



การวิเคราะห์ใบหน้าเพื่อยืนยันตัวบุคคลสำหรับระบบบันทึกเวลาการเข้าทำงาน
กรณีศึกษา บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน)

**Facial recognition system for time stamping for Metro System
Corporation Plc.**

นาย ธีรพงศ์ อธิพิทักษ์เมธี

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีไทย-ญี่ปุ่น

พ.ศ. 2561

การวิเคราะห์ใบหน้าเพื่อยืนยันตัวบุคคลสำหรับระบบบันทึกเวลาการทำงาน
กรณีศึกษา บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน)

Facial recognition system for time stamping for Metro System Corporation Plc.

นาย ณริพงษ์ อธิพิทักษ์เมธี

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยี ไทย-ญี่ปุ่น

พ.ศ. 2561

คณะกรรมการสอบ

..... ประธานกรรมการสอบ

(ผศ. ตรีรัตน์ เมตต์การุณจิต)

..... กรรมการสอบ

(ดร. ปราณิสรา อิศรเสนา)

..... อาจารย์ที่ปรึกษา

(อาจารย์ สลิตา ณ หนองคาย)

..... ประธานสหกิจศึกษาสาขาวิชา

(อ.สลิตา ชีวกิตการ)

ลิขสิทธิ์ของสถาบันเทคโนโลยีไทย - ญี่ปุ่น

ชื่อโครงการ	การวิเคราะห์ใบหน้าเพื่อยืนยันตัวตนบุคคลสำหรับระบบบันทึกเวลาการเข้าทำงาน กรณีศึกษา บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) Facial recognition system for time stamping for Metro System Corporation Plc.
ผู้เขียน	นาย ฌธิพงษ์ อธิพิทักษ์เมธี
คณะวิชา	เทคโนโลยีสารสนเทศ สาขาวิชา เทคโนโลยีสารสนเทศ
อาจารย์ที่ปรึกษา	อาจารย์ สลิตา ฌ หนองคาย
พนักงานที่ปรึกษา	นาย สุภกร พึ่งเกียรติขจร
ชื่อบริษัท	บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน)
ประเภทธุรกิจ/สินค้า	Web Application, Software Application

บทสรุป

จากการปฏิบัติงานสหกิจ บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) ได้รับมอบหมายให้ทำโครงการศึกษาตรวจสอบการเข้าทำงานของพนักงานโดยการเข้าทำงานในสมัยก่อน และปัจจุบันต้องใช้การเขียน หรือ ใช้บัตรเพื่อยืนยันตัวตน จึงเกิดความไม่สะดวกสบายในการเข้าทำงานขึ้น ดังนั้น จึงจัดทำระบบวิเคราะห์ใบหน้าและยืนยันตัวตนเพื่อให้เกิดความสะดวกสบาย และยืนยันตัวตนของบุคคลได้ดียิ่งขึ้น โดยใช้ ภาษา Python ในการเขียน และใช้ควบคู่กับ กล้องคอมพิวเตอร์พกพา

ระบบนี้สามารถจับภาพบุคคลเพื่อบันทึกวันที่และเวลา โดยมีการแสดงชื่อและเปอร์เซ็นต์ความคล้ายของบุคคลนั้นๆ และระบบนี้สามารถช่วยลดเวลาและเพิ่มความสะดวกในบันทึกการเข้าทำงาน โดยพนักงานสามารถเข้าทำงานและบันทึกเวลาได้โดยไม่ต้องพกบัตรหรือใช้บัตรในการเข้าทำงาน

กิตติกรรมประกาศ

จากการที่ได้ไปสหกิจศึกษาที่ ณ บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) นับตั้งแต่ วันที่ 04 พฤษภาคม พ.ศ. 2561 จนถึง 28 กันยายน พ.ศ. 2561 ทำให้ได้รับความรู้และประสบการณ์จากการทำงานจริงที่สำคัญ ต่อการพัฒนาตัวเองในอนาคต และ สำหรับรายงานปฏิบัติงานสหกิจนี้ ขอกราบขอบคุณผู้ที่ให้ความรู้และสนับสนุนการฝึกสหกิจจนสำเร็จลุล่วงไปได้ด้วยดี

และสุดท้ายนี้ขอกราบขอบคุณพี่ๆ ทุกคนใน บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) และ อาจารย์ ลลิตา ณ หนองคาย ที่ให้ความช่วยเหลือและ ให้คำปรึกษาต่างๆ ตลอดทั้ง 4 เดือน ในการปฏิบัติสหกิจที่ผ่านมานจนสำเร็จ ลุล่วงไปได้ด้วยดี

นาย ณริพงษ์ อธิพิทักษ์เมธี

ผู้จัดทำ

TNI

สารบัญ

หน้า

บทสรุป.....	ก
กิตติกรรมประกาศ.....	ข
สารบัญ.....	ค
สารบัญภาพ.....	ณ
สารบัญตาราง.....	ช

บทที่

บทที่ 1 บทนำ.....	1
1.1 ชื่อและที่ตั้งของสถานประกอบการ	1
1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร	2
1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร.....	4
1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย	4
1.5 พนักงานที่ปรึกษา และ ตำแหน่งของพนักงานที่ปรึกษา.....	5
1.6 ระยะเวลาที่ปฏิบัติงาน	5
1.7 ที่มาและความสำคัญของปัญหา.....	5
1.8 วัตถุประสงค์หรือจุดมุ่งหมายของโครงการ	5
1.9 ผลที่คาดว่าจะได้รับการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย.....	6
1.10 นิยามศัพท์เฉพาะ.....	6
บทที่ 2 ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน.....	7
2.1 เทคโนโลยีที่ใช้ในการปฏิบัติงาน.....	7
2.1.1 ความหมายของโปรแกรม Oracle VM VirtualBox	7
2.1.2 โปรแกรม PyCharm	9
2.1.3 ภาษา Python	11
2.1.4 SQLite.....	13

สารบัญ(ต่อ)

หน้า

บทที่ 2 ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

2.1.5 Django Framework	16
2.1.6 เทคนิค Haarcascade_frontalface_default	18
2.1.7 เทคนิค Local Binary Pattern Histogram.....	18

บทที่ 3 แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน19

3.1 แผนงานการปฏิบัติงาน	19
3.2 รายละเอียดที่นักศึกษาปฏิบัติในการฝึกงาน	19
3.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติงาน	20
3.3.1 งานประจำที่ได้รับมอบหมาย	20
3.3.2 งานโครงการที่ได้รับมอบหมาย	23
3.3.3 ขั้นตอนการทำงานของระบบ Face Recognition	26
3.3.4 ทดสอบระบบ	28

บทที่ 4 สรุปผลการดำเนินงาน การวิเคราะห์และสรุปผลต่างๆ30

4.1 ขั้นตอนและผลการดำเนินงาน	30
4.1.1 ส่วนจับภาพใบหน้าบุคคลสำหรับบันทึกภาพ	31
4.1.2 ส่วนที่การฝึกให้ระบบจดจำภาพของบุคคล.....	32
4.1.3 ส่วนที่แสดงชื่อในภาพแบบ Real Time และบันทึกเวลา	33
4.2 ผลการวิเคราะห์ข้อมูล	33
4.3 วิเคราะห์และวิจารณ์ข้อมูลโดยเปรียบเทียบผลที่ได้รับกับวัตถุประสงค์และ จุดมุ่งหมายในการปฏิบัติงานหรือการจัดทำโครงการ	33

สารบัญ(ต่อ)

หน้า

บทที่ 5 บทสรุปและข้อเสนอแนะ.....	35
5.1 สรุปผลการดำเนินงาน	35
5.2 แนวทางการแก้ไขปัญหา	35
5.3 ข้อเสนอแนะจากการดำเนินงาน	36
เอกสารอ้างอิง	37
ภาคผนวก.....	38
ก. รายงานประจำสัปดาห์.....	39
ประวัติผู้จัดทำโครงการ.....	56



TNI

TAHT - NICHI INSTITUTE OF TECHNOLOGY

สารบัญภาพ

ภาพที่	หน้า
ภาพที่ 1.1 สถานที่ตั้ง บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน)	1
ภาพที่ 1.2 Metro Proud Awards	3
ภาพที่ 1.3 คณะผู้บริหาร บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน).....	4
ภาพที่ 2.1 ไอคอนของโปรแกรม Oracle VM VirtualBox	7
ภาพที่ 2.2 หน้าตาของโปรแกรม Oracle VM VirtualBox	9
ภาพที่ 2.3 ไอคอนของโปรแกรม PyCharm.....	10
ภาพที่ 2.4 หน้าการใช้งานของ โปรแกรม PyCharm.....	11
ภาพที่ 2.5 โลโก้ของภาษา Python	12
ภาพที่ 2.6 ตัวอย่างของภาษา Python	13
ภาพที่ 2.7 โลโก้ของ SQLite.....	14
ภาพที่ 2.8 ตัวอย่างโค้ดของ SQLite	15
ภาพที่ 2.9 ตัวอย่างหน้าเว็บไซต์ที่สร้างโดย SQLite.....	16
ภาพที่ 2.10 โลโก้ของ Django Framework	16
ภาพที่ 2.11 ตัวอย่างโค้ดของ Django Framework	17
ภาพที่ 2.12 พื้นฐานการทำงานของ Local Binary Pattern	18
ภาพที่ 3.1 ตัวอย่างโค้ด Stored Procedures ใช้ and/or	20
ภาพที่ 3.2 ตัวอย่างโค้ด Stored Procedures การสร้าง Temp Table	21
ภาพที่ 3.3 ตัวอย่างโค้ด Javascript การทำ Web Chat	23
ภาพที่ 3.4 โค้ดส่วนการใส่ไอดีบุคคล	24
ภาพที่ 3.5 โค้ดส่วนชื่อของภาพที่ถ่าย และจำนวนและเวลาของการถ่ายภาพ	24
ภาพที่ 3.6 โค้ดส่วนการใช้เทคนิค Haarcascade และ Local Binary Patterns Histograms	25
ภาพที่ 3.7 โค้ดส่วนการแสดงชื่อและเปอร์เซ็นต์ความคล้าย.....	25
ภาพที่ 3.8 ส่วนการถ่ายภาพใบหน้าบุคคล	26
ภาพที่ 3.9 ส่วนการประมวลผลเพื่อสร้างโมเดล	27
ภาพที่ 3.10 ส่วนการจับภาพใบหน้าบุคคลและบันทึกวันที่และเวลา	28

สารบัญภาพ(ต่อ)

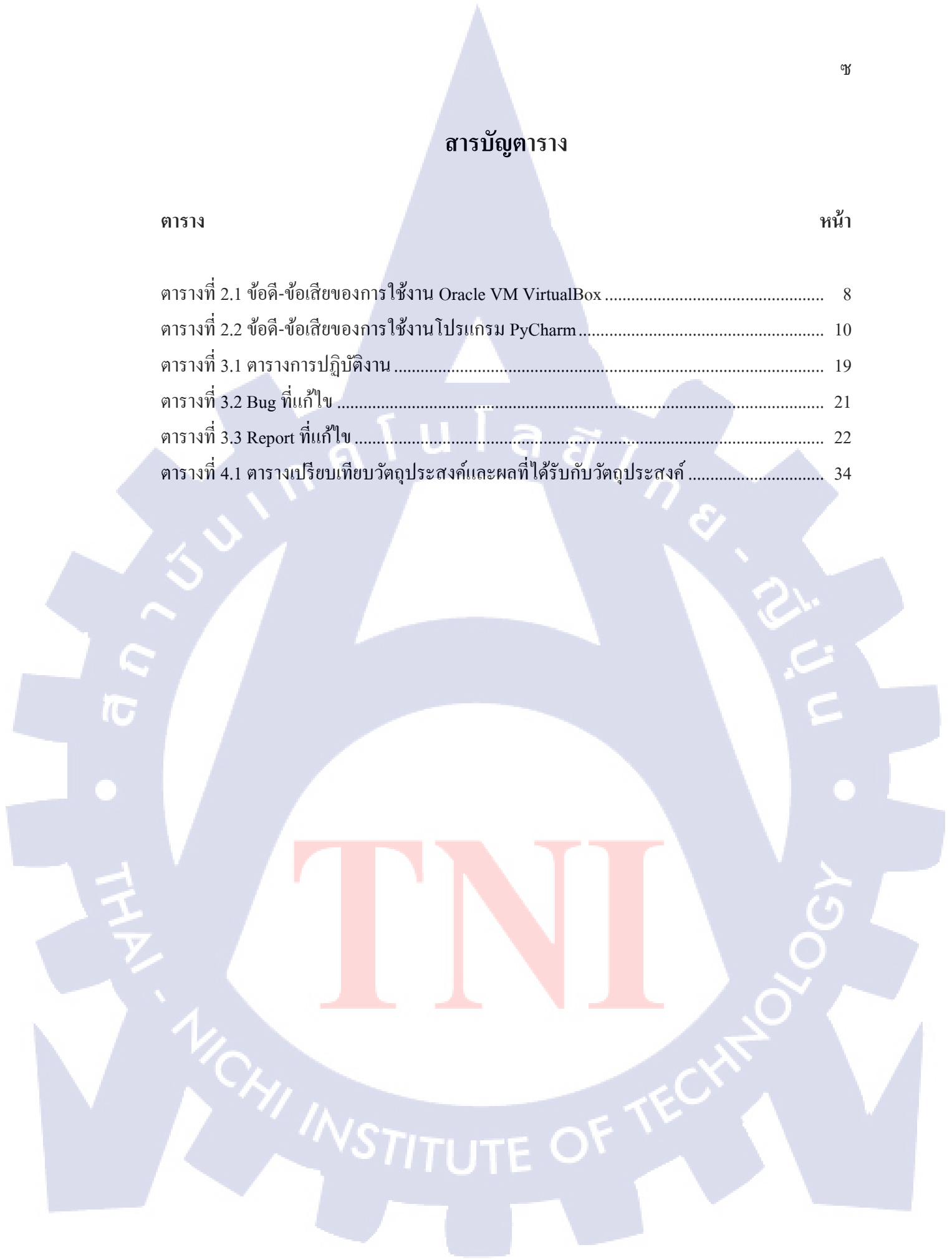
ภาพที่	หน้า
ภาพที่ 3.11 การแสดงชื่อและเปอร์เซ็นต์ความคล้ายของบุคคล	29
ภาพที่ 3.12 การบันทึกวันที่และเวลา	29
ภาพที่ 4.1 การแสดงชื่อและรหัส และข้อความเพื่อให้ใส่เลขไอดีของบุคคล.....	30
ภาพที่ 4.2 การจับภาพของบุคคล	31
ภาพที่ 4.3 การฝึกให้ระบบจดจำภาพของบุคคล	31
ภาพที่ 4.4 การแสดงชื่อในภาพและเปอร์เซ็นต์ความเหมือนแบบ Real Time	32
ภาพที่ 4.5 การบันทึกวันที่และเวลา	32

TNI

THAI - NICHI INSTITUTE OF TECHNOLOGY

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 2.1 ข้อดี-ข้อเสียของการใช้งาน Oracle VM VirtualBox	8
ตารางที่ 2.2 ข้อดี-ข้อเสียของการใช้งานโปรแกรม PyCharm	10
ตารางที่ 3.1 ตารางการปฏิบัติงาน	19
ตารางที่ 3.2 Bug ที่แก้ไข	21
ตารางที่ 3.3 Report ที่แก้ไข	22
ตารางที่ 4.1 ตารางเปรียบเทียบวัตถุประสงค์และผลที่ได้รับกับวัตถุประสงค์	34



TNI

บทที่ 1

บทนำ

1.1 ชื่อและที่ตั้งของสถานประกอบการ

ชื่อหน่วยงาน

บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน)

(Metro Systems Corporation Public Company Limited)

ที่ตั้ง

เลขที่ 400 ถนนเฉลิมพระเกียรติ ร.9 แขวงหนองบอน เขตประเวศ

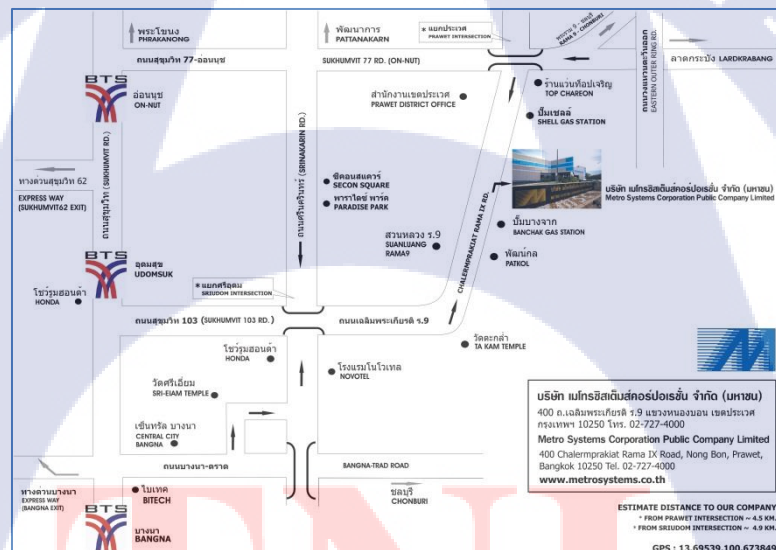
กรุงเทพมหานคร 10250

เว็บไซต์

<http://www.metrosystems.co.th>

อีเมลล์

recruitment@metrosystems.co.th



ภาพที่ 1.1 สถานที่ตั้ง บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน)

1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร

บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) ได้ก่อตั้งขึ้นในปี พ.ศ. 2529 ด้วยทุนจดทะเบียน 4 ล้านบาท เริ่มต้นดำเนินธุรกิจจากการเป็นตัวแทนจำหน่ายผลิตภัณฑ์ไอทีเอ็มเป็นรายแรกในภูมิภาคเอเชียตะวันออกเฉียงใต้ ต่อมาได้มีการขยายการจัดจำหน่ายโซลูชันและบริการจากบริษัทไอทีชั้นนำอื่น ๆ เพื่อการบริการที่ครบวงจรและครอบคลุมความต้องการของลูกค้าได้มากยิ่งขึ้น

ในปี พ.ศ. 2538 บริษัทได้เพิ่มทุนจดทะเบียนเป็น 180 ล้านบาท เพื่อปรับเปลี่ยนรูปแบบธุรกิจโดยจดทะเบียนเป็นบริษัทมหาชนในตลาดหลักทรัพย์แห่งประเทศไทยและทำการซื้อขายหุ้นสามัญครั้งแรกเมื่อวันที่ 9 พฤษภาคม พ.ศ. 2539 ตลอดระยะเวลาการดำเนินงาน บริษัทมีการปรับเปลี่ยนมูลค่าทุนจดทะเบียนให้ตามความเหมาะสมกับขนาดของธุรกิจ โดย ณ ปัจจุบันบริษัทมีทุนจดทะเบียน 360 ล้านบาท

ภายหลังการได้รับรองระบบคุณภาพ ISO 9001: 2008 ในทุกขั้นตอนการทำงาน และระบบ ISO/IEC 20000:2005 และ ISO/IEC 27001:2005 ซึ่งเกี่ยวข้องกับการป้องกันสารสนเทศในองค์กรที่อาจส่งผลกระทบต่อการทำงาน ทำให้ในปี 2554 ทางบริษัทได้เดินหน้าจัดทำระบบบริหารความต่อเนื่องทางธุรกิจ ((Business Continuity Management System : BCMS) และกำหนดเพิ่มเติมไว้ในพันธกิจหลักขององค์กร กล่าวคือ “...การมุ่งมั่นสู่ความเป็นเลิศและบริหารความต่อเนื่องในธุรกิจ เทคโนโลยีสารสนเทศครบวงจร โดยบุคลากรระดับมืออาชีพ เพื่อความสำเร็จของลูกค้า ตลอดจนมีส่วนร่วมในการพัฒนาสังคมแห่งความรู้ ทั้งนี้ เป็นการป้องกันบรรเทาอุบัติเหตุการณ์ที่อาจก่อให้เกิดความเสียหายต่อธุรกิจ และทำให้บริษัทสามารถดำเนินธุรกิจได้อย่างต่อเนื่อง ซึ่งมุ่งเน้นการดูแลความปลอดภัย การปฏิบัติตามกฎหมาย กฎระเบียบ ข้อตกลงที่เกี่ยวข้อง ตลอดจนดูแลตรวจสอบความถูกต้องของข้อมูล ภาพลักษณ์และความน่าเชื่อถือของบริษัท

ด้วยแนวคิดที่ยึดเอาลูกค้าเป็นศูนย์กลาง บริษัทฯ จึงมุ่งแสวงหาแนวทางใหม่ๆ เพื่อตอบสนองความต้องการของลูกค้า และยังคงเดินหน้าแสวงหาผลิตภัณฑ์และบริการต่างๆ เพื่อให้ทันกับการเปลี่ยนแปลงของอุตสาหกรรมไอที การมุ่งแสวงหาบริการที่จะทำให้ลูกค้าดำเนินธุรกิจได้อย่างมีประสิทธิภาพและต่อเนื่องทั้งในสภาวะปกติ หรือในยามเผชิญวิกฤต และยังให้ความสำคัญในการการร่วมมือกับพันธมิตรธุรกิจเพื่อนำเสนอโซลูชันให้ครอบคลุมความต้องการของกลุ่มลูกค้า บริษัทฯ จะยังคงเดินหน้าพัฒนาบุคลากร และผสมผสานความสามารถของซัพพลายเออร์ คู่ค้า ตลอดจนนวัตกรรมใหม่ๆ ที่เกี่ยวข้อง เพื่อตอบสนองความต้องการ สร้างความสำเร็จ และสร้างคุณค่าเพิ่มให้แก่ธุรกิจของลูกค้า

จากการเติบโตทางธุรกิจอย่างรวดเร็ว ทำให้ บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) ได้รับมอบรางวัล “IBM Business Partner of the Year” ซึ่งเป็นรางวัลที่ได้รับติดต่อกันเป็นเวลาถึง 25 ปี พร้อมทั้งได้รับรางวัล IBM The Longest Relation Business Partner, Citrix - Platinum Partner of the Year และรางวัลอื่นๆจากลูกค้าอีกมากมาย ดังภาพที่ 1.2



ภาพที่ 1.2 Metro Proud Awards

ปัจจุบันบริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) มีประเภทของผลิตภัณฑ์และการบริการ ซึ่งสามารถแบ่งออกเป็น 3 กลุ่มใหญ่ๆ ได้แก่

- กลุ่มธุรกิจดิจิทัลโซลูชัน (Digital Solution Group – DSG) จำหน่ายผลิตภัณฑ์คอมพิวเตอร์ฮาร์ดแวร์ ประกอบด้วย คอมพิวเตอร์ขนาดกลาง พีซี อุปกรณ์จัดเก็บข้อมูล ระบบคำปพลิเคชัน เครื่องพิมพ์สำหรับระบบคอมพิวเตอร์ขนาดใหญ่
- กลุ่มธุรกิจซอฟต์แวร์โซลูชัน (Solutions Integration Group – SIG) จำหน่ายซอฟต์แวร์โซลูชันและบริการด้านไอทีเพื่อการบริหารงานที่มีประสิทธิภาพ
- กลุ่มธุรกิจดิจิทัลพริ้นติ้ง (Digital Printing Group – DPG) จำหน่ายวัสดุสิ้นเปลืองที่เกี่ยวข้องกับคอมพิวเตอร์ เครื่องพิมพ์ขนาดเล็ก กล้องถ่ายรูป และเครื่องโปรเจกเตอร์ เป็นต้น

1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร



ภาพที่ 1.3 คณะผู้บริหารบริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน)

1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย

ตำแหน่ง	โปรแกรมเมอร์ (Programmer)
หน้าที่งานที่ได้รับมอบหมาย	เขียนโค้ด C#, เขียนโค้ด Stored Procedures, แก้ไขบั๊ก, จัดทำรายงาน, จัดทำเว็บไซต์

1.5 พนักงานที่ปรึกษา และ ตำแหน่งของพนักงานที่ปรึกษา

ชื่อ – นามสกุล

นาย ศุภกร พึ่งเกียรติจิกร

ตำแหน่ง

Program Developer

1.6 ระยะเวลาที่ปฏิบัติงาน

ปฏิบัติงานสหกิจศึกษาเป็นระยะเวลา 4 เดือน

นับตั้งแต่วันที่ 04 มิถุนายน 2561 – 28 กันยายน 2561

1.7 ที่มาและความสำคัญของปัญหา

เนื่องจากในอดีตและปัจจุบันการบันทึกการเข้าทำงานนั้นทำด้วยวิธี เขียน หรือ ใช้บัตรเพื่อเข้าและยืนยันตัวตนในการเข้าทำงาน จึงทำให้การเข้าและยืนยันตัวตนเข้าทำงานนั้นไม่มีความสะดวกสบายในการใช้งาน และในบางครั้งทำให้สูญเสียเวลามากขึ้น

ระบบวิเคราะห์ใบหน้าและยืนยันตัวตนบุคคลเพื่อการบันทึกเวลาเข้าทำงาน สำหรับ บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) จึงได้ถูกพัฒนาขึ้นมาเพื่อทำให้การเช็คชื่อเข้าและยืนยันตัวตนในการเข้าทำงานมีประสิทธิภาพยิ่งขึ้น โดยนำมาใช้ร่วมกับโปรแกรม Oracle VM VirtualBox, PyCharm, Python, Linux Ubuntu และ Django โดยมีการพัฒนาระบบเป็นสองส่วน ส่วนที่หนึ่งคือ ระบบวิเคราะห์ใบหน้าและยืนยันตัวตนบุคคลเพื่อการเช็คชื่อเข้าทำงาน และ ส่วนที่สองคือ Django ที่จัดเก็บข้อมูลลง SQLite ซึ่งในโครงงานนี้ทำในส่วนของ ระบบวิเคราะห์ใบหน้าและยืนยันตัวตนบุคคลเพื่อการเช็คชื่อเข้าทำงาน

1.8 วัตถุประสงค์หรือจุดมุ่งหมายของโครงการ

1. เพื่อพัฒนาต้นแบบของระบบวิเคราะห์ใบหน้าและยืนยันตัวตนบุคคลเพื่อการบันทึกเวลาเข้าทำงานสำหรับพนักงาน
2. เพื่อบริษัทสามารถนำต้นแบบของระบบวิเคราะห์ใบหน้าและยืนยันตัวตนบุคคลเพื่อการบันทึกเวลาเข้าทำงานไปใช้งานจริงในอนาคต
3. เพื่อพัฒนาทักษะในการทำงานจากประสบการณ์จริง

1.9 ผลที่คาดว่าจะได้รับการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย

1. สามารถวิเคราะห์ใบหน้า, ยืนยันตัวบุคคลและบันทึกเวลาเข้าทำงานได้
2. สามารถนำต้นแบบของระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการบันทึกเวลาเข้าทำงานไปใช้ได้จริงในอนาคต
3. สามารถนำความรู้และทักษะในการทำงานมาพัฒนาตนเอง

1.10 นิยามศัพท์เฉพาะ

1. โปรแกรม Oracle VM VirtualBox คือ โปรแกรมที่ใช้จำลองการทำงานของระบบปฏิบัติการ เช่น Window และ Linux เป็นต้น
2. โปรแกรม PyCharm คือ โปรแกรมที่ใช้สำหรับเขียนคำสั่ง Python มีตัวช่วยสำหรับผู้เริ่มต้นเขียนภาษา Python
3. Face Recognition คือ ระบบการตรวจหาใบหน้าของมนุษย์และปรับภาพใบหน้าโดยอัตโนมัติ กรอบจะปรากฏขึ้นบนใบหน้าที่ถูกตรวจจับ
4. Django คือ ชุดเครื่องมือ Framework สำหรับ การพัฒนาเว็บไซต์ด้วยภาษา Python

บทที่ 2

ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

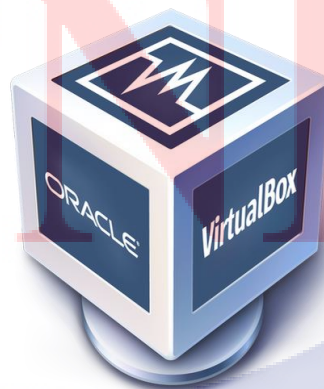
ระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการเช็คชื่อเข้าทำงาน สำหรับ บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) นี้จัดทำขึ้นมาโดยมีเครื่องมือเป็นโปรแกรมในการ เขียนโค้ด และโปรแกรมที่จำลองระบบปฏิบัติการโดยเฉพาะ ซึ่งโปรแกรมที่ใช้สำหรับทำระบบมี ดังนี้

2.1 เทคโนโลยีที่ใช้ในการปฏิบัติงาน

2.1.1 ความหมายของโปรแกรม Oracle VM VirtualBox

โปรแกรม Oracle VM VirtualBox เป็นโปรแกรมเครื่องมือที่ช่วยให้สามารถ จำลองระบบปฏิบัติการที่ใช้ในการทำระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการเช็คชื่อเข้า ทำงาน ซึ่งสามารถลงระบบปฏิบัติการ Linux ได้โดยไม่ยุ่งเกี่ยวกับระบบปฏิบัติการ Window ที่ใช้ อยู่บนเครื่องคอมพิวเตอร์พกพา

โปรแกรม Oracle VM VirtualBox สามารถแบ่งความจำเครื่องบางส่วนไปใช้ สำหรับเก็บข้อมูลของระบบปฏิบัติการที่ติดตั้งลงไปและสามารถนำโปรแกรมต่างๆไปติดตั้งภายใน ระบบปฏิบัติการได้ทำให้สามารถทำงานได้โดยที่ข้อมูลทั้งหมดไม่มีผลกระทบต่อ ระบบปฏิบัติการ Window ที่ใช้ และโปรแกรมสามารถโคลน นำเข้า ลบ และ ส่งออก ระบบปฏิบัติการที่สร้างขึ้นได้



ภาพที่ 2.1 ไอคอนของโปรแกรม Oracle VM VirtualBox

ตารางที่ 2.1 ข้อดี-ข้อเสียของการใช้งาน Oracle VM VirtualBox

ข้อดี	ข้อเสีย
สะดวกต่อการติดตั้งและใช้งาน	มีการทำงานค่อนข้างช้า
ไม่ส่งผลกระทบต่อระบบปฏิบัติการ Window ที่ใช้	มีปัญหา Error ที่ไม่ทราบสาเหตุ
สามารถโคลน ลบ นำเข้า และ ส่งออก ข้อมูลระบบปฏิบัติการที่ทำได้	

2.1.1.1 ประวัติของโปรแกรม Oracle VM VirtualBox

VirtualBox ถูกเสนอครั้งแรกโดย Innotek GmbH จาก Weinstadt , เยอรมนี ซอฟต์แวร์นี้เป็นกรรมสิทธิ์ภายใต้ใบอนุญาตเวอร์ชันหนึ่งของผลิตภัณฑ์ที่ไม่มีค่าใช้จ่ายสำหรับการใช้งานส่วนบุคคลหรือการประเมินผล ขึ้นอยู่กับการใช้งาน VirtualBox ส่วนบุคคลและใบอนุญาตการประเมินผล (Puel) ใน และในเดือนมกราคม ปี ค.ศ. 2007 บนพื้นฐานของคำแนะนำโดย LiSoG , Innotek GmbH เปิดตัว VirtualBox Open Source Edition (OSE) เป็นซอฟต์แวร์เสรีซึ่งเป็น Open Source ภายใต้ข้อกำหนดของ GNU General Public License (GPL) ในเวอร์ชันที่ 2

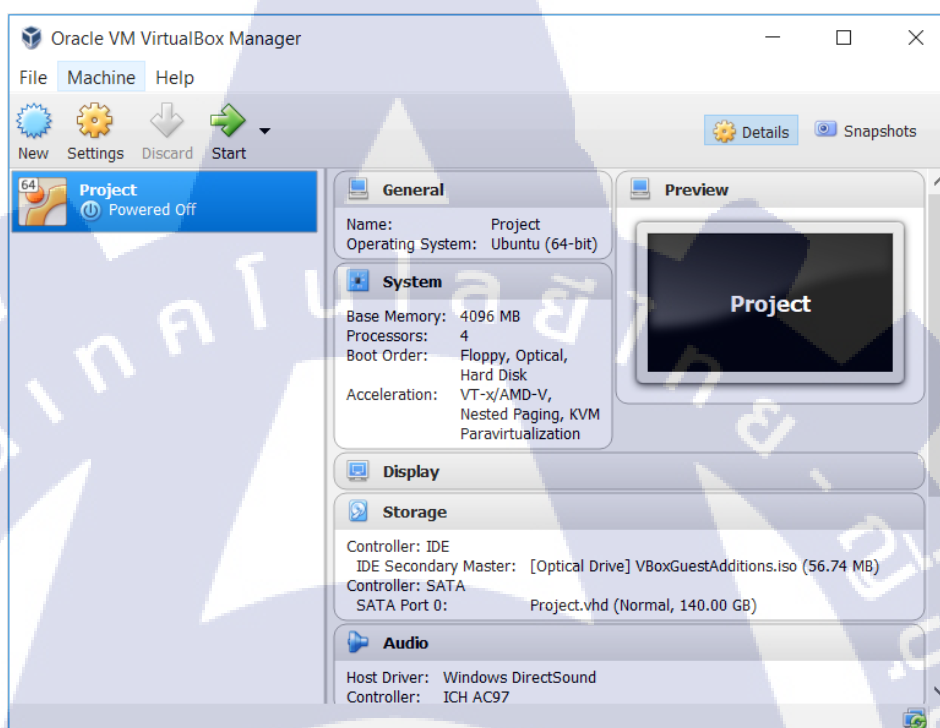
Innotek GmbH ยังสนับสนุนการพัฒนาของ OS / 2 และลินุกซ์สนับสนุนในการทำงานแบบเสมือนและ OS / 2 พอร์ตของผลิตภัณฑ์จาก Connectix ซึ่งได้รับมาในภายหลังโดยไมโครซอฟท์ โดยเฉพาะอย่างยิ่ง Innotek ได้พัฒนาโค้ด "additions" ในทั้ง Microsoft Virtual PC และ Microsoft Virtual Server ซึ่งช่วยให้สามารถโต้ตอบกับโฮสต์ผู้อื่นได้ เช่น การแชร์คลิปบอร์ดที่ใช้ร่วมกันหรือการปรับขนาดมุมมองแบบไดนามิก และบริษัท Sun Microsystems ชื่อ Innotek ในเดือนกุมภาพันธ์ ปี ค.ศ. 2008

ในเดือน มกราคม ค.ศ.2010 บริษัท ออราเคิลคอร์ปอเรชั่น ได้ซื้อกิจการ Sun Microsystems และเปลี่ยนชื่อผลิตภัณฑ์เป็น "Oracle VM VirtualBox"

2.1.1.2 การจำลองสภาพแวดล้อม

ผู้ใช้ VirtualBox สามารถดาวน์โหลดระบบปฏิบัติการต่างๆ จำนวนมากภายใต้ระบบปฏิบัติการที่เป็น Single Host operating-system (host OS) แต่ละโฮสต์สามารถเริ่มต้น, หยุดชั่วคราวและหยุดทำงานได้โดยอิสระภายในเครื่องเสมือน (VM) ของตนเอง ผู้ใช้สามารถกำหนดค่าแต่ละ VM ได้โดยอิสระและเรียกใช้งานภายใต้การเลือกใช้การจำลองเสมือนของซอฟต์แวร์หรือฮาร์ดแวร์ที่ได้รับการสนับสนุน

ระบบปฏิบัติการหลัก และ OS ของ Guest และแอปพลิเคชันสามารถติดต่อสื่อสารกันผ่านทางกลไกต่างๆ รวมถึงคลิปบอร์ดทั่วไป และระบบเครือข่ายเสมือนจริง และ Guest ของเครื่องเสมือน (VM) สามารถสื่อสารกันโดยตรงหากกำหนดค่าให้ทำเช่นนั้น



ภาพที่ 2.2 หน้าตาของโปรแกรม Oracle VM VirtualBox

2.1.2 โปรแกรม PyCharm

โปรแกรม PyCharm เป็นโปรแกรมสำหรับเขียนภาษา Python ที่ช่วยให้สามารถเขียนภาษา Python ได้สะดวกมากขึ้น โปรแกรม PyCharm นั้นสามารถใช้ได้ทั้งบนระบบปฏิบัติการ Window และ ระบบปฏิบัติการ Linux

โปรแกรม PyCharm มีการแจ้ง Error และสามารถแสดงตัวช่วยคำสั่งต่างๆ โดยอัตโนมัติ จึงมีความสะดวกต่อการใช้เขียนภาษา Python



ภาพที่ 2.3 ไอคอนของโปรแกรม PyCharm

ตารางที่ 2.2 ข้อดี-ข้อเสียของการใช้งานโปรแกรม PyCharm

ข้อดี	ข้อเสีย
ใช้งานง่ายสำหรับมือใหม่ในการเขียนภาษา Python	ติดตั้งยากบนระบบปฏิบัติการ Linux
มีการแจ้งเตือนต่างๆให้ทราบ	

2.1.2.1 ประวัติของโปรแกรม PyCharm

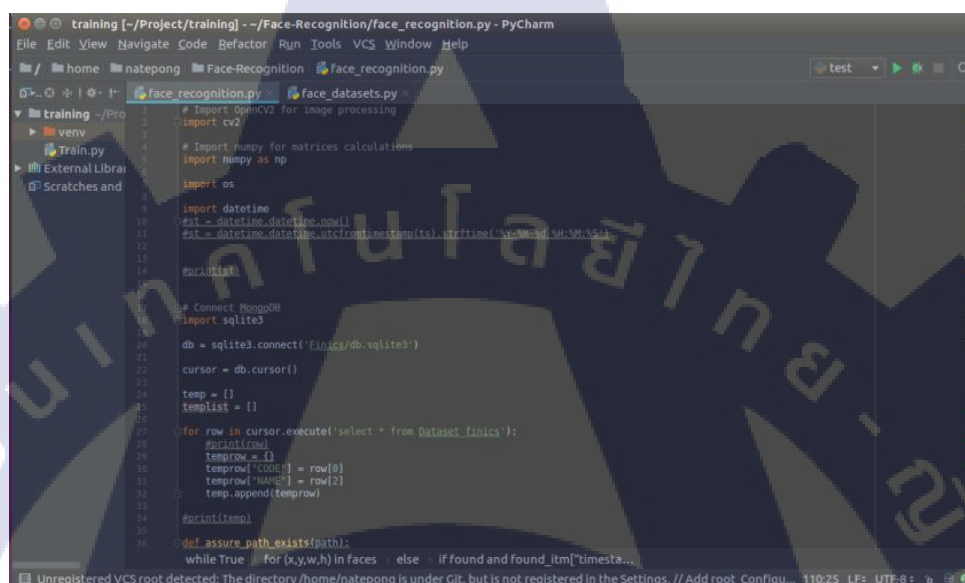
PyCharm รุ่นเบต้าได้รับการเผยแพร่ออกมาในเดือนกรกฎาคม ปี ค.ศ. 2010 โดยมี เวอร์ชัน 1.0 จนมาถึง 3 เดือนหลังจากนั้น เวอร์ชัน 2.0 ได้รับการเผยแพร่ออกมาเมื่อวันที่ 13 ธันวาคม ค.ศ. 2013 และ เวอร์ชัน 3.0 ได้รับการเผยแพร่ออกมาเมื่อวันที่ 24 กันยายน ค.ศ. 2013 และ เวอร์ชัน 4.0 ได้รับการเผยแพร่ออกมาเมื่อวันที่ 19 พฤศจิกายน ค.ศ. 2014

PyCharm Community Edition ซึ่งเป็น PyCharm เวอร์ชัน Open Source สามารถเริ่มใช้งานได้เมื่อวันที่ 22 ตุลาคม ค.ศ. 2013

2.1.2.2 คุณสมบัติของโปรแกรม PyCharm

- สามารถช่วยเหลือและการวิเคราะห์โค้ด และช่วยเกี่ยวกับไวยากรณ์และการเน้นข้อผิดพลาดและการ linter integration และการแก้ไขปัญหาอย่างรวดเร็ว

- สามารถนำทางโปรเจกต์ และโค้ดได้ โดยสามารถเลือกดูเฉพาะโปรเจกต์ และสามารถดูโครงสร้างไฟล์และสามารถเดินทางข้ามระหว่างไฟล์, คลาส และ Method ได้อย่างรวดเร็ว
- ซัพพอร์ต Web Framework เช่น Django, Web2py และ Flask เป็นต้น
- สามารถ Debug โค้ด Python ได้



ภาพที่ 2.4 หน้าการใช้งานของโปรแกรม PyCharm

2.1.3 ภาษา Python

Python เป็นภาษาเขียนโปรแกรมระดับสูงที่ใช้กันอย่างกว้างขวางในการเขียนโปรแกรมสำหรับวัตถุประสงค์ทั่วไป ภาษา Python นั้นสร้างโดย Guido van Rossum และถูกเผยแพร่ครั้งแรกในปี ค.ศ. 1991 Python นั้นเป็นภาษาแบบ Interpret ที่ถูกออกแบบโดยมีปรัชญาที่จะทำให้โค้ดอ่านได้ง่ายขึ้น และโครงสร้างของภาษานั้นจะทำให้โปรแกรมเมอร์สามารถเข้าใจแนวคิดการเขียนโค้ดโดยใช้บรรทัดที่น้อยลงกว่าภาษาอย่าง C++ และ Java ซึ่งภาษานั้นถูกกำหนดให้มีโครงสร้างที่ดัดใจให้การเขียนโค้ดเข้าใจง่ายทั้งในโปรแกรมเล็กไปจนถึงโปรแกรมขนาดใหญ่ ทั้งยังมีคุณสมบัติเป็นภาษาเขียนโปรแกรมแบบไดนามิกส์และมีระบบการจัดการหน่วยความจำอัตโนมัติและสนับสนุนการเขียนโปรแกรมหลายรูปแบบ ที่ประกอบไปด้วย การเขียนโปรแกรมเชิงวัตถุ Imperative การเขียนโปรแกรมแบบฟังก์ชัน และการเขียนโปรแกรมแบบขั้นตอน มันมีไลบรารีที่ครอบคลุมการทำงานอย่างหลากหลาย ตัวแปรในภาษา Python นั้นมีให้ใช้ในหลายระบบปฏิบัติการ ทำให้โค้ดของภาษา Python สามารถรันในระบบต่างๆ ได้อย่างแพร่หลาย



ภาพที่ 2.5 โลโก้ของภาษา Python

2.1.3.1 ประวัติของภาษา Python

Python เกิดขึ้นในช่วงปลายปี ค.ศ. 1980 และ Python ได้เริ่มต้นดำเนินการขึ้นในเดือนธันวาคม ปี ค.ศ. 1989 โดย Guido van Rossum ที่ Centrum Wiskunde & Informatica (CWI) ใน ประเทศ เนเธอร์แลนด์ ซึ่งภาษา Python เป็นภาษาที่ต่อมาจากภาษา ABC (ได้รับแรงบันดาลใจจาก SETL) ภาษา Python สามารถจัดการข้อบกพร่องต่างๆและเชื่อมต่อกับระบบปฏิบัติการ Amoeba ได้ และ Van Rossum remains เป็นผู้เขียนหลักของภาษา Python และ Van Rossum remains ได้หยุดการเขียนภาษา Python ในวันที่ 12 กรกฎาคม ค.ศ. 2018

Python 2.0 เผยแพร่ออกมาในวันที่ 16 ตุลาคม ค.ศ. 2000 และมีคุณสมบัติใหม่ๆ มากมายรวมถึงตัวเก็บรวบรวมข้อมูลที่เป็น cycle-detecting garbage collector และการสนับสนุน Unicode และด้วยการเผยแพร่ครั้งนี้ทำให้กระบวนการพัฒนามีความโปร่งใสและได้รับการสนับสนุนจากชุมชนมากขึ้น

Python 3.0 (เรียกอีกอย่างว่า Python 3000 หรือ py3k) เผยแพร่ในวันที่ 03 ธันวาคม ค.ศ. 2008 หลังจากผ่านการทดสอบเป็นระยะเวลานานเป็นการปรับปรุงที่สำคัญของภาษา Python ที่ไม่สามารถใช้งานร่วมกับรุ่นก่อนหน้านี้ได้อย่างสมบูรณ์ อย่างไรก็ตามคุณสมบัติที่สำคัญหลายอย่างได้รับการย้อนกลับสู่ Python เวอร์ชัน 2.6.x และ 2.7.x และการเผยแพร่ของ Python 3 ประกอบด้วยประโยชน์ที่เรียกว่า 2to3 ซึ่งคือการแปลโค้ด Python 2 ให้ Python 3 โดยอัตโนมัติ

Python 2.7 ได้กำหนดวันสิ้นสุดตั้งแต่ปี ค.ศ. 2015 จากนั้นเลื่อนออกไปจนถึงปี ค.ศ. 2020 จากความกังวลว่าโค้ดขนาดใหญ่ที่มีอยู่ไม่สามารถส่งต่อไปยัง Python 3 ได้โดยง่าย และในเดือน มกราคม ปี ค.ศ. 2017 Google ได้ประกาศงาน Python 2.7 ไปยัง Go transcompiler เพื่อปรับปรุงประสิทธิภาพภายใต้งานที่เกิดขึ้นพร้อมกัน

```

# Import numpy for matrices calculations
import numpy as np

import os

import datetime
#st = datetime.datetime.now()
#st = datetime.datetime.utcfromtimestamp(ts).strftime('%Y-%m-%d %H:%M:%S')

#print(st)

# Connect MongoDB
import sqlite3

db = sqlite3.connect('Finics/db.sqlite3')
cursor = db.cursor()

temp = []
templist = []

for row in cursor.execute('select * from Dataset_finics'):
    #print(row)
    temprow = {}
    temprow["CODE"] = row[0]
    temprow["NAME"] = row[2]
    temp.append(temprow)

```

ภาพที่ 2.6 ตัวอย่างของภาษา Python

2.1.4 SQLite

SQLite เป็นระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ ที่มีอยู่ใน C programming library ในทางตรงกันข้ามกับระบบการจัดการฐานข้อมูลอื่น ๆ SQLite ไม่ใช่เครื่องมือฐานข้อมูลฝั่ง Client

SQLite เป็นตัวเลือกที่ได้รับความนิยมเป็นซอฟต์แวร์ฐานข้อมูลสำหรับการจัดเก็บข้อมูลภายใน / Client ซอฟต์แวร์แอปพลิเคชัน เช่น เว็บเบราว์เซอร์ เป็นต้น SQLite เป็นเครื่องมือฐานข้อมูลที่มีการใช้กันอย่างแพร่หลายมากที่สุด เนื่องจากมีการใช้งานกันอย่างแพร่หลายในปัจจุบัน ทั้งบน เว็บเบราว์เซอร์, ระบบปฏิบัติการ, ระบบแบบฝังตัว (เช่น โทรศัพท์มือถือ) และอื่น ๆ และ SQLite มีการเชื่อมกับภาษาโปรแกรมมิ่งหลายภาษา



ภาพที่ 2.7 โลโก้ของ SQLite

2.1.4.1 ประวัติของ SQLite

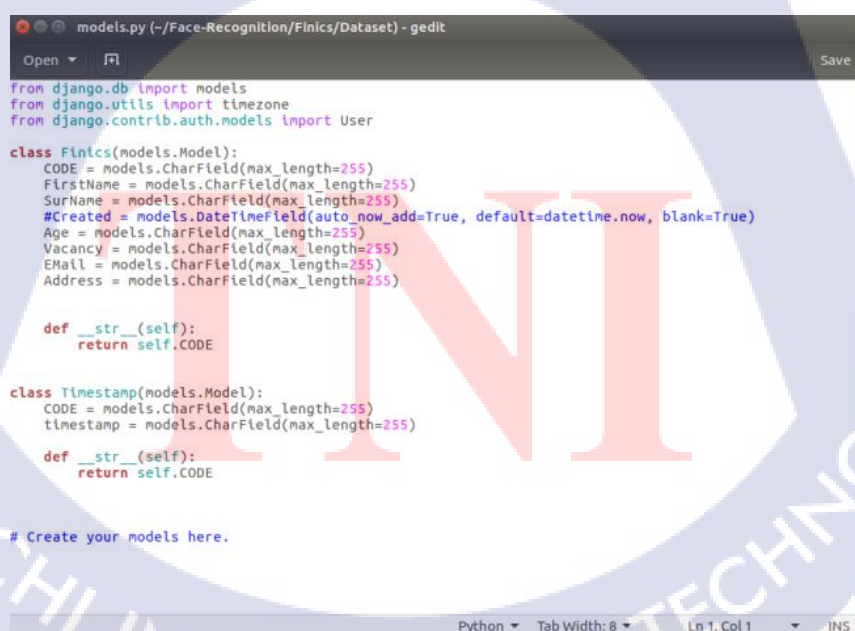
D. Richard Hipp ออกแบบ SQLite ในฤดูใบไม้ผลิ ปี ค.ศ. 2000 ขณะที่ทำงานอยู่ที่ General Dynamics ในช่วงการทำสัญญากับกองทัพเรือสหรัฐฯ Hipp กำลังออกแบบซอฟต์แวร์ที่ใช้บนเรือพิฆาตติดขีปนาวุธนำวิถีซึ่งเดิมใช้ HP-UX กับฐานข้อมูล IBM Informix back-end SQLite เริ่มจากส่วนขยาย Tcl

เป้าหมายการออกแบบของ SQLite คือการอนุญาตให้ใช้งานโปรแกรมได้โดยไม่ต้องติดตั้งระบบการจัดการฐานข้อมูลหรือต้องมีผู้ดูแลระบบฐานข้อมูล ซึ่ง Hipp ได้ใช้หลักไวยากรณ์ และความหมายบนพื้นฐานของ PostgreSQL 6.5 และในเดือนสิงหาคม ปี ค.ศ. 2000 ได้มีการเผยแพร่ SQLite เวอร์ชัน 1.0 โดยมีที่จัดเก็บข้อมูลตามพื้นฐานของ gdbm (GNU Database Manager) และ SQLite 2.0 ถูกนำมาใช้แทนที่ gdbm โดยการใช้ B-tree ที่กำหนดเองและเพิ่มความสามารถในการติดต่อกัน และ SQLite 3.0 ได้รับการสนับสนุนบางส่วนจาก America Online และเพิ่มความเป็นนานาชาติ, การพิมพ์ข้อความ และการปรับปรุงที่สำคัญอื่น ๆ

ในปี ค.ศ. 2011 Hipp ประกาศแผนการที่จะเพิ่ม NoSQL Interface (เอกสารการจัดการที่แสดงใน JSON) ไปยังฐานข้อมูล SQLite และเพื่อพัฒนา UnQLite ซึ่งเป็นฐานข้อมูลแบบฝังตัว และ UnQLite ได้เผยแพร่เป็นฐานข้อมูลอิสระ

2.1.4.2 คุณสมบัติของ SQLite

- SQLite ไม่จำเป็นต้องใช้กระบวนการแยกเซิร์ฟเวอร์หรือระบบปฏิบัติการ (ไม่มีเซิร์ฟเวอร์)
- SQLite ไม่จำเป็นต้องมีการกำหนดค่าซึ่งหมายความว่าไม่มีการติดตั้งหรือการจัดการ
- SQLite มีขนาดเล็กมาก, น้ำหนักเบา และเมื่อมีขนาดน้อยกว่าค่าที่ปรับแต่งแบบเต็มรูปแบบที่ 400KiB และจะทำการละเว้นการตั้งค่าฟังก์ชันตัวเลือกเมื่อมีขนาดน้อยกว่า 250KiB
- SQLite ไม่มีการอ้างอิงจากภายนอก
- การติดต่อกันของ SQLite อนุญาตให้เข้าถึงได้จากหลายกระบวนการหรือตามหัวข้อความปลอดภัย
- SQLite query language สนับสนุน SQL92 (SQL2) มาตรฐาน มากที่สุด
- SQLite เขียนโดยใช้มาตรฐาน ANSI-C, มีความเรียบง่ายและใช้งานง่ายสำหรับ API
- SQLite สามารถเรียกใช้งานบนระบบปฏิบัติการยูนิกซ์ (Linux, Mac OS-X, Android, ios) และ Windows (Win32, WinCE, WinRT)



```
models.py (~/.Face-Recognition/Finics/Dataset) - gedit
Open Save

from django.db import models
from django.utils import timezone
from django.contrib.auth.models import User

class Finics(models.Model):
    CODE = models.CharField(max_length=255)
    FirstName = models.CharField(max_length=255)
    SurName = models.CharField(max_length=255)
    #Created = models.DateTimeField(auto_now_add=True, default=datetime.now, blank=True)
    Age = models.CharField(max_length=255)
    Vacancy = models.CharField(max_length=255)
    Email = models.CharField(max_length=255)
    Address = models.CharField(max_length=255)

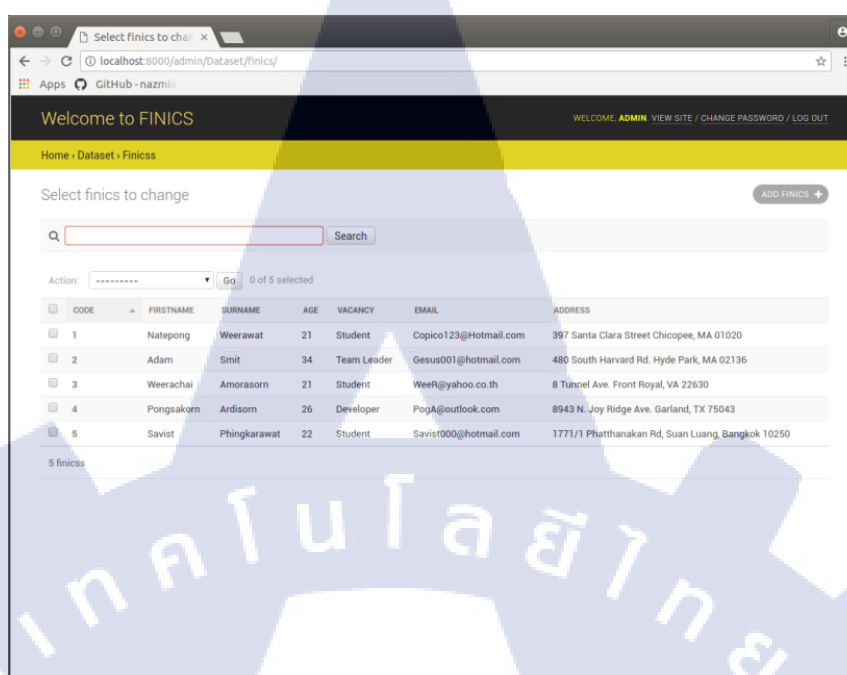
    def __str__(self):
        return self.CODE

class Timestamp(models.Model):
    CODE = models.CharField(max_length=255)
    timestamp = models.CharField(max_length=255)

    def __str__(self):
        return self.CODE

# Create your models here.
```

ภาพที่ 2.8 ตัวอย่างโค้ดของ SQLite



ภาพที่ 2.9 ตัวอย่างหน้าเว็บไซต์ที่สร้างโดย SQLite

2.1.5 Django Framework

Django Framework เป็นชุดเครื่องมือ Framework สำหรับ การพัฒนาเว็บไซต์ด้วยภาษา Python ซึ่งความเป็นจริงแล้วทุกวันนี้มี Framework สำหรับการเขียนเว็บไซต์ด้วยภาษา Python ก่อนข้างเยอะ ซึ่ง Django Framework ก็เป็นหนึ่งใน Framework สำหรับการพัฒนาเว็บไซต์ และทำเว็บไซต์ด้วยภาษา Python ด้วยเช่นกัน โดยปัจจุบันภาษา Python นั้นค่อนข้างได้รับความนิยมเพิ่มมากขึ้นเรื่อย ๆ ทั้งในต่างประเทศ และในประเทศไทย



ภาพที่ 2.10 โลโก้ของ Django Framework

2.1.5.1 ประวัติของ Django Framework

Django ถูกสร้างขึ้นในช่วงฤดูใบไม้ร่วงปี ค.ศ. 2003 เมื่อนักเขียนโปรแกรมบนเว็บที่ทำงานกันอยู่ที่สำนักพิมพ์ Lawrence Journal-World ชื่อ Adrian Holovaty และ Simon Willison เริ่มใช้ Python เพื่อสร้างแอปพลิเคชัน Django ถูกเผยแพร่ขออกสู่สาธารณะภายใต้ใบอนุญาต BSD ในเดือนกรกฎาคมปี ค.ศ. 2005 Framework นี้ถูกตั้งชื่อตามนักกีตาร์ Django Reinhardt

ในเดือนมิถุนายน ปี ค.ศ. 2008 มีการประกาศว่า Django Software Foundation (DSF) ซึ่งเป็นรูปแบบใหม่จะสามารถรักษา Django ไว้ในอนาคตได้

2.1.5.2 คุณสมบัติของ Django Framework

- Object-relational mapper คือ การกำหนด Data Model ในภาษา Python เพื่อการทำงานด้านข้อมูล และสนับสนุน dynamic database-access API
- Automatic admin interface คือ ส่วนของการสร้าง Interface อัตโนมัติสำหรับการเพิ่ม, แก้ไข , ลบ และ ค้นหา ด้วย Django Framework
- Elegant URL design คือ การทำให้ URL มีความสวยงาม ตื่น กระชับ และสื่อความหมายของหน้านั้น ๆ ได้อย่างชัดเจน เหมาะสมกับการทำ SEO ในปัจจุบัน
- Template system คือ Django นั้นมีการออกแบบ Template Language เพื่อการเขียนแยกส่วนระหว่าง Design และ Business Logic
- Cache system คือ ส่วนของการบันทึก หรือจัดการข้อมูลที่มีการดาวน์โหลดไปแล้ว เพื่อเพิ่มประสิทธิภาพการทำงานของเว็บไซต์ด้านความเร็ว และด้านอื่น ๆ
- Internationalization คือ Django สนับสนุน Application ที่มีความหลากหลายด้านภาษาในการแสดงผล



ภาพที่ 2.11 ตัวอย่างโค้ดของ Django Framework

2.1.6 เทคนิค Haarcascade_frontalface_default

Haarcascade คือ การตรวจสอบวัตถุที่ใช้พื้นฐานของ Haar-feature ซึ่งได้ทำการปรับกระบวนการทำงานโดยนำแนวคิด Haar wavelet และพัฒนาคุณลักษณะของแนวคิด Haar-like feature โดยพิจารณาจากรูปสี่เหลี่ยมที่อยู่ติดกัน พื้นที่ต่างๆในภาพจะถูกกำหนดและปรากฏในหน้าต่างการตรวจสอบ จากนั้นทำการสรุปความเข้มของแต่ละพิกเซลจากแต่ละส่วนของภาพและคำนวณผลรวมเพื่อทำกระบวนว่าบริเวณใดคือใบหน้า โดยจะใช้ Haar-like features ในการตรวจสอบความเข้มของสีและกำหนดค่าน้ำหนัก โดย Haarcascade_frontalface_default จะทำการจับภาพเฉพาะส่วนใบหน้าของบุคคลเท่านั้น

2.1.7 เทคนิค Local Binary Pattern Histogram

Local Binary Pattern (LBP) เป็นเทคนิคที่มีประสิทธิภาพมากในการในการเก็บคุณสมบัติทางด้านพื้นผิว การทำงาน พื้นฐานของ LBP คือคำนวณหาค่าค่าหนึ่งเพื่อใช้เป็นตัวแทนของพื้นที่ขนาด 3x3 พิกเซล โดยใช้จุดศูนย์กลางของพื้นที่ เป็นค่าที่ใช้อ้างอิงในการคำนวณ ผลที่ได้จะอยู่ในรูปของ แพทเทิร์นของเลขฐานสองซึ่งสามารถแปลงเป็น Histogram เพื่อแสดงคุณสมบัติทางด้านพื้นผิวของภาพได้ดังภาพที่ 2.12 แสดงการทำงานพื้นฐานของ LBP

3 x 3 Pixel Data			Threshold			Weight		
8	5	2	1	0	0	1	2	4
9	5	4	1		0	128		8
1	7	6	0	1	1	64	32	16
Pattern = 10110001								
LBP = 128 + 32 + 16 + 1 = 177								

ภาพที่ 2.12 พื้นฐานการทำงานของ Local Binary Pattern

บทที่ 3

แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน

3.1 แผนงานการปฏิบัติงาน

ตารางที่ 3.1 ตารางการปฏิบัติงาน

หัวข้องาน	มิ.ย. 61			ก.ค. 61			ส.ค. 61			ก.ย. 61		
ปฏิบัติงานที่ได้รับมอบหมาย												
ศึกษาภาษาและโค้ด												
ศึกษาภาษา Python												
ศึกษา SQLite												
ศึกษา Face Recognition												
เขียนโค้ด Face Recognition												
เขียนโค้ดเชื่อมกับ SQLite												
ทดสอบระบบ												

3.2 รายละเอียดที่นักศึกษาปฏิบัติในการฝึกงาน

จากการได้มาสหกิจศึกษาที่ บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) ทางบริษัท ได้มีโครงการใหญ่ที่ต้องส่งมอบให้ลูกค้า และมีความต้องการทำให้รายงานแสดงผลเร็วขึ้น จึงได้มีส่วนร่วมในการเข้าทำโครงการนี้ด้วย ซึ่งงานที่ได้รับมอบหมายนั้นมีตั้งแต่ ทำ Stored Procedures, แก้ไขบั๊กต่างๆ, จัดทำรายงาน, จัดทำหน้าเว็บฯ จากนั้นทางบริษัทมีความต้องการที่จะทำ ระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการเช็คชื่อเข้าทำงานขึ้น จึงได้รับหน้าที่ศึกษาและจัดทำระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการเช็คชื่อเข้าทำงาน เพื่อที่จะนำไปพัฒนาและนำไปใช้ต่อไป

3.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติงาน

3.3.1 งานประจำที่ได้รับมอบหมาย

งานประจำที่ได้รับมอบหมายจากการปฏิบัติสหกิจศึกษาทั้ง 4 เดือน ที่ บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) แบ่งออกเป็นงานหลักที่ทำต่อเนื่อง และงานย่อยที่ทำในบางครั้ง โดยที่งานประจำที่เป็นงานหลัก คือ การทำ Stored Procedures และ งานงานประจำที่เป็นงานย่อย คือ การแก้ไข Bug, การแก้ไข Report และการทำ Web Chat โดยมีรายละเอียดดังนี้

1). การทำ Stored Procedures

การทำ Stored Procedures คือ การเขียนโค้ด SQL ลงใน SQL Server เพื่อให้หน้า Report มีการประมวลที่เร็วขึ้น และสามารถแสดง Report ได้เร็วยิ่งขึ้น

```
where
(
  ((@TapCheck = '0') and ((DRED.SrOwner in (select * from dbo.[SplitString](@brn,','))) or (DRED.SrOwner in (case when @brn = 'GSC' then ('')end ))))
  or
  ((@TapCheck = 'P') and ((DRED.SrLocCode in (select * from dbo.[SplitString](@brn,','))) or (DRED.SrLocCode in (case when @brn = 'GSC' then ('')end ))))
)
or
(
  ((@TapCheck = '' or @TapCheck is null) and ((DRED.SrOwner in (select * from dbo.[SplitString](@brn,','))) or
  (DRED.SrLocCode in (case when @brn = '' or @brn is null then (DRED.SrLocCode) end))
  or (DRED.SrLocCode in (select * from dbo.[SplitString](@brn,',')))
  or (DRED.SrOwner in (case when @brn = 'GSC' then ('')end ))
  or (DRED.SrLocCode in (case when @brn = 'GSC' then ('')end ))))
)
)
and
(
  (@Type = '' or @Type is null) and (DRED.SrType = DRED.SrType) or
  (@Type = 'R') and (DRED.SrType = 'R') or
  (@Type = 'P') and (DRED.SrType = 'P') or
  (@Type = 'C') and (DRED.SrType = 'C')
)
)
and
(
  (@Status = '' or @Status is null) and (DRED.SrStatus = DRED.SrStatus) or
  (@Status = 'AT') and (DRED.SrStatus = 'AT') or
  (@Status = 'IT') and (DRED.SrStatus = 'IT')
)
)
and
(
  (@State = '' or @State is null) and (DRED.SrState = DRED.SrState) or
  (@State = 'IN') and (DRED.SrState = 'IN') or
  (@State = 'OUT') and (DRED.SrState = 'OUT') or
  (@State = 'LOST') and (DRED.SrState = 'LOST')
)
)
and
(
  ((@PlateNo is null or @PlateNo = '') and (DRED.SrPlate = DRED.SrPlate)) or
  ((@PlateNo <> null or @PlateNo <> '') and (DRED.SrPlate = @PlateNo))
)
)
and
(
  ((@Compcode = '' or @Compcode is null) and (DRED.SrComp = DRED.SrComp)) or
  ((@Compcode <> '' or @Compcode <> null) and (DRED.SrComp = @Compcode))
)
)
```

ภาพที่ 3.1 ตัวอย่างโค้ด Stored Procedures ใช้ and/or

จากภาพที่ 3.1 เป็นการทำให้ Stored Procedures โดยเป็นการเขียนเงื่อนไขของการทำงานของตัวโปรแกรมตามความต้องการของลูกค้า โดยที่การตั้งเงื่อนไขมากกว่า 1 เงื่อนไข จึงมีการใช้ and/or เพื่อบ่งบอกว่าแต่ละเงื่อนไขมีการทำงานด้วยกันหรือไม่ เช่น ถ้า เป็น and เงื่อนไขทั้ง 2 จะต้องเข้าเงื่อนไขนั้นทั้งคู่ และ ถ้าเป็น or เงื่อนไขอย่างใดอย่างหนึ่งถูกก็จะเข้าเงื่อนไขนั้น เป็นต้น

```

select * into #BFOC from (
select * from DSAL_BFOC
where BmRemoveParts = 'N'
) as BFOC

select * into #qry from (
select
cts.EntryId as EntryId,
cts.CsMtocCode as mtocCode,
cts.CsBranch as branch,
cts.CsFrameNo as frameNo,
cts.CsEngine as engineNo,
(case
when cts.CsRecR3Date is null then CAST(NULL AS datetime2)
else cts.CsRecR3Date end) as r3hDate,
(case
when cts.CsRecBranchDate is null then CAST(NULL AS datetime2)
else cts.CsRecBranchDate end) as rcvDate,
(case
when cts.CsAllocateDate is null then CAST(NULL AS datetime2)
else cts.CsAllocateDate end) as allDate,
(case
when cts.CsAllocateDate is null then 'N'
else 'Y' end) as Allocate,
cts.CsBkNo as bookingNo,
cts.CsBmemoNo as bmemoNo,
cts.CsParking as CsParking,
BHD.BmSINO as CsSINO,
cts.CsStsDesc as CsStsDesc,
cts.CsAdvSts as CsAdvSts,
BHD.BmPDIOpen as CsPDISts,
BHD.BmPDIClose as BmPDIClose,
BHD.BmPDIOpen as BmPDIOpen,

```

ภาพที่ 3.2 ตัวอย่างโค้ด Stored Procedures การสร้าง Temp Table

จากภาพที่ 3.2 เป็นการสร้าง Temp Table โดยมีการตั้งชื่อของ Temp Table ที่จะทำการสร้างและข้างในของ Temp Table นั้นก็จะทำการเขียน SQL แบบปกติได้ โดยสามารถ Join Database หรือ สร้างเงื่อนไขได้ หลังจากทำโค้ดภายใน Temp Table เสร็จต้องทำการปิดและประกาศชื่อให้แต่ละ Temp Table เพื่อไม่ให้เกิดการ Error ขึ้น

2). การแก้ไข Bug

การแก้ไข Bug ที่ทำการแก้ไขตามที่พี่เลี้ยงมีการสั่งให้ช่วยแก้ไข โดยการแก้ไขส่วนมากจะเน้นไปทางการแสดงข้อมูลที่ผิดหรือไม่ตรงกัน โดยทางแก้ไขที่ดีที่สุดคือการเช็คเงื่อนไขต่างๆ ใหม่เนื่องจากการแสดงข้อมูลที่ผิดพลาดส่วนใหญ่เกิดจากการเขียนเงื่อนไขที่ผิด

ตารางที่ 3.2 Bug ที่แก้ไข

Bug ที่เกิดขึ้น	การแก้ปัญหา
1. การแสดงข้อมูลไม่ตรงกัน เนื่องจากการเขียนเงื่อนไขโปรแกรมผิด เช่น การใช้ and/or ที่ผิดจะส่งผลให้มีการแสดงข้อมูลที่มากกว่าหรือน้อยกว่า ข้อมูลจริงที่ควรแสดง	ทำการเปลี่ยนเงื่อนไขใหม่ เช่น การเปลี่ยน and เป็น or หรือ ทำการเขียนเงื่อนไขใหม่ทั้งหมดเพื่อให้ได้ข้อมูลที่ถูกต้อง

ตารางที่ 3.2 Bug ที่แก้ไข (ต่อ)

Bug ที่เกิดขึ้น	การแก้ปัญหา
2. เกิดปัญหา Timeout ในการรัน Report เนื่องจากการเขียน Logic ไม่มีประสิทธิภาพมากพอ	เขียนโค้ดด้วย Logic ใหม่ เช่น ทำการเปลี่ยนเงื่อนไขที่เคยทำ หรือ ทำการสร้าง Temp Table เพื่อกระจายการทำงาน เป็นต้น
การแสดงผลวันที่จากเครื่อง Format ไม่ถูกต้อง เนื่องจากนำโค้ดเก่ามาใช้	ทำการเปลี่ยนคำสั่งการแสดงผลเวลาให้ถูกต้องตาม Format ที่ลูกค้าต้องการ

3). การแก้ไข Report

การแก้ไข Report จะทำหลังจากเมื่อโค้ดเบื้องต้นเสร็จสมบูรณ์แล้ว จากนั้นก็จะทำการสร้าง Report ให้ตรงกับความต้องการของลูกค้า โดยที่ Report จะรูปแบบและขนาดที่ใกล้เคียงกันทั้งหมด ตัวอย่างดังแสดงที่ตารางที่ 3.3

ตารางที่ 3.3 Report ที่แก้ไข

Report ที่แก้ไข
1. ทำการจัดหน้าของ Report ใหม่ให้ตรงตามความต้องการของลูกค้า
2. ทำส่วนการคำนวณให้กับ Report ที่มีการคำนวณ
3. ทำการสร้างปุ่มกดตามความต้องการลูกค้า
4. ทำการเชื่อมข้อมูลของ Report และ โค้ดที่เขียนเข้าด้วยกัน
5. ทำการสร้างหน้า Report ใหม่ทั้งหมด

4). การทำ Web Chat

การทำ Web Chat นั้นได้รับมอบหมายจากพี่เลี้ยงเพื่อให้อำนาจในการจัดทำหน้าตา และรูปแบบในบางส่วนของ Web Chat โดยทำการเขียนโค้ด Javascript (ดังภาพที่ 3.3) เพื่อให้มีรูปแบบตามแบบที่ลูกค้าต้องการ หลังจากนั้นจะทำการปรับขนาด, รูปแบบ และ ตำแหน่งของตัวอักษรให้ดูสวยงาม ต่อมาจะทำการตกแต่งให้มีสีสันและความสวยงามตามความต้องการของ

ลูกค้า และสุดท้ายการทำ Responsive เพื่อให้หน้า Web Chat มีการขยาย - ลด หน้าจอได้โดยที่ข้อมูลภายในไม่เสียรูปแบบไป

```
<div className="right-mid">
  <div className="info">
    <table>
      <tbody>
        <tr>
          <td className="leftID">
            <span>Customer Type</span>
          </td>
          <td className="rightID">
            <span>{this.state.customer}</span>
          </td>
        </tr>
        <tr>
          <td className="leftID">
            <span>FirstName</span>
            <span className="required">*</span>
          </td>
          <td className="rightID">
            <input
              type="text"
              className="input-field"
              name="field3"
              placeholder=" "
              pattern="[A-Za-z\s]+"
              title="กรุณาพิมพ์ชื่อ"
              value={this.state.new_firstname}
              ref={this.firstname}
              onChange={this.handlefirstname}
              .bind(this)} />
            <labelerror className='hide' id='firstname'>{this.state.errFirstname}</labelerror>
          </td>
        </tr>
      </tbody>
    </table>
  </div>
</div>
```

ภาพที่ 3.3 ตัวอย่างโค้ด JavaScript การทำ Web Chat

3.3.2 งานโครงการที่ได้รับมอบหมาย

1). ศึกษา Python

Python เป็นภาษาเขียนโปรแกรมระดับสูง ซึ่งจำเป็นต้องใช้ภาษานี้ เนื่องจากเป็นภาษาที่สะดวกต่อการเขียนโค้ดระดับสูง ซึ่งเป็นภาษาที่ต้องมีการศึกษาเพิ่มเติมเนื่องจากไม่เคยศึกษาและเขียนภาษา Python มาก่อน แต่มีโปรแกรมที่เป็นเครื่องมือในการช่วยจัดทำโค้ด ทำให้การเขียนโค้ดด้วยภาษา Python มีความสะดวกมากขึ้น

2). ศึกษา SQLite

SQLite เป็นโปรแกรมฐานข้อมูลที่ไว้สำหรับจัดเก็บฐานข้อมูลเป็นไฟล์โดยไม่จำเป็นต้องมีเซิร์ฟเวอร์ ซึ่งการเก็บข้อมูลต่างๆ จากโค้ดจำเป็นต้องมีพื้นที่ไว้สำหรับเก็บบันทึกข้อมูลต่างๆ ไว้ จึงต้องทำการศึกษา SQLite เพื่อเขียนโค้ดทำให้สามารถเชื่อมระบบที่ทำกับ SQLite ได้

3). ศึกษา Face Recognition

Face Recognition คือระบบการตรวจหาใบหน้าของมนุษย์และปรับภาพใบหน้าโดยอัตโนมัติ กรอบจะปรากฏขึ้นบนใบหน้าที่ถูกตรวจจับ เป็นระบบที่ไม่เคยศึกษาก่อน จึงต้องทำการศึกษาและทำความเข้าใจเกี่ยวกับ Face Recognition โดยแบ่งออกเป็น 3 ไฟล์ คือ

3.1). ไฟล์ face_datasets.py

ไฟล์ face_datasets.py เป็นไฟล์สำหรับถ่ายภาพใบหน้าของบุคคลโดยในการถ่ายภาพนั้น ผู้ควบคุมระบบต้องมีการกำหนดไอดีให้กับบุคคลก่อนรันไฟล์ face_datasets.py เพื่อถ่ายภาพ เนื่องจากการถ่ายภาพจะบันทึกภาพตามไอดีที่มีการใส่ค่าลงไปโปรแกรม (ภาพที่ 3.1) หลังจากนั้นเมื่อทำการรันไฟล์ face_datasets.py เรียบร้อย ก็จะต้องทำงานอัตโนมัติและทำการถ่ายภาพจำนวน 100 ภาพ โดยใช้เวลา 0.1 วินาทีต่อการถ่ายภาพ 1 ภาพ (ดังภาพที่ 3.2) หลังจากทำการถ่ายภาพครบ 100 ภาพก็จะต้องทำการปิดเองอัตโนมัติ และภาพจะถูกจัดเก็บไว้ที่ Folder ชื่อ dataset

```
# For each person, one face id
face_id = 1
```

ภาพที่ 3.4 โค้ดส่วนการใส่ไอดีบุคคล

```
# Save the captured image into the datasets folder
cv2.imwrite("dataset/User." + str(face_id) + '.' + str(count) + ".jpg", gray[y:y+h,x:x+w])

# Display the video frame, with bounded rectangle on the person's face
cv2.imshow('frame', image_frame)

# To stop taking video, press 'q' for at least 100ms
if cv2.waitKey(100) & 0xFF == ord('q'):
    break

# If image taken reach 100, stop taking video
elif count>100:
    break
```

ภาพที่ 3.5 โค้ดส่วนชื่อของภาพที่ถ่าย และจำนวนและเวลาของการถ่ายภาพ

3.2). ไฟล์ training.py

ไฟล์ training.py เป็นไฟล์สำหรับเทรนให้ระบบสามารถวิเคราะห์ใบหน้าบุคคลได้ โดยใช้เทคนิค Haarcascade และ Local Binary Patterns Histograms ในการสร้างโมเดลโดยเมื่อรันไฟล์ training.py รูปภาพจาก Folder dataset ที่ได้จากไฟล์ face_datasets.py จะถูกนำเข้าสู่เทคนิค Haarcascade และ Local Binary Patterns Histograms (ดังภาพที่ 3.3)โดยมีการใช้ข้อมูลจากภาพถ่ายทั้งหมดและไอดีของบุคคลเพื่อให้สามารถเทรนบุคคลได้ถูกตามภาพและไอดีของบุคคลนั้นๆ และเมื่อรันไฟล์ training.py เสร็จเรียบร้อยแล้วระบบจะปิดโดยอัตโนมัติและจะได้ไฟล์โมเดล trainer ที่ถูกเก็บไว้ใน Folder ชื่อ trainer

```
# Create Local Binary Patterns Histograms for face recognition
recognizer = cv2.face.LBPHFaceRecognizer_create()

# Using prebuilt frontal face training model, for face detection
detector = cv2.CascadeClassifier("haarcascade_frontalface_default.xml");
```

ภาพที่ 3.6 โค้ดส่วนการใช้เทคนิค Haarcascade และ Local Binary Patterns Histograms

3.3). ไฟล์ face_recognition.py

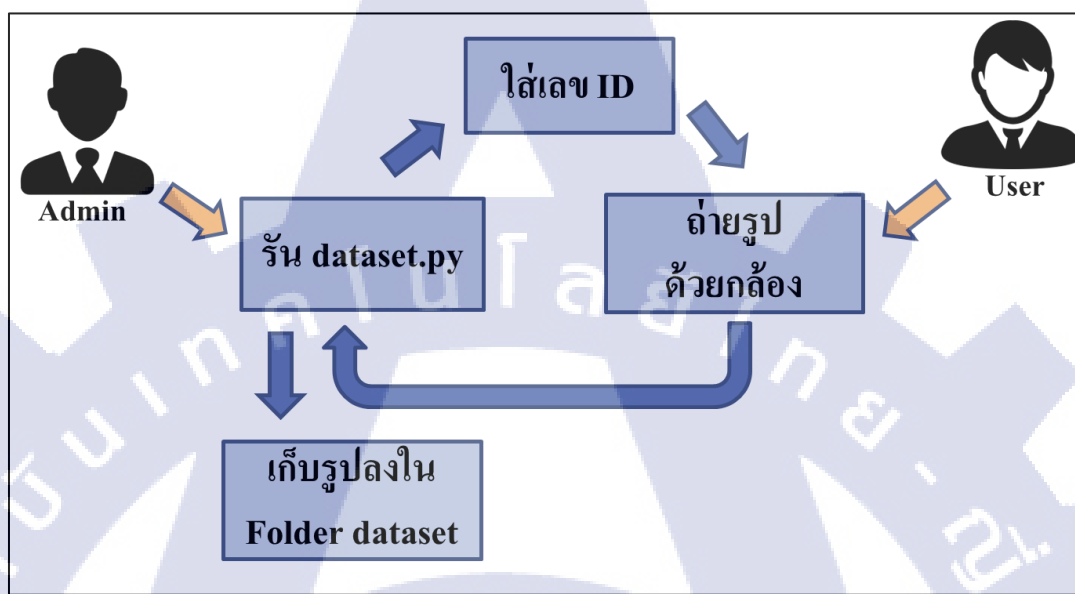
ไฟล์ face_recognition.py เป็นไฟล์สำหรับวิเคราะห์ใบหน้าบุคคลและทำการแสดงชื่อและเปอร์เซ็นต์ความคล้ายของบุคคล โดยเมื่อรันไฟล์ face_recognition.py ระบบจะทำการใช้โมเดล trainer ที่ได้จากไฟล์ training.py เพื่อนำมาเปรียบเทียบกับหน้าของบุคคลที่อยู่หน้ากล้องแบบ real time และแสดงชื่อและเปอร์เซ็นต์ความคล้ายของบุคคลตามไอดีที่มีการกำหนดไว้ (ดังภาพที่ 3.4)

```
# Check the ID if exist
if(Id == 1):
    Id = "Nazmi {0:.2f}%".format(round(100 - confidence, 2))
```

ภาพที่ 3.7 โค้ดส่วนการแสดงผลชื่อและเปอร์เซ็นต์ความคล้าย

3.3.3 ขั้นตอนการทำงานของระบบ Face Recognition

1). ส่วนการถ่ายภาพใบหน้าบุคคล

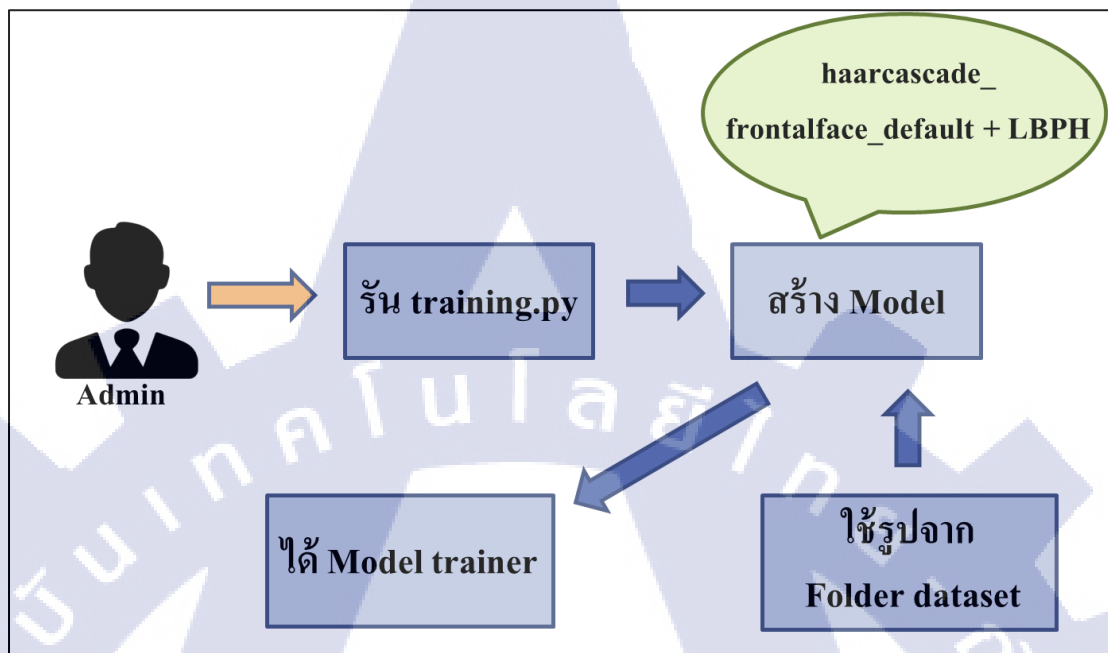


ภาพที่ 3.8 ส่วนการถ่ายภาพใบหน้าบุคคล

จากภาพที่ 3.5 การทำงานของส่วนการถ่ายภาพใบหน้าบุคคลเริ่มโดยการที่ผู้ควบคุมระบบทำการรันไฟล์ face_dataset.py เพื่อเริ่มกระบวนการทำงานของระบบในส่วนการถ่ายภาพใบหน้าบุคคล จากทำการรันไฟล์แล้วจะมีข้อมูลของบุคคลแสดงเพื่อให้ผู้ควบคุมระบบตรวจสอบชื่อ, นามสกุล และเลขไอดี เพื่อทำการใส่เลขไอดีลงไป

จากนั้นกล้องจะทำการเปิดโดยอัตโนมัติ และผู้ใช้จะทำการถ่ายภาพโดยอยู่ต่อหน้ากล้อง โดยทำตามกระบวนการที่เขียนโค้ดไว้ในไฟล์ face_dataset.py เมื่อถ่ายภาพเสร็จเรียบร้อยแล้วจะทำการปิดโดยอัตโนมัติ และภาพทั้งหมดจะถูกจัดเก็บไว้ที่ Folder dataset

2). ส่วนการประมวลผลเพื่อสร้างโมเดล

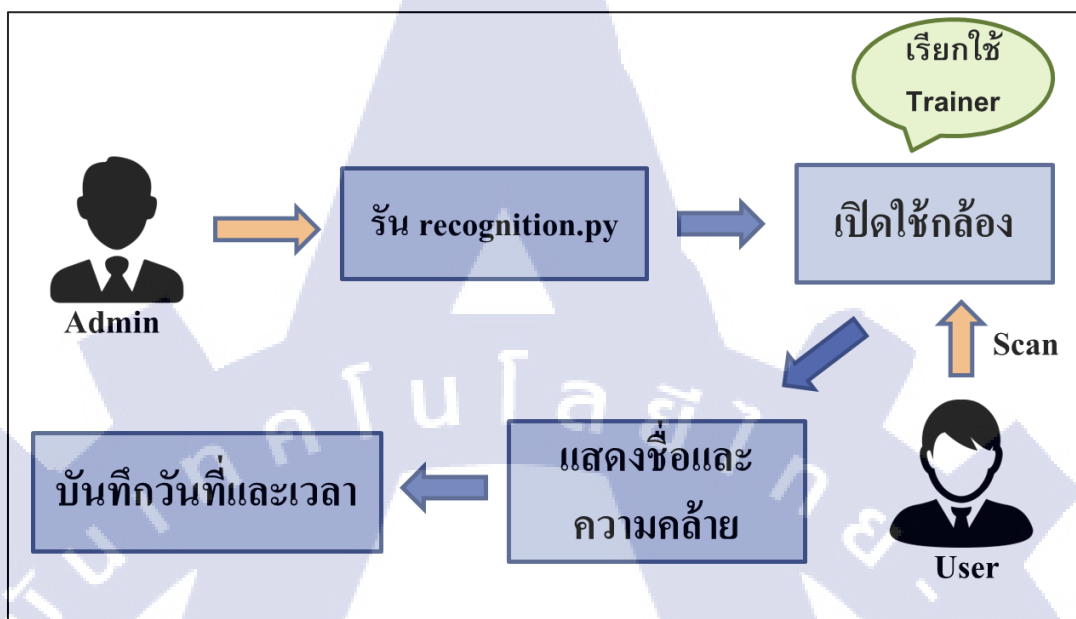


ภาพที่ 3.9 ส่วนการประมวลผลเพื่อสร้างโมเดล

จากภาพที่ 3.6 การทำงานของการประมวลผลเพื่อสร้างโมเดลนั้นเริ่มต้นโดยให้ผู้ควบคุมระบบทำการรัน `training.py` จากนั้นระบบจะทำการสร้างโมเดลผ่านเทคนิค **Haarcascade** และ **Local Binary Patterns Histograms** โดยเทคนิค **Haarcascade** นั้นใช้สำหรับการสร้างกรอบเพื่อจับเฉพาะส่วนของใบหน้าเท่านั้น และเทคนิค **Local Binary Patterns Histograms** นั้นใช้สำหรับการประมวลผลค่าของภาพเพื่อให้สามารถจดจำและนำไปเปรียบเทียบใบหน้าบุคคลได้

จากนั้นระบบจะทำการนำข้อมูลรูปภาพจาก **Folder dataset** ที่ได้จากการรันไฟล์ `face_dataset.py` จากนั้นระบบในส่วนนี้จะปิดการทำงานโดยอัตโนมัติ และได้โมเดล **trainer** ที่ถูกเก็บไว้ใน **Folder trainer**

3). ส่วนการจับภาพใบหน้าบุคคลและบันทึกวันที่และเวลา



ภาพที่ 3.10 ส่วนการจับภาพใบหน้าบุคคลและบันทึกวันที่และเวลา

จากภาพที่ 3.7 การจับภาพใบหน้าบุคคลและบันทึกวันที่และเวลาเริ่มต้นจากผู้ควบคุมระบบทำการรันไฟล์ face_recognition.py (รันไฟล์ครั้งเดียว) จากนั้นกล้องจะทำการเปิดขึ้นโดยอัตโนมัติ และทำการเรียกใช้โมเดล trainer ที่ได้จากการรันไฟล์ training.py

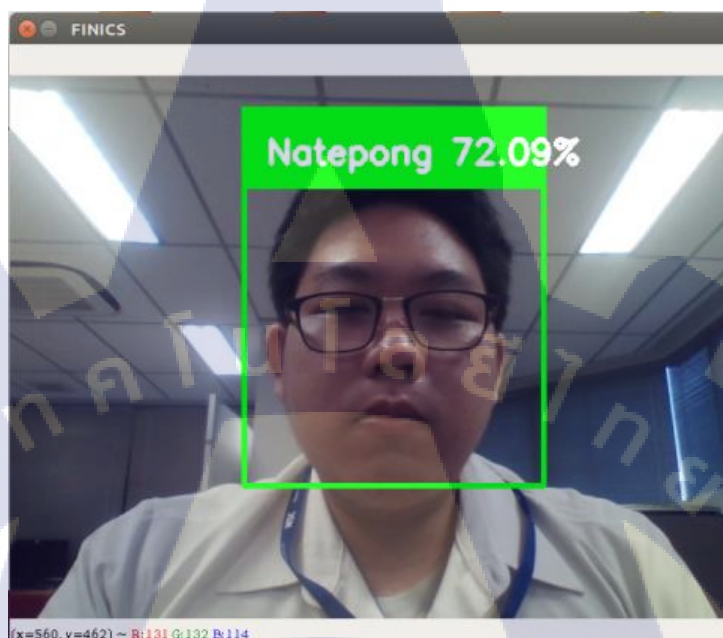
เมื่อผู้ใช้ทำการสแกนใบหน้าทีหน้ากล้องระบบจะทำการตรวจจับใบหน้าของผู้ใช้และแสดงชื่อและเปอร์เซ็นต์ความคล้ายของผู้ใช้ทันทีที่กล้องตรวจจับใบหน้าของผู้ใช้และแสดงชื่อและเปอร์เซ็นต์ความคล้ายของผู้ใช้ระบบจะทำการบันทึกวันที่และเวลา

3.3.4 ทดสอบระบบ

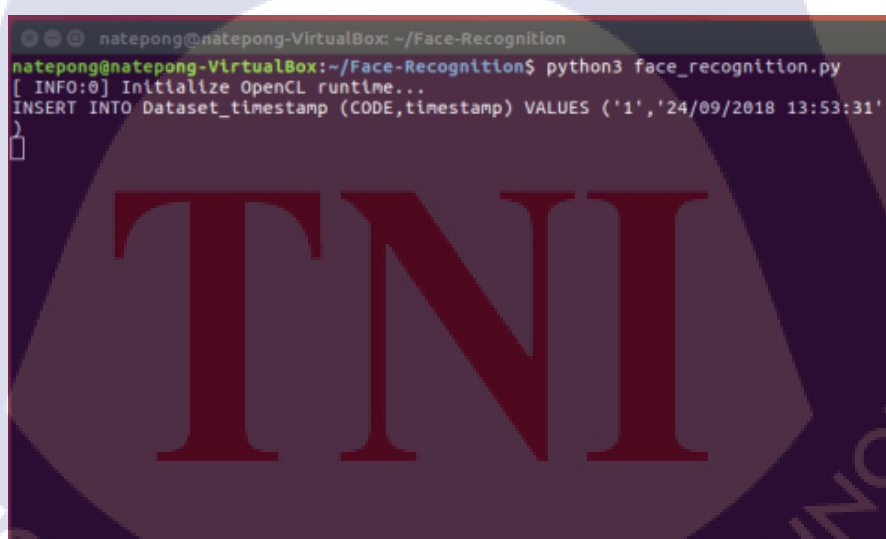
จากการทดสอบ พบว่าระบบสามารถแสดงชื่อและเปอร์เซ็นต์ความคล้ายของบุคคลได้ (ดังภาพที่ 3.8) โดยเปอร์เซ็นต์ความคล้ายของบุคคลนั้น ขึ้นอยู่กับปัจจัยหลายอย่าง ยกตัวอย่างเช่น ความสว่าง และ จำนวนภาพที่เก็บ เป็นต้น แต่ถ้ามีบุคคลที่ไม่ได้ทำการบันทึกข้อมูลลงตัวระบบ ระบบจะแสดงข้อความว่า Unknown

เมื่อระบบสามารถจับใบหน้าของบุคคลแล้ว สามารถยืนยันตัวตนของบุคคลนั้นได้แล้ว ระบบจะทำการบันทึกวันที่และเวลา (ดังภาพที่ 3.9) โดย ระบบจะทำการบันทึกให้ทุกๆ 1 นาทีเมื่อ

เป็นบุคคลคนเดิม แต่ถ้าหากไม่ใช่บุคคลคนเดิมระบบจะทำการบันทึกใหม่ทันที และถ้าหากไม่ใช่บุคคลที่อยู่ในระบบ หรือมีการแสดงชื่อว่า 'Unknown' ระบบจะไม่ทำการบันทึกข้อมูล



ภาพที่ 3.11 การแสดงชื่อและเปอร์เซ็นต์ความคล้ายของบุคคล



ภาพที่ 3.12 การบันทึกวันที่และเวลา

บทที่ 4

สรุปผลการดำเนินงาน การวิเคราะห์และสรุปผลต่าง ๆ

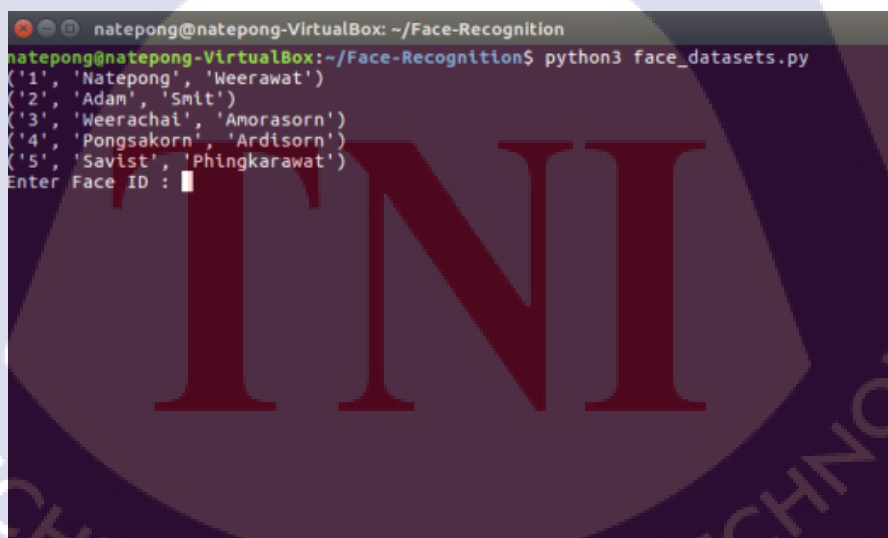
จากขั้นตอนการทดสอบระบบ ในบทที่ 3 สามารถสรุปผลการดำเนินงานผลการวิเคราะห์ข้อมูล รวมถึงการวิจารณ์ข้อมูล โดยเปรียบเทียบผลที่ได้รับกับวัตถุประสงค์ในการจัดทำโครงการ

4.1 ขั้นตอนและผลการดำเนินงาน

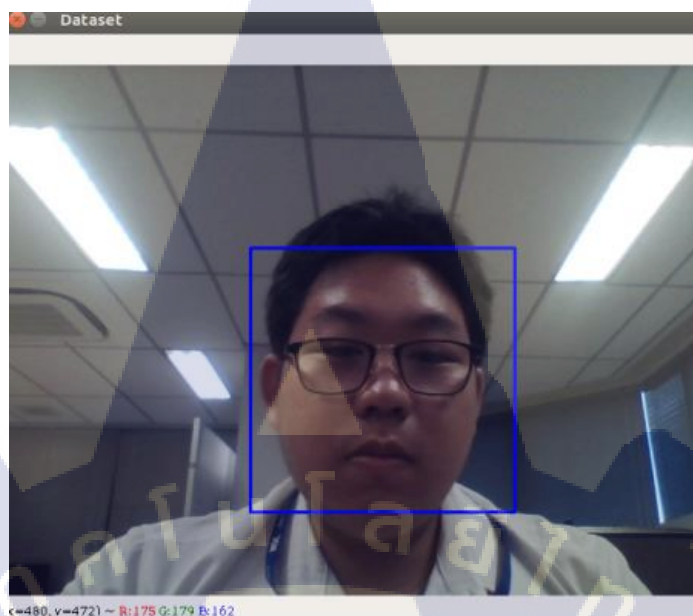
ขั้นตอนการดำเนินงาน มีการแบ่งส่วนทั้งหมด 3 ส่วน คือ

4.1.1 ส่วนจับภาพใบหน้าบุคคลสำหรับบันทึกภาพ

ระบบจะทำการแสดงชื่อ, นามสกุล และเลขไอดี และมีการให้ใส่เลขไอดีลงไป เพื่อที่จะถ่ายภาพของบุคคลที่ตรงกับเลขไอนั้น และหลังจากใส่เลขไอดีของบุคคล ดังแสดงใน ภาพที่ 4.1 จากนั้นระบบจะทำการเปิดกล้องและถ่ายภาพใบหน้าของบุคคล โดยถ่ายภาพทั้งหมด 500 ภาพ โดยแบ่งเป็นรอบ ใน 1 รอบ จะทำการจับภาพทั้งหมด 100 ภาพ และทำการถ่ายภาพไปจนครบ 500 ภาพ และระบบจะปิดกล้องโดยอัตโนมัติ ดังแสดงใน ภาพที่ 4.2



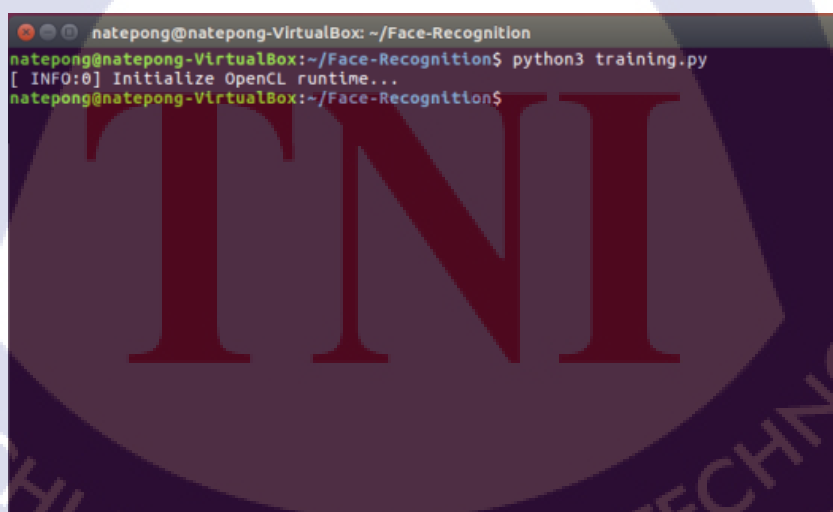
ภาพที่ 4.1 การแสดงชื่อและรหัส และข้อความเพื่อให้ใส่เลขไอดีของบุคคล



ภาพที่ 4.2 การจับภาพของบุคคล

4.1.2 ส่วนที่การฝึกให้ระบบจดจำภาพของบุคคล

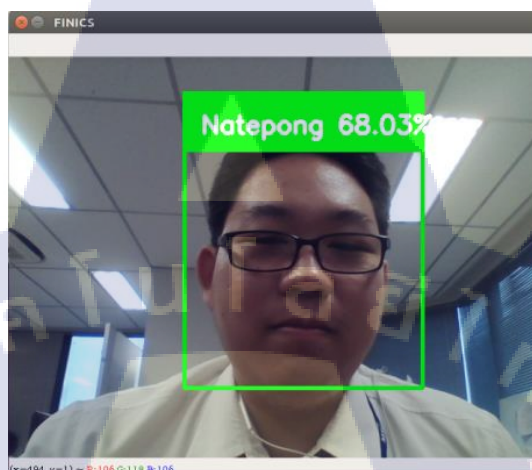
ระบบจะทำการหาค่าความแตกต่างระหว่างภาพและเลขไอดีเพื่อระบุตัวบุคคล ระบบจะไม่ทำการแสดงข้อความใดๆ ออกมาให้ผู้ใช้หรือผู้ควบคุมระบบเห็น แต่จะมีการแสดงเพียงว่าระบบกำลังประมวลผลอยู่เท่านั้น ดังแสดงใน ภาพที่ 4.3



ภาพที่ 4.3 การฝึกให้ระบบจดจำภาพของบุคคล

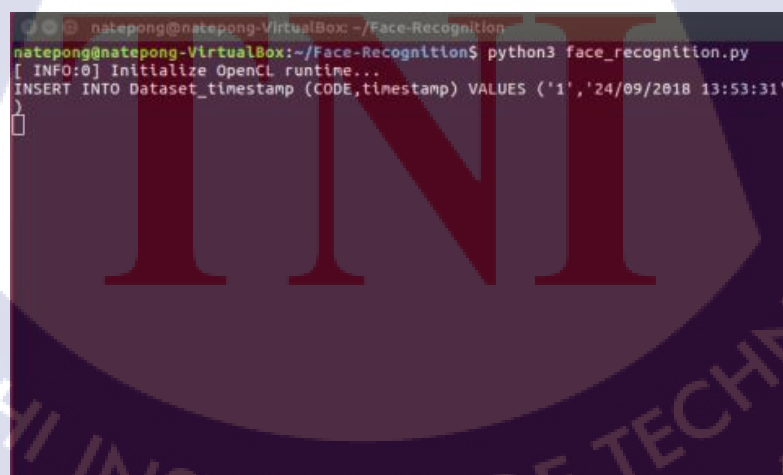
4.1.3 ส่วนที่แสดงชื่อในภาพแบบ Real Time และบันทึกเวลา

ระบบจะทำการเปิดกล้องและจับภาพใบหน้าบุคคลที่อยู่ต่อหน้ากล้อง และแสดงชื่อของบุคคล และแสดงค่าความเหมือนของบุคคลเป็นเปอร์เซ็นต์ และระบบ ดังแสดงในภาพที่ 4.4



ภาพที่ 4.4 การแสดงชื่อในภาพและเปอร์เซ็นต์ความเหมือนแบบ Real Time

หลังจากที่กล้องตรวจจับใบหน้าและได้ค่าเปอร์เซ็นต์ความคล้ายตามเกณฑ์ที่ตั้งไว้ (มีความคล้ายมากกว่า 50%) ระบบจะทำการบันทึกวันที่และเวลาตามเวลาของเครื่อง ดังแสดงในภาพที่ 4.5 โดยถ้าผู้ใช้งานยังคงอยู่ต่อหน้ากล้อง ระบบจะทำการสแกนหน้าอีกครั้งและบันทึกลงระบบในทุกๆ 1 นาที



ภาพที่ 4.5 การบันทึกวันที่และเวลา

4.2 ผลการวิเคราะห์ข้อมูล

จากการทดสอบระบบ เป็นการทดสอบเพื่อยืนยันระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการเช็คชื่อเข้าทำงานนั้น ใช้งานได้และสามารถเชื่อมต่อข้อมูลระหว่าง Face Recognition กับ SQLite เพื่อแสดงผลได้อย่างถูกต้อง จากการทดสอบระบบทดสอบจากผู้ทดสอบทั้งหมดจำนวน 3 คน มีการทำงานโดยระบุตัวบุคคลถูกต้องทั้งหมด 3 คน โดยที่การใช้งานจะต้องนั่งในระดับหนึ่ง และมีระยะห่างระหว่างใบหน้าบุคคลกับกล้องประมาณ 35 เซนติเมตร และมีระดับความสว่างอยู่ในระดับใกล้เคียงกับที่ทำการถ่ายภาพไว้

ในบางครั้งมีความผิดพลาดเกิดขึ้นจากการที่ผู้ทดสอบมีการขยับใบหน้ามากเกินไป หรือทำการใช้ระบบในสถานที่ที่มีความสว่างน้อยเกินไป ทำให้ระบบอาจจะมีการแสดงชื่อและเลขไอดีผิดพลาดได้

ในกรณีที่มีการใช้งานระบบโดยมีผู้ใช้มากกว่า 1 คนอาจจะทำให้การแสดงผลชื่อและเลขไอดีมีความผิดพลาดได้เนื่องจากการจับภาพใบหน้าของบุคคลมากกว่า 1 คน นั้นสามารถทำให้ระยะห่างระหว่างใบหน้าและกล้องมีระยะห่างที่มากเกินไปจนทำให้เกิดการแสดงผลชื่อและเลขไอดีที่ผิดพลาด

4.3 วิเคราะห์และวิจารณ์ข้อมูลโดยเปรียบเทียบผลที่ได้รับกับวัตถุประสงค์และจุดมุ่งหมายในการปฏิบัติงานหรือการจัดทำโครงการ

จุดมุ่งหมายในการทำ Face Recognition และ Admin site ในครั้งนี้เป็นการทำเพื่อสร้างจุดเริ่มต้นของการบันทึกเวลาเข้าทำงานที่สามารถระบุตัวบุคคลได้โดยผ่านกล้องและสามารถบันทึกวันที่และเวลาได้ ซึ่งสามารถนำไปปรับใช้ได้มากมาย และสามารถนำไปพัฒนาต่อเพื่อให้มีความสมบูรณ์ยิ่งขึ้น

ซึ่งผลลัพธ์จากการทดสอบพบว่า ระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการบันทึกเวลาเข้าทำงานนั้น ใช้อุปกรณ์ที่สามารถจัดเก็บข้อมูลและแสดงผลได้ในระดับหนึ่งในอนาคตสามารถเพิ่มความสามารถด้านความแม่นยำของระบบอาจต้องเปลี่ยนอุปกรณ์ตรวจจับให้มีประสิทธิภาพมากขึ้น

ตารางที่ 4.1 ตารางเปรียบเทียบวัตถุประสงค์และผลที่ได้รับกับวัตถุประสงค์

วัตถุประสงค์	ผลที่ได้รับกับวัตถุประสงค์
1. เพื่อพัฒนาต้นแบบของระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการบันทึกเวลาเข้าทำงานสำหรับพนักงาน	สร้างตัวต้นแบบของระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการบันทึกเวลาเข้าทำงานสำหรับพนักงานที่สามารถใช้ได้จริง
2. เพื่อบริษัทสามารถนำต้นแบบของระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการบันทึกเวลาเข้าทำงานไปใช้งานจริงในอนาคต	บริษัทสามารถนำต้นแบบของระบบวิเคราะห์ใบหน้าและยืนยันตัวบุคคลเพื่อการบันทึกเวลาเข้าทำงานไปใช้งานจริงและสามารถนำไปพัฒนาต่อได้
3. เพื่อพัฒนาทักษะในการทำงานจากประสบการณ์จริง	นำทักษะที่ได้เรียนรู้มามารับใช้ในการเรียนและการทำงานได้

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการที่ได้ปฏิบัติงานสหกิจศึกษาที่ บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) ผู้ปฏิบัติงานได้รับมอบหมายให้ทำ Stored Procedures เพื่อให้เว็บแอปพลิเคชันทำการแสดงผลรีพอร์ตได้อย่างมีประสิทธิภาพและรวดเร็วขึ้น และผู้ปฏิบัติงานได้รับมอบหมายให้จัดทำส่วนหน้ารีพอร์ตและส่วนหน้าเว็บแอปพลิเคชัน เพื่อให้ตรงกับความต้องการของลูกค้า และผู้ปฏิบัติงานได้รับมอบหมายให้ทำโปรแกรมระบบวิเคราะห์ใบหน้าและยืนยันตัวตนบุคคลเพื่อการบันทึกเวลาเข้าทำงานสำหรับ บริษัท เมโทรซิสเต็มส์คอร์ปอเรชั่น จำกัด (มหาชน) โดยดำเนินงานโดยใช้ภาษา Python ในการเขียนผลการดำเนินงานเป็นไปได้อย่างดีและประสบความสำเร็จเป็นอย่างดี ได้มีการสร้างโปรแกรมที่สามารถบันทึกใบหน้าบุคคลและแสดงชื่อของบุคคลและความคล้ายของบุคคล และมีการเชื่อมข้อมูลกับหน้าเว็บ Admin เพื่อเป็นแนวทางในการพัฒนาโปรแกรมระบบวิเคราะห์ใบหน้าและยืนยันตัวตนเพื่อการบันทึกเวลาเข้าทำงานในอนาคต

ปัญหาที่เกิดขึ้นประจำในการทำงาน ได้แก่

1. การแสดงชื่อบุคคลและแสดงเปอร์เซ็นต์ความคล้ายมีความผิดพลาด
2. การเชื่อมข้อมูลไปยังหน้าเว็บ Admin ผิดพลาด

5.2 แนวทางการแก้ไขปัญหา

จากปัญหาที่กล่าวมามีแนวทางแก้ปัญหาดังตามอันดับ ดังนี้

1. ทำการถ่ายภาพเพิ่มเพื่อให้โปรแกรมมีการจดจำใบหน้าของบุคคลเพิ่มขึ้นและสามารถแสดงชื่อและเปอร์เซ็นต์ความคล้ายเพิ่มขึ้น
2. ทำการเพิ่มโค้ดส่วนที่เชื่อมต่อโปรแกรมกับหน้าเว็บ Admin เพื่อรับ – ส่งข้อมูลต่างๆ ใน SQLite

5.3 ข้อเสนอแนะจากการดำเนินงาน

ควรศึกษาภาษา Python เพื่อช่วยทำให้การเขียนหรืออ่านโค้ดมีความสะดวกในการเขียนโปรแกรมมากขึ้น แล้วควรระมัดระวังการเขียนโค้ด Python เนื่องจากภาษา Python หากมีการเว้นหรือมีพื้นที่ว่างเกินมาจากปกติภาษา Python นั้นจะเกิด Error ขึ้น และการเขียนภาษา Python ควรใช้โปรแกรม PyCharm เพื่อความสะดวกในการเขียนภาษา Python เนื่องจาก โปรแกรม PyCharm สามารถแสดงการเว้นวรรคหรือช่องว่างที่เกินมา และแสดงว่ามีส่วนไหนของโค้ดมี Error ที่ควรแก้ไข



เอกสารอ้างอิง

1. Python Software Foundation, 2554, **Python3 Tutorial** [Online], Available : <https://docs.python.org/3/tutorial/> [30 พฤษภาคม 2561]
2. JetBrains s.r.o. Developed with drive and IntelliJ IDEA, 2553, **Pycharm for Ubuntu 16.04** [Online], Available : <https://www.jetbrains.com/pycharm/download/#section=linux> [30 มิถุนายน 2561]
3. Wasin Thiengkunakrit, 2560, **เริ่มพัฒนา Web Application กับภาษา Python ด้วย Django Framework** [Online], Available : <https://codeburst.io/เริ่มพัฒนา-Web-Application-กับภาษา-Python-ด้วย-Django-Framework-38ce132ac706> [16 กรกฎาคม 2561]
4. wannaphong, 2560, **Face Recognition API for Python 3** [Online], Available : <https://python3.wannaphong.com/2017/03/face-recognition-python.html> [01 สิงหาคม 2561]

สถาบันเทคโนโลยีไทย-ญี่ปุ่น

ภาคผนวก

TNI

THAI - JAPANESE INSTITUTE OF TECHNOLOGY

ภาคผนวก ก.
รายงานประจำสัปดาห์

TNI

ประวัติผู้จัดทำโครงการ



ชื่อ – สกุล

ณธิพงศ์ อธิพิทักษ์เมธี

วัน เดือน ปีเกิด

11 ตุลาคม 2539

ประวัติการศึกษา

ระดับประถมศึกษา

โรงเรียนสารสาสน์วิเทศร่มเกล้า พ.ศ. 2546

ระดับมัธยมศึกษา

มัธยมศึกษาตอนต้น โรงเรียนปทุมคงคา พ.ศ. 2552

มัธยมศึกษาตอนปลาย โรงเรียนปทุมคงคา พ.ศ. 2555

ระดับอุดมศึกษา

คณะเทคโนโลยีสารสนเทศ สาขาเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย-ญี่ปุ่น พ.ศ. 2558

ทุนการศึกษา

- ไม่มี -

ประวัติการฝึกอบรม

TNI Internship Development Program

ผลงานที่ได้รับการตีพิมพ์

- ไม่มี -