



การพัฒนาส่วนฟรอนต์เอนด์สำหรับ
ไอเซอร์วิส: แอปพลิเคชันเฮลป์เดสก์สำหรับระบบพีโอเอส
IMPLEMENTING A FRONT-END FOR ISERVICE: A HELP-DESK APPLICATION
FOR POS SYSTEM

นางสาวสุภาสินี ยมนา

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยี ไทย-ญี่ปุ่น

พ.ศ. 2560

การพัฒนาส่วนฟรอนต์เอนด์สำหรับไอเซอร์วิส:แอปพลิเคชันเซลล์ปัดสก์สำหรับระบบพีโอเอส

IMPLEMENTING A FRONT-END FOR ISERVICE: A HELP-DESK APPLICATION
FOR POS SYSTEM

นางสาวสุภาสินี ยมนา

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีไทย - ญี่ปุ่น
ปีการศึกษา 2560

คณะกรรมการสอบ

..... ประธานกรรมการสอบ

(ผศ.ดร. อรรถพร ห่มั่นสกุล)

..... กรรมการสอบ

(อาจารย์ อมรพันธ์ ขมกลืน)

..... อาจารย์ที่ปรึกษา

(อาจารย์ ดร. โสภณ มงคลลักษณ์)

..... ประธานสหกิจศึกษาสาขาวิชา

(อาจารย์ อมรพันธ์ ขมกลืน)

ลิขสิทธิ์ของสถาบันเทคโนโลยีไทย - ญี่ปุ่น

ชื่อโครงการ	การพัฒนาส่วนฟรอนต์เอนด์สำหรับไอเซอร์วิส:แอปพลิเคชันเฮลป์เดสก์ สำหรับระบบพีโอเอส IMPLEMENTING A FRONT-END FOR ISERVICE: A HELP-DESK APPLICATION FOR POS SYSTEM
ผู้เขียน	นางสาวสุภาสินี ขมณา
คณะวิชา	เทคโนโลยีสารสนเทศ สาขาวิชา เทคโนโลยีสารสนเทศ
อาจารย์ที่ปรึกษา	ดร.โสภณ มงคลลักษณ์
พนักงานที่ปรึกษา	นายชนกฤต รัศมี
ชื่อบริษัท	บริษัท ไอเก็ท ยูทีลิตี้ จำกัด
ประเภทธุรกิจ/สินค้า	บริษัทพัฒนาซอฟต์แวร์

บทสรุป

บริษัท ไอเก็ท ยูทีลิตี้ จำกัด ผู้พัฒนาซอฟต์แวร์ระบบ POS หรือ Point of Sale ซึ่งเป็นระบบคิดเงินที่สามารถเก็บข้อมูลการขาย และข้อมูลการบันทึกค่าใช้จ่ายเมื่อมีการขายสินค้า เมื่อเกิดปัญหาในการใช้งาน ฝ่ายสนับสนุนต้องเข้ามาดูแล ให้คำแนะนำเพื่อแก้ไขปัญหาต่าง ๆ ของลูกค้า โดยในปัจจุบันใช้แอปพลิเคชันไลน์ เฟซบุ๊ก และโทรศัพท์เป็นช่องทางการติดต่อเพื่อแก้ปัญหา ซึ่งไม่มีการเก็บข้อมูลในการแก้ไขปัญหาอย่างมีรูปแบบ เพื่อเป็นการแก้ปัญหาทางบริษัทจึงมอบหมายให้สร้างแอปพลิเคชัน iService สำหรับสมาร์ตโฟน เพื่อใช้เป็นช่องทางการแจ้งปัญหา พร้อมทั้งสามารถให้ข้อมูลการแก้ไขปัญหาเบื้องต้น ในรายงานนี้เป็นส่วนของการพัฒนาส่วนฟรอนต์เอนด์คือการเขียนหน้าอินเตอร์เฟซ ในการพัฒนานี้ใช้ ionic framework ภาษา HTML/CSS และ JavaScript

จากการดำเนินโครงการทำให้ได้แอปพลิเคชัน iService ที่สามารถช่วยเหลือและแก้ไขปัญหาให้ลูกค้าผู้ใช้งานระบบ POS ได้รวดเร็วยิ่งขึ้น พร้อมทั้งลูกค้าสามารถส่งปัญหาไปยังฝ่ายสนับสนุนได้โดยตรง และสามารถจัดเก็บข้อมูลได้อย่างมีประสิทธิภาพ

กิตติกรรมประกาศ

ขอขอบคุณบริษัท ไอเก็ด ยูทียู จำกัด ที่มอบโอกาสให้ข้าพเจ้าได้มาเป็นส่วนหนึ่งในการปฏิบัติงาน ได้ทดลองทำงานจริง ๆ เป็นประสบการณ์ที่หาไม่ได้ในห้องเรียน ทั้งเชิงทฤษฎีและการปฏิบัติ ช่วยชี้แนะวิธีการทำงาน อีกทั้งยังสอนให้รู้จักการเรียนรู้ และแก้ไขปัญหาด้วยตนเอง ทำให้ข้าพเจ้าสามารถปฏิบัติงานได้อย่างราบรื่น

ขอขอบคุณพี่พนักงานในบริษัททุกคนที่คอยให้ความช่วยเหลือ ให้คำแนะนำ คอยดูแลอย่างดีตลอดการปฏิบัติงานครั้งนี้ อีกทั้งยังได้แนวคิดใหม่ ๆ ที่สามารถนำมาปรับใช้จริงในการทำงาน และรับฟังความคิดเห็นของนักศึกษาสหกิจในการวางแผนงาน รวมถึงอาจารย์ที่ปรึกษา ดร.โสภณ มงคลลักษณ์ และพนักงานที่ปรึกษา นายธนกฤต รัตมี ที่คอยให้คำปรึกษาเกี่ยวกับโครงการตลอดมา ทำให้การสหกิจศึกษาของข้าพเจ้าสำเร็จลุล่วง สุดท้ายนี้ข้าพเจ้าขอขอบคุณคณะเทคโนโลยีสารสนเทศ และสถาบันเทคโนโลยีไทย-ญี่ปุ่น ที่ให้โอกาสให้ข้าพเจ้าได้มาสหกิจในครั้งนี้ ขอขอบคุณมา ณ ที่นี้ค่ะ

สุภาสินี ขมนา

ผู้จัดทำโครงการ

TNI

THAI - NICHI INSTITUTE OF TECHNOLOGY

สารบัญ

บทสรุป	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
สารบัญตาราง	ฅ
สารบัญภาพประกอบ	ช

บทที่

บทที่ 1 บทนำ	1
1.1 ชื่อและที่ตั้งของสถานประกอบการ	1
1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร	2
1.3 รูปแบบการบริหารและจัดการองค์กร	2
1.4 ตำแหน่งและหน้าที่งานที่ได้รับมอบหมาย	2
1.5 พนักงานที่ปรึกษาและตำแหน่งของพนักงานที่ปรึกษา	2
1.6 ระยะเวลาที่ปฏิบัติงาน	2
1.7 ที่มาและความสำคัญของปัญหา	3
1.8 วัตถุประสงค์ หรือจุดมุ่งหมายของโครงการ	3
1.9 ผลที่คาดว่าจะได้รับจากการปฏิบัติงาน หรือโครงการที่ได้รับมอบหมาย	3
1.10 นิยามศัพท์เฉพาะ	4
บทที่ 2 ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน	6
2.1 ทฤษฎีที่ใช้ในการปฏิบัติงาน	6
2.2.1 ภาษา HTML	6
2.2.2 ภาษา JavaScript	8
2.2.3 ภาษา CSS	9
2.2.4 วัฏจักรการพัฒนาระบบงาน	10
2.2 เทคโนโลยีที่ใช้ในการปฏิบัติงาน	13
2.1.1 ionic framework	13
2.1.2 Node.js	14

สารบัญ (ต่อ)

2.1.3	Android Studio	15
2.1.4	Visual Studio Code	16
2.1.5	Xcode	17
บทที่ 3	แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน	19
3.1	แผนการปฏิบัติงาน	19
3.2	รายละเอียดโครงการ	20
3.2.1	แอปพลิเคชัน iService	20
3.2.2	งานอื่นๆ	20
3.3	ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติหรือโครงการ	21
3.3.1	รวบรวม requirement	21
3.3.2	ขอบเขตของงาน	22
3.3.3	นำเสนอในที่ประชุม	22
3.3.4	พัฒนาแอปพลิเคชัน	22
3.3.5	ทำการเขียนหน้าฟรอนเอนด์ที่ได้รับมอบหมาย	25
3.3.6	ทดสอบและแก้ไขแอปพลิเคชัน	25
บทที่ 4	ผลการดำเนินงาน การวิเคราะห์และสรุปผล	27
4.1	ผลการดำเนินงาน	27
4.1.1	ผลการดำเนินงานของหน้าแอปพลิเคชัน	27
บทที่ 5	บทสรุปและข้อเสนอแนะ	40
5.1	สรุปผลการดำเนินงาน	40
5.2	ข้อเสนอแนะจากการดำเนินงาน	40
5.3	ปัญหาและอุปสรรคที่เกิดขึ้น	40
5.4	แนวทางการแก้ไขปัญหา	40
	เอกสารอ้างอิง	41

สารบัญ (ต่อ)

ภาคผนวก	43
ก. รายงานการปฏิบัติงานประจำสัปดาห์ (Weekly Report)	44
ประวัติผู้จัดทำโครงการ	63



สารบัญตาราง

ตารางที่

หน้า

3.1 แผนงานการปฏิบัติงาน

19



สารบัญภาพประกอบ

ภาพที่	หน้า
1.1 สัญลักษณ์ของบริษัท	1
1.2 แผนที่แสดงตำแหน่งของบริษัท	1
2.1 สัญลักษณ์ของ HTML 6	6
2.2 สัญลักษณ์ของ JavaScript	8
2.3 สัญลักษณ์ของ CSS	10
2.4 แผนภาพวัฏจักรการพัฒนาระบบงาน	11
2.5 สัญลักษณ์ของ ionic framework	13
2.6 ภาพเปรียบเทียบความแตกต่าง Platform ของระบบปฏิบัติการ iOS และ Android	14
2.7 สัญลักษณ์ของ Node.js	14
2.8 สัญลักษณ์ของ npm	15
2.9 สัญลักษณ์ของโปรแกรม Android Studio	16
2.10 สัญลักษณ์ของ Visual Studio	17
2.11 สัญลักษณ์ของโปรแกรม Xcode	17
2.12 หน้าตาของ XcodeIDE	18
3.1 สัญลักษณ์ของโปรแกรม React Native	21
3.2 ความแตกต่างของการแสดงผลระหว่างของระบบปฏิบัติการ Android และ iOS	21
3.3 แสดงหน้าตาของการดาวโหลดโปรแกรม Node.js	23
3.4 คำสั่งในการติดตั้ง ionic framework ผ่านคำสั่งคอมมานไลน์	23
3.5 คำสั่งในการติดตั้ง ionic framework ผ่านคำสั่งคอมมานไลน์ (ต่อ)	23
3.6 แสดงหน้าตา Document ของ ionic framework	24
3.7 แสดงหน้าตาของการดาวโหลดโปรแกรม Visual Studio Code	24
3.8 ตัวอย่างหน้าตาของ Visual Studio Code	25
3.9 หน้าตาของการรันโปรแกรม ionic framework ผ่านเซิร์ฟเวอร์	26
3.10 ภาพ User Interface ของโปรแกรม Simulator กรณีเลือกให้แสดง simulator สำหรับ iphone 6	26
4.1 ภาพรวมของหน้าแจ้งปัญหา	27

สารบัญภาพประกอบ (ต่อ)

ภาพที่	หน้า
4.2 หน้าแจ้งปัญหาส่วนเลือกหัวข้อปัญหา	28
4.3 หน้าแจ้งปัญหาส่วนหัวข้อย่อยและเลือกร้านค้า	29
4.4 หน้าแจ้งปัญหาส่วนเบอร์โทรศัพท์และอีเมล	30
4.5 หน้าแจ้งปัญหาส่วนยืนยันรายการ	31
4.6 หน้าแจ้งปัญหาส่วนหน้า Splash	32
4.7 ภาพรวมของหน้าคำถามที่พบบ่อย	33
4.8 หน้าคำถามที่พบบ่อยแสดงหัวข้อย่อย	34
4.9 หน้าคำถามที่พบบ่อยแสดงรายละเอียดของปัญหา	35
4.10 หน้าคำถามที่พบบ่อยแสดงผลการค้นหา	36
4.11 หน้าคำถามที่พบบ่อยเมื่อค้นหาในระบบไม่พบ	37
4.12 ภาพรวมของหน้าตั้งค่า	38
4.13 หน้าตั้งค่าส่วนแสดงเวอร์ชัน	39
4.14 แถบเมนูบาร์	39

บทที่ 1

บทนำ

1.1 ชื่อและที่ตั้งของสถานประกอบการ

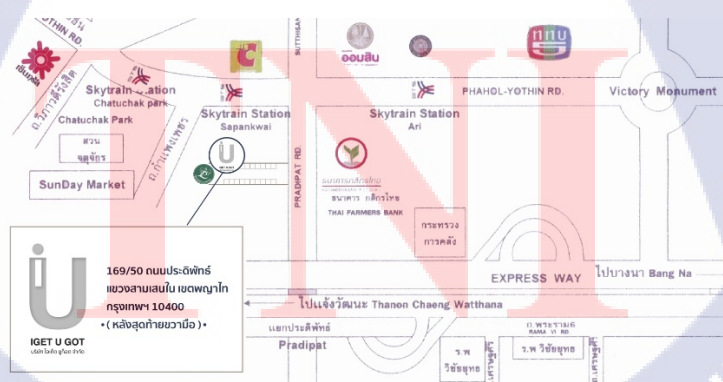


IGET UGOT Co.,Ltd.
บริษัท ไอเก็ท ยูก็อต จำกัด

ภาพที่ 1.1 สัญลักษณ์ของบริษัท

ชื่อ บริษัท ไอเก็ท ยูก็อต จำกัด (IGET U GOT Co., Ltd.)

ที่ตั้ง 169/50 ถนนประดิพัทธ์ แขวงสามเสนใน เขตพญาไท กรุงเทพฯ 10400



ภาพที่ 1.2 แผนที่แสดงตำแหน่งของบริษัท

1.2 ลักษณะธุรกิจของสถานประกอบการ หรือการให้บริการหลักขององค์กร

บริษัท ไอเก็ต ยูก็อต จำกัด (IGET U GOT Co., Ltd.) เป็นบริษัทที่ให้บริการด้านงานพัฒนาซอฟต์แวร์ประเภท POS คือ iGeT POS และ iGeT M โดยเน้นกลุ่มลูกค้าไปที่ร้านค้าต่างๆ อาทิร้านค้าขนาดเล็กจนถึงร้านค้าขนาดกลาง

และยังประกอบกิจการอีกประเภท ทั้งการรับออกแบบ พัฒนา และเป็นที่ปรึกษาทางด้านโปรแกรมซอฟต์แวร์ทุกชนิด ทุกภาษา พร้อมทั้งรับผิดชอบโปรแกรมคอมพิวเตอร์ตามความต้องการของลูกค้า ทั้งเว็บไซต์ วินโดวส์ หรือระบบอื่น ๆ ทุกชนิด มีการจำหน่ายตัวซอฟต์แวร์ ไลเซนส์ ทุกชนิด ประกอบกิจการรับออกแบบ รับดูแล และพัฒนาระบบเซิร์ฟเวอร์ทุกรูปแบบ เช่น ยูนิค โซลาลิส จำหน่ายคอมพิวเตอร์และอุปกรณ์ต่อพ่วงทุกชนิดเกี่ยวกับเทคโนโลยี คอมพิวเตอร์ไฟฟ้า และอิเล็กทรอนิกส์ อีกทั้งยังรับเป็นที่ปรึกษาและให้คำแนะนำทางด้านการบริหารงานทุกชนิด ทุกประเภท

1.3 รูปแบบการจัดการองค์กรและบริหารองค์กร

บริษัท ไอเก็ต ยูก็อต จำกัด (IGET U GOT Co., Ltd.) มีแผนกในการดำเนินการดังนี้ แผนกบริหาร แผนกขาย แผนกบัญชี แผนกการตลาด และ แผนกพัฒนาซอฟต์แวร์

1.4 ตำแหน่งและหน้าที่งานที่ได้รับมอบหมาย

ตำแหน่ง Front-End Development

หน้าที่ จัดทำแอปพลิเคชัน iService ให้บริการลูกค้าในระบบปฏิบัติการ iOS และ Android โดยจะทำในส่วนหน้าของหน้าแจ้งปัญหา หน้าตั้งค่า และหน้าคำถามที่พบบ่อย

1.5 พนักงานที่ปรึกษาและตำแหน่งของพนักงานที่ปรึกษา

ชื่อ นายชนกฤต รัศมี

ตำแหน่ง โปรแกรมเมอร์

1.6 ระยะเวลาที่ปฏิบัติงาน

29 พฤษภาคม - 29 กันยายน 2560

1.7 ที่มาและความสำคัญของปัญหา

ในปัจจุบัน เทคโนโลยีสารสนเทศมีความก้าวหน้าอย่างมาก จึงทำให้มีการพัฒนาและเปลี่ยนแปลงอย่างต่อเนื่อง บริษัท ไอเก็ท ยูทีกอต จำกัด เป็นผู้พัฒนาซอฟต์แวร์ SaaS หรือ Software as a service และมีซอฟต์แวร์ที่ชื่อว่า iGET POS และ iGET M ซึ่งที่เรียกว่า POS คือ Point Of Sale ระบบคิดเงินที่สามารถเก็บข้อมูลการขาย และข้อมูลการบันทึกค่าใช้จ่ายที่เกิดขึ้น เมื่อมีการขายสินค้า การบริการ ระบบนี้จะมีความเกี่ยวข้องกับเทคโนโลยีทุกอย่างผสม ๆ กัน

บริษัท ไอเก็ท ยูทีกอต จำกัด ได้พัฒนาซอฟต์แวร์ระบบ POS ซึ่งการใช้งานในบางครั้งอาจมีปัญหาก่อเกิดขึ้น ทำให้ฝ่ายซัพพอร์ตต้องเข้ามาดูแลในการช่วยให้คำแนะนำในการแก้ปัญหาส่วนนี้ โดยจะใช้ไลน์ เฟซบุ๊ก และโทรศัพท์ ซึ่งบางครั้งปัญหาที่ต้องแก้เป็นหาเดิม ๆ ที่ผู้ใช้งานสามารถแก้ไขเองได้ ทางบริษัทเล็งเห็นว่าควรจัดทำแอปพลิเคชันเพื่ออำนวยความสะดวกกับผู้ใช้งานและฝ่าย Support ในแอปพลิเคชันนี้จะให้ข้อมูลเรื่องการแก้ไขปัญหาเบื้องต้น เพื่อให้ผู้ใช้งานแก้ไขเบื้องต้นด้วยตนเองได้โดยไม่ต้องรอฝ่ายซัพพอร์ตและมีช่องทางให้ผู้ใช้งานแจ้งปัญหากับฝ่ายซัพพอร์ตโดยตรง อีกทั้งผู้ใช้งานยังสามารถเข้าไปดูประวัติการแจ้งย้อนหลังได้อีกด้วย

1.8 วัตถุประสงค์หรือจุดมุ่งหมายของโครงการ

สร้างแอปพลิเคชัน iService บนระบบปฏิบัติการ iOS และ Android เพื่ออำนวยความสะดวกแก่ผู้ใช้งาน โดยมีวัตถุประสงค์ดังต่อไปนี้

- 1) เพื่อรวบรวมปัญหาที่พบเจอบ่อยให้ลูกค้าเข้ามาค้นหาเพื่อแก้ไขปัญหาเบื้องต้นด้วยตนเอง
- 2) เพื่อเพิ่มช่องทางติดต่อระหว่างลูกค้ากับฝ่ายซัพพอร์ต
- 3) เพื่อจัดการการแก้ไขปัญหาให้มีประสิทธิภาพมากยิ่งขึ้น
- 4) เพื่อประหยัดเวลาทั้งลูกค้าและฝ่ายซัพพอร์ต

1.9 ผลที่คาดว่าจะได้รับการปฏิบัติงาน หรือโครงการที่ได้รับมอบหมาย

สามารถสร้างฟีเจอร์ตามที่กำหนดไว้ให้เสร็จสมบูรณ์มากที่สุดและได้ประสบการณ์จากการฝึกงานครั้งนี้ไปปรับใช้ต่อยอดได้ในอนาคต

1.10 นิยามศัพท์เฉพาะ

1. User Experience (UX)

User Experience [3] คือ การออกแบบที่เกี่ยวข้องกับความรู้สึกของการใช้งาน ว่ามันยากหรือง่าย จะสะดวกกับผู้ใช้งานหรือไม่ ต้องทำการเก็บข้อมูล ทดลองเพื่อทำตามความต้องการของผู้ใช้งาน

2. User Interface (UI)

User Interface [3] คือ การออกแบบส่วนที่ใช้ติดต่อกับผู้ใช้งาน ไม่ว่าจะเป็น เมนู ปุ่ม หรือการจัดวางภาพ ขนาดตัวอักษรของแต่ละหน้า ต้องคำนึงว่าจะสะดวกกับผู้ใช้งานไหม

3. Compiler

Compiler [4] คือ โปรแกรมที่ทำหน้าที่แปลโปรแกรมที่เขียนขึ้นด้วยภาษาคอมพิวเตอร์ระดับสูง ให้เป็นรหัสที่อยู่ในรูปแบบที่เครื่องคอมพิวเตอร์สามารถทำงานได้ ตัวอย่างของโปรแกรมแปลภาษา เช่น ตัวแปลภาษาจาวา ตัวแปลภาษาซี

4. Terminal

Terminal [5] เป็นโปรแกรมสำหรับรับคำสั่งคอมพิวเตอร์ไลน์ (ข้อความ) คล้ายกับโปรแกรม command.com หรือ cmd.exe บน Windows

5. Front-end Development

Front-end Development คือ การพัฒนาแอปพลิเคชันในส่วนที่มีการติดต่อกับผู้ใช้งาน มีหน้าที่พัฒนาให้ใช้งานง่ายสะดวกกับผู้ใช้งาน เช่น ปุ่มกด แถบเมนู การวางคอนเทนท์

6. Back-end Development

Back-end Development คือ การพัฒนาแอปพลิเคชันในส่วน of ข้อมูล ฐานข้อมูล คอยควบคุมระบบต่าง ๆ ของแอปพลิเคชัน

7. Framework

Framework [6] คือ โครงสร้างการเขียนโปรแกรม มีโค้ดที่วางไว้อย่างเป็นระบบ มีรูปแบบแผน และลักษณะการเขียนเป็นมาตรฐานของโปรแกรมต่าง ๆ เมื่อเกิดปัญหาจะสามารถแก้ไขได้ง่าย

8. Application Programming Interface (API)

API [7] คือ ช่องทางการเชื่อมต่อระหว่างผู้ใช้งานกับ Server หรือจาก Server เชื่อมต่อไปหา Server ซึ่ง API เปรียบเหมือนภาษาคอมพิวเตอร์ที่ทำให้คอมพิวเตอร์สามารถสื่อสารข้อมูลกันได้อย่างอิสระ

9. Point of Sale (POS)

POS [1] คือ ระบบคิดเงินที่สามารถเก็บข้อมูลการขาย และข้อมูลการบันทึกค่าใช้จ่ายที่เกิดขึ้น เมื่อมีการขายสินค้า การบริการ ระบบนี้จะมีความเกี่ยวข้องกับเทคโนโลยีหาอย่างผสม ๆ

10. iGet POS

iGet POS คือ ซอฟต์แวร์ที่ถูกพัฒนาขึ้นเพื่อให้พนักงานหน้าร้านใช้งาน โดยจะมีลักษณะการทำงานเป็น POS กล่าวคือใช้ในการจัดการหน้าร้านต่าง ๆ

11. iGet M

iGet M คือ ซอฟต์แวร์ที่ถูกพัฒนาขึ้นเพื่อให้พนักงานบัญชี และผู้บริหารใช้งาน โดยซอฟต์แวร์นี้จะใช้การจัดการ iGET POS เช่น การนำข้อมูลจากฐานข้อมูลมาทำการเชื่อมโยงกับ iGET POS

12. Controller

Controller [8] คือ การจัดการโปรแกรม เช่น การประมวลผล หรือการรับข้อมูล

บทที่ 2

ทฤษฎีและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

2.1 ทฤษฎีที่ใช้ในการปฏิบัติงาน

2.1.1 ภาษา HTML

2.1.1.1 HTML คืออะไร

HTML (Hypertext Markup Language) [20] คือ ภาษาหลักที่ใช้ในการเขียนเว็บเพจ โดยใช้ Tag ในการกำหนดการแสดงผล โดย Hypertext หมายถึง ข้อความที่เชื่อมต่อกันผ่านลิงค์ (Hyperlink) Markup language หมายถึงภาษาที่ใช้ Tag ในการกำหนดการแสดงผลสิ่งต่าง ๆ ที่แสดงอยู่บนเว็บเพจ ดังนั้น HTML จึงหมายถึง ภาษาที่ใช้ Tag ในการกำหนดการแสดงผลเว็บเพจที่ต่างก็เชื่อมถึงกันใน Hyperspace ผ่าน Hyperlink



ภาพที่ 2.1 สัญลักษณ์ของ HTML

2.1.1.2 ประวัติความเป็นมา

ความเป็นมาของ HTML เริ่มขึ้นเมื่อปี 1980 เมื่อ Tim Berners Lee เสนอต้นแบบสำหรับนักวิจัยใน CERN เพื่อแลกเปลี่ยนเอกสาร ข้อมูลด้านการวิจัย โดยใช้ชื่อว่า Enquire ในปี 1990 ได้เขียนโปรแกรมเบราเซอร์ และทดลองรันบนเซิร์ฟเวอร์ที่เค้าพัฒนาขึ้น HTML ได้รับการรู้จักจาก HTML Tag ซึ่งมีอยู่ 18 Tag ในปี 1991

HTML ถูกพัฒนาจาก SGML และ Tim ก็คิดเสมือนว่า HTML เป็นโปรแกรมย่อยของ SGML อยู่ในตอนนั้น ต่อมาในปี 1996 เพื่อกำหนดมาตรฐานให้ตรงกัน W3C World Wide Web Consortium จึงเป็นผู้กำหนดสเปกทั้งหมดของ HTML และปี 1999 HTML 4.01 ก็ถือกำเนิดขึ้น โดยมี HTML 5 ซึ่งเป็น Web Hypertext Application ถูกพัฒนาต่อมาในปี 2004 นอกจากนี้ยังมีการพัฒนาไปเป็น

XHTML ซึ่ง คือ Extended HTML ซึ่งมีความสามารถและมาตรฐานที่รัดกุมกว่าอีกด้วย โดยอยู่ภายใต้การควบคุมของ W3C (World Wide Web Consortium)

2.1.1.3 หลักการเขียนภาษา HTML

หลักการเขียนภาษา HTML มีดังนี้

1) คำสั่ง หรือ Tag

Tag เป็นลักษณะเฉพาะของภาษา HTML ใช้ในการระบุรูปแบบคำสั่งหรือการลงรหัสคำสั่ง HTML ภายในเครื่องหมาย less-than bracket (<) และ greater-than bracket (>) โดยที่ Tag HTML แบ่งได้ 2 ลักษณะ คือ

Tag เดี่ยว เป็น Tag ที่ไม่ต้องมีการปิดรหัส เช่น <HR>,
 เป็นต้น

Tag เปิด/ปิด รูปแบบของ tag นี้จะเป็นแบบ <tag> </tag> โดยที่

<tag> เราเรียกว่า tag เปิด

</tag> เราเรียกว่า tag ปิด

2) Attributes

Attributes เป็นตัวบอกรายละเอียดของ tag นั้นเช่น ... เป็นการบอกว่าให้อักขรที่อยู่ใน tag นี้ชิดซ้าย

3) Case Insensitivity

ภาษา HTML เป็นภาษาแบบ case-insensitive ซึ่งลักษณะของภาษาชนิดนี้ ทำให้การเขียน Lowercase หรือ Uppercase จึงไม่มีความแตกต่างกันในมุมมองของ Interpreter

4) โครงสร้างของหลักของ HTML

โครงสร้างหลักของ HTML ก็จะเริ่มด้วย <html> และจบด้วย </html> เสมอ ซึ่งชุดคำสั่งที่ใช้จะแยกเป็น 2 ส่วนคือ

1. **head** คำสั่งที่อยู่ในส่วนนี้จะใช้บรรยายรายละเอียดเกี่ยวกับ web page ซึ่งจะไม่แสดงผลที่ web page โดยตรง

2. **body** คำสั่งที่อยู่ในส่วนนี้จะใช้ในการจัดรูปแบบตัวอักษร จัดหน้า ใส่รูปภาพ ซึ่งตัวอักษรในส่วนนี้จะแสดงที่เว็บเบราว์เซอร์โดยตรง

2.1.2 ภาษา JavaScript

2.1.2.1 JavaScript คืออะไร

JavaScript [9] คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง Java JavaScript เป็น ภาษาสคริปต์เชิงวัตถุ ที่เรียกกันว่า "สคริปต์" (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ ใช้ร่วมกับ HTML เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะแปลความและดำเนินงานไปทีละคำสั่ง (interpret) หรือเรียกว่า อ็อบเจกต์โอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และ ภาษา Java ได้ทั้งทางฝั่ง Client และ ทางฝั่ง Server



ภาพ 2.2 สัญลักษณ์ของ JavaScript

2.1.2.2 ข้อดีของภาษา JavaScript

1) สามารถใช้ตรวจสอบข้อมูลได้ สังเกตว่าเมื่อเรากรอกข้อมูลบางเว็บไซต์ เช่น Email เมื่อเรากรอกข้อมูลผิดจะมีหน้าต่างฟ้องขึ้นมาว่าเรากรอกผิด หรือลืมกรอกอะไรบางอย่าง เป็นต้น

2) สามารถเขียนหรือเปลี่ยนแปลง HTML Element ได้ นั่นคือสามารถเปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้ หรือหน้าแสดงเนื้อหาสามารถซ่อนหรือแสดงเนื้อหาได้แบบง่าย ๆ นั่นเอง

3) คำสั่งที่ตอบสนองกับผู้ใช้งาน เช่นเมื่อผู้ใช้คลิกที่ปุ่ม หรือ Checkbox ก็สามารถสั่งให้เปิดหน้าต่างใหม่ได้ ทำให้เว็บไซต์ของเรามีปฏิสัมพันธ์กับผู้ใช้งานมากขึ้น นี่ก็คือข้อดีของ JavaScript เลยก็ว่าได้ที่ทำให้เว็บไซต์ต่าง ๆ ทั้งหลายเช่น Google Map ต่างหันมาใช้

2.1.2.3 ข้อเสียของภาษา JavaScript

JavaScript มีข้อจำกัด คือไม่สามารถรับและส่งข้อมูลต่าง ๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ชม เพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น

2.1.3 CSS

2.1.3.1 CSS คืออะไร

CSS [10] ย่อมาจาก Cascading Style Sheet มักเรียกโดยย่อว่า "สไตลชีท" คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (หรือ "Style") ของเนื้อหาในเอกสาร อันได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ หรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสาร ไม่ขึ้นอยู่กับเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลผลลัพธ์ของเอกสาร HTML โดยเฉพาะในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสารบ่อยครั้ง หรือต้องการควบคุมให้รูปแบบการแสดงผลเอกสาร HTML มีลักษณะของความสม่ำเสมอทั่วกันทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน โดยกฎเกณฑ์ในการกำหนดรูปแบบ (Style) เอกสาร HTML ถูกเพิ่มเข้ามาครั้งแรกใน HTML 4.0 เมื่อปีพ.ศ. 2539 ในรูปแบบของ CSS level 1 Recommendations ที่กำหนดโดยองค์กร World Wide Web Consortium หรือ W3C



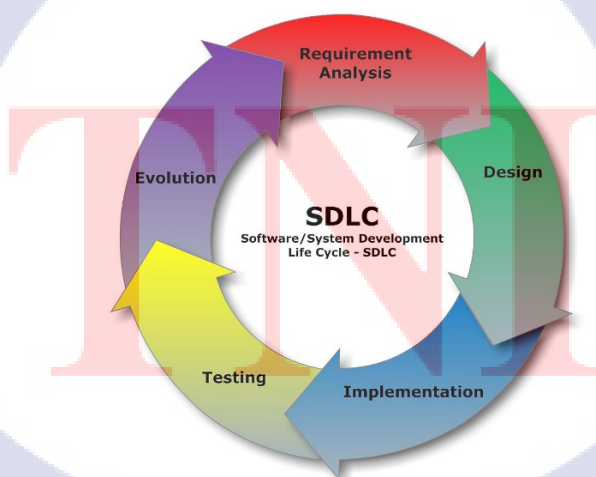
ภาพที่ 2.3 สัญลักษณ์ของ CSS

2.1.3.2 ประโยชน์ของ CSS

ประโยชน์ของ CSS มีดังนี้

- 1) ภาษา CSS จะช่วยในการจัดรูปแบบแสดงผลให้กับภาษา HTML ซึ่งจะช่วยลดการใช้ภาษา HTML ให้น้อยลง โดยเหลือเพียงแต่ส่วนที่เป็นเอกสารที่เป็นภาษา HTML เท่านั้นทำให้มีการแก้ไขและทำความเข้าใจได้ง่ายขึ้น
- 2) ทำให้ขนาดไฟล์ HTML น้อยลงเนื่องจาก ภาษา CSS จะช่วยลดการใช้ภาษา HTML ลงทำให้ขนาดไฟล์นั้นก็เล็กลงไปด้วยเช่นกัน
- 3) ภาษา CSS เป็นภาษา Style Sheets โดย Style Sheets ชุดเดียวสามารถใช้กำหนดรูปแบบการแสดงผลให้เอกสาร HTML ทั้งหน้า หรือทุกหน้ามีผลเหมือนกันได้ จึงทำให้เวลาที่มีการแก้ไขก็จะแก้ไขได้ง่ายขึ้นเพียงแก้ไข Style Sheets ที่ใช้งานเพียงชุดเดียวเท่านั้น
- 4) ทำให้เว็บไซต์มีมาตรฐานเพราะการใช้งาน CSS นั้นจะทำให้การแสดงผลในสื่อต่าง ๆ ถูกปรับเปลี่ยนไปได้เหมาะสม เช่น การแสดงผลบนหน้าจอ และการแสดงผลในมือถือ
- 5) สามารถที่จะใช้งานได้หลากหลาย เว็บเบราว์เซอร์ ทำให้การใช้งานนั้นสะดวกมากยิ่งขึ้น
- 6) สามารถกำหนดแยกไว้ต่างหากจากไฟล์เอกสาร HTML และสามารถนำมาใช้ร่วมกับเอกสารหลายไฟล์ได้ การแก้ไขก็แก้ไขเพียง จุดเดียวก็มีผลกับเอกสารทั้งหมด

2.1.4 วัฏจักรการพัฒนาระบบงาน) System development Life Cycle : SDLC)



ภาพที่ 2.4 แผนภาพวัฏจักรการพัฒนาระบบงาน [19]

วัฏจักรการพัฒนากระบวนการ (System development Life Cycle : SDLC) [18] หมายถึง ขั้นตอนหรือกระบวนการในการพัฒนาระบบงาน ซึ่งครอบคลุมตั้งแต่จุดเริ่มต้นในการทำงานและจุดสิ้นสุดของการปฏิบัติงาน

การพัฒนาซอฟต์แวร์ ตามปกติแล้วจะประกอบไปด้วยกลุ่มกิจกรรม 3 ส่วนหลัก ๆ ด้วยกัน คือ การวิเคราะห์ (Analysis), การออกแบบ (Design) และการนำไปใช้ (Implementation) ซึ่งกิจกรรมทั้งสามนี้สามารถใช้งานได้ดีกับโครงการซอฟต์แวร์ขนาดเล็ก ในขณะที่โครงการซอฟต์แวร์ขนาดใหญ่ มักจำเป็นต้องใช้แบบแผนการพัฒนาซอฟต์แวร์ตามแนวทางของ SDLC จนครบทุกกิจกรรม

วงจรการพัฒนากระบวนการ (System Development Life Cycle :SDLC) ระบบสารสนเทศทั้งหลายมีวงจรชีวิตที่เหมือนกันตั้งแต่เกิดจนตายวงจรนี้จะเป็นขั้นตอน ที่เป็นลำดับตั้งแต่ต้นจนเสร็จเรียบร้อย เป็นระบบที่ใช้งานได้ ซึ่งนักวิเคราะห์ระบบต้องทำความเข้าใจให้ดีว่าในแต่ละขั้นตอนจะต้องทำอะไร และอย่างไร

จากภาพที่ 2.1 เป็นการแสดงขั้นตอนการพัฒนากระบวนการประกอบด้วย 5 ขั้นตอน ได้แก่

- 1) วิเคราะห์ความต้องการ (Requirement Analysis)
- 2) ออกแบบ (Design)
- 3) สร้างหรือพัฒนาระบบ (Implementation)
- 4) ทดสอบ (Testing)
- 5) บำรุงรักษาและประเมินผล (Maintenance and Evaluation)

ขั้นตอนที่ 1 วิเคราะห์ความต้องการ (Requirement Analysis)

ระบบสารสนเทศจะเกิดขึ้นได้ก็ต่อเมื่อผู้บริหารองค์กรมีความตระหนักถึงปัญหาของระบบเดิมที่ใช้งานอยู่ เช่น ระบบจัดเก็บเอกสารในตู้เอกสาร ก่อให้เกิดปัญหาทั้งค่าใช้จ่ายสำหรับสร้างพื้นที่จัดเก็บเอกสาร ค่าบูรณะกระดาษให้มีอายุการคงอยู่ที่ยืนนาน รวมถึงปัญหาในการค้นหาเอกสารในแต่ละครั้ง หรืออาจเป็นปัญหาในการจัดเรียงเอกสารให้ถูกต้อง ซึ่งปัญหาเหล่านี้สามารถแก้ไขได้โดยการสร้างระบบจัดเก็บข้อมูลเอกสารอิเล็กทรอนิกส์ ที่มีคุณสมบัติคือพื้นที่จัดเก็บน้อยลง เอกสารอิเล็กทรอนิกส์ไม่มีวันสลาย และสามารถค้นหาและจัดเรียงเอกสารได้อย่างง่ายดาย และถูกต้องแม่นยำ เป็นต้น จะเห็นว่าหากมีความชัดเจนในการจำแนกปัญหาเพียงพอ จะสามารถสร้างระบบเพื่อแก้ไขปัญหาได้ แต่ในทางกลับกัน หากการจำแนกปัญหาของระบบยังมีความคลุมเครือ ไม่ชัดเจน การออกแบบระบบขึ้นมาใหม่ อาจต้องเสียค่าใช้จ่ายที่ไม่จำเป็น เพื่อรองรับปัญหาและความต้องการที่ไม่มีความแน่นอน

จากที่กล่าวมาข้างต้นเป็นการเข้าใจปัญหาของระบบที่ยังไม่ถูกสร้างขึ้น แต่ในความเป็นจริง การพัฒนาระบบส่วนใหญ่มักจะเป็นการพัฒนาแบบที่มีอยู่ก่อนแล้ว ทำให้ทางเลือกสำหรับ นักพัฒนาจึงมีอยู่ 2 ลักษณะ คือ พัฒนาระบบโดยใช้พื้นฐานของระบบเดิม ซึ่งวิธีนี้หากระบบเดิมมี Infrastructure ที่ดี การพัฒนาต่อก็สามารถทำได้ไม่ยาก แต่หาก Infrastructure เดิมของระบบไม่ดี ความยากในการพัฒนาต่อก็จะมีมากขึ้น ทำให้นักพัฒนาต้องใช้ทางเลือกหนึ่งคือ สร้างระบบใหม่ ขึ้นมา ซึ่งจะเสียค่าใช้จ่ายสูง เนื่องจากต้องวิเคราะห์และวางโครงสร้างของระบบขึ้นมาใหม่ ส่งผลให้ก่อนเลือกหนึ่งในสองทางเลือกดังกล่าวนี้ นักพัฒนาจะต้องมีการศึกษาความเป็นไปได้เสียก่อน

ขั้นตอนที่ 2 ออกแบบ (Design)

หลังจากที่ได้วิเคราะห์และจำแนกความต้องการของระบบออกมาอย่างชัดเจนแล้ว ถึงเวลาที่ทีมพัฒนาจะต้องออกแบบระบบ ซึ่งขั้นตอนการออกแบบนี้จะครอบคลุมตั้งแต่การออกแบบการทำงานของระบบ เช่น ออกแบบความสัมพันธ์ของฐานข้อมูล ออกแบบ Work Flow ของระบบ ไปจนถึงการออกแบบ User Interface โดยคำนึงถึงความต้องการของระบบเป็นหลัก ผลลัพธ์ที่ได้จากขั้นตอนนี้อาจเป็นได้ทั้งแผนภาพ (Diagram) แสดงการไหลผ่านของข้อมูล หรือ Low Definition Prototype ก็ได้

ขั้นตอนที่ 3 สร้างหรือพัฒนาระบบ (Implementation)

ขั้นตอนนี้เป็นการนำผลลัพธ์ที่ได้จากขั้นตอนที่แล้ว มาพัฒนาและสร้างออกมาเป็นระบบที่ใช้งานได้จริง โดยการใช้เครื่องมือต่าง ๆ ทั้งในลักษณะของซอฟต์แวร์และฮาร์ดแวร์ ขั้นตอนการพัฒนา ความยากง่ายขึ้นอยู่กับรูปแบบในขั้นตอนก่อนหน้านี้ หากออกแบบ Infrastructure ได้ดี การ Implement ก็ง่าย แต่ถ้าออกแบบได้ไม่ดี การ Implement ก็เกิดปัญหาตามมา

ขั้นตอนที่ 4 ทดสอบ (Testing)

หลังจากที่นักพัฒนาได้สร้างระบบเสร็จสมบูรณ์แล้ว ระบบจะต้องถูกนำมาทดสอบ เพื่อให้แน่ใจว่าระบบจะสามารถทำงานได้อย่างสมบูรณ์ ไม่มีข้อผิดพลาด หากพบเจอข้อผิดพลาด หรือ Bug ระบบต้องถูกส่งกลับไปยังฝ่ายพัฒนาเพื่อแก้ไขให้เรียบร้อย และนำกลับมาทดสอบต่อ

ขั้นตอนนี้เป็นขั้นตอนที่ใช้เวลานานที่สุด เนื่องจากยิ่งระบบมีความซับซ้อน ความเสี่ยงที่จะเกิด Bug ก็จะสูงขึ้นเช่นกัน ดังนั้นฝ่ายทดสอบจะต้องทำความเข้าใจกับระบบ และจำเตรียมชุดทดสอบที่มีความเป็นไปได้อย่างละเอียด เพื่อให้การทดสอบมีความแม่นยำมากยิ่งขึ้น

ขั้นตอนที่ 5 บำรุงรักษาและประเมินผล (Maintenance and Evaluation)

หลังจากที่ส่งมอบระบบให้กับลูกค้าแล้ว ทีมผู้พัฒนาจะต้องมีการติดตามผลหลังจากใช้งาน หากผู้ใช้พบเจอปัญหา จะต้องทำการแก้ไขระบบให้เรียบร้อย อาจทำแบบประเมินขึ้นมาเพื่อติดตามผลปละข้อแสดงความคิดเห็นจากผู้ใช้ เพื่อนำไปปรับปรุงแก้ไข ทั้งนี้ทีมพัฒนาควรจะต้องทำเอกสารสรุปงาน เพื่อใช้เป็นแหล่งอ้างอิงในอนาคต

2.2 เทคโนโลยีที่ใช้ในการปฏิบัติงาน

2.2.1 ionic framework

ionic framework [2] เป็นเครื่องมือที่ใช้ในการพัฒนา Mobile Application แบบ Hybrid หรือ "Hybrid App หรือ Hybrid Mobile App" คือเราสามารถพัฒนาแอปพลิเคชันครั้งเดียวแล้วรันได้หลาย Platform เช่น Android iOS ที่ได้รับความนิยมในปัจจุบัน ตัว ionic framework เองจะใช้เทคโนโลยีในการพัฒนาคือ HTML5 CSS3 และ JAVA Script ตัวการพัฒนาหลัก ๆ เลยที่จะต้องใช้คือ JAVA Script เพราะว่า ionic framework ใช้ Core หลักเป็น Angular JS

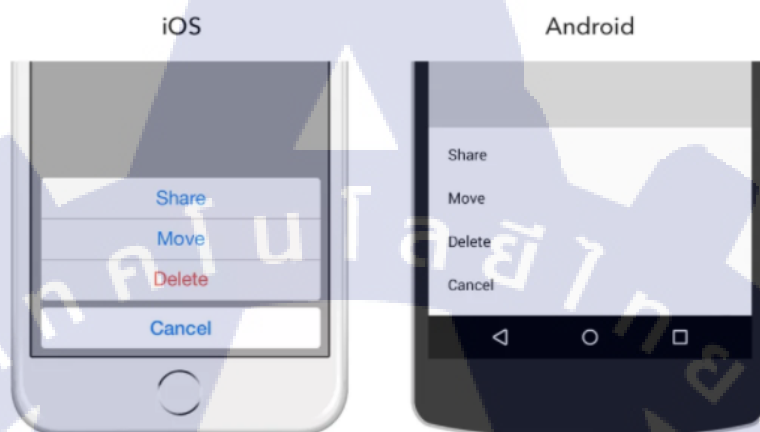
ข้อดีที่ของ ionic framework ที่เห็นได้ชัดคือมีความใกล้เคียงกับ Native App มาก ทั้งหน้าตา UI การใช้งาน แต่ก็ไม่ถึงกับเหมือน Native App เลยสักทีเดียว และยังสามารถพัฒนาให้ติดต่อกับ Hardware ของอุปกรณ์ได้ด้วย เช่น กล้อง ไมโครโฟน การตรวจสอบการเชื่อมต่ออินเทอร์เน็ต เป็นต้น โดยใช้งานร่วมกับ PhoneGap/Cordova

นอกจากนั้นแล้วยังสามารถทดสอบแอปพลิเคชันที่พัฒนาขึ้นผ่านอุปกรณ์จริงได้เลย โดยการทดสอบจาก Application ที่ชื่อว่า ionicview ที่สามารถรองรับทั้ง Android และ iOS โดยใช้คำสั่ง command line ไม่กี่บรรทัดก็สามารถอัปโหลดแล้วรันได้เลย



ภาพที่ 2.5 สัญลักษณ์ของ ionic framework

สิ่งที่เป็นข้อดีของ ionic framework อีกอย่างหนึ่งคือการมี Platform Continuity โดย ionic framework เปิดโอกาสให้ปรับแต่ง Widget และ UI ต่าง ๆ ได้อย่างอิสระ เช่น Font, Action sheet, Nav bar, Back button และ check boxes โดยจะมีค่าเริ่มต้นสำหรับ iOS และ Android มาให้ก่อนอยู่แล้ว ยกตัวอย่าง Action sheet ของทั้ง 2 ระบบปฏิบัติการดังภาพ



ภาพที่ 2.6 ภาพเปรียบเทียบความแตกต่าง Platform ของระบบปฏิบัติการ iOS และ Android

2.2.2 Node.js

node.js [11] คือ Programming language ที่ใช้โครงสร้างภาษา JavaScript ในการเขียน และมีการรันด้วย Chrome's V8 JavaScript engine โดยตอนเริ่มต้น V8 มาจากการที่google พัฒนาไว้แล้วให้ใช้กันอย่างเสรี หลักการทำงานคือจะรัน JavaScript ได้อย่างรวดเร็ว หลังจากนั้นเลยมีคนนำมาพัฒนาต่อเป็น server interpreter



ภาพที่ 2.7 สัญลักษณ์ของ node.js

node.js มี ส่วนเสริม ที่ชื่อว่า node package management [12] หรือที่เรียกกันว่า npm เป็นเหมือน google play ใน android หรือ app store ใน iOS นั่นเอง ที่เราสามารถเอา package ที่คนอื่นเขียนเอาไว้ มาต่อยอดเพื่อใช้ในงานของเราได้ โดยตัวอย่างที่ได้รับความนิยมมาก ๆ ก็จะเป็น underscore, async, request, express เป็นต้น อย่างเช่น ตัว underscore หากเราเอามาใช้ เราก็จะสามารถใช้งาน function เช่น `_.each([1, 2, 3], alert);` ได้เลย หรือ `_.filter(list, iterator, [context])` ได้เลย หรือ express ก็เหมือนเป็น web framework ตัวนี้เลยติดตั้งตัวนี้แล้วตั้งค่านิดหน่อย ก็รันเว็บได้โดยไม่ต้องมี Apache เลย ซึ่ง package เหล่านี้ช่วยให้เราไม่ต้องมาเขียน function พื้นฐานเหล่านี้ด้วยตัวเราเอง ปัจจุบัน มีให้เราได้เลือกใช้เยอะมาก (ณ วันที่เขียน มี 37,784 ตัวแล้ว) จริง ๆ เราก็ใช้เท่าที่งานเราจำเป็นนั่นแหละครับ (เท่าที่ผมเขียนมาก็ใช้ 1-4 ตัว ต่องานประมาณนี้) การติดตั้ง ก็แสนง่าย แค่สั่ง `npm install underscore` เท่านั้นเอง ก็จะได้ package underscore มาใช้เลย



ภาพที่ 2.8 สัญลักษณ์ของ npm

2.2.3 Android Studio

Android Studio [13] เป็นเครื่องมือพัฒนา IDE (Integrated Development Environment) ที่ถูกสร้างขึ้นมาเพื่อการพัฒนาแอนดรอยด์แอปพลิเคชัน บนพื้นฐานของแนวคิด IntelliJ IDEA คล้าย ๆ กับการทำงานของ Eclipse และ Android ADT Plugin และเป็น IDE Tools ล่าสุดจาก Google ไว้พัฒนาโปรแกรม Android เป็นระบบปฏิบัติการของอุปกรณ์เคลื่อนที่ ที่เห็นกันชัด ๆ คือ โทรศัพท์ แท็บเล็ต เป็นต้น

การเขียน Android บน Android Studio จะมีขั้นตอนอยู่ 2 ขั้นตอนก็คือ ติดตั้ง Java SDK และดาวน์โหลด Android Studio มาติดตั้งก็จะสามารถใช้งานได้ทันที ที่ละยั้งรวมไปถึงตัว Emulator อย่างเช่น Genymotion ที่เราต้องโหลดมาติดตั้งเพื่อช่วยในการทดสอบโปรเจก



ภาพที่ 2.9 สัญลักษณ์ของโปรแกรม Android Studio

ในส่วนของ SDK [14] ที่ย่อมาจาก Software Development Kit คือ เครื่องมือที่เอาไว้สำหรับพัฒนาโปรแกรมหรือแอปพลิเคชันบนระบบ Android OS ซึ่งทาง Google พัฒนาออกมาเพื่อแจกจ่ายให้นักพัฒนาแอปพลิเคชัน หรือผู้สนใจทั่วไปดาวน์โหลดไปใช้กันโดยไม่มีค่าใช้จ่าย และนี่ก็เป็นหนึ่งในปัจจัยที่ทำให้แอปพลิเคชันบนแอนดรอยด์นั้นเพิ่มขึ้น อย่างรวดเร็ว ซึ่งในชุด SDK นั้นจะมีโปรแกรมและไลบรารีต่างๆ ที่จำเป็นต่อการพัฒนาแอปพลิเคชันบนแอนดรอยด์ อย่างเช่น Emulator ซึ่งทำให้ผู้ใช้สามารถสร้างแอปพลิเคชันและนำมาทดลองรันบนตัวอิมูเลเตอร์ก่อน โดยมีสถานะแวดล้อมเหมือนมือถือที่รันระบบปฏิบัติการแอนดรอยด์จริง ๆ

ส่วน Genymotion [15] เป็น Android Emulator ที่มาพร้อมกับ Android System Image และ AVD สำเร็จรูปมากกว่า 10 ตัว ช่วยให้นักพัฒนาไม่ต้องกังวลเรื่องการตั้งค่า AVD อีกต่อไป ที่สำคัญ Genymotion ทำงานเร็วกว่า Emulator ที่มากับ Android SDK มาก ๆ Genymotion มีเครื่องมือช่วยนักพัฒนาหลายอย่างทั้งการเชื่อมต่อกับ Webcam, GPS, Screencast และอีกมากมาย Genymotion สามารถดาวน์โหลดมาใช้งานได้ฟรี หรือเสียเงินเพื่อใช้งาน feature เสริมอื่น ๆ

2.2.4 Visual Studio Code

Visual Studio Code [17] คือ โปรแกรมตัวหนึ่งที่เป็นเครื่องมือที่ช่วยพัฒนาซอฟต์แวร์และระบบต่าง ๆ ซึ่งสามารถติดต่อสื่อสารพูดคุยกับคอมพิวเตอร์ได้ในระดับหนึ่งแล้ว แต่ยังไม่สามารถพัฒนาเป็นระบบเองได้ เหมาะสมสำหรับภาษา VB และ VB.NET เนื่องจากไมโครซอฟต์ได้พัฒนาโปรแกรมและภาษาขึ้นมาควบคู่กันเพื่อให้ใช้งานได้ซึ่งกันและกัน ซึ่งนักโปรแกรมเมอร์จะนำเครื่องมือมาใช้ในการพัฒนาต่อยอดให้เกิดเป็นระบบต่าง ๆ หรือเป็นเว็บไซต์ และแอปพลิเคชันต่าง ๆ

ดังนั้น วิศวลศตุดิโอิจึงเป็นโปรแกรมตัวหนึ่งที่เป็นเครื่องมือที่ช่วยพัฒนาซอฟต์แวร์และระบบต่าง ๆ ซึ่งโปรแกรมได้มีการติดต่อดสื่อสารกับคอมพิวเตอร์ในระดับหนึ่งแล้ว แต่ไม่สามารถพัฒนาเป็นระบบได้ด้วยตนเอง นักพัฒนาจะนำเครื่องมือของโปรแกรมมาใช้พัฒนาต่อให้เกิดเป็นซอฟต์แวร์หรือระบบต่าง ๆ เพื่อช่วยอำนวยความสะดวก และลดเวลาการทำงานและข้อผิดพลาดได้เป็นอย่างมาก



ภาพที่ 2.10 สัญลักษณ์ของ Visual Studio

2.2.5 Xcode

Xcode [16] เป็นเครื่องมือของนักพัฒนา Apple เอาไว้สร้างแอปพลิเคชันสำหรับเครื่อง Mac, iPhone และ iPad ถ้าเทียบกับเครื่องมือพัฒนาของฝั่งไมโครซอฟท์ก็คือ Visual Studio นั่นเอง Xcode เวอร์ชันล่าสุด ณ ที่เขียนบทความนี้ คือเวอร์ชัน 4 ต้องติดตั้งลงบนเครื่องที่มี OS เป็น Mac OS X เวอร์ชัน 10.6 ขึ้นไป สามารถหาซื้อได้จาก Mac App Store ในราคา \$4.99 เหรียญสหรัฐ แต่ถ้าคุณสมัครเป็นสมาชิกของ iOS หรือ Mac Developer Program เสียค่าสมาชิกปีละ \$99 เหรียญสหรัฐ สามารถโหลดได้ฟรี Xcode 4 ประกอบด้วย Xcode IDE, Instruments และ SDK สำหรับ Mac OS X และ iOS

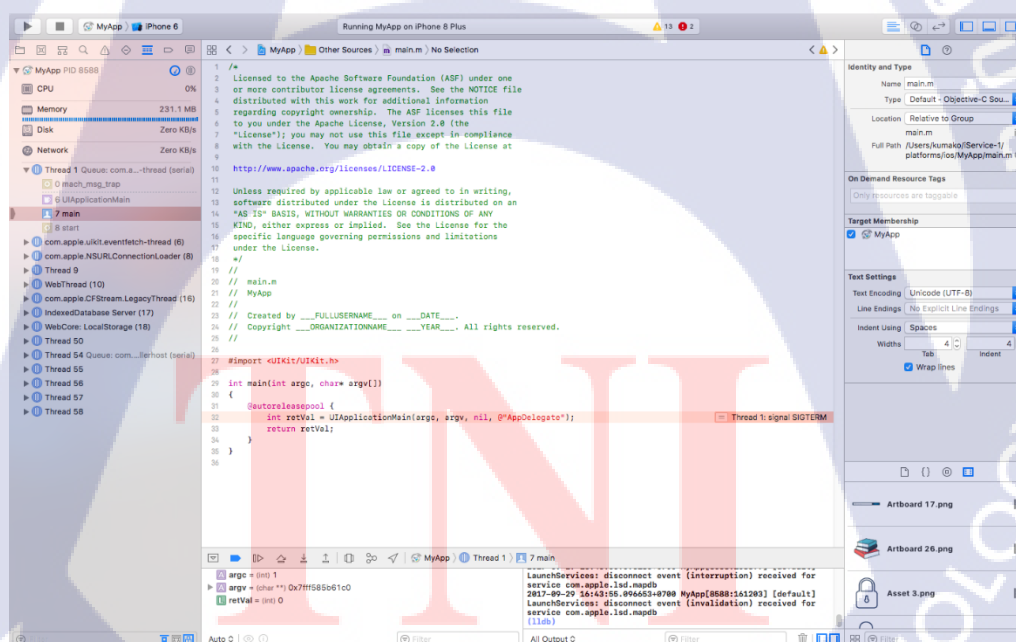


ภาพที่ 2.11 สัญลักษณ์ของโปรแกรม Xcode

Xcode IDE คือ สภาพแวดล้อมในการพัฒนาแอปพลิเคชัน ประกอบด้วยพื้นที่ทำงานสำหรับเขียน code , พื้นที่สำหรับออกแบบหน้าจอ user interface เรียกว่า Interface builder, มีคอมไพเลอร์ที่มีประสิทธิภาพสูง ชื่อว่า LLVM Compiler 2.0 สามารถคอมไพล์โค้ดได้เร็วกว่า GCC ถึง 2 เท่า สร้างแอปพลิเคชันให้ทำงานได้เร็วขึ้น มีระบบตรวจสอบโค้ดที่มีประสิทธิภาพ แก้ไขโค้ดที่ผิดให้อัตโนมัติ, มี Version Editor ที่ตรวจสอบ Source Code ทั้ง 2 เวอร์ชัน แบบเทียบกันหน้าต่อหน้าให้เห็นบรรทัดที่แตกต่างกันได้อย่างมีประสิทธิภาพ, มี Debugger Engine ที่ชื่อว่า LLDB สามารถ track code ได้ขณะที่โปรแกรมกำลัง run ทดสอบอยู่, มี Source Control ในการควบคุมเวอร์ชันของ Project ได้

Instruments เป็นเครื่องมือที่ใช้ตรวจสอบประสิทธิภาพของแอปพลิเคชันที่พัฒนาเสร็จแล้ว โดยวัดการใช้งาน memory, CPU Time, Overhead ต่าง ๆ

SDK สำหรับ Mac OS X และ iOS คือ Class Library สำหรับใช้พัฒนาแอปพลิเคชันแบ่งเป็น Mac OS X และ iOS



ภาพที่ 2.12 หน้าตาของ XcodeIDE

บทที่ 3

แผนงานการปฏิบัติงานและขั้นตอนการดำเนินงาน

3.1 แผนงานการปฏิบัติงาน

รายละเอียด	มิถุนายน				กรกฎาคม				สิงหาคม				กันยายน			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
ศึกษาความรู้ฝั่ง Frontend Develop, Backend Develop																
ศึกษาโปรแกรม React Native																
ศึกษาโปรแกรม ionic framework																
ประชุม, รวบรวม Requirement																
จัดทำ wireframe นำเสนอในที่ประชุม แก่ไข																
เขียนโปรแกรมหน้าพีทเจอร์รี่ที่ได้รับมอบหมาย																
ใส่ฟังก์ชันเชื่อมหลังบ้าน และแก้ไขหน้าแอปพลิเคชันให้สมบูรณ์																
รวบรวม เทสต์และปรับปรุงให้สมบูรณ์																

ตาราง 3.1 แสดงแผนการปฏิบัติงานสหกิจศึกษา

3.2 รายละเอียดโครงการ

3.2.1 แอปพลิเคชัน iService

จัดทำแอปพลิเคชัน iService เวอร์ชัน 1.0.0 สำหรับระบบปฏิบัติการ Android และ iOS เพื่อช่วยเหลือลูกค้าที่ใช้งานระบบ iGet POS ของทางบริษัท โดยสร้างฟีเจอร์ทั้งหมด 3 ฟีเจอร์มีรายละเอียดดังนี้

3.2.1.1 หน้าแจ้งปัญหาของลูกค้า

ฟีเจอร์นี้ทำขึ้นเพื่อให้ลูกค้าแจ้งปัญหาลงในหน้าเพื่อส่งไปให้ฝ่ายซัพพอร์ตโดยตรงได้ และสามารถเลือกหัวข้อใหญ่ หัวข้อย่อยของปัญหา และอัปโหลดรูปภาพได้

3.2.1.2 หน้าปัญหาที่พบบ่อย

ฟีเจอร์นี้จะทำการรวบรวมปัญหาที่พบบ่อยที่ทำการเก็บรวบรวมมาจากฝ่ายซัพพอร์ตให้ลูกค้าสามารถค้นหาและแก้ไขปัญหาลำพังด้วยตนเอง สร้างความสะดวกให้ลูกค้าโดยการรวบรวมเป็นหัวข้อต่าง ๆ สามารถทำการค้นหาด้วยเวิร์ดได้จากช่องค้นหา

3.2.1.3 หน้าตั้งค่า

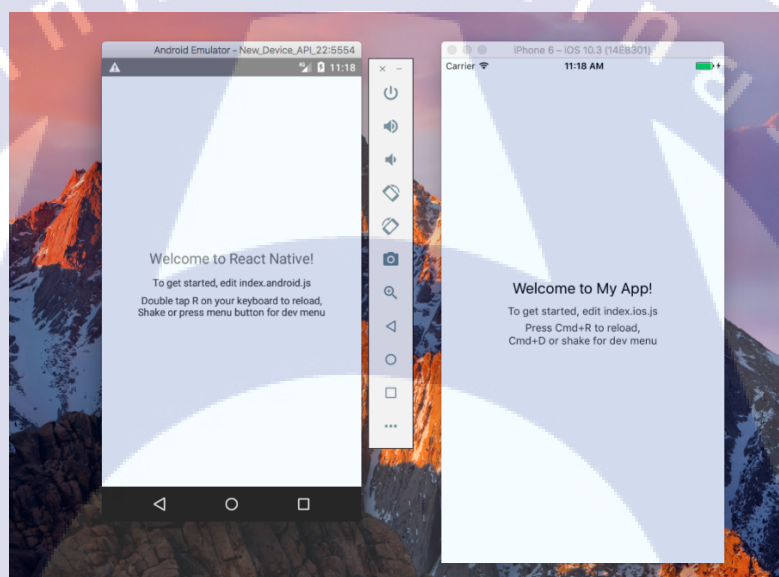
ฟีเจอร์นี้จะแสดงชื่อ ตำแหน่ง เบอร์โทรศัพท์ สาขาของร้านค้าของลูกค้าที่ทำการเข้าระบบมาใช้งาน และมีปุ่มแสดงเวอร์ชันของแอปพลิเคชัน อีกทั้งถ้าต้องการออกจากระบบสามารถออกได้ในหน้านี้

3.2.2 งานอื่น ๆ

ทดลองเขียนแอปพลิเคชันโดยใช้โปรแกรม React Native เนื่องจากในตอนแรกทางบริษัทอยากให้ลองใช้อะไรใหม่ ๆ ในการเขียนแอปพลิเคชันครั้งนี้ **ตอนเริ่มต้นจึงเริ่มจาก React Native** ข้อดีคือรองรับทั้งระบบปฏิบัติการ iOS และ Android เขียนครั้งเดียวสามารถรันได้ทั้งสองอย่าง และรันโปรแกรมได้รวดเร็ว แต่ติดปัญหาอยู่หลายอย่างอาทิเช่น พนักงานในบริษัทก็ยังไม่เคยเขียนแอปพลิเคชันโดย React Native เหมือนกัน เมื่อติดปัญหาจึงไม่มีใครสามารถช่วยแก้ไขได้ และ React Native ยังเป็นโปรแกรมที่ใหม่อยู่ บางปัญหาที่เจอลองหาคำตอบด้วยตนเองแล้วแต่ยังไม่สามารถแก้ปัญหาได้ จึงเปลี่ยนมาทำ ionic framework



ภาพที่ 3.1 สัญลักษณ์ของโปรแกรม React Native



ภาพที่ 3.2 ความแตกต่างของการแสดงผลระหว่างของระบบปฏิบัติการ Android (ซ้าย) และระบบปฏิบัติการ iOS ขวา

3.3 ขั้นตอนการดำเนินงานที่นักศึกษาปฏิบัติหรือโครงการ

3.3.1 รวบรวม requirement

เมื่อได้หัวข้อและรู้จุดประสงค์ของการสร้างแอปพลิเคชันนี้แล้ว จากนั้นก็ทำการรวบรวมความต้องการ (requirement) จากพี่ ๆ พนักงานฝ่ายซัพพอร์ตว่าลูกค้าต้องการอะไรเป็นหลัก ปัญหาหลัก ๆ ที่ลูกค้าต้องพบเจอคืออะไร และแนวทางแก้ไขปัญหาคืออะไร เราได้ทำการรวบรวมและสรุปผลกันภายในทีม

3.3.2 ขอบเขตของงาน

ทุกคนในทีมช่วยกันร่างแบบและเขียนฟังก์ชันคร่าว ๆ ว่าแอปพลิเคชันนี้ต้องมีอะไรบ้าง จากนั้นจึงทำการมอบหมายงานให้เพื่อนในทีมทำการออกแบบแอปพลิเคชัน แล้วค่อยมาถามความเห็นคนในทีมว่ามีความเห็นตรงกันหรือไม่ แก้ไขไปเรื่อย ๆ จนกว่าจะเป็นที่น่าพอใจ และกำหนดขอบเขตการทำงาน รวมไปถึงระบบฐานข้อมูลด้วย แอปพลิเคชันนี้สามารถเข้าระบบได้ 2 ฝั่งได้แก่ ฝั่งพนักงานฝ่ายซัพพอร์ต และฝั่งลูกค้า โดยมีหน้าฟีเจอร์ดังนี้

หน้าของฝั่งพนักงานฝ่ายซัพพอร์ต

- 1) หน้าโฮมของพนักงาน
- 2) หน้าประวัติที่ลูกค้าแจ้ง
- 3) หน้าหัวข้อการแจ้งปัญหา
- 4) หน้าแก้ไข FAQ
- 5) หน้าสร้างปัญหาใหม่

หน้าของฝั่งลูกค้า

- 1) หน้าโฮมของลูกค้า
- 2) หน้าแจ้งปัญหา
- 3) หน้าประวัติย้อนหลัง
- 4) หน้าปัญหาที่พบบ่อย
- 5) หน้าตั้งค่า

3.3.3 นำเสนอในที่ประชุม

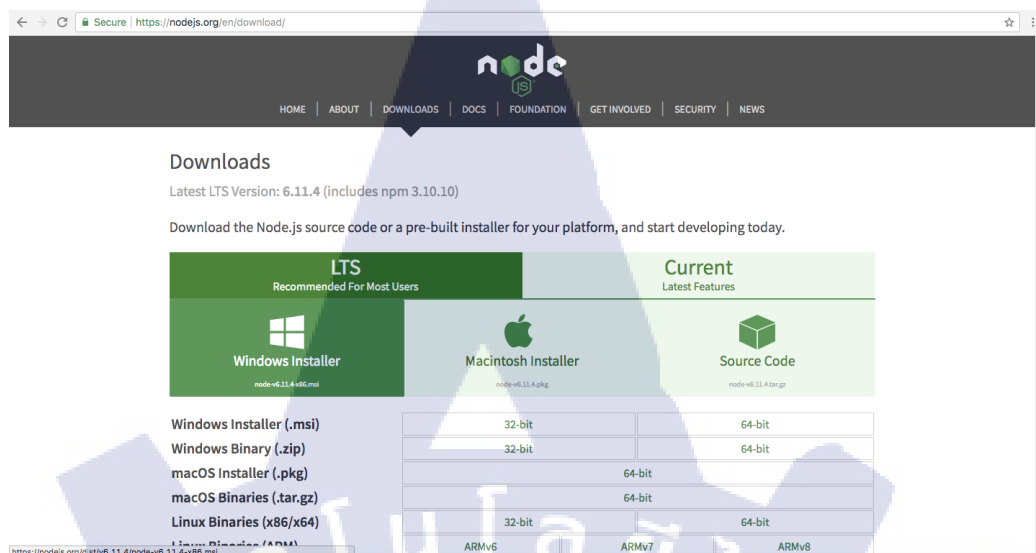
เมื่อได้ผลสรุปออกมาแล้ว จึงได้นำหน้าตาของแอปพลิเคชันที่ออกแบบไว้ไปนำเสนอในที่ประชุมกับพี่พนักงานจากหลายแผนกกว่ายังต้องการให้ปรับเปลี่ยน หรืออยากให้ใส่ฟังก์ชันอะไรเพิ่มอีกหรือไม่

3.3.4 พัฒนาแอปพลิเคชัน

เริ่มจากการติดตั้งที่ต้องใช้ก่อนเป็นอันดับแรกได้แก่

- 1) ติดตั้งโปรแกรม Node.js

เพื่อใช้ในการรันคอมมานต่าง ๆ โดยเข้าไปที่เว็บไซต์และเลือกโหลดตามสเปคของเครื่อง



ภาพที่ 3.3 แสดงหน้าต่างของการดาวน์โหลดโปรแกรม Node.js

- 2) ติดตั้งและศึกษาระบบการทำงานของโปรแกรม ionic framework
ขั้นแรกก็ทำการติดตั้งโปรแกรม ionic framework ผ่านคำสั่งคอมมานไลน์ใน terminal
ก่อนเป็นอันดับแรก

1 Install Ionic

```
npm install -g cordova ionic
```

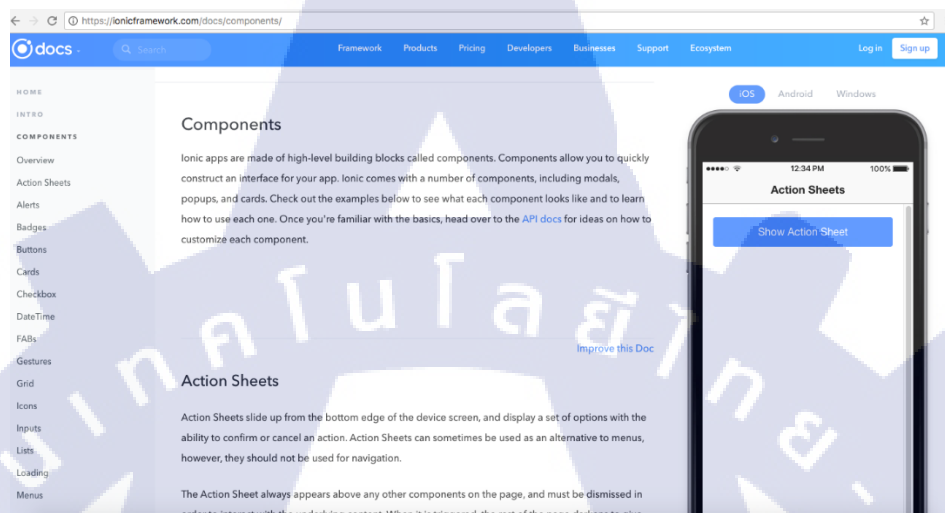
ภาพที่ 3.4 คำสั่งในการติดตั้ง ionic framework ผ่านคำสั่งคอมมานไลน์

2 Start an App

```
ionic start myApp tabs
```

ภาพที่ 3.5 คำสั่งในการติดตั้ง ionic framework ผ่านคำสั่งคอมมานไลน์ (ต่อ)

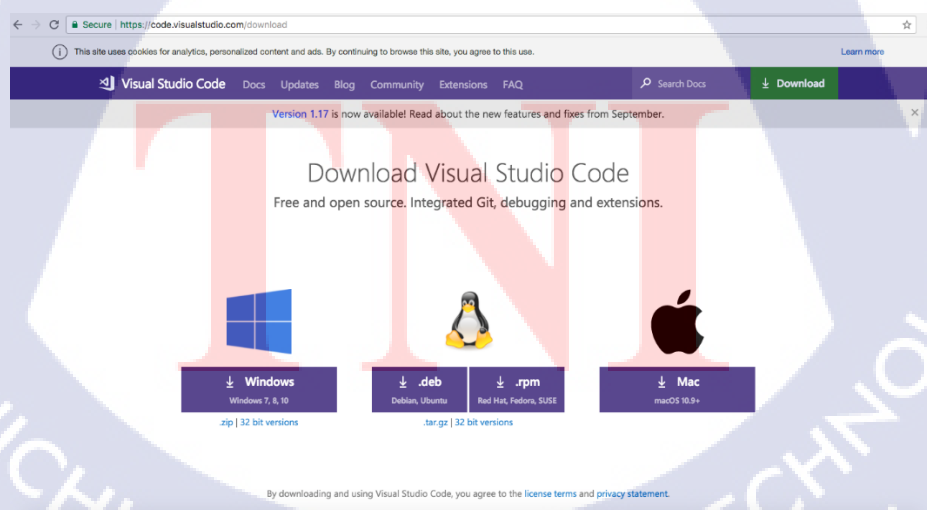
เมื่อติดตั้งเรียบร้อยแล้วขั้นตอนต่อไปก็ทำการศึกษาเบื้องต้นทั้งวิธีการใช้งานและฟังก์ชันต่าง ๆ ผ่าน Doc ของ ionic framework



ภาพที่ 3.6 แสดงหน้าต่าง Document ของ ionic framework

3) ติดตั้งโปรแกรม Visual Studio Code

เมื่อศึกษาการทำงาน ionic framework คร่าว ๆ แล้วก็มาทำการติดตั้งโปรแกรม Visual Studio Code ที่ใช้ในการเขียนแอปพลิเคชัน

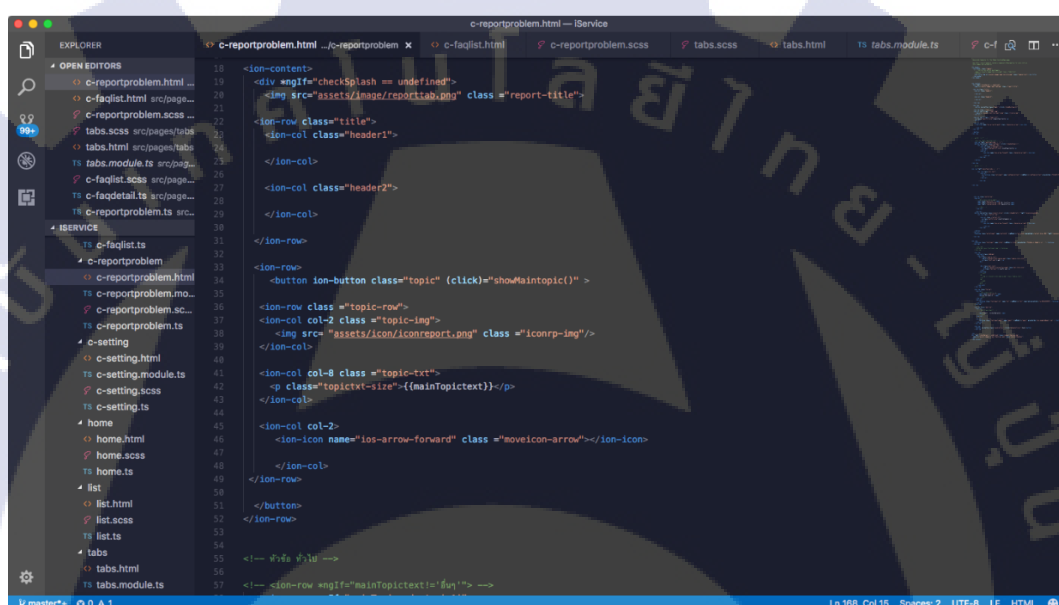


ภาพที่ 3.7 แสดงหน้าต่างการขอการดาวน์โหลดโปรแกรม Visual Studio Code

3.3.5 ทำการเขียนหน้าฟิเจอร์ที่ได้รับมอบหมาย

หน้าฟิเจอร์ที่ได้รับมอบหมายมี 3 ส่วนทำทีละขั้นตอนเริ่มจากการทำ UI Components และใส่ฟังก์ชันให้ครบ โดยพัฒนาแอปพลิเคชันด้วย ionic framework โดยเขียน 3 ฟิเจอร์หลักดังนี้

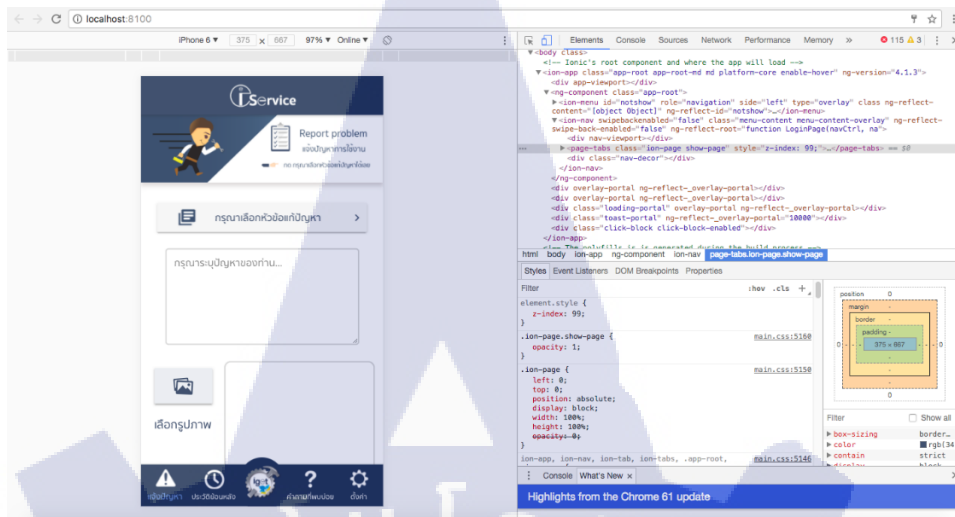
- 1) หน้าแจ้งปัญหาของลูกค้า
- 2) หน้าคำถามที่พบบ่อย
- 3) หน้าตั้งค่า



ภาพที่ 3.8 ตัวอย่างหน้าต่างของ Visual Studio Code

3.3.6 ทดสอบและแก้ไขแอปพลิเคชัน

ทำการทดสอบโดยรันผ่านเซิร์ฟเวอร์ด้วยคำสั่ง ionic serve ใน terminal คอยทดสอบและแก้ไข CSS เรื่อย ๆ จนหน้าต่างออกมาตามที่ดีไซน์ไว้



ภาพที่ 3.9 หน้าต่างของการรันโปรแกรม ionic framework ผ่านเชิร์ฟเวอร์

จากนั้นจึงส่งไปให้เพื่อนในทีมรวมไฟล์ทั้งหมด เมื่อรวมจนเสร็จสิ้นค่อยให้เพื่อนในทีมส่งกลับมาเพื่อทดสอบในซิมูเลเตอร์ (Simulator) ของตัว Xcode ในระบบปฏิบัติการ iOS



ภาพที่ 3.10 ภาพ User Interface ของโปรแกรม Simulator

กรณีเลือกให้แสดง simulator สำหรับ iphone 6

บทที่ 4

ผลการดำเนินงาน การวิเคราะห์และสรุปผล

4.1 ผลการดำเนินงาน

จากดำเนินโครงการทั้งการพัฒนาและแก้ไขแอปพลิเคชัน iService ในส่วนของ UI Components และทดสอบในระบบปฏิบัติการ iOS หลังจากเสร็จสิ้นขั้นตอนการพัฒนา ได้มีการทดสอบและแก้ไขจุดบกพร่องที่พบในแอปพลิเคชันเพื่อให้เข้าใกล้แอปพลิเคชันที่มีความสมบูรณ์แบบมากที่สุด

จากการได้รับมอบหมายงานในโครงการซึ่งได้ผลดังนี้

4.1.1 ผลการดำเนินงานของหน้าแอปพลิเคชัน

หน้าแอปพลิเคชันที่ดำเนินงานเสร็จแล้วมีดังนี้

1) หน้าแจ้งปัญหา



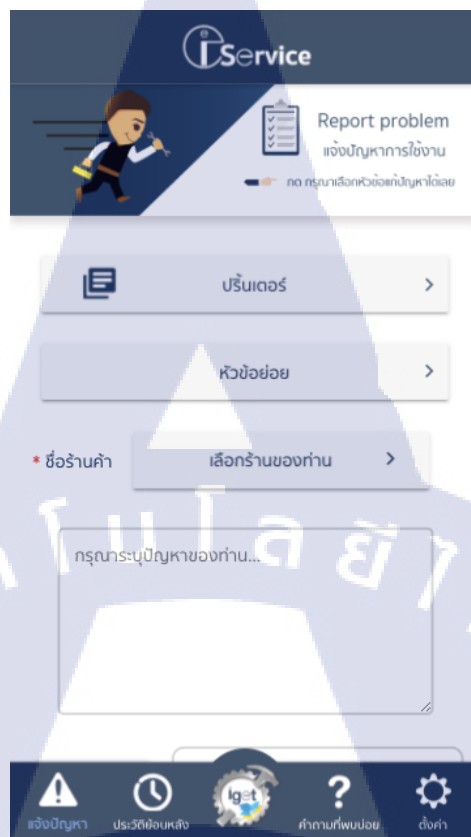
ภาพที่ 4.1 ภาพรวมของหน้าแจ้งปัญหา

ซึ่งในหน้านี้ฟังก์ชันข้างบนจะเป็นส่วนที่ลูกค้าสามารถแจ้งปัญหาได้จากการกรอกข้อมูลก่อนอื่นให้คลิกตรงปุ่มกรุณาเลือกหัวข้อปัญหา



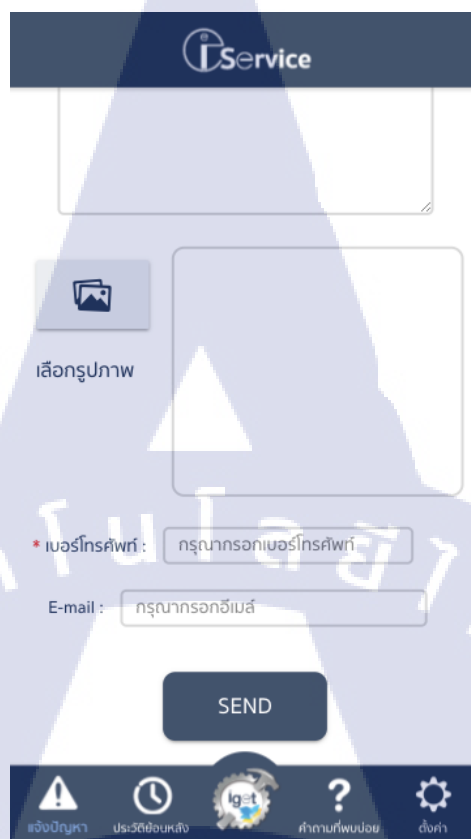
ภาพที่ 4.2 หน้าแจ้งปัญหาส่วนเลือกหัวข้อปัญหา

ตรงนี้จะเป็นตัว Component ของ ionic framework โผล่มาให้เราเลือกหัวข้อก่อนเมื่อเลือกหัวข้อไปแล้ว หัวข้อย่อยถึงจะปรากฏขึ้นมา



ภาพที่ 4.3 หน้าแจ้งปัญหาส่วนหัวข้อย่อยและเลือกร้านค้า

เมื่อหัวข้อย่อยปรากฏออกมาแล้วจะมี Component แบบเดิมขึ้นมาให้ทำการเลือกหัวข้อย่อย
ต่อด้วยเลือกร้านค้า และสามารถพิมพ์ข้อความได้ในหน้านี้ ระบุปัญหาที่เกิดขึ้น



I-Service

เลือกรูปภาพ

* เบอร์โทรศัพท์ : กรุณากรอกเบอร์โทรศัพท์

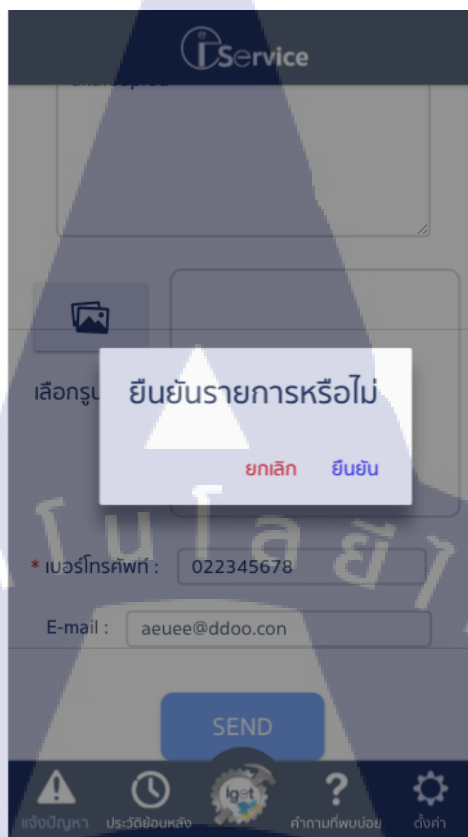
E-mail : กรุณากรอกอีเมล

SEND

แจ้งปัญหา ประวัติข้อร้องเรียน คำถามที่พบบ่อย ตั้งค่า

ภาพที่ 4.4 หน้าแจ้งปัญหาส่วนเบอร์โทรศัพท์และอีเมล

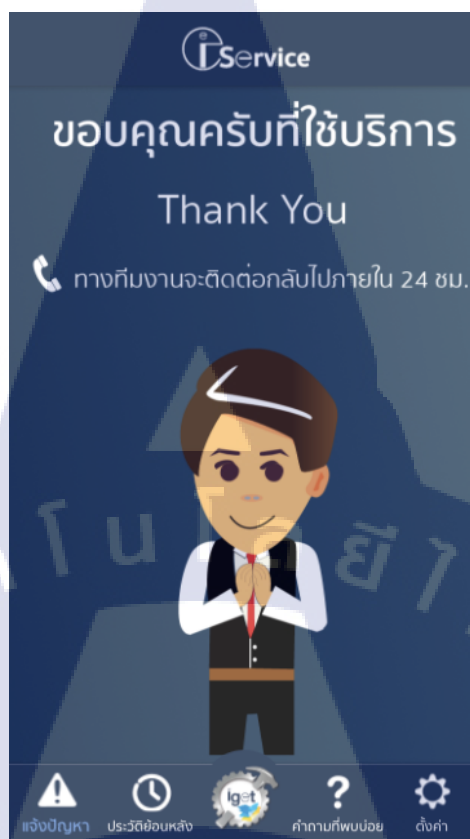
สามารถเลือกรูปภาพได้ หรือถ้าไม่มีก็ข้ามไป ตามด้วยกรอกเบอร์โทรศัพท์และอีเมลให้ถูกต้อง ตรงนี้ได้ทำการใส่ฟังก์ชันการทำงานไว้เรียบร้อยแล้วต้องกรอกให้ถูกต้อง หากลืมกรอกข้อมูลส่วนใดส่วนหนึ่งหรือมีข้อผิดพลาด ระบบจะให้ทำการกรอกใหม่จนกว่าจะสมบูรณ์



ภาพที่ 4.5 หน้าแจ้งปัญหาส่วนยืนยันรายการ

ก่อนจะส่งข้อมูลจะมี Prop ให้ยืนยันการทำรายการอีกครั้ง ถ้าจะแก้ไขข้อมูลหรือใส่ข้อมูลเพิ่มเพิ่มเติมให้กดยกเลิกกลับไปเพิ่มข้อมูลได้ ถ้าต้องการส่งก็กดยืนยัน

TNI



ภาพที่ 4.6 หน้าแจ้งปัญหาส่วนหน้า Splash

หลังจากทำการส่งข้อมูลระบบก็จะขึ้นมาที่หน้า Splash ขอบคุณที่ใช้บริการจะค้างอยู่ประมาณ 5 วินาทีก่อนแดงกลับไปหน้าจอโฮม

TNI

2) หน้าคำถามที่พบบ่อย



ภาพที่ 4.7 หน้าคำถามที่พบบ่อย

หน้านี้จะแสดงหัวข้อของปัญหาที่พบบ่อยมาให้ผู้ใช้งานสามารถเลือกเข้าไปดูด้วยตนเองได้โดยที่ไม่ต้องรอฝ่ายซัพพอร์ตตอบกลับ สามารถกดปุ่มเลือกดูตามหัวข้อได้



ภาพที่ 4.8 หน้าคำถามที่พบบ่อยแสดงหัวข้อย่อ

เมื่อกดที่ปุ่มหัวข้อสีเทา หัวข้อย่อของปัญหานั้น ๆ ก็จะโผล่มาเราสามารถเลือกดูปัญหาที่ต้องการจะเข้าไปอ่านรายละเอียดต่อได้



ภาพที่ 4.9 หน้าคำถามที่พบบ่อยแสดงรายละเอียดของปัญหา

เมื่อกดปุ่มหัวข้อแล้วรายละเอียดวิธีการแก้ปัญหาก็จะขึ้นมาทั้งขั้นตอนและรูปภาพ

TNI

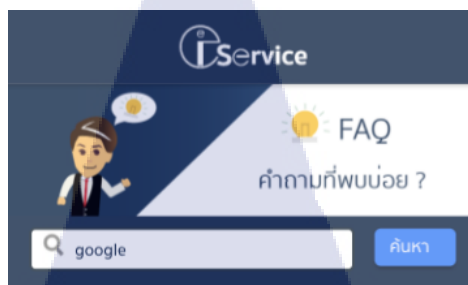
THAI - NICHI INSTITUTE OF TECHNOLOGY



ภาพที่ 4.10 หน้าคำถามที่พบบ่อยแสดงผลการค้นหา

ตรงช่องค้นหาเราสามารถใส่คีย์เวิร์ดในการค้นหาคำได้ อย่างตัวอย่างในภาพนี้ เมื่อค้นหาแล้วหัวข้อย่อยของปัญหานั้นก็จะแสดงขึ้นมา

TNI



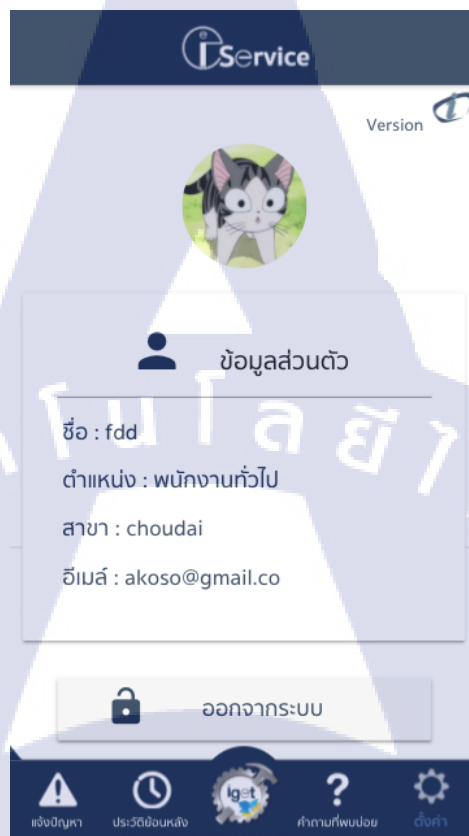
ไม่สามารถค้นหาในระบบ!



ภาพที่ 4.11 หน้าคำถามที่พบบ่อยเมื่อค้นหาในระบบไม่พบ

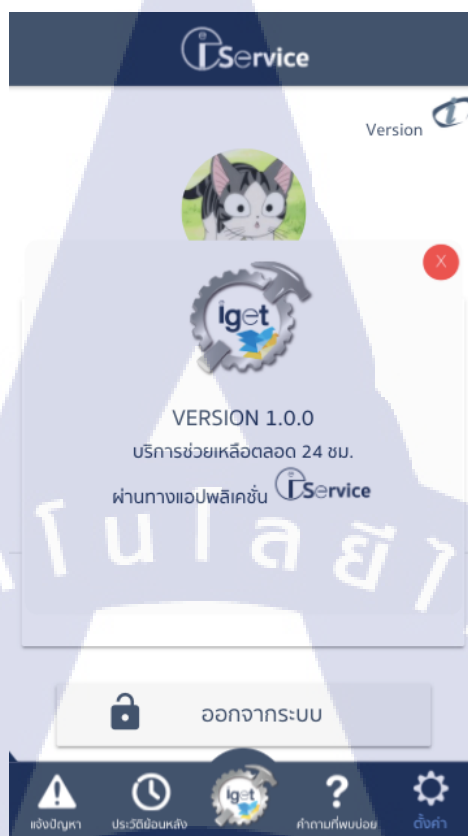
ถ้าคำค้นหาไม่ตรงกับคำที่มีในระบบจะขึ้นข้อความว่า ไม่สามารถค้นหาในระบบ

3) หน้าตั้งค่า



ภาพที่ 4.12 หน้าตั้งค่า

หน้าตั้งค่าจะแสดงข้อมูลส่วนตัวของลูกค้าที่เข้ามาดูโดยจะแสดงชื่อ ตำแหน่ง สาขา อีเมล
ตามลำดับและสามารถออกจากระบบได้ในหน้านี้



ภาพที่ 4.13 หน้าตั้งค่าส่วนแสดงเวอร์ชัน

ถ้ากดปุ่มคำว่าเวอร์ชันที่มุมขวบนจะมี Prop เด้งขึ้นมาตรวจสอบเวอร์ชันได้ เมื่อต้องการให้มันหายไปก็คลิกกากบาทสีแดงที่มุมขวบนของ Prop

4) แถบเมนูบาร์

แถบเมื่อนั้นต้องใช้คำสั่งของ ionic framework สร้างขึ้นมา แต่รูปแบบที่ได้ออกมาจะเป็นแบบเรียบง่าย ถ้าเราอยากได้แบบไหนต้องตกแต่งใน CSS เองเอง เลยทำการปรับแต่งขึ้นมาตามดีไซน์ที่เพื่อนในทีมออกแบบไว้ที่แรก



ภาพที่ 4.14 แถบเมนูบาร์

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

ตัวแอปพลิเคชันสามารถทำงานได้จริง สามารถเขียนฟีเจอร์ได้ตามที่รับมอบหมายแต่บางครั้งยังมีข้อผิดพลาดเกิดขึ้นเช่นกัน เนื่องจากก่อนหน้านี้ได้ทำการศึกษาโปรแกรม React Native เพื่อจะใช้พัฒนาแอปพลิเคชันนี้มาก่อนหน้าจะใช้ ionic framework ทำให้กินเวลามากจนเลยกำหนดแผนงานคร่าว ๆ ตอนแรก จึงไม่สามารถเขียนออกมาให้สมบูรณ์ครบถ้วนตามที่กำหนดไว้ยังขาดการทำงานบางส่วน

5.2 ข้อเสนอแนะจากการดำเนินงาน

- 1) ในอนาคตสามารถพัฒนาให้อยู่ในรูปแบบของเว็บไซต์ได้
- 2) เพิ่มฟังก์ชันต่อออกจากเดิมที่มีอยู่
- 3) หากต้องการพัฒนาต่อต้องมีความรู้เกี่ยวกับ ionic framework

5.3 ปัญหา และอุปสรรคที่เกิดขึ้น

- 1) ต้องศึกษาโปรแกรมที่ไม่เคยเรียนมาก่อน
- 2) Platform ของมือถือแต่ละรุ่น ไม่เหมือนกันจึงต้องปรับแต่งตามรูปแบบ
- 3) มีการเปลี่ยนแปลง requirement ระหว่างการทำงาน

5.4 แนวทางการแก้ไขปัญหา

- 1) ศึกษาด้วยตนเองเพิ่มเติมหรือปรึกษาพี่พนักงานเวลาที่ติดปัญหา
- 2) ศึกษาคำสั่งเพิ่มเติมน่าจะมีวิธีการแก้ไขสำหรับการเขียนหลายรูปแบบ
- 3) เสนอข้อเสนอแนะอื่น ๆ ให้ลูกค้าเผื่อจะถูกปรับเปลี่ยนน้อยลง

เอกสารอ้างอิง

1. Anonymous, **POS**[Online].Available:
http://www.chaophayacomputech.com/news/news_content_id=25570226-1.php
[2017,October 20].
2. Kittiphan Sornsakda, **Ionic Framework**[Online].Available: <http://ionicframework-docs.blogspot.com/> [2017,October 20].
3. Anonymous, **UX/UI**[Online].Available: <http://www.rushshop.com/article-preview/371391791792> [2017,October 20].
4. Anonymous, **Compiler**[Online].Available: <https://th.wikipedia.org/wiki/คอมไพเลอร์>
[2017,October 20].
5. Nawapong Areekul, **Terminal**[Online].Available:
<https://iphonemactool.blogspot.com/2014/01/terminal.html> [2017,October 20].
6. Anonymous, **Framework**[Online].Available: <http://www.siammodify.com/framework-คืออะไร.html> [2017,October 20].
7. Anonymous, **API**[Online].Available: <http://www.xn--12cg1cxchd0a2gzclc5d5a.net/api/>
[2017,October 20].
8. Anonymous, **Controller**[Online].Available: <https://www.arnondora.in.th/what-is-mvc/>
[2017,October 20].
9. Anonymous, **JavaScript**[Online].Available: <http://www.mindphp.com/คู่มือ/73-คืออะไร/2187-java-javascript-คืออะไร.html> [2017,October 20].
10. Anonymous, **CSS**[Online].Available: https://www.wynnsoft-solution.com/CSS_คืออะไร.html [2017,October 20].
11. Anonymous, **Node.js**[Online].Available: <https://beyourcyber.com/2016/node-js-is-programming-language-by-javascript/> [2017,October 20].
12. Anonymous, **npm**[Online].Available: <http://meewebfree.com/site/nodejs/441-learn-about-node-js-with-basic-of-node-js> [2017,October 20].
13. Anonymous, **Android Studio**[Online].Available: <http://www.mindphp.com/คู่มือ/73-คืออะไร/3505-android-studio.html> [2017,October 20].

เอกสารอ้างอิง (ต่อ)

14. Anonymous, **SDK**[Online].Available: <http://www.mindphp.com/คู่มือ/73-คืออะไร/2261-sdk-คืออะไร.html> [2017,October 20].
15. Anuchit Chalothorn, **Genymotion**[Online].Available: <http://thaiopensource.org/มาใช้-genymotion-android-emulator-กัน/> [2017,October 20].
16. Anonymous, **Xcode**[Online].Available: <http://takato.exteen.com/20110826/xcode> [2017,October 20].
17. Anonymous, **VS Code**[Online].Available: <http://www.mindphp.com/คู่มือ/73-คืออะไร/3639-visual-studio-วิชาวสดูดิโอ-คืออะไร.html> [2017,October 20].
18. Anonymous, **SDLC**[Online].Available: <http://goo.gl/QRv74W> [2017,October 20].
19. Anonymous, **SDLC Life-cycle**[Online].Available: <http://www.stylusinc.com/down-to-earth-ideation/the-software-development-life-cycle-sdlc/> [2017,October 20].
20. Anonymous, **HTML**[Online].Available: <http://goo.gl/TUC0pL> [2017,October 20].



TNI

ภาคผนวก

TNI

ภาคผนวก ก

รายงานการปฏิบัติงานประจำสัปดาห์ (Weekly Report)

TNI

ประวัติผู้จัดทำโครงการ

ชื่อ-สกุล	นางสาวสุภาสินี ขมณา
วัน เดือน ปีเกิด	5 สิงหาคม 2538
ประวัติการศึกษา	
ระดับประถมศึกษา	โรงเรียนแม่พระฟาติมา กรุงเทพฯ พ.ศ. 2546
ระดับมัธยมศึกษา	โรงเรียนสาธิตมหาวิทยาลัยรามคำแหง กรุงเทพฯ พ.ศ. 2551
ระดับอุดมศึกษา	คณะเทคโนโลยีสารสนเทศ สาขาเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีไทย-ญี่ปุ่น พ.ศ. 2558
ทุนการศึกษา	- ไม่มี -
ประวัติการฝึกอบรม	1. การอบรม Production of Creative Movie โดย M Pictures 2. โครงการ gBLP (Global Project Base Learning) ในหัวข้อ HTML5 and TECHTILE Toolkit ร่วมกับ Shibaura Institute of Technology ที่สถาบันเทคโนโลยีไทย-ญี่ปุ่น 3. โครงการ AKB-TNI Japanese Language Special Course 4. การอบรม Cyber Security Summit
ผลงานที่ได้รับการตีพิมพ์	- ไม่มี -

TNI

