การพิสูจน์ตัวตนด้วยจังหวะการพิมพ์
KEYSTROKE DYNAMICS AUTHENTICATION

โดย
นายขจรยศ สีกล่อม
นายคมสันต์ เจนใจ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

การพิสูจน์ตัวตนด้วยจังหวะการพิมพ์
KEYSTROKE DYNAMICS AUTHENTICATION

โดย

นายขจรยศ	สีกล่อม	57010124
นายคมสันติ์	เจนใจ	57010139

อาจารย์ที่ปรึกษา

ผศ.ดร.กฤษณ์ วงรุจิระ

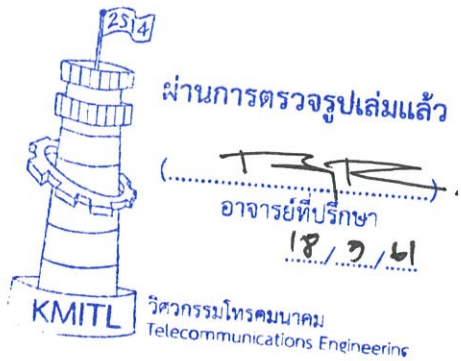
ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

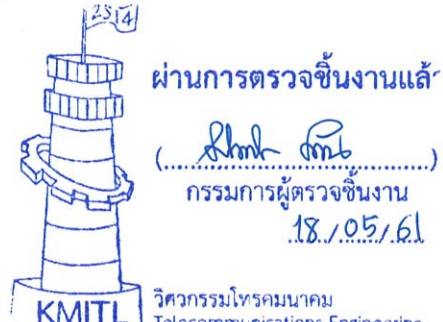
ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560





ปริญญานิพนธ์ปีการศึกษา 2560
ภาควิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง การพิสูจน์ตัวตนด้วยจังหวะการพิมพ์
KEYSTROKE DYNAMICS AUTHENTICATION

ผู้จัดทำ
1. นายขจรยศ สีกล่อม 57010124
2. นายคมสันติ์ เจนใจ 57010139

..... อาจารย์ที่ปรึกษา
(ผศ.ดร.กฤษณ์ วงรุจิระ)

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จลุล่วงไปได้ด้วยดี เนื่องจากได้รับความเมตตากรุณาจากอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.กฤษณ์ วงรุจิระ ที่คอยให้คำแนะนำ คำสั่งสอน ที่เป็นประโยชน์ตลอดระยะเวลาในการทำปริญญานิพนธ์นี้ ขอขอบพระคุณท่านในความห่วงใยและความหวังดีให้แก่ผู้จัดทำเป็นอย่างยิ่ง

ขอขอบพระคุณอาจารย์ประจำสาขาวิชาโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่าน ที่คอยช่วยสั่งสอนและให้คำแนะนำในการเรียนมาตลอดเป็นระยะเวลา 4 ปี

ขอขอบคุณเพื่อน พี่ และน้อง ทุกคนที่คอยช่วยเหลือ สละเวลาในการเก็บข้อมูล โดยเฉพาะอย่างยิ่งขอขอบคุณพ่อ แม่ ญาติพี่น้องที่คอยอยู่เป็นกำลังใจ จนกระทั่งปริญญานิพนธ์นี้สำเร็จลุล่วงไปได้ด้วยดี

นายขจรยศ สีกล่อม
นายคมสันต์ เจนใจ
ผู้จัดทำ

การพิสูจน์ตัวตนด้วยจังหวะการพิมพ์
KEYSTROKE DYNAMICS AUTHENTICATION

โดย นายขจรยศ สีกล่อม 57010124
นายคมสันติ์ เจนใจ 57010139

อาจารย์ที่ปรึกษา ผศ.ดร.กฤษณ์ วงรุจิระ

บทคัดย่อ

ในปัจจุบันการรักษาความปลอดภัยของเว็บไซต์ต่างๆผ่านระบบคอมพิวเตอร์มีหลายรูปแบบด้วยกัน โดยทั่วไปจะใช้ Username และ Password ในการล็อกอินเข้าเว็บไซต์เพื่อระบุตัวตนผู้ใช้งาน ซึ่งวิธีนี้มีความเสี่ยงหากมีผู้อื่นรู้ Username และ Password ของระบบ อาจนำมาซึ่งความเสียหายต่างๆได้ ปริญญานิพนธ์นี้จึงได้มีแนวคิดที่จะเสริมสร้างความปลอดภัยของระบบโดยการผนวกจังหวะการพิมพ์ (Keystroke Dynamics) ซึ่งเป็นมาตรวัดทางชีวะแบบพฤติกรรมอย่างหนึ่ง โดยใช้ คำภาษาไทย สำหรับ Username และ Password โดยได้นำการวิเคราะห์และศึกษาพฤติกรรมจังหวะการพิมพ์ของ Username และ Password ประยุกต์ใช้การวิเคราะห์ความแตกต่างของเส้นเวลาจังหวะการพิมพ์ เพื่อใช้เป็นอัลกอริทึมหลัก ในการพิสูจน์ตัวตน

ABSTRACT

Nowadays, the security of websites through computer systems has various ways of authentication. In general, system login with "Username" and "Password" has been intensive used to authentication the user. This method is flawed in security point when someone knows about login name and password of the user. This thesis has the concept to enhance the security of the system by integrating Keystroke Dynamics, a measure of biological behavior in sequence and time, with the unique of combination sequence in typing of Thai word, the time of interchange between pressing a key and hold a key. We are able to extract this sequence with time stamp between stroke and time duration. Then, the rithm of key stroke sequence will be used to analysis and authenticate a user.

สารบัญ

		หน้า
กิตติกรรมประกาศ		I
บทคัดย่อ		II
สารบัญ		III
สารบัญรูป		V
สารบัญตาราง		VI
บทที่ 1	**บทนำ**	**1**
	1.1 ความเป็นมาและความสำคัญของปัญหา	1
	1.2 วัตถุประสงค์	1
	1.3 ขอบเขตของปริญญานิพนธ์	1
บทที่ 2	**ทฤษฎีและหลักการที่เกี่ยวข้อง**	**2**
	2.1 BIOMETRIC	2
	2.2 KEYSTROKE DYNAMICS	3
	2.3 การพิสูจน์ตัวตนผ่านลำดับการกดแป้นพิมพ์ของคำภาษาไทย	4
	2.4 ทฤษฎีที่ใช้ในการคำนวณ	6
	2.5 ภาษาจาวาสคริปต์	10
	2.6 ภาษาไพธอน	12
	2.7 ภาษาเฮชทีเอ็มแอล	14
	2.8 ภาษาพีเอชพี	16
	2.9 พีเอชพีมายแอดมิน	18
	2.10 มายเอสคิวแอล	19
บทที่ 3	**การออกแบบและการจัดทำปริญญานิพนธ์**	**24**
	3.1 การออกแบบ	24
	3.2 เครื่องมือที่ใช้ในการทดลอง	30
	3.3 การจัดเก็บผลการทดลอง	30

สารบัญ (ต่อ)

		หน้า
บทที่ 4	**ผลการทดลอง**	**32**
	4.1 ผลการทดสอบการเก็บค่าจังหวะการพิมพ์	32
	4.2 ผลการทดสอบการคำนวณหาค่าเวลาในการกดคีย์บนแป้นพิมพ์	32
	4.3 ผลการทดสอบลำดับคำในการกดคีย์บนแป้นพิมพ์	34
	4.4 ผลการทดสอบการคำนวณหาค่าเส้นเวลาจังหวะการพิมพ์ต้นแบบ	37
	4.5 ผลการทดสอบการคำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์	38
	4.6 ผลการทดสอบการพิสูจน์ตัวตน	41
	4.7 ผลการทดสอบใช้งานเว็บไซต์ล็อกอินต้นแบบสำหรับระบบการพิสูจน์ตัวตนด้วยจังหวะการพิมพ์	50
บทที่ 5	**สรุปผลและข้อเสนอแนะ**	**56**
	5.1 สรุปผล	56
	5.2 ข้อเสนอแนะ	56
บรรณานุกรม		**57**
ภาคผนวก ก	โค๊ดเว็บไซต์เก็บค่าจังหวะการพิมพ์	59
ภาคผนวก ข	โค๊ดโปรแกรมคำนวณหาค่าจังหวะการพิมพ์เพื่อจัดเก็บลงฐานข้อมูล	63
ภาคผนวก ค	โค๊ดโปรแกรมพิสูจน์ตัวตนด้วยจังหวะการพิมพ์	70
ภาคผนวก ง	โค๊ดเว็บไซต์ล็อกอินต้นแบบ สำหรับระบบการพิสูจน์ตัวตนด้วยจังหวะการพิมพ์	76
ภาคผนวก จ	ผลการทดสอบประสิทธิภาพ	102

สารบัญรูป

รูปที่		หน้า
2.1	Feature พื้นฐานของ Keystroke Dynamics	4
2.2	ลำดับการพิมพ์คำว่า “ซ้ำเติม” จากผู้ทดลอง 20 คน	5
2.3	ลำดับการพิมพ์คำว่า “ที่อยู่” จากผู้ทดลอง 20 คน	5
2.4	วัดความแตกต่างของเส้นเวลาจังหวะการพิมพ์ด้วยวิธีวัดระยะทางแบบยูคลิด	8
2.5	โลโก้ของจาวาสคริปต์ (JavaScript)	10
2.6	โลโก้ของไพธอน (Python)	12
2.7	โลโก้เฮชทีเอ็มแอล (HTML)	14
2.8	ทิม เบอร์เนิร์ส-ลี ผู้คิดค้น HTML	15
2.9	โลโก้ของพีเอชพี (PHP)	16
2.10	โลโก้ของพีเอชพีมายแอดมิน (phpMyAdmin)	18
2.11	ตัวอย่างการใช้งานพีเอพีมายแอดมิน (phpMyAdmin)	19
2.12	โลโก้ของมายเอสคิวแอล (MySQL)	19
2.13	คำสั่งติดต่อ MySQL	21
2.14	คำสั่งสร้างฐานข้อมูล	21
2.15	คำสั่งลบฐานข้อมูล	22
2.16	คำสั่งยกเลิกการเชื่อมต่อฐานข้อมูล	22
2.17	คำสั่งเลือกข้อมูลโดยเจาะจงค่าของ Field Name	22
2.18	คำสั่งเลือกข้อมูลโดยกำหนดค่าของ Field Name และเช็คค่าเทียบกับตัวแปร	22
2.19	คำสั่งดึงข้อมูลเฉพาะ Field ที่ต้องการ	22
2.20	คำสั่งดึงข้อมูลแบบ cell เดียว	22
2.21	คำสั่งใช้ mysql_fetch_array( )	23
2.22	คำสั่งใช้ mysql_fetch_assoc( )	23
3.1	บล็อกไดอะแกรมการทำงานของระบบ	24
3.2	แผนภาพการออกแบบหน้าเว็บไซต์เก็บค่าจังหวะการพิมพ์	25

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.3	แผนภาพการออกแบบการคำนวณหาค่าจังหวะการพิมพ์เพื่อเก็บลงฐานข้อมูล	26
3.4	แผนภาพการออกแบบการพิสูจน์ตัวตน	28
3.5	แผนภาพการทำงานของเว็บไซต์ล็อกอินต้นแบบ	29
4.1	ผลการทดสอบการเก็บค่าจังหวะการพิมพ์	32
4.2	ผลการทดสอบการคำนวณหาค่าเวลาในการกดคีย์บนแป้นพิมพ์	33
4.3	ผลการคำนวณหาค่าเวลาในแบบ Interkey Time ของข้อมูล 20 ชุด	33
4.4	ผลการคำนวณหาค่าเวลาในแบบ Latency Time ของข้อมูล 20 ชุด	34
4.5	ผลลำดับการพิมพ์คำว่า “ศรัทธา” ของผู้ใช้งาน 20 คน	35
4.6	ผลลำดับการพิมพ์คำว่า “ซ้ำเติม” ของผู้ใช้งาน 20 คน	36
4.7	ผลลำดับการพิมพ์คำว่า “ที่อยู่” ของผู้ใช้งาน 20 คน	37
4.8	ผลการทดสอบการคำนวณหาค่าเส้นเวลาจังหวะการพิมพ์ต้นแบบ	37
4.9	ผลการคำนวณหาค่า Master X	38
4.10	ผลการคำนวณหาค่า Master Y	38
4.11	ผลการทดสอบการคำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์	38
4.12	ผลการคำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์	39
4.13	กราฟความสัมพันธ์ของ Latency Time และ Interkey Time เพื่อสร้างเส้นเวลาจังหวะการพิมพ์ต้นแบบ	39
4.14	เปรียบเทียบเส้นเวลาจังหวะการพิมพ์ต้นฉบับกับเส้นเวลาจังหวะการพิมพ์ที่ต้องการตรวจสอบ	40
4.15	กราฟการหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์	40
4.16	ผลการทดสอบการพิสูจน์ตัวตน กรณีระบบอนุญาต	41
4.17	ผลการทดสอบการพิสูจน์ตัวตน กรณีระบบไม่อนุญาต	42
4.18	ผลการทดสอบการพิสูจน์ตัวตน กรณีจังหวะการพิมพ์ถูกแต่ลำดับในการพิมพ์ไม่ถูกต้อง	42
4.19	ทดสอบการเข้าสู่ระบบ	50

สารบัญรูป (ต่อ)

หน้า

รูปที่

4.20 ทดสอบการพิมพ์คำภาษาไทยเพื่อพิสูจน์จังหวะการพิมพ์ 51

4.21 จังหวะการพิมพ์ที่ถูกส่งมายังเครื่องเซิร์ฟเวอร์ 51

4.22 ระบบตรวจสอบว่าเป็นผู้ใช้งานจริง 52

4.23 ระบบตรวจสอบว่าไม่ใช่ผู้ใช้งานจริง 52

4.24 ฐานข้อมูลที่สร้างขึ้น 53

4.25 ตารางสำหรับบันทึกข้อมูลผู้ใช้งาน 54

4.26 ตารางสำหรับบันทึกคำภาษาไทยที่ผู้ใช้งานสร้างขึ้น 54

4.27 ตารางสำหรับบันทึกจังหวะการพิมพ์ 55

สารบัญตาราง

หน้า

ตารางที่

4.1 ประสิทธิภาพของคำง่ายในระบบพิสูจน์ตัวตน 43
4.2 ประสิทธิภาพของคำยาวในระบบพิสูจน์ตัวตน 43
4.3 ประสิทธิภาพของคำยากในระบบพิสูจน์ตัวตน 44
4.4 ผลรวมกรทดสอบประสิทธิภาพของทั้งระบบ 44
4.5 ประสิทธิภาพของระบบพิสูจน์ตัวตนของกลุ่มคำง่าย 46
4.6 ประสิทธิภาพของระบบพิสูจน์ตัวตนของกลุ่มคำยาว 47
4.7 ประสิทธิภาพของระบบพิสูจน์ตัวตนของกลุ่มคำยาก 48
4.8 ผลรวมประสิทธิภาพของระบบพิสูจน์ตัวตนทุกคำของทั้งระบบ 49

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันการเข้าใช้งานเว็บไซต์บนระบบอินเทอร์เน็ตผ่านระบบคอมพิวเตอร์มีจำนวนมากขึ้นและมีบทบาทสำคัญ ส่งผลให้การรักษาความปลอดภัยบนระบบคอมพิวเตอร์เพื่อรักษาสิทธิและข้อมูลของผู้ใช้งานมีความสำคัญมากขึ้น การพิสูจน์ตัวตนบนระบบคอมพิวเตอร์ด้วย Username และ Password เป็นวิธีการที่ใช้งานกันอย่างแพร่หลาย ซึ่งระบบการพิสูจน์ตัวตนดังกล่าวยังมีจุดอ่อนเนื่องจากผู้โจมตีสามารถทำการคาดเดา Password ได้ และ Username ไม่เป็นความลับ คาดเดาได้ง่าย ทำให้เสี่ยงต่อการถูกเจาะข้อมูลเข้าใช้งานโดยผู้อื่นได้

ปริญญานิพนธ์นี้จึงได้มีแนวคิดโดยใช้คำภาษาไทย และ Username ผนวกเข้ากับรหัสลับบางอย่างที่เป็นความลับต่อผู้โจมตี แต่ง่ายและไม่ต้องจดจำสำหรับผู้ใช้งานที่เป็นเจ้าของตัวจริง จึงนำมาซึ่งการใช้จังหวะการพิมพ์ (Keystroke Dynamics) ซึ่งเป็นมาตรวัดทางชีวะแบบพฤติกรรมมาทำการสร้างเป็นรหัสลับ

1.2 วัตถุประสงค์

1) เพื่อวิเคราะห์และศึกษาพฤติกรรมการพิมพ์ของผู้ใช้งาน
2) เพื่อจัดเก็บค่าจังหวะการพิมพ์ของแต่ละบุคคล ลงในระบบฐานข้อมูล
3) เพื่อสร้างระบบรักษาความปลอดภัยในการเข้าสู่ระบบผ่านระบบคอมพิวเตอร์

1.3 ขอบเขตของปริญญานิพนธ์

ระบบการพิสูจน์ตัวตนด้วยจังหวะการพิมพ์ผนวกกับการนำลำดับคำในภาษาไทยมาใช้สามารถจำแนกบุคคลในการเข้าสู่ระบบ จากการวิเคราะห์พฤติกรรมจังหวะการพิมพ์ได้ ด้วยการเปรียบเทียบค่าที่รับเข้ามากับฐานข้อมูล ซึ่งถ้าหากค่าที่รับเข้ามามีค่าอยู่ในช่วงการตัดสินใจของระบบ ระบบจะยืนยันว่าเป็นผู้ใช้ตัวจริง และยินยอมให้เข้าสู่ระบบได้

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 Biometric

ไบโอเมตริก (Biometric) เป็นเทคโนโลยีชีวภาพ ซึ่งเป็นการผสมผสานกันระหว่างเทคโนโลยีทางด้านชีวภาพและทางการแพทย์ กับเทคโนโลยีทางคอมพิวเตอร์เข้าด้วยกัน โดยการตรวจวัดคุณลักษณะทางกายภาพ (Physical Characteristics) และลักษณะทางพฤติกรรม (Behaviors) ที่เป็นลักษณะเฉพาะของแต่ละคนมาใช้ในการระบุตัวบุคคลนั้นๆ แล้วนำสิ่งเหล่านั้นมาเปรียบเทียบกับคุณลักษณะที่ได้มีการบันทึกไว้ในฐานข้อมูลก่อนหน้านี้ เพื่อใช้แยกแยะบุคคลนั้นจากบุคคลอื่นๆ ซึ่งคุณลักษณะทางกายภาพของคนเรานั้น ส่วนใหญ่จะไม่ค่อยเปลี่ยนแปลงไปตามกาลเวลา ในขณะที่พฤติกรรมของมนุษย์อาจมีการเปลี่ยนแปลงได้ไม่ว่าจะเป็นเสียงพูด การลงลายมือชื่อ การใช้แป้นพิมพ์ ซึ่งจัดเป็นคุณลักษณะทางพฤติกรรมของบุคคล ที่สามารถเปลี่ยนแปลงได้ตามกาลเวลาและการเรียนรู้ของบุคคลนั้นๆ [1]

2.1.1 ประเภทของ Biometric

ไบโอเมตริกซ์สามารถแบ่งออกเป็น 2 ประเภทใหญ่ๆ คือ การใช้ลักษณะทางกายภาพ (Physiological Biometrics) และการใช้ลักษณะทางพฤติกรรม (Behavioural Biometrics) ในการระบุตัวบุคคล [2]

2.1.1.1 ลักษณะทางกายภาพ (Physiological Biometrics)

1) ลายนิ้วมือ Fingerprint
2) ลักษณะใบหน้า Facial Recognition
3) ลักษณะของมือ Hand Geometry
4) ลักษณะของนิ้วมือ Finger Geometry
5) ลักษณะใบหู Ear Shape
6) ภายในดวงตา Iris Retina
7) กลิ่น Human Scen

2.1.1.2 ลักษณะทางพฤติกรรม (Behavioural Biometrics)

1) การพิมพ์ Keystroke Dynamics
2) การเดิน Gait Recognition
3) เสียง Voice Recognition
4) การเซ็นชื่อ Signature

2.1.2 กระบวนการในการตรวจสอบ

กระบวนการในการตรวจสอบ หรือระบุตัวบุคคลด้วยการใช้ไบโอเมตริกซ์ไม่ว่าจะเป็นการใช้ลักษณะเฉพาะแบบใดก็ตามจะมีขั้นตอนเหมือนกันดังต่อไปนี้

1) ผู้ใช้ระบบต้องทำการให้ตัวอย่าง ของลักษณะทางไบโอเมตริกซ์ที่ต้องการจะใช้หรือเป็นการลงทะเบียนเริ่มต้นก่อนที่จะทำการใช้ระบบ

2) ตัวอย่างทางไบโอเมตริกซ์ที่ถูกเก็บมาในขั้นตอนแรก จะถูกทำการแปลงและจัดเก็บให้เป็นแม่แบบ(Template)ที่จะใช้ในการเปรียบเทียบ

3) เมื่อผู้ใช้ต้องการที่จะใช้ระบบก็จะถูกตรวจสอบ หรือระบุผู้ใช้ โดยทำการเก็บตัวอย่างทางไบโอเมตริกซ์ของผู้ใช้ และทำการเปรียบเทียบกับแม่แบบ (Template) ที่เก็บไว้แล้วทำการตรวจสอบความเหมือนของตัวอย่างกับแม่แบบ จากนั้นก็จะทำการอนุญาต หรือปฏิเสธการเข้ามาใช้งานระบบของผู้ใช้

เราเรียกขั้นตอนที่ 1 และ 2 ว่าเป็นขั้นตอนของการลงทะเบียน (Enrolment) ซึ่งเป็นการทำเพียงครั้งเดียวก่อนการที่จะเริ่มใช้งาน ส่วนในขั้นตอนที่ 3 จะเป็นกระบวนการตรวจสอบ (Authentication) หรือ ระบุตัวผู้ใช้ (Identification) ซึ่งผลของการตรวจสอบหรือระบุตัวผู้ใช้นี้มีผลออกมาได้ 4 กรณีดังนี้

1) Correct Accept : ให้ผู้ใช้ที่มีสิทธิใช้ระบบเข้าใช้ระบบ

2) Correct Reject : ปฏิเสธผู้ที่ไม่มีสิทธิใช้ระบบ

3) False Accept : อนุญาตให้ผู้ที่ไม่มีสิทธิเข้าใช้ระบบ

จำนวนของ False Accept ถ้าคำนวณออกมาเป็นเปอร์เซ็นต์ จะเรียกว่า อัตราการอนุญาตผิดพลาด (False Accept Rate หรือ FAR)

4) False Reject : ปฏิเสธผู้ใช้ที่มีสิทธิใช้ระบบ ไม่ให้เข้าใช้ระบบ

จำนวนของ False Reject ถ้าคำนวณออกมาเป็นเปอร์เซ็นต์ จะเรียกว่า อัตราการปฏิเสธผิดพลาด (False Reject Rate หรือ FRR)

2.2 Keystroke Dynamics

Keystroke Dynamics เป็นหลักการที่ตั้งอยู่บนพื้นฐานของเวลาในการกดคีย์บนแป้นพิมพ์ (Keyboard) ซึ่งตั้งสมมติฐานว่าจังหวะการพิมพ์ของแต่ละคนจะมีลักษณะเฉพาะตัวและสามารถนำมาใช้ ในระบบการพิสูจน์ตัวตนได้ Feature พื้นฐานของ Keystroke Dynamics มีดังนี้ [3]

1) Key Hold Time

เวลาที่กดคีย์ค้างไว้ โดยจับเวลาตั้งแต่เริ่มกดคีย์จนกระทั่งปล่อยคีย์

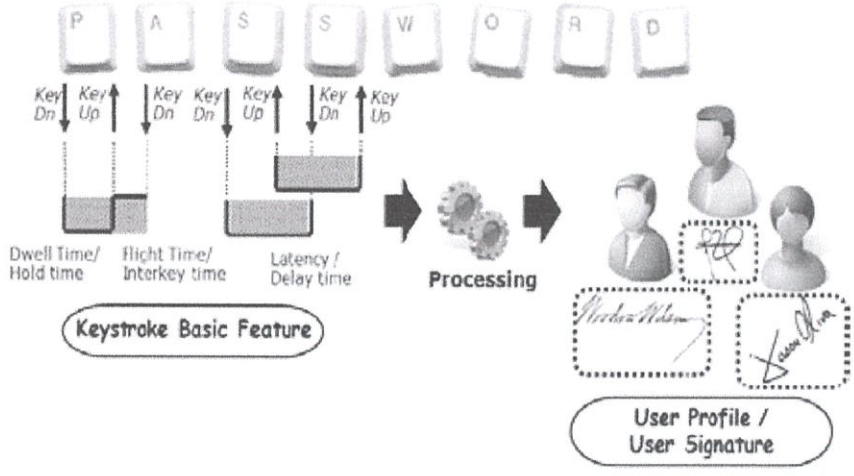
2) Interkey Time

เวลาที่เปลี่ยนคีย์ใด ไปสู่อีกคีย์หนึ่งซึ่งอาจได้เป็นตัวเลขค่าบวกหรือลบ กรณีที่ได้ค่าบวกคือมีการปล่อยคีย์ก่อนหน้าก่อนที่จะกดคีย์ถัดไป หากได้ค่าลบ คือ มีการกดคีย์ถัดไปก่อนที่จะปล่อยคีย์ก่อนหน้า หรือมีการกดคีย์ซ้อนกันในช่วงเวลาที่เปลี่ยนคีย์

3) Latency Time

เวลาที่เริ่มกดคีย์จนกระทั่ง กดคีย์ถัดไปหรืออีกนัยหนึ่งคือเวลาที่เริ่มจากปล่อยคีย์จนกระทั่งปล่อยคีย์ถัดไป

ซึ่งสามารถอธิบายให้เข้าใจ ได้ง่ายขึ้นดังรูปที่ 2.1



รูปที่ 2.1 Feature พื้นฐานของ Keystroke Dynamics [3]

2.3 การพิสูจน์ตัวตนผ่านลำดับการกดแป้นพิมพ์ของคำภาษาไทย

ภาษาไทยมีพยัญชนะ 44 รูป สระ 21 รูป และวรรณยุกต์อีก 4 รูป โดยในการพิมพ์บนแป้นคีย์บอร์ดบนระบบคอมพิวเตอร์ต้องมีการกด Shift เพื่อพิมพ์คำหรือสระที่ต้องการ ซึ่งในการกด Shift จะแบ่งเป็น Shift Left และ Shift Right โดยในคำหนึ่งคำ แต่ละบุคคลจะมีลำดับในการกดแป้นพิมพ์ไม่เหมือนกัน แต่ในบุคคลเดิมจะมีลำดับในการกดเหมือนเดิมตลอด เช่น การกดพิมพ์คำว่า “ซ้ำเติม” ประกอบด้วยพยัญชนะ คือ ซ.โซ่ ต.เต่า ม.ม้า สระคือ สระอำ สระเอ สระอิ และวรรณยุกต์คือ ไม้โท สามารถมีลำดับในการกดได้หลายแบบ เช่น Shift (Left/Right) + ซ.โซ่ + ไม้โท + สระอำ + สระเอ + ต.เต่า + สระอิ + ม.ม้า โดยในการพิมพ์คำว่า “ซ้ำเติม” เราสามารถพิมพ์สระอำก่อนไม้โทได้ จะมีลำดับการพิมพ์เป็น Shift (Left/Right) + ซ.โซ่ + สระอำ + ไม้โท + สระเอ + ต.เต่า + สระอิ + ม.ม้า จะเห็นว่าลำดับการกดแป้นพิมพ์เพื่อพิมพ์คำว่า “ซ้ำเติม” มีลำดับที่แตกต่างกัน ดังรูปที่ 2.2 หรือการกดพิมพ์คำว่า “ที่อยู่” ประกอบด้วยพยัญชนะ คือ ท.ทหาร อ.

อ่าง ย.ยักษ์ และสระคือ สระอี สระอู และวรรณยุกต์ คือ ไม้เอก สามารถมีลำดับการพิมพ์ได้หลายแบบ เช่น ท.ทหาร + สระอี + ไม้เอก + อ.อ่าง + ย.ยักษ์ + Shift (Left/Right) + สระอู + ไม้เอก โดยในการพิมพ์คำว่า “ที่อยู่” เราสามารถพิมพ์ไม้เอกก่อน สระอูได้ จะมีลำดับในการพิมพ์เป็น ท.ทหาร + สระอี + ไม้เอก + อ.อ่าง + ย.ยักษ์ + Shift (Left/Right) + ไม้โท + สระอู จะเห็นว่าลำดับการกดแป้นพิมพ์เพื่อพิมพ์คำว่า “ที่อยู่” มีลำดับที่แตกต่างกัน ดังรูปที่ 2.3

key.txt

```
ShiftLeft Semicolon KeyE KeyH KeyG Digit9 KeyB Comma
ShiftLeft Semicolon KeyE KeyH KeyG Digit9 KeyB Comma
ShiftLeft Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftLeft Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftLeft Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftLeft Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftLeft Semicolon KeyE KeyH KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyE KeyH KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftLeft Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftRight Semicolon KeyH KeyE KeyG Digit9 KeyB Comma
ShiftLeft Semicolon KeyE KeyH KeyG Digit9 KeyB Comma
```

รูปที่ 2.2 ลำดับการพิมพ์คำว่า “ซ้ำเติม” จากผู้ทดลอง 20 คน

key.txt

```
KeyM KeyU KeyJ KeyV KeyP ShiftLeft Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftLeft Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftRight Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftRight Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftLeft Digit6 KeyJ
KeyM KeyJ KeyU KeyV KeyP ShiftLeft Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftLeft Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftLeft Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftRight Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftLeft Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftRight Digit6 KeyJ
KeyM KeyJ KeyU KeyV KeyP ShiftRight Digit6 KeyJ
KeyM KeyJ KeyU KeyV KeyP ShiftRight Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftRight Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftRight Digit6 KeyJ
KeyM KeyJ KeyU KeyV KeyP ShiftLeft Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftRight Digit6 KeyJ
KeyM KeyU KeyJ KeyV KeyP ShiftLeft Digit6 KeyJ
KeyM KeyJ KeyU KeyV KeyP ShiftLeft Digit6 KeyJ
KeyM KeyJ KeyU KeyV KeyP ShiftRight Digit6 KeyJ
```

รูปที่ 2.3 ลำดับการพิมพ์คำว่า “ที่อยู่” จากผู้ทดลอง 20 คน

จากรูปที่ 2.2 และ 2.3 จะเห็นได้ว่าลำดับในการพิมพ์คำภาษาไทยของแต่ละบุคคลเมื่อพิมพ์คำเดียวกัน จะมีรูปแบบหรือลำดับในการพิมพ์ที่แตกต่างกันออกไป ซึ่งลำดับในการพิมพ์ของคำภาษาไทยนั้นสามารถนำมาช่วยในการพัฒนาการพิสูจน์ตัวตนผ่านจังหวะการพิมพ์ได้

2.4 ทฤษฎีที่ใช้ในการคำนวณ

2.4.1 ค่าเฉลี่ยเลขคณิต (Arithmetic Mean) [4]

ค่าเฉลี่ยเลขคณิต หมายถึง การหารผลรวมของข้อมูลทั้งหมดด้วยจำนวนข้อมูลทั้งหมด การหาค่าเฉลี่ยเลขคณิต สามารถหาได้ 2 วิธี

1) ค่าเฉลี่ยเลขคณิตของข้อมูลที่ไม่ได้แจกแจงความถี่ สามารถคำนวณได้จากสูตร

$$\bar{x} = \frac{\sum x}{n} \tag{2.1}$$

เมื่อ $\bar{x}$ คือ ค่าเฉลี่ยเลขคณิต

$\sum x$ คือ ผลบวกของข้อมูลทุกค่า

n คือ จำนวนข้อมูลทั้งหมด

2) ค่าเฉลี่ยเลขคณิตของข้อมูลที่แจกแจงความถี่ สามารถคำนวณได้จากสูตร

$$\bar{x} = \frac{\sum fx}{n} \tag{2.2}$$

เมื่อ $\bar{x}$ คือ ค่าเฉลี่ยเลขคณิต

f คือ ความถี่ของข้อมูล

x คือ ค่าของข้อมูล (ในกรณีการแจกแจงความถี่ไม่เป็นอันตรภาคชั้น) หรือจุดกึ่งกลางของอันตรภาคชั้น (ในกรณีการแจกแจงความถี่เป็นอันตรภาคชั้น)

หาได้จาก $\frac{\text{ค่าสูงสุดของอันตรภาคชั้น} + \text{ค่าต่ำสุดของอันตรภาคชั้น}}{2}$

n คือ ผลรวมความถี่ทั้งหมด หรือ จำนวนข้อมูลทั้งหมด

2.4.2 ส่วนเบี่ยงเบนมาตรฐาน (Standard Deviation : S.D.,S,s) [5]

ส่วนเบี่ยงเบนมาตรฐานเป็นค่าวัดการกระจายที่สำคัญทางสถิติ เพราะเป็นค่าที่ใช้บอกถึงการกระจายของข้อมูลได้ดีกว่าค่าพิสัย และค่าส่วนเบี่ยงเบนเฉลี่ย การหาส่วนเบี่ยงเบนมาตรฐานสามารถหาได้ 2 วิธี

1) การหาส่วนเบี่ยงเบนมาตรฐาน(S.D.) ในกรณีข้อมูลไม่ได้มีการแจกแจงความถี่สามารถหาได้จากสูตร

$$S.D. = \sqrt{\frac{(x-\bar{x})^2}{n-1}} \tag{2.3}$$

$$S.D. = \sqrt{\frac{n\sum x^2-(\sum x)^2}{n(n-1)}} \quad (2.4)$$

เมื่อ $S.D.$ คือ ส่วนเบี่ยงเบนมาตรฐาน

x คือ ข้อมูล (ตัวที่1,2,3..,n)

$\bar{x}$ คือ ค่าเฉลี่ยเลขคณิต

n คือ จำนวนข้อมูลทั้งหมด

2) การหาส่วนเบี่ยงเบนมาตรฐาน (S.D.) ในกรณีข้อมูลมีการแจกแจงความถี่ สามารถหาได้จากสูตร

$$S.D. = \sqrt{\frac{\sum f(x-\bar{x})^2}{n-1}} \quad (2.5)$$

$$S.D. = \sqrt{\frac{n\sum fx^2-(\bar{x})^2}{n(n-1)}} \quad (2.6)$$

เมื่อ $S.D.$ คือ ส่วนเบี่ยงเบนมาตรฐาน

f คือ ความถี่

x คือ จุดกึ่งกลางชั้น

$\bar{x}$ คือ ค่าเฉลี่ยเลขคณิต

n คือ จำนวนข้อมูล

2.4.3 ระยะทางแบบยูคลิด (Euclidean distance)

คือระยะทางปกติระหว่างจุดสองจุดในแนวเส้นตรง ซึ่งอาจสามารถวัดได้ด้วยไม้บรรทัด มีที่มาจากทฤษฎีบทพีทาโกรัส เหตุที่เรียกว่า แบบยุคลิด เนื่องจากเป็นการวัดระยะทางในปริภูมิแบบยุคลิด (หรือแม้แต่ปริภูมิผลคูณภายใน) คือไม่มีความโค้งและไม่สามารถทำให้โค้งงอ และการใช้สูตรนี้วัดระยะทางทำให้กลายเป็นปริภูมิอิงระยะทาง ค่าประจำ (norm) ที่เกี่ยวข้องก็จะเรียกว่าเป็น ค่าประจำแบบยุคลิด (Euclidean norm) เช่นกัน (งานเขียนสมัยก่อนเรียกการวัดอย่างนี้ว่า ระยะทางแบบพีทาโกรัส) [6]

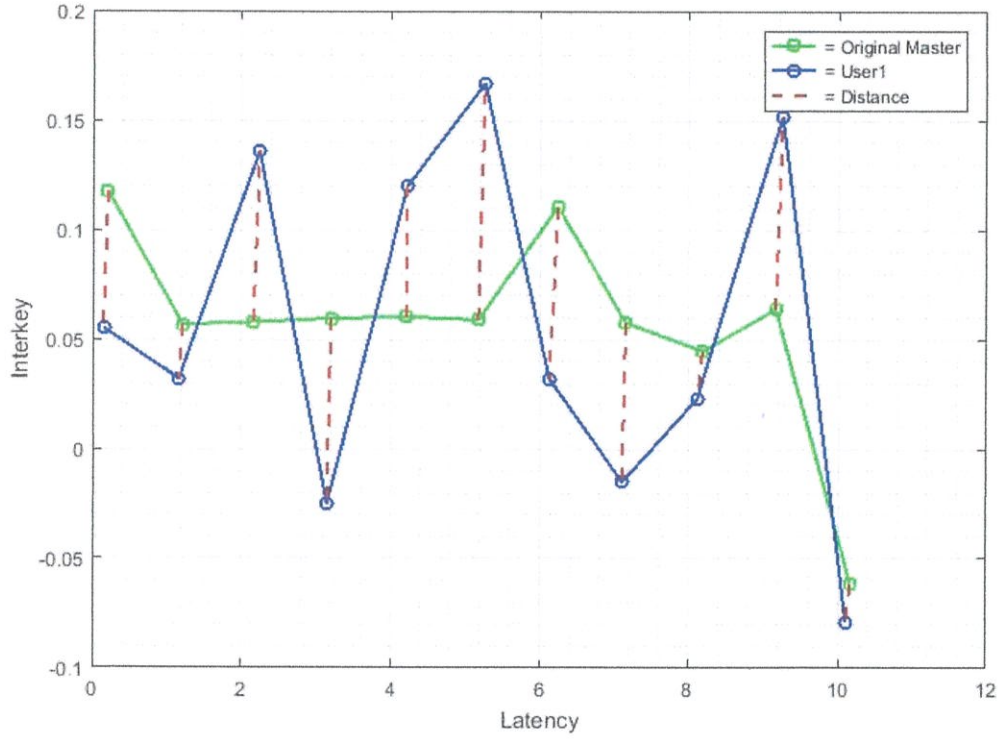
ระยะทางแบบยุคลิดระหว่างจุดสองจุด p และ q คือความยาวของส่วนของเส้นตรง $\overline{pq}$ ถ้า $p = (p_1, p_2, \dots, p_n)$ และ $q = (q_1, q_2, \dots, q_n)$ ในระบบพิกัดคาร์ทีเซียน เป็นจุดสองจุดบนปริภูมิยุคลิด n มิติ ระยะทางระหว่างจุด p กับ q คำนวณได้จาก

$$d(p,q) = \sqrt{(p_1-q_1)^2 + (p_2-q_2)^2 + \cdots + (p_n-q_n)^2} \quad (2.7)$$

ในสองมิติแบบยุคลิด ถ้า $p = (p_1, p_2)$ และ $q = (q_1, q_2)$ แล้ว ระยะทางระหว่าง p และ q สามารถคำนวณได้ดังนี้ ซึ่งมีสูตรเหมือนกับทฤษฎีบทพีทาโกรัส

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2} \qquad (2.8)$$

การประยุกต์ใช้การวัดระยะทางแบบยูคลิดเพื่อหาความแตกต่างของเส้นเวลาจังหวะการพิมพ์ โดยวัดผลรวมระยะทางจุดต่อจุด ระหว่างเส้นเวลาจังหวะการพิมพ์ของผู้ใช้งานตัวจริง (Real User) กับ ผู้บุกรุก (Imposter User) โดยเปรียบเทียบค่ากับเส้นต้นฉบับได้ทำการเก็บบันทึกค่าไว้ หากเป็นเส้นจังหวะเวลาการพิมพ์ที่มาจากผู้ใช้งานตัวจริง (Real User) จะมีผลรวมของระยะทางที่ต่ำกว่าเส้นจังหวะเวลาการพิมพ์ที่มาจากผู้บุกรุก (Imposter User) ดังรูปที่ 2.4



รูปที่ 2.4 วัดความแตกต่างของเส้นเวลาจังหวะการพิมพ์ด้วยวิธีวัดระยะทางแบบยูคลิด

2.4.4 การวัดสมรรถภาพของเทคโนโลยีไบโอเมตริก (Biometric Performance)

การวัดสมรรถภาพของเทคโนโลยีไบโอเมตริก เป็นการวัดความแม่นยำ (Accuracy) ทางเทคนิค ซึ่งเป็นสิ่งแรกในการพิจารณา เพื่อตัดสินใจเลือกรูปแบบของเทคโนโลยีไบโอเมตริกเข้ามาใช้ในหน่วยงาน โดยหลักๆแล้ว ความแม่นยำวัดได้จากร้อยละของความผิดพลาดที่เกิดขึ้นขณะทำการตรวจสอบ ดังต่อไปนี้ [7]

1) อัตราการปฏิเสธที่ผิดพลาด (False Rejection Rate : FRR)

เป็นอัตราการปฏิเสธตัวจริง ค่า FRR (บางงานวิจัย ใช้ False Non - MatchRate : FNMR) จะบอกถึงร้อยละของความผิดพลาดที่เกิดขึ้นเมื่อตัวจริงถูกปฏิเสธ หากค่า FRR ของระบบต่ำ แสดงว่าไม่ว่าตัวจริงหรือตัวปลอมก็มีโอกาสได้รับอนุญาตให้ผ่านเข้าสู่ระบบได้ง่าย ฉะนั้นหากระบบต้องการความปลอดภัยค่อนข้างสูง ก็ควรจะปรับแต่งให้ FRR มีค่าสูงขึ้น

2) อัตราการยอมรับที่ผิดพลาด (False Acceptance Rate : FAR)

FAR (บางงานวิจัย ใช้ False Match Rate : FMR) เป็นร้อยละของความผิดพลาดที่เกิดขึ้น จากการที่ตัวปลอมได้รับการยอมรับค่า FAR ที่ต่ำ หมายถึง ไม่ว่าตัวจริงหรือตัวปลอมก็มีโอกาสถูกปฏิเสธสูง อย่างไรก็ตาม การปรับแต่งให้ FAR มีค่าต่ำจะช่วยป้องกันความเสียหายอันเนื่องมาจากการยอมรับตัวปลอมให้ผ่านเข้าสู่ระบบได้โดยง่าย

2.4.5 ความไวและความจำเพาะ

ความไวและความจำเพาะเป็นค่าวัดทางสถิติที่ใช้ประเมินประสิทธิภาพของการทดสอบ ที่ให้ผลเป็นสองส่วน (เช่นเป็นบวกและลบ) [8]

1) ความไว (Sensitivity)

ความไวคือสัดส่วนของผลบวกที่เป็นจริงสาหรับภาวะนั้น ๆ (เช่น สัดส่วนของการตรวจพบโรคในผู้ที่ป่วยจริง) มีไวพจน์เป็นคาอื่น ๆ รวมทั้ง อัตราผลบวกจริง (true positive rate) สามารถคำนวณได้ตามสมการที่ (2.7)

2) ความจำเพาะ (Specificity)

ความจำเพาะคือสัดส่วนของผลลบที่เป็นจริงสำหรับภาวะนั้น ๆ (เช่น สัดส่วนของการตรวจไม่พบโรคในผู้ที่ไม่ป่วย) มีไวพจน์เป็นคำอื่น ๆ รวมทั้ง อัตราผลลบจริง (true negative rate) สามารถคำนวณได้ตามสมการที่ (2.8)

3) ค่าทำนายผลบวกหรือค่าพยากรณ์ผลบวก(Positive predictive value)

ค่าทำนายผลบวกหรือค่าพยากรณ์ผลบวก คือสัดส่วนของจำนวนผลการตรวจที่เป็นผลบวกแท้ ("การวินิจฉัยถูกต้อง") ต่อจำนวนผลการตรวจที่เป็นผลบวกทั้งหมด เป็นค่าที่มีความสำคัญมากในการประเมินประสิทธิภาพของวิธีหรือเครื่องมือในการวินิจฉัยโรค เนื่องจากเป็นค่าที่บ่งบอกว่าผลบวกที่ได้จากการตรวจนั้นแสดงว่าเป็นโรคจริงๆ ได้ดีเพียงใด ค่าทำนายผลบวกหรือค่าพยากรณ์ผลบวกสามารถคำนวณได้ตามสมการที่ (2.9)

4) ค่าทำนายผลลบหรือค่าพยากรณ์ผลลบ (Negative predictive value)

ค่าทำนายผลลบหรือค่าพยากรณ์ผลลบ คือสัดส่วนของจำนวนผลการตรวจที่เป็นผลลวง ต่อจำนวนผลการตรวจที่เป็นผลลบทั้งหมด เป็นค่าที่บ่งบอกว่าผลลบที่ได้จากการตรวจนั้นแสดงว่าไม่ได้เป็นโรคจริงๆ ได้ดีเพียงใด ดังนั้นความไวจึงมีประโยชน์ในการวินิจฉัยแยกกับผลลบลวง (false negative) เพราะว่าการทดสอบยิ่งไวเท่าไร โอกาสการได้ผลลบ (เช่น การพบว่าไม่มีโรค) ที่ไม่เป็นจริง (เช่น บุคคลจริง ๆ มีโรค) ก็น้อยลงเท่านั้น สามารถคำนวณได้ตามสมการที่ (2.10) ดังนั้น ถ้าความไวอยู่ที่ 100% โอกาสได้ผลลบลวงก็อยู่ที่ 0% และความจำเพาะจึงมีประโยชน์ในการยืนยันภาวะที่มี โดยกันผลบวกลวง (false positive) เพราะว่าการทดสอบยิ่งจำเพาะเท่าไร โอกาสการได้ผลบวก (เช่น การพบว่ามีโรค) ที่ไม่เป็นจริง (เช่น บุคคลจริง ๆ ไม่มีโรค) ก็น้อยลงเท่านั้น และดังนั้น ถ้าความจำเพาะอยู่ที่ 100% โอกาสได้ผลบวกลวงก็อยู่ที่ 0%

$$\text{Sensitivity} = TP / (TP + FN) \quad (2.7)$$

$$\text{Specificity} = TN / (FP + TN) \quad (2.8)$$

$$\text{Positive predictive} = TP / (TP + FP) \quad (2.9)$$

$$\text{Negative} = TN / (FN + TN) \quad (2.10)$$

โดยที่

TP (True Positive)	คือ ผลบวกจริง (ผู้ใช้งานจริงเข้าได้)
FP (False Positive)	คือ ผลบวกลวง (ผู้บุรุกเข้าได้)
TN (True Negative)	คือ ผลลบจริง (ผู้ใช้งานจริงเข้าไม่ได้)
FN (False Negative)	คือ ผลลบลวง (ผู้บุกรุกเข้าไม่ได้)

2.5 ภาษาจาวาสคริปต์ (JavaScript)

รูปที่ 2.5 โลโก้ของจาวาสคริปต์ (JavaScript)

JavaScript คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต เป็นภาษาสคริปต์เชิงวัตถุ ที่เรียกกันว่า "สคริปต์" (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปทีละคำสั่ง" (interpret) หรือเรียกว่า อ็อบเจ็กโอเรียลเต็ด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และทางฝั่งเซิร์ฟเวอร์ (Server) [9]

JavaScript ถูกพัฒนาขึ้นโดย เน็ตสเคปคอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่า Live Script ออกมาพร้อมกับ Netscape Navigator2.0 เพื่อใช้สร้างเว็บเพจโดยติดต่อกับเซิร์ฟเวอร์แบบ Live Wire ต่อมาเน็ตสเคปจึงได้ร่วมมือกับ บริษัทซันไมโครซิสเต็มส์ ปรับปรุงระบบของบราวเซอร์เพื่อให้สามารถติดต่อใช้งานกับภาษาจาวาได้ และได้ปรับปรุง LiveScript ใหม่เมื่อ ปี 2538 แล้วตั้งชื่อใหม่ว่า JavaScript โดย JavaScript สามารถทำให้ การสร้างเว็บเพจ มีลูกเล่น ต่าง ๆ มากมาย และยังสามารถโต้ตอบกับผู้ใช้ได้อย่างทันที เช่น การใช้เมาส์คลิก หรือ การกรอกข้อความในฟอร์ม เป็นต้น

เนื่องจาก JavaScript ช่วยให้ผู้พัฒนา สามารถสร้างเว็บเพจได้ตรงกับความต้องการและมีความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิด ที่ใครก็สามารถนำไปใช้ได้ ดังนั้นจึงได้รับความนิยมเป็นอย่างสูง มีการใช้งานอย่างกว้างขวาง รวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดย ECMA การทำงานของ JavaScript จะต้องมีการแปลความคำสั่ง ซึ่งขั้นตอนนี้จะถูกจัดการโดยบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้น JavaScript จึงสามารถทำงานได้ เฉพาะบนบราวเซอร์ที่สนับสนุน ซึ่งปัจจุบันบราวเซอร์เกือบทั้งหมดก็สนับสนุน JavaScript แล้ว อย่างไรก็ดี สิ่งที่ต้องระวังคือ JavaScript มีการพัฒนาเป็นเวอร์ชั่นใหม่ๆออกมาด้วย ดังนั้น ถ้านำโค้ดของเวอร์ชั่นใหม่ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุน ก็อาจจะทำให้เกิด error ได้

2.5.1 ประโยชน์ของจาวาสคริปต์ (JavaScript)

1) JavaScript ทำให้สามารถใช้เขียนโปรแกรมแบบง่ายๆได้ โดยไม่ต้องพึ่งภาษาอื่น

2) JavaScript มีคำสั่งที่ตอบสนองกับผู้ใช้งาน เช่นเมื่อผู้ใช้คลิกที่ปุ่ม หรือ Checkbox ก็สามารถสั่งให้เปิดหน้าใหม่ได้ ทำให้เว็บไซต์ของเรามีปฏิสัมพันธ์กับผู้ใช้งานมากขึ้น นี่คือข้อดีของ JavaScript เลยก็ว่าได้ที่ทำให้เว็บไซต์ดังๆทั้งหลายเช่น Google Map ต่างหันมาใช้

3) JavaScript สามารถเขียนหรือเปลี่ยนแปลง HTML Element ได้ นั่นคือสามารถเปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้ หรือหน้าแสดงเนื้อหาสามารถซ่อนหรือแสดงเนื้อหาได้แบบง่ายๆนั่นเอง

4) JavaScript สามารถใช้ตรวจสอบข้อมูลได้ สังเกตว่าเมื่อเรากรอกข้อมูลบางเว็บไซต์ เช่น Email เมื่อเรากรอกข้อมูลผิดจะมีหน้าต่างฟ้องขึ้นมาว่าเรากรอกผิด หรือลืมกรอกอะไรบางอย่าง เป็นต้น

5) JavaScript สามารถใช้ในการตรวจสอบผู้ใช้ได้เช่น ตรวจสอบว่าผู้ใช้ใช้ web browser อะไร

6) JavaScript สร้าง Cookies (เก็บข้อมูลของผู้ใช้ในคอมพิวเตอร์ของผู้ใช้เอง) ได้

2.5.2 ข้อดี และข้อเสียของจาวาสคริปต์ (JavaScript)

การทำงานของ JavaScript เกิดขึ้นบนบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้นไม่ว่าจะใช้เซิร์ฟเวอร์อะไร หรือที่ไหน ก็ยังคงสามารถใช้ JavaScript ในเว็บเพจได้ ต่างกับภาษาสคริปต์อื่น เช่น Perl, PHP หรือ ASP ซึ่งต้องแปลความและทำงานที่ตัวเครื่องเซิร์ฟเวอร์ (เรียกว่า server-side script) ดังนั้นจึงต้องใช้บนเซิร์ฟเวอร์ ที่สนับสนุนภาษาเหล่านี้เท่านั้น อย่างไรก็ดี จากลักษณะดังกล่าวก็ทำให้ JavaScript มีข้อจำกัด คือไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ชม เพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานลักษณะนี้ จึงยังคงต้องอาศัยภาษา server-side script อยู่ (ความจริง JavaScript ที่ทำงานบนเซิร์ฟเวอร์ก็มี ซึ่งต้องอาศัยเซิร์ฟเวอร์ที่สนับสนุนโดยเฉพาะเช่นกัน แต่ไม่เป็นที่นิยมนัก)

2.6 ภาษาไพธอน (Python)

รูปที่ 2.6 โลโก้ของไพธอน (Python)

2.6.1 ประวัติของไพธอน

ผู้ที่คิดค้นภาษาไพธอนขึ้นมาคือกุยโด ฟาน รอสซึม (Guido van Rossum) ชาวฮอลันดา แม้ว่าไพธอนจะหมายถึงงู แต่เดิมทีผู้คิดค้นนั้นได้ชื่อนี้มาจากชื่อของซีรีส์รายการตลกเรื่อง Monty Python's Flying Circus ของอังกฤษ ซึ่งฉายตั้งแต่ปี 1969 ถึง 1974

ผู้คิดค้นนั้นทำงานอยู่กับสถาบันวิจัยคณิตศาสตร์และวิทยาศาสตร์คอมพิวเตอร์แห่งชาติฮอลันดา เดิมทีร่วมพัฒนาภาษา ABC ซึ่งถูกคิดขึ้นและใช้มาก่อนหน้า แต่ว่าตอนหลังเปลี่ยนมาพัฒนาภาษาขึ้นใหม่เป็นของตัวเอง ดังนั้นภาษาไพธอนจึงได้รับอิทธิพลจากภาษา ABC มามาก ภาษาไพธอนเริ่มกำเนิดขึ้นในปี 1989 จากนั้นในปี 1991 ก็ถูกปล่อยออกมาเป็นโอเพนซอร์สเป็นครั้งแรก [10]

2.6.2 ที่มาของชื่อ ไพธอน

คำว่าไพธอน (python) เป็นชื่องูสกุลหนึ่ง ซึ่งในภาษาไทยเรียกว่า "งูเหลือม" หรือ "งูหลาม" เป็นงูไม่มีพิษ มีชื่อทางวิทยาศาสตร์ว่าสกุล Pythonidae รากศัพท์เดิมมาจากภาษากรีกคำว่า **πύθων** อ่านแบบกรีกโบราณว่า "ปือทอน" อ่านแบบกรีกสมัยใหม่ว่า "พีโธน" แต่พอมาใช้ในภาษาอังกฤษก็แผลงเป็น "ไพธอน"

"ปือทอน" เป็นชื่อของงูยักษ์รูปร่างคล้ายมังกร ซึ่งปรากฏตัวในเทพปกรณัมกรีก แต่ตอนหลังถูกนำมาใช้เป็นชื่อเรียกงูที่มีอยู่จริง

อนึ่ง อักษร "ธ" ในการเขียนทับศัพท์คำว่า "ไพธอน" ในที่นี้ไม่ได้แทนเสียง "ท" แต่แทนเสียง th ในภาษาอังกฤษ ซึ่งไม่เหมือน th ในภาษาไทย แต่เป็นเสียงที่ไม่มีอยู่ในภาษาไทย เสียงนี้จริงๆแล้วใกล้เคียง "ซ" มากกว่า "ท" เสียอีก

ในภาษากรีกใช้อักษร **θ** "เธตา" อย่างไรก็ตามกรีกโบราณไม่มีเสียงนี้ แต่ออกเสียง **θ** เป็นเสียง "ท" แทน [10]

2.6.3 จุดเด่นของภาษาไพธอน

1) สามารถใช้ได้ทุกแพลตฟอร์ม

สามารถทำงานได้ทุก ๆ CPU หลาย ๆ ระบบปฏิบัติการ เพียงแต่ผู้เขียนโปรแกรมเขียนจากแพลตฟอร์มใด ๆ แล้วนำโปรแกรมที่ได้ไปให้ทำงานต่างแพลตฟอร์มกันได้

2) ไม่ต้องเสียค่าใช้จ่ายในการจัดซื้อโปรแกรมต้นฉบับ

โดยปกติแล้วโปรแกรมภาษาทั่ว ๆ ไปจะต้องจัดซื้อโปรแกรมต้นฉบับเพื่อนำมาติดตั้งในราคาที่แพงมาก แต่โปแกรมภาษาไพธอน สามารถดาวน์โหลดจาก www.python.org ได้โดยตรง แล้วนำมาติดตั้งและศึกษาการใช้ด้วยตนเอง เพราะเป็นโปรแกรมประเภทโอเพนซอร์ส (Open Source)

3) ภาษาไพธอนได้นำเอาข้อดีของโปรแกรมในอดีตเข้ามาไว้ด้วยกัน เช่น ภาษา C, C++, Java และ Perl เป็นต้น

4) มีความปลอดภัยสูง

เนื่องจากภาษาไพธอนทำงานอยู่ด้าน Server เป็นหลัก เมื่อมีการร้องขอจากเครื่อง Client จะประมวลผลที่เครื่อง Server ทำให้ผู้ใช้ทั่วไปไม่สามารถเข้าถึงเครื่อง Server ได้โดยตรงจึงมีความปลอดภัยสูงกว่า

5) ใช้ในการพัฒนา Web Service

ในปัจจุบันการพัฒนาซอฟต์แวร์ได้เน้นที่มีการแลกเปลี่ยนข้อมูลซึ่งกันและกันทั้งในองค์กรเดียวกันหรือแม้แต่ต่างองค์กรกัน ทำให้เกิดความ สะดวกสบาย ไม่ต้องใช้ซอฟต์แวร์อื่น ๆ มาแปลงข้อมูลเพื่อให้เข้ากันได้อีกต่อไปเรียนรู้ได้เร็วกว่าโปรแกรมภาษาอื่น ๆ เพราะมีโครงสร้างภาษาที่ไม่ซับซ้อน ซึ่งโครงสร้างภาษาคล้ายคลึงกับภาษา C นอกจากนี้การเขียนโปรแกรมด้วยภาษาไพธอนจะมีความกระชับและสั้นกว่าภาษาซี [11]

2.7 ภาษาเฮชทีเอ็มแอล (HTML)

รูปที่ 2.7 โลโก้เฮชทีเอ็มแอล (HTML)

2.7.1 ความหมายของเฮชทีเอ็มแอล

HTML ย่อมาจาก HyperText Markup Language เป็น ภาษาคอมพิวเตอร์ที่ใช้สร้างหน้าเว็บ (Web Page) ในรูปแบบของ ไฟล์ HTML (คือไฟล์ที่มีนามสกุลเป็น .htm หรือ .html) ซึ่งมีเว็บเบราว์เซอร์ (Web Browser) เป็นโปรแกรมที่ใช้แปลงไฟล์ HTML เพื่อ แสดงผลในรูปของหน้าเว็บ

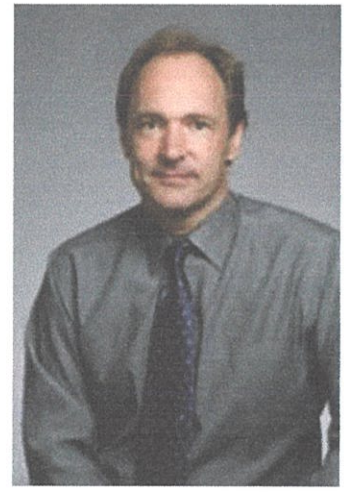
ไฟล์ HTML เป็นไฟล์รหัสแอสกี (ASCII) ถูกบันทึกในรูปของ ไฟล์เอกสาร (Text File) ที่สามารถถูกสร้างจากโปรแกรมสร้างไฟล์ ข้อความ (Text Editor) เช่น Notepad หรือ Word Processing ทั่วๆ ไป ซึ่งลักษณะของไฟล์ HTML ประกอบไปด้วยแท็ก (Tag) ต่างๆ ที่เป็นคำสั่งของ HTML ซึ่งแท็กจะอยู่ภายในเครื่องหมาย < และ >

แท็กใน HTML แบ่งเป็น 2 ประเภท คือ คอนเทนเนอร์แท็ก (Container Tag) และแท็กเปล่า (Empty Tag) โดยที่คอนเทนเนอ ร์แท็ก ประกอบไปด้วยแท็กเปิด และแท็กปิด โดยที่แท็กปิดจะมี เครื่องหมาย / นำหน้าแท็ก เช่น <H1> . . . </H1> ส่วนแท็กเปล่า จะมีแท็กเปิดอย่างเดียว เช่น <HR> ซึ่งแท็กจะถูกเขียนด้วยตัวอักษร พิมพ์ใหญ่ หรือพิมพ์เล็กก็ได้ จะไม่มีผลต่อการ

แสดงผลของเว็บเบราว์เซอร์ เช่น <BR>, <br>,
 หรือ
 เว็บเบราว์เซอร์จะ แปลความหมายเหมือนกัน

โครงสร้างไฟล์ HTML แบ่งออกเป็น 2 ส่วน คือ ส่วนหัว เรื่อง (Head Section) และส่วนเนื้อหา (Body Section) โดยจะมีแท็ก <HTML> และ </HTML> เป็นตัวกำหนดขอบเขตไฟล์ ซึ่ง ส่วนหัวเรื่อง มีไว้กำหนดข้อมูลเฉพาะของหน้าเว็บ เช่น ชื่อเรื่อง ของเว็บภายในแท็ก <HEAD> และ </HEAD> และสำหรับส่วน เนื้อหามีไว้กำหนดรายละเอียดต่างๆ ที่ต้องการแสดงบนหน้าเว็บ เช่น ข้อความ และรูปภาพภายในแท็ก <BODY> และ </BODY>

2.7.2 ประวัติเฮชทีเอ็มแอล



รูปที่ 2.8 ทิม เบอร์เนิร์ส-ลี ผู้คิดค้น HTML

HTML (Hypertext Markup Language) ได้ถูกพัฒนาโดย ทิมเบอร์เนิร์ส-ลี (Tim Berners Lee) แห่งศูนย์ปฏิบัติการวิจัยทาง อนุภาคฟิสิกส์ของยุโรป (CERN) แห่งกรุงเจนีวา ประเทศสวิตเซอร์แลนด์ ซึ่งมีแนวความคิดที่จะให้นักวิทยาศาสตร์ภายในสถาบันค้นหา และสื่อสารข้อมูลถึงกันและกันได้สะดวกขึ้น จึงคิดค้นวิธีการที่จะถ่ายทอดข้อมูลเอกสารในรูปของ ไฮเปอร์เท็กซ์ (Hypertext) ซึ่งเป็นรูป แบบเอกสารที่แต่ละหน้าเชื่อมโยงถึงกันได้ จากนั้นได้มีการพัฒนาเครื่องมือที่เรียกว่า เว็บเบราว์เซอร์ (Web Browser) เพื่อใช้ในการอ่าน ข้อมูลเอกสารไฮเปอร์เท็กซ์ โดยมีโมเสค (MOSAIC) เป็นเว็บเบราว์เซอร์ ตัวแรกที่ได้ถือกำเนิดขึ้นมา และมี HTTP (Hypertext Transport Protocol) เป็นโปรโตคอล (Protocol) ที่ใช้ในการสื่อสารรับส่งข้อมูล จากนั้นเอกสารไฮเปอร์เท็กซ์ได้ถูกพัฒนาขึ้นใหม่ เรียกว่า ภาษา HTML

2.8 ภาษาพีเอชพี (PHP)

รูปที่ 2.9 โลโก้ของพีเอชพี (PHP)

PHP ย่อมาจาก PHP Hypertext Preprocessor แต่เดิมย่อมาจาก Personal Home Page Tools เป็นภาษาคอมพิวเตอร์จำพวก scripting language ภาษาจำพวกนี้คำสั่งต่างๆจะเก็บอยู่ในไฟล์ที่เรียกว่า script และเวลาใช้งานต้องอาศัยตัวแปรชุดคำสั่ง ตัวอย่างของภาษาสคริปก็เช่น JavaScript, Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมา เพื่อใช้งานในการสร้างเอกสารแบบ HTMLโดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษาที่เรียกว่า server-side หรือ HTML-embedded scripting language นั้นคือในทุกๆ ครั้งก่อนที่เครื่องคอมพิวเตอร์ซึ่งให้บริการเป็น Web server จะส่งหน้าเว็บเพจที่เขียนด้วย PHP ให้เรา มันจะทำการประมวลผลตามคำสั่งที่มีอยู่ให้เสร็จเสียก่อน แล้วจึงค่อยส่งผลลัพธ์ที่ได้ให้เรา ผลลัพธ์ที่ได้นั้นก็คือเว็บเพจที่เราเห็นนั่นเอง ถือได้ว่า PHP เป็นเครื่องมือที่สำคัญชนิดหนึ่งที่ช่วยให้เราสามารถสร้าง Dynamic Web pages (เว็บเพจที่มีการโต้ตอบกับผู้ใช้) ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น

PHP เป็นผลงานที่เติบโตมาจากกลุ่มของนักพัฒนาในเชิงเปิดเผยรหัสต้นฉบับ หรือ OpenSource ดังนั้น PHP จึงมีการพัฒนาไปอย่างรวดเร็ว และแพร่หลายโดยเฉพาะอย่างยิ่งเมื่อใช้ร่วมกับ Apache Web server ระบบปฏิบัติอย่างเช่น Linux หรือ FreeBSD เป็นต้น ในปัจจุบัน PHP สามารถใช้ร่วมกับ Web Server หลายๆตัวบนระบบปฏิบัติการอย่างเช่น Windows 95/98/NT เป็นต้น [13]

2.8.1 การพัฒนาของพีเอชพี (PHP)

1) PHP

PHP เกิดในปี 1994 โดยใช้ข้อดีของภาษา C และ Perl เรียกว่า Personal Home Page และได้สร้างส่วนติดต่อกับฐานข้อมูลที่ชื่อว่า Form Interpreter (FI) รวมทั้งสองส่วน เรียกว่า PHP/FI ซึ่งก็เป็นจุดเริ่มต้นของ PHP พัฒนาต่อ ในลักษณะของ

Open Source ภายหลังมีความนิยมขึ้นเป็นอย่างมากภายใน 3 ปีมีเว็บไซต์ที่ใช้ PHP/FI ในติดต่อฐานข้อมูลและแสดงผลแบบ ไดนามิกและอื่นๆ มากกว่า 50000 ไซต์

2) PHP2 (ในตอนนั้นใช้ชื่อว่า PHP/FI)

ในช่วงระหว่าง 1995-1997 Rasmus Lerdorf ได้มีผู้ที่มาช่วยพัฒนาอีก 2 คนคือ Zeev Suraski และ Andi Gutmans ชาวอิสราเอล ซึ่งปรับปรุงโค้ดของ Lerdorf ใหม่โดยใช้ C++ ให้มีความสามารถจัดการเกี่ยวกับแบบฟอร์มข้อมูลที่ถูกสร้างมาจากภาษา HTML และสนับสนุนการติดต่อกับโปรแกรมจัดการฐานข้อมูล mSQL จึงทำให้ PHP เริ่มถูกใช้มากขึ้นอย่างรวดเร็ว และเริ่มมีผู้สนับสนุนการใช้งาน PHP มากขึ้น โดยในปลายปี 1996 PHP ถูกนำไปใช้ประมาณ 15,000 เว็บทั่วโลก และเพิ่มจำนวนขึ้นเรื่อยๆ ต่อมาก็มผู้เข้ามาช่วยพัฒนาอีก 3 คน คือ Stig Bakken รับผิดชอบความสามารถในการติดต่อ Oracle, Shane Caraveo รับผิดชอบดูแล PHP บน Window 9x/NT,และ Jim Winstead รับผิดชอบการตรวจความบกพร่องต่างๆ และได้เปลี่ยนชื่อเป็น Professional Home Page ในเวอร์ชั่นที่ 2

3) PHP3

ออกมาในช่วงระหว่างเดือน มิถุนายน 1997 ถึง 1999 ได้ออกสู่สายตาของนักโปรแกรมเมอร์ มีคุณสมบัติเด่นคือสนับสนุนระบบปฏิบัติการทั้ง Window 95/98/ME/NT, Linux และเว็บเซร์ฟเวอร์ อย่าง IIS, PWS, Apache, OmniHTTPd สนับสนุน ระบบฐานข้อมูลได้หลายรูปแบบเช่น SQL Server, MySQL, mSQL, Oracle, Informix, ODBC

4) PHP4

ตั้งแต่ 1999 - 2007 ซึ่งได้เพิ่ม Functionการทำงานในด้านต่างๆให้มากและง่ายขึ้น โดย บริษัท Zend ซึ่งมี Zeev และ Andi Gutmans ได้ร่วมก่อตั้งขึ้น (http:.zend.com) ในเวอร์ชั่นนี้จะเป็น compile script ซึ่งในเวอร์ชั่นหน้านี้จะเป็น embed script interpreter ในปัจจุบันมีคนได้ใช้ PHP สูงกว่า 5,100,000 ไซต์ แล้วทั่วโลก และผู้พัฒนาได้ตั้งชื่อของ PHP ใหม่ว่า PHP: Hypertext Preprocessor ซึ่งหมายถึงมีประสิทธิภาพระดับโปรเฟสเซอร์สำหรับไฮเปอร์เท็กซ์

5) PHP5

ตั้งแต่ 2007-ปัจจุบัน มี ได้เพิ่ม Functions การทำงานในด้านต่าง ๆ เช่น Object Oriented Model, การกำหนดสโคป public/private/protected, Exception handling , XML และ Web Service, MySQLi และ SQLite, Zend Engine 2.0 เป็นต้น [14]

2.9 พีเอชพีมายแอดมิน (phpMyAdmin)

รูปที่ 2.10 โลโก้ของพีเอชพีมายแอดมิน (phpMyAdmin)

phpMyAdmin คือโปรแกรมที่ถูกพัฒนาโดยใช้ภาษา PHP เพื่อใช้ในการบริหารจัดการฐานข้อมูล MySQL แทนการคีย์คำสั่ง เนื่องจากถ้าเราจะใช้ฐานข้อมูลที่เป็น MySQL บางครั้งจะมีความลำบากและยุ่งยากในการใช้งาน ดังนั้นจึงมีเครื่องมือในการจัดการฐานข้อมูล MySQL ขึ้นมา เพื่อให้สามารถจัดการ ตัว DBMS ที่เป็น MySQL ได้ง่ายและสะดวกยิ่งขึ้น โดย phpMyAdmin ก็ถือเป็นเครื่องมือชนิดหนึ่งในการจัดการนั้นเอง[15]

phpMyAdmin เป็นส่วนต่อประสานที่สร้างโดยภาษาพีเอชพี ซึ่งใช้จัดการฐานข้อมูล MySQL ผ่านเว็บเบราว์เซอร์ โดยสามารถที่จะทำการสร้างฐานข้อมูลใหม่ หรือทำการสร้าง TABLE ใหม่ๆ และยังมี function ที่ใช้สำหรับการทดสอบการ query ข้อมูลด้วยภาษา SQL พร้อมกันนั้นยังสามารถทำการ insert delete update หรือแม้กระทั่งใช้ คำสั่งต่างๆ เหมือนกับกันการใช้ภาษา SQL ในการสร้างตารางข้อมูล

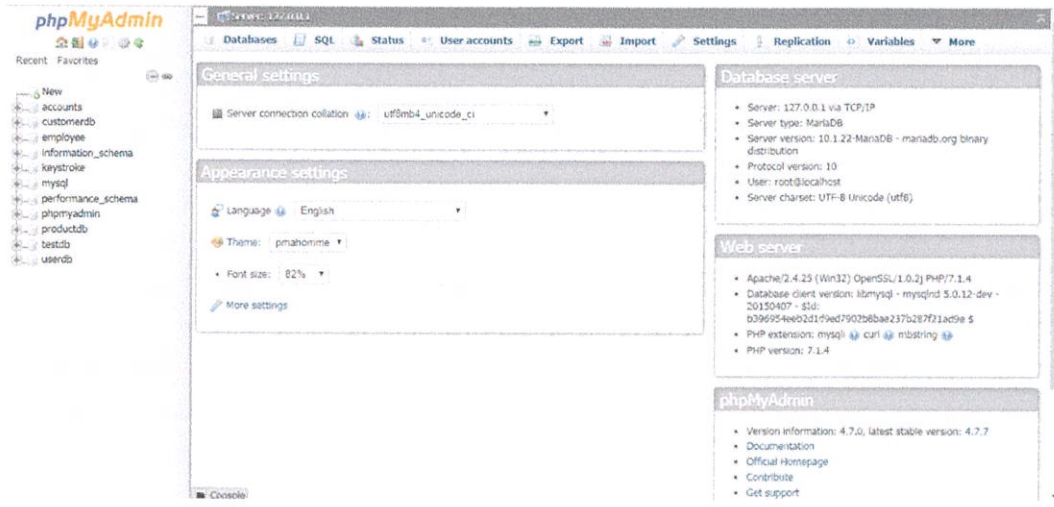
phpMyAdmin เป็นโปรแกรมประเภท MySQL Client ตัวหนึ่งที่ใช้ในการจัดการข้อมูล MySQL ผ่าน Web browser ได้โดยตรง phpMyAdmin ตัวนี้จะทำงานบน Web server เป็น PHP Application ที่ใช้ควบคุมจัดการ MySQL Server ความสามารถของ phpMyAdmin คือ

1) สร้างและลบ Database

2) สร้างและจัดการ Table เช่น แทรก record, ลบ record, แก้ไข record, ลบ Table, แก้ไข field

3) โหลดเท็กซ์ไฟล์เข้าไปเก็บเป็นข้อมูลในตารางได้

4) หาผลสรุป (Query) ด้วยคำสั่ง SQL



รูปที่ 2.11 ตัวอย่างการใช้งานพีเอพีมายแอดมิน (phpMyAdmin)

2.10 มายเอสคิวแอล (MySQL)

รูปที่ 2.12 โลโก้ของมายเอสคิวแอล (MySQL)

MySQL คือ โปรแกรมระบบจัดการฐานข้อมูล ที่พัฒนาโดยบริษัท MySQL AB มีหน้าที่เก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่ง SQL เป็นเครื่องมือสำหรับเก็บข้อมูล ที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่นอย่างบูรณาการ เพื่อให้ได้ระบบงานที่รองรับ ความต้องการของผู้ใช้ เช่นทำงานร่วมกับเครื่องบริการเว็บ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษา php ภาษา aps.net หรือภาษาเจเอสพี เป็นต้น หรือทำงานร่วมกับโปรแกรมประยุกต์ (Application Program) เช่น ภาษาวิชวลเบสิกดอทเน็ต ภาษาจาวา หรือภาษาซีชาร์ป เป็นต้น โปรแกรมถูกออกแบบให้สามารถทำงานได้

บนระบบปฏิบัติการที่หลากหลาย และเป็นระบบฐานข้อมูลโอเพนทซอร์ท (Open Source) ที่ถูกนำไปใช้งานมากที่สุด [16]

MySQL : มายเอสคิวแอล เป็นระบบจัดการฐานข้อมูลโดยใช้ภาษา SQL. แม้ว่า MySQL เป็นซอฟต์แวร์โอเพนซอร์ส แต่แตกต่างจากซอฟต์แวร์โอเพนซอร์สทั่วไป โดยมีการพัฒนาภายใต้บริษัท MySQL AB ในประเทศสวีเดน โดยจัดการ MySQL ทั้งในแบบที่ให้ใช้ฟรี และแบบที่ใช้ในเชิงธุรกิจ

MySQL สร้างขึ้นโดยชาวสวีเดน 2 คน และชาวฟินแลนด์ ชื่อ David Axmark, Allan Larsson และ Michael "Monty" Widenius. ในปัจจุบันบริษัทซันไมโครซิสเต็มส์ (Sun Microsystems, Inc.) เข้าซื้อกิจการของ MySQL AB เรียบร้อยแล้ว ฉะนั้นผลิตภัณฑ์ภายใต้ MySQL AB ทั้งหมดจะตกเป็นของซัน

ชื่อ "MySQL" อ่านออกเสียงว่า "มายเอสคิวเอล" หรือ "มายเอสคิวแอล" (ในการอ่านอักษร L ในภาษาไทย) ซึ่งทางซอฟต์แวร์ไม่ได้อ่าน มายซีเควล หรือ มายซีควล เหมือนกับซอฟต์แวร์จัดการฐานข้อมูลตัวอื่น

2.10.1 ความสามารถและการทำงานของโปรแกรม MySQL

1) MySQL เป็นระบบจัดการฐานข้อมูล (DataBase Management System : DBMS)

ฐานข้อมูลมีลักษณะเป็นโครงสร้างของการเก็บรวบรวมข้อมูล การที่จะเพิ่มเติม เข้าถึงหรือประมวลผลข้อมูลที่เก็บในฐานข้อมูลจำเป็นจะต้องอาศัยระบบจัดการฐานข้อมูล ซึ่งจะทำหน้าที่เป็นตัวกลางในการจัดการกับข้อมูลในฐานข้อมูลทั้งสำหรับการ ใช้งานเฉพาะ และรองรับการทำงานของแอพลิเคชันอื่นๆ ที่ต้องการใช้งานข้อมูลในฐานข้อมูล เพื่อให้ได้รับความสะดวกในการจัดการกับข้อมูลจำนวนมาก MySQL ทำหน้าที่เป็นทั้งตัวฐานข้อมูลและระบบจัดการฐานข้อมูล

2) MySQL เป็นระบบจัดการฐานข้อมูลแบบ relational

ฐานข้อมูลแบบ relational จะทำการเก็บข้อมูลทั้งหมดในรูปแบบของตารางแทนการเก็บข้อมูลทั้งหมดลงในไฟล์ เพียงไฟล์เดียว ทำให้ทำงานได้รวดเร็วและมีความยืดหยุ่น นอกจากนั้น แต่ละตารางที่เก็บข้อมูลสามารถเชื่อมโยงเข้าหากันทำให้สามารถรวมหรือจัดกลุ่มข้อมูลได้ตามต้องการ โดยอาศัยภาษา SQL ที่เป็นส่วนหนึ่งของโปรแกรม MySQL ซึ่งเป็นภาษามาตรฐานในการเข้าถึงฐานข้อมูล

3) MySQL แจกจ่ายให้ใช้งานแบบ Open Source

ผู้ใช้งาน MySQL ทุกคนสามารถใช้งานและปรับแต่งการทำงานได้ตามต้องการ สามารถดาวน์โหลดโปรแกรม MySQL ได้จากอินเทอร์เน็ตและนำมาใช้งานโดยไม่มีค่าใช้จ่ายใดๆ

2.10.2 ข้อควรระวังที่เกี่ยวข้องกับความปลอดภัยของโปรแกรม MySQL

การใช้งานโปรแกรม MySQL ให้มีความปลอดภัยนั้น ผู้ดูแลจะต้องพิจารณาถึงวิธีการที่ผู้ใช้หรือผู้อื่นๆ จะเข้าถึงข้อมูลต่างๆ ในฐานข้อมูล และจำกัดสิทธิ์การใช้งานของผู้ที่จะเข้าใช้งานให้ได้รับสิทธิ์ให้น้อยที่สุด เท่าที่จะเป็นไปได้ เท่าที่จำเป็นต่อการใช้งานเท่านั้น มีข้อควรระวังดังต่อไปนี้

1) นอกจากผู้ดูแลฐานข้อมูลแล้ว ไม่ควรให้ผู้ใช้งานคนอื่นๆ เข้าถึงตาราง user ของฐานข้อมูล

2) ผู้ดูแลฐานข้อมูลควรศึกษาถึงระบบการให้สิทธิ์การเข้าถึงฐานข้อมูลโดยละเอียด

3) รหัสผ่านที่ใช้งานจะต้องเป็นรหัสผ่านที่ดี และการเก็บค่ารหัสผ่านจะต้องได้รับการเข้ารหัส

4) ถ้ามีการส่งข้อมูลผ่านอินเทอร์เน็ต ข้อมูลที่ส่งจะต้องได้รับการเข้ารหัสทุกครั้ง

5) ตรวจสอบการส่งข้อมูลโดยใช้คำสั่ง tcpdump และ strings ของระบบปฏิบัติการ

6) ต้องกำหนดให้ผู้ใช้งานฐานข้อมูลทุกคนมีรหัสผ่านในการเข้าใช้งาน

2.10.3 คำสั่งใน MySQL

2.10.3.1 คำสั่งพื้นฐานทั่วไป [17]

- คำสั่งติดต่อ MySQL

```php
<?
$host = "localhost" ;
$username = "xxx" ; // ชื่อผู้ใช้ในการติดต่อกับฐานข้อมูล
$password = "xxx" ; // password ในการเชื่อมต่อกับฐานข้อมูล
$dbname = "xxx" ;  // ชื่อฐานข้อมูล

$c = mysql_connect($host,$username,$password); // ติดต่อฐานข้อมูล

if (!$c){
    echo "<h3>ERROR : ไม่สามารถติดต่อฐานข้อมูลได้</h3>"; // ไว้แสดง error
    exit();
}
mysql_query("SET NAMES UTF8"); // สำหรับการติดต่อฐานข้อมูลแบบ UTF8
mysql_select_db($dbname,$c); // สำหรับเลือกชื่อฐานข้อมูลในการติดต่อไว้เลย
?>
```

รูปที่ 2.13 คำสั่งติดต่อ MySQL

- การสร้างฐานข้อมูล

```php
$c= mysql_connect("localhost","username","password");
mysql_create_db("dbname",$c);
```

รูปที่ 2.14 คำสั่งสร้างฐานข้อมูล

- การลบฐานข้อมูล

```
$c = mysql_connect("localhost","username","password");
mysql_drop_db("dbname",$c);
```

รูปที่ 2.15 คำสั่งลบฐานข้อมูล

- การยกเลิกการเชื่อมต่อฐานข้อมูล

```
$c = mysql_connect("localhost","username","password");
...
...
...
mysql_close($c);
```

รูปที่ 2.16 คำสั่งยกเลิกการเชื่อมต่อฐานข้อมูล

2.10.3.2 คำสั่งในการเลือกข้อมูล

- การเลือกข้อมูลโดยเจาะจงค่าของ Field Name

```
select * from table_name where field_name="xxx"
```

รูปที่ 2.17 คำสั่งเลือกข้อมูลโดยเจาะจงค่าของ Field Name

- การเลือกข้อมูลโดยกำหนดค่าของ Field Name และเช็คค่าเทียบกับตัวแปร

```
$data = "'xxx','yyy'";
$query = "select * from table_name where field_name in($data);"
```

รูปที่ 2.18 คำสั่งเลือกข้อมูลโดยกำหนดค่าของ Field Name และเช็คค่าเทียบกับตัวแปร

- การดึงข้อมูลเฉพาะ Field ที่ต้องการ

```
select field_name1,field_name2 from table_name
```

รูปที่ 2.19 คำสั่งดึงข้อมูลเฉพาะ Field ที่ต้องการ

2.10.3.3 การแสดงผลข้อมูลที่ได้จากการคิวรี่

- การดึงข้อมูลแบบ cell เดียว

```
$query = "SELECT filed_name1 FROM table_name where filed_name2='xxx'";
$rs = mysql_query($query);
$cell_data = mysql_result($rs,0,0);
```

รูปที่ 2.20 คำสั่งดึงข้อมูลแบบ cell เดียว

- แบบใช้ mysql_fetch_array( )

```
$sql ="select * from table_name";
$result = mysql_db_query($dbname,$sql);
while($rs=mysql_fetch_array($result)) {
    $variable1 = $rs["filed_name1"];
    $variable2 = $rs["filed_name2"];
    $variable3 = $rs["filed_name3"];
    echo $variable1;
    echo $variable2;
    echo $variable3;
}
```

รูปที่ 2.21 คำสั่งใช้ mysql_fetch_array( )

- แบบใช้ mysql_fetch_assoc( )

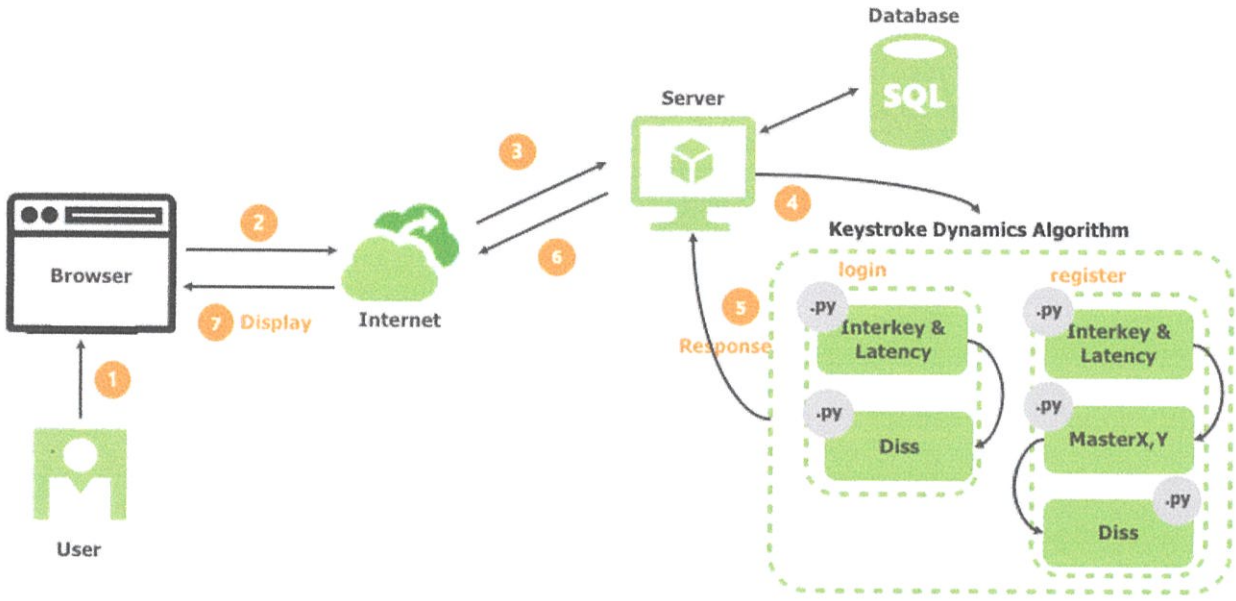
```
$sql ="select * from table_name";
$result = mysql_db_query($dbname,$sql);
while($rs=mysql_fetch_assoc($result)) {
    $variable1 = $rs["filed_name1"];
    $variable2 = $rs["filed_name2"];
    $variable3 = $rs["filed_name3"];
    echo $variable1;
    echo $variable2;
    echo $variable3;
}
```

รูปที่ 2.22 คำสั่งใช้ mysql_fetch_assoc( )

ทั้ง 2 คำสั่งนี้มีความแตกต่างตรงค่าใน Array ที่ถูกเก็บไว้ โดยที่ mysql_fetch_assoc( ) จะทำให้ประหยัด memory มากกว่า

บทที่ 3

การออกแบบและการจัดทำปริญญานิพนธ์



รูปที่ 3.1 บล็อกไดอะแกรมการทำงานของระบบ

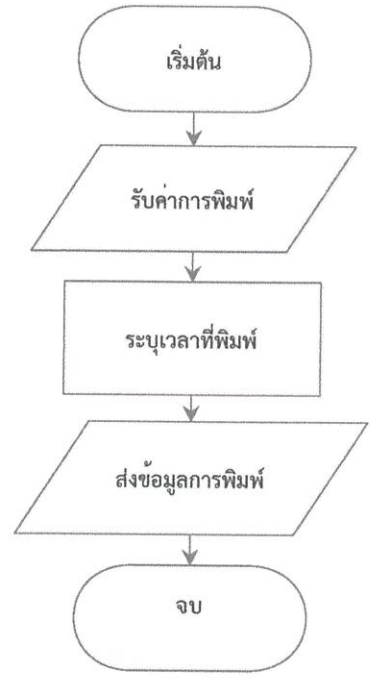
จากบล็อกไดอะแกรมสามารถอธิบายการทำงานได้ดังนี้ เมื่อผู้ใช้งานทำการล็อกอินเข้าใช้งานระบบหรือทำการสมัครสมาชิก ระบบจะส่งค่าจังหวะการพิมพ์ไปที่เครื่องเซิร์ฟเวอร์ เพื่อทำการประมวลผล โดยถ้าเลือกทำการล็อกอิน ระบบจะนำค่าจังหวะการพิมพ์ มาทำการหาค่า Interkey Time และค่า Latency Time เมื่อได้ค่าทั้งสองจะถูกนำไปคำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์ โดยเปรียบเทียบค่าจากฐานข้อมูล เพื่อจะทำการพิสูจน์จังหวะการพิมพ์ที่ป้อนเข้ามาว่าเป็นของผู้ใช้งานจริงหรือไม่ ถ้าข้อมูลออกมาเป็นจริง ระบบจะยอมให้เข้าสู่ระบบได้ แต่ถ้าไม่เป็นจริง ระบบจะไม่ยอมให้เข้าสู่ระบบ ในส่วนที่ผู้ใช้งานเลือกสมัครสมาชิก ระบบจะนำค่าจังหวะการพิมพ์มาทำการประมวลผลเพื่อเก็บข้อมูลลงสู่ฐานข้อมูล

3.1 การออกแบบ

3.1.1 การออกแบบหน้าเว็บไซต์เก็บค่าจังหวะการพิมพ์

การเก็บค่าจังหวะการพิมพ์จะเริ่มทำงานจากการเรียกหน้าเว็บไซต์เก็บจังหวะการพิมพ์ เพื่อรับค่าการกดแป้นพิมพ์เข้ามาจากคีย์บอร์ด จากนั้นระบบทำการระบุเวลาที่ทำการกดแป้นพิมพ์

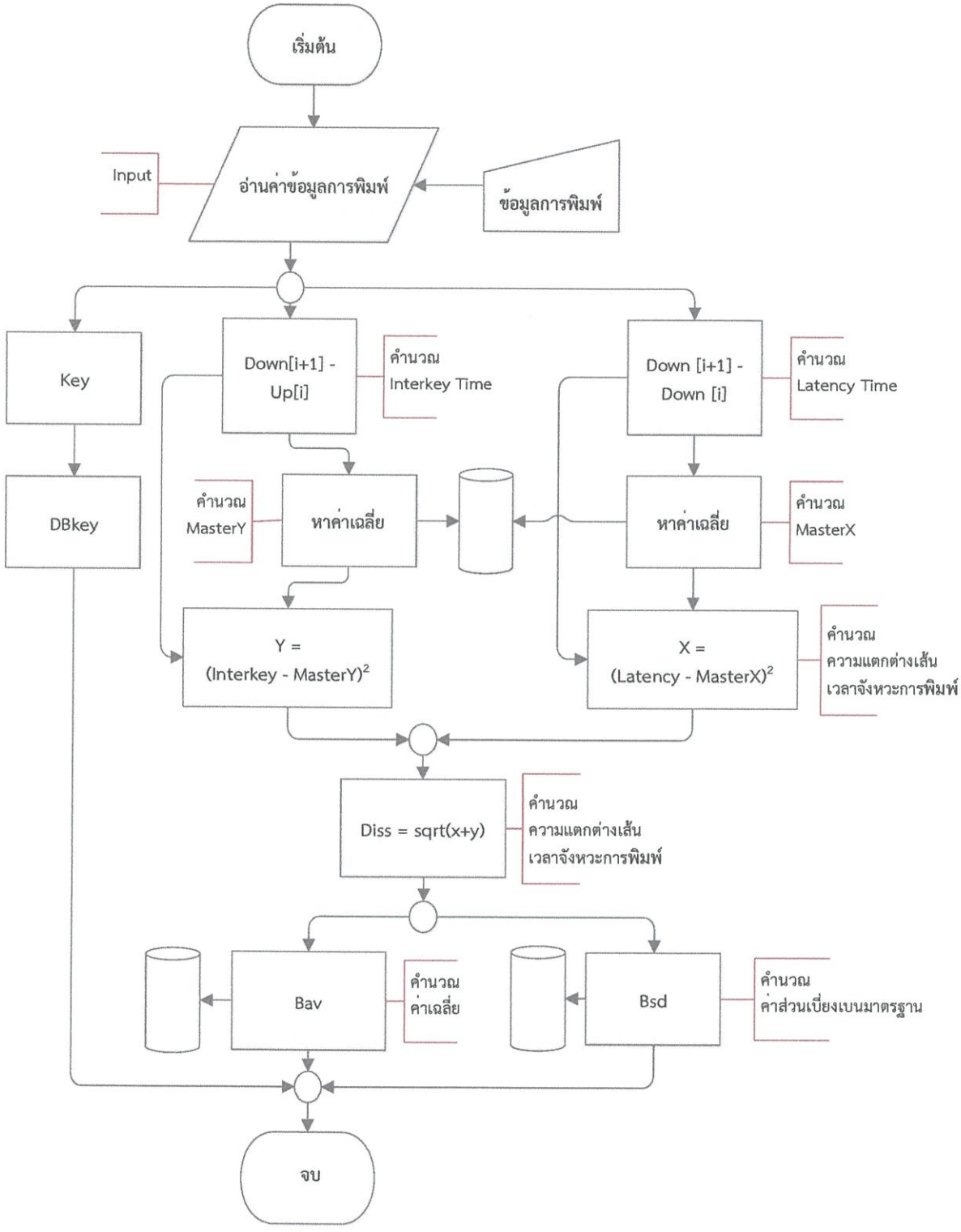
และเวลาที่ทำการยกแป้นพิมพ์ขึ้นในแต่ละคีย์ ซึ่งแผนภาพการออกแบบหน้าเว็บไซต์เก็บค่าจังหวะการพิมพ์ เป็นไปดังรูปที่ 3.2



รูปที่ 3.2 แผนภาพการออกแบบหน้าเว็บไซต์เก็บค่าจังหวะการพิมพ์

3.1.2 การออกแบบการคำนวณหาค่าจังหวะการพิมพ์เพื่อเก็บลงฐานข้อมูล

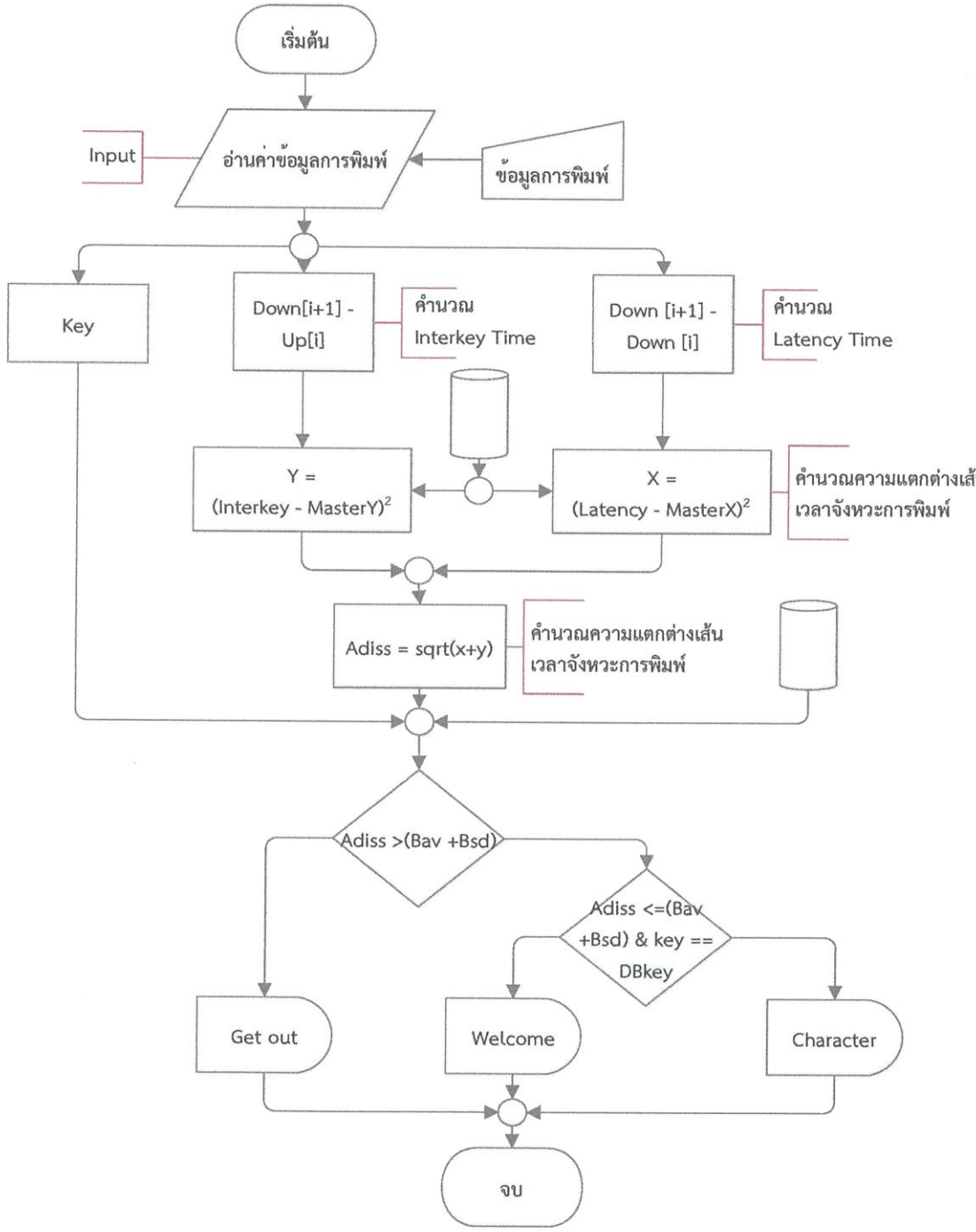
การออกแบบการคำนวณหาค่าจังหวะการพิมพ์ เพื่อเก็บลงฐานข้อมูล เป็นดังรูปที่ 3.3 โดยเมื่อได้รับข้อมูลการพิมพ์มาแล้ว ระบบจะทำการระบุคีย์ที่ทำการกด เก็บลงในฐานข้อมูล และทำการคำนวณเวลาในการกดคีย์บนแป้นพิมพ์ออกเป็น 2 แบบ คือ Interkey Time และ Latency Time จากนั้นจะนำเวลาที่ได้ไปหาค่าเฉลี่ยเพื่อทำเป็นเส้นเวลาจังหวะการพิมพ์ต้นแบบ X และ Y เก็บลงในฐานข้อมูล จากนั้นระบบจะทำการหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์ โดยจะนำ Interkey Time , Latency Time และเส้นเวลาจังหวะการพิมพ์ต้นแบบ X และ Y มาทำการคำนวณ เมื่อคำนวณได้ค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์มาแล้ว จะนำค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์มาคำนวณหาค่าเฉลี่ย และค่าส่วนเบี่ยงเบนมาตรฐาน เพื่อเก็บลงฐานข้อมูล



รูปที่ 3.3 แผนภาพการออกแบบการคำนวณหาค่าจังหวะการพิมพ์เพื่อเก็บลงฐานข้อมูล

3.1.3 การออกแบบการพิสูจน์ตัวตน

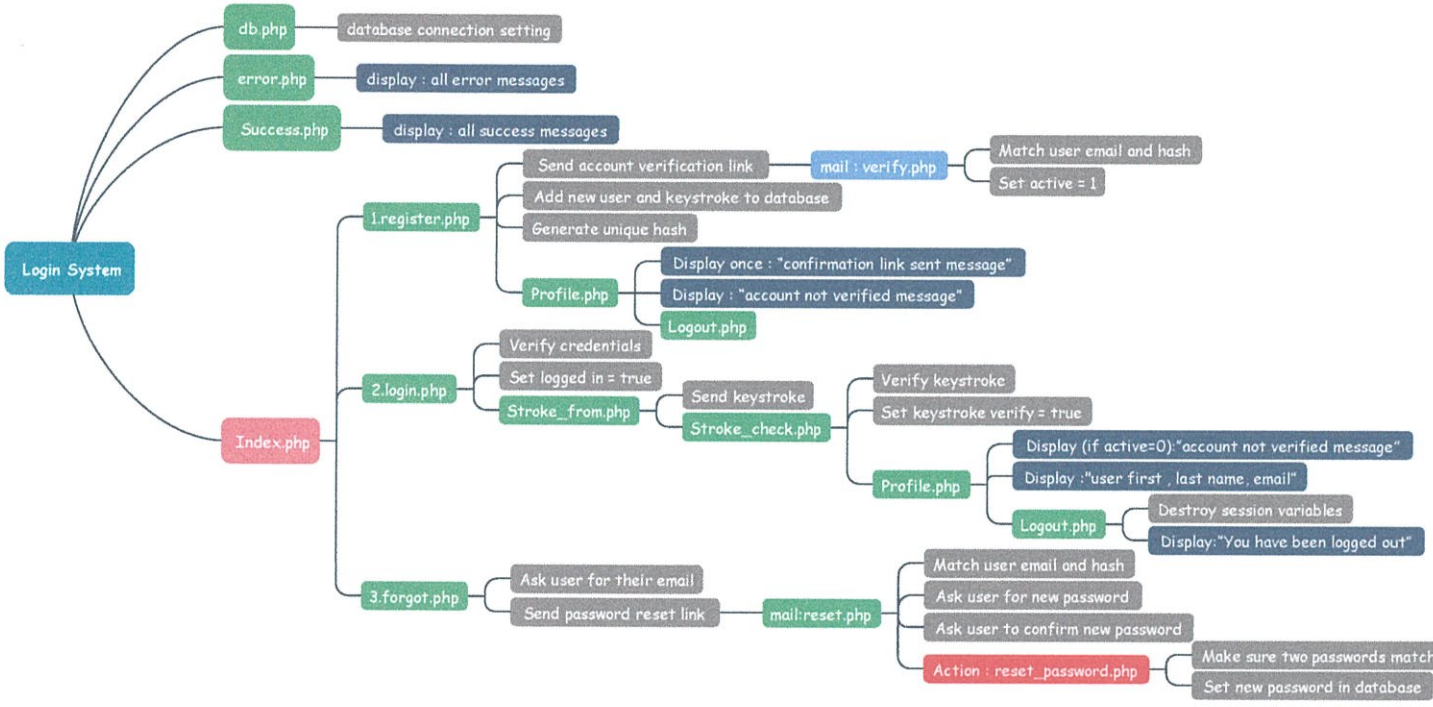
การออกแบบการพิสูจน์ตัวตน เป็นดังรูปที่ 3.4 โดยเมื่อได้รับข้อมูลการพิมพ์มาแล้ว ระบบจะทำการระบุคีย์ที่ทำการกด และทำการคำนวณเวลาในการกดบนแป้นพิมพ์ออกมา 2 แบบ คือ Interkey Time และ Latency Time จากนั้นจะนำเวลาที่ได้มาทำการคำนวณความแตกต่างของเส้นเวลาจังหวะการพิมพ์เทียบกับเส้นจังหวะการพิมพ์ต้นแบบที่เก็บไว้ในฐานข้อมูล หลังจากนั้นจะนำค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์ที่ได้ นำมาเทียบกับข้อมูลในฐานข้อมูล ซึ่งถ้าระบบประมวลผลแล้วว่าเป็นตัวจริง จะแสดงข้อความว่า “Welcome” แต่ถ้าจังหวะการพิมพ์ของผู้ใช้ถูกต้อง แต่คีย์ที่กดไม่ถูกต้อง จะแสดงข้อความว่า “Character” และถ้าไม่ใช่จะแสดงข้อความว่า “Get out”



รูปที่ 3.4 แผนภาพการออกแบบการพิสูจน์ตัวตน

3.1.3 การออกแบบเว็บไซต์ล็อกอินต้นแบบสำหรับระบบการพิสูจน์ตัวตนด้วยจังหวะการพิมพ์

การออกแบบเว็บไซต์ล็อกอินต้นแบบสำหรับระบบการพิสูจน์ตัวตนด้วยจังหวะการพิมพ์ และการทำงานของส่วนต่างๆของเว็บไซต์ล็อกอินต้นแบบ เป็นดังรูปที่ 3.4



รูปที่ 3.5 แผนภาพการทำงานของเว็บไซต์ล็อกอินต้นแบบ

3.2 เครื่องมือที่ใช้ในการทดลอง

3.2.1 อุปกรณ์ด้านฮาร์ดแวร์

เครื่องคอมพิวเตอร์ (Computer) เพื่อใช้เป็นเครื่องมือสำหรับเขียนโปรแกรม ใช้เป็นเครื่องเซิร์ฟเวอร์ และใช้สำหรับการแสดงผล

3.2.2 อุปกรณ์ด้านซอฟแวร์

โปรแกรมและภาษาที่ต้องใช้ในการพัฒนาเว็บไซต์แสดงผลและเขียนโปรแกรมประมวลผล

1) โปรแกรม Apache 2 web server

2) โปรแกรม Atom

3) ภาษา JavaScript

4) ภาษา Python

5) ภาษา html

3.3 การจัดเก็บผลการทดลอง

3.3.1 เก็บค่าจังหวะการพิมพ์ของคำต่างๆ ตามที่กำหนด

ในส่วนนี้จะทำการเขียนหน้าเว็บไซต์ขึ้นมา เพื่อทำการเก็บค่าจังหวะการพิมพ์ของคำต่างๆ เพื่อนำไปประมวลผลต่อไป

3.3.2 คำนวณหาค่าเวลาในการกดคีย์บนแป้นพิมพ์ของคำต่างๆ

ในส่วนนี้จะทำการเขียนโค้ดโปรแกรมขึ้นมา เพื่อคำควณค่าเวลาในการกดคีย์บนแป้นพิมพ์ในแบบต่างๆ โดยนำค่าจังหวะการพิมพ์มาทำการประมวลผล เพื่อจะหาค่า Interkey Time และ Latency Time

3.3.3 คำนวณหาค่าเส้นเวลาจังหวะการพิมพ์ต้นแบบของคำต่างๆ

ในส่วนนี้จะทำการเขียนโค้ดโปรแกรมขึ้นมา เพื่อคำนวณหาค่าเส้นเวลาจังหวะการพิมพ์ต้นแบบ โดยนำค่าเวลาในการกดคีย์บนแป้นพิมพ์ในแบบ Latency Time และ Interkey Time มาประมวลผล หาค่าเฉลี่ย เพื่อนำมาเป็นเส้นเวลาจังหวะการพิมพ์ต้นแบบ

3.3.4 คำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์

ในส่วนนี้จะทำการเขียนโค๊ดโปรแกรมขึ้นมา เพื่อคำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์ โดยนำค่าเส้นเวลาจังหวะการพิมพ์ของข้อมูล 3,6,9 และ12 ข้อมูลแรก มาเปรียบเทียบกับเส้นเวลาจังหวะการพิมพ์ต้นแบบ และคำนวณหาค่าความแตกต่าง

3.3.5 การทดสอบผลการพิสูจน์ตัวตน

ในส่วนนี้จะทำการเขียนโค๊ดโปรแกรมเพื่อทดสอบผลการพิสูจน์ตัวตน โดยประมวลผลข้อมูลจากอาสาสมัครที่สามารถเข้าใช้งานระบบได้ 20 คน และอาสาสมัครที่ไม่สามารถเข้าใจงานระบบได้ 5 คน ในการทดสอบการพิสูจน์ตัวตน

3.3.6 ทดสอบการใช้งานเว็บไซต์ ล็อกอินต้นแบบ สำหรับระบบการพิสูจน์ตัวตนด้วยจังหวะการพิมพ์

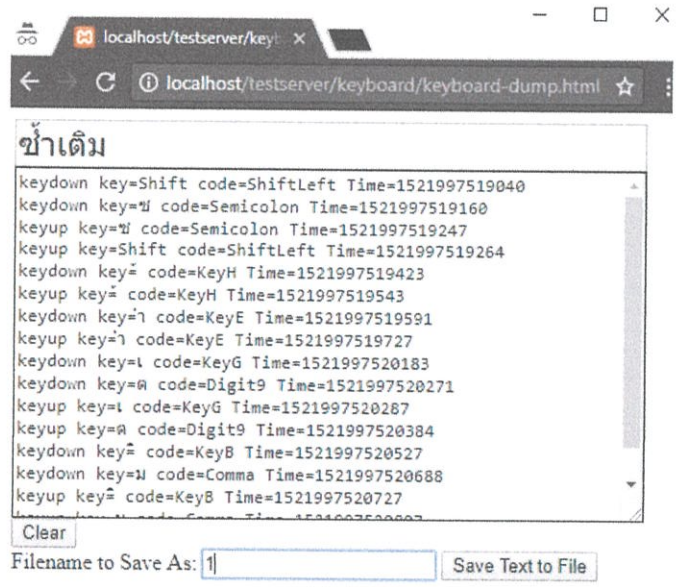
ในส่วนนี้จะทำการทดสอบการทำงานของเว็บไซต์ล็อกอินต้นแบบในส่วนต่างๆ และทำการทดสอบระบบการพิสูจน์ตัวตนด้วยจังหวะการพิมพ์ เพื่อนำไปใช้งานในเว็บไซต์

บทที่ 4

ผลการทดลอง

4.1 ผลการทดสอบการเก็บค่าจังหวะการพิมพ์

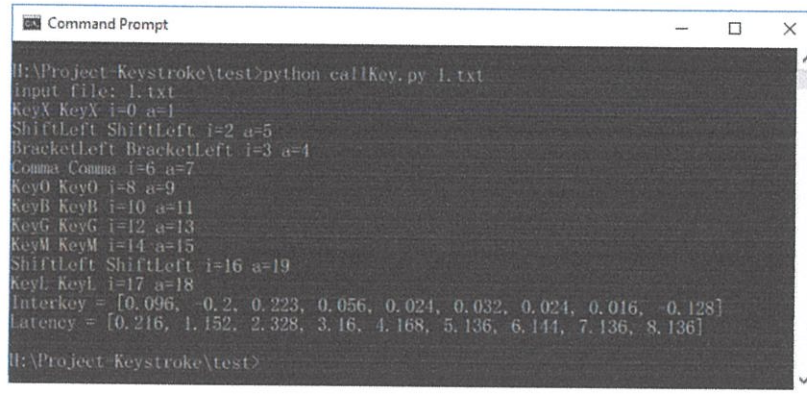
ในการทดสอบการเก็บค่าจังหวะการพิมพ์ จะทำการทดสอบผ่านหน้าเว็บไซน์ที่สร้างขึ้นโดยทำการป้อนข้อความตามที่กำหนด โดยผลลัพธ์เป็นไปดังรูปที่ 4.1



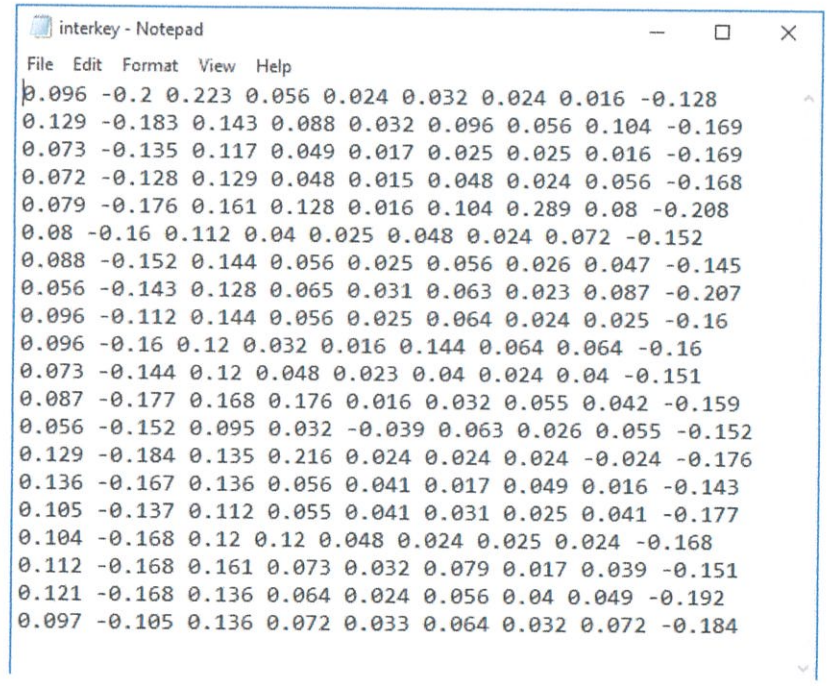
รูปที่ 4.1 ผลการทดสอบการเก็บค่าจังหวะการพิมพ์

4.2 ผลการทดสอบการคำนวณหาค่าเวลาในการกดคีย์บนแป้นพิมพ์

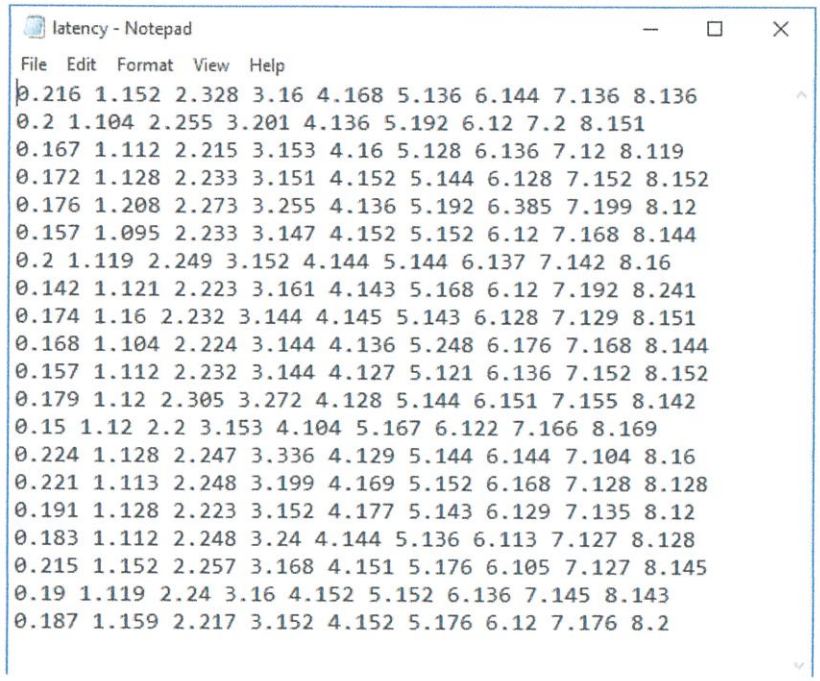
การคำนวณหาค่าเวลาในการกดคีย์บนแป้นพิมพ์ ทำโดยการเขียนโค๊ดโปรแกรมขึ้นมา เมื่อป้อนค่าข้อมูลจังหวะการพิมพ์เข้าไป โค๊ดโปรแกรมจะทำการคำนวณค่าเวลาในการกดคีย์บนแป้นพิมพ์ออกมาเป็น 2 ค่า คือ Interkey Time และ Latency Time พร้อมทั้งทำการจัดลำดับคำในการพิมพ์ โดยมีผลการทดสอบดังรูปที่ 4.2 ถึงรูปที่ 4.4



รูปที่ 4.2 ผลการทดสอบการคำนวณหาค่าเวลาในการกดคีย์บนแป้นพิมพ์



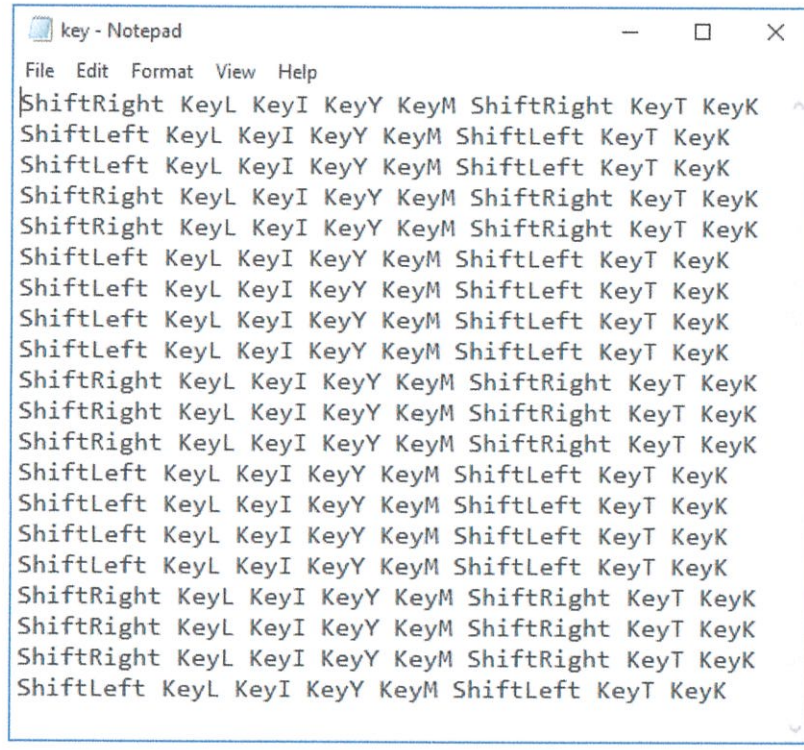
รูปที่ 4.3 ผลการคำนวณหาค่าเวลาในแบบ Interkey Time ของข้อมูล 20 ชุด



รูปที่ 4.4 ผลการคำนวณหาค่าเวลาในแบบ Latency Time ของข้อมูล 20 ชุด

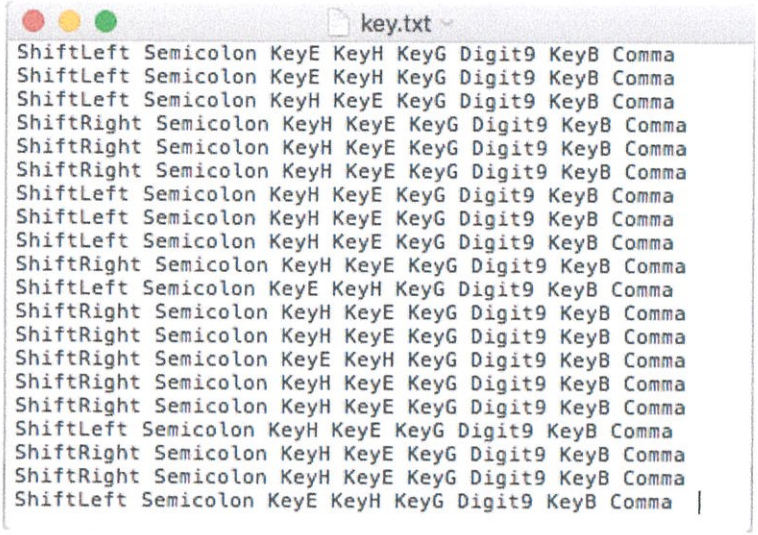
4.3 ผลการทดสอบลำดับคำในการกดคีย์บนแป้นพิมพ์

การทดสอบลำดับคำในการกดคีย์บนแป้นพิมพ์ ทำโดยการเขียนโค๊ดโปรแกรมขึ้นมา เมื่อป้อนค่าข้อมูลจังหวะการพิมพ์เข้าไป โค๊ดโปรแกรมจะทำการจัดลำดับคำในการกดคีย์บนแป้นพิมพ์ โดยมีผลการทดสอบดังรูปที่ 4.5 ถึงรูปที่ 4.7



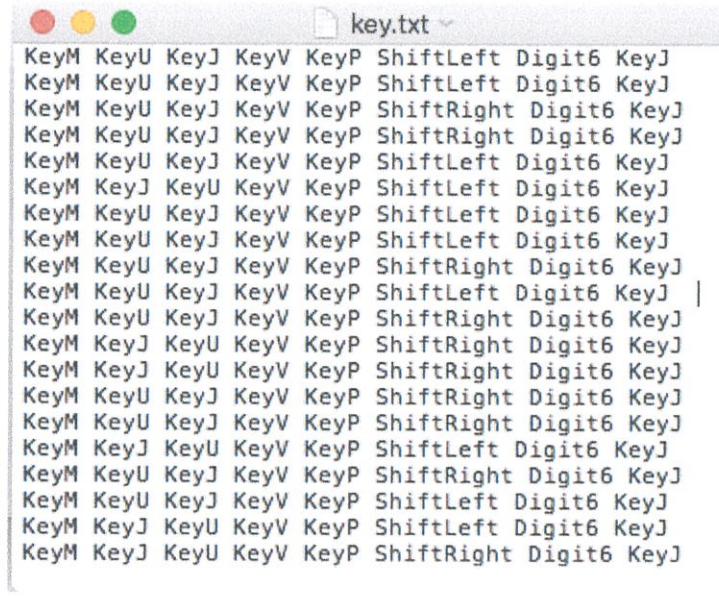
รูปที่ 4.5 ผลลำดับการพิมพ์คำว่า “ศรัทธา” ของผู้ใช้งาน 20 คน

จากรูปที่ 4.5 เป็นการทดสอบลำดับคำในการกดคีย์บนแป้นพิมพ์ ของคำว่า “ศรัทธา” ซึ่งคำว่า “ศรัทธา” จะต้องมีการกด Shift สองตำแหน่ง คือตำแหน่ง “ศ” และตำแหน่ง “ธ” จากการทดสอบจะเห็นว่าผู้ใช้งาน 20 คน กด Shift-Left จำนวน 11 คน และกด Shift-Right จำนวน 9 คน ซึ่งจุดนี้ทำให้เราสามารถใช้ลำดับคำในการกดคีย์บนแป้นพิมพ์เป็น Filter ในการพิสูจน์ตัวตนได้ โดยสามารถคัดกรองผู้บุกรุกได้ถึงเกือบ 50 เปอร์เซ็นต์



รูปที่ 4.6 ผลลำดับการพิมพ์คำว่า “ซ้ำเติม” ของผู้ใช้งาน 20 คน

จากรูปที่ 4.6 เป็นการทดสอบลำดับคำในการกดคีย์บนแป้นพิมพ์ ของคำว่า “ซ้ำเติม” ซึ่งคำว่า “ซ้ำเติม” จะต้องมีการกด Shift หนึ่งตำแหน่ง คือตำแหน่ง “ซ” และมีลำดับการกดสระอำ และไม้โท ที่สามารถแยกความแตกต่างของผู้ใช้งานแต่ละคนได้ จากการทดสอบจะเห็นว่าผู้ใช้งาน 20 คน กด Shift-Left และไม้โทก่อนสระอำ จำนวน 5 คน กด Shift-Left และสระอำก่อนไม้โท จำนวน 4 คน กด Shift-Right และไม้โทก่อนสระอำ จำนวน 10 คน และกด Shift-Right และสระอำก่อนไม้โท จำนวน 1 คน ซึ่งจุดนี้ทำให้เราสามารถใช้ลำดับคำในการกดคีย์บนแป้นพิมพ์เป็น Filter ในการพิสูจน์ตัวตนได้

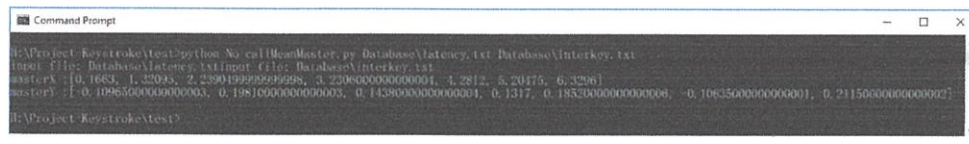


รูปที่ 4.7 ผลลำดับการพิมพ์คำว่า “ที่อยู่” ของผู้ใช้งาน 20 คน

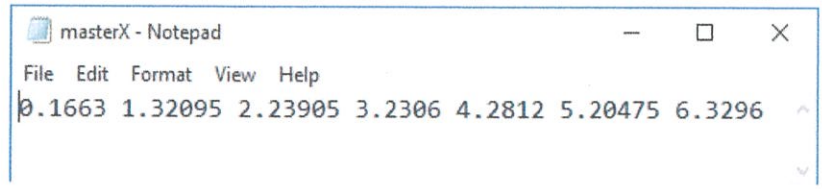
จากรูปที่ 4.7 เป็นการทดสอบลำดับคำในการกดคีย์บนแป้นพิมพ์ ของคำว่า “ที่อยู่” ซึ่งคำว่า “ที่อยู่” จะต้องมีการกด Shift หนึ่งตำแหน่ง คือตำแหน่ง “สระอู” และมีลำดับการกดสระอี ไม้เอกและสระอู ที่สามารถแยกความแตกต่างของผู้ใช้งานแต่ละคนได้ จากการทดสอบจะเห็นว่าผู้ใช้งาน 20 คน กด Shift-Left และสระอีก่อนไม้เอก จำนวน 3 คน กด Shift-Left และไม้เอกก่อนสระอี จำนวน 7 คน กด Shift-Right และไม้เอกก่อนสระอี จำนวน 3 คน และกด Shift-Right และสระอีก่อนไม้เอก จำนวน 7 คน ซึ่งจุดนี้ทำให้เราสามารถใช้ลำดับคำในการกดคีย์บนแป้นพิมพ์เป็น Filter ในการพิสูจน์ตัวตนได้

4.4 ผลการทดสอบการคำนวณหาค่าเส้นเวลาจังหวะการพิมพ์ต้นแบบ

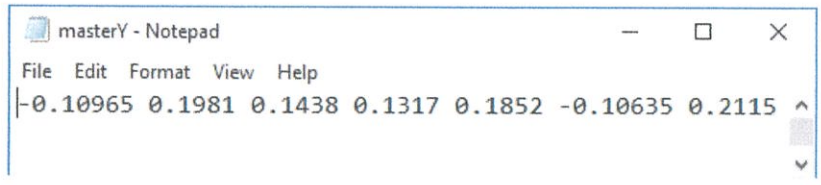
การทดสอบการคำนวณหาค่าเส้นเวลาจังหวะการพิมพ์ต้นแบบ จะทำการทดสอบผ่านโค๊ดโปรแกรมที่เขียนขึ้น โดยการนำข้อมูลเวลาแบบ Latency Time และ Interkey Time ทั้งหมด 20 ชุดข้อมูล มาทำการคำนวณ ซึ่งผลลัพธ์ของการคำนวณเป็นดังรูปที่ 4.8 ถึงรูปที่ 4.10



รูปที่ 4.8 ผลการทดสอบการคำนวณหาค่าเส้นเวลาจังหวะการพิมพ์ต้นแบบ



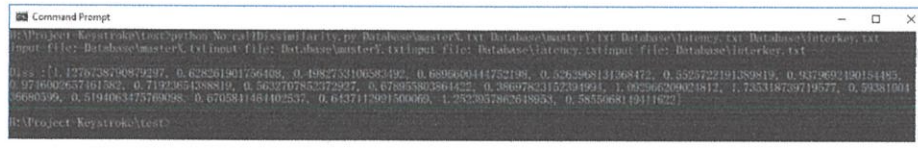
รูปที่ 4.9 ผลการคำนวณหาค่า Master X



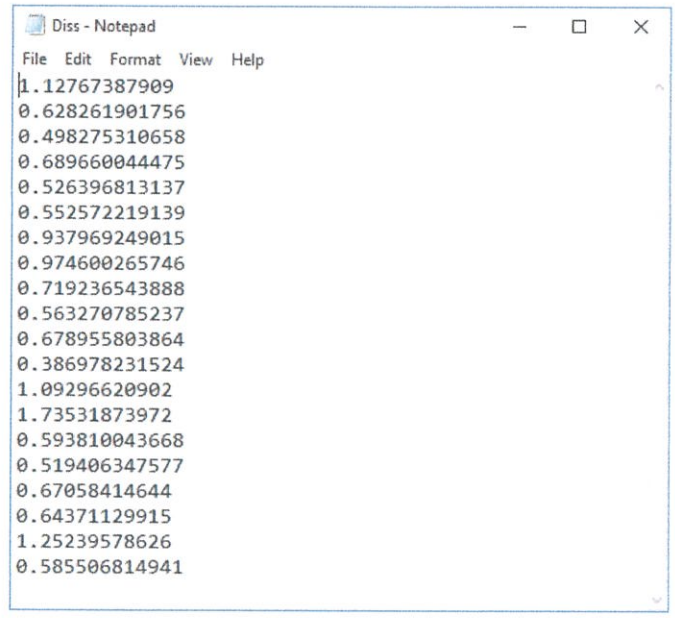
รูปที่ 4.10 ผลการคำนวณหาค่า Master Y

4.5 ผลการทดสอบการคำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์

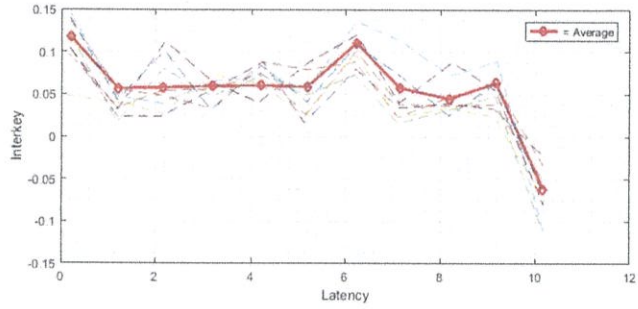
การทดสอบการคำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์ จะทำการทดสอบผ่านโค๊ดโปรแกรมที่เขียนขึ้น โดยการนำข้อมูลเวลาแบบ Latency Time และ Interkey Time ทั้งหมด 20 ชุดข้อมูล มาทำการคำนวณเปรียบเทียบกับเส้นเวลาจังหวะการพิมพ์ต้นแบบ ซึ่งผลลัพธ์ของการคำนวณเป็นดังรูปที่ 4.11 และรูปที่ 4.12



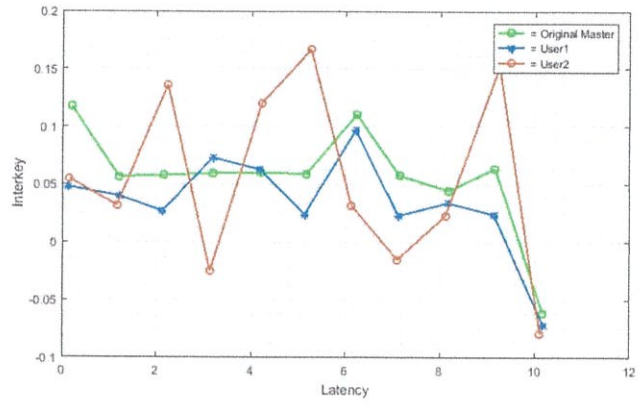
รูปที่ 4.11 ผลการทดสอบการคำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์



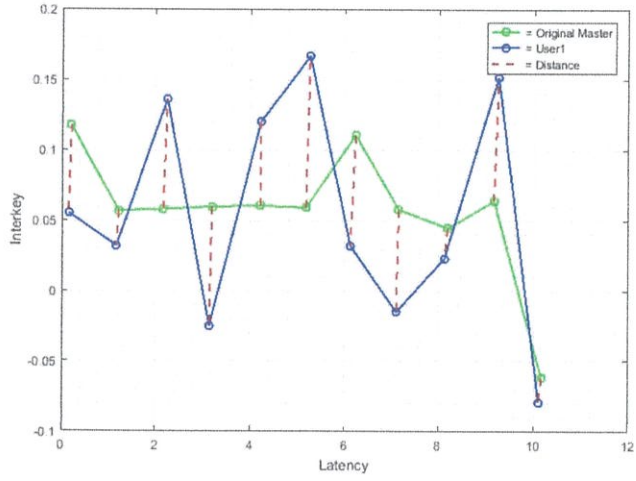
รูปที่ 4.12 ผลการคำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์



รูปที่ 4.13 กราฟความสัมพันธ์ของ Latency Time และ Interkey Time เพื่อสร้างเส้นเวลาจังหวะการพิมพ์ต้นแบบ



รูปที่ 4.14 เปรียบเทียบเส้นเวลาจังหวะการพิมพ์ต้นฉบับ
กับเส้นเวลาจังหวะการพิมพ์ที่ต้องการตรวจสอบ



รูปที่ 4.15 กราฟการหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์

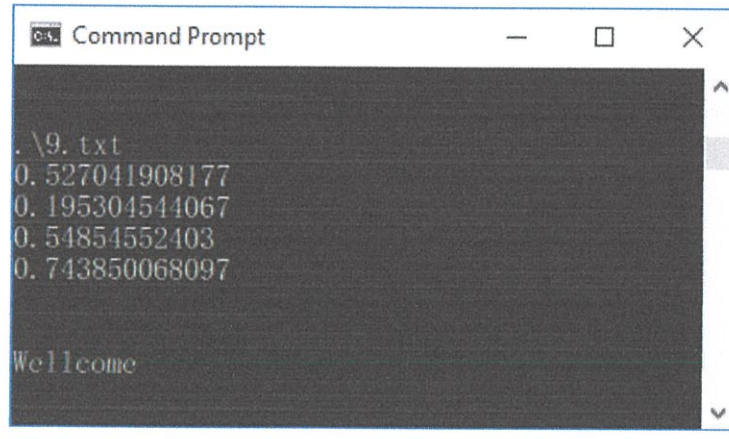
จากกราฟรูปที่ 4.13 เป็นกราฟเฉลี่ยเส้นเวลาจังหวะการพิมพ์ทั้งหมด 12 เส้น เพื่อหาเส้นเวลาจังหวะการพิมพ์ต้นแบบ นำไปเก็บไว้ในฐานข้อมูล เพื่อใช้ในการตรวจสอบเส้นเวลาจังหวะการพิมพ์ที่ต้องการตรวจสอบ

จากกราฟรูปที่ 4.14 แสดงให้เห็นว่าถ้าเป็นข้อมูลจังหวะการพิมพ์ของผู้ใช้งานจริง จะให้ค่าเส้นเวลาจังหวะการพิมพ์ที่มีความใกล้เคียงกับเส้นเวลาจังหวะการพิมพ์ต้นแบบดังเช่น เส้นกราฟ User1 (สีน้ำเงิน) ที่มีลักษณะคล้ายคลึงกันกับ เส้นกราฟ Original Master (สีเขียว) ซึ่งเป็นเส้นเวลาจังหวะการพิมพ์ต้นแบบ มากกว่า เส้นกราฟ User2 (สีแดง) ที่มีลักษณะต่างกันอย่างชัดเจน กับเส้นเวลาจังหวะการพิมพ์ต้นฉบับ

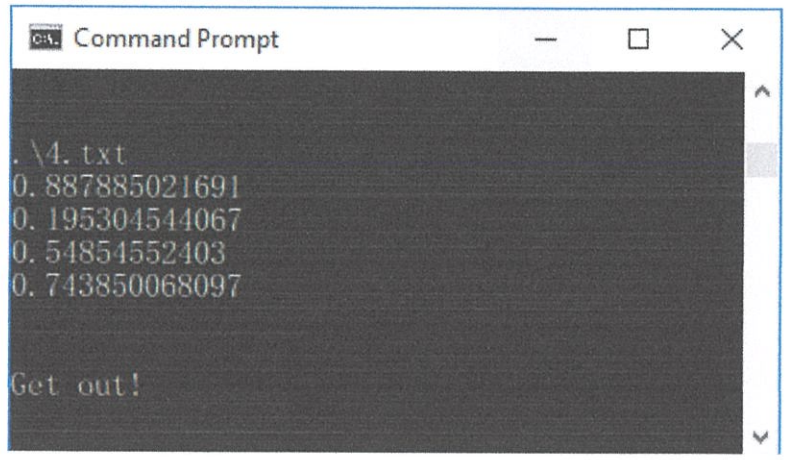
จากกราฟรูปที่ 4.15 แสดงวิธีการตรวจสอบว่าเส้นเวลาจังหวะการพิมพ์นั้นเป็นของผู้ใช้งานตัวจริงหรือไม่ โดยการหาค่าความแตกต่างระหว่างเส้นเวลาจังหวะการพิมพ์ ซึ่งถ้าหากเป็นเส้นเวลาจังหวะการพิมพ์ของผู้ใช้งานตัวจริง ค่าความแตกต่างที่ได้จะมีค่าน้อย และลำดับในการพิมพ์ถูกต้อง ระบบจะแสดงผลว่า “Welcome” ถ้าหากค่าความแตกต่างที่ได้มีค่าน้อย แต่ลำดับในการพิมพ์ไม่ถูกต้อง ระบบจะแสดงผลว่า “Character” และถ้าค่าความแตกต่างที่ได้มีค่ามาก ระบบจะแสดงผลว่า “Get out”

4.6 ผลการทดสอบการพิสูจน์ตัวตน

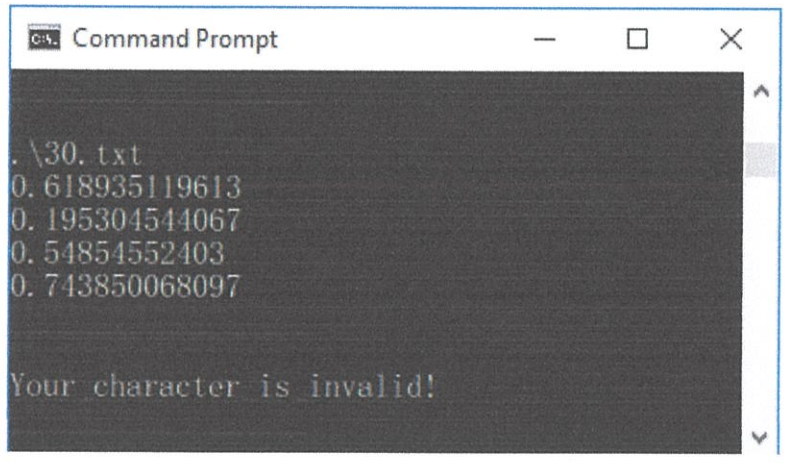
การทดสอบการพิสูจน์ตัวตน จะทำการทดสอบผ่านโค๊ดโปรแกรมที่เขียนขึ้น โดยการนำข้อมูลจังหวะการพิมพ์ที่ต้องการพิสูจน์มาทำการทดสอบ ซึ่งผลลัพธ์ของการทดสอบสามารถแยกออกได้เป็น 3 กรณี คือ กรณีที่ 1 ระบบอนุญาตให้เข้าสู่ระบบ จะแสดงผลว่า “Welcome”เป็นดังรูปที่ 4.16 กรณีที่ 2 ระบบไม่อนุญาตให้เข้าสู่ระบบ จะแสดงผลว่า “Get out” ดังรูปที่ 4.17 และกรณีที่ 3 ระบบตรวจสอบแล้วว่าจังหวะเวลาในการพิมพ์ถูกต้อง แต่ลำดับในการพิมพ์ไม่ถูกต้องตามฐานข้อมูล จะแสดงผลว่า “Your Character is invalid!” ดังรูปที่ 4.18



รูปที่ 4.16 ผลการทดสอบการพิสูจน์ตัวตน กรณีระบบอนุญาต



รูปที่ 4.17 ผลการทดสอบการพิสูจน์ตัวตน กรณีระบบไม่อนุญาต



รูปที่ 4.18 ผลการทดสอบการพิสูจน์ตัวตน
กรณีจังหวะการพิมพ์ถูก แต่ลำดับในการพิมพ์ไม่ถูกต้อง

ตารางที่ 4.1 ภาพรวมการทดสอบประสิทธิภาพของระบบการพิสูจน์ตัวตนในกลุ่มของคำง่าย

Word			Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
					Character Fail	Fail			Character Fail	Fail		
คำง่าย	ที่อยู่	3 ครั้ง	600	326	1	273	600	38	34	528	45.67	6.33
		6 ครั้ง	600	458	1	141	600	66	59	475	23.67	11.00
		9 ครั้ง	600	465	0	135	600	44	39	517	22.50	7.33
		12 ครั้ง	600	457	0	143	600	51	45	504	23.83	8.50
	บ่อยครั้ง	3 ครั้ง	600	241	0	359	600	8	0	592	59.83	1.33
		6 ครั้ง	600	467	0	133	600	88	0	512	22.17	14.67
		9 ครั้ง	600	495	0	105	600	89	0	511	17.50	14.83
		12 ครั้ง	600	503	0	97	600	88	0	512	16.17	14.67
	ซ้ำเติม	3 ครั้ง	600	163	23	414	600	18	18	564	72.83	3.00
		6 ครั้ง	600	399	1	200	600	62	35	503	33.50	10.33
		9 ครั้ง	600	474	1	125	600	78	72	450	21.00	13.00
		12 ครั้ง	600	488	1	111	600	80	75	445	18.67	13.33
ผลรวมของคำง่าย			7200	4936	28	2236	7200	710	377	6113	31.44	9.86
เปอร์เซ็นของคำง่าย				68.56	0.39	31.06		9.86	5.24	84.90		

ตารางที่ 4.2 ภาพรวมการทดสอบประสิทธิภาพของระบบการพิสูจน์ตัวตนในกลุ่มของคำยาว

Word			Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
					Character Fail	Fail			Character Fail	Fail		
คำยาว	พระจอมเกล้า	3 ครั้ง	600	239	0	361	600	15	0	585	60.17	2.50
		6 ครั้ง	600	420	0	180	600	69	0	531	30.00	11.50
		9 ครั้ง	600	479	0	121	600	86	0	514	20.17	14.33
		12 ครั้ง	600	498	0	102	600	112	0	488	17.00	18.67
	คอมพิวเตอร์	3 ครั้ง	600	258	0	342	600	23	10	567	57.00	3.83
		6 ครั้ง	600	401	0	199	600	54	26	520	33.17	9.00
		9 ครั้ง	600	452	0	148	600	55	42	503	24.67	9.17
		12 ครั้ง	600	478	0	122	600	57	49	494	20.33	9.50
	อนุญาต	3 ครั้ง	600	262	1	337	600	6	8	586	56.33	1.00
		6 ครั้ง	600	423	1	176	600	25	24	551	29.50	4.17
		9 ครั้ง	600	475	0	125	600	30	23	547	20.83	5.00
		12 ครั้ง	600	495	0	105	600	26	47	527	17.50	4.33
ผลรวมของคำยาว			7200	4880	2	2318	7200	558	229	6413	32.22	7.75
เปอร์เซ็นของคำยาว				67.78	0.03	32.19		7.75	3.18	89.07		

ตารางที่ 4.3 ภาพรวมการทดสอบประสิทธิภาพของระบบการพิสูจน์ตัวตนในกลุ่มของคำยาก

Word			Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
					Character Fail	Fail			Character Fail	Fail		
คำยาก	ศรัทธา	3 ครั้ง	600	239	0	361	600	29	11	560	60.17	4.83
		6 ครั้ง	600	438	0	162	600	42	63	495	27.00	7.00
		9 ครั้ง	600	454	0	146	600	29	20	551	24.33	4.83
		12 ครั้ง	600	488	0	112	600	43	58	499	18.67	7.17
	สาบสูญ	3 ครั้ง	600	229	0	371	600	1	6	593	61.83	0.17
		6 ครั้ง	600	355	0	245	600	11	6	583	40.83	1.83
		9 ครั้ง	600	432	0	168	600	17	27	556	28.00	2.83
		12 ครั้ง	600	461	3	136	600	24	47	529	23.17	4.00
	ปฐมนิเทศ	3 ครั้ง	600	243	0	357	600	1	7	592	59.50	0.17
		6 ครั้ง	600	413	0	187	600	18	27	555	31.17	3.00
		9 ครั้ง	600	439	1	160	600	17	31	552	26.83	2.83
		12 ครั้ง	600	473	1	126	600	39	25	536	21.17	6.50
ผลรวมของคำยาก			7200	4664	5	2531	7200	271	328	6601	35.22	3.76
เปอร์เซ็นของคำยาก				64.78	0.07	35.15		3.76	4.56	91.68		

ตารางที่ 4.4 ภาพรวมการทดสอบประสิทธิภาพของระบบการพิสูจน์ตัวตนของทั้งระบบ

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
			Character Fail	Fail			Character Fail	Fail		
ผลรวมของระบบ	21600	14480	35	7085	21600	1539	934	19127	32.96	7.13
เปอร์เซ็นของระบบ		67.04	0.16	32.80		7.13	4.32	88.55		

จากตารางที่ 4.1 ภาพรวมการทดสอบประสิทธิภาพของระบบการพิสูจน์ตัวตนในกลุ่มของคำง่าย ของผู้ใช้งานจำนวน 20 คน ทั้งหมด 7200 ครั้ง จากจำนวน 3 คำ โดยการแบ่งการเก็บข้อมูลลงฐานข้อมูลเป็นการเก็บจำนวน 3 ครั้ง 6 ครั้ง 9 ครั้ง และ 12 ครั้ง ซึ่งเมื่อทำการทดสอบประสิทธิภาพของระบบพบว่า เมื่อข้อมูลจังหวะการพิมพ์ที่เข้ามาเป็นของผู้ใช้งานจริง ระบบอนุญาต 68.56% อักขระการพิมพ์ไม่ถูกต้อง 0.39% ระบบไม่อนุญาต 32.19% และเมื่อข้อมูลจังหวะการพิมพ์ที่เข้ามาเป็นของผู้บุกรุก ระบบอนุญาต 9.86% อักขระการพิมพ์ไม่ถูกต้อง 5.24% ระบบไม่อนุญาต 84.90% และมีค่าอัตราการปฏิเสธที่ผิดพลาด (FRR) 31.44% กล่าวคือผู้ใช้งานจริงไม่สามารถเข้าสู่ระบบได้ 31.44% และอัตราการยอมรับที่ผิดพลาด (FAR) 9.86% กล่าวคือผู้บุกรุกสามารถเข้าสู่ระบบได้ 9.86%

จากตารางที่ 4.2 ภาพรวมการทดสอบประสิทธิภาพของระบบการพิสูจน์ตัวตนในกลุ่มของคำยาว ของผู้ใช้งานจำนวน 20 คน ทั้งหมด 7200 ครั้ง จากจำนวน 3 คำ โดยการแบ่งการเก็บข้อมูลลงฐานข้อมูลเป็นการเก็บจำนวน 3 ครั้ง 6 ครั้ง 9 ครั้ง และ 12 ครั้ง ซึ่งเมื่อทำการทดสอบประสิทธิภาพของระบบพบว่า เมื่อข้อมูลจังหวะการพิมพ์ที่เข้ามาเป็นของผู้ใช้งานจริง ระบบอนุญาต 67.78% อักขระการพิมพ์ไม่ถูกต้อง 0.03% ระบบไม่อนุญาต 31.06% และเมื่อข้อมูลจังหวะการพิมพ์ที่เข้ามาเป็นของผู้บุกรุก ระบบอนุญาต 7.75% อักขระการพิมพ์ไม่ถูกต้อง 3.18% ระบบไม่อนุญาต 89.07% และมีค่าอัตราการปฏิเสธที่ผิดพลาด (FRR) 32.22% กล่าวคือผู้ใช้งานจริงไม่สามารถเข้าสู่ระบบได้ 32.22% และอัตราการยอมรับที่ผิดพลาด (FAR) 7.75% กล่าวคือผู้บุกรุกสามารถเข้าสู่ระบบได้ 7.75%

จากตารางที่ 4.3 ภาพรวมการทดสอบประสิทธิภาพของระบบการพิสูจน์ตัวตนในกลุ่มของคำยาก ของผู้ใช้งานจำนวน 20 คน ทั้งหมด 7200 ครั้ง จากจำนวน 3 คำ โดยการแบ่งการเก็บข้อมูลลงฐานข้อมูลเป็นการเก็บจำนวน 3 ครั้ง 6 ครั้ง 9 ครั้ง และ 12 ครั้ง ซึ่งเมื่อทำการทดสอบประสิทธิภาพของระบบพบว่า เมื่อข้อมูลจังหวะการพิมพ์ที่เข้ามาเป็นของผู้ใช้งานจริง ระบบอนุญาต 64.78% อักขระการพิมพ์ไม่ถูกต้อง 0.07% ระบบไม่อนุญาต 35.15% และเมื่อข้อมูลจังหวะการพิมพ์ที่เข้ามาเป็นของผู้บุกรุก ระบบอนุญาต 3.76% อักขระการพิมพ์ไม่ถูกต้อง 4.56% ระบบไม่อนุญาต 91.68% และมีค่าอัตราการปฏิเสธที่ผิดพลาด (FRR) 35.22% กล่าวคือผู้ใช้งานจริงไม่สามารถเข้าสู่ระบบได้ 35.22% และอัตราการยอมรับที่ผิดพลาด (FAR) 3.76% กล่าวคือผู้บุกรุกสามารถเข้าสู่ระบบได้ 3.76%

จากตารางที่ 4.4 ภาพรวมการทดสอบประสิทธิภาพของระบบการพิสูจน์ตัวตนของทั้งระบบของผู้ใช้งานจำนวน 20 คน ทั้งหมด 21600 ครั้ง จากจำนวน 3 กลุ่ม กลุ่มละ 3 คำ โดยการแบ่งการเก็บข้อมูลลงฐานข้อมูลเป็นการเก็บจำนวน 3 ครั้ง 6 ครั้ง 9 ครั้ง และ 12 ครั้ง ซึ่งเมื่อทำการทดสอบประสิทธิภาพของระบบพบว่า เมื่อข้อมูลจังหวะการพิมพ์ที่เข้ามาเป็นของผู้ใช้งานจริง ระบบอนุญาต 67.04% อักขระการพิมพ์ไม่ถูกต้อง 0.16% ระบบไม่อนุญาต 32.80% และเมื่อข้อมูลจังหวะการพิมพ์ที่เข้ามาเป็นของผู้บุกรุก ระบบอนุญาต 7.13% อักขระการพิมพ์ไม่ถูกต้อง 4.32% ระบบไม่อนุญาต 88.55% และมีค่าอัตราการปฏิเสธที่ผิดพลาด (FRR) 32.96% กล่าวคือผู้ใช้งานจริงไม่สามารถเข้าสู่ระบบได้ 32.96% และอัตราการยอมรับที่ผิดพลาด (FAR) 7.13% กล่าวคือผู้บุกรุกสามารถเข้าสู่ระบบได้ 7.13%

ตารางที่ 4.5 ประสิทธิภาพของระบบพิสูจน์ตัวตนของกลุ่มคำง่าย

	Condition positive	Condition negative	
Test outcome positive	True positive (TP) = 68.56	Fale positive (FP) = 9.86	Positive predictive value 87.43%
Test outcome negative	False negative (FN) = 31.44	True negative (TN) = 90.14	Negative predictive value 74.14%
	Sensitivity 68.56%	Specificity 90.14%	

จากตารางที่ 4.5 สามารถแสดงค่าประสิทธิภาพของระบบพิสูจน์ตัวตนของกลุ่มคำง่าย โดยค่าความไว (Sensitivity) บ่งบอกว่าระบบสามารถจำแนกผู้ใช้งานจริงได้ดีเพียงใด โดยระบบมีความไว (Sensitivity) 68.56% ซึ่งอยู่ในเกณฑ์ที่พอใช้ ค่าความจำเพาะ (Specificity) บ่งบอกการจำแนกผู้บุกรุกได้ดีเพียงใด ระบบมีค่าความจำเพาะ (Specificity) 90.14% ถือว่าอยู่ในเกณฑ์ที่ดีมาก ค่าทำนายผลบวกบ่งบอกผลบอกที่ได้จากระบบนั้นแสดงว่าเป็นผู้ใช้งานจริงๆ ได้ดีเพียงใด ในที่นี้คือ 87.43% ซึ่งถือว่าอยู่ในเกณฑ์ที่ดีมาก และค่าทำนายผลลบบ่งบอกว่าผลลบที่ได้จากระบบนั้นแสดงว่าเป็นผู้บุกรุกจริงๆ ได้เพียงใด ค่าทำนายผลลบอยู่ที่ 74.14% ถือว่าระบบอยู่ในระดับที่ดี

ตารางที่ 4.6 ประสิทธิภาพของระบบพิสูจน์ตัวตนของกลุ่มคำยาว

	Condition positive	Condition negative	
Test outcome positive	True positive (TP) = 67.78	Fale positive (FP) = 7.75	Positive predictive value 89.74%
Test outcome negative	False negative (FN) = 32.22	True negative (TN) = 92.25	Negative predictive value 74.11%
	Sensitivity 67.78%	Specificity 92.25%	

จากตารางที่ 4.6 สามารถแสดงค่าประสิทธิภาพของระบบพิสูจน์ตัวตนของกลุ่มคำยาวโดยค่าความไว (Sensitivity) บ่งบอกว่าระบบสามารถจำแนกผู้ใช้งานจริงได้ดีเพียงใด โดยระบบมีความไว (Sensitivity) 67.78% ซึ่งอยู่ในเกณฑ์ที่พอใช้ ค่าความจำเพาะ (Specificity) บ่งบอกการจำแนกผู้บุกรุกได้ดีเพียงใด ระบบมีค่าความจำเพาะ (Specificity) 92.25% ถือว่าอยู่ในเกณฑ์ที่ดีมาก ค่าทำนายผลบวกบ่งบอกผลบอกที่ได้จากระบบนั้นแสดงว่าเป็นผู้ใช้งานจริงๆ ได้ดีเพียงใด ในที่นี้คือ 89.74% ซึ่งถือว่าอยู่ในเกณฑ์ที่ดีมาก และค่าทำนายผลลบบ่งบอกว่าผลลบที่ได้จากระบบนั้นแสดงว่าเป็นผู้บุกรุกจริงๆ ได้เพียงใด ค่าทำนายผลลบอยู่ที่ 74.11% ถือว่าระบบอยู่ในระดับที่ดี

ตารางที่ 4.7 ประสิทธิภาพของระบบพิสูจน์ตัวตนของกลุ่มคำยาก

	Condition positive	Condition negative	
Test outcome positive	True positive (TP) = 64.78	Fale positive (FP) = 3.76	Positive predictive value 94.51%
Test outcome negative	False negative (FN) = 35.22	True negative (TN) = 96.24	Negative predictive value 73.21%
	Sensitivity 64.78%	Specificity 96.24%	

จากตารางที่ 4.7 สามารถแสดงค่าประสิทธิภาพของระบบพิสูจน์ตัวตนของกลุ่มคำยาก โดยค่าความไว (Sensitivity) บ่งบอกว่าระบบสามารถจำแนกผู้ใช้งานจริงได้ดีเพียงใด โดยระบบมีความไว (Sensitivity) 64.78% ซึ่งอยู่ในเกณฑ์ที่พอใช้ ค่าความจำเพาะ (Specificity) บ่งบอกการจำแนกผู้บุกรุกได้ดีเพียงใด ระบบมีค่าความจำเพาะ (Specificity) 96.24% ถือว่าอยู่ในเกณฑ์ที่ดีมาก ค่าทำนายผลบวกบ่งบอกผลบอกที่ได้จากระบบนั้นแสดงว่าเป็นผู้ใช้งานจริงๆ ได้ดีเพียงใด ในที่นี้คือ 94.51% ซึ่งถือว่าอยู่ในเกณฑ์ที่ดีมาก และค่าทำนายผลลบบ่งบอกว่าผลลบที่ได้จากระบบนั้นแสดงว่าเป็นผู้บุกรุกจริงๆ ได้เพียงใด ค่าทำนายผลลบอยู่ที่ 73.21% ถือว่าระบบอยู่ในระดับที่ดี

ตารางที่ 4.8 ผลรวมประสิทธิภาพของระบบพิสูจน์ตัวตนทุกคำของทั้งระบบ

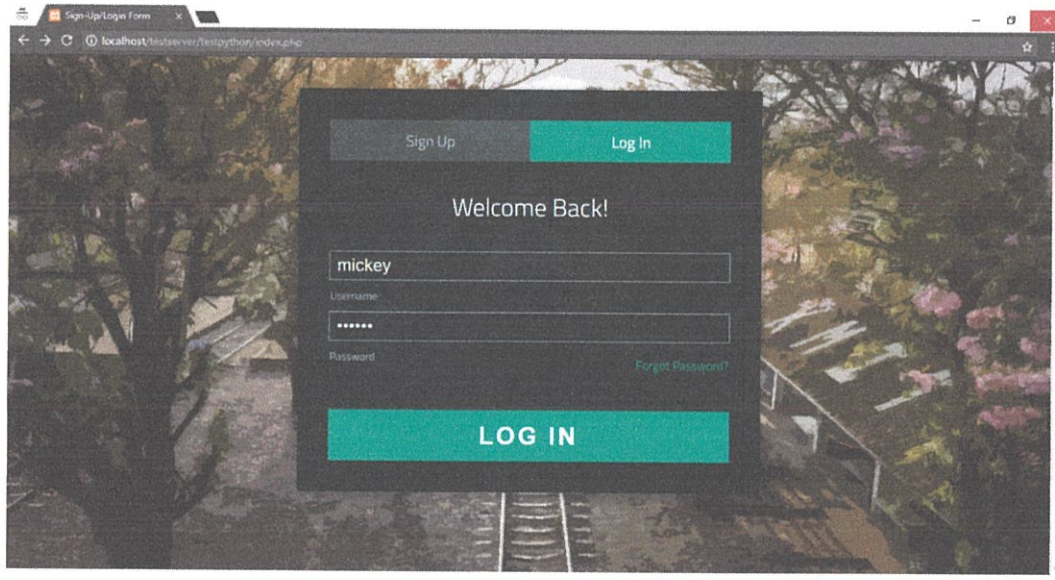
	Condition positive	Condition negative	
Test outcome positive	True positive (TP) = 67.04	Fale positive (FP) = 7.13	Positive predictive value 90.39%
Test outcome negative	False negative (FN) = 32.96	True negative (TN) = 92.87	Negative predictive value 73.81%
	Sensitivity 67.04%	Specificity 92.87%	

จากตารางที่ 4.8 สามารถแสดงค่าประสิทธิภาพของระบบพิสูจน์ตัวตนของทั้งระบบ โดยค่าความไว (Sensitivity) บ่งบอกว่าระบบสามารถจำแนกผู้ใช้งานจริงได้ดีเพียงใด โดยระบบมีความไว (Sensitivity) 67.04% ซึ่งอยู่ในเกณฑ์ที่พอใช้ ค่าความจำเพาะ (Specificity) บ่งบอกการจำแนกผู้บุกรุกได้ดีเพียงใด ระบบมีค่าความจำเพาะ (Specificity) 92.87% ถือว่าอยู่ในเกณฑ์ที่ดีมาก ค่าทำนายผลบวกบ่งบอกผลบอกที่ได้จากระบบนั้นแสดงว่าเป็นผู้ใช้งานจริงๆ ได้ดีเพียงใด ในที่นี้คือ 90.39% ซึ่งถือว่าอยู่ในเกณฑ์ที่ดีมาก และค่าทำนายผลลบบ่งบอกว่าผลลบที่ได้จากระบบนั้นแสดงว่าเป็นผู้บุกรุกจริงๆ ได้เพียงใด ค่าทำนายผลลบอยู่ที่ 73.81% ถือว่าระบบอยู่ในระดับที่ดี

4.7 ผลการทดสอบใช้งานเว็บไซต์ล็อกอินต้นแบบสำหรับระบบการพิสูจน์ตัวตนด้วยจังหวะการพิมพ์

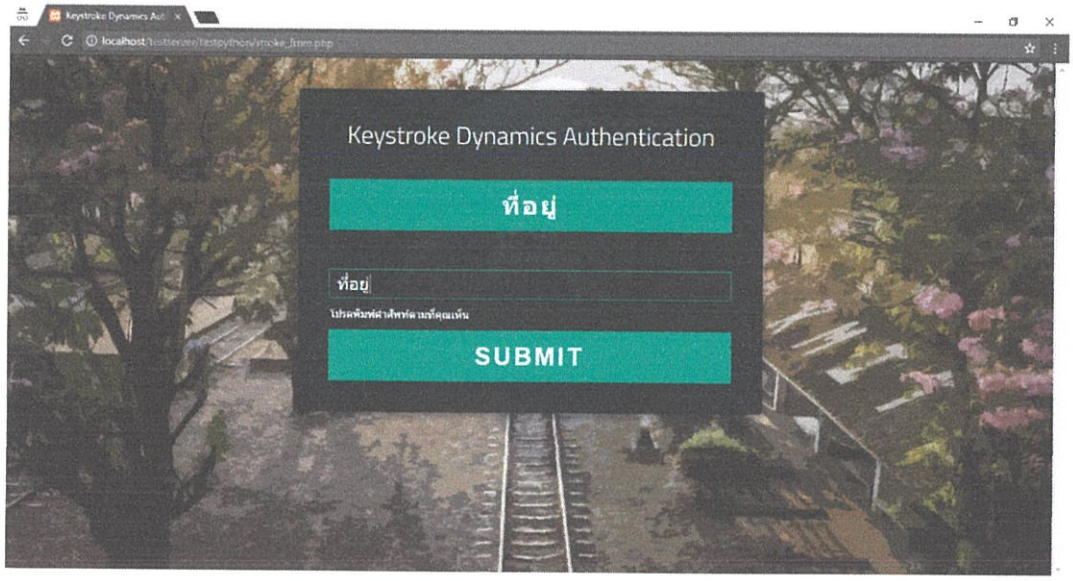
4.7.1 ทดสอบการล็อกอินเข้าใช้งาน

ขั้นตอนนี้จะทำการทดสอบโดยการใส่ username และ password และทำการกดเข้าสู่ระบบ ดังรูปที่ 4.19 เพื่อทำการตรวจสอบกับฐานข้อมูลว่าเข้ามูลที่เข้ามานั้น ตรงกับที่เก็บไว้ในฐานข้อมูลหรือไม่ ถ้าข้อมูลถูกต้องระบบจะส่งต่อข้อมูลไปยังขั้นตอนถัดไป

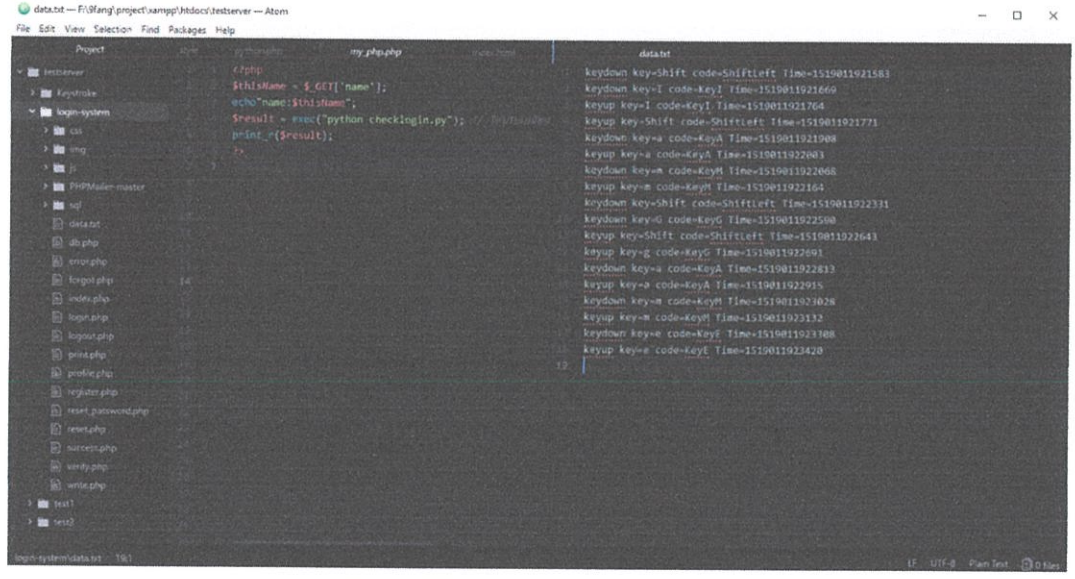


รูปที่ 4.19 ทดสอบการเข้าสู่ระบบ

เมื่อระบบตรวจสอบข้อมูล username และ password เรียบร้อยแล้ว ขั้นตอนถัดมาระบบจะทำการให้ผู้ใช้งานพิมพ์คำภาษาไทยที่ถูกสุ่มออกมาจากฐานข้อมูลของ username นั้นๆ เพื่อทำการพิสูจน์จังหวะการพิมพ์ว่าใช่ของผู้ใช่คนนั้นจริงๆ หรือไม่ ดังรูปที่ 4.20

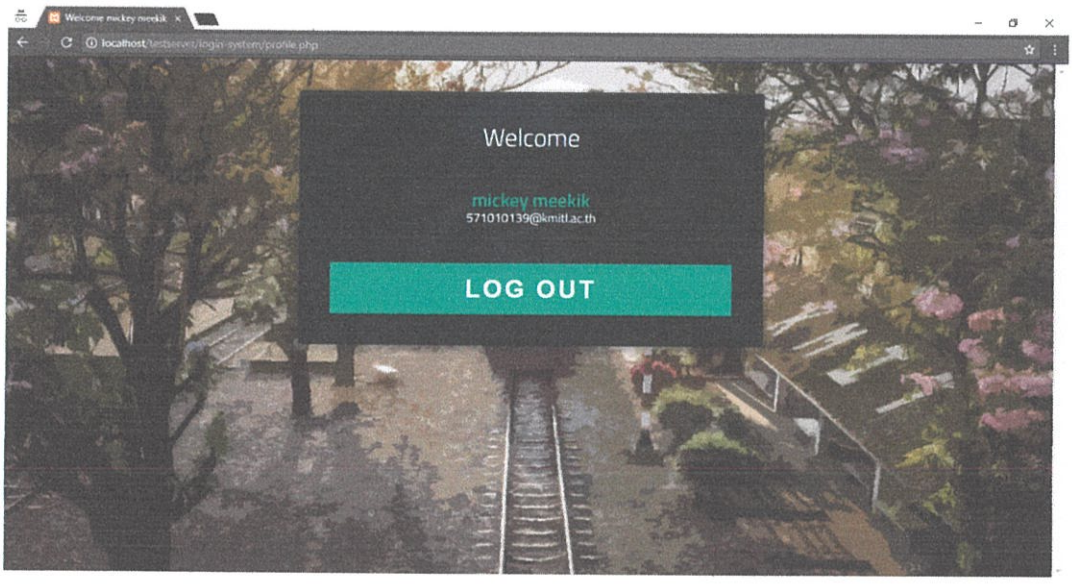


รูปที่ 4.20 ทดสอบการพิมพ์คำภาษาไทยเพื่อพิสูจน์จังหวะการพิมพ์

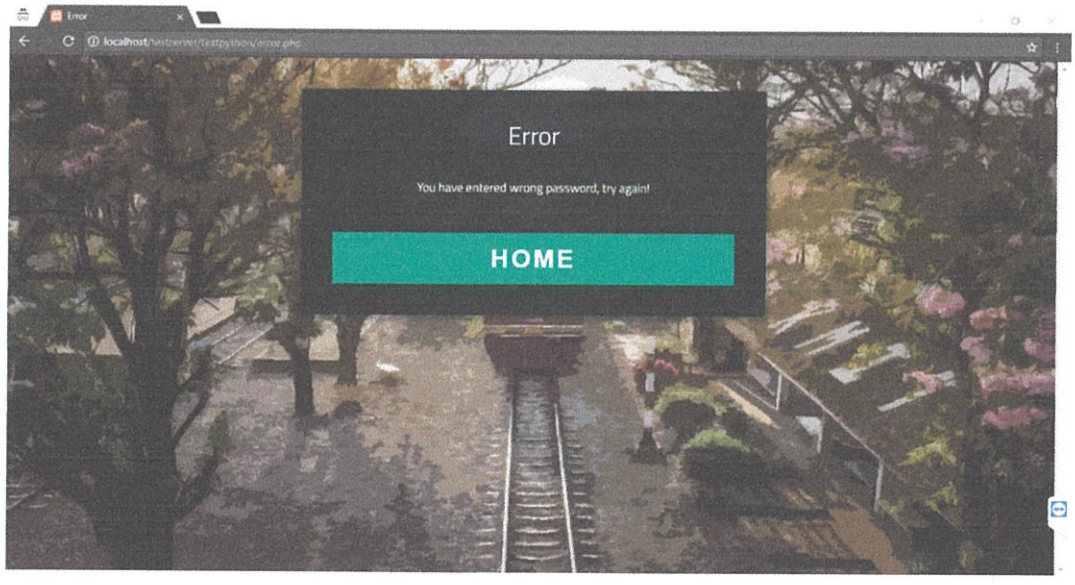


รูปที่ 4.21 จังหวะการพิมพ์ที่ถูกส่งมายังเครื่องเซิร์ฟเวอร์

ในขณะที่พิมพ์จังหวะการพิมพ์ของผู้ใช้งานจะถูกส่งค่ามายังเซิร์ฟเวอร์ ดังรูปที่ 4.21 เมื่อระบบประมวลผลแล้วพบว่าเป็นผู้ใช้งานจริงระบบก็จะยินยอมให้เข้าสู่ระบบได้ ดังรูปที่ 4.22 แต่ถ้าพบว่าไม่ใช่ผู้ใช้งานจริง ระบบจะไม่ยินยอมให้เข้าสู่ระบบ ดังรูปที่ 4.23



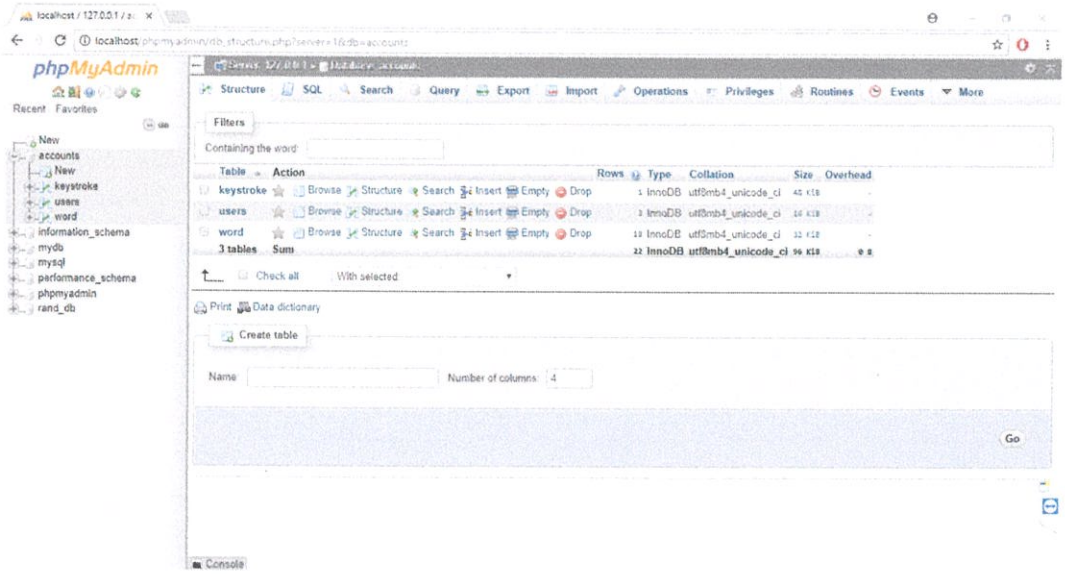
รูปที่ 4.22 ระบบตรวจสอบว่าเป็นผู้ใช้งานจริง



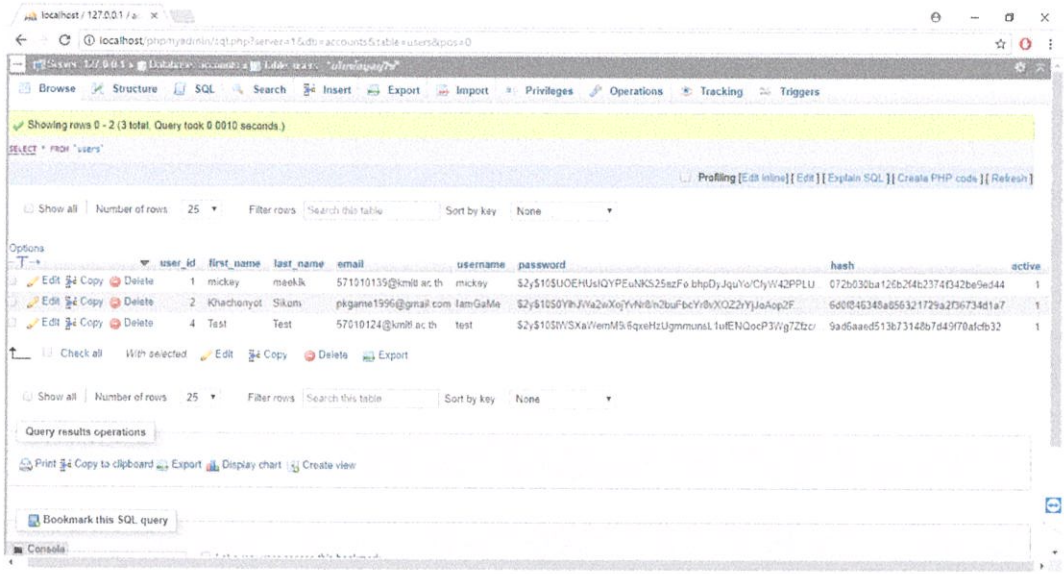
รูปที่ 4.23 ระบบตรวจสอบว่าไม่ใช่ผู้ใช้งานจริง

4.7.2 การสร้างฐานข้อมูล

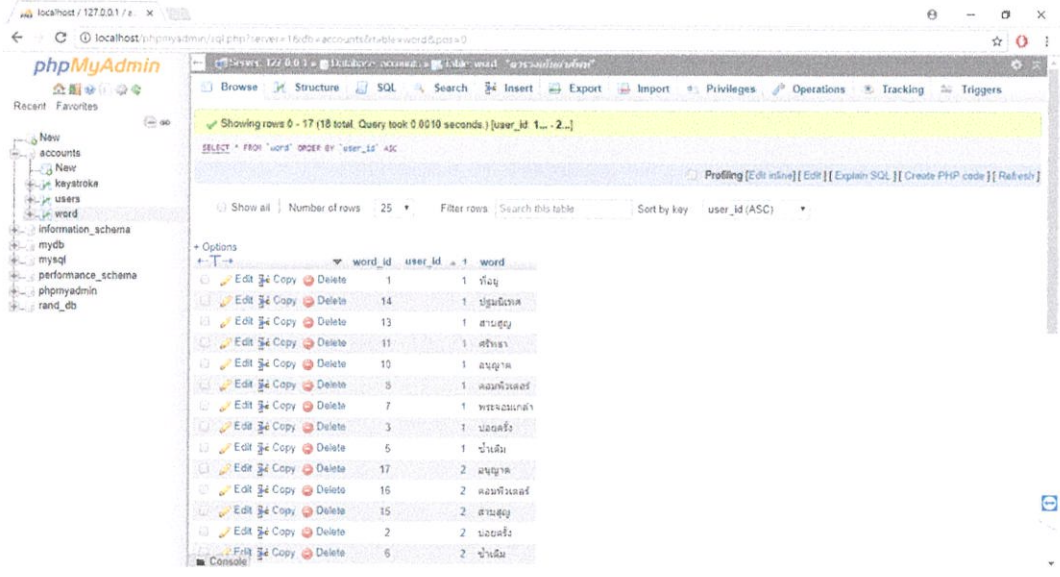
ในปริญญานิพนธ์นี้จะสร้างฐานข้อมูลโดยใช้ phpMyAdmin ซึ่งเป็นโปรแกรมที่ใช้ในการจัดการฐานข้อมูล MySQL โดยฐานข้อมูลที่ผู้จัดทำได้สร้างขึ้นแสดงดังรูปที่ 4.24 จะเห็นว่าในการสร้างฐานข้อมูลนี้จะประกอบด้วยตาราง 3 ตารางคือ ตาราง users ตาราง word และตาราง keystroke โดยตาราง users จะมีไว้เพื่อทำการเก็บข้อมูลผู้ใช้งานที่ทำการสมัครสมาชิกไว้ แสดงดังรูปที่ 4.25 ตาราง word จะมีไว้เก็บคำภาษาไทยต่างๆ ที่ผู้ใช้งานได้สร้างไว้ เพื่อจะใช้ในการสุ่มออกมาเพื่อทำการพิสูจน์จังหวะการพิมพ์ แสดงดังรูปที่ 4.26 ส่วนตาราง keystroke จะมีไว้เพื่อเก็บจังหวะการพิมพ์ของคำต่างๆ แสดงดังรูปที่ 4.27



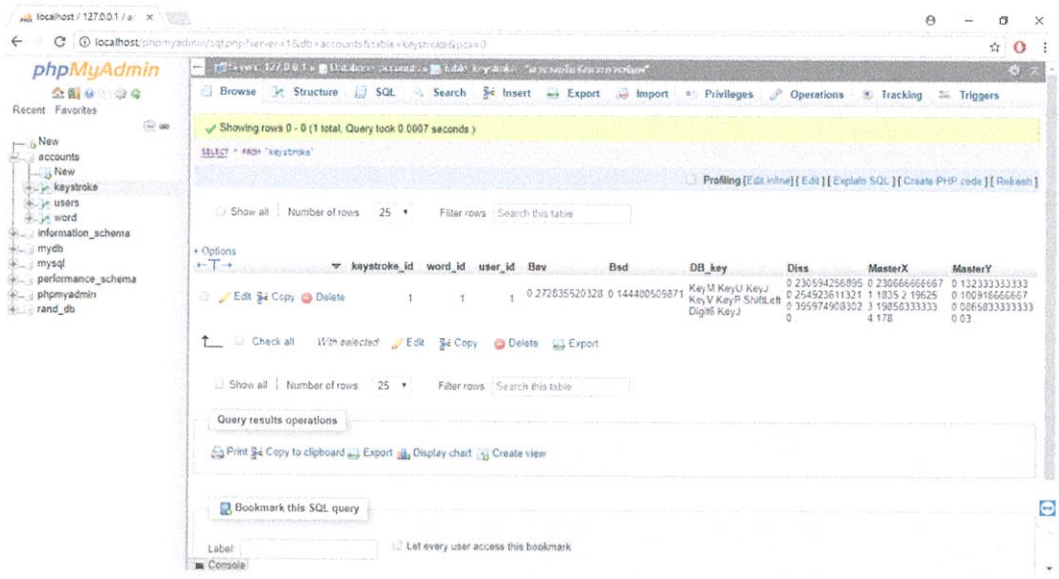
รูปที่ 4.24 ฐานข้อมูลที่สร้างขึ้น



รูปที่ 4.25 ตารางสำหรับบันทึกข้อมูลผู้ใช้งาน



รูปที่ 4.26 ตารางสำหรับบันทึกคำภาษาไทยที่ผู้ใช้งานสร้างขึ้น



รูปที่ 4.27 ตารางสำหรับบันทึกจังหวะการพิมพ์

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

ปริญญานิพนธ์นำเสนอการระบุตัวตนผ่านจังหวะการพิมพ์ สำหรับการเข้าถึงระบบที่ต้องมีการล็อกอินเข้าใช้งานบนคอมพิวเตอร์ ที่มีการพิมพ์ด้วยคีย์บอร์ด โดยการประยุกต์ใช้หลักการหาระยะทางแบบยูคลิด เริ่มจากการนำ 2 Features ของ Keystroke Dynamics Authentication ซึ่งได้แก่ Interkey Time หมายถึง ช่วงเวลาที่เปลี่ยนคีย์ใดไปสู่อีกคีย์หนึ่ง ซึ่งได้เป็นตัวเลขค่าบวกหรือค่าลบ และ Latency Time หมายถึง ช่วงเวลาที่เริ่มกดคีย์จนกระทั่งกดคีย์ถัดไป หรือช่วงเวลาที่เริ่มจากปล่อยคีย์จนกระทั่งปล่อยคีย์ถัดไป มาสร้างเส้นต้นฉบับของผู้ใช้งานตัวจริง และใช้หลักการหาระยะทางแบบยูคลิด มาวัดหาค่าความแตกต่างระหว่างผู้ใช้งานตัวจริง กับเส้นที่ต้องการทดสอบผนวกกับการเช็คลำดับในการกดแป้นพิมพ์ของคำภาษาไทย โดยผลการทดสอบประสิทธิภาพของระบบ มีค่าความไวซึ่งบ่งบอกถึงการจำแนกผู้ใช้งานจริงได้ดีเพียงใด อยู่ที่ 67.04% และค่าความจำเพาะซึ่งบ่งบอกถึงการจำแนกผู้บุกรุกได้ดีเพียงใดอยู่ที่ 92.87% ค่าทำนายผลบวกบ่งบอกถึง ระบบแสดงว่าเป็นผู้ใช้งานจริงได้ดีเพียงใด มีค่าอยู่ที่ 90.39% และค่าทำนายผลลบบ่งบอกถึง ระบบแสดงว่าเป็นผู้บุกรุกได้ดีเพียงใด มีค่าอยู่ที่ 73.81% นอกจากนี้ระบบยังมีค่าอัตราการปฏิเสธที่ผิดพลาด (FRR) 32.96% ซึ่งหมายถึงว่า ผู้ใช้งานจริงจะไม่สามารถเข้าสู่ระบบได้ 32.96% และอัตราการยอมรับที่ผิดพลาด (FAR) 7.13% หมายถึงว่า ผู้บุกรุกสามารถเข้าสู่ระบบได้เพียง 7.13% เท่านั้น ถือว่าระบบมีประสิทธิภาพการทำงานอยู่ในระดับที่ดี

ซึ่งปริญญานิพนธ์นี้ได้แสดงให้เห็นถึงแนวทาง และประโยชน์ของการนำพฤติกรรมจังหวะการพิมพ์ของมนุษย์ในการพิมพ์คำภาษาไทยมาพัฒนา และเพิ่มประสิทธิภาพในการพิสูจน์ตัวตนผ่านจังหวะการพิมพ์บนระบบคอมพิวเตอร์

5.2 ข้อเสนอแนะ

5.2.1 จังหวะการพิมพ์ของแต่ละบุคคล เป็นลักษณะทางพฤติกรรมของมนุษย์ ซึ่งเป็นลักษณะที่มีการเปลี่ยนแปลงหรือพัฒนาได้ตลอดเวลา ในการเก็บข้อมูลการพิมพ์จะต้องไม่พิมพ์ต่อเนื่องกัน ควรเว้นระยะห่างในการเก็บข้อมูลการพิมพ์

5.2.2 ตัวอย่างคำที่นำมาใช้ให้ผู้ทดลองพิมพ์ ควรเป็นคำที่ผู้ทดลองได้พิมพ์อยู่เป็นประจำเพื่อศึกษาถึงธรรมชาติของผู้ทดลอง

บรรณานุกรม

[1] สำนักงานคณะกรรมการกำกับกิจการพลังงาน . "ไบโอเมตริก (Biometric)."
http://www.erc.or.th/OERCWeb/Front/Article/ArticleDetail.aspx?Type=1&CatId=6&rid=4&muid=0&prid=0.

[2] NovaBizz. "ไบโอเมตริก (Biometric)."
https://www.novabizz.com/CDC/System/Biometrics.htm.

[3] เกษม หวังสุข และ ธนภัทร์ อนุศาสน์อมรกุล. "*การพิสูจน์ตัวตนโดยจังหวะการพิมพ์ด้วยวิธีการวัดค่าความต่างของเส้นโคจร.*" วิทยานิพนธ์ปริญญาวิทยาศาสตร์มหาบัณฑิต, ภาควิชาวิทยาการคอมพิวเตอร์และสารสนเทศ คณะวิทยาศาสตร์ประยุกต์, มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2557.

[4] ธีระพงษ์ กระการดี. "ค่าเฉลี่ยเลขคณิต."
http://www.stvc.ac.th/elearning/stat/csu2.html.

[5] ธีระพงษ์ กระการดี. "ส่วนเบี่ยงเบนมาตรฐาน."
http://www.stvc.ac.th/elearning/stat/csu3.html.

[6] Wikipedia. "ระยะทางแบบยูคลิด."
https://th.wikipedia.org/wiki/%E0%B8%A3%E0%B8%B0%E0%B8%A2%E0%B8%B0%E0%B8%97%E0%B8%B2%E0%B8%87%E0%B9%81%E0%B8%9A%E0%B8%9A%E0%B8%A2%E0%B8%B8%E0%B8%84%E0%B8%A5%E0%B8%B4%E0%B8%94.

[7] Paul Fearnhead. "*Biometrika*", Oxford University Press on behalf of the Biometrika Trust,1991

[8] Wikipedia. "ความไวและความจำเพาะ."
https://th.wikipedia.org/wiki/%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1%E0%B9%84%E0%B8%A7%E0%B9%81%E0%B8%A5%E0%B8%B0%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1%E0%B8%88%E0%B8%B3%E0%B9%80%E0%B8%9E%E0%B8%B2%E0%B8%B0.

[9] Mindphp. "Javascript."
http://www.mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A1%E0%B8%B7%E0%B8%AD/73-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/2187-java-javascript%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3.html.

[10] Ikamiso Phyblas. "ประวัติไพธอน ."
https://phyblas.hinaboshi.com/tsuchinoko01.
[11] ทวีรัตน์ นวลช่วย. "จุดเด่นของภาษาไพธอน."
https://sites.google.com/site/dotpython/installation.
[12] พิชิต วิจิตรบุญยรักษ์. "HTML : ภาษาเขียนเว็บ." , มหาวิทยาลัยกรุงเทพ
[13] Mindphp. "PHP."
http://www.mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A 1%E0%B8%B7%E0%B8%AD/73-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/2127-php-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3.html.
[14] JustUsers.net. "การพัฒนาPHP."
http://www.justusers.net/forum/index.php?topic=2828.0.
[15] Mindphp. "phpMyAdmin."
http://www.mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A1%E0%B8%B7%E0%B8%AD/73-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/2285-phpmyadmin-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3.html.
[16] Easy Branches. "MySQL."
http://www.th.easyhostdomain.com/dedicated-servers/mysql.html.
[17] Neoxteen. "คำสั่งใน MySQL."
http://www.neoxteen.com/knowledge/PHP/%E0%B8%A3%E0%B8%A7%E0%B8%A1%E0%B8%84%E0%B8%B3%E0%B8%AA%E0%B8%B1%E0%B9%88%E0%B8%87MySQL%E0%B8%97%E0%B8%B5%E0%B9%88%E0%B9%83%E0%B8%8A%E0%B9%89%E0%B8%87%E0%B8%B2%E0%B8%99%E0%B8%9A%E0%B9%88%E0%B8%AD%E0%B8%A2%E0%B9%86%E0%B8%AA%E0%B8%B3%E0%B8%AB%E0%B8%A3%E0%B8%B1%E0%B8%9A-PHP.

ภาคผนวก ก

โค๊ดเว็บไซต์เก็บค่าจังหวะการพิมพ์

ไฟล์เว็บไซต์

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<link rel="stylesheet" href="style.css">
</head>
<body>
<form id="form" onsubmit="return false">
<input type="text" placeholder="Press keys here" id="kinput">
<textarea id="area"></textarea>
<input type="button" value="Clear" onclick="area.value = ""/>
</form>
<div>
 <tr>
      <td>Filename to Save As:</td>
      <td><input id="inputFileNameToSaveAs"></input></td>
      <td><button onclick="saveTextAsFile()">Save Text to File</button></td>
 </tr>
</div>
<script src="script.js"></script>
</body>
</html>
```

ไฟล์ script.js

```
kinput.onkeydown=kinput.onkeyup=handle;
let lastTime=Date.now();
function handle(e){
let k = new Date();
let charCode = e.which || e.keyCode;
let text=e.type+' key='+ e.key+' code='+ e.code+
                (e.shiftKey?':')+
                (e.ctrlKey?':ctrlKey':')+
                (e.altKey?':altKey':')+
                (e.metaKey?':metaKey':')+
                (e.repeat?':(repeat)':') + ' Time='+ k.getTime()+ "\n";
```

```
 if(area.value&&Date.now()- lastTime>1000){
          area.value+=new Array(81).join('-')+'\n';
 }
 lastTime=Date.now();
 area.value+=text;
 if(charCode==8){
        alert("You pressed the Backspace key!");
        area.value = '';
                    kinput.value = '';
 }
 var xmlhttp = new XMLHttpRequest();
  xmlhttp.onreadystatechange = function() {
          if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {
              document.getElementById("demo").innerHTML = xmlhttp.responseText;
           }
        };
        xmlhttp.open("GET", "print.php?text=" + text, true);
        xmlhttp.send();
}

function saveTextAsFile()
{
    var textToSave = document.getElementById("area").value;
    var textToSaveAsBlob = new Blob([textToSave], {type:"text/plain"});
    var textToSaveAsURL = window.URL.createObjectURL(textToSaveAsBlob);
    var fileNameToSaveAs = document.getElementById
("inputFileNameToSaveAs").value;
    var downloadLink = document.createElement("a");
    downloadLink.download = fileNameToSaveAs;
    downloadLink.innerHTML = "Download File";
    downloadLink.href = textToSaveAsURL;
    downloadLink.onclick = destroyClickedElement;
    downloadLink.style.display = "none";
    document.body.appendChild(downloadLink);
```

```
    downloadLink.click();
}

function destroyClickedElement(event)
{
    document.body.removeChild(event.target);
}
```

ไฟล์ style.css

```
#kinput{font-size:150%;box-sizing:border-box;width:95%;}
#area{width:95%;box-sizing:border-box;height:250px;border:1px solid
black;display:block;}
form label{display:inline;white-space:nowrap;}
```

ภาคผนวก ข

โค๊ดโปรแกรมคำนวณหาค่าจังหวะการพิมพ์เพื่อจัดเก็บลงฐานข้อมูล

```
def callkey(input_x):
    filename = str(input_x)
    mynumbers = []
    Hold = []
    Interkey = []
    Latency = []
    time_down = []
    time_up = []
    time = []
    key = []
    code = []
    c = []
    with open(filename) as f:
        for line in f:
            mynumbers.append([str(n) for n in line.strip().split(' ')])
    for pair in mynumbers:
        try:
            w,x,y,z = pair[0],pair[1],pair[2],pair[3]
            key.append(w)
        except IndexError:
            print "A line in the file doesn't have enough entries."
        try:
            c = y.split('code=')
            code.append(c[1])
        except IndexError:
            print "A line in the file doesn't have enough entries."
        try:
            t = z.split('Time=')
            time.append(t[1])
        except IndexError:
            print "A line in the file doesn't have enough entries."
    #จัดลำดับ key ที่เข้ามา และ แยกเวลากดแป้นแล้วยกแป้น----------
    outfile_key = open('Database/key.txt', "a") #เปิดไฟล์ key
    for i in range(len(code)): #วนลูปจำนวน i ครั้ง โดยที่ i = จำนวน code
        for a in range(len(code)): #วนลูปจำนวน a ครั้ง โดยที่ a = จำนวน code
            if (code[i]==code[a] and time[i] != time[a] and code[i] != '0'):
```

```
            time = map(float, time)
            c.append(code[i])
            outfile_key.write( str(code[i]) + " " ) #เขียนข้อมูลใน code ลงไฟล์
            time_down.append(time[i]) #เก็บข้อมูลที่เป็น time_down
            time_up.append(time[a]) #เก็บข้อมูลที่เป็น time_up
            code[i] = '0' #ให้ code[i] = 0
            code[a] = '0' #ให้ code[a] = 0
    outfile_key.write(" \r\n")
    outfile_key.close()
    #หา interkey ----------------------------------------------------
    outfile_Interkey = open('Database/interkey.txt', "a")  #เปิดไฟล์ interkey เพื่อเขียนเพิ่ม
    for i in range(len(time_down)-1):
        Interkey.append( (time_down[i+1] - time_up[i]) / 1000 )
        outfile_Interkey.write( str((time_down[i+1] - time_up[i]) / 1000 ) + " " )
    outfile_Interkey.write(" \r\n")
    outfile_Interkey.close()
    #หา latency ----------------------------------------------------
    outfile_Latency = open('Database/latency.txt', "a") #เปิดไฟล์ latency เพื่อเขียนเพิ่ม
    for i in range(len(time_down)-1):
        Latency.append( i + ((time_down[i+1] - time_down[i]) / 1000) )
        outfile_Latency.write( str(i + ((time_down[i+1] - time_down[i]) / 1000) ) + " " )
    outfile_Latency.write(" \r\n")
    outfile_Latency.close()
    #----------------------------------------------------------------

def callMeanMaster(): #หาค่าเฉลี่ยของ interkey และ latency เพื่อเป็นเส้นต้นแบบ
    FeatureX = []
    FeatureY = []
    sum = 0;
    sum_x = []
    sum_y = []
    mean_x = []
    mean_y = []
    with open('Database/latency.txt') as f:
        for line in f:
            FeatureX.append([float(n) for n in line.strip().split(' ')])
```

```
    out_FeatureX = open('Database/masterX.txt', "w")
    for column in range(len(FeatureX[0])):
        sum = 0
        for row in range(len(FeatureX)):
                sum += FeatureX[row][column]
        sum_x.append(sum)
    for column in range(len(sum_x)):
        mean_x.append(sum_x[column]/len(FeatureX))
        out_FeatureX.write(str(sum_x[column]/len(FeatureX)) + " " )
    out_FeatureX.write(" \r\n")
    out_FeatureX.close()

    with open('Database/interkey.txt') as f:
       for line in f:
         FeatureY.append([float(n) for n in line.strip().split(' ')])
    out_FeatureY = open('Database/masterY.txt', "w")
    for column in range(len(FeatureY[0])):
        sum = 0
        for row in range(len(FeatureY)):
                sum += FeatureY[row][column]
        sum_y.append(sum)
    for column in range(len(sum_y)):
        mean_y.append(sum_y[column]/len(FeatureX))
        out_FeatureY.write(str(sum_y[column]/len(FeatureX)) + " " )
    out_FeatureY.write(" \r\n")
    out_FeatureY.close()
    #----------------------------------------------------------------------

def callDiss(): # คำนวณหาค่าความแตกต่างของเส้นเวลาจังหวะการพิมพ์
    FeatureX = []
    FeatureY = []
    input_x = []
    input_y =[]
    sum = 0;
    sum_x = []
```

```
sum_y = []
mean_x = []
mean_y = []
Dissimilarity = []
x = []
y = []
z = []
with open('Database/masterX.txt') as f:
    for line in f:
        FeatureX.append([float(n) for n in line.strip().split(' ')])
with open('Database/masterY.txt') as f:
    for line in f:
        FeatureY.append([float(n) for n in line.strip().split(' ')])
file_X = str(open('Database/latency.txt'))
with open('Database/latency.txt') as f:
    for line in f:
        input_x.append([float(n) for n in line.strip().split(' ')])
file_Y = str(open('Database/interkey.txt'))
with open('Database/interkey.txt') as f:
    for line in f:
        input_y.append([float(n) for n in line.strip().split(' ')])
outfile = open('Database/Diss.txt', "w")
for row in range(len(input_x)):
      sum = 0
      for column in range(len(input_x[0])):
            x = (input_x[row][column] - FeatureX[0][column])**2
            y = (input_y[row][column] - FeatureY[0][column])**2
            z = math.sqrt(x + y)
            sum += z
      Dissimilarity.append(sum)
      outfile.write(str(sum) + "\n")
outfile.close()
#-----------------------------------------------------------------
```

```
def callCalculate(): #คำนวณหาค่าเฉลี่ยและค่าส่วนเบี่ยงเบนมาตรฐานเพื่อนำไปเป็นช่วง
    mynumbers = [] #การตัดสินใจ
    data = []
    sum = 0;
    a = []
    sum_t = []
    mean = []
    total = 0;
    total_t = []
    sd = []
    with open('Database/Diss.txt') as f:
        for line in f:
            #print "Line =" + Line
             mynumbers.append([float(n) for n in line.strip().split(' ')])
    outfile_mean = open('Database/Bav.txt', "w")
    outfile_sd = open('Database/Bsd.txt', "w")
    for column in range(len(mynumbers[0])):
            sum = 0
            for row in range(len(mynumbers)):
                    sum += mynumbers[row][column]
            sum_t.append(sum)
    for column in range(len(sum_t)):
            mean.append(sum_t[column]/len(mynumbers))
            outfile_mean.write(str(sum_t[column]/len(mynumbers)) + " " )
    for column in range(len(mynumbers[0])):
            for row in range(len(mynumbers)):
                    total += (mynumbers[row][column] - mean[column])**2
            total_t.append(total)
    for column in range(len(total_t)):
            sd.append(math.sqrt(total_t[column]/len(mynumbers)))
            outfile_sd.write(str(math.sqrt(total_t[column]/len(mynumbers))) + " " )
    outfile_mean.write(" \r\n")
    outfile_sd.write(" \r\n")
    outfile_mean.close()
    outfile_sd.close()
    #-----------------------------------------------------------------------
```

```
def callDBKey():
    key = []
    code = []
    cod = []
    c=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
    D = 0
    with open('Database/key.txt') as f:
        for line in f:
            key.append([str(n) for n in line.strip().split(' ')])
    DB_key = open('Database/DB_key.txt', "w")
    for i in range(len(key)):
        for a in range(len(key)):
            if(key[i]==key[a]):
                c[i]=c[i]+1
    for i in range(len(c)):
        if(max(c)==c[i]):
            code = key[i]
    for i in range(len(code)):
        DB_key.write(str(code[i]) + " " )
    DB_key.write(" \r\n")
    DB_key.close()
    #-----------------------------------------------------------------------

import sys
import math
import os
import glob
path = "."
for filename in glob.glob(os.path.join(path, '*.txt')):
    key = callkey(str(filename))
key = callMeanMaster()
Diss = callDiss()
callCalcul = callCalculate()
DB = callDBKey()
```

ภาคผนวก ค

โค๊ดโปรแกรมพิสูจน์ตัวตนด้วยจังหวะการพิมพ์

```
def callkey(input_x):
    filename = str(input_x)
    mynumbers = []
    Hold = []
    Interkey = []
    Latency = []
    time_down = []
    time_up = []
    time = []
    key = []
    code = []
    c = []
    with open(filename) as f:
        for line in f:
            mynumbers.append([str(n) for n in line.strip().split(' ')])
    for pair in mynumbers:
        try:
            w,x,y,z = pair[0],pair[1],pair[2],pair[3]
            key.append(w)
        except IndexError:
            print "A line in the file doesn't have enough entries."
        try:
            c = y.split('code=')
            code.append(c[1])
        except IndexError:
            print "A line in the file doesn't have enough entries."
        try:
            t = z.split('Time=')
            time.append(t[1])
        except IndexError:
            print "A line in the file doesn't have enough entries."
    #-----------------------------------------------------------------
    outfile_key = open('Data/key.txt', "w")
    for i in range(len(code)):
        for a in range(len(code)):
            if (code[i]==code[a] and time[i] != time[a] and code[i] != '0'):
```

```
                time = map(float, time)
                c.append(code[i])
                outfile_key.write( str(code[i]) + " " )
                time_down.append(time[i])
                time_up.append(time[a])
                code[i] = '0'
                code[a] = '0'
    outfile_key.write(" \r\n")
    outfile_key.close()
    #-----------------------------------------------------------------
    outfile_Interkey = open('Data/interkey.txt', "w")
    for i in range(len(time_down)-1):
        Interkey.append( (time_down[i+1] - time_up[i]) / 1000 )
        outfile_Interkey.write( str((time_down[i+1] - time_up[i]) / 1000 ) + " " )
    outfile_Interkey.write(" \r\n")
    outfile_Interkey.close()
    #-----------------------------------------------------------------
    outfile_Latency = open('Data/latency.txt', "w")
    for i in range(len(time_down)-1):
        Latency.append( i + ((time_down[i+1] - time_down[i]) / 1000) )
        outfile_Latency.write( str(i + ((time_down[i+1] - time_down[i]) / 1000) ) + " " )
    outfile_Latency.write(" \r\n")
    outfile_Latency.close()
    #-----------------------------------------------------------------

def callDiss():
    FeatureX = []
    FeatureY = []
    input_x = []
    input_y =[]
    sum = 0;
    sum_x = []
    sum_y = []
    mean_x = []
    mean_y = []
    Dissimilarity = []
```

```
    x = []
    y = []
    z = []
    with open('Database/masterX.txt') as f:
        for line in f:
            FeatureX.append([float(n) for n in line.strip().split(' ')])
    with open('Database/masterY.txt') as f:
        for line in f:
            FeatureY.append([float(n) for n in line.strip().split(' ')])
    file_X = str(open('Data/latency.txt'))
    with open('Data/latency.txt') as f:
        for line in f:
            input_x.append([float(n) for n in line.strip().split(' ')])
    file_Y = str(open('Data/interkey.txt'))
    with open('Data/interkey.txt') as f:
        for line in f:
            input_y.append([float(n) for n in line.strip().split(' ')])
    outfile = open('Data/Adiss.txt', "w")
    for row in range(len(input_x)):
            sum = 0
            for column in range(len(input_x[0])):
                    x = (input_x[row][column] - FeatureX[0][column])**2
                    y = (input_y[row][column] - FeatureY[0][column])**2
                    z = math.sqrt(x + y)
                    sum += z
            outfile.write(str(sum) + "\n")
    outfile.close()
    #-----------------------------------------------------------------------


def callProcess():
    Adiss = []
    Bav = []
    Bstd = []
    key_d = []
    key_i = []
    Sig = 1;
```

```
    with open('Data/Adiss.txt') as f:
        for line in f:
            Adiss.append([float(n) for n in line.strip().split(' ')])
    with open('Database/Bav.txt') as f:
        for line in f:
            Bav.append([float(n) for n in line.strip().split(' ')])
    with open('Database/Bsd.txt') as f:
        for line in f:
            Bstd.append([float(n) for n in line.strip().split(' ')])
    with open('Database/DB_key.txt') as f:
        for line in f:
            key_d.append([str(n) for n in line.strip().split(' ')])
    with open('Data/key.txt') as f:
        for line in f:
            key_i.append([str(n) for n in line.strip().split(' ')])
    x = (Sig*Bstd[0][0])
    y = (Bav[0][0] + (Sig * Bstd[0][0]))
    checklogin = open('Data/checklogin.txt', "a")
    if (Adiss[0][0] > (Bav[0][0] + (Sig * Bstd[0][0]))):
        print "Get_out!"
        checklogin.write(str("Get_out!") + "\n")
    if (Adiss[0][0] <= (Bav[0][0] + (Sig * Bstd[0][0]))) and key_d != key_i:
        print "Your_character_is_invalid!"
        checklogin.write(str("Your_character_is_invalid!") + "\n")
    if (Adiss[0][0] <= (Bav[0][0] + (Sig * Bstd[0][0]))) and key_d == key_i:
        print "Wellcome"
        checklogin.write(str("Wellcome") + "\n")
    checklogin.close()
    #----------------------------------------------------------------------

import sys
import math
import os
import glob
checklogin = []
```

```
path = "."
for filename in glob.glob(os.path.join(path, '*.txt')):
    key = callkey(str(filename))
    Adiss = callDiss()
    process = callProcess()
Data = open('Data/checklogin.txt',"r")
checklogin = Data.read()
str = checklogin
sub = "Get_out!";
print "Get_out! : ", str.count(sub)
sub = "Your_character_is_invalid!";
print "Your_character_is_invalid! : ", str.count(sub)
sub = "Wellcome";
print "Wellcome : ", str.count(sub)
Data = open('Data/checklogin.txt',"w")
Data.close()
```

ภาคผนวก ง

โค๊ดเว็บไซต์ล็อกอินต้นแบบ สำหรับระบบการพิสูจน์ตัวตนด้วยจังหวะการพิมพ์

ไฟล์ connection.php

```
<?php
$host = 'localhost';
$user = 'root';
$pass = '';
$db = 'accounts';
$con= mysqli_connect($host, $user, $pass,$db) or die("Error: " . mysqli_error($con));
mysqli_query($con, "SET NAMES 'utf8' "); //เชื่อมต่อกับฐานข้อมูล
?>
```

ไฟล์ db.php

```
<?php
/* Database connection settings */
$host = 'localhost';
$user = 'root';
$pass = '';
$db = 'accounts';
$mysqli = new mysqli( $host, $user, $pass,$db ) or die($mysqli->error);
//เชื่อมต่อกับฐานข้อมูล
```

ไฟล์ DataFrom.php

```
<?php
ob_start();
/* Main page with two forms: sign up and log in */
require 'db.php'; //เรียก db
session_start();
?>
<?php
    if (isset($_REQUEST['login'])) { //user logging in

        require 'login.php'; //เรียกหน้า login

    }

    elseif (isset($_REQUEST['register'])) { //user registering
```

```
        require 'register.php'; //เรียกหน้า require
    }
?>
```

ไฟล์ index.php

```
<?php
ob_start();
/* Main page with two forms: sign up and log in */
require 'db.php';
session_start();
?>
<?php
$myfile = fopen("Data/data.txt","w+");
        fclose($myfile);
?>

<!DOCTYPE html>
<html>
<head>
  <title>Sign-Up/Login Form</title>
  <?php include 'css/css.html'; ?>
</head>

<body>
  <div class="form">

      <ul class="tab-group">
        <li class="tab"><a href="#signup">Sign Up</a></li>
        <li class="tab active"><a href="#login">Log In</a></li>
      </ul>

      <div class="tab-content">

        <div id="login">
         <h1>Welcome Back!</h1>
```

```
        <form action="DataFrom.php" method="post" autocomplete="on">

          <div class="field-wrap">
          <label>
            Username<span class="req">*</span>
          </label>
          <input type="username" required autocomplete="off" name="username"
id='kinput' />
        </div>

        <div class="field-wrap">
          <label>
            Password<span class="req">*</span>
          </label>
          <input type="password" required autocomplete="off" name="password"/>
        </div>

        <div class="field-wrap">
          <label>
            <span class="req"></span>
          </label>
          <!--<textarea id="area"></textarea>-->
        </div>

        <p class="forgot"><a href="forgot.php">Forgot Password?</a></p>


         <button class="button button-block" name="login" />Log In</button>

        </form>

      </div>

      <div id="signup">
        <h1>Sign Up for Free</h1>
```

```
<form action="DataFrom.php" method="post" autocomplete="off">

<div class="top-row">
  <div class="field-wrap">
    <label>
      First Name<span class="req">*</span>
    </label>
    <input type="text" required autocomplete="off" name='firstname' />
  </div>

  <div class="field-wrap">
    <label>
      Last Name<span class="req">*</span>
    </label>
    <input type="text"required autocomplete="off" name='lastname' />
  </div>
</div>

<div class="field-wrap">
  <label>
    Email Address<span class="req">*</span>
  </label>
  <input type="email"required autocomplete="off" name='email' />
</div>

<div class="top-row">
  <div class="field-wrap">
    <label>
      Set A Username<span class="req">*</span>
    </label>
    <input type="username"required autocomplete="off" name='username'
id='kinput1'/>
  </div>

  <div class="field-wrap">
    <label>
```

```
            Set A Password<span class="req">*</span>
          </label>
          <input type="password"required autocomplete="off" name='password'/>
        </div>
      </div>


      <button type="submit" class="button button-block" name="register"
/>Register</button>

      </form>

    </div>

  </div><!-- tab-content -->

</div> <!-- /form -->
<script src="js/script.js"></script>
<script src='http://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js'></script>


<script src="js/index.js"></script>
</body>
</html>
```

ไฟล์ error.php

```
<?php
/* Displays all error messages */
session_start();
?>
<!DOCTYPE html>
<html>
<head>
  <title>Error</title>
  <?php include 'css/css.html'; ?>
</head>
```

```
<body>
<div class="form">
    <h1>Error</h1>
    <p>
    <?php
    if( isset($_SESSION['message']) AND !empty($_SESSION['message']) ):
        echo $_SESSION['message'];
    else:
        header( "location: index.php" ); //ย้ายไปยังหน้า index
    endif;
    ?>
    </p>
    <a href="index.php"><button class="button button-block"/>Home</button></a>
</div>
</body>
</html>
```

ไฟล์ success.php

```
<?php
/* Displays all successful messages */
session_start();
?>
<!DOCTYPE html>
<html>
<head>
  <title>Success</title>
  <?php include 'css/css.html'; ?>
</head>
<body>
<div class="form">
    <h1><?= 'Success'; ?></h1>
    <p>
    <?php
    if( isset($_SESSION['message']) AND !empty($_SESSION['message']) ):
        echo $_SESSION['message'];
    else:
```

```
        header( "location: index.php" ); //ย้ายไปยังหน้า index
    endif;
    ?>
    </p>
    <a href="index.php"><button class="button button-block"/>Home</button></a>
</div>
</body>
</html>
```

ไฟล์ forgot.php

```
<?php
/* Reset your password form, sends reset.php password link */
require 'db.php';
session_start();

// Check if form submitted with method="post"
if ( $_SERVER['REQUEST_METHOD'] == 'POST' )
{
    $email = $mysqli->escape_string($_POST['email']);
    $result = $mysqli->query("SELECT * FROM users WHERE email='$email'");

    if ( $result->num_rows == 0 ) // User doesn't exist
    {
        $_SESSION['message'] = "User with that email doesn't exist!";
        header("location: error.php"); //ย้ายไปยังหน้า error
    }
    else { // User exists (num_rows != 0)

        $user = $result->fetch_assoc(); // $user becomes array with user data

        $email = $user['email'];
        $hash = $user['hash'];
        $first_name = $user['first_name'];

        // Session message to display on success.php
        $_SESSION['message'] = "<p>Please check your email <span>$email</span>"
```

```
        . " for a confirmation link to complete your password reset!</p>";

        // Send registration confirmation link (reset.php)
        require 'PHPMailer-master/PHPMailerAutoload.php';
        $mail = new PHPMailer();
        $mail ->IsSmtp();
        $mail ->SMTPDebug = 0;
        $mail ->SMTPAuth = true;
        $mail ->SMTPSecure = 'ssl';
        $mail ->Host = "smtp.gmail.com";
        $mail ->Port = 465; // or 587
        $mail ->IsHTML(true);
        $mail ->Username = "keystorkeproject@gmail.com";
        $mail ->Password = "Project2018";
        $mail ->SetFrom("keystorkeproject@gmail.com");
        $to      = $email;
        $subject = 'Password Reset Link';
        $message_body = '
        Hello '.$first_name.',

        You have requested password reset!

        Please click this link to reset your password:

        http://localhost/login-system/reset.php?email='.$email.'&hash='.$hash;

        $mail ->Subject = $subject;
        $mail ->Body = $message_body;
        $mail ->AddAddress($to);
        $mail->Send();

        header("location: success.php"); //ย้ายไปยังหน้า success
    }
}
?>
<!DOCTYPE html>
```

```
<html>
<head>
  <title>Reset Your Password</title>
  <?php include 'css/css.html'; ?>
</head>

<body>

  <div class="form">

    <h1>Reset Your Password</h1>

    <form action="forgot.php" method="post">
     <div class="field-wrap">
      <label>
        Email Address<span class="req">*</span>
      </label>
      <input type="email"required autocomplete="off" name="email"/>
    </div>
    <button class="button button-block"/>Reset</button>
    </form>
  </div>

<script src='http://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js'></script>
<script src="js/index.js"></script>
</body>

</html>
```

ไฟล์ login.php

```
<?php
/* User login process, checks if user exists and password is correct */

// Escape email to protect against SQL injections
$username = $mysqli->escape_string($_REQUEST['username']);
$result = $mysqli->query("SELECT * FROM users WHERE username='$username'");
```

```
if ( $result->num_rows == 0 ){ // User doesn't exist
    $_SESSION['message'] = "User with that email doesn't exist!";
    header("location: error.php"); //ย้ายไปยังหน้า error
}
else { // User exists
    $user = $result->fetch_assoc();

    if ( password_verify($_POST['password'], $user['password']) ) {

        $_SESSION['username'] = $user['username'];
        $_SESSION['email'] = $user['email'];
        $_SESSION['first_name'] = $user['first_name'];
        $_SESSION['last_name'] = $user['last_name'];
        $_SESSION['active'] = $user['active'];

        // This is how we'll know the user is logged in
        $_SESSION['logged_in'] = true;

        header("location: stroke_from.php"); //ย้ายไปยังหน้า stroke_from
    }
    else {
        $_SESSION['message'] = "You have entered wrong password, try again!";
        header("location: error.php"); //ย้ายไปยังหน้า error
    }
}
?>
```

ไฟล์ logout.php

```
<?php
/* Log out process, unsets and destroys session variables */
session_start();
session_unset();
session_destroy();
?>
<!DOCTYPE html>
```

```
<html>
<head>
  <meta charset="UTF-8">
  <title>Error</title>
  <?php include 'css/css.html'; ?>
</head>

<body>
    <div class="form">
        <h1>Thanks for stopping by</h1>

        <p><?= 'You have been logged out!'; ?></p>

        <a href="index.php"><button class="button button-
block"/>Home</button></a>

    </div>
</body>
</html>
```

ไฟล์ openfile_db.php

```
<?php
$fBav = fopen("Database/Bav.txt","w+"); //เปิดไฟล์ Bav
    fwrite($fBav,$user['Bav']); //เขียนข้อมูลที่เก็บไว้ใน Bav ลงไปในไฟล์
$fBsd = fopen("Database/Bsd.txt","w+"); //เปิดไฟล์ Bsd
    fwrite($fBsd,$user['Bsd']); //เขียนข้อมูลที่เก็บไว้ใน Bsd ลงไปในไฟล์
$fDB_kry = fopen("Database/DB_key.txt","w+"); //เปิดไฟล์ DB_key
    fwrite($fDB_kry,$user['DB_key']); //เขียนข้อมูลที่เก็บไว้ใน DB_key ลงไปในไฟล์
$fDiss = fopen("Database/Diss.txt","w+"); //เปิดไฟล์ Diss
    fwrite($fDiss,$user['Diss']); //เขียนข้อมูลที่เก็บไว้ใน Diss ลงไปในไฟล์
$fMasterX = fopen("Database/MasterX.txt","w+"); //เปิดไฟล์ MasterX
    fwrite($fMasterX,$user['MasterX']); //เขียนข้อมูลที่เก็บไว้ใน MasterX ลงไปในไฟล์
$fMasterY = fopen("Database/MasterY.txt","w+"); //เปิดไฟล์ MasterY
    fwrite($fMasterY,$user['MasterY']); //เขียนข้อมูลที่เก็บไว้ใน MasterY ลงไปในไฟล์
?>
```

ไฟล์ print.php

```
<?php
$r = $_GET["text"]; //รับค่า text มา เก็บที่ตัวแปร r
$myfile = fopen("Data/data.txt", "a+") or die("Unable to open file!"); //เปิดไฟล์ data
fwrite($myfile,$r."\n"); //เพิ่มข้อมูลที่เก็บไว้ใน r ลงไปในไฟล์ data
fclose($myfile); //ปิดไฟล์ data
?>
```

ไฟล์ profile.php

```
<?php
/* Displays user information and some useful messages */
session_start();

// Check if user is logged in using the session variable
if ( $_SESSION['logged_in'] != 1 ) {
  $_SESSION['message'] = "You must log in before viewing your profile page!";
  header("location: error.php"); //ย้ายไปยังหน้า error
}
else {
    // Makes it easier to read
    $username = $_SESSION['username'];
    $first_name = $_SESSION['first_name'];
    $last_name = $_SESSION['last_name'];
    $email = $_SESSION['email'];
    $active = $_SESSION['active'];
}
?>
<!DOCTYPE html>
<html >
<head>
  <meta charset="UTF-8">
  <title>Welcome <?= $first_name.' '.$last_name ?></title>
  <?php include 'css/css.html'; ?>
</head>

<body>
```

```
<div class="form">

    <h1>Welcome</h1>

    <p>
    <?php

    // Display message about account verification link only once
    if ( isset($_SESSION['message']) )
    {
        echo $_SESSION['message'];

        // Don't annoy the user with more messages upon page refresh
        unset( $_SESSION['message'] );
    }

    ?>
    </p>

    <?php

    // Keep reminding the user this account is not active, until they activate
    if ( !$active ){
        echo
        '<div class="info">
        Account is unverified, please confirm your email by clicking
        on the email link!
        </div>';
    }

    ?>

    <h2><?php echo $first_name.' '.$last_name; ?></h2>
    <p><?= $email ?></p>
```

```
        <a href="logout.php"><button class="button button-block"
name="logout"/>Log Out</button></a>

    </div>

<script src='http://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js'></script>
<script src="js/index.js"></script>
</body>
</html>
```

ไฟล์ register.php

```
<?php
/* Registration process, inserts user info into the database
   and sends account confirmation email message
 */

 // Set session variables to be used on profile.php page
 $_SESSION['email'] = $_POST['email'];
 $_SESSION['first_name'] = $_POST['firstname'];
 $_SESSION['last_name'] = $_POST['lastname'];

 // Escape all $_POST variables to protect against SQL injections
 $first_name = $mysqli->escape_string($_POST['firstname']);
 $last_name = $mysqli->escape_string($_POST['lastname']);
 $email = $mysqli->escape_string($_POST['email']);
 $username = $mysqli->escape_string($_POST['username']);
 $password = $mysqli->escape_string(password_hash($_POST['password'],
PASSWORD_BCRYPT));
 $hash = $mysqli->escape_string( md5( rand(0,1000) ) );

 // Check if user with that email already exists
 $result = $mysqli->query("SELECT * FROM users WHERE email='$email'") or
die($mysqli->error());
 $result1 = $mysqli->query("SELECT * FROM users WHERE username='$username'") or
die($mysqli->error());
```

```
// We know user email exists if the rows returned are more than 0
if ( $result->num_rows > 0 ) {

    $_SESSION['message'] = 'User with this email already exists!';
    header("location: error.php"); //ย้ายไปยังหน้า error

}
else if ($result1->num_rows > 0) {
   $_SESSION['message'] = 'User with this username already exists!';
   header("location: error.php"); //ย้ายไปยังหน้า error
}
else { // Email doesn't already exist in a database, proceed...

    // active is 0 by DEFAULT (no need to include it here)
    $sql = "INSERT INTO users (first_name, last_name, email, username, password,
hash) "
          . "VALUES ('$first_name','$last_name','$email','$username','$password',
'$hash')";

    // Add user to the database
    if ( $mysqli->query($sql) ){

        $_SESSION['active'] = 0; //0 until user activates their account with verify.php
        $_SESSION['logged_in'] = true; // So we know the user has logged in
        $_SESSION['message'] =

             "Confirmation link has been sent to $email, please verify
             your account by clicking on the link in the message!";

        // Send registration confirmation link (verify.php)

        require 'PHPMailer-master/PHPMailerAutoload.php';
        $mail = new PHPMailer();
        $mail ->IsSmtp();
        $mail ->SMTPDebug = 0;
        $mail ->SMTPAuth = true;
```

```
        $mail ->SMTPSecure = 'ssl';
        $mail ->Host = "smtp.gmail.com";
        $mail ->Port = 465; // or 587
        $mail ->IsHTML(true);
        $mail ->Username = "keystorkeproject@gmail.com";
        $mail ->Password = "Project2018";
        $mail ->SetFrom("keystorkeproject@gmail.com");

        $to      = $email;
        $subject = 'Account Verification';
        $message_body = '
        Hello '.$first_name.',

        Thank you for signing up!

        Please click this link to activate your account:

        http://localhost/login-system/verify.php?email='.$email.'&hash='.$hash;

        $mail ->Subject = $subject;
        $mail ->Body = $message_body;
        $mail ->AddAddress($to);
        $mail->Send();
        //mail( $to, $subject, $message_body );

        header("location: profile.php"); //ย้ายไปยังหน้า profile

    }

    else {
        $_SESSION['message'] = 'Registration failed!';
        header("location: error.php"); //ย้ายไปยังหน้า error
    }

}
?>
```

ไฟล์ reset_password.php

```
<?php
/* Password reset process, updates database with new user password */
require 'db.php';
session_start();

// Make sure the form is being submitted with method="post"
if ($_SERVER['REQUEST_METHOD'] == 'POST') {

    // Make sure the two passwords match
    if ( $_POST['newpassword'] == $_POST['confirmpassword'] ) {

        $new_password = password_hash($_POST['newpassword'], PASSWORD_BCRYPT);

        // We get $_POST['email'] and $_POST['hash'] from the hidden input field of
reset.php form
        $email = $mysqli->escape_string($_POST['email']);
        $hash = $mysqli->escape_string($_POST['hash']);

        $sql = "UPDATE users SET password='$new_password', hash='$hash' WHERE
email='$email'";

        if ( $mysqli->query($sql) ) {

        $_SESSION['message'] = "Your password has been reset successfully!";
        header("location: success.php");    //ย้ายไปยังหน้า success

        }

    }
    else {
        $_SESSION['message'] = "Two passwords you entered don't match, try again!";
        header("location: error.php");    //ย้ายไปยังหน้า error
    }
```

```
}
?>
```

ไฟล์ reset.php

```
<?php
/* The password reset form, the link to this page is included
   from the forgot.php email message
*/
require 'db.php';
session_start();

// Make sure email and hash variables aren't empty
if( isset($_GET['email']) && !empty($_GET['email']) AND isset($_GET['hash']) &&
!empty($_GET['hash']) )
{
    $email = $mysqli->escape_string($_GET['email']);
    $hash = $mysqli->escape_string($_GET['hash']);

    // Make sure user email with matching hash exist
    $result = $mysqli->query("SELECT * FROM users WHERE email='$email' AND
hash='$hash'");

    if ( $result->num_rows == 0 )
    {
        $_SESSION['message'] = "You have entered invalid URL for password reset!";
        header("location: error.php"); //ย้ายไปยังหน้า error
    }
}
else {
    $_SESSION['message'] = "Sorry, verification failed, try again!";
    header("location: error.php");  //ย้ายไปยังหน้า error
}
?>
<!DOCTYPE html>
<html >
<head>
```

```
  <meta charset="UTF-8">
  <title>Reset Your Password</title>
  <?php include 'css/css.html'; ?>
</head>

<body>
   <div class="form">

        <h1>Choose Your New Password</h1>

        <form action="reset_password.php" method="post">

        <div class="field-wrap">
          <label>
            New Password<span class="req">*</span>
          </label>
          <input type="password"required name="newpassword" autocomplete="off"/>
        </div>

        <div class="field-wrap">
          <label>
            Confirm New Password<span class="req">*</span>
          </label>
          <input type="password"required name="confirmpassword"
autocomplete="off"/>
        </div>

        <!-- This input field is needed, to get the email of the user -->
        <input type="hidden" name="email" value="<?= $email ?>">
        <input type="hidden" name="hash" value="<?= $hash ?>">

        <button class="button button-block"/>Apply</button>

        </form>

   </div>
```

```
<script src='http://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js'></script>
<script src="js/index.js"></script>

</body>
</html>
```

ไฟล์ stroke_check.php

```
<?php
require 'db.php';
session_start();

if (isset($_REQUEST['stroke'])) {
  include("connection.php");
  $username = $_SESSION['username'];
  $first_name = $_SESSION['first_name'];
  $last_name = $_SESSION['last_name'];
  $email = $_SESSION['email'];
  $active = $_SESSION['active'];
  $word = $_REQUEST['stroke'];



  $sql = "
  SELECT
  *
  FROM
  users
  INNER JOIN word ON word.user_id = users.user_id
  INNER JOIN keystroke ON keystroke.user_id = word.user_id AND keystroke.word_id =
word.word_id
  Where username='$username' and word='$word'
  ";

  $result = mysqli_query($con,$sql);
  if ( $result->num_rows == 0 ){ // User doesn't exist
      $_SESSION['message'] = $word; //"User with that word doesn't exist!";
```

```
    header("location: error.php"); //ย้ายไปยังหน้า error
  }
  else { // User exists
    $user = $result->fetch_assoc();
    include("openfile_db.php");
    $result = exec("python py\checklogin.py");
    $check = fopen("Data/checklogin.txt","r");
    $data = fgets($check);

    if ($data == "Wellcome") {

      $_SESSION['username'] = $user['username'];
      $_SESSION['email'] = $user['email'];
      $_SESSION['first_name'] = $user['first_name'];
      $_SESSION['last_name'] = $user['last_name'];
      $_SESSION['active'] = $user['active'];

      // This is how we'll know the user is logged in
      $_SESSION['logged_in'] = true;

      header("location: profile.php"); //ย้ายไปยังหน้า profile
    }
    else {
      $_SESSION['message'] = "You have entered wrong password, try again!";
      header("location: error.php"); //ย้ายไปยังหน้า error
    }
  }

}else {
  Header("Location: index.php"); //user & password incorrect back to login again
}
?>
```

ไฟล์ stroke_from.php

```
<?php
ob_start();
```

```
/* Main page with two forms: sign up and log in */
require 'db.php';
session_start();
?>
<?php
$myfile = fopen("Data/data.txt","w+");
        fclose($myfile);
?>
<?php
// Check if user is logged in using the session variable
if ( $_SESSION['logged_in'] != 1 ) {
  $_SESSION['message'] = "You must log in before viewing your profile page!";
  header("location: error.php"); //ย้ายไปยังหน้า error
}
else {
    // Makes it easier to read
    $username = $_SESSION['username'];
    $first_name = $_SESSION['first_name'];
    $last_name = $_SESSION['last_name'];
    $email = $_SESSION['email'];
    $active = $_SESSION['active'];

    $_SESSION['username'] = $username;
    $_SESSION['email'] = $email;
    $_SESSION['first_name'] = $first_name;
    $_SESSION['last_name'] = $last_name;
    $_SESSION['active'] = $active;
}
?>

<!DOCTYPE html>
<html >
<head>
  <meta charset="UTF-8">
  <title>Keystroke Dynamics Authentication From</title>
  <?php include 'css/css.html'; ?>
```

```
</head>

<body>
  <div class="form">
    <form action="stroke_check.php" method="post" autocomplete="on">
    <h1>Keystroke Dynamics Authentication</h1>
    <button class="button button-block"/>
    <?php
    include("connection.php");
    $query = "
    SELECT *
    FROM
    users
    INNER JOIN word ON word.user_id = users.user_id
    Where username='$username'
    ORDER BY RAND()
    LIMIT 0,1
    ";
    $result3 = mysqli_query($con,$query);
    while ($line = mysqli_fetch_array($result3)){
      print $line["word"]." ";
    }
    ?>
    </button>
    <p></p>
    <div class="field-wrap">
      <label>
        โปรดพิมพ์คำศัพท์ตามที่คุณเห็น<span class="req">*</span>
      </label>
      <input type="text"required autocomplete="off" name='stroke' id='stroke'>
    </div>

    <div class="field-wrap">
      <label>
        <span class="req"></span>
      </label>
```

```
    <!--<textarea id="area"></textarea>-->
  </div>
      <button class="button button-block" name="SUBMIT"/>SUBMIT</button></a>

  </div>

<script src="js/script0.js"></script>
<script src='http://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js'></script>
<script src="js/index.js"></script>
</body>
</html>
```

ไฟล์ verify.php

```
<?php
/* Verifies registered user email, the link to this page
   is included in the register.php email message
*/
require 'db.php';
session_start();

// Make sure email and hash variables aren't empty
if(isset($_GET['email']) && !empty($_GET['email']) AND isset($_GET['hash']) &&
!empty($_GET['hash']))
{
   $email = $mysqli->escape_string($_GET['email']);
   $hash = $mysqli->escape_string($_GET['hash']);

   // Select user with matching email and hash, who hasn't verified their account yet
(active = 0)
   $result = $mysqli->query("SELECT * FROM users WHERE email='$email' AND
hash='$hash' AND active='0'");

   if ( $result->num_rows == 0 )
   {
      $_SESSION['message'] = "Account has already been activated or the URL is
invalid!";
```

```
        header("location: error.php"); //ย้ายไปยังหน้า error
    }
    else {
        $_SESSION['message'] = "Your account has been activated!";

        // Set the user status to active (active = 1)
        $mysqli->query("UPDATE users SET active='1' WHERE email='$email'") or
die($mysqli->error);
        $_SESSION['active'] = 1;

        header("location: success.php"); //ย้ายไปยังหน้า success
    }
}
else {
    $_SESSION['message'] = "Invalid parameters provided for account verification!";
    header("location: error.php"); //ย้ายไปยังหน้า error
}
?>
```

ภาคผนวก จ

ผลการทดสอบประสิทธิภาพ

ตารางภาพรวมการเก็บข้อมูลทั้งหมด

ลำดับ	รายชื่อ	คำ									รวม
		คำสั้น			คำยาว			คำยาก			
		ที่อยู่	บ่อยครั้ง	ซ้ำเติม	พระจอมเกล้า	คอมพิวเตอร์	อนุญาต	ศรัทธา	สาบสูญ	ปฐมนิเทศ	
1	Jame	30	30	30	30	30	30	30	30	30	270
2	Fang	30	30	30	30	30	30	30	30	30	270
3	Mickey	30	30	30	30	30	30	30	30	30	270
4	May-Math	30	30	30	30	30	30	30	30	30	270
5	Bank	30	30	30	30	30	30	30	30	30	270
6	Khanin	30	30	30	30	30	30	30	30	30	270
7	View	30	30	30	30	30	30	30	30	30	270
8	Nan	30	30	30	30	30	30	30	30	30	270
9	Muay	30	30	30	30	30	30	30	30	30	270
10	Ging-Math	30	30	30	30	30	30	30	30	30	270
11	Prem	30	30	30	30	30	30	30	30	30	270
12	Fair	30	30	30	30	30	30	30	30	30	270
13	Jui	30	30	30	30	30	30	30	30	30	270
14	Ying	30	30	30	30	30	30	30	30	30	270
15	Kratae-Math	30	30	30	30	30	30	30	30	30	270
16	Jib-Math	30	30	30	30	30	30	30	30	30	270
17	Mazda-Math	30	30	30	30	30	30	30	30	30	270
18	Fern-Math	30	30	30	30	30	30	30	30	30	270
19	Ploy-Math	30	30	30	30	30	30	30	30	30	270
20	Deedo	30	30	30	30	30	30	30	30	30	270
	รวม	600	600	600	600	600	600	600	600	600	5400

หมายเหตุ :	
	คำสั้น : คำไม่เกิน 2 พยางค์ มีหรือไม่มี shift ก็ได้
	คำยาว : คำ 2 พยางค์ขึ้นไป มีหรือไม่มี shift ก็ได้
	คำยาก : คำ 2 พยางค์ขึ้นไป และมี shift มากกว่า 1 ครั้ง

ตารางการทดสอบประสิทธิของคำง่าย เมื่อเก็บข้อมูลลงฐานข้อมูล 3 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
ที่อยู่	Jame	30	9	0	21	30	0	2	28	70.00	100.00
	Fang	30	10	0	20	30	2	1	27	66.67	93.33
	Mickey	30	8	0	22	30	0	0	30	73.33	100.00
	May-Math	30	4	0	26	30	0	0	30	86.67	100.00
	Bank	30	29	0	1	30	4	6	20	3.33	86.67
	Khanin	30	19	0	11	30	0	0	30	36.67	100.00
	View	30	20	0	10	30	5	1	24	33.33	83.33
	Nan	30	14	0	16	30	0	0	30	53.33	100.00
	Muay	30	25	0	5	30	0	0	30	16.67	100.00
	Ging-Math	30	5	0	25	30	0	0	30	83.33	100.00
	Prem	30	26	0	4	30	10	3	17	13.33	66.67
	Fair	30	9	1	20	30	4	2	24	70.00	86.67
	Jui	30	11	0	19	30	0	0	30	63.33	100.00
	Ying	30	28	0	2	30	5	4	21	6.67	83.33
	Kratae-Math	30	11	0	19	30	0	0	30	63.33	100.00
	Jib-Math	30	26	0	4	30	0	6	24	13.33	100.00
	Mazda-Math	30	9	0	21	30	5	0	25	70.00	83.33
	Fern-Math	30	20	0	10	30	0	0	30	33.33	100.00
	Ploy-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Deedo	30	21	0	9	30	3	9	18	30.00	90.00
ผลรวม		600	326	1	273	600	38	34	528	45.67	93.67
เปอร์เซ็น			54.33	0.17	45.50		6.33	5.67	88.00		
บ่อยครั้ง	Jame	30	12	0	18	30	1	0	29	60.00	96.67
	Fang	30	12	0	18	30	0	0	30	60.00	100.00
	Mickey	30	17	0	13	30	0	0	30	43.33	100.00
	May-Math	30	8	0	22	30	0	0	30	73.33	100.00
	Bank	30	6	0	24	30	0	0	30	80.00	100.00
	Khanin	30	24	0	6	30	0	0	30	20.00	100.00
	View	30	25	0	5	30	4	0	26	16.67	86.67
	Nan	30	2	0	28	30	0	0	30	93.33	100.00
	Muay	30	4	0	26	30	0	0	30	86.67	100.00
	Ging-Math	30	26	0	4	30	3	0	27	13.33	90.00
	Prem	30	3	0	27	30	0	0	30	90.00	100.00
	Fair	30	10	0	20	30	0	0	30	66.67	100.00
	Jui	30	5	0	25	30	0	0	30	83.33	100.00
	Ying	30	11	0	19	30	0	0	30	63.33	100.00

	Kratae-Math	30	10	0	20	30	0	0	30	66.67	100.00
	Jib-Math	30	15	0	15	30	0	0	30	50.00	100.00
	Mazda-Math	30	5	0	25	30	0	0	30	83.33	100.00
	Fern-Math	30	5	0	25	30	0	0	30	83.33	100.00
	Ploy-Math	30	20	0	10	30	0	0	30	33.33	100.00
	Deedo	30	21	0	9	30	0	0	30	30.00	100.00
ผลรวม		600	241	0	359	600	8	0	592	59.83	98.67
เปอร์เซ็น			40.17	0.00	59.83		1.33	0.00	98.67		
ซ้ำเติม	Jame	30	2	0	28	30	0	0	30	93.33	100.00
	Fang	30	4	0	26	30	3	0	27	86.67	90.00
	Mickey	30	6	0	24	30	0	0	30	80.00	100.00
	May-Math	30	5	0	25	30	3	0	27	83.33	90.00
	Bank	30	18	0	12	30	0	0	30	40.00	100.00
	Khanin	30	6	0	24	30	0	0	30	80.00	100.00
	View	30	10	0	20	30	0	0	30	66.67	100.00
	Nan	30	2	0	28	30	0	0	30	93.33	100.00
	Muay	30	4	0	26	30	0	0	30	86.67	100.00
	Ging-Math	30	6	0	24	30	0	0	30	80.00	100.00
	Prem	30	2	0	28	30	1	0	29	93.33	96.67
	Fair	30	1	23	6	30	9	18	3	96.67	70.00
	Jui	30	16	0	14	30	0	0	30	46.67	100.00
	Ying	30	3	0	27	30	0	0	30	90.00	100.00
	Kratae-Math	30	4	0	26	30	0	0	30	86.67	100.00
	Jib-Math	30	10	0	20	30	1	0	29	66.67	96.67
	Mazda-Math	30	9	0	21	30	0	0	30	70.00	100.00
	Fern-Math	30	18	0	12	30	0	0	30	40.00	100.00
	Ploy-Math	30	24	0	6	30	1	0	29	20.00	96.67
	Deedo	30	13	0	17	30	0	0	30	56.67	100.00
ผลรวม		600	163	23	414	600	18	18	564	72.83	97.00
เปอร์เซ็น			27.17	3.83	69.00		3.00	3.00	94.00		
ผลรวมของระบบ		1800	730	24	1046	1800	64	52	1684	59.44	96.44
เปอร์เซ็นของระบบ			40.56	1.33	58.11		3.56	2.89	93.56		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	730	24	1046	1800	64	52	1684
FRR	59.44							
FAR	3.56							

ตารางการทดสอบประสิทธิของคำง่าย เมื่อเก็บข้อมูลลงฐานข้อมูล 6 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
ที่อยู่	Jame	30	22	0	8	30	1	7	22	26.67	96.67
	Fang	30	17	0	13	30	2	1	27	43.33	93.33
	Mickey	30	24	0	6	30	0	0	30	20.00	100.00
	May-Math	30	20	0	10	30	1	2	27	33.33	96.67
	Bank	30	22	0	8	30	0	0	30	26.67	100.00
	Khanin	30	17	0	13	30	0	0	30	43.33	100.00
	View	30	28	0	2	30	6	5	19	6.67	80.00
	Nan	30	14	0	16	30	0	0	30	53.33	100.00
	Muay	30	21	0	9	30	0	0	30	30.00	100.00
	Ging-Math	30	27	0	3	30	0	0	30	10.00	100.00
	Prem	30	25	0	5	30	10	2	18	16.67	66.67
	Fair	30	27	1	2	30	11	12	7	10.00	63.33
	Jui	30	26	0	4	30	5	2	23	13.33	83.33
	Ying	30	29	0	1	30	10	6	14	3.33	66.67
	Kratae-Math	30	19	0	11	30	1	0	29	36.67	96.67
	Jib-Math	30	26	0	4	30	10	10	10	13.33	66.67
	Mazda-Math	30	24	0	6	30	0	0	30	20.00	100.00
	Fern-Math	30	23	0	7	30	0	0	30	23.33	100.00
	Ploy-Math	30	25	0	5	30	0	0	30	16.67	100.00
	Deedo	30	22	0	8	30	9	12	9	26.67	70.00
ผลรวม		600	458	1	141	600	66	59	475	23.67	89.00
เปอร์เซ็น			76.33	0.17	23.50		11.00	9.83	79.17		
บ่อยครั้ง	Jame	30	29	0	1	30	14	0	16	3.33	53.33
	Fang	30	27	0	3	30	15	0	15	10.00	50.00
	Mickey	30	20	0	10	30	0	0	30	33.33	100.00
	May-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Bank	30	28	0	2	30	5	0	25	6.67	83.33
	Khanin	30	19	0	11	30	0	0	30	36.67	100.00
	View	30	28	0	2	30	0	0	30	6.67	100.00
	Nan	30	24	0	6	30	3	0	27	20.00	90.00
	Muay	30	23	0	7	30	0	0	30	23.33	100.00
	Ging-Math	30	27	0	3	30	2	0	28	10.00	93.33
	Prem	30	17	0	13	30	3	0	27	43.33	90.00
	Fair	30	29	0	1	30	21	0	9	3.33	30.00
	Jui	30	19	0	11	30	0	0	30	36.67	100.00
	Ying	30	21	0	9	30	0	0	30	30.00	100.00

	Kratae-Math	30	18	0	12	30	1	0	29	40.00	96.67
	Jib-Math	30	22	0	8	30	1	0	29	26.67	96.67
	Mazda-Math	30	18	0	12	30	0	0	30	40.00	100.00
	Fern-Math	30	26	0	4	30	1	0	29	13.33	96.67
	Ploy-Math	30	23	0	7	30	1	0	29	23.33	96.67
	Deedo	30	27	0	3	30	21	0	9	10.00	30.00
ผลรวม		600	467	0	133	600	88	0	512	22.17	85.33
เปอร์เซ็น			77.83	0.00	22.17		14.67	0.00	85.33		
ซ้ำเติม	Jame	30	24	0	6	30	10	2	18	20.00	66.67
	Fang	30	8	0	22	30	3	8	19	73.33	90.00
	Mickey	30	22	0	8	30	0	0	30	26.67	100.00
	May-Math	30	12	0	18	30	4	0	26	60.00	86.67
	Bank	30	25	0	5	30	0	0	30	16.67	100.00
	Khanin	30	21	0	9	30	0	0	30	30.00	100.00
	View	30	25	0	5	30	0	1	29	16.67	100.00
	Nan	30	12	0	18	30	1	0	29	60.00	96.67
	Muay	30	19	0	11	30	0	0	30	36.67	100.00
	Ging-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Prem	30	26	0	4	30	6	13	11	13.33	80.00
	Fair	30	28	1	1	30	17	8	5	6.67	43.33
	Jui	30	24	0	6	30	0	0	30	20.00	100.00
	Ying	30	14	0	16	30	1	0	29	53.33	96.67
	Kratae-Math	30	26	0	4	30	15	0	15	13.33	50.00
	Jib-Math	30	17	0	13	30	2	1	27	43.33	93.33
	Mazda-Math	30	21	0	9	30	2	1	27	30.00	93.33
	Fern-Math	30	17	0	13	30	0	0	30	43.33	100.00
	Ploy-Math	30	19	0	11	30	1	0	29	36.67	96.67
	Deedo	30	17	0	13	30	0	1	29	43.33	100.00
ผลรวม		600	399	1	200	600	62	35	503	33.50	89.67
เปอร์เซ็น			66.50	0.17	33.33		10.33	5.83	83.83		
ผลรวมของระบบ		1800	1324	2	474	1800	216	94	1490	26.44	88.00
เปอร์เซ็นของระบบ			73.56	0.11	26.33		12.00	5.22	82.78		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	1324	2	474	1800	216	94	1490
FRR	26.44							
FAR	12.00							

ตารางการทดสอบประสิทธิของคำง่าย เมื่อเก็บข้อมูลลงฐานข้อมูล 6 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
ที่อยู่	Jame	30	20	0	10	30	1	5	24	33.33	96.67
	Fang	30	16	0	14	30	2	2	26	46.67	93.33
	Mickey	30	19	0	11	30	0	0	30	36.67	100.00
	May-Math	30	23	0	7	30	0	0	30	23.33	100.00
	Bank	30	26	0	4	30	0	0	30	13.33	100.00
	Khanin	30	24	0	6	30	0	0	30	20.00	100.00
	View	30	28	0	2	30	5	3	22	6.67	83.33
	Nan	30	19	0	11	30	0	0	30	36.67	100.00
	Muay	30	25	0	5	30	0	0	30	16.67	100.00
	Ging-Math	30	27	0	3	30	0	0	30	10.00	100.00
	Prem	30	25	0	5	30	10	2	18	16.67	66.67
	Fair	30	21	0	9	30	3	1	26	30.00	90.00
	Jui	30	26	0	4	30	3	0	27	13.33	90.00
	Ying	30	28	0	2	30	7	5	18	6.67	76.67
	Kratae-Math	30	16	0	14	30	0	0	30	46.67	100.00
	Jib-Math	30	24	0	6	30	1	2	27	20.00	96.67
	Mazda-Math	30	21	0	9	30	0	0	30	30.00	100.00
	Fern-Math	30	27	0	3	30	5	5	20	10.00	83.33
	Ploy-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Deedo	30	28	0	2	30	7	14	9	6.67	76.67
ผลรวม		600	465	0	135	600	44	39	517	22.50	92.67
เปอร์เซ็น			77.50	0.00	22.50		7.33	6.50	86.17		
บ่อยครั้ง	Jame	30	29	0	1	30	12	0	18	3.33	60.00
	Fang	30	28	0	2	30	15	0	15	6.67	50.00
	Mickey	30	26	0	4	30	1	0	29	13.33	96.67
	May-Math	30	26	0	4	30	1	0	29	13.33	96.67
	Bank	30	20	0	10	30	1	0	29	33.33	96.67
	Khanin	30	19	0	11	30	0	0	30	36.67	100.00
	View	30	28	0	2	30	0	0	30	6.67	100.00
	Nan	30	27	0	3	30	4	0	26	10.00	86.67
	Muay	30	25	0	5	30	0	0	30	16.67	100.00
	Ging-Math	30	28	0	2	30	2	0	28	6.67	93.33
	Prem	30	22	0	8	30	13	0	17	26.67	56.67
	Fair	30	19	0	11	30	3	0	27	36.67	90.00
	Jui	30	25	0	5	30	0	0	30	16.67	100.00
	Ying	30	27	0	3	30	2	0	28	10.00	93.33

	Kratae-Math	30	25	0	5	30	3	0	27	16.67	90.00
	Jib-Math	30	27	0	3	30	14	0	16	10.00	53.33
	Mazda-Math	30	23	0	7	30	3	0	27	23.33	90.00
	Fern-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Ploy-Math	30	25	0	5	30	2	0	28	16.67	93.33
	Deedo	30	24	0	6	30	13	0	17	20.00	56.67
ผลรวม		600	495	0	105	600	89	0	511	17.50	85.17
เปอร์เซ็น			82.50	0.00	17.50		14.83	0.00	85.17		
ซ้ำเติม	Jame	30	27	0	3	30	13	1	16	10.00	56.67
	Fang	30	24	0	6	30	7	15	8	20.00	76.67
	Mickey	30	24	0	6	30	0	0	30	20.00	100.00
	May-Math	30	17	0	13	30	1	0	29	43.33	96.67
	Bank	30	25	0	5	30	0	0	30	16.67	100.00
	Khanin	30	23	0	7	30	0	0	30	23.33	100.00
	View	30	24	0	6	30	0	0	30	20.00	100.00
	Nan	30	15	0	15	30	1	0	29	50.00	96.67
	Muay	30	26	0	4	30	0	0	30	13.33	100.00
	Ging-Math	30	24	0	6	30	0	0	30	20.00	100.00
	Prem	30	26	0	4	30	4	15	11	13.33	86.67
	Fair	30	28	1	1	30	18	8	4	6.67	40.00
	Jui	30	26	0	4	30	1	6	23	13.33	96.67
	Ying	30	25	0	5	30	5	0	25	16.67	83.33
	Kratae-Math	30	26	0	4	30	14	0	16	13.33	53.33
	Jib-Math	30	20	0	10	30	8	2	20	33.33	73.33
	Mazda-Math	30	19	0	11	30	1	1	28	36.67	96.67
	Fern-Math	30	28	0	2	30	1	8	21	6.67	96.67
	Ploy-Math	30	24	0	6	30	3	0	27	20.00	90.00
	Deedo	30	23	0	7	30	1	16	13	23.33	96.67
ผลรวม		600	474	1	125	600	78	72	450	21.00	87.00
เปอร์เซ็น			79.00	0.17	20.83		13.00	12.00	75.00		
ผลรวมของระบบ		1800	1434	1	365	1800	211	111	1478	20.33	88.28
เปอร์เซ็นของระบบ			79.67	0.06	20.28		11.72	6.17	82.11		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	1434	1	365	1800	211	111	1478
FRR	20.33							
FAR	11.72							

ตารางการทดสอบประสิทธิของคำง่าย เมื่อเก็บข้อมูลลงฐานข้อมูล 12 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
ที่อยู่	Jame	30	26	0	4	30	1	8	21	13.33	96.67
	Fang	30	16	0	14	30	2	2	26	46.67	93.33
	Mickey	30	29	0	1	30	2	1	27	3.33	93.33
	May-Math	30	12	0	18	30	0	0	30	60.00	100.00
	Bank	30	25	0	5	30	0	1	29	16.67	100.00
	Khanin	30	17	0	13	30	0	0	30	43.33	100.00
	View	30	28	0	2	30	5	3	22	6.67	83.33
	Nan	30	19	0	11	30	0	0	30	36.67	100.00
	Muay	30	25	0	5	30	0	0	30	16.67	100.00
	Ging-Math	30	27	0	3	30	0	0	30	10.00	100.00
	Prem	30	25	0	5	30	10	2	18	16.67	66.67
	Fair	30	16	0	14	30	1	0	29	46.67	96.67
	Jui	30	26	0	4	30	6	2	22	13.33	80.00
	Ying	30	29	0	1	30	7	5	18	3.33	76.67
	Kratae-Math	30	16	0	14	30	0	0	30	46.67	100.00
	Jib-Math	30	25	0	5	30	9	8	13	16.67	70.00
	Mazda-Math	30	23	0	7	30	0	0	30	23.33	100.00
	Fern-Math	30	27	0	3	30	5	4	21	10.00	83.33
	Ploy-Math	30	25	0	5	30	0	0	30	16.67	100.00
	Deedo	30	21	0	9	30	3	9	18	30.00	90.00
ผลรวม		600	457	0	143	600	51	45	504	23.83	91.50
เปอร์เซ็น			76.17	0.00	23.83		8.50	7.50	84.00		
บ่อยครั้ง	Jame	30	29	0	1	30	10	0	20	3.33	66.67
	Fang	30	27	0	3	30	10	0	20	10.00	66.67
	Mickey	30	19	0	11	30	1	0	29	36.67	96.67
	May-Math	30	20	0	10	30	0	0	30	33.33	100.00
	Bank	30	21	0	9	30	0	0	30	30.00	100.00
	Khanin	30	22	0	8	30	0	0	30	26.67	100.00
	View	30	28	0	2	30	0	0	30	6.67	100.00
	Nan	30	28	0	2	30	7	0	23	6.67	76.67
	Muay	30	27	0	3	30	0	0	30	10.00	100.00
	Ging-Math	30	28	0	2	30	2	0	28	6.67	93.33
	Prem	30	26	0	4	30	15	0	15	13.33	50.00
	Fair	30	22	0	8	30	2	0	28	26.67	93.33
	Jui	30	25	0	5	30	0	0	30	16.67	100.00
	Ying	30	28	0	2	30	4	0	26	6.67	86.67

	Kratae-Math	30	21	0	9	30	1	0	29	30.00	96.67
	Jib-Math	30	27	0	3	30	8	0	22	10.00	73.33
	Mazda-Math	30	26	0	4	30	5	0	25	13.33	83.33
	Fern-Math	30	27	0	3	30	2	0	28	10.00	93.33
	Ploy-Math	30	27	0	3	30	4	0	26	10.00	86.67
	Deedo	30	25	0	5	30	17	0	13	16.67	43.33
ผลรวม		600	503	0	97	600	88	0	512	16.17	85.33
เปอร์เซ็น			83.83	0.00	16.17		14.67	0.00	85.33		
ซ้ำเติม	Jame	30	28	0	2	30	12	1	17	6.67	60.00
	Fang	30	26	0	4	30	6	19	5	13.33	80.00
	Mickey	30	28	0	2	30	0	0	30	6.67	100.00
	May-Math	30	14	0	16	30	0	0	30	53.33	100.00
	Bank	30	25	0	5	30	0	0	30	16.67	100.00
	Khanin	30	25	0	5	30	0	0	30	16.67	100.00
	View	30	24	0	6	30	0	0	30	20.00	100.00
	Nan	30	16	0	14	30	1	0	29	46.67	96.67
	Muay	30	25	0	5	30	0	0	30	16.67	100.00
	Ging-Math	30	25	0	5	30	0	0	30	16.67	100.00
	Prem	30	29	0	1	30	8	17	5	3.33	73.33
	Fair	30	28	1	1	30	19	6	5	6.67	36.67
	Jui	30	26	0	4	30	2	8	20	13.33	93.33
	Ying	30	25	0	5	30	3	0	27	16.67	90.00
	Kratae-Math	30	28	0	2	30	15	1	14	6.67	50.00
	Jib-Math	30	22	0	8	30	7	2	21	26.67	76.67
	Mazda-Math	30	23	0	7	30	1	1	28	23.33	96.67
	Fern-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Ploy-Math	30	23	0	7	30	2	0	28	23.33	93.33
	Deedo	30	26	0	4	30	4	20	6	13.33	86.67
ผลรวม		600	488	1	111	600	80	75	445	18.67	86.67
เปอร์เซ็น			81.33	0.17	18.50		13.33	12.50	74.17		
ผลรวมของระบบ		1800	1448	1	351	1800	219	120	1461	19.56	87.83
เปอร์เซ็นของระบบ			80.44	0.06	19.50		12.17	6.67	81.17		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	1448	1	351	1800	219	120	1461
FRR	19.56							
FAR	12.17							

ตารางการทดสอบประสิทธิของคำยาว เมื่อเก็บข้อมูลลงฐานข้อมูล 3 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
พระจอมเกล้า	Jame	30	7	0	23	30	0	0	30	76.67	100
	Fang	30	3	0	27	30	0	0	30	90.00	100
	Mickey	30	14	0	16	30	0	0	30	53.33	100
	May-Math	30	6	0	24	30	1	0	29	80.00	96.66666667
	Bank	30	19	0	11	30	1	0	29	36.67	96.66666667
	Khanin	30	10	0	20	30	0	0	30	66.67	100
	View	30	22	0	8	30	1	0	29	26.67	96.66666667
	Nan	30	14	0	16	30	1	0	29	53.33	96.66666667
	Muay	30	20	0	10	30	2	0	28	33.33	93.33333333
	Ging-Math	30	19	0	11	30	0	0	30	36.67	100
	Prem	30	16	0	14	30	0	0	30	46.67	100
	Fair	30	4	0	26	30	0	0	30	86.67	100
	Jui	30	10	0	20	30	1	0	29	66.67	96.66666667
	Ying	30	4	0	26	30	0	0	30	86.67	100
	Kratae-Math	30	20	0	10	30	7	0	23	33.33	76.66666667
	Jib-Math	30	17	0	13	30	1	0	29	43.33	96.66666667
	Mazda-Math	30	14	0	16	30	0	0	30	53.33	100
	Fern-Math	30	15	0	15	30	0	0	30	50.00	100
	Ploy-Math	30	3	0	27	30	0	0	30	90.00	100
	Deedo	30	2	0	28	30	0	0	30	93.33	100
ผลรวม		600	239	0	361	600	15	0	585	60.17	97.50
เปอร์เซ็น			39.83	0.00	60.17		2.50	0.00	97.50		
คอมพิวเตอร์	Jame	30	7	0	23	30	0	2	28	76.67	100
	Fang	30	7	0	23	30	2	1	27	76.67	93.33333333
	Mickey	30	13	0	17	30	0	0	30	56.67	100
	May-Math	30	17	0	13	30	6	1	23	43.33	80
	Bank	30	24	0	6	30	4	0	26	20.00	86.66666667
	Khanin	30	25	0	5	30	0	0	30	16.67	100
	View	30	23	0	7	30	2	0	28	23.33	93.33333333
	Nan	30	10	0	20	30	6	4	20	66.67	80
	Muay	30	11	0	19	30	0	0	30	63.33	100
	Ging-Math	30	8	0	22	30	0	0	30	73.33	100
	Prem	30	14	0	16	30	2	1	27	53.33	93.33333333
	Fair	30	2	0	28	30	0	0	30	93.33	100
	Jui	30	9	0	21	30	0	0	30	70.00	100
	Ying	30	7	0	23	30	0	0	30	76.67	100

	Kratae-Math	30	3	0	27	30	0	0	30	90.00	100
	Jib-Math	30	26	0	4	30	0	0	30	13.33	100
	Mazda-Math	30	15	0	15	30	0	0	30	50.00	100
	Fern-Math	30	22	0	8	30	1	0	29	26.67	96.66666667
	Ploy-Math	30	9	0	21	30	0	1	29	70.00	100
	Deedo	30	6	0	24	30	0	0	30	80.00	100
ผลรวม		600	258	0	342	600	23	10	567	57.00	96.17
เปอร์เซ็น			43.00	0.00	57.00		3.83	1.67	94.50		
อนุญาต	Jame	30	22	0	8	30	2	0	28	26.67	93.33333333
	Fang	30	8	0	22	30	0	1	29	73.33	100
	Mickey	30	4	0	26	30	0	0	30	86.67	100
	May-Math	30	10	0	20	30	0	0	30	66.67	100
	Bank	30	18	0	12	30	0	0	30	40.00	100
	Khanin	30	27	0	3	30	0	1	29	10.00	100
	View	30	9	0	21	30	0	0	30	70.00	100
	Nan	30	9	0	21	30	0	0	30	70.00	100
	Muay	30	20	0	10	30	0	0	30	33.33	100
	Ging-Math	30	5	0	25	30	0	0	30	83.33	100
	Prem	30	4	0	26	30	0	0	30	86.67	100
	Fair	30	3	1	26	30	2	2	26	90.00	93.33333333
	Jui	30	22	0	8	30	0	2	28	26.67	100
	Ying	30	8	0	22	30	1	0	29	73.33	96.66666667
	Kratae-Math	30	14	0	16	30	0	0	30	53.33	100
	Jib-Math	30	15	0	15	30	0	0	30	50.00	100
	Mazda-Math	30	6	0	24	30	0	0	30	80.00	100
	Fern-Math	30	21	0	9	30	0	2	28	30.00	100
	Ploy-Math	30	22	0	8	30	1	0	29	26.67	96.66666667
	Deedo	30	15	0	15	30	0	0	30	50.00	100
ผลรวม		600	262	1	337	600	6	8	586	56.33	99.00
เปอร์เซ็น			43.67	0.17	56.17		1.00	1.33	97.67		
ผลรวมของระบบ		1800	759	1	1040	1800	44	18	1738	57.83	97.56
เปอร์เซ็นของระบบ			42.17	0.06	57.78		2.44	1.00	96.56		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	759	1	1040	1800	44	18	1738
FRR	57.83							
FAR	2.44							

ตารางการทดสอบประสิทธิของคำยาว เมื่อเก็บข้อมูลลงฐานข้อมูล 6 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
พระจอมเกล้า	Jame	30	14	0	16	30	0	0	30	53.33	100.00
	Fang	30	11	0	19	30	2	0	28	63.33	93.33
	Mickey	30	18	0	12	30	0	0	30	40.00	100.00
	May-Math	30	28	0	2	30	5	0	25	6.67	83.33
	Bank	30	28	0	2	30	5	0	25	6.67	83.33
	Khanin	30	23	0	7	30	1	0	29	23.33	96.67
	View	30	22	0	8	30	0	0	30	26.67	100.00
	Nan	30	23	0	7	30	3	0	27	23.33	90.00
	Muay	30	26	0	4	30	5	0	25	13.33	83.33
	Ging-Math	30	26	0	4	30	0	0	30	13.33	100.00
	Prem	30	27	0	3	30	15	0	15	10.00	50.00
	Fair	30	19	0	11	30	2	0	28	36.67	93.33
	Jui	30	22	0	8	30	13	0	17	26.67	56.67
	Ying	30	15	0	15	30	0	0	30	50.00	100.00
	Kratae-Math	30	21	0	9	30	4	0	26	30.00	86.67
	Jib-Math	30	22	0	8	30	9	0	21	26.67	70.00
	Mazda-Math	30	18	0	12	30	1	0	29	40.00	96.67
	Fern-Math	30	9	0	21	30	0	0	30	70.00	100.00
	Ploy-Math	30	24	0	6	30	0	0	30	20.00	100.00
	Deedo	30	24	0	6	30	4	0	26	20.00	86.67
ผลรวม		600	420	0	180	600	69	0	531	30.00	88.50
เปอร์เซ็น			70.00	0.00	30.00		11.50	0.00	88.50		
คอมพิวเตอร์	Jame	30	27	0	3	30	10	3	17	10.00	66.67
	Fang	30	11	0	19	30	3	2	25	63.33	90.00
	Mickey	30	17	0	13	30	0	0	30	43.33	100.00
	May-Math	30	25	0	5	30	7	0	23	16.67	76.67
	Bank	30	21	0	9	30	0	0	30	30.00	100.00
	Khanin	30	21	0	9	30	0	0	30	30.00	100.00
	View	30	18	0	12	30	0	0	30	40.00	100.00
	Nan	30	14	0	16	30	11	3	16	53.33	63.33
	Muay	30	26	0	4	30	0	0	30	13.33	100.00
	Ging-Math	30	21	0	9	30	4	0	26	30.00	86.67
	Prem	30	23	0	7	30	3	5	22	23.33	90.00
	Fair	30	16	0	14	30	1	0	29	46.67	96.67
	Jui	30	16	0	14	30	2	0	28	46.67	93.33
	Ying	30	21	0	9	30	0	0	30	30.00	100.00

	Kratae-Math	30	26	0	4	30	10	2	18	13.33	66.67
	Jib-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Mazda-Math	30	11	0	19	30	0	0	30	63.33	100.00
	Fern-Math	30	22	0	8	30	1	0	29	26.67	96.67
	Ploy-Math	30	19	0	11	30	0	2	28	36.67	100.00
	Deedo	30	24	0	6	30	2	9	19	20.00	93.33
ผลรวม		600	401	0	199	600	54	26	520	33.17	91.00
เปอร์เซ็น			66.83	0.00	33.17		9.00	4.33	86.67		
อนุญาต	Jame	30	17	0	13	30	0	0	30	43.33	100.00
	Fang	30	25	0	5	30	1	10	19	16.67	96.67
	Mickey	30	16	0	14	30	0	0	30	46.67	100.00
	May-Math	30	25	0	5	30	2	0	28	16.67	93.33
	Bank	30	24	0	6	30	1	0	29	20.00	96.67
	Khanin	30	19	0	11	30	0	0	30	36.67	100.00
	View	30	23	0	7	30	0	0	30	23.33	100.00
	Nan	30	11	0	19	30	0	0	30	63.33	100.00
	Muay	30	20	0	10	30	0	0	30	33.33	100.00
	Ging-Math	30	20	0	10	30	1	0	29	33.33	96.67
	Prem	30	24	0	6	30	0	3	27	20.00	100.00
	Fair	30	18	1	11	30	1	4	25	40.00	96.67
	Jui	30	24	0	6	30	0	5	25	20.00	100.00
	Ying	30	27	0	3	30	4	0	26	10.00	86.67
	Kratae-Math	30	30	0	0	30	7	0	23	0.00	76.67
	Jib-Math	30	23	0	7	30	8	0	22	23.33	73.33
	Mazda-Math	30	27	0	3	30	0	1	29	10.00	100.00
	Fern-Math	30	18	0	12	30	0	1	29	40.00	100.00
	Ploy-Math	30	17	0	13	30	0	0	30	43.33	100.00
	Deedo	30	15	0	15	30	0	0	30	50.00	100.00
ผลรวม		600	423	1	176	600	25	24	551	29.50	95.83
เปอร์เซ็น			70.50	0.17	29.33		4.17	4.00	91.83		
ผลรวมของระบบ		1800	1244	1	555	1800	148	50	1602	30.89	91.78
เปอร์เซ็นของระบบ			69.11	0.06	30.83		8.22	2.78	89.00		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	1244	1	555	1800	148	50	1602
FRR	30.89							
FAR	8.22							

ตารางการทดสอบประสิทธิของคำยาว เมื่อเก็บข้อมูลลงฐานข้อมูล 9 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
พระจอมเกล้า	Jame	30	26	0	4	30	7	0	23	13.33	76.67
	Fang	30	15	0	15	30	2	0	28	50.00	93.33
	Mickey	30	23	0	7	30	0	0	30	23.33	100.00
	May-Math	30	25	0	5	30	1	0	29	16.67	96.67
	Bank	30	28	0	2	30	6	0	24	6.67	80.00
	Khanin	30	21	0	9	30	0	0	30	30.00	100.00
	View	30	24	0	6	30	1	0	29	20.00	96.67
	Nan	30	28	0	2	30	5	0	25	6.67	83.33
	Muay	30	25	0	5	30	4	0	26	16.67	86.67
	Ging-Math	30	27	0	3	30	0	0	30	10.00	100.00
	Prem	30	24	0	6	30	1	0	29	20.00	96.67
	Fair	30	26	0	4	30	8	0	22	13.33	73.33
	Jui	30	20	0	10	30	10	0	20	33.33	66.67
	Ying	30	21	0	9	30	0	0	30	30.00	100.00
	Kratae-Math	30	28	0	2	30	11	0	19	6.67	63.33
	Jib-Math	30	29	0	1	30	16	0	14	3.33	46.67
	Mazda-Math	30	17	0	13	30	1	0	29	43.33	96.67
	Fern-Math	30	18	0	12	30	0	0	30	40.00	100.00
	Ploy-Math	30	25	0	5	30	2	0	28	16.67	93.33
	Deedo	30	29	0	1	30	11	0	19	3.33	63.33
ผลรวม		600	479	0	121	600	86	0	514	20.17	85.67
เปอร์เซ็น			79.83	0.00	20.17		14.33	0.00	85.67		
คอมพิวเตอร์	Jame	30	27	0	3	30	9	3	18	10.00	70.00
	Fang	30	19	0	11	30	7	4	19	36.67	76.67
	Mickey	30	18	0	12	30	0	0	30	40.00	100.00
	May-Math	30	28	0	2	30	6	0	24	6.67	80.00
	Bank	30	23	0	7	30	2	0	28	23.33	93.33
	Khanin	30	23	0	7	30	0	0	30	23.33	100.00
	View	30	15	0	15	30	0	0	30	50.00	100.00
	Nan	30	13	0	17	30	11	2	17	56.67	63.33
	Muay	30	26	0	4	30	0	0	30	13.33	100.00
	Ging-Math	30	26	0	4	30	5	0	25	13.33	83.33
	Prem	30	21	0	9	30	3	8	19	30.00	90.00
	Fair	30	20	0	10	30	0	2	28	33.33	100.00
	Jui	30	28	0	2	30	3	7	20	6.67	90.00
	Ying	30	27	0	3	30	0	0	30	10.00	100.00

	Kratae-Math	30	24	0	6	30	3	0	27	20.00	90.00
	Jib-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Mazda-Math	30	16	0	14	30	0	0	30	46.67	100.00
	Fern-Math	30	26	0	4	30	1	0	29	13.33	96.67
	Ploy-Math	30	23	0	7	30	0	2	28	23.33	100.00
	Deedo	30	27	0	3	30	5	14	11	10.00	83.33
ผลรวม		600	452	0	148	600	55	42	503	24.67	90.83
เปอร์เซ็น			75.33	0.00	24.67		9.17	7.00	83.83		
อนุญาต	Jame	30	28	0	2	30	13	0	17	6.67	56.67
	Fang	30	27	0	3	30	0	9	21	10.00	100.00
	Mickey	30	23	0	7	30	0	0	30	23.33	100.00
	May-Math	30	27	0	3	30	2	0	28	10.00	93.33
	Bank	30	25	0	5	30	0	0	30	16.67	100.00
	Khanin	30	24	0	6	30	0	1	29	20.00	100.00
	View	30	23	0	7	30	0	1	29	23.33	100.00
	Nan	30	14	0	16	30	0	0	30	53.33	100.00
	Muay	30	24	0	6	30	0	0	30	20.00	100.00
	Ging-Math	30	22	0	8	30	1	0	29	26.67	96.67
	Prem	30	23	0	7	30	0	3	27	23.33	100.00
	Fair	30	25	0	5	30	1	4	25	16.67	96.67
	Jui	30	24	0	6	30	0	3	27	20.00	100.00
	Ying	30	28	0	2	30	5	0	25	6.67	83.33
	Kratae-Math	30	25	0	5	30	0	0	30	16.67	100.00
	Jib-Math	30	22	0	8	30	6	0	24	26.67	80.00
	Mazda-Math	30	23	0	7	30	0	0	30	23.33	100.00
	Fern-Math	30	22	0	8	30	0	1	29	26.67	100.00
	Ploy-Math	30	25	0	5	30	2	1	27	16.67	93.33
	Deedo	30	21	0	9	30	0	0	30	30.00	100.00
ผลรวม		600	475	0	125	600	30	23	547	20.83	95.00
เปอร์เซ็น			79.17	0.00	20.83		5.00	3.83	91.17		
ผลรวมของระบบ		1800	1406	0	394	1800	171	65	1564	21.89	90.50
เปอร์เซ็นของระบบ			78.11	0.00	21.89		9.50	3.61	86.89		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	1406	0	394	1800	171	65	1564
FRR	21.89							
FAR	9.50							

ตารางการทดสอบประสิทธิของคำยาว เมื่อเก็บข้อมูลลงฐานข้อมูล 12 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
พระจอมเกล้า	Jame	30	28	0	2	30	15	0	15	6.67	50.00
	Fang	30	24	0	6	30	6	0	24	20.00	80.00
	Mickey	30	24	0	6	30	0	0	30	20.00	100.00
	May-Math	30	15	0	15	30	0	0	30	50.00	100.00
	Bank	30	27	0	3	30	3	0	27	10.00	90.00
	Khanin	30	21	0	9	30	1	0	29	30.00	96.67
	View	30	24	0	6	30	2	0	28	20.00	93.33
	Nan	30	28	0	2	30	7	0	23	6.67	76.67
	Muay	30	27	0	3	30	6	0	24	10.00	80.00
	Ging-Math	30	26	0	4	30	0	0	30	13.33	100.00
	Prem	30	26	0	4	30	10	0	20	13.33	66.67
	Fair	30	27	0	3	30	8	0	22	10.00	73.33
	Jui	30	22	0	8	30	10	0	20	26.67	66.67
	Ying	30	21	0	9	30	0	0	30	30.00	100.00
	Kratae-Math	30	26	0	4	30	12	0	18	13.33	60.00
	Jib-Math	30	25	0	5	30	11	0	19	16.67	63.33
	Mazda-Math	30	25	0	5	30	2	0	28	16.67	93.33
	Fern-Math	30	25	0	5	30	0	0	30	16.67	100.00
	Ploy-Math	30	29	0	1	30	9	0	21	3.33	70.00
	Deedo	30	28	0	2	30	10	0	20	6.67	66.67
ผลรวม		600	498	0	102	600	112	0	488	17.00	81.33
เปอร์เซ็น			83.00	0.00	17.00		18.67	0.00	81.33		
คอมพิวเตอร์	Jame	30	21	0	9	30	3	2	25	30.00	90.00
	Fang	30	26	0	4	30	8	8	14	13.33	73.33
	Mickey	30	23	0	7	30	0	0	30	23.33	100.00
	May-Math	30	28	0	2	30	5	0	25	6.67	83.33
	Bank	30	24	0	6	30	2	0	28	20.00	93.33
	Khanin	30	25	0	5	30	0	0	30	16.67	100.00
	View	30	19	0	11	30	0	0	30	36.67	100.00
	Nan	30	14	0	16	30	11	2	17	53.33	63.33
	Muay	30	25	0	5	30	0	0	30	16.67	100.00
	Ging-Math	30	28	0	2	30	6	0	24	6.67	80.00
	Prem	30	26	0	4	30	5	10	15	13.33	83.33
	Fair	30	21	0	9	30	0	2	28	30.00	100.00
	Jui	30	24	0	6	30	2	5	23	20.00	93.33
	Ying	30	26	0	4	30	0	0	30	13.33	100.00

	Kratae-Math	30	23	0	7	30	5	2	23	23.33	83.33
	Jib-Math	30	28	0	2	30	0	0	30	6.67	100.00
	Mazda-Math	30	19	0	11	30	0	0	30	36.67	100.00
	Fern-Math	30	22	0	8	30	1	0	29	26.67	96.67
	Ploy-Math	30	29	0	1	30	4	4	22	3.33	86.67
	Deedo	30	27	0	3	30	5	14	11	10.00	83.33
ผลรวม		600	478	0	122	600	57	49	494	20.33	90.50
เปอร์เซ็น			79.67	0.00	20.33		9.50	8.17	82.33		
อนุญาต	Jame	30	28	0	2	30	7	0	23	6.67	76.67
	Fang	30	27	0	3	30	0	5	25	10.00	100.00
	Mickey	30	18	0	12	30	0	0	30	40.00	100.00
	May-Math	30	27	0	3	30	3	0	27	10.00	90.00
	Bank	30	27	0	3	30	0	0	30	10.00	100.00
	Khanin	30	26	0	4	30	0	1	29	13.33	100.00
	View	30	25	0	5	30	0	1	29	16.67	100.00
	Nan	30	14	0	16	30	0	0	30	53.33	100.00
	Muay	30	28	0	2	30	0	3	27	6.67	100.00
	Ging-Math	30	24	0	6	30	0	0	30	20.00	100.00
	Prem	30	25	0	5	30	0	4	26	16.67	100.00
	Fair	30	27	0	3	30	1	8	21	10.00	96.67
	Jui	30	28	0	2	30	2	16	12	6.67	93.33
	Ying	30	28	0	2	30	7	0	23	6.67	76.67
	Kratae-Math	30	24	0	6	30	1	0	29	20.00	96.67
	Jib-Math	30	20	0	10	30	4	0	26	33.33	86.67
	Mazda-Math	30	28	0	2	30	0	1	29	6.67	100.00
	Fern-Math	30	23	0	7	30	0	2	28	23.33	100.00
	Ploy-Math	30	24	0	6	30	1	0	29	20.00	96.67
	Deedo	30	24	0	6	30	0	6	24	20.00	100.00
ผลรวม		600	495	0	105	600	26	47	527	17.50	95.67
เปอร์เซ็น			82.50	0.00	17.50		4.33	7.83	87.83		
ผลรวมของระบบ		1800	1471	0	329	1800	195	96	1509	18.28	89.17
เปอร์เซ็นของระบบ			81.72	0.00	18.28		10.83	5.33	83.83		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	1471	0	329	1800	195	96	1509
FRR	18.28							
FAR	10.83							

ตารางการทดสอบประสิทธิของคำยาก เมื่อเก็บข้อมูลลงฐานข้อมูล 3 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
ศรัทธา	Jame	30	24	0	6	30	6	0	24	20.00	80.00
	Fang	30	11	0	19	30	9	4	17	63.33	70.00
	Mickey	30	15	0	15	30	0	0	30	50.00	100.00
	May-Math	30	4	0	26	30	0	0	30	86.67	100.00
	Bank	30	8	0	22	30	0	0	30	73.33	100.00
	Khanin	30	7	0	23	30	0	0	30	76.67	100.00
	View	30	11	0	19	30	0	0	30	63.33	100.00
	Nan	30	4	0	26	30	3	0	27	86.67	90.00
	Muay	30	24	0	6	30	0	2	28	20.00	100.00
	Ging-Math	30	9	0	21	30	1	0	29	70.00	96.67
	Prem	30	25	0	5	30	2	4	24	16.67	93.33
	Fair	30	4	0	26	30	0	0	30	86.67	100.00
	Jui	30	9	0	21	30	0	0	30	70.00	100.00
	Ying	30	18	0	12	30	0	0	30	40.00	100.00
	Kratae-Math	30	3	0	27	30	0	0	30	90.00	100.00
	Jib-Math	30	8	0	22	30	0	0	30	73.33	100.00
	Mazda-Math	30	20	0	10	30	7	1	22	33.33	76.67
	Fern-Math	30	8	0	22	30	0	0	30	73.33	100.00
	Ploy-Math	30	23	0	7	30	1	0	29	23.33	96.67
	Deedo	30	4	0	26	30	0	0	30	86.67	100.00
ผลรวม		600	239	0	361	600	29	11	560	60.17	95.17
เปอร์เซ็น			39.83	0.00	60.17		4.83	1.83	93.33		
สาบสูญ	Jame	30	25	0	5	30	0	0	30	16.67	100.00
	Fang	30	2	0	28	30	0	0	30	93.33	100.00
	Mickey	30	26	0	4	30	0	2	28	13.33	100.00
	May-Math	30	6	0	24	30	0	0	30	80.00	100.00
	Bank	30	6	0	24	30	0	0	30	80.00	100.00
	Khanin	30	26	0	4	30	0	0	30	13.33	100.00
	View	30	7	0	23	30	0	0	30	76.67	100.00
	Nan	30	7	0	23	30	0	0	30	76.67	100.00
	Muay	30	13	0	17	30	0	0	30	56.67	100.00
	Ging-Math	30	17	0	13	30	0	0	30	43.33	100.00
	Prem	30	11	0	19	30	1	0	29	63.33	96.67
	Fair	30	7	0	23	30	0	0	30	76.67	100.00
	Jui	30	1	0	29	30	0	0	30	96.67	100.00
	Ying	30	21	0	9	30	0	0	30	30.00	100.00

	Kratae-Math	30	21	0	9	30	0	0	30	30.00	100.00
	Jib-Math	30	5	0	25	30	0	0	30	83.33	100.00
	Mazda-Math	30	8	0	22	30	0	0	30	73.33	100.00
	Fern-Math	30	10	0	20	30	0	0	30	66.67	100.00
	Ploy-Math	30	0	0	30	30	0	0	30	100.00	100.00
	Deedo	30	10	0	20	30	0	4	26	66.67	100.00
ผลรวม		600	229	0	371	600	1	6	593	61.83	99.83
เปอร์เซ็น			38.17	0.00	61.83		0.17	1.00	98.83		
ปฐมนิเทศ	Jame	30	5	0	25	30	0	0	30	83.33	100.00
	Fang	30	3	0	27	30	0	0	30	90.00	100.00
	Mickey	30	6	0	24	30	0	0	30	80.00	100.00
	May-Math	30	7	0	23	30	0	0	30	76.67	100.00
	Bank	30	2	0	28	30	0	0	30	93.33	100.00
	Khanin	30	22	0	8	30	0	2	28	26.67	100.00
	View	30	18	0	12	30	0	0	30	40.00	100.00
	Nan	30	3	0	27	30	0	0	30	90.00	100.00
	Muay	30	18	0	12	30	0	0	30	40.00	100.00
	Ging-Math	30	2	0	28	30	0	0	30	93.33	100.00
	Prem	30	29	0	1	30	1	5	24	3.33	96.67
	Fair	30	17	0	13	30	0	0	30	43.33	100.00
	Jui	30	12	0	18	30	0	0	30	60.00	100.00
	Ying	30	6	0	24	30	0	0	30	80.00	100.00
	Kratae-Math	30	20	0	10	30	0	0	30	33.33	100.00
	Jib-Math	30	18	0	12	30	0	0	30	40.00	100.00
	Mazda-Math	30	5	0	25	30	0	0	30	83.33	100.00
	Fern-Math	30	17	0	13	30	0	0	30	43.33	100.00
	Ploy-Math	30	23	0	7	30	0	0	30	23.33	100.00
	Deedo	30	10	0	20	30	0	0	30	66.67	100.00
ผลรวม		600	243	0	357	600	1	7	592	59.50	99.83
เปอร์เซ็น			40.50	0.00	59.50		0.17	1.17	98.67		
ผลรวมของระบบ		1800	711	0	1089	1800	31	24	1745	60.50	98.28
เปอร์เซ็นของระบบ			39.50	0.00	60.50		1.72	1.33	96.94		

Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
		Character Fail	Fail			Character Fail	Fail
1800	711	0	1089	1800	31	24	1745

FRR	60.50
FAR	1.72

ตารางการทดสอบประสิทธิของคำยาก เมื่อเก็บข้อมูลลงฐานข้อมูล 6 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
ศรัทธา	Jame	30	23	0	7	30	4	0	26	23.33	86.67
	Fang	30	15	0	15	30	9	5	16	50.00	70.00
	Mickey	30	19	0	11	30	0	0	30	36.67	100.00
	May-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Bank	30	22	0	8	30	1	0	29	26.67	96.67
	Khanin	30	20	0	10	30	0	0	30	33.33	100.00
	View	30	30	0	0	30	7	15	8	0.00	76.67
	Nan	30	12	0	18	30	3	1	26	60.00	90.00
	Muay	30	29	0	1	30	0	8	22	3.33	100.00
	Ging-Math	30	23	0	7	30	2	0	28	23.33	93.33
	Prem	30	28	0	2	30	3	11	16	6.67	90.00
	Fair	30	26	0	4	30	0	3	27	13.33	100.00
	Jui	30	22	0	8	30	0	0	30	26.67	100.00
	Ying	30	26	0	4	30	0	0	30	13.33	100.00
	Kratae-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Jib-Math	30	12	0	18	30	0	0	30	60.00	100.00
	Mazda-Math	30	24	0	6	30	5	0	25	20.00	83.33
	Fern-Math	30	27	0	3	30	3	3	24	10.00	90.00
	Ploy-Math	30	12	0	18	30	0	0	30	60.00	100.00
	Deedo	30	24	0	6	30	5	17	8	20.00	83.33
ผลรวม		600	438	0	162	600	42	63	495	27.00	93.00
เปอร์เซ็น			73.00	0.00	27.00		7.00	10.50	82.50		
สาบสูญ	Jame	30	24	0	6	30	0	0	30	20.00	100.00
	Fang	30	5	0	25	30	1	1	28	83.33	96.67
	Mickey	30	22	0	8	30	0	0	30	26.67	100.00
	May-Math	30	12	0	18	30	0	0	30	60.00	100.00
	Bank	30	18	0	12	30	0	0	30	40.00	100.00
	Khanin	30	19	0	11	30	0	0	30	36.67	100.00
	View	30	23	0	7	30	0	1	29	23.33	100.00
	Nan	30	18	0	12	30	2	0	28	40.00	93.33
	Muay	30	28	0	2	30	0	2	28	6.67	100.00
	Ging-Math	30	18	0	12	30	0	0	30	40.00	100.00
	Prem	30	17	0	13	30	0	0	30	43.33	100.00
	Fair	30	19	0	11	30	0	0	30	36.67	100.00
	Jui	30	12	0	18	30	2	1	27	60.00	93.33
	Ying	30	27	0	3	30	2	1	27	10.00	93.33

	Kratae-Math	30	9	0	21	30	0	0	30	70.00	100.00
	Jib-Math	30	22	0	8	30	2	0	28	26.67	93.33
	Mazda-Math	30	18	0	12	30	0	0	30	40.00	100.00
	Fern-Math	30	17	0	13	30	0	0	30	43.33	100.00
	Ploy-Math	30	10	0	20	30	0	0	30	66.67	100.00
	Deedo	30	17	0	13	30	2	0	28	43.33	93.33
ผลรวม		600	355	0	245	600	11	6	583	40.83	98.17
เปอร์เซ็น			59.17	0.00	40.83		1.83	1.00	97.17		
ปฐมนิเทศ	Jame	30	24	0	6	30	4	0	26	20.00	86.67
	Fang	30	9	0	21	30	0	0	30	70.00	100.00
	Mickey	30	25	0	5	30	0	0	30	16.67	100.00
	May-Math	30	21	0	9	30	0	0	30	30.00	100.00
	Bank	30	24	0	6	30	1	0	29	20.00	96.67
	Khanin	30	22	0	8	30	0	1	29	26.67	100.00
	View	30	17	0	13	30	0	0	30	43.33	100.00
	Nan	30	10	0	20	30	0	0	30	66.67	100.00
	Muay	30	17	0	13	30	0	0	30	43.33	100.00
	Ging-Math	30	20	0	10	30	1	0	29	33.33	96.67
	Prem	30	24	0	6	30	0	0	30	20.00	100.00
	Fair	30	22	0	8	30	0	0	30	26.67	100.00
	Jui	30	20	0	10	30	0	0	30	33.33	100.00
	Ying	30	23	0	7	30	2	1	27	23.33	93.33
	Kratae-Math	30	25	0	5	30	2	0	28	16.67	93.33
	Jib-Math	30	10	0	20	30	1	0	29	66.67	96.67
	Mazda-Math	30	24	0	6	30	0	0	30	20.00	100.00
	Fern-Math	30	28	0	2	30	1	5	24	6.67	96.67
	Ploy-Math	30	19	0	11	30	0	0	30	36.67	100.00
	Deedo	30	29	0	1	30	6	20	4	3.33	80.00
ผลรวม		600	413	0	187	600	18	27	555	31.17	97.00
เปอร์เซ็น			68.83	0.00	31.17		3.00	4.50	92.50		
ผลรวมของระบบ		1800	1206	0	594	1800	71	96	1633	33.00	96.06
เปอร์เซ็นของระบบ			67.00	0.00	33.00		3.94	5.33	90.72		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	1206	0	594	1800	71	96	1633
FRR	33.00							
FAR	3.94							

ตารางการทดสอบประสิทธิของคำยาก เมื่อเก็บข้อมูลลงฐานข้อมูล 9 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
ศรัทธา	Jame	30	25	0	5	30	2	0	28	16.66666667	93.33
	Fang	30	15	0	15	30	9	11	10	50	70.00
	Mickey	30	23	0	7	30	0	0	30	23.33333333	100.00
	May-Math	30	20	0	10	30	0	0	30	33.33333333	100.00
	Bank	30	21	0	9	30	0	0	30	30	100.00
	Khanin	30	18	0	12	30	0	0	30	40	100.00
	View	30	27	0	3	30	0	0	30	10	100.00
	Nan	30	25	0	5	30	2	1	27	16.66666667	93.33
	Muay	30	26	0	4	30	0	0	30	13.33333333	100.00
	Ging-Math	30	28	0	2	30	3	0	27	6.666666667	90.00
	Prem	30	22	0	8	30	1	1	28	26.66666667	96.67
	Fair	30	25	0	5	30	0	0	30	16.66666667	100.00
	Jui	30	21	0	9	30	0	0	30	30	100.00
	Ying	30	25	0	5	30	1	0	29	16.66666667	96.67
	Kratae-Math	30	24	0	6	30	0	0	30	20	100.00
	Jib-Math	30	19	0	11	30	1	0	29	36.66666667	96.67
	Mazda-Math	30	24	0	6	30	6	1	23	20	80.00
	Fern-Math	30	28	0	2	30	4	6	20	6.666666667	86.67
	Ploy-Math	30	23	0	7	30	0	0	30	23.33333333	100.00
	Deedo	30	15	0	15	30	0	0	30	50	100.00
ผลรวม		600	454	0	146	600	29	20	551	24.33	95.17
เปอร์เซ็น			75.67	0.00	24.33		4.83	3.33	91.83		
สาบสูญ	Jame	30	27	0	3	30	0	0	30	10	100.00
	Fang	30	12	0	18	30	6	6	18	60	80.00
	Mickey	30	22	0	8	30	0	0	30	26.66666667	100.00
	May-Math	30	21	0	9	30	0	0	30	30	100.00
	Bank	30	20	0	10	30	0	0	30	33.33333333	100.00
	Khanin	30	26	0	4	30	0	0	30	13.33333333	100.00
	View	30	24	0	6	30	0	1	29	20	100.00
	Nan	30	18	0	12	30	0	0	30	40	100.00
	Muay	30	24	0	6	30	0	1	29	20	100.00
	Ging-Math	30	22	0	8	30	0	0	30	26.66666667	100.00
	Prem	30	26	0	4	30	1	1	28	13.33333333	96.67
	Fair	30	27	0	3	30	0	1	29	10	100.00
	Jui	30	8	0	22	30	0	2	28	73.33333333	100.00
	Ying	30	27	0	3	30	1	0	29	10	96.67

	Kratae-Math	30	22	0	8	30	3	0	27	26.66666667	90.00
	Jib-Math	30	23	0	7	30	2	1	27	23.33333333	93.33
	Mazda-Math	30	21	0	9	30	0	0	30	30	100.00
	Fern-Math	30	28	0	2	30	2	3	25	6.666666667	93.33
	Ploy-Math	30	19	0	11	30	0	0	30	36.66666667	100.00
	Deedo	30	15	0	15	30	2	11	17	50	93.33
ผลรวม		600	432	0	168	600	17	27	556	28.00	97.17
เปอร์เซ็น			72.00	0.00	28.00		2.83	4.50	92.67		
	Jame	30	21	0	9	30	0	0	30	30	100.00
	Fang	30	9	1	20	30	6	0	24	70	80.00
	Mickey	30	24	0	6	30	0	0	30	20	100.00
	May-Math	30	24	0	6	30	0	0	30	20	100.00
	Bank	30	20	0	10	30	0	0	30	33.33333333	100.00
	Khanin	30	22	0	8	30	0	2	28	26.66666667	100.00
	View	30	24	0	6	30	0	2	28	20	100.00
	Nan	30	22	0	8	30	0	0	30	26.66666667	100.00
	Muay	30	26	0	4	30	0	0	30	13.33333333	100.00
	Ging-Math	30	25	0	5	30	0	0	30	16.66666667	100.00
ปฐมนิเทศ	Prem	30	21	0	9	30	1	0	29	30	96.67
	Fair	30	20	0	10	30	0	0	30	33.33333333	100.00
	Jui	30	25	0	5	30	0	8	22	16.66666667	100.00
	Ying	30	21	0	9	30	1	0	29	30	96.67
	Kratae-Math	30	28	0	2	30	3	3	24	6.666666667	90.00
	Jib-Math	30	23	0	7	30	2	1	27	23.33333333	93.33
	Mazda-Math	30	20	0	10	30	0	0	30	33.33333333	100.00
	Fern-Math	30	22	0	8	30	2	0	28	26.66666667	93.33
	Ploy-Math	30	20	0	10	30	0	0	30	33.33333333	100.00
	Deedo	30	22	0	8	30	2	15	13	26.66666667	93.33
ผลรวม		600	439	1	160	600	17	31	552	26.83	97.17
เปอร์เซ็น			73.17	0.17	26.67		2.83	5.17	92.00		
ผลรวมของระบบ		1800	1325	1	474	1800	63	78	1659	26.39	96.50
เปอร์เซ็นของระบบ			73.61	0.06	26.33		3.50	4.33	92.17		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	1325	1	474	1800	63	78	1659
FRR	26.39							
FAR	3.50							

ตารางการทดสอบประสิทธิของคำยาก เมื่อเก็บข้อมูลลงฐานข้อมูล 12 ครั้ง

Word	User	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail		FRR	FAR
				Character Fail	Fail			Character Fail	Fail		
ศรัทธา	Jame	30	23	0	7	30	2	0	28	23.33	93.33
	Fang	30	24	0	6	30	9	11	10	20.00	70.00
	Mickey	30	28	0	2	30	0	0	30	6.67	100.00
	May-Math	30	21	0	9	30	0	0	30	30.00	100.00
	Bank	30	19	0	11	30	0	0	30	36.67	100.00
	Khanin	30	19	0	11	30	0	0	30	36.67	100.00
	View	30	30	0	0	30	5	12	13	0.00	83.33
	Nan	30	28	0	2	30	3	1	26	6.67	90.00
	Muay	30	29	0	1	30	0	5	25	3.33	100.00
	Ging-Math	30	28	0	2	30	2	0	28	6.67	93.33
	Prem	30	23	0	7	30	2	2	26	23.33	93.33
	Fair	30	26	0	4	30	0	0	30	13.33	100.00
	Jui	30	22	0	8	30	0	0	30	26.67	100.00
	Ying	30	28	0	2	30	4	0	26	6.67	86.67
	Kratae-Math	30	23	0	7	30	0	0	30	23.33	100.00
	Jib-Math	30	20	0	10	30	1	0	29	33.33	96.67
	Mazda-Math	30	25	0	5	30	5	1	24	16.67	83.33
	Fern-Math	30	28	0	2	30	5	10	15	6.67	83.33
	Ploy-Math	30	21	0	9	30	0	0	30	30.00	100.00
	Deedo	30	23	0	7	30	5	16	9	23.33	83.33
ผลรวม		600	488	0	112	600	43	58	499	18.67	92.83
เปอร์เซ็น			81.33	0.00	18.67		7.17	9.67	83.17		
สาบสูญ	Jame	30	21	0	9	30	0	0	30	30.00	100.00
	Fang	30	21	0	9	30	7	13	10	30.00	76.67
	Mickey	30	25	0	5	30	0	0	30	16.67	100.00
	May-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Bank	30	24	0	6	30	0	0	30	20.00	100.00
	Khanin	30	24	0	6	30	0	0	30	20.00	100.00
	View	30	29	0	1	30	2	7	21	3.33	93.33
	Nan	30	21	0	9	30	0	0	30	30.00	100.00
	Muay	30	27	0	3	30	0	1	29	10.00	100.00
	Ging-Math	30	22	0	8	30	0	0	30	26.67	100.00
	Prem	30	24	0	6	30	2	0	28	20.00	93.33
	Fair	30	27	0	3	30	0	1	29	10.00	100.00
	Jui	30	17	3	10	30	2	5	23	43.33	93.33
	Ying	30	26	0	4	30	1	0	29	13.33	96.67

	Kratae-Math	30	23	0	7	30	2	0	28	23.33	93.33
	Jib-Math	30	23	0	7	30	2	0	28	23.33	93.33
	Mazda-Math	30	20	0	10	30	0	0	30	33.33	100.00
	Fern-Math	30	28	0	2	30	2	2	26	6.67	93.33
	Ploy-Math	30	18	0	12	30	0	0	30	40.00	100.00
	Deedo	30	19	0	11	30	4	18	8	36.67	86.67
ผลรวม		600	461	3	136	600	24	47	529	23.17	96.00
เปอร์เซ็น			76.83	0.50	22.67		4.00	7.83	88.17		
ปฐมนิเทศ	Jame	30	27	0	3	30	11	2	17	10.00	63.33
	Fang	30	17	1	12	30	0	0	30	43.33	100.00
	Mickey	30	24	0	6	30	0	0	30	20.00	100.00
	May-Math	30	25	0	5	30	0	0	30	16.67	100.00
	Bank	30	23	0	7	30	0	0	30	23.33	100.00
	Khanin	30	22	0	8	30	0	2	28	26.67	100.00
	View	30	25	0	5	30	0	3	27	16.67	100.00
	Nan	30	21	0	9	30	3	0	27	30.00	90.00
	Muay	30	24	0	6	30	0	0	30	20.00	100.00
	Ging-Math	30	25	0	5	30	2	0	28	16.67	93.33
	Prem	30	22	0	8	30	0	0	30	26.67	100.00
	Fair	30	21	0	9	30	0	0	30	30.00	100.00
	Jui	30	23	0	7	30	0	0	30	23.33	100.00
	Ying	30	26	0	4	30	5	1	24	13.33	83.33
	Kratae-Math	30	27	0	3	30	12	1	17	10.00	60.00
	Jib-Math	30	21	0	9	30	2	0	28	30.00	93.33
	Mazda-Math	30	24	0	6	30	1	3	26	20.00	96.67
	Fern-Math	30	27	0	3	30	1	2	27	10.00	96.67
	Ploy-Math	30	26	0	4	30	0	1	29	13.33	100.00
	Deedo	30	23	0	7	30	2	10	18	23.33	93.33
ผลรวม		600	473	1	126	600	39	25	536	21.17	93.50
เปอร์เซ็น			78.83	0.17	21.00		6.50	4.17	89.33		
ผลรวมของระบบ		1800	1422	4	374	1800	106	130	1564	21.00	94.11
เปอร์เซ็นของระบบ			79.00	0.22	20.78		5.89	7.22	86.89		

	Total Real Test	Total Real Pass	Total Real Fail		Total Impostor Test	Total Impostor Pass	Total Impostor Fail	
			Character Fail	Fail			Character Fail	Fail
	1800	1422	4	374	1800	106	130	1564
FRR	21.00							
FAR	5.89							