



การพัฒนาแอปพลิเคชันการชำระเงินและการเก็บข้อมูลส่วนบุคคล
DEVELOPMENT OF PAYMENT AND PERSONAL INFORMATION COLLECTION
APPLICATION

นางสาวชลธิชา กาบตุ้ม

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีไทย-ญี่ปุ่น

พ.ศ. 2561

การพัฒนาแอปพลิเคชันการชำระเงินและการเก็บข้อมูลส่วนบุคคล
DEVELOPMENT OF PAYMENT AND PERSONAL INFORMATION COLLECTION
APPLICATION

นางสาวชลธิชา กาบตุ้ม

โครงงานสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย - ญี่ปุ่น
ปีการศึกษา 2561

คณะกรรมการสอบ

..... ประธานกรรมการสอบ

(รศ.ดร.อรรณพ หมั่นสกุล)

..... กรรมการสอบ

(อาจารย์ ดร.บุษราพร เหลืองมาลาวัฒน์)

..... อาจารย์ที่ปรึกษา

(อาจารย์ ดร.โสภณ มงคลลักษณ์)

..... ประธานสหกิจศึกษาสาขาวิชา

(อาจารย์ สติลา ชีวกิดาการ)

ลิขสิทธิ์ของสถาบันเทคโนโลยีไทย - ญี่ปุ่น

ชื่อโครงการ	การพัฒนาแอปพลิเคชันการชำระเงินและการเก็บข้อมูลส่วนบุคคล
ผู้เขียน	นางสาวชลธิชา กาบคุ้ม
คณะวิชา	เทคโนโลยีสารสนเทศ สาขาวิชา เทคโนโลยีสารสนเทศ
อาจารย์ที่ปรึกษา	ดร.โสภณ มงคลลักษณ์
พนักงานที่ปรึกษา	นางสาวพริมา ผลสินธุ์
ชื่อบริษัท	ดิจิทัล (ประเทศไทย) จำกัด
ประเภทธุรกิจ/สินค้า	ผู้สร้างและพัฒนาระบบประเภทเว็บไซต์และแอปพลิเคชัน

บทสรุป

โครงการนี้มีวัตถุประสงค์เพื่อช่วยพัฒนาและปรับปรุงระบบการทำงานของแอปพลิเคชันด้านการจัดการข้อมูลส่วนบุคคลและด้านการชำระเงินผ่านบัตรเครดิตโดยสมาร์ทโฟนและอุปกรณ์ของทางบริษัทหรือเรียกว่า mPOS (Mobile Point of Sale) ซึ่งการพัฒนาแอปพลิเคชันด้านการจัดการข้อมูลส่วนบุคคลนั้นช่วยให้แอปพลิเคชันสามารถใช้งานร่วมกับเครื่อง EDC โดยสแกนบัตรได้ทันทีโดยไม่ต้องเชื่อมต่อผ่านเครือข่ายบลูทูธ และการพัฒนาแอปพลิเคชันด้านการชำระเงินช่วยปรับปรุงการทำงานให้มีประสิทธิภาพมากขึ้นโดยการตรวจสอบระบบการทำงานของแอปพลิเคชันและรายงานผลกลับไปให้ลูกค้า

- จึงได้มีการพัฒนาแอปพลิเคชันด้านการจัดการข้อมูลบัตรประชาชนและการชำระเงินผ่านบัตรเครดิตโดยสมาร์ทโฟนขึ้น เพื่อให้ตอบโจทย์ความต้องการจากทางด้านของลูกค้าของทางบริษัทและเพื่อเพิ่มฟังก์ชันทำให้แอปพลิเคชันมีความสามารถหลากหลายขึ้นในอนาคต

กิตติกรรมประกาศ

โครงการนี้สำเร็จลุล่วงได้ด้วยความกรุณาจากนางสาวพริมา ผลาสินธุ์, นางสาวพรประภา จริงกิจจานุกูล และพนักงานทุกท่านที่ได้ให้คำแนะนำ แนวคิด ตลอดจนการแก้ไขข้อบกพร่องต่าง ๆ มาโดยตลอดจนโครงการเล่มนี้เสร็จสมบูรณ์

ขอขอบคุณอาจารย์ ดร. โสภณ มงคลลักษณ์ อาจารย์ที่ปรึกษาโครงการ ที่ให้คำปรึกษาและคำแนะนำต่าง ๆ ในการทำโครงการ ทำให้โครงการสำเร็จลุล่วงไปด้วยดี

นางสาวชลธิชา กาบคุ้ม
ผู้จัดทำ

TNI

สารบัญ

บทสรุป	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
สารบัญรูป	ง
สารบัญตาราง	ช

บทที่

บทที่ 1 บทนำ	1
---------------------	----------

1.1 ชื่อและที่ตั้งของสถานประกอบการ	1
1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร	2
1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร	2
1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย	2
1.5 พนักงานที่ปรึกษา และ ตำแหน่งของพนักงานที่ปรึกษา	2
1.6 ระยะเวลาที่ปฏิบัติงาน	3
1.7 ที่มาและความสำคัญของปัญหา	3
1.8 วัตถุประสงค์หรือจุดมุ่งหมายของโครงการ	4
1.9 ผลที่คาดว่าจะได้รับจากการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย	4
1.10 นิยามศัพท์เฉพาะ	4

บทที่ 2 ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน	6
---	----------

2.1 ทฤษฎีที่เกี่ยวข้องกับคลังข้อมูล	6
2.1.1 สถาปัตยกรรมของ Android	8
2.1.1.1 ชั้นแอปพลิเคชัน (Application)	9
2.1.1.2 ชั้นแอปพลิเคชันเฟรมเวิร์ค (Application Framework)	9
2.1.1.3 ชั้นแอปพลิเคชัน (Application)	10
2.1.1.3.1 Android Runtime	11

	ง
2.1.1.4 ขั้นแอปพลิเคชัน (Application)	11
2.2 เทคโนโลยีที่ใช้ในการปฏิบัติงาน	11
2.2.1 โปรแกรม Android Studio	11
2.2.2 โปรแกรม Xampp	12
2.2.3 โปรแกรม MySQL	12
2.2.4 โปรแกรม Visual Code	13
2.2.5 โปรแกรม FileZilla	13
2.2.6 โปรแกรม Git	14
2.3 ภาษาที่ใช้พัฒนา	15
2.3.1 ภาษา Java	15
2.3.2 ภาษา PHP	16
บทที่ 3 แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน	17
3.1 แผนงานการฝึกงาน	17
3.2 รายละเอียดที่นักศึกษาปฏิบัติในการฝึกงาน	18
3.2.1 การส่งข้อมูลจากแอปพลิเคชันแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติขึ้น เซิร์ฟเวอร์ (OCR)	18
3.2.2 การหาสาเหตุปัญหาที่เกิดขึ้นจากการทำธุรกรรมการเงินผ่านแอปพลิเคชัน	24
3.2.3 การช่วยพัฒนาแอปเก็บข้อมูลบัตรประชาชนผ่านแอปพลิเคชัน	31
3.2.4 การส่งข้อมูลจากแอปพลิเคชันแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติขึ้น เซิร์ฟเวอร์ (OCR)	34
บทที่ 4 สรุปผลการดำเนินงาน การวิเคราะห์และสรุปผลต่าง ๆ	39
4.1 ขั้นตอนและผลการดำเนินงาน	39
4.1.1 การส่งข้อมูลจากแอปพลิเคชันแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติขึ้น เซิร์ฟเวอร์ (OCR)	39

	จ
4.1.2 การหาสาเหตุปัญหาที่เกิดขึ้นจากการทำธุรกรรมการเงินผ่านแอปพลิเคชัน	39
4.1.3 การช่วยพัฒนาแอปเก็บข้อมูลบัตรประชาชนผ่านแอปพลิเคชัน	39
4.1.4 การพัฒนาแอปพลิเคชัน E-Passport	40
บทที่ 5 บทสรุปและข้อเสนอแนะ	41
5.1 สรุปผลการดำเนินงาน	41
5.2 แนวทางการแก้ไขปัญหา	41
5.3 ข้อเสนอแนะจากการดำเนินงาน	41
เอกสารอ้างอิง	42
ภาคผนวก	44
รายงานประจำสัปดาห์	45
ประวัติผู้จัดทำโครงการ	62

TNI

สารบัญรูป

รูป

หน้า

รูปที่ 1.1 ตัวอย่างโลโก้ของบริษัท ดิจิโอ (ประเทศไทย) จำกัด	1
รูปที่ 1.2 ตัวอย่างแผนที่บริษัท ดิจิโอ (ประเทศไทย) จำกัด	1
รูปที่ 1.3 ตัวอย่างรูปแบบการจัดองค์กรบริษัท ดิจิโอ (ประเทศไทย) จำกัด	2
รูปที่ 2.1 ตัวอย่างรูปโครงสร้างสถาปัตยกรรมของ Android (ในบทที่ 2)	9
รูปที่ 2.2 ตัวอย่างโลโก้ Android Studio (ในบทที่ 2)	11
รูปที่ 2.3 ตัวอย่างโลโก้ Xampp (ในบทที่ 2)	12
รูปที่ 2.4 ตัวอย่างโลโก้ MySQL (ในบทที่ 2)	12
รูปที่ 2.5 ตัวอย่างโลโก้ Visual Code (ในบทที่ 2)	13
รูปที่ 2.6 ตัวอย่างโลโก้ FileZilla (ในบทที่ 2)	13
รูปที่ 2.7 ตัวอย่างโลโก้ Git (ในบทที่ 2)	14
รูปที่ 2.8 ตัวอย่างโลโก้ Java (ในบทที่ 2)	15
รูปที่ 2.9 ตัวอย่างโลโก้ PHP (ในบทที่ 2)	16
รูปที่ 3.1 ตัวอย่างทำงานของ Retrofit2 (ในบทที่ 3)	19
รูปที่ 3.2 ตัวอย่างการสร้าง Model (ในบทที่ 3)	20
รูปที่ 3.3 ตัวอย่างการสร้าง API Interface (ในบทที่ 3)	21
รูปที่ 3.4 ตัวอย่างการสร้าง Class เชื่อมต่อเซิร์ฟเวอร์ (ในบทที่ 3)	21
รูปที่ 3.5 ตัวอย่างการสร้างฐานข้อมูล (ในบทที่ 3)	22
รูปที่ 3.6 ตัวอย่าง PHP เชื่อมต่อเซิร์ฟเวอร์ (ในบทที่ 3)	22
รูปที่ 3.7 ตัวอย่างโปรแกรม PuTTY Key Generator (ในบทที่ 3)	23
รูปที่ 3.8 ตัวอย่าง Public Key (ในบทที่ 3)	23
รูปที่ 3.9 แอปพลิเคชัน ThirdParty (ในบทที่ 3)	25
รูปที่ 3.10 ตัวอย่างแอปพลิเคชันของทางบริษัทหน้า Login (ในบทที่ 3)	26
รูปที่ 3.11 แอปพลิเคชันของทางบริษัทหน้าการต่ออุปกรณ์อ่านบัตร (ในบทที่ 3)	26
รูปที่ 3.12 รอกการรูดบัตรผ่านเครื่อง MSR90 (ในบทที่ 3)	27

รูปที่ 3.13 หน้า Login ของแอปพลิเคชันการทำธุรกรรมทางการเงิน (ในบทที่ 3)	27
รูปที่ 3.14 การทำ Bypass โดยเปลี่ยนจากค่าตั้ง false เป็น true (ในบทที่ 3)	28
รูปที่ 3.15 เมื่อทำ Bypass จะเข้าเงื่อนไขการ Login สำเร็จ (ในบทที่ 3)	28
รูปที่ 3.16 หน้าโฮมของแอปพลิเคชันการทำธุรกรรมทางการเงิน (ในบทที่ 3)	28
รูปที่ 3.17 ตัวอย่างฟังก์ชันการ Login สำเร็จ (ในบทที่ 3)	29
รูปที่ 3.18 ตัวอย่างหน้าแสดงรายการ (ในบทที่ 3)	29
รูปที่ 3.19 เครื่อง MSR90 (ในบทที่ 3)	30
รูปที่ 3.20 หัวขารัจสมาร์ทโฟน (ในบทที่ 3)	30
รูปที่ 3.21 อุปกรณ์อ่านบัตรประชาชน (ในบทที่ 3)	32
รูปที่ 3.22 เครื่อง EDC (ในบทที่ 3)	32
รูปที่ 3.23 หน้าแรกของแอปพลิเคชันตัวอย่าง (ในบทที่ 3)	33
รูปที่ 3.24 หน้าแสดงข้อมูลแอปพลิเคชันตัวอย่าง (ในบทที่ 3)	33
รูปที่ 3.25 หน้าการตั้งค่าแอปพลิเคชัน (ในบทที่ 3)	34
รูปที่ 3.26 เมื่อเลือกการตั้งค่าต่าง ๆ (ในบทที่ 3)	34
รูปที่ 3.27 ตัวอย่างการเขียนไฟล์.xml (ในบทที่ 3)	35
รูปที่ 3.28 ตัวอย่างผู้ใช้งานไม่ได้กดเลือก URL (ในบทที่ 3)	35
รูปที่ 3.29 ตัวอย่างผู้ใช้งานกดเลือก URL (ในบทที่ 3)	35
รูปที่ 3.30 ตัวอย่างโค้ดการส่งข้อมูลไปยังอีเมล.xml (ในบทที่ 3)	36
รูปที่ 3.31 ตัวอย่างการกรอกข้อมูล (ในบทที่ 3)	36
รูปที่ 3.32 ตัวอย่างการส่งอีเมล (ในบทที่ 3)	37
รูปที่ 3.33 ตัวอย่างอีเมลที่ได้รับ (ในบทที่ 3)	37
รูปที่ 3.34 ตัวอย่างเนื้อหาอีเมลที่ได้รับ (ในบทที่ 3)	37
รูปที่ 3.35 ตัวอย่างไฟล์.csv (ในบทที่ 3)	38

สารบัญตาราง

ตาราง

หน้า

ตารางที่ 2.1	การพัฒนาหุ่นของ Android ตั้งแต่รุ่นที่ 1.0 – 7.0	7
ตารางที่ 2.1	การพัฒนาหุ่นของ Android ตั้งแต่รุ่นที่ 1.0 – 7.0 (ต่อ)	8
ตารางที่ 3.1	แผนงานการฝึกงาน	17



บทที่ 1

บทนำ

1.1 ชื่อและที่ตั้งของสถานประกอบการ



ภาพที่ 1.1 ตัวอย่างโลโก้ของบริษัท ดิจิโอ (ประเทศไทย) จำกัด

ชื่อ
ที่ตั้ง

บริษัท ดิจิโอ ประเทศไทย จำกัด

972/1 อาคารวรสุบิน ชั้น 4 ซอยโรงพยาบาลพระราม 9 ถนนริมคลองสามเสน
แขวงบางกะปิ เขตห้วยขวาง กรุงเทพฯ 10310

โทรศัพท์

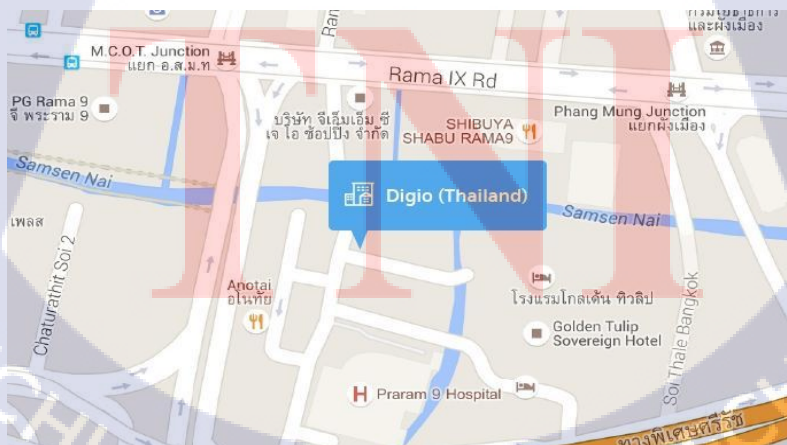
02-641-5170

โทรสาร

02-019-4499

อีเมล

support@digio.co.th

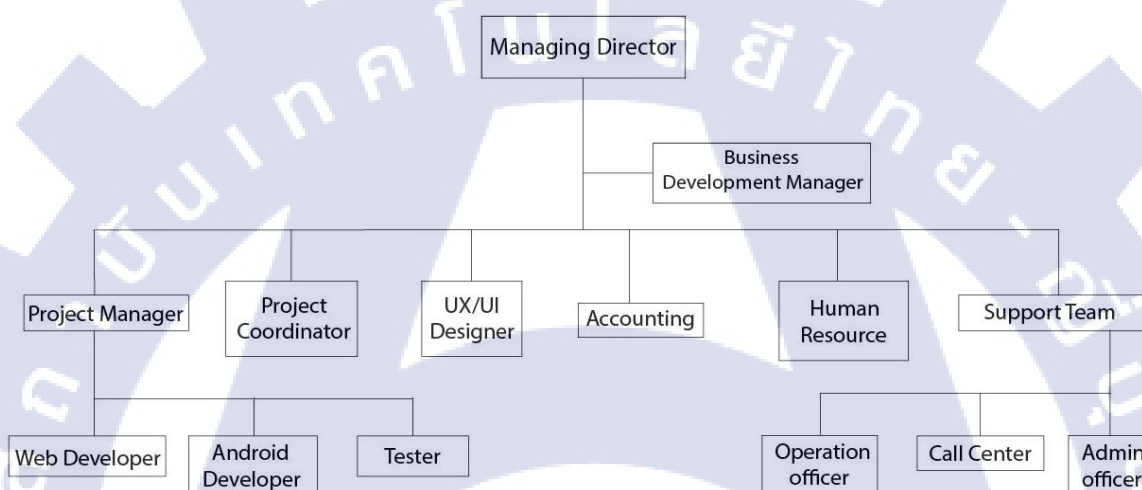


ภาพที่ 1.2 ตัวอย่างแผนที่บริษัท ดิจิโอ (ประเทศไทย) จำกัด

1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร

ออกแบบพัฒนาแอปพลิเคชันที่ใช้สำหรับการชำระเงิน โดยใช้ร่วมกับอุปกรณ์เสริมที่ทางบริษัทได้ออกแบบและพัฒนาขึ้นมา โดยสามารถเปลี่ยนสมาร์ตโฟนหรือแท็บเล็ตให้สามารถชำระเงินผ่านบัตรเครดิตได้ในทุกที่และทุกเวลา รวมไปถึงการพัฒนา ระบบ mPOS สำหรับการชำระเงินในหน้าร้านค้าและรับออกแบบเว็บแอปพลิเคชันทางการเงินของธนาคารอย่างมืออาชีพ

1.3 รูปแบบการจัดองค์กรและการบริหารองค์กร



ภาพที่ 1.3 ตัวอย่างรูปแบบการจัดองค์กรบริษัท ดิจิไอ (ประเทศไทย) จำกัด

1.4 ตำแหน่งและหน้าที่งานที่นักศึกษาได้รับมอบหมาย

ตำแหน่ง Android Developer พัฒนาแอปพลิเคชันมือถือระบบปฏิบัติการแอนดรอยด์ โดยใช้ Android studio ในการพัฒนาแอปพลิเคชันได้รับมอบหมายงานให้นำข้อมูลจากแอปพลิเคชันไปเก็บบน Server

1.5 พนักงานที่ปรึกษา และ ตำแหน่งของพนักงานที่ปรึกษา

- 1) นางสาวพริมา ผลาสินธุ์ ตำแหน่ง Project Coordinator
- 2) นางสาวพรประภา จริ่งกิจจานุกุล ตำแหน่ง Project Coordinator

1.6 ระยะเวลาที่ปฏิบัติงาน

ระยะเวลาในการปฏิบัติงานสหกิจตั้งแต่ 4 มิถุนายน พ.ศ. 2561 ถึง 28 กันยายน พ.ศ. 2561 เวลาในการปฏิบัติงานคือวันจันทร์ - วันศุกร์ เวลา 9.00 – 18.00 น.

1.7 ที่มาและความสำคัญของปัญหา

บริษัท ดิจิโอ ประเทศไทย จำกัด เป็นบริษัทที่มุ่งเน้นการพัฒนาระบบการชำระเงินผ่านสมาร์ตโฟน และให้บริการเครื่องรับชำระเงินด้วยบัตรเครดิต ซึ่งผลิตภัณฑ์ของทางบริษัทนั้นประกอบไปด้วย 4 ผลิตภัณฑ์หลักๆ ด้วยกัน ได้แก่

- 1) ระบบสำหรับรับชำระเงินของร้านค้า Merchant Solution (เป็นเครื่องให้ร้านค้าสามารถรับชำระด้วยช่องทางต่าง ๆ ได้ เช่น บัตรเครดิต, Alipay, WeChat, พร้อมเพย์ รองรับทั้งร้านค้ารายย่อยและรายใหญ่)
- 2) ระบบจัดการการขายของร้านค้า Tab2Pay (Point of Sale)
- 3) ระบบกระเป๋าเงินอิเล็กทรอนิกส์และตัวแทนธนาคาร (E-Wallet and Agent Banking Solution)
- 4) ระบบอ่านข้อมูลและยืนยันตัวตนอิเล็กทรอนิกส์ (Printable Signature and Identification Solution)
- 5) บริการด้านความปลอดภัยของระบบ (Security Qualification Service)

โดยผลิตภัณฑ์จะได้รับการพัฒนาอย่างต่อเนื่องเพื่อประสิทธิภาพการใช้งานที่ดีขึ้น การฝึกสหกิจครั้งนี้ทำให้ข้าพเจ้าได้มีโอกาสฝึกปฏิบัติงานด้วยการทำการปรับปรุงและแก้ไขปัญหาระบบของบริษัท ข้าพเจ้าได้รับมอบหมายให้ทำงานปรับปรุงแก้ไขระบบการทำงานบางส่วนของ 2 แอปพลิเคชัน คือ แอปพลิเคชันการชำระเงินด้วยบัตรเครดิต และแอปพลิเคชันการเก็บข้อมูลส่วนบุคคลและบันทึกลายเซ็น ซึ่ง 2 งานนี้คืองานที่ทางบริษัทดำเนินการอยู่ ณ เวลานั้น นอกจากนี้ข้าพเจ้าได้รับมอบหมายอีก 2 เรื่อง ดังนี้ เรื่องคือการทำงานกับเพื่อนร่วมงานสหกิจต่างสถาบันการทำแอปพลิเคชันอ่านไฟล์ภาพเอกสารให้แปลงเป็นไฟล์ข้อความโดยส่วนนี้คือการอ่าน MRZ บน Passport และส่งข้อมูลที่จัดเก็บไปยังเซิร์ฟเวอร์ เรื่องที่สองคือการพัฒนาแอปพลิเคชัน E-Passport ให้สามารถส่งข้อมูลได้หลายรูปแบบ โดยแอปพลิเคชันจะต้องสามารถส่งข้อมูลเป็นไฟล์ .csv ผ่านทาง Email จากที่กล่าวมาจะถูกอธิบายและขยายความในบทที่ 3

1.8 วัตถุประสงค์หรือจุดมุ่งหมายของโครงการ

1. ปฏิบัติงานตามที่ได้รับมอบหมาย
2. ปรับปรุงแอปพลิเคชันให้ตรงตามความต้องการของบริษัท
3. พัฒนาการทักษะการเขียนโปรแกรมจากการปฏิบัติงาน
4. แอปพลิเคชันสามารถทำงานได้จริงและมีประสิทธิภาพ
5. ฝึกการทำงานร่วมกับผู้อื่น

1.9 ผลที่คาดว่าจะได้รับการปฏิบัติงานหรือโครงการที่ได้รับมอบหมาย

1. ทักษะในการเขียนภาษาจาวามากขึ้น
2. ทักษะในการเขียนโปรแกรมพัฒนาแอปพลิเคชันแอนดรอยด์มากขึ้น
3. เรียนรู้การทำงานภายในบริษัท
4. ได้แนวคิดหรือหลักการเขียนโปรแกรมได้ดียิ่งขึ้น
5. สามารถนำความรู้ที่ได้ไปต่อยอดให้เกิดประโยชน์ในอนาคต

1.10 นิยามศัพท์เฉพาะ

1. API [1] ย่อมาจาก Application Programming Interface คือ ช่องทางการเชื่อมต่อ, ช่องทางหนึ่งที่จะเชื่อมต่อกับเว็บไซต์ผู้ให้บริการ API จากที่อื่น เป็นตัวกลางที่ทำให้โปรแกรมประยุกต์เชื่อมต่อกับโปรแกรมประยุกต์อื่น หรือเชื่อมการทำงานเข้ากับระบบปฏิบัติการ
2. Activity [2] คือ Application Component ที่ทำหน้าที่ควบคุมการแสดงผล User Interface รวมถึงควบคุมการมีปฏิสัมพันธ์ระหว่างผู้ใช้กับ User Interface ด้วย เช่น การโทรออก, การถ่ายรูป, การส่งอีเมล
3. Fragment [3] เป็น Component ที่มีความสามารถของ Activity มาใช้ ช่วยแยกการทำงานออกเป็นส่วนๆ โดยที่ไม่ให้ Activity รับภาระการทำงานจนมากเกินไป ซึ่งจุดเด่นของ Fragment ก็คือสามารถนำไปใช้งานซ้ำได้ (Reusable) โดย Activity หนึ่งตัวสามารถมี Fragment หลายๆตัวก็ได้
4. Androids [4] คือ ระบบปฏิบัติการสำหรับอุปกรณ์พกพา เช่น โทรศัพท์มือถือ แท็บเล็ต คอมพิวเตอร์ เน็ตบุ๊ก ทำงานบนลินุกซ์ เคอร์เนล เริ่มพัฒนาโดยบริษัทแอนดรอยด์

5. Clone [5] การคัดลอก หรือ ทำสำเนาให้เหมือนกับต้นฉบับ จัดเป็นการ Backup ข้อมูลรูปแบบหนึ่ง ถูกนำมาใช้ในไมโครคอมพิวเตอร์ หมายถึงคอมพิวเตอร์ที่สร้างขึ้นให้เหมือนต้นแบบ



บทที่ 2

ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

ในการปฏิบัติงานสหกิจศึกษาครั้งนี้ เป็นการนำความรู้ทางด้านทฤษฎีและเทคโนโลยีมาใช้ในการปฏิบัติงานทุกส่วนตลอดการปฏิบัติงานสหกิจศึกษา ซึ่งเป็นการนำความรู้ทั้งที่เคยเรียนมาประยุกต์ใช้ และเป็นการศึกษาเรียนรู้สิ่งใหม่ ๆ ที่ได้จากการปฏิบัติงาน

2.1 ทฤษฎีที่เกี่ยวข้องกับคลังข้อมูล

Android [6] เป็นระบบปฏิบัติการสำหรับอุปกรณ์พกพา เช่น โทรศัพท์มือถือ, แท็บเล็ต, คอมพิวเตอร์ เน็ตบุ๊กทำงานบนลินุกซ์เคอร์เนล เริ่มพัฒนาโดยบริษัท Android จากนั้นบริษัท Android ถูกซื้อโดยกูเกิลและนำ Android ไปพัฒนาต่อ

ภายหลังถูกพัฒนาในนามของ Open Handset Alliance ทางกูเกิลได้เปิดให้นักพัฒนาสามารถแก้ไขโค้ดต่างๆ ด้วยภาษาจาวาและควบคุมอุปกรณ์ผ่านทางชุด Java libraries ที่กูเกิลพัฒนาขึ้น โดย Android ถูกตั้งชื่อเลียนแบบหุ่นยนต์ในเรื่องสตาร์วอร์สที่ชื่อดรอยด์ ซึ่งเป็นหุ่นยนต์ที่สร้างขึ้นมาเลียนแบบมนุษย์เป็นซอฟต์แวร์ระบบปฏิบัติการที่มีโครงสร้างแบบเรียงทับซ้อนหรือแบบสแต็ก(Stack) โดยลินุกซ์เคอร์เนล เป็นพื้นฐานของระบบ และใช้ภาษา Java ในการพัฒนา มี AndroidSDK เป็นเครื่องมือสำหรับการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการ Android อีกทีหนึ่ง

ระบบปฏิบัติการ Android เริ่มพัฒนาเมื่อปีพ.ศ. 2550 โดยบริษัท Android ร่วมกับกูเกิลจากนั้นเมื่อปีพ.ศ. 2550 ได้มีการร่วมมือกันกว่า 30 บริษัทชั้นนำเพื่อพัฒนาระบบ Android ทั้งบริษัทผู้ให้บริการโทรศัพท์เคลื่อนที่ บริษัทผู้ผลิตอุปกรณ์โทรศัพท์เคลื่อนที่ บริษัทซอฟต์แวร์และบริษัทเอกชนต่างๆ เช่น HTC, LG, Motorola, Samsung, China Mobile Communications, KDDI, DoCoMo, Sprint/Nextel, T-Mobile, Telecom Italia, Audience, Broadcom, Intel, Marvel, NVidia, eBay, Packet Video, TAT และ Wind River เป็นต้น โดยใช้ชื่อกลุ่มว่า OHA (Open Handset Alliances) หรือ Open Source โดยมีลิขสิทธิ์ตาม Apache Versions 2 License ซึ่งลิขสิทธิ์ของ Apache จะอนุญาตให้ผู้พัฒนาสามารถนำโค้ดที่มีอยู่ไปพัฒนาต่อได้ทั้งในส่วนของการค้าหรือซอฟต์แวร์ลิขสิทธิ์และแบบใช้ฟรีหรือฟรีแวร์ก็ได้ จนถึงปัจจุบันมีการพัฒนา Android ออกมาหลายรุ่นตั้งแต่รุ่นที่ 1.0 จนกระทั่งถึงรุ่น 9.0 ดังตารางที่ 2.1

ตารางที่ 2.1 การพัฒนารุ่นของ Android ตั้งแต่รุ่นที่ 1.0 – 7.0

รุ่น	ชื่อเล่น	ระดับเอพีไอ
1.0	Alpha (แอลฟา)	1
1.1	Beta (เบต้า)	2
1.5	Cupcake (คัพเค้ก)	3
1.6	Donut (โดนัท)	4
2.0	Eclair (เอแคลร์)	5
2.0.1		6
2.1		7
2.2	Froyo (โฟรอสเซนโยเกิร์ต)	8
2.3	Gingerbread (ขนมปังจิง)	9
2.3.3		10
3.0	Honeycomb (รวงผึ้ง)	11
3.1		12
3.2		13
4.0	Ice Cream Sandwich (แซนด์วิชไอศกรีม)	14
4.0.3		15
4.1	Jelly Bean (เจลลี่빈)	16
4.2		17
4.3		18
4.4	KitKat (คิตแคต)	19
4.4w		20
5.0	Lollipop (อมยิ้ม)	21
5.1		22
6.0	Marshmallow (มาร์ชเมลโลว์)	23
7.0	Nougat (นูกัต)	24
7.1		25
8.0	Oreo (โอรีโอ)	26
8.1		27

ตารางที่ 2.1 การพัฒนารุ่นของ Android ตั้งแต่รุ่นที่ 1.0 – 7.0 (ต่อ)

รุ่น	ชื่อเล่น	ระดับเอพีไป
9.0	Pie (พาย)	28

เนื่องจาก Android เปิดให้นักพัฒนาเข้าไปชมรหัสต้นฉบับได้ทำให้มีผู้พัฒนาจากหลายฝ่ายนำเอารหัสต้นฉบับมาปรับแต่งและสร้าง Android ในแบบฉบับของตนเองขึ้นแบ่งประเภทของ Android ออกได้เป็น 3 ประเภท ดังต่อไปนี้

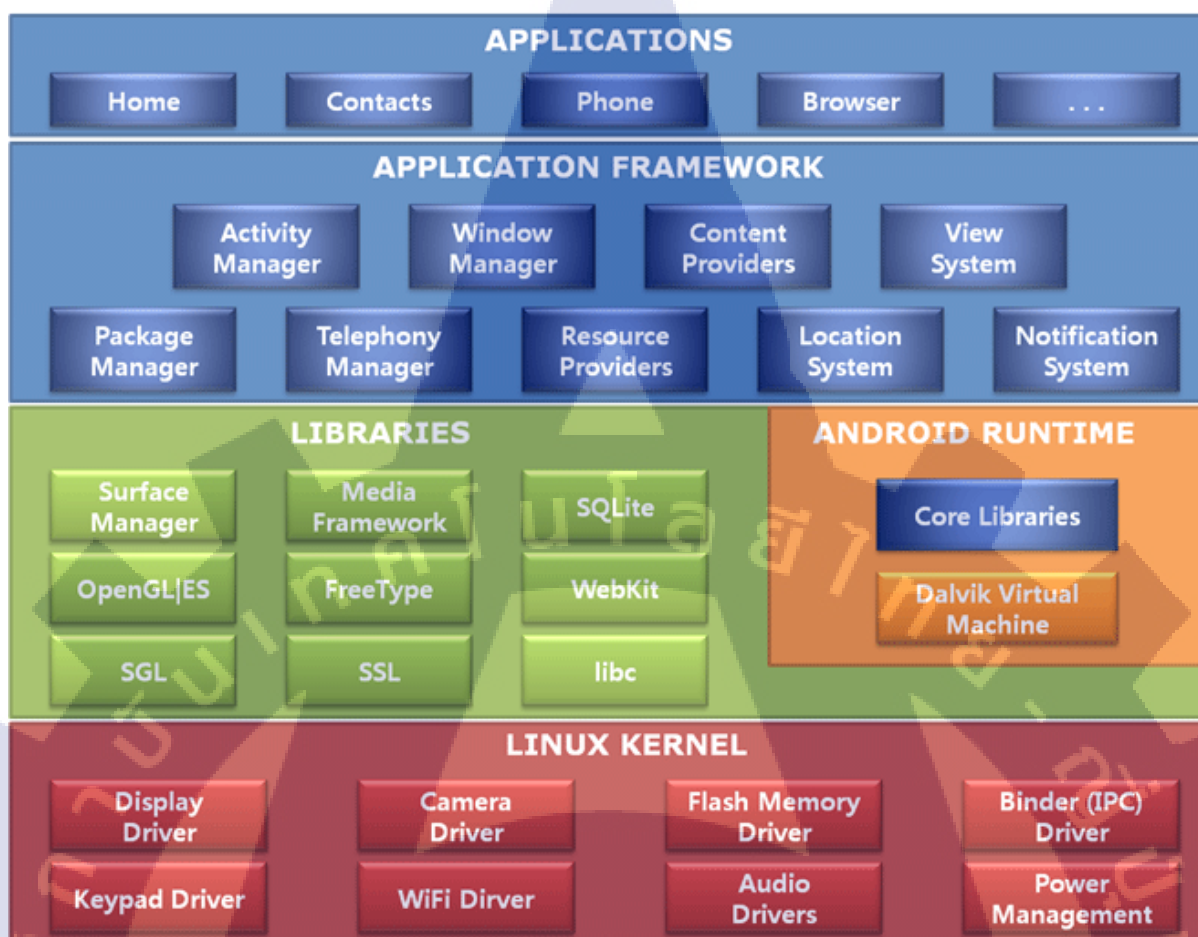
1) Android Open Source Project (AOSP) AOSP เป็น Android ประเภทแรกที่ถูกเปิดให้สามารถนำต้นฉบับแบบเปิดไปดัดแปลงและใช้งานในอุปกรณ์ต่าง ๆ ได้โดยไม่ต้องเสียค่าใช้จ่ายใด ๆ

2) Open Handset Mobile (OHM) เป็น Android ที่ได้รับการพัฒนาร่วมกับกลุ่มบริษัทผู้ผลิตอุปกรณ์พกพาที่เข้าร่วมกับถูกเปิดในนาม Open Handset Alliances (OHA) บริษัทเหล่านี้จะพัฒนา Android ในแบบฉบับของตนเองออกมา โดยรูปร่างหน้าตาการแสดงผล และฟังก์ชันการใช้งานจะมีความเป็นเอกลักษณ์และมีลิขสิทธิ์เป็นของตนเอง พร้อมได้รับสิทธิในการมีบริการเสริมต่าง ๆ จากถูกเปิดที่เรียกว่า Google Mobile Service (GMS) ซึ่งเป็นบริการเสริมที่ทำให้ Android มีประสิทธิภาพเป็นไปตามจุดประสงค์ของ Android แต่การจะได้มาซึ่ง GMS นั้นผู้ผลิตจะต้องทำการทดสอบระบบและขออนุญาตกับทางถูกเปิดก่อนจึงจะนำเครื่องออกสู่ตลาดได้

3) Cooking หรือ Customize เป็น Android ที่นักพัฒนานำเอารหัสต้นฉบับจากแหล่งต่าง ๆ มาปรับแต่งในแบบฉบับของตนเอง โดยจะต้องทำการปลดล็อกสิทธิ์การใช้งานอุปกรณ์หรือ Unlock เครื่องก่อนจึงจะสามารถดัดแปลงได้ Android ประเภทนี้ถือเป็นประเภทที่มีความสามารถมากที่สุดเท่าที่อุปกรณ์เครื่องนั้น ๆ จะรองรับได้ เนื่องจากได้รับการปรับแต่งให้เข้ากับอุปกรณ์นั้น ๆ จากผู้ใช้งานจริง

2.1.1 สถาปัตยกรรมของ Android

สถาปัตยกรรมของ Android จะถูกแบ่งออกเป็นลำดับชั้นเรียกว่า Layer โดยที่แต่ละชั้นจะเรียกใช้บริการจากระดับชั้นที่อยู่ด้านล่างของตัวเอง ซึ่งสามารถแบ่งออกเป็น 4 ชั้น ชั้นหลักคือ ชั้นลินุกซ์ (Linux Kernel), ชั้นไลบรารี (Libraries), ชั้นแอปพลิเคชันเฟรมเวิร์ก (Application Framework) และชั้นแอปพลิเคชัน (Application) ดังภาพที่ 2.1



ภาพที่ 2.1 ตัวอย่างรูปโครงสร้างสถาปัตยกรรมของ Android

2.1.1.1 ชั้นแอปพลิเคชัน (Application)

ชั้นนี้เป็นชั้นบนสุดของโครงสร้าง Android ซึ่งเป็นส่วนของแอปพลิเคชันที่พัฒนาขึ้นมาใช้งาน เช่น แอปพลิเคชันรับส่งอีเมล, แอปพลิเคชันโทรศัพท์, แอปพลิเคชันเว็บเบราว์เซอร์ เป็นต้น ทั้งนี้โปรแกรมในชั้นแอปพลิเคชันนั้นจะอยู่ในรูปแบบของไฟล์ .apk ซึ่งโดยทั่วไปแล้วจะอยู่ในไดเรกทอรี data/app ของโทรศัพท์

2.1.1.2 ชั้นแอปพลิเคชันเฟรมเวิร์ค (Application Framework)

โดยปกติแล้วนักพัฒนาสามารถเรียกใช้งาน Android ผ่าน API (Application Programming Interface) ได้ซึ่ง Android ได้ออกแบบไว้เพื่อลดความซับซ้อนในการใช้งานซ้ำของ Application Component โดยประกอบด้วยแอปพลิเคชันเฟรมเวิร์คดังนี้

- 1) View System เป็นส่วนที่ใช้ในการควบคุมการทำงานสำหรับการสร้างแอปพลิเคชัน เช่น lists, grids, textboxes, buttons และembeddable web browser
- 2) Location Manager เป็นส่วนที่จัดการเกี่ยวกับตำแหน่งของเครื่องอุปกรณ์พกพาเคลื่อนที่
- 3) Content Provider เป็นส่วนที่ใช้ควบคุมการเข้าถึงข้อมูลที่มีการใช้งานร่วมกัน (Share data) ระหว่างแอปพลิเคชันที่แตกต่างกัน เช่น ข้อมูลผู้ติดต่อ (Contact)
- 4) Resource Manager เป็นส่วนที่จัดการข้อมูลต่าง ๆ ที่ไม่ใช่ส่วนของโค้ดโปรแกรม เช่น รูปภาพ, localized strings, layout ซึ่งจะอยู่ในไดเรกทอรี res/
- 5) Notification Manager เป็นส่วนที่ควบคุมอีเวนต์ (Event) ต่าง ๆ ที่แสดงบนแถบสถานะ (Status bar) เช่น ในกรณีที่ได้รับข้อความหรือสายที่ไม่ได้รับและการแจ้งเตือนอื่น ๆ เป็นต้น
- 6) Activity Manager เป็นส่วนควบคุม Life Cycle ของแอปพลิเคชัน

2.1.1.3 ชั้นแอปพลิเคชัน (Application)

ได้รวบรวมกลุ่มของไลบรารีต่าง ๆ ที่สำคัญและมีความจำเป็นเอาไว้จำนวนมาก เพื่ออำนวยความสะดวกให้กับนักพัฒนาและง่ายต่อการพัฒนาโปรแกรม โดยตัวอย่างของไลบรารีที่สำคัญ เช่น

- 1) System C library เป็นกลุ่มของไลบรารีมาตรฐานที่อยู่บนพื้นฐานของภาษา C ไลบรารี(libc) สำหรับ embedded system ที่มีพื้นฐานมาจาก Linux
- 2) Media Libraries เป็นกลุ่มการทำงานมัลติมีเดีย เช่น MPEG4, H.264, MP3, AAC, AMR, JPG, และ PNG
- 3) Surface Manager เป็นกลุ่มการจัดการรูปแบบหน้าจอการวาดหน้าจอ
- 4) 2D/3D library เป็นกลุ่มของกราฟิกแบบ 2 มิติ หรือ SGL (Scalable Graphics Library) และแบบ 3 มิติ หรือ OpenGL
- 5) FreeType เป็นกลุ่มของบิตแมป (Bitmap) และเวกเตอร์ (Vector) สำหรับการเรนเดอร์ (Render) ภาพ
- 6) SQLite เป็นกลุ่มของฐานข้อมูล โดยนักพัฒนาสามารถใช้ฐานข้อมูลนี้เก็บข้อมูลแอปพลิเคชันต่าง ๆ ได้
- 7) Browser Engine เป็นกลุ่มของการแสดงผลบนเว็บเบราว์เซอร์โดยอยู่บนพื้นฐานของ Webkit ซึ่งจะมีลักษณะคล้ายกับ Google Chrome

2.1.1.3.1 Android Runtime

เป็นชั้นย่อยที่อยู่ในชั้นไลบรารีซึ่งจะประกอบด้วย 2 ส่วนหลัก คือ

1) Dalvik VM (Virtual Machine) ส่วนนี้ถูกเขียนด้วยภาษา Java เพื่อใช้เฉพาะการใช้งานในอุปกรณ์เคลื่อนที่ Dalvik VM จะแตกต่างจาก Java VM (Virtual Machine) คือ Dalvik VM จะรันไฟล์ .dex ที่คอมไพล์มาจากไฟล์ .class และ .jar โดยมี tool ที่ชื่อว่า dx ทำหน้าที่ในการบีบอัดคลาส Java ทั้งนี้ไฟล์ .dex จะมีขนาดกะทัดรัดและเหมาะสมกับอุปกรณ์เคลื่อนที่มากกว่า .class เพื่อต้องการใช้พลังงานจากแบตเตอรี่อย่างมีประสิทธิภาพสูงสุด

2) Core Java Library ส่วนนี้เป็นไลบรารีมาตรฐาน แต่มีความแตกต่างจากไลบรารีของ Java SE (Java Standard Edition) และ Java ME (Java Mobile Edition)

2.1.1.4 ชั้นแอปพลิเคชัน (Application)

ระบบ Android อยู่บนพื้นฐานของระบบปฏิบัติการลินุกซ์ โดยชั้นลินุกซ์เคอร์เนลนั้นมีฟังก์ชันการทำงานหลายๆ ส่วน ซึ่งแต่ละส่วนถูกพัฒนาขึ้นด้วยภาษา c เช่น การจัดการหน่วยความจำ (Memory Management), การจัดการโพรเซส (Process Management), การเชื่อมต่อเครือข่าย (Networking) และฟังก์ชันการทำงานส่วนอื่นที่เกี่ยวข้องกับระบบปฏิบัติการ ทั้งนี้นักพัฒนาจะไม่มีสิทธิ์เข้าถึงส่วนนี้ได้โดยตรง ซึ่งนักพัฒนาสามารถเข้าถึงระบบปฏิบัติการลินุกซ์ได้จากชุดคำสั่ง Command Prompt เช่น adb shell ซึ่งจะสามารถใช้เครื่องมือต่าง ๆ ได้ เช่น การเข้าสู่ระบบไฟล์ (File System), โพรเซสการคัดลอกไฟล์ (Copy File) เป็นต้น

2.2 เทคโนโลยีที่ใช้ในการปฏิบัติงาน

2.2.1 โปรแกรม Android Studio



ภาพที่ 2.2 ตัวอย่างโลโก้ Android Studio

2.2.2 โปรแกรม Xampp



ภาพที่ 2.3 ตัวอย่างโลโก้ Xampp

Xampp [8] คืออะไร เป็นโปรแกรม Apache web server ไว้จำลอง web server เพื่อไว้ทดสอบสคริปหรือเว็บไซต์ในเครื่องของเรา โดยที่ไม่ต้องเชื่อมต่ออินเทอร์เน็ตและไม่ต้องมีค่าใช้จ่ายใดๆ ง่ายต่อการติดตั้งและใช้งานโปรแกรม Xampp จะมาพร้อมกับ PHP ภาษาสำหรับพัฒนาเว็บแอปพลิเคชันที่เป็นที่นิยม , MySQL ฐานข้อมูล, Apache จะทำหน้าที่เป็นเว็บ เซิร์ฟเวอร์, Perl อีกทั้งยังมาพร้อมกับ OpenSSL , phpMyadmin (ระบบบริหารฐานข้อมูลที่พัฒนาโดย PHP เพื่อใช้เชื่อมต่อไปยังฐานข้อมูลสนับสนุนฐานข้อมูล MySQL และ SQLite

2.2.3 โปรแกรม MySQL



ภาพที่ 2.4 ตัวอย่างโลโก้ MySQL

MySQL [9] โปรแกรมระบบจัดการฐานข้อมูล ที่พัฒนาโดยบริษัท MySQL AB มีหน้าที่เก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่ง SQL เป็นเครื่องมือสำหรับเก็บข้อมูล ที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่นอย่างบูรณาการ เพื่อให้ได้ระบบงานที่รองรับ ความต้องการของผู้ใช้ เช่นทำงานร่วมกับเครื่องบริการเว็บ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษา php ภาษา asp.net หรือภาษาเจเอสพี เป็นต้น หรือทำงานร่วมกับโปรแกรมประยุกต์ (Application Program) เช่น ภาษาวิซวลเบสิกคอตเน็ต ภาษาจาวา หรือภาษาซีชาร์ป เป็นต้น โปรแกรมถูกออกแบบให้สามารถทำงานได้

บนระบบปฏิบัติการที่หลากหลาย และเป็นระบบฐานข้อมูลโอเพนซอร์ส (Open Source) ที่ถูกนำไปใช้งานมากที่สุด

2.2.4 โปรแกรม Visual Code



ภาพที่ 2.5 ตัวอย่างโลโก้ Visual Code

Visual Code [10] เป็นโปรแกรม Code Editor ที่ใช้ในการแก้ไขและปรับแต่งโค้ด จากค่ายไมโครซอฟท์ มีการพัฒนาออกมาในรูปแบบของ OpenSource จึงสามารถนำมาใช้งานได้แบบฟรี ๆ ที่ต้องการความเป็นมืออาชีพ ซึ่ง Visual Studio Code นั้น เหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานข้ามแพลตฟอร์ม รองรับการใช้งานทั้งบน Windows, macOS และ Linux สนับสนุนทั้งภาษา JavaScript, TypeScript และ Node.js สามารถเชื่อมต่อกับ Git ได้

2.2.5 โปรแกรม FileZilla



ภาพที่ 2.6 ตัวอย่างโลโก้ FileZilla

FileZilla [11] เป็นโปรแกรมที่ใช้ติดต่อกับ FTP server เพื่อดาวน์โหลดหรืออัปโหลดไฟล์ โดยเฉพาะกับเว็บไซต์ ทำให้เหมาะสำหรับนักออกแบบเว็บ นอกจากนี้ FileZilla ยังรองรับการถ่ายไฟล์อย่างปลอดภัยผ่าน SSH (SFTP) อีกด้วย FileZilla รองรับการกลับมาถ่ายไฟล์ต่อในกรณีที่อัปโหลดหรือดาวน์โหลดล้มเหลว และทำงานได้ดีผ่านไฟร์วอลล์และพร็อกซี FileZilla มีวิธีใช้งานที่สะดวก แบ่งวินโดว์ออกเป็นสองส่วนคือแสดงฝั่ง local กับ remote จากนั้นผู้ใช้งานสามารถ drag and drop ไฟล์ระหว่างสองฝั่งเพื่ออัปโหลดหรือดาวน์โหลดได้ทันที ผู้ใช้งานสามารถ login อย่างรวดเร็วโดยป้อนข้อมูลของเซิร์ฟเวอร์ที่อยู่ ชื่อผู้ใช้ และรหัสผ่าน ได้บนทูลบาร์เลยทันที นอกจากนี้ FileZilla ยังมี host manager ที่เก็บรายละเอียดของเซิร์ฟเวอร์ที่คุณใช้เป็นประจำ เพื่อให้คุณเชื่อมต่อได้อย่างง่ายและรวดเร็ว

2.2.6 โปรแกรม Git



ภาพที่ 2.7 ตัวอย่างโลโก้ Git

Git [12] คือ Version Control แบบ Distributed ตัวหนึ่ง เป็นระบบที่ใช้จัดเก็บและควบคุมการเปลี่ยนแปลงที่เกิดขึ้นกับไฟล์ชนิดใดก็ได้ ไม่ว่าจะเป็น Text File หรือ Binary File (จากนี้จะขอเรียก Text File หรือ Binary File รวมกันว่า Source Code) สาเหตุที่ควรใช้ Git มีดังนี้

- 1) Track version ของ Source Code ย้อนกลับได้เมื่อจัดเก็บไฟล์เข้าไปในระบบของ Git จะเรียกว่า Git Repository ซึ่งเก็บสำรองข้อมูลและการเปลี่ยนแปลงของ Source Code ทำให้สามารถย้อนกลับไปเวอร์ชันใด ๆ ก่อนหน้า และดูรายละเอียดการเปลี่ยนแปลงของแต่ละเวอร์ชันได้ นอกจากนี้ยังสามารถดูได้ว่าใครเป็นคนแก้ไข
- 2) ช่วยในการพัฒนาซอฟต์แวร์เป็นทีม Git สามารถเก็บบันทึกการเปลี่ยนแปลงของ Source Code เวอร์ชันล่าสุดไว้ที่ Local Repository ซึ่งสามารถทำงานได้โดยไม่ต้องต่อกับอินเทอร์เน็ต และเมื่อต้อง Update การเปลี่ยนแปลงของ Source Code เวอร์ชันล่าสุดให้กับเพื่อนร่วมทีมก็สามารถที่จะ Push ขึ้นไปเก็บ

ที่ Remote Repository(Git Hosting) และเพื่อนร่วมทีมก็สามารถ Pull เวอร์ชันล่าสุดนั้นมารวม(Auto Merge) ที่เครื่องของเขาเอง ทำให้ Source Code ที่พัฒนาร่วมกันกับคนภายในทีมเป็นเวอร์ชันล่าสุดเสมอ

2.3 ภาษาที่ใช้พัฒนา

2.3.1 ภาษา Java



ภาพที่ 2.8 ตัวอย่างโลโก้ Java

Java programming language [13] คือภาษาโปรแกรมเชิงวัตถุ พัฒนาโดย เจมส์ กอสลิง และวิศวกรคนอื่นๆ ที่บริษัท ซัน ไมโครซิสเต็มส์ ภาษานี้มีจุดประสงค์เพื่อใช้แทนภาษาซีพลัสพลัส C++ โดยรูปแบบที่เพิ่มเติมขึ้นคล้ายกับภาษาอ็อบเจกทีฟซี (Objective-C) แต่เดิมภาษานี้เรียกว่า ภาษาโอ๊ก (Oak) ซึ่งตั้งชื่อตามต้นไม้โอ๊กใกล้ที่ทำงานของ เจมส์ กอสลิง แล้วภายหลังจึงเปลี่ยนไปใช้ชื่อ “จาวา” ซึ่งเป็นชื่อกาแฟแทน จุดเด่นของภาษา Java อยู่ที่ผู้เขียนโปรแกรมสามารถใช้หลักการของ Object-Oriented Programming มาพัฒนาโปรแกรมของตนด้วย Java ได้

ภาษา Java เป็นภาษาสำหรับเขียนโปรแกรมที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (OOP : Object-Oriented Programming) โปรแกรมที่เขียนขึ้นถูกสร้างภายในคลาส ดังนั้นคลาสคือที่เก็บเมทอด (Method) หรือพฤติกรรม (Behavior) ซึ่งมีสถานะ (State) และรูปพรรณ (Identity) ประจำพฤติกรรม (Behavior)

2.3.2 ภาษา PHP



ภาพที่ 2.9 ตัวอย่างโลโก้ PHP

PHP [14] ย่อมาจาก PHP Hypertext Preprocessor แต่เดิมย่อมาจาก Personal Home Page Tools คือ ภาษาคอมพิวเตอร์จำพวก scripting language ภาษาจำพวกนี้คำสั่งต่าง ๆ จะเก็บอยู่ในไฟล์ที่เรียกว่า script และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างของภาษาสคริปก็เช่น JavaScript, Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมาเพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ PHP จึงเป็นภาษาที่เรียกว่า server-side หรือ HTML-embedded scripting language นั่นคือในทุก ๆ ครั้งก่อนที่เครื่องคอมพิวเตอร์ซึ่งให้บริการเป็น Web server จะส่งหน้าเว็บเพจที่เขียนด้วย PHP ทำการประมวลผลตามคำสั่งที่มีอยู่ แล้วจึงส่งผลลัพธ์ที่ได้ก็คือเว็บเพจ PHP เป็นเครื่องมือที่สำคัญชนิดหนึ่งที่ช่วยสร้าง Dynamic Web pages (เว็บเพจที่มีการโต้ตอบกับผู้ใช้) ได้อย่างมีประสิทธิภาพ

บทที่ 3

แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน

การปฏิบัติสหกิจศึกษาที่บริษัท ดิจิโอ (ประเทศไทย) จำกัด เป็นระยะเวลา 4 เดือน โดยแผนการปฏิบัติงานนั้นมีรายละเอียดการปฏิบัติงานทั้งหมด 4 งาน ดังนี้

1. การส่งข้อมูลจากแอปพลิเคชันแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติขึ้นเซิร์ฟเวอร์ (OCR)
2. การหาสาเหตุการเกิดข้อผิดพลาดของแอปการทำธุรกรรมการเงินผ่านแอปพลิเคชัน
3. การช่วยพัฒนาแอปเก็บข้อมูลบัตรประชาชนผ่านแอปพลิเคชัน
4. การพัฒนาแอปพลิเคชัน E-Passport

3.1 แผนงานการฝึกงาน

ตารางที่ 3.1 แผนงานการฝึกงาน

หัวข้องาน	เดือนที่			
	1	2	3	4
1. โปรเจกต์การแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติผ่านแอปพลิเคชัน				
2. โปรเจกต์การทำธุรกรรมการเงินผ่านแอปพลิเคชัน				
3. โปรเจกต์เก็บข้อมูลบัตรประชาชนผ่านแอปพลิเคชัน				
4. การพัฒนาแอปพลิเคชัน E-Passport				

3.2 รายละเอียดที่นักศึกษาปฏิบัติในการฝึกงาน

จากแผนฝึกงานในช่วงระยะเวลา 17 สัปดาห์ เริ่มแผนปฏิบัติงานตั้งแต่วันที่ 4 มิถุนายน 2561 ถึงวันที่ 28 กันยายน 2561 นั้นได้รับมอบมีทั้งหมด 4 งาน โดยข้อมูลต่อไปนี้จะเป็นการอธิบายการปฏิบัติงานในการทำงานโปรเจกต์ทั้ง 4 งาน โดยแต่ละโปรเจกต์จะแบ่งหัวข้อออกเป็น 2 ส่วน ดังนี้

1. รายละเอียดงาน
2. ขั้นตอนการดำเนินงาน

3.2.1 การส่งข้อมูลจากแอปพลิเคชันแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติขึ้นเซิร์ฟเวอร์ (OCR)

3.2.1.1 รายละเอียดงาน

วัตถุประสงค์ของโปรเจกต์การแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติผ่านแอปพลิเคชันคือต้องการให้พัฒนาแอปพลิเคชันขึ้นมา 1 แอป โดยคุณสมบัติของแอปนั้นคือสามารถอ่านค่าจากแถบข้อมูล MRZ (Machine Readable Zone) ซึ่งเป็นแถบชุดตัวอักษร 2 แถว แต่ละแถวบรรจุข้อมูลตัวอักษรไว้จำนวน 44 ตัว โดยตัวอักษรพิมพ์ด้วยฟอนต์ การแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติผ่านแอปพลิเคชัน ซึ่งแถบข้อมูล MRZ บรรจุข้อมูลต่าง ๆ ไว้ดังนี้

1. ชนิดของพาสปอร์ต
2. หมายเลขของพาสปอร์ต
3. ประเทศผู้ออกพาสปอร์ต
4. ชื่อเจ้าของพาสปอร์ต
5. สัญชาติเจ้าของพาสปอร์ต
6. วันเกิดเจ้าของพาสปอร์ต
7. เพศเจ้าของพาสปอร์ต
8. วันหมดอายุ
9. หมายเลขประจำตัวประชาชน

เมื่อได้ข้อมูลจากการอ่านแถบข้อมูล MRZ โดยใช้ การแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติผ่านแอปพลิเคชัน (Optical Character Recognition) ซึ่งเป็นกระบวนการของการแปลงสื่อสิ่งพิมพ์ เช่น กระดาษ นิตยสาร สัญญา หรือข้อมูลที่อยู่ในรูปของเอกสารกระดาษ ให้กลายเป็นข้อความ หรือให้

มีความฉลาดมากขึ้นกว่าการเป็นข้อความธรรมดาสามารถบันทึกไปเป็นไฟล์ประมวลผลค่าที่สามารถแก้ไขได้ง่าย ในที่นี้คือการอ่านแถบข้อมูล MRZ เพื่อนำข้อมูลที่ไปใช้งานต่อซึ่งในโปรเจกการแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติผ่านแอปพลิเคชันต้องการนำข้อมูลที่จัดเก็บไว้บนเซิร์ฟเวอร์

3.2.1.2 ขั้นตอนการดำเนินงาน

ในการดำเนินงาน OCR นั้น ข้าพเจ้าได้รับหน้าที่ให้ทำหน้าที่นำข้อมูลที่ได้จากแอปพลิเคชันไปจัดเก็บบนฐานข้อมูล ซึ่งก่อนจะลงมือปฏิบัติงานนั้นจึงจำเป็นต้องศึกษาหาข้อมูลให้เข้าใจการทำงานการเชื่อมต่อข้อมูลระหว่างแอปพลิเคชันและเซิร์ฟเวอร์ ซึ่งข้าพเจ้าได้ทราบหลักการทำงาน ดังนี้

1) การใช้ไลบรารี Retrofit (REST Client API) ที่เป็น API ตัวหนึ่งของโปรแกรม Android Studio ที่ใช้การเชื่อมต่อ Http สำหรับจัดการข้อมูล Json หรือ XML จุดเด่นของ Retrofit คือ แปลงข้อมูลเป็น POJO (Plain Old Java Object) สามารถใช้ได้ทั้ง GET หรือ POST จุดเด่นของ Retrofit อีกอย่างคือ มี OkHttp (ไลบรารีตัวหนึ่งที่ใช้ในการเชื่อมต่อ Http) และ Gson (ไลบรารีตัวหนึ่งใช้แปลงข้อมูลระหว่าง Object และ JSON) เป็น built-in อยู่ในนี้ด้วย



ภาพที่ 3.1 ตัวอย่างทำงานของ Retrofit2

2) การทำเซิร์ฟเวอร์จำลองโดยใช้โปรแกรม Xampp ที่มีการใช้ Apache เป็นเซิร์ฟเวอร์ MySQL เป็นตัวจัดการฐานข้อมูลซึ่งมี phpMyAdmin เป็น Interface ช่วยให้การสร้างหรือแก้ไขตารางฐานข้อมูลได้สะดวกขึ้น

3) การจัดทำฐานข้อมูลเพื่อเก็บข้อมูลที่ส่งมาจากแอปพลิเคชัน การติดต่อระหว่างเซิร์ฟเวอร์และฐานข้อมูลข้าพเจ้าเลือกใช้ภาษา PHP

4) การจัดทำแอปพลิเคชันให้เชื่อมต่อกับคลาวด์เซิร์ฟเวอร์ของทางบริษัท จำเป็นต้องสร้าง Public key โดยใช้โปรแกรม Putty เพื่อให้ทางบริษัทเพิ่มเครื่องคอมพิวเตอร์ของข้าพเจ้าในการใช้งานคลาวด์เซิร์ฟเวอร์

- 1) จากที่กล่าวไปข้างต้นมาถึงเรื่องของขั้นตอนการดำเนินงานในการปฏิบัติงานนั้นข้าพเจ้าได้พบทั้งปัญหาและเรียนรู้การแก้ไขปัญหาไปด้วย โดยมีที่พนักงานให้การช่วยเหลือในเรื่องที่ข้าพเจ้าไม่สามารถทำความเข้าใจได้ ซึ่งในเนื้อหาต่อไปนี้ข้าพเจ้าจะกล่าวถึงการดำเนินงานและปัญหาที่พบพร้อมวิธีการแก้ไข ดังนี้เพิ่มไลบรารี Retrofit และทำการเขียนองค์ประกอบต่าง ๆ ที่จำเป็นในการทำให้แอปพลิเคชันเชื่อมต่อกับเซิร์ฟเวอร์ได้ โดยหลักๆประกอบไปด้วย 3 ส่วนด้วยกัน

1. สร้าง Model (POJO) เก็บข้อมูลที่ Server ส่ง Json มา

```
public class InsertResponseModel {
    @SerializedName("success")
    private int status;
    @SerializedName("message")
    private String message;

    public InsertResponseModel(int status, String message) {...}

    public InsertResponseModel() {}

    public int getStatus() { return status; }

    public void setStatus(int status) { this.status = status; }

    public String getMessage() { return message; }

    public void setMessage(String message) { this.message = message; }
}
```

ภาพที่ 3.2 ตัวอย่างการสร้าง Model

2. สร้าง API Interface ในการระบุ url และ parameters ต่าง ๆ รายละเอียด

```

package com.test.epassportreadertesseract.SaveToDB;

import ...

public interface ApiInterface {
    @FormUrlEncoded
    @POST("passport/insertData.php")
    Call<InsertResponseModel> insertData (@Field("PassportType") String PassType,
                                         @Field("Surname") String Surname,
                                         @Field("Firstname") String Firstname,
                                         @Field("PassportNo") String ID,
                                         @Field("Nationality") String Nation,
                                         @Field("CitizenID") String CitizenNo,
                                         @Field("DOB") String DOB,
                                         @Field("Sex") String Sex,
                                         @Field("ExpDate") String ExpDate
    );
    @FormUrlEncoded
    @POST("passport/resultData.php")
    Call<List<PassportModel>> getData(@Field("CitizenID") String CitizenNo) ;
}

```

ภาพที่ 3.3 ตัวอย่างการสร้าง API Interface

3. สร้าง Class เชื่อมต่อเซิร์ฟเวอร์ และ Response จากเซิร์ฟเวอร์

```

public class ConnectServer {
    public static final String URL = "http://IP Address/";
    public static Retrofit RETROFIT = null;

    public static Retrofit getClient(){
        if (RETROFIT == null){
            OkHttpClient client = new OkHttpClient.Builder()
                .addInterceptor(new LoggingInterceptor())
                .build();

            Gson gson = new GsonBuilder()
                .setLenient().create();

            RETROFIT = new Retrofit.Builder()
                .baseUrl(URL)
                .client(client)
                .addConverterFactory(GsonConverterFactory.create(gson))
                .build();
        }
        return RETROFIT;
    }
}

```

ภาพที่ 3.4 ตัวอย่างการสร้าง Class เชื่อมต่อเซิร์ฟเวอร์

2) สร้างฐานข้อมูลโดยใช้โปรแกรม Xampp

id	PassportType	Surname	Firstname	PassportNo	Nationality	CitizenID	DOB	Sex	ExpDate
1	P	PRAWEEENA	PORNDEE	AA8854000	THA	1409901.....	96/10/17	F	22/08/15

ภาพที่ 3.5 ตัวอย่างการสร้างฐานข้อมูล

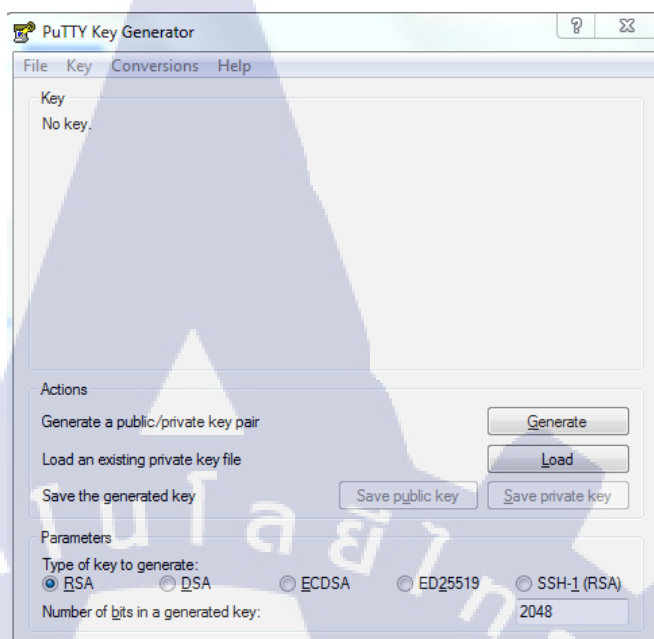
3) เขียน PHP เพื่อเชื่อมต่อเซิร์ฟเวอร์ และเขียนโค้ดการเพิ่มข้อมูลที่ได้รับมาจากแอปพลิเคชันไปยังฐานข้อมูล

```
<?php
$host      = "ip address";
$username  = "namehost";
$password  = "your password";
$dbname    = "name database";

$connect = mysqli_connect($host,$username,$password,$dbname);
if(!$connect){
    // echo "Connection Fails";
}else{
    // echo "Connected Success";
}
?>
```

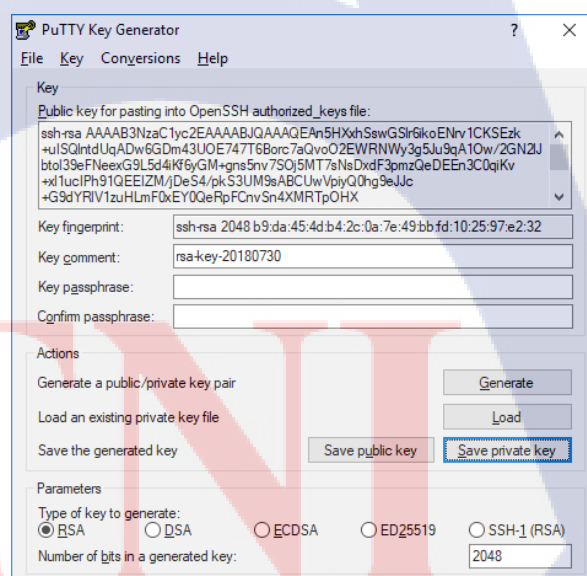
ภาพที่ 3.6 ตัวอย่าง PHP เชื่อมต่อเซิร์ฟเวอร์

- 4) จากการปฏิบัติตามข้อที่ผ่านมาได้ทำการทดสอบการใช้แอปพลิเคชันพบว่าแอปพลิเคชันสามารถเก็บข้อมูลไปยังเซิร์ฟเวอร์สำเร็จ
- 5) ดำเนินการทำเรื่องขอทดสอบการเชื่อมต่อแอปพลิเคชันบนเซิร์ฟเวอร์จริงของทางบริษัท โดยทางบริษัทได้ขอ Public key จากข้าพเจ้า จึงได้ทำการสร้าง Public key ไปไว้ในขั้นตอนนี้ข้าพเจ้าใช้เวลา 1 วัน ในการทำ Public key เนื่องจากเข้าใจวิธีการทำผิดไป ซึ่งวิธีที่ถูกต้องในการสร้าง Public key คือ
 1. เปิดโปรแกรม PuTTY Key Generator ขึ้นมา จากนั้นทำการเลือก Type of key to generate เป็น RSA แล้วกดปุ่ม Generate



ภาพที่ 3.7 ตัวอย่างโปรแกรม PuTTY Key Generator

2. จะได้ Public key เมื่อทำการ Generate เสร็จสิ้น แล้วทำการกดปุ่ม Save private key



ภาพที่ 3.8 ตัวอย่าง Public key

6) เมื่อทำการส่งไฟล์ Public key ไปให้ทางบริษัทแล้ว ทางบริษัทจะส่ง IP Address ที่ใช้เข้าไปยังคลาวด์เซิร์ฟเวอร์มาให้ทางอีเมล เมื่อได้รับอีเมลแล้วจึงเริ่มดำเนินการเข้าไปยังคลาวด์เซิร์ฟเวอร์ ในขั้นตอนนี้อัปเจ้าพบที่ต้องเขียนคำสั่งการทุกอย่างผ่าน Cmd ซึ่งการเข้าไปยังคลาวด์เซิร์ฟเวอร์มีคำสั่ง คือ `ssh root@IP Address -lroot` เมื่อเข้าไปยังคลาวด์เซิร์ฟเวอร์ได้แล้วนั้น อัปเจ้าพบที่ต้องทำการติดตั้งโปรแกรมทุกอย่างที่จำเป็นต้องใช้ในการสร้างฐานข้อมูลบนพื้นที่บริการคลาวด์เซิร์ฟเวอร์ที่ได้รับมา ซึ่งโปรแกรมที่ต้องติดตั้งผ่าน Command Line มีทั้งหมด 4 อย่าง ดังนี้

1. Apache
2. MySQL
3. PHP
4. phpMyAdmin

7) จากข้อ 6 อัปเจ้าพบปัญหาในการติดตั้งคืออัปเจ้าไม่ชำนาญในใช้คำสั่ง command Line ของ Linux เป็นเหตุให้อัปเจ้าต้องค้นหาคำสั่งการใช้งานของ Linux ในการสร้างไฟล์ สร้างโฟลเดอร์ หรือการลบไฟล์ เป็นต้น เมื่อทำการติดตั้งโปรแกรมทั้งหมดที่จำเป็นแล้วทำการสร้างไฟล์ Text ขึ้น เพื่อเขียนโค้ด PHP ในการเชื่อมต่อฐานข้อมูลและการเพิ่มข้อมูล ในขั้นตอนนี้อัปเจ้าทำได้ค่อนข้างลำบากเพราะต้องเขียนทุกอย่างบน command Line อัปเจ้าคิดว่าถ้ามีโปรแกรมไฟล์ FileZilla เข้ามาช่วยในการจัดจะช่วยประหยัดเวลาได้มากทีเดียว

8) หลังจากสร้างไฟล์ในการเชื่อมต่อคลาวด์เซิร์ฟเวอร์แล้ว จึงทดสอบแอปพลิเคชันพบว่าสามารถเพิ่มข้อมูลในฐานข้อมูลได้

9) แจ้งผลการปฏิบัติงานกับทางบริษัท

3.2.2 การหาสาเหตุปัญหาที่เกิดขึ้นจากการทำธุรกรรมการเงินผ่านแอปพลิเคชัน

แอปการทำธุรกรรมการเงินผ่านแอปพลิเคชันเป็นแอปพลิเคชันที่ใช้สำหรับร้านค้าที่ต้องการรับชำระเงินผ่านบัตรเครดิต บริษัทช่วยจัดการปัญหาด้านธุรกรรมการเงินจากการจ่ายเงินสดเปลี่ยนมาเป็นการจ่ายผ่านบัตรเครดิตแทนเพื่อให้บุคคลที่ไปเก็บเงินกับทางร้านค้าไม่ต้องจับเงินสด และทางร้านค้าที่เป็นลูกค้ากับบริษัทไม่จำเป็นต้องเอาเงินไปฝากธนาคารโดยตรง เพราะแอปพลิเคชันการธุรกรรมทางการเงินจะจัดการเรื่องนี้ให้ ซึ่งช่วยในเรื่องของลดต้นทุนในด้านต่าง ๆ โดยแอปพลิเคชันนี้ได้เกิดข้อผิดพลาดขึ้นคือ

1. การรู้บัตรเครดิตไม่ติด

2. เกิด Error 30000 (Invalid parameter)
3. เกิด Error 32000 (Transaction pending for start)

ทางธนาคารต้องการทราบสาเหตุที่ทำให้เกิดปัญหาดังกล่าว ซึ่งข้าพเจ้าและเพื่อนร่วมงานได้ปฏิบัติงานร่วมกับพนักงานที่รับผิดชอบโปรเจกต์นี้ ซึ่งหน้าที่ของข้าพเจ้าและเพื่อนร่วมงานคือคอยช่วยเหลืองานและปฏิบัติงานตามที่พนักงานมอบหมายให้คือหาสาเหตุของการเกิด Error 32000 โดยลักษณะการทำงานคือ คอยหาสาเหตุของปัญหาและทำการแก้ไขไปที่สาเหตุเพื่อให้เป็นไปตามเป้าหมายที่วางไว้

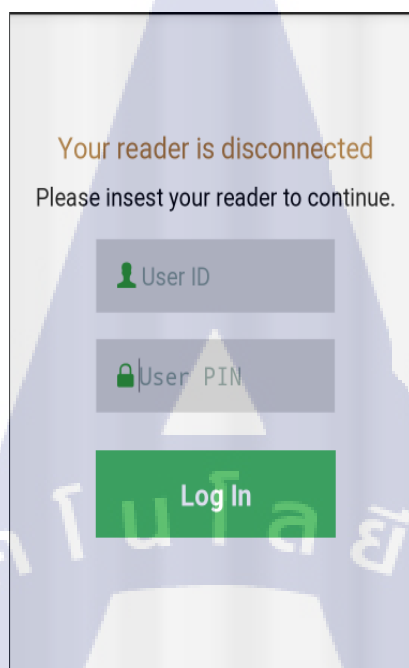
3.2.2.1 รายละเอียดงานและขั้นตอนการดำเนินงาน

นำแอปพลิเคชันมาโคลนลงเครื่องคอมพิวเตอร์แต่ไม่สามารถรันแอปพลิเคชันให้เปิดขึ้นมาได้ เนื่องจากบางคำสั่งของแอปพลิเคชันถูกยกเลิกการใช้งานไปแล้ว จึงต้องมีการปรับเปลี่ยนคำสั่งของแอปพลิเคชันให้เป็นคำสั่งที่มีใช้อยู่ในปัจจุบันแทน จากที่กล่าวมาทำให้ใช้เวลาในการปรับเปลี่ยนคำสั่งแอปพลิเคชันนานพอสมควร เมื่อทำการปรับเปลี่ยนและแก้ไขให้แอปพลิเคชันให้สามารถรันได้ปกติแล้ว จึงเริ่มการปฏิบัติงานดังนี้

1) ศึกษาแอปพลิเคชันการทำธุรกรรมทางการเงิน ในตัวโปรเจกต์นั้นมีแอปพลิเคชันอยู่จำนวน 2 แอปพลิเคชัน ThirdParty คือ แอปพลิเคชันที่ช่วยให้แอปพลิเคชันของทางบริษัทใช้งานร่วมกับแอปพลิเคชันของทางธนาคารได้, แอปพลิเคชันของทางบริษัท



ภาพที่ 3.9 แอปพลิเคชัน ThirdParty

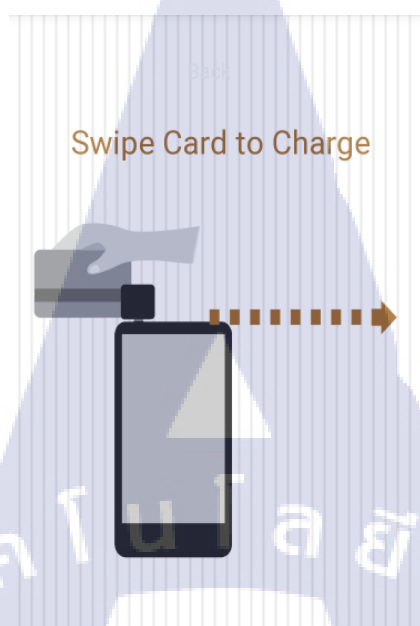


ภาพที่ 3.10 แอปพลิเคชันของทางบริษัทหน้า Login

2) จุดมุ่งหมายในการทำงานของทีมงานและข้าพเจ้าต้องการที่จะเข้าถึงหน้า Activity ที่อุปกรณ์อ่านบัตรเครดิต MSR90 สามารถอ่านบัตรได้ เพื่อทำการเขียน Log ให้สามารถดูค่าตัวแปรต่าง ๆ ได้ผ่าน Logcat



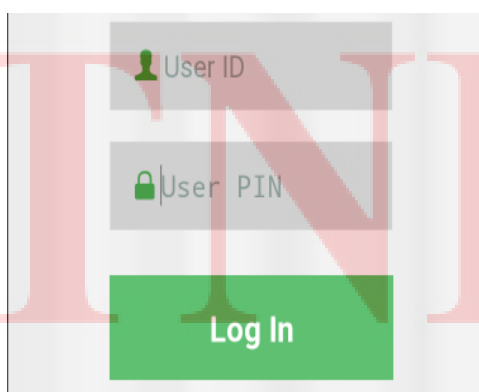
ภาพที่ 3.11 แอปพลิเคชันของทางบริษัทหน้าการต่ออุปกรณ์อ่านบัตร



ภาพที่ 3.12 รอคการรูดบัตรผ่านเครื่อง MSR90

3) การที่จะเข้าหน้า Activity ของอุปกรณ์อ่านบัตรเครดิต MSR90 ได้นั้น ทางทีมงานสันนิษฐานว่ามีอยู่ 2 ทางเลือก คือ การเข้าผ่านแอปพลิเคชัน Third Party และทางเลือกที่สองคือการเข้าผ่านแอปพลิเคชันการทำธุรกรรมการเงิน

4) หาวิธีเข้าหน้า Activity อุปกรณ์อ่านบัตรเครดิต MSR90 ผ่านทางแอปพลิเคชันการทำธุรกรรมการเงินก่อน แต่พบว่าแอปพลิเคชันการทำธุรกรรมการเงินนั้นต้องทำการ Login เพื่อเข้าใช้งานแอปพลิเคชัน ซึ่งทางทีมงานและข้าพเจ้าไม่มี Username และ Password



ภาพที่ 3.13 หน้า Login ของแอปพลิเคชันการทำธุรกรรมทางการเงิน

5) จากข้อที่ 4 ทำให้ต้องใช้วิธี Bypass เข้าไปแอปพลิเคชันแทนการใส่ Username และ Password

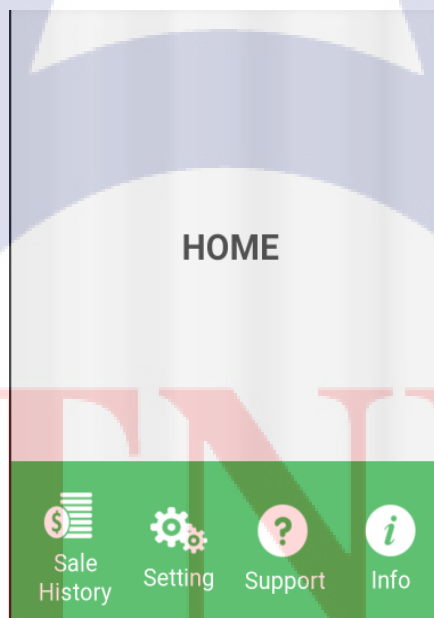
```
public final static Boolean BYPASS_LOGIN = false;
public final static Boolean BYPASS_LOGIN = true;
```

ภาพที่ 3.14 การทำ Bypass โดยเปลี่ยนจากคำสั่ง false เป็น true

```
if ( Config.BYPASS_LOGIN ) {
    loginSuccessful();
}
```

ภาพที่ 3.15 เมื่อทำ Bypass จะเข้าเงื่อนไขการ Login สำเร็จ

6) แต่เมื่อทำการ Bypass เข้าไปได้นั้น แอปพลิเคชันการทำธุรกรรมทางการเงินได้เข้าสู่หน้าหลักหรือหน้าโฮม ซึ่งไม่สามารถเข้าไปยังหน้าอุปกรณ์อ่านบัตรเครดิต MSR90 อย่างที่ต้องการ ทำให้ข้อสมมติฐานนี้ผิด



ภาพที่ 3.16 หน้าโฮมของแอปพลิเคชันการทำธุรกรรมทางการเงิน

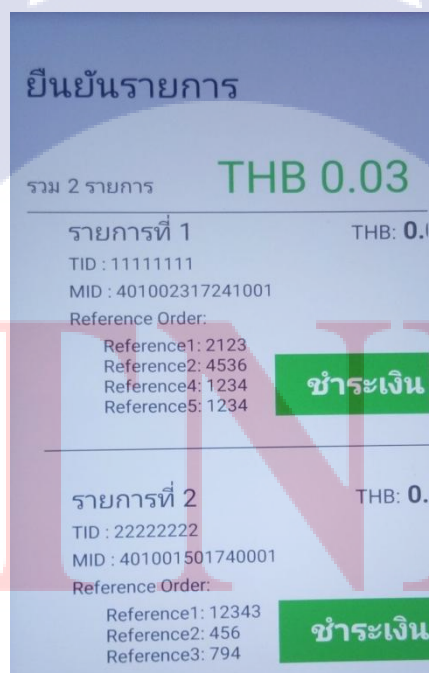
```
private void loginSuccessful() {
    setState("loginSuccessful");

    Intent intent = new Intent();
    intent.setClass(packageContext.this, HomeMainActivity.class);
    startActivity(intent);
}
```

ภาพที่ 3.17 ตัวอย่างฟังก์ชันการ Login สำเร็จ

7) ทดลองเข้าแอปผ่าน Third Party แต่จะเข้าแอปได้ต้องมี Username กับ Password เช่นกัน ที่พนักงานที่รับหน้าที่รับผิดชอบโปรเจกต์นี้จึงได้ทำการสร้าง Service จำลองขึ้นมา เพื่อที่จะใช้งานแอปพลิเคชันการทำธุรกรรมทางการเงินได้

8) จากข้อที่แล้ว เมื่อจำลอง service และเข้าแอปพลิเคชันการทำธุรกรรมทางการเงินผ่านแอปพลิเคชัน Third Party ได้แล้ว จึงพบหน้าการทำงานที่ปรากฏขึ้นมาคือหน้าแสดงรายการสินค้า จากนั้นทดลองกดปุ่มชำระเงินพบภายในหน้าแสดงรายการสินค้าพบว่าสามารถเข้าต่อไปยัง Activity ที่เป็น อุปกรณ์อ่านบัตรเครดิต MSR90 ตามที่ต้องการได้แล้ว



ภาพที่ 3.18 ตัวอย่างหน้าแสดงรายการ

จากภาพตัวอย่างหน้าแสดงรายการสินค้าที่พนักงานต้องการดู Log ว่าหน้าดังกล่าวได้รับตัวแปรอะไรมาบ้าง ข้าพเจ้าและเพื่อนร่วมงานจึงทำการเขียนโค้ดในการดูค่าตัวแปรผ่าน Logcat

9) จากข้อที่แล้ว เมื่อข้าพเจ้าและเพื่อนร่วมงานต้องการดูข้อมูลที่ได้รับจากการรูดบัตรเครดิตผ่านเครื่อง MSR90 ก็พบปัญหาว่าการดู Log นั้นจะสามารถดูได้ผ่านการรันแอปพลิเคชันธุรกรรมทางการเงินที่เชื่อมต่อสายชาร์จของสมาร์ทโฟนที่ด้านหนึ่งเป็นหัวแบบ USB เพื่อเชื่อมต่อกับคอมพิวเตอร์และหากดึงสายออกจะทำให้ Log หยุดการแสดงผล ซึ่งการที่จะรูดบัตรเครดิตผ่านเครื่อง MSR90 นั้น ตัวเครื่องจะต้องเชื่อมต่อกับสมาร์ทโฟนโดยสายของตัวเครื่อง MSR90 เป็นดังภาพที่ 3.20 เป็นเหตุให้ไม่สามารถดู Log การทำงานของแอปพลิเคชันการทำธุรกรรมทางการเงินพร้อมกับการรูดบัตรเครดิตผ่านเครื่อง MSR90 ได้



ภาพที่ 3.19 เครื่อง MSR90



ภาพที่ 3.20 หัวสายชาร์จสมาร์ทโฟน

10) ปัญหาจากข้อที่แล้ว ทำให้ต้องหาวิธีการดู Log การทำงานของแอปพลิเคชันการทำธุรกรรมทางการเงินแบบไร้สายเชื่อมต่อ โดยข้าพเจ้าและเพื่อนร่วมงานมีความเห็นว่าจะใช้วิธีการเชื่อมต่อผ่านเครือข่าย ซึ่งเครือข่ายที่จะนำมาทดสอบมี 2 เครือข่ายคือ เครือข่ายบลูทูธ และเครือข่าย wireless

11) จากข้อที่ 10 พบว่าการเชื่อมต่อผ่าน wireless นั้นทำได้ง่ายกว่า โดยหากต้องการดู Log ผ่านเครือข่าย wifi ทำได้โดยพิมพ์คำสั่งบน Command Line ดังนี้

1. adb tcpip 5555 คือใช้ tcp 5555 ในการเชื่อมต่อ Android studio
2. adb connect ip address:5555 เชื่อมต่อ ip address ที่ระบุ (ไอพีของสมาร์ทโฟนซึ่งอยู่ในเครือข่าย wifi เดียวกันกับคอมพิวเตอร์)

12) เมื่อสามารถดู Log ผ่านการรันแอปพลิเคชันโดยใช้ wifi ได้แล้ว จึงทำการเขียนโค้ดเพื่อดู Log เพิ่มเติมเพื่อดูข้อมูลที่ได้รับจากโค้ดส่วนของฟังก์ชันการตรวจสอบเงื่อนไขบัตรเครดิตดังนี้

1. ตัวเลขในบัตรเครดิต 4 ตัวแรก เป็นไปตามที่กำหนดหรือไม่
2. ตัวเลขในบัตรเครดิตมีจำนวน 16 ตัวจริงหรือไม่

3. เช็คว่าบัตรที่ได้มาเป็นบัตรเครดิตจริงหรือไม่

การวาง Log ตรงส่วนนี้เพื่อช่วยให้นำข้อมูลที่ได้ไปวิเคราะห์ปัญหาของการรูดบัตรเครดิตด้วยเครื่อง MSR90 ไม่ติด

13) จากแผนที่วางไว้ว่าคือเข้าไปยังหน้า อุปกรณ์อ่านบัตรเครดิต MSR90 เพื่อดูปัญหาที่เกิดขึ้น และเขียน Log เพื่อวิเคราะห์ข้อมูลจากการค่าที่แสดงบน Logcat ในขณะที่ทำการรูดบัตรเครดิต ซึ่งจากที่กล่าวมาสามารถทำได้ตามแผนที่วางไว้เรียบร้อยแล้วทางการเงินแบบไร้สายเชื่อมต่อ โดยข้าพเจ้าและเพื่อนร่วมงานมีความเห็นว่าจะใช้วิธีการเชื่อมต่อผ่านเครือข่าย ซึ่งเครือข่ายที่จะนำมาทดสอบมี 2 เครือข่ายคือ เครือข่าย บูลดอต และเครือข่าย wireless

14) จากการทำการทดลองรูดบัตรเครดิตในหลายๆ กรณีทำให้ทีมงานพบว่าของการเกิด Error 32000 และการรูดบัตรเครดิตไม่ติดนั้นคือในขณะที่ทำการรูดบัตรนั้นแอปพลิเคชันเก็บเลขบัตรเครดิตไม่ครบ 16 ตัว นอกจากนี้สมาร์ทโฟนที่ใช้คีย์บอร์ดอัจฉริยะที่สามารถตรวจจับตัวอักษรและแปรงเป็นคำได้ จะทำให้การอ่านบัตรผิดพลาด เนื่องจากคีย์บอร์ดอัจฉริยะจะตรวจจับคำที่มีความหมาย แต่ค่าที่ได้จากเครื่อง MSR90 เป็นค่าที่ไม่เป็นคำและไม่มีความหมาย ทำให้คีย์บอร์ดอัจฉริยะตรวจจับไม่ได้ส่งผลให้ไม่มีการรีเทิร์นค่ากลับมายังแอปการทำธุรกรรมทางการเงิน

จากที่กล่าวมาสามารถทำการแก้ปัญหาเบื้องต้นได้แล้ว และส่งรายงานปัญหาที่พบแก่ทางธนาคารเป็นที่เรียบร้อย ปัจจุบันโปรเจกต์ยังคงได้รับการพัฒนาต่อไปไปตามความต้องการของทางธนาคาร โดยรุ่นที่ที่รับผิดชอบงานโปรเจกต์ดำเนินการต่อไป

3.2.3 การช่วยพัฒนาแอปเก็บข้อมูลบัตรประชาชนผ่านแอปพลิเคชัน

3.2.3.1 รายละเอียดงาน

แรกเริ่มนั้นแอปพลิเคชันเก็บข้อมูลบัตรประชาชนผ่านแอปพลิเคชันนี้ใช้งานร่วมกับอุปกรณ์ฮาร์ดแวร์ที่เป็นเครื่องสำหรับอ่านบัตรประชาชนโดยเฉพาะดังภาพด้านล่าง โดยเชื่อมต่อแอปพลิเคชันอ่านบัตรประชาชนผ่านเครือข่ายบูลดอต แต่ทางบริษัทต้องการพัฒนาให้แอปพลิเคชันนี้ใช้งานได้กับตัวผลิตภัณฑ์ของบริษัทซึ่งเป็นอุปกรณ์ EDC (Electronic Data Capture) คืออุปกรณ์รับชำระเงินทางอิเล็กทรอนิกส์ หรือที่เรียกว่าเครื่องรับบัตรหรือเครื่องรูดบัตรสามารถอ่านบัตรประชาชนได้ ดังนั้นงานนี้คือการเปลี่ยนแอปพลิเคชันให้สามารถรองรับการทำงานร่วมกับตัวผลิตภัณฑ์ของบริษัทได้ โดยใช้คุณสมบัติการอ่านบัตรของตัวผลิตภัณฑ์ของบริษัทแทนเครื่องอ่านบัตรประชาชนโดยใช้บูลดอต อีกทั้งยังต้องการให้แอปพลิเคชันสามารถ

ส่งต่อข้อมูลที่อ่านได้จากบัตรประชาชนไปยังแอปพลิเคชันอื่น จึงจำเป็นต้องสร้างแอปพลิเคชันตัวอย่างที่ใช้ในการรับข้อมูลจากแอปพลิเคชันเก็บข้อมูลบัตรประชาชน ซึ่งข้าพเจ้าได้ทำงานนี้ร่วมกับเพื่อนร่วมงาน โดยหน้าที่ของข้าพเจ้าคือการสร้างแอปพลิเคชันตัวอย่างในการรับข้อมูลจากแอปเก็บข้อมูลบัตร



ภาพที่ 3.21 อุปกรณ์อ่านบัตรประชาชน



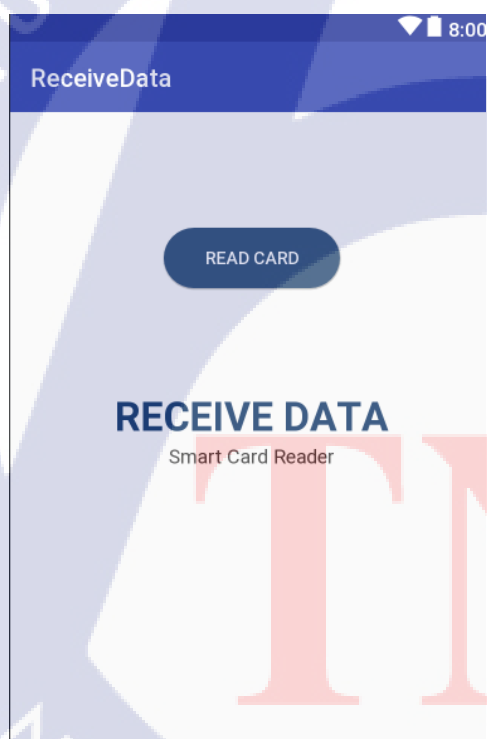
ภาพที่ 3.22 เครื่อง EDC

3.2.3.2 ขั้นตอนการดำเนินงาน

ในขั้นตอนการดำเนินงานนั้นพนักงานที่รับผิดชอบโปรเจกต์นี้เป็นผู้ชี้แจงรายละเอียดในการเลือก Branch ที่ต้องจำเป็นต้อง Clone โปรแกรมมาปฏิบัติงาน ซึ่งสิ่งที่ข้าพเจ้าต้อง Clone นั้นประกอบไปด้วย 2 Branch คือ Branch ที่มีโค้ดตัวอย่างในการอ่านบัตรประชาชนบนผลิตภัณฑ์ของบริษัท และ Branch ตัวโปร

เจตที่ทางบริษัทต้องการปรับเปลี่ยนฟังก์ชันการทำงาน เมื่อ Clone ลงบนคอมพิวเตอร์สำเร็จจึงลงมือปฏิบัติงานที่ได้รับมอบหมาย ดังนี้

1. ศึกษาโค้ดของแอปพลิเคชันการเก็บข้อมูลบัตรประชาชน โดยเฉพาะส่วนที่เป็นการทำงานของบลูทูธเพื่อที่จะลบส่วนการทำงานของบลูทูธ เนื่องจากผลิตภัณฑ์ของบริษัทอ่านบัตรประชาชนได้โดยไม่ต้องใช้งานเครื่องอ่านบลูทูธ
2. ลบโค้ดที่เป็นการทำงานต่างๆ ของบลูทูธออกจากแอปพลิเคชันการเก็บข้อมูลบัตรประชาชน
3. ทำการเขียนโค้ดให้แอปพลิเคชันการเก็บข้อมูลบัตรประชาชนสามารถส่งข้อมูลไปยังแอปพลิเคชันตัวอย่าง
4. สร้างแอปพลิเคชันตัวอย่างทดลองมาหนึ่งแอปเพื่อเช็คว่าแอปพลิเคชันการเก็บข้อมูลบัตรประชาชนสามารถส่งข้อมูลได้จริงหรือไม่ และแอปตัวอย่างสามารถรับข้อมูลจากแอปพลิเคชันการเก็บข้อมูลบัตรประชาชนได้นั้นจะต้องมีการเขียนโค้ดส่วนใดบ้าง เพื่อรองรับการส่งข้อมูลมาจากแอปพลิเคชันการเก็บข้อมูลบัตรประชาชน



ภาพที่ 3.23 หน้าแรกของแอปพลิเคชันตัวอย่าง



ภาพที่ 3.24 หน้าแสดงข้อมูลแอปพลิเคชันตัวอย่าง

ตลอดระยะเวลาในการปรับปรุงแอปพลิเคชัน ข้าพเจ้า เพื่อนร่วมงานและพนักงานที่รับผิดชอบโปรเจกต์นี้ได้ปรึกษาและแก้ไขงานร่วมกันจนโปรเจกต์สำเร็จตามเป้าหมายที่วางไว้

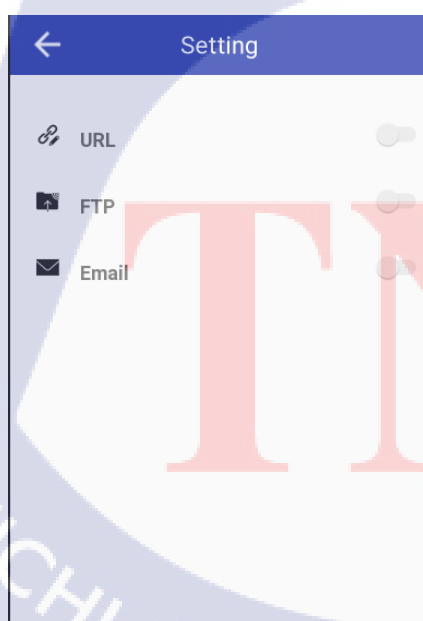
3.2.4 การส่งข้อมูลจากแอปพลิเคชันแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติขึ้นเซิร์ฟเวอร์ (OCR)

3.2.4.1 รายละเอียดงาน

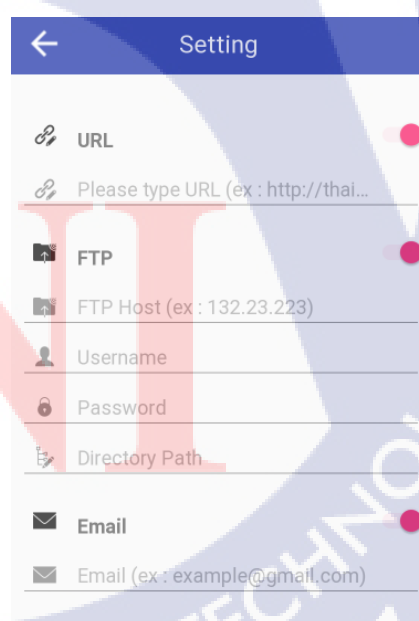
แอปพลิเคชัน E-Passport นั้นคือแอปพลิเคชันที่ใช้ OCR ในการอ่านข้อมูลบน Passport ซึ่งนำพัฒนาให้แอปพลิเคชัน E-Passport สามารถส่งข้อมูลไปยังที่อื่นได้ โดยส่งต่อได้ผ่าน Email และในอนาคตจะพัฒนาให้สามารถส่งผ่าน File Transfer Protocol (FTP), Uniform Resource Locator (URL) โดยหน้าที่ของข้าพเจ้าคือการทำหน้า Setting layout และ พัฒนาให้แอปสามารถส่งข้อมูลผ่านระบบอีเมลได้

3.2.4.2 ขั้นตอนการดำเนินงาน

ศึกษาการส่งข้อมูลด้วยอีเมลเพื่อให้เข้าใจหลักการทำงาน จากนั้นทำการออกแบบหน้าการตั้งค่าของแอปพลิเคชัน ในส่วนการส่งข้อมูลจะถูกจัดไว้ในโหมดของการตั้งค่า เมื่อออกแบบเสร็จแล้วจึงเริ่มทำการเขียนโค้ดการทำงานให้แอปพลิเคชันสามารถส่งข้อมูลผ่านอีเมล โดยเริ่มปฏิบัติงานดังนี้

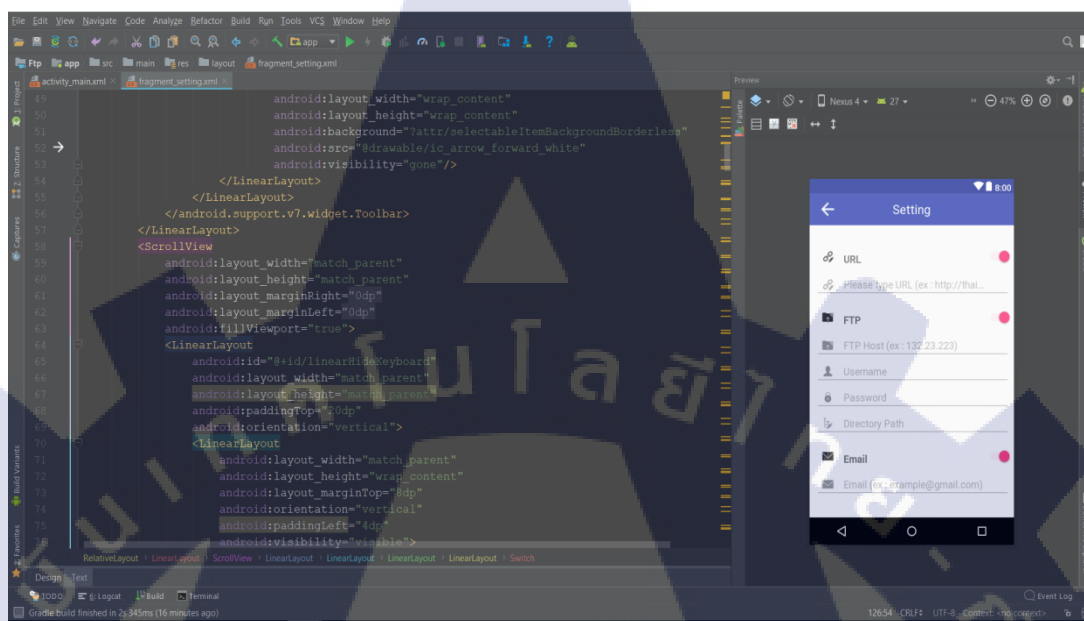


ภาพที่ 3.25 หน้าการตั้งค่าแอปพลิเคชัน



ภาพที่ 3.26 เมื่อเลือกการตั้งค่าต่าง ๆ

- 1) ออกแบบการทำหน้าการตั้งค่า และจัดการเขียนไฟล์.xml ตามการรูปแบบที่ทำการออกแบบไว้



ภาพที่ 3.27 ตัวอย่างการเขียนไฟล์.xml

- 2) ดำเนินการเขียนโค้ดในส่วนของการซ่อน Layout ขั้นตอนนี้คือการทำให้ซ่อนช่องกรอกข้อมูลหากฟังก์ชันที่ไม่ได้ถูกเลือก เช่น ผู้ใช้งานกดเลือกตั้งค่า URL แอปพลิเคชันจะแสดงช่องให้ระบุ URL ที่ต้องการ แต่หากไม่ได้กดเลือกการตั้งค่า URL ช่องสำหรับกรอกข้อมูลจะถูกซ่อนไว้นั่นเอง



ภาพที่ 3.28 ตัวอย่างผู้ใช้งานไม่ได้กดเลือก URL



ภาพที่ 3.29 ตัวอย่างผู้ใช้งานกดเลือก URL

- 3) ดำเนินปรับปรุงโค้ดในส่วนของการตั้งค่าและส่งข้อมูลผ่านอีเมลเพื่อให้แอปพลิเคชันสามารถส่งข้อมูลผ่านอีเมลได้

```

//////////function Email
private void sendToEmail(String filename){
    File filelocation = new File(getAlbumStorageDir( albumName: "CSV/"), filename);
    Uri path = Uri.fromFile(filelocation);
    Intent emailIntent = new Intent(Intent.ACTION_SEND);
    // set the type to 'email'
    // emailIntent.setType("vnd.android.cursor.dir/email");
    emailIntent.setType("message/rfc822");
    String to = emailPreference.getEmail();//"fsmoon.ck@gmail.com";//dataSetting.getDataEmail();
    emailIntent.putExtra(Intent.EXTRA_EMAIL, new String[]{to} );
    //// the attachment
    emailIntent.putExtra(Intent.EXTRA_STREAM, path);
    // the mail subject
    emailIntent.putExtra(Intent.EXTRA_SUBJECT, value: "Passport Data From Application");
    startActivityForResult(Intent.createChooser(emailIntent, title: "Send email..."), requestCode: 33);
}

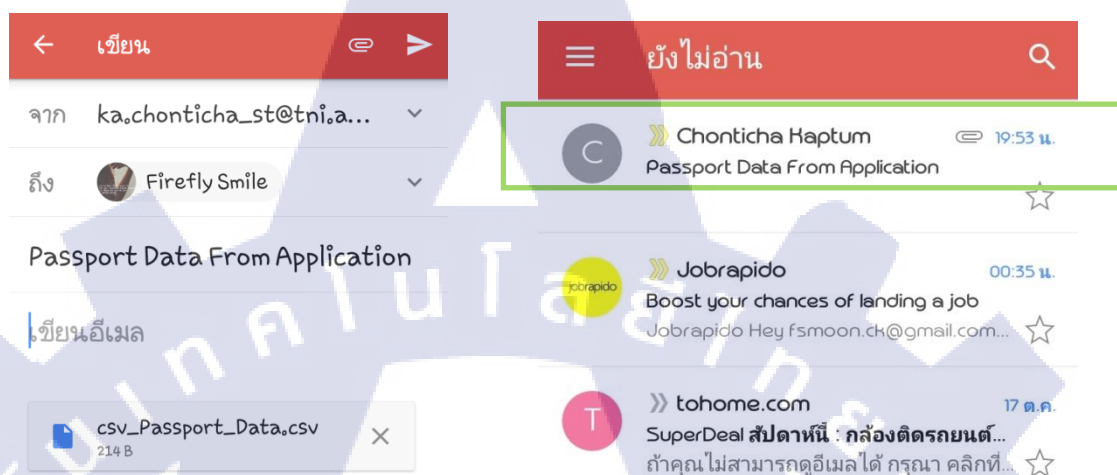
```

ภาพที่ 3.30 ตัวอย่างโค้ดการส่งข้อมูลไปยังอีเมล.xml

4) ตัวอย่างการทำงานของแอปพลิเคชัน E- Passport เมื่อเปิดแอปพลิเคชันขึ้นมาจะสามารถกรอกข้อมูล Passport ของผู้ใช้งานได้เลย หรือ ถ่ายรูปจาก Passport ก็ได้ เมื่อได้ข้อมูลครบแล้ว ทำการเลือกส่งข้อมูลผ่านอีเมล(พัฒนาในส่วนของข้าพเจ้าคือการทำฟังก์ชันส่งข้อมูลผ่านอีเมล) แล้วกดปุ่ม Send ดังภาพที่ 3.30

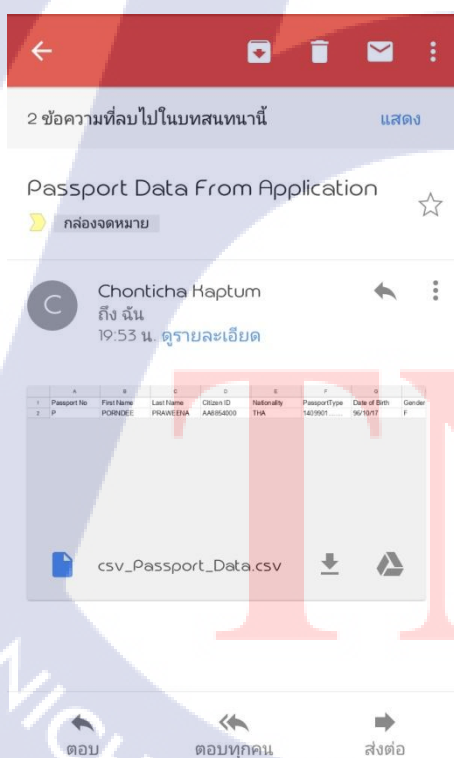
ภาพที่ 3.31 ตัวอย่างการกรอกข้อมูล

5) จากภาพที่ 3.31 ข้อมูลที่ถูกส่งผ่านระบบอีเมลนั้น ถูกเก็บเป็นไฟล์ .CSV และภาพที่ 3.32 คือได้รับอีเมลจากแอปพลิเคชัน E-Passport



ภาพที่ 3.32 ตัวอย่างการส่งอีเมล

ภาพที่ 3.33 ตัวอย่างอีเมลที่ได้รับ



ภาพที่ 3.34 ตัวอย่างเนื้อหาอีเมลที่ได้รับ

5) จากภาพที่ 3.31 ข้อมูลที่ถูกส่งผ่านระบบอีเมลนั้น ถูกเก็บเป็นไฟล์ .CSV และภาพที่ 3.32 คือได้รับอีเมลจากแอปพลิเคชัน E-Passport



The screenshot shows a mobile phone interface with a status bar at the top displaying '4G', signal strength, time '19:54 น.', and battery level '4G 82%'. Below the status bar, the title bar of an application window reads 'csv_Passport_Data'. The main content area shows a table with the same title. The table has 10 columns labeled A through I, corresponding to the headers in the first row. The data rows show passport details for a person named PORNDÉE PRAWEENA.

	A	B	C	D	E	F	G	H	I
1	Passport No	First Name	Last Name	Citizen ID	Nationality	Passport Type	Date of Birth	Gender	Date of Expire
2	P	PORNDÉE	PRAWEENA	AA8854000	THA	1409901.....	10/17/1996	F	8/22/2015
3									

ภาพที่ 3.35 ตัวอย่างไฟล์ .CSV

TNI

บทที่ 4

สรุปผลการดำเนินงาน การวิเคราะห์และสรุปผลต่าง ๆ

4.1 ผลการดำเนินงาน

จากบทที่ 3 ในการปฏิบัติงานได้อธิบายรายละเอียดและขั้นตอนการปฏิบัติงานตลอด 17 สัปดาห์ สามารถสรุปผลการดำเนินงาน โดยแบ่งออกเป็น 4 งาน ดังนี้

4.1.1 การส่งข้อมูลจากแอปพลิเคชันแปลงไฟล์ภาพเอกสารให้เป็นไฟล์ข้อความโดยอัตโนมัติขึ้นเซิร์ฟเวอร์ (OCR)

การทำโปรเจกต์ถือว่าประสบความสำเร็จ เนื่องจากสามารถทำการส่งข้อมูลจากแอปพลิเคชันแปลงไฟล์ภาพให้เป็นไฟล์ข้อความโดยอัตโนมัติขึ้นเซิร์ฟเวอร์ได้จริง โดยในอนาคตข้างหน้าตัวแอปพลิเคชันนี้จะถูกนำไปใช้บนเครื่อง EDC ของบริษัท

4.1.2 การหาสาเหตุปัญหาที่เกิดขึ้นจากการทำธุรกรรมการเงินผ่านแอปพลิเคชัน

โปรเจกต์นี้เป็นโปรเจกต์ที่ถูกพัฒนาในระยะยาว และจากการร่วมทำงานโปรเจกต์นี้เป็นระยะเวลาหนึ่งได้ปฏิบัติงานให้บรรลุตามเป้าหมายเบื้องต้น โดยสาเหตุของการเกิด Error 32000 มีดังนี้

- 1) กรณีรู้บัตรและอ่านบัตรไม่ได้
- 2) ข้อมูลบัตรไม่ครบ 16 ตัวอักษร
- 3) กรณีบัตรไม่ขึ้นต้นด้วย 0004
- 4) กรณีเลขบัตรไม่ได้มาตรฐาน

ซึ่งได้จัดทำรายงานสาเหตุของการเกิดปัญหา Error 32000 กับทางธนาคารแล้ว โปรเจกต์นี้ปัจจุบันยังคงดำเนินต่อไปภายใต้ความรับผิดชอบของทีมงานที่รับผิดชอบโปรเจกต์นี้

4.1.3 การช่วยพัฒนาแอปเก็บข้อมูลบัตรประชาชนผ่านแอปพลิเคชัน

โปรเจกต์นี้สามารถอ่านบัตรประชาชนผ่านเครื่อง EDC ของทางบริษัทได้ รวมไปถึงส่งข้อมูลไปยังแอปพลิเคชันตัวอย่างถือว่าบรรลุตามเป้าหมาย ในอนาคตแอปพลิเคชันนี้จะได้รับการพัฒนาให้สามารถใช้งานได้มีประสิทธิภาพมากยิ่งขึ้น

4.1.4 การพัฒนาแอปพลิเคชัน E-Passport

โปรเจกต์นี้คือการพัฒนาต่อจากแอปพลิเคชัน OCR เพื่อให้ข้อมูลที่ได้นำไปใช้งานต่อได้ โดยงานในส่วนที่ข้าพเจ้าได้รับมอบหมาย คือ สร้าง layout หน้าการตั้งค่า และแปลงข้อมูลเป็นไฟล์ .csv ส่งไปยังอีเมล สามารถทำได้ตามเป้าหมายที่วางไว้ ซึ่งในอนาคตข้างหน้าจะทำการเพิ่มการรับข้อมูลจากบัตรประชาชนและนำข้อมูลไปใช้งานต่อได้ โดยการส่งข้อมูลเพิ่มอีก 2 ช่องทางคือ 1. FTP และ 2. URL ต่อไป



บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการฝึกสหกิจที่บริษัท ดิจิโอ (ประเทศไทย) จำกัด มาเป็นระยะเวลา 4 เดือน ทำให้ได้รับประสบการณ์ในการทำงานร่วมกันเป็นทีมและมีทักษะในการพัฒนาแอปพลิเคชันเพิ่มมากขึ้น โดยผ่านการเรียนรู้และปฏิบัติงานจริงกับพนักงานของทางบริษัท โดยได้ปฏิบัติงานช่วยพัฒนาแอปพลิเคชันด้านการจัดการข้อมูลส่วนบุคคลและช่วยตรวจสอบหาสาเหตุการเกิดข้อผิดพลาดของแอปพลิเคชันระบบชำระเงินในการดำเนินงานต่างๆ ผ่านไปด้วยดี

5.2 แนวทางการแก้ไขปัญหา

ปัญหาในการปฏิบัติงานโดยรวมคือการขาดทักษะและองค์ความรู้ในการทำงานที่ได้รับมอบหมาย ซึ่งต้องใช้ระยะเวลาในการศึกษางาน เมื่อทำความเข้าใจกับงานที่ได้รับมอบหมายจึงสามารถดำเนินการพัฒนาต่อไป ทั้งนี้ยังมีปัญหาที่พบเจอในการปฏิบัติงานบางโครงการคือสมาร์ทโฟนที่ใช้ในการทดลองบางครั้งตัวเครื่องไม่รองรับการทำงานของแอปพลิเคชันที่มีการพัฒนาในระดับที่สูงกว่าตัวเครื่องสมาร์ทโฟน ก่อนใช้งานจึงต้องตรวจสอบเป็นอันดับต้นๆ เพื่อแก้ไขการเขียนแอปพลิเคชันให้รองรับการทำงานกับตัวเครื่องสมาร์ทโฟนที่ใช้ทดลอง

5.3 ข้อเสนอแนะจากการดำเนินงาน

ในการพัฒนาแอปพลิเคชันที่ใช้งานร่วมกับอุปกรณ์ฮาร์ดแวร์ต่าง ๆ อาจจะต้องคอยตรวจสอบและปรับปรุงให้แอปพลิเคชันและตัวอุปกรณ์ฮาร์ดแวร์ถึงเวอร์ชันที่ใช้ให้ตรงกับปัจจุบันมากที่สุด เนื่องจากหากไม่ทำการตรวจสอบหรือปรับปรุงเป็นเวลานานส่งผลให้คำสั่งในแอปพลิเคชันบางคำสั่งถูกยกเลิกไปส่งผลให้แอปพลิเคชันทำงานได้ไม่เต็มประสิทธิภาพ

เอกสารอ้างอิง

- [1] Mindphp.com (14 มีนาคม 2560) *API คืออะไร*. [ออนไลน์], แหล่งที่มา <http://mindphp.com/คู่มือ/73-คืออะไร/2038-api-คืออะไร.html> [1 สิงหาคม 2561].
- [2] Funtion.in.th (ไม่ระบุ) *Activity คือ*. [ออนไลน์], <http://tutorial.function.in.th/android/what-is-activity> [1 สิงหาคม 2560].
- [3] SleepingForLess (22 ตุลาคม 2557) *[Android Code] Fragment Principle - มารู้จักกับ Fragment กันเถอะ*. [ออนไลน์], <http://www.akexorcist.com/2014/10/android-fragment-principle-fragment.html> [1 สิงหาคม 2560].
- [4] SaimTechnology (ไม่ระบุ) *[Android Code] Androids คืออะไร*. [ออนไลน์], www.similantechology.com/news&article/android.html [1 สิงหาคม 2560].
- [5] mindphp.com (14 มีนาคม 2560) *Clone คืออะไร*. [ออนไลน์], mindphp.com/คู่มือ/73-คืออะไร/2090-clone-คืออะไร.html [1 สิงหาคม 2560].
- [6] Supachoke Pienpoldeesakul (3 มีนาคม 2555) *ทำความรู้จัก Androids*. [ออนไลน์], <http://kadroidz.blogspot.com/2012/03/android-architecture.html> [1 สิงหาคม 2560].
- [7] archai. (17 มกราคม 2559). *ทำความรู้จักกับ Android Studio* [ออนไลน์], แหล่งที่มา: <http://arctech.in.th/articles/125> [1 สิงหาคม 2560].
- [8] Mindphp (23 สิงหาคม 2560). *xampp* [ออนไลน์], แหล่งที่มา: <http://www.mindphp.com/คู่มือ/73-คืออะไร/2637-xampp-คืออะไร.html> [10 สิงหาคม 2561].

[9] Genius. (19 กันยายน 2556). *(MySQL) คืออะไร* [ออนไลน์], แหล่งที่มา [https://www.itgenius.co.th/article/\(MySQL\)%20คืออะไร.html](https://www.itgenius.co.th/article/(MySQL)%20คืออะไร.html) [10 สิงหาคม 2561].

[10] Mindphp (23 สิงหาคม 2560). *Visual Studio Code* [ออนไลน์], แหล่งที่มา: <http://www.mindphp.com/บทความ/microsoft/4829-visual-studio-code.html> [10 สิงหาคม 2561].

[11] Mindphp (23 สิงหาคม 2560). *Filezilla คืออะไร วิธีการใช้งาน Filezilla* [ออนไลน์], แหล่งที่มา: <http://www.mindphp.com/forums/viewtopic.php?t=13645> [10 สิงหาคม 2561].

[12] Medium (23 สิงหาคม 2560). *Git คืออะไร* [ออนไลน์], แหล่งที่มา: <https://medium.com/@pakin/git-คืออะไร-git-is-your-friend-c609c5f8efea> [10 สิงหาคม 2561].

[13] Mindphp (14 สิงหาคม 2560). *PHP คืออะไร พี่เอกพี คือภาษาคอมพิวเตอร์ ใช้ในการเขียนโปรแกรมในเว็บ* [ออนไลน์], แหล่งที่มา: <http://mindphp.com/คู่มือ/73-คืออะไร/2127-php-คืออะไร.html> [10 สิงหาคม 2561].

[14] ครูไอทีบทเรียนออนไลน์ (23 ตุลาคม 2560). *รู้จักภาษาจาวา (JAVA)* [ออนไลน์], แหล่งที่มา: <https://www.krui3.com/content/knowledge-of-java/> [10 สิงหาคม 2561]

ภาคผนวก
รายงานประจำสัปดาห์

TNI

ประวัติผู้จัดทำโครงการ

ชื่อ – สกุล

นางสาวชลธิชา กาบคุ้ม

วัน เดือน ปีเกิด

วันที่ 11 กุมภาพันธ์ 2540

ประวัติการศึกษา

ระดับประถมศึกษา

ประถมศึกษาตอนปลาย พ.ศ. 2552

โรงเรียนวัดบางปลานัก

ระดับมัธยมศึกษา

มัธยมศึกษาตอนปลาย พ.ศ. 2558

โรงเรียนคัคครุณี

ระดับอุดมศึกษา

คณะเทคโนโลยีสารสนเทศ สาขาเทคโนโลยีสารสนเทศ พ.ศ. 2561

สถาบันเทคโนโลยีไทย-ญี่ปุ่น

ทุนการศึกษา

- ไม่มี -

ประวัติการฝึกอบรม

1. เข้าร่วมการบรรยายพิเศษในหัวข้อ “Mobile Technology and Internet of Things in Japan” ณ สถาบันเทคโนโลยีไทย-ญี่ปุ่น

2. Work Shop งาน Security Girl Community ครั้งที่ 1 ณ บริษัท เพลย์ทอเรียม โซลูชันส์ จำกัด (PTS)

ผลงานที่ได้รับการตีพิมพ์ - ไม่มี -

