



รายงานสหกิจศึกษาฉบับสมบูรณ์

ระบบรายงาน KPI เพื่อเพิ่มประสิทธิภาพในเชิงธุรกิจ

KPI Reporting System

นางสาวญาดา อาทโรประยูร

สาขาวิชาวิศวกรรมสารสนเทศ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2562

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงการสหกิจศึกษา ระบบรายงาน KPI เพื่อเพิ่มประสิทธิภาพในเชิงธุรกิจ
ชื่อ-สกุล นักศึกษา นางสาวณาดา อาทโรประยูร
คณะ วิศวกรรมศาสตร์ **ภาควิชา** วิศวกรรมคอมพิวเตอร์ **สาขาวิชา** วิศวกรรมสารสนเทศ
ชื่อ-สกุล อาจารย์นิเทศ ผศ. มยุรี เลิศเวชกุล, รศ. ดร. มนต์รี วิบูลย์รัตน์
ชื่อ-สกุล ผู้นิเทศงาน คุณจักรกฤษ ฤทธิเพชรอัมพร
สถานประกอบการ บริษัทแอดวานซ์ ไวร์เลส เน็ตเวิร์ค จำกัด

บทคัดย่อ

การจัดทำโครงการระบบรายงาน KPI เพื่อเพิ่มประสิทธิภาพในเชิงธุรกิจนี้ มีจุดประสงค์ในการจัดทำขึ้นเพื่ออำนวยความสะดวกในการรวบรวม จัดการ และวิเคราะห์ข้อมูลด้านการขาย สำหรับบุคลากรภายในบริษัทแอดวานซ์ ไวร์เลส เน็ตเวิร์ค จำกัด จากกระบวนการแสดงผลแบบเดิมที่ใช้เวลานาน และมีการแสดงผลในรูปแบบที่จำกัด ระบบนี้จะมีการแสดงผลใน 2 รูปแบบ แบ่งตามบุคลากรที่ใช้งาน ซึ่งบุคลากรที่อยู่ในร้านค้ารายย่อยในเครือข่ายของบริษัทจะใช้งานระบบผ่านทาง LINE Bot โดยสามารถดูได้เฉพาะร้านค้าของตนเองเท่านั้น และบุคลากรที่อยู่ในแผนกการตลาด สามารถดูข้อมูลผ่านทางโปรแกรม Power BI โดยการจัดทำในครั้งนี้ใช้ระบบฐานข้อมูล MySQL ใช้ภาษา PHP ร่วมกับ LINE Messaging API และภาษา DAX ในการพัฒนา ผลการจัดทำโครงการในครั้งนี้ทำให้พนักงานสามารถรวบรวมจัดการ วิเคราะห์ข้อมูลได้อย่างถูกต้องและรวดเร็ว และสามารถนำข้อมูลไปใช้ในการตัดสินใจทางธุรกิจต่อไปได้อย่างมีประสิทธิภาพ

คำสำคัญ : ระบบแสดงผล, KPI, LINE Bot, Power BI

Co-operative Title: KPI Reporting System
Student Intern Name: Miss Yada Artaroprayoon
Faculty: Engineering **Department:** Computer Engineering
Program: Information Engineering
Advisor Name: Asst. Prof. Mayuree Lertwatechakul
Assoc. Prof. Dr. Montri Wiboonrat
Mentor Name: Mr. Jakkitt Littipetumporn
Company: Advanced Wireless Network Co., Ltd.

ABSTRACT

KPI Reporting System was developed to facilitate the collection, management, and analysis of sales data for personnel within Advanced Wireless Network Co., Ltd. Since the former system took a long time to process and response and also problem of displaying in a limited format. This system was designed for 2 group of users. The first group is the company staffs who look after retail stores. User of this group will be able to use the system via LINE Bot to view their responsible retail stores' information. The second group is the marketing staffs who can view all information through the Power BI program. The project uses the MySQL database, PHP in conjunction with Line Messaging API and DAX in the development. As a result, the employees can collect, view and analyze data more easily and quickly. Thus, the business decisions can be made much more efficiently.

Keyword: Reporting Systems, KPI, LINE Bot, Power BI

กิตติกรรมประกาศ

การปฏิบัติงานสหกิจศึกษา ณ บริษัทแอดวานซ์ ไรร์เลส เน็ตเวิร์ค จำกัด ในตำแหน่งนักศึกษาฝึกงาน ระหว่างวันที่ 5 สิงหาคม 2562 ถึงวันที่ 22 พฤศจิกายน 2562 ส่งผลให้ได้รับความรู้ใหม่นอกเหนือจากตำราเรียน และประสบการณ์ในการทำงาน สำหรับรายงานสหกิจศึกษาฉบับนี้ สำเร็จสมบูรณ์ เพราะได้รับความช่วยเหลือและคำแนะนำที่เป็นประโยชน์ จากผู้มีพระคุณหลายท่าน

ขอขอบพระคุณ คุณจักรกฤษ ฤทธิเพชรอัมพร ซึ่งเป็นผู้ควบคุมดูแลโครงการ และพี่ ๆ บริษัทแอดวานซ์ ไรร์เลส เน็ตเวิร์ค จำกัด ทุกท่านที่ได้กรุณาเสียสละเวลาในการให้ความรู้ คำแนะนำ และถ่ายทอดประสบการณ์ในด้านการทำงาน ตลอดจนให้ความช่วยเหลือในด้านต่าง ๆ ทำให้การดำเนินสหกิจศึกษาครั้งนี้สำเร็จลุล่วงไปได้ด้วยดีทุกประการ

ขอขอบพระคุณอาจารย์ที่ปรึกษา ผศ.มยุรี เลิศเวชกุล และ รศ. ดร.มนตรี วิบูลย์รัตน์ ที่คอยให้คำปรึกษาและแนวทางการแก้ไขต่าง ๆ รวมไปถึงการติดตามความคืบหน้าของโครงการ ทำให้โครงการนี้สำเร็จไปได้ด้วยดี

สุดท้ายนี้ขอขอบพระคุณบิดา มารดา ที่คอยให้กำลังใจ ให้คำปรึกษา และให้การสนับสนุนมาโดยตลอด รวมถึงเพื่อน ๆ พี่ ๆ และผู้มีส่วนเกี่ยวข้องทุกท่านที่ให้คำแนะนำ และให้ความช่วยเหลือในการดำเนินสหกิจศึกษาในครั้งนี้

นางสาวญาดา อาทโรประยูร

สารบัญ

หน้า

บทคัดย่อ	I
ABSTRACT.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญภาพ	VI
สารบัญตาราง.....	X
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 วิธีการดำเนินงาน	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 Heroku	3
2.2 Git.....	3
2.3 PHP	5
2.4 Composer.....	6
2.5 SQL	6
2.5.1 ส่วนประกอบของ SQL.....	7
2.5.2 ประเภทของคำสั่งภาษา SQL.....	7
2.6 IMAP.....	8
2.7 JSON.....	8
2.8 API.....	9
2.8.1 รูปแบบการนำ API ไปใช้งาน.....	9
2.9 LINE.....	10
2.9.1 LINE Bot.....	10
2.10 LINE Messaging API.....	11

สารบัญ (ต่อ)

	หน้า
2.11 BI.....	14
2.12 Power BI.....	15
2.12.1 ส่วนประกอบต่าง ๆ ของ Power BI.....	15
2.13 DAX (Data Analysis Expression).....	16
2.13.1 โครงสร้าง DAX.....	16
2.14 KPI (Key Performance Indicator).....	17
บทที่ 3 ขั้นตอนการดำเนินงาน.....	18
3.1 กรอบการดำเนินงาน.....	18
3.2 วิธีการดำเนินงาน.....	19
3.2.1 ศึกษาข้อมูลที่ใช้ในการทำโครงการ.....	19
3.2.2 ออกแบบและสร้างฐานข้อมูล.....	20
3.2.3 การสร้างระบบบันทึกข้อมูลจากอีเมลลงฐานข้อมูลอัตโนมัติ.....	24
3.2.4 การจัดทำระบบ LINE Bot.....	30
3.2.5 การจัดทำแดชบอร์ดรายงานข้อมูล โดยใช้โปรแกรม Power BI.....	38
บทที่ 4 การทำงานของระบบรายงาน KPI.....	43
4.1 ระบบบันทึกข้อมูลจากอีเมลลงฐานข้อมูลอัตโนมัติ.....	43
4.2 ระบบ LINE Bot.....	44
4.2.1 การลงทะเบียนเข้าใช้งาน LINE Bot.....	44
4.2.2 การใช้งาน LINE Bot.....	46
4.3 แดชบอร์ดรายงานข้อมูล โดยใช้โปรแกรม Power BI.....	57
บทที่ 5 สรุปผลการดำเนินงาน ปัญหา และข้อเสนอแนะ.....	67
5.1 สรุปผลการดำเนินงาน.....	67
5.2 ปัญหาและอุปสรรคที่พบ.....	67
5.3 ข้อเสนอแนะ.....	67
เอกสารอ้างอิง.....	68

สารบัญภาพ

	หน้า
รูปที่ 2.1 สัญลักษณ์ของ Heroku	3
รูปที่ 2.2 สัญลักษณ์ของ Git	3
รูปที่ 2.3 สัญลักษณ์ของ PHP.....	5
รูปที่ 2.4 ตัวอย่าง code ภาษา PHP	5
รูปที่ 2.5 สัญลักษณ์ของ Composer	6
รูปที่ 2.6 ส่วนประกอบของ SQL	7
รูปที่ 2.7 แผนภาพแสดงการทำงานของ IMAP.....	8
รูปที่ 2.8 สัญลักษณ์ของ JSON.....	8
รูปที่ 2.9 แผนภาพการแสดงตัวอย่างระบบที่มีการใช้ API.....	10
รูปที่ 2.10 สัญลักษณ์ของ LINE.....	10
รูปที่ 2.11 แผนภาพการทำงานของ LINE Messaging API.....	11
รูปที่ 2.12 การ Push Messages และการ Reply Messages	12
รูปที่ 2.13 ตัวอย่าง Flex Message แบบ Bubble.....	13
รูปที่ 2.14 ตัวอย่าง Flex Message แบบ Carousel.....	13
รูปที่ 2.15 หน้าจอ LIFF ในขนาดต่าง ๆ.....	14
รูปที่ 2.16 ตัวอย่างหน้ารายงานของ Power BI๗	15
รูปที่ 2.17 ตัวอย่างสูตรคำนวณ DAX	16
รูปที่ 3.1 หน้าจอของ MySQL Workbench.....	20
รูปที่ 3.2 flowchart การทำงานระบบบันทึกข้อมูลจากอีเมลลงฐานข้อมูล.....	24
รูปที่ 3.3 หน้าจอสมัครเข้าใช้งานบัญชี Heroku.....	25
รูปที่ 3.4 หน้าจอการติดตั้ง Git	25
รูปที่ 3.5 การติดตั้ง Heroku Command Line Interface	26
รูปที่ 3.6 การสร้างแอปพลิเคชันบน Heroku	26
รูปที่ 3.7 การส่งไฟล์จากคอมพิวเตอร์ ไปยัง Heroku Server.....	27
รูปที่ 3.8 การเปิดการใช้งาน IMAP	28
รูปที่ 3.9 โค้ดที่ใช้ในการรันไฟล์สคริปต์ PHP	28

สารบัญภาพ (ต่อ)

	หน้า
รูปที่ 3.10 หน้าจอที่ใช้ในการสร้าง Task บนโปรแกรม Task Scheduler	29
รูปที่ 3.11 หน้าจอของแถบ New Action	29
รูปที่ 3.12 หน้าจอของแถบ New Trigger	30
รูปที่ 3.13 โครงสร้างการทำงานของ LINE Bot	30
รูปที่ 3.14 Activity Diagram ของการลงชื่อเข้าใช้งาน LINE Bot	31
รูปที่ 3.15 Activity Diagram การทำงานของ LINE Bot	32
รูปที่ 3.16 การสร้างแอปพลิเคชัน สำหรับ Line bot บน Heroku	33
รูปที่ 3.17 หน้าจอ https://developers.line.biz/console	34
รูปที่ 3.18 การสร้างแชแนล	34
รูปที่ 3.19 ตัวอย่างโค้ดที่ใช้ในการพัฒนา LINE Bot	35
รูปที่ 3.20 การส่งไฟล์จากคอมพิวเตอร์ ไปยัง Heroku Server	36
รูปที่ 3.21 การกำหนด Webhook URL	36
รูปที่ 3.22 การกำหนด LIFF App	37
รูปที่ 3.23 การกำหนด Rich Menu	37
รูปที่ 3.24 ภาพรวมของการพัฒนาระบบสารสนเทศเพื่อการแสดงข้อมูลของ บริษัทแอดวานซ์ ไรร์เลส เน็ตเวิร์ค จำกัด โดยใช้โปรแกรม Power BI	38
รูปที่ 3.25 การเชื่อมต่อฐานข้อมูลกับแดชบอร์ด	39
รูปที่ 3.26 หน้าต่าง Power Query Editor	39
รูปที่ 3.27 หน้าจอในส่วนของ Panel: Data View	40
รูปที่ 3.28 ตัวอย่างภาษา DAX	40
รูปที่ 3.29 หน้าต่างแก้ไขความสัมพันธ์ระหว่างข้อมูล	40
รูปที่ 3.30 Entity Relationship Diagram บน Panel: Model View	41
รูปที่ 3.31 การเผยแพร่ไปยัง Power BI cloud service	42
รูปที่ 3.32 การแสดงผลรายงานบน Power BI cloud service	42
รูปที่ 4.1 การเปิดหน้าเว็บ PHP บนโปรแกรม Internet Explorer	43
รูปที่ 4.2 หน้าจอการทำงาน Batch File	43

สารบัญภาพ (ต่อ)

	หน้า
รูปที่ 4.3 หน้าจอเมื่อผู้ใช้ทำการเพิ่ม LINE Bot เป็นเพื่อน	44
รูปที่ 4.4 หน้าจอการลงทะเบียนใช้งาน LINE Bot.....	45
รูปที่ 4.5 หน้าจอยืนยันข้อมูลการลงทะเบียน	45
รูปที่ 4.6 Dialog Box เตือนผู้ใช้ว่าไม่สามารถลงทะเบียนได้	46
รูปที่ 4.7 หน้าจอแสดงเมนูทั้งหมด กรณีที่ผู้ใช้พิมพ์รหัสยืนยันตัวตนถูกต้อง	47
รูปที่ 4.8 เมนูทั้งหมด	47
รูปที่ 4.9 หน้าจอแสดงเมนูทั้งหมด กรณีที่ผู้ใช้พิมพ์รหัสยืนยันตัวตนไม่ถูกต้อง	47
รูปที่ 4.10 Rich Menu.....	48
รูปที่ 4.11 Flex Message เมนูยอดขายรวม.....	48
รูปที่ 4.12 Flex Message แสดงยอดขาย Postpaid.....	49
รูปที่ 4.13 Flex Message แสดงยอดขาย Prepaid.....	50
รูปที่ 4.14 Flex Message แสดงยอดขายจำนวนแพ็คเกจเสริมที่ขายได้.....	50
รูปที่ 4.15 ข้อความตอบกลับเมื่อผู้ใช้กดเลือกเมนู ยอดขายย้อนหลัง 6 เดือน.....	51
รูปที่ 4.16 LIFF APP แสดงกราฟยอดขายซิมประเภทเติมเงิน 6 เดือนย้อนหลัง.....	51
รูปที่ 4.17 Flex Message แสดง เป้าหมายของยอดขายซิม	52
รูปที่ 4.18 Flex Message เมนูสิทธิพิเศษ.....	52
รูปที่ 4.19 ข้อความ ตอบกลับเมื่อผู้ใช้กดเลือกเมนู สิทธิที่ร้านจะได้รับ (ยอดขาย 0 ซิม)	53
รูปที่ 4.20 รูปภาพและ Flex Message ตอบกลับเมื่อผู้ใช้กดเลือกเมนู สิทธิที่ร้านจะได้รับ (ยอดขาย 1-15 ซิม)	53
รูปที่ 4.21 รูปภาพและ Flex Message ตอบกลับเมื่อผู้ใช้กดเลือกเมนู สิทธิที่ร้านจะได้รับ (ยอดขาย 16-29 ซิม).....	54
รูปที่ 4.22 รูปภาพและ Flex Message ตอบกลับเมื่อผู้ใช้กดเลือกเมนู สิทธิที่ร้านจะได้รับ (ยอดขาย 30-44 ซิม).....	54
รูปที่ 4.23 รูปภาพและ Flex Message ตอบกลับเมื่อผู้ใช้กดเลือกเมนู สิทธิที่ร้านจะได้รับ	55
(ยอดขาย 45 ซิมขึ้นไป).....	55
รูปที่ 4.24 ข้อความตอบกลับเมื่อผู้ใช้กดเลือกสิทธิพิเศษ	55

สารบัญภาพ (ต่อ)

	หน้า
รูปที่ 4.25 ข้อความตอบกลับ กรณีผู้ใช้เคยกดเลือกสิทธิพิเศษไปแล้ว.....	56
รูปที่ 4.26 รูปและ Flex Message ตอบกลับเมื่อผู้ใช้กดเลือกเมนู เปลี่ยนสิทธิพิเศษ.....	56
รูปที่ 4.27 ข้อความตอบกลับเมื่อผู้ใช้กดเลือก (เปลี่ยน) สิทธิพิเศษ	57
รูปที่ 4.28 ข้อความตอบกลับ กรณีผู้ใช้ส่งข้อความนอกเหนือจากเมนูที่กำหนดไว้.....	57
รูปที่ 4.29 รายงานเปรียบเทียบยอดขายประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน	57
รูปที่ 4.30 รายงานยอดขายแบ่งตามสถานที่ตั้ง	58
รูปที่ 4.31 รายงานยอดขายในภาพรวมตามเวลาที่กำหนด แบ่งตามชื่อร้าน.....	60
รูปที่ 4.32 รายงานแสดงยอดขายและเป้าหมาย แบ่งตามชื่อร้าน.....	62
รูปที่ 4.33 รายงานการเลือกสิทธิของร้านต่าง ๆ แบ่งตามเวลา.....	64
รูปที่ 4.34 รายงานการใช้งาน LINE Bot ของร้านค้า.....	65

สารบัญตาราง

	หน้า
ตารางที่ 2.1 สถานะของ Git.....	4
ตารางที่ 2.2 ตัวอย่างคำสั่งของ Git.....	4
ตารางที่ 2.3 คำสั่งที่ใช้กับ Composer.....	6
ตารางที่ 2.4 คุณสมบัติของการส่ง Flex Message.....	12
ตารางที่ 3.1 โครงสร้างฐานข้อมูลของ shopdata (ข้อมูลของร้านค้า).....	21
ตารางที่ 3.2 โครงสร้างฐานข้อมูลของ loguser (ข้อมูลการใช้งาน LINE Bot ของร้านค้า).....	21
ตารางที่ 3.3 โครงสร้างฐานข้อมูลของ sales (ข้อมูลการขายในแต่ละวัน).....	22
ตารางที่ 3.4 โครงสร้างฐานข้อมูลของ privilege (ข้อมูลสิทธิพิเศษที่ร้านค้าเลือก).....	23
ตารางที่ 3.5 โครงสร้างฐานข้อมูลของ dataforsystems (ข้อมูลที่ใช้ในการพัฒนาระบบ).....	23

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

เนื่องจากบริษัท แอดวานซ์ ไวร์เลส เน็ทเวิร์ค จำกัด ต้องการระบบแสดงผลข้อมูลยอดขายแบบใหม่ ทดแทนระบบแสดงผลข้อมูลยอดขายแบบเดิม ซึ่งแต่เดิมนั้นเป็นการแสดงผลข้อมูลยอดขายสำหรับร้านค้าในเครือข่ายของบริษัทโดยใช้เอสเอ็มเอส (Short Message Service: SMS) ซึ่งสามารถแสดงผลข้อมูลได้ในปริมาณที่จำกัด และมีรูปแบบที่อ่านยาก ไม่สวยงาม ประกอบกับแอปพลิเคชันไลน์เป็นแอปพลิเคชันที่ได้รับความนิยมเป็นอย่างมากในประเทศไทย จึงทำให้ผู้จัดทำได้รับมอบหมายในการสร้าง LINE Chat Bot เพื่อให้เจ้าของกิจการร้านค้าในเครือข่ายของบริษัทสามารถสอบถามข้อมูลได้ตรงกับความต้องการมากยิ่งขึ้น

และเนื่องจากการรวบรวมและวิเคราะห์ข้อมูลที่บริษัทดำเนินการอยู่นั้น ใช้โปรแกรม Microsoft Excel ซึ่งมีข้อจำกัดคือสามารถวิเคราะห์ข้อมูลได้ในปริมาณจำกัด มีการแสดงผลในรูปแบบที่จำกัด และใช้เวลานานในการวิเคราะห์ข้อมูล รวมถึงพนักงานในบริษัทสามารถใช้บริการโปรแกรม Power BI ได้โดยไม่เสียค่าใช้จ่าย ทางบริษัทจึงได้มอบหมายให้ผู้จัดทำใช้โปรแกรม Power BI ในการแสดงผลข้อมูลเพื่อให้ง่ายต่อการตัดสินใจทางธุรกิจต่อไป

1.2 วัตถุประสงค์ของโครงการ

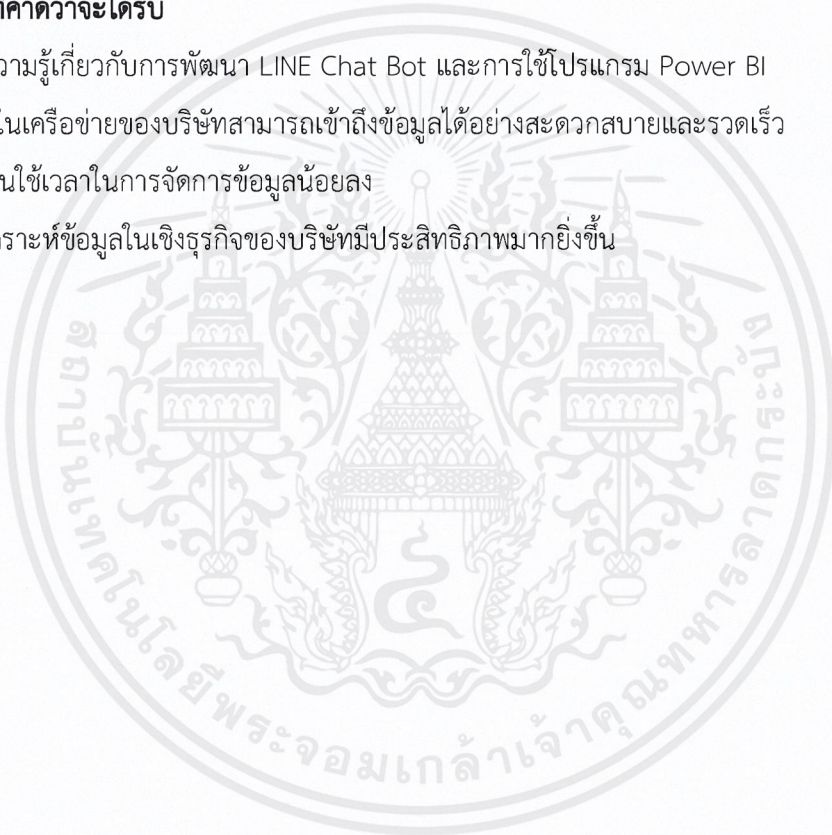
- 1) เพื่อศึกษาการใช้เทคโนโลยีและเครื่องมือที่ใช้การพัฒนาระบบ
- 2) เพื่ออำนวยความสะดวกให้กับร้านค้าในเครือข่ายของบริษัทในการสอบถามข้อมูล
- 3) เพื่อลดระยะเวลาที่ใช้ในการจัดการข้อมูลของพนักงาน
- 4) เพื่อช่วยในการเพิ่มประสิทธิภาพในการวิเคราะห์ข้อมูลในเชิงธุรกิจ

1.3 วิธีการดำเนินงาน

- 1) ทำความเข้าใจเกี่ยวกับข้อมูลยอดขายของบริษัท
- 2) ศึกษาเทคโนโลยีที่ต้องใช้ในการพัฒนาระบบ
- 3) จัดการข้อมูลตัวอย่างให้อยู่ในรูปแบบที่เหมาะสมต่อการพัฒนาระบบ
- 4) พัฒนาระบบที่ได้รับมอบหมาย
- 5) ทดสอบการใช้งานและแก้ไขระบบให้สมบูรณ์

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ได้รับความรู้เกี่ยวกับการพัฒนา LINE Chat Bot และการใช้โปรแกรม Power BI
- 2) ราคาค่าในเครือข่ายของบริษัทสามารถเข้าถึงข้อมูลได้อย่างสะดวกสบายและรวดเร็ว
- 3) พนักงานใช้เวลาในการจัดการข้อมูลน้อยลง
- 4) การวิเคราะห์ข้อมูลในเชิงธุรกิจของบริษัทมีประสิทธิภาพมากยิ่งขึ้น



บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 Heroku

Heroku คือ Platform as a Service (PaaS) บนคลาวด์ ผู้พัฒนาจะสามารถควบคุมจัดการใน ระดับ Application Container เท่านั้นหรืออีกนัยหนึ่งคือ การนำ Application Server หรือ Web Server มาให้บริการอยู่บนการประมวลผลกลุ่มเมฆ ยึดตามแพลตฟอร์มการพัฒนาต่าง ๆ เช่น Java, .Net, PHP, Python และอื่น ๆ รวมไปถึงสามารถสร้าง buildpack สำหรับภาษาอื่น ๆ ได้ เช่น Lua ที่รันอยู่บน OpenResty โดยแอปพลิเคชันที่ทำงานบน Heroku จะมีโดเมนที่ไม่ซ้ำกัน ผู้พัฒนาสามารถรันคำสั่งบน Heroku โดยการ push code ผ่าน git ขึ้นไปบน heroku ผ่านคำสั่ง git init, git add ., git commit -m "Initial project" และ git push heroku master



รูปที่ 2.1 สัญลักษณ์ของ Heroku

(ที่มา: <https://seekvectorlogo.net/heroku-vector-logo-svg>)

2.2 Git

Git คือ Version Control แบบหนึ่งซึ่งเป็นระบบที่มีหน้าที่ในการจัดเก็บการเปลี่ยนแปลงของไฟล์ ในโปรเจกต์ ซึ่งเป็นไฟล์ชนิดใดก็ได้ มีการสำรองไฟล์ข้อมูลทำให้สามารถที่จะเรียกดูหรือย้อนกลับไปได้ เวอร์ชันต่าง ๆ ของโปรเจกต์ที่ใดเวลาใดก็ได้ รวมไปถึงสามารถดูว่าผู้ใดบ้างที่เขียนและแก้ไขไฟล์ ทำให้ Git เป็นที่นิยมสำหรับโปรเจกต์ที่มีผู้พัฒนาจำนวนมาก



รูปที่ 2.2 สัญลักษณ์ของ Git

(ที่มา: <https://git-scm.com/images/logos/downloads/Git-Logo-1788C.png>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 สถานะของ Git

สถานะ	คำอธิบายสถานะ
Untracked	สถานะที่ Source Code ถูกเพิ่มเข้ามาใหม่และยังไม่ได้ถูกเก็บไว้ในระบบของ Git
Working Directory/Modified	สถานะที่กำลังมีการเปลี่ยนแปลงหรือแก้ไข Source Code
Staged	สถานะที่ Source Code กำลังเตรียมที่จะ Commit เพื่อยืนยันการเปลี่ยนแปลงก่อนที่จะเก็บลงในสถานะ Local Repository
Local Repository	สถานะที่มีการเก็บบันทึกข้อมูลการเปลี่ยนแปลงของ Source Code ลงไปที่ Git Repository ที่เป็น Local
Remote Repository	สถานะที่มีการเก็บบันทึกข้อมูลการเปลี่ยนแปลงของ Source Code ลงไปที่ Git Repository ที่เป็น Hosting

ตารางที่ 2.2 ตัวอย่างคำสั่งของ Git

คำสั่ง	การใช้งาน
Git Init	คำสั่งที่ใช้สร้างระบบของ Git ขึ้นมาภายใต้ Path นั้น โดยจะสร้างโฟลเดอร์ .git ขึ้นมาเพื่อใช้เก็บสำรองข้อมูลการเปลี่ยนแปลงและคุณสมบัติอื่น ๆ ของ Git
Git Status	คำสั่งที่ใช้ตรวจสอบสถานะของ Source Code ในระบบของ Git
Git Add	คำสั่งที่ใช้เพิ่มการเปลี่ยนแปลงของ Source Code เข้าไปที่สถานะ Staged
Git Commit	คำสั่งที่ยืนยัน Source Code ที่อยู่ในสถานะ Staged ขึ้นไปเก็บไว้บน Local Repository
Git Push	คำสั่งที่ใช้ส่ง Source Code ที่เก็บอยู่บน Local Repository ขึ้นไปยัง Remote Repository

2.3 PHP

PHP (PHP Hypertext Preprocessor) คือภาษาคอมพิวเตอร์ในรูปแบบ open source ออกแบบมาเพื่อการพัฒนาเว็บไซต์ที่สามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ เวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง และถูกเรียกใช้งานโดย Command Line Interface (CLI) ซึ่งฝังอยู่ในโค้ด HTML หรือใช้ร่วมกับระบบแม่แบบของเว็บต่าง ๆ และ web frameworks

PHP เป็นเครื่องมือสำคัญที่ช่วยให้สามารถสร้าง dynamic web pages (เว็บเพจที่มีการโต้ตอบกับผู้ใช้) ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น ลักษณะเด่นของ PHP คือ ไม่ต้องเสียค่าใช้จ่ายในการใช้งาน สามารถทำงานบนระบบปฏิบัติการ UNIX, Linux, Windows โปรแกรมที่ใช้ PHP เป็นโครงสร้างหลักคือ WordPress, Joomla! เป็นต้น



รูปที่ 2.3 สัญลักษณ์ของ PHP

(ที่มา: <https://upload.wikimedia.org/wikipedia/th/7/74/PHP.png>)

```
<?php
srand( (double)microtime()*1000000);

if (rand(1,2)==1)
{
    include("file1.html");
}
else
{
    include("file2.html");
};
?>
```

รูปที่ 2.4 ตัวอย่าง code ภาษา PHP

(ที่มา: <https://www.tomwerf.nl/stel-de-juiste-tijdzone-in-binnen-php5>)

2.4 Composer

เป็นเครื่องมือจัดการแพ็คเกจ (package) ระดับโปรแกรมประยุกต์ ใช้ในการเขียนโปรแกรมด้วยภาษา PHP ที่มีรูปแบบมาตรฐานในการจัดการซอฟต์แวร์ PHP และไลบรารี (library)

เมื่อต้องการใช้งานส่วนเสริมของ PHP จะต้องติดตั้งแพ็คเกจของส่วนเสริมนั้นซึ่ง composer จะทำหน้าที่จัดการแพ็คเกจของส่วนเสริมให้เป็นหมวดหมู่ ทำให้สามารถค้นหาและจัดการได้ง่าย

ตารางที่ 2.3 คำสั่งที่ใช้กับ Composer

คำสั่ง	การใช้งาน
Require	คำสั่งเพิ่มไลบรารีในพารามิเตอร์ลงในไฟล์ composer.json และติดตั้ง
Install	คำสั่งติดตั้งไลบรารีทั้งหมดจาก composer.json
Update	คำสั่งอัปเดตไลบรารีทั้งหมดจาก composer.json ตามรุ่นที่ระบุไว้
remove	คำสั่งถอนการติดตั้งไลบรารีและลบออกจาก composer.json



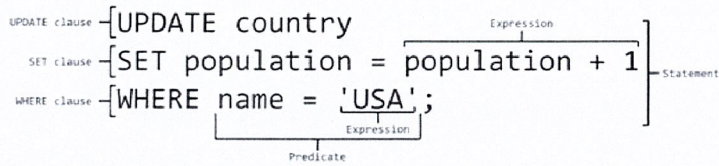
รูปที่ 2.5 สัญลักษณ์ของ Composer

(ที่มา: <https://cdn.freebiesupply.com/logos/thumbs/2x/composer-logo.png>)

2.5 SQL

SQL (Structured Query Language) เป็นภาษาแบบ domain-specific ใช้ในการบริหารจัดการฐานข้อมูล (Database) โดยเฉพาะ Relational Database Management System (RDBMS) เช่น Oracle, MySQL, Microsoft SQL Server, PostgreSQL อีกทั้งยังเป็นมาตรฐานกลางของ ANSI (American National Standard Institute) ที่ใช้ในระบบฐานข้อมูลต่าง ๆ โดย SQL นั้นสามารถใช้ในการสืบค้นข้อมูล, ปรับปรุงข้อมูล, สร้างฐานข้อมูล, สร้าง Stored Procedure ในฐานข้อมูล รวมถึงกำหนดสิทธิ์ให้กับตาราง (Table), Procedure และ Views อีกด้วย

2.5.1 ส่วนประกอบของ SQL



รูปที่ 2.6 ส่วนประกอบของ SQL

(ที่มา: [https://upload.wikimedia.org/wikipedia/commons/thumb/a/aa/](https://upload.wikimedia.org/wikipedia/commons/thumb/a/aa/SQL_ANATOMY_wiki.svg/550px-SQL_ANATOMY_wiki.svg.png)

[SQL_ANATOMY_wiki.svg/550px-SQL_ANATOMY_wiki.svg.png](https://upload.wikimedia.org/wikipedia/commons/thumb/a/aa/SQL_ANATOMY_wiki.svg/550px-SQL_ANATOMY_wiki.svg.png))

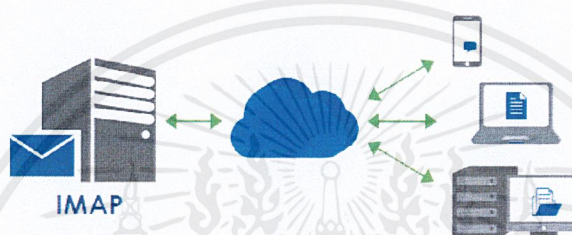
- 1) Queries: การดึงข้อมูลตามเงื่อนไข (clause)
- 2) Clauses: เป็นองค์ประกอบหนึ่งของ Statement และ Query
- 3) Expressions: การสร้างผลลัพธ์ออกมาในรูปแบบตารางข้อมูลที่ประกอบด้วยแถวและคอลัมน์
- 4) Predicates: รูปแบบเงื่อนไขที่มีผลลัพธ์เป็น true หรือ false หรือ unknown
- 5) Statements: คำสั่งที่ผลต่อโครงสร้างข้อมูล, จัดการข้อมูล transactions, program flow และ session ต้องลงท้ายด้วย semicolon (;) เสมอ

2.5.2 ประเภทของคำสั่งภาษา SQL

- 1) ภาษานิยามข้อมูล (Data Definition Language: DDL): เป็นคำสั่งที่ใช้ในการสร้างฐานข้อมูล กำหนดโครงสร้างข้อมูลว่าประกอบด้วยแอตทริบิวต์หรือชนิดของข้อมูลใดบ้าง รวมทั้งการเปลี่ยนแปลงตาราง และการสร้างดัชนี คำสั่งที่ใช้คือ CREATE, DROP และ ALTER
- 2) ภาษาจัดการข้อมูล (Data Manipulation Language: DML) เป็นคำสั่งที่ใช้ในการเรียกใช้, เพิ่ม, ลบ และเปลี่ยนแปลงข้อมูลในตาราง คำสั่งที่ใช้คือ SELECT, INSERT, UPDATE และ DELETE
- 3) ภาษาควบคุมข้อมูล (Data Control Language: DCL) เป็นคำสั่งที่ใช้ในการกำหนดสิทธิ์การอนุญาตหรือยกเลิกการเข้าถึงฐานข้อมูล เพื่อป้องกันความปลอดภัยของฐานข้อมูล คำสั่งที่ใช้คือ GRANT, REVOKE

2.6 IMAP

IMAP (Internet Message Access Protocol) เป็นโพรโทคอลที่ใช้ในการรับส่งอีเมลโดยจะเชื่อมต่อกับเครื่องคอมพิวเตอร์แม่ข่าย (Mail Server) ตลอดเวลาเพื่อจัดการข้อมูลต่าง ๆ ซึ่งการจัดการข้อมูลต่าง ๆ นี้จะจัดการที่เครื่องแม่ข่ายทั้งหมด และทำงานควบคู่กับอุปกรณ์ต่าง ๆ โดยจะส่งผลการจัดการมาที่เครื่องคอมพิวเตอร์ลูกข่าย (Mail Client) บนอุปกรณ์ทุกอุปกรณ์ที่มีการเชื่อมต่ออยู่ เป็นการรองรับการลงชื่อเข้าใช้หลายบัญชี ผู้ใช้สามารถเข้าถึงอีเมลได้ตลอดเวลา ทำให้ผู้ใช้สามารถจัดระเบียบข้อความ ติดตามข้อความที่สำคัญและเร่งด่วน หรือติดตามและบันทึกข้อความฉบับร่างบน ใดบนอุปกรณ์ต่าง ๆ พร้อมกัน

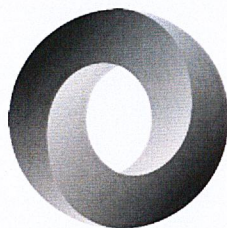


รูปที่ 2.7 แผนภาพแสดงการทำงานของ IMAP

(ที่มา: https://s3.amazonaws.com/s3.timetoast.com/public/uploads/photos/12611907/descarga_%281%29.png)

2.7 JSON

JSON (JavaScript Object Notation) เป็นรูปแบบมาตรฐานของการจัดเก็บข้อมูลของ JavaScript จัดเก็บในลักษณะของ JSON Object เพื่อใช้ในการส่งข้อมูลขนาดเล็กระหว่างแอปพลิเคชัน หรือ API (Application Programming Interface) โดย JSON มีลักษณะคือ มนุษย์สามารถเข้าใจได้ง่าย และมีหลายภาษาที่ใช้งาน JSON โดยการสร้างและแปลงรูปแบบไปมาได้ เช่น ภาษา PHP, ภาษา Python เป็นต้น การเก็บข้อมูลในรูปแบบของ JSON สามารถเก็บลงตัวแปร (ในหน่วยความจำ) เก็บเป็นไฟล์ข้อมูล JSON หรือ บันทึกในไฟล์ .js ก็ได้



รูปที่ 2.8 สัญลักษณ์ของ JSON

(ที่มา: <https://cdn.freebiesupply.com/logos/thumbs/2x/json-logo.png>)

2.8 API

API (Application Programming Interface) เป็นโปรโตคอลที่ใช้ในการสื่อสารระหว่างโปรแกรมประยุกต์หลายโปรแกรม ทำให้คอมพิวเตอร์สามารถสื่อสารและแลกเปลี่ยนข้อมูลกันได้อย่างอิสระ ทำให้นักพัฒนาสามารถพัฒนาระบบได้อย่างรวดเร็ว และผู้ใช้งานสามารถใช้งานระบบได้อย่างสะดวกสบาย โดย API มีส่วนประกอบสำคัญ 2 อย่าง คือ ข้อกำหนดที่อธิบายการแลกเปลี่ยนข้อมูลระหว่างโปรแกรมในรูปแบบเอกสารเพื่อบอกลักษณะของคำขอกับการตอบสนอง (Request/Response) และซอฟต์แวร์ (Software) ที่เขียนขึ้นตามข้อกำหนดเพื่อนำไปใช้งาน

2.8.1 รูปแบบการนำ API ไปใช้งาน

1) Libraries and Frameworks

ใช้ API ในรูปแบบของ software library ซึ่งเขียนตามเอกสารข้อกำหนดของภาษาที่ใช้เขียนโปรแกรมนั้น ๆ สำหรับนำไปทำเป็น framework ให้กับระบบในขั้นตอนต่อไป

2) Operating Systems

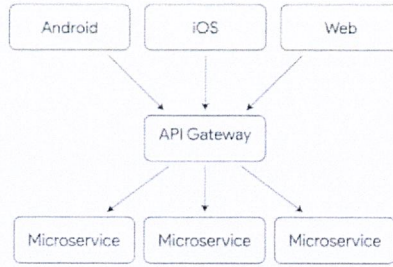
ใช้ API ในการสื่อสารระหว่างแอปพลิเคชันและระบบปฏิบัติการ (Operating System: OS) เช่น การใช้ command line (API) ควบคุมการสื่อสารการทำงานของระบบปฏิบัติการ

3) Remote APIs

ใช้ API ในการกำหนดมาตรฐานการเข้าถึงทรัพยากรของผู้พัฒนา มักถูกใช้งานพัฒนาและปรับปรุงระบบ ด้วยการที่ฝั่งเครื่องคอมพิวเตอร์ลูกข่ายดึงข้อมูลจากเครื่องคอมพิวเตอร์แม่ข่ายกลับมาทำงานผ่าน API

4) Web APIs

เป็น API ในกลุ่มของ HTTP และขยายรูปแบบไปเป็น XML และ JSON อยู่บน Web Service



รูปที่ 2.9 แผนภาพการแสดงตัวอย่างระบบที่มีการใช้ API

(ที่มา: https://lh3.googleusercontent.com/RCHRZWq2FCkBzR_D-CIN5ZBmXOEj-_e4RDEFigByiiYutZFLYae4Awlg05uQh49Hk6sx4zY=s168)

2.9 LINE

LINE คือโปรแกรมสำหรับการส่งข้อความโต้ตอบแบบ Instant Messaging (IM) ซึ่งเป็นการส่งข้อความระหว่างสองคนหรือกลุ่มคนในเน็ตเวิร์กเดียวกันผ่านอุปกรณ์อิเล็กทรอนิกส์ เช่น โทรศัพท์, แท็บเล็ต, คอมพิวเตอร์ และคอมพิวเตอร์ส่วนบุคคล ผู้ใช้สามารถแลกเปลี่ยนข้อความ, รูปภาพ, วิดีโอ, เสียง, ดำเนินการสนทนาทาง VoIP และดำเนินการประชุมทางวิดีโอได้ นอกจากนี้ LINE ยังเป็นแพลตฟอร์มที่ให้บริการต่าง ๆ รวมถึง LINE Pay (กระเป๋าเงินออนไลน์), LINE TV (เว็บไซต์และแอปพลิเคชันที่ให้บริการวิดีโอตามคำขอ) และ LINE Webtoon (แพลตฟอร์มการ์ตูนดิจิทัล)



รูปที่ 2.10 สัญลักษณ์ของ LINE

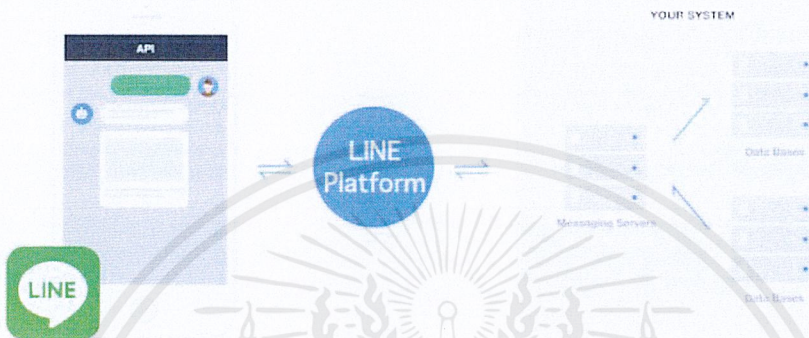
(ที่มา: https://encryptedtbn0.gstatic.com/images?q=tbn%3AANd9GcRdwJ2QMgVh2YJauteN_R7ACZmh_Ljj97czxwSjXbOz9G11hoxv)

2.9.1 LINE Bot

LINE Bot คือ LINE Official Account ที่ได้นำ Messaging API มาใช้ โดยผู้พัฒนาสามารถกำหนดหรือตั้งค่าที่ back-end เพื่อให้สามารถโต้ตอบกับผู้ใช้ได้อัตโนมัติ ทำให้ผู้ดูแลไม่ต้องตอบคำซ้ำหลายครั้ง และผู้ใช้สามารถใช้งานได้ง่ายมากขึ้น

2.10 LINE Messaging API

LINE Messaging API ใช้เพื่อการสร้าง Chat Bot สำหรับผู้ใช้โปรแกรม ผู้พัฒนาต้องเชื่อม Bot ที่สร้างขึ้นกับแชนเนล (Channel) บนแพลตฟอร์มของไลน์ เพื่อให้ Line Messaging API สามารถส่งผ่านข้อมูลระหว่างเครื่องคอมพิวเตอร์แม่ข่าย, Bot และแพลตฟอร์มของไลน์ได้ โดยข้อมูลคำขอ (Request) จะถูกส่งผ่าน HTTPS ในรูปแบบของ JSON



รูปที่ 2.11 แผนภาพการทำงานของ LINE Messaging API

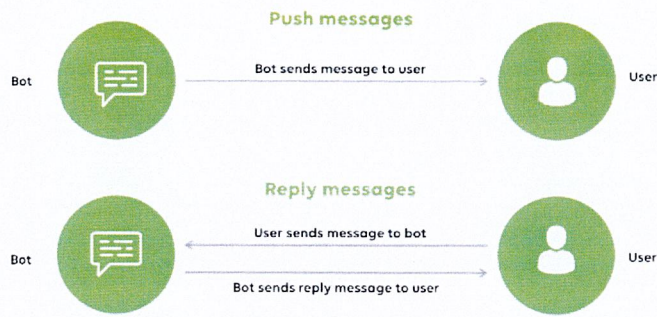
(ที่มา: https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcSgF1jRtzexsgYuGV_v-28zpQ8UpSG5jp59yg4Myez_LL4M-A5B)

2.10.1 หลักการทำงานของ LINE Messaging API

- 1) ผู้ใช้งานส่งข้อความมายังบัญชีทางการของ LINE
- 2) แพลตฟอร์มของ LINE ส่ง webhook event (เหตุการณ์ต่าง ๆ ที่เกิดขึ้นกับ LINE Bot) ไปยัง webhook URL ของ Bot Server
- 3) Bot Server ตอบสนองผู้ใช้ผ่านทางแพลตฟอร์มของ LINE

2.10.2 การดำเนินการเกี่ยวกับข้อมูลผ่าน LINE Messaging API

- 1) Reply Messages
เป็นการตอบกลับข้อความไปยังผู้ใช้ที่ได้ตอบกับบัญชีทางการของ LINE จำเป็นต้องใช้ Reply Token ในการตอบกลับข้อความ
- 2) Push Messages
เป็นการส่งข้อความโดยตรงไปยังผู้ใช้เมื่อใดก็ได้ตามที่ต้องการ ไม่จำเป็นต้องใช้ Reply Token ในการส่งข้อความ



รูปที่ 2.12 การ Push Messages และการ Reply Messages

(ที่มา: https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcTQ4_-gmxuWqwOaRLzhklFfRem14Ypbgi0IsdjE2dtUWhmT4fmL)

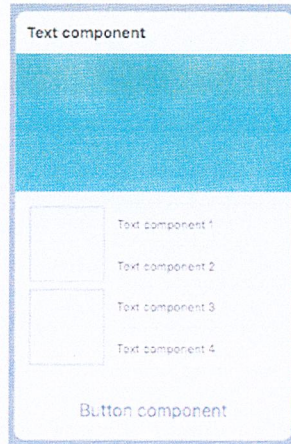
2.10.3 Flex Message

Flex Message เป็นข้อความที่นักพัฒนาสามารถกำหนดรูปแบบเองได้ โดยสามารถปรับแต่งรูปแบบได้อย่างอิสระตามข้อกำหนดของ CSS Flexbox

ตารางที่ 2.4 คุณสมบัติของการส่ง Flex Message

คุณสมบัติ	ประเภท	คำอธิบาย
type	String	ข้อความประเภท flex
altText	String	ข้อความยาวได้มากที่สุด 400 ตัวอักษร
contents	Object	เนื้อหาของ Flex Message

Flex Message ประกอบด้วยโครงสร้าง 2 ระดับ คือ Bubble และ Carousel (Bubble หลาย ๆ Bubble มาต่อกัน)



รูปที่ 2.13 ตัวอย่าง Flex Message แบบ Bubble

(ที่มา: <https://developers.line.biz/en/docs/messaging-api/flex-message-elements>)



รูปที่ 2.14 ตัวอย่าง Flex Message แบบ Carousel

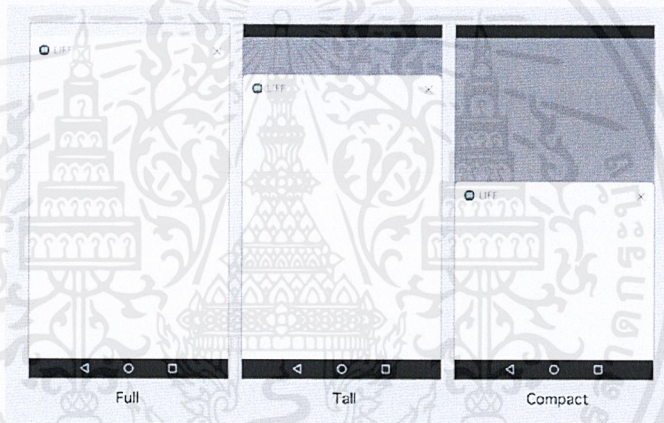
(ที่มา: <https://developers.line.biz/en/docs/messaging-api/flex-message-elements>)

2.10.4 LIFF

LIFF (Line Front-end Framework) เป็นแพลตฟอร์มสำหรับเว็บแอปที่ให้บริการโดย LINE เว็บแอปที่รันบนแพลตฟอร์มนี้เรียกว่า LIFF Apps ซึ่งหน้าตาที่แสดง LIFF Apps นั้นจะคล้ายกันกับหน้าตาที่แสดงหน้าเพจทั่วไป เพียงแต่จะแสดงบนโปรแกรม LINE เท่านั้น ไม่ใช่แสดงบนเบราว์เซอร์อื่น ๆ

ขนาดของหน้าจอ LIFF มีทั้งหมด 3 ขนาด คือ

- 1) Full: แสดงขนาด Web View 100% ของหน้าจอ
- 2) Tall: แสดงขนาด Web View 75% ของหน้าจอ
- 3) Compact: แสดงขนาด Web View 50% ของหน้าจอ



รูปที่ 2.15 หน้าจอ LIFF ในขนาดต่าง ๆ

(ที่มา: <https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcSiEPRHoTyYBswWMKYVYIs2Q3YxNaEKBCSTthEoZXcKm1csrsxf>)

2.11 BI

BI (Business Intelligence) คือกระบวนการเปลี่ยนแปลงสถาปัตยกรรมทางเทคโนโลยีที่แปลงข้อมูลดิบเป็นข้อมูลสารสนเทศ ซึ่งจะแสดงผลอยู่ในรายงานในรูปแบบต่าง ๆ ที่เหมาะสมกับการนำไปวิเคราะห์ แสดงความสัมพันธ์ และทำนายผลลัพธ์ของแนวโน้มที่อาจเกิดขึ้นได้ตรงตามความความต้องการขององค์กร เพื่อนำไปวางแผนกลยุทธ์และตัดสินใจกระบวนการดำเนินงานธุรกิจขององค์กรต่อไป

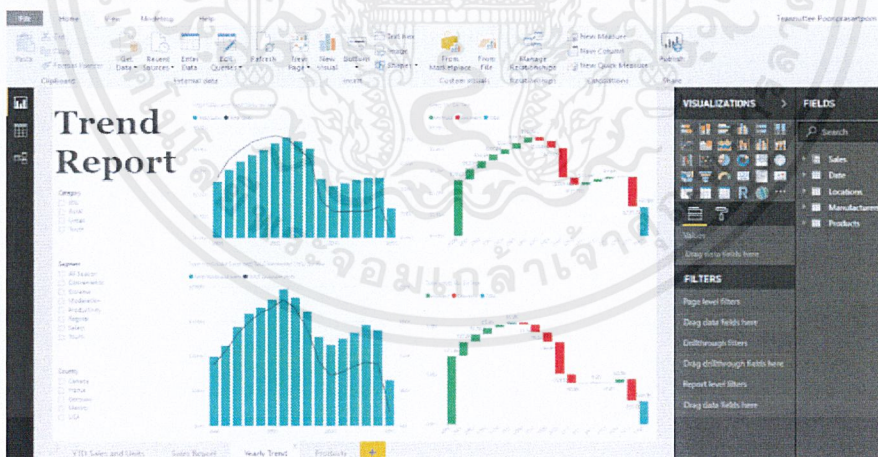
2.12 Power BI

Power BI คือโปรแกรมและบริการคลาวด์โดย Microsoft ใช้ในการค้นหาและจัดการข้อมูลเชิงลึกจากข้อมูลรวมขององค์กรเพื่อนำไปวิเคราะห์ผลทางธุรกิจและนำเสนอออกมาในรูปแบบ Interactive ทั้งแบบตาราง กราฟ แผนที่ และรูปร่างต่าง ๆ โดยสามารถแบ่งปันข้อมูลให้กับพนักงานในองค์กรและบุคคลภายนอกได้

2.12.1 ส่วนประกอบต่าง ๆ ของ Power BI

- 1) Power Query: เครื่องมือค้นหาข้อมูลและแปลงข้อมูล
- 2) Power Pivot: เครื่องมือสร้างโมเดลข้อมูลแบบตาราง
- 3) Power View: เครื่องมือสร้างภาพจากข้อมูล
- 4) Power Map: เครื่องมือสร้างภาพข้อมูลภูมิศาสตร์ 3 มิติ
- 5) Power Q&A: เครื่องมือสร้างภาพด้วยคำถาม

นอกจากนี้ Power BI ยังสามารถเชื่อมต่อแหล่งข้อมูลมากมาย ตั้งแต่ไฟล์ (Excel, PDF, SharePoint, XML), ฐานข้อมูล (SQL Server Database, ฐานข้อมูล Oracle Database, IBM databases, Amazon Redshift, Google BigQuery), BI อื่น ๆ และบริการออนไลน์ (Dynamics 365, Google Analytics, Adobe Analytics, Facebook) ได้อีกด้วย



รูปที่ 2.16 ตัวอย่างหน้ารายงานของ Power BIข

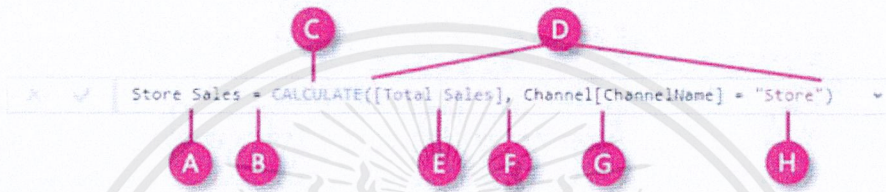
(ที่มา: <https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcQXMX-bJw4FV5kf5GWVwoTUm9s5FbedTlcmovSHhQ9ryGCrjvhi>)

2.13 DAX (Data Analysis Expression)

DAX (Data Analysis Expression) คือชุดของฟังก์ชัน ตัวแปร และการดำเนินการทางคณิตศาสตร์ที่ใช้เพื่อการคำนวณในงานที่ไม่สามารถทำได้ด้วยการคำนวณปกติและให้ค่ากลับมาอย่างน้อยหนึ่งค่า หรือกลับมาเป็นตาราง (Table) โดยสูตรของ DAX จะมีลักษณะที่คล้ายกับสูตรที่ใช้ใน Microsoft Excel เพื่อใช้ในการคำนวณและสร้างผลลัพธ์ในรูปแบบของตัวเลข, ตารางสรุป Slicer หรือไปสร้าง Visualization ต่อไป

2.13.1 โครงสร้าง DAX

โครงสร้างของ DAX จะเปลี่ยนแปลงไปตามสูตรที่ใช้ในคำนวณค่านั้น ๆ



รูปที่ 2.17 ตัวอย่างสูตรคำนวณ DAX

(ที่มา: https://docs.microsoft.com/en-us/power-bi/media/desktop-quickstart-learn-dax-basics/qsdx_4_context.png)

- ชื่อหน่วยวัด (Measure) ในตัวอย่างนี้คือ Store Sales
- เครื่องหมายเท่ากับ (=) เป็นสิ่งที่ใช้ระบุจุดเริ่มต้นของสูตร
- ชื่อของฟังก์ชัน ในตัวอย่างนี้คือ ฟังก์ชัน CALCULATE ซึ่งจะประเมินนิพจน์เป็นอาร์กิวเมนต์ในบริบทที่ปรับเปลี่ยนตามตัวกรองที่ระบุ
- วงเล็บ () ซึ่งจะล้อมรอบนิพจน์ที่ประกอบด้วยอย่างน้อยหนึ่งอาร์กิวเมนต์
- ชื่อหน่วยวัด (Measure) ในตัวอย่างนี้คือ [Total Sales] นิพจน์ในตารางเดียวกันกับ Store Sales
- เครื่องหมายจุลภาค (,) ซึ่งแยกอาร์กิวเมนต์นิพจน์แรกออกจากอาร์กิวเมนต์ตัวกรอง
- คอลัมน์อ้างอิง (Column) ในตัวอย่างนี้คือ Channel[ChannelName] แต่ละแถวในคอลัมน์นี้จะระบุช่องทาง เช่น Store หรือ Online
- บริบทตัวกรองที่เป็นค่าเฉพาะ (Filter) ในตัวอย่างนี้คือ Store

2.14 KPI (Key Performance Indicator)

KPI (Key Performance Indicator) หรือดัชนีชี้วัดความสำเร็จ คือหน่วยวัดผลความสำเร็จของการปฏิบัติงานหรือการดำเนินการที่เกิดขึ้น โดยการวัดผลจะแสดงผลลัพธ์ในรูปแบบของตัวเลข จำนวน หรือ ปริมาณที่ชัดเจน การวัดผลสามารถวัดได้ 2 ประเภทคือ การวัดผลทางตรง และการแสดงผลทางอ้อม



บทที่ 3

ขั้นตอนการดำเนินงาน

3.1 กรอบการดำเนินงาน

การดำเนินงานประกอบด้วย 3 ส่วน ดังนี้

1) การจัดทำระบบบันทึกข้อมูลจากอีเมลลงฐานข้อมูลอัตโนมัติ

การนำข้อมูลเข้าสู่ฐานข้อมูลนั้นอาศัยข้อมูลจากระบบอัตโนมัติของบริษัท โดยระบบของบริษัทจะดึงข้อมูลจากฐานข้อมูลใหญ่ของบริษัท มาเข้ารหัสข้อมูลเพื่อความปลอดภัยและจัดทำเป็นไฟล์ CSV ส่งเข้ามายังอีเมลที่กำหนด ซึ่งหน้าที่นักศึกษาที่ได้รับมอบหมายคือส่วนของการเขียนโค้ดเพื่ออ่านข้อมูลจากไฟล์ .csv ซึ่งเป็นไฟล์แนบของอีเมล และบันทึกลงฐานข้อมูลแบบเจาะจง โดยกำหนดเวลาในการอ่านและบันทึกลงในฐานข้อมูลในเวลา 09:00 น. ของทุกวัน

2) การจัดทำระบบ Line Bot

ทางบริษัทต้องการกระตุ้นยอดขายและดูแลแนวโน้มความต้องการของร้านค้าเพื่อนำไปดำเนินกลยุทธ์ทางธุรกิจต่อไป ดังนั้นบริษัทจึงมอบหมายให้ทำ LINE Bot เพื่อให้สามารถแสดงข้อมูลได้หลากหลายประเภทดังนี้

- ยอดขายรวมของซิมประเภทรายเดือนในเดือนปัจจุบัน
- ยอดขายรวมของซิมประเภทเติมเงินในเดือนปัจจุบัน
- ยอดขายแพ็คเกจเสริมรวมในเดือนปัจจุบัน
- ยอดขายซิมประเภทเติมเงินในเงื่อนไข ย้อนหลัง 6 เดือน
- เป้าหมายที่บริษัทคาดหวัง
- สิทธิพิเศษที่ร้านค้าจะได้รับเมื่อยอดขายซิมประเภทเติมเงินในเงื่อนไขถึงเป้าหมายที่กำหนด

3) การจัดทำ Dashboard โดยใช้โปรแกรม Power BI

นักศึกษาได้รับมอบหมายให้จัดทำ Dashboard โดยใช้โปรแกรม Power BI เพื่อเพิ่มประสิทธิภาพของการแสดงผลข้อมูล โดยมีการจัดทำ Dashboard แสดงยอดขาย, สิทธิพิเศษ และข้อมูลการใช้งาน Line Bot ของร้านค้า

3.2 วิธีการดำเนินงาน

3.2.1 ศึกษาข้อมูลที่ใช้ในการทำโครงการงาน

ข้อมูลเกี่ยวข้องกับโครงการนี้ประกอบด้วย

- 1) ข้อมูลของร้านค้า ได้แก่
 - a. ชื่อร้านค้า
 - b. ชื่อและนามสกุลเจ้าของร้านค้า
 - c. สถานที่ตั้งของร้านค้า ประกอบด้วย
 - i. ละติจูด
 - ii. ลองจิจูด
 - iii. เมือง
 - iv. รัฐ
 - v. ประเทศ

ซึ่งข้อมูลนี้ได้มีการปรับเปลี่ยนเพื่อความปลอดภัยของทางบริษัท

- 2) ข้อมูลการขาย ได้แก่
 - a. วันที่ขาย
 - b. จำนวนยอดขายชิมประเภทรายเดือน พิจารณาจากลูกค้าใหม่และลูกค้าที่ย้ายมาจากค่ายอื่นในแต่ละวัน
 - c. จำนวนยอดขายชิมประเภทรายเดือน พิจารณาจากลูกค้าที่ย้ายมาจากชิมประเภทเติมเงินในแต่ละวัน
 - d. จำนวนยอดขายชิมประเภทเติมเงินที่อยู่ในเงื่อนไขในแต่ละวัน โดยเงื่อนไขนั้น พิจารณาจากชิมที่ลูกค้ามีการลงทะเบียนชิมและเติมเงินเรียบร้อยแล้ว
 - e. จำนวนยอดขายชิมประเภทเติมเงินที่อยู่ในเงื่อนไขแบบพิเศษในแต่ละวัน โดยเงื่อนไขพิเศษนั้น พิจารณาจากชิมที่ลูกค้ามีการลงทะเบียนแต่ยังไม่มีการเติมเงิน

- f. จำนวนยอดขายชิมประเภทเติมเงินที่อยู่นอกเงื่อนไขในแต่ละวัน (ชิมที่ยังไม่มีการลงทะเบียนชิม)
- g. จำนวนชิมที่มีลงทะเบียนและการเติมเงินในแต่ละวัน โดยแบ่งตามราคาของการเติมเงิน ดังนี้
 - i. จำนวนชิมที่มีการเติมเงิน 20 - 49 บาท
 - ii. จำนวนชิมที่มีการเติมเงิน 50 - 99 บาท
 - iii. จำนวนชิมที่มีการเติมเงิน 100 - 299 บาท
 - iv. จำนวนชิมที่มีการเติมเงิน 300 บาทขึ้นไป
- h. เป้าหมายของยอดขายชิมในแต่ละเดือน ซึ่งยอดขายชิมนี้ต้องเป็นชิมประเภทรายเดือนที่อยู่ในเงื่อนไข
- i. จำนวนแพ็คเกจเสริมที่ขายได้ในแต่ละวัน โดยแบ่งตามราคาของแพ็คเกจเสริม ดังนี้
 - i. จำนวนแพ็คเกจเสริมที่ราคาน้อยกว่า 20 บาท
 - ii. จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 20 - 49 บาท
 - iii. จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 50 - 99 บาท
 - iv. จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 100 - 299 บาท
 - v. จำนวนแพ็คเกจเสริมที่ราคาตั้งแต่ 300 บาทขึ้นไป

3.2.2 ออกแบบและสร้างฐานข้อมูล

การสร้างฐานข้อมูลจะสร้างที่ SQL Server บน Azure VM และใช้โปรแกรม MySQL Workbench ในการจัดการฐานข้อมูล



รูปที่ 3.1 หน้าจอของ MySQL Workbench

ตารางที่ 3.1 โครงสร้างฐานข้อมูลของ shopdata (ข้อมูลของร้านค้า)

ชื่อข้อมูล	ชนิดข้อมูล	คำอธิบาย
cyp_location_code	Varchar	ชื่อร้านค้า
owner_first_name	Varchar	ชื่อเจ้าของร้านค้า
owner_last_name	Varchar	นามสกุลของเจ้าของร้านค้า
country	Varchar	ประเทศที่ตั้งของร้านค้า
state	Varchar	รัฐที่ตั้งของร้านค้า
city	Varchar	เมืองที่ตั้งของร้านค้า
longitude	Varchar	ลองจิจูดที่ตั้งของร้านค้า
latitude	Varchar	ละติจูดที่ตั้งของร้านค้า
secretID	Varchar	รหัสยืนยันตัวตนของร้านค้า
lineUserID	Varchar	รหัสประจำตัวผู้ใช้บนระบบ LINE

ตารางที่ 3.2 โครงสร้างฐานข้อมูลของ loguser (ข้อมูลการใช้งาน LINE Bot ของร้านค้า)

ชื่อข้อมูล	ชนิดข้อมูล	คำอธิบาย
u_id	Int	ลำดับข้อมูลการเข้าใช้งาน LINE Bot ของร้านค้า
user_id	Varchar	รหัสประจำตัวผู้ใช้บนระบบ LINE
user_text	Varchar	ข้อความที่ร้านค้าส่งเข้ามาใน LINE Bot
user_Replytoken	Varchar	Token ของ LINE Messaging API ที่ใช้ในการส่งข้อความตอบกลับ
user_Timestamp	Varchar	เวลาที่ร้านค้าส่งข้อความเข้ามาใน LINE Bot

ตารางที่ 3.3 โครงสร้างฐานข้อมูลของ sales (ข้อมูลการขายในแต่ละวัน)

ชื่อข้อมูล	ชนิดข้อมูล	คำอธิบาย
sales_shopName	Varchar	ชื่อร้านค้า
sales_date	Date	วันที่ขาย
sales_target	Int	เป้าหมายของยอดขายชิมในแต่ละเดือน
sales_postp_newam	Int	จำนวนยอดขายชิมประเภทรายเดือนพิจารณาจากลูกค้าใหม่ และลูกค้าที่ย้ายมาจากค่ายอื่น
sales_postp_prep	Int	จำนวนยอดขายชิมประเภทรายเดือนพิจารณาจากลูกค้าที่ย้ายมาจากชิมประเภทเติมเงิน
sales_piact	Int	จำนวนยอดขายชิมประเภทเติมเงิน
sales_piact_incon	Int	จำนวนยอดขายชิมประเภทเติมเงินที่อยู่ในเงื่อนไข
sales_piact_outcon	Int	จำนวนยอดขายชิมประเภทเติมเงินที่อยู่นอกเงื่อนไข
sales_piact_inconspe	Int	จำนวนยอดขายชิมประเภทเติมเงินที่อยู่ในเงื่อนไขแบบพิเศษ
sales_topupmt20	Int	จำนวนชิมที่มีการเติมเงิน 20 - 49 บาท
sales_topupmt50	Int	จำนวนชิมที่มีการเติมเงิน 50 99 บาท
sales_topupmt100	Int	จำนวนชิมที่มีการเติมเงิน 100 - 299 บาท
sales_topupmt300	Int	จำนวนชิมที่มีการเติมเงิน 300 บาทขึ้นไป
sales_vaslt20	Int	จำนวนแพ็คเกจเสริมที่ราคาน้อยกว่า 20 บาท
sales_vasmt20	Int	จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 20 - 49 บาท
sales_vasmt50	Int	จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 50 -99 บาท
sales_vasmt100	Int	จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 100 - 299 บาท
sales_vasmt300	Int	จำนวนแพ็คเกจเสริมที่ราคาตั้งแต่ 300 บาทขึ้นไป

ตารางที่ 3.4 โครงสร้างฐานข้อมูลของ privilege (ข้อมูลสิทธิพิเศษที่ร้านค้าเลือก)

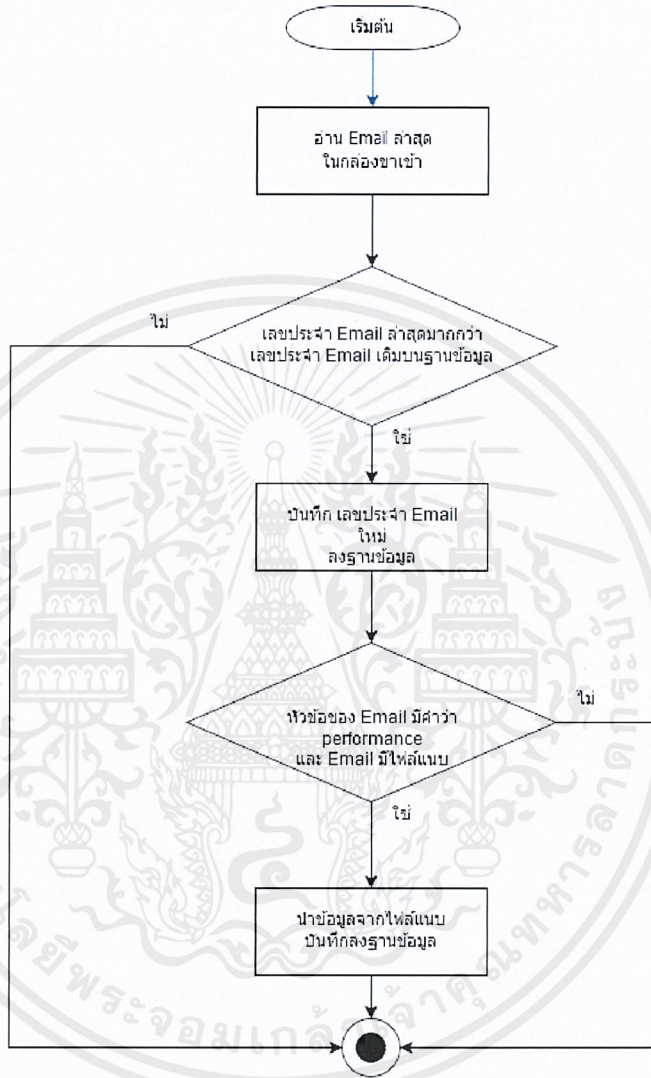
ชื่อข้อมูล	ชนิดข้อมูล	คำอธิบาย
priv_id	Varchar	ลำดับข้อมูลการกรอกรับสิทธิพิเศษของร้านค้า
priv_UserID	Varchar	รหัสประจำตัวผู้ใช้บนระบบ Line
priv_ShopName	Varchar	ชื่อของร้านค้า
priv_UserRequest	Varchar	สิทธิที่ร้านค้าเลือกกรอกรับ
priv_RequestTime	Varchar	เวลาที่ร้านค้ากรอกรับสิทธิพิเศษ
priv_Status	Varchar	สถานะของสิทธิพิเศษ ประกอบด้วย 2 กรณี คือ Y: สิทธิพิเศษที่ร้านค้าเลือกล่าสุดในเดือนนั้น N: สิทธิพิเศษที่ร้านค้าที่ร้านค้าเคยเลือก และหลังจากนั้นได้มีการเปลี่ยนแปลงสิทธิพิเศษไปเป็นสิทธิพิเศษใหม่ ภายในเดือนนั้น

ตารางที่ 3.5 โครงสร้างฐานข้อมูลของ dataforsystems (ข้อมูลที่ใช้ในการพัฒนาระบบ)

ชื่อข้อมูล	ชนิดข้อมูล	คำอธิบาย
list	Varchar	รายการการเก็บข้อมูล
numberemail	Int	หมายเลขของ E-mail ล่าสุด
useriddataforgraph	Varchar	รหัสประจำตัวผู้ใช้บนระบบ Line
pictureurl	Varchar	URL ของรูปภาพที่ใช้แสดงสิทธิพิเศษ

3.2.3 การสร้างระบบบันทึกข้อมูลจากอีเมลลงฐานข้อมูลอัตโนมัติ

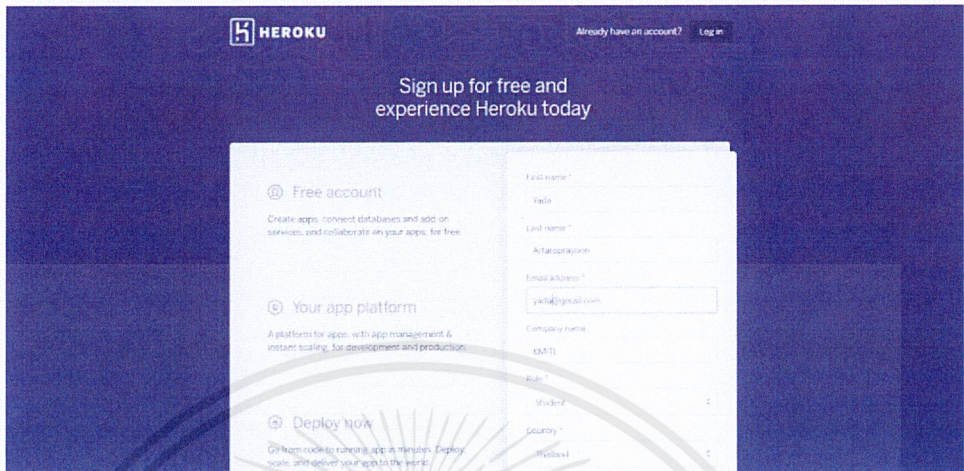
3.2.3.1 flowchart การทำงานของระบบ



รูปที่ 3.2 flowchart การทำงานระบบบันทึกข้อมูลจากอีเมลลงฐานข้อมูล

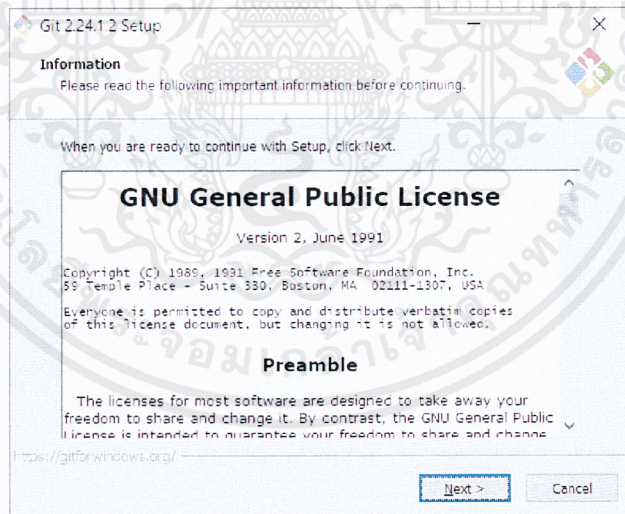
3.2.3.2 การพัฒนาระบบบันทึกข้อมูลจากอีเมลลงฐานข้อมูลอัตโนมัติ

- 1) สมัครเข้าใช้งานบัญชี Heroku ที่ <https://signup.heroku.com>



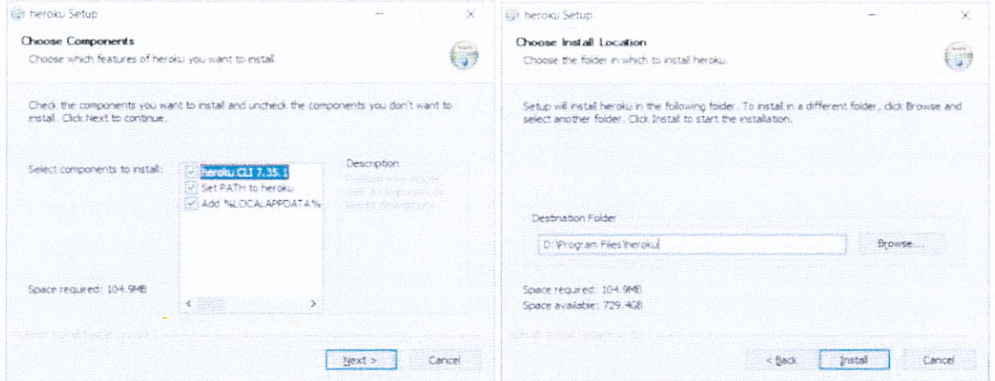
รูปที่ 3.3 หน้าจอสมัครเข้าใช้งานบัญชี Heroku

- 2) ติดตั้ง Git เพื่อใช้ในการจัดการไฟล์บน Heroku
ดาวน์โหลด Git จาก <https://git-scm.com/download> และติดตั้งลงบนเครื่องคอมพิวเตอร์



รูปที่ 3.4 หน้าจอการติดตั้ง Git

- 3) ติดตั้ง Heroku Command Line Interface (CLI) เพื่อใช้จัดการแอปพลิเคชันและส่วนเสริม โดย ดาวน์โหลด Heroku Command Line Interface ที่ <https://devcenter.heroku.com/articles/getting-started-with-php#set-up> และติดตั้งลงบนเครื่องคอมพิวเตอร์



รูปที่ 3.5 การติดตั้ง Heroku Command Line Interface

4) สร้างแอปพลิเคชันผ่าน Heroku Command Line Interface

```
D:\herokufolder\project1>git clone https://github.com/heroku/php-getting-started.git
Cloning into 'php-getting-started'...
remote: Enumerating objects: 189, done.
remote: Total 189 (delta 0), reused 0 (delta 0), pack-reused 189
Receiving objects: 100% (189/189), 49.79 KiB | 3.00 KiB/s, done.
Resolving deltas: 100% (77/77), done.

D:\herokufolder\project1>heroku login
» Warning: heroku update available from 7.33.0 to 7.33.3.
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/browser/84fb0f66-6cbc-4091-9a8d-ff016c443886
Logging in... done
Logged in as yada39nco@gmail.com

D:\herokufolder\project1>heroku create
» Warning: heroku update available from 7.33.0 to 7.33.3.
Creating app... done, ⬢ peaceful-brook-75249
https://peaceful-brook-75249.herokuapp.com/ | https://git.heroku.com/peaceful-brook-75249.git

D:\herokufolder\project1>git push heroku master
```

รูปที่ 3.6 การสร้างแอปพลิเคชันบน Heroku

จากรูปที่ 3.6 สามารถใช้คำสั่งการสร้าง Application บน Heroku ได้ดังนี้

- คำสั่ง git clone https://github.com/heroku/php-getting-started.git คือ คำสั่งที่ใช้คัดลอก Repository เบื้องต้นที่ใช้ในการเขียนคำสั่งภาษา PHP มาลงที่เครื่องคอมพิวเตอร์
- คำสั่ง heroku login คือคำสั่งที่ใช้ในการลงชื่อเข้าใช้ Heroku Server
- คำสั่ง heroku create คือคำสั่งที่ใช้ในการสร้างแอปพลิเคชันบน Heroku Server
- คำสั่ง git push heroku master คือการส่งไฟล์ที่เกี่ยวข้องอยู่บน Local Repository ขึ้นไปยัง Remote Repository

5) ติดตั้ง Composer ของ IMAP ผ่าน Heroku Command Line Interface โดยใช้ คำสั่ง composer require php-imap/php-imap

6) เขียนโค้ด และติดตั้งโค้ดบน Heroku Server

```
C:\Users\DELL> heroku login
» Warning: heroku update available from 7.33.0 to 7.33.3.
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/browser/a5545a31-73c5-4ee4-80d4-8132e74416c6
Logging in... done
Logged in as yada356co@gmail.com

C:\Users\DELL> cd D:\herokufolder\project1\php-getting-started

C:\Users\DELL>

D:\herokufolder\project1\php-getting-started> heroku git:remote -a peaceful-brook-75249
» Warning: heroku update available from 7.33.0 to 7.33.3.
set git remote heroku to https://git.heroku.com/peaceful-brook-75249.git

D:\herokufolder\project1\php-getting-started> git add .

D:\herokufolder\project1\php-getting-started> git commit -am "upload file"
[master 41a18a5] upload file
1 file changed, 7 insertions(+), 25 deletions(-)
rewrite web/index.php (91%)

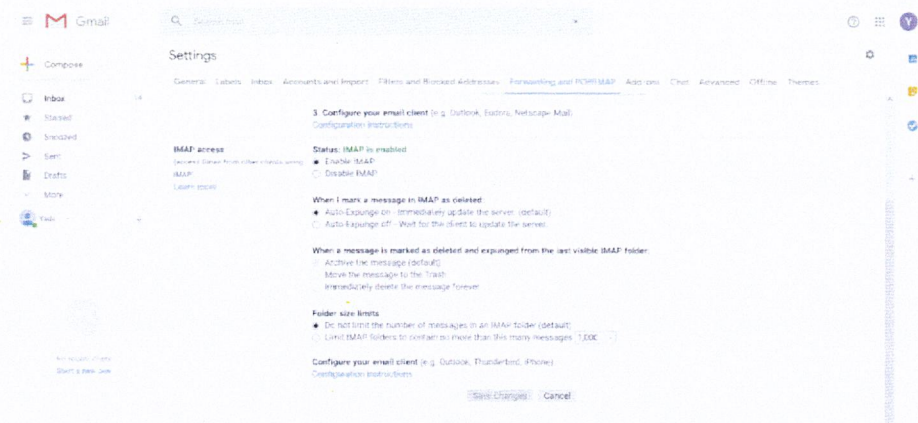
D:\herokufolder\project1\php-getting-started> git push heroku master
Enumerating objects: 182, done.
Counting objects: 100% (182/182), done.
Delta compression used up to 8.14Mk
```

รูปที่ 3.7 การส่งไฟล์จากคอมพิวเตอร์ ไปยัง Heroku Server

จากรูปที่ 3.7 สามารถใช้คำสั่งการส่งไฟล์จากคอมพิวเตอร์ ไปยัง Heroku Server ดังนี้

- คำสั่ง heroku login คือ คำสั่งที่ใช้ในการลงชื่อเข้าใช้ Heroku Server
- คำสั่ง heroku git:remote -a peaceful-brook-75249 คือการกำหนดปลายทางไปยังแอปพลิเคชันบน Heroku Server (Remote Repository)
- คำสั่ง git add . คือการเพิ่มไฟล์ทั้งหมดที่อยู่ในโฟลเดอร์การทำงาน เข้าไปที่สถานะ Staged
- คำสั่ง git commit -am "upload file" คือการยืนยันไฟล์ที่อยู่ในสถานะ Staged เข้าไปเก็บไว้ที่ Local Repository
- คำสั่ง git push heroku master คือการส่งไฟล์ที่เก็บอยู่บน Local Repository ขึ้นไปยัง Remote Repository

- 7) เปิดการใช้งาน IMAP บนอีเมลที่ใช้รับข้อมูลโดยตั้งค่าที่ Forwarding and POP/IMAP ซึ่งอยู่ในส่วนของหน้า Settings ของอีเมลที่ใช้รับข้อมูล



รูปที่ 3.8 การเปิดการใช้งาน IMAP

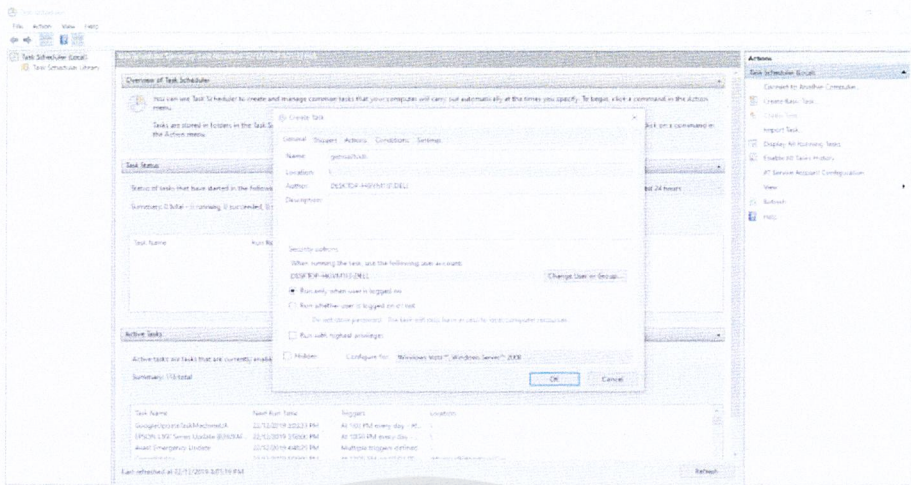
จากรูปที่ 3.8 ทำการเปิดการใช้งาน IMAP โดยกดเลือก Enable IMAP เพื่อเป็นการเปิดใช้งาน กดเลือก Auto-Expunge on - Immediately update the server. (default) เพื่อทำให้การล้างข้อมูลอัตโนมัติทำงานเมื่อผู้ใช้ทำเครื่องหมายข้อความใน IMAP ว่าลบ และกดเลือก Auto-Expunge on - Immediately update the server. (default) เพื่อตั้งค่าให้จำนวนข้อความในโฟลเดอร์ IMAP สามารถมีได้ไม่จำกัด

- 8) ตั้งค่าเวลาที่ใช้ในการรันไฟล์สคริปต์ PHP นำข้อมูลจากไฟล์แนบของอีเมลลงฐานข้อมูลโดยโค้ดที่ใช้ในการรันไฟล์สคริปต์ PHP จะอยู่ถูกเขียนในไฟล์ Batch หลังจากนั้นนำไฟล์ Batch ไปตั้งค่าเวลาให้ทำงานบนเครื่องคอมพิวเตอร์แม่ข่ายของบริษัทโดยใช้โปรแกรม Task Scheduler บนระบบปฏิบัติการ Windows

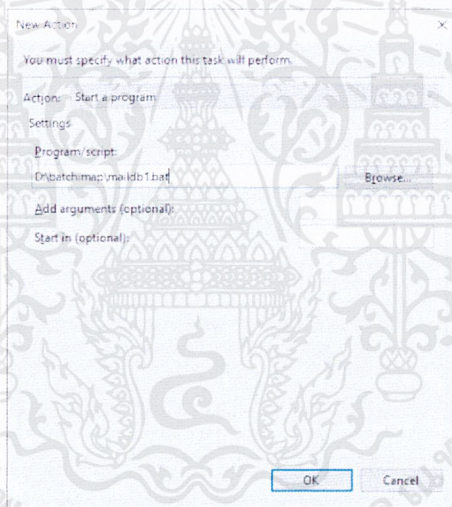
```
1 START "C:\Program Files\Internet Explorer\iexplore.exe" "https://peachful234.herokuapp.com/getemailtodb.php"
2 TIMEOUT 200
3 taskkill /f /im iexplore.exe
4 EXIT
```

รูปที่ 3.9 โค้ดที่ใช้ในการรันไฟล์สคริปต์ PHP

จากรูปที่ 3.9 คำสั่งในบรรทัดแรก คือคำสั่งที่ใช้ในการเปิด <https://peachful234.herokuapp.com/getemailtodb.php> ซึ่งเป็นไฟล์สคริปต์คำสั่งภาษา PHP บนโปรแกรม Internet Explorer คำสั่งในบรรทัดที่ 2 คือคำสั่งที่ใช้ในการกำหนดเวลาการทำงานของสคริปต์ 200 วินาที คำสั่งในบรรทัดที่ 3 คือคำสั่งที่ใช้ในการปิดโปรแกรม Internet Explorer และคำสั่งในบรรทัดที่ 4 คือ คำสั่งที่ใช้ในการปิดการทำงานของไฟล์ Batch นี้

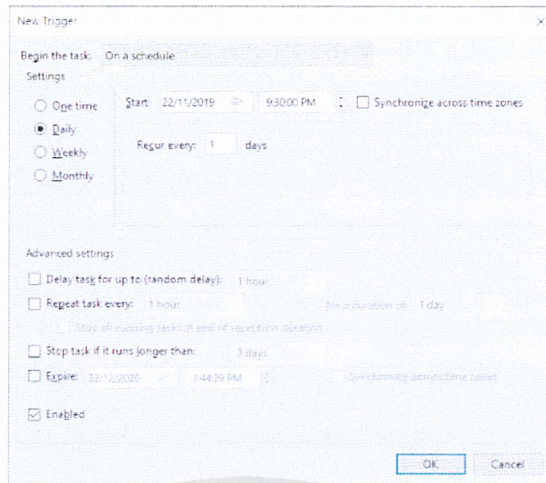


รูปที่ 3.10 หน้าจอที่ใช้ในการสร้าง Task บนโปรแกรม Task Scheduler



รูปที่ 3.11 หน้าจอของแถบ New Action

จากรูปที่ 3.10 ทำการสร้าง Task ได้โดยการกด Create Task บริเวณแถบด้านขวามือ โปรแกรมจะแสดงหน้าต่าง Create Task ขึ้น หลังจากนั้นทำการใส่ชื่อของ Task ที่ช่อง Name ในแถบ General และกำหนดไฟล์ Batch ที่ต้องการให้ทำงานในแถบ Action ดังรูปที่ 3.11

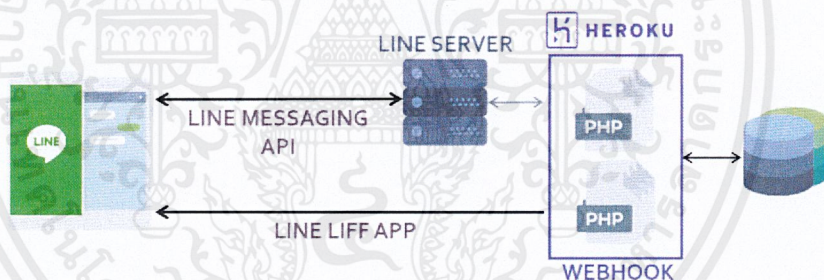


รูปที่ 3.12 หน้าจอของแถบ New Trigger

จากรูปที่ 3.12 ทำการกำหนดวันที่และเวลาที่ต้องการให้ไฟล์ Batch ทำงานในแถบ Trigger

3.2.4 การจัดทำระบบ LINE Bot

3.2.4.1 โครงสร้างการทำงานของ LINE Bot



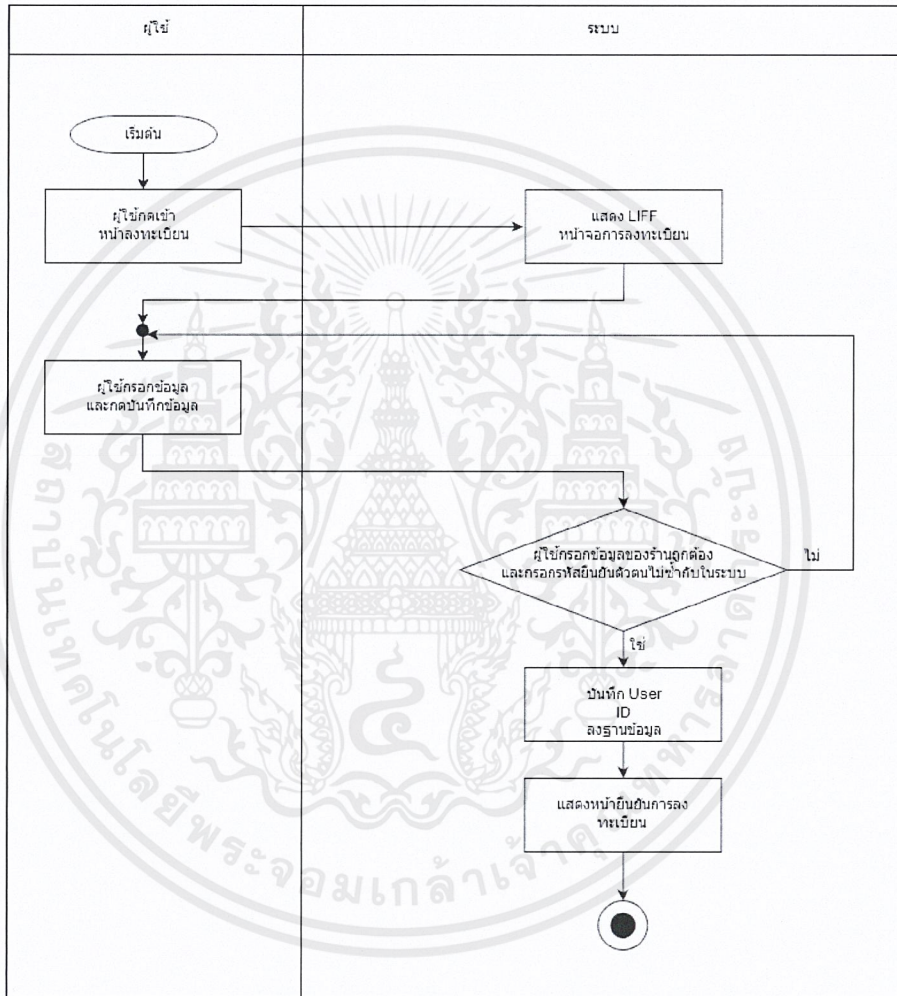
รูปที่ 3.13 โครงสร้างการทำงานของ LINE Bot

จากรูปที่ 3.13 เมื่อผู้ใช้มีการส่งข้อความมายัง Bot Application แพลตฟอร์มบน Bot Application จะส่ง Webhook Event (เหตุการณ์ต่าง ๆ ที่เกิดขึ้นกับ Line Bot) ไปยัง LINE Server และ LINE Server จะทำการส่งข้อมูลไปยัง Webhook โดยจะติดต่อสื่อสารกันโดยใช้ Line Messaging API เมื่อ Webhook ประมวลผลเรียบร้อยแล้ว จะตอบกลับผู้ใช้งานผ่านแพลตฟอร์มของ Bot Application

เมื่อผู้ใช้มีการส่งข้อความมายัง Bot Application โดยข้อความที่ส่งมานั้นมีการตอบกลับเป็นหน้าเว็บไซต์ที่อยู่นอกเหนือระบบของ LINE Server แพลตฟอร์มบน Bot

Application จะแสดงผลหน้าเว็บไซต์นั้นในรูปแบบของ LIFF (LINE Front-End Framework)

3.2.4.2 Activity Diagram การทำงานของระบบ



รูปที่ 3.14 Activity Diagram ของการลงชื่อเข้าใช้งาน LINE Bot

3.2.4.3 ขั้นตอนการพัฒนาระบบ

1) สร้างแอปพลิเคชันผ่าน Heroku Command Line Interface

```
D:\herokufolder\project2>heroku login
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/browser/2df638ea-5738-40f8-bfa8-a63fe3538f19
Logging in... done
Logged in as yada356co@gmail.com

D:\herokufolder\project2>heroku create
Creating app... done, ☑ sleepy-ocean-02437
https://sleepy-ocean-02437.herokuapp.com/ | https://git.heroku.com/sleepy-ocean-02437.git

D:\herokufolder\project2>git clone https://github.com/line/line-bot-sdk-php
Cloning into 'line-bot-sdk-php'...
remote: Enumerating objects: 34, done.
remote: Counting objects: 100% (34/34), done.
remote: Compressing objects: 100% (28/28), done.

Receiving objects: 100% (3716/3716), 3.27 MiB | 1.01 MiB/s, done.
Resolving deltas: 100% (2610/2610), done.

D:\herokufolder\project2>git push heroku master
```

รูปที่ 3.16 การสร้างแอปพลิเคชัน สำหรับ Line bot บน Heroku

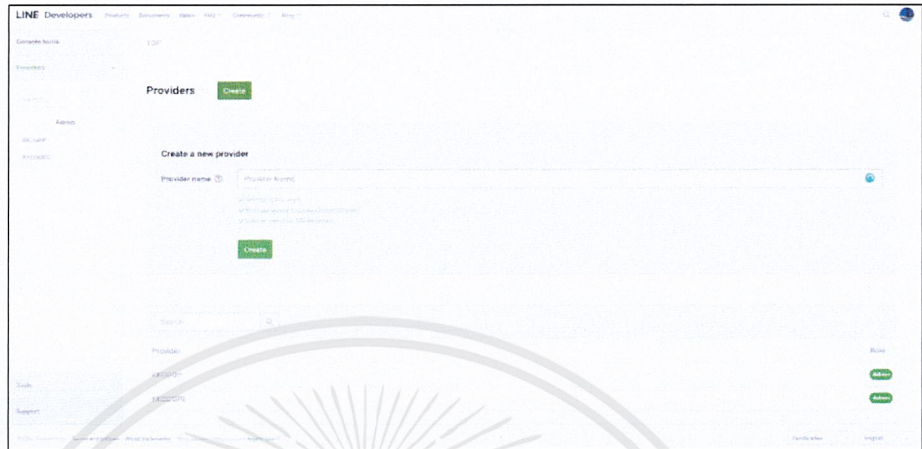
จากรูปที่ 3.16 สามารถอธิบายคำสั่งได้ดังนี้

- คำสั่ง heroku login คือคำสั่งที่ใช้ในการลงชื่อเข้าใช้ Heroku Server
- คำสั่ง heroku create คือคำสั่งที่ใช้ในการสร้างแอปพลิเคชันบน Heroku Server
- คำสั่ง git clone https://github.com/line/line-bot-sdk-php คือคำสั่งที่ใช้คัดลอก Repository เบื้องต้นที่ใช้ในการเขียนคำสั่งภาษา PHP สำหรับ LINE Bot มาลงที่เครื่องคอมพิวเตอร์
- คำสั่ง git push heroku master คือการส่งไฟล์ที่เก็บอยู่บน Local Repository ขึ้นไปยัง Remote Repository

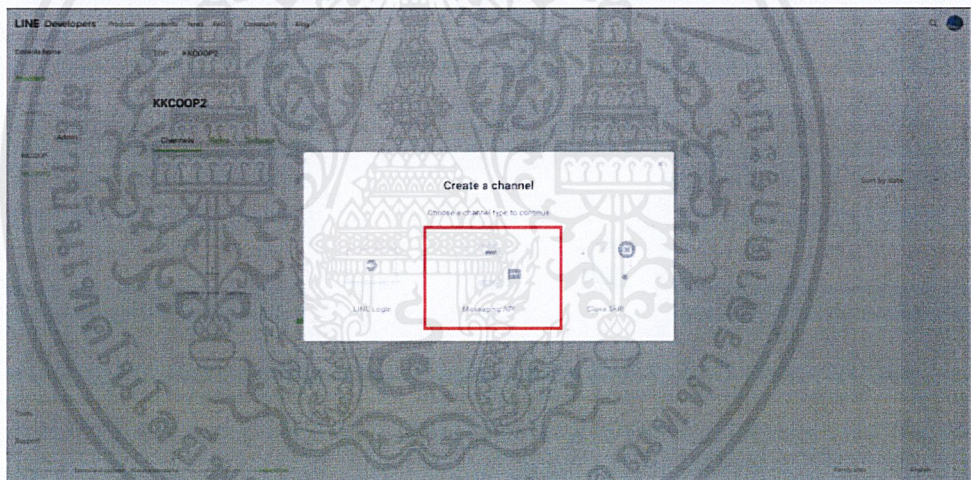
2) ติดตั้ง Composer ของ LINE Bot ผ่าน Heroku Command Line Interface โดยใช้คำสั่ง composer require linecorp/line-bot-sdk

3) สร้าง LINE Channel ที่ <https://developers.line.biz/consol>

เริ่มต้นด้วยการกรอก Provider Name ดังรูปที่ 3.17



รูปที่ 3.17 หน้าจอ <https://developers.line.biz/consol>



รูปที่ 3.18 การสร้างแชนเนล

จากรูปที่ 3.18 ทำการสร้างแชนเนล โดยเลือกที่ Messaging API และกรอกข้อมูลทั่วไปเกี่ยวกับแชนเนล ซึ่งได้แก่ ชื่อของแชนเนล คำอธิบายแชนเนล ประเภทของแชนเนล และอีเมลที่ใช้ติดตามข้อมูลของแชนเนลนี้

เมื่อทำการสร้างแชนเนลเรียบร้อยแล้ว จะได้รับ Channel secret และ Channel access token สำหรับใช้ในกระบวนการเขียนโค้ด

4) เขียนโค้ด และติดตั้งโค้ดบน Heroku Server

```
1 <?php
2 $API_URL = 'https://api.line.me/v2/bot/message';
3 $ACCESS_TOKEN = '12gv1u10y267956/D3c4q/xoZZeb68Dkm6j/1l';
4 $channelSecret = 'c7efd525d7e7c99dbf8530798tyd9146';
5 $POST_HEADER = array('Content-Type: application/json', 'Authorization: Bearer ' . $ACCESS_TOKEN);
6
7 $request = file_get_contents('php://input');
8 $request_array = json_decode($request,true);
9
10 $replyTokenn = $request_array["events"][0]["replyToken"];
11 $userID = $request_array["events"][0]["source"]["userId"];
12 $userText = $request_array["events"][0]["message"]["text"];
13 $timestamp = $request_array["events"][0]["timestamp"];
14
15 $replyText["type"] = "text";
16 $replyText["text"] = "ข้อความตอบกลับ";
17
18 $replyarrayJson["replyToken"] = $replyTokenn;
19 $replyarrayJson["messages"][0] = $replyText;
20
21 $post_body = json_encode($replyarrayJson,JSON_UNESCAPED_UNICODE);
22 $send_result = send_reply_message($API_URL.'/reply', $POST_HEADER, $post_body);
23 echo $send_result."\r\n";
24
25 function send_reply_message($url, $post_header, $post_body)
26 {
27     $ch = curl_init($url);
28     curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "POST");
29     curl_setopt($ch, CURLOPT_HTTPHEADER, $post_header);
30     curl_setopt($ch, CURLOPT_POSTFIELDS, $post_body);
31     curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
32     $result = curl_exec($ch);
33     curl_close($ch);
34
35     return $result;
36 }
```

รูปที่ 3.19 ตัวอย่างโค้ดที่ใช้ในการพัฒนา LINE Bot

จากรูปที่ 3.19 มีคำสั่งดังนี้ ในบรรทัดที่ 1 คือคำสั่งที่ใช้ระบุว่าจะโค้ดส่วนใดที่เป็นภาษา PHP ในบรรทัดที่ 2 - 5 เป็นการกำหนดตัวแปรที่ใช้สื่อสารกับ LINE Server ประกอบด้วย URL ที่ใช้ในการตอบกลับ, Channel secret, Channel access token และประเภทของเนื้อหาที่ใช้ในการตอบกลับ ในบรรทัดที่ 7 - 8 คือคำสั่งที่ใช้รับข้อความที่ผู้ใช้ส่งเข้ามาทาง Line Bot และแปลงจากข้อมูลรูปแบบ JSON ให้อยู่ในรูปแบบสตริงบนภาษา PHP ในบรรทัดที่ 10 - 13 คือคำสั่งที่ใช้กำหนดตัวแปรให้กับสตริงที่ระบบได้รับจากการที่ผู้ใช้ส่งเข้ามาทาง Line Bot ในบรรทัดที่ 15 - 19 คือคำสั่งที่ใช้ในการกำหนดรูปแบบข้อความเพื่อตอบกลับไปยังผู้ใช้ ในบรรทัดที่ 21 - 36 คือคำสั่งที่ใช้ในการแปลงข้อมูลในรูปแบบสตริงให้อยู่ในรูปแบบ JSON และตอบกลับไปยังผู้ใช้

```

D:\herokufolder\project2\line-bot-sdk-php>heroku login
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/browser/402ffc03-ec83-4f06-ba24-11ec86983cfa
Logging in... done
Logged in as yada356co@gmail.com

D:\herokufolder\project2\line-bot-sdk-php>heroku git:remote -a sleepy-ocean-02437
set git remote heroku to https://git.heroku.com/sleepy-ocean-02437.git

D:\herokufolder\project2\line-bot-sdk-php>git add .

D:\herokufolder\project2\line-bot-sdk-php>git commit -am "upload file"
[master 33d1ff1] upload file
1 file changed, 2 insertions(+), 1 deletion(-)

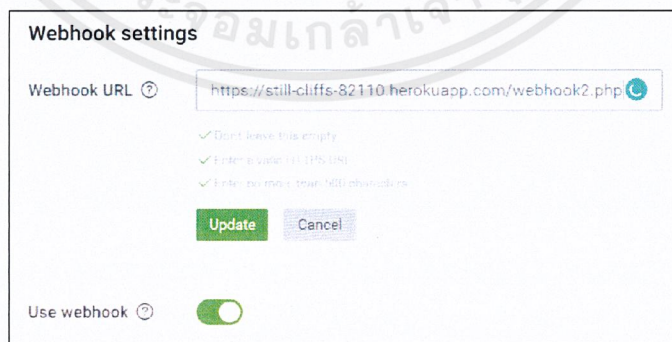
D:\herokufolder\project2\line-bot-sdk-php>git push heroku master

```

รูปที่ 3.20 การส่งไฟล์จากคอมพิวเตอร์ ไปยัง Heroku Server

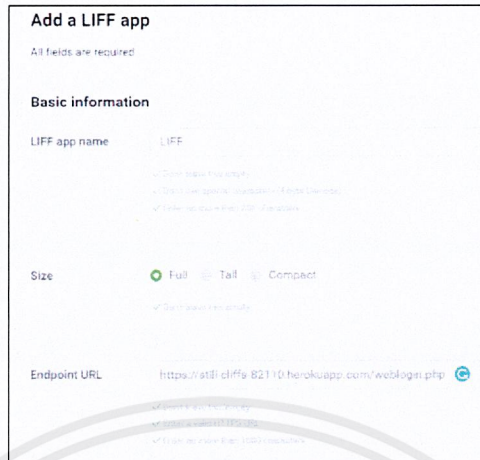
จากรูปที่ 3.20 สามารถใช้คำสั่งการส่งไฟล์จากคอมพิวเตอร์ ไปยัง Heroku Server ดังนี้

- คำสั่ง heroku login คือ คำสั่งที่ใช้ในการลงชื่อเข้าใช้ Heroku Server
 - คำสั่ง heroku git:remote -a peaceful-brook-75249 คือการกำหนดปลายทางไปยังแอปพลิเคชันบน Heroku Server (Remote Repository)
 - คำสั่ง git add . คือการเพิ่มไฟล์ทั้งหมดที่อยู่ในโฟลเดอร์การทำงาน เข้าไปที่สถานะ Staged
 - คำสั่ง git commit -am "upload file" คือการยืนยันไฟล์ที่อยู่ในสถานะ Staged เข้าไปเก็บไว้ที่ Local Repository
 - คำสั่ง git push heroku master คือการส่งไฟล์ที่เก็บอยู่บน Local Repository ขึ้นไปยัง Remote Repository
- 5) กำหนด Webhook URL สำหรับแชนเนล เพื่อใช้ในการเชื่อมต่อและสื่อสารข้อมูลระหว่าง LINE Server กับ Heroku Server



รูปที่ 3.21 การกำหนด Webhook URL

6) กำหนด LIFF App สำหรับแชทเพื่อใช้แสดง WebView บน Line Bot



รูปที่ 3.22 การกำหนด LIFF App

จากรูปที่ 3.22 กำหนด LIFF App โดยการกรอกชื่อของ LIFF App กำหนดขนาดในการแสดงผล และกำหนด URL ของเว็บไซต์ที่ต้องการให้แสดงบน LINE Bot

7) กำหนด Rich Menu ที่ LINE Official Account Manager เพื่อใช้ในการแสดงแถบเมนูลัดใน LINE Bot

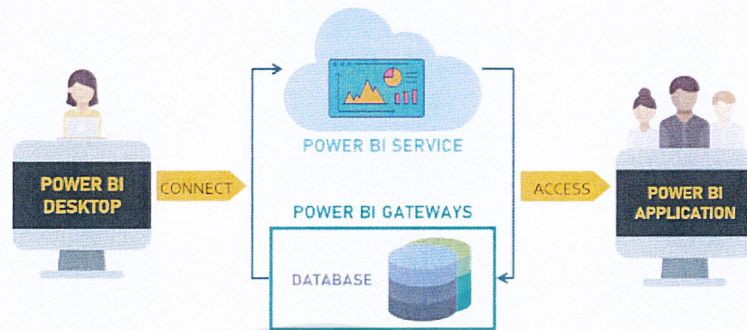


รูปที่ 3.23 การกำหนด Rich Menu

จากรูปที่ 3.23 กำหนด Rich Menu โดยการกรอกชื่อของ Rich Menu, กำหนดช่วงเวลาในการเปิด/ปิดเมนู, กำหนดข้อความบนเมนูบาร์ เทมเพลต และข้อความตอบกลับเมื่อผู้ใช้งานมีการกดเมนู

3.2.5 การจัดทำแดชบอร์ดรายงานข้อมูล โดยใช้โปรแกรม Power BI

3.2.5.1 โครงสร้างการทำงานของระบบสารสนเทศเพื่อการแสดงข้อมูลบน Power BI



รูปที่ 3.24 ภาพรวมของการพัฒนาระบบสารสนเทศเพื่อการแสดงข้อมูลของบริษัทแอดวานซ์ ไวร์เลส เน็ตเวิร์ค จำกัด โดยใช้โปรแกรม Power BI

การพัฒนาระบบสารสนเทศเพื่อการแสดงข้อมูลของบริษัท แอดวานซ์ ไวร์เลส เน็ตเวิร์ค จำกัด โดยใช้โปรแกรม Power BI แบ่งออกได้เป็น 4 ส่วน คือ

- 1) Administrator: เป็นส่วนของผู้พัฒนา มีหน้าที่ในการสร้างแดชบอร์ดตามความต้องการของผู้ใช้งาน ซึ่งในการพัฒนาระบบนี้ได้ใช้ Power BI Desktop ในการทำงาน
- 2) Power BI service: เป็นบริการของ Power BI ในรูปแบบของ Software as a Service บนคลาวด์ ทำหน้าที่เก็บข้อมูลการทำงานของระบบ ทำให้ผู้ใช้สามารถเรียกใช้งานระบบได้ตลอดเวลา
- 3) Power BI gateway: เป็นส่วนที่ทำหน้าที่เป็นสื่อกลางในการเชื่อมต่อข้อมูลระหว่าง Power service, Power BI Desktop และฐานข้อมูล
- 4) User: เป็นส่วนของผู้ใช้งาน โดยสามารถเรียกใช้ระบบได้ด้วย Microsoft Power BI application บนคอมพิวเตอร์

sales_shop_name	sales_date	sales_target	sales_postp_newsum	sales_postp_prep	sales_p
wp20wp3CmKk	Wednesday, 27 November 2019	80	0	0	0
wp20wp3CmKk	Wednesday, 27 November 2019	80	0	0	0
wp20wp3CmKk	Wednesday, 27 November 2019	80	0	0	0

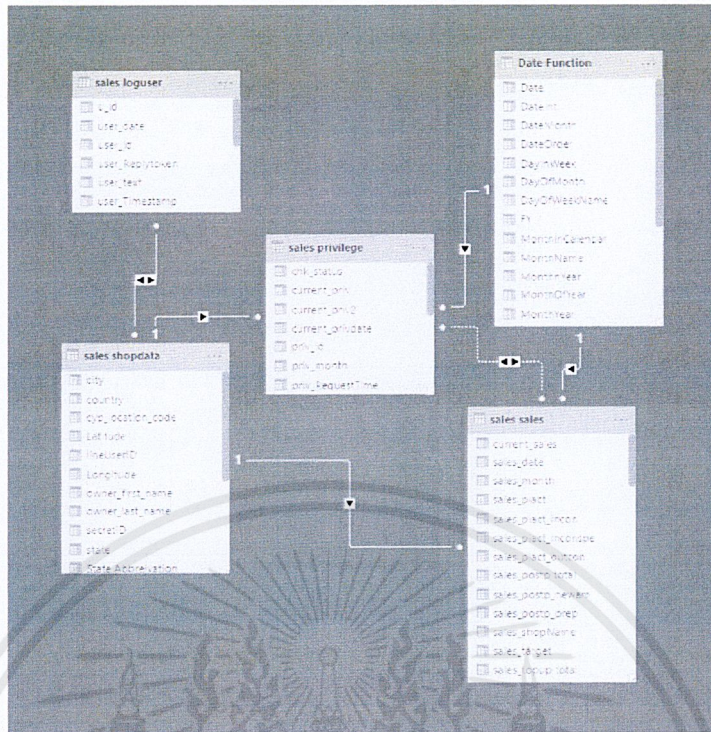
รูปที่ 3.27 หน้าจอในส่วนของ Panel: Data View

- 1 current_month_piact_thismonth = CALCULATE(SUM('sales sales'[sales_piact]),
- 2 FILTER('Date Function', MONTH('Date Function'[MonthInCalendar])=MONTH(TODAY()-0))

รูปที่ 3.28 ตัวอย่างภาษา DAX

- 3) สร้างความสัมพันธ์ระหว่างข้อมูลจากตารางต่าง ๆ ที่ Panel: Model View

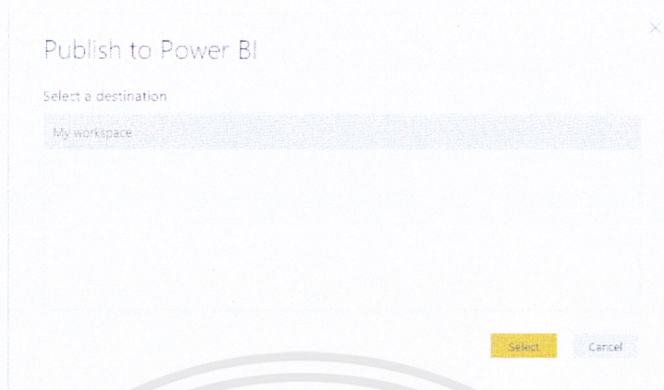
รูปที่ 3.29 หน้าต่างแก้ไขความสัมพันธ์ระหว่างข้อมูล



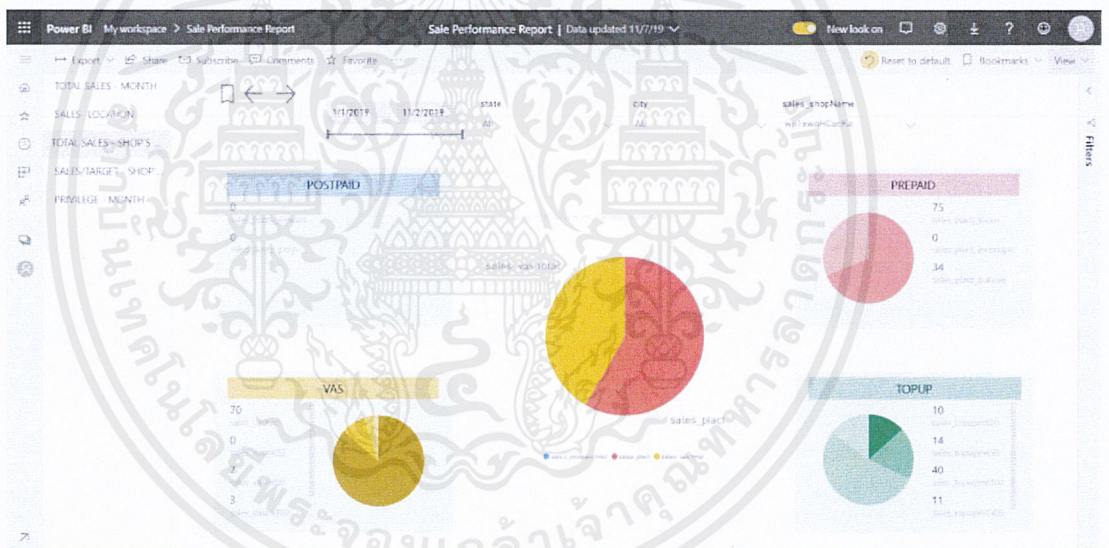
รูปที่ 3.30 Entity Relationship Diagram บน Panel: Model View

- 4) จัดทำแดชบอร์ดด้วย Visualizations Plane, Bookmark, Plane, Filter Plane, Fields บน Panel: Model View และภาษา DAX รายงานผล 6 รูปแบบ ดังนี้
- รายงานเปรียบเทียบยอดขายประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน
 - รายงานยอดขายแบ่งตามสถานที่ตั้ง
 - รายงานยอดขายในภาพรวมตามเวลาที่กำหนดแบ่งตามชื่อร้าน
 - รายงานยอดขายทุกประเภท แสดงผลรายปี ไตรมาส เดือน และวัน รวมถึงแสดงเป้าหมายแบ่งตามชื่อร้าน
 - รายงานการเลือกสิทธิของร้านต่าง ๆ แบ่งตามเวลา
 - รายงานการใช้งาน LINE Bot ของร้านค้า

5) เผยแพร่รายงาน จาก Power BI Desktop ไปยัง Power BI cloud service



รูปที่ 3.31 การเผยแพร่ไปยัง Power BI cloud service



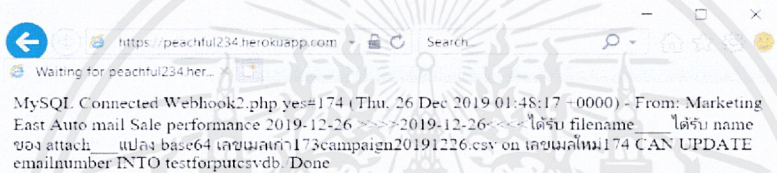
รูปที่ 3.32 การแสดงผลรายงานบน Power BI cloud service

บทที่ 4

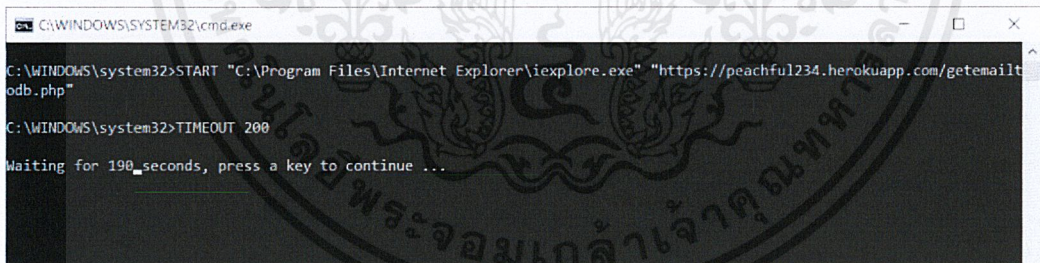
การทำงานของระบบรายงาน KPI

4.1 ระบบบันทึกข้อมูลจากอีเมลลงฐานข้อมูลอัตโนมัติ

ไฟล์สคริปต์ (Batch File) จะทำงานโดยมีการเปิดหน้าเว็บ PHP บนโปรแกรม Internet Explorer ดังรูปที่ 4.27 เพื่อทำการบันทึกข้อมูลจากไฟล์แนบอีเมลลงฐานข้อมูล ตามที่มีการตั้งค่าเอาไว้ที่โปรแกรม Task Scheduler บนเครื่องคอมพิวเตอร์แม่ข่าย ดังรูปที่ 4.28



รูปที่ 4.1 การเปิดหน้าเว็บ PHP บนโปรแกรม Internet Explorer



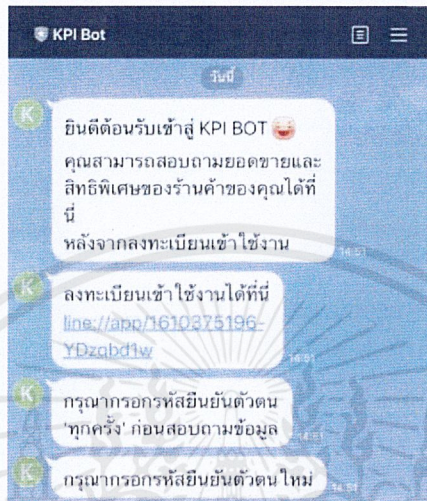
รูปที่ 4.2 หน้าจอการทำงาน Batch File

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ระบบ LINE Bot

4.2.1 การลงทะเบียนเข้าใช้งาน LINE Bot

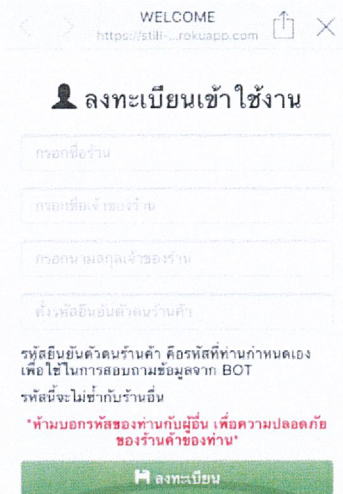
- 1) เมื่อผู้ใช้ทำการเพิ่ม LINE Bot เป็นเพื่อน ทาง LINE Bot จะส่งข้อความมายังผู้ใช้ ดังรูปที่ 4.3



รูปที่ 4.3 หน้าจอเมื่อผู้ใช้ทำการเพิ่ม LINE Bot เป็นเพื่อน

ก่อนที่ผู้ใช้จะสามารถส่งข้อความเพื่อสอบถามข้อมูลต่าง ๆ ได้นั้น จะต้องคลิกไปยัง <line://app/1610375196-YDzqbd1w> เพื่อทำการลงทะเบียนเข้าใช้งาน

- 2) เมื่อผู้ใช้เข้าสู่หน้าจอบน <line://app/1610375196-YDzqbd1w> จะต้องใส่ข้อมูลต่าง ๆ ให้ถูกต้องและตั้งรหัสยืนยันตัวตนของร้านค้า โดยรหัสยืนยันตัวตนต้องไม่ซ้ำกับบัญชีผู้ใช้อื่น ๆ มิเช่นนั้นจะไม่สามารถลงทะเบียนได้



รูปที่ 4.4 หน้าจอการลงทะเบียนเข้าใช้งาน LINE Bot

เมื่อผู้ใช้ลงทะเบียนสำเร็จจะแสดงหน้ายืนยันข้อมูลการลงทะเบียน ดังรูปที่ 4.5



รูปที่ 4.5 หน้าจอยืนยันข้อมูลการลงทะเบียน

ในกรณีที่ไม่สามารถลงทะเบียนได้ จะแสดง Dialog Box แจ้งเตือนข้อผิดพลาด และให้
ผู้ใช้ลงทะเบียนใหม่ ดังรูปที่ 4.6



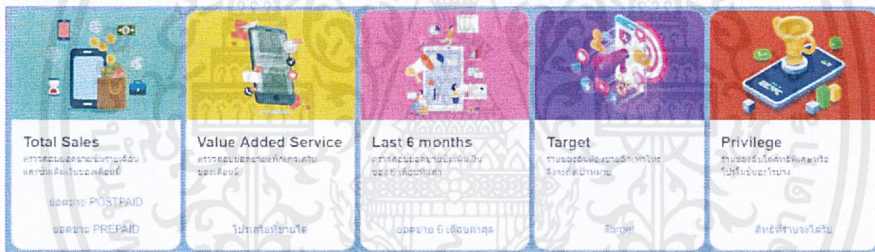
รูปที่ 4.6 Dialog Box เตือนผู้ใช้ว่าไม่สามารถลงทะเบียนได้

4.2.2 การใช้งาน LINE Bot

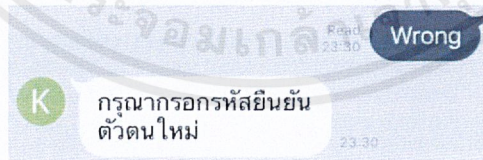
- 1) เมื่อผู้ใช้ลงทะเบียนเรียบร้อยแล้ว การเข้าใช้งาน LINE Bot ในครั้งแรกจะต้องพิมพ์รหัสยืนยันตัวตนเพื่อเปิดระบบการใช้งาน ในกรณีที่ผู้ใช้พิมพ์รหัสยืนยันตัวตนถูกต้อง จะปรากฏเมนูทั้งหมด ดังรูปที่ 4.8 บนหน้าจอการทำงานของผู้ใช้ ดังรูปที่ 4.7 ในกรณีที่ผู้ใช้พิมพ์รหัสยืนยันตัวตนไม่ถูกต้อง จะปรากฏข้อความ กรุณากรอกรหัสยืนยันตัวตนใหม่ ดังรูปที่ 4.9



รูปที่ 4.7 หน้าจอแสดงเมนูทั้งหมด กรณีที่ผู้ใช้พิมพ์รหัสยืนยันตัวตนถูกต้อง



รูปที่ 4.8 เมนูทั้งหมด



รูปที่ 4.9 หน้าจอแสดงเมนูทั้งหมด กรณีที่ผู้ใช้พิมพ์รหัสยืนยันตัวตนไม่ถูกต้อง

ในครั้งต่อ ๆ ไปของการใช้งาน ผู้ใช้ไม่จำเป็นต้องพิมพ์รหัสยืนยันตัวตนใหม่อีกครั้ง สามารถเลือกใช้งานเมนูต่าง ๆ โดยการกด Rich Menu ได้ทันที ดังภาพที่ 4.10



รูปที่ 4.10 Rich Menu

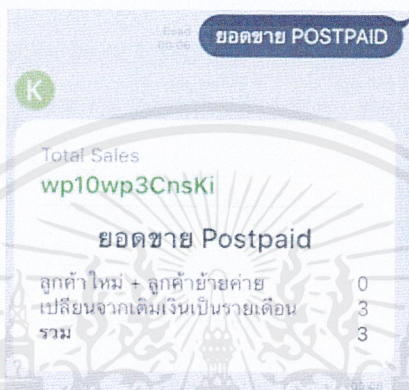
2) เมื่อมีผู้ใช้กดเมนู LINE Bot จะมีการตอบกลับต่างกันในแต่ละเมนู ดังนี้

กรณีผู้ใช้กดเลือกเมนู “ยอดขายรวม” จะปรากฏ Flex Message ให้ผู้ใช้ได้เลือกเมนู “ยอดขาย POSTPAID” และ “ยอดขาย PREPAID” ดังรูปที่ 4.11



รูปที่ 4.11 Flex Message เมนูยอดขายรวม

เมื่อผู้ใช้กดเลือกเมนู “ยอดขาย POSTPAID” LINE Bot จะตอบกลับด้วย Flex Message แสดงจำนวนยอดขายซิมประเภทรายเดือน พิจารณาจากลูกค้าใหม่และลูกค้าที่ย้ายมาจากค่ายอื่นในเดือนปัจจุบัน, จำนวนยอดขายซิมประเภทรายเดือน พิจารณาจากลูกค้าที่ย้ายมาจากซิมประเภทเติมเงินในเดือนปัจจุบัน และยอดขายรวมของซิมประเภทรายเดือนทุกเงื่อนไขในเดือนปัจจุบัน ดังรูปที่ 4.12



รูปที่ 4.12 Flex Message แสดงยอดขาย Postpaid

เมื่อผู้ใช้กดเลือกเมนู “ยอดขาย PREPAID” LINE Bot จะตอบกลับด้วย Flex Message แสดงจำนวนยอดขายซิมประเภทเติมเงินที่อยู่ในเงื่อนไขในเดือนปัจจุบัน, จำนวนยอดขายซิมประเภทเติมเงินที่อยู่นอกเงื่อนไขในเดือนปัจจุบัน, จำนวนยอดขายซิมประเภทเติมเงินที่อยู่ในเงื่อนไขแบบพิเศษในเดือนปัจจุบัน และจำนวนซิมที่มีลงทะเบียนและการเติมเงินในเดือนปัจจุบันโดยแยกแสดงตามราคาของการเติมเงิน ดังรูปที่ 4.13

ยอดขาย Prepaid	
PI+ACT ในเงื่อนไข	1
PI+ACT นอกเงื่อนไข	1
รวม	2
SPECIAL MODEL	0
การเติมเงินภายใน 10 วัน	
฿20 - ฿49	0
฿50 - ฿99	1
฿100 - ฿299	0
มากกว่า ฿300	0

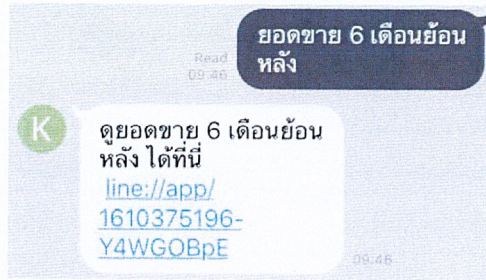
รูปที่ 4.13 Flex Message แสดงยอดขาย Prepaid

กรณีผู้ใช้กดเลือกเมนู “โปรเสริมที่ขายได้” LINE Bot จะตอบกลับด้วย Flex Message แสดงจำนวนแพ็คเกจเสริมที่ขายได้ในเดือนปัจจุบัน โดยแยกแสดงตามราคาของแพ็คเกจเสริม ดังรูปที่ 4.14

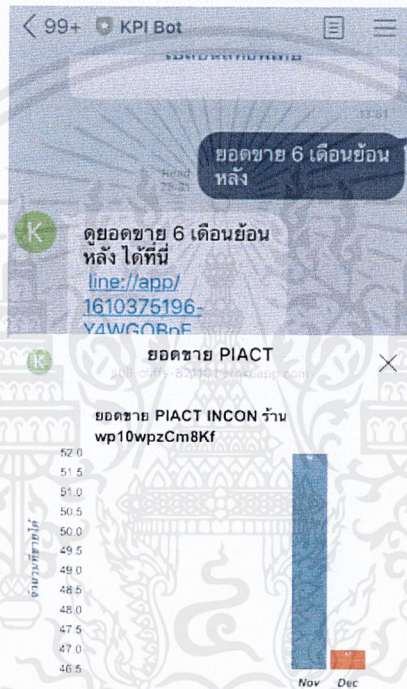
โปรเสริมที่ขายได้	
น้อยกว่า ฿20	0
฿20 - ฿49	2
฿50 - ฿99	0
฿100 - ฿299	0
มากกว่า ฿300	0
รวม	2

รูปที่ 4.14 Flex Message แสดงยอดขายจำนวนแพ็คเกจเสริมที่ขายได้

กรณีผู้ใช้กดเลือกเมนู “ยอดขายย้อนหลัง 6 เดือน” LINE Bot จะตอบกลับด้วยลิงก์ของ LIFF ดังรูปที่ 4.15 เมื่อผู้ใช้กดลิงก์ จะปรากฏ LIFF ที่มีกราฟแสดงจำนวนยอดขายซิมประเภทเติมเงินที่อยู่ในเงื่อนไขในเดือนปัจจุบัน และในระยะเวลา 6 เดือนย้อนหลัง หากไม่มีข้อมูลยอดขายในเดือนนั้น จะไม่มีรูปกราฟปรากฏขึ้น ดังรูปที่ 4.16

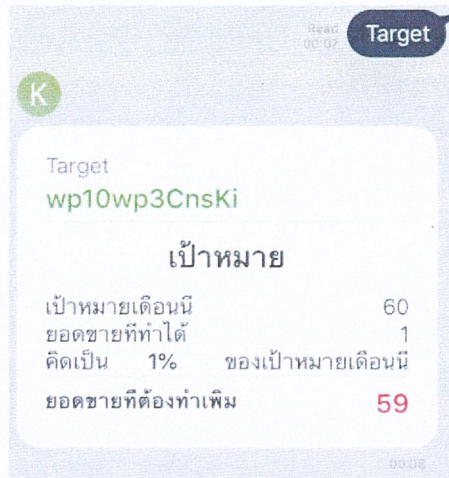


รูปที่ 4.15 ข้อความตอบกลับเมื่อผู้ใช้กดเลือกเมนู ยอดขายย้อนหลัง 6 เดือน



รูปที่ 4.16 LIFF APP แสดงกราฟยอดขายชิมประเภทเดิมเงิน 6 เดือนย้อนหลัง

กรณีผู้ใช้กดเลือกเมนู “Target” LINE Bot จะตอบกลับด้วย Flex Message แสดงเป้าหมายของยอดขายชิมที่ทางบริษัทต้องการให้ร้านค้าขายได้ในเดือนปัจจุบัน, ยอดขายที่ร้านค้าทำได้ทั้งในรูปแบบตัวเลขและเปอร์เซ็นต์ และยอดขายที่ต้องทำเพิ่มสำหรับเดือนปัจจุบัน ดังรูปที่ 4.17



รูปที่ 4.17 Flex Message แสดง เป้าหมายของยอดขายชิม

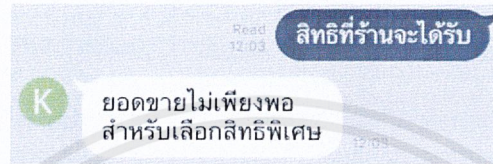
กรณีผู้ใช้กดเลือกเมนู “สิทธิพิเศษ” จะปรากฏ Flex Message ให้ผู้ใช้ได้เลือกเมนู “สิทธิที่ร้านจะได้รับ” และ “เปลี่ยนสิทธิพิเศษ” ดังรูปที่ 4.18



รูปที่ 4.18 Flex Message เมนูสิทธิพิเศษ

กรณีผู้ใช้กดเลือกเมนู “สิทธิที่ร้านจะได้รับ” LINE Bot จะตอบกลับด้วยรูปภาพที่ประกอบด้วย ตัวเลือกสิทธิพิเศษในเดือนปัจจุบัน และ Flex Message ที่มีปุ่มให้ลูกค้ากดเลือกสิทธิพิเศษที่ต้องการ ซึ่งเกณฑ์ในการเลือกสิทธิพิเศษ คัดจากยอดขายชิมประเภทเติมเงินที่อยู่ในเงื่อนไขในเดือนปัจจุบัน หากยอดขายเท่ากับ 0 ผู้ใช้จะไม่สามารถเลือกสิทธิพิเศษได้ LINE Bot จะไม่แสดง Flex Message แต่จะตอบกลับผู้ใช้เป็นข้อความ “ยอดขายไม่เพียงพอสำหรับเลือกสิทธิพิเศษ” แทน ดังรูปที่ 4.19 หากยอดขายอยู่ระหว่าง 1-15 ชิม ผู้ใช้จะสามารถเลือกสิทธิพิเศษได้ 1 สิทธิจาก 1 สิทธิ Flex Message จะ

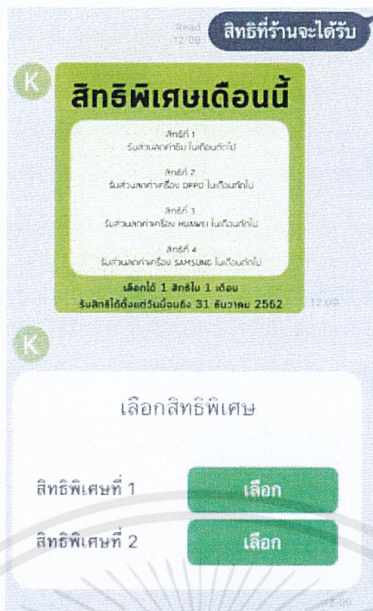
ปรากฏสิทธิพิเศษ 1 ปุ่ม ดังรูปที่ 4.20 หากยอดขายอยู่ระหว่าง 16-29 ซิม ผู้ใช้จะสามารถเลือกสิทธิพิเศษได้ 1 สิทธิจาก 2 สิทธิ Flex Message จะปรากฏสิทธิพิเศษ 2 ปุ่ม ดังรูปที่ 4.21 หากยอดขายอยู่ระหว่าง 30-44 ซิม ผู้ใช้จะสามารถเลือกสิทธิพิเศษได้ 1 สิทธิจาก 3 สิทธิ Flex Message จะปรากฏสิทธิพิเศษ 3 ปุ่ม ดังรูปที่ 4.22 หากยอดขายตั้งแต่ 45 ซิม ขึ้นไป ผู้ใช้จะสามารถเลือกสิทธิพิเศษได้ 1 สิทธิจาก 4 สิทธิ ตามลำดับ Flex Message จะปรากฏสิทธิพิเศษ 4 ปุ่ม ดังรูปที่ 4.23



รูปที่ 4.19 ข้อความ ตอบกลับเมื่อผู้ใช้กดเลือกเมนู สิทธิที่ร้านจะได้รับ (ยอดขาย 0 ซิม)



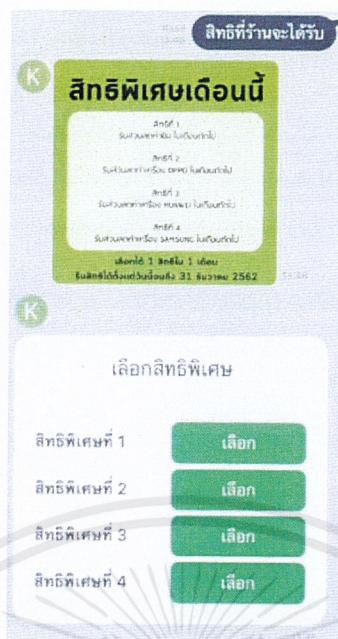
รูปที่ 4.20 รูปภาพและ Flex Message ตอบกลับเมื่อผู้ใช้กดเลือกเมนู สิทธิที่ร้านจะได้รับ (ยอดขาย 1-15 ซิม)



รูปที่ 4.21 รูปภาพและ Flex Message ตอบกลับเมื่อผู้ใช้กดเลือกเมนู สิทธิที่ร้านจะได้รับ (ยอดขาย 16-29 ซิม)

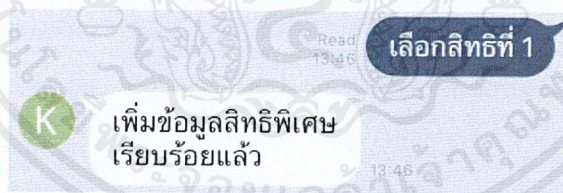


รูปที่ 4.22 รูปภาพและ Flex Message ตอบกลับเมื่อผู้ใช้กดเลือกเมนู สิทธิที่ร้านจะได้รับ (ยอดขาย 30-44 ซิม)



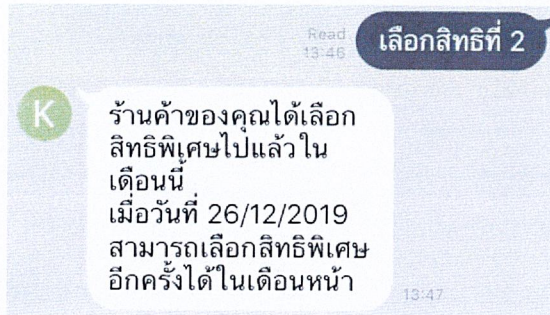
รูปที่ 4.23 รูปภาพและ Flex Message ตอบกลับเมื่อผู้ใช้กดเลือกเมนู สิทธิที่ร้านจะได้รับ (ยอดขาย 45 ซิมขึ้นไป)

หากภายในเดือนปัจจุบัน ผู้ใช้ยังไม่เคยกดเลือกสิทธิพิเศษ เมื่อผู้ใช้กดปุ่มเลือกสิทธิ KPI Bot จะตอบกลับด้วยข้อความ “เพิ่มข้อมูลสิทธิพิเศษเรียบร้อยแล้ว” ดังรูปที่ 4.24



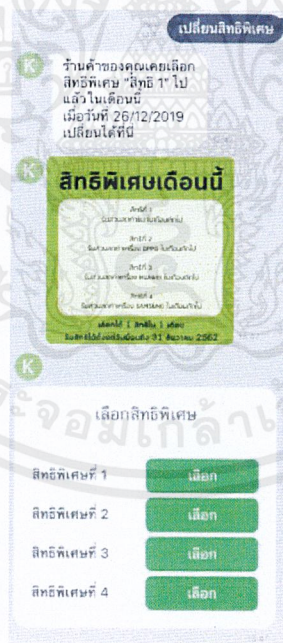
รูปที่ 4.24 ข้อความตอบกลับเมื่อผู้ใช้กดเลือกสิทธิพิเศษ

หากภายในเดือนปัจจุบัน ผู้ใช้เคยกดเลือกสิทธิพิเศษไปแล้ว เมื่อผู้ใช้กดปุ่มเลือกสิทธิ KPI Bot จะตอบกลับด้วยข้อความว่าผู้ใช้เคยกดเลือกสิทธิพิเศษไปแล้วในเดือนปัจจุบัน ดังรูปที่ 4.25



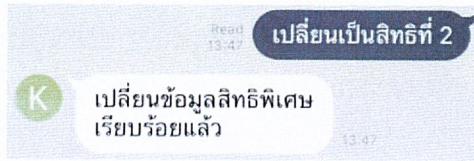
รูปที่ 4.25 ข้อความตอบกลับ กรณีผู้ใช้เคยกดเลือกสิทธิพิเศษไปแล้ว

กรณีผู้ใช้กดเลือกเมนู “เปลี่ยนสิทธิพิเศษ” KPI Bot จะตอบกลับด้วยข้อความแสดงสิทธิพิเศษที่เคยเลือกไปในครั้งก่อนหน้า, รูปภาพที่ประกอบด้วยตัวเลือกสิทธิพิเศษในเดือนปัจจุบัน และ Flex Message ที่มีปุ่มให้ลูกค้ากดเปลี่ยนสิทธิพิเศษ ซึ่งเกณฑ์ในการเปลี่ยนสิทธิพิเศษจะใช้เกณฑ์เดียวกับการเลือกสิทธิพิเศษ ดังรูปที่ 4.26



รูปที่ 4.26 รูปและ Flex Message ตอบกลับเมื่อผู้ใช้กดเลือกเมนู เปลี่ยนสิทธิพิเศษ

เมื่อผู้ใช้กดปุ่มเลือก(เปลี่ยน) สิทธิพิเศษจะตอบกลับด้วยข้อความว่า “เปลี่ยนข้อมูลสิทธิพิเศษเรียบร้อยแล้ว” ดังรูปที่ 4.27



รูปที่ 4.27 ข้อความตอบกลับเมื่อผู้ใช้กดเลือก (เปลี่ยน) สิทธิพิเศษ

หากในกรณีที่ผู้ใช้ส่งข้อความนอกเหนือจากเมนูที่กำหนดไว้ จะปรากฏข้อความ กรณารอกรรหัตถ์ ยืนยันตัวตนใหม่ ดังรูปที่ 4.28

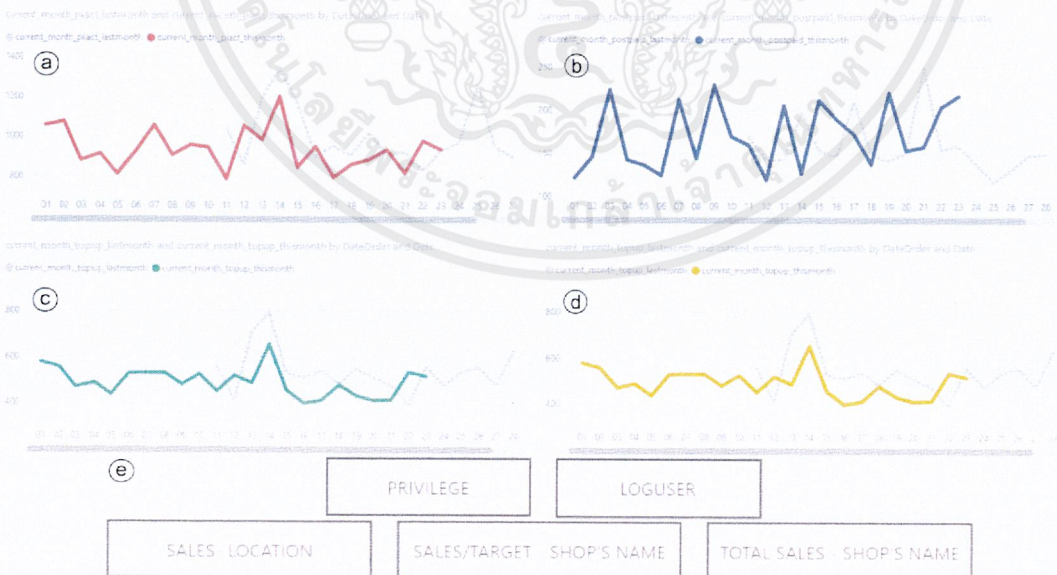


รูปที่ 4.28 ข้อความตอบกลับ กรณีผู้ใช้ส่งข้อความนอกเหนือจากเมนูที่กำหนดไว้

4.3 แดชบอร์ดรายงานข้อมูล โดยใช้โปรแกรม Power BI

การจัดทำแดชบอร์ดเพื่อรายงานข้อมูลโดยใช้โปรแกรม Power BI ประกอบด้วย 6 รายงาน ดังนี้

รายงานที่ 1: รายงานเปรียบเทียบยอดขายประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน

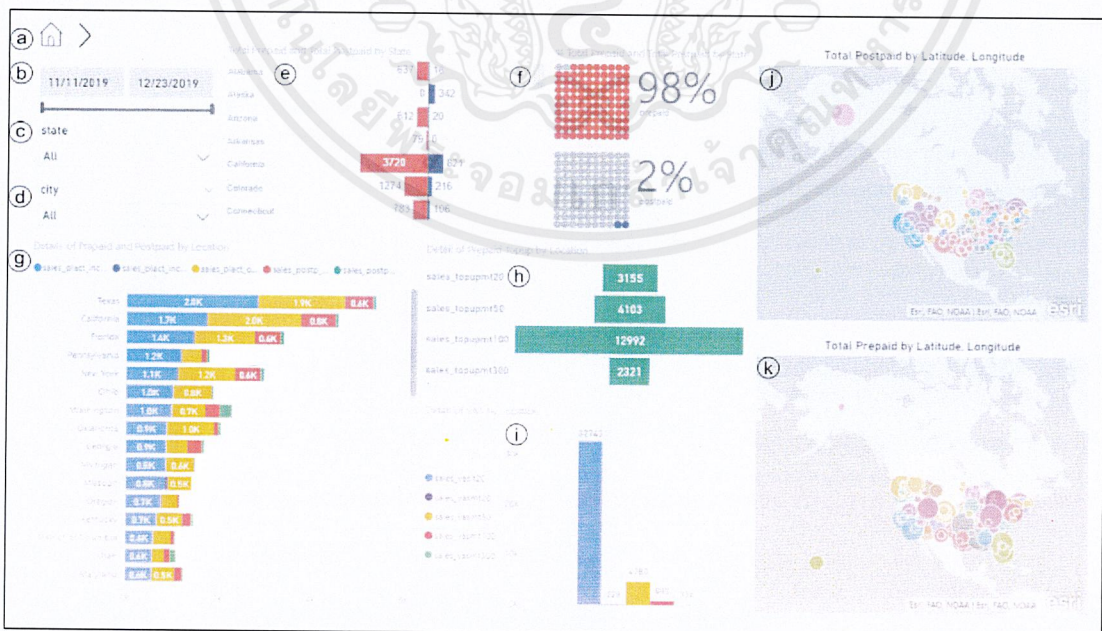


รูปที่ 4.29 รายงานเปรียบเทียบยอดขายประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน

จากรูปที่ 4.29 ประกอบด้วย

- กราฟเส้นแสดงยอดขายรายภูมิภาคเติบเงินประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน
- กราฟเส้นแสดงยอดขายรายภูมิภาครายเดือนประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน
- กราฟเส้นแสดงจำนวนซิมที่มีลงทะเบียนและการเติบเงินประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน
- กราฟเส้นแสดงยอดขายแพ็คเกจเสริมประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน
- Bookmark ที่ใช้เชื่อมต่อไปยังหน้ารายงานต่าง ๆ ดังนี้
 - PRIVILEGE: เชื่อมต่อไปยังหน้ารายงานรายงานการเลือกสิทธิของร้านต่าง ๆ แบ่งตามเวลา
 - LOGUSER: เชื่อมต่อไปยังหน้ารายงานการใช้งาน Line Bot ของร้านค้า
 - SALES- LOCATION: เชื่อมต่อไปยังหน้ารายงานยอดขายแบ่งตามสถานที่ตั้ง
 - SALES/TARGET - SHOP'S NAME: เชื่อมต่อไปยังหน้ารายงานยอดขายในภาพรวมตามเวลาที่กำหนด
 - TOTAL SALES - SHOP'S NAME: เชื่อมต่อไปยังหน้ารายงานยอดขายทุกประเภทและเป้าหมายของร้านค้า

รายงานที่ 2: รายงานยอดขายแบ่งตามสถานที่ตั้ง

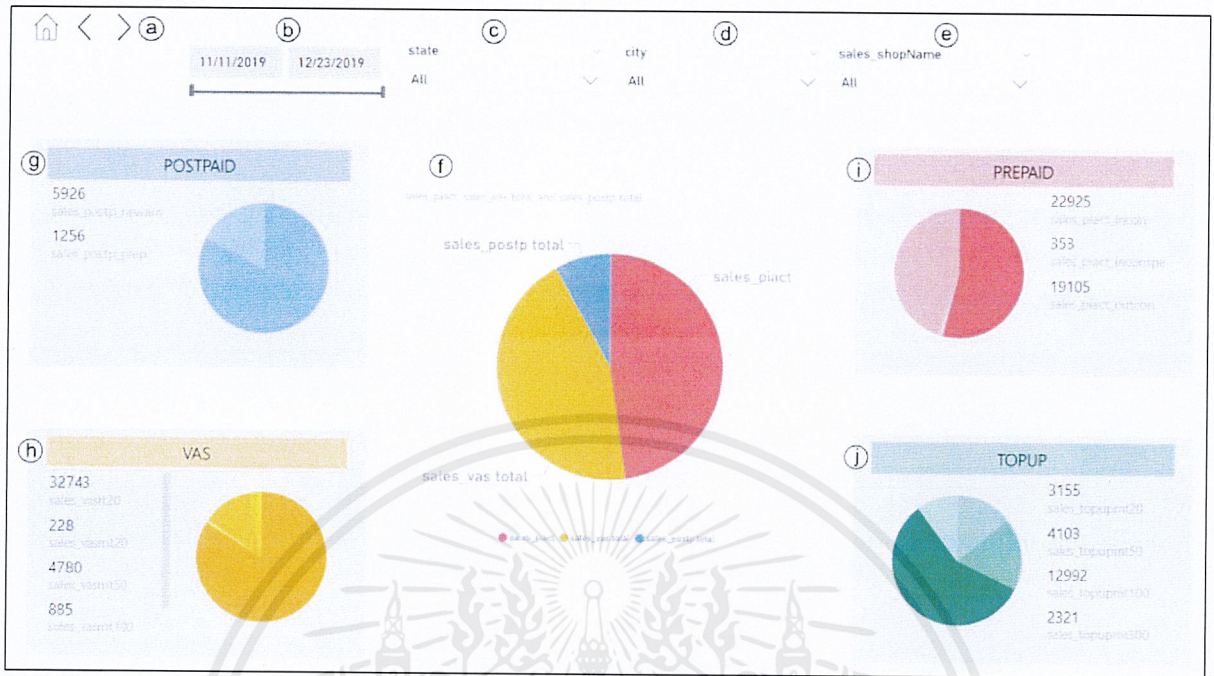


รูปที่ 4.30 รายงานยอดขายแบ่งตามสถานที่ตั้ง

จากรูปที่ 4.30 ประกอบด้วย

- a) Bookmark ที่ใช้เชื่อมต่อไปยังหน้ารายงานต่าง ๆ ดังนี้
 - i. เครื่องหมายรูปบ้าน: เชื่อมต่อไปยังหน้ารายงานเปรียบเทียบยอดขายประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน
 - ii. เครื่องหมายไปข้างหน้า: เชื่อมต่อไปยังหน้ารายงานยอดขายทุกประเภทและเป้าหมายของร้านค้า
- b) Slicer สำหรับกำหนดเวลา
- c) Slicer สำหรับกำหนดรัฐที่ตั้งของร้านค้า
- d) Slicer สำหรับกำหนดเมืองที่ตั้งของร้านค้า
- e) กราฟแท่งเปรียบเทียบยอดขายภูมิภาคเติบเงินและยอดขายภูมิภาครายเดือน แบ่งตามรัฐที่ตั้งของร้านค้า
- f) กราฟแบบ Waffle เปรียบเทียบยอดขายภูมิภาคเติบเงินและยอดขายภูมิภาครายเดือน แสดงผลในรูปแบบเปอร์เซ็นต์ สามารถดูยอดขายในภาพรวมทั้งหมดทุกรัฐหรือเลือกดูเฉพาะรัฐได้
- g) กราฟแท่งแบบซ้อน เปรียบเทียบยอดขายภูมิภาคเติบเงินทุกเงื่อนไขและยอดขายภูมิภาครายเดือนทุกเงื่อนไข
- h) กราฟแท่ง เปรียบเทียบจำนวนซิมที่มีลงทะเบียนและการเติบเงินตามราคาของการเติบเงิน
- i) กราฟแท่ง เปรียบเทียบจำนวนแพ็คเกจเสริมที่ขายได้ตามราคาของแพ็คเกจเสริม
- j) แผนที่แสดงตำแหน่งของร้านค้าและยอดขายภูมิภาครายเดือน โดยสีแสดงถึงรัฐที่ตั้งของร้านค้านั้น ๆ และขนาดของจุดแสดงยอดขายภูมิภาครายเดือน
- k) แผนที่แสดงตำแหน่งของร้านค้าและยอดขายภูมิภาคเติบเงิน โดยสีแสดงถึงรัฐที่ตั้งของร้านค้านั้น ๆ และขนาดของจุดแสดงยอดขายภูมิภาคเติบเงิน

รายงานที่ 3: รายงานยอดขายในภาพรวมตามเวลาที่กำหนด แบ่งตามชื่อร้าน



รูปที่ 4.31 รายงานยอดขายในภาพรวมตามเวลาที่กำหนด แบ่งตามชื่อร้าน

จากรูปที่ 4.31 ประกอบด้วย

- Bookmark ที่ใช้เชื่อมต่อไปยังหน้ารายงานต่าง ๆ ดังนี้
 - เครื่องหมายรูปบ้าน: เชื่อมต่อไปยังหน้ารายงานเปรียบเทียบยอดขายประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน
 - เครื่องหมายย้อนกลับ: เชื่อมต่อไปยังหน้ารายงานยอดขายแบ่งตามสถานที่ตั้ง
 - เครื่องหมายไปข้างหน้า: เชื่อมต่อไปยังหน้ารายงานแสดงยอดขายและเป้าหมาย แบ่งตามชื่อร้าน
- Slicer สำหรับกำหนดเวลา
- Slicer สำหรับกำหนดรัฐที่ตั้งของร้านค้า
- Slicer สำหรับกำหนดเมืองที่ตั้งของร้านค้า
- Slicer สำหรับกำหนดชื่อของร้านค้า
- กราฟวงกลม แสดงอัตราส่วนระหว่างยอดขายซิมประเภทเติมเงินรวมทุกเดือนไซ, ยอดขายซิมประเภทรายเดือนรวมทุกเดือนไซ และจำนวนแพ็คเกจเสริมที่ขายได้รวมทุกราคา
- ข้อความแสดงจำนวน และกราฟวงกลมแสดงอัตราส่วน ยอดขายซิมประเภทรายเดือนทุกเดือนไซ ดังนี้

- i. จำนวนยอดขายชิมประเภทรายเดือนพิจารณาจากลูกค้าที่ย้ายมาจากชิมประเภทเติมเงิน
 - ii. จำนวนยอดขายชิมประเภทรายเดือนพิจารณาจากลูกค้าใหม่และลูกค้าที่ย้ายมาจากค่ายอื่น
- h) ข้อความแสดงจำนวน และกราฟวงกลมแสดงจำนวนแพ็คเกจเสริมที่ขายได้ ทุกเดือนไซ ดังนี้
- i. จำนวนแพ็คเกจเสริมที่ราคาน้อยกว่า 20 บาท
 - ii. จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 20 - 49 บาท
 - iii. จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 50 -99 บาท
 - iv. จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 100 - 299 บาท
 - v. จำนวนแพ็คเกจเสริมที่ราคาตั้งแต่ 300 บาทขึ้นไป
- i) ข้อความแสดงจำนวน และกราฟวงกลมแสดงอัตราส่วน ยอดขายชิมประเภทเติมเงินทุกเดือนไซ ดังนี้
- i. จำนวนยอดขายชิมประเภทเติมเงินที่อยู่ในเดือนไซ (ชิมที่ลูกค้ามีการลงทะเบียนชิมและเติมเงินเรียบร้อยแล้ว)
 - ii. จำนวนยอดขายชิมประเภทเติมเงินที่อยู่ในเดือนไซ (ชิมที่ลูกค้ามีการลงทะเบียนชิมและเติมเงินเรียบร้อยแล้ว)
 - iii. จำนวนยอดขายชิมประเภทเติมเงินที่อยู่ในเดือนไซแบบพิเศษ (ชิมที่ลูกค้ามีการลงทะเบียนแต่ยังไม่มีการเติมเงิน)
 - iv. จำนวนยอดขายชิมประเภทเติมเงินที่อยู่นอกเดือนไซ (ชิมที่ยังไม่มีการลงทะเบียนชิม)
- j) ข้อความแสดงจำนวน และกราฟวงกลมแสดงอัตราส่วนจำนวนชิมที่มีลงทะเบียนและการเติมเงินทุกเดือนไซ ดังนี้
- i. จำนวนชิมที่มีการเติมเงิน 20 - 49 บาท
 - ii. จำนวนชิมที่มีการเติมเงิน 20 - 49 บาท
 - iii. จำนวนชิมที่มีการเติมเงิน 50 - 99 บาท
 - iv. จำนวนชิมที่มีการเติมเงิน 100 - 299 บาท
 - v. จำนวนชิมที่มีการเติมเงิน 300 บาทขึ้นไป

รายงานที่ 4: รายงานแสดงยอดขายและเป้าหมาย แบ่งตามชื่อร้าน



รูปที่ 4.32 รายงานแสดงยอดขายและเป้าหมาย แบ่งตามชื่อร้าน

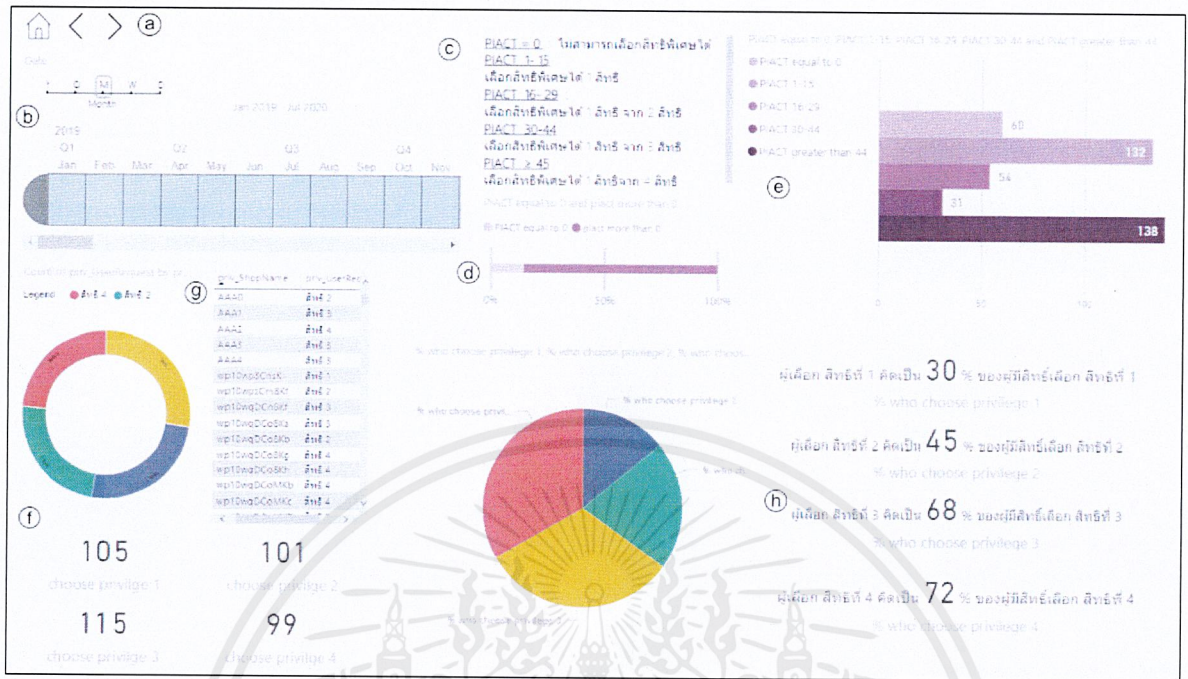
จากรูปที่ 4.32 ประกอบด้วย

- a) Bookmark ที่ใช้เชื่อมต่อไปยังหน้ารายงานต่าง ๆ ดังนี้
 - i. เครื่องหมายรูปบ้าน: เชื่อมต่อไปยังหน้ารายงานเปรียบเทียบยอดขายประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน
 - ii. เครื่องหมายย้อนกลับ: เชื่อมต่อไปยังหน้ารายงานยอดขายในภาพรวมตามเวลาที่กำหนด แบ่งตามชื่อร้าน
 - iii. เครื่องหมายไปข้างหน้า: เชื่อมต่อไปยังหน้ารายงานการเลือกสิทธิของร้านต่าง ๆ แบ่งตามเวลา
- b) Slicer สำหรับกำหนดเวลา
- c) Slicer สำหรับกำหนดชื่อของร้านค้า
- d) Slicer สำหรับกำหนดรัฐที่ตั้งของร้านค้า
- e) Slicer สำหรับกำหนดเมืองที่ตั้งของร้านค้า
- f) กราฟแสดงเป้าหมายของยอดขายซิมประเภทเติมเงินที่อยู่ในเงื่อนไข ถ้าร้านค้ามียอดขายถึงเป้าหมายแล้วจะแสดงกราฟในช่วงของยอดขายที่เกินเป้าหมายเป็นเป็นสีเขียว

- g) กราฟแท่งแสดงจำนวนยอดขายชิมประเภทเติมเงินรวมกันทุกเดือน และ กราฟเส้นแสดงจำนวนยอดขายชิมประเภทเติมเงินแยกตามเดือน ดังนี้
- จำนวนยอดขายชิมประเภทเติมเงินที่อยู่ในเดือน (ชิมที่ลูกค้ามีการลงทะเบียนและเติมเงินเรียบร้อยแล้ว)
 - จำนวนยอดขายชิมประเภทเติมเงินที่อยู่ในเดือนแบบพิเศษ (ชิมที่ลูกค้ามีการลงทะเบียนแต่ยังไม่มีการเติมเงิน)
 - จำนวนยอดขายชิมประเภทเติมเงินที่อยู่นอกเดือน (ชิมที่ยังไม่มีการลงทะเบียนชิม)
- h) กราฟแท่งแสดงจำนวนยอดขายชิมประเภทรายเดือนรวมกันทุกเดือน และ กราฟเส้นแสดงจำนวนยอดขายชิมประเภทรายเดือนแยกตามเดือน ดังนี้
- จำนวนยอดขายชิมประเภทรายเดือนพิจารณาจากลูกค้าที่ย้ายมาจากชิมประเภทเติมเงิน
 - จำนวนยอดขายชิมประเภทรายเดือนพิจารณาจากลูกค้าใหม่และลูกค้าที่ย้ายมาจากค่ายอื่น
- i) กราฟแท่งแสดงจำนวนชิมที่มีลงทะเบียนและการเติมเงินรวมทุกเดือน และ กราฟเส้นแสดงจำนวนชิมที่มีลงทะเบียนและการเติมเงินทุกเดือน ดังนี้
- จำนวนชิมที่มีการเติมเงิน 20 - 49 บาท
 - จำนวนชิมที่มีการเติมเงิน 50 - 99 บาท
 - จำนวนชิมที่มีการเติมเงิน 100 - 299 บาท
 - จำนวนชิมที่มีการเติมเงิน 300 บาทขึ้นไป
- j) กราฟแท่งแสดงจำนวนแพ็คเกจเสริมที่ขายได้รวมทุกเดือน และ กราฟเส้นแสดงจำนวนแพ็คเกจเสริมที่ขายได้ ทุกเดือน ดังนี้
- จำนวนแพ็คเกจเสริมที่ราคาต่ำกว่า 20 บาท
 - จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 20 - 49 บาท
 - จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 50 - 99 บาท
 - จำนวนแพ็คเกจเสริมที่ราคาอยู่ระหว่าง 100 - 299 บาท
 - จำนวนแพ็คเกจเสริมที่ราคาตั้งแต่ 300 บาทขึ้นไป

โดยทั้ง g) h) i) และ j) สามารถแสดงรูปแบบได้ทั้ง รายปี รายไตรมาส รายเดือน และ รายวัน จากรูปที่ 4.32 ซึ่ง g) แสดงผลในรูปแบบรายปี h) แสดงผลในรูปแบบรายไตรมาส i) แสดงผลในรูปแบบรายเดือน และ j) แสดงผลในรูปแบบรายวัน

รายงานที่ 5: รายงานการเลือกสิทธิของร้านต่าง ๆ แบ่งตามเวลา



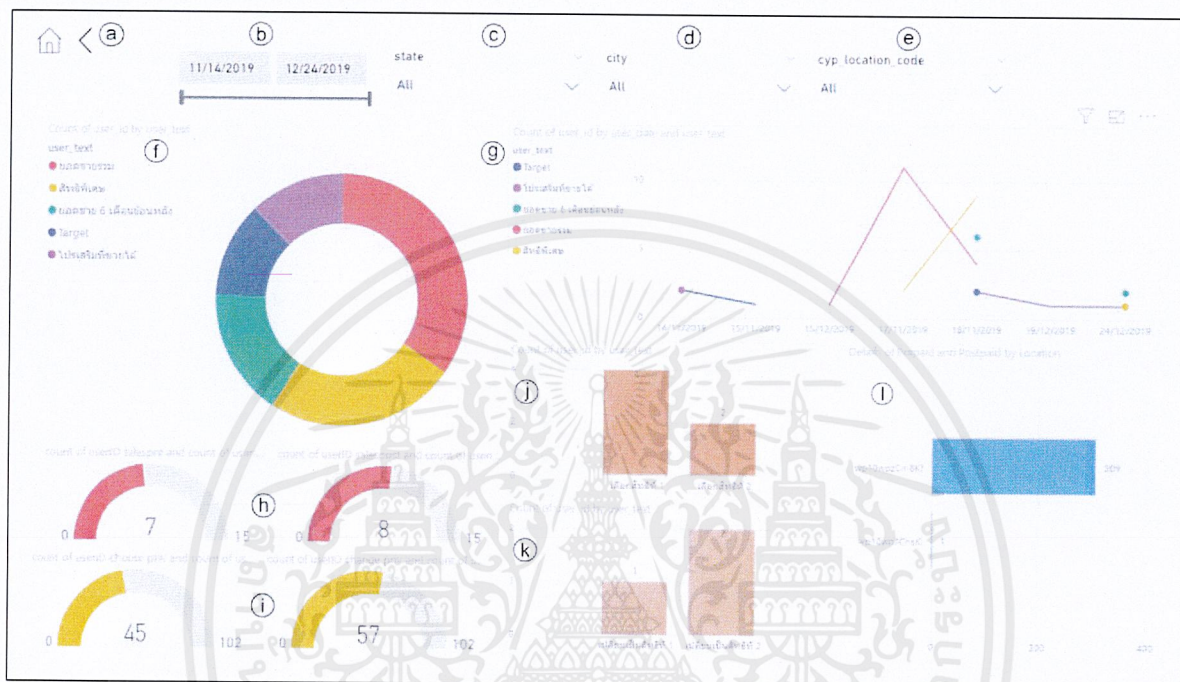
รูปที่ 4.33 รายงานการเลือกสิทธิของร้านต่าง ๆ แบ่งตามเวลา

จากรูปที่ 4.33 ประกอบด้วย

- a) Bookmark ที่ใช้เชื่อมต่อไปยังหน้ารายงานต่าง ๆ ดังนี้
 - i. เครื่องหมายรูปบ้าน: เชื่อมต่อไปยังหน้ารายงานเปรียบเทียบยอดขายประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน
 - ii. เครื่องหมายย้อนกลับ: เชื่อมต่อไปยังหน้ารายงานแสดงยอดขายและเป้าหมาย แบ่งตามชื่อร้าน
 - iii. เครื่องหมายไปข้างหน้า: เชื่อมต่อไปยังหน้ารายงานการใช้งาน Line Bot ของร้านค้า
- b) Slicer สำหรับกำหนดเวลา
- c) กล่องข้อความแสดงข้อมูลเกี่ยวกับการเลือกสิทธิพิเศษ
- d) กราฟแท่งเปรียบเทียบระหว่าง ร้านค้าที่มีจำนวนยอดขายซิมประเภทเติมเงินที่อยู่ในเงื่อนไขเท่ากับ 0 ซิม และมากกว่า 0 ซิม
- e) กราฟแท่งแสดงจำนวนร้านค้าแบ่งตามยอดขายซิมประเภทเติมเงินที่อยู่ในเงื่อนไข
- f) กราฟ sunburst และตัวเลข แสดงอัตราส่วนการเลือกสิทธิพิเศษของร้านค้า
- g) ตารางแสดงการเลือกสิทธิพิเศษของร้านค้า

h) กราฟวงกลม และข้อความ แสดงจำนวนร้านค้าที่เลือกสิทธิพิเศษเทียบกับจำนวนร้านค้าที่มีสิทธิเลือกสิทธิพิเศษ แบ่งตามสิทธิอันดับที่ 1-4 ในรูปแบบเปอร์เซ็นต์

รายงานที่ 6: รายงานการใช้งาน Line Bot ของร้านค้า



รูปที่ 4.34 รายงานการใช้งาน LINE Bot ของร้านค้า

จากรูปที่ 4.34 ประกอบด้วย

- a) Bookmark ที่ใช้เชื่อมต่อไปยังหน้ารายงานต่าง ๆ ดังนี้
 - i. เครื่องหมายรูปบ้าน: เชื่อมต่อไปยังหน้ารายงานเปรียบเทียบยอดขายประจำวันระหว่างเดือนที่แล้วและเดือนปัจจุบัน
 - ii. เครื่องหมายย้อนกลับ: เชื่อมต่อไปยังหน้ารายงานการเลือกสิทธิของร้านค้าต่าง ๆ แบ่งตามเวลา
- b) Slicer สำหรับกำหนดเวลา
- c) Slicer สำหรับกำหนดรัฐที่ตั้งของร้านค้า
- d) Slicer สำหรับกำหนดเมืองที่ตั้งของร้านค้า
- e) Slicer สำหรับกำหนดชื่อของร้านค้า
- f) กราฟ Sunburst แสดงจำนวนครั้งของการกดเลือกเมนูบน LINE Bot

- g) กราฟเส้นแสดงจำนวนครั้งของการกดเลือกเมนูบน Line Bot แสดงผลในรูปแบบรายวัน
- h) กราฟ Gauge แสดงจำนวนครั้งของการกดเลือกเมนู “ยอดขาย POSTPAID” และ “ยอดขาย PREPAID” บน LINE Bot
- i) กราฟ Gauge แสดงจำนวนครั้งของการกดเลือกเมนู “สิทธิที่ร้านจะได้รับ” และ “เปลี่ยนสิทธิพิเศษ” บน LINE Bot
- j) กราฟแท่งแสดงจำนวนครั้งของการกดเลือกใช้สิทธิพิเศษ
- k) กราฟแท่งแสดงจำนวนครั้งของการกดเปลี่ยนสิทธิพิเศษ
- l) กราฟแท่งแสดงจำนวนครั้งของการส่งข้อความมายัง LINE Bot ของร้านค้าแต่ละร้าน



บทที่ 5

สรุปผลการดำเนินงาน ปัญหา และข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

ระบบรายงานและสอบถามข้อมูลผ่าน LINE Bot สำหรับผู้ใช้ที่ทำงานอยู่ในร้านค้าในเครือข่ายของบริษัท และระบบรายงานข้อมูลผ่านแดชบอร์ดบน Power BI สำหรับพนักงานฝ่ายขายของบริษัท สามารถแสดงผลข้อมูลได้ถูกต้องและรวดเร็ว ลดเวลาในการทำงานและลดข้อผิดพลาดในการทำงานที่เกิดขึ้นจากการรวบรวม วิเคราะห์ และสรุปผลข้อมูล ทำให้ร้านค้าในเครือบริษัท สามารถสอบถามข้อมูลได้ตรงตามความต้องการ เป็นการกระตุ้นยอดขาย ทำให้ประสิทธิภาพในการขายสูงขึ้นและทำให้พนักงานฝ่ายขายสามารถนำผลจากรายงานนี้ไปใช้ในการตัดสินใจทางธุรกิจต่อไปได้อย่างรวดเร็วมากขึ้น

5.2 ปัญหาและอุปสรรคที่พบ

- 1) แอปพลิเคชัน LINE มีข้อจำกัดในการพัฒนาทำให้มีตัวเลือกในการแสดงข้อมูลไม่มากเท่าที่ต้องการ
- 2) โปรแกรม Power BI มีขั้นตอนและสูตรที่ใช้ในการสร้างรายงานที่ซับซ้อนทำให้ใช้เวลาในการศึกษาและทำงานเป็นเวลานาน

5.3 ข้อเสนอแนะ

ควรมีความรู้พื้นฐานในการใช้โปรแกรม Microsoft Excel เพื่อที่จะใช้งานโปรแกรม Power BI ได้รวดเร็วมากยิ่งขึ้น เนื่องจากสูตรที่ใช้สำหรับ 2 โปรแกรมนี้มีความคล้ายคลึงกัน

เอกสารอ้างอิง

- [1] JSON. (ม.ป.ป.). [ออนไลน์]. สืบค้นจาก: <https://www.json.org/json-en.html> [6 ธันวาคม 2562]
- [2] IMAP (Internet Message Access Protocol). (2559). [ออนไลน์]. สืบค้นจาก <https://www.json.org/json-en.html> [6 ธันวาคม 2562]
- [3] LINE. (ม.ป.ป.). [ออนไลน์]. สืบค้นจาก [https://en.wikipedia.org/wiki/Line_\(software\)](https://en.wikipedia.org/wiki/Line_(software)) [6 ธันวาคม 2562]
- [4] LINE Messaging API. (ม.ป.ป.). [ออนไลน์]. สืบค้นจาก <https://developers.line.biz/en/docs/messaging-api/overview> [6 ธันวาคม 2562]
- [5] API คืออะไร. (2560). [ออนไลน์]. สืบค้นจาก <https://saixiii.com/what-is-api> [8 ธันวาคม 2562]
- [6] Prisan Pimprasan.(2560).*การสร้าง Line Chat Bot.*[ออนไลน์]. สืบค้นจาก <https://medium.com/@prisan6046/สร้าง-chatbot-line-แบบง่ายด้วย-php-d45921863fcf> [8 ธันวาคม 2562]
- [7] Heroku. (ม.ป.ป.). [ออนไลน์]. สืบค้นจาก <https://www.heroku.com/developers> [8 ธันวาคม 2562]
- [8] Git. (ม.ป.ป.). [ออนไลน์]. สืบค้นจาก <https://git-scm.com> [9 ธันวาคม 2562]
- [9] PHP. (ม.ป.ป.). [ออนไลน์]. สืบค้นจาก <https://en.wikipedia.org/wiki/PHP> [9 ธันวาคม 2562]
- [10] SQL. (ม.ป.ป.). [ออนไลน์]. สืบค้นจาก <https://en.wikipedia.org/wiki/SQL> [9 ธันวาคม 2562]
- [11] องค์ประกอบของ Business Intelligence. (2556). [ออนไลน์]. สืบค้นจาก <http://bi-thailand.blogspot.com/2013/12/business-intelligence.html> [9 ธันวาคม 2562]
- [12] Margaret Rouse.(ม.ป.ป.). *Microsoft Power BI.*[ออนไลน์]. สืบค้นจาก <https://searchcontentmanagement.techtarget.com/definition/Microsoft-Power-BI> [10 ธันวาคม 2562]
- [13] ความหมายของ KPI. (ม.ป.ป.). [ออนไลน์]. สืบค้นจาก <https://th.hrnote.asia/orgdevelopment/190402-kpi> [10 ธันวาคม 2562]