ระบบจัดการห้องแลบโปรแกรมมิ่ง

PROGRAMMING LAB MANAGEMENT SYSTEM

นิรันดร์ ก่อตั้งทรัพย์
ปัญญาวุธ นกงาม
วรรณพงศ์ เจริญเรืองทรัพย์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

ระบบจัดการห้องแลบโปรแกรมมิ่ง

PROGRAMMING LAB MANAGEMENT SYSTEM

นิรันดร์ ก่อตั้งทรัพย์
ปัญญาวุธ นกงาม
วรรณพงศ์ เจริญเรืองทรัพย์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

ปริญญานิพนธ์ปีการศึกษา 2558

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบจัดการห้องแลบโปรแกรมมิ่ง

PROGRAMMING LAB MANAGEMENT SYSTEM

ผู้จัดทำ

1. นายนิรันดร์ ก่อตั้งทรัพย์ รหัสนักศึกษา 55010668
2. นายปัญญาวุธ นกงาม รหัสนักศึกษา 55010753
3. นายวรรณพงศ์ เจริญเรืองทรัพย์ รหัสนักศึกษา 55011082

..... อาจารย์ที่ปรึกษา

(อาจารย์คณัฐ ตังติสานนท์)

ระบบจัดการห้องแลบโปรแกรมมิ่ง

นายนิรันดร์ ก่อตั้งทรัพย์ 55010668
นายปัญญาวุธ นกงาม 55010753
นายวรรณพงศ์ เจริญเรืองทรัพย์ 55011082
อาจารย์คณัฐ ตังติสานนท์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2558

บทคัดย่อ

ปัจจุบันนี้การเรียนการสอนในหลาย ๆ วิชามีส่วนเกี่ยวข้องกับการเขียนโปรแกรมเป็นจำนวนมาก ซึ่งมักจะดำเนินการสอนควบคู่กันไประหว่างภาคทฤษฎีกับภาคปฏิบัติ เพราะการสอนภาคปฏิบัติจำต้องให้นักศึกษาทดลองทำแบบฝึกหัดด้วยตนเอง โดยอาจารย์ผู้สอนคอยให้คำแนะนำปรึกษาและตรวจแบบฝึกหัดให้ ทั้งนี้ ปัญหาอาจเกิดขึ้นมาได้เมื่อจำนวนนักศึกษาเพิ่มมากขึ้น โครงงานนี้จึงได้พยายามพัฒนาระบบจัดการห้องแลบโปรแกรมมิ่งขึ้นมาให้อยู่ในรูปแบบของเว็บแอพพลิเคชัน โดยใช้ภาษาพีเอชพีและเอสคิวแอลเป็นหลักในการพัฒนาระบบร่วมกับไทนีซีคอมไพเลอร์ เพื่อแปลรหัสภาษาซีต้นฉบับ ระบบที่พัฒนาขึ้นมานี้จะแบ่งผู้ใช้งานออกเป็น 3 ประเภท คือ นักศึกษา อาจารย์ผู้สอน และผู้ดูแลระบบ การดำเนินการของระบบเริ่มขึ้นโดยนักศึกษาส่งรหัสต้นฉบับของแบบฝึกหัดเข้าไปเพื่อให้ระบบตรวจสอบความถูกต้องและให้คะแนน

ส่วนอาจารย์ผู้สอนทำหน้าที่สร้างโจทย์แบบฝึกหัดออกมาหลายรูปแบบ และสุ่มโจทย์เหล่านี้ให้ได้รูปแบบที่แตกต่างกันก่อนที่จะแจกจ่ายให้นักศึกษา โดยเฉพาะอย่างยิ่งสำหรับนักศึกษาผู้มีตำแหน่งเลขที่นั่งติดกัน และรับผิดชอบทำตารางกำหนดวันเวลาเปิด-ปิดแลบไว้ล่วงหน้า รวมทั้งของแลบปัจจุบัน และของแลบที่เปิดให้ซ่อมย้อนหลังได้ ตรวจสอบการเข้าสู่-ออกจากระบบของนักศึกษาว่าตรงกับตำแหน่งเลขที่นั่งในห้องแลบหรือไม่ นอกจากนี้อาจารย์ยังสามารถตรวจคะแนนในแต่ละแลบของนักศึกษาแต่ละคนได้อีกด้วย

PROGRAMMING LAB MANAGEMENT SYSTEM

Mr. Niran	Kortangasap	55010668
Mr. Panyawat	Nokngam	55010753
Mr. Wannapong	Charoenruangsap	55011082
Mr. Kanat	Tangtisanon	Advisor

Academic Year 2015

ABSTRACT

Nowadays, the teaching of many subjects is involved in various programming on the computer. Most of the teaching operations are performed alongside between the theory and the practice with some exercises. It is necessary that the students do their works by themselves but under the monitoring of their supervisors. However, some difficulties might inevitably occur when the number of the students is successively increased.

Therefore, this project was attempted to develop the management of the Laboratory Programming System, as a teaching tool, in the archetype of web application by employing PHP and SQL as the primary language in conjunction with the Tiny C Compiler for the development in order to decode C language in an original program. The developed program has divided its users into 3 categories: the student, the supervisor, and the administrator. The operation of the system begins the task when a student upload an original coding of the problems of his/her exercises into the system so that they will have the system examined their correctness in the form of scoring for the exercises.

The supervisor's function are to construct the problem exercises into different forms or types, and to appropriately random them before being handed out to the students in the lab, especially the students who have the seat number adjacent to each other in lab class will have the different form of lab exercises, and being responsible for the scheduling of date and time for the opening-closing the lab both of the missions of the current lab practice and the compensable one on the back lab. Also, is to check the log-in, the log-out of the students if they are true to their seat numbers in the lab. In addition, the supervisor enables to check the scores of each student in each of any lab as well.

กิตติกรรมประกาศ

ปริญญานิพนธ์เล่มนี้สามารถสำเร็จลงไปได้ เนื่องจากได้รับความช่วยเหลือเป็นอย่างดีมากจาก อาจารย์คณัฐ ตังติสานนท์ ซึ่งเป็นอาจารย์ที่ปรึกษา ผู้ให้การแนะนำชี้แนะถึงปัญหาที่กำลังเจอหรือสิ่งที่เกิดขึ้นให้ผ่านไปได้ และยังช่วยเหลือให้คำแนะนำตลอดเวลาในการทำงานในทุกๆช่วงของการทำปริญญานิพนธ์เล่มนี้ และคอยกำกับดูแลเกี่ยวกับความคืบหน้าของงานในทุก ๆ ช่วงเวลา เพื่อที่จะได้เกิดความประสบความสำเร็จให้ได้มากที่สุด รวมทั้งยังคอยช่วยดูแลในสถานการณ์ต่าง ๆ และได้ใช้งานจริงกับนักศึกษา เพื่อให้เกิดความประสบความสำเร็จกับโครงงานนี้มากยิ่งขึ้น รวมถึงข้อควรปรับปรุงแก้ไขในโครงงานนี้

ขอขอบพระคุณบิดา และมารดาของบรรดาข้าพเจ้าที่ให้การสนับสนุนในทุกด้านมาโดยตลอด และคอยให้กำลังใจในยามเหน็ดเหนื่อย อ่อนล้า และท้อแท้จากการเรียนและการทำโครงงาน ในทุกช่วงของเวลา เพื่อให้การทำโครงงานนั้นเกิดความประสบความสำเร็จไปได้โดยไม่เกิดปัญหาใด ๆ เกิดขึ้น จนกระทั่งทุกอย่างสำเร็จลุล่วงไปได้

สุดท้ายนี้ขอขอบคุณรุ่นพี่โครงงานที่คอยให้คำปรึกษาในทุก ๆ ช่วงเวลาในการทำโครงงานในส่วนต่าง ๆ ตั้งแต่ต้นจนจบเพื่อคอยแนะนำปัญหาต่าง ๆ ที่จะเกิดขึ้นกับโครงงานนี้และช่วยหาแนวคิดและการแก้ปัญหาต่าง ๆ ที่เกิดขึ้นในการทำโครงงานและรุ่นพี่ที่ช่วยเหลือภายในห้องแลบในการใช้งานได้ จริงของนักศึกษาและคอยให้คำแนะนำต่าง ๆ เวลามีปัญหาได้เกิดขึ้นมาและสามารถผ่านไปได้อย่างประสบความสำเร็จและอีกทั้งเพื่อนๆทั้งหลายที่ค่อยให้คำแนะนำและปัญหาที่เกิดขึ้นเพื่อแก้ไขและพัฒนาได้ดียิ่งขึ้น จึงทำให้โครงงานผ่านไปด้วยความประสบความสำเร็จตลอดทุกท่านที่ให้ความสนใจต่อการดำเนินงานโครงงานนี้ด้วย

นิรันดร์ ก่อตั้งทรัพย์
ปัญญาวุธ นกงาม
วรรณพงศ์ เจริญเรืองทรัพย์

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VI

สารบัญรูป VII

บทที่ 1 บทนำ 1

1.1 ความสำคัญและที่มาของโครงงาน 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขอบเขตของโครงงาน 1

1.4 วิธีการดำเนินการ 2

1.5 ประโยชน์ที่คาดว่าจะได้รับ 2

1.6 ข้อจำกัดของระบบ 2

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 3

2.1 เทคโนโลยีเว็บ 3

2.2 ไคลเอนต์ - เซิร์ฟเวอร์ โมเดล 10

2.3 ภาษาพีเอชพี 12

2.4 ภาษาเอสคิวแอล 17

2.5 โครงร่างพีเอชพี (PHP Framework) 20

2.6 จาวาสคริปต์ (JavaScript) 25

2.7 เอแจ็กซ์ (AJAX) 26

2.8 คอมไพเลอร์ (Compiler) 30

2.9 นิพจน์ปรกติ (Regular expression) 33

สารบัญ (ต่อ)

หน้า

บทที่ 3 การวิเคราะห์และออกแบบระบบ 34

3.1 ความต้องการของระบบ และลักษณะสำคัญ 34

3.2 รายการตรวจสอบความต้องการของระบบ (System Requirement Checklist) 37

3.3 การเปรียบเทียบของระบบเก่าและระบบใหม่ 38

3.4 การใช้ระบบ 56

3.5 การออกแบบฐานข้อมูล 58

3.6 การทำงานระหว่าง Folder (Folder Tree) 61

บทที่ 4 การทดลองและผลการทดลอง 62

4.1 สภาพแวดล้อมที่ใช้ในระบบ 62

4.2 ขั้นตอนการทดลอง 62

บทที่ 5 บทสรุป 73

5.1 สรุปผลการทดลอง 73

5.2 ปัญหา อุปสรรค และแนวทางการแก้ไข 73

5.3 แนวทางในการพัฒนาต่อไป 74

5.4 สรุประบบที่สร้างและสถานะโครงงาน 75

5.5 คำถามที่พบบ่อย (FAQ) 77

บรรณานุกรม 78

ภาคผนวก ก การติดตั้งระบบ 79

ก.1 โปรแกรมที่ใช้ในการติดตั้งระบบ 79

ก.2 ขั้นตอนการติดตั้ง 79

สารบัญตาราง

ตาราง หน้า

2.1 รูปแบบแท็กแสดงส่วนของภาษาพีเอชพี....14

2.2 อักขระต้องห้าม....15

2.3 ตัวดำเนินการต่าง ๆ ในภาษาพีเอชพี....16

2.4 ฟิลด์ต่าง ๆ ของการเพิ่มข้อมูล....18

2.5 ฟิลด์ต่าง ๆ ของการปรับปรุงข้อมูล....19

2.6 ฟิลด์ต่าง ๆ ของการลบข้อมูล....20

2.7 ประเภทไฟล์และวิธีการใช้ไฟล์ต่าง ๆ ในซีไอไซต์....22

2.8 โฟล์เดอร์ต่าง ๆ ในซีไอไซต์....24

2.9 สรุปออบชันของทีซีซี....32

2.10 การอธิบายสายอักขระ....33

3.1 การใช้งานระบบแบ่งตามประเภทผู้ใช้งาน....56

5.1 สรุประบบที่สร้างและสถานะโครงงาน....75

สารบัญรูป

รูป หน้า

2.1 เว็บเพจและเว็บเบราว์เซอร์ 4
2.2 การให้บริการข้อมูลผ่านเว็บ 5
2.3 สถาปัตยกรรมแบบไคลเอนต์-เซิร์ฟเวอร์ หรือแบบชิงเกิลเทียร์ 7
2.4 สถาปัตยกรรมแบบทูเทียร 7
2.5 สถาปัตยกรรมแบบทรีเทียร์ 8
2.6 การทำงานของภาษาสคริปต์ที่ฝั่งไคลเอนต์ 9
2.7 การทำงานของภาษาสคริปต์ที่ฝั่งเซิร์ฟเวอร์ 10
2.8 ลำดับขั้นตอนการทำงานของเว็บไซต์แบบพลวัต 13
2.9 การไหลของข้อมูลในสถาปัตยกรรมแบบเอ็มซีวี 21
2.10 การเปรียบเทียบรูปแบบเว็บแอพพลิเคชันแบบปกติและเอแจ็กซ์ 27
2.11 สถาปัตยกรรมของเอแจ็กซ์ 28
2.12 การปรับปรุงส่วนติดต่อผู้ใช้บางส่วนและการสื่อสารแบบไม่เข้าจังหวะ 29
2.13 กระบวนการคอมไพล์โค้ดโปรแกรม 31
3.1 แผนภาพการทำงานส่วนตรวจโปรแกรมและให้คะแนน 35
3.2 แผนภาพการทำงานส่วนการแจกโจทย์แบบฝึกให้โปรแกรมมิ่งแก่นักศึกษา 36
3.3 หน้า log-in ระบบเก่า 38
3.4 หน้า log-in ระบบใหม่ 38
3.5 หน้า log-out ระบบเก่า 39
3.6 หน้า log-out ระบบใหม่ 39
3.7 การเข้าสู่หน้าจอเพิ่มโจทย์ของระบบเก่า 40
3.8 หน้าจอเพิ่มโจทย์ของระบบเก่า 40
3.9 การเข้าสู่หน้าจอเพิ่มโจทย์ของระบบใหม่ 41
3.10 หน้าจอเพิ่มโจทย์ของระบบใหม่ 41
3.11 การเข้าสู่หน้าจอแก้ไขโจทย์ของระบบเก่า 42
3.12 หน้าจอแก้ไขโจทย์ของระบบเก่า 42
3.13 การเข้าสู่หน้าจอแก้ไขโจทย์ของระบบใหม่ 43
3.14 หน้าจอโจทย์ของระบบใหม่ 43
3.15 การลบโจทย์ของระบบเก่า 44

สารบัญรูป (ต่อ)

รูป หน้า

3.16 การลบโจทย์ของระบบใหม่ 44
3.17 การแจ้งเตือนการลบโจทย์ของระบบใหม่ 44
3.18 การดูโจทย์ทั้งหมดของระบบเก่า 45
3.19 การดูโจทย์บทที่ 2 ของระบบเก่า 45
3.20 การดูโจทย์ทั้งหมดของระบบใหม่ 46
3.21 การดูโจทย์บทที่ 4 ของระบบใหม่ 46
3.22 การแสดงเลเวลของระบบเก่า 47
3.23 การกรองเลเวล 2 ในระบบใหม่ 47
3.24 การกรองเลเวล 3 ในระบบใหม่ 48
3.25 การดูจำนวนโจทย์ของระบบเก่า 48
3.26 การดูจำนวนโจทย์ของระบบใหม่ 49
3.27 การสุ่มโจทย์ให้นักศึกษาของระบบใหม่ 49
3.28 การแสดงโจทย์ที่สุ่มได้ของระบบใหม่ 50
3.29 การดูสถานะเปิด-ปิดแลบของระบบเก่า 50
3.30 การดูสถานะเปิด-ปิดแลบของระบบใหม่ 51
3.31 การเข้าสู่หน้าจอทดสอบโจทย์ของระบบเก่า 51
3.32 การทดสอบโจทย์ของระบบเก่า 52
3.33 การเข้าสู่หน้าจอทดสอบโจทย์ของระบบใหม่ 52
3.34 การทดสอบโจทย์ของระบบใหม่ 53
3.35 ผลการทดสอบโจทย์ของระบบเก่า 53
3.36 การแสดงโค้ดที่ส่งไปของระบบเก่า 54
3.37 ผลการทดสอบโจทย์ของระบบใหม่ 54
3.38 การแสดงโค้ดที่ส่งไปของระบบใหม่ 55
3.39 เอนทิตี้และความสัมพันธ์ระหว่างเอนทิตี้ของฐานข้อมูลในระบบ 58
3.40 เอนทิตี้และความสัมพันธ์ระหว่างเอนทิตี้ของฐานข้อมูลในระบบ (admin) 59
3.41 เอนทิตี้และความสัมพันธ์ระหว่างเอนทิตี้ของฐานข้อมูลในระบบ (Student) 59
3.42 เอนทิตี้และความสัมพันธ์ระหว่างเอนทิตี้ของฐานข้อมูลในระบบ (Supervisor) 60
4.1 หน้าต่างกรอกข้อมูลเพื่อเข้าระบบ 62

สารบัญรูป (ต่อ)

รูป		หน้า
4.2	หน้าต่างเพิ่มผู้ใช้	63
4.3	รายการผู้ใช้	63
4.4	หน้าต่างนำเข้าข้อมูลลงทะเบียนนักศึกษา	64
4.5	รูปแบบของไฟล์ .csv ที่ใช้ในการนำเข้าข้อมูลนักศึกษา	64
4.6	หน้าต่างแสดงการแก้ไขข้อมูลส่วนตัว	65
4.7	หน้าต่างการการลบข้อมูลทั้งหมดในฐานข้อมูล	65
4.8	หน้าต่างแสดงจำนวนโจทย์	66
4.9	หน้าต่างสำหรับกดเข้าสู่หน้าการสร้างโจทย์	66
4.10	หน้าต่างสำหรับการสร้างโจทย์	67
4.11	หน้าต่างสำหรับคลิกเข้าสู่หน้าการทดสอบโจทย์	67
4.12	หน้าต่างสำหรับการทดสอบโจทย์	68
4.13	หากคำตอบถูกจะแสดงเครื่องหมายถูก	68
4.14	หน้าต่างสำหรับคลิกเพื่อแก้ไขและลบโจทย์	69
4.15	หน้าต่างแสดงคะแนนแลบของนักศึกษา	69
4.16	หน้าต่างแสดงการแก้ไขข้อมูลของอาจารย์	70
4.17	หน้าต่างแสดงรายละเอียดการทำแลบ	71
4.18	หน้าต่างแสดงรายละเอียดโจทย์ที่นักศึกษาต้องทำ	71
4.19	หน้าต่างแสดงรายละเอียดโจทย์และผลลัพธ์ตัวอย่าง	72
4.20	หน้าต่างแสดงการแก้ไขข้อมูลส่วนตัว	72
ก.1	ทำการดาวน์โหลดโปรแกรม XAMPP จากเว็บไซต์	79
ก.2	ดับเบิ้ลคลิกที่ตัวโปรแกรมเพื่อเริ่มการติดตั้ง	80
ก.3	กด next เพื่อไปยังขั้นตอนถัดไป	80
ก.4	ทำการเลือกสิ่งที่ต้องการจะติดตั้ง	81
ก.5	กด next เพื่อไปยังขั้นตอนถัดไป	81
ก.6	เข้าสู่สถานะการติดตั้งโปรแกรม	82
ก.7	การลงโปรแกรมเสร็จสิ้น	82
ก.8	เริ่มต้นการใช้งาน Control Panel	83
ก.9	นำโฟลเดอร์โปรเจคมาใส่ในไดเรคทอรี่ที่ทำการติดตั้งโปรแกรม	83

สารบัญรูป (ต่อ)

รูป หน้า

ก.10 นำโฟลเดอร์ TinyC มาใส่ในไดร์ฟ C 84
ก.11 Import ไฟล์ compro14s2v2.sql เข้าสู่ฐานข้อมูล 84
ก.12 ทำการเข้าสู่หน้าแก้ไข password 85
ก.13 ทำการแก้ไข password ใน MySQL 85
ก.14 ทำการแก้ไข password ในไฟล์ config.inc.php 86
ก.15 ทำการแก้ไข short_open_tag เป็น On 86
ก.16 ทำการแก้ไข date.timezone เป็น Asia/Bangkok 86
ก.17 ทำการ restart Apache และ MySQL 87

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของโครงงาน

ในการเรียนวิชาโปรแกรมมิ่งนั้น นอกจากจะต้องเรียนทฤษฎีแล้ว ยังต้องทำแลบหรือภาคปฏิบัติควบคู่กันไปด้วย เพื่อให้นักศึกษาได้ฝึกเขียนโปรแกรมจริงตามที่ได้เรียนทฤษฎีมาและสามารถนำไปประยุกต์ได้ โดยจะต้องมีการจัดตารางเรียนแลบก่อน เมื่อนักศึกษาเข้ามาทำแลบ อาจารย์จะกำหนดโจทย์มาให้เพื่อให้นักศึกษาทำความเข้าใจและเขียนโปรแกรมให้ได้ผลลัพธ์ตามที่กำหนดไว้ เมื่อเสร็จแล้วจึงเรียกอาจารย์ให้มาตรวจผลลัพธ์จึงให้คะแนน อาจจะมีการซักถามเพื่อทดสอบความเข้าใจ ซึ่งในการตรวจผลลัพธ์ในการให้คะแนนนั้น อาจทำให้เกิดความล่าช้า เนื่องจากนักศึกษามีจำนวนมาก และจำนวนอาจารย์ไม่เพียงพอต่อความต้องการของนักศึกษา

โครงงานชิ้นนี้จึงได้จัดทำขึ้นมาเพื่อจัดตารางเรียนแลบวิชาโปรแกรมมิ่งเพื่อให้นักศึกษาได้เรียนรู้การเขียนโปรแกรมทั้งด้านไวยากรณ์ และขั้นตอนวิธีการเขียนโปรแกรมจนสามารถนำมาประยุกต์เขียนโปรแกรมสำหรับโจทย์ต่าง ๆ เพื่อให้นักศึกษาฝึกแก้ไขปัญหาในการเขียนโปรแกรมได้ด้วยตนเอง รวมทั้งสามารถส่งผลลัพธ์ผ่านเว็บแอพพลิเคชั่นเพื่อตรวจสอบความถูกต้องได้

1.2 วัตถุประสงค์ของโครงงาน

วัตถุประสงค์ของโครงงานนี้เพื่อให้นักศึกษาทำการส่งซอร์สโค้ดผ่านเว็บแอพพลิเคชั่นที่สร้างขึ้น เพื่อตรวจสอบผลลัพธ์ที่ได้ ซึ่งจะทำให้ช่วยลดเวลาในการตรวจโปรแกรมของนักศึกษา และสามารถจัดทำรายงานผลคะแนนแลบวิชาโปรแกรมมิ่งได้สะดวกและรวดเร็วมากขึ้น อีกทั้งช่วยจัดตารางเวลาเรียนแลบวิชาโปรแกรมมิ่งได้อีกด้วย

1.3 ขอบเขตของโครงงาน

เป็นระบบที่ผู้ดูแลระบบนำข้อมูลการลงทะเบียนของนักศึกษาเข้ามา จากนั้นระบบจะทำการจัดตารางแลบของ วิชาโปรแกรมมิ่ง สำหรับนักศึกษาที่ลงวิชาโปรแกรมมิ่ง โดยผู้ดูแลระบบสามารถเพิ่ม-ลดกลุ่มที่แลบหรือเปิด-ปิดกลุ่มเรียนเพื่ออนุญาตให้นักศึกษาเข้ามาทำแลบได้ ระบบจะให้นักศึกษาสามารถอัพโหลดไฟล์ซอสโค้ดเพื่อให้ระบบนำไปคอมไพล์และนำไปตรวจสอบกับผลลัพธ์ที่กำหนดให้ว่าตรงกันหรือไม่ พร้อมทั้งแสดงผลการตรวจออกมา นอกจากนี้ ระบบยังสามารถแสดงประวัติการส่งงาน คะแนนที่ได้รับ รายละเอียดการส่งงานของนักศึกษา และให้นักศึกษาสามารถแก้ไขข้อมูลส่วนตัวได้อีกด้วย

1.4 วิธีการดำเนินการ

1.4.1 การวางแผน

ทำการศึกษาระบบเก่า เพื่อหาแนวทางการปรับปรุงให้มีประสิทธิภาพยิ่งขึ้น รวมถึงกำหนดวัตถุประสงค์ ขอบเขตและระยะเวลาดำเนินโครงงาน และศึกษาเกี่ยวกับเทคโนโลยีใหม่ๆ ที่มีความจำเป็นต่อระบบ เพื่อนำมาปรับปรุงระบบให้มีความทันสมัยขึ้น และใช้งานการใช้งานที่ง่ายขึ้น รวมไปถึงการปรับปรุงในภายภาคหน้าที่สามารถทำได้อย่างสะดวกและรวดเร็วขึ้น

1.4.2 การวิเคราะห์และออกแบบระบบ

กำหนดรายการสิ่งที่ระบบต้องมี , เครื่องมือและซอฟต์แวร์ที่จำเป็น ออกแบบฐานข้อมูลเพื่อรองรับข้อมูลได้ในทุกกรณี , แผนภาพการทำงานของระบบและส่วนติดต่อผู้ใช้ให้เป็นไปตามวัตถุประสงค์ของโครงงาน

1.4.3 การทดลอง

ทดลองเขียนโปรแกรมส่วนย่อยเพื่อทดสอบแนวคิดที่ได้วิเคราะห์และออกแบบไว้ ศึกษาและหาข้อผิดพลาดของการทำงาน ทำการปรับปรุงแก้ไขเพื่อให้ส่วนต่าง ๆ ทำงานได้อย่างถูกต้องสมบูรณ์มากยิ่งขึ้น

1.4.4 สรุปผล

สรุปผลการทดลอง บันทึกปัญหาที่พบเพื่อนำมาปรับปรุงแก้ไขระบบให้มีประสิทธิภาพและสามารถใช้งานได้จริง

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1) ได้รับความรู้ความเข้าใจในด้านเว็บเทคโนโลยี
2) ได้รับความรู้ความเข้าใจเกี่ยวกับการติดต่อสื่อสารระหว่างลูกข่ายกับเซิฟเวอร์
3) ได้รับความรู้ในการเขียนเว็บแอพพลิเคชั่นโดยใช้ภาษาต่าง ๆ เช่น พีเอชพี เอชทีเอ็มแอล จาวาสคริป เป็นต้น
4) ได้รับความรู้ความเข้าใจในการวางแผนระบบ วิเคราะห์ ออกแบบ ตลอดจนการสร้างและการนำไปใช้งาน

1.6 ข้อจำกัดของระบบ

1) ระบบจะตรวจได้เฉพาะโปรแกรมที่เขียนโดยภาษาซีเท่านั้น
2) ในการแก้ไขข้อมูลส่วนตัว ให้ใช้รูปประจำตัวที่มีขนาดไม่เกิน 100 กิโลไบต์
3) ห้ามใช้คำสั่ง getch() เนื่องจากจะทำให้เกิดการวนลูปในเซิฟเวอร์อย่างไม่สิ้นสุด

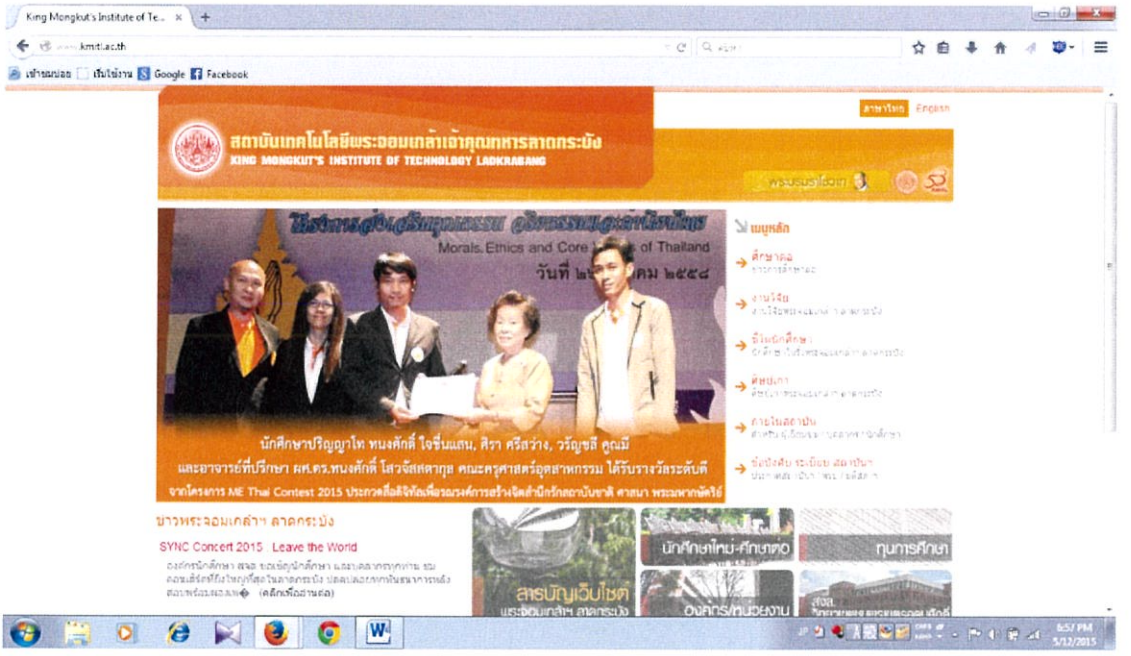
บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 เทคโนโลยีเว็บ

เว็บเป็นเทคโนโลยีของการสื่อสารที่ทำให้ผู้ใช้จากทั่วโลกสามารถเข้าใช้ข้อมูลที่มีให้ได้อย่างสะดวกและรวดเร็ว มีการเก็บเอกสารไว้ที่ฐานข้อมูลซึ่งจะถูกเข้าใช้และแสดงผลโดยเว็บเบราว์เซอร์ ซึ่งเป็นโปรแกรมที่ถูกติดตั้งอยู่ที่เครื่องคอมพิวเตอร์ปลายทาง โดยในการใช้เว็บเบราว์เซอร์นี้ ผู้ใช้สามารถร้องขอข้อมูลหรือเอกสารได้จากคอมพิวเตอร์เครื่องอื่นที่อยู่นอกระบบเครือข่าย และเมื่อได้รับข้อมูลมาแล้ว เว็บเบราว์เซอร์ก็จะแสดงผลข้อมูลเหล่านั้นให้แก่ผู้ใช้

2.1.1 วิวัฒนาการของเว็บ

แนวความคิดของการให้บริการเว็บเริ่มต้นจากผู้ให้บริการ ผู้ให้บริการจำเป็นต้องติดตั้งเครื่องบริการ ที่เรียกว่าเว็บเซิร์ฟเวอร์ขึ้นมา โดยอาจเรียกเว็บเซิร์ฟเวอร์นั้นว่าเว็บไซต์ จากนั้นนำข้อมูลที่ต้องการเผยแพร่ขึ้นไปไว้บนเว็บเซิร์ฟเวอร์ที่ได้รับอนุญาตให้สามารถเข้าใช้บริการจากภายนอกได้ เว็บเซิร์ฟเวอร์ที่ตั้งขึ้นนั้นจะต้องเชื่อมต่อกับระบบอินเทอร์เน็ตด้วยโพรโตคอลทีซีพีไอพี (TCP/IP: Transmission control protocol/Internet protocol) ผ่านทางพอร์ต 80 และจะต้องสนับสนุนการให้บริการข้อมูลผ่านโพรโตคอลเอชทีทีพี (HTTP: Hypertext transfer protocol) ในชั้นแอพพลิเคชัน ในส่วนของผู้ใช้งานนั้นจำเป็นที่จะต้องติดตั้งโปรแกรมเว็บเบราว์เซอร์ ซึ่งในการใช้งานผู้ใช้จะต้องพิมพ์ยูอาร์แอล (URL: Unified Resource Locator) ซึ่งเป็นตำแหน่งที่บ่งบอกถึงเว็บเซิร์ฟเวอร์และข้อมูลที่ต้องการใช้ เมื่อเว็บเซิร์ฟเวอร์ได้รับการร้องขอจากผู้ใช้ก็จะตรวจสอบเงื่อนไขต่าง ๆ และส่งข้อมูลกลับไปให้แก่ผู้ใช้ที่ทำการร้องขอข้อมูลมา



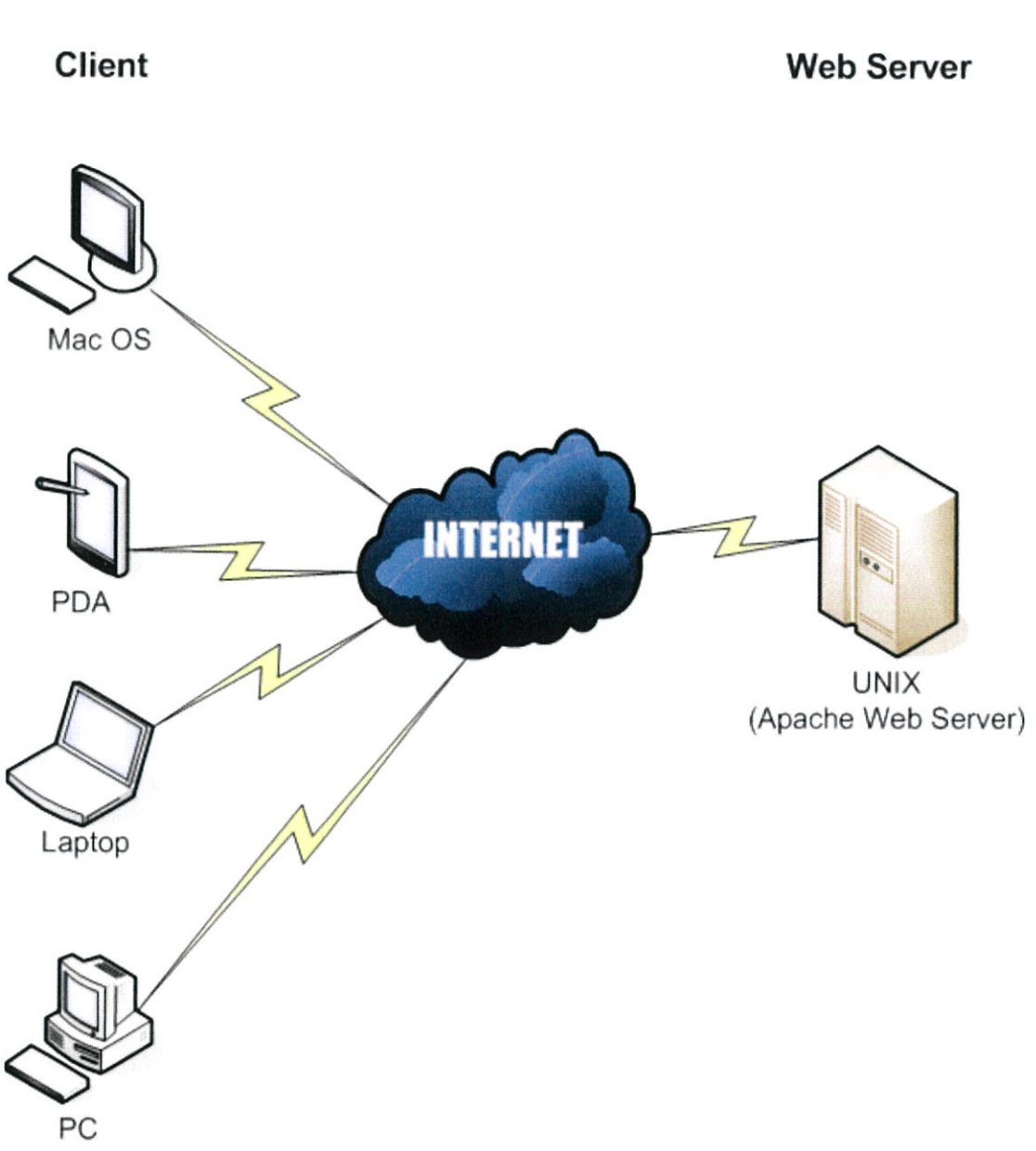
รูป 2.1 เว็บเพจและเว็บเบราว์เซอร์

ในส่วนของข้อมูลที่ถูกเก็บอยู่ภายในเซิร์ฟเวอร์นั้น เพื่อให้เกิดความเป็นกลางในการนำเสนอข้อมูลของเว็บ จึงได้มีการจัดตั้งมาตรฐานของรูปแบบข้อมูลสำหรับเว็บขึ้น มีชื่อเรียกว่า เอชทีเอ็มแอล (HTML: Hyper Text Markup Language) โดยผู้ให้บริการต้องจัดรูปแบบข้อมูลของตนให้อยู่ในรูปแบบของเอชทีเอ็มแอล และเมื่อข้อมูลถูกส่งไปยังผู้ใช้ โปรแกรมเว็บเบราว์เซอร์จะทำการแปลงรูปแบบของเอชทีเอ็มแอลให้มีโครงสร้างและหน้าตาตามที่ผู้ให้บริการต้องการ

จากพื้นฐานของการให้บริการเว็บจะเห็นได้ว่า เว็บเป็นการให้บริการข้อมูลในทิศทางเดียว กล่าวคือ จากผู้ให้บริการไปยังผู้ใช้บริการ ซึ่งอินเตอร์เน็ตนั้นมีการเติบโตที่รวดเร็ว โดยสาเหตุหนึ่งก็มาจากการให้บริการเว็บ ทำให้เกิดความสะดวกรวดเร็วในการเข้าถึงข้อมูลต่าง ๆ

เมื่อผู้ใช้เว็บมีจำนวนมากขึ้น จากรูปแบบเริ่มต้นของการให้บริการแบบทิศทางเดียวในรูปแบบของข้อความและรูปภาพ ก็ได้มีการขยายไปสู่ความหลากหลายของชนิดข้อมูล เช่น เสียง และวิดีโอ นอกจากนี้ยังมีการรับข้อมูลโต้ตอบกับผู้ใช้ได้

จากความสามารถในการรับข้อมูลและโต้ตอบกับผู้ใช้นั่นเองที่ส่งผลให้เกิดแอพพลิเคชันรูปแบบใหม่ คือ เว็บแอพพลิเคชัน โดยทั่วไปโปรแกรมที่ผู้ใช้ใช้งานกันนั้นจะเป็นโปรแกรมที่ติดตั้งอยู่บนคอมพิวเตอร์ ซึ่งมีข้อจำกัดคือผู้ใช้ต้องติดตั้งโปรแกรมลงบนเครื่องของตนเอง ซึ่งการให้บริการแอพพลิเคชันแบบนี้ก็สามารถช่วยอำนวยความสะดวกได้อีกรูปแบบหนึ่ง



รูป 2.2 การให้บริการข้อมูลผ่านเว็บ

จากความสามารถของเว็บในการสื่อสารข้อมูลแบบสองทางและความเป็นหนึ่งในการนำเสนอข้อมูล ทำให้นักพัฒนาได้เป็นประโยชน์และเริ่มพัฒนาแอพพลิเคชันบนพื้นฐานของเว็บออกมา ซึ่งมีข้อได้เปรียบกว่าแบบเก่า คือ ผู้ใช้สามารถเข้าถึงแอพพลิเคชันจากที่ใดก็ได้ที่มีโปรแกรมเว็บเบราว์เซอร์ ซึ่งไม่ได้ถูกจำกัดอยู่แต่เพียงคอมพิวเตอร์ตั้งโต๊ะเท่านั้น แอพพลิเคชันนี้ยังสามารถเข้าถึงได้จากโทรศัพท์เคลื่อนที่ รวมถึงอุปกรณ์ต่าง ๆ ได้เช่นเดียวกัน

2.1.2 การพัฒนาเว็บ

ในอดีตการพัฒนาเว็บนั้นเป็นการสร้างและนำเสนอข้อมูลที่ต้องการให้บริการเท่านั้น หรือถ้าเว็บไซต์มีขนาดใหญ่ก็อาจจะต้องมีการกำหนดโครงสร้างของเว็บขึ้นมาเพื่อให้ง่ายต่อการ

พัฒนาจัดการและแก้ไข แต่สำหรับการให้บริการเว็บในปัจจุบันไม่เพียงแต่ความหลากหลายของข้อมูลและรูปแบบการให้บริการ ตัวเว็บไซต์ยังมีขนาดใหญ่และมีความซับซ้อนมากขึ้นด้วยเช่นกัน

สำหรับการพัฒนาเว็บแอพพลิเคชั่นสามารถแบ่งออกได้เป็นสามระดับคือ

1) การพัฒนาเว็บสำหรับให้บริการข้อมูล

2) การพัฒนาเว็บแบบโต้ตอบได้

3) การพัฒนาเว็บแอพพลิเคชั่น

โดยการพัฒนาเว็บสำหรับให้บริการข้อมูลนั้นเป็นการสร้างเพื่อให้บริการข้อมูลแบบทิศทางเดียว ถ้าเป็นการพัฒนาเว็บที่มีขนาดไม่ใหญ่มาก ผู้พัฒนาสามารถสร้างเว็บไซต์ขึ้นมาได้ทันทีโดยไม่ต้องดำเนินกระบวนการออกแบบ แต่ถ้าเว็บมีขนาดใหญ่ขึ้นมา ผู้พัฒนาอาจจำเป็นต้องมีการออกแบบโครงสร้างเว็บไซต์ในการพัฒนาเว็บไซต์สำหรับให้บริการข้อมูลผู้พัฒนาจะต้องเน้นความสะดวกของผู้ใช้ในการเข้าถึงข้อมูล ซึ่งจำเป็นต้องออกแบบโครงสร้างของเว็บให้สามารถเข้าถึงแต่ละส่วนได้ง่าย ไม่ซับซ้อนเกินไป พร้อมทั้งจะต้องสามารถเพิ่มเติมหรือแก้ไขข้อมูลเว็บที่มีอยู่ได้ง่ายด้วย

2.1.3 สถาปัตยกรรมในการติดตั้งเว็บไซต์

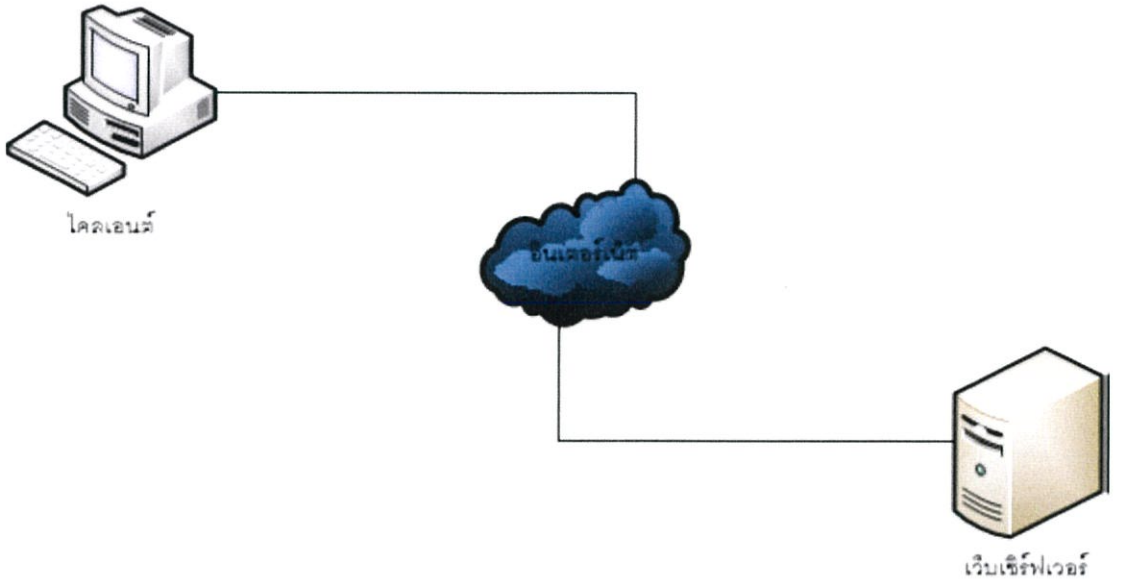
สถาปัตยกรรมในการติดตั้งเว็บไซต์สามารถแบ่งออกได้เป็น 3 รูปแบบ ตามการให้บริการของเว็บ คือ

1) สถาปัตยกรรมแบบไคลเอนต์ – เซิร์ฟเวอร์ (Client - Server) หรือแบบซิงเกิลเทียร์ (Single - tier)

2) สถาปัตยกรรมแบบทูเทียร์ (Two -tier)

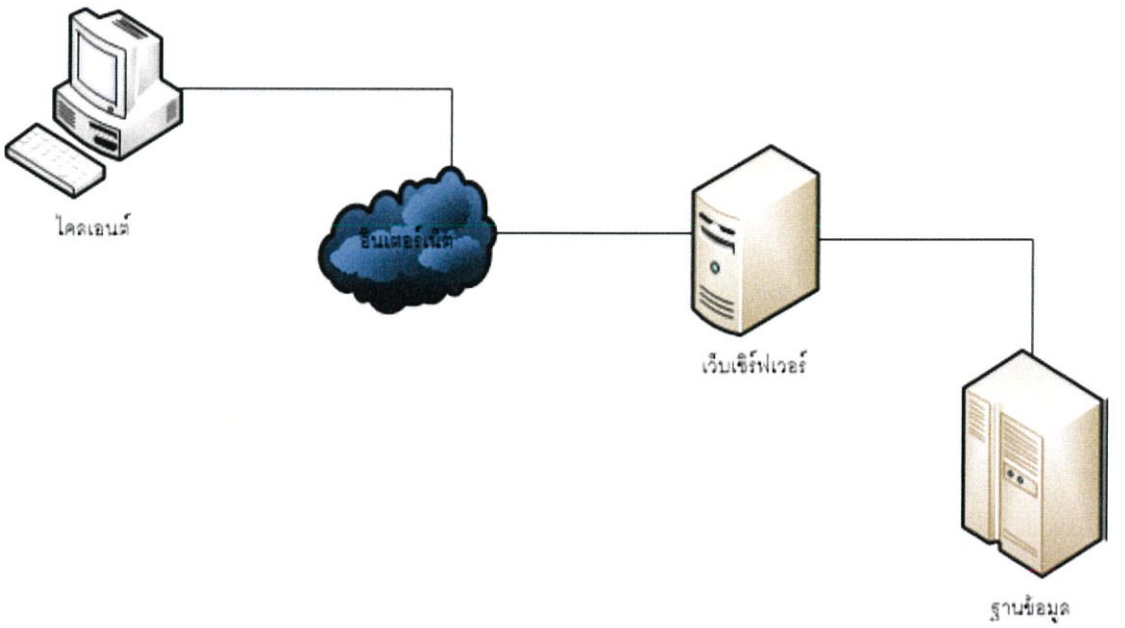
3) สถาปัตยกรรมแบบทรีเทียร์ (Three - tier)

สำหรับสถาปัตยกรรมแบบไคลเอนต์-เซิร์ฟเวอร์หรือสถาปัตยกรรมแบบซิงเกิลเทียร์ เป็นสถาปัตยกรรมที่ประกอบด้วยเว็บเซิร์ฟเวอร์และไคลเอนต์ โดยเว็บเซิร์ฟเวอร์นั้นจะทำหน้าที่ในการให้บริการข้อมูลเว็บ ส่วนไคลเอนต์ทำหน้าที่รับข้อมูลจากเว็บเซิร์ฟเวอร์มาแสดงผลบนเว็บเบราว์เซอร์และอาจจะทำการส่งข้อมูลจากผู้ใช้กลับไปให้กับเซิร์ฟเวอร์ได้



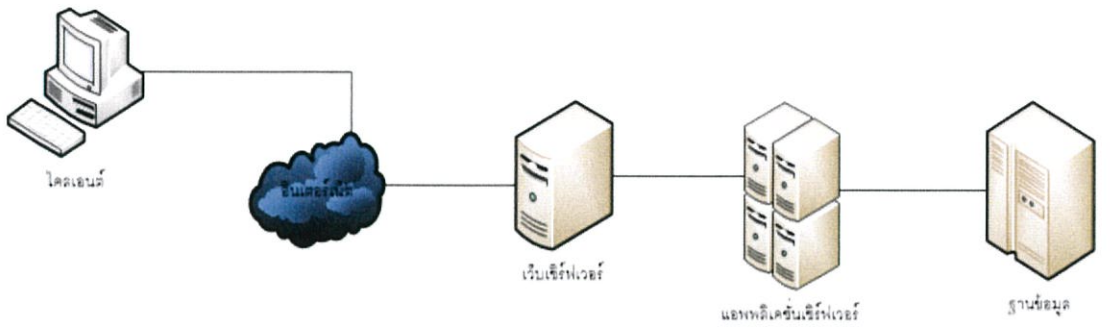
รูป 2.3 สถาปัตยกรรมแบบไคลเอนต์-เซิร์ฟเวอร์ หรือแบบชิงเกิลเทียร์

สถาปัตยกรรมแบบทูเทียร์ จะเป็นสถาปัตยกรรมของเว็บที่มีการเพิ่มระบบฐานข้อมูล (Database) เข้ามา ซึ่งสถาปัตยกรรมแบบนี้จะเหมาะกับเว็บแอพพลิเคชันที่เน้นการเก็บข้อมูลของผู้ใช้เป็นหลัก ซึ่งการนำระบบฐานข้อมูลเข้ามาใช้นั้นจะทำให้การเก็บและดึงข้อมูลสามารถทำได้ผ่านภาษาเอสคิวแอล (SQL) หากเลือกใช้ระบบฐานข้อมูลแบบสัมพันธ์ (Relational DBMS)



รูป 2.4 สถาปัตยกรรมแบบทูเทียร์

สำหรับสถาปัตยกรรมแบบทรีเทียร์นั้น เป็นสถาปัตยกรรมที่ได้แยกระบบออกเป็น ระบบฐานข้อมูล แอพพลิเคชันเซิร์ฟเวอร์ เว็บเซิร์ฟเวอร์ และไคลเอนต์ ในสถาปัตยกรรมแบบนี้จะมีการแยกส่วนแอพพลิเคชันเซิร์ฟเวอร์กับเว็บเซิร์ฟเวอร์ออกจากกัน โดยส่วนของแอพพลิเคชันเซิร์ฟเวอร์นั้นจะทำหน้าที่ในการประมวลผลหรือทำธุรกรรมต่าง ๆ เป็นหลัก ส่วนเว็บเซิร์ฟเวอร์จะนำผลลัพธ์ที่ได้จากการประมวลผลมาจัดรูปแบบในแบบของเอชทีเอ็มแอล เพื่อส่งไปแสดงผลยังไคลเอนต์



รูป 2.5 สถาปัตยกรรมแบบทรีเทียร์

2.1.4 เทคโนโลยีที่สนับสนุน

เนื่องจากการให้บริการเว็บในปัจจุบันไม่ได้เป็นเพียงการให้บริการข้อมูลแบบเอชทีเอ็มแอลเท่านั้น เว็บไซต์ยังมีความสามารถในการให้บริการแบบโต้ตอบ การประมวลผล และการจัดเก็บข้อมูลของผู้ใช้ด้วย ดังนั้นผู้พัฒนาจึงจำเป็นต้องมีเครื่องมือและมีภาษาในการเขียนโปรแกรมที่เหมาะสมเพื่อให้การพัฒนาเว็บเป็นไปได้อย่างมีประสิทธิภาพ

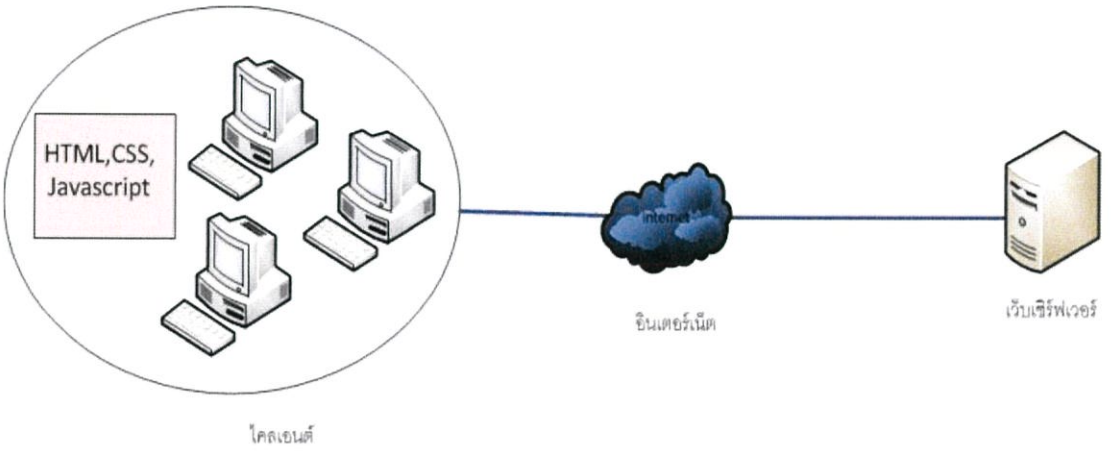
สำหรับเครื่องมือที่เกี่ยวข้องกับการพัฒนาเว็บสามารถแบ่งออกได้เป็น 2 ประเภท คือ เครื่องมือสำหรับการสร้างเพจเอชทีเอ็มแอล ซึ่งรวมความสามารถในการวางโครงสร้างเว็บไซต์และเครื่องมือช่วยในการเขียนโปรแกรมซึ่งรวมถึงการติดต่อกับฐานข้อมูลด้วย สำหรับโปรแกรมในการสร้างเพจเอชทีเอ็มแอล และวางโครงสร้างเว็บไซต์ จะเป็นเครื่องมือที่เข้ามาช่วยในการสร้างส่วนติดต่อผู้ใช้เป็นหลัก ตัวอย่างเช่น Adobe Dreamweaver, Microsoft FrontPage เป็นต้น

การตัดสินใจเลือกภาษาและเครื่องมือในการพัฒนานั้นก็มีหลากหลายปัจจัยที่เข้ามาเกี่ยวข้อง เช่น หากเว็บที่ต้องการสร้างขึ้นเป็นแอพพลิเคชันพื้นฐานที่มีการทำงานไม่ซับซ้อน และให้บริการต่อผู้ใช้จำนวนน้อย การเลือกใช้ภาษาสคริปต์อย่าง เพิร์ล (Perl), พีเอชพี (PHP) หรือเอเอสพี (ASP) ก็เป็นสิ่งที่เหมาะสม เนื่องจากภาษาสคริปต์เป็นสิ่งที่เรียนรู้และใช้งานได้ง่าย แต่มีข้อเสียคือ ภาษาสคริปต์เหล่านี้มีแนวโน้มที่จะทำให้โครงสร้างของโปรแกรมไม่มั่นคงและยากต่อการแก้ไข ซึ่งไม่เหมาะกับแอพพลิเคชันขนาดใหญ่และซับซ้อน แต่สามารถนำโครงร่าง (Framework) มาใช้งานร่วมเพื่อช่วยในการรักษาโครงสร้างของเว็บแอพพลิเคชันขนาดใหญ่ได้

2.1.5 ภาษาสคริปต์สำหรับเว็บไซต์

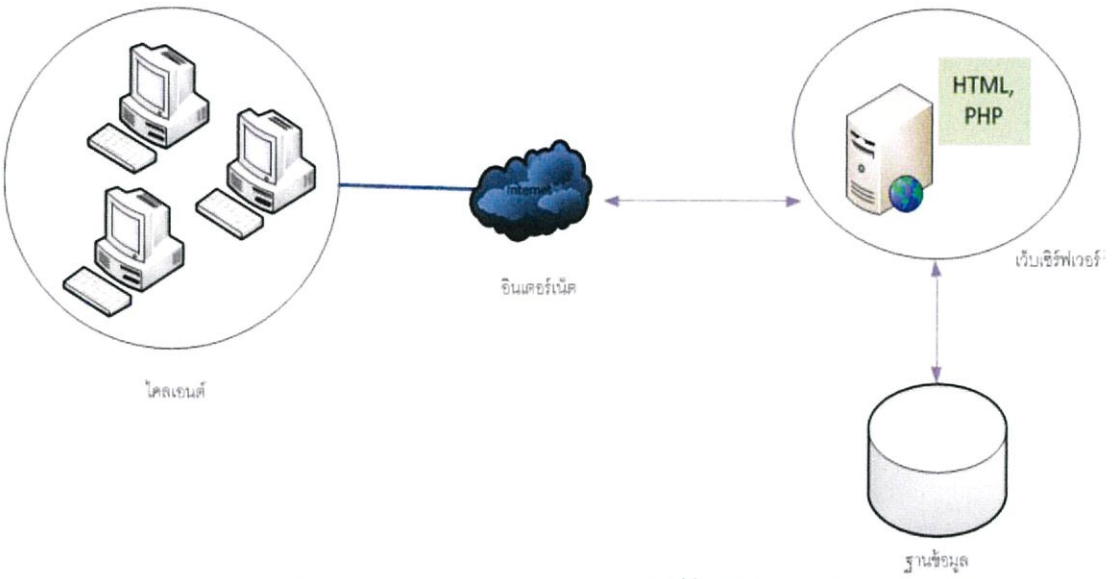
ภาษาเอชทีเอ็มแอลนั้นเป็นเพียงภาษาที่ใช้ในการแสดงผล ดังนั้นเพื่อให้เว็บสามารถคำนวณหรือตัดสินใจได้ จำเป็นต้องใช้ภาษาในการเขียนโปรแกรมหรือภาษาสคริปต์อื่น ๆ ช่วยเพิ่มความสามารถลงในเพจเอชทีเอ็มแอล ซึ่งภาษาสคริปต์สามารถแบ่งออกได้เป็น 2 ระดับ คือ ภาษาสคริปต์ที่ฝั่งไคลเอนต์ (Client - side Script) และภาษาสคริปต์ที่ฝั่งเซิร์ฟเวอร์ (Server - side Script)

สำหรับภาษาสคริปต์ที่ฝั่งไคลเอนต์นั้น โค้ดของภาษาสคริปต์เหล่านี้จะฝังตัวอยู่ในเพจเอชทีเอ็มแอล ดังนั้น เมื่อผู้ใช้ดาวน์โหลดเพจเอชทีเอ็มแอลมา ภาษาสคริปต์เหล่านี้ก็จะติดมาด้วย และเมื่อตัวแปรความหมายของภาษาเอชทีเอ็มแอลในโปรแกรมเว็บเบราว์เซอร์พบแท็กที่บ่งบอกว่าเป็นส่วนของโค้ดภาษาสคริปต์ โปรแกรมจะเรียกส่วนของอินเทอร์พรีเตอร์ (Interpreter) ขึ้นมาประมวลผลและทำงานตามภาษาสคริปต์ที่ได้เขียนไว้



รูป 2.6 การทำงานของภาษาสคริปต์ที่ฝั่งไคลเอนต์

ในส่วนของภาษาสคริปต์ที่ฝั่งเซิร์ฟเวอร์นั้นเป็นภาษาสคริปต์ที่เขียนขึ้นมาเพื่อใช้ในการสร้างเว็บเพจตามข้อมูลหรือตามความต้องการของผู้ใช้ ภาษาสคริปต์ที่ฝั่งเซิร์ฟเวอร์ก็จะมีการฝังโค้ดไว้ในเพจเอชทีเอ็มแอลเช่นกัน แต่ในการทำงานนั้น เมื่อมีการร้องขอเพจเอชทีเอ็มแอลเข้ามา เว็บเซิร์ฟเวอร์จะไม่ได้ส่งไฟล์เอชทีเอ็มแอลนั้นไปในทันที แต่จะมีการเรียกอินเทอร์พรีเตอร์ขึ้นมาทำงานก่อน เพื่อที่จะเปลี่ยนแท็กที่ฝังอยู่ในเอชทีเอ็มแอลให้กลายเป็นแท็กเอชทีเอ็มแอลตามที่ได้เขียนโปรแกรมภาษาสคริปต์ไว้



รูป 2.7 การทำงานของภาษาสคริปต์ที่ฝั่งเซิร์ฟเวอร์

จากคุณลักษณะของการทำงานที่แตกต่างกัน ทำให้ภาษาสคริปต์ทั้งสองแบบมีความสามารถที่แตกต่างกันด้วย ทั้งนี้ภาษาสคริปต์ที่ฝั่งไคลเอนต์นั้นจะทำงานที่ไคลเอนต์เลย ทำให้จำเป็นต้องมีการจำกัดความสามารถของภาษาสคริปต์ มิฉะนั้นแล้วอาจมีผู้ไม่ประสงค์ดีเขียนโปรแกรมที่อาจทำอันตรายแก่ไคลเอนต์ที่ดาวน์โหลดเพจไปใช้งานได้

จากสาเหตุข้างต้นนั้น ทำให้ภาษาสคริปต์ฝั่งไคลเอนต์มีความสามารถไม่มากนัก ส่วนใหญ่นักพัฒนาเว็บจะใช้ภาษาสคริปต์ในการสร้างลูกเล่นเพื่อทำให้หน้าตาของเว็บไซต์มีความน่าสนใจมากขึ้น

แต่ภาษาสคริปต์ในฝั่งเซิร์ฟเวอร์นั้นจะแตกต่างกัน เนื่องจากภาษาสคริปต์ที่ฝั่งเซิร์ฟเวอร์นั้นถูกใช้ในการสร้างเพจเอชทีเอ็มแอล ที่ประกอบด้วยข้อมูลตามที่ผู้ใช้ต้องการ และภาษาสคริปต์ก็มีการทำงานอยู่ที่เซิร์ฟเวอร์ ดังนั้นความสามารถของภาษาสคริปต์ที่ฝั่งเซิร์ฟเวอร์จึงมีความสามารถมากกว่า เช่น สามารถเชื่อมต่อกับระบบฐานข้อมูล และอาจสามารถเชื่อมต่อกับเซิร์ฟเวอร์เครื่องอื่น ๆ ได้ด้วย

2.2 ไคลเอนต์ - เซิร์ฟเวอร์ โมเดล

ไคลเอนต์ - เซิร์ฟเวอร์ เป็นการอธิบายความสัมพันธ์ระหว่างโปรแกรมคอมพิวเตอร์สองโปรแกรม โดยโปรแกรมหนึ่งเป็นไคลเอนต์ จะสร้างคำร้องบริการจากอีกโปรแกรมหรือเซิร์ฟเวอร์ แนวคิดนี้เป็นแนวคิดสำคัญในระบบเครือข่ายแบบจำลองไคลเอนต์ - เซิร์ฟเวอร์ให้แบบแผนการติดต่อภายในโปรแกรมที่ให้ประสิทธิภาพการกระจายข้ามตำแหน่งที่ต่างกัน

แบบจำลองไคลเอนต์ - เซิร์ฟเวอร์จะมีเซิร์ฟเวอร์หนึ่งเครื่อง และคอยคำร้องขอของ ไคลเอนต์ โดยปกติโปรแกรมไคลเอนต์หลายโปรแกรมใช้บริการเซิร์ฟเวอร์ร่วมกัน ซึ่งมักจะเป็นส่วนหนึ่งของโปรแกรมประยุกต์ขนาดใหญ่ เมื่อเปรียบเทียบกับอินเทอร์เน็ต เว็บเบราว์เซอร์ของผู้ใช้โปรแกรมเป็นไคลเอนต์ที่ขอบริการจากเว็บเซิร์ฟเวอร์ ในทางเทคนิคเรียกว่า Hypertext Transfer Protocol Server ซึ่งเป็นคอมพิวเตอร์อีกเครื่องบนอินเทอร์เน็ต

เครือข่ายแบบไคลเอนต์ - เซิร์ฟเวอร์เหมาะกับระบบเครือข่ายที่ต้องการเชื่อมต่อกับเครื่องไคลเอนต์จำนวนมาก โดยการรองรับจำนวนเครื่องไคลเอนต์อาจมากถึงหลักพัน เพราะฉะนั้นเครื่องที่จะนำมาทำหน้าที่ให้บริการจะต้องเป็นเครื่องที่มีประสิทธิภาพสูงเนื่องจากต้องสามารถทนทานต่อความผิดพลาด (Fault Tolerance) และต้องคอยให้บริการทรัพยากรให้กับเครื่องลูกข่ายตลอดเวลา สำหรับองค์กรขนาดใหญ่อาจมีการติดตั้งหลาย ๆ เซิร์ฟเวอร์เพื่อแบ่งภาระการทำงาน

2.2.1 ไคลเอนต์

เป็นโปรแกรมที่ถูกรันอยู่บนเครื่องของผู้ใช้เพื่อเรียกใช้บริการจากเซิร์ฟเวอร์ ไคลเอ็นต์จะเปิดช่องทางสื่อสารกับเซิร์ฟเวอร์โดยใช้ไอพีแอดเดรส และหมายเลขพอร์ตของเซิร์ฟเวอร์ ซึ่งเรียกวิธีแบบนี้ว่า active open เมื่อช่องทางการสื่อสารเปิดออกแล้วไคลเอ็นต์สามารถส่งคำร้องและรับบริการจากเซิร์ฟเวอร์ได้

2.2.2 เซิร์ฟเวอร์

เป็นโปรแกรมที่ถูกรันอยู่บนเครื่องที่จะคอยให้บริการกับเครื่องอื่น ๆ เมื่อมีการรันโปรแกรม และมีการเปิดช่องทางการสื่อสารเอาไว้เพื่อให้ไคลเอ็นต์สามารถติดต่อเข้ามาได้ แต่จะยังไม่ให้บริการใดๆ จนกระทั่งมีคำร้องขอมาจากไคลเอ็นต์ เราเรียกวิธีการแบบนี้ว่า passive open โดยตัวอย่างประเภทของเซิร์ฟเวอร์มีดังต่อไปนี้

2.2.2.1 เว็บ เซิร์ฟเวอร์ (Web Server)

คือเครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรมคอมพิวเตอร์ที่ทำหน้าที่ให้บริการข้อมูลในรูปแบบสื่อผสมผ่านระบบเครือข่ายโดยสามารถแสดงผลผ่านโปรแกรมอินเตอร์เน็ตบราวเซอร์ทางด้านของผู้ขอใช้บริการ

2.2.2.2 เมล์ เซิร์ฟเวอร์ (Mail Server)

คือเครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรม Mail Server เพื่อที่ให้บริการรับ – ส่งจดหมายอิเล็คทรอนิคส์ (E-Mail) หากองค์กรได้ทำการขอจดทะเบียนโดเมนนั้นแล้ว ก็สามารถใช้ Mail Server ภายใน องค์กร ทำหน้าที่ ส่งจดหมายอิเล็คทรอนิคส์ภายในองค์กรและภายนอกได้โดยไม่จำกัด สามารถสร้างผู้ใช้อีเมล์ขึ้นเองได้อย่างไม่จำกัดภายใต้ชื่อโดเมนของตนเอง

2.2.2.3 ดาต้าเบส เซิร์ฟเวอร์ (Database Server)

คือเครื่องคอมพิวเตอร์ที่ทำหน้าที่ให้บริการจัดเก็บข้อมูลต่าง ๆ ด้วยระบบฐานข้อมูล

2.2.2.4 โดเมน เนม เซิร์ฟเวอร์ (Domain Name Server)

คือเครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรมสำหรับทำหน้าที่จัดการชื่อต่าง ๆ ของ Server ทำให้สามารถเรียกใช้งานได้สะดวกรวดเร็วขึ้น

2.2.2.5 ดีเอชซีพี เซิร์ฟเวอร์ (DHCP Server)

คือเครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรม ที่ทำหน้าที่แจกจ่าย IP Address (หมายเลขประจำเครื่องในระบบเครือข่าย) ไปยังเครื่องลูกข่ายในเวลาที่เครื่องลูกข่ายเปิดใช้งานช่วยลดการกำหนดค่าคุณสมบัติต่างๆ ที่เครื่องลูกข่าย

2.2.2.6 พร็อกซี เซิร์ฟเวอร์ (Proxy Server)

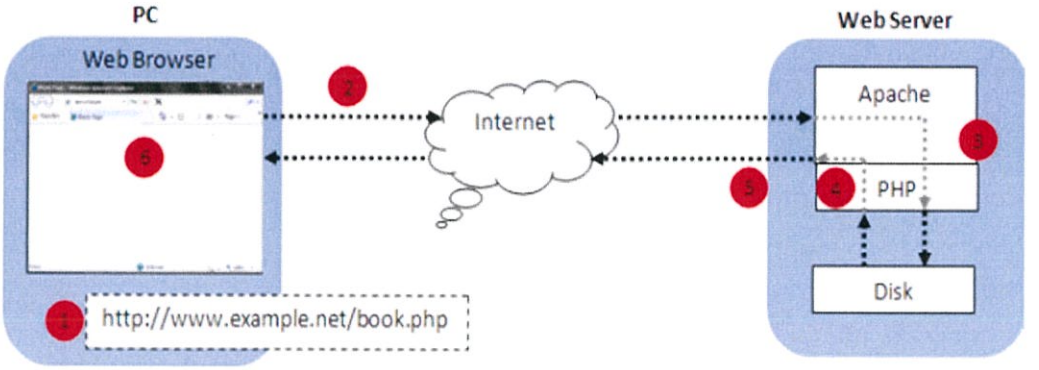
คือเครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรม Proxy ไว้สำหรับเก็บบันทึกการเรียกใช้บริการ www ทำให้สามารถเรียกใช้บริการจาก อินเตอร์เน็ตได้รวดเร็วขึ้นเนื่องจากว่า ได้เรียกใช้ภายในระบบเดียวกัน ซึ่งมีความเร็วสูงกว่าการเรียกใช้งานผ่านอินเตอร์เน็ตโดยตรง ช่วยลดปัญหาการสื่อสารผ่านอินเตอร์เน็ตลงได้ ทำให้ความเร็วในการเรียกใช้บริการอินเตอร์เน็ตสูงขึ้น และ ยังสามารถกำหนดสิทธิของ ผู้ใช้งานภายในเครือข่ายได้ เช่น จำกัดสิทธิในการใช้บริการ บางเว็บไซต์หรือจะต้องมีรหัสจึงจะสามารถใช้งานได้

2.3 ภาษาพีเอชพี

พีเอชพี (PHP: PHP Hypertext Preprocessor) เป็นภาษาสคริปต์ฝั่งเซิร์ฟเวอร์ซึ่งจะถูกประมวลผลบนเซิร์ฟเวอร์เป็นเว็บเพจ แล้วส่งเว็บเพจแสดงผลบนเบราว์เซอร์และสามารถเขียนรวมกับ เอชทีเอ็มแอล, จาวาสคริปต์ และภาษาสาหรับเว็บอื่น ๆ

2.3.1 โครงสร้างการทำงานของพีเอชพี

เว็บไซต์นั้นแบบออกเป็น 2 ประเภทตามลักษณะของการประมวลข้อมูล คือ เว็บไซต์แบบสถิต (Static) และ เว็บไซต์แบบพลวัต (Dynamic) สำหรับเว็บไซต์แบบสถิตนั้นเป็นเว็บไซต์ที่มีข้อมูลเป็นเอกสารเอชทีเอ็มแอลเพียงอย่างเดียว ส่วนเว็บไซต์แบบพลวัต นั้น นอกจากเอกสารเอชทีเอ็มแอลแล้ว ยังมีการเพิ่มโค้ดของโปรแกรมที่เป็นภาษาสคริปต์ที่ฝั่งเซิร์ฟเวอร์อย่างเช่น พีเอชพี, เจเอสพี หรือ เอเอสพี เข้าไปด้วย ซึ่งช่วยเพิ่มความสามารถในการทำงาน เช่น การรับส่งข้อมูลจากผู้ใช้หรือการติดต่อฐานข้อมูล เป็นต้น



รูป 2.8 ลำดับขั้นตอนการทำงานของเว็บไซต์แบบพลวัต

1) เริ่มต้นโดยเมื่อเราต้องการเปิดดูเว็บไซต์ เราจะเปิดโปรแกรมเรียกดูเว็บขึ้นมาก่อน เช่น Internet Explorer จากนั้นพิมพ์ยูอาร์แอลของเว็บไซต์ที่ต้องการ ซึ่งในตัวอย่างคือ http://www.example.net/book.php
2) โปรแกรมเรียกดูเว็บจะส่งข้อมูลไปยังเว็บไซต์ www.example.net เพื่อขอข้อมูล ในที่นี้คือไฟล์ book.php
3) เว็บเซิร์ฟเวอร์ที่ทำงานอยู่บนเว็บไซต์ปลายทาง (ในที่นี้คือ Apache) ได้รับข้อมูลและตรวจสอบพบว่าไฟล์ที่ต้องการเป็นไฟล์พีเอชพี จึงส่งให้พีเอชพีเอนจิน (PHP engine) ทำการประมวลผล
4) พีเอชพีเอนจินอ่านไฟล์ข้อมูล book.php และทำการประมวลผลข้อมูลตามโค้ดที่เขียนไว้ในไฟล์นี้ (ในขั้นตอนนี้อาจมีการประมวลผลและติดต่อกับบริการอื่น ๆ เช่น ติดต่อกับฐานข้อมูลหรือติดต่อกับไดเรกทอรี่เซอร์วิส เป็นต้น) จากนั้นจึงส่งผลลัพธ์ที่ได้ซึ่งอยู่ในรูปแบบของเอชทีเอ็มแอลกลับไปยัง Apache
5) Apache ส่งผลลัพธ์กลับไปให้โปรแกรมเรียกดูเว็บที่ฝั่งไคลเอนต์
6) เมื่อโปรแกรมเรียกดูเว็บที่ได้รับข้อมูลซึ่งเป็นเอกสารเอชทีเอ็มแอลก็จะแสดงผลลัพธ์ให้ผู้ชม

2.3.2 ความรู้พื้นฐานภาษาพีเอชพี

2.3.2.1 การเปิด-ปิดแท็กพีเอชพี

การเริ่มต้นเขียนภาษาพีเอชพีนั้นจะต้องมีการกำหนดขอบเขตส่วนของภาษาพีเอชพีไว้ในเพจ เพราะเพจหนึ่งอาจมีหลายภาษาประกอบกัน จึงต้องทำการแบ่งส่วนให้ชัดเจนเพื่อปกป้องไม่ให้อินเทอร์พรีเตอร์ของแต่ละภาษาเกิดความสับสนในการนำชุดคำสั่งไปแปลภาษา ซึ่งรูปแบบการเปิด-ปิดแท็กนั้นมี 4 รูปแบบ คือ

ตาราง 2.1 รูปแบบแท็กแสดงส่วนของภาษาพีเอชพี

รูปแบบแท็ก	เปิดแท็กพีเอชพี	ปิดแท็กพีเอชพี
แบบมาตรฐาน	<?php	?>
แบบสั้น	<?	?>
แบบ ASP	<%	%>
แบบ Script	<script language="PHP">	</script>

2.3.2.2 รูปแบบคำสั่ง (statement)

เมื่อเขียนคำสั่งของภาษาพีเอชพีแล้วจะต้องปิดท้ายด้วยเครื่องหมายอัฒภาค (;) ทุกครั้งเพื่อบ่งบอกถึงการสิ้นสุดคำสั่ง

โปรแกรม 2.1 โค้ดแสดงผลลัพธ์ "Hello world"

```
<HTML>
<BODY>
<?php
echo "Hello, World";
?>
</BODY>
</HTML>
```

2.3.2.3 ตัวแปร (Variable)

การประกาศตัวแปรของภาษาพีเอชพีนั้นมีกฎการตั้งชื่อที่คล้ายคลึงกับภาษาโปรแกรมมิงอื่น ๆ ซึ่งจะต้องเริ่มต้นด้วยอักษรภาษาอังกฤษ หรือเครื่องหมายเส้นใต้อักขระ (Underscore) แต่ตัวแปรของภาษาพีเอชพีนั้นต้องมีเครื่องหมายสกุลเงินดอลลาร์ ($) นำหน้าด้วย

2.3.2.4 หมายเหตุ (Comment)

หมายเหตุมีไว้สำหรับเขียนคำอธิบายคำสั่ง หรือชุดคำสั่ง โดยส่วนที่เป็นหมายเหตุจะไม่ถูกนำไปประมวลผลโดนอินเทอร์พรีเตอร์

2.3.2.5 การกำหนดค่าให้ตัวแปร (variable initialization)

โดยปกติแล้วการกำหนดค่าให้แก่ตัวแปรจะกำหนดผ่านเครื่องหมายเท่ากับ (=) ซึ่งมีทั้งการกำหนดค่าให้ตัวเลขและสายอักขระ โดยที่การกำหนดค่าให้ตัวแปรนั้นสามารถทำได้พร้อมกับการประกาศตัวแปรทันที

โปรแกรม 2.3 การกำหนดค่าให้ตัวแปร

```
<?php
$total = 10;
?>
// Difference between single quote and double quote
<php
$total = 10;
$example1 = 'The total is $total';
$example2 = "The total is $total";
echo $ example1
echo $ example2
// Output of $example1 is The total is $total
// Output of $example1 is The total is 10
?>
```

2.3.2.6 การเชื่อมสายอักขระ (String concatenation)

สามารถเชื่อมต่อสายอักขระได้มากกว่าหนึ่งสายด้วยเครื่องหมายมหัพภาค (.)

โปรแกรม 2.4 การเชื่อมต่อสายอักขระ

```
<php
$a = 'apples';
$b = 'bananas';
$c = '';
$c = $a . ' and ' . $b;
?>
```

2.3.2.7 อักขระต้องห้าม (Escaped characters)

อักขระต้องห้าม คือ อักขระพิเศษที่ทำหน้าที่ต่างจากอักขระทั่วไป วิธีการใช้งานจะต้องนำหน้าอักขระเหล่านี้ด้วยเครื่องหมายทับย้อนหลัง (\)

ตาราง 2.2 อักขระต้องห้าม

Character	Escaped Character	Description
ไม่มี	\n	ป้อนบรรทัด
ไม่มี	\r	ปัดแคร่
ไม่มี	\t	เพิ่มย่อหน้า
\	\\	แสดงเครื่องหมาย \
$	\$	แสดงเครื่องหมาย $
"	\"	แสดงเครื่องหมาย "

2.3.2.8 แถวลำดับ (Array)

คือตัวแปรชนิดหนึ่งที่สามารถเก็บค่าได้หลายค่าในตัวแปรเดียวกัน การสร้างตัวแปรแถวลำดับให้เรียกใช้ฟังก์ชั่น array() ซึ่งจะถูกชี้ตำแหน่งโดยคีย์ โดยปกติคีย์จะมีค่าเป็นตัวเลข หากตัวแปรแถวลำดับใด ๆ มีคีย์เป็นสายอักขระจะเรียกตัวแปรแถวลำดับนั้นว่าแถวลำดับเชื่อมโยง (Associative array) ซึ่งมีลักษณะคล้ายโครงสร้างข้อมูลแบบพจนานุกรม

โปรแกรม 2.5 การประกาศและกำหนดค่าให้ตัวแปรแถวลำดับ

```
$World = array(1 => "moon",2 => "sun");
$Computer = new array();
$Computer[0] = "Mouse";
$Computer['cpu'] = "Processor";
```

2.3.2.9 ตัวดำเนินการ (Operator)

ตัวดำเนินการใช้สัญลักษณ์ต่าง ๆ ในการกำหนด ซึ่งมีทั้งการกำหนดค่าให้แก่ตัวแปร, การคำนวณทางคณิตศาสตร์, การเปรียบเทียบเชิงตรรกศาสตร์ เป็นต้น

ตาราง 2.3 ตัวดำเนินการต่าง ๆ ในภาษาพีเอชพี

Operator	ความหมาย
==	เท่ากับ
!=,<>	ไม่เท่ากับ
<,>	น้อยกว่า,มากกว่า
<=,>=	น้อยกว่าหรือเท่ากับ,มากกว่าหรือเท่ากับ

2.3.2.10 คำสั่งควบคุม (Control structure)

คำสั่งควบคุมใช้ในการตรวจสอบเงื่อนไขต่าง ๆ รวมถึงการทำซ้ำเพื่อให้สอดคล้องและทำงานตามแนวคิดของผู้เขียนโปรแกรม คำสั่งควบคุมมีหลายชนิด เช่น if - else condition, for loop และ while loop แต่คำสั่ง for each loop จะอ่านตัวแปรแถวลำดับทุกส่วนย่อยแบบเรียงลำดับโดยไม่ต้องมีเงื่อนไข หากในคำสั่งควบคุมมีรูปแบบคำสั่ง (statement) มากกว่า 1 คำสั่ง จะต้องครอบด้วยเครื่องหมายปีกกาเปิดและปีกกาปิด

2.3.2.11 ฟังก์ชั่น (Function)

ฟังก์ชั่นมีไว้สำหรับเก็บรูปแบบคำสั่งที่ใช้งานบ่อยครั้ง เพื่อให้ง่ายต่อการเรียกใช้งานและเก็บรูปแบบคำสั่งให้เป็นระเบียบ ลดจำนวนบรรทัดของโค้ด ซึ่งมีผลให้ขนาดของไฟล์ลดลงได้

2.4 ภาษาเอสคิวแอล

ภาษาเอสคิวแอล (SQL: Structured Query Language) เป็นภาษามาตรฐานที่ใช้ติดต่อกับระบบฐานข้อมูลเชิงสัมพันธ์ (RDBMS: Relational Database Management System) ซึ่งระบบฐานข้อมูลในปัจจุบันเกือบทั้งหมดเป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ เช่น Oracle, Sybase, Microsoft SQL Server, Microsoft Access, DB2 เป็นต้น ถึงแม้ว่าระบบจัดการฐานข้อมูลเชิงสัมพันธ์ทุกระบบจะสามารถใช้คำสั่งพื้นฐานของเอสคิวแอลเหมือนกัน แต่ทุกระบบอาจจะมีคำสั่งพิเศษของตัวเองที่ระบบอื่นอาจจะใช้ไม่ได้

ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) เป็นโครงสร้างฐานข้อมูลที่สามารถกำหนดความสัมพันธ์ขึ้นเมื่อใดก็ได้ โดยที่ไม่จำเป็นต้องกำหนดไว้ตั้งแต่เริ่มต้นออกแบบฐานข้อมูล และเปลี่ยนโครงสร้างได้ง่าย ฐานข้อมูลเก็บข้อมูลและสารสนเทศในออบเจคเรียกว่า ตาราง (Table)

ตารางประกอบด้วยสดมภ์ (Column) และแถวข้อมูล (Row) โดยสดมภ์ประกอบด้วยชื่อสดมภ์ ประเภทข้อมูล และคุณสมบัติอื่น ๆ (เช่น การกำหนดเป็นคีย์หลัก, ค่าเริ่มต้น) ส่วนแถวเก็บข้อมูลของสดมภ์

เอสคิวแอลสามารถแบ่งคำสั่งออกเป็น 4 กลุ่ม คือ

1) Data Manipulate (DML) เป็นคำสั่งจัดการข้อมูล ได้แก่ INSERT, UPDATE, DELETE, ROLLBACK, COMMIT
2) Data Definition (DDL) เป็นคำสั่งจัดการกับไฟล์ในฐานข้อมูล ได้แก่ CREATE, ALTER, DROP
3) Query เป็นคำสั่งการเรียกดูข้อมูล คือ SELECT
4) Data Control เป็นคำสั่งจัดการความปลอดภัย

2.4.1 เพิ่มข้อมูล (INSERT INTO)

เพิ่มระเบียนหนึ่ง หรือหลายระเบียนลงในตาราง

โปรแกรม 2.8 ไวยากรณ์ของการเพิ่มข้อมูล

```
INSERT INTO target [(field1[, field2[, ...]])]
VALUES (value1[, value2[, ...])
```

ตาราง 2.4 ฟิลด์ต่าง ๆ ของการเพิ่มข้อมูล

ส่วน	คำอธิบาย
target	ชื่อของตารางหรือแบบสอบถามที่ต้องการผนวกข้อมูลเข้า
field1, field2	ชื่อของเขตข้อมูลที่ต้องการผนวกข้อมูลเข้าถ้าตามหลังอาร์กิวเมนต์ target หรือชื่อของเขตข้อมูลที่ต้องการจะนำข้อมูลมาถ้าตามหลังอาร์กิวเมนต์ source
source	ชื่อของตารางหรือแบบสอบถามที่ต้องการคัดลอกระเบียนมา
value1, value2	ค่าที่ต้องการจะแทรกลงในเขตข้อมูลที่ระบุของระเบียนใหม่ แต่ละค่าจะถูกแทรกลงในเขตข้อมูลที่ตรงกับตำแหน่งของค่าในรายการ เช่น value1 จะถูกแทรกลงใน field1 ของระเบียนใหม่ และ value2 จะถูกแทรกลงในลงใน field2 และเป็นเช่นนี้เรื่อย ๆ คุณจะต้องคั่นระหว่างค่าต่าง ๆ ด้วยเครื่องหมายจุลภาค (,) และใส่ขอบเขตข้อมูลข้อความไว้ในเครื่องหมายอัญประกาศเดี่ยว (' ')

2.4.2 เลือกข้อมูล (SELECT)

เป็นคำสั่งที่ใช้สำหรับการเรียกดูข้อมูลในตาราง คำสั่งเอสคิวแอล SELECT สามารถเรียกได้ทั้งตาราง หรือสามารถระบุฟิลด์ที่ต้องการเรียกดูข้อมูลได้

โปรแกรม 2.10 ไวยากรณ์ของการเลือกข้อมูล

```
SELECT Column1, Column2, Column3,... FROM [Table-Name]
```

2.4.3 การระบุเงื่อนไข (WHERE)

เป็นคำสั่งที่ใช้สำหรับการระบุเงื่อนไขการเลือกข้อมูลในตาราง คำสั่งเอสคิวแอล WHERE สามารถระบุเงื่อนไขในการเลือกข้อมูลได้ตั้งแต่หนึ่งเงื่อนไขเป็นต้นไป

โปรแกรม 2.11 ไวยากรณ์ของการระบุเงื่อนไข

```
SELECT Column1, Column2, Column3,... FROM Table-Name
WHERE [Field] = 'Value' ;
```

2.4.4 เรียงลำดับข้อมูล (ORDER BY)

คำสั่งเอสคิวแอล ORDER BY เป็นคำสั่งที่ใช้สำหรับการระบุเงื่อนไขในการเลือกข้อมูลในตาราง โดยจะจัดเรียงข้อมูลตามต้องการ

โปรแกรม 2.12 ไวยากรณ์ของการเรียงลำดับข้อมูล

```
SELECT Column1,Column2,Culumn3,... FROM [Table-Name]
ORDER BY [Field] [ASC/DESC],[Field]
[ASC/DESC],...
```

2.4.5 ปรับปรุงข้อมูล (UPDATE)

คำสั่งเอสคิวแอล UPDATE เป็นคำสั่งที่ใช้สำหรับแก้ไขข้อมูลในตาราง โดยสามารถทำการแก้ไขได้หลายฟิลด์และหลายระเบียน ภายในคำสั่งหนึ่งคำสั่ง ทั้งนี้ขึ้นอยู่กับเงื่อนไขที่ผู้ใช้ได้เขียนขึ้น

โปรแกรม 2.13 ไวยากรณ์ของการปรับปรุงข้อมูล

```
UPDATE table SET newvalue WHERE criteria;
```

ตาราง 2.5 ฟิลด์ต่าง ๆ ของการปรับปรุงข้อมูล

ส่วน	คำอธิบาย
table	ชื่อของตารางมีข้อมูลที่คุณต้องการจะปรับเปลี่ยน
newvalue	นิพจน์ที่ระบุค่าที่จะถูกแทรกลงในบางเขตข้อมูลในระเบียนที่มีการปรับปรุง
criteria	นิพจน์ที่กำหนดระเบียนที่จะถูกปรับปรุง ระเบียนที่เป็นไปตามนิพจน์เท่านั้นที่จะถูกปรับปรุง

2.4.6 ลบข้อมูล (DELETE)

คำสั่งเอสคิวแอล DELETE เป็นคำสั่งที่ใช้สำหรับลบข้อมูลในตาราง โดยสามารถทำการลบได้หลายระเบียนภายในคำสั่งเดียว ทั้งนี้ขึ้นอยู่กับเงื่อนไขที่ผู้ใช้เขียนขึ้นด้วย

โปรแกรม 2.14 ไวยากรณ์ของการลบข้อมูล

```
DELETE [table.*] FROM table
WHERE criteria
```

ตาราง 2.6 ฟิลด์ต่าง ๆ ของการลบข้อมูล

ส่วน	คำอธิบาย
table	ชื่อของตารางระเบียนจะถูกลบออกซึ่งไม่จำเป็นต้องมีก็ได้
table	ชื่อของตารางระเบียนจะถูกลบออก
criteria	นิพจน์ที่กำหนดว่าระเบียนใดบ้างที่จะถูกลบ

2.5 โครงร่างพีเอชพี (PHP Framework)

2.5.1 โครงร่าง (Framework)

โครงร่าง คือ ชุดของไลบรารี (Libraries) หรือคลาส (Classes) สำหรับใช้ในการพัฒนาโปรแกรม โดยมีโครงสร้างของโปรแกรมที่เป็นมาตรฐาน โดยที่ภายในโครงร่างนั้น จะมีโค้ดที่นำกลับมาใช้ใหม่ได้อยู่เป็นจำนวนมาก ซึ่งจะทำให้ประหยัดเวลาในการทำแอพพลิเคชันหนึ่ง ๆ ได้อย่างมาก เนื่องจากไม่จำเป็นต้องเขียนโค้ดเดิมซ้ำ ๆ ทุกครั้งที่พัฒนาแอพพลิเคชันใหม่ หรือแอพพลิเคชันที่ทำงานคล้าย ๆ กัน จึงควรเก็บโค้ดซ้ำ ๆ ไว้เป็นฟังก์ชั่นในไฟล์ย่อย ๆ แล้วใช้วิธีการรวม (include) ไฟล์เข้ามาใช้งาน ซึ่งเป็นหลักการของโครงร่าง

2.5.2 โค้ดอิกไนเตอร์ (CodeIgniter)

เป็นโครงร่างพีเอชพี ซึ่งพัฒนาขึ้นโดย Rick Ellis ใช้รูปแบบการเขียนโปรแกรมแบบเอ็มซีวี (MCV: Model-View-Controller) ซึ่งจะแยกเอาส่วนของการแสดงผลออกจากส่วนประมวลผล ทำให้โค้ดที่ได้มีความเป็นระเบียบเป็นระบบและง่ายต่อการบำรุงรักษา เนื่องจากตัวโครงร่างเองนั้นมาพร้อมกับฟังก์ชั่น หรือ คลาสที่อยู่ในรูปของไลบารีช่วยเหลือ (library helper) หรือส่วนเสริม (plugin) ซึ่งสิ่งเหล่านี้มีส่วนช่วยให้นักพัฒนาใช้เวลาในการพัฒนาระบบลดลงสามารถศึกษาทำความเข้าใจได้ในระยะเวลาสั้น ๆ

โค้ดอิกไนเตอร์ถูกสร้างบนพื้นฐานการพัฒนาแบบเอ็มซีวี ซึ่งเป็นวิธีการทางซอฟต์แวร์ที่แยกระบบการประมวลผลของแอพพลิเคชันออกจากส่วนแสดงผล ในทางปฏิบัติ มันอนุญาตให้หน้าเว็บมีสคริปต์เล็ก ๆ เนื่องจากส่วนแสดงต้องแยกออกจากสคริปต์พีเอชพี

สามารถอธิบายรูปแบบการเขียนโปรแกรมแบบเอ็มซีวีได้ดังต่อไปนี้

2.5.2.1 โมเดล (Model)

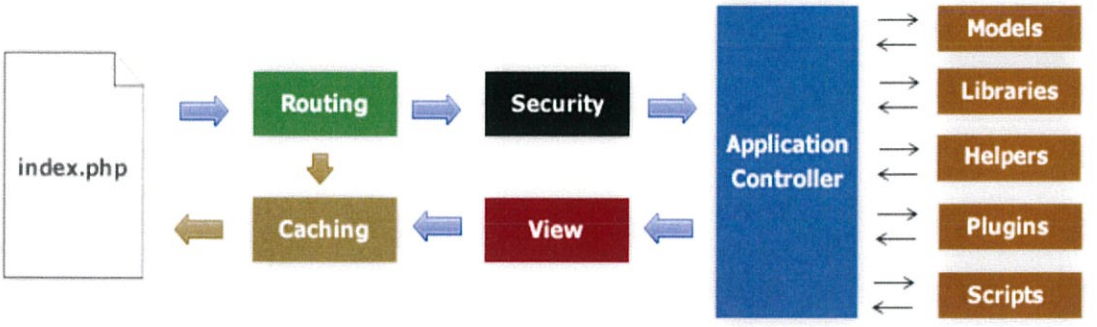
อธิบายโครงสร้างข้อมูล แบบฉบับของคลาสแบบจำลองจะมีฟังก์ชั่นที่ช่วยให้สามารถ รับ, ใส่, ปรับปรุงข้อมูล ในฐานข้อมูล

2.5.2.2 วิว (View)

คือข้อมูลที่ถูกแสดงให้กับผู้ใช้ ส่วนใหญ่จะเป็นเว็บเพจ แต่ในโค้ดอิกไนเตอร์, วิว สามารถเป็นเพียงบางส่วนของหน้าได้ เช่น ส่วนหัว ส่วนท้ายได้

2.5.2.3 คอลโทรลเลอร์ (Controller)

ถูกใช้เป็นสื่อกลางระหว่างโมเดล, วิว และแหล่งที่มาอื่น ๆ ที่ต้องการในการทำงานร้องขอเอชทีทีพี (HTTP request) และการสร้างเว็บเพจ



รูป 2.9 การไหลของข้อมูลในสถาปัตยกรรมแบบเอ็มซีวี

สามารถอธิบายขั้นตอนการไหลของข้อมูลของสถาปัตยกรรมแบบเอ็มซีวีได้ดังนี้

1) หน้า index.php เป็นตัวควบคุมส่วนหน้า, สร้างทรัพยากรพื้นฐานที่ต้องการในการรันโค้ดอิกไนเตอร์
2) ส่วน Routing ทำการตรวจสอบการร้องขอเอชทีทีพีกำหนดว่าควรจะทำอะไร
3) ถ้ามีไฟล์แคช (Cache) อยู่ Caching จะส่งข้อมูลกลับไปยังเบราว์เซอร์ทันที
4) ก่อนที่จะโหลดตัวควบคุมของแอพพลิเคชัน (Application Controller) การร้องขอเอชทีทีพีและผู้ใช้ใด ๆ ที่ส่งข้อมูลมาจะถูกกรองข้อมูลที่ Security เพื่อความปลอดภัย
5) โหลดแบบจำลอง (Models), ไลบรารี่หลัก (Libraries), ส่วนเสริม (Plugins), ผู้ช่วย (Helpers) และทรัพยากรอื่น ๆ ที่จำเป็นในการทำงานที่ถูกร้องขอมาผ่านคอนโทรลเลอร์
6) สุดท้าย View ปฏิบัติงานและถูกส่งกลับไปยังเบราว์เซอร์เพื่อโชว์หน้าจอ ถ้าระบบแคชถูกใช้งาน หน้าจอจะถูกแคชก่อนแล้วจึงค่อยส่งสิ่งที่ร้องขอมาเป็นลำดับถัดไป

2.5.3 ซีไอไซต์ (CI Site)

ภายในโฟลเดอร์ application จะมีโฟลเดอร์ย่อยที่ทำหน้าที่ต่างกันอีกหลายโฟลเดอร์ ในหัวข้อก่อนหน้านี้ เราได้ทำการศึกษาหน้าที่การทำงานของโฟลเดอร์ controller, config และ views กันแล้ว สำหรับในหัวข้อนี้จะเป็นการศึกษาชนิดของไฟล์ หรือคลาส และหน้าที่การทำงาน ของโฟลเดอร์อื่น ๆ ซีไอไซต์ในทางเทคนิคแล้ว เราจะสามารถเขียนโค้ดพีเอชพีแล้วบันทึกไว้ใน

โฟลเดอร์ใดก็ได้ภายในซีไอไซต์ แต่โค้ดอิกไนเตอร์ได้ทำการสร้างโครงสร้างโฟลเดอร์ไว้เพื่อทำหน้าที่ที่แตกต่างกัน เพื่อความสะดวกในการพัฒนา ดูแลรักษา และการปรับปรุงซีไอไซต์

ตัวอย่างที่จะกล่าวต่อไปนี้ จะเป็นการเรียกใช้งานส่วนของโค้ดที่เรียกว่า "display" โดยอาจจะเรียกใช้งานผ่านฟังก์ชัน "mainpage" โดยมี 4 วิธีการที่แตกต่างกันแต่ให้ผลลัพธ์เหมือนกัน ได้แก่การสร้างไฟล์ของโค้ดอิกไนเตอร์เป็นประเภทต่าง ๆ ดังนี้ ได้แก่ model, library, helper และ plug-in

ตาราง 2.7 ประเภทไฟล์และวิธีการใช้ไฟล์ต่าง ๆ ในซีไอไซต์

ประเภทไฟล์	วิธีการใช้

Model	เป็นคลาสไฟล์ (ตามหลักการเขียนโปรแกรมแบบเชิงวัตถุ)
	• การ Load : $this->load->model('display');
	• การใช้ : $this->display->mainpage();
	• คลาสไฟล์นี้ต้องเป็นคลาสที่สืบทอดมาจาก model (โดยเริ่มต้นด้วย class Display extends Model)
	• คลาสจะต้องมี constructor ที่มีส่วนของโค้ดดังนี้
	function display()
	{
	parent::Model();
	}
	และมีฟังก์ชันที่ชื่อ mainpage()
	Function mainpage(){}
	• โดยหลักการ Model เป็นคลาสที่ถูกออกแบบมาเพื่อทำงานกับฐานข้อมูล

ตาราง 2.7 ประเภทไฟล์และวิธีการใช้ไฟล์ต่าง ๆ ในซีไอไซต์ (ต่อ)

ประเภทไฟล์	วิธีการใช้

Library	• Library มี 2 โฟลเดอร์คือ ในโฟลเดอร์ system/libraries และ application/libraries
	• การ Load : $this->load->library('display');
	• การใช้ : $this->display->mainpage();
	• ไฟล์ library เป็นคลาสไฟล์ (ไม่ต้องสืบทอดคลาส Model และ ไม่จำเป็นต้องมี constructor)
	class Display()
	{
	function mainpage(){
	// code here
	}
	}
	• libraries จะเป็นที่สาหรับเก็บโค้ดที่เราเขียนขึ้นเพื่อทำงานเฉพาะด้าน

plug-in	plug-in มี 2 โฟลเดอร์คือ ในโฟลเดอร์ system/helper และ application/helper ไฟล์ plug-in จะเป็นแบบ procedural ไม่ใช่แบบ OOP • การ Load : $this->load->plugin('display'); • การใช้โดยเรียกฟังก์ชันโดยตรง: mainpage(); • ชื่อไฟล์ต้องเพิ่ม _pi.php คือ display_pi.php • mainpage() ต้องเป็นฟังก์ชัน ไม่ใช่คลาส • ข้อแตกต่างระหว่าง helper และ plug-in คือ helper เป็นที่รวบรวม ฟังก์ชันหลาย ๆ ฟังก์ชัน แต่ plug-in เป็นที่เก็บฟังก์ชันเดียวต่อหนึ่งไฟล์

ไฟล์ต่าง ๆ ที่กล่าวถึงข้างต้นจะต้องไม่สามารถเรียกใช้งานได้โดยตรง ข้อสังเกตคือ helpers, plug-in และ libraries มี 2 ชุดเพื่อประโยชน์ดังนี้

1) ในโฟลเดอร์ system เป็นส่วนหนึ่งของโค้ดซีไอคอร์ (CI Core) จะถูกแชร์ให้ application ทั้งหมด สามารถทำการอัพเดทโค้ดอิกไนเตอร์เป็นเวอร์ชันใหม่โดยการเขียนทับ (overwrite) โฟลเดอร์ system
2) ในโฟลเดอร์ application จะใช้งานได้เฉพาะ application นั้น ๆ เท่านั้น ถ้าต้องการอัพเดทเวอร์ชันของโค้ดอิกไนเตอร์เป็นเวอร์ชันใหม่ โฟลเดอร์ application จะไม่ถูกเขียนทับ

3) เมื่อเราทำการโหลด helper, plug-in หรือ library ของโค้ดอิกไนเตอร์จะไปค้นหาไฟล์จากทั้งสองที่โดยอัตโนมัติ โดยทำการค้นหาไฟล์ในโฟลเดอร์ system ก่อน ถ้าไม่เจอไฟล์ก็จะไปหาในโฟลเดอร์ application เป็นลำดับถัดไป
4) จากข้อ 3 ทำให้สามารถ Overwrite Core, library, helpers และ plug-in ได้โดยการสร้างไฟล์ที่มีชื่อไฟล์เหมือนกัน แต่เก็บในโฟลเดอร์ application

ตาราง 2.8 โฟล์เดอร์ต่าง ๆ ในซีไอไซต์

โฟล์เดอร์	คำอธิบาย
Config	เก็บข้อมูลพื้นฐานเกี่ยวกับซีไอไซต์
Controllers	ส่วนควบคุม
Errors	เก็บไฟล์แจ้งเตือนข้อผิดพลาด (Error) ที่เกิดขึ้น
Hooks	ควบคุมการโหลดไฟล์
Libraries	เก็บไฟล์ไลบรารี
Models	เก็บไฟล์ส่วนของการทำงาน
Views	เก็บไฟล์ส่วนการแสดงผล
Cache	ใช้เก็บไฟล์แคช
Codeigniter	เก็บไฟล์ระบบ
Database	ไฟล์ไลบรารีสำหรับการทำงานกับฐานข้อมูล
Fonts	เก็บฟอนต์
Helpers	ไฟล์ผู้ช่วยระบบ
Plugins	เก็บไฟล์ส่วนเสริม
Scaffolding	ไลบรารีที่ช่วยในการจัดการกับฐานข้อมูล

โค้ดอิกไนเตอร์เป็นโครงร่างพีเอชพีที่มีขนาดเล็ก ใช้งานง่าย มีรูปแบบการเขียน โปรแกรมแบบเอ็มซีวี ซึ่งจะแยกเอาส่วนของการแสดงผลออกจากส่วนประมวลผล ทำให้โค้ดที่ได้ มีความเป็นระเบียบ เป็นระบบและง่ายต่อการพัฒนา ช่วยให้ผู้เรียนรู้สามารถศึกษาทำความเข้าใจได้ในระยะเวลาอันสั้น เพราะมีคู่มือช่วยในการศึกษา ไม่ว่าจะเป็นการติดตั้ง การแนะนำ การใช้งานรูปแบบการเขียนโค้ดต่าง ๆ เป็นต้น และเนื่องจากตัวโครงร่างมาพร้อมกับฟังก์ชั่น หรือคลาสที่อยู่ในรูปของไลบรารีผู้ช่วย (library helper) หรือส่วนเสริม (plugin) จึงมีส่วนช่วยในการเขียนและพัฒนาโปรแกรมโดยใช้เวลาในการพัฒนาระบบลดลง

2.6 จาวาสคริปต์ (JavaScript)

จาวาสคริปต์นั้นเป็นภาษาสคริปต์เชิงวัตถุ (Object-Oriented) ภาษาหนึ่งสำหรับการเขียนโปรแกรมเพื่อนำเสนอความสามารถ รวมถึงลูกเล่นของเว็บไซต์ผ่านทางเว็บ โดยสามารถเพิ่มคำสั่งจาวาสคริปต์ลงไปได้เพื่อใช้ในประโยชน์ในด้านต่าง ๆ สำหรับจาวาสคริปต์นี้ใช้สำหรับการสร้างเว็บไซต์เพื่อให้มีลักษณะพลวัต กล่าวคือ สามารถตอบโต้กับผู้ใช้ได้ทันที อีกทั้งยังต้องมีความเกี่ยวข้องกับไฟล์ที่เป็นเอชทีเอ็มแอลอีกด้วย ดังนั้นเพื่อให้การเขียนจาวาสคริปต์เป็นไปได้ด้วยดี ผู้เขียนควรที่จะมีความรู้เกี่ยวกับภาษาเอชทีเอ็มแอลด้วย

2.6.1 ลักษณะการทำงานของจาวาสคริปต์

1) นาวิเกเตอร์จาวาสคริปต์ (Navigator JavaScript) เป็นจาวาสคริปต์ฝั่งลูกข่าย (Client-Side JavaScript) ซึ่งหมายถึงจาวาสคริปต์ที่ถูกแปลทางฝั่งไคลเอนต์ จึงมีความเหมาะสมต่อการใช้งานของผู้ใช้ทั่วไปเป็นส่วนใหญ่
2) ไลฟ์ไวร์จาวาสคริปต์ (LiveWire JavaScript) เป็นจาวาสคริปต์ฝั่งแม่ข่าย (Server-Side JavaScript) ซึ่งหมายถึงจาวาสคริปต์ที่ถูกแปลทางฝั่งเซิร์ฟเวอร์ สามารถใช้ได้เฉพาะกับไลฟ์ไวร์ของเน็ตสเคปโดยตรง

2.6.2 รูปแบบของจาวาสคริปต์

การเขียนจาวาสคริปต์เป็นการเขียนโปรแกรมเสริมเข้าไปในโปรแกรมที่เป็นโครงสร้างหลักของเว็บเพจที่เขียนด้วยเอชทีเอ็มแอล เพื่อช่วยให้การทำงานมีความสามารถมากขึ้นจากที่เคยทำได้ด้วยเอชทีเอ็มแอล การใช้จาวาสคริปต์จะช่วยทำให้การทำเว็บเพจหรือโปรแกรมเชิงวัตถุมีความสามารถมากขึ้น

2.6.3 ความสามารถของจาวาสคริปต์

1) จาวาสคริปต์ทำให้สามารถเขียนโปรแกรมแบบง่าย ๆ ได้โดยไม่ต้องพึ่งภาษาอื่น เช่น พีเอชพี โดยส่วนใหญ่จะเป็นรูปแบบของการแสดงผล
2) จาวาสคริปต์มีคำสั่งที่ตอบสนองกับผู้ใช้งาน เช่น เมื่อผู้ใช้คลิกที่ปุ่ม สามารถสั่งให้เปิดหน้าใหม่ได้ ทำให้เว็บไซต์มีปฏิสัมพันธ์กับผู้ใช้งานมากขึ้น
3) จาวาสคริปต์สามารถเขียนหรือเปลี่ยนแปลงองค์ประกอบของเอชทีเอ็มแอล (HTML Element) ได้ สามารถเปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้
4) จาวาสคริปต์มีความสามารถในการตรวจสอบ เปรียบเทียบ ตัดสินใจ ประมวลผลและสามารถสร้างฟังก์ชันเองได้
5) จาวาสคริปต์สามารถใช้ในการตรวจสอบผู้ใช้ได้ เช่น ตรวจสอบเว็บเบราว์เซอร์ของผู้ใช้
6) จาวาสคริปต์มีลักษณะเป็นอินเทอร์พรีเตอร์ (Interpreter) แบบไฟล์ข้อความฝังอยู่ในเอกสารเอชทีเอ็มแอล ทำให้สามารถทำงานบนเว็บเบราว์เซอร์ได้ทันที

2.6.4 ข้อจำกัดของจาวาสคริปต์

1) ไม่สามารถติดต่อ หรือทำงานบนฝั่งของเครื่องแม่ข่ายได้
2) ไม่สามารถสร้างส่วนต่อประสาน (Interface) ของเว็บแอพลิเคชันได้โดยตรง ต้องใช้งานร่วมกับโปรแกรมภาษาอื่น

2.7 เอแจ็กซ์ (AJAX)

เอแจ็กซ์ (AJAX: Asynchronous JavaScript and XML) ไม่ใช่ชื่อของการเขียนโปรแกรมหรือเป็นชื่อของภาษาที่ใช้ในการโปรแกรม แต่เป็นชุดของเทคโนโลยีต่าง ๆ มาจากการทำงานร่วมกันของจาวาสคริปต์ (JavaScript) และเอ็กเอ็มแอล (XML) แบบไม่เข้าจังหวะ (Asynchronous) มีหลักการทำงาน 2 ประเด็นคือ การอัพเดทหน้าจอแบบบางส่วน และการติดต่อสื่อสารกับเซิร์ฟเวอร์โดยใช้หลักการไม่เข้าจังหวะ ทำให้ผู้ใช้ไม่ต้องหยุดการทำงานเพื่อรอการประมวลผลจากเซิร์ฟเวอร์ รวมถึงการโหลดและการรีเฟรชหน้าจอของเบราว์เซอร์ทางฝั่งไคลเอนต์มีการใช้เอแจ็กซ์ โดยการเพิ่มเลเยอร์ระหว่างเบราเซอร์ผู้ใช้กับเซิร์ฟเวอร์ ทำให้ผู้ใช้สามารถทำงานได้โดยไม่ต้องรอให้ไคลเอนต์ติดต่อไปยังเซิร์ฟเวอร์รวมถึงการโหลดและการรีเฟรชหน้าจอทั้งหมดด้วย ดังนั้นผู้ใช้จึงสามารถใช้งานแอพพลิเคชันได้อย่างมีประสิทธิภาพมากขึ้น

เอแจ็กซ์เอนจิน (AJAX engine) ทำหน้าที่เป็นตัวกลางระหว่างไคลเอนต์และเซิร์ฟเวอร์ฉะนั้นเมื่อไคลเอนต์มีต้องการร้องขอ แทนที่จะการร้องขอส่งเอชทีทีพี (HTTP request) ไปยังเซิร์ฟเวอร์โดยตรงไคลเอนต์จะส่งจาวาสคริปต์คอล (JavaScript call) ไปเอแจ็กซ์เอนจิน เพื่อโหลดข้อมูลผู้ใช้ต้องการ และหากเอแจ็กซ์เอนจินต้องการข้อมูลเพิ่มเติมในการตอบสนองต่อผู้ใช้ เอแจ็กซ์เอนจินจะส่งการร้องขอไปยังเซิร์ฟเวอร์โดยใช้เอ็กเอ็มแอล โดยส่วนประกอบจะเป็นขั้นพื้นฐานที่ขาดไม่ได้ในเอแจ็กซ์ ได้แก่ เอชทีเอ็มแอล, เอ็กเอชทีเอ็มแอล, ดีโอเอ็ม (DOM) และจาวาสคริปต์

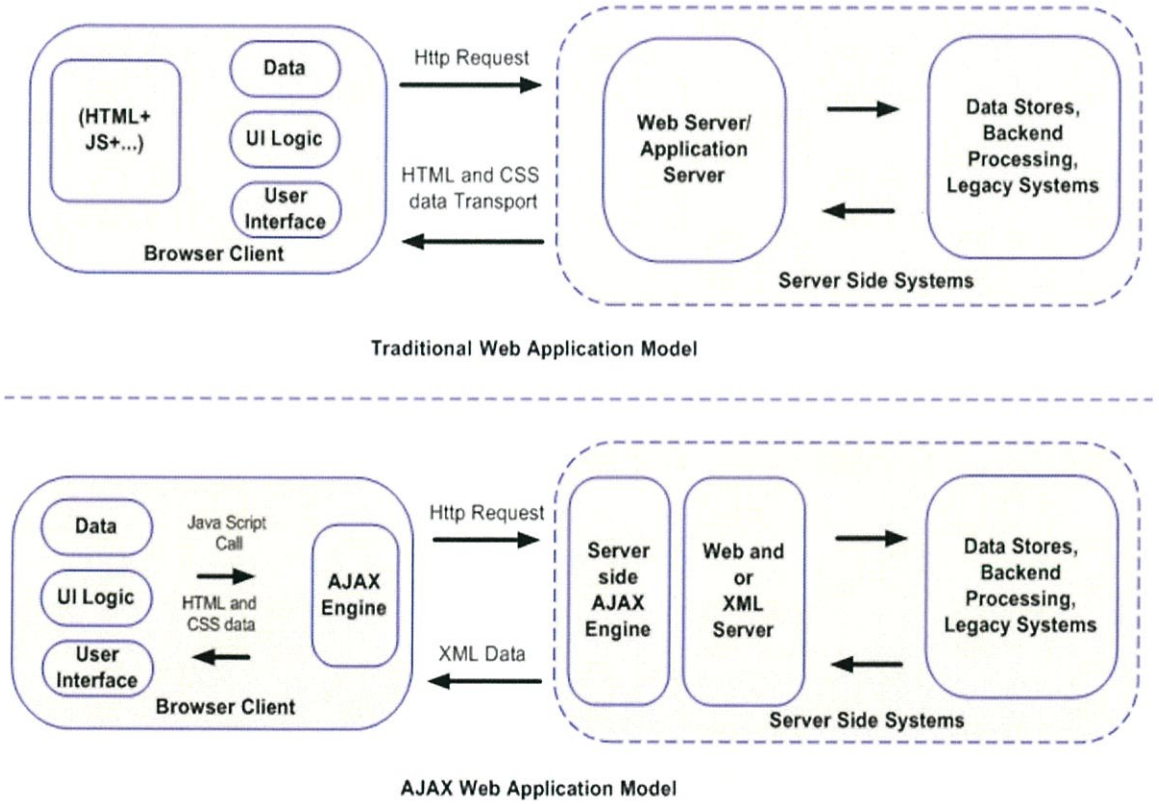
2.7.1 ที่มาของปัญหา

เนื่องจากแอพพลิเคชันที่ใช้งานในปัจจุบันนี้ มีหลักการที่ทำงานแล้วเกิดการสูญเสียเวลาและทรัพยากรของผู้ใช้ในการรอคอยการทำงานต่าง ๆ ทำให้ผู้ใช้ต้องหยุดคอย ดังนั้นการทำงานของผู้ใช้จึงเป็นไปอย่างไม่ต่อเนื่อง ซึ่งหลักการดังกล่าวคือ

1) User interaction paradigm การที่เบราว์เซอร์ตอบสนองต่อการทำงานของผู้ใช้ โดยจะทิ้งหน้าเว็บที่แสดงอยู่ในขณะนั้น แล้วไปทำการส่งการร้องขอส่งเอชทีทีพีกลับไปยังเซิร์ฟเวอร์แทน ซึ่งทำให้ผู้ใช้ไม่สามารถทำอะไรได้ นอกจากการรอคอย เมื่อเซิร์ฟเวอร์ทำการประมวลเสร็จก็จะส่งหน้าเอชทีเอ็มแอลกลับมายังเบราว์เซอร์ ต่อจากนั้นเบราว์เซอร์ก็จะรีเฟรชและแสดงหน้าเอชทีเอ็มแอลหน้าใหม่ และเหตุนี้เองที่ทำให้ผู้ใช้สามารถใช้งานต่อไปได้ จะเห็นว่า ผู้ใช้มีช่วงเวลาของการหยุดรอคอยเป็น

เวลานานสำหรับการประมวลผลของเซิร์ฟเวอร์และการรีเฟรชหน้าเอชทีเอ็มแอลใหม่ทั้งหน้า ซึ่งเป็นสิ่งที่ไม่มีประสิทธิภาพในเชิงพลวัตของการทำงานบนเว็บแอพพลิเคชัน

2) Synchronous request and response communication mode การที่เบราว์เซอร์เริ่มที่การร้องขอข้อมูล และเซิร์ฟเวอร์ก็ตอบสนองเฉพาะการร้องขอที่เบราว์เซอร์ร้องขอมา เซิร์ฟเวอร์จะไม่สามารถส่งข้อมูลได้ถ้าเบราว์เซอร์ไม่ได้ร้องขอข้อมูลในขณะนั้น ซึ่งถือว่าเป็นการติดต่อสื่อสารเป็นแบบทิศทางเดียว วงจรการร้องขอและตอบสนองแบบเข้าจังหวะ (Synchronous) คือ การทำงานแบบประสานจังหวะระหว่างเบราว์เซอร์กับเซิร์ฟเวอร์ ทำให้เกิดความล่าช้าในการทำงานทำให้ผู้ใช้ทำอะไรไม่ได้อีก นอกจากการคอยการตอบสนองกลับมาจากเซิร์ฟเวอร์เมื่อเซิร์ฟเวอร์ประมวลผลเสร็จ

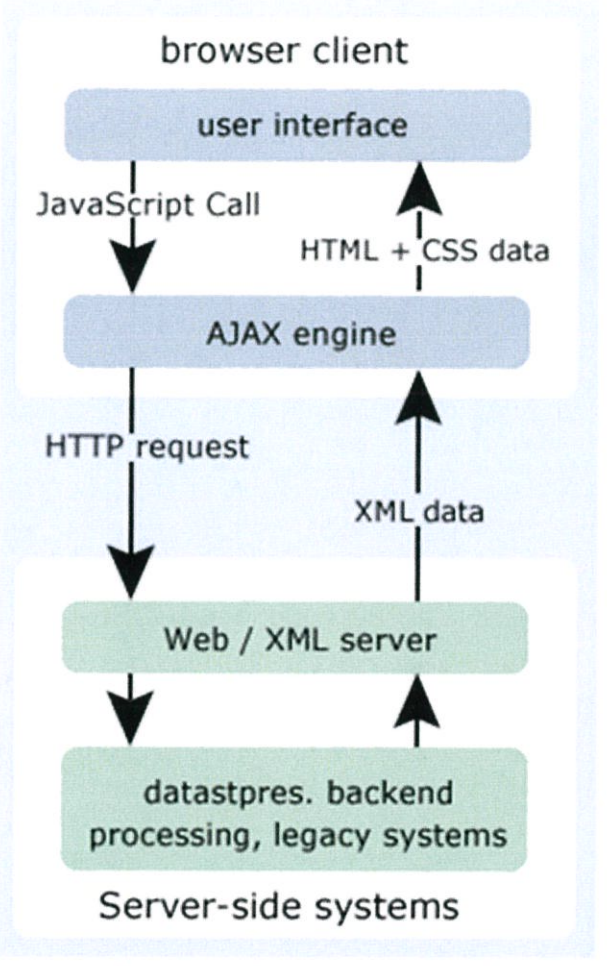


รูป 2.10 การเปรียบเทียบรูปแบบเว็บแอพพลิเคชันแบบปกติและเอแจ็กซ์

2.7.2 โครงสร้างของเอแจ็กซ์

มุมมองของโครงสร้างทางซอฟต์แวร์ของเอแจ็กซ์ต่างจากเว็บแอพพลิเคชันในทุกวันนี้ เนื่องจากมีการเพิ่มเอนจินทางฝั่งไคเอนต์ เอแจ็กซ์เอนจินนี้ อยู่ส่วนติดต่อผู้ใช้ (User interface) กับเซิร์ฟเวอร์ซึ่งจะมองว่าเป็นการทำงานที่ไคลเอนต์การทำงานต่าง ๆ ของผู้ใช้ โปรแกรมจะไปเรียกเอ

แจ็กซ์เอนจินตัวนี้ขึ้นมา แทนที่การร้องขอหน้าเว็บจากเซิร์ฟเวอร์โดยตรง และจะใช้โครงสร้างข้อมูลแบบเอ็กเอ็มแอลในการขนย้ายข้อมูลระหว่างเซิร์ฟเวอร์กับเอแจ็กซ์เอนจิน เมื่อเบราว์เซอร์ทำการร้องขอข้อมูลจากเซิร์ฟเวอร์ นอกจากนี้เอแจ็กซ์เอนจินไม่ต้องทำการติดตั้ง ไม่ใช้ส่วนเสริม (plugin) และไม่สามารถดาวน์โหลด (download) ได้ เพราะเอแจ็กซ์ เป็นแนวคิดในการแก้ปัญหาการหยุดชะงักการทำงานของผู้ใช้



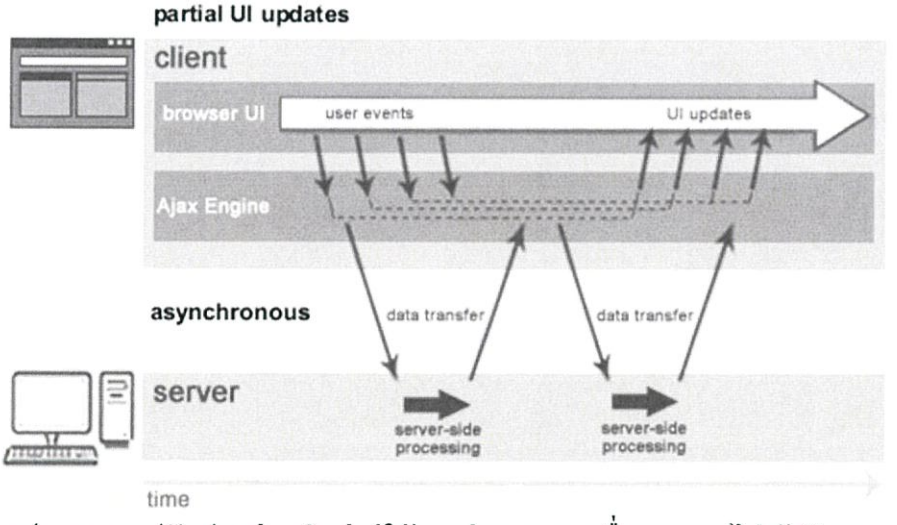
รูป 2.11 สถาปัตยกรรมของเอแจ็กซ์

2.7.3 การทำงานของเอแจ็กซ์

เอแจ็กซ์จะช่วยลดการติดต่อระหว่างไคลเอนต์กับเซิร์ฟเวอร์โดยในการโหลดหน้าเว็บนั้น เบราว์เซอร์จะโหลดข้อมูลเอแจ็กซ์เอนจิน แทนการร้องขอข้อมูลจากเซิร์ฟเวอร์โดยตรง นอกจากเอแจ็กซ์จะทำหน้าที่ทั้งการแสดงส่วนติดต่อกับผู้ใช้และติดต่อไปยังเซิร์ฟเวอร์แล้ว เอแจ็กซ์เอนจินอนุญาตให้การกระทำต่าง ๆ ในเว็บแอพพลิเคชันเป็นแบบไม่เข้าจังหวะ คือความเป็นอิสระในการติดต่อไปยังเซิร์ฟเวอร์นั่นเอง

การอัพเดทหน้าจอบางส่วน แทนที่การ "click, wait, and refresh" ระหว่างที่เกิด การทำงานแบบการติดต่อสื่อสารของผู้ใช้ ส่วนติดต่อผู้ใช้ที่ต้องนำมาแสดงซ้ำในหน้าเว็บที่ร้องขอไปยังเซิร์ฟเวอร์จะถูกจัดเป็นข้อมูลใหม่เมื่อถูกอัพเดทแล้ว การหยุดชะงักส่วนติดต่อผู้ใช้จึงไม่เกิดขึ้น เพราะหน้าเว็บนั้นยังคงถูกแสดงอยู่และสามารถใช้งานได้โดยปราศจากการหยุดชะงักการทำงานของผู้ใช้ การอัพเดทหน้าเว็บบางส่วนสามารถทำให้หน้าเว็บทำงานต่อไปได้

การติดต่อแบบไม่เข้าจังหวะ เข้ามาแทนที่การ “synchronous request and response model” สำหรับเอแจ็กซ์ การร้องขอและตอบสนองจะทำแบบไม่เข้าจังหวะ ซึ่งคือการติดต่อสื่อสารกับเซิร์ฟเวอร์แบบอิสระโดยทำการลดการติดต่อระหว่างเบราว์เซอร์ กับเซิร์ฟเวอร์ผลที่ได้ก็คือผู้ใช้สามารถใช้งานเว็บแอพพลิเคชันได้ในขณะที่ไคลเอนต์ทำการร้องขอข้อมูลจากเซิร์ฟเวอร์อยู่เบื้องหลัง (การทำงานแบบพร้อมกันแต่มองเป็น 2 ฝั่ง เช่นหน้าร้านกับหลังร้าน) เมื่อข้อมูลเดินทางมาถึงเบราว์เซอร์ ก็จะอัพเดทส่วนติดต่อผู้ใช้ที่ต้องการข้อมูลใหม่ ส่วนติดต่อผู้ใช้ที่ไม่ต้องการอัพเดทก็จะแสดงส่วนนั้นต่อไป



รูป 2.12 การปรับปรุงส่วนติดต่อผู้ใช้บางส่วนและการสื่อสารแบบไม่เข้าจังหวะ

2.7.4 ข้อดีของเอแจ็กซ์

1) ตอบสนองต่อผู้ใช้ได้อย่างรวดเร็วเนื่องจากการอัพเดทแบบบางส่วน
2) ผู้ใช้ไม่ต้องหยุดรอคอยการประมวลของเซิร์ฟเวอร์เนื่องจากการติดต่อแบบ Asynchronous
3) รองรับกับเว็บเบราว์เซอร์หลัก ๆ ที่สามารถใช้จาวาสคริปต์ได้
4) ทำให้การประมวลผลที่เซิร์ฟเวอร์มีความรวดเร็วขึ้นเนื่องจากการประมวลผลที่เซิร์ฟเวอร์ลดลง

5) ไม่ต้องทำการติดตั้ง หรือใช้ส่วนเสริม
6) ไม่ยึดติดกับแพลทฟอร์ม (Platform) หรือภาษาที่ใช้ในการเขียนโปรแกรม
7) เป็นเทคโนโลยีใหม่ที่ไม่ได้เป็นของนักพัฒนาเว็บแอพพลิเคชันคนใด กล่าวคือทุกคนมีสิทธิ์เข้ามาพัฒนาแอพพลิเคชันตัวนี้

2.8 คอมไพเลอร์ (Compiler)

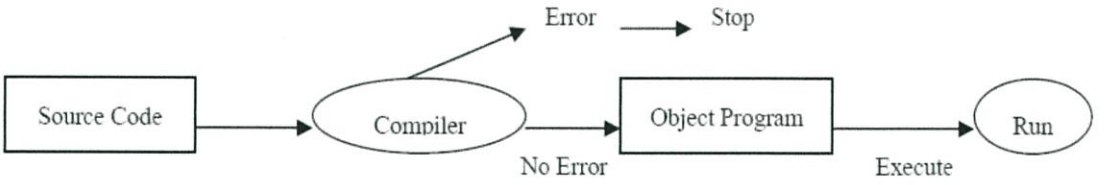
2.8.1 ความหมายของคอมไพเลอร์

คอมไพเลอร์ เป็นโปรแกรมแปลภาษาที่ทำหน้าที่แปลภาษาระดับสูงมาเป็นภาษาเป้าหมาย (Object Program) ซึ่งอาจหมายถึง ภาษาเครื่อง หรือภาษาแอสแซมบลี หากเกิดข้อผิดพลาดเกิดขึ้น คอมไพเลอร์จะหยุดทำการแปลเพื่อให้ผู้ใช้งานทำการแก้ไขข้อผิดพลาด แต่ถ้าไม่มีข้อผิดพลาดก็จะทำงานต่อไปจนจบ และเกิดภาษาเป้าหมายขึ้น เพื่อสื่อสารกับเครื่องคอมพิวเตอร์ต่อไป

ในการพัฒนาซอฟต์แวร์คอมพิวเตอร์นั้น โปรแกรมเมอร์ (Programmer) จะเขียนโปรแกรมในภาษาคอมพิวเตอร์แบบต่าง ๆ ตามความชำนาญของแต่ละคน โปรแกรมที่ได้จะเรียกว่าโปรแกรมต้นฉบับ (Source Code) ซึ่งมนุษย์จะอ่านโปรแกรมต้นฉบับนี้ได้ แต่เครื่องคอมพิวเตอร์จะไม่สามารถเข้าใจคำสั่งเหล่านั้น เนื่องจากคอมพิวเตอร์เข้าใจแต่ภาษาเครื่อง (Machine Language) ซึ่งมีลักษณะเป็นชุดของบิตที่สร้างขึ้นจากรหัสของระบบเลขฐานสอง (เลข 0 และ 1 เท่านั้น) ที่หมายถึงสถานะของไฟฟ้าที่มีสองสถานะ คือ เปิดและปิด นั่นทำให้การเขียนโปรแกรมเป็นเรื่องไม่สะดวก เนื่องจากจะต้องจารหัสคำสั่ง ผู้เขียนจะเข้าใจได้ยาก ทำการตรวจสอบหาข้อผิดพลาดของโปรแกรมได้ยาก ฉะนั้นจึงต้องอาศัยโปรแกรมตัวแปลภาษาคอมพิวเตอร์ (Translator) ในการแปลภาษาคอมพิวเตอร์ภาษาต่าง ๆ ไปเป็นภาษาเครื่อง โปรแกรมที่แปลจากโปรแกรมต้นฉบับ แล้วจะเรียกว่ารหัสจุดหมาย (Object Code) ซึ่งจะประกอบด้วย รหัสคำสั่งที่คอมพิวเตอร์สามารถเข้าใจและนำไปปฏิบัติต่อได้

2.8.2 การทำงานของคอมไพเลอร์

ถ้าต้องการรันโปรแกรม ต้องสั่งงานคอมไพเลอร์ให้ทำการแปลภาษานั้นแล้วทำการ เอ็กซิคิวต์ (Execute) โปรแกรมต้นฉบับ



รูป 2.13 กระบวนการคอมไพล์โค้ดโปรแกรม

การทำงานของคอมไพเลอร์จะประกอบด้วย 5 ขั้นตอนหลัก

1) Lexical Analyzer เป็นขั้นตอนในการตรวจจับและจัดเรียงคำโปรแกรมต้นฉบับที่ต้องการแปลภาษา
2) Syntax Analyzer เป็นขั้นตอนในการตรวจสอบไวยากรณ์ของภาษา
3) Semantic Analyzer เป็นขั้นตอนในการตรวจสอบชนิดของข้อมูลที่จะนำมาประมวลผลและทำการเปลี่ยนรูปแบบของภาษาให้อยู่ในรูปของรูปแบบขั้นกลาง (Intermediate form) เพื่อรอการแปลงให้เป็นรหัสจุดหมาย
4) Code Generation เป็นการแปลรูปแบบขั้นกลางให้เป็นรหัสจุดหมาย ซึ่งส่วนใหญ่จะเป็นภาษาแอสเซมบลี
5) Code Optimization เป็นการลดขั้นตอนการทำงานที่ซ้ำซ้อนกันของภาษาเครื่อง เพื่อให้การประมวลผลโดยเครื่องคอมพิวเตอร์ทำงานได้เร็วขึ้น

2.8.3 โปรแกรมคอมพิวเตอร์

หลักการทำงานโดยทั่วไปของคอมพิวเตอร์จะต้องมีการป้อนข้อมูล ให้แก่คอมพิวเตอร์เสียก่อน แล้วคอมพิวเตอร์จึงจะสามารถทำการประมวลผล (Process) ตามโปรแกรมที่เราได้สร้างขึ้นมา เมื่อประมวลผลเสร็จ คอมพิวเตอร์จะนำผลลัพธ์ที่ได้ออกมาแสดงผลในรูปแบบต่าง ๆ ส่วนประกอบของโปรแกรมคอมพิวเตอร์ โดยทั่วไปจะมีส่วนประกอบ 3 ส่วนดังนี้

2.8.3.1 ข้อมูลนำเข้า (Input)

ส่วนที่คอมพิวเตอร์มีไว้สำหรับรับนำข้อมูลเข้ามาใช้ในการประมวลผล ในการเขียนโปรแกรมถ้าจะต้องมีการใช้ข้อมูลจากภายนอก ก็จะต้องมีการกำหนดหน่วยรับข้อมูล ไว้ในโปรแกรมเสียก่อน เช่น อาจจะเลือกใช้แป้นพิมพ์ (Keyboard) เป็นต้น

2.8.3.2 กระบวนการ (Process)

การเขียนชุดคำสั่งในการดำเนินงานของโปรแกรมเพื่อให้เครื่องคอมพิวเตอร์ดำเนินงานตามวัตถุประสงค์ของผู้เขียน ในส่วนนี้อาจจะมีการสร้างตัวแปรในรูปแบบต่าง ๆ ขึ้นมาเพื่อรองรับข้อมูลที่จะนำเข้ามาใช้ในการประมวลผล

2.8.3.3 ข้อมูลส่งออก (Output)

ส่วนที่คอมพิวเตอร์มีไว้สำหรับนำผลลัพธ์ที่ได้จากการประมวลผลออกมาแสดงผลหรือจัดเก็บ โดยจะต้องมีการกำหนดหน่วยที่เป็นข้อมูลส่งออกไว้ในส่วนของโปรแกรมเสียก่อน ตัวอย่างหน่วยข้อมูลส่งออก ที่เห็นได้ง่าย คือ จอภาพ

2.8.4 ไทนีซี คอมไพเลอร์ (Tiny C Compiler)

ในการพัฒนาโครงงานนี้ ได้เลือกใช้ทีซีซี (TCC: Tiny C Compiler) เพราะมีขนาดเล็กและสามารถทำงานได้ดีบนระบบปฏิบัติการวินโดว์ และทำงานได้ทั้งสถาปัตยกรรมหน่วยประมวลแบบ x86 และ x64 อีกทั้ง พีเอชพีฟังก์ชั่นในการเรียกใช้คำสั่งบนคอมมานด์พรอมท์ (command prompt) ได้โดยตรง คุณสมบัติที่โดดเด่นกว่าคอมไพเลอร์ตัวอื่น ๆ คือ สามารถคอมไพล์, ลิงค์ออบเจคโค้ดและได้โปรแกรมสกุล .exe ภายในคำสั่งเดียว

ทีซีซีมีคุณลักษณะอื่น ๆ อีก เช่น เมื่อโค้ดโปรแกรมที่นำไปคอมไพล์มีลักษณะเหมือนกันทุกประการ ทีซีซีใช้เวลาคอมไพล์น้อยกว่าจีซีซี (GCC) ถึง 9 เท่า, สามารถใช้ C dynamic library ได้โดยตรงและมีฟีเจอร์ (Feature) หลากหลายที่เป็นไปตามมาตรฐาน ISO/IEC 9899:1999 และ GNU C extensions และมีทุกฟีเจอร์ตามมาตรฐาน ANSI C extension

ในการคอมไพล์ด้วยคำสั่งทีซีซีสามารถทำได้โดยใช้คำสั่ง tcc source.c โดยที่ source.c คือ โปรแกรมต้นฉบับที่จะทำการคอมไพล์ หลังจากนั้นจะได้ไฟล์ source.exe ซึ่งเป็น executable file พร้อมนำไปรันได้ รูปแบบการใช้คำสั่งโดยทั่วไปของ TCC คือ tcc [options] [infile1 infile2...] [`-run' infile args...] และป้อนอินพุตเข้าโปรแกรมโดยคำสั่ง echo ผ่านเครื่องหมายไพป์ (|) โดยรูปแบบคำสั่ง คือ echo input1 input2 ... | source.exe ซึ่งการไพป์คือการนำผลลัพธ์ที่ได้จากโปรแกรมหนึ่งไปเป็นอินพุทของอีกโปรแกรมหนึ่ง

ตาราง 2.9 สรุปออปชันของทีซีซี

Option	คำอธิบาย
-v	แสดงเวอร์ชั่นปัจจุบันของทีซีซี
-c	สร้างออปเจคไฟล์ (object file)
-o outfile	ป้อนออปเจคไฟล์, executable หรือ dll ให้ “outfile”
-Bdir	กำหนดเส้นทาง (path) ที่ไลบรารีภายในของทีซีซี (tcc internal libraries) ถูกเก็บไว้
-bench	ส่งออกสถิติการคอมไพล์
-run source [args...]	คอมไพล์ไฟล์ “source” และรันด้วยอาร์กิวเมนต์แบบคอมมานด์ไลน์ “args”

2.9 นิพจน์ปรกติ (Regular expression)

นิพจน์ปกติ (regular expression) คือ สายอักขระ (string) ที่อธิบายถึงรูปแบบของสายอักขระตามโครงสร้างรูปแบบที่กำหนด นิพจน์ปกตินั้นมีการใช้อย่างแพร่หลายในโปรแกรมประเภท text editor ในการค้นหาและปรับเปลี่ยนข้อความ นิพจน์ปกติใช้อธิบายรูปแบบของสายอักขระ โดยการอธิบายจะกระชับและรัดกุมโดยไม่จำเป็นต้องอธิบายเป็นรายชื่อทั้งหมด เช่น "ex(pres|pan)sion" มีความหมายว่า expression หรือ expansion

ตาราง 2.10 การอธิบายสายอักขระ

เครื่องหมาย	คำอธิบาย
|	เครื่องหมายไพป์ (pipe) ใช้สำหรับสร้างทางเลือกที่จะใช้ค้นหานิพจน์ เช่น "express|expand" มีความหมายว่า express หรือ expand
()	เครื่องหมายนขลิขิต ใช้ในการกำหนดลาดับและขอบเขตของนิพจน์ เช่น "ex(pres|pan)sion" มีความหมายว่า expression หรือ expansion
?	เครื่องหมายปรัศนี ใช้บ่งบอกว่ามีนิพจน์ก่อนหน้านี้จำนวน 0 หรือ 1 นิพจน์) มีนิพจน์ หรือไม่มีก็ได้(เช่น " ปร?กติ " มีความหมายว่า ปกติ หรือ ปรกติ
*	เครื่องหมายดอกจัน ใช้บ่งบอกว่ามีนิพจน์ก่อนหน้านี้จำนวน 0 นิพจน์หรือมากกว่า เช่น "go*gle" มีความหมายว่า ggle gogle google goooogle เป็นต้น
+	เครื่องหมายบวก ใช้บ่งบอกว่ามีนิพจน์ก่อนหน้านี้จำนวน 1 นิพจน์หรือมากกว่า เช่น "go+gle" มีความหมายว่า gogle google goooogle เป็นต้น แต่ไม่เท่ากับ ggle
{n}	วงเล็บปีกกาใช้บ่งบอกว่าต้องมีนิพจน์ก่อนหน้านี้จำนวน n นิพจน์ เช่น "go{5}gle" มีความหมายว่า gooooogle เพียงตัวเดียว
{m,n}	วงเล็บปีกกาที่มีจุลภาคข้างใน ใช้บ่งบอกว่าต้องมีนิพจน์ก่อนหน้านี้จำนวน m นิพจน์ถึง n นิพจน์ เช่น "go{2,4}gle" มีความหมายว่า google goooogle และ goooogle หากไม่ระบุตัวเลข n จะหมายถึงต้องมีนิพจน์จำนวนตั้งแต่ m นิพจน์เป็นต้นไป
^	เครื่องหมายหมวก ใช้เขียนเป็นตัวแรกสุดของนิพจน์ปกติ เพื่อบ่งบอกว่านิพจน์ที่ต้องการค้นหาอยู่ต้นบรรทัดหรือต้นสตริงเท่านั้น
$	เครื่องหมายสกุลเงิน ดอลลาร์ ใช้เขียนเป็นตัวท้ายสุดของนิพจน์ปกติ เพื่อบ่งบอกว่านิพจน์ที่ต้องการค้นหาอยู่ท้ายบรรทัดหรือท้ายสตริงเท่านั้น

บทที่ 3
การวิเคราะห์และออกแบบระบบ

3.1 ความต้องการของระบบ และลักษณะสำคัญ

3.1.1 วิเคราะห์รูปแบบของโจทย์

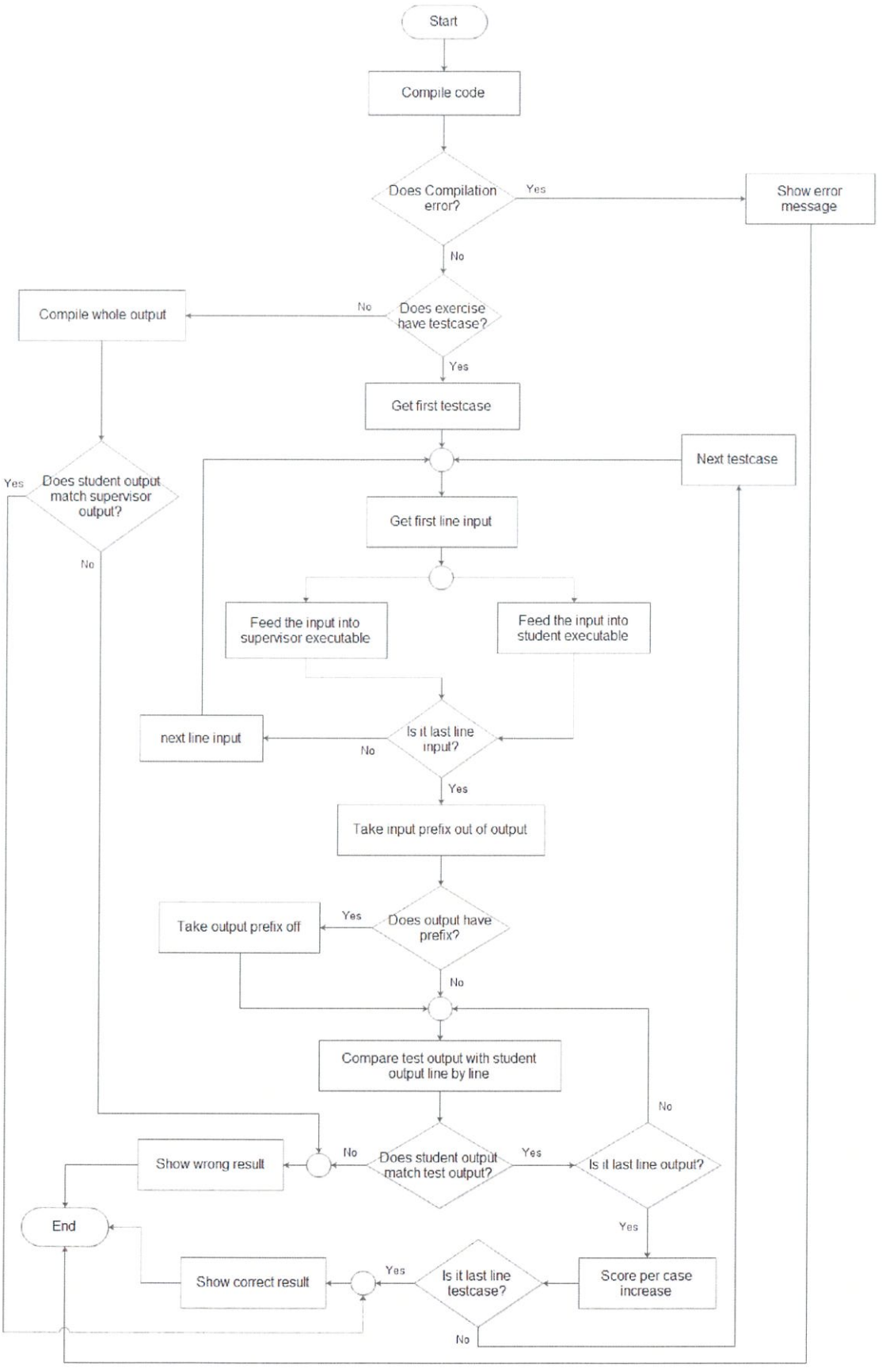
โจทย์ต่างๆ ที่อาจารย์จะมอบหมายให้นักศึกษาทำมีหลากหลายรูปแบบ ก่อนการออกแบบและสร้างระบบตรวจผลลัพธ์ของโปรแกรมควรต้องวิเคราะห์รูปแบบการป้อนอินพุทและส่งออกเอาท์พุตของแต่ละโจทย์ ซึ่งมีรูปแบบของโจทย์ดังนี้

1) โจทย์ที่ไม่ต้องการรับอินพุท เช่น ให้แสดงจำนวนระหว่าง 1 ถึง 10
2) โจทย์ที่รับอินพุทค่าเดียว เช่น ให้แสดงผลคูณระหว่าง 1 ถึง ตัวเลขที่รับเข้ามา
3) โจทย์ที่รับอินพุทมากกว่า 1 ค่า เช่น หาค่าที่น้อยที่สุดจากเลขจำนวนเต็มที่รับเข้ามา
4) โจทย์ที่ส่งออกเอาท์พุทค่าเดียว เช่น ให้หาค่าที่มากที่สุด
5) โจทย์ที่ส่งออกเอาท์พุทมากกว่า 1 ค่า เช่น ให้คำนวณพื้นที่แล้วเส้นรอบวงของรูปสี่เหลี่ยมจากความยาวและความกว้างที่ป้อนเข้ามา

3.1.2 ความต้องการของระบบตรวจผลลัพธ์ของโปรแกรม

ความต้องการของระบบตรวจผลลัพธ์ของโปรแกรมมีดังต่อไปนี้

1) ระบบต้องสามารถตรวจผลลัพธ์ของโปรแกรมภาษาซีได้
2) ตรวจโปรแกรมที่รันบนคอมมานด์พรอมท์ (command prompt) ได้
3) สามารถตรวจความถูกต้องผลลัพธ์ของโปรแกรมของนักศึกษาด้วยการเทียบผลการรันกับผลลัพธ์จากโค้ดโปรแกรมตัวอย่างที่อาจารย์ส่งเข้าระบบ
4) ระบบต้องสามารถตรวจจับข้อผิดพลาด (error) และแจ้งผลการตรวจโปรแกรมได้
5) ระบบมีการแสดงผลการตรวจหลังจากส่งโค้ดโปรแกรมทันที โดยหลังจากส่งงานแล้วจะมีหน้าต่างแสดงผลการตรวจขึ้นมา โดยมีรายละเอียด คือ ผลลัพธ์ของโปรแกรมจากโค้ดของนักศึกษาแต่ละกรณีทดสอบ, ชื่อ-นามสกุล และรหัสประจำตัวของนักศึกษา, วันเวลาที่ตรวจ, ผลการคอมไพล์, คะแนนเต็ม, คะแนนที่ได้, หน่วยความจำที่ใช้
6) สามารถตรวจโปรแกรมที่มีหรือไม่มีอินพุท และมีหลายเอาท์พุทได้
7) ระบบต้องสามารถตรวจสอบความถูกต้องของนามสกุลซอร์สโค้ดโปรแกรมที่นักศึกษาส่งเข้ามาในระบบได้

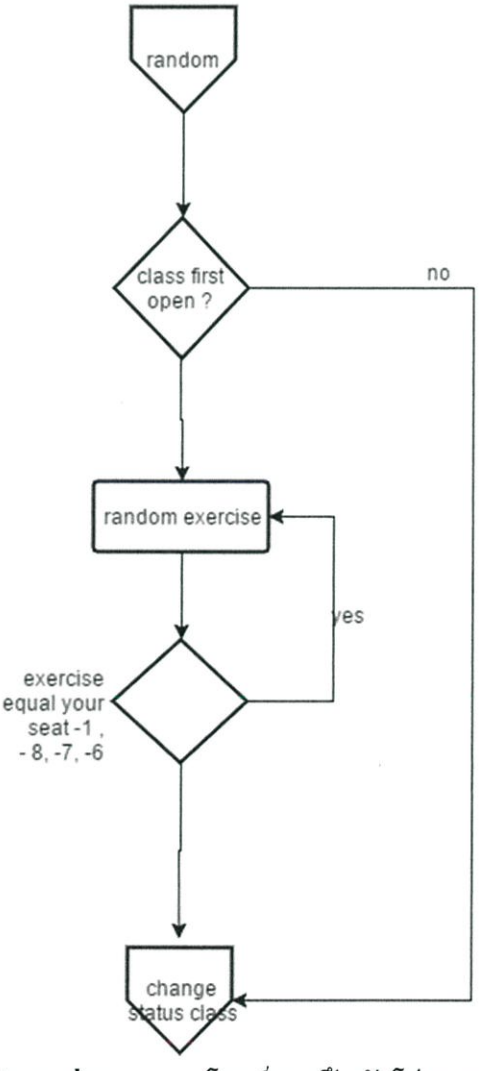


รูป 3.1 แผนภาพการทำงานส่วนตรวจโปรแกรมและให้คะแนน

3.1.3 ความต้องการของระบบแจกโจทย์อัตโนมัติของโปรแกรม

ความต้องการของระบบแจกโจทย์อัตโนมัติของโปรแกรมมีดังต่อไปนี้

1) ระบบต้องสามารถสุ่มและแจกโจทย์แบบฝึกหัดให้นักศึกษาแต่ละคนได้โดยอัตโนมัติ เมื่ออาจารย์ได้ทำการเปิดแลบให้นักศึกษาทำ
2) นักเรียนที่นั่งอยู่ตำแหน่งติดกัน จะต้องได้โจทย์แบบฝึกหัดที่แตกต่างกัน เพื่อป้องกันการคัดลอกคำตอบของโจทย์โปรแกรมข้อเดียวกัน
3) ระบบต้องสามารถสุ่มโจทย์แบบฝึกหัดจากคลังโจทย์ทั้งหมดที่อาจารย์ผู้สอนสร้างไว้ได้ตรงกับเนื้อหาที่นักศึกษาจะต้องใช้ทำแลบในแต่ละครั้ง



รูป 3.2 แผนภาพการทำงานส่วนการแจกโจทย์แบบฝึกหัดโปรแกรมมิ่งให้แก่นักศึกษา

3.2 รายการตรวจสอบความต้องการของระบบ (System Requirement Checklist)

3.2.1 ข้อมูลนำเข้า (input)

1) นักศึกษาต้องอัพโหลดไฟล์โค้ดโปรแกรมภาษาซีเข้าสู่ระบบเพื่อทำการตรวจและให้คะแนน
2) นักศึกษาสามารถปรับปรุงโค้ดโปรแกรมภาษาซีและอัพโหลดเข้าสู่ระบบได้ หากโจทย์ข้อนั้นได้คะแนนไม่เป็นที่น่าพอใจ
3) อาจารย์ต้องเพิ่มโจทย์ให้ระบบพร้อมทั้งกำหนดกรณีทดสอบและซอร์สโค้ดโปรแกรมคำตอบ
4) อาจารย์สามารถแก้ไขโจทย์ กรณีทดสอบ (testcase) และซอร์สโค้ดโปรแกรมคำตอบได้แล้วส่งกลับไปให้ระบบ
5) อาจารย์สามารถทำการทดสอบความถูกต้องของผลลัพธ์ของโจทย์ได้โดยการอัพโหลดโค้ดโปรแกรมเพื่อคอมไพล์และรันโปรแกรม

3.2.2 ข้อมูลส่งออก (output)

1) ระบบต้องแสดงรายชื่อนักศึกษาเรียนแลบโปรแกรมมิ่ง
2) ระบบต้องแสดงคะแนนและผลการตรวจผลลัพธ์ของโปรแกรมของนักศึกษา
3) ระบบต้องแสดงรายละเอียดการส่งซอร์สโค้ด เช่น วัน-เวลาที่ส่งโค้ด เป็นต้น

3.2.3 กระบวนการ (process)

1) ระบบต้องตรวจผลลัพธ์ของโปรแกรมและให้คะแนนได้
2) ระบบสามารถเตรียมโจทย์และกรณีทดสอบพร้อมกำหนดคะแนนแต่ละอินพุตเพื่อให้นักศึกษาทำโจทย์ได้
3) ระบบสามารถติดต่อกับฐานข้อมูลผ่านเว็บไซต์ได้
4) ระบบมีการแสดงตัวอย่างผลลัพธ์ของโปรแกรมเพื่อให้นักศึกษาสามารถทำความเข้าใจคำสั่งของโจทย์ได้ง่ายขึ้น

3.2.4 ประสิทธิภาพ (performance)

1) ระบบต้องสามารถตรวจผลลัพธ์ของโปรแกรมให้คะแนนได้ทันทีหลังจากได้รับไฟล์โค้ดโปรแกรมจากนักศึกษา
2) ระบบต้องสามารถรองรับผู้ใช้งานจำนวนมากได้ขณะที่เข้าเว็บไซต์ในเวลาเดียวกัน

3.2.5 ส่วนควบคุม (control)

1) โค้ดโปรแกรมที่นักศึกษาส่งเข้าสู่ระบบไม่ควรเกิดการผิดพลาดขณะคอมไพล์ (compilation error) และการผิดพลาดขณะรัน (runtime error) หลังจากผ่านการคอมไพล์และเอ็กซิคิวต์ (execute) ในระบบ
2) โค้ดโปรแกรมต้องอยู่ในรูปแบบของไฟล์สกุล .c เท่านั้น

3) ผู้ใช้ระบบต้องป้อนชื่อผู้ใช้และรหัสผ่านเพื่อเข้าสู่ระบบ

4) ชื่อผู้ใช้และรหัสผ่านต้องถูกเพิ่ม, แก้ไขหรือลบโดยผู้ดูแลระบบเท่านั้น

5) ระบบต้องแบ่งสิทธิของผู้ใช้เพื่อกำหนดตารางฐานข้อมูลที่สามารถเข้าถึงได้

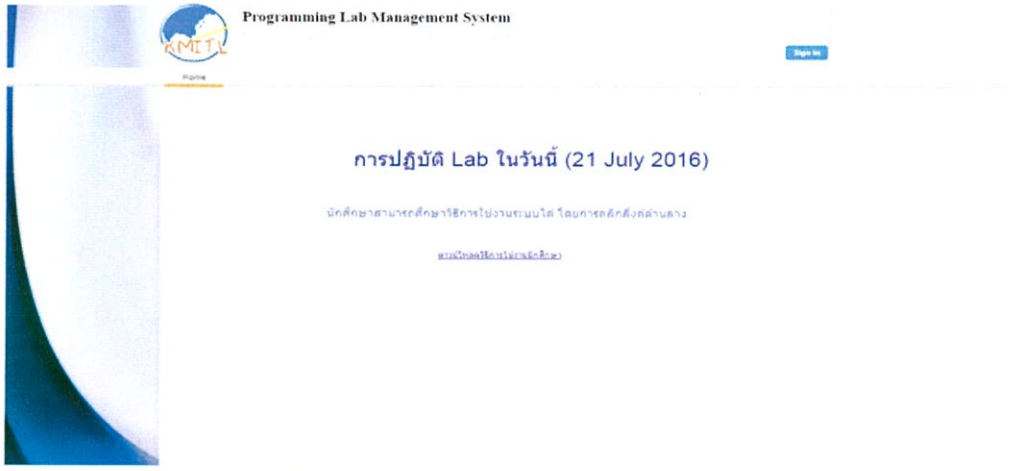
6) ระบบคอมไฟล์และรันโปรแกรมบนคอมมานด์พรอมท์

3.3 การเปรียบเทียบของระบบเก่าและระบบใหม่

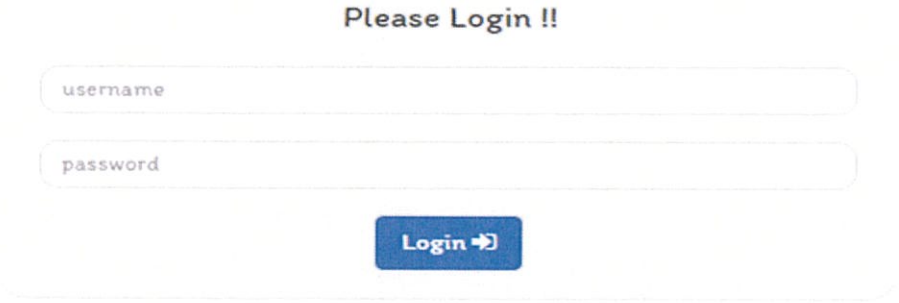
3.3.1 หน้า log-in

ระบบเก่า จะอยู่ด้านบนสุดเหนือเมนูบาร์ขึ้นไป และตรงกลางจอจะบอกถึงวันที่ของการปฏิบัติแลป

ระบบใหม่มีความเป็นทันสมัยและใช้งานได้ง่ายมากยิ่งขึ้น จะมีแค่การให้ใส่ไอดีและพาสเวิร์ดเท่านั้น ที่บริเวณกลางหน้าจอ



รูป 3.3 หน้า log-in ระบบเก่า



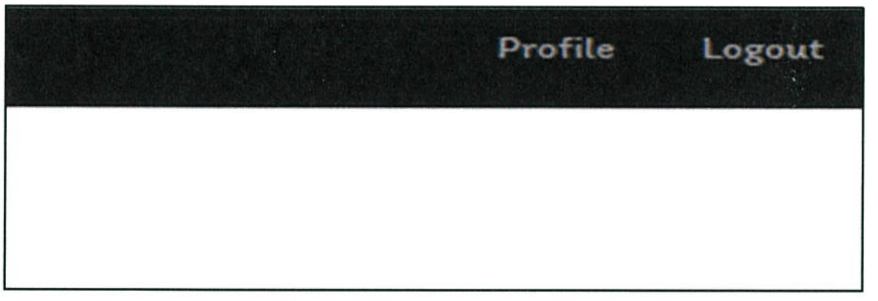
รูป 3.4 หน้า log-in ระบบใหม่

3.3.2 หน้า log-out

ระบบเก่าในส่วนของ log-out จะอยู่ตรงข้างปุ่ม home ทำให้บางที่อาจกดผิดพลาดโดนปุ่ม log-out ได้โดยง่าย

ระบบใหม่มีรูปแบบที่ทันสมัยมากยิ่งขึ้นโดยที่เราจะปรับปรุงโดยนำปุ่ม log-out ไปไว้ที่บนขวาแทนเพื่อสะดวกในการมองเห็นและจัดการได้ง่ายขึ้น

รูป 3.5 หน้า log-out ระบบเก่า



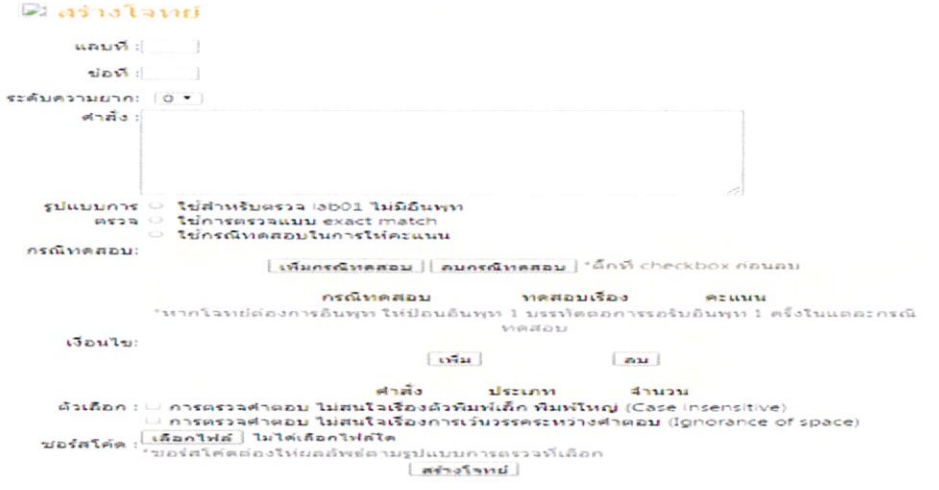
รูป 3.6 หน้า log-out ระบบใหม่

3.3.3 การเพิ่มโจทย์

ระบบเก่ารูปแบบที่ยากต่อการที่จะใช้งานและตัวหนังสืออ่านยาก ทำให้เข้าใจได้ยากโดยจะทำให้ผู้ใช้งานเกิดสับสนมึนงงในคำเขียนที่แสดงออกมา หากเรากดคลิกปุ่มที่ หากต้องการสร้างโจทย์คลิกที่นี่ ก็จะพาไปสู่หน้าสร้างโจทย์ถัดไป

หน้าถัดไปเป็นการสร้างโจทย์โดยที่เราจะต้องกำหนดแลบ ข้อที่เท่าไร ระดับความยากอะไร และคำสั่งที่จะต้องให้นักศึกษาทำ รูปแบบการตรวจ กรณีทดสอบ เงื่อนไข และการตรวจคำตอบว่าเป็นแบบไหนเสร็จแล้วค่อยนำไฟล์โจทย์มาอัพใส่ลงในระบบแต่ต้องเป็น .c เท่านั้น

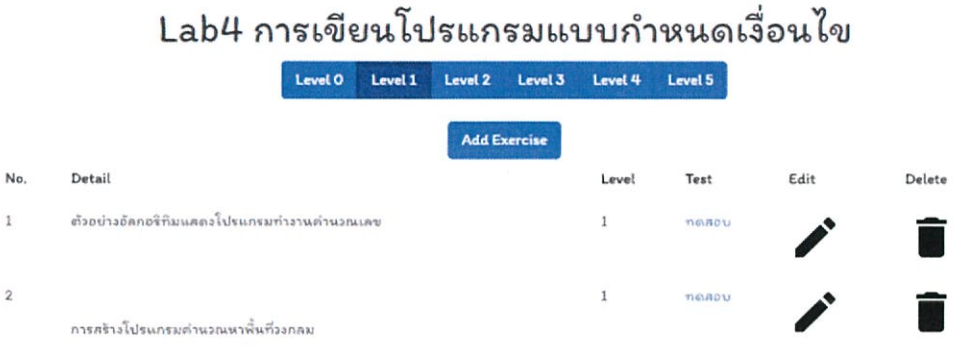
รูป 3.7 การเข้าสู่หน้าจอเพิ่มโจทย์ของระบบเก่า



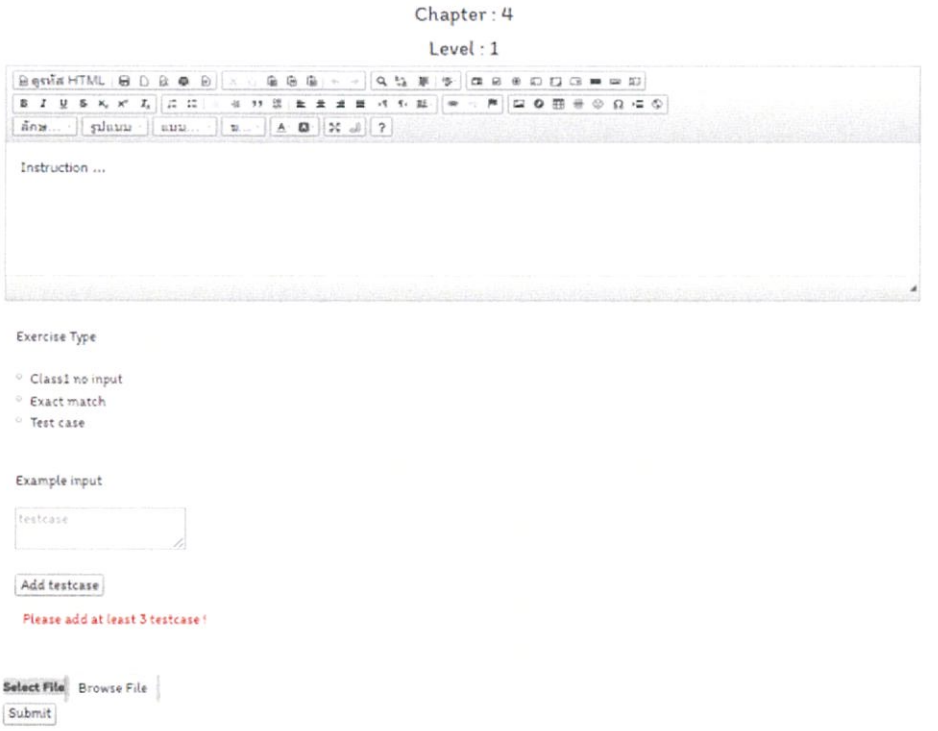
รูป 3.8 หน้าจอเพิ่มโจทย์ของระบบเก่า

ระบบใหม่มีความทันสมัยสวยงามทำให้ดูเข้าใจได้ง่ายยิ่งขึ้น และยังสามารถเพิ่มโจทย์ได้อย่างสะดวกสบาย มีรูปแบบเสริมต่าง ๆ เป็นตัวช่วยโดยเพียงแค่เรากดคลิกปุ่ม Add Exercise ก็จะพาเข้าสู่ไปที่หน้าต่อไปในการสร้างโจทย์

หน้าถัดไปหลังจากการสร้างโจทย์โดยเราจะเป็นคำกำหนดว่าโจทย์นั้นจะมีคำสั่งอย่างไร การตรวจแบบไหน ตัวอย่างในการตรวจ เงื่อนไขต่าง ๆ โดยที่เราไม่ต้องกำหนดเลเวลอะไร แลบอะไร เพราะได้ถูกเลือกแต่แรกแล้วทำให้เกิดความสะดวกสบายและเข้าใจได้ง่ายยิ่งขึ้น



รูป 3.9 การเข้าสู่หน้าจอเพิ่มโจทย์ของระบบใหม่



รูป 3.10 หน้าจอเพิ่มโจทย์ของระบบใหม่

3.3.4 การแก้ไขโจทย์

ระบบเก่ามีตัวอักษรที่หนาเยอะอ่านลำบากว่ารายละเอียดที่ต้องการนั้นคืออะไร ทำให้เกิดความสับสนต่อผู้ที่ใช้งาน โดยที่เราจะแก้ไขจะต้องคลิกที่ปากกาเพื่อไปยังหน้าแก้ไขหน้าถัดไป

หน้าถัดไปเป็นหน้าสำหรับการแก้ไขโจทย์โดยจะมีกรณีต่าง ๆ ที่เราสามารถที่จะเปลี่ยนหรือไขได้รวมถึงคำสั่งของโจทย์แต่อาจจะดูยุ่งยากและเยอะที่ใช้ในการแก้โจทย์รมถึงคำสั่งที่อ่านยากทำให้ใช้เวลาในการแก้ไขมากขึ้น

แลบ	ข้อ	level	รายละเอียด	ทดสอบ	แก้ไข	ลบ
1	4	0	ทดลองกำหนดรูปแบบการพิมพ์ด้วย \n และ \t ของคำสั่ง printf ให้นักศึกษาเขียนโปรแกรมแสดงข้อมูลของตัวนักศึกษาเอง ซึ่งประกอบด้วย รหัสนักศึกษา,เบอร์โทรศัพท์,ชื่อ-นามสกุล และจังหวัดที่อยู่ โดยมีรูปแบบการแสดงผลดังตัวอย่าง เก็บไว้ภายใต้โปรเจคเดิม คือ Lab01 ชื่อโปรแกรม Lab01_4 ตัวอักษรแรกของบรรทัดที่ 1 จะมีการเว้นระยะจากขอบด้านซ้าย 8 ช่องตัวอักษร, ตัว N ในบรรทัดที่ 2 จะอยู่ตำแหน่งเดียวกับ I ,ในบรรทัดแรก ตัวอักษร p ในบรรทัดที่ 1 และ 2 อยู่ตำแหน่งเดียวกัน, โคลอน : จะมีการเว้นระยะด้านซ้ายและขวา 1 ช่องตัวอักษร	ทดสอบ		

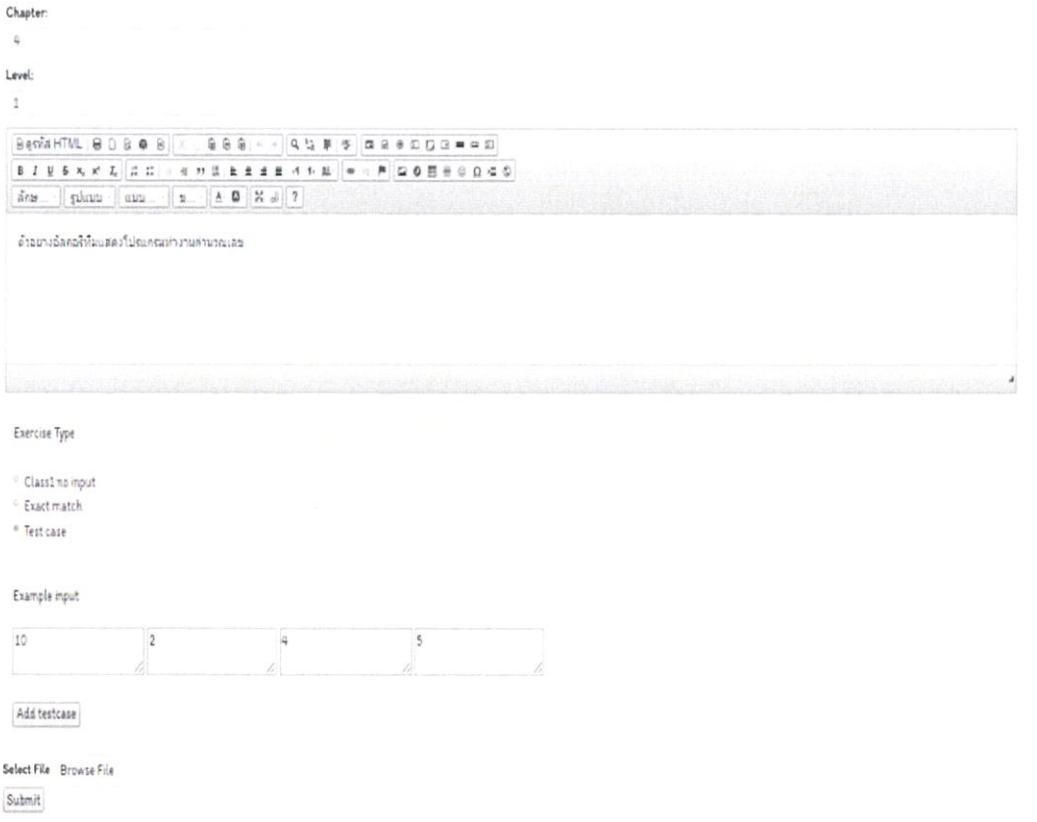
รูป 3.11 การเข้าสู่หน้าจอแก้ไขโจทย์ของระบบเก่า

รูป 3.12 หน้าจอแก้ไขโจทย์ของระบบเก่า

ระบบใหม่มีการจัดเรียงสวยงามและกะทัดรัดง่ายต่อการที่จะใช้งานทำให้ผู้ใช้งานนั้นมองง่ายและดูสบายตาโดยเพียงแค่กดคลิกปุ่มปากกาก็จะพาไปที่หน้าต่อไป

ต่อไปเป็นหน้าถัดมาจากการแก้ไขโจทย์ โดยเราสามารถแก้ไขกรณีเงื่อนไขต่าง ๆ ได้โดยง่ายไม่ยุ่งยากมองง่ายและสะดวกต่อการใช้งาน แถมภาษาที่อ่านง่ายและกะทัดรัด

รูป 3.13 การเข้าสู่หน้าจอแก้ไขโจทย์ของระบบใหม่



รูป 3.14 หน้าจอโจทย์ของระบบใหม่

3.3.5 การลบโจทย์

ระบบเก่าทำการลบโจทย์โดยการคลิกที่ปุ่มกากบาทโดยโจทย์จะหายไปเลย ไม่มีการแจ้งเตือนใด ๆ ทั้งสิ้นทำให้อาจเกิดความผิดพลาดเวลาที่กดโดนปุ่มผิดทำให้โจทย์ที่สร้างมานั้นหายไป เกิดความเสียเวลาในการสร้างโจทย์ขึ้นมาใหม่ และเกิดความสับสนกับผู้ที่ใช้งานเป็นครั้งแรก

ระบบใหม่ เวลาลบโจทย์จะกดที่ปุ่มถังขยะเพื่อทำการลบและจะแจ้งเตือนมาก่อนว่าจะทำการลบโจทย์จริงหรือใหม่เพื่อทำการยืนยันในการลบโจทย์เพื่อแก้ปัญหาของระบบเก่าที่เกิดขึ้นเพื่อ

ไม่ให้ผู้ใช้งานต้องลำบากเวลาที่เกิดความผิดพลาดเวลาที่เผลอกดลบโจทย์ไปและจะมีส่วนตัวจะเป็นส่วนที่แจ้งเตือนเวลากรณีที่ต้องการลบโจทย์ เพื่อทำการยืนยันต่อผู้ใช้ว่าต้องการที่จะลบจริง ๆ หรือไม่

ลบ	ข้อ	level	รายละเอียด	ทดสอบ	แก้ไข	ลบ
1	4	0	ทดลองกำหนดรูปแบบการพิมพ์ด้วย \n และ \t ของคำสั่ง printf ให้นักศึกษาเขียนโปรแกรมแสดงข้อมูลของตัวนักศึกษาเอง ซึ่งประกอบด้วย รหัสนักศึกษา,เบอร์โทรศัพท์,ชื่อ-นามสกุล และจังหวัดที่อยู่ โดยมีรูปแบบการแสดงผลดังตัวอย่าง เก็บไว้ภายใต้โปรเจคเดิม คือ Lab01 ชื่อโปรแกรม Lab01_4 ตัวอักษรแรกของบรรทัดที่ 1 จะมีการเว้นระยะจากขอบด้านซ้าย 8 ช่องตัวอักษร, ตัว N ในบรรทัดที่ 2 จะอยู่ตำแหน่งเดียวกับ I ,ในบรรทัดแรก ตัวอักษร p ในบรรทัดที่ 1 และ 2 อยู่ตำแหน่งเดียวกัน, โคลอน : จะมีการเว้นระยะด้านซ้ายและขวา 1 ช่องตัวอักษร	ทดสอบ		

รูป 3.15 การลบโจทย์ของระบบเก่า

รูป 3.16 การลบโจทย์ของระบบใหม่

รูป 3.17 การแจ้งเตือนการลบโจทย์ของระบบใหม่

3.3.6 การดูโจทย์ในแต่ละบท

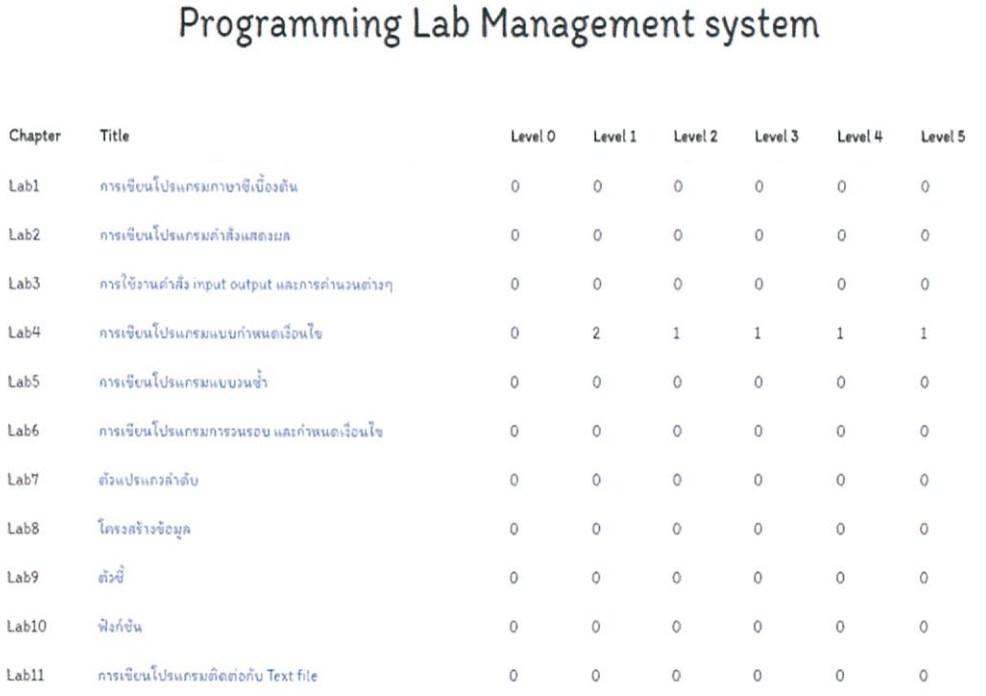
ระบบเก่าต้องทำการกดเลือกบทและยืนยันถึงจะดูโจทย์ในแต่ละบทได้ และถ้าดูทุกบทจะทำให้มีความหลากหลายของจำนวนข้อมากไป เพราะแต่ละข้อจะมีโจทย์ที่ยาวและอ่านยากทำให้เกิดความสับสนของผู้ที่เคยใช้ครั้งแรกถ้าเราอยากดูบทไหนก็จะต้องเป็นผู้กดเลือกเองตามภาพ

รูป 3.18 การดูโจทย์ทั้งหมดของระบบเก่า

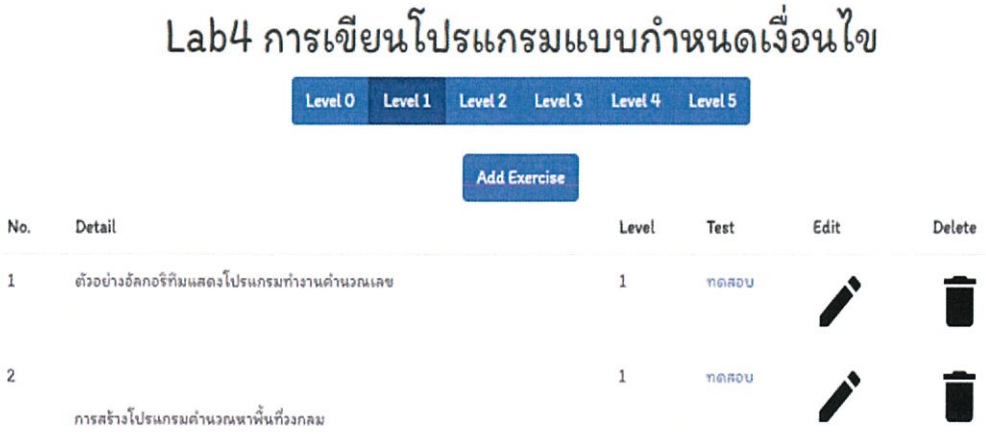
รูป 3.19 การดูโจทย์บทที่ 2 ของระบบเก่า

ระบบใหม่ทำให้เกิดความสะดวกสบายในการเข้าใจและการใช้เพิ่มมากขึ้น กดตอนแรกก็แสดงโจทย์ทุกบทมาให้ดูเป็นหัวข้อต่าง ๆ ถ้าเกิดเราอยากดูบทนี้ก็เพียงแค่คลิกไปที่บท ๆ นั้นก็จะแสดงข้อมูลของข้อนั้น ๆ ขึ้นมาแสดงให้ดู

ส่วนภาพนี้จะเป็นการเลือกโจทย์บท 4 เพียงแค่คลิกไปในชื่อหัวเรื่องสีน้ำเงินของบทที่ 4 เท่านั้น ก็จะพามาหน้าดังนี้และสามารถที่จะเลือกโจทย์ตามระดับความยากต่าง ๆ ที่มีได้ตามต้องการ



รูป 3.20 การดูโจทย์ทั้งหมดของระบบใหม่



รูป 3.21 การดูโจทย์บทที่ 4 ของระบบใหม่

3.3.7 การกรองโจทย์แบบเป็นเลเวลได้

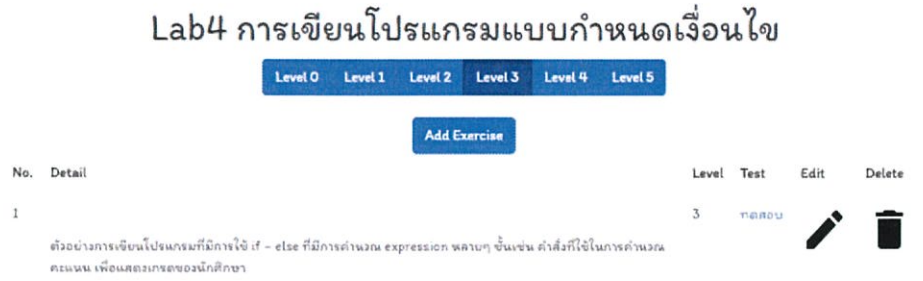
ระบบเก่าไม่ได้มีการทำระบบนี้ขึ้นมา โดยจะบอกแค่ว่าบทนี้มีเลเวลนี้ทั้งหมดกี่ข้อโดยเรียงมาให้ดูทุกเลเวลและจะทำการเลือกอีกที

ระบบใหม่โดยเวลากดเข้ามาแลบที่ต้องการเราจะสามารถเลือกเลเวลการเปิดแลบได้ตามต้องการโดยการที่กดคลิกที่เลเวลของแต่ละอันภายในแลบ อันนี้กดเลือกเลเวล 2 ในบท 4 ก็จะมีข้อที่อยู่ในเลเวลนี้ให้เลือก

ถ้าเราอยากที่จะเปลี่ยนเลเวลก็เพียงคลิกเลเวลที่ต้องการ โดยหน้านี้เราคลิกเปลี่ยนเป็นเลเวล 3 โดยที่จะมีโจทย์ในเลเวล3ให้เราเลือกได้ตามต้องการ

รูป 3.22 การแสดงเลเวลของระบบเก่า

รูป 3.23 การกรองเลเวล 2 ในระบบใหม่

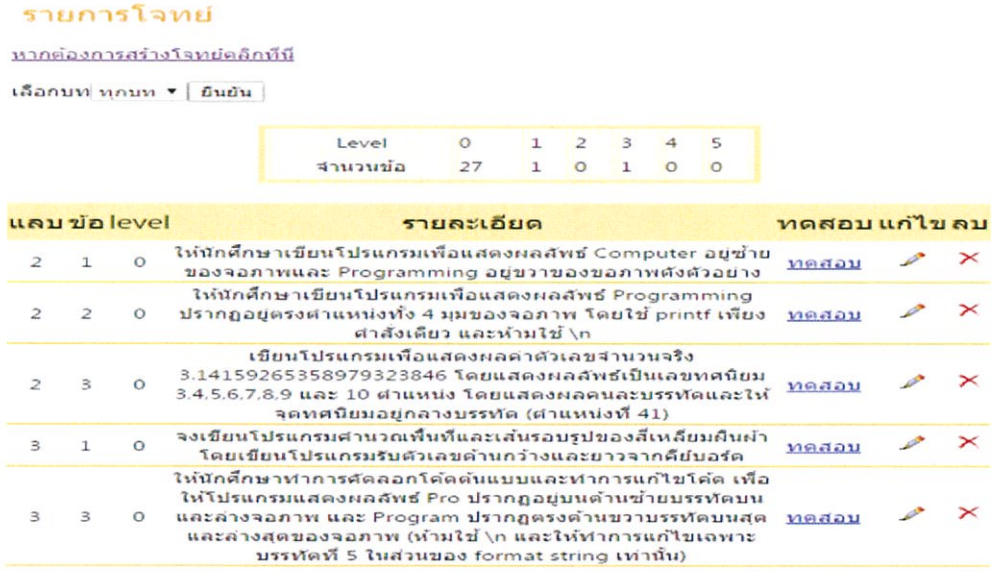


รูป 3.24 การกรองเลเวล 3 ในระบบใหม่

3.3.8 การดูจำนวนโจทย์

ระบบเก่าจะแสดงมาในภาพที่อาจจะยากต่อการเข้าใจ เพราะเริ่มต้นมาจะแสดงให้ดูทุกบทและยังเอาเนื้อหาคำสั่งเพิ่มมาด้วย ถ้าอยากจะแสดงบทไหนต้องทำการเลือกบทนั้น ๆ และกดยืนยัน

ระบบใหม่ออกแบบมาแสดงให้เห็นโจทย์ทั้งหมดรวมถึงเลเวลแต่ละข้อที่มีอยู่ในโจทย์ แสดงให้เข้าใจง่ายและกะทัดรัดมายิ่งขึ้น โดยจะบอกเพียงแค่แลบอะไรหัวข้อเรื่องอะไรและแต่ละเลเวลในแลบมีกี่ข้อบ้าง



รูป 3.25 การดูจำนวนโจทย์ของระบบเก่า

Programming Lab Management system

Chapter	Title	Level 0	Level 1	Level 2	Level 3	Level 4	Level 5
Lab1	การเขียนโปรแกรมภาษาซีเบื้องต้น	0	0	0	0	0	0
Lab2	การเขียนโปรแกรมคำสั่งแสดงผล	0	0	0	0	0	0
Lab3	การใช้งานคำสั่ง input output และการคำนวนต่างๆ	0	0	0	0	0	0
Lab4	การเขียนโปรแกรมแบบกำหนดเงื่อนไข	0	2	1	1	1	1
Lab5	การเขียนโปรแกรมแบบวนซ้ำ	0	0	0	0	0	0
Lab6	การเขียนโปรแกรมการวนรอบ และกำหนดเงื่อนไข	0	0	0	0	0	0
Lab7	ตัวแปรแถวลำดับ	0	0	0	0	0	0
Lab8	โครงสร้างข้อมูล	0	0	0	0	0	0
Lab9	ตัวชี้	0	0	0	0	0	0
Lab10	ฟังก์ชัน	0	0	0	0	0	0
Lab11	การเขียนโปรแกรมติดต่อกับ Text file	0	0	0	0	0	0

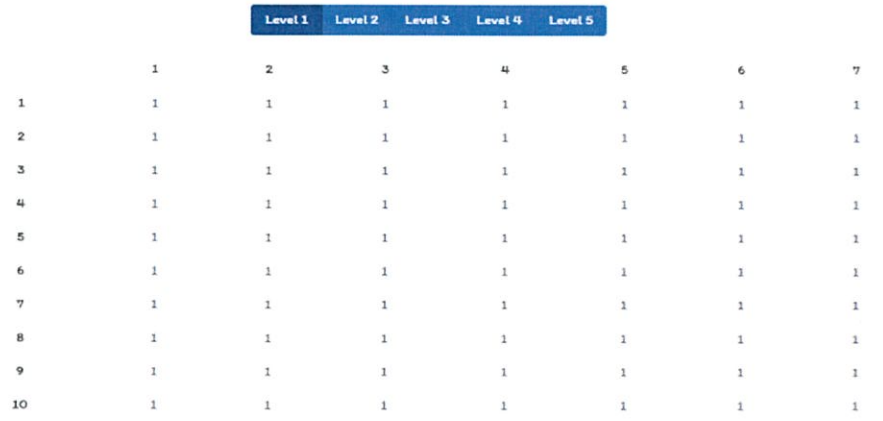
รูป 3.26 การดูจำนวนโจทย์ของระบบใหม่

3.3.9 สุ่มโจทย์ให้นักศึกษา 5 เลเวล

ระบบเก่า ในการสุ่มโจทย์นั้นจะมีการทำงานแบบแจกโจทย์ให้นักศึกษาทั้งหมด 3 ชุด โดยแต่ละคนนั้นจะมีโอกาสที่จะได้โจทย์จำนวน 1 ข้อภายใน 3 ชุดนี้ ทำให้โอกาสที่จะซ้ำกันนั้นง่าย และทำให้มีโอกาสที่จะลอกกันได้ง่าย และต้องสร้างโจทย์ถึง 3 ชุดด้วยกันในการเปิดแลบ

ระบบใหม่ ระบบสุ่มจะทำงานโดยกดเปิดแลบที่เซคที่ต้องการจะเปิดและบทที่ต้องการจะเปิดก็จะทำการสุ่มโจทย์ที่มีทั้งหมดในบทนั้นแจกให้กับนักศึกษา โดยที่นักศึกษาที่นั่งติดกันจะไม่มีโอกาสที่ได้โจทย์ซ้ำกัน

รูป 3.27 การสุ่มโจทย์ให้นักศึกษาของระบบใหม่



	1	2	3	4	5	6	7
1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1

รูป 3.28 การแสดงโจทย์ที่สุ่มได้ของระบบใหม่

3.3.10 การดูสถานะเปิด-ปิดแลบ

ระบบเก่าจะออกแบบมาเพื่อการกำหนดโจทย์ 3 ชุดเพื่อแจกให้กับนักศึกษา ถ้ากำหนดโจทย์ไม่ครบทั้ง 3 ชุดก็จะไม่สามารถที่จะทำการเปิดแลบได้ และมีระยะเวลาในการเปิดแลบได้มากสุดเพียงแค่ 2 ชั่วโมงเท่านั้น โดยปัญหาที่พบจะมีปัญหาเรื่องเวลาในการสร้างโจทย์ที่ไม่ตรงที่ระบบต้องการทำให้ไม่สามารถที่จะเปิดแลบได้ ทำให้เกิดความสับสนต่อผู้ใช้เป็นอันมาก

ระบบใหม่จะแสดงให้เห็นสถานะเปิดปิดแลบของทุกบท เวลาที่จะกดเปิดปิดแลบก็เพียง แค่คลิกบทที่ต้องการให้เปิดและเซตไหน สีแดงคือแลบได้ถูกปิดอยู่ส่วนสีเขียวคือแลบได้ทำการเปิด ขึ้นมาแล้วทำให้เพิ่มความสะดวกสบายกับผู้ใช้งาน ไม่มีการกำหนดว่าสร้างโจทย์กี่ข้อแก้ไขปัญหาเวลาในการสร้างโจทย์ ไม่มีการกำหนดเวลาเพียงแค่ปิดเปิดแลบได้ตามที่ผู้ใช้ต้องการ

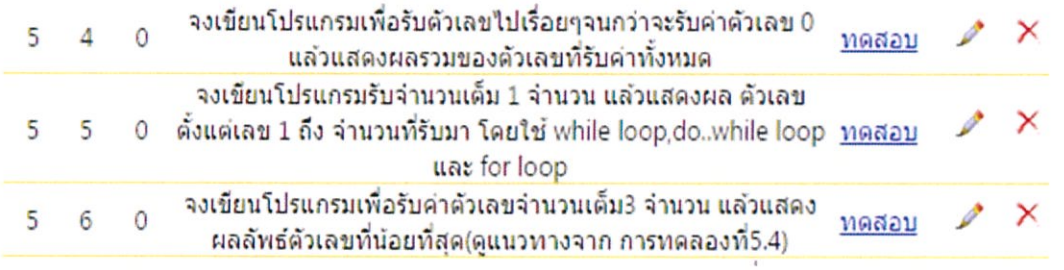
รูป 3.29 การดูสถานะเปิด-ปิดแลบของระบบเก่า

รูป 3.30 การดูสถานะเปิด-ปิดแลบของระบบใหม่

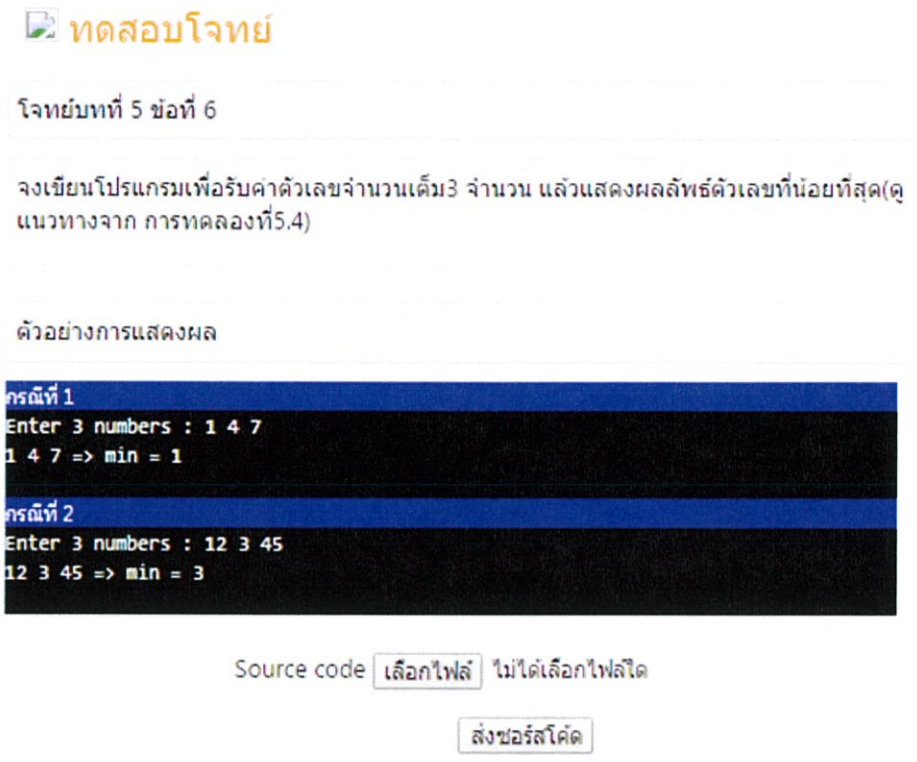
3.3.11 การทดสอบโจทย์

ระบบเก่าก็จะมีรูปแบบที่ยากต่อการเข้าใจเนื่องจากมีคำสั่งโจทย์ที่ยากทำให้เกิดสับสนได้ จะใช้งานโดยคลิกที่ปุ่มทดสอบเพื่อทำการทดสอบในหน้าถัดไป

พอเข้ามาที่หน้าทดสอบโจทย์ก็จะทำการนำไฟล์ที่ต้องการทำการทดสอบอัพโหลดขึ้นไปในระบบ



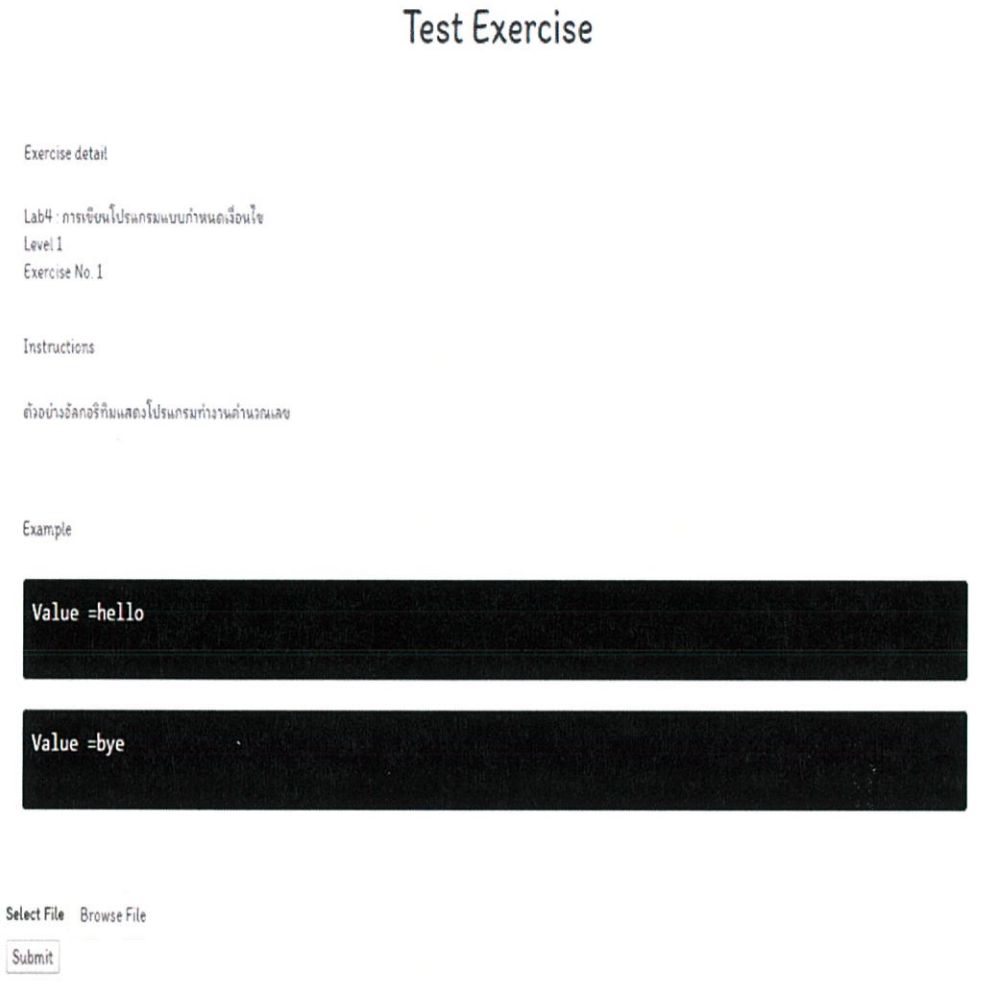
รูป 3.31 การเข้าสู่หน้าจอทดสอบโจทย์ของระบบเก่า



รูป 3.32 การทดสอบโจทย์ของระบบเก่า

ระบบใหม่ มีรูปแบบที่กะทัดรัดเข้าใจง่ายสะดวกในการใช้งานของผู้ใช้ระบบมากยิ่งขึ้น โดยมีเนื้อหาและรูปแบบและง่ายต่อผู้ใช้งานเป็นอันมาก ทำให้มองสบายตาละไม่ซับซ้อนในการที่เราจะทดสอบโจทย์

รูป 3.33 การเข้าสู่หน้าจอทดสอบโจทย์ของระบบใหม่



รูป 3.34 การทดสอบโจทย์ของระบบใหม่

3.3.12 ผลการทดสอบโจทย์

ระบบเก่าทดสอบเสร็จก็จะแสดงว่าถูกหรือไม่และแสดงโค้ดที่ได้ทำการส่งเข้าไปในระบบว่ามีการเขียนอย่างไร

รูป 3.35 ผลการทดสอบโจทย์ของระบบเก่า

```c
#include<stdio.h>
void main(){
	int a,b,c  ;
	printf("Enter 3 numbers : ");
	scanf("%d %d %d",&a,&b,&c);
	if(a<b&&a<c){
		printf("%d %d %d => min = %d\n",a,b,c,a );
	}else if(b<a&&b<c){
		printf("%d %d %d => min = %d\n",a,b,c,b );
	}else {
		printf("%d %d %d => min = %d\n",a,b,c,c );
	}
}
```

รูป 3.36 การแสดงโค้ดที่ส่งไปของระบบเก่า

ระบบใหม่จะทำการเพิ่มเติมของตัวคอมไฟล์โดยจะมีวันและเวลากำกับอยู่เวลาทดสอบโจทย์สำเร็จโดยจะบอกว่ามาจะแลบอะไรและระดับความยากของข้อนั้นเป็นอย่างไรเพิ่มความสะดวกสบายแต่การใช้งานและการตรวจ

รูป 3.37 ผลการทดสอบโจทย์ของระบบใหม่

Your code here !!

```
//Date :2016-07-21 17:25:28
//Username : supervisor
//Lab : 4 level : 1 Index : 1

#include
int main(){
		int a;
		printf("Value =");
		scanf("%d",&a);

		if(a==10)
			printf("hello");
		else
			printf("bye");
		return 0;
}
```

รูป 3.38 การแสดงโค้ดที่ส่งไปของระบบใหม่

3.4 การใช้ระบบ

3.4.1 ตารางการใช้งานระบบ

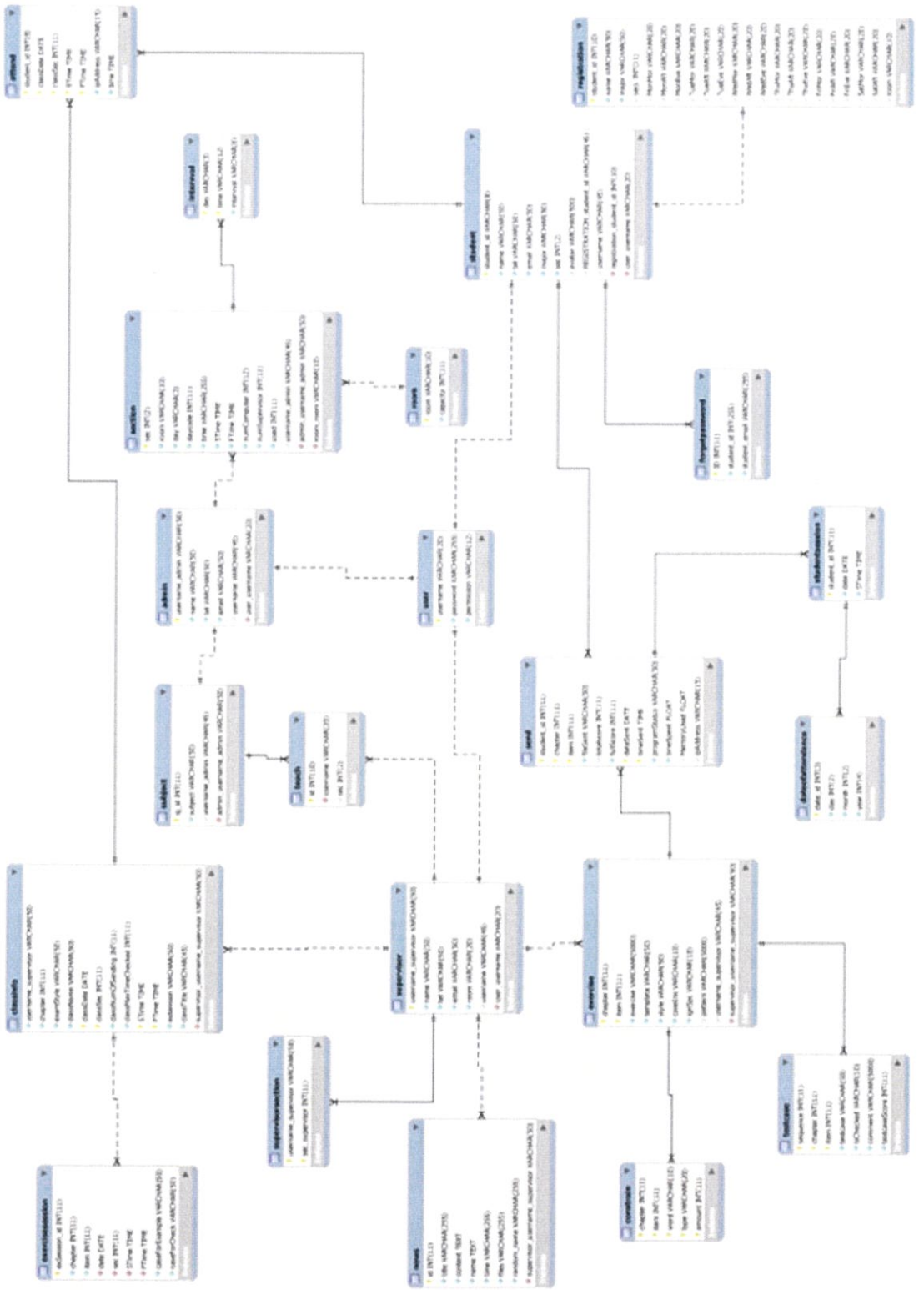
ตาราง 3.1 การใช้งานระบบแบ่งตามประเภทผู้ใช้งาน

ผู้ดูแลระบบ Admin	1. บริหารจัดการข้อมูลที่เกี่ยวข้องกับห้องแลบ ○ เพิ่มผู้ใช้งาน ได้แก่ ผู้ดูแลระบบ อาจารย์ และนักศึกษา เข้าสู่ระบบ ○ ลบผู้ใช้งาน ได้แก่ ผู้ดูแลระบบ อาจารย์ และนักศึกษา จากในระบบ ○ แก้ไขผู้ใช้งาน ได้แก่ ผู้ดูแลระบบ อาจารย์ และนักศึกษา ภายในระบบ ○ นำเข้าข้อมูลลงทะเบียนนักศึกษาเป็นไฟล์ .csv ○ ล้างหรือรีเซ็ตข้อมูลทั้งหมดในฐานข้อมูล ○ แก้ไขข้อมูลส่วนตัวของผู้ดูแลระบบ (admin) เช่น Name, Password, Telephone number, E-mail
อาจารย์ Supervisor	1. สร้างโจทย์แบบฝึกหัดเข้าสู่ระบบ ○ สามารถเขียนอธิบายคำสั่งโจทย์ได้เพื่อให้ง่ายต่อการเข้าใจ ○ มีการกำหนดรูปแบบ เงื่อนไข และระดับความยากของโจทย์ได้ ○ ตั้งค่ากรณีทดสอบ (testcase) ใช้ในการให้คะแนนเวลาตรวจคำตอบ 2. ลบโจทย์แบบฝึกหัดที่ไม่ต้องการออกจากระบบได้ 3. แก้ไขโจทย์แบบฝึกหัดให้ตรงตามที่เราต้องการได้ 4. ทดสอบโจทย์แบบฝึกหัดที่ได้สร้างหรือแก้ไขไว้ 5. ตรวจสอบประวัติการส่งแลบย้อนหลังทั้งหมดของนักศึกษาได้ 6. สามารถดูได้ว่า มีนักศึกษาส่งแลบครั้งล่าสุดแล้วกี่คน 7. เลือกดูซอร์สโค้ดของนักศึกษาแต่ละแลบของทุกคนได้ 8. ดูสถานะ login และ logout ของนักศึกษาตามตำแหน่งที่นั่งของตนเอง ○ แสดงจำนวนข้อและโจทย์แบบฝึกหัดที่นักศึกษาแต่ละคนสุ่มได้ 9. แก้ไขข้อมูลส่วนตัวของอาจารย์ผู้สอน (Supervisor) เช่น Name, Password, Telephone number, E-mail

ตาราง 3.1 การใช้งานระบบแบ่งตามประเภทผู้ใช้งาน (ต่อ)

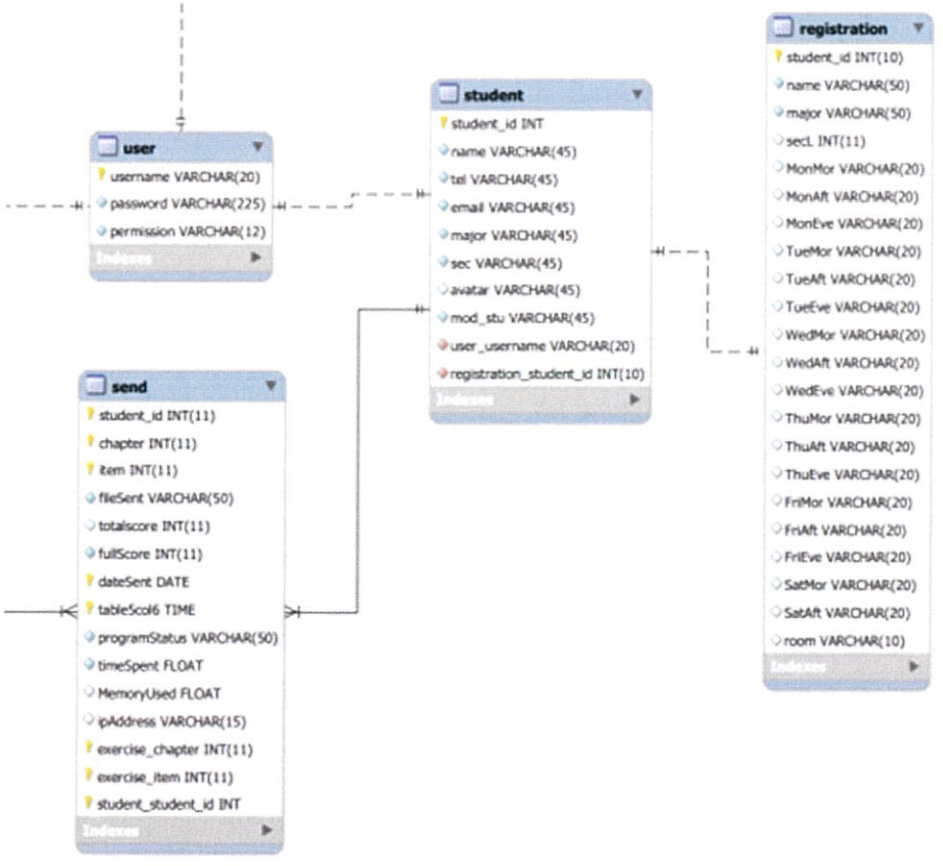
นักศึกษา Student	1. เข้าทำโจทย์ด้วยการอัพโหลดไฟล์ซอร์สโค้ดของตนเองขึ้นสู่ระบบ 2. ทำการตรวจผลลัพธ์ของซอร์สโค้ดแบบฝึกหัดที่นักศึกษานำเข้าสู่ระบบ 3. แสดงผลลัพธ์การตรวจคำตอบ แล้วแสดงเป็นคะแนนที่ได้ในแต่ละแลบ 4. ตรวจประวัติการส่งแลบย้อนหลังทั้งหมด ของตัวนักศึกษาเองได้ 5. สามารถตรวจสอบตารางเรียนส่วนบุคคลของตนเองได้ 6. แจ้งปัญหาและข้อเสนอแนะเกี่ยวกับระบบแก่ผู้ดูแลระบบได้ 7. แก้ไขข้อมูลส่วนตัวของนักศึกษา (Student) เช่น Password, Name, Telephone number, E-mail, Portrait

3.5 การออกแบบฐานข้อมูล



รูป 3.39 เอนทิตี้และความสัมพันธ์ระหว่างเอนทิตี้ของฐานข้อมูลในระบบ

รูป 3.40 เอนทิตี้และความสัมพันธ์ระหว่างเอนทิตี้ของฐานข้อมูลในระบบ (admin)



รูป 3.41 เอนทิตี้และความสัมพันธ์ระหว่างเอนทิตี้ของฐานข้อมูลในระบบ (Student)

รูป 3.42 เอนทิตี้และความสัมพันธ์ระหว่างเอนทิตี้ของฐานข้อมูลในระบบ (Supervisor)

3.6 การทำงานระหว่าง Folder (Folder Tree)

3.6.1 โฟลเดอร์ Project

Application folder

Config folder (ส่วนการตั้งค่า)

config.php – ตั้งค่า font path

database.php – เป็นการ config เกี่ยวกับ database

folder.php – ตั้งค่า folder ต่าง ๆ ที่ใช้เก็บไฟล์

controller folder (ส่วนการควบคุมระบบ)

MainCtrl.php – ไฟล์ฟังก์ชั่นก่อนล็อคอินจะอยู่ในไฟล์นี้

Admin.php – ไฟล์ฟังก์ชันของแอดมิน

Supervisor.php – ไฟล์ฟังก์ชันของอาจารย์

Student.php – ไฟล์ฟังก์ชันของนักศึกษา

models folder (ส่วน Function ติดต่อ Database)

Admin_db.php – ไฟล์ฟังก์ชั่นการติดต่อกับ database ของแอดมินจะอยู่ในนี้

Supervisor_db.php – ไฟล์ฟังก์ชั่นการติดต่อกับ database ของอาจารย์

Studetn_db.php – ไฟล์ฟังก์ชั่นการติดต่อกับ database ของนักศึกษา

views folder (ส่วนการแสดงผลทางหน้า Website)

login.php

admin – แฟ้มสำหรับเก็บหน้าแสดงผลของผู้ดูแลระบบ

supervisor – แฟ้มสำหรับเก็บหน้าแสดงผลของอาจารย์

student – แฟ้มสำหรับเก็บหน้าแสดงผลของนักศึกษา

assets folder

css – แฟ้มเก็บไฟล์ที่ช่วยในการแสดงผลของระบบ

font – แฟ้มเก็บรูปแบบอักษรพิเศษของระบบ

images – แฟ้มเก็บรูปที่ใช้ในระบบ

js – แฟ้มเก็บไฟล์จาวาสคริป

บทที่ 4

การทดลองและผลการทดลอง

4.1 สภาพแวดล้อมที่ใช้ในระบบ

ในการพัฒนาและทดลองระบบจัดการห้องแลบโปรแกรมมิ่ง ต้องจัดสภาพแวดล้อมและเครื่องมือรวมถึงซอฟท์แวร์ต่าง ๆ ดังนี้

1) ระบบปฏิบัติการ: Windows 8

2) Web server: Apache 5.5.24 for Windows 32 bits

3) Database Server: PhpMyAdmin

4) Web browser: Google Chrome

5) คอมพิวเตอร์เพื่อใช้ในการจำลองเป็นตัวเซิร์ฟเวอร์

4.2 ขั้นตอนการทดลอง

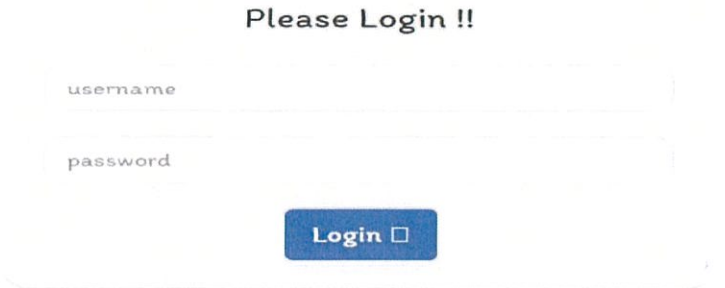
4.2.1 การเข้าสู่ระบบ

ในการเข้าใช้งานระบบจัดการห้องแลบโปรแกรมมิ่ง เริ่มแรกจะต้องเข้าไปที่เว็บไซต์ที่ผู้ติดตั้งได้ทำการกำหนด จากนั้นผู้ใช้จะต้องทำการเข้าสู่ระบบโดยการกรอกชื่อผู้ใช้และรหัสผ่าน ซึ่งหากกรอกชื่อผู้ใช้หรือรหัสผ่านผิด ก็จะไม่สามารถเข้าระบบได้ โดยผู้ใช้งานแบ่งออกเป็น 3 ประเภท คือ

1) ผู้ดูแลระบบ

2) อาจารย์

3) นักศึกษา



รูป 4.1 หน้าต่างกรอกข้อมูลเพื่อเข้าระบบ

4.2.2 การเพิ่มข้อมูลผู้ใช้

การเพิ่มข้อมูลผู้ใช้นั้นสามารถกระทำได้โดยผู้ใช้ที่มีสิทธิ์เป็นผู้ดูแลระบบเท่านั้น เมื่อเลือกรายการเพิ่มข้อมูลผู้ใช้ จะปรากฏช่องกรอกรายละเอียดของผู้ใช้

รูป 4.2 หน้าต่างเพิ่มผู้ใช้

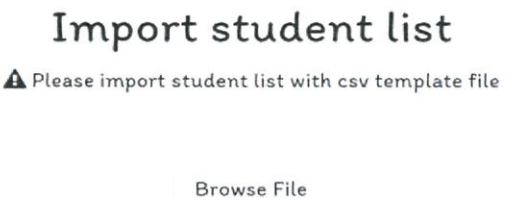
โดยสามารถกรอกรายละเอียดต่าง ๆ ของผู้ใช้ เช่น ชื่อผู้ใช้, รหัสผ่าน, หมายเลขโทรศัพท์, อีเมล์ พร้อมทั้งกำหนดสิทธิ์ผู้ใช้ด้วย รวมทั้งสามารถแก้ไขหรือลบผู้ใช้ได้

ชื่อผู้ใช้	ชื่อสกุล	โทร. e-mail	สาขาวิชา	กลุ่มเรียน	แก้ไข	ลบ
51010013	นายกรณ์ทรงพล ตรางา		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57010160	นายจักรภัทร จันดง		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57010854	นายพลภัทร ลิมะตำรงค์		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57010879	นายพันธกานต์ พิมพ์จันทร์		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57010887	นางสาวพิชชาพร แก้วพรม		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57010908	นางสาวพิรญาณ์ เมืองโคตร		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57010936	นางสาวเพ็ญฟ้า เจ็ดตีสกุล		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57010937	นายเพิมศักดิ์ พิมพะนิตย์		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57010944	นายฟาฮัด อาหมัด		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57010965	นางสาวภัทรา พรมโสภา		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57010990	นายภูชิชย์ ชูเชิด		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ
57011000	นายภูมิภักดิ์ แจมจำรัส		วิศวกรรมเครื่องกล	17	แก้ไข	ลบ

รูป 4.3 รายการผู้ใช้

4.2.3 การนำเข้าข้อมูลลงทะเบียนนักศึกษา

การนำเข้าข้อมูลลงทะเบียนนักศึกษาสามารถกระทำได้โดยผู้ใช้ที่มีสิทธิ์เป็นผู้ดูแลระบบเท่านั้น ใช้ในกรณีที่ผู้ดูแลระบบต้องการเพิ่มผู้ใช้งานที่เป็นนักศึกษาเป็นจำนวนมาก สามารถทำได้โดยนำข้อมูลของนักศึกษาใส่ลงในไฟล์ .csv



รูป 4.4 หน้าต่างนำเข้าข้อมูลลงทะเบียนนักศึกษา

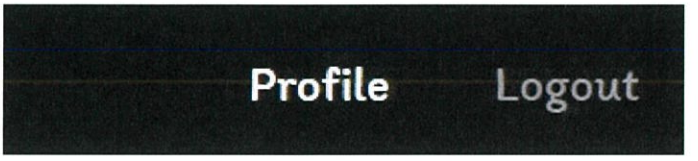
โดยไฟล์ .csv ที่ใช้จะต้องมีรูปแบบดังนี้

	A	B	C	D	E
1	รหัส	ชื่อ-นามสกุล	ภาควิชา	กลุ่มเรียน	วันเวลา
2	51010013	นายกรณ์ไทรงพล ตรางา	วิศวกรรมเครื่องกล	17	TueMor
3	57010160	นายจักรภัทร จันดง	วิศวกรรมเครื่องกล	17	TueMor
4	57010854	นายพลภัทร ลิมะสารงค์	วิศวกรรมเครื่องกล	17	TueMor
5	57010879	นายพันธกานต์ พิมพ์จันทร์	วิศวกรรมเครื่องกล	17	TueMor
6	57010887	นางสาวพิชชาพร แก้วพรม	วิศวกรรมเครื่องกล	17	TueMor
7	57010908	นางสาวพิรญาณ์ เมืองโคตร	วิศวกรรมเครื่องกล	17	TueMor
8	57010936	นางสาวเพ็ญฟ้า เจ็ดดีสกุล	วิศวกรรมเครื่องกล	17	TueMor
9	57010937	นายเพิ่มศักดิ์ พิมพะนิตย์	วิศวกรรมเครื่องกล	17	TueMor
10	57010944	นายฟาฮัด อาหมัด	วิศวกรรมเครื่องกล	17	TueMor
11	57010965	นางสาวภัทรา พรมโสภา	วิศวกรรมเครื่องกล	17	TueMor
12	57010990	นายภูชิชย์ ชูเชิด	วิศวกรรมเครื่องกล	17	TueMor
13	57011000	นายภูมิภักดิ์ แจมจำรัส	วิศวกรรมเครื่องกล	17	TueMor
14	57011001	นายภูริชา ตันสิริ	วิศวกรรมเครื่องกล	17	TueMor
15	57011004	นายภูริวัชร์ โชติประดิษฐ์	วิศวกรรมเครื่องกล	17	TueMor
16	57011005	นายภูวณัฏฐ์ เลิศอัษฎมงคล	วิศวกรรมเครื่องกล	17	TueMor
17	57011034	นายยศวัจน์ เหลืองรัตนากร	วิศวกรรมเครื่องกล	17	TueMor
18	57011043	นางสาวรชกรณ์ แพงสภา	วิศวกรรมเครื่องกล	17	TueMor
19	57011065	นายรัตนะ ตอติด	วิศวกรรมเครื่องกล	17	TueMor
20	57011109	นายวรรณพงศ์ ขุนโต	วิศวกรรมเครื่องกล	17	TueMor
21	57011123	นายวรินทร นาวงษ์	วิศวกรรมเครื่องกล	17	TueMor
22	57011144	นายวสุ โกมลรัตน์มงคล	วิศวกรรมเครื่องกล	17	TueMor
23	57011148	นายวัชรพงศ์ อินทร์กลั่น	วิศวกรรมเครื่องกล	17	TueMor
24	57011154	นายวันเฉลิม พิรพัฒนภูมิ	วิศวกรรมเครื่องกล	17	TueMor
25	57011189	นายวิศรุต หยานุวัฒน์	วิศวกรรมเครื่องกล	17	TueMor
26	57011193	นายวิศิษฏ์ ตั้งเจริญ	วิศวกรรมเครื่องกล	17	TueMor
27	57011196	นายวิสสุต อนันเอื้อ	วิศวกรรมเครื่องกล	17	TueMor
28	57011214	นางสาวศนทกานต์ เหลืองวิเศษ	วิศวกรรมเครื่องกล	17	TueMor
29	57011235	นายศักย์ศรณ์ รักตพงศ์ไพศาล	วิศวกรรมเครื่องกล	17	TueMor
30	57011245	นางสาวศิโรรัตน์ ท้าวสูงเนิน	วิศวกรรมเครื่องกล	17	TueMor
31	57011248	นายศิวกร เนตรสุวรรณ์	วิศวกรรมเครื่องกล	17	TueMor
32	57011257	นายศุภกฤต สนธิศิริกฤตย์	วิศวกรรมเครื่องกล	17	TueMor
33	57011260	นายศุภกานต์ มูลหนองแวง	วิศวกรรมเครื่องกล	17	TueMor
34	57011263	นายศุภโชค ชูนิรันดร์	วิศวกรรมเครื่องกล	17	TueMor
35	57011276	นายศุภวิชญ์ ศตุงคารินทร์	วิศวกรรมเครื่องกล	17	TueMor
36	57011281	นายศุภากร ดังคุปานันท์	วิศวกรรมเครื่องกล	17	TueMor
37	57011290	นายสกล บัดสำราญ	วิศวกรรมเครื่องกล	17	TueMor
38	57011315	นายสรวิศ เอี่ยมสนธิ	วิศวกรรมเครื่องกล	17	TueMor
39	57011324	นายสวรรค์ ทองธรรมชาติ	วิศวกรรมเครื่องกล	17	TueMor
40	57011334	นายสัตยา ที่ภักดี	วิศวกรรมเครื่องกล	17	TueMor
41	57011361	นายสิรวิชญ์ รัตนเมธาชาติ	วิศวกรรมเครื่องกล	17	TueMor

รูป 4.5 รูปแบบไฟล์ .csv ที่ใช้ในการนำเข้าข้อมูลนักศึกษา

4.2.4 การแก้ไขข้อมูลส่วนตัว

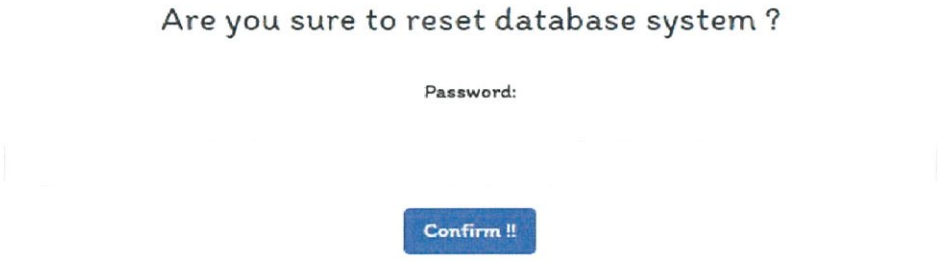
ผู้ดูแลระบบสามารถแก้ไขข้อมูลส่วนตัวของตนเองได้ โดยจะต้องทำการใส่รหัสผ่านเก่าเพื่อยืนยันตัวตนของผู้ใช้งาน และสามารถแก้ไขข้อมูลส่วนตัวที่มุมบนขวามือ



รูป 4.6 หน้าต่างแสดงการแก้ไขข้อมูลส่วนตัว

4.2.5 การลบข้อมูลทั้งหมดในฐานข้อมูล

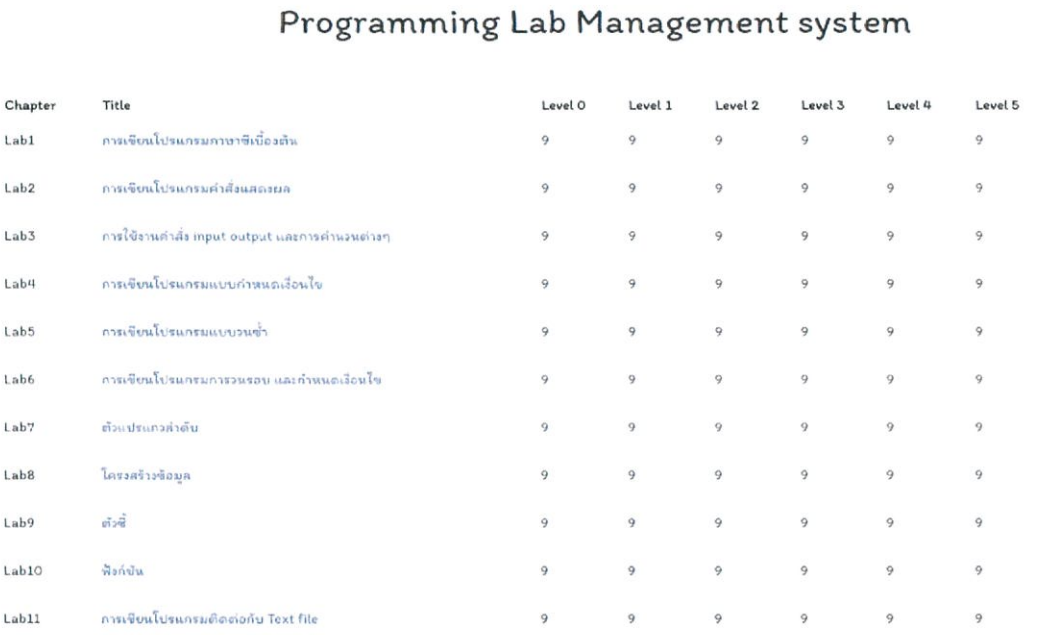
การลบข้อมูลทั้งหมดในฐานข้อมูลนั้นสามารถกระทำได้โดยผู้ใช้ที่มีสิทธิ์เป็นผู้ดูแลระบบเท่านั้น เมื่อกระทำการนี้ข้อมูลในฐานข้อมูลจะถูกลบไปทั้งหมด



รูป 4.7 หน้าต่างการการลบข้อมูลทั้งหมดในฐานข้อมูล

4.2.6 การแสดงรายการโจทย์

การแสดงรายการโจทย์นั้นสามารถกระทำได้โดยผู้ใช้ที่มีสิทธิ์เป็นอาจารย์เท่านั้น เมื่อเข้าสู่หน้ารายการโจทย์จะแสดงโจทย์ทั้งหมดที่มีอยู่ในระบบ โดยจะมีสรุปจำนวนข้อของแต่ละ Level ว่าโจทย์ Level นี้มีจำนวนข้อทั้งหมดเท่าไหร่ และสามารถฟิลเตอร์เลือกดูโจทย์เฉพาะบทได้



รูป 4.8 หน้าต่างแสดงจำนวนโจทย์

หากต้องการสร้างโจทย์ให้คลิกที่ปุ่มหากต้องการสร้างโจทย์คลิกที่นี่

รูป 4.9 หน้าต่างสำหรับกดเข้าสู่หน้าการสร้างโจทย์

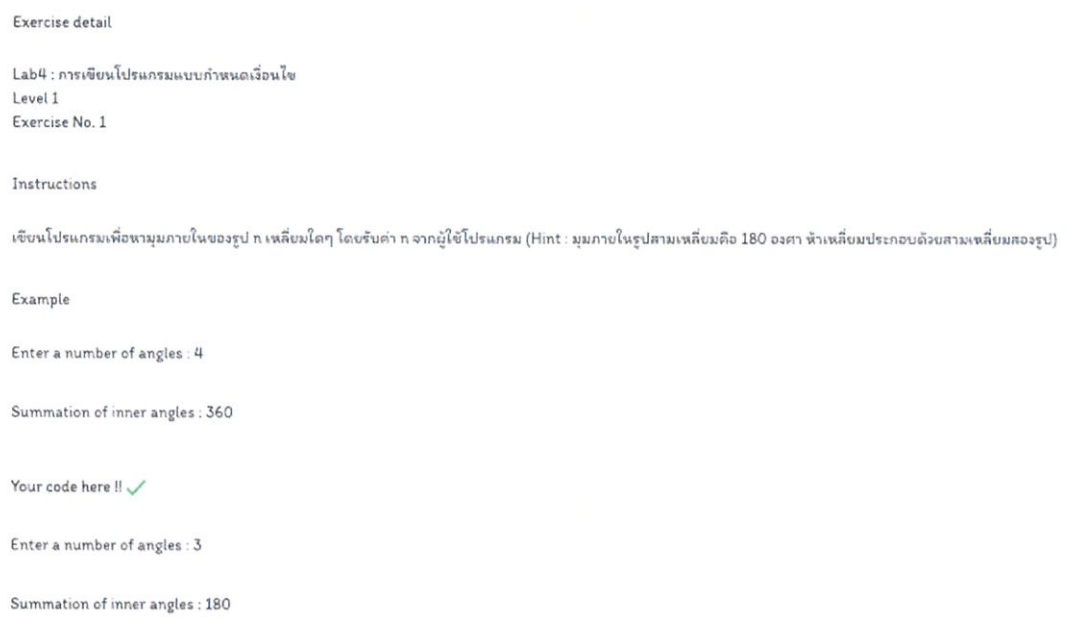
เมื่อเข้าสู่หน้าสร้างโจทย์ให้ทำการกรอกรายละเอียดข้อมูลต่าง ๆ ของโจทย์ และทำการอัพโหลดโค้ดที่ใช้เป็นต้นแบบในการตรวจ

รูป 4.10 หน้าต่างสำหรับการสร้างโจทย์

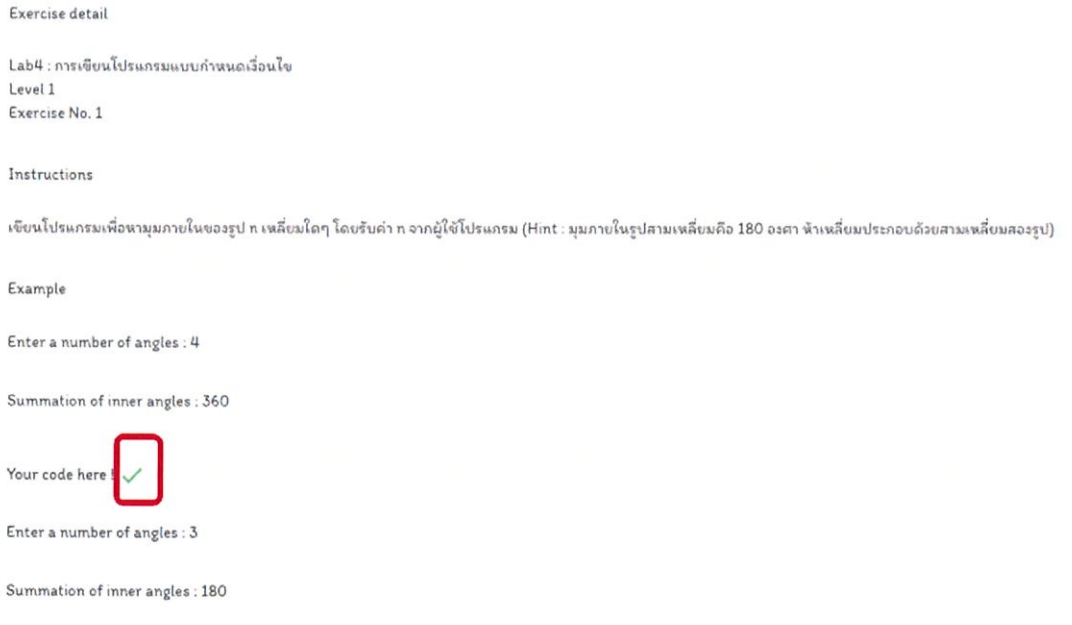
หากอาจารย์ต้องการทดสอบโจทย์ที่สร้างขึ้นให้คลิกที่ทดสอบโจทย์ในส่วนท้าย

รูป 4.11 หน้าต่างสำหรับคลิกเข้าสู่หน้าการทดสอบโจทย์

ในหน้าทดสอบโจทย์ จะแสดงตัวอย่างผลการคอมไพล์ และหากอาจารย์ต้องการทดสอบโจทย์ให้ทำการอัพโหลดไฟล์ที่จะทดสอบเข้าระบบ



รูป 4.12 หน้าต่างสำหรับการทดสอบโจทย์



รูป 4.13 หากคำตอบถูกจะแสดงเครื่องหมายถูก

อาจารย์สามารถทำการแก้ไขโจทย์ และลบโจทย์ได้

รูป 4.14 หน้าต่างสำหรับคลิกเพื่อแก้ไขและลบโจทย์

4.2.7 การแสดงคะแนนแลบของนักศึกษา

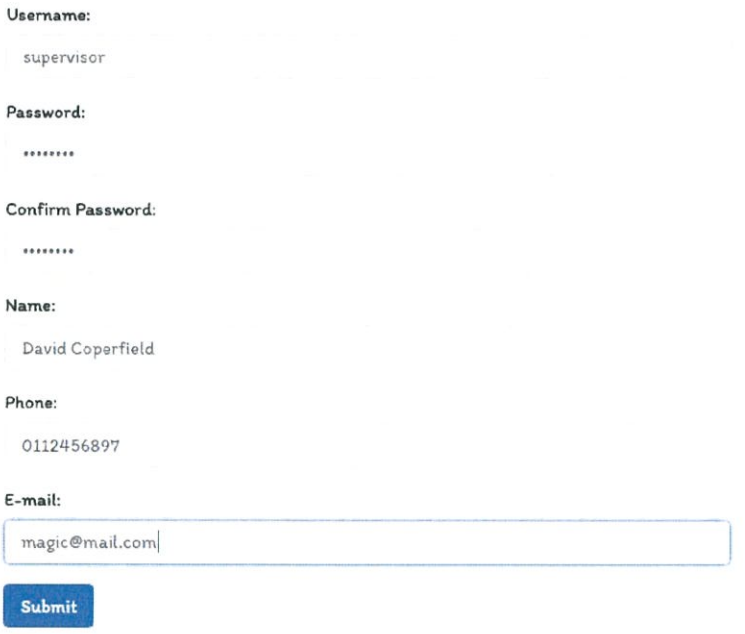
การแสดงคะแนนแลบของนักศึกษานั้นสามารถดูได้เฉพาะผู้ใช้ที่มีสิทธิ์เป็นอาจารย์เท่านั้น เมื่อเข้าสู่หน้าการแสดงคะแนนแลบของนักศึกษา จะแสดงตารางสรุปคะแนนของนักศึกษาทุกคนโดยแสดงคะแนนของแต่ละแลบ

			Score				
StudentID	Firstname	Lastname	Level 1	Level 2	Level 3	Level 4	Level 5s
55011257	Dolby	Coldzy	0	0	0	0	0
55011258	Moe	Craszy	0	0	0	0	0
55011259	Dooley	Bunnty	0	0	0	0	0

รูป 4.15 หน้าต่างแสดงคะแนนแลบของนักศึกษา

4.2.8 การแก้ไขข้อมูลส่วนตัว

การแก้ไขข้อมูลส่วนตัว อาจารย์สามารถแก้ไขข้อมูลส่วนตัวของตนเองได้ โดยทำการกรอกข้อมูลที่ต้องการแก้ไข และใส่รหัสผ่านเดิมเพื่อยืนยันตัวตน



รูป 4.16 หน้าต่างแสดงการแก้ไขข้อมูลของอาจารย์

4.2.9 การแสดงรายละเอียดการทำแลบของนักศึกษา

การแสดงรายละเอียดการทำแลบของนักศึกษาสามารถใช้ได้เฉพาะผู้ใช้ที่มีสิทธิ์เป็นนักศึกษาเท่านั้น หน้าการแสดงรายละเอียดการทำแลบของนักศึกษาจะแสดงรายละเอียดแลบทั้งหมดของนักศึกษาที่จะต้องทำในวิชานี้

Programming Lab Management system

Chapter	Title	Score	Status
Lab1	การเขียนโปรแกรมภาษาซีเบื้องต้น	0	close
Lab2	การเขียนโปรแกรมคำสั่งแสดงผล	0	close
Lab3	การใช้งานคำสั่ง input output และการคำนวนต่างๆ	0	close
Lab4	การเขียนโปรแกรมแบบกำหนดเงื่อนไข	0	close
Lab5	การเขียนโปรแกรมแบบวนซ้ำ	0	close
Lab6	การเขียนโปรแกรมการวนรอบ และกำหนดเงื่อนไข	0	close
Lab7	ตัวแปรแถวลำดับ	0	close
Lab8	โครงสร้างข้อมูล	0	close
Lab9	ตัวชี้	0	close
Lab10	ฟังก์ชัน	0	close
Lab11	การเขียนโปรแกรมติดต่อกับ Text file	0	close

รูป 4.17 หน้าต่างแสดงรายละเอียดการทำแลบ

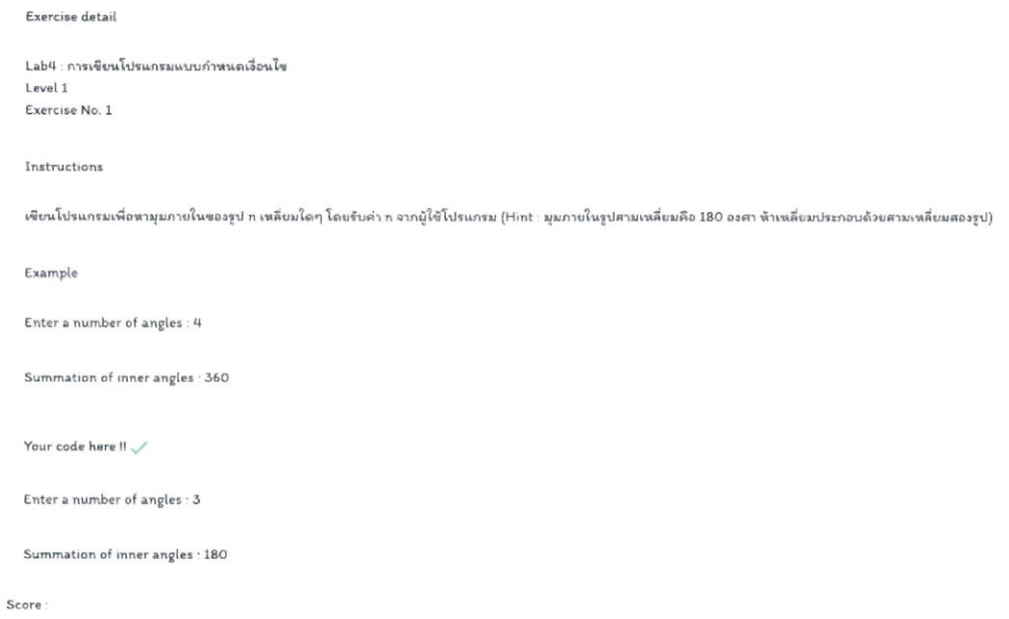
สเตตัสจะหมายถึงแลบเปิดหรือไม่ นักศึกษาจะได้เลือกเข้าไปทำแลปได้จากหน้านี้

Instructions

เขียนโปรแกรมเพื่อหามุมภายในของรูป n เหลี่ยมใดๆ โดยรับค่า n จากผู้ใช้โปรแกรม (Hint : มุมภายในรูปสามเหลี่ยมคือ 180 องศา ห้าเหลี่ยมประกอบด้วยสามเหลี่ยมสองรูป)

รูป 4.18 หน้าต่างแสดงรายละเอียดโจทย์ที่นักศึกษาต้องทำ

เมื่อเข้าสู่หน้าทำโจทย์ จะแสดงรายละเอียดโจทย์ที่นักศึกษาจะต้องทำ และแสดงผลลัพธ์ตัวอย่างของโจทย์

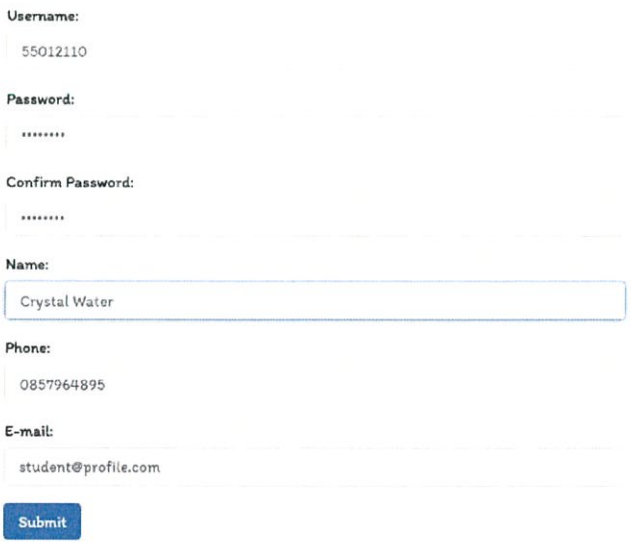


รูป 4.19 หน้าต่างแสดงรายละเอียดโจทย์และผลลัพธ์ตัวอย่าง

เมื่อนักศึกษาส่งโค้ดเข้าไปในระบบ ระบบจะแสดงผลลัพธ์การคอมไพล์โค้ดของนักศึกษาและแสดงคะแนนที่นักศึกษาได้

4.2.10 การแก้ไขข้อมูลส่วนตัวของนักศึกษา

การแก้ไขข้อมูลส่วนตัวของนักศึกษา นักศึกษาสามารถแก้ไขข้อมูลของตนเองได้ โดยจะต้องมีการใส่รหัสผ่านเก่าเพื่อยืนยันตัวตนของนักศึกษาด้วย



รูป 4.20 หน้าต่างแสดงการแก้ไขข้อมูลส่วนตัว

บทที่ 5
บทสรุป

5.1 สรุปผลการทดลอง

1) ระบบที่สร้างขึ้นมานั้นเป็นการทำขึ้นมาโดยยึดหลักของระบบเดิมบางส่วน เช่นส่วนของการล็อกอิน หรือส่วนของคอมไพเลอร์
2) จากการทดลองส่งโค้ดโปรแกรมเข้าสู่ระบบ พบว่าระบบสามารถตรวจผลลัพธ์จากโปรแกรมและให้คะแนนได้อย่างถูกต้อง เป็นการเปรียบเทียบระหว่างเอาท์พุทของโปรแกรมนักศึกษากับเอาท์พุทของโปรแกรมอาจารย์หรือโจทย์ที่กำหนด หากเอาท์พุทไม่เหมือนกันก็จะไม่ได้คะแนนในส่วนนั้น
3) โค้ดโปรแกรมที่นักศึกษาส่งเข้าไปตรวจ หากมีการรับอินพุท จำเป็นต้องมีคำนำหน้าอินพุท (input prefix) แต่ไม่จำเป็นต้องเหมือนกับตัวอย่าง และจะมีคำนำหน้าเอาท์พุท (output prefix) หรือไม่ก็ได้แล้วแต่โจทย์กำหนด แต่การรับอินพุททุกครั้งต้องมีเครื่องหมายทวิภาค (:) เสมอ เพื่อใช้สำหรับตรวจจับอินพุท
4) ในการจัดตารางเรียนแลบโปรแกรมมิ่งนั้น เพื่อให้นักศึกษาทุกคนปฏิบัติได้ และอาจารย์สามารถเปิด-ปิดกลุ่มเรียนได้
5) ระบบมีการเพิ่มความปลอดภัยให้กับผู้ใช้งาน โดยมีระบบอัพโหลดรูปประจำตัว ไว้อัพโหลดรูปถ่ายหรือรูปแสกนบัตรนักศึกษา เพื่อตรวจสอบตัวตน และมีระบบเปลี่ยนแปลง Password ได้
6) ระบบออกแบบการป้องกันการลอกกันของนักศึกษา โดยนักศึกษาที่นักติดกันจะไม่สามารถได้โจทย์ที่เหมือนกันได้

5.2 ปัญหา อุปสรรค และแนวทางการแก้ไข

1) กรณีนักศึกษาลืมรหัสผ่าน นักศึกษาต้องแจ้งผู้ดูแลระบบโดยตรงทำให้เกิดความล่าช้าในการแก้ไข ดังนั้น ควรที่จะให้สิทธิการดูแลฐานข้อมูลของระบบในส่วนข้อมูลประจำตัวนักศึกษาให้แก่อาจารย์ผู้คุมแลบ
2) ระบบไม่สามารถตรวจรูปแบบการเขียนโค้ดโปรแกรมของนักศึกษาได้ แต่จุดประสงค์ของการเขียนโค้ดที่ดี นอกจากได้ผลลัพธ์ที่ถูกต้องแล้ว โค้ดโปรแกรมควรจะมีความเป็นระเบียบเพื่อจะได้อ่านทำความเข้าใจได้ง่าย และฝึกให้นักศึกษาเป็นผู้พัฒนาโปรแกรมที่มีคุณภาพ
3) ระบบยังไม่สามารถตรวจโค้ดโปรแกรมที่มีข้อห้ามหรือเงื่อนไขเพิ่มเติมได้ เช่น ต้องใช้

ฟังก์ชัน (function), ให้ใช้ตัวแปรประเภท pointer เท่านั้น เป็นต้น ดังนั้นระบบควรเพิ่มความสามารถโดยใช้ Lexical analyzer ในการวิเคราะห์โครงสร้างของโปรแกรมหรือไวยากรณ์ของโปรแกรม

4) เนื่องจากโปรแกรมมีผลลัพธ์หลากหลายรูปแบบมาก ทำให้ระบบตรวจได้ไม่ครอบคลุมในโจทย์ทุก ๆ ประเภท จึงควรพัฒนาระบบให้ตรวจรูปแบบผลลัพธ์ที่หลากหลายมากขึ้น
5) ในส่วนของโค้ดภาษาพีเอชพีที่ใช้ในการสร้างระบบนั้นมีความยุ่งยากซับซ้อน ยากต่อการเข้าใจ เนื่องจากขาดการจัดเรียงในแต่ละส่วนให้เป็นระเบียบ ทำให้ยากต่อการทำความเข้าใจและพัฒนาต่อ ดังนั้นจึงควรแบ่งหมวดหมู่การทำงานของโค้ดออกเป็นส่วน ๆ เพื่อให้เข้าใจได้ง่าย
6) หากโปรแกรมของนักศึกษาที่ส่งมาเป็นแบบลูปที่ไม่มีวันจบสิ้นหรืออินฟินิติจะทำให้ตัวระบบอาจค้างได้

5.3 แนวทางในการพัฒนาต่อไป

1) ให้เพิ่มความสามารถในการตรวจรูปแบบผลลัพธ์ของระบบให้หลากหลายมากขึ้นได้ เช่น กรณีที่ผลลัพธ์ของแต่ละบุคคลไม่จำเป็นต้องเหมือนกัน ซึ่งโจทย์วัดการจัดรูปแบบการแสดงของผลลัพธ์ จึงอาจมีผลลัพธ์ที่แตกต่างกันไป
2) ให้สามารถตรวจรูปแบบการเขียนโค้ดโปรแกรม เช่น การย่อหน้า การจัดลำดับความสำคัญของโค้ด เพื่อฝึกให้นักศึกษาเป็นผู้พัฒนาโปรแกรมที่ดีต่อไปในอนาคตได้
3) สามารถพัฒนาระบบให้สามารถตรวจโค้ดภาษาอื่น ๆ ได้นอกเหนือจากโค้ดโปรแกรมภาษาซี เช่น ภาษาซีพลัสพลัส ภาษาจาวา จึงจำเป็นต้องติดตั้งคอมไพเลอร์หรือเขียนโปรแกรมเพิ่มความสามารถให้แก่ระบบมากยิ่งขึ้น
4) ในระบบการแสดงผลของโปรแกรมควรจัดทำให้มีรูปแบบที่แน่นอน ไม่แปรเปลี่ยนตามค่า resolution หรือค่า graphics ในแต่ละเครื่อง
5) ทำส่วนติดต่อผู้ใช้งานให้ง่ายขึ้น

5.4 สรุประบบที่สร้างและสถานะโครงงาน

ตาราง 5.1 สรุประบบที่สร้างและสถานะโครงงาน

Supervisor	Priority	Status
1. Log in	5	100%
2. Log out	5	100%
3. เพิ่มโจทย์เข้าสู่ระบบ	5	100%
4. แก้ไขโจทย์ภายในระบบ	5	100%
5. คัดลอกโจทย์ในกรณีที่สร้างโจทย์คล้ายโจทย์เก่า	3	0%
6. ดูโจทย์ในแต่ละบท	5	100%
7. ดูโจทย์ในแต่ละlevel	5	100%
8. ทดสอบโจทย์ในแต่ละข้อภายในระบบ	5	100%
9. ดูจำนวนโจทย์ทั้งหมดภายในระบบ	5	100%
10. เปิด/ปิดแลป และการสุ่มโจทย์อัตโนมัติ	5	100%
11. ดูว่านักศึกษาแต่ละคนนั้นได้โจทย์ข้อไหนที่แจกให้	5	100%
12. ดูสถานะlog in และข้อมูลของนักศึกษาในระบบ	5	0%
13. ดูสรุปคะแนนนักศึกษาของแต่ละกลุ่มทุกแลป	5	0%
14. class information config lab เบื้องต้น	2	0%
15. ประวัติการส่งงานของนักศึกษาแต่ละคน	5	0%
16. สถานะปัจจุบัน	5	0%
17. การแก้ไขข้อมูลส่วนตัวของตัวเอง	5	0%
18. reset password ของนักศึกษาภายในระบบ	4	75%

ตาราง 5.1 สรุประบบที่สร้างและสถานะโครงงาน(ต่อ)

Student	Priority	Status
1. log in	5	100%
2. log out	5	100%
3. เลือกที่นั่งในการทำโจทย์	3	0%
4. ดูประวัติการส่งงานในแต่ละแลป	5	100%
5. กดเลือกเพื่อการทำโจทย์	5	100%
6. แก้ไขข้อมูลส่วนตัว และ upload รูป	5	0%
7. แก้ไขที่นั่งในการทำโจทย์	3	0%
Admin	**Priority**	**Status**
1. log in	5	100%
2. log out	5	100%
3. นำรายชื่อนักศึกษาเข้าสู่ระบบ	5	50%
4. เพิ่ม/แก้ไข user ต่าง ๆ ภายในระบบ	5	0%
5. ย้ายกลุ่มนักศึกษาภายในระบบ	3	75%
6. backup ข้อมูลของระบบ	5	75%
7. ลบ/reset รายชื่อนักศึกษาและข้อมูลที่เกี่ยวข้องภายในระบบ	5	75%
8. แก้ไขสิทธิของ supervisor	3	0%
9. ดู account ในระบบและกดดูตาม permission	2	0%

5.5 คำถามที่พบบ่อย (FAQ)

1) หากนักศึกษาส่งโปรแกรม แล้วปรากฏว่า ผลลัพธ์คะแนนออกมาเป็น 0 อาจเกิดจากกรณีใดได้บ้าง ?

ตอบ : ผลลัพธ์ที่ได้จากการ Compile โปรแกรมที่นักศึกษาส่งผิดทำให้นักศึกษาไม่ได้คะแนนในกรณีทดสอบต่างๆ

ตอบ : อาจเกิดจากรูปแบบ หรือแบบฟอร์มการแสดงผลลัพธ์ของนักศึกษาผิดไปจากแบบฟอร์มที่กำหนด (ควรศึกษาเพิ่มเติมในส่วนของการแสดงผล)

2) สามารถเข้าสู่ระบบการจัดการ Database ได้อย่างไร ?

ตอบ : เข้าไปที่ http://localhost/ เลือก PHPMyAdmin

3) Admin สามารถทำการแก้ไขข้อมูล user ได้อย่างไร ?

ตอบ : เลือก เมนู แก้ไขข้อมูลผู้ใช้ จากนั้นเลือกประเภทผู้ใช้งาน admin, supervisor, student แล้วจึงทำการเลือก user ที่ต้องการแก้ไข

4) การนำเข้าข้อมูลลงทะเบียนนักศึกษาทำได้อย่างไร ?

ตอบ : เตรียมไฟล์นำเข้าเป็นไฟล์สกุล .csv จากนั้นนำไฟล์มา import เข้าสู่ระบบ

5) Supervisor สร้างโจทย์ แล้วต้องการให้นักศึกษาทำโจทย์ต้องทำอย่างไร ?

ตอบ : ต้องเปิด เซสชั่น เพื่อระบุวัน เวลา และ โจทย์ที่ต้องการให้นักศึกษาทำ

6) Supervisor สามารถกำหนดให้นักศึกษาแต่ละคนทำไม่เหมือนกันได้อย่างไร ?

ตอบ : เปิดเซสชั่น 3 เซสชั่น กำหนดโจทย์ในแต่ละเซสชั่นเป็นคนละข้อกัน

7) หาก Supervisor ต้องการกำหนดหลักเกณฑ์การให้คะแนนนักศึกษาต้องทำอย่างไร ?

ตอบ : ตั้งค่าการเพิ่มกรณีทดสอบ เพื่อการให้คะแนนนักศึกษา ตอนสร้างโจทย์

8) Supervisor สามารถกำหนดเวลา เปิด-ปิดเซสชั่น เพื่อให้นักศึกษาทำโจทย์ได้หรือไม่ ?

ตอบ : ทำได้โดยการเลือกที่เมนู รายการเซสชั่น แล้วเลือกแก้ไข

9) หากนักศึกษาลืมรหัสผ่านต้องทำอย่างไร ?

ตอบ : ติดต่อทางผู้ดูแลระบบ (Admin)

10) นักศึกษาสามารถเรียนรู้วิธีการใช้งานระบบได้อย่างไร ?

ตอบ : สามารถ Download คู่มือการใช้งานที่หน้า Homepage

11) ในขณะที่นักศึกษาต้องการส่งไฟล์ source code ต้องเป็นไฟล์ประเภทใด ?

ตอบ : ไฟล์ที่มีสกุล .c เท่านั้น

12) โครงงานนี้มีระบบตรวจสอบหรือบังคับแนวทางการเขียนโค้ดของนักศึกษาหรือไม่ ? (เช่น ห้ามเขียนโค้ด Printf แสดงผลตามผลลัพธ์ แต่ให้ใช้ for loop แทน)

ตอบ : ทำได้โดยการกำหนดค่าตอนสร้างโจทย์/กำหนดในส่วนของเงื่อนไข

บรรณานุกรม

บัญชา ปะสีละเตสัง. 2555. พัฒนาเว็บแอปพลิเคชั่นด้วย PHP ร่วมกับ MySQL และ Dreamweaver. กรุงเทพฯ : ซีเอ็ดยูเคชั่น.

อนรรฆนงค์ คุณมณี. 2555. พัฒนา Web App แบบมืออาชีพด้วย PHP + AJAX และ JQUERY. นนทบุรี : บริษัท ไอดีซี พรีเมียร์ จำกัด.

อาณัติ รัตนถิรกุล. 2549. ก้าวสู่อาชีพผู้ดูแลระบบเครือข่ายคอมพิวเตอร์ในองค์กร (ภาคปฏิบัติ). กรุงเทพฯ : ซีเอ็ดยูเคชั่น.

Johnson and Vlissides. 2012. "Model-View-Controller" [Online]. Available : http://msdn.microsoft.com/en-us/library/ff649643.aspx.

Refsnes Data. 1999. PHP Tutorial. [Online]. Available : http://www.w3schools.com/php/default.asp.

www.stackoverflow.com

http://www.codeigniter.com/user guide/

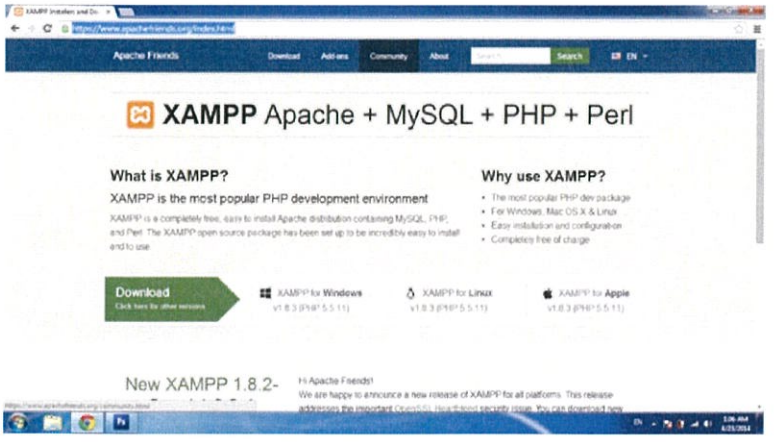
ภาคผนวก ก
การติดตั้งระบบ

ก.1 โปรแกรมที่ใช้ในการติดตั้งระบบ

ระบบจัดการห้องแลบโปรแกรมมิ่งมีการใช้งานโปรแกรม XAMPP ซึ่งเป็นโปรแกรมสำหรับจำลองเครื่องคอมพิวเตอร์ส่วนบุคคลของเรา ให้ทำงานในลักษณะของ Web Server นั่นคือเครื่องคอมพิวเตอร์ของเราจะเป็นทั้งเครื่องแม่ และเครื่องลูกในเครื่องเดียวกัน ทำให้ไม่ต้องเชื่อมต่อกับ Internet คุณก็สามารถทดสอบเว็บไซต์ที่คุณสร้างขึ้น ได้ทุกที่ทุกเวลา ปัจจุบันได้รับความนิยมจากผู้ใช้ CMS ในการสร้างเว็บไซต์ ฉะนั้นในการใช้งานระบบจะต้องมีการติดตั้งโปรแกรม XAMPP ก่อนจึงจะสามารถใช้งานได้ XAMPP ประกอบด้วย Apache, PHP, MySQL, PHP MyAdmin, Perl ซึ่งเป็นโปรแกรมพื้นฐานที่รองรับการทำงาน CMS ซึ่งเป็นชุดโปรแกรม สำหรับออกแบบเว็บไซต์ที่ได้รับความนิยมในปัจจุบัน ไฟล์สำหรับติดตั้ง xampp นั้นอาจมีขนาดใหญ่สักหน่อย เนื่องจาก มีชุดควบคุมการทำงานที่ช่วยให้การปรับแต่งส่วนต่าง ๆ ง่ายขึ้น XAMPP นั้นรองรับระบบปฏิบัติการหลายตัว เช่น Windows, Linux, Apple ทำงานได้ทั้งบนระบบปฏิบัติการแบบ 32 bit และ 64 bit สิ่งที่โดดเด่นกว่าโปรแกรมอื่นคือมีตัวช่วยติดตั้ง CMS ที่เรียกว่า BitNami ซึ่งช่วยให้คุณติดตั้ง CMS รุ่นใหม่ ๆ ที่ได้รับความนิยมอีกด้วย

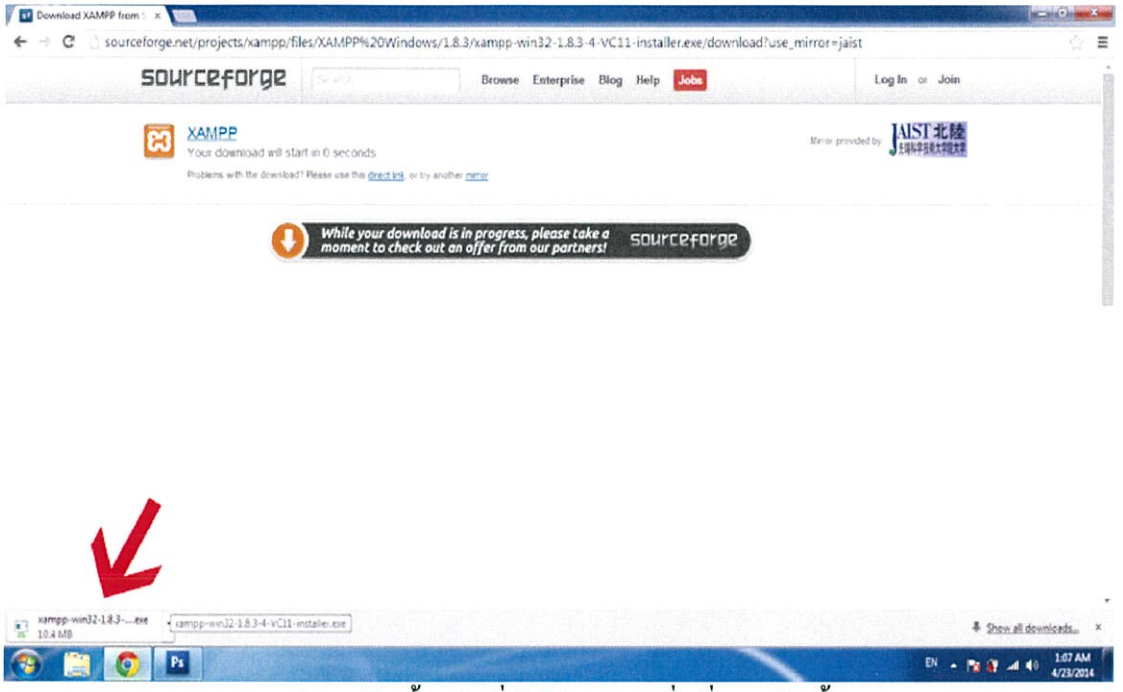
ก.2 ขั้นตอนการติดตั้ง

1) ดาวน์โหลดโปรแกรม XAMPP จากเว็บไซต์ https://www.apachefriends.org โดยเลือกเวอร์ชันที่ต้องการติดตั้ง



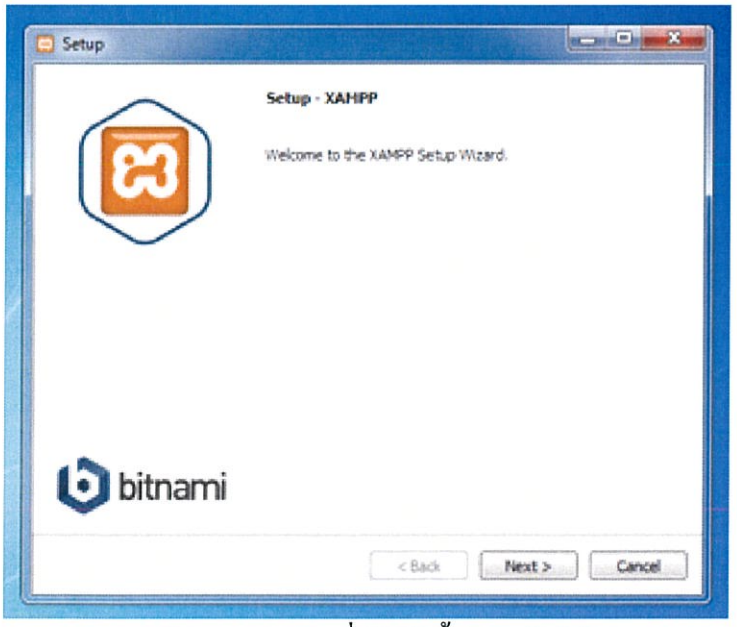
รูป ก.1 ทำการดาวน์โหลดโปรแกรม XAMPP จากเว็บไซต์

2) เมื่อดาวน์โหลดเสร็จเรียบร้อยดับเบิ้ลคลิกที่ตัวโปรแกรม



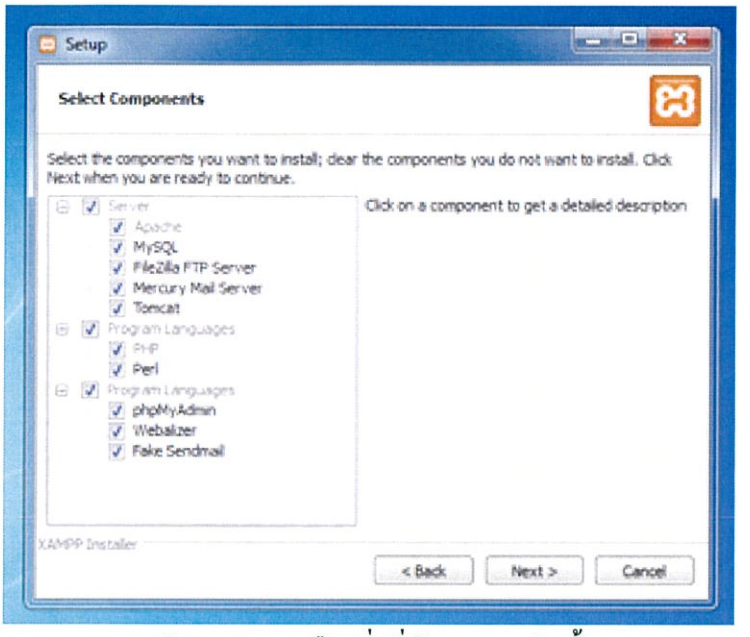
รูป ก.2 ดับเบิ้ลคลิกที่ตัวโปรแกรมเพื่อเริ่มการติดตั้ง

3) กด next เพื่อไปยังขั้นตอนถัดไป



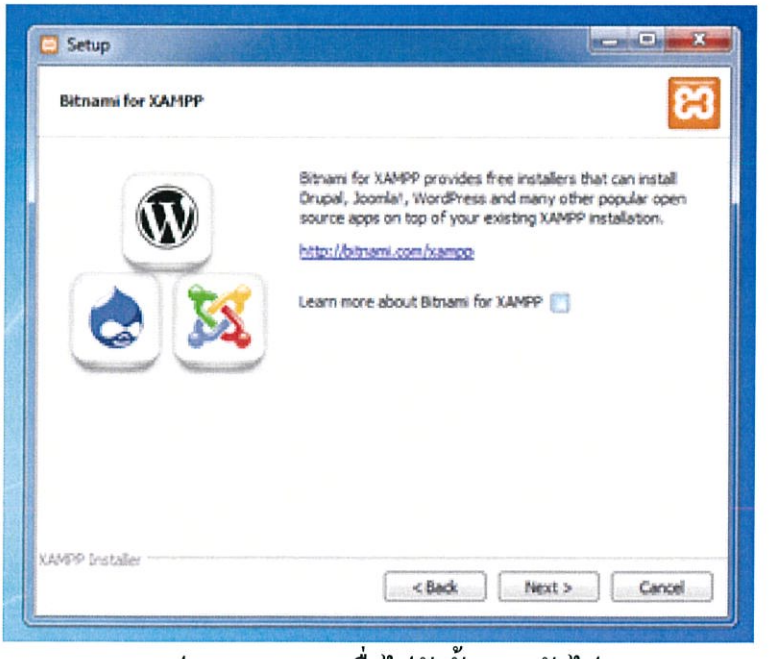
รูป ก.3 กด next เพื่อไปยังขั้นตอนถัดไป

4) ระบบจะมีตัวเลือกให้ผู้ใช้เลือกว่าต้องการติดตั้งสิ่งใดลงในระบบบ้าง เมื่อเลือกเสร็จเรียบร้อยให้กด next เพื่อไปยังขั้นตอนถัดไป



รูป ก.4 ทำการเลือกสิ่งที่ต้องการจะติดตั้ง

5) กด next เพื่อไปยังขั้นตอนถัดไป

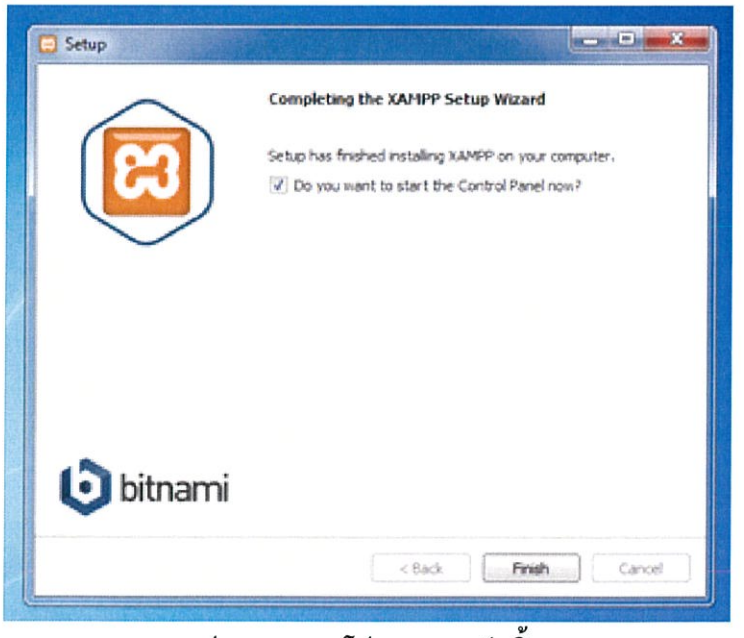


รูป ก.5 กด next เพื่อไปยังขั้นตอนถัดไป

6) ระบบจะเข้าสู่การติดตั้งตัวโปรแกรม XAMPP

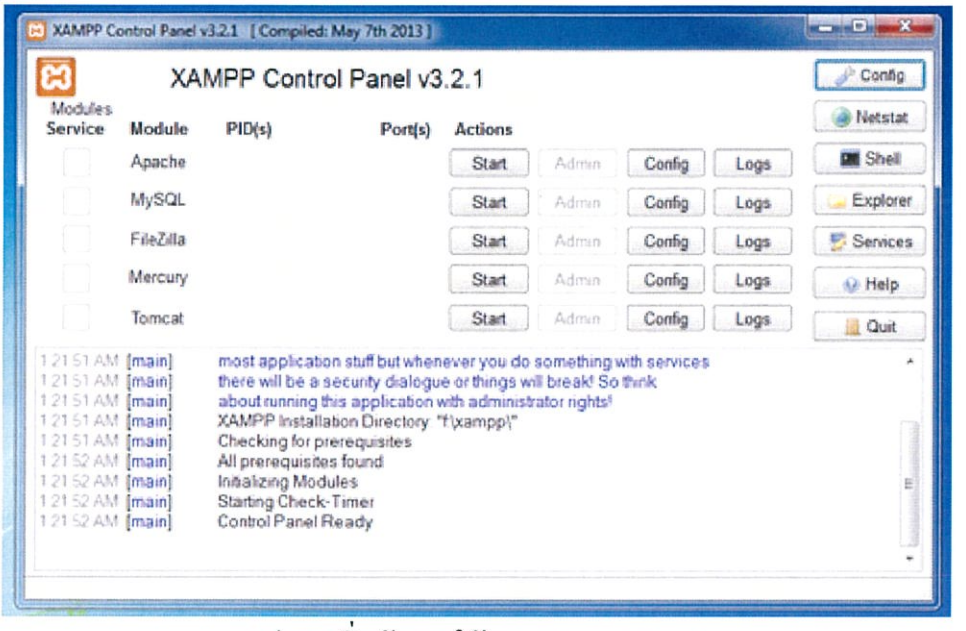
รูป ก.6 เข้าสู่สถานการณ์ติดตั้งโปรแกรม

7) เมื่อติดตั้งโปรแกรมเสร็จเรียบร้อยจะขึ้นหน้าต่างดังรูป



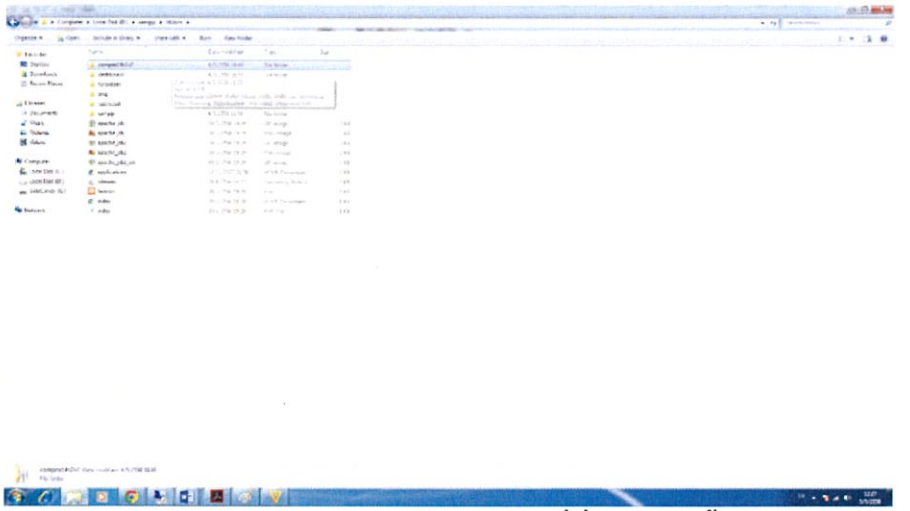
รูป ก.7 การลงโปรแกรมเสร็จสิ้น

8) ทดลองการใช้งานส่วน Control Panel กด start ที่ Apache และ MySQL เพื่อเริ่มต้นใช้งาน



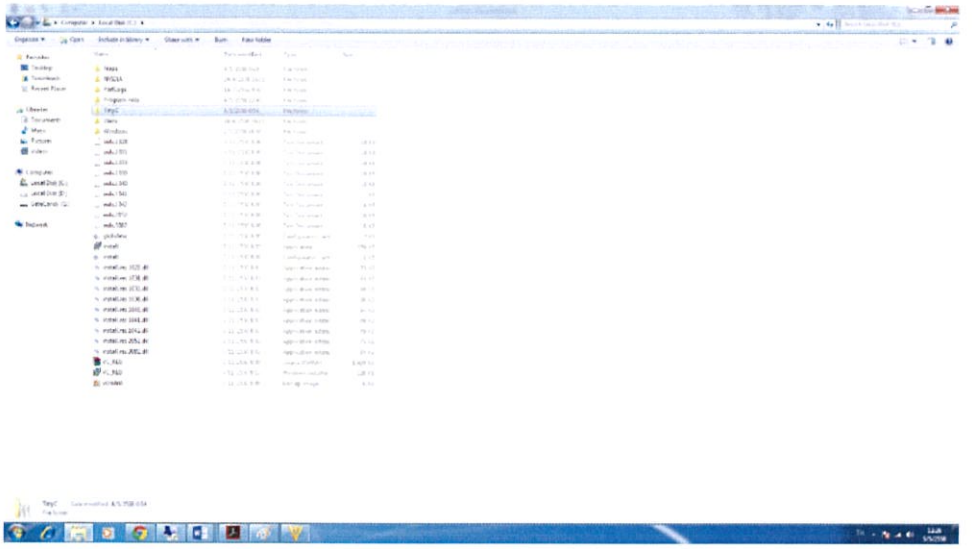
รูป ก.8 เริ่มต้นการใช้งาน Control Panel

9) นำโฟลเดอร์ compro14s2v2 มาใส่ใน xampp/htdocs



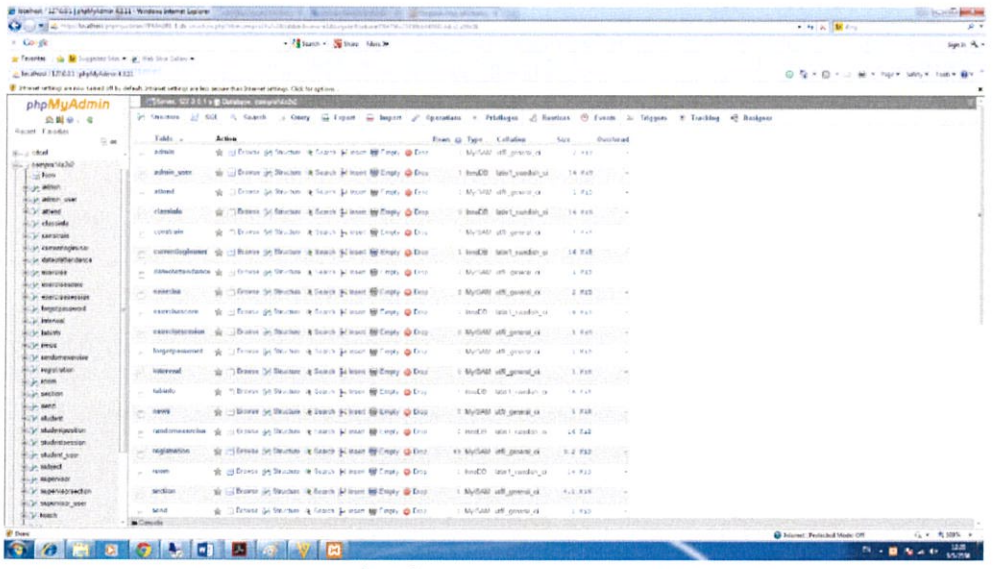
รูป ก.9 นำโฟลเดอร์โปรเจคมาใส่ในไดเรคทอรี่ที่ทำการติดตั้งโปรแกรม

10) นำโฟลเดอร์ TinyC มาใส่ไว้ในไดร์ฟ C



รูป ก.10 นำโฟลเดอร์ TinyC มาใส่ในไดร์ฟ C

11) Import ไฟล์ compro14s2v2.sql เข้าสู่ฐานข้อมูล

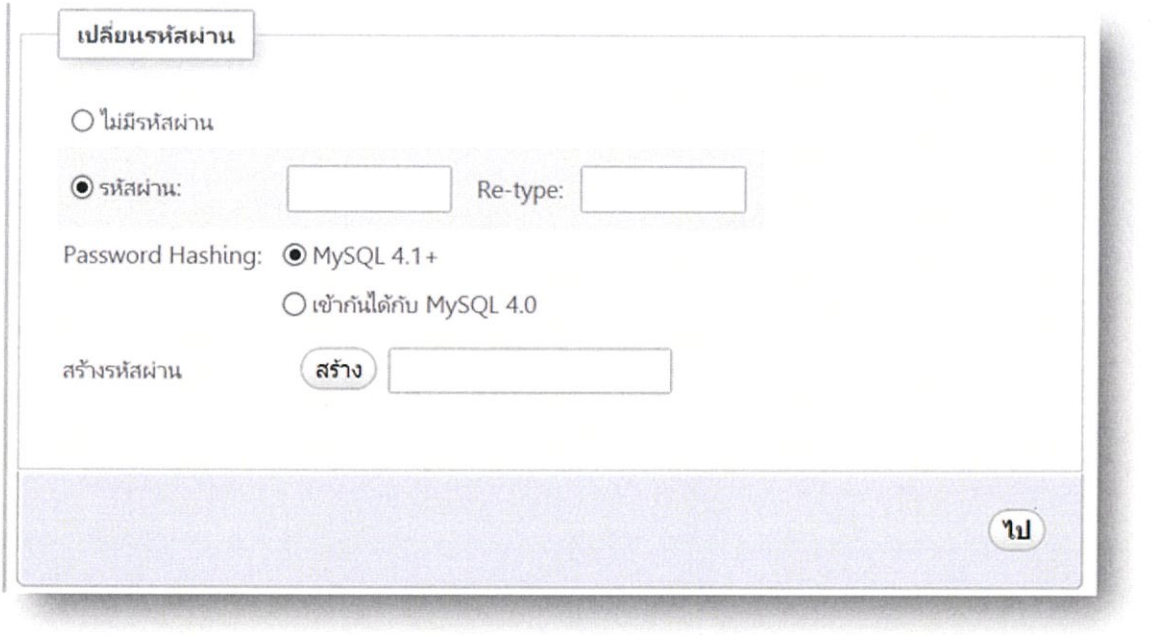


รูป ก.11 Import ไฟล์ compro14s2v2.sql เข้าสู่ฐานข้อมูล

12) ไปที่ Privileges และคลิกที่ Edit Privileges ของ localhost

รูป ก.12 ทำการเข้าสู่หน้าแก้ไข password

13) ทำการแก้ไข password เป็น 2536



รูป ก.13 ทำการแก้ไข password ใน MySQL

14) แก้ไขการตั้งค่าในไฟล์ใน phpMyAdmin/config.inc.php โดยแก้ password เป็น 2536

```
/* Authentication type and info */
$cfg['Servers'][$i]['auth_type'] = 'config';
$cfg['Servers'][$i]['user'] = 'root';
$cfg['Servers'][$i]['password'] = '2536';
$cfg['Servers'][$i]['extension'] = 'mysqli';
$cfg['Servers'][$i]['AllowNoPassword'] = true;
$cfg['Lang'] = '';
```

รูป ก.14 ทำการแก้ไข password ในไฟล์ config.inc.php

15) แก้ไขการตั้งค่าในไฟล์ใน php/php.ini โดยแก้ short_open_tag เป็น On และ แก้ date.timezone เป็น Asia/Bangkok

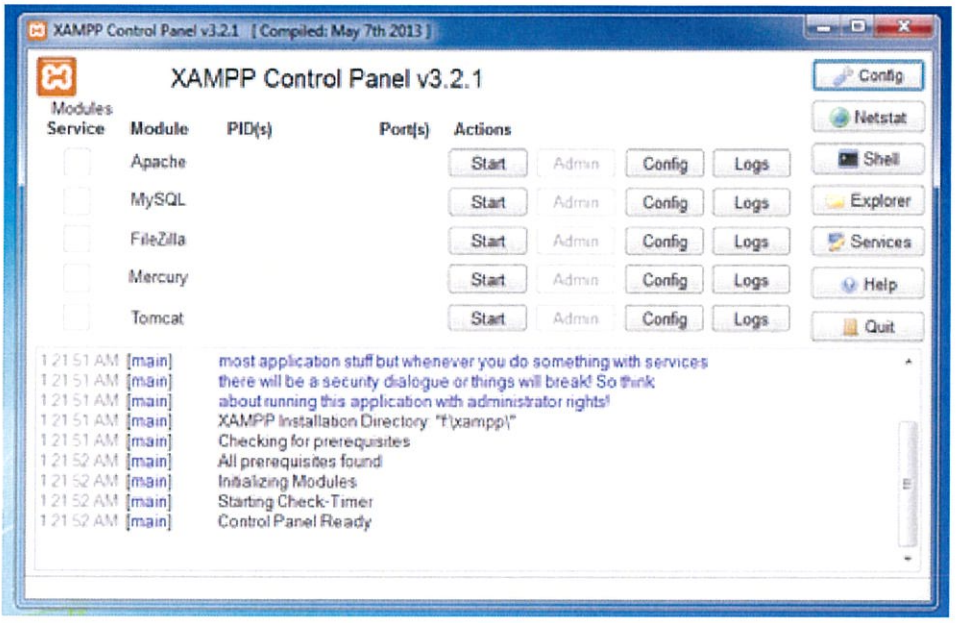
```
;short_open_tag=Off
; XAMPP for Linux is currently old fashioned
short_open_tag = On
```

รูป ก.15 ทำการแก้ไข short_open_tag เป็น On

```
[Date]
; Defines the default timezone used by the date functions
; http://php.net/date.timezone
date.timezone=Asia/Bangkok
```

รูป ก.16 ทำการแก้ไข date.timezone เป็น Asia/Bangkok

16) ทำการ restart Apache และ MySQL



รูป ก.17 restart Apache และ MySQL