าจอลายทางอัตโนมัติใน <span id="counter">'.$redirect_time.'</span> วินาที)';

// Set Text Show and redirect url

//-----ในสถานะนี้
if($_SESSION["finish_page"]["mode"] == "complete") # สำเร็จ
{
    $arr_result["status"] = "green";
    $arr_result["txt_header"] = "นำเข้าข้อมูลสำเร็จแล้ว";
    $arr_result["txt_info"] = "กรุณากดที่ปุ่มด้านล่างนี้เพื่อไปสูหน้าหลัก";
    $arr_result["txt_btn"] = "ไปเก็บข้อมูล";
    $arr_result["url"] = "../inputdata/";
}
else if($_SESSION["finish_page"]["mode"] == "cannot_complete") # ไม่สำเร็จ
{
    $arr_result["status"] = "red";
    $arr_result["txt_header"] = "นำเข้าข้อมูลไม่สำเร็จ";
    $arr_result["txt_info"] = "กรุณากดที่ปุ่มด้านล่างนี้เพื่อกลับไปกรอกข้อมูลอีกครั้ง";
    $arr_result["txt_btn"] = "ไปเก็บข้อมูล";
    $arr_result["url"] = "../inputdata/";
}
}
```

รูป 3.24 ตัวอย่าง Code เงื่อนไขสถานะของหน้า Finish Page

## 3.4 Web Service

คือระบบซอฟต์แวร์ที่ออกแบบมาเพื่อสนับสนุนการแลกเปลี่ยนข้อมูลกัน ระหว่างเครื่องคอมพิวเตอร์ผ่านระบบเครือข่าย โดยที่ภาษาที่ใช้ในการติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์คือเอกซ์เอ็มแอล เว็บเซอร์วิซมีอินเทอร์เน็ตเฟส ที่ใช้อธิบายรูปแบบข้อมูลที่เครื่องคอมพิวเตอร์ประมวลผลได้

### 3.4.1 Config

ในส่วนของการตั้งค่าเชื่อมต่อกับฐานข้อมูลเพื่อให้ง่ายในการเปลี่ยนที่อยู่ของฐานข้อมูล จึงสร้างไฟล์ขึ้นมาและการนำไปใช้ให้ใช้คำสั่ง require ไฟล์ที่เราสร้างขึ้นมา

```
<?php
require("../dbconfig.php");
header('Content-Type: text/html; charset=utf-8');
conndb();
?>
```

รูป 3.25 ตัวอย่าง Code เรียกใช้งานไฟล์ dbconfig.php

ไฟล์ dbconfig.php เป็นการตั้งค่า mysqli โดยกำหนด ip username password และชื่อตาราง เป็นส่วนของการเชื่อมต่อกับฐานข้อมูล

```
<?php
$mysqli;
$dbserver = "127.0.0.1"; // database_server_ip
$dbuser = "root";
$dbpass = "1234";
$dbname = "project2";

function conndb() {
    global $conn;
    global $dbserver;
    global $dbuser;
    global $dbpass;
    global $dbname;
    $conn = new mysqli($dbserver, $dbuser, $dbpass, $dbname);
    $conn->set_charset("utf8");
}

function closedb() {
    global $conn;
    mysql_close($conn);
}

?>
```

รูป 3.26 ตัวอย่าง Code สร้างการเชื่อมต่อกับฐานข้อมูล

### 3.4.2 SQL

เป็นส่วนของคำสั่ง sql เพื่อใช้กับฐานข้อมูลแบ่งออกเป็น 3 ส่วนคือ Select Update และ Insert

#### 1) Select

เป็นคำสั่งในการเลือกข้อมูลจากฐานข้อมูล เมื่อได้ข้อมูลมาแล้วจะต้องเก็บข้อมูลที่ได้อีกมาให้อยู่ในรูปแบบ array ที่เรากำหนดเอง

```

$sql = 'SELECT `num`,
        `date`,
        `soil_temp`,
        `wind_dir`,
        `wind_speed`,
        `gust_speed`,
        `water_crop2`,
        `water_crop3`,
        `temp`,
        `rh`,
        `dew`,
        `battery`,
        `time`
FROM `outputdata`
WHERE `$.stm.` ORDER BY date DESC
';

$result = $conn->query($sql);
while($row = mysqli_fetch_array($result ,MYSQLI_ASSOC) ){
    $check = 1;
    foreach ($row as $k => $v) {
        $row_data['pj_'].$k = $v;
    }

    $data_set[] = $row_data;
}

```

รูป 3.27 ตัวอย่าง Code sql การ Select ข้อมูล

## 2) Insert

เป็นคำสั่งในการเพิ่มข้อมูลลงในฐานข้อมูล

```

$i=0;
foreach($titlenum as $title){
    $sql2 = 'INSERT INTO `inputdata_detail`(`group`,`title`,`score`)
            VALUES (
                " ".$g." " , " ".$title." " , " ".$inputdata[$i]."
            )
            ';

    $result2 = $conn->query($sql2);
    $i++;
}

```

รูป 3.28 ตัวอย่าง Code sql การ Insert ข้อมูล

## 3) Update

เป็นคำสั่งในการอัปเดตข้อมูลใหม่ลงในฐานข้อมูลแทนข้อมูลเดิม

```

$sql = 'UPDATE `performance`
SET
    `soil_temp_score`='.$soil_temp_score.',
    `wind_dir_score`='.$wind_dir_score.',
    `wind_speed_score`='.$wind_speed_score.',
    `gust_speed_score`='.$gust_speed_score.',
    `water_crop2_score`='.$water_crop2_score.',
    `water_crop3_score`='.$water_crop3_score.',
    `temp_score`='.$temp_score.',
    `rh_score`='.$rh_score.',
    `dew_score`='.$dew_score.',
    `battery_score`='.$battery_score.'
WHERE `date` = ".$date.'"
';
$result = $conn->query($sql);

```

รูป 3.29 ตัวอย่าง Code sql การ Update ข้อมูล

### 3.4.3 ส่งข้อมูลที่ได้จากฐานข้อมูลให้ Web Application

สร้างเงื่อนไขแบ่งออก 2 รูปแบบคือ มีข้อมูลส่งกลับมาจากฐานข้อมูลและไม่มีข้อมูลส่งกลับมาจากฐานข้อมูลเพื่อใช้สำหรับหน้า finish page ในกรณี insert และ update เมื่อจัดข้อมูลในรูปแบบ array ที่ต้องการแล้วใช้คำสั่ง json\_encode เพื่อส่งข้อมูลกลับไปให้ส่วนของ Web Application

```

if($check == "1")
{
    $data_return['status'] = "100";
    $data_return['statusDetails'] = "สมบูรณ์";
    $data_return['results'] = $data_set;
    $data_return['num_row'] = count($data_set);
}
else
{
    $data_return['status'] = "101";
    $data_return['statusDetails'] = "ไม่สมบูรณ์";
    $data_return['results'] = $data_set;
}

}

echo json_encode($data_return);

```

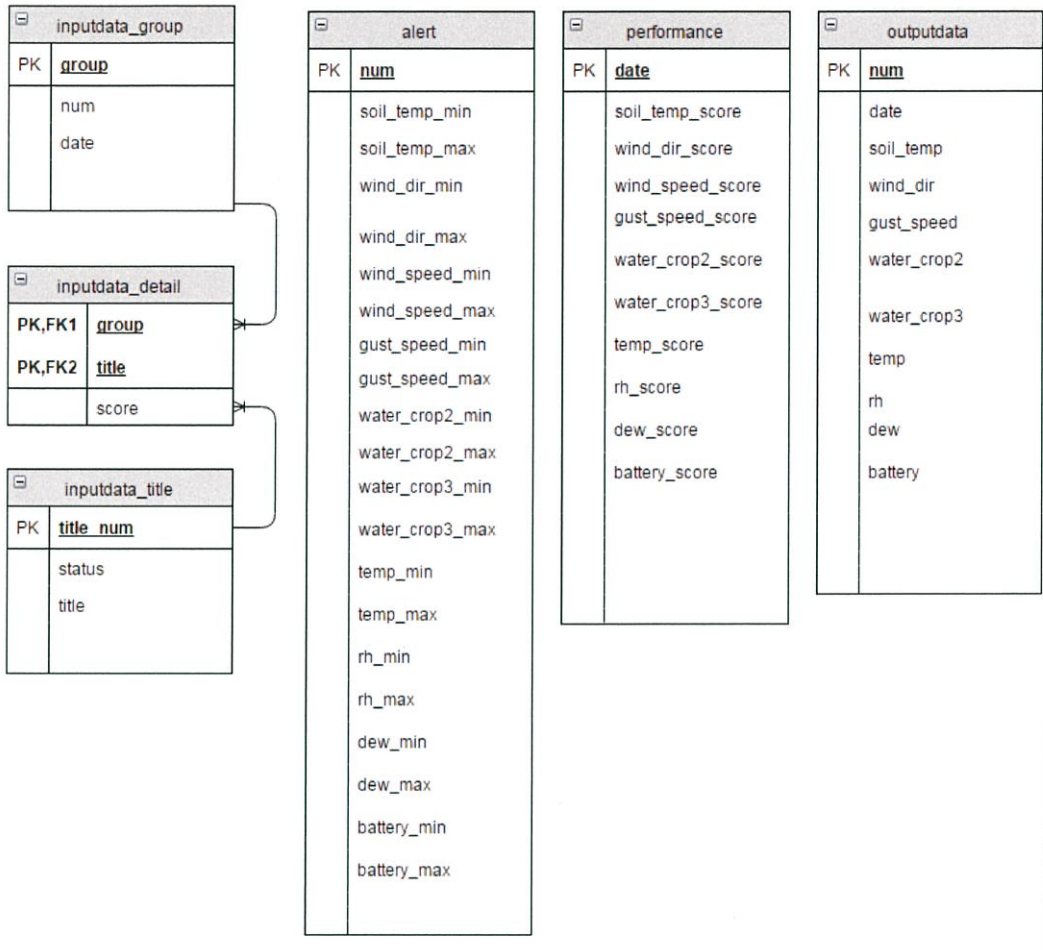
รูป 3.30 ตัวอย่าง Code กำหนดค่าสถานะข้อมูลและการส่งข้อมูลกลับ

## 3.5 Database

คือระบบที่รวบรวมข้อมูลต่าง ๆ ที่เกี่ยวข้องกันเข้าไว้ด้วยกันอย่างมีระบบมีความสัมพันธ์ระหว่างข้อมูลต่าง ๆ ที่ชัดเจน ในระบบฐานข้อมูลจะประกอบด้วยเพิ่มข้อมูลหลายเพิ่มที่มีข้อมูลเกี่ยวข้องสัมพันธ์กันเข้าไว้ด้วยกันอย่างเป็นระบบและเปิดโอกาสให้ผู้ใช้สามารถใช้งานและดูแลรักษาป้องกันข้อมูลเหล่านี้ ได้อย่างมีประสิทธิภาพ

### 3.5.1 ER Diagram

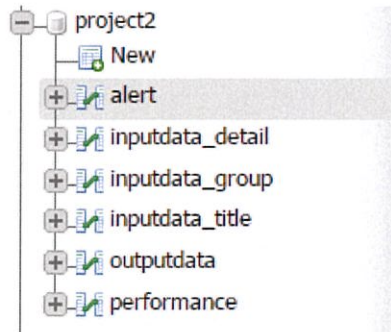
แบบจำลองที่ใช้อธิบายโครงสร้างของฐานข้อมูลซึ่งเขียนออกมาในลักษณะของรูปภาพ



รูป 3.31 ER Diagram

### 3.5.2 phpMyAdmin

ฐานข้อมูลแบ่งออกเป็น 6 ตาราง ดังรูป



รูป 3.32 แสดงตารางฐานข้อมูล

## 1) ตาราง alert

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 num 🔑	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2 soil_temp_min	float			No	None	
<input type="checkbox"/>	3 soil_temp_max	float			No	None	
<input type="checkbox"/>	4 wind_dir_min	int(11)			No	None	
<input type="checkbox"/>	5 wind_dir_max	int(11)			No	None	
<input type="checkbox"/>	6 wind_speed_min	float			No	None	
<input type="checkbox"/>	7 wind_speed_max	float			No	None	
<input type="checkbox"/>	8 gust_speed_min	float			No	None	
<input type="checkbox"/>	9 gust_speed_max	float			No	None	
<input type="checkbox"/>	10 water_crop2_min	float			No	None	
<input type="checkbox"/>	11 water_crop2_max	float			No	None	
<input type="checkbox"/>	12 water_crop3_min	float			No	None	
<input type="checkbox"/>	13 water_crop3_max	float			No	None	
<input type="checkbox"/>	14 temp_min	float			No	None	
<input type="checkbox"/>	15 temp_max	float			No	None	
<input type="checkbox"/>	16 rh_min	float			No	None	
<input type="checkbox"/>	17 rh_max	float			No	None	
<input type="checkbox"/>	18 dew_min	float			No	None	
<input type="checkbox"/>	19 dew_max	float			No	None	
<input type="checkbox"/>	20 battery_min	float			No	None	
<input type="checkbox"/>	21 battery_max	float			No	None	

รูป 3.33 ฐานข้อมูลตาราง alert

## 2) ตาราง inputdata\_detail

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 num 🔑	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2 group	int(11)			No	None	
<input type="checkbox"/>	3 title	tinyint(4)			No	None	
<input type="checkbox"/>	4 score	tinyint(4)			No	None	

รูป 3.34 ฐานข้อมูลตาราง inputdata\_detail

## 3) ตาราง inputdata\_group

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 num 🔑	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2 date	date			No	None	
<input type="checkbox"/>	3 group	int(11)			No	None	

รูป 3.35 ฐานข้อมูลตาราง inputdata\_group

## 4) ตาราง inputdata\_title

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 title	text	utf8_general_ci		No	None	
<input type="checkbox"/>	2 status	int(11)			No	None	
<input type="checkbox"/>	3 title_num 🔑	int(11)			No	None	AUTO_INCREMENT

รูป 3.36 ฐานข้อมูลตาราง inputdata\_title

## 5) ตาราง outputdata

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 num 🔑	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2 date	datetime			No	None	
<input type="checkbox"/>	3 soil_temp	float			No	None	
<input type="checkbox"/>	4 wind_dir	int(11)			No	None	
<input type="checkbox"/>	5 wind_speed	float			No	None	
<input type="checkbox"/>	6 gust_speed	float			No	None	
<input type="checkbox"/>	7 water_crop2	float			No	None	
<input type="checkbox"/>	8 water_crop3	float			No	None	
<input type="checkbox"/>	9 temp	float			No	None	
<input type="checkbox"/>	10 rh	float			No	None	
<input type="checkbox"/>	11 dew	float			No	None	
<input type="checkbox"/>	12 battery	float			No	None	
<input type="checkbox"/>	13 time	time			No	None	

รูป 3.37 ฐานข้อมูลตาราง outputdata

## 6) ตาราง performance

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 num 🔑	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2 date	date			No	None	
<input type="checkbox"/>	3 soil_temp_score	int(11)			No	None	
<input type="checkbox"/>	4 wind_dir_score	int(11)			No	None	
<input type="checkbox"/>	5 wind_speed_score	int(11)			No	None	
<input type="checkbox"/>	6 gust_speed_score	int(11)			No	None	
<input type="checkbox"/>	7 water_crop2_score	int(11)			No	None	
<input type="checkbox"/>	8 water_crop3_score	int(11)			No	None	
<input type="checkbox"/>	9 temp_score	int(11)			No	None	
<input type="checkbox"/>	10 rh_score	int(11)			No	None	
<input type="checkbox"/>	11 dew_score	int(11)			No	None	
<input type="checkbox"/>	12 battery_score	int(11)			No	None	

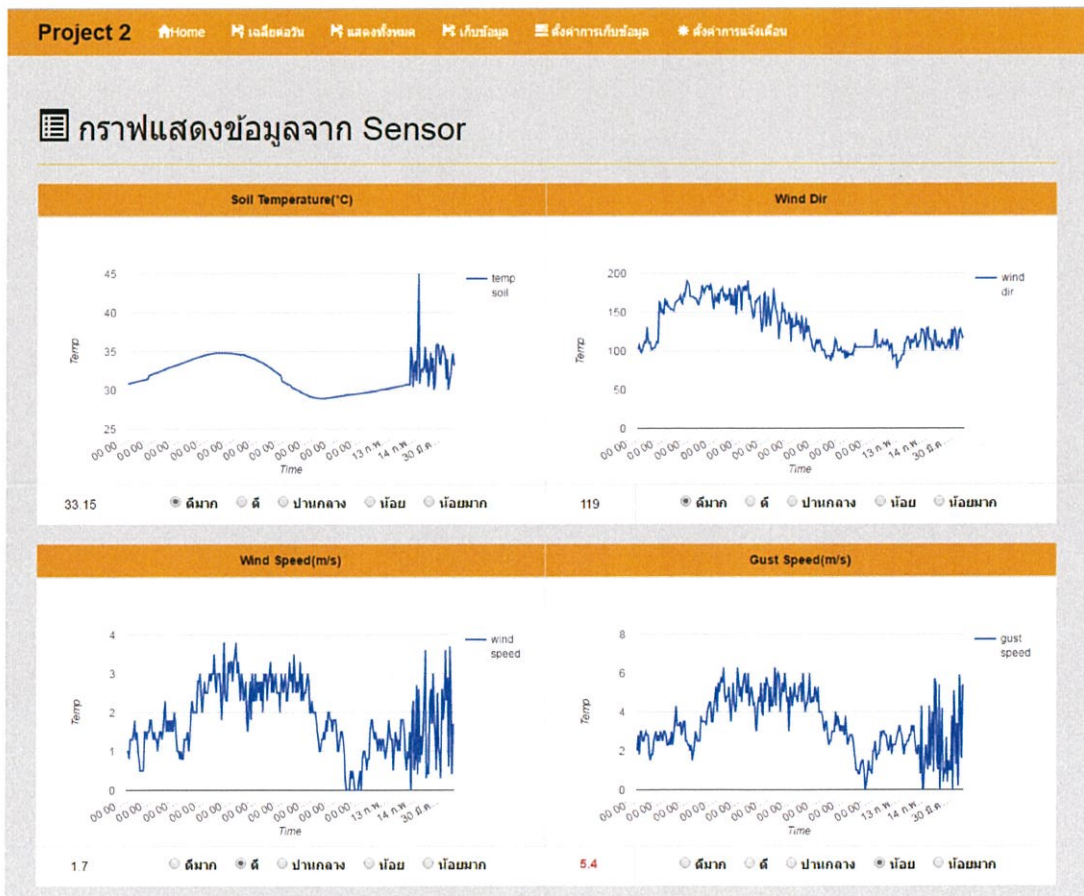
รูป 3.38 ฐานข้อมูลตาราง performance

# บทที่ 4

## ผลการวิจัย

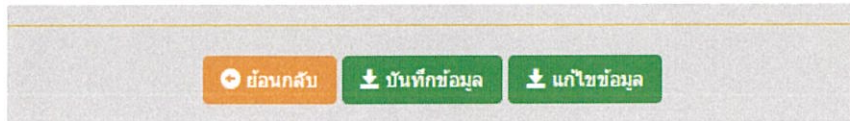
### 4.1 หน้าหลักของระบบ

หน้าหลักของระบบแสดงกราฟข้อมูลของ Sensor เพื่อใช้ในการดูการเปลี่ยนแปลงของข้อมูล โดยแบ่งออกเป็น 10 กราฟตามข้อมูลของ Sensor และแสดงค่าข้อมูลจาก Sensor ค่าล่าสุดที่ได้รับ พร้อมแจ้งเตือนด้วยรูปแบบตัวอักษรสีแดง แสดงถึงมีค่าข้อมูลเกินที่กำหนดไว้ ส่วนสุดท้ายแสดงค่าการประเมิน 5 ระดับ โดยหนึ่งวันจะเก็บค่าประเมิน 1 ค่าพร้อมตั้งสามารถแก้ไขค่าประเมินได้ ข้อมูลของ Sensor มีดังนี้ Soil Temperature(\*C) ,Wind Dir ,Wind Speed(m/s) ,Gust Speed(m/s) ,Water Content Crop 2 ,Water Content Crop 3 ,Temp(\*C) ,Relative Humidity ,Dew Point(\*C) ,Battery(V)



รูป 4.1 แสดงหน้าหลักของระบบ

ปุ่มการทำงานต่างๆแบ่งออกเป็น 3 รูปแบบคือ ปุ่มย้อนกลับไปหน้าที่แล้ว ปุ่มบันทึกข้อมูลใหม่สำหรับบันทึกข้อมูลการประเมินใหม่ลงในฐานข้อมูลและปุ่มแก้ไขข้อมูลสำหรับแก้ไขข้อมูลการประเมินของวันปัจจุบันลงในฐานข้อมูลใหม่แทนที่ข้อมูลเดิม



รูป 4.2 แสดงปุ่มการทำงาน

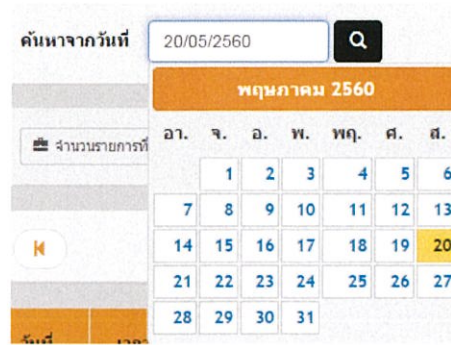
## 4.2 หน้าแสดงข้อมูลเฉลี่ยจาก Sensor

หน้าแสดงข้อมูลเฉลี่ยจะแสดงข้อมูลค่าเฉลี่ยในหนึ่งวันของข้อมูลที่ได้รับมาจาก Sensor สามารถเลือกวันที่ที่จะให้แสดงข้อมูลของวันนั้นๆได้

วันที่	Soil Temperature(°C)	Wind Dir	Wind Speed(m/s)	Gust Speed(m/s)	Water Content Crop 2	Water Content Crop 3	Temp(°C)	Relative Humidity	Dew Point(°C)	Battery(V)
31/03/2560	32.49	119.33	1.97	3.43	1.31	1.02	28	92.39	25.28	4.31

รูป 4.3 แสดงหน้าอ่านข้อมูลจาก Sensor

ปุ่มค้นหาจากวันที่เมื่อกดที่กรอบวันที่จะมีเมนูวันเดือนปีให้สามารถเลือกวันจากปฏิทินได้ โดยค่าวันที่พื้นฐานในการกดครั้งแรกจะเป็นวันที่ปัจจุบัน



รูป 4.4 ปฏิทินในหน้าอ่านข้อมูลเฉลี่ยจาก Sensor

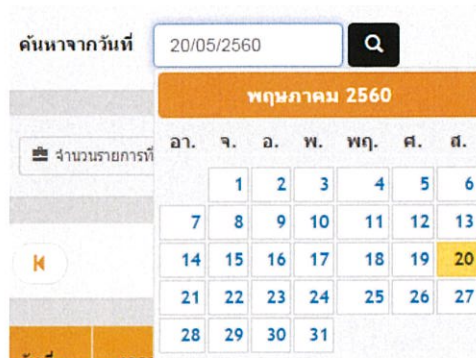
### 4.3 หน้าแสดงข้อมูลจาก Sensor

แสดงข้อมูลจาก Sensor ในหนึ่งวัน โดยจะแสดงข้อมูลทั้งหมดที่ได้รับจาก Sensor ซึ่งสามารถเลือกวันที่แสดงข้อมูลออกมาได้ ในหน้านี้จะมีแถบแสดงรายการทั้งหมดและแถบเปลี่ยนหน้าอีก 2 แถบทั้งแถบบนและล่าง ซึ่งสามารถเปลี่ยนเป็นหน้าแรก หน้าก่อนหน้า หน้าถัดไปและหน้าสุดท้ายได้ โดยจะมีบอกเลขหน้าปัจจุบันและหน้าทั้งหมดอยู่ด้วย แถบบนสามารถใส่ตัวเลขหน้าที่ต้องการไปค้นหาและกดปุ่มค้นหาได้ ปริมาณข้อมูลที่แสดงในตารางในหนึ่งหน้าสามารถได้ในโปรแกรมพื้นฐานจะใช้ที่ 10 ข้อมูลต่อหนึ่งหน้า

วันที่	เวลา	Soil Temperature(°C)	Wind Dir	Wind Speed(m/s)	Gust Speed(m/s)	Water Content Crop 2	Water Content Crop 3	Temp(°C)	Relative Humidity	Dew Point(°C)	Battery(V)
31/03/2560	07:05:00	35.51	117	1.8	4.5	0.75	1.03	30.84	80.02	26.14	4.31
31/03/2560	07:00:00	32.76	101	3.1	1.2	1.91	1.96	22.2	96.24	23.16	4.31
31/03/2560	06:55:00	30.59	123	2.1	2.3	0.97	0.88	39.63	98.64	25.05	4.31
31/03/2560	06:50:00	30.26	107	2.6	0	1.75	0.66	29.37	83.34	22.51	4.31
31/03/2560	06:45:00	35.75	117	1.9	1.8	1.86	0.81	30.5	74.51	27.97	4.31
31/03/2560	06:40:00	31.69	111	0.7	5.2	1.06	0.13	23	92.7	24.85	4.31
31/03/2560	06:35:00	31.49	107	1.1	5.1	1.75	1.18	24.74	78.39	22.42	4.31
31/03/2560	06:30:00	32	129	1	2.5	1.11	1.9	33.36	97.77	24.83	4.31
31/03/2560	06:25:00	33.15	119	1.7	5.4	1.44	1.19	22.76	96.1	25.44	4.31
31/03/2560	06:20:00	34.74	117	0.4	1.6	1.87	1.53	39.02	82	23.28	4.31

รูป 4.5 แสดงหน้าอ่านข้อมูลจาก Sensor

ปุ่มค้นหาจากวันที่เมื่อกดที่รอบวันที่จะมีเมนูวันเดือนปีให้สามารถเลือกวันจากปฏิทินได้ โดยค่าวันที่พื้นฐานในการกดครั้งแรกจะเป็นวันที่ปัจจุบัน



รูป 4.6 ปฏิทินในหน้าอ่านข้อมูลจาก Sensor

#### 4.4 หน้าแสดงข้อมูลที่เก็บจากผู้ใช้งาน

แสดงข้อมูลที่เก็บได้จากผู้ใช้งานระบบโดยจะแสดงข้อมูลระหว่างวันที่เลือกจากเมนูด้านบน สามารถเพิ่มข้อมูลที่ต้องการเก็บและแก้ไขได้

วันที่	ผลผลิต	คุณภาพผลผลิต	โรค	สภาพดิน	อุณหภูมิดิน	ลม	ปริมาณน้ำ Crop2	ปริมาณน้ำ Crop3	อุณหภูมิ	ความชื้น	
31/03/2560	5	5	5	5	5	5	5	5	4	4	แก้ไข
30/03/2560	5	5	4	5	5	3	4	5	5	5	แก้ไข
29/03/2560	4	3	4	2	5	5	2	4	5	5	แก้ไข
28/03/2560	5	4	4	3	5	5	5	3	5	5	แก้ไข
27/03/2560	4	5	5	3	5	5	5	5	3	5	แก้ไข
26/03/2560	4	5	3	5	5	4	5	3	5	5	แก้ไข
25/03/2560	4	4	5	3	5	3	5	5	5	5	แก้ไข
24/03/2560	4	5	5	2	5	1	5	5	1	5	แก้ไข
23/03/2560	4	5	5	5	2	5	5	4	2	2	แก้ไข
22/03/2560	5	4	5	5	3	5	4	1	5	1	แก้ไข

รูป 4.7 แสดงหน้าแสดงข้อมูลการเก็บข้อมูล

ปุ่มค้นหาจากวันที่เมื่อกดที่กรอบวันที่จะมีเมนูวันเดือนปีให้สามารถเลือกวันจากปฏิทินได้ โดยค่าวันที่พื้นฐานในการกดครั้งแรกจะเป็นเดือนปัจจุบัน เป็นช่วงเวลาตั้งแต่วันที่แรกของเดือนถึงวันสุดท้ายของเดือน สามารถปรับแก้ไขให้ค้นหาระหว่างช่วงวันที่ได้

รูป 4.8 ปฏิทินในหน้าเก็บข้อมูล

#### 4.5 หน้าแก้ไขข้อมูลจากผู้ใช้งาน

หน้าแก้ไขข้อมูลจากผู้ใช้งาน แก้ไขข้อมูลการประเมินโดยให้ผู้ใช้งานระบบเป็นผู้ใส่รายละเอียด

รูป 4.9 แสดงหน้าแก้ไขข้อมูล



รูป 4.10 ปุ่มการทำงานของหน้าแก้ไขข้อมูล

#### 4.6 เก็บข้อมูลจากผู้ใช้งาน

หน้าเก็บข้อมูลจากผู้ใช้งาน เก็บข้อมูลการประเมิน โดยให้ผู้ใช้งานระบบเป็นผู้ใส่รายละเอียด

รูป 4.11 แสดงหน้าเก็บข้อมูล



รูป 4.12 ปุ่มการทำงานของหน้าเก็บข้อมูล

#### 4.7 ตั้งค่าการเก็บข้อมูลจากผู้ใช้งาน

หน้าตั้งค่าการเก็บข้อมูลเพื่อตั้งค่าข้อมูลที่จะเก็บหรือลบข้อมูลที่ไม่ต้องการเก็บเพิ่มเติม

**Project 2** Home เติมน้ำต้นไม้ แสดงสถานะ เก็บข้อมูล ตั้งค่าการเก็บข้อมูล ตั้งค่าการแจ้งเตือน

## ตั้งค่าการเก็บข้อมูล

- ผลผลิต
  - เก็บ  ไม่เก็บ
- คุณภาพผลผลิต
  - เก็บ  ไม่เก็บ
- โรค
  - เก็บ  ไม่เก็บ
- สภาพแวดล้อม
  - เก็บ  ไม่เก็บ
- ลุคนภูมิดิน
  - เก็บ  ไม่เก็บ
- ลม
  - เก็บ  ไม่เก็บ
- ปริมาณน้ำ Crop2
  - เก็บ  ไม่เก็บ
- ปริมาณน้ำ Crop3
  - เก็บ  ไม่เก็บ
- ลุคนภูมิ
  - เก็บ  ไม่เก็บ
- ความชื้น

รูป 4.13 หน้าตั้งค่าการเก็บข้อมูล



รูป 4.14 ปุ่มการทำงานหน้าตั้งค่าการเก็บข้อมูล

## 4.8 ตั้งค่าการแจ้งเตือน

หน้าสำหรับตั้งค่าการแจ้งเตือนโดยตั้งค่าให้ Sensor มีค่าระหว่างตัวเลขสองจำนวน ถ้าค่าจาก Sensor มีค่าไม่อยู่ระหว่างตัวเลขที่ตั้งค่าไว้ จะแสดงเป็นตัวอักษรสีแดงที่หน้าหลัก



รูป 4.15 แสดงหน้าตั้งค่าการแจ้งเตือน

#### 4.9 หน้า Finish Page

หน้าสำหรับแสดงค่าสถานะการนำเข้าข้อมูลหรืออัปเดตข้อมูลจากผู้ใช้โดยสีเขียวคือสำเร็จ และสีแดงคือผิดพลาด



รูป 4.16 แสดงหน้า Finish Page

## บทที่ 5

# บทสรุปและข้อเสนอแนะ

### 5.1 สรุป

- 1) Web Application นี้เป็นระบบสำหรับแสดงผลข้อมูลจาก Sensor และเก็บข้อมูลการประเมินจากผู้ใช้งานระบบ
- 2) งานนี้มีการแบ่งออกเป็น 3 ส่วนหลักได้แก่ส่วนของ Web Application ซึ่งใช้ภาษา Html CSS JavaScript PHP เป็นส่วนหลักของการสร้าง ส่วนที่สองเป็นส่วนของ Web Service ซึ่งทำหน้าที่เป็นตัวกลางในการติดต่อระหว่าง Web Application กับ Database โดยใช้ JSON เป็นรูปแบบในการส่งผ่านข้อมูล และส่วนสุดท้ายคือส่วนของ Database ที่เป็นฐานข้อมูลในการเก็บข้อมูลทั้งหมด ทั้งจาก Sensor และผลการประเมินจากผู้ใช้งานระบบ
- 3) หน้าหลักของระบบเป็นการแสดงผลในรูปแบบของกราฟเส้นเพื่อให้สามารถดูแนวโน้มของข้อมูลจาก Sensor ทั้ง 10 ข้อมูลในหนึ่งวันได้ มีการแจ้งเตือนเมื่อมีข้อมูลที่ผิดปกติและสามารถเพิ่มผลประเมินระดับค่าของข้อมูลในวันนั้นๆ ได้อีกด้วย
- 4) ระบบแสดงข้อมูลจาก Sensor มีทั้งแบบเฉลี่ยในหนึ่งวันและข้อมูลแบบละเอียดทั้งหมดภายในระยะเวลาที่กำหนดได้แสดงในรูปแบบของตารางแสดงข้อมูล
- 5) ระบบแสดงข้อมูลที่ได้จากการเก็บผลการประเมินแสดงในรูปแบบตารางสามารถเพิ่มและแก้ไขข้อมูลการประเมินได้
- 6) ระบบตั้งค่าการแจ้งเตือนใช้สำหรับหน้าหลักเป็นการกำหนดค่าความผิดปกติต่างๆของข้อมูลจาก Sensor เพื่อใช้อ้างอิงในการแจ้งเตือนผู้ใช้งานระบบ

### 5.2 ปัญหา

- 1) การเก็บข้อมูลการประเมินข้อมูล Sensor จากผู้ใช้งาน สามารถเก็บได้เฉพาะในวันนั้นๆ ระบบยังไม่สามารถรองรับการเก็บข้อมูลย้อนหลังหรือแก้ไขข้อมูลย้อนหลังได้
- 2) ในหน้าแสดงข้อมูลจาก Sensor ถ้าในวันใหม่ไม่มีการส่งข้อมูลจาก Sensor มาที่ฐานข้อมูล บน Web Application จะมีการแจ้งเตือน Code Notice ว่าไม่สามารถหาค่าของข้อมูล Sensor ได้
- 3) หน้าหลักของระบบเมื่อขึ้นวันใหม่บางครั้งจะมี Code Notice แจ้งเตือนว่าไม่มีข้อมูลการประเมินจากผู้ใช้งาน เนื่องจากโปรแกรมที่สร้างขึ้นมีการดึงข้อมูลการประเมินจากในวันนั้นๆ มาแสดงในหน้าหลักให้ผู้ใช้งานด้วยเพื่อให้ผู้ใช้งานเห็นว่ามีการประเมินไว้แล้วเมื่อไม่มีข้อมูลชุดนั้นเลยมีการแจ้งเตือนในหน้าหลัก

- 4) Web Application นี้ไม่มีระบบ Login จำเป็นการใช้งานร่วมกันไม่มีการแบ่งผู้ใช้งาน ข้อมูลที่ได้จากการประเมินอาจจะน้อยเกินไป

### 5.3 แนวทางการแก้ไข

- 1) เพิ่มระบบแก้ไขข้อมูลการประเมินข้อมูล Sensor จากผู้ใช้งานย้อนหลังให้สามารถเพิ่ม และแก้ไขข้อมูลการประเมินจากผู้ใช้งานได้
- 2) ตั้งค่าระบบให้ปิดการแจ้งเตือน Code Notice หรือใส่เงื่อนไขเพิ่มตรวจสอบว่ามีข้อมูล Sensor จากวันนั้นๆหรือไม่ถ้าไม่มีให้แสดงผลต่างจากปกติ หรือให้แจ้งด้วยข้อความจากระบบ
- 3) สามารถแก้ปัญหาได้ด้วยวิธีเดียวกับข้อ 2)
- 4) เพิ่มระบบ Login ให้สามารถแยกผู้ใช้งานระบบเพิ่มขึ้น เพื่อให้สามารถเก็บข้อมูลการประเมินได้เพิ่มขึ้น

### 5.4 แนวทางในการพัฒนาต่อ

- 1) พัฒนาระบบแก้ไขและเพิ่มข้อมูลการประเมินย้อนหลังให้แก่ผู้ใช้งานระบบ
- 2) เพิ่มระบบ Login ให้กับระบบเพื่อเพิ่มปริมาณข้อมูลการประเมินที่ต้องนำมาวิเคราะห์
- 3) นำข้อมูลจาก Sensor และข้อมูลการประเมินที่ได้จากผู้ใช้งานมาวิเคราะห์ด้วยวิธีการต่างๆ ให้สามารถทำนายชุดคำตอบทั้งเรื่องผลผลิต สภาพอากาศ โรค และอื่นๆในอนาคตได้
- 4) พัฒนาระบบ Android Application เพื่อให้ง่ายแก่การใช้งานนอกสถานที่หรือสถานที่ เกษตรจริง
- 5) พัฒนาต่อยอดจากระบบการแจ้งเตือนของ Sensor ให้สามารถควบคุมการทำงานของ อุปกรณ์ Sensor นั้นๆได้เช่น Sensor ตรวจวัดความชื้นถ้าความชื้นน้อยกว่าที่ตั้งค่าไว้ให้ เปิดการทำงานของระบบพ่นน้ำ เป็นต้น

## บรรณานุกรม

เสาวนีย์ ลีละวัฒนพันธ์. ภาษา HTML. [Online]. Available : <http://61.19.202.164/resource/courseware/html/k01-01.html>.

HTML คืออะไร?. [Online]. Available : [http://www.enjoyday.net/webtutorial/html/html\\_chapter01.html](http://www.enjoyday.net/webtutorial/html/html_chapter01.html).

JavaScript คืออะไร. 9 เมษายน 2013. [Online]. Available : <http://www.mindphp.com/คู่มือ/73-คืออะไร/2187-java-javascript-คืออะไร.html>.

ความรู้พื้นฐานเกี่ยวกับ JavaScript. 16 พฤษภาคม 2010. [Online]. Available : <http://www.hellomyweb.com/index.php/main/content/131>.

JavaScript คืออะไร. [Online]. Available : [www2.cvc.ac.th/trsai/it51/39012009/JavaScript.doc](http://www2.cvc.ac.th/trsai/it51/39012009/JavaScript.doc).

โครงสร้างของภาษา PHP. [Online]. Available : <http://www.mwit.ac.th/~jeab/40201/ch3.php>.

PHP คืออะไร. 13 มิถุนายน 2014. [Online]. Available : <http://www.mindphp.com/คู่มือ/73-คืออะไร/2127-php-คืออะไร.html>.

Suranart Niamcome. Bootstrap คืออะไร + สอนวิธีใช้แบบเข้าใจง่าย. 14 กรกฎาคม 2013. [Online]. Available : <http://www.siamhtml.com/bootstrap-คืออะไร-สอนวิธีใช้/>.

มานพ กองอ่อน. Bootstrap คืออะไร?. [Online]. Available : <https://www.programmerthailand.com/tutorial/post/view/96/bootstrap-คืออะไร>.

กลุ่มพัฒนาการบริหารข้อมูล ศูนย์เทคโนโลยีสารสนเทศและการสื่อสาร สำนักงานปลัดกระทรวงสาธารณสุข. รู้จักกับ MySQL. 18 ธันวาคม 2012. [Online]. Available : <http://kmops.moph.go.th/index.php/km-test/ict/124-mysql>.

MySQL คืออะไร มาเรียนรู้กัน. 21 มีนาคม 2013. [Online]. Available : <http://www.amplysoft.com/knowledge/what-is-mysql.html>.

มายเอสคิวเอล. 19 มกราคม 2016. [Online]. Available : <https://th.wikipedia.org/wiki/มายเอสคิวเอล>

MySQL มีความสำคัญอย่างไรกับเซิร์ฟเวอร์. [Online]. Available : <http://www.th.easyhostdomain.com/dedicated-servers/mysql.html>.

RiderOOU. เว็บแอปพลิเคชัน (Web Application) คืออะไร ?. [Online]. Available : <http://aicomputer.co.th/sArticle/002-what-is-Web-Application.aspx>.

mospichit. Front-End และ Back-End คืออะไร. 4 ตุลาคม 2016. [Online]. Available : <http://blog.sogoodweb.com/Article/Detail/54053>.

Pijitra.ict5. ทำความรู้จัก FRONT - END และ BACKEND. 28 ตุลาคม 2016. [Online]. Available : [https://www.wynnsoft-solution.com/ทำความรู้จัก\\_Front\\_-end\\_และ\\_Backend](https://www.wynnsoft-solution.com/ทำความรู้จัก_Front_-end_และ_Backend).

CSS คืออะไร?. [Online]. Available : [http://www.enjoyday.net/webtutorial/css/css\\_chapter01.html](http://www.enjoyday.net/webtutorial/css/css_chapter01.html)

เขียน JavaScript สะดวกและง่ายขึ้นด้วย jQuery หรือ JavaScript Library. [Online]. Available : [https://www.mdsoft.co.th/ความรู้/101-what-is\\_jquery.html](https://www.mdsoft.co.th/ความรู้/101-what-is_jquery.html).

สอน jQuery คืออะไร ?. [Online]. Available : <http://www.webub.com/สอน%20jQuery%20คืออะไร%20%20%20jQuery-693-51/>.

gigkokman. ฟังก์ชัน json\_encode ของ PHP. 22 กุมภาพันธ์ 2010. [Online]. Available : [http://www.imooh.com/ฟังก์ชัน-json\\_encode-ของ-php](http://www.imooh.com/ฟังก์ชัน-json_encode-ของ-php).

Sheroku. การใช้งาน JSON เพื่อการสื่อสารกับ Javascript ในรูปแบบของ Ajax. 7 กรกฎาคม 2009. [Online]. Available : [http://articles.modoeye.com/Design\\_and\\_Development/PHP/การใช้งาน\\_JSON\\_เพื่อการสื่อสารกับ\\_Javascript\\_ในรูปแบบของ\\_Ajax.html](http://articles.modoeye.com/Design_and_Development/PHP/การใช้งาน_JSON_เพื่อการสื่อสารกับ_Javascript_ในรูปแบบของ_Ajax.html).

PHP file\_get\_contents() Function. [Online]. Available : [http://www.w3im.com/th/php/func\\_file\\_system\\_file\\_get\\_contents.html](http://www.w3im.com/th/php/func_file_system_file_get_contents.html).

XML คืออะไร. 14 มีนาคม 2017. [Online]. Available : <http://www.mindphp.com/คู่มือ/73-คืออะไร/2129-xml-คืออะไร.html>.

- amdmanzero. **เซสชัน (SESSION) คืออะไรแล้วใช้งานอย่างไร**. 31 พฤษภาคม 2016. [Online]. Available : [http://www.phpdevthailand.com/2016/05/31/php\\_session/](http://www.phpdevthailand.com/2016/05/31/php_session/).
- Google Developers. **Populating Data Using Server-Side Code**. 28 มกราคม 2016. [Online]. Available : [https://developers.google.com/chart/interactive/docs/php\\_example](https://developers.google.com/chart/interactive/docs/php_example).
- Esmerelda. **เว็บเซอร์วิส (Web Service)**. 30 สิงหาคม 2010. [Online]. Available : <http://oknation.nationtv.tv/blog/Siraprapa/2010/08/30/entry-2>.
- Web Services คืออะไร**. 19 กุมภาพันธ์ 2011. [Online]. Available : <http://www.rightsoftcorp.com/?name=news&file=readnews&id=31>.
- Web Services คืออะไร**. 22 พฤษภาคม 2012. [Online]. Available : <http://www.mindphp.com/คู่มือ/73-คืออะไร/2194-web-services-คืออะไร.html>.
- จุฬามาศ อนุสติภูษกุล. **Database คืออะไร ?**. 18 สิงหาคม 2006. [Online]. Available : <https://www.gotknow.org/posts/45256>.
- Database (ฐานข้อมูล) คืออะไร ?**. [Online]. Available : <http://pioneer.chula.ac.th/~vduangna/2200199/page3.html>.
- บทความ E-R Diagram คืออะไร**. [Online]. <http://www.9experttraining.com/articles/บทความ-e-r-diagram-คืออะไร>.
- ชาคริต กุลไกรศรี. **การออกแบบข้อมูลด้วย E-R Diagram (Entity-Relationship Diagrams) (ตอนที่ 1)**. 17 กันยายน 2013. [Online]. Available : <https://msit5.wordpress.com/2013/09/17/การออกแบบข้อมูลด้วย-e-r-diagram-entity-relation/>.

## ภาคผนวก

### ข้อมูลจากเซ็นเซอร์

ข้อมูลจากเซ็นเซอร์คือข้อมูลที่ได้มาจากการทำโครงการเก็บข้อมูลด้วยเซ็นเซอร์แล้วนำมาพัฒนาต่อยอดเป็นโครงการครั้งนี้ ข้อมูลเซ็นเซอร์มีการเก็บข้อมูลต่างๆของสวนเกษตรเช่น เวลาที่เก็บข้อมูล อุณหภูมิในดิน ทิศทางของลม ความเร็วลม ความเร็วลมกัน โชนก ปริมาณน้ำในแปลงผัก อุณหภูมิ ค่าความเป็นกรดเบส ปริมาณแบคทีเรีย เป็นต้น