



รายงานสหกิจศึกษาฉบับสมบูรณ์
ระบบเก็บข้อมูลสินค้าคงคลัง
Web-based Inventory System

ณัฐญา บุราณรักษ์
NATTHAYA BURARNRAK

ภาควิชา วิศวกรรมคอมพิวเตอร์ สาขาวิชา วิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560



รายงานสหกิจศึกษาฉบับสมบูรณ์
ระบบเก็บข้อมูลสินค้าคงคลัง
Web-based Inventory System

ณัฐญา บุราณรักษ์
NATTHAYA BURARNRAK

ภาควิชา วิศวกรรมคอมพิวเตอร์ สาขาวิชา วิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

ชื่อโครงการสหกิจศึกษา	ระบบเก็บข้อมูลสินค้าคงคลัง
ชื่อ-สกุลนักศึกษา	นางสาวณัฐญา บุราณรักษ์
คณะ	วิศวกรรมศาสตรบัณฑิต สาขาวิชา วิศวกรรมสารสนเทศ
ชื่อ-สกุลอาจารย์นิเทศน์	ผศ.มยุรี เลิศเวชกุล
ชื่อ-สกุล ผู้นิเทศน์งาน	คุณพิบูลย์ พรประสิทธิ์
ชื่อสถานประกอบการ	บริษัท ไอบีเอ็ม โซลูชั่นส์ ดิลิเวอรี จำกัด

บทคัดย่อ

ในการปฏิบัติงานในทีมเน็ตเวิร์ค บริษัท ไอบีเอ็ม โซลูชั่นส์ ดิลิเวอรี จำกัด จะมีการนำเอาอุปกรณ์เครือข่ายออกไปเพื่อใช้ในการติดตั้งให้แก่ลูกค้า หรือนำอุปกรณ์ไปเพื่อประกอบการทำงานด้วยเสมอ ซึ่งระเบียบในการทำงานได้กำหนดไว้ให้พนักงานจะต้องลงบันทึกการนำอุปกรณ์ออกและการนำเข้าอุปกรณ์ทุกครั้ง แต่อย่างไรก็ตามมักเกิดปัญหาที่เกิดจากการที่มีผู้นำอุปกรณ์ออกไปโดยไม่ลงบันทึกข้อมูล หรือเกิดจากความล่าช้าในการรวมไฟล์ข้อมูลซึ่งจัดเก็บในรูปแบบของไฟล์เอ็กเซล (Microsoft Excel) ทำให้ข้อมูลที่มีอยู่ในระบบไม่เป็นข้อมูลที่ล่าสุด เมื่อเช็คสินค้าคงคลังจะพบว่าจำนวนอุปกรณ์ที่มีอยู่ในระบบไม่ตรงกับจำนวนสินค้าที่มีอยู่จริง จึงมีแนวคิดในการจัดทำระบบข้อมูลสินค้าคงคลังขึ้นมาเพื่อใช้ประโยชน์ในการติดตามอุปกรณ์ สามารถรู้สถานที่เก็บอุปกรณ์ และจำนวนอุปกรณ์ได้แม่นยำมากยิ่งขึ้น ระบบข้อมูลสินค้าคงคลังที่พัฒนาขึ้นมีส่วนช่วยให้การทำงานในทีมเน็ตเวิร์คเป็นระบบและมีประสิทธิภาพมากยิ่งขึ้น

Co-operative Title	Web-based Inventory System		
Student Intern Name	Natthaya Burarnrak		
Faculty	Engineering	Department	Information Engineering
Advisor Name	Asst.Prof. Mayuree Lertwatechakul		
Mentor Name	Piboon Pornprasit		
Company	IBM Solution Delivery Co., Ltd.		

ABSTRACT

Generally, task of the network operation team, IBM Solutions Delivery Co., Ltd. is to install networking devices at customer's site. The operation rules defined that storing and taking-out the network device activities must be recorded correctively, and up-to-date. But the problem always occurs when network operation team member does not record some device taking activity or because of delaying of file merging procedure since the inventory data could be kept in many separated Excel files. This makes the network device data in the inventory system incorrect. Web-based inventory system is developed as to solve this problem. The web-based inventory system could be used to record the inventory activities from anywhere, anytime, user able to query the devices data to show a device's location, the number of available devices more easily. By utilizing the web-based inventory system, the correct information of the inventory system makes the network operation team working is pretty more systematically and more efficiently.

กิตติกรรมประกาศ

ข้าพเจ้าได้รับผิดชอบและปฏิบัติหน้าที่ในบริษัท ไอพีเอ็ม โซลูชั่นส์ ดิลิเวอรี จำกัด ระหว่างวันที่ 7 สิงหาคม ถึงวันที่ 30 พฤศจิกายน พ.ศ.2560 ในโครงการวิชาสหกิจศึกษาที่ทางคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ได้จัดทำสินค้าคงคลังเว็บไซต์ขึ้นมา ซึ่งข้าพเจ้าได้รับความรู้ ความเข้าใจและประสบการณ์ในการทำงานที่เป็นประโยชน์อย่างมาก อีกทั้งการดูแลและการช่วยเหลือต่าง ๆ ตลอดเวลาการทำงาน โดยการปฏิบัติงานสหกิจศึกษาในครั้งนี้สำเร็จลุล่วงได้ เพราะมีการชี้แนะและได้รับความร่วมมือจากคุณพิบูลย์ พรประสิทธิ์ ทีม เน็ตเวิร์ค ทุกคน และข้าพเจ้าขอขอบคุณอาจารย์ที่ปรึกษา ผศ.มยุรี เลิศเวชกุล ที่คอยให้คำแนะนำ คำปรึกษาและคอยรับฟังและช่วยเหลือปัญหาต่าง ๆ ในการทำโครงการครั้งนี้ และท้ายที่สุดข้าพเจ้าขอขอบคุณครอบครัวที่คอยให้กำลังใจที่ดีแก่ข้าพเจ้าเสมอมาทำให้ปริญญาโทฉบับนี้ให้สำเร็จลุล่วงไปได้ด้วยดี

ณัฐญา บุราณรักษ์

สารบัญ

หน้า

บทคัดย่อ	ii
ABSTRACT	iii
กิตติกรรมประกาศ	iv
สารบัญ.....	v
สารบัญรูป	viii
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของการปฏิบัติงาน	1
1.3 วิธีการดำเนินงาน	1
1.4 ขอบเขตของงาน.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 Hyper Text Markup Language	3
2.1.1 รูปแบบของ HTML.....	3
2.1.1.1 ส่วนของคำสั่ง	3
2.1.1.2 ส่วนของบทความทั่วไป	4
2.1.2 โครงสร้างคำสั่งของ HTML.....	4
2.1.3 โครงสร้างของหลักของ HTML.....	4
2.1.3.1. head.....	5
2.1.3.2. body.....	5
2.2 Cascading Style Sheet	9

2.2.1	ประโยชน์ของ CSS	9
2.2.2	โครงสร้างคำสั่ง	9
2.3	JavaScript	11
2.3.1	ประโยชน์ JavaScript	11
2.3.2	ลักษณะการทำงานของ JavaScript.....	12
2.3.3	JavaScript กับ HTML	12
2.3.4	การเรียกใช้ Script.....	13
2.3.5	รูปแบบการเขียน	14
2.3.6	สัญลักษณ์การกระทำทางคณิตศาสตร์และอื่นๆ	14
2.3.7	ตัวแปรและ Syntax.....	14
2.3.8	เงื่อนไข และการวนรอบ.....	15
2.3.9	ฟังก์ชันและตัวแปรของภาษา.....	16
2.4	Bootstrap.....	16
2.4.1	โครงสร้างของ Bootstrap Framework.....	16
2.5	DataTables	17
2.6	Sublime Text	18
2.7	หลักการออกแบบเว็บไซต์	18
2.7.1	การออกแบบโครงสร้างเว็บไซต์	18
2.7.2	รูปแบบของโครงสร้างเว็บไซต์.....	19
2.7.3	การออกแบบระบบเนวิเกชัน.....	19
2.7.4	ออกแบบขนาดของเว็บเพจ	20
2.7.5	ลำดับความสำคัญของเนื้อหา	20
2.7.6	การเลือกใช้สี.....	21
2.7.7	ความสำคัญกับส่วนบนของเว็บเพจ	21
2.7.8	ส่วนประกอบของหน้าเว็บเพจ.....	21
2.7.10	แนวคิดในการออกแบบเว็บเพจ.....	22

บทที่ 3	ขั้นตอนการดำเนินงาน.....	23
3.1	วางแผนจัดทำเว็บไซต์	23
3.1.1	เก็บความต้องการของผู้ใช้งานในทีม	23
3.1.2	ติดตั้งโปรแกรมที่ต้องใช้	23
3.1.2.1	การติดตั้ง Xampp	23
3.1.2.2	การติดตั้ง Bootstrap.....	27
3.1.2.3	การติดตั้ง Sublime Text	28
3.2	กำหนดโครงสร้างเว็บไซต์.....	30
3.3	กำหนดการเชื่อมโยงระหว่างเว็บเพจ	30
3.4	ออกแบบหน้าเว็บแต่ละหน้า.....	30
บทที่ 4	ผลการดำเนินการ.....	31
4.1	การเข้าสู่ระบบ	31
4.2	Admin Page.....	32
4.3	การเปลี่ยนรหัสผ่าน.....	39
4.4	Implementer Page.....	40
4.5	SCoA&D Page.....	41
4.6	Network Monitor Page.....	42
บทที่ 5	สรุปผลการดำเนินการ.....	43
5.1	สรุปผลการดำเนินงาน.....	43
5.2	อุปสรรค.....	43
5.3	ข้อเสนอแนะ	43
เอกสารอ้างอิง	44

สารบัญรูป

รูปที่ 2.1 ตัวอย่าง DataTables.....	17
รูปที่ 2.2 ตัวอย่าง Sublime Text.....	18
รูปที่ 3.1 เว็บดาวนโหลด XAMPP	23
รูปที่ 3.2 การติดตั้ง XAMPP.....	24
รูปที่ 3.3 การเลือก Components ของ XAMPP.....	24
รูปที่ 3.4 การเลือกตำแหน่งของ XAMPP	24
รูปที่ 3.5 Bitnami ของ XAMPP	25
รูปที่ 3.6 การเตรียมการติดตั้ง XAMPP	25
รูปที่ 3.7 รอการติดตั้ง XAMPP	25
รูปที่ 3.8 การติดตั้ง XAMPP เรียบร้อย.....	26
รูปที่ 3.9 XAMPP Control Panel.....	26
รูปที่ 3.10 หน้าเว็บ Localhost ของ XAMPP.....	26
รูปที่ 3.11 เว็บไซต์ Sublime Text	28
รูปที่ 3.12 Sublime Text Setup.....	28
รูปที่ 3.13 Location Sublime Text.....	28
รูปที่ 3.14 Additional Task.....	29
รูปที่ 3.15 เตรียมติดตั้ง Sublime Text	29
รูปที่ 3.16 Console Sublime Text	29
รูปที่ 4.1 Sign in Page	31
รูปที่ 4.2 Register Page	31
รูปที่ 4.3 ตาราง movement ของ admin	32
รูปที่ 4.4 แสดงข้อมูล ตาราง movement.....	32
รูปที่ 4.5 การแก้ไขข้อมูลตาราง movement.....	33
รูปที่ 4.6 การลบข้อมูลในตาราง movement.....	33
รูปที่ 4.7 การค้นหาข้อมูล.....	34
รูปที่ 4.8 การเพิ่มข้อมูลไปยัง movement	34
รูปที่ 4.9 Serial Page	35

รูปที่ 4.10 การเพิ่มข้อมูลตาราง Serial.....	35
รูปที่ 4.11 การแสดงข้อมูลในตาราง Serial	36
รูปที่ 4.12 การแก้ไขข้อมูลในตาราง Serial.....	36
รูปที่ 4.13 การลบข้อมูลในตาราง Serial.....	36
รูปที่ 4.14 การเพิ่มข้อมูลไปยัง movement	37
รูปที่ 4.15 Member Page.....	37
รูปที่ 4.16 การแสดงข้อมูลสมาชิก	38
รูปที่ 4.17 การแก้ไขข้อมูลสมาชิก	38
รูปที่ 4.18 การลบสมาชิก	38
รูปที่ 4.19 ปุ่มแสดงการเปลี่ยนรหัสผ่าน	39
รูปที่ 4.20 การเปลี่ยนรหัสผ่าน.....	39
รูปที่ 4.21 ตาราง movement ของ Implementer.....	40
รูปที่ 4.22 การแสดงข้อมูลตาราง movement ของ Implementer	40
รูปที่ 4.23 การแก้ไขข้อมูลจากตาราง movement ของ Implementer	40
รูปที่ 4.24 SCoA&D Page	41
รูปที่ 4.25 การแสดงข้อมูลในตาราง movement ของ SCoA&D.....	41
รูปที่ 4.26 การตรวจสอบข้อมูลของ SCoA&D	41
รูปที่ 4.27 Network monitor Page.....	42
รูปที่ 4.28 การแสดงข้อมูลตาราง movement ของ Network monitor	42
รูปที่ 4.29 การตรวจสอบข้อมูลของ Network monitor.....	42

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

ในการปฏิบัติงานในทีมเน็ตเวิร์ค บริษัท ไอบีเอ็ม โซลูชันส์ ดิลิเวอรี จำกัด จะมีการนำเอาอุปกรณ์เครือข่ายออกไปเพื่อใช้ในการติดตั้งให้แก่ลูกค้า หรือนำอุปกรณ์ไปเพื่อประกอบการทำงานด้วยเสมอ ซึ่งระเบียบในการทำงานได้กำหนดไว้ให้พนักงานจะต้องลงบันทึกการนำอุปกรณ์ออก และการนำเข้าอุปกรณ์ทุกครั้ง แต่อย่างไรก็ตามมักจะเกิดปัญหาที่เกิดจากการที่มีผู้นำอุปกรณ์ออกไปโดยไม่ลงบันทึกข้อมูล หรือเกิดจากความล่าช้าในการรวมไฟล์ข้อมูลซึ่งจัดเก็บในรูปแบบของไฟล์เอ็กเซล (Microsoft Excel) ทำให้ข้อมูลที่มีอยู่ในระบบไม่เป็นข้อมูลที่ล่าสุด เมื่อเช็คสินค้าคงคลังจะพบว่าจำนวนอุปกรณ์ที่มีอยู่ในระบบไม่ตรงกับจำนวนสินค้าที่มีอยู่จริง จึงมีแนวคิดในการจัดทำระบบข้อมูลสินค้าคงคลังขึ้นมาเพื่อใช้ประโยชน์ในการติดตามอุปกรณ์ สามารถรู้สถานะที่เก็บอุปกรณ์ และจำนวนอุปกรณ์ได้แม่นยำมากยิ่งขึ้น ระบบข้อมูลสินค้าคงคลังที่พัฒนาขึ้นมีส่วนช่วยให้การทำงานในทีมเน็ตเวิร์คเป็นระบบและมีประสิทธิภาพมากยิ่งขึ้น

1.2 วัตถุประสงค์ของการปฏิบัติงาน

ในการปฏิบัติงานในทีมเน็ตเวิร์ค บริษัท ไอบีเอ็ม โซลูชันส์ ดิลิเวอรี จำกัด จะมีการนำเอาอุปกรณ์เครือข่ายออกไปเพื่อใช้ในการติดตั้งให้แก่ลูกค้า หรือนำอุปกรณ์ไปเพื่อประกอบการทำงานด้วยเสมอ ซึ่งระเบียบในการทำงานได้กำหนดไว้ให้พนักงานจะต้องลงบันทึกการนำอุปกรณ์ออก และการนำเข้าอุปกรณ์ทุกครั้ง แต่อย่างไรก็ตามมักจะเกิดปัญหาที่เกิดจากการที่มีผู้นำอุปกรณ์ออกไปโดยไม่ลงบันทึกข้อมูล หรือเกิดจากความล่าช้าในการรวมไฟล์ข้อมูลซึ่งจัดเก็บในรูปแบบของไฟล์เอ็กเซล (Microsoft Excel) ทำให้ข้อมูลที่มีอยู่ในระบบไม่เป็นข้อมูลที่ล่าสุด เมื่อเช็คสินค้าคงคลังจะพบว่าจำนวนอุปกรณ์ที่มีอยู่ในระบบไม่ตรงกับจำนวนสินค้าที่มีอยู่จริง จึงมี แนวคิดในการจัดทำระบบข้อมูลสินค้าคงคลังขึ้นมาเพื่อใช้ประโยชน์ในการติดตามอุปกรณ์ สามารถรู้สถานะที่เก็บอุปกรณ์ และจำนวนอุปกรณ์ได้แม่นยำมากยิ่งขึ้น ระบบข้อมูลสินค้าคงคลังที่พัฒนาขึ้นมีส่วนช่วยให้การทำงานในทีมเน็ตเวิร์คเป็นระบบและมีประสิทธิภาพมากยิ่งขึ้น

1.3 วิธีการดำเนินงาน

- ศึกษาระบบการทำงานในองค์กร
- ศึกษาโปรแกรม และภาษาที่เกี่ยวข้องในการทำเว็บไซต์
- วางแผนจัดทำเว็บไซต์เพื่อให้ตรงตามความต้องการ
- กำหนดโครงสร้างเว็บไซต์

- กำหนดการเชื่อมโยงระหว่างเว็บเพจ
- ออกแบบหน้าเว็บเพจแต่ละหน้า
- ทดสอบการใช้งาน และแก้ไขปัญหาที่เกิดขึ้น

1.4 ขอบเขตของงาน

- ศึกษาเรียนรู้ระบบการทำงานในทีมเน็ตเวิร์ค
- จัดเก็บข้อมูลเกี่ยวกับความต้องการของระบบเว็บไซต์
- ศึกษา Frontend-framework ผ่าน Bootstrap เพื่อช่วยในการจัดทำเว็บไซต์
- ออกแบบเว็บไซต์ข้อมูลสินค้าคงคลัง โดยมีรายละเอียด ประกอบด้วย
 - Admin Page
 - Movement
 - Serial
 - Member
 - Implementer Page
 - SCoA&D Page
 - Network Monitor Page
- เว็บไซต์ที่ออกแบบรองรับการแสดงผลผ่านทาง เว็บเบราว์เซอร์ตามขนาดจอภาพของเครื่องคอมพิวเตอร์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- นักศึกษาได้รับความรู้และเข้าใจในด้านของเน็ตเวิร์ค
- นักศึกษาได้พัฒนาระบบการทำงานให้สะดวกยิ่งขึ้น
- นักศึกษาได้ความรู้ความเข้าใจของ frontend website มากขึ้น
- บริษัทฯ ได้ใช้งานเว็บไซต์เพื่อความสะดวกและการติดตามสินค้าได้ง่ายขึ้น

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 Hyper Text Markup Language

Hyper Text Markup Language (HTML) คือภาษาคอมพิวเตอร์ที่ใช้ในการแสดงผลของเอกสารบนเว็บไซต์ (website) หรือว่าเว็บเพจ (webpage) นั้นเอง ซึ่งถูกพัฒนาและกำหนดมาตรฐานโดยองค์กร World Wide Web Consortium (W3C) และจากการพัฒนาทางด้านซอฟต์แวร์ของ Microsoft ทำให้ภาษา HTML เป็นอีกภาษาหนึ่งที่ใช้เขียนโปรแกรมได้ หรือ HTML Application นั้นเอง ภาษา HTML เป็นประเภทมาร์กอัป (Markup) สำหรับการการสร้างเว็บเพจโดยใช้ภาษา HTML สามารถทำได้โดยใช้โปรแกรม Text Editor ต่างๆ เช่น Notepad, Edit Plus, Sublime Text หรือจะอาศัยโปรแกรมที่เป็นเครื่องมือช่วยสร้างเว็บเพจ เช่น Microsoft FrontPage, Dream Weaver ซึ่งอำนวยความสะดวกในการสร้างหน้า HTML การเรียกใช้งานหรือทดสอบการทำงานของเอกสาร HTML จะใช้โปรแกรมเว็บเบราว์เซอร์ (web browser) เช่น Microsoft Internet Explorer (IE), Mozilla Firefox, Safari, Opera, Netscape Navigator เป็นต้น

2.1.1 รูปแบบของ HTML จะแบ่งออกเป็น 2 ส่วน คือ

- 2.1.1.1 ส่วนของคำสั่ง (tag) เป็นส่วนที่กำหนดรูปแบบของข้อความที่แสดง ซึ่งเราเรียกว่า Tag โดยจะอยู่ในเครื่องหมาย < ... >
 1. <html> </html> ในการใช้งาน HTML เราจะต้องเริ่มด้วย <html> และปิดด้วย </html> เสมอ
 2. <head> </head> เป็นส่วนที่ใช้ให้รายละเอียดเกี่ยวกับเว็บเพจหน้านี้ ซึ่งจะไม่แสดงให้เห็นในส่วนของการแสดงผลของเว็บเบราว์เซอร์แต่จะมีผลกับส่วนอื่นๆ เช่น การหาของ search engine ในเว็บ google หรือ yahoo นั้นเอง การใช้งานก็จะมีคำสั่งย่อยเพื่อบรรยายรายละเอียด เช่น <title> </title>, <meta> และอื่นๆ
 3. <title> </title> ในส่วนตัวอักษรที่อยู่ในคำสั่งนี้จะอยู่ใน title bar ของเว็บเพจ
 4. <body> </body> ตัวอักษรที่อยู่ในคำสั่งนี้จะแสดงส่วนแสดงผลของเว็บเบราว์เซอร์

2.1.1.2. ส่วนของบทความทั่วไป เป็นส่วนของข้อความที่เราต้องการแสดงผล

ตัวอย่างการใช้งานภาษา HTML

```
<html>
  <head>
    <title> หัวข้อเรื่อง </title>
  </head>
  <body>
    เนื้อหา
  </body>
</html>
```

2.1.2 โครงสร้างคำสั่งของ HTML

1. คำสั่ง หรือ Tag เป็นลักษณะเฉพาะของภาษา HTML ใช้ในการระบุรูปแบบคำสั่ง หรือการลงรหัสคำสั่ง HTML ภายในเครื่องหมายน้อยกว่า (<) และ เครื่องหมายมากกว่า (>) โดยที่ Tag HTML แบ่งได้ 2 ลักษณะ คือ Tag เดี่ยว เป็น Tag ที่ไม่ต้องมีการปิดรหัส เช่น <HR>,
 เป็นต้น และ Tag เปิด/ปิด รูปแบบของ tag นี้จะเป็นแบบ <tag> </tag> โดยที่ <tag> เราเรียกว่า tag เปิด </tag> เราเรียกว่า tag ปิด

2. Attributes เป็นตัวบอกรายละเอียดของ tag นั้น เช่น ... เป็นการบอกให้ให้อักษรที่อยู่ใน tag นี้ชิดขวา

3. Not case sensitive คือการพิมพ์ตัวพิมพ์ใหญ่หรือตัวพิมพ์เล็กก็ได้ผลลัพธ์เหมือนกัน เช่น การใช้คำสั่ง
 หรือ
 ผลลัพธ์ที่ได้ออกมาเป็นการขึ้นบรรทัดใหม่เหมือนกัน

2.1.3 โครงสร้างของหลักของ HTML

```
<html>
  <head> คำสั่งในหัวข้อของ head </head>
  <body>
    เป็นส่วนที่ใช้แสดงผล
  </body>
</html>
```

ภาษา HTML นั้นจะเริ่มด้วย <html> และจบด้วย </html> เสมอ ซึ่งชุดคำสั่งที่ใช้จะแยกเป็น 2 ส่วน

2.1.3.1. head คำสั่งที่อยู่ในส่วนนี้จะใช้บรรยายรายละเอียดเกี่ยวกับเว็บเพจซึ่งจะไม่แสดงผลที่เว็บเพจโดยตรง คำสั่งในหัวข้อของ head (Head Section) เป็นส่วนที่ใช้อธิบายเกี่ยวกับข้อมูลเฉพาะของหน้าเว็บนั้นๆ เช่น ชื่อเรื่องของหน้าเว็บ (Title), ชื่อผู้จัดทำเว็บ (Author), คีย์เวิร์ดสำหรับการค้นหา (Keyword) โดยมี Tag สำคัญ คือ

<HEAD>

<TITLE>ข้อความอธิบายชื่อเรื่องของเว็บ</TITLE>

<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=TIS-620">

<META NAME="Author" CONTENT="ชื่อผู้พัฒนาเว็บ">

<META NAME="KeyWords" CONTENT="ข้อความ 1, ข้อความ 2 ">

</HEAD>

1. TITLE ข้อความที่ใช้เป็น TITLE ไม่ควรพิมพ์เกิน 64 ตัวอักษร, ไม่ต้องใส่ลักษณะพิเศษ เช่น ตัวหนา, เอียง หรือสี โดยข้อความในส่วนนี้จะแสดงผลใน title bar ของเว็บเบราว์เซอร์

2. META Tag META จะไม่ปรากฏผลบนเบราว์เซอร์ แต่จะเป็นส่วนสำคัญ ในการจัดอันดับบัญชีเว็บ สำหรับผู้ให้บริการสืบค้นเว็บ (Search Engine เช่น google, yahoo)

3.charset=TIS-620 ใช้บอกว่าใช้ชุดตัวอักษรแบบใดในการแสดงผลภาษาไทยเราใช้ charset=TIS-620, charset=windows-874 หรือcharset=utf-8 ก็ได้

4. keywords ดังภาพด้านบนจะเห็นว่าเราสามารถใส่ keywords มากกว่า 1 คำได้โดยใช้เครื่องหมาย (,) ในการคั่นระหว่างคำ

การพิมพ์ชุดคำสั่ง HTML สามารถพิมพ์ได้ทั้งตัวพิมพ์เล็ก ตัวพิมพ์ใหญ่ หรือผสม การย่อหน้า เว้นบรรทัด หรือช่องว่าง สามารถกระทำได้อิสระ โปรแกรมเบราว์เซอร์จะไม่สนใจเกี่ยวกับระยะเว้นบรรทัดหรือย่อหน้า หรือช่องว่าง

2.1.3.2. body คำสั่งที่อยู่ในส่วนนี้จะใช้ในการจัดรูปแบบตัวอักษร จัดหน้า ใส่รูปภาพ ซึ่งตัวอักษรในส่วนนี้จะแสดงที่เว็บเบราว์เซอร์โดยตรง คำสั่งในส่วนของ (Body Section) เป็นส่วนเนื้อหาหลักของหน้าเว็บซึ่งการแสดงผลจะต้องใช้ Tag จำนวนมาก ขึ้นอยู่กับลักษณะของข้อมูล เช่น ข้อความ รูปภาพเสียง หรือไฟล์ต่างๆ ส่วนเนื้อหาเอกสารเว็บ เป็นส่วนการทำงานหลักของหน้าเว็บ ประกอบด้วย Tag มากมายตามลักษณะของข้อมูลที่ต้องการนำเสนอ การป้อนคำสั่งในส่วนนี้ ไม่มีข้อจำกัดสามารถป้อนติดกันหรือ 1 บรรทัดต่อ 1 คำสั่งก็ได้ แต่มักจะยึดรูปแบบที่อ่านง่ายคือการทำย่อหน้าในชุดคำสั่งที่เกี่ยวข้องกันทั้งหมดนี้ให้ป้อนคำสั่งทั้งหมดภายใต้ Tag <BODY> </BODY>

โดยการแบ่งกลุ่มคำสั่งได้ดังนี้

1. กลุ่มคำสั่งเกี่ยวกับการจัดรูปแบบเอกสาร

- คำสั่งที่ใช้ในการจัดย่อหน้า คำสั่ง P นี้จะเพิ่มบรรทัดว่างก่อนและหลังตัวอักษรที่เราพิมพ์ไป

โดยอัตโนมัติ ตามที่แสดงในตัวอย่าง

```
<p> ย่อหน้าที่ 1 </p>
```

```
<p> ย่อหน้าที่ 2 </p>
```

```
<p> ย่อหน้าที่ 3 </p>
```

- คำสั่งที่ใช้ในการขึ้นบรรทัดใหม่ การขึ้นบรรทัดใหม่นั้นปกติเว็บเบราว์เซอร์จะทำการตัดให้อยู่แล้วแต่การตัดคำของเว็บเบราว์เซอร์จะถูกตัดก็ต่อเมื่อแสดงผลไม่ได้ แต่ถ้าเราใส่คำสั่ง
 เข้าไปเว็บเบราว์เซอร์จะตัดให้ทันที ซึ่งอาจจำเป็นที่จะต้องตัดคำเป็นบรรทัดสั้นๆ เช่น การเขียนกลอนดังตัวอย่าง

```
บรรทัดที่ 1 <br>
```

```
บรรทัดที่ 2 <br>
```

```
บรรทัดที่ 3 <br>
```

- คำสั่งที่ใช้กับข้อความที่เป็นหัวเรื่อง โดยใช้คำสั่ง h จะมีทั้งหมด 6 ลำดับด้วยกัน เริ่มตั้งแต่ h1, h2, h3, h4, h5, h6 ซึ่งขนาดของ h1 จะมีขนาดตัวหนังสือใหญ่ที่สุดดังตัวอย่างที่แสดง โดยเราจะใช้กับตัวอักษรที่ต้องการให้เป็นหัวเรื่องเพื่อให้อักษรนั้นโดดเด่นขึ้นมา แต่จะสังเกตได้ว่าเมื่อใช้ h tag จะตัดตัวอักษรที่ต่อจาก h tag เป็นบรรทัดใหม่อัตโนมัติ

```
<h1>head 1</h1>
```

```
<h2>head 2</h2>
```

```
<h3>head 3</h3>
```

```
<h4>head 4</h4>
```

```
<h5>head 5</h5>
```

```
<h6>head 6</h6>
```

- คำสั่งที่ใช้ในการขีดเส้นคั่น

```
<p>เนื้อหาบทที่ 1</p>
```

```
<hr>
```

- คำสั่งที่ใช้ในการจัดตัวอักษรชิดซ้าย ชิดขวา หรือกึ่งกลาง การจัดให้ตัวอักษรให้ชิดซ้าย ขวา หรือกึ่งกลาง โดยใช้ Attributes ให้รายละเอียดของ tag โดยเราจะใช้ align เพื่อบอกว่าให้ชิดซ้าย (align = 'left') ชิดขวา (align = 'right') และ จัดกึ่งกลาง (align = 'center')

```
<h3 align = 'left'>ชิดซ้าย</h3>
```

```
<h3 align = 'right'>ชิดขวา</h3>
```

```
<h3 align = 'center'>จัดเข้ากลาง</h3>
```

- คำสั่งที่ใช้ในการเปลี่ยนสีพื้นหลัง โดยใช้คำสั่ง bgcolor ซึ่งเป็นอีกหนึ่ง Attributes เหมือนกันที่ใช้กำหนดสี อาจเปลี่ยนจาก สีเขียว(green) เป็น เหลือง(yellow) หรือสีอื่นๆก็ได้

```
<body bgcolor = 'green'>
```

```
<h1> ดูสีพื้นหลัง </h1>
```

```
</body>
```

- การเขียนคำบรรยาย source code ในส่วนของคำบรรยาย source code นั้นจะไม่แสดงที่เว็บเบราว์เซอร์เขียนเพื่อบรรยายว่า source code ส่วนนี้ใช้ทำอะไรเพื่อความสะดวกเมื่อกลับมาแก้ไข source code ในภายหลัง

```
<!-- คำบรรยาย source code --!>
```

2. คำสั่งในการกำหนดลักษณะตัวอักษร และแสดงสัญลักษณ์ต่างๆ

- คำสั่งที่ใช้ในการจัดลักษณะตัวอักษรเป็นคำสั่งที่ใช้ในการจัดรูปแบบทั่วไป เช่น ตัวเอียง ตัวหนา คำสั่งเราจะแบ่งได้เป็น 2 พวกดังนี้

1. แบ่งตามลักษณะที่ปรากฏ เช่นตัวเอียง ตัวหนา

```
<B> ตัวอักษรแบบตัวหนา (bold)
```

```
<I> ตัวอักษรแบบตัวเอน (italic)
```

```
<S> ตัวอักษรแบบตัวขีดฆ่า (strike)
```

```
<Sub> ตัวอักษรแบบตัวห้อย (subscripted)
```

```
<Sup> ตัวอักษรแบบตัวยก (superscripted)
```

```
<U> ตัวอักษรแบบขีดเส้นใต้ (underline)
```

2. แบ่งตามการใช้งาน เช่น ใช้กับคำพูดหรือวลี ใช้กับข้อความที่สำคัญมาก

```
<Em> ใช้เน้นข้อความ คำพูดหรือวลี (emphasized)
```

```
<Strong> ใช้เน้นข้อความที่สำคัญมากๆ (strong)
```

- <Ins> ใช้เน้นข้อความที่แก้ไขเพิ่มเติม (inserted)
- ใช้บอกว่าข้อความนี้ถูกลบไปแล้ว (deleted)
- <Code> ใช้บอกว่าข้อความที่เป็นโปรแกรม (computer code)
- <Address> ใช้บอกว่าข้อความที่เป็นที่อยู่ (computer code)

- คำสั่งที่ทำให้รูปแบบตัวอักษรใน source code เหมือนกับที่แสดงผล Pre tag จะมีประโยชน์มากในการที่เราจะแสดงบทความที่มีเนื้อหาหมาก หรือ คัดลอกเนื้อหาจากที่อื่นมาทำให้เราไม่ต้องขึ้นบรรทัดใหม่ด้วยคำสั่ง br และใช้ในการแสดง source code ได้ดีอีกด้วย

```
<pre> ..... </pre>
```

- คำสั่งแสดงสัญลักษณ์พิเศษต่างๆ สัญลักษณ์พิเศษบางตัวเช่น เครื่องหมายมากกว่า (>) หรือ เครื่องหมายน้อยกว่า (<) ซึ่งเครื่องหมายเหล่านี้ใช้ในภาษา HTML ด้วยทำให้ตัวอักษรที่อยู่ในเครื่องหมายเหล่านี้กลายเป็น Tag หมด หรือเครื่องหมายที่ไม่มีบนคีย์บอร์ด แต่ถ้าเราต้องการแสดงเครื่องหมายเหล่านี้เราต้องใช้ Entity Name แทน

3. คำสั่งเกี่ยวกับการทำลิงก์ HTML นั่นคือ ไฮเปอร์ลิงก์ (hyperlink) หรือเราเรียกกันสั้นว่าลิงก์ (link) ซึ่งเราจะเห็นอยู่ในทุกเว็บไซต์ ใช้เพื่อเปิดเว็บเพจอื่นๆใน เว็บไซต์ของเรา หรือเชื่อมโยงไปที่เว็บไซต์หรือเว็บเพจอื่นๆ ซึ่งมีความสำคัญมากในเว็บไซต์

- การสร้าง hyperlink มีการสร้างเพื่อลิงก์ภายในเว็บไซต์ด้วยกัน ลิงก์ไปยังเว็บไซต์ภายนอก และ ลิงก์ไปยังรูปภาพซึ่งมีหลักการสร้าง hyperlink ดังรูปแบบด้านล่าง

รูปแบบของ hyperlink ภายใน directory เดียวกัน

```
<a href='ชื่อไฟล์ที่ต้องการลิงก์'> ตัวอักษรที่จะใช้แสดง </a>
```

รูปแบบของ hyperlink จาก directory ที่สูงกว่า

```
<a href='ชื่อ directory ที่เก็บไฟล์ไว้/ชื่อไฟล์ที่ต้องการลิงก์'> ตัวอักษรที่จะใช้แสดง </a>
```

รูปแบบของ hyperlink จาก directory ที่ต่ำกว่า 1 ชั้น

```
<a href='../ชื่อไฟล์ที่ต้องการลิงก์'> ตัวอักษรที่จะใช้แสดง </a>
```

รูปแบบของ hyperlink กับเว็บภายนอก

```
<a href='http://domain name'> ตัวอักษรที่จะใช้แสดง </a>
```

รูปแบบของการสร้าง hyperlink ด้วยรูปภาพ

```
<a href='หน้าที่ต้องการเชื่อมโยง'>
```

```
<img border = '0' src = 'ที่อยู่ภาพ/ชื่อภาพ'> </a>
```

2.2 Cascading Style Sheet

Cascading Style Sheet (CSS) หรือเรียกอีกอย่างหนึ่งว่าสไตล์ชีต (stylesheet) คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (Style) ของเนื้อหาในเอกสารนั้น ได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบนี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสาร ไม่ขึ้นอยู่กับเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลลัพท์ของเอกสาร HTML โดยเฉพาะในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสารบ่อยครั้งหรือต้องการควบคุมให้รูปแบบการแสดงผลเอกสาร HTML มีลักษณะของความสม่ำเสมอทั่วกันทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน โดยกฎเกณฑ์ในการกำหนดรูปแบบ เอกสาร HTML ถูกเพิ่มเข้ามาครั้งแรก ใน HTML 4.0 เมื่อปีพ.ศ. 2539 ในรูปแบบของ CSS level 1 Recommendations ที่กำหนดโดย องค์กร World Wide Web Consortium หรือ W3C

2.2.1 ประโยชน์ของ CSS

- CSS มีคุณสมบัติมากกว่า tag ของ html เช่น การกำหนดกรอบให้ข้อความ สีข้อความ รูปแบบของข้อความ เป็นต้น
- CSS นั้นสามารถกำหนดดึงมาใช้โดยการกำหนดคำสั่งไว้ที่ต้นของไฟล์ html หรือตำแหน่งอื่น ๆ ก็ได้ และสามารถมีผลกับเอกสารทั้งหมด แสดงว่ากำหนดชุดคำสั่ง CSS เพียงครั้งเดียว จุดเดียวก็มีผลกับการแสดงผลทั้งหมดทำให้เวลาแก้ไขหรือปรับปรุงทำได้สะดวกไม่ต้องไล่ตามแก้ tag ต่างๆ ทั่วทั้งเอกสาร
- CSS สามารถกำหนดแยกไว้ต่างหากจากไฟล์เอกสาร html และสามารถนำมาใช้ร่วมกับเอกสารหลายไฟล์ได้ ซึ่งการแก้ไขก็แก้ไขเพียงจุดเดียวก็ใช้ได้กับเอกสารทั้งหมด
- CSS กับ HTML นั้นทำหน้าที่คนละอย่างกัน โดย HTML / XHTML จะทำหน้าที่ในการวางโครงร่างเอกสารอย่างเป็นรูปแบบ ถูกต้อง เข้าใจง่าย ไม่เกี่ยวข้องกับการแสดงผล ส่วน CSS จะทำหน้าที่ในการตกแต่งเอกสารให้สวยงาม เรียกได้ว่า HTML คือส่วน coding ส่วน CSS คือส่วน design นั่นเอง

2.2.2 โครงสร้างคำสั่ง

คำสั่งของ CSS จะประกอบด้วย selector, property และ value

```
selector { property:value }
```

```
selector { property1:value1; property2:value2 }
```

1. selector สามารถเป็น HTML Tag ต่างๆ เช่น <body>, <p> หรือเป็น Class name หรือ ID ที่เรที่ตั้งชื่อให้ก็ได้

2. property คือ คุณสมบัติในการจัดรูปแบบการแสดงผล เช่น color สำหรับกำหนดสี, font-size สำหรับกำหนดขนาดตัวอักษร

3. value เป็น ค่า ที่เรากำหนดให้กับ property ต่างๆ เช่น color:white, font-size:14px
ตัวอย่างคำสั่ง CSS

- กำหนดให้ข้อความที่อยู่ใน Tag <p> เป็นสีขาว และวางอยู่ด้านขวา

```
/* selector ที่เป็น HTML Tag */  
p {  
    color:#FFFFFF;  
    text-align:right  
}
```

- กำหนดให้ข้อความที่ class name topic เป็นสีน้ำเงิน ชนิดอักษรเป็น Arial ตัวหนา และจัดวางอยู่ด้านซ้าย

```
/* selector ที่เป็น Class name */  
.topic{  
    color:blue;  
    font-family:Arial;  
    font-weight:bold;  
    text-align:left;  
}
```

กรณีที่ selector มีค่า property เหมือนกัน สามารถเขียนรวมกันได้ โดยใช้เครื่องหมาย "," คั่นระหว่าง selector

- กำหนดให้ข้อความใน Tag <h1>,<h2> และ <h3> เป็นสีแดง

```
h1, h2, h3 {  
    color:red;  
}
```

Comment ใน Style Sheet Comment จะใช้เครื่องหมาย "/*" เป็นการเปิด และ "*/" เป็นการปิด เช่น

```
/* comment กำหนดสีตัวอักษรเป็นสีดำ ขนาด16px */  
body {  
color:#000000;  
font-size:16px  
}
```

2.3 JavaScript

JavaScript คือภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง JavaScript เป็นภาษาสคริปต์เชิงวัตถุที่เรียกกันว่าสคริปต์ (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์จะใช้ร่วมกับ HTML เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะแปลความและดำเนินงานไปที่คำสั่ง (interpret) หรือเรียกว่า อ็อบเจ็กโอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server)

2.3.1 ประโยชน์ JavaScript

- JavaScript ทำให้สามารถใช้เขียนโปรแกรมแบบง่ายๆได้ โดยไม่ต้องพึ่งภาษาอื่น
- JavaScript มีคำสั่งที่ตอบสนองกับผู้ใช้งาน เช่นเมื่อผู้ใช้คลิกที่ปุ่ม หรือ Checkbox ก็สามารถสั่งให้เปิดหน้าต่างใหม่ได้ ทำให้เว็บไซต์ของเรามีปฏิสัมพันธ์กับผู้ใช้งานมากขึ้น ซึ่งเป็นข้อดีของ JavaScript ที่ทำให้เว็บไซต์ต่างๆทั้งหลายเช่น Google Map ต่างหันมาใช้ JavaScript ในการพัฒนา
- JavaScript สามารถเขียนหรือเปลี่ยนแปลง HTML Element ได้ นั่นคือสามารถเปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้ หรือหน้าแสดงเนื้อหาสามารถซ่อนหรือแสดงเนื้อหาได้แบบง่ายๆนั่นเอง
- JavaScript สามารถใช้ตรวจสอบข้อมูลได้ สังเกตว่าเมื่อเรากรอกข้อมูลบางเว็บไซต์ เช่น Email เมื่อเรากรอกข้อมูลผิดจะมีหน้าต่างฟ้องขึ้นมาว่าเรากรอกผิด หรือลืมกรอกอะไรบางอย่าง
- JavaScript สามารถใช้ในการตรวจสอบผู้ใช้ได้เช่น ตรวจสอบได้ว่าผู้ใช้งานกำลังใช้เว็บ

เบราว์เซอร์ใดในการทำงาน

- JavaScript สร้าง Cookies เก็บข้อมูลของผู้ใช้ในคอมพิวเตอร์ของผู้ใช้เองได้

2.3.2 ลักษณะการทำงานของ JavaScript

JavaScript มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนเอกสารด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ทำงานร่วมกับ ภาษา HTML และภาษาจาวาได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server) โดยมีลักษณะการทำงานดังนี้

1. Navigator JavaScript เป็น Client-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่งไคลเอนต์จึงมีความเหมาะสมต่อการใช้งานของผู้ใช้ทั่วไปเป็นส่วนใหญ่
2. LiveWire JavaScript เป็น Server-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่งเซิร์ฟเวอร์สามารถใช้ได้เฉพาะกับ LiveWire ของเน็ตสเคป โดยตรง

2.3.3 JavaScript กับ HTML

การเขียน JavaScript เราอาจเขียนรวมอยู่ในไฟล์เดียวกับ HTML ได้ ซึ่งแตกต่างจากการเขียนโปรแกรมภาษา Java ที่ต้อง เขียนแยกออกเป็นไฟล์ต่างหาก ไม่สามารถเขียนรวมอยู่ในไฟล์เดียวกับ HTML ได้ วิธีการเขียน JavaScript เพื่อสั่งให้เว็บเพจทำงาน มีอยู่ด้วยกัน 2 วิธี ดังนี้ เขียนด้วยชุดคำสั่ง และฟังก์ชันของ JavaScript เอง หรือ เขียนตามเหตุการณ์ที่เกิดขึ้นตามการใช้งานจากชุดคำสั่งของ HTML เมื่อเริ่มใช้งาน โปรแกรมเบราว์เซอร์ จะอ่านข้อมูลจากส่วนบนของเพจ HTML และทำงานไปตามลำดับจาก บนลงล่าง (top-down) โดยเริ่มที่ส่วน < HEAD >...< /HEAD > ก่อนจากนั้นจึงทำงานในส่วน < BODY >...< /BODY > เป็นลำดับต่อมา การทำงานของ JavaScript ดูไม่แตกต่างไปจาก HTML เท่าไรนัก แต่ HTML จะวางเลย์เอาต์โครงสร้างของอ็อบเจกต์ภายใน และส่วนเชื่อมโยงกับเว็บเพจเท่านั้น ในขณะที่ JavaScript สามารถเพิ่มเติมส่วนของการเขียนโปรแกรมและล่อจิกเข้าไป

```
< FORM NAME = "statform" >  
< INPUT type="text" name="username" size = 10 >  
< INPUT type="text" name="userage" size = 5 >  
< /FORM >
```

สมาชิกในที่นี้คือ INPUT 2 ชุดในแบบฟอร์ม statform ทำหน้าที่สะท้อนไปยังอ็อบเจกต์ document.statform.username และ document.statform.userage จากการอ้างอิงโดย

JavaScript ทำให้เราสามารถนำอ็อบเจ็กต์นี้มาใช้งานได้ทันทีที่แบบฟอร์มนี้ถูก กำหนดขึ้นมา แต่ไม่สามารถใช้อ็อบเจ็กต์นี้ก่อนที่แบบฟอร์ม statform จะถูกกำหนดขึ้นมาได้ ตัวอย่างต่อไปนี้จะแสดงถึงค่าต่าง ๆ ของอ็อบเจ็กต์ในสคริปต์ที่อ้างถึงแบบฟอร์ม

ตัวอย่าง

```
< SCRIPT >

document.write(document.statform.username.value)

document.write(document.statform.userage.value)

< /SCRIPT >
```

2.3.4 การเรียกใช้ Script

โดยปกติต้องเรียกผ่านเหตุการณ์ต่างๆ ที่เกิดขึ้นกับ Element ของ HTML เช่นอีเวนต์ onclick ของปุ่มในฟอร์ม เป็นต้น นอกจากนั้นหากต้องการให้มีการเรียกใช้โดยตรงสามารถเรียกชื่อฟังก์ชันตรงๆ ผ่านการแทรกแท็ก <script> </script> ในตัว body ได้

```
<head>
  <script>
    //การประกาศฟังก์ชัน
    function func1(){ .... }
    function funcscript(){ .... }
  </script>
</head>
<body>
  // การเรียกใช้ func1() ผ่านอีเวนต์ onclick
  <input type="button" value="Click" onclick="func1()"/>
  // การเรียกใช้ func2() โดยตรง
  <script type="text/javascript"> func2(); </script>
</body>
```

2.3.5 รูปแบบการเขียน มีกฎดังนี้

1. ชื่อตัวแปรที่ใช้ห้ามมีสัญลักษณ์พิเศษหรือช่องว่าง
2. การใช้ตัวอักษรในชื่อตัวแปรตัวพิมพ์เล็กกับตัวพิมพ์ใหญ่เป็นตัวแปรคนละตัวกัน (Case Sensitive) เช่น `firstname` กับ `firstName` ถือเป็นคนละตัวกัน
3. เนื่องจาก Java Script เป็นภาษาที่ไม่มีตัวระบุชนิดของข้อมูล (Un-Type) ดังนั้นชนิดของข้อมูลจะขึ้นอยู่กับค่าที่เรากำหนดให้ในฝั่งขวามือของเครื่องหมาย =

เช่น `var x=10; //ตอนนี้ x เป็นข้อมูลชนิดตัวเลข (integer)`
`var x=1.321; //ตอนนี้ x เป็นข้อมูลชนิดทศนิยม (float)`
`var x='A', y="b"; //ตอนนี้ x และ y เป็นข้อมูลชนิดตัวอักษร (String)`

4. ตัวแปรที่ประกาศไว้โดยไม่มีการกำหนดค่า จะมีค่าปริยายเป็น `undefined` เมื่อจบคำสั่งการทำงาน ต้องปิดด้วยเครื่องหมาย ; (Semi Colon) เสมอ
5. การคอมเมนต์ที่ใช้เหมือน c กับ Java คือ `//` สำหรับบรรทัดเดียว `/* ... */` สำหรับหลายบรรทัด

2.3.6 สัญลักษณ์การกระทำทางคณิตศาสตร์และอื่นๆ

`+, -, *, /, %` บวก, ลบ, คูณ,หาร,หารเอาเศษ
`>, <, >=, <=` มากกว่า, น้อยกว่า, มากกว่าเท่ากับ, น้อยกว่าเท่ากับ
`==, ===` เท่ากันแบบไม่สนชนิด, เท่ากันแบบดูชนิด เช่น `1 == "1"` จะได้ `true` แต่ถ้า `1 === "1"` จะได้ `false`
`!=, !==` ไม่เท่ากันแบบไม่สนชนิด, ไม่เท่ากันแบบดูชนิด
`&&, ||` การเชื่อมสมการเข้าด้วยกัน แบบ AND และ OR
`++, --` เพิ่มทีละ 1 ลดทีละ 1

2.3.7 ตัวแปรและ Syntax

การประกาศตัวแปรใน Java Script มี 3 รูปแบบแบ่งตามขอบเขต (Scope) การใช้งานคือ ตัวแปรที่ใช้ร่วมกัน (Global Variable) จะประกาศด้วยประโยค `var` และวางอยู่นอกฟังก์ชัน ตัวแปรที่ประกาศแบบนี้จะใช้ได้ตลอดทั้งเอกสาร ตัวแปรที่ใช้เฉพาะในฟังก์ชัน (Local Variable) จะใช้ `var` หรือไม่ก็ได้ โดยตัวแปรนี้จะใช้งานได้ตั้งแต่บรรทัดที่ตัวแปรถูกประกาศ จนถึงปิด } โดยจะใช้ได้เฉพาะ

ในฟังก์ชันนั้นๆ เท่านั้น ตัวแปรที่เป็นพารามิเตอร์ของฟังก์ชัน (Parameter Variable) ขอบเขตสามารถอ้างถึงได้ตั้งแต่เปิดปีกกา { ของฟังก์ชัน จนถึงปิด }

```
//global_var เป็นตัวแปรชนิด global สามารถใช้ได้ทั่วทั้งเอกสาร

var global_var=10;

// a และ b เป็น function parameter สามารถใช้ในฟังก์ชันเท่านั้น

function funcX(a , b){

    // resultX เป็น local variable ใช้ได้เฉพาะในฟังก์ชัน funcX เท่านั้น

    var resultX = (a + b) * global_var;

    return resultY;

}

function funcY(c , d){

    var resultY= (c * d) * global_var;

    return resultY;

}
```

2.3.8 เงื่อนไข และการวนรอบ

if / else, else if การตรวจสอบด้วยสมการที่ให้ผล true, false

switch การตรวจสอบแบบหลายเงื่อนไขพร้อมๆกัน

for loop การวนรอบแบบรู้จุดสิ้นสุด มักใช้กับ array

while, do/while loop ใช้วนรอบแบบไม่จำเป็นต้องรู้จุดสิ้นสุด เช่นการวนโดยผู้ใช้ตัดสินใจเอง

2.3.9 ฟังก์ชันและตัวแปรของภาษา

การใช้งานออบเจ็กต์ (Object) และอาร์เรย์ (Array)

Math ฟังก์ชัน: abs, acos, asin, atan, atan2, ceil, cos, exp, floor, log, max, min, pow, random, round, sin, sqrt, tan

ค่าคงที่: E, LN10, LN2, LOG10E, PI, SQRT1_2, SQRT2

String ฟังก์ชัน: charAt, indexOf, lastIndexOf, substring, toLowerCase, toUpperCase

RegExp ฟังก์ชัน: match, replace, search, split

2.4 Bootstrap

เป็น Front-end Framework ที่ประกอบด้วยโครงสร้าง CSS , HTML และ JavaScript ที่ช่วยให้เราสามารถสร้างหน้าจอ User Interface ได้ง่าย สวยงาม และรวดเร็ว ลดเวลาในการออกแบบ Design หน้าจอ layout หรือรายการ Element อื่นๆ ที่เกี่ยวข้องกับหน้า Form ทั้งหมด ก็สามารถใช้ Bootstrap เข้ามาจัดการได้ทั้งหมด สำหรับ Bootstrap มีทั้ง CSS Component และ JavaScript Plugin ที่ทำงานร่วมกับ jQuery ที่สามารถเรียกใช้งานได้มากมาย และ Bootstrap มีการแสดงผลในรูปแบบของ Responsive ซึ่งจะแสดงสภาพแวดล้อมที่เหมาะสมกับอุปกรณ์ที่ทำการเปิดดูในขณะนั้น เช่น PC Desktop, Tablets, Mobile หรืออุปกรณ์อื่นๆ ที่เกี่ยวข้อง ซึ่งทำให้เรานั้นออกแบบเว็บและเขียนคำสั่งต่างๆ ก็สามารถที่จะรองรับอุปกรณ์ทั้งหมดได้เลย Bootstrap ถูกพัฒนาขึ้นโดย Mark Otto และ Jacob Thornton ทีมพัฒนาของ Twitter Inc. ก่อนหน้านี้ใช้ชื่อว่า Twitter Blueprint และเปิดให้นักพัฒนาสามารถนำไปใช้งานพัฒนาเว็บไซต์ได้แบบฟรี (Open Source) ในชื่อว่า Bootstrap Framework ปัจจุบันมีการพัฒนาขึ้นด้วยกลุ่มนักพัฒนาจากทุกแห่งในโลก มีการอัปเดตแก้ไข bug อยู่ตลอดเวลา เพื่อรองรับการทำงานได้อย่างไม่มีปัญหาพร้อมๆ กับการพัฒนาเทคโนโลยีบนเว็บเบราว์เซอร์

2.4.1 โครงสร้างของ Bootstrap Framework

- Scaffolding grid system จำนวน 12 คอลัมน์ สามารถเลือกใช้ได้ทั้งแบบ fixed และแบบ fluid เป็นโครงสร้างของ Layout ที่จะแสดงผลในหน้าจอ ซึ่งจำนวน Column นี้จะแสดงผลตามความกว้างของแต่ละอุปกรณ์ที่เรียกใช้งาน

- Base CSS style sheets สำหรับ html elements พื้นฐาน เช่น typography, tables, forms และ images เป็น Stylesheet พื้นฐานที่เราสามารถเรียกใช้งานได้เลย เช่น Button ที่อยู่ในรูปแบบของ สีต่างๆ การแสดงรูปภาพ ตาราง และอื่นๆ

- Components style sheets เป็นโครงสร้างพื้นฐานของ Bootstrap ที่ไว้จัดการเมนูและเนวิเกชั่นซึ่งจะแปรผันกับขนาดของหน้าจอ ของอุปกรณ์ที่เรียกใช้งาน เช่น navigation, breadcrumbs และ pagination

- JavaScript jQuery plugins ต่างๆ ไม่ว่าจะเป็น modal, carousel หรือ tooltip ช่วยในการสร้าง Popup, Dialog, Tooltip ต่างๆ ซึ่งเรียกใช้งานได้ง่าย

2.5 DataTables

plug-in JQuery Javascript สำหรับจัดการข้อมูล Html Table โดยจะมีการจัดรูปแบบข้อมูลใน Table ของเรา ให้สามารถค้นหา(Search), เรียงลำดับ(Sort), แบ่งหน้า (Pagination) ได้ และสามารถปรับแต่ง Options ต่าง ๆ ได้อีกด้วย

ตัวอย่างตาราง DataTables

Name	Position	Office	Age	Start date	Salary
Airi Satou	Accountant	Tokyo	33	2008/11/28	\$162,700
Angelica Ramos	Chief Executive Officer (CEO)	London	47	2009/10/09	\$1,200,000
Ashton Cox	Junior Technical Author	San Francisco	66	2009/01/12	\$86,000
Bradley Greer	Software Engineer	London	41	2012/10/13	\$132,000
Brenden Wagner	Software Engineer	San Francisco	28	2011/06/07	\$206,850
Brielle Williamson	Integration Specialist	New York	61	2012/12/02	\$372,000
Bruno Nash	Software Engineer	London	38	2011/05/03	\$163,500
Caesar Vance	Pre-Sales Support	New York	21	2011/12/12	\$106,450
Cara Stevens	Sales Assistant	New York	46	2011/12/06	\$145,600
Cedric Kelly	Senior Javascript Developer	Edinburgh	22	2012/03/29	\$433,060

Showing 1 to 10 of 57 entries

Previous 2 3 4 5 6 Next

รูปที่ 2.1 ตัวอย่าง DataTables

2.6 Sublime Text



รูปที่ 2.2 ตัวอย่าง Sublime Text

Sublime Text ถูกออกแบบมาให้ทำงานกับพวก code สำหรับการเขียนเว็บหรือเขียนโปรแกรมต่างๆ โดยเฉพาะดั่งนั้นถ้าการทำงานด้านพัฒนากับออกแบบเว็บไซต์หรือเว็บแอปพลิเคชัน (Web application) และต้องการเครื่องมือที่ไม่แพงแต่ช่วยให้ทำงานเสร็จเร็วขึ้น Sublime Text เปิดให้ดาวน์โหลดมาใช้ฟรี ซึ่งมีภาษาที่ Sublime Text รองรับการทำงานส่วนใหญ่ก็มี แพคเกจ (Package) หรือส่วนเสริมการทำงาน ช่วยอำนวยความสะดวกให้กับภาษาหลักของ Web Designer และ Web Developer ตัวอย่างเช่น HTML, CSS, Javascript, JQuery, Python, Ruby on Rails เป็นต้น

2.7 หลักการออกแบบเว็บไซต์

การออกแบบเว็บไซต์นั้นจะต้องนำข้อมูลต่างๆ ที่เก็บรวบรวมไว้เริ่มตั้งแต่วัตถุประสงค์ของเว็บไซต์ กลุ่มผู้ชมเว็บไซต์ เป้าหมาย จนกระทั่งเนื้อหาทั้งหมด เพื่อนำมาวิเคราะห์ จัดระบบ และสรุปหาแนวคิดของเว็บไซต์ให้มีการจัดการโครงสร้างและกำหนดรูปแบบของเว็บไซต์เพื่อที่จะนำเสนอสู่ผู้ชมให้ใช้งานได้ง่ายและเข้าถึงตัวผู้ใช้งานได้ง่ายมากยิ่งขึ้น การออกแบบเว็บไซต์มีองค์ประกอบ 2 ส่วน คือ

- การออกแบบโครงสร้างเว็บไซต์ (Site structure design)
- การออกแบบระบบเนวิเกชัน (Site navigation design)

2.7.1 การออกแบบโครงสร้างเว็บไซต์ (Site Structure Design)

โครงสร้างเว็บไซต์ (Site structure) เป็นแผนผังของการวางลำดับเนื้อหาของเว็บเพจหรือการจัดวางตำแหน่งเว็บเพจทั้งหมด ซึ่งจะทำให้รู้ว่าทั้งเว็บไซต์ประกอบไปด้วยเนื้อหาอะไรและมีเว็บเพจหน้าไหนที่เกี่ยวข้องเชื่อมโยงถึงกันบ้าง การออกแบบโครงสร้างเว็บไซต์จึงเป็นเรื่องสำคัญเปรียบได้กับการเขียนแปลนบ้านหลังหนึ่งก่อนที่จะลงมือสร้างบ้านหลังนั้นนั่นเอง การออกแบบโครงสร้างเว็บไซต์นี้จะทำให้มองเห็นหน้าตาของเว็บไซต์ เป็นรูปธรรมมากขึ้น สามารถออกแบบระบบเนวิเกชันได้เหมาะสม และมี

แนวทางการทำงานที่ชัดเจนในเรื่องของการวางแผนสำหรับขั้นตอนต่อไป นอกจากนี้โครงสร้างเว็บไซต์ที่ดีช่วยให้ผู้ใช้งานไม่สับสนและในการค้นหาข้อมูลที่ต้องการได้อย่างรวดเร็วโครงสร้างเว็บไซต์สามารถทำได้หลายแบบ แต่แนวคิดหลักๆที่นิยมใช้กันมีอยู่ 2 แบบ คือ จัดตามกลุ่มเนื้อหา (Content- Based Structure) และจัดตามกลุ่มผู้ชม (User-Based structure)

2.7.2 รูปแบบของโครงสร้างเว็บไซต์

สามารถวางรูปแบบโครงสร้างเว็บไซต์ได้หลายแบบตามความเหมาะสม เช่น แบบเรียงลำดับ (Sequence) เหมาะสำหรับ เว็บไซต์ที่มีจำนวนเว็บเพจไม่มากนัก หรือเว็บไซต์ที่มีการนำเสนอ ข้อมูลแบบทีละขั้นตอน หรือแบบระดับชั้น (Hierarchy) เหมาะสำหรับเว็บไซต์ ที่มีจำนวนเว็บเพจมากขึ้น เป็นรูปแบบที่พบได้ทั่วไป และสุดท้ายแบบผสม (Combination) เหมาะสำหรับเว็บไซต์ ที่ซับซ้อน เป็นการนำข้อดีของรูปแบบทั้ง 2 ข้างต้นมาผสมกัน

2.7.3 การออกแบบระบบเนวิเกชัน (Site Navigation Design) เป้าหมายของระบบนำทางหรือเนวิเกชันคือช่วยให้ผู้ใช้งานเข้าถึงข้อมูลที่ต้องการได้อย่างรวดเร็วและไม่หลงทางตั้งนั้น

องค์ประกอบของระบบนำทางจึงมี 2 ส่วนด้วยกันคือ

1. เครื่องนำทาง (Navigation Controls) คือ เครื่องมือสำหรับใช้ผู้ใช้งานเปิดไปยังเว็บต่างๆภายในเว็บไซต์ โดยแยก ได้เป็น

- เมนูหลัก เป็นเมนูสำหรับเปิดไปยังหัวข้อเนื้อหา หลักของเว็บไซต์ มักอยู่ในรูปของกลุ่มลิงก์ที่เป็นข้อความหรือ ภาพกราฟิก และควรมีปรากฏอยู่บนเว็บเพจทุกหน้า

- เมนูเฉพาะกลุ่ม เป็นเมนูที่เชื่อมโยงเว็บเพจ ปัจจุบันกับเว็บเพจอื่นภายในกลุ่มย่อยที่มีเนื้อหาเกี่ยวเนื่องเท่านั้น มักอยู่ในรูปของกลุ่มลิงก์ข้อความหรือกราฟิกเช่นกัน

- เครื่องมือเสริม สำหรับช่วยเสริมการทำงานของเมนูนี้ได้หลากหลายรูปแบบ เช่น การค้นหาข้อมูล (search box) เมนูแบบดริอปดาวน์ (drop-down menu) อิมเมจแมพ (image map) และแผนผังเว็บไซต์ (Site map) เป็นต้น

2. เครื่องบอกตำแหน่ง (Location Indicator) เป็นลิงก์ที่ใช้แสดงว่าขณะนี้ผู้ใช้งานกำลังอยู่ที่ตำแหน่งในเว็บไซต์ เครื่องบอกตำแหน่งนี้ได้หลายรูปแบบ เช่น ข้อความหรือภาพกราฟิกที่แสดงชื่อเว็บเพจหรือข้อความบ่งชี้และบ่อยครั้งที่เครื่องบอกตำแหน่งถูกรวมไว้กับตัวเมนูเลย โดยแสดงด้วยลิงก์หรือรูปแบบที่แตกต่างจากปุ่มเมนูอื่นๆ ลักษณะระบบเนวิเกชันที่ดีจะต้อง อยู่ในตำแหน่งที่เห็นได้ชัดและเข้าถึงง่าย เช่น ส่วนบน หรือด้านซ้ายของเว็บเพจ มีไอคอน (icon) หรือปุ่มกดที่ชัดเจนโดดเด่น

เข้าใจง่ายหรือมีข้อความกำกับชัดเจน ผู้ใช้งานสามารถเห็นได้กดคลิกใช้งานโดยไม่ต้องเสียเวลาศึกษามีความสม่ำเสมอ และเป็นระบบไม่ชวนให้สับสนหรือกลับไปกลับมา มีการตอบสนองกับผู้ใช้ใช้งานเมื่อใช้งาน เช่น เปลี่ยนสีเมื่อผู้ใช้งานใช้เมาส์หรือคลิกที่ปุ่มกดหรือไอคอนนั้นควรมีจำนวนรายการพอเหมาะไม่มากจนเกินไปและหลายทางเลือกให้ใช้ เช่น เมนูกราฟฟิก เมนูข้อความ ช่องค้นหาข้อมูล (search box) เมนูแบบดรอปดาวน์ (drop-down menu) และแผนผังเว็บไซต์ (site map) เป็นต้น ระบบเนวิเกชันควรมีลิงก์ให้คลิกกลับไปยังโฮมเพจได้เสมอเพื่อให้ผู้ใช้งานกลับไปเริ่มต้นไม่ให้เกิดกรณีที่หลงทางไม่รู้ว่าจะตัวเองอยู่ที่ตำแหน่งใดในเว็บไซต์นี้

2.7.4 ออกแบบขนาดของเว็บเพจ

การออกแบบเว็บเพจที่ดีนั้น ต้องคำนึงถึงกลุ่มผู้ใช้งาน เป้าหมายส่วนใหญ่จะใช้จอภาพที่มีความละเอียด (resolution) ที่ฟิกเซลและกำหนดว่าจะทำให้เว็บเพจของคุณแสดงผลได้ดีบนจอความละเอียดเท่าใดความละเอียดของจอภาพที่ใช้กันมากที่สุดในปัจจุบัน มีอยู่ 2 ระดับ คือ

- 1023 X 768 พิกเซล
- 1280 X 800 หรือ 1280 X 1024 พิกเซล

หากออกแบบเว็บเพจสำหรับจอ 1024 X 768 พิกเซล ผู้ใช้งานที่ใช้จอความละเอียดเท่ากันนี้ก็ยังสามารถชมเนื้อหาทั้งหมดได้พอดี เต็มจอภาพ แต่ถ้าผู้ใช้งานใช้จอความละเอียด 1280 X 1024 ขนาดของเว็บเพจและองค์ประกอบต่างๆจะเล็กลง และแสดงไม่เต็มจอภาพ ในทางกลับกัน ถ้าออกแบบเว็บเพจสำหรับจอ 1280 X 1024 พิกเซล ผู้ใช้งานที่ใช้จอความละเอียดเท่านี้ จะชมได้พอดีเต็มจอภาพ แต่สำหรับผู้ใช้งานที่ใช้จอความละเอียด 1024 X 768 หน้าเว็บเพจนี้จะล้นจอ โดยประเด็นสำคัญที่สุดอยู่ที่ความกว้างของเว็บเพจ ซึ่งไม่ควรออกแบบให้กว้างเกินจอภาพของผู้ใช้งานเพราะจะทำให้ต้องเลื่อนหน้าจอในแนวนอนเพื่อดูเนื้อหาที่ตกไป ซึ่งไม่สะดวกเป็นอย่างยิ่งส่วนทางความสูงนั้นไม่ค่อยเป็นปัญหา แต่ก็ควรให้เนื้อหาที่สำคัญ สามารถมองเห็นได้ทันทีโดยผู้ใช้งานไม่ต้องเลื่อนจอลง เนื่องจากปัจจุบันยังมีผู้ชมจำนวนหนึ่งที่ใช้จอความละเอียดเพียง 800 X 600 ดังนั้นจึงมีหลายเว็บไซต์ที่ออกแบบเว็บเพจกว้างไม่เกิน 800 พิกเซลเท่านั้น

2.7.5 ลำดับความสำคัญของเนื้อหา

การสร้างลำดับความสำคัญขององค์ประกอบต่างๆ บนเว็บเพจ จะช่วยเน้นให้ผู้ใช้งานมองเห็นว่าส่วนไหนสำคัญมากหรือน้อยได้อย่างรวดเร็ว ตัวอย่างเช่น ข้อความขนาดใหญ่แสดงว่าสำคัญกว่าข้อความขนาดเล็ก เป็นต้น ลักษณะองค์ประกอบที่ต้อง คำนึงถึงมีดังนี้

- ตำแหน่งและลำดับขององค์ประกอบเว็บเพจ

- สีและขนาดขององค์ประกอบ

-การใช้ภาพเคลื่อนไหว หรือข้อความที่ เคลื่อนไหวได้ ซึ่งจะดึงดูดสายตามากกว่า องค์ประกอบทั่วไป ให้กราฟฟิกให้พอดี การใช้กราฟฟิกบนเว็บเพจจะให้ผลทั้ง 2 ด้าน คือด้านหนึ่งช่วยให้เว็บเพจสวยงามและน่าสนใจแต่อีกด้านหนึ่งถ้าใส่ กราฟฟิกมากเกินไปอาจจะทำให้ ดูรกตาและเว็บเพจโหลดกราฟฟิกช้าต้องเสียเวลารอนาน ดังนั้นจึงควรจำกัดปริมาณของภาพกราฟฟิก ที่นำมาใช้ให้เหมาะสม

2.7.6 การเลือกใช้สี

การออกแบบเว็บเพจให้สวยงามนั้น ส่วนหนึ่งจะขึ้นอยู่กับเลือกใช้ชุดสีที่ผสมกลมกลืนกัน เช่น สี ของพื้นเว็บเพจ สีข้อความ และสีขององค์ประกอบอื่นๆ เช่น กราฟฟิก ปุ่มเมนูหรือลิงก์ และการใช้สี นั้นยังบ่งบอกบุคลิกของเว็บไซต์ด้วยว่าเนื้อหา ออกมาในแนวไหน

2.7.7 ความสำคัญกับส่วนบนของเว็บเพจ

เว็บเพจคือด้านบนสุดของหน้าเป็นส่วนที่สำคัญที่สุดส่วนหนึ่งในเว็บเพจเลยก็ว่าได้ ซึ่งผู้ใช้งานจะ สามารถสัมผัสได้ทันทีเมื่อเปิดเข้าไปที่หน้านั้น เว็บเพจส่วนใหญ่นิยม จัดวางองค์ประกอบดังนี้

- โลโก้และชื่อของเว็บไซต์
- ระบบนำทาง ลิงก์ที่สำคัญ และเครื่องมือเสริมสำหรับค้นหาข้อมูลภายในเว็บไซต์
- ป้ายแบนเนอร์โฆษณา หรือข้อความที่สำคัญ

2.7.8 ส่วนประกอบของหน้าเว็บเพจ

โดยทั่วไปหน้าเว็บเพจจะแบ่งออกเป็นส่วนๆหลักๆ ดังนี้

- ส่วนหัว (Page Header) อยู่ตอนบนสุดของหน้า เป็นบริเวณที่สำคัญที่สุด เนื่องจาก ผู้ใช้จะมองเห็นก่อนบริเวณอื่น ส่วนใหญ่นิยมใช้วางโลโก้ ชื่อเว็บไซต์ ป้ายโฆษณา ลิงก์สำหรับการ ติดต่อหรือลิงก์ที่สำคัญ และระบบทำงาน

- ส่วนของเนื้อหา (Page Body) ตอนกลางหน้า ใช้แสดงเนื้อหาของเว็บเพจนั้นซึ่ง อาจจะประกอบไปด้วยข้อความ ภาพกราฟฟิก ตารางข้อมูล และอื่นๆ บางครั้งเมนูหลักหรือเมนูเฉพาะ กลุ่มอาจมาอยู่ในส่วนนี้ก็ได โดยมักจะวางไว้ด้านซ้ายมือสุด เนื่องจากผู้ใช้จะมองเห็นได้ง่ายกว่า

- ส่วนท้าย (Page Footer) อยู่ด้านล่างสุดของหน้า ส่วนใหญ่จะนิยมใช้วางระบบนำ ทางแบบที่เป็นลิงก์ข้อความง่ายๆนอกจากนั้นก็อาจจะมีชื่อเจ้าของ ข้อความแสดงลิขสิทธิ์และอีเมล แอดเดรสของผู้ดูแลเว็บไซต์

- แถบข้าง (Side Bar) ในปัจจุบันนิยมออกแบบ ด้านข้างของหน้าเว็บเพจให้น่าสนใจ เพื่อใช้วางป้ายแบนเนอร์ หรือ ลิงก์แนะนำเกี่ยวกับการบริการของเว็บไซต์ เป็นต้น

2.7.9 การออกแบบเว็บเพจ (Page Design)

วิธีที่สะดวกที่สุดในการออกแบบเว็บเพจคือใช้โปรแกรมสร้างภาพกราฟิก เช่น Photoshop หรือ Firework วางเค้าโครงของหน้าและสร้างองค์ประกอบต่างๆขึ้นมาให้ครบสมบูรณ์ในภาพเดียวไม่ว่าจะเป็นโลโก้ ชื่อเว็บไซต์ ปุ่มเมนู ปุ่มไอคอน แถว สี ภาพเคลื่อนไหว และอื่นๆ เนื่องจากโปรแกรมเหล่านี้มีเครื่องมือพร้อมสำหรับงานดังกล่าวและในขั้นตอนสุดท้ายควรสามารถที่จะบันทึกองค์ประกอบทั้งหมดแยกเป็นไฟล์กราฟิกย่อยๆพร้อมกับไฟล์ HTML เพื่อนำไปใช้เป็นคั่นแบบในโปรแกรมสร้างเว็บเพจได้ทันที

2.7.10 แนวคิดในการออกแบบเว็บเพจ

- ศึกษาจากเว็บไซต์อื่นๆ เป็นวิธีที่ง่ายที่สุด โดยสามารถศึกษามาจากเว็บไซต์ๆ ที่มีอยู่แล้วและนำรูปแบบที่เหมาะสม นำมาประยุกต์ใช้แต่ไม่ใช้การลอกเลียนแบบออกมาทั้งหมดซึ่งจะเป็นการละเมิดลิขสิทธิ์ได้ ไม่ว่าจะเป็นในเรื่องแนวคิดการออกแบบ การใช้โทนสี ข้อความ กราฟฟิก การจัดองค์ประกอบศิลป์ และเทคนิคที่ใช้สร้างเว็บ

- ประยุกต์รูปแบบจากสิ่งพิมพ์ สิ่งพิมพ์ เช่น นิตยสาร แผ่นพับ โบรชัวร์ โปสเตอร์ และหนังสือบางเล่มที่มีรูปแบบที่สวยงามซึ่งอาจจะนำมาใช้เป็นแนวทางการออกแบบเว็บเพจได้ในด้านของข้อความและกราฟฟิก เว็บเพจที่มีข้อความมากๆอาจทำให้ผู้ใช้งานรู้สึกกลัวและเบื่อหน่าย เพราะไม่ยอมอ่านเนื้อหายาวๆ ในทางตรงกันข้ามถ้าเว็บเพจมีแต่รูปภาพอย่างเดียว ผู้ใช้งานก็อาจสงสัยเมื่อรูปภาพนั้นไม่สื่อไปทางใดทางหนึ่งผู้ใช้งานจึงไม่เข้าใจว่าต้องการนำเสนออะไร ดังนั้นควรออกแบบเว็บเพจในแนวทางกลางๆ ไม่มีองค์ประกอบอย่างใดอย่างหนึ่งมากหรือน้อยจนเกินไป แต่ดูให้เหมาะสมกับเนื้อหาและกลุ่มผู้ใช้งานเว็บเพจที่จัดแบบเรียบง่ายแต่ดูดีจะสร้างความน่าสนใจและน่าเชื่อถือกับผู้ใช้งาน นอกจากนี้การวางตำแหน่งของเนื้อหาต่างๆ อย่างมีระเบียบเป็นสัดส่วนก็ทำให้ผู้ใช้งานหาข้อมูลที่ ต้องการได้อย่างรวดเร็ว

บทที่ 3

ขั้นตอนการดำเนินงาน

วิธีการดำเนินสามารถแบ่งออกได้เป็น 4 ส่วนด้วยกันคือ

- วางแผนจัดทำเว็บไซต์
- กำหนดโครงสร้างเว็บไซต์
- กำหนดการเชื่อมโยงระหว่างเว็บเพจ
- ออกแบบหน้าเว็บเพจแต่ละหน้า

3.1 วางแผนจัดทำเว็บไซต์

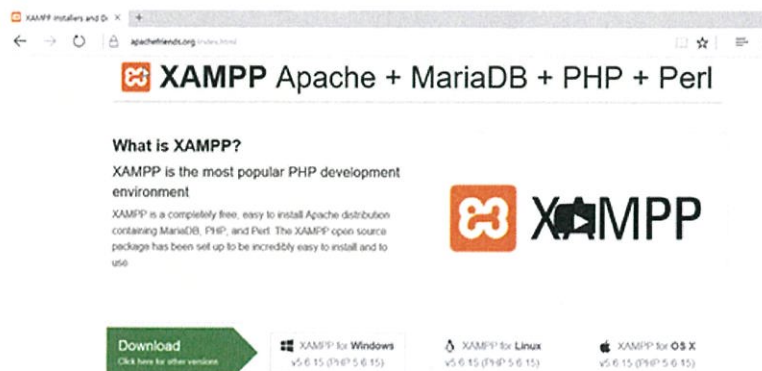
3.1.1 เก็บความต้องการของผู้ใช้งานในทีม ซึ่งประกอบด้วย

- Admin Page
 - Movement
 - Serial
 - Member
- Implementer Page
- SCoA&D Page
- Network Monitor Page

3.1.2 ติดตั้งโปรแกรมที่ต้องใช้

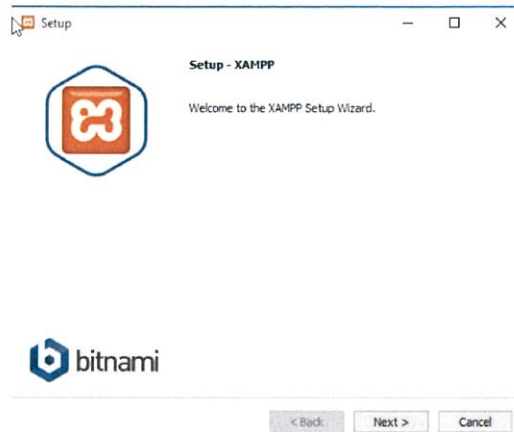
3.1.2.1 การติดตั้ง Xampp

1. ทำการดาวน์โหลด XAMPP สำหรับ Windows [1]



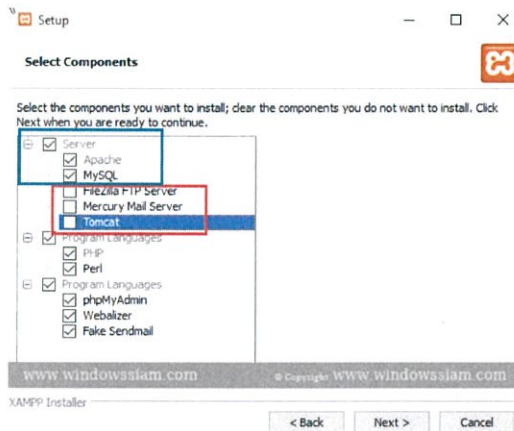
รูปที่ 3.1 เว็บไซต์ดาวน์โหลด XAMPP

2. ทำการติดตั้ง คคลิก Next



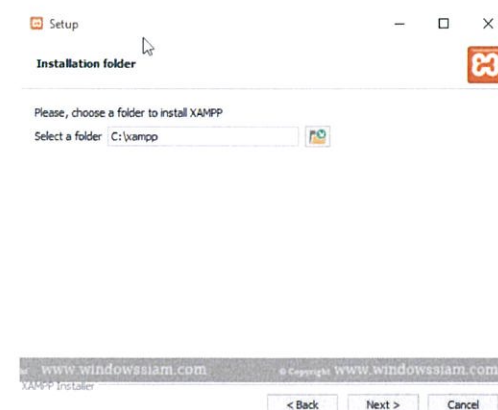
รูปที่ 3.2 การติดตั้ง XAMPP

3. ทำการเลือก Components ที่เราต้องการ



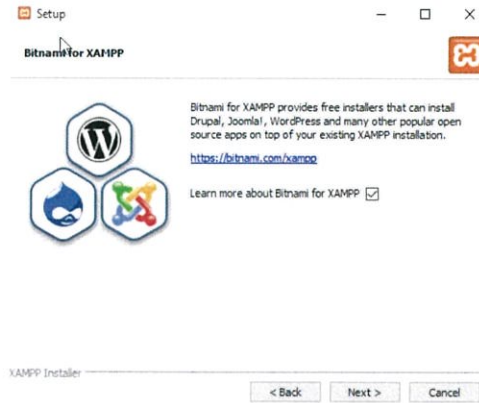
รูปที่ 3.3 การเลือก Components ของ XAMPP

4. เลือกตำแหน่งในการติดตั้ง XAMPP



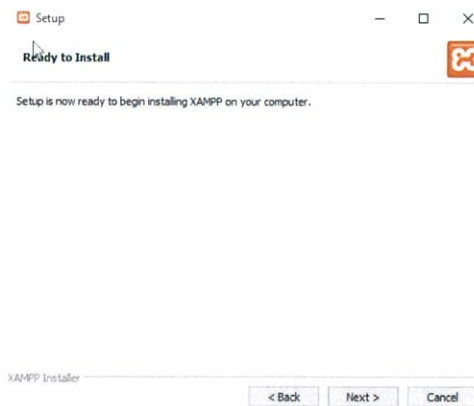
รูปที่ 3.4 การเลือกตำแหน่งของ XAMPP

5. คลิก Next



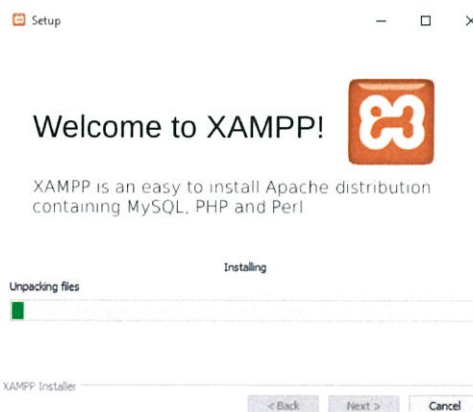
รูปที่ 3.5 Bitnami ของ XAMPP

6. คลิก Next



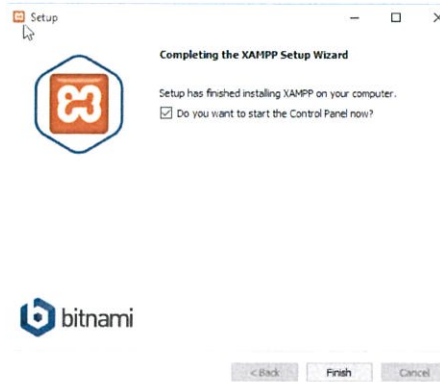
รูปที่ 3.6 การเตรียมการติดตั้ง XAMPP

7. รอทำการติดตั้ง



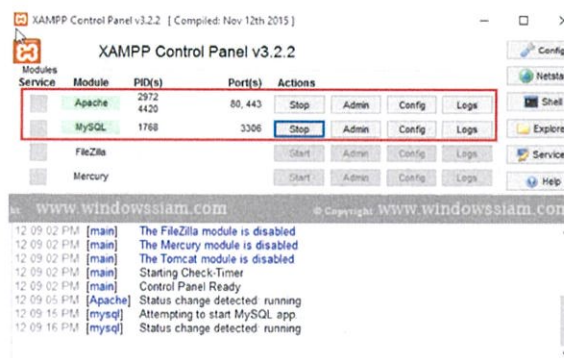
รูปที่ 3.7 รอการติดตั้ง XAMPP

8. ทำการติดตั้งเรียบร้อย และทำการ Restart Computer 1 ครั้ง



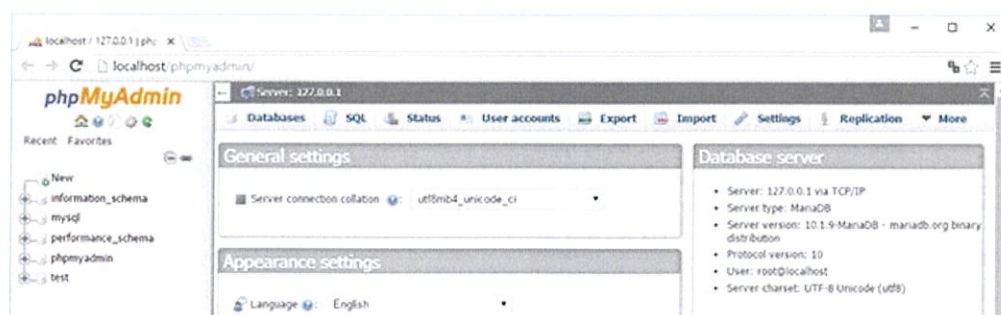
รูปที่ 3.8 การติดตั้ง XAMPP เรียบร้อย

9. จากนั้นเข้า Control Panel ของ XAMPP และทำการคลิก Start Apache และ Mysql ตามลำดับ



รูปที่ 3.9 XAMPP Control Panel

10. จากนั้นลองเข้า Phpmysql และ localhost ดู <http://localhost/phpmyadmin>



รูปที่ 3.10 หน้าเว็บ Localhost ของ XAMPP

3.1.2.2 การติดตั้ง Bootstrap

1. ดาวน์โหลด Bootstrap จากเว็บไซต์ <http://getbootstrap.com/> เพื่อติดตั้ง แล้วเลือก "Download Bootstrap"
2. แดก zip ไฟล์ จะได้โฟลเดอร์ "bootstrap-3.2.0-dist" ซึ่งภายในจะประกอบด้วย 3 โฟลเดอร์ย่อย คือ โฟลเดอร์ "css" โฟลเดอร์ "fonts" และโฟลเดอร์ "js"
3. ให้ copy 3 โฟลเดอร์ ข้างต้นไปยังโฟลเดอร์ของโปรเจ็ค เปิด text editor เพื่อสร้างไฟล์ index.html ขึ้นมา และให้เราพิมพ์โค้ดลงไปดังนี้

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap </title>
</head>
<body>
</body>
</html>
```

4. ไฟล์ index.html จะต้อง include ไฟล์ "bootstrap.css" ซึ่งอยู่ในโฟลเดอร์ css โดยใส่ภายในแท็ก <head></head> แต่อยู่ก่อนแท็ก <tittle>

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

5. ดาวน์โหลด JQuery ซึ่งดาวน์โหลดไปเก็บไว้ในโฟลเดอร์ "js"
6. ที่ไฟล์ index.html เราจะต้อง include ไฟล์ JQuery.js และ Bootstrap.js โดยต้องอยู่ภายในแท็ก

```
<body>
<script src="js/jquery-1.11.1.js"></script>
<script src="js/bootstrap.js"></script>
</body>
```

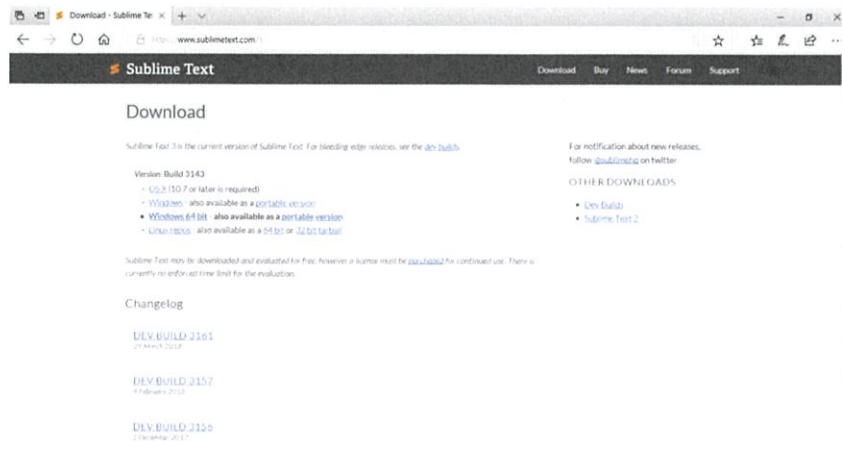
include ไฟล์ JQuery.js และ Bootstrap.js ไว้ในแท็ก <body></body>

7. ต้องกำหนดภาษาในแท็ก <meta> เพื่อให้เว็บเพจของเราสามารถแสดงผลได้อย่างถูกต้อง

```
<meta charset="UTF-8" />
```

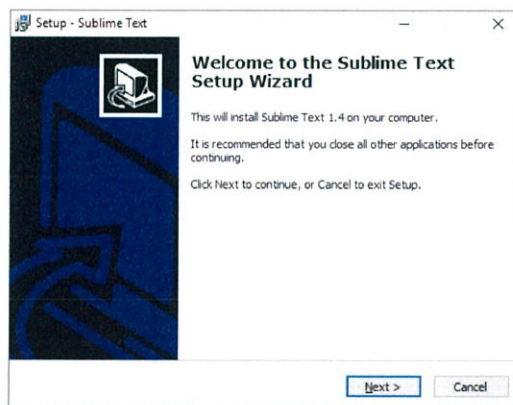
3.1.2.3 การติดตั้ง Sublime Text

1. ดาวน์โหลด Sublime Text ในเว็บไซต์ <https://www.sublimetext.com/> [3]



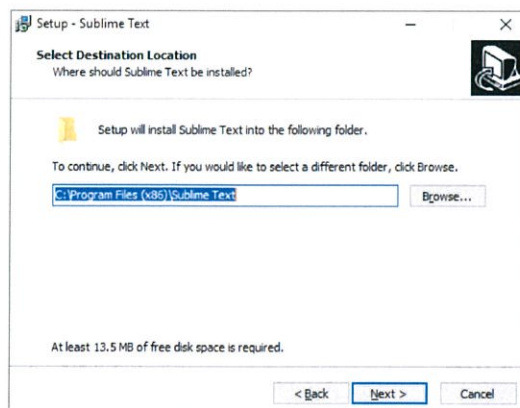
รูปที่ 3.11 เว็บไซต์ Sublime Text

2. คลิกไฟล์ Sublime Text Setup.exe



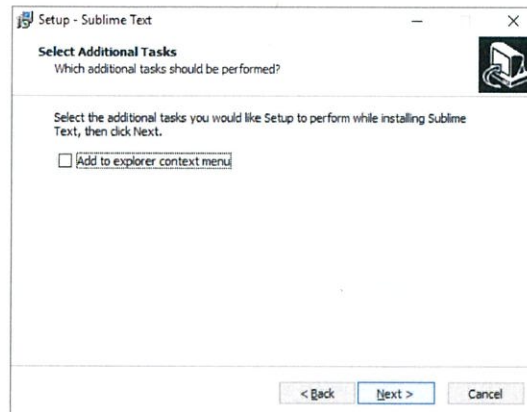
รูปที่ 3.12 Sublime Text Setup

3. เลือกตำแหน่งในการติดตั้ง



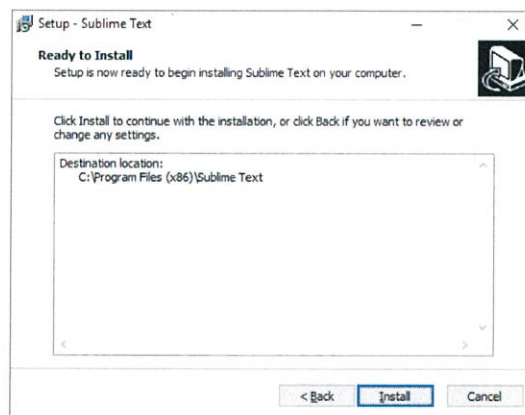
รูปที่ 3.13 Location Sublime Text

5. คลิก Next



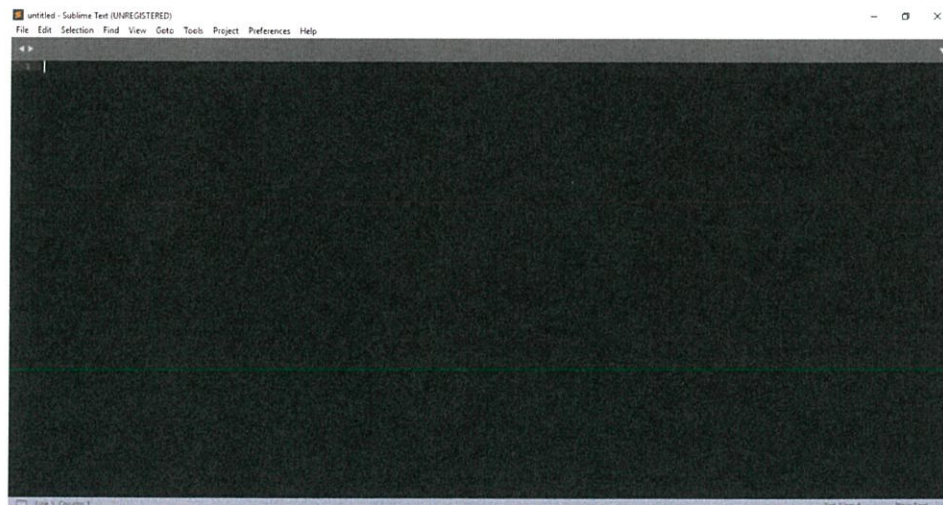
รูปที่ 3.14 Additional Task

6. คลิก Install



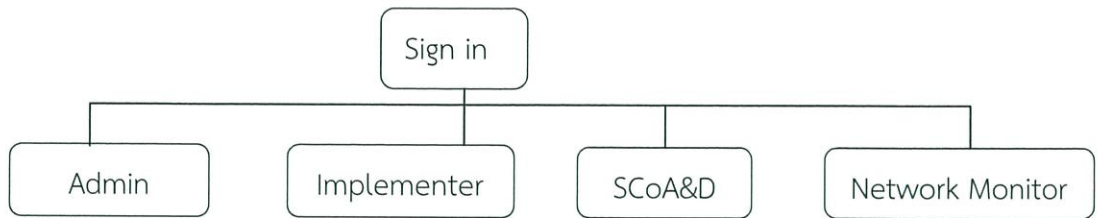
รูปที่ 3.15 เตรียมติดตั้ง Sublime Text

8. ทำการติดตั้งเรียบร้อยแล้ว และทำการ Restart Computer 1 ครั้ง

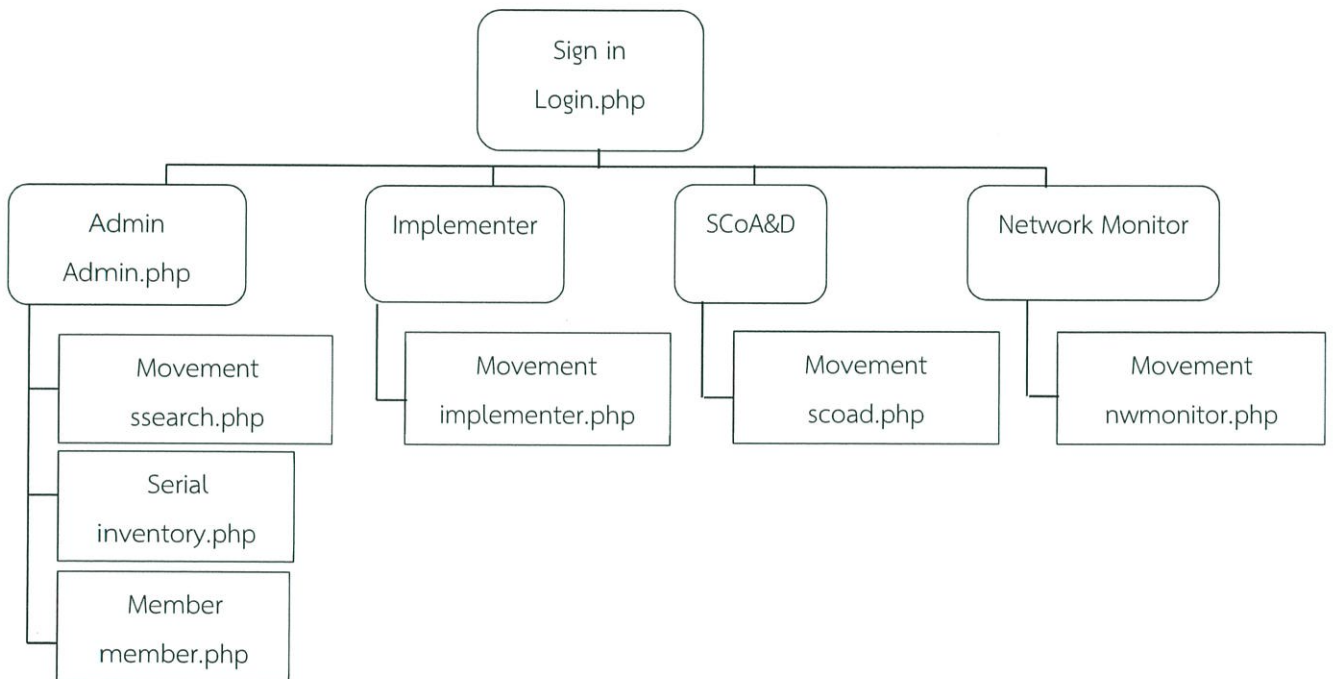


รูปที่ 3.16 Console Sublime Text

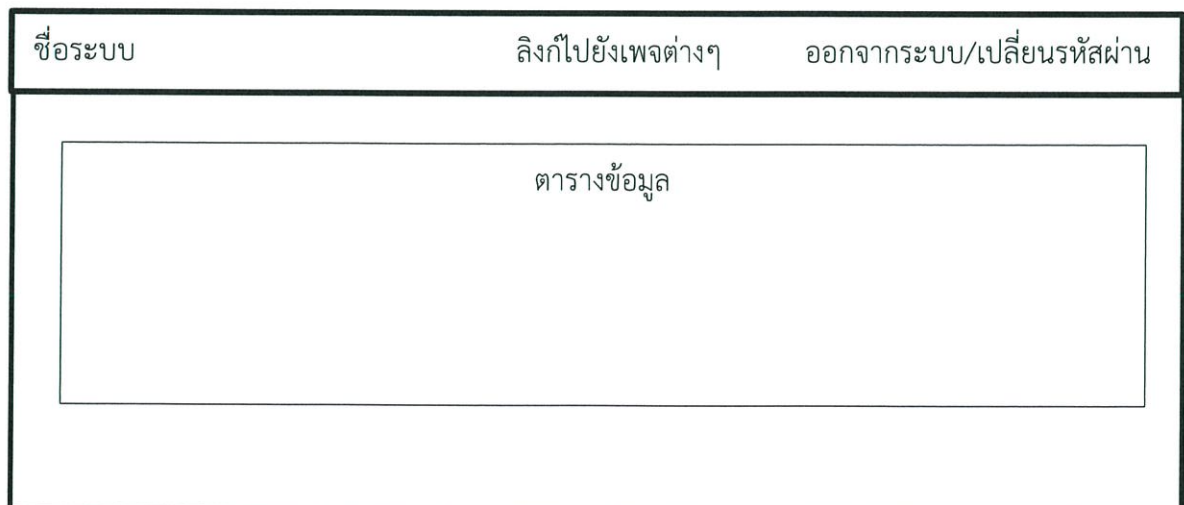
3.2 กำหนดโครงสร้างเว็บไซต์



3.3 กำหนดการเชื่อมโยงระหว่างเว็บเพจ



3.4 ออกแบบหน้าเว็บแต่ละหน้า



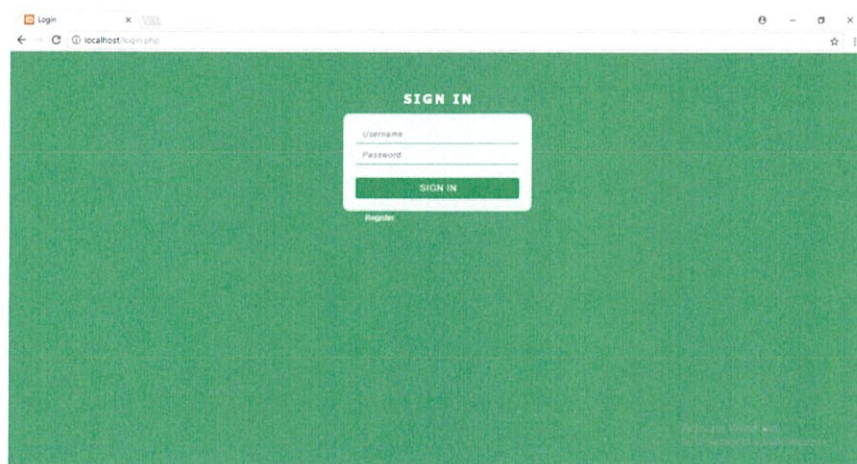
บทที่ 4

ผลการดำเนินการ

เว็บไซต์ข้อมูลสินค้าคงคลังส่วนหน้าเว็บ เพื่อความสะดวกในการติดตามอุปกรณ์ที่นำไปใช้งาน เซ็คสินค้าคงคลังได้ถูกต้อง มีการทำงานอย่างเป็นระบบโดยการนำอุปกรณ์เข้าและออกผ่านเน็ตเวิร์ค admin ผู้ดูแลระบบเท่านั้น โดยประกอบด้วยลักษณะของการใช้งาน 4 ส่วน คือ admin, implementer, SCoA&D และNetwork monitor มีรายละเอียดดังนี้

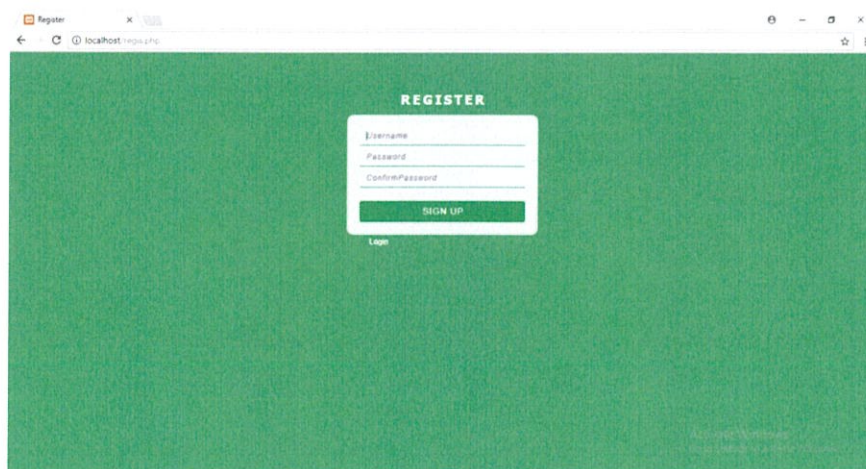
4.1 การเข้าสู่ระบบ

- Sign in



รูปที่ 4.1 Sign in Page

- Register การลงทะเบียนเพื่อเข้าสู่ระบบ

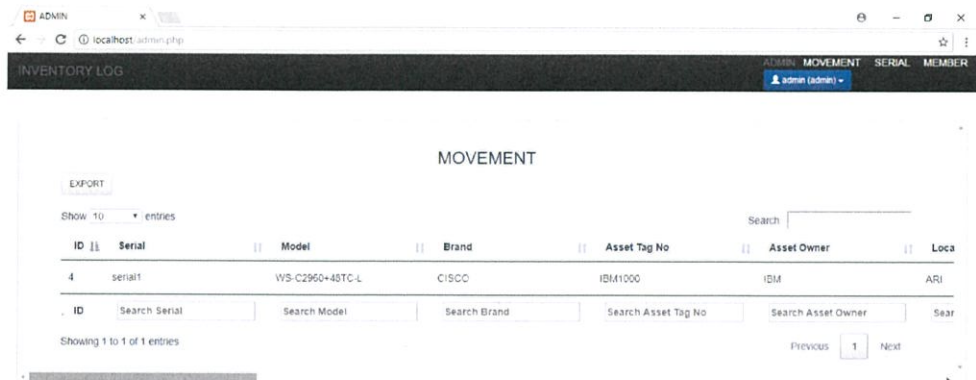


รูปที่ 4.2 Register Page

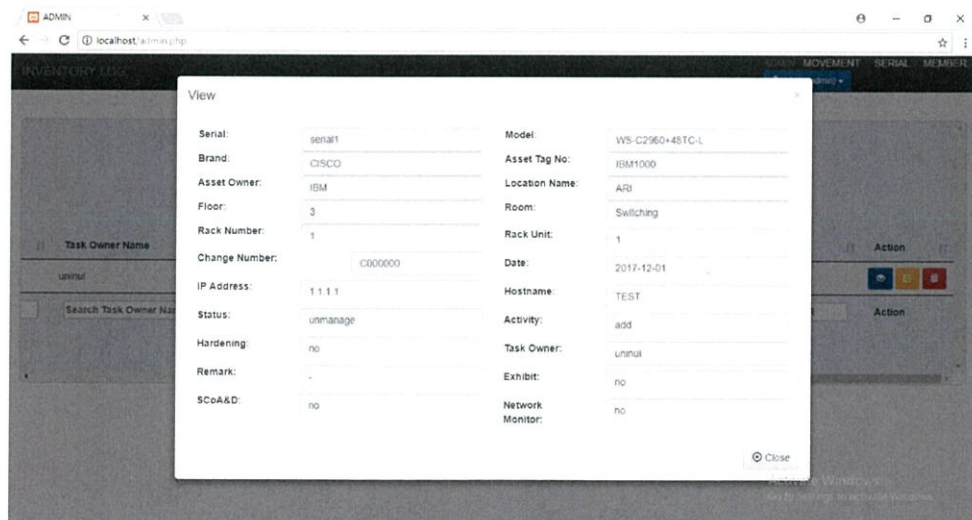
4.2 Admin Page

- Admin

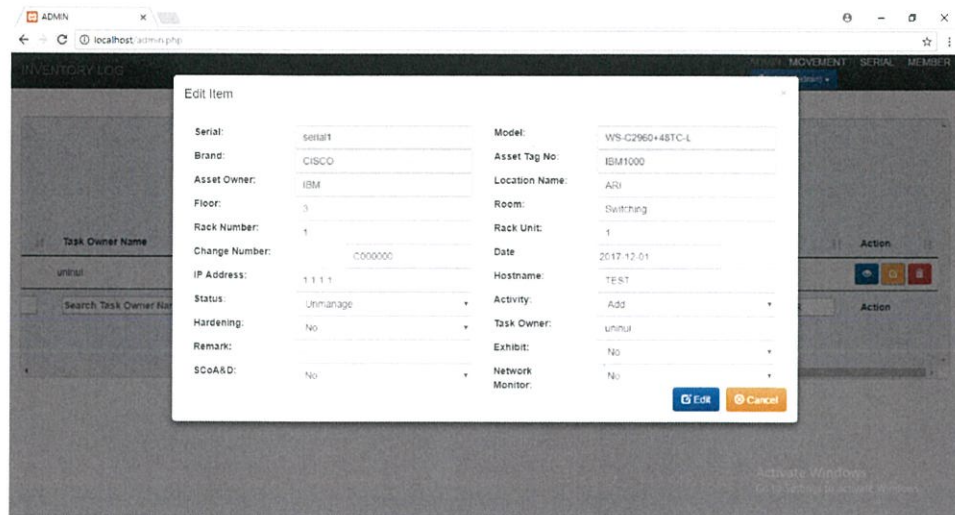
เป็นหน้าที่ admin ได้เพิ่มข้อมูลของอุปกรณ์มาไว้ เพื่อให้ implementer ได้การแก้ไขข้อมูลต่างๆ ในการนำเอาอุปกรณ์เข้าและออก ซึ่ง admin สามารถแสดงข้อมูลทั้งหมด แก้ไขข้อมูลได้ และลบข้อมูลออกไปได้



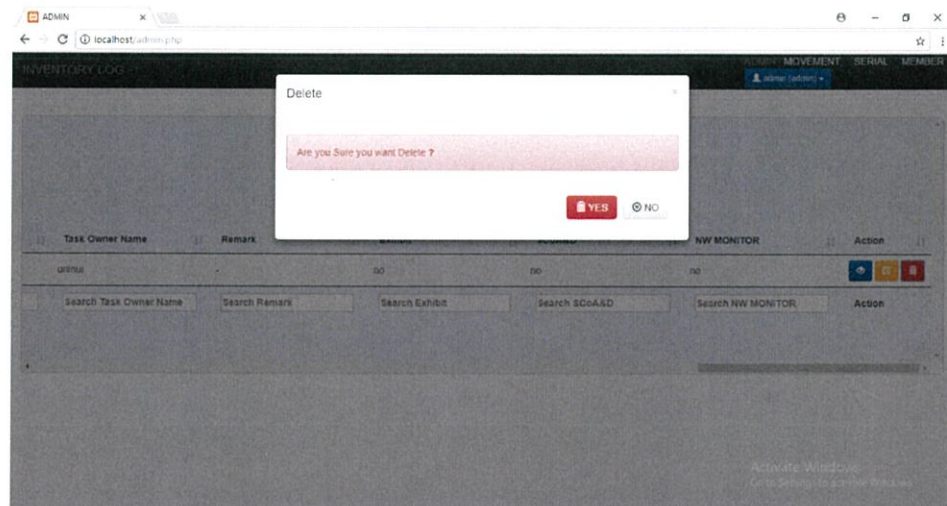
รูปที่ 4.3 ตาราง movement ของ admin



รูปที่ 4.4 แสดงข้อมูล ตาราง movement



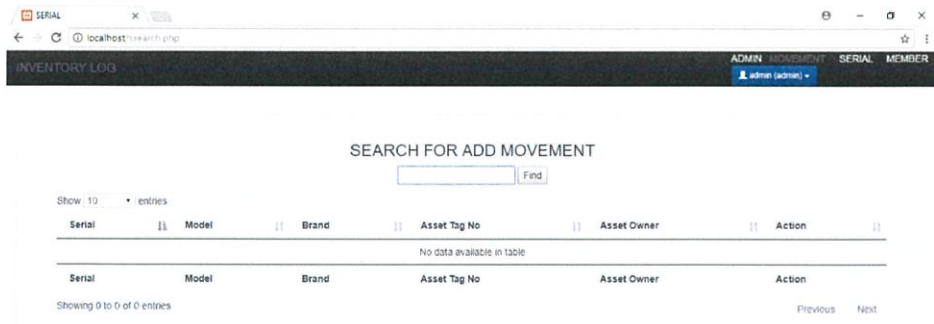
รูปที่ 4.5 การแก้ไขข้อมูลตาราง movement



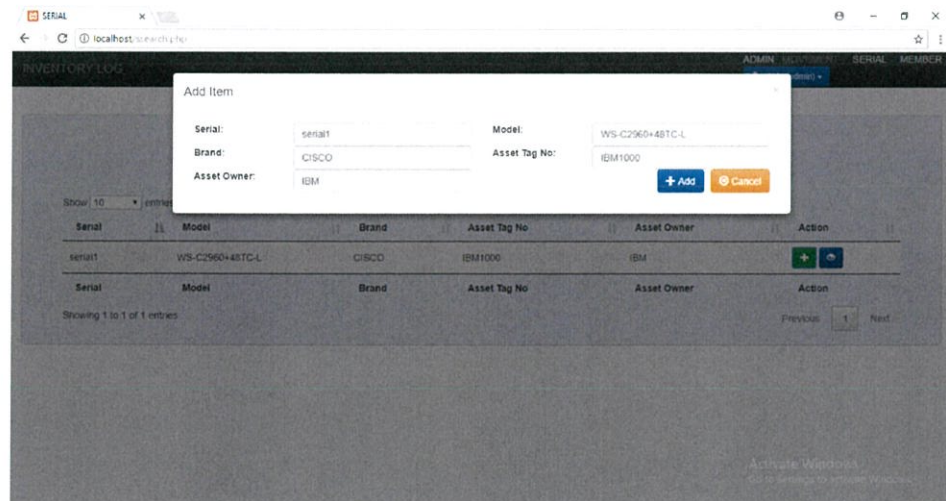
รูปที่ 4.6 การลบข้อมูลในตาราง movement

- Search

หน้า search เป็นการค้นหาข้อมูลอุปกรณ์ในการเพิ่มข้อมูล สามารถแสดงข้อมูลทั้งหมดได้ และเพิ่มข้อมูลเข้าไปในหน้า movement



รูปที่ 4.7 การค้นหาข้อมูล

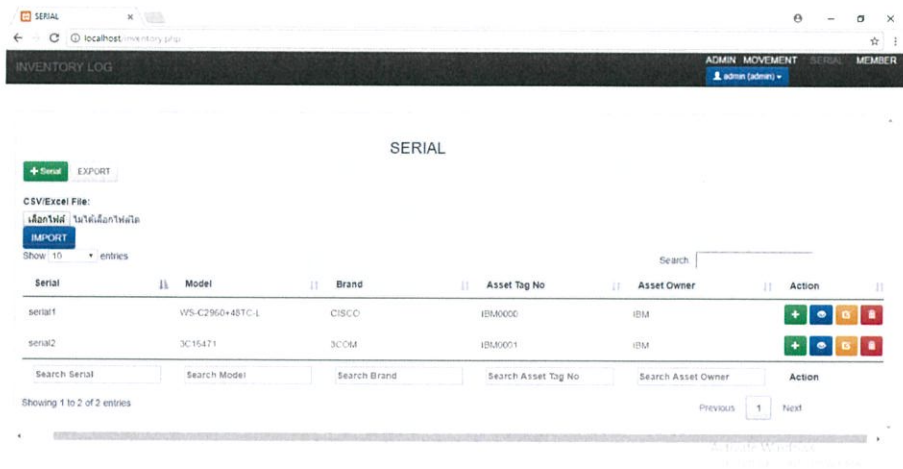


รูปที่ 4.8 การเพิ่มข้อมูลไปยัง movement

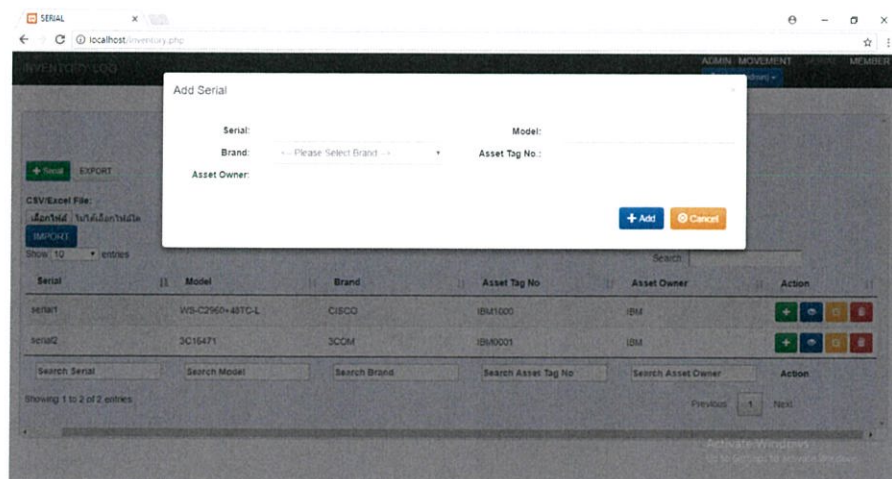
- Serial

เป็นหน้าแสดงข้อมูลอุปกรณ์ทั้งหมด มีรายละเอียดดังนี้

- เพิ่มข้อมูลอุปกรณ์
- แสดงข้อมูลอุปกรณ์
- แก้ไขข้อมูล
- ลบข้อมูล
- เพิ่มข้อมูลไปยังหน้า movement
- นำเข้าข้อมูลจาก csv ไฟล์
- นำข้อมูลออกเป็น excel ไฟล์



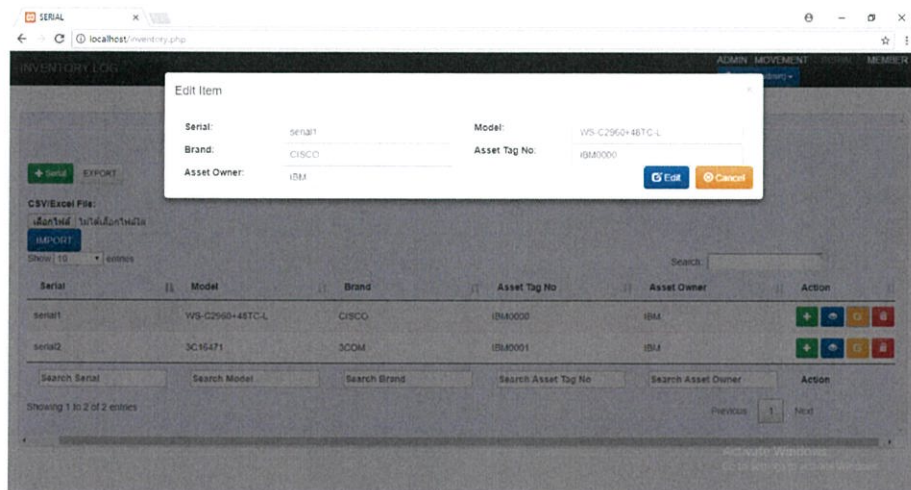
รูปที่ 4.9 Serial Page



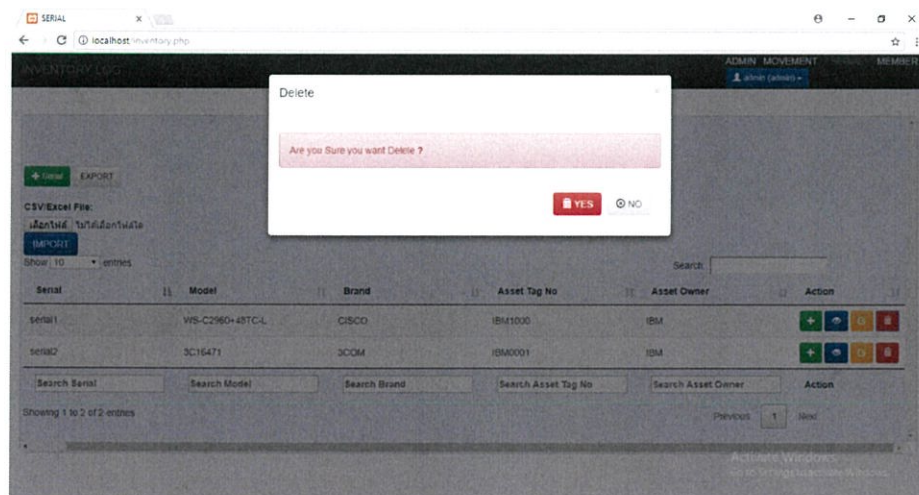
รูปที่ 4.10 การเพิ่มข้อมูลตาราง Serial



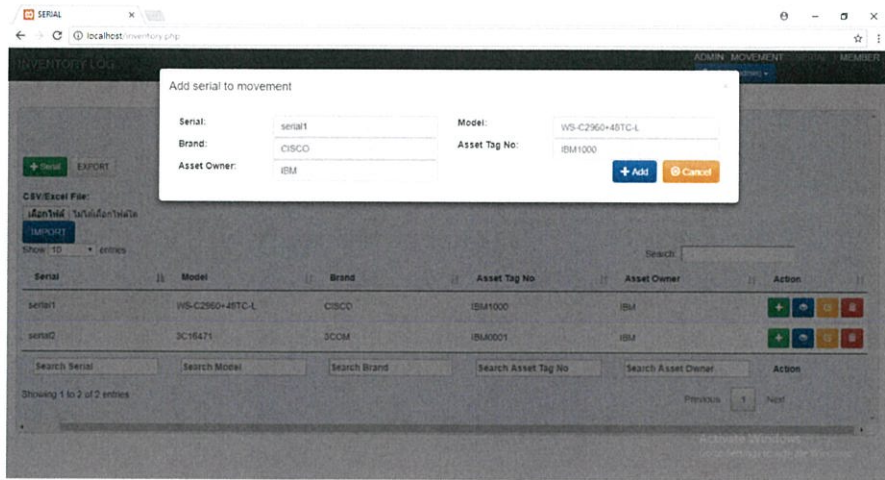
รูปที่ 4.11 การแสดงข้อมูลในตาราง Serial



รูปที่ 4.12 การแก้ไขข้อมูลในตาราง Serial



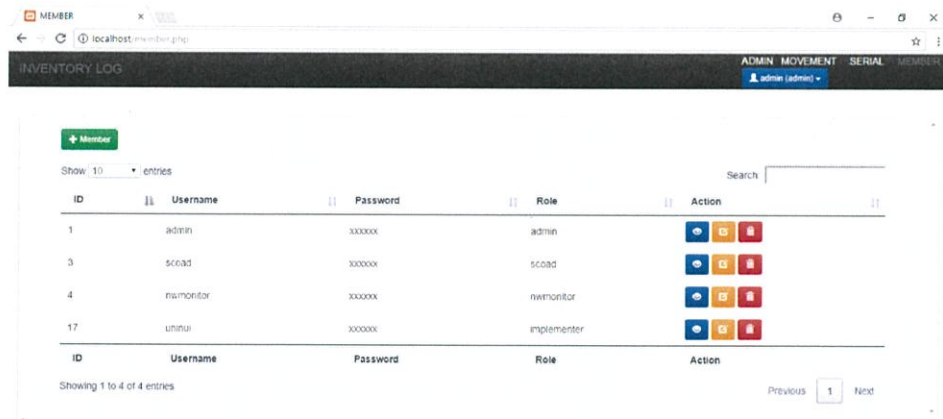
รูปที่ 4.13 การลบข้อมูลในตาราง Serial



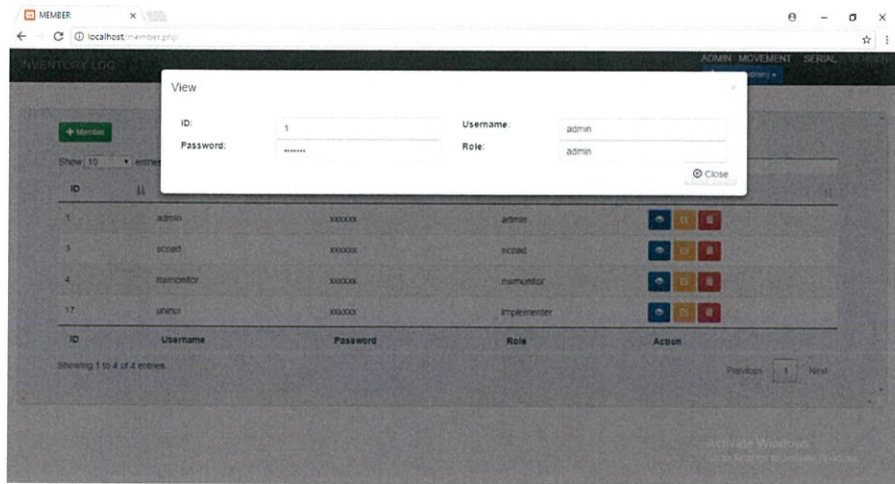
รูปที่ 4.14 การเพิ่มข้อมูลไปยัง movement

- Member

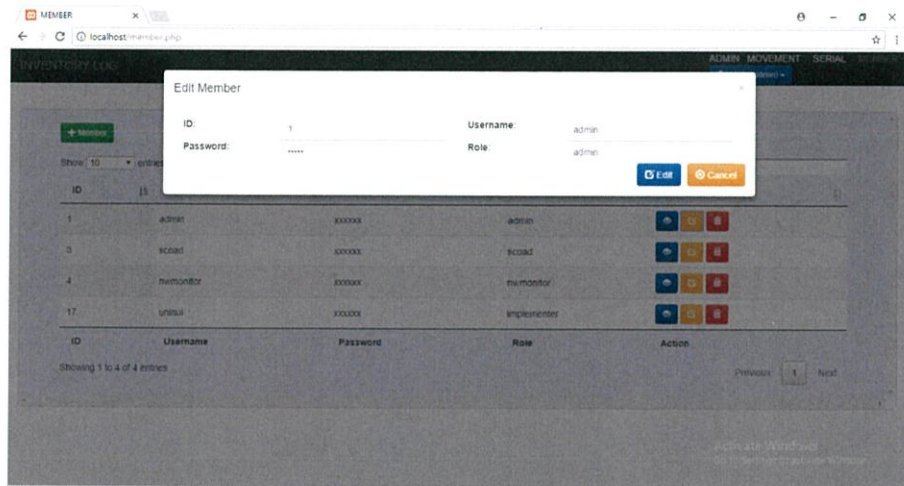
เป็นหน้าแสดงสมาชิกทั้งหมด โดย admin จะเป็นผู้เปลี่ยนสถานะให้กับสมาชิกในระบบ แต่ admin ไม่สามารถรับรู้รหัสผ่านที่สมาชิกลงทะเบียนเข้ามาในระบบได้ แต่แสดงข้อมูล แก้ไขข้อมูล และลบสมาชิกออกได้



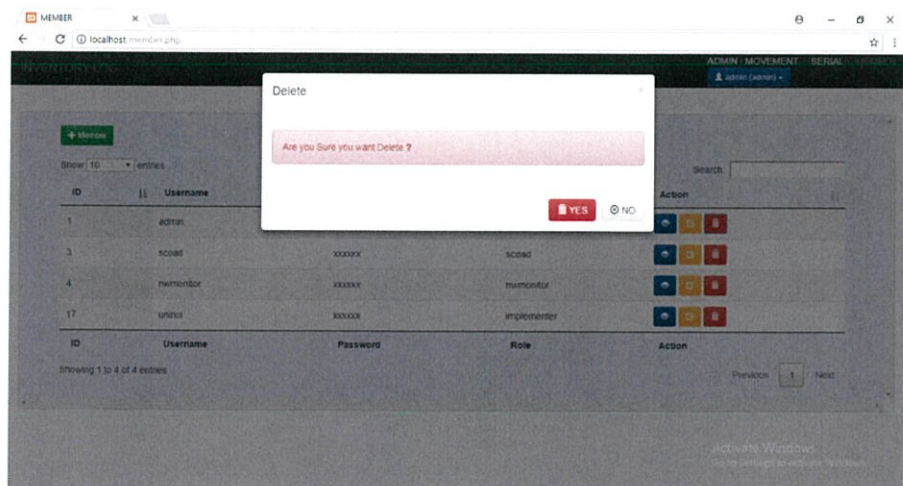
รูปที่ 4.15 Member Page



รูปที่ 4.16 การแสดงข้อมูลสมาชิก



รูปที่ 4.17 การแก้ไขข้อมูลสมาชิก



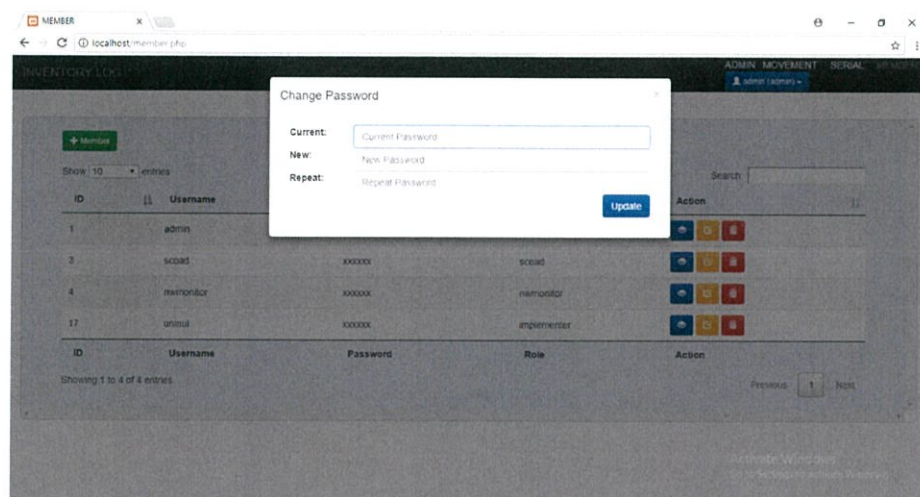
รูปที่ 4.18 การลบสมาชิก

4.3 การเปลี่ยนรหัสผ่าน

การเปลี่ยนรหัสผ่านนั้น สมาชิกทุกคนจะสามารถทำได้ในตอนที่เข้าสู่ระบบเข้ามาแล้ว หรือเปลี่ยนกับ admin



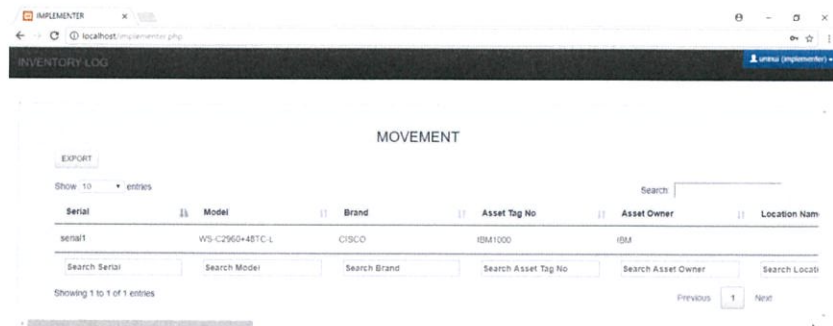
รูปที่ 4.19 ปุ่มแสดงการเปลี่ยนรหัสผ่าน



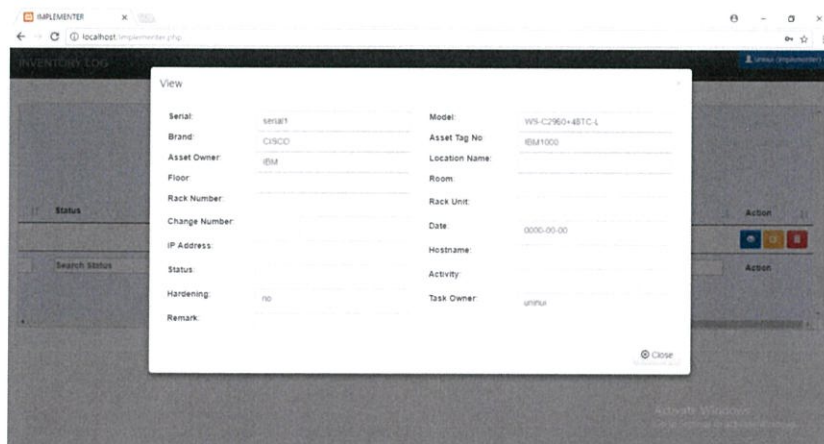
รูปที่ 4.20 การเปลี่ยนรหัสผ่าน

4.4 Implementer Page

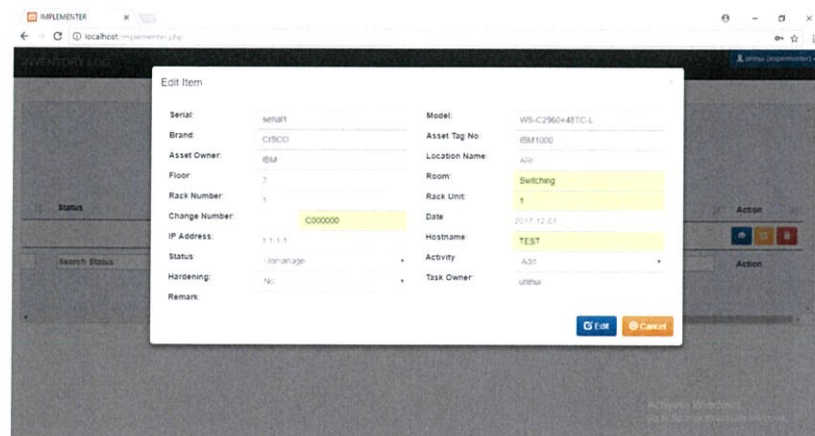
Implementer จะทำการแก้ไขข้อมูลที่ admin ได้เพิ่มเข้ามาโดยการค้นหา อุปกรณ์ของตนเองที่จะนำเข้าหรือนำออก ซึ่งสามารถแสดงข้อมูล แก้ไขข้อมูล และลบข้อมูลได้



รูปที่ 4.21 ตาราง movement ของ Implementer



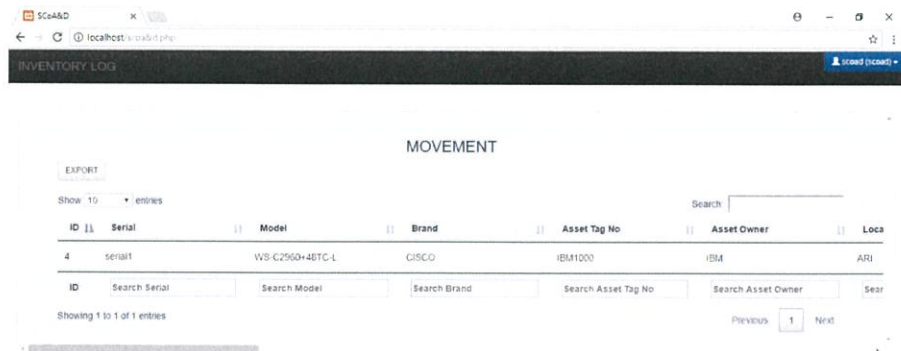
รูปที่ 4.22 การแสดงข้อมูลตาราง movement ของ Implementer



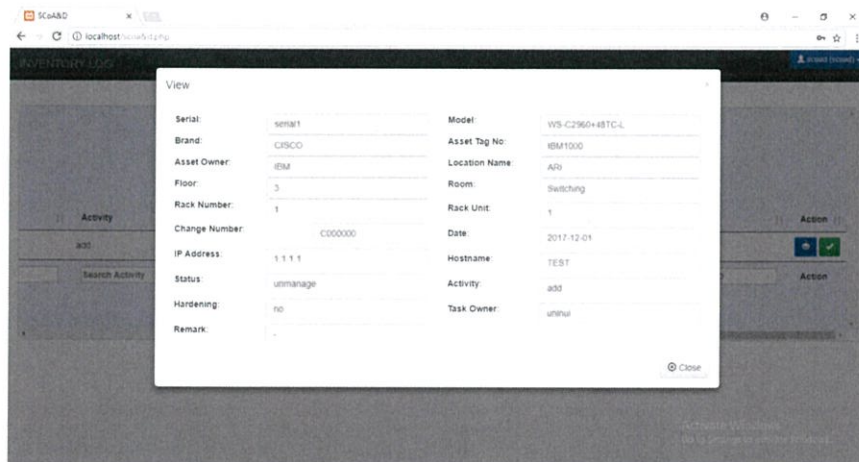
รูปที่ 4.23 การแก้ไขข้อมูลจากตาราง movement ของ Implementer

4.5 SCoA&D Page

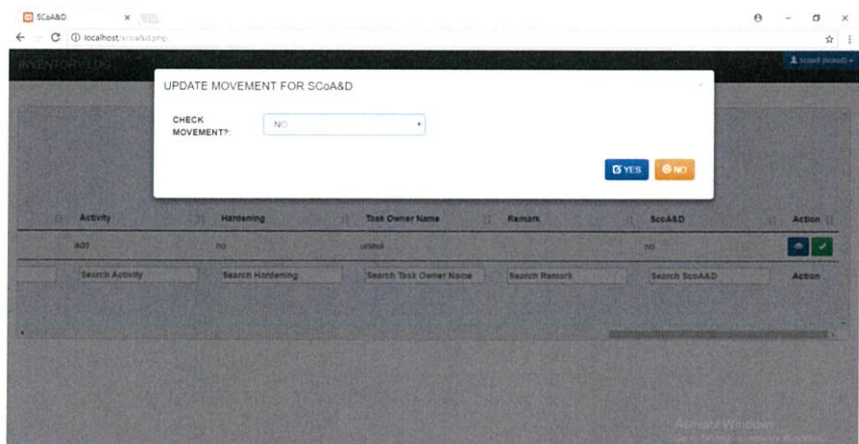
เป็นหน้าสำหรับผู้ตรวจสอบข้อมูล ตรงตามข้อมูลจริงหรือไม่ ซึ่งจะไม่สามารถแก้ไขข้อมูลได้ สามารถแสดงข้อมูลทั้งหมด และตรวจสอบได้



รูปที่ 4.24 SCoA&D Page



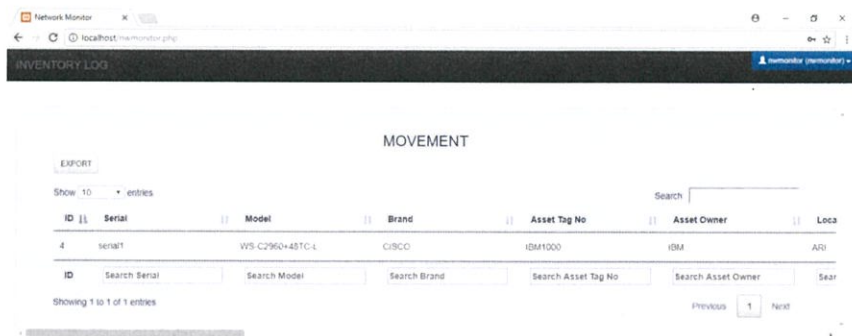
รูปที่ 4.25 การแสดงข้อมูลในตาราง movement ของ SCoA&D



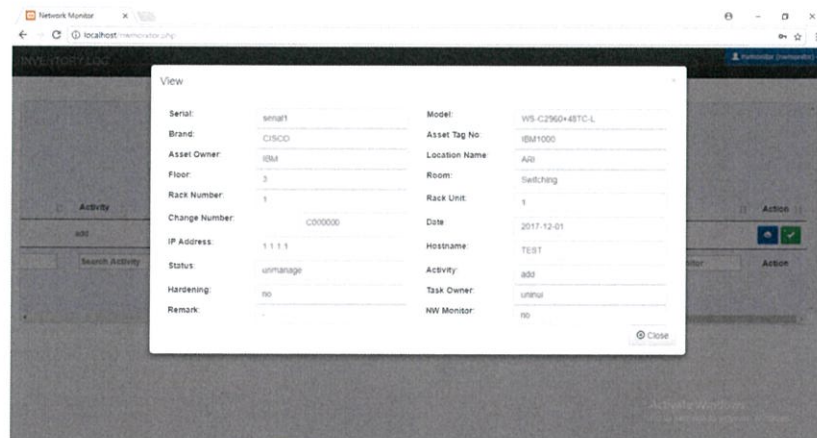
รูปที่ 4.26 การตรวจสอบข้อมูลของ SCoA&D

4.6 Network Monitor Page

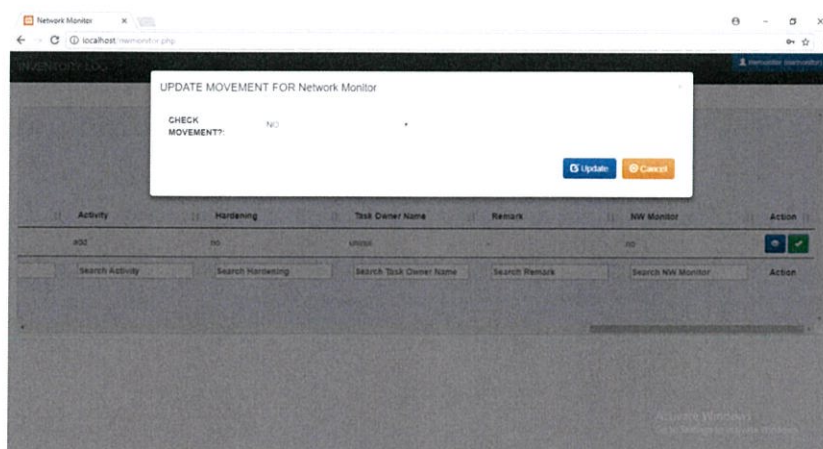
เป็นหน้าของผู้ควบคุมติดตามการทำงาน ซึ่งไม่สามารถแก้ไขข้อมูลได้ ตรวจสอบข้อมูลและแสดงข้อมูลทั้งหมดได้



รูปที่ 4.27 Network monitor Page



รูปที่ 4.28 การแสดงข้อมูลตาราง movement ของ Network monitor



รูปที่ 4.29 การตรวจสอบข้อมูลของ Network monitor

บทที่ 5

สรุปผลการดำเนินการ

5.1 สรุปผลการดำเนินงาน

เว็บไซต์ข้อมูลสินค้าคงคลัง สามารถพัฒนาการทำงานในทีมเน็ตเวิร์คให้เป็นระบบมากขึ้น มีความสะดวกต่อผู้ดูแลระบบ ผู้ควบคุมการทำงาน ผู้ตรวจสอบข้อมูล และผู้ปฏิบัติงานอีกด้วย ในส่วนนี้ยังสามารถนำไปพัฒนาต่อยอดขึ้นไปอีกโดยการเพิ่มหน้าเว็บเพจที่ต้องการเข้าไป ประสิทธิภาพของการทำงานหรือนำไปประยุกต์ใช้งานการทีมอื่นๆได้ เพื่อความสะดวกและรวดเร็วในการติดตามข้อมูล มีการอัปเดตข้อมูลอยู่ตลอดเวลา ทำให้ข้อมูลตรงตามความจริง ไม่มีปัญหาของการสูญหายของอุปกรณ์อีกต่อไป

5.2 อุปสรรค

- การออกแบบเว็บไซต์ซึ่งไม่ตรงตามความต้องการของคนในทีม ต้องทำการเก็บความต้องการเพิ่มเติมอย่างมาก ทำความเข้าใจในระบบการทำงานในทีม
- ความต้องการของผู้ใช้งานมีการเปลี่ยนแปลงอยู่บ่อยครั้งทำให้ต้องแก้ไขอยู่เรื่อย ไม่มีเวลาในการศึกษาการทำงานอย่างอื่นในทีม

5.3 ข้อเสนอแนะ

- ในการทำเว็บไซต์ข้อมูลสินค้าคงคลังควรทำการศึกษาข้อมูลและเก็บความต้องการของผู้ใช้งาน ทำความเข้าใจกระบวนการทำงานให้ดีก่อนทำออกแบบเว็บไซต์
- ในการทำเว็บไซต์ควรมีการทดสอบ และตรวจสอบเรื่อยๆว่า มีความถูกต้องและสมบูรณ์ตามความต้องการของผู้ใช้งานหรือไม่

เอกสารอ้างอิง

[1] “Download XAMPP” [ออนไลน์]. เข้าถึงได้จาก :

<https://www.apachefriends.org/index.html>.2560

[2] “Download Bootstrap” [ออนไลน์]. เข้าถึงได้จาก : <http://getbootstrap.com/>.2560

[3] “Download Sublime Text” [ออนไลน์]. เข้าถึงได้จาก :

<https://www.sublimetext.com/>.2560