



## รายงานการปฏิบัติงานสหกิจศึกษา

การพัฒนา Line Bot สำหรับ Service Desk

Developing Line bot for Service Desk



โดย

นาย ปารามศ บัวงาม 6204000009


นาย สรวิษฐ์ สุขผล 6204000012

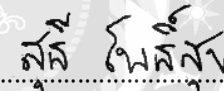
รายงานนี้เป็นส่วนหนึ่งของรายวิชาสหกิจศึกษา  
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์  
มหาวิทยาลัยสยาม  
ภาคการศึกษา 3 ปีการศึกษา 2564


หัวข้อโครงการ            การพัฒนา Line Bot สำหรับ Service Desk  
Deverloping Line bot for Service Desk  
รายชื่อคณะผู้จัดทำ        นาย ปารามศ บัวงาม  
   นาย สรวิชญ์ สุขผล  
ภาควิชา                        วิศวกรรมคอมพิวเตอร์  
อาจารย์ที่ปรึกษา            อาจารย์ สรายุทธ อินทรเสมา


อนุมัติให้โครงการนี้เป็นส่วนหนึ่งของการปฏิบัติงานสหกิจศึกษาภาควิชาวิศวกรรม  
คอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม ภาคการศึกษาที่ 3 ปีการศึกษา 2564

คณะกรรมการสอบสอบโครงการ

  
.....อาจารย์ที่ปรึกษา  
(อาจารย์ สรายุทธ อินทรเสมา)

  
.....พนักงานที่ปรึกษา  
(นาย สุทธิ โปธิ์สุข)

  
.....กรรมการกลาง  
(อาจารย์ ธนากรณ์ รอดชีวิต)

  
.....ผู้ช่วยอธิการบดีและผู้อำนวยการสำนักสหกิจศึกษา  
(ผู้ช่วยศาสตราจารย์ ดร.มารุจ ลิ้มประวัฒน์นะ)

ชื่อโครงการ	: การพัฒนา LINE Bot สำหรับ Service Desk
หน่วยกิต	: 6 หน่วยกิต
ผู้จัดทำ	: นาย ปารามศ บัวงาม , นาย สรวิชญ์ สุขผล
อาจารย์ที่ปรึกษา	: อาจารย์ สราวุธ อินทรเสมา
ระดับการศึกษา	: ปริญญาตรี
ภาควิชา	: วิศวกรรมคอมพิวเตอร์
คณะ	: วิศวกรรมศาสตร์
ภาคการศึกษา/ปีการศึกษา	: 3/2564

### บทคัดย่อ

พีไฟฟ์ แมเนจเม้นท์ จำกัด เป็นผู้ให้บริการเกี่ยวกับการบริการเครื่อง POS ในประเทศไทย ให้บริการในด้านของ Hardware และ Software ในปัจจุบันผู้ประกอบการต่างๆส่วนหันมาให้ความสนใจในการใช้เครื่อง POS (Point to sale system) มากขึ้นเพื่อให้คิดเงินในการขายสินค้าได้ไวขึ้น ซึ่ง พีไฟฟ์ แมเนจเม้นท์จำกัดก็มีแผนกคอยให้บริการแก้ไขปัญหาทางด้าน Hardware และ Software ให้กับทางลูกค้าเพื่อให้ธุรกิจของลูกค้าไม่ติดขัด ดังนั้นการมีตัวช่วยในการหาข้อมูลของช่างแบบรวดเร็วจึงเป็นเรื่องสำคัญ ซึ่งจากข้อความที่กล่าวมาข้างต้นตัวช่วยที่ทำให้พนักงานในแผนก Service desk สามารถค้นหาข้อมูลในการแก้ไขปัญหาให้กับลูกค้าได้เร็ว และ ส่งข้อมูลได้อย่างถูกต้อง ก็คือ LINE Bot เป็น โปรแกรมที่ใช้ในการส่งข้อมูลคู่มือช่างให้สำหรับพนักงานแผนก Service desk

**คำสำคัญ:** เครื่องPOS , เครื่องคิดเงิน , Hardware , Software , LINE Bot

**Project Title** : Development of a LINE Bot with Python  
**Credits** : 6 credits  
**By** : Mr. Paramet Buangam , Mr. Sorawit Sukphon  
**Advisor** : Mr. Sarayut Intarasema  
**Degree** : Bachelor of Engineering  
**Major** : Computer  
**Department** : Engineering  
**Semester / Academic year** : 3/2021

### Abstract

P5 Management Co. is a company that focuses on selling the cash registers and offers technical support for repairs, cleaning, and checking the cash register's system, With the company's limited technicians and difficulties that might be encounters made progress slow and they could not give service to all customer's request. The study aimed at helping them by developing a LINE bot to send a repair document they need. The design used Python as main coding languages for development, we used Python Flask for sending code into the webapp to make LINE Bot processing and Python request for sending or receiving requests between the webserver and LINE Bot, We used Heroku for the PC's Public IP address so LINE Bot can use our IP address, then, upload documents into LINE Bot. The results suggested that a LINE Bot was working as expected and can help technician to make their progress more quickly.

**Keywords:** POS, cash register, hardware, software, LINE Bot



## กิตติกรรมประกาศ (Acknowledgement)

การที่คณะผู้จัดทำได้มาปฏิบัติงานในโครงการสหกิจศึกษา ณ บริษัท พีไฟฟ์ แมเนจเม้นท์ จำกัด ตั้งแต่วันที่ 23 พฤษภาคม พ.ศ. 2565 ถึง 2 กันยายน พ.ศ. 2565 ส่งผลให้คณะผู้จัดทำได้รับความรู้และประสบการณ์ต่างๆ ที่เป็นประโยชน์ต่อการเรียนและการปฏิบัติงานในอนาคต ในตำแหน่ง Service desk และ Service on site ณ บริษัท พีไฟฟ์ แมเนจเม้นท์ จำกัด สามารถนำความรู้ที่ได้ไปใช้ในการประกอบอาชีพในอนาคตได้ โดยได้รับความร่วมมือจากพนักงาน บริษัท บริษัท พีไฟฟ์ แมเนจเม้นท์ จำกัด ได้สอน ได้เรียนรู้งานและให้โอกาสในการการเสนอความคิดและแก้ไขปัญหาต่างๆ ด้วยตนเองในงานที่ได้รับมอบหมายส่งผลให้คณะผู้จัดทำได้พัฒนาทักษะในด้านต่างๆ มากขึ้น จึงขอขอบคุณผู้สนับสนุนจากหลายฝ่าย มา ณ ที่นี้ด้วย ดังนี้

1. คุณ ชนสาร กระจุกแก้ว Sennior Manager
2. คุณ นรชัย พันธุ์จินดาทรัพย์ IT Service Director
3. คุณ ขงยุทธ คำหริ่ม Assistance Manager
4. คุณ จิระศักดิ์ แซ่ตั้ง Supervisor
5. คุณ สุทธิ โพธิ์สุข Supervisor
6. อาจารย์ สรายุทธ อินทรเสมา อาจารย์ที่ปรึกษาสหกิจศึกษา

และบุคคลท่านอื่นๆที่ไม่ได้กล่าวนามทุกท่านที่ได้ให้คำแนะนำช่วยเหลือในการจัดทำรายงาน

คณะผู้จัดทำขอขอบพระคุณผู้ที่มีส่วนเกี่ยวข้องทุกท่านที่มีส่วนร่วมในการให้ข้อมูลและเป็นที่ยอมรับในการทำรายงานฉบับนี้จนเสร็จสมบูรณ์ตลอดจนให้การดูแลและให้ความเข้าใจในชีวิตการทำงานจริง ซึ่งผู้จัดทำขอขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้ด้วย

คณะผู้จัดทำ

นาย ปารามศ บัวงาม

นาย สรวิชญ์ สุขผล

2 กันยายน 2565

## จดหมายนำส่งรายงาน

วันที่ 2 เดือน กันยายน พ.ศ. 2565

เรื่อง ขอส่งรายงานการปฏิบัติสหกิจศึกษา  
เรียน อาจารย์ที่ปรึกษาสหกิจศึกษา ภาควิชา วิศวกรรมคอมพิวเตอร์  
อาจารย์ สราวุธ อินทรเสมา

ตามที่ นาย ปารามศ บัวงาม และ นาย สรวิชญ์ สุขผล นักศึกษาภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม ได้ไปปฏิบัติงานสหกิจศึกษาระหว่างวันที่ 23 พฤษภาคม พ.ศ. 2565 ถึง 2 กันยายน พ.ศ. 2565 ในตำแหน่ง Service desk และ Service on site ณ บริษัท พีไฟฟ์ แมเนจเม้นท์ จำกัด และได้รับมอบหมายจากพนักงานที่ปรึกษาให้ศึกษาและทำรายงานเรื่องการสร้างไลน์แชทบอทส่งคู่มือช่าง

บัดนี้การปฏิบัติงานสหกิจศึกษาได้สิ้นสุดแล้ว นาย ปารามศ บัวงาม และ นาย สรวิชญ์ สุขผล คณะผู้จัดทำจึงขอส่งรายงานดังกล่าวมาพร้อมกันนี้จำนวน 1 เล่ม และ CD จำนวน 1 แผ่น เพื่อขอรับคำปรึกษาต่อไป

จึงเรียนมาเพื่อโปรดพิจารณา

ขอแสดงความนับถือ

นาย ปารามศ บัวงาม

นาย สรวิชญ์ สุขผล

นักศึกษาสหกิจศึกษา

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยสยาม

## สารบัญ

	หน้า
กิตติกรรมประกาศ	ก
จดหมายนำส่ง	ข
บทคัดย่อ	ค
Abstract	ง
<b>บทที่ 1 บทนำ</b>	
1.1 ประวัติความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ขอบเขตของโครงการ.....	2
1.4 ประโยชน์ที่ได้รับ.....	2
<b>บทที่ 2 การทบทวนวรรณกรรมที่เกี่ยวข้อง</b>	
2.1 Line Developer.....	3
2.2 Line.....	5
2.3 Visual Studio Code.....	6
2.4 Webhook .....	9
2.5 LineBotDesigner.....	9
2.6 Python.....	12
2.7 Python Flask .....	13
2.8 Dialogflow.....	13
2.8.1 Agents.....	14
2.8.2 Intents.....	14
2.8.3 Responses.....	14
2.8.4 Entities.....	14
2.8.5 Fulfillments.....	14
2.9 Heroku.....	15

### บทที่ 3 รายละเอียดการปฏิบัติงาน

3.1	ชื่อและที่ตั้งของสถานประกอบการ.....	16
3.2	ลักษณะการประกอบการ.....	17
3.3	ตำแหน่งและลักษณะงานที่ได้รับมอบหมาย.....	19
3.4	ชื่อและตำแหน่งของพนักงานที่ปรึกษา.....	19
3.5	ระยะเวลาในการปฏิบัติบ้ดงาน.....	19
3.6	หลักการทำงานของ Flowchart.....	20
3.7	คำอธิบายขั้นตอนและวิธีการดำเนินงาน.....	21
3.8	อุปกรณ์และเครื่องที่ใช้.....	30

### บทที่ 4 ผลการปฏิบัติโครงการงาน

4.1	การเปิดใช้งานชุดคำสั่งเพื่อไว้สำหรับเชื่อมต่อกับ Line Channel.....	31
4.2	การแสดงผลการทำงานของชุดคำสั่งบน Visual Studio Code.....	32
4.3	การแสดงผลรวมของ Heroku.....	32
4.4	นำ URL Webhook ที่ได้จาก Heroku ไปใช้งานกับ Line Developers.....	33
4.5	รูปภาพต่างแสดงผลของช่องแชท.....	34





## สารบัญ (ต่อ)

หน้า

### บทที่ 5 สรุปผลรายงานและข้อเสนอแนะ

5.1 สรุปผลโครงการ.....	36
5.1.1 ข้อจำกัดหรือปัญหาของโครงการ.....	36
5.1.1.1 สรุปผลการใช้งานไลน์บอท.....	36
5.1.1.2 ข้อจำกัดของไลน์บอท.....	36
5.1.1.3 สรุปผลการใช้งานไลน์บอท.....	36
5.1.1.4 ข้อจำกัดของไลน์บอท.....	36
5.1.2 ข้อเสนอแนะ.....	36
5.1.2.1 ข้อเสนอแนะ.....	36
5.1.2.2 ข้อเสนอแนะ.....	36
5.2 สรุปผลการปฏิบัติงานสหกิจศึกษา.....	37
5.2.1 ข้อดีของการปฏิบัติงานสหกิจศึกษา.....	37
5.2.1.1 ข้อดีของการปฏิบัติงานสหกิจศึกษา.....	37
5.2.2 ปัญหาที่พบของการปฏิบัติงานสหกิจศึกษา.....	37
5.2.2.1 ปัญหาที่พบของการปฏิบัติสหกิจศึกษา.....	37
5.2.3 ข้อเสนอแนะ.....	37
5.2.3.1 ข้อเสนอแนะในการปฏิบัติสหกิจศึกษา.....	37
บรรณานุกรม.....	38
ภาคผนวก ก ภาพของการสร้างบอท.....	39
ภาคผนวก ข ภาพขณะปฏิบัติงาน.....	42
ประวัติผู้จัดทำ.....	47

## สารบัญตาราง

หน้า

ตารางที่ 3.1 ระยะเวลาในการดำเนินโครงการ.....	19
--	----



## สารบัญรูปภาพ

หน้า

รูปที่ 2.1 ตัวอย่าง Product ของ LINE Developer.....	4
รูปที่ 2.2 ไอคอนของแอปพลิเคชัน LINE.....	5
รูปที่ 2.3 หน้าต่างของโปรแกรม Visual Studio Code.....	6
รูปที่ 2.4 หน้าต่างของโปรแกรม Visual Studio Code Intellisense.....	7
รูปที่ 2.5 หน้าต่างของโปรแกรม VisualStudioCode การมอง.....	8
รูปที่ 2.6 หน้าต่างของโปรแกรม VisualStudioCode CLI.....	8
รูปที่ 2.7 ภาพการทำงานของ webhook.....	9
รูปที่ 2.8 ตัวอย่างของเว็บไซต์ Line Bot Desiner.....	10
รูปที่ 2.9 ตัวอย่างของ Bot Items Message.....	10
รูปที่ 2.10 ตัวอย่างของ Bot Items Rich Menu.....	11
รูปที่ 2.11 ตัวอย่างของ Bot Items WebApp.....	11
รูปที่ 2.12 รูปภาพ Python.....	12
รูปที่ 2.13 Python Flask.....	13
รูปที่ 2.14 ตัวอย่างchatbotสำหรับรับแจ้งข้อความโดย Trainingphrase.....	14
รูปที่ 2.15 Dialogflow.....	15
รูปที่ 2.15 Heroku.....	15
รูปที่ 3.1 แผนที่ตั้งของสถานประกอบการ.....	16
รูปที่ 3.2 บริการ Helpdesk Support ของทาง บริษัทพีไฟฟ์ แมเนจเม้นท์ จำกัด.....	17
รูปที่ 3.3 บริการ Network ของทาง บริษัทพีไฟฟ์ แมเนจเม้นท์ จำกัด.....	17
รูปที่ 3.4 บริการการติดตั้งของทาง บริษัทพีไฟฟ์ แมเนจเม้นท์ จำกัด.....	18
รูปที่ 3.5 ตัวอย่างผู้ใช้บริการของทางบริษัทพีไฟฟ์ แมเนจเม้นท์ จำกัด.....	18
รูปที่ 3.6 Flowchart ของระบบ Messaging API.....	20
รูปที่ 3.7 ตัวอย่างของกำทำงานเบื้องต้นของ Line Bot.....	22
รูปที่ 3.8 การสร้าง Provider.....	22
รูปที่ 3.9 การสร้าง Messaging API channel.....	23
รูปที่ 3.10 การตั้งค่าในแถบ Messaging API.....	24
รูปที่ 3.11 ตัวอย่างการปิดการใช้งาน Auto-reply Message.....	24

รูปที่ 3.12	เว็บไซต์ของ Heroku.....	25
รูปที่ 3.13	ตัวอย่างการ Create a new app บนเว็บไซต์ของ Heroku.....	25
รูปที่ 3.14	หน้าตาสำหรับดูภาพรวมของ App บน Heroku.....	26
รูปที่ 3.15	หน้าการเขียนและทำงานของชุดคำสั่งบน Visual Studio Code.....	27
รูปที่ 3.16	หน้าตาโปรแกรม Command line.....	27
รูปที่ 3.17	หน้าเว็บไซต์ของ Dialogflow.....	28
รูปที่ 3.18	การเปิดใช้งาน Webhook บน Dialogflow.....	29
รูปที่ 3.19	การเปิดใช้งาน Webhook บน Line Developers.....	30
รูปที่ 4.1	ภาพแสดงการเปิดใช้งานชุดคำสั่งเพื่อไว้สำหรับเชื่อมต่อกับ Line Channel.....	31
รูปที่ 4.2	ภาพแสดงผลการทำงานของชุดคำสั่งบน Visual Studio Code.....	32
รูปที่ 4.3	หน้าตาแสดงผลรวมของ Heroku.....	32
รูปที่ 4.4	หน้าแถบ Webhook Setting ของ Line Developers.....	33
รูปที่ 4.5	ตัวอย่างการใช้งานเมื่อเลือกตัวเลือกที่ 1.....	34
รูปที่ 4.6	ตัวอย่างการใช้งานเมื่อเลือกตัวเลือกที่ 2.....	35
รูปที่ 4.7	ตัวอย่างการใช้งานเมื่อเลือกตัวเลือกที่ 3.....	35



# บทที่ 1

## บทนำ

### 1.1 ประวัติความเป็นมาและความสำคัญของปัญหา

บริษัทพีไฟฟ์ แมเนจเม้นท์ จำกัด เป็นผู้ให้บริการเกี่ยวกับอุปกรณ์เครื่องคิดเงิน บริการด้านการติดตั้งและการซ่อมบำรุงให้กับร้านค้าปลีกทั่วประเทศ ทั้งในด้านของ Hardware และ Software

คณะผู้จัดทำได้มาปฏิบัติงานในโครงการสหกิจศึกษาในตำแหน่งของ Service Desk และช่างติดตั้ง (Service on site) โดยได้รับมอบหมายจากทางบริษัทให้ทำการเรียนรู้เกี่ยวกับ Hardware และ Software ที่เกี่ยวข้องกับการทำงานของบริษัทตั้งแต่ขั้นพื้นฐาน ไปจนถึงขั้นกลาง โดยเริ่มจากการเรียนรู้เรื่อง Hardware ตั้งแต่ถอดประกอบเครื่องคิดเงิน เครื่องพิมพ์ใบเสร็จ หรือการลูกค้าแก้ปัญหาด้าน Software ผ่านทาง Service Desk โดยการใช้โปรแกรมของบริษัทในการแก้ไขปัญหา และรับเรื่องปัญหาของลูกค้าผ่านทางไลน์และทางโทรศัพท์ โดยมีพี่เลี้ยงและพนักงานในบริษัทให้คำแนะนำปรึกษาอย่างใกล้ชิด

เมื่อคณะผู้จัดทำได้ศึกษางานเกี่ยวกับระบบงานของบริษัทเบื้องต้นแล้วทางบริษัทจึงได้มอบหมายให้ทำการสร้างแชทบอทไลน์ขึ้นมา เพื่อให้ทางพนักงานบริษัทสามารถค้นหาข้อมูลเพื่อทำการซ่อมแซมหรือแก้ไขปัญหาของลูกค้าได้อย่างรวดเร็ว โดยใช้ไลน์ในการส่งข้อมูลคู่มือช่างไปให้พนักงานเมื่อพนักงานได้มีส่งตัวคำสั่งเพื่อต้องการรับข้อมูลเข้าภายในช่องพูดคุยของไลน์

โดยแชทบอทไลน์ที่คณะผู้จัดทำต้องทำการสร้างเป็นแชทบอทไลน์เกี่ยวกับการส่งข้อมูลและการจัดระเบียบของคู่มือภายในบริษัท โดยจะใช้คู่มือที่ได้จากพนักงานภายในแผนก Service Desk โดยทางด้านแชทบอทไลน์จะสามารถส่งคู่มือให้กับพนักงานตามที่พนักงานต้องการได้ รวมถึงสามารถแยกหมวดหมู่ของคู่มือได้และผู้เข้าใช้บอทไลน์ต้องมีสิทธิ์ในการใช้เท่านั้นแชทบอทไลน์ถึงจะส่งคู่มือช่างให้

### 1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อเอาความสามารถของไลน์แชทบอทมาแบ่งเบาภาระการทำงานของพนักงานในแผนกของ Service desk ในการส่งคู่มือแก้ไขปัญหาในเรื่องของ Hardwar และ Software
- 1.2.2 เพื่อให้ไลน์แชทบอทสามารถส่งข้อมูลให้กับผู้ที่สแกนคิวอาร์โค้ดเพียงเท่านั้น
- 1.2.3 เพื่อสร้างไลน์แชทบอทให้พนักงานสามารถแก้ไขปัญหาในเรื่อง Hardwar และ Software ให้กับลูกค้าได้รวดเร็วมากขึ้น

### 1.3 ขอบเขตของการทำโครงการ

- 1.3.1 ไลน์เซทบอทสามารถส่งคู่มือในรูปแบบของข้อความได้
- 1.3.2 มีแถบรายการในการแยกประเภทของคู่มือ
- 1.3.3 ไลน์เซทบอทสามารถตอบกลับผู้ใช้โดยการใช้คำสำคัญได้

### 1.4 ประโยชน์ที่ได้รับ

- 1.4.1 ทำให้พนักงานสามารถตอบและแก้ไขปัญหาให้กับลูกค้าได้เร็ว และมีประสิทธิภาพในการตอบปัญหาที่เกิดขึ้นกับลูกค้าได้อย่างตรงประเด็นมากขึ้น
- 1.4.2 ทำให้พนักงานสามารถหาคู่มือได้เร็วมากขึ้น โดยแยกตามหมวดหมู่



## บทที่ 2

### การทบทวนเอกสารและวรรณกรรมที่เกี่ยวข้อง

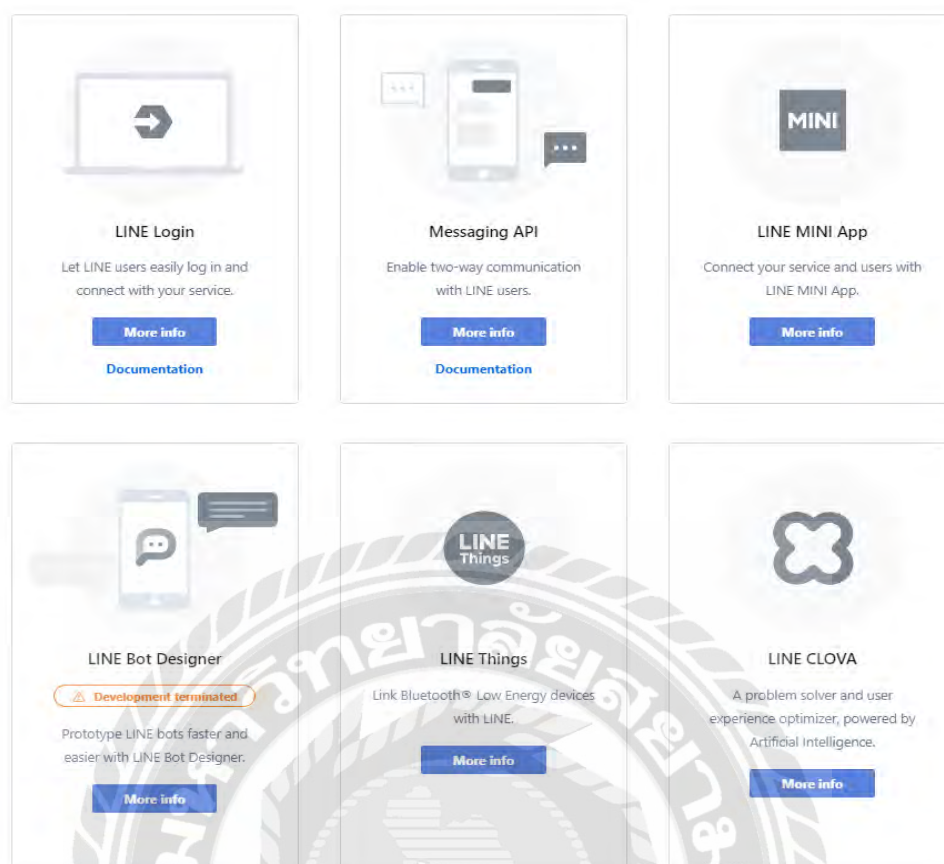
ในการจัดทำโครงการสหกิจศึกษานี้ คณะผู้จัดทำได้ทำการศึกษาค้นคว้า ทฤษฎี แนวคิดและโปรแกรมทั้งหมดที่ใช้ในการทำโครงการให้โครงการสำเร็จตามวัตถุประสงค์และตอบโจทย์ตามที่ต้องการ

#### 2.1 Line Developer

Line Developer เป็นหน้าเว็บไซต์สำหรับการพัฒนาระบบที่เกี่ยวข้องกับ LINE เช่น LINE login, Messaging API, LINE MINI APP, LINE BOT Designer เป็นต้น โดยเป็นคอมมูนิตี้ที่รวมตัวของนักพัฒนาโดยใช้ตัว API ของทาง LINE ในการพัฒนาเพื่อสร้างแอปพลิเคชันหรือบริการขึ้นมาสำหรับใช้งานใน LINE

#### ตัว Product ของ LINE Developer

- LINE Login : เพื่อให้ตัวผู้ใช้งาน LINE สามารถเข้าใช้ระบบได้ง่ายมากขึ้นและสามารถเชื่อมต่อหรือใช้งานบริการของผู้พัฒนาได้ง่ายขึ้น
- Messaging API : เพื่อสร้างตัวบอทหรือบัญชีที่สามารถสื่อสารกับทางผู้ใช้งานได้โดยตรงผ่านทางช่องทางแชท
- LINE MINI APP : เพื่อช่วยในตัวผู้ใช้งานสามารถเชื่อมต่อและใช้งานกับตัวแอปพลิเคชันหรือบริการของผู้พัฒนาได้
- LINE Bot Designer : สร้างแบบจำลองของ LINE Bot ขึ้นมาโดยที่ไม่ต้องสร้างใหม่ตั้งแต่ต้นและช่วยให้ผู้พัฒนาสามารถพัฒนาตัวบอทได้ง่ายมากขึ้น
- LINE Thing : สามารถเชื่อมต่ออุปกรณ์ Bluetooth กับทางตัว API ของ LINE ได้
- LINE Pay : เพิ่มวิธีการชำระเงินให้กับตัวแอปพลิเคชันหรือบริการของผู้พัฒนา



รูปที่ 2.1 ตัวอย่าง Product ของ LINE Developer

### ตัวอย่างคำสั่งใน Line Developer

- Channel = สำหรับการกักเก็บและสร้างตัว Product ขึ้นมา
- Role = สำหรับการกำหนดสมาชิกที่สามารถใช้งานตัว Channel นั้นๆได้
- Setting = สำหรับตรวจสอบข้อมูลเบื้องต้น, Provider ID, เปลี่ยนชื่อ และการลบ Provider
- Product = สำหรับการดู API กับแอปพลิเคชันที่สามารถใช้งานได้



## 2.2 LINE

เป็น โปรแกรมแชทที่สามารถใช้งาน ได้ทั้ง โทรศัพท์มือถือที่มีระบบปฏิบัติการ iOS, Android, Windows Phone ล่าสุดสามารถใช้งาน ได้บนคอมพิวเตอร์ PC และ Mac ได้แล้ว ด้วยความ ที่มีลูกเล่นมากมาย สามารถแชท ส่งรูป ส่งไอคอน ส่ง Sticker ตั้งค่าคุยกันเป็นกลุ่ม ฯลฯ ทำให้มี ผู้ใช้งานแอปนี้ เป็นจำนวนมาก นอกจากนี้ LINE ยังมีโปรแกรมเสริม ทั้ง LINE Camera ที่ถ่ายภาพ ฟรี พร้อมกรอบกว่า 100 แบบ และแสดมป์แต่งภาพมากกว่า 600 แบบเก้ๆ โดยจุดเด่นของ แอปพลิเคชันนี้คือ dki ตกแต่งภาพหลากหลายรวมไปถึงการถ่ายภาพผ่านฟิลเตอร์ถึง 14 แบบ ที่ช่วย ปรับแต่งภาพและรายละเอียดให้ภาพดูดีขึ้น พร้อมด้วยฟู้กันกว่า 156 ชนิด เพื่อให้ผู้ใช้ได้แต่ง แด้มด้วยแสดมป์และเลือกแบบตัวอักษรต่างๆ พิมพ์ข้อความลงบนภาพตามสไตล์ของตัวเองและ สามารถแชร์ภาพได้โดยตรง

### สิ่งทีโดดเด่นของไลน์

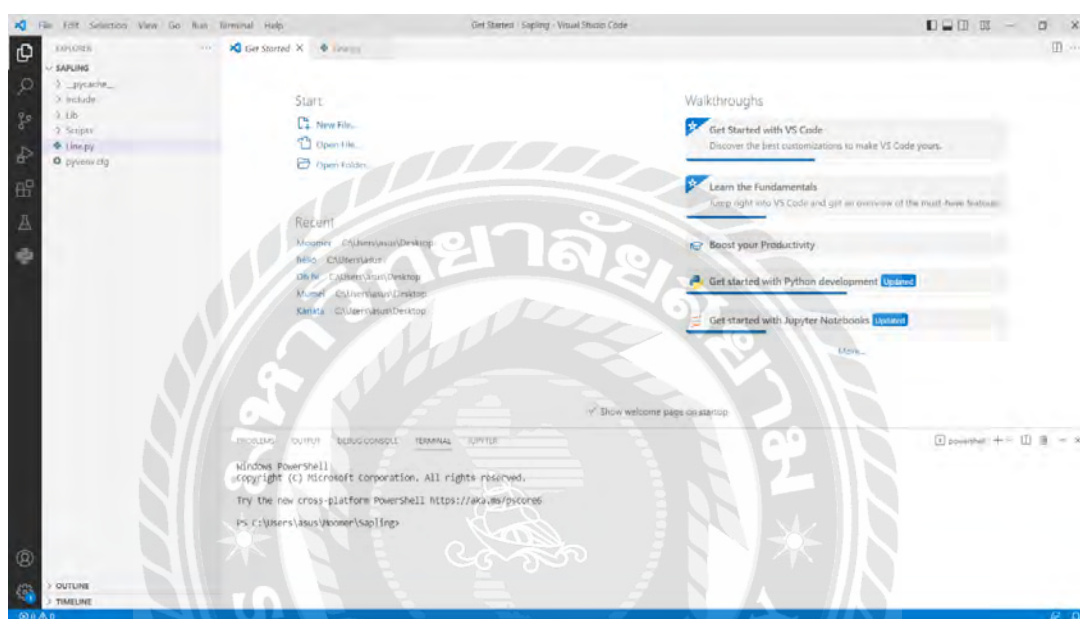
- 1.สามารถเพิ่มกลุ่มสนทนาหรือเชิญเพื่อน ได้ถึง100คน
- 2.ออกแบบให้สามารถโทร.หากันฟรีแบบ1ต่อ1
- 3.พัฒนาคุณภาพของการ โทร.ให้ดีขึ้น โดยตัดเสียงรบกวนและเสียงแทรกจากบริเวณรอบๆ ทำให้ผู้ใช้สามารถพูดคุย
- 4.ส่งวิดีโอและข้อความเสียงฟรี



รูป 2.2 ไอคอนของแอปพลิเคชัน LINE

## 2.3 Visual Studio Code

Visual Studio Code (VS Code) เป็นโปรแกรมแก้ไขซอร์สโค้ดที่มีขนาดเล็กแต่ทรงพลัง ซึ่งทำงานบน Desktop และพร้อมใช้งานสำหรับ Windows, macOS และ Linux ซึ่งมาพร้อมกับการสนับสนุนในตัวสำหรับ JavaScript, TypeScript และ Node.js และมีระบบนิเวศที่สมบูรณ์ของส่วนขยายสำหรับภาษาอื่น ๆ (เช่น C++, C#, Java, Python, PHP และ Go) และรันไทม์ (เช่น .NET และ Unity)



รูปที่ 2.3 หน้าต่างของโปรแกรม Visual Studio Code

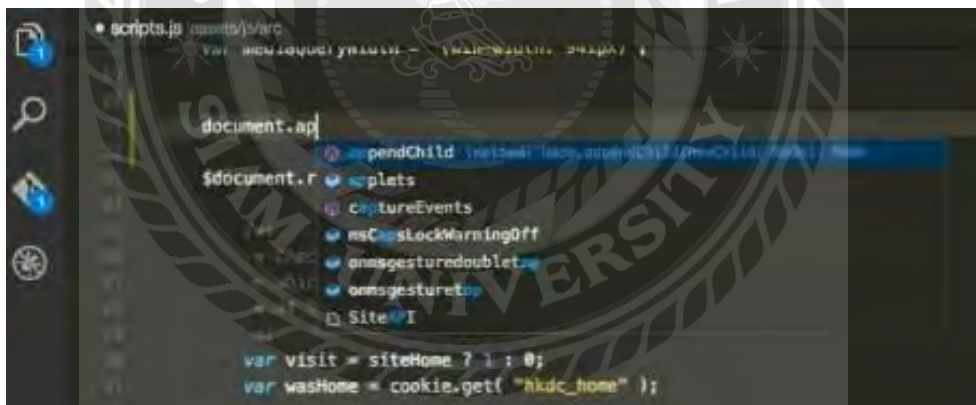
### เทคนิคการใช้งาน Visual Studio Code (Editor) ที่น่าสนใจ 15 เทคนิค

1. Ctrl+P เปิดไฟล์อย่างรวดเร็วโดยการพิมพ์ชื่อไฟล์ได้เลย
2. Ctrl+Shift+F ค้นหาแบบรวดเร็วในไฟล์ต่างๆ
3. Ctrl+Shift+P แสดงชุดคำสั่งต่างๆ เช่น ค้นหาการตั้งค่าของ editor ลองพิมพ์ color theme เพื่อเปลี่ยน theme ของ editor
4. ลองใช้ฟอนต์ Menlo-Regular ติดตั้งลง windows ก่อน แล้วตั้งค่าที่ User Settings ฟอนต์จะดูง่ายสบายตา
5. Ctrl+B เปิด-ปิด Sidebar
6. ซ่อนแถบ OpenEditors ใน EXPLORER ตั้งค่าที่ User Settings
7. ซ่อน minimap ด้านขวาจะได้ดูสะอาดๆ ตั้งค่าที่ User Settings

8. สำหรับคนที่เขียนPHPให้ติดตั้ง extensionsphpintelephense และ phpdebug
9. ลองติดตั้งextensionชื่อว่า snippetcreator ไว้สร้างsnippetใช้เองได้
10. ใช้ Ctrl+K เพื่อเคลียร์ข้อความในterminal
11. จัดรูปแบบ โค้ดใช้ Shift+Alt+F
12. สำหรับคนที่เขียน Vue.js ก็ใช้extensionชื่อว่า Vetur
13. สำหรับคนที่ใช้ Git แนะนำ extensions ชื่อว่า GitHistory และ GitLens
14. กดAltค้างไว้แล้วคลิกเมาส์ที่บรรทัดนั้นๆหากต้องการใช้ multi-cursor
15. หากต้องการจัดรูปแบบโค้ดตอนบันทึกไฟล์ทุกครั้งก็ให้ตั้งค่า UserSettings

### ประโยชน์ของVisualstudiocode

1. Intellisense เป็นรหัส เติมข้อความอัตโนมัติตรงตามปัญญาประดิษฐ์. ยูทิลิตีนี้มีรายการคำแนะนำพร้อมกับคำใบ้หรือคำอธิบายสั้น ๆ ในขณะที่เรากำลังเขียนรหัส เหล่านี้คือ ได้มาจาก บัญชีตามบริบทหลายประการ เช่นภาษาการเขียนโปรแกรมไวยากรณ์ตัวแปรฟังก์ชัน เช่นเดียวกับ รหัสทั้งหมดที่อยู่ในไฟล์IntelliSense รองรับภาษาการเขียนโปรแกรมจำนวนมาก รวมถึง Sass, LESS, JavaScript, TypeScript



รูปที่ 2.4 หน้าต่างของโปรแกรม Visual Studio Code Intellisense

2. การมองในขณะที่เขียน โค้ดมักจะลิมฟังก์ชันบางอย่างโดยที่ฟังก์ชันนั้นถูกกำหนดไว้ในตอนแรกและสิ่งที่เป็นพารามิเตอร์ที่จำเป็น กับ แอบมอง สามารถเลือกฟังก์ชันจากนั้นกด Shift + F12 การคัดลอก ขยายเข้าไปในหน้าต่างออนไลน์แสดงคำจำกัดความที่สมบูรณ์ของฟังก์ชัน เช่นเดียวกับที่กำหนดฟังก์ชัน ปัจจุบันคุณลักษณะนี้ทำงานใน C, C #, JavaScript, TypeScript, .NET และภาษาการเขียนโปรแกรมอื่นๆ

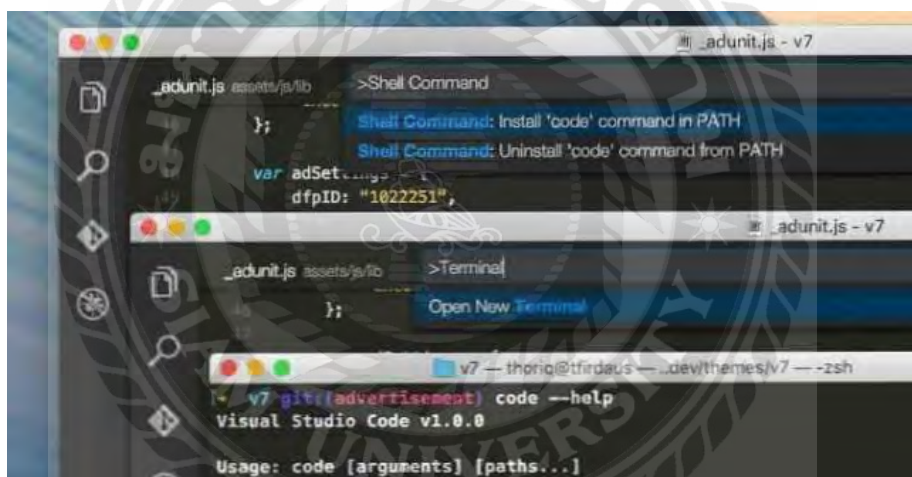
```

19 ( !$( this ).is( ".sharing-url--email" ) ) {
20   event.preventDefault();
21   openShare( this.href, 600, 420 );
22 }
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

รูปที่ 2.5 หน้าต่างของโปรแกรม VisualStudioCode การมอง

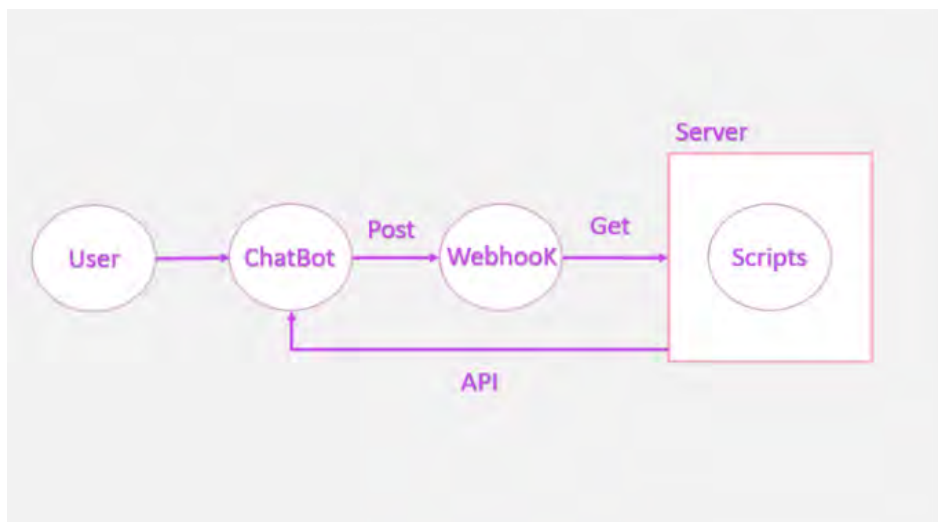
3. CLI คล้ายกับ SublimeText ที่มี subl บรรทัดคำสั่งรหัส Visual Studio ติดตั้ง CLI ชื่อ รหัส และติดตั้งง่าย ใน Windows CLI จะถูกติดตั้งพร้อมกับแอปแล้ว CLI สามารถเข้าถึงได้ผ่าน Command Prompt ใน OS X สามารถติดตั้งและถอนการติดตั้งผ่าน Command Palette



รูปที่ 2.6 หน้าต่างของโปรแกรม VisualStudioCode CLI

## 2.4 Webhook

Webhook คือการใช้งาน API แบบสลับข้างกัน โดยปกติแล้วจะมี “ผู้ให้บริการ” เปิดตัว API ให้ผู้ใช้ได้ใช้งาน เมื่อ “ผู้ใช้” ต้องการใช้งานตัวบริการหรือแอปพลิเคชันนั้น ก็จะส่ง Request ไปที่ URL ดังกล่าวจึงจะได้รับข้อมูลกลับมา แต่ Webhook จะไม่มีตัว URL ให้ “ผู้ให้บริการ” แต่กลับกัน “ผู้ใช้” จำเป็นที่จะต้อง มี URL หรือก็คือ API ของผู้ให้บริการเพื่อจะรับข้อมูลกลับมา



รูปที่ 2.7 ภาพการทำงานของ Webhook

#### ข้อดี Webhook

ข้อดีคือ Confirm ว่า Realtime แน่แน่นอน และทำให้ไม่ต้องเสียเวลา Query ซ้ำๆ

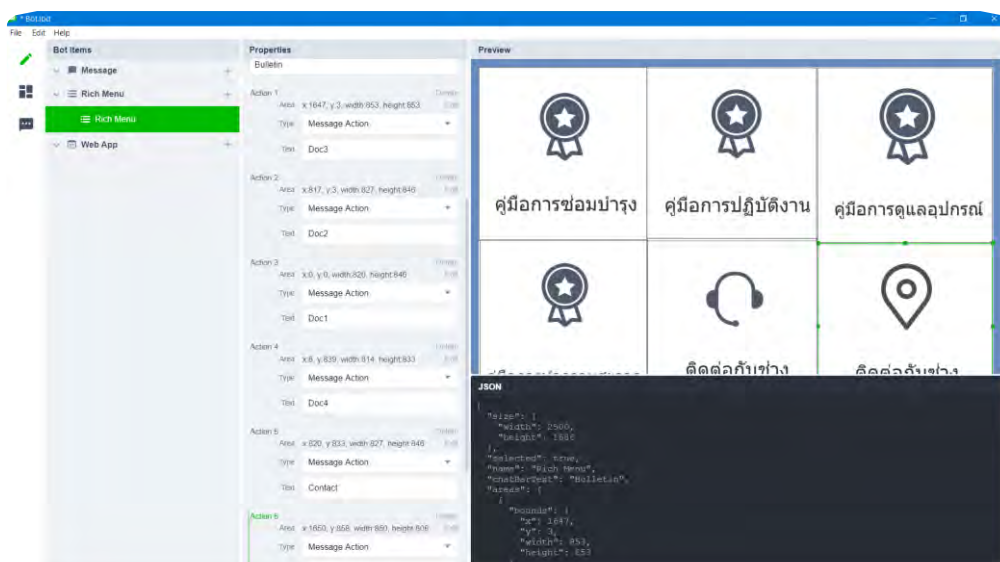
#### ข้อเสีย Webhook

ข้อเสียคือ ทางต้นทางจะต้องเพิ่มงานมีการ Provide API ไว้ทำ Webhook

### 2.5 Line Bot Designer

เป็น โปรแกรมแชทที่มีไว้สำหรับการจำลองแชท Line ขึ้นมาเพื่อไว้ให้สำหรับผู้พัฒนาสร้าง และพัฒนาตัวของ Line Bot ได้ง่ายขึ้น โดยใน โปรแกรมจะสามารถสร้างข้อความธรรมดา, Rich Menu, ข้อความแบบบอลลูน และ Web app ภายใน LINE ได้เป็นต้น โดยโปรแกรมจะมีแถบ Chat ไว้สำหรับจำลองหน้าต่างแชทของ LINE ขึ้นมาให้ผู้พัฒนาสามารถทดสอบระบบได้



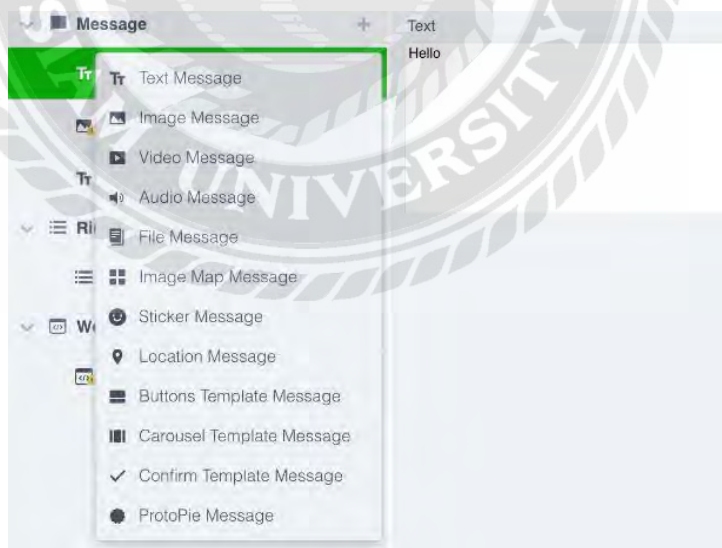


รูปที่ 2.8 ตัวอย่างของเว็บไซต์ Line Bot Designer

Bot Items รูปแบบข้อความทั้งหมดบน LINE

### 1.Message

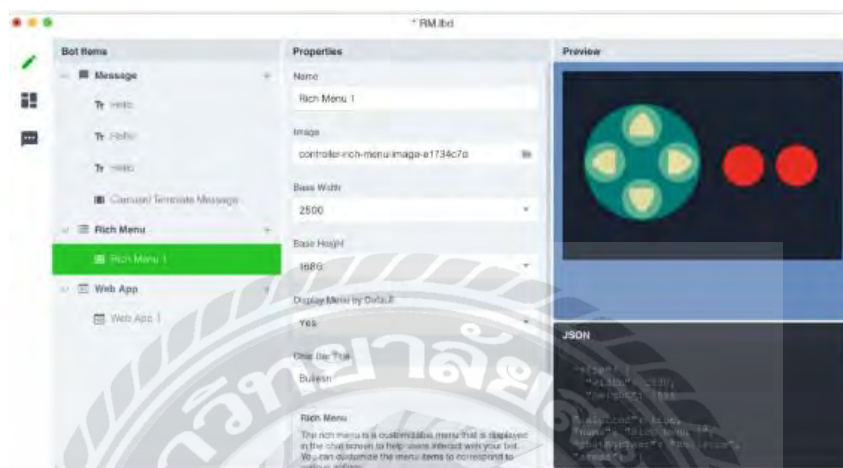
รูปแบบข้อความทั่วไปที่เราจะได้อีกน้อยที่สุด มีข้อความทั้งหมด 12 รูปแบบ เมื่อเราเลือกมาวางบน list ของ item ระบบจะให้เราใส่ข้อมูลต่างๆ ลงในฟิลด์ จากนั้นจะขึ้น preview ให้เราดูทางด้านขวา



รูปที่ 2.9 ตัวอย่างของ Bot Items Message

## 2. Rich Menu

เมนูที่เหมาะสมสำหรับการทำ Call to Action มีความยืดหยุ่นในการออกแบบเมนู นำไปประยุกต์ใช้ได้หลากหลาย บนโปรแกรมนี้ทำได้เพียงวางภาพเพื่อให้เห็นหน้าตาของเมนูเท่านั้น ไม่ได้ถึงกับสร้าง Rich Menu แบบทดลองใช้งานจริงได้ เพราะเป็นรูปแบบข้อความที่ค่อนข้างซับซ้อน ต้องเขียนโปรแกรมจริงเพื่อระบุตำแหน่งของปุ่มต่างๆ



รูปที่ 2.10 ตัวอย่างของ Bot Items Rich Menu

## 3. WebApp

เป็นรูปแบบการแสดงผลหน้าเว็บในข้อความแชทของ LINE เหมือนมี web browser อยู่ในกล่องแชท สามารถใช้งาน ได้เหมือนอยู่บนหน้าเว็บ แต่ไม่ต้องสลับแอปไปเปิดบน browser ในโปรแกรมเพียงเราวาง URL ลงใน property



รูปที่ 2.11 ตัวอย่างของ Bot Items WebApp

## 2.6 Python

เป็นภาษาการเขียนโปรแกรมที่ใช้อย่างแพร่หลายในเว็บแอปพลิเคชัน การพัฒนาซอฟต์แวร์ วิทยาศาสตร์ข้อมูล และ Machine Learning (ML) นักพัฒนาใช้ Python เนื่องจากมีประสิทธิภาพ เรียนรู้ง่าย และสามารถทำงานบนแพลตฟอร์มต่างๆ ได้มากมาย



รูปที่ 2.12 รูปภาษา Python

### ข้อดีของPython

1. นักพัฒนาสามารถอ่านและทำความเข้าใจโปรแกรมPythonได้อย่างง่ายดาย
2. Python ทำให้นักพัฒนาทำงานได้อย่างมีประสิทธิภาพมากขึ้น เนื่องจากพวกเขาสามารถเขียนโปรแกรม Python ได้โดยใช้โค้ดน้อยลงเมื่อเปรียบเทียบกับภาษาอื่นๆ อีกมากมาย
3. Python มีไลบรารีมาตรฐานขนาดใหญ่ที่มีโค้ดที่ใส่ซ้ำได้สำหรับเกือบทุกงาน ด้วยเหตุนี้ นักพัฒนาจึงไม่ต้องเขียนโค้ดขึ้นใหม่ทั้งหมด
4. โดยนักพัฒนาสามารถใช้ Python ร่วมกับภาษาการเขียนโปรแกรมชนิดอื่น ๆ เช่น Java, C และ C++ ได้ง่าย ๆ
5. ทั้งนี้ชุมชน Python ในปัจจุบันมีนักพัฒนาที่พร้อมให้การสนับสนุนหลายด้านคนทั่วโลก หากประสบปัญหาคุณสามารถรับการสนับสนุนอย่างรวดเร็วได้จากชุมชน
6. Python สามารถใช้งานได้ในระบบปฏิบัติการคอมพิวเตอร์ต่างๆ เช่น Windows, macOS, Linux และ Unix



## 2.7 Python Flask

คือ Web framework ที่เขียนขึ้นมาสำหรับ Python เพื่อใช้ร่วมกัน webserver เช่น Apache และได้รับการยอมรับจาก community we pages ชื่อนำเช่น Pinterest, LinkedIn เป็นต้น โดย Flask ถูกเรียกว่า micro framework เพราะว่ามันไม่ต้องการเครื่องมือหรือ library อะไรมาก อีกทั้งไม่จำเป็นต้องมี database ด้วย

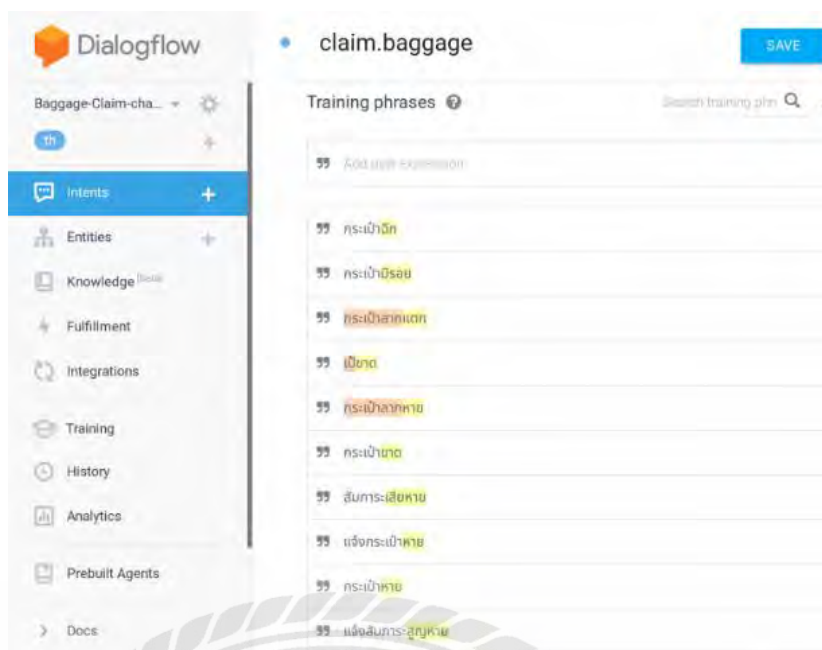


รูปที่ 2.13 Python Flask

## 2.8 Dialogflow

Dialogflow เป็นเครื่องมือสร้าง chatbot จาก Google ซึ่งมีความโดดเด่นในเรื่อง Natural Language Processing หรือ NLP หมายถึงการที่ chatbot สามารถทำความเข้าใจความหมายของประโยคที่ผู้ใช้พิมพ์ส่งเข้ามาได้อย่างถูกต้องทำให้ chatbot สามารถโต้ตอบกับผู้ใช้งานได้แม่นยำและตรงประเด็น

การใช้งาน Dialogflow นั้น ช่วยลดความยุ่งยากในการสร้าง engine NLP เนื่องจาก Dialogflow จะจัดการส่วนนี้ให้ สิ่งที่เราต้องทำคือการกำหนด intent ของ chatbot ว่าต้องการให้ chatbot สามารถตอบอะไรได้บ้าง จากนั้น เราแค่ใส่ประโยคตัวอย่าง (training phrase) ที่ตรงกับ intent นั้นๆ เข้าไปให้ chatbot เรียนรู้ ( Google แนะนำว่าควรมี 10 ประโยคสำหรับแต่ละ intent )



รูปที่ 2.14 ตัวอย่าง chatbot สำหรับรับแจ้งข้อความโดย Training phrase

Dialogflow ยังมีเครื่องมือที่ถูกออกแบบมาให้สามารถรองรับภาษาโปรแกรมได้หลากหลายภาษาเพื่อสำหรับการออกแบบหรือสร้างระบบติดต่อสื่อสารระหว่างผู้คนอย่างง่ายโดยส่วนมากมักจะเน้นไปที่แอปพลิเคชันสำหรับติดต่อสื่อสารในโทรศัพท์พกพาและตัว Bot เป็นส่วนใหญ่ เช่น JSO หรือ Python เป็นต้น โดยที่ตัว Dialogflow นั้นจะประกอบไปด้วย

**2.8.1 Agents** เป็นส่วนของตัวแทนที่ทำหน้าที่คอยตอบกลับข้อความของผู้ใช้ โดยที่แต่ละโปรเจกต์จะสามารถมี Agent ได้เพียงแค่ตัวเดียวเท่านั้น

**2.8.2 Intents** เป็นส่วนหัวข้อย่อยลงมาในตัว Agents มีไว้สำหรับสร้างหรือเพิ่มสิ่งที่ต้องการให้ตัว Agents ตอบกลับ ไปหาผู้ใช้งาน

**2.8.3 Responses** เป็นส่วนที่อยู่ข้างในของ Intents โดยมีหน้าที่ในการตอบกลับผู้ใช้งานด้วยการตอบกลับรูปแบบต่างๆ เช่น ข้อความ, รูปภาพ และเสียง เป็นต้น

**2.8.4 Entities** เป็นส่วนที่มีไว้สำหรับกำหนดประเภทของข้อมูลที่เราต้องการดึงมาจากผู้ใช้งานหรือดึงมาจากภายนอก ไม่ว่าจะมียูเออยู่แล้วหรือสร้างขึ้นก็ตาม

**2.8.5 Fulfillments** เป็นส่วนที่มีไว้สำหรับดึงข้อมูลจากภายนอกหรือภายใน Dialogflow เข้ามาในระบบ เพื่อไว้สำหรับใช้งานหรือเพิ่มลงไปในตัว Intents ไม่ว่าจะทำการดึงผ่าน URL หรือผ่านทางใดทางหนึ่งก็ตาม



รูปที่ 2.15 Dialogflow

## 2.9 Heroku

คือแพลตฟอร์มที่ให้บริการด้านคลาวด์ที่สามารถรองรับภาษาโปรแกรมได้หลากหลายภาษา เช่น Python, PHP, Java, Node, Go, Scala และ Clojure เป็นต้น เป็นแพลตฟอร์มที่มีไว้สำหรับการทดสอบและสร้าง Webapp ขึ้นมาโดยสามารถปรับให้ยืดหยุ่นได้ตามความต้องการของผู้ใช้



รูปที่ 2.16 Heroku

# บทที่ 3

## รายละเอียดและการปฏิบัติงาน

### 3.1 ชื่อและที่ตั้งของสถานประกอบการ

บริษัท พีไฟฟ์ แมเนจเม้นท์ จำกัด

ที่อยู่ : 84 10 ถนนพระราม2 ซอย69 แขวง แสมดำ เขต บางขุนเทียน กรุงเทพมหานคร 10150

โทรศัพท์ : 02-464-7019

อีเมล : wilawan@p5-management.com

เว็บไซต์ : <https://www.p5-management.com/>



รูปที่ 3.1 แผนที่ตั้งสถานประกอบการ

### 3.2 ลักษณะการประกอบการ

บริษัท พีไฟฟ์ แมเนจเม้นท์ จำกัด เป็นผู้ให้บริการเกี่ยวกับ เครื่อง POS ในประเทศไทย ให้บริการในด้านของ Smart IT Solution for Your Business

บริษัท พีไฟฟ์ แมเนจเม้นท์ จำกัด เป็นผู้เชี่ยวชาญในด้านนำเสนอวิธีการ กระบวนการ และ แนวทางต่างๆ ที่เกี่ยวข้องกับ Smart IT Solution for Your Business ทั้งนี้จะตอบสนองความต้องการของผู้รับบริการด้วยเหตุนี้ บริษัท พีไฟฟ์ แมเนจเม้นท์ จำกัด จึงมีบริการที่ครอบคลุมความต้องการของผู้รับบริการ

จากประสบการณ์ 9 ปีที่ผ่านมา บริษัท พีไฟฟ์ แมเนจเม้นท์ จำกัด จึงได้รับความไว้วางใจจากองค์กรชั้นนำระดับประเทศ



รูปที่ 3.2 บริการ Helpdesk Support ของทาง บริษัทพีไฟฟ์ แมเนจเม้นท์ จำกัด

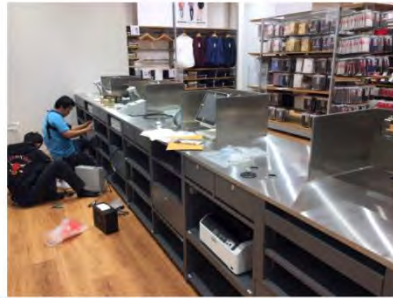


รูปที่ 3.3 บริการ Network ของทาง บริษัทพีไฟฟ์ แมเนจเม้นท์ จำกัด



**MA**

สัญญาดูแลรักษาอุปกรณ์ IT โดยให้บริการ สิ่งผิดปกติเบื้องต้น โดยทีม Help Desk หากไม่สามารถแก้ไขปัญหาได้ในระยะเวลาที่กำหนด ทางบริษัทจะส่งทีมช่างลงไปแก้ไขเพื่อให้งานของลูกค้าเป็นไปได้อย่างปกติ อย่างรวดเร็วที่สุด โดยระบบการมีมาตรฐานภายใต้ข้อกำหนด SLA (Service Level Agreement) เพื่อให้ลูกค้ามั่นใจได้ว่า ปัญหาที่เกิดขึ้นจะถูกแก้ไขภายในเวลาที่กำหนด



**บริการติดตั้ง**

เป็นงานติดตั้งอุปกรณ์ Point of Sales รวมไปถึงอุปกรณ์ IT เช่น Access point, CCTV, Weight scale, ระบบ Network เป็นต้น

รูปที่ 3.4 บริการการติดตั้งของทาง บริษัทพีไฟฟ์ แมเนจเม้นท์ จำกัด



รูปที่ 3.5 ตัวอย่างผู้ใช้บริการของทางบริษัทพีไฟฟ์ แมเนจเม้นท์ จำกัด

### 3.3 ตำแหน่งและลักษณะงานที่ได้รับมอบหมาย

3.3.1 ตำแหน่ง Service desk และ Service on site

3.3.2 งานที่ได้รับมอบหมาย

- ทำการเรียนรู้เกี่ยวกับงานในบริษัทด้าน Hardware ซ่อม,ทดสอบ,ติดตั้งเครื่อง,ติดตั้งสายLan และออกแก้ปัญหาหน้างานเกี่ยวกับอุปกรณ์เครื่อง POS
- ทำงานเกี่ยวกับการแก้ปัญหาของตัวเครื่องPOS โดยทำการรีโมทเพื่อแก้ปัญหา

### 3.4 ชื่อและตำแหน่งของพนักงานที่ปรึกษา

1. นาย ชนสาร สระกบแก้ว
2. นาย สุทธิ โพธิ์สุข

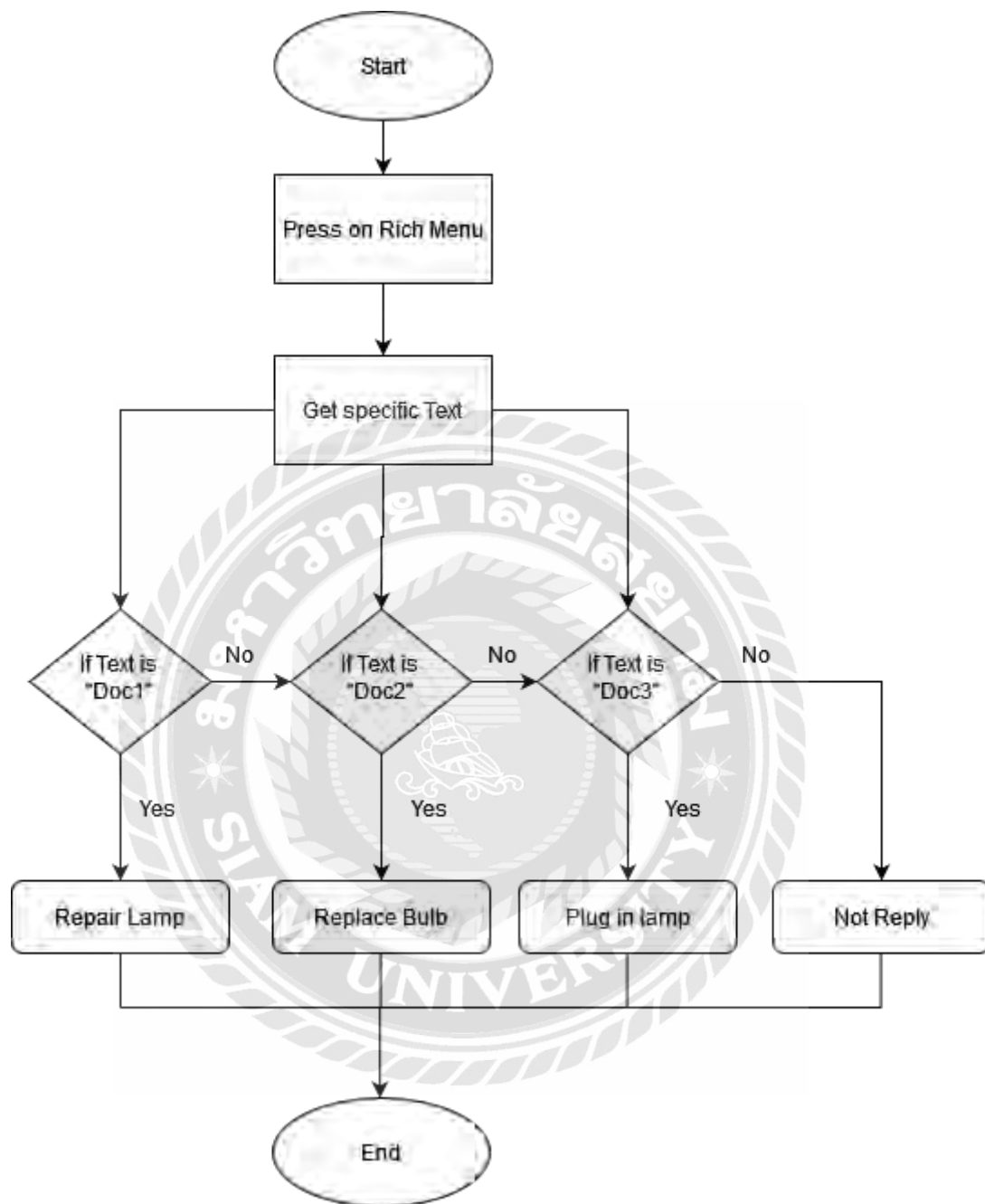
### 3.5 ระยะเวลาในการปฏิบัติงาน

ระยะเวลาในการปฏิบัติงาน เริ่มวัน ที่ 23 พฤษภาคม พ.ศ. 2565 ถึง วันที่ 2 กันยายน พ.ศ. 2565

ตารางที่ 3.1 ระยะเวลาในการดำเนินโครงการ

ขั้นตอนการดำเนินงาน	พ.ค. 65	มิ.ย 65	ก.ค. 65	ส.ค. 64
1. ศึกษา Line Developer	←→			
2. ออกแบบหน้าตาบอทไลน์		←→		
3. สร้างตัวบอทไลน์		←→		
3. เพิ่มข้อความการตอบกลับ			←→	
4. เพิ่มหน้าต่าง Rich Menu			←→	
5. ทดสอบการใช้งาน			←→	
6. จัดทำเอกสารและส่งมอบ				←→

## 3.6 หลักการทำงานของ Flowchart



รูปที่ 3.6 Flowchart ของระบบ Messaging API



Messaging API ของ Line Bot จะเริ่มต้นโดยการอยู่ในสภาพเตรียมพร้อมเพื่อที่จะรับข้อความจากผู้ใช้งาน หลังจากที่ผู้ใช้งานดำเนินการเลือกคำสั่งจากหน้าของ Rich Menu แล้ว ตัว Rich Menu จะทำการส่ง Keywords ที่ถูกกำหนดไว้ไปหา Messaging API หาก Keywords นั้นตรงกับเงื่อนไขที่ค้กำหนดไว้ ชุดคำสั่งจะทำการส่งข้อความกลับไปหาผู้ใช้งานตามที่ได้ถูกกำหนดไว้เช่นกัน แต่หาก Keywords ที่ถูกส่งมานั้นไม่ตรงกับเงื่อนไขที่กำหนดไว้ ชุดคำสั่งจะทำการตรวจสอบกับเงื่อนไขต่อไปเรื่อยๆจนกว่าจะพบเงื่อนไขที่ตรงหรือไม่พบเงื่อนไขที่ตรง หากตรวจสอบจนครบแล้วไม่พบเงื่อนไขที่ตรง ชุดคำสั่งจะไม่ทำการตอบกลับข้อความใดๆกับผู้ใช้งาน

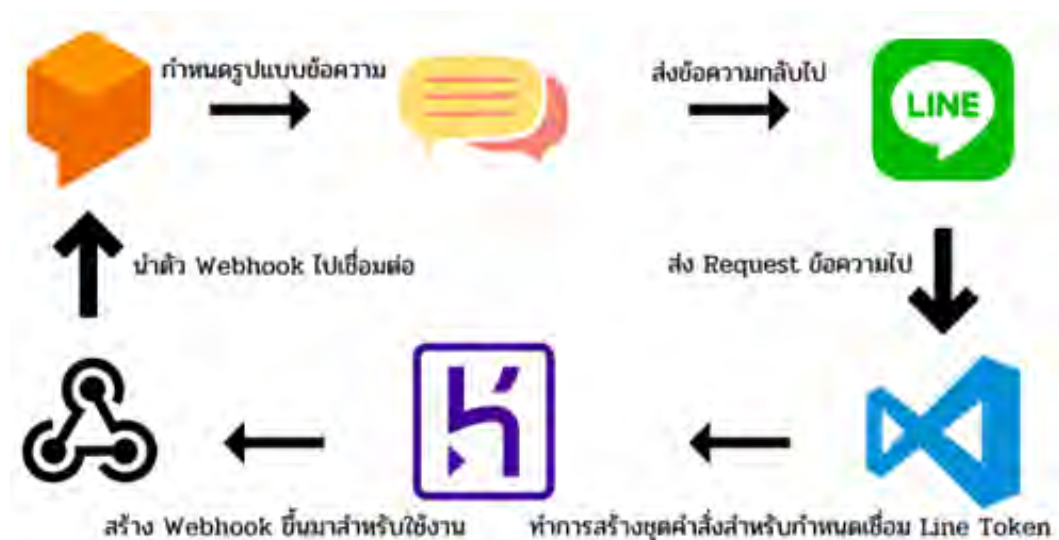
### 3.7 คำอธิบายขั้นตอนและวิธีการดำเนินงาน

ผู้ใช้งานทำการเลือกคำสั่งจากการคลิกแถบรายการที่ขึ้นทางหน้าจอ หลังจากคลิกแถบรายการตัวผู้ใช้งานจะต้องพิมพ์ข้อความค่าสำคัญไปที่ตัวไลน์และตัวไลน์หลังจากนั้นระบบจะเอาค่าสำคัญนั้นไปตรวจสอบกับเงื่อนไขที่กำหนดไว้ว่าใช่หรือไม่ใช่ถ้าตรงกับที่กำหนดไว้ระบบก็จะส่งวิธีซ่อมแซมที่เป็นข้อความกลับไปให้ถ้าไม่ตรงระบบจะไปตรวจสอบในเงื่อนไขต่อไป

3.7.1 ทำการเขียนวิธีการทำงานของตัว Linebot ขึ้นมาเพื่อนำไปใช้เป็นแนวทางในการสร้างตัว

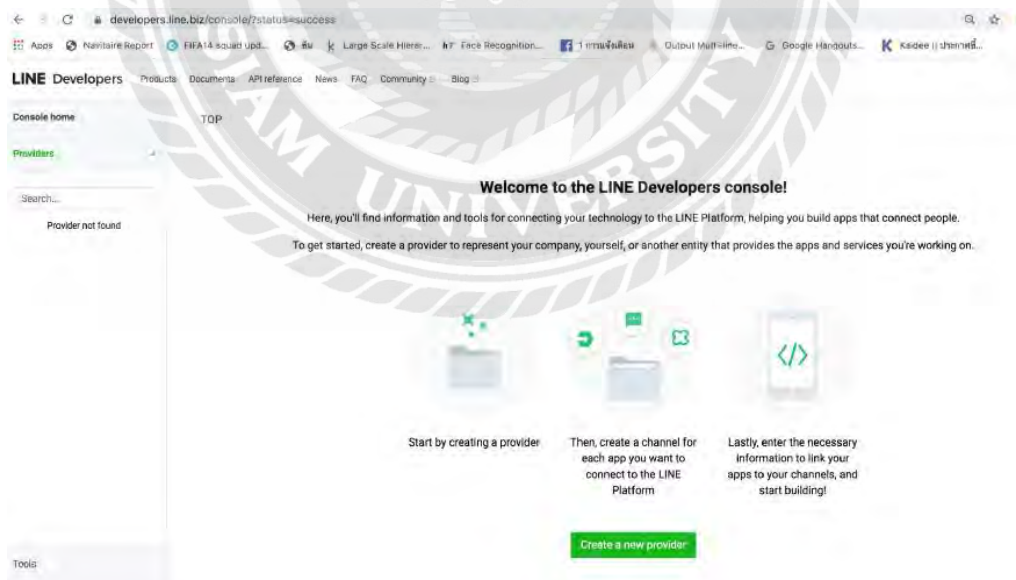
Line bot ขึ้นมาในภายหลังโดยมีหน้าที่การทำงานคร่าวๆ ดังนี้

- Line = ทำหน้าที่เป็นสื่อกลางในการรับและส่งข้อความและผลลัพธ์ที่จะเกิดขึ้น
- Visual Studio code = ทำหน้าที่ในการคงการทำงานของชุดคำสั่งที่ไว้สำหรับกำหนดประเภทของข้อความและเชื่อมต่อกับ Messaging API ของ Line
- Heroku = มีหน้าที่ในการสร้างและเปิด Webhook ที่ไว้สำหรับการเชื่อมต่อกับ Messaging API และ Dialogflow
- Dialogflow = ไว้สำหรับการกำหนดข้อความและจำนวนบอลลูนข้อความที่ต้องการให้ Messaging API ตอบกลับได้



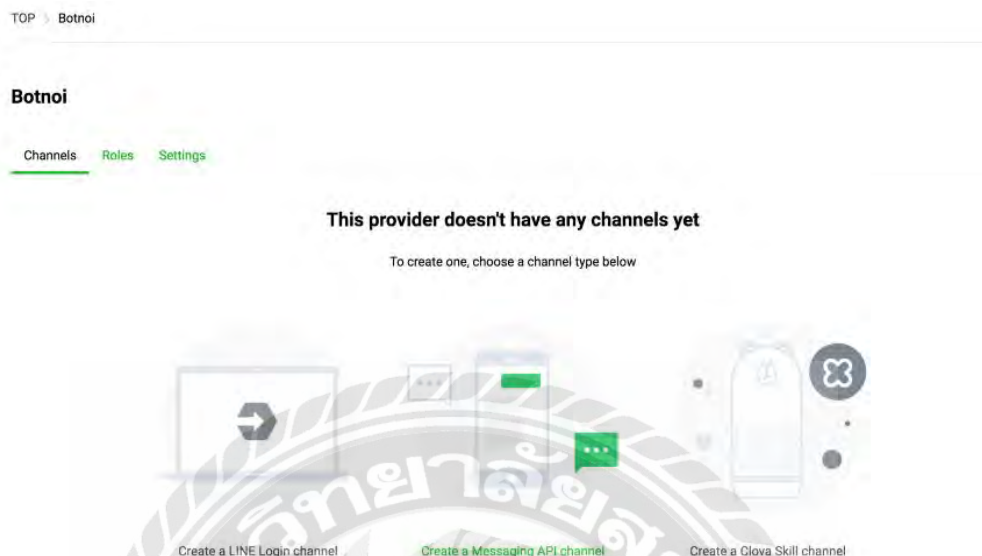
รูปที่ 3.7 ตัวอย่างของการทำงานเบื้องต้นของ Line Bot

3.7.2 ทำการ Create Provider ในหน้าเว็บไซต์ของ Line Developer เพื่อเอาไว้สำหรับจัดเก็บ Channel ของ Line app แต่ละอัน โดย Line app แต่ละอันจะมีจุดประสงค์ในการใช้งานที่ต่างกันไป โดยจะมีให้เลือกดังนี้ Line Login channel, Messaging API channel และ Clova Skill Channel เป็นต้น



รูปที่ 3.8 การสร้าง Provider

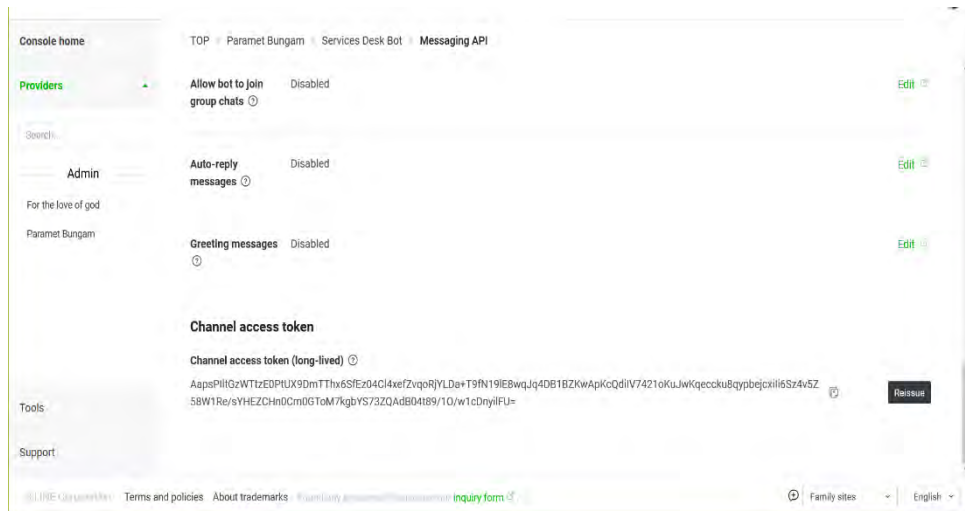
3.7.3 สร้าง Channel ใน Provider ของ Line Developer เพื่อไว้สำหรับการสร้าง Line app โดยในโครงการนี้ทางผู้จัดทำได้ทำการสร้าง Messaging API channel เพื่อไว้สำหรับสร้าง Line Bot ที่สามารถตอบโต้ข้อความกับผู้ใช้งานได้



รูปที่ 3.9 การสร้าง Messaging API channel

3.7.4 หลังจากทำการสร้าง Messaging API แล้ว ให้เลือกไปที่แถบที่มีชื่อว่า Messaging API เพื่อที่จะทำการเพิ่มเพื่อนกับ Line Bot และกำหนดการตั้งค่าต่างๆ เช่น

- Webhook Setting = ไว้สำหรับกำหนด URL ของ Webhook ที่มีอยู่แล้วหรือที่ถูกสร้างขึ้น
- Line Official Account Features = ไว้สำหรับกำหนดให้ Messaging API สามารถตอบกลับผู้ใช้งานด้วยข้อความที่ถูกกำหนดไว้ในเว็บไซต์ Line Official Account หรือไม่
- Channel access Token = ไว้สำหรับการนำ Token ของ Messaging API ไปเชื่อมกับ Visual Studio code ในภายหลัง



รูปที่ 3.10 การตั้งค่าในแถบ Messaging API

3.7.5 ทำการปิดการใช้งานของ Auto-reply Message ในหมวด Line Official Account Features ในแถบของ Messaging API เพื่อไม่ให้ Messaging API ดึงข้อความตอบกลับที่ถูกตั้งไว้ในเว็บไซต์ Line Official Account มาใช้งานแทน Webhook

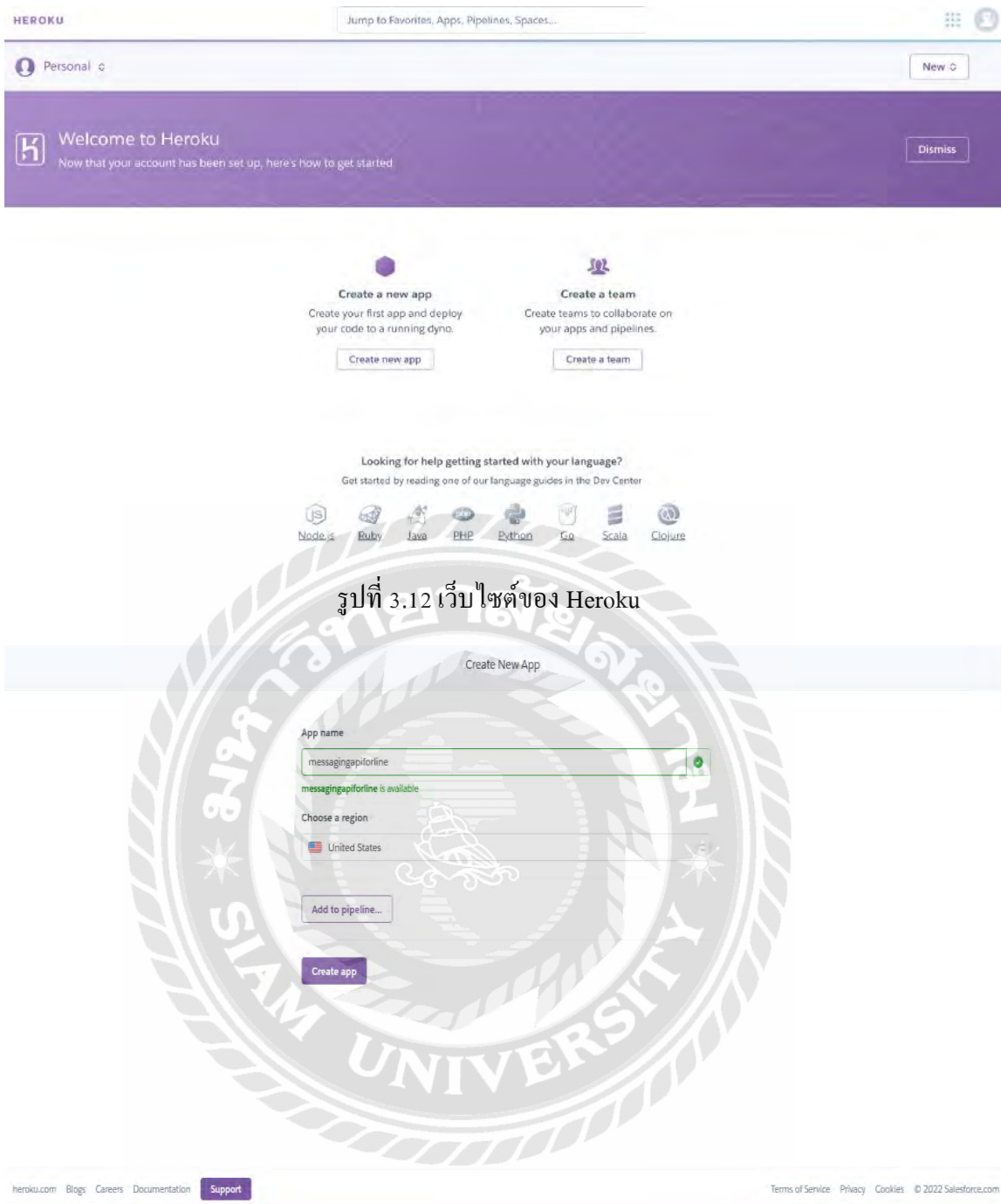
#### LINE Official Account features

Edit the message text and other settings for these features in the LINE Official Account Manager



รูปที่ 3.11 ตัวอย่างการปิดการใช้งาน Auto-reply Message

3.7.6 ทำการสร้าง App บนเว็บไซต์ของ Heroku โดยการเลือกไปที่ Create a new app แล้วทำการเลือกภูมิภาคของ App โดยจะมีให้เลือกอยู่ 2 ที่ในปัจจุบันคือ USA และ Europe แล้วทำการตั้งชื่อ App เป็นอันเสร็จสิ้น



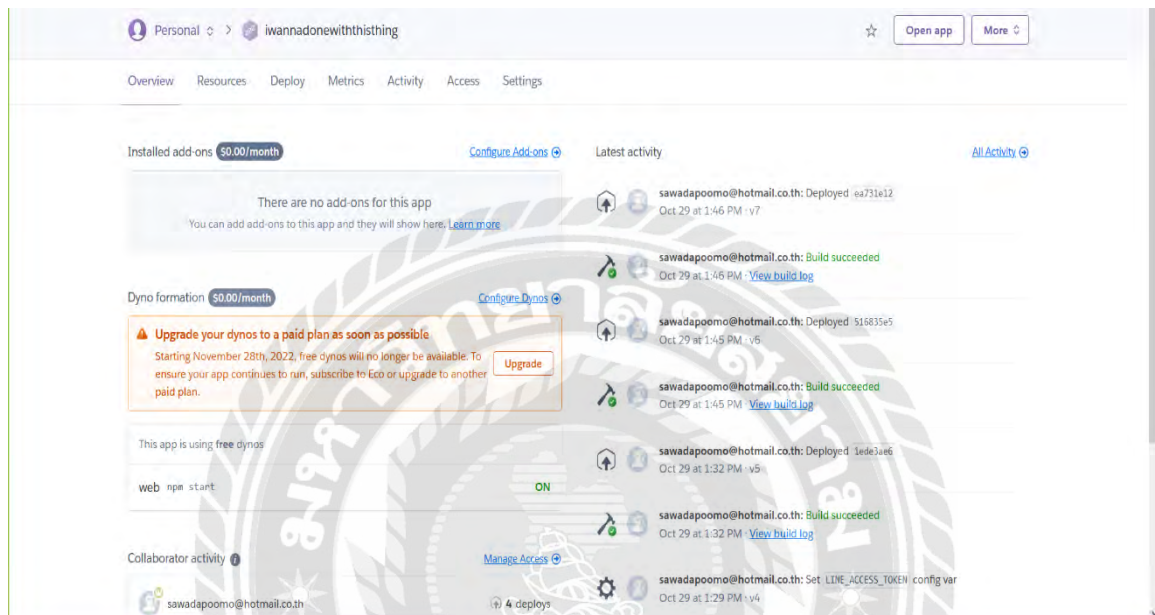
รูปที่ 3.12 เว็บไซต์ของ Heroku

รูปที่ 3.13 ตัวอย่างการ Create a new app บนเว็บไซต์ของ Heroku

3.7.7 หลังจากสร้าง App เสร็จสิ้น ทาง Heroku จะนำเราเข้าไปที่หน้า Dashboard ของ App และทำการสร้าง Git Repository ให้ไว้สำหรับต้นชุดคำสั่งขึ้นไปบนตัว Webserver ที่ได้สร้างขึ้น โดยสามารถทำการต้นชุดคำสั่งนี้ผ่านทาง Command Line ได้โดยตรง

- Git Repository = โพลเดอร์ที่ใช้สำหรับเก็บข้อมูลของโปรเจกต์นั้นๆ ที่ถูกสร้างขึ้น ส่วนมากมักจะไว้สำหรับเก็บชุดคำสั่งที่ต้องการใช้งาน โดยใน Heroku มีไว้สำหรับเก็บชุดคำสั่งที่ถูกดึงขึ้นมาจากการใช้ Command Line

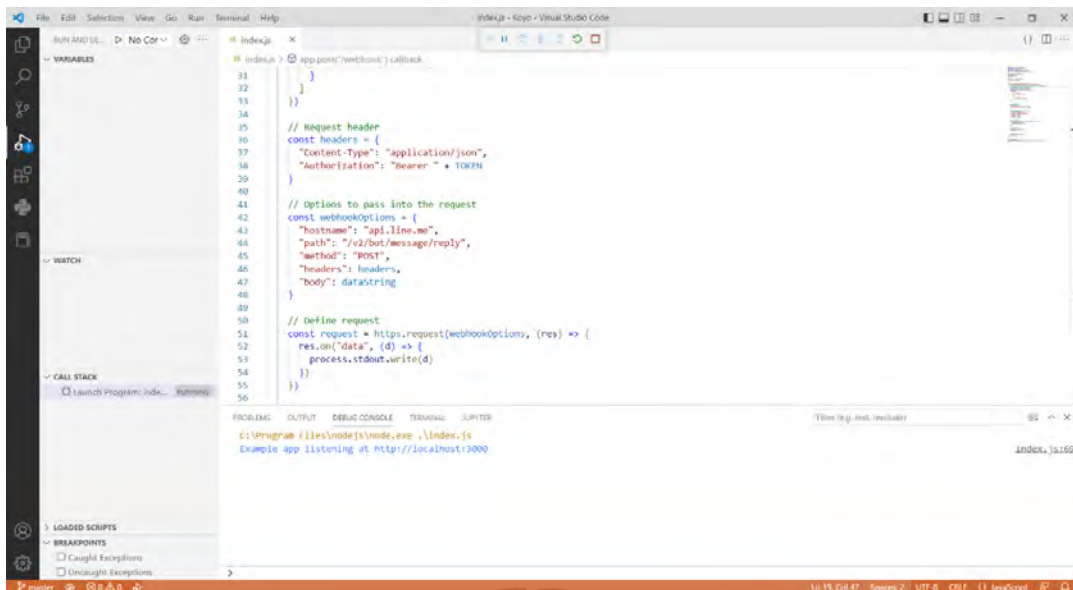
- Command Line เป็นหน้าโปรแกรมที่จะรับคำสั่งแบบข้อความเข้าไปที่ละบรรทัดแล้วทำการหาความหมายของคำสั่งบรรทัดนั้นๆ เพื่อนำไปใช้งานในระบบปฏิบัติการของเราด้วยคำสั่งนั้นๆต่อไป



รูปที่ 3.14 หน้าต่างสำหรับดูภาพรวมของ App บน Heroku

3.7.8 ทำการสร้างชุดคำสั่งที่ไว้สำหรับกำหนดประเภทของข้อความและเพื่อไว้สำหรับเชื่อม Line Channel Token กับชุดคำสั่งบน Visual Studio code โดยการเชื่อมชุดคำสั่งกับ Line Channel Token มีประโยชน์ในการช่วยให้ชุดคำสั่งสามารถรู้ได้ว่าชุดคำสั่งนี้ควรจะเป็นของ Line channel ไหร





```

31   }
32   }
33   })
34
35   // request header
36   const headers = {
37     "Content-Type": "application/json",
38     "Authorization": "bearer " + TOKEN
39   }
40
41   // Options to pass into the request
42   const webhookOptions = {
43     "hostname": "api.inline.me",
44     "path": "/v2/bot/message/reply",
45     "method": "POST",
46     "headers": headers,
47     "body": datastring
48   }
49
50   // Define request
51   const request = https.request(webhookOptions, (res) => {
52     res.on("data", (d) => {
53       process.stdout.write(d)
54     })
55   })
56

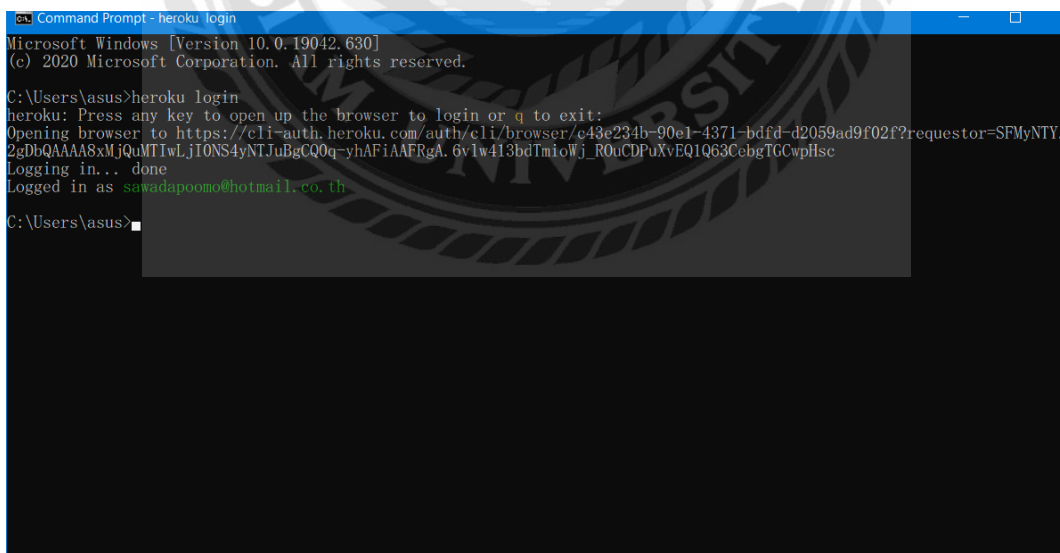
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPITER

Example app listening at http://localhost:3000

รูปที่ 3.15 หน้าการเขียนและทำงานของชุดคำสั่งบน Visual Studio Code

3.7.9 เปิดโปรแกรม Command line ขึ้นมาเพื่อเริ่มขั้นตอนการค้นชุดคำสั่งโดยการเริ่มจากการใช้งานคำสั่ง Heroku login เพื่อสำหรับเข้าสู่ระบบของ Heroku ผ่านทางโปรแกรม Command line



```

Command Prompt - heroku login
Microsoft Windows [Version 10.0.19042.630]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\asus>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/cli/browser/c43e234b-90e1-4371-bdfd-d2059ad9f02f?requester=SFMyNTY.2gDbQAAAASxMjQuMTIwLjT0NS4yNTJubGQ0Qq-yhAFiAAFRgA.6v1w413bdTmoWj_R0uCDPuXvEQ1Q63CebgTGCwpHsc
Logging in... done
Logged in as sawadapoom@hotmail.co.th

C:\Users\asus>

```

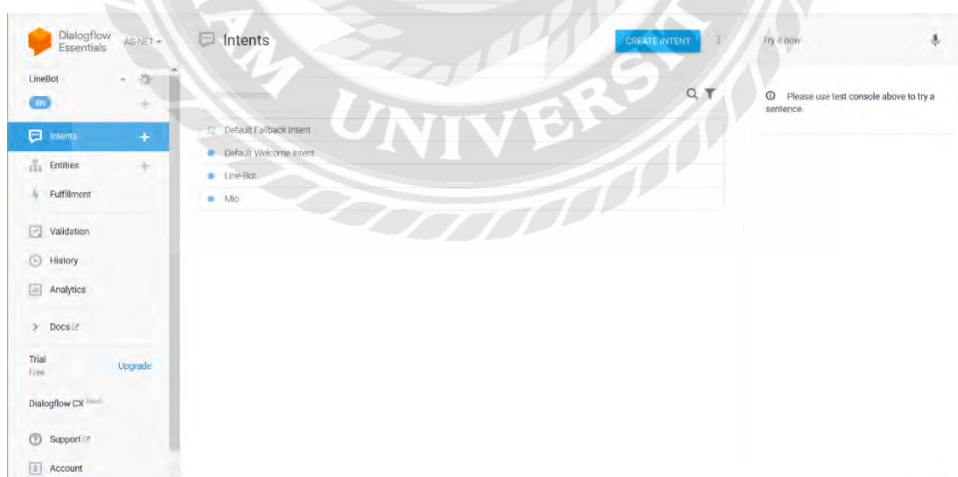
รูปที่ 3.16 หน้าต่างโปรแกรม Command line

3.7.10 หลังจากนั้นทำการตั้งค่าของ Heroku เพื่อที่จะทำการดันชุดคำสั่งขึ้นไปยัง Heroku บนเว็บไซต์ ดังนี้

- Git Init = เพื่อสร้างโฟลเดอร์สำหรับเก็บโปรเจกบนคอมพิวเตอร์
- Heroku git:remote -a (ชื่อของ App ที่สร้างบนหน้าเว็บไซต์ Heroku) = เพื่อให้โปรแกรม Command line สามารถเชื่อมต่อกับ App บน Heroku ได้ โดยที่ -a มีไว้สำหรับกำหนดให้คำสั่งนี้ส่งผลกับ App ใดโดยตรง
- git commit -am “(ชื่อตามต้องการ)” = เพื่อไว้สำหรับการเก็บหลักฐานการเปลี่ยนแปลงของชุดคำสั่งใน Git Repo บนเว็บไซต์ของ Heroku
- git push heroku master = เป็นคำสั่งสุดท้ายสำหรับทำการดันชุดคำสั่งขึ้นไปบน Heroku Master หรือ App บนเว็บไซต์ Heroku

3.7.11 หลังจากทำตามขั้นตอนด้านบนเสร็จแล้ว Heroku จะสร้าง URL สำหรับ Webhook ขึ้นมาให้ โดยที่จะสามารถนำไปใช้ได้กับ Line Developer และ Dialogflow

3.7.12 ทำการสร้าง Agents บนหน้าเว็บไซต์ของ Dialogflow และจะต้องตั้งชื่อของ Agents และกำหนดภูมิภาคให้กับตัว Agents นั้นๆ โดย Agents เปรียบเสมือนคนกลางที่คอยรับข้อความที่ได้จาก Webserver ของ Heroku และส่งข้อความกลับไป Messaging API

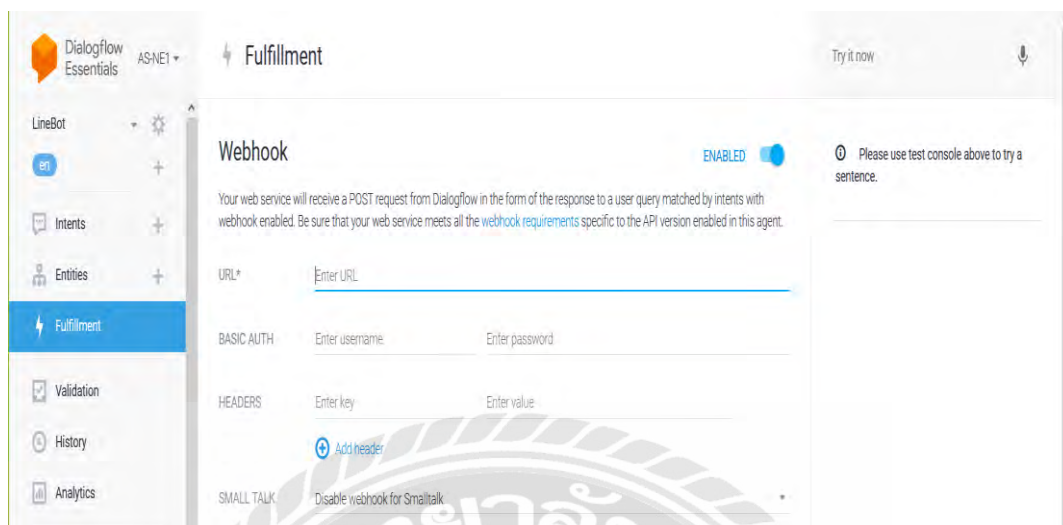


รูปที่ 3.17 หน้าเว็บไซต์ของ Dialogflow

3.7.13 สร้าง Intents ของ Agents ขึ้นมาเพื่อมีไว้สำหรับการกำหนดข้อความตอบกลับของตัว Messaging API โดยจะกำหนดเป็นข้อความธรรมดาหรือแบบพิเศษได้

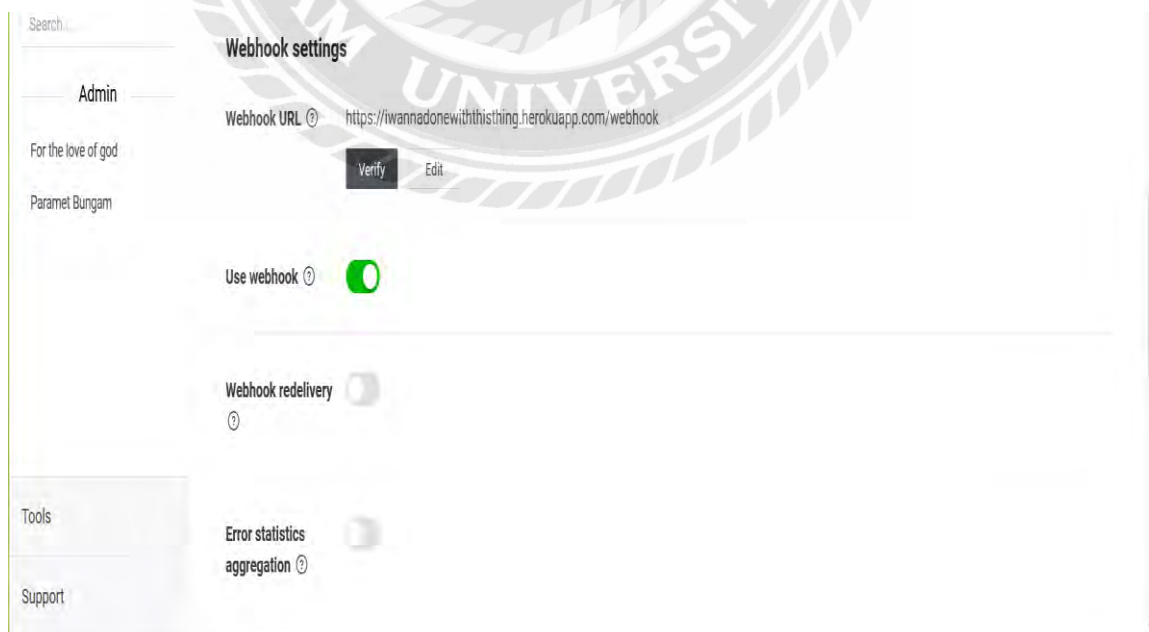


3.7.14 หลังจากกำหนดตัว Intents ให้ไปยังแถบของ Fulfillment เพื่อทำการเปิดใช้งานการเชื่อมต่อกับ Webhook ของตัว Dialogflow เอง โดยใส่ URL ของ Webhook ที่ได้จาก Heroku ลงไป



รูปที่ 3.18 การเปิดใช้งาน Webhook บน Dialogflow

3.7.15 หลังจากเสร็จทุกขั้นตอนแล้วให้นำ Webhook ที่ได้จาก Heroku แล้วเป็น Webhook เดียวกับ Dialogflow ใส่ไปที่ช่อง Webhook Setting ของ Messaging API บนเว็บไซต์ Line Developers เพื่อเปิดใช้งานและเชื่อมต่อ Webserver กับ Messaging API ก็เป็นอันเสร็จสิ้น



รูปที่ 3.19 การเปิดใช้งาน Webhook บน Line Developers

### 3.8 อุปกรณ์และเครื่องที่ใช้

#### 3.9.1 ฮาร์ดแวร์

- คอมพิวเตอร์โน้ตบุ๊ก 1 เครื่อง
- สมาร์ทโฟน 1 เครื่อง

#### 3.9.2 ซอฟต์แวร์

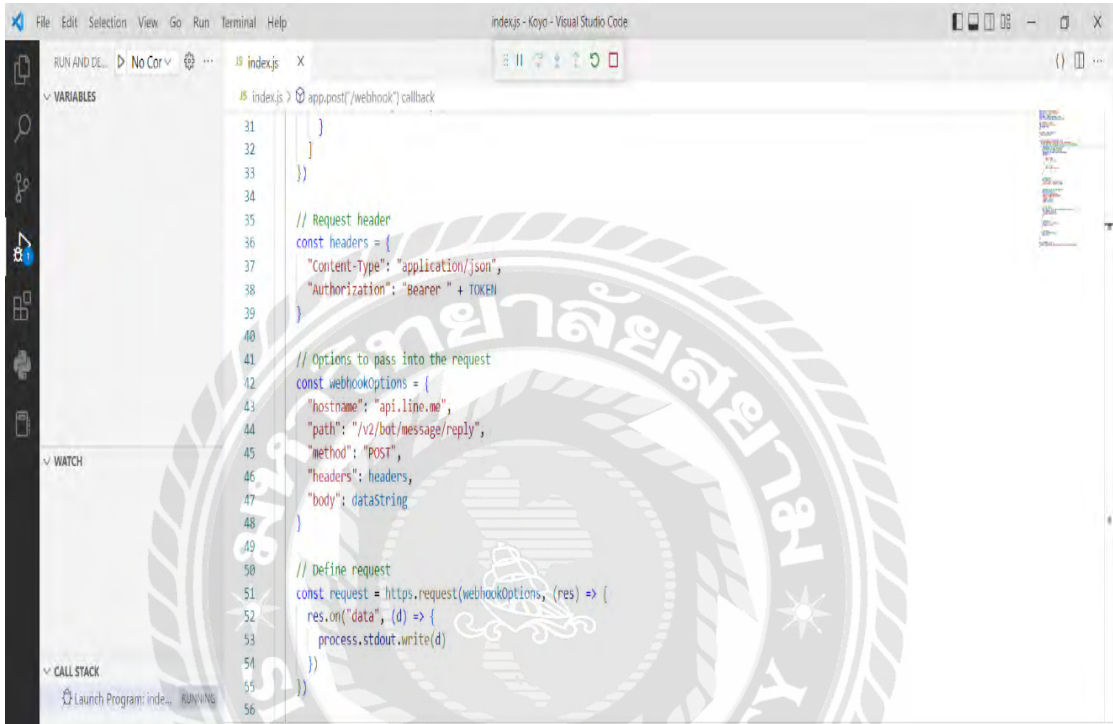
- โปรแกรม LINE
- โปรแกรม LINE Bot Designer



## บทที่ 4

### ผลการปฏิบัติโครงการ

4.1 ทำเปิดใช้งานชุดคำสั่งบน Visual Studio Code เพื่อให้ตัวชุดคำสั่งสามารถระบุได้ว่าชุดคำสั่งนี้ควรที่จะใช้งานกับ Line Channel ใดๆ



```
indexjs > app.post("/webhook") callback
31   }
32   ]
33   })
34
35   // Request header
36   const headers = {
37     "Content-Type": "application/json",
38     "Authorization": "Bearer " + TOKEN
39   }
40
41   // Options to pass into the request
42   const webhookOptions = {
43     "hostname": "api.line.me",
44     "path": "/v2/bot/message/reply",
45     "method": "POST",
46     "headers": headers,
47     "body": dataString
48   }
49
50   // Define request
51   const request = https.request(webhookOptions, (res) => {
52     res.on("data", (d) => {
53       process.stdout.write(d)
54     })
55   })
56
```

รูปที่ 4.1 ภาพแสดงการเปิดใช้งานชุดคำสั่งเพื่อไว้สำหรับเชื่อมต่อกับ Line Channel

4.2 หลังจากที่เปิดใช้งานชุดคำสั่งแล้ว โปรแกรม Visual Studio Code จะทำการสร้าง Webhook ของตัวเองค้นหาโดยใช้ IP Address ของคอมพิวเตอร์ที่ใช้งานชุดคำสั่งนั้นเป็น Webhook แต่ Webhook นี้ไม่สามารถนำไปใช้งานต่อได้ จึงจำเป็นต้องมีทางเลือกอื่นเข้ามาช่วย

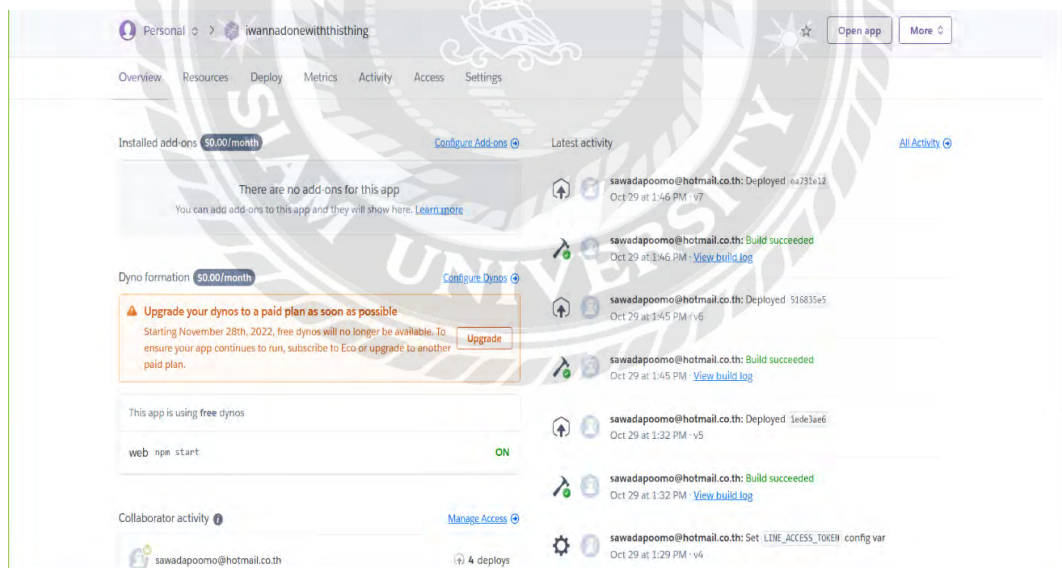
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
C:\Program Files\nodejs\node.exe .\index.js
Example app listening at http://localhost:3000

```

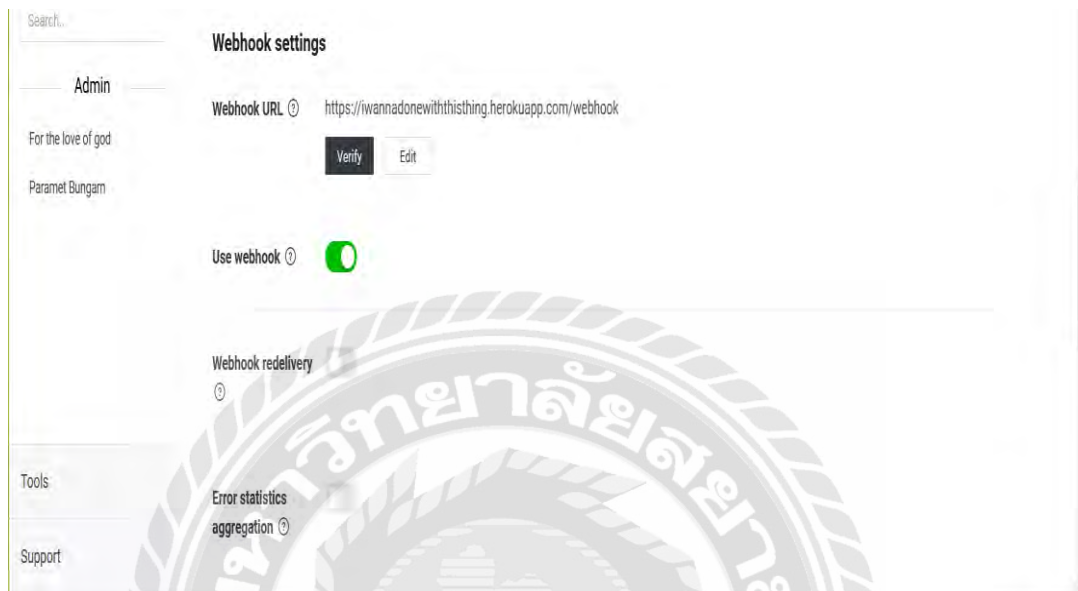
รูปที่ 4.2 ภาพแสดงผลการทำงานของชุดคำสั่งบน Visual Studio Code

4.3 ทำการเปิดหน้าต่างแสดงผลของ App บนเว็บไซต์ของ Heroku เพื่อที่จะนำ URL Webhook ที่ได้จากการสร้างและรันชุดคำสั่งขึ้นไปบน Heroku ไปใช้งานกับ Line Developer



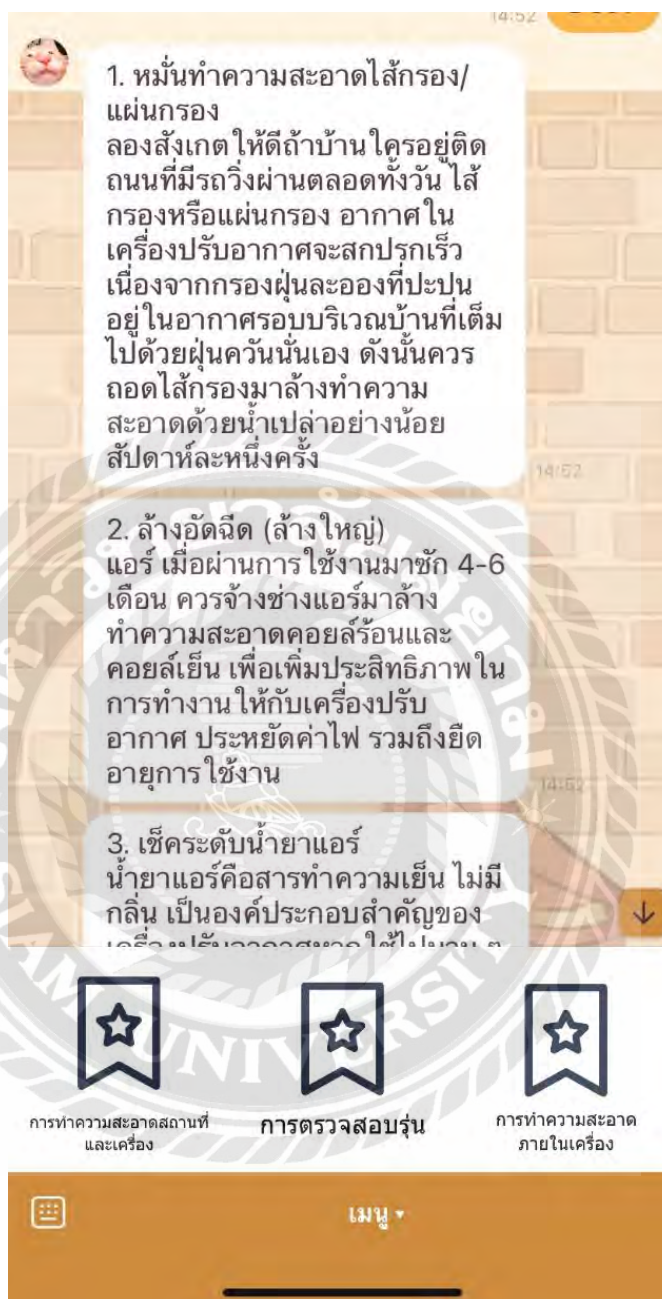
รูปที่ 4.3 หน้าต่างแสดงผลรวมของ Heroku

4.4 นำ URL Webhook ที่ได้จาก Heroku ไปใช้งานกับ Line Developers โดยการนำไปใส่ในช่องของ Webhook Setting บนหน้าเว็บไซต์เพื่อตรวจสอบเช็คการทำงานของ Webhook



รูปที่ 4.4 หน้าแถบ Webhook Setting ของ Line Developers

4.5 หลังจากที่ตรวจสอบความพร้อมทั้งหมดแล้วให้ทดสอบใช้งานตัว Messaging API เพื่อตรวจสอบว่าระบบสามารถทำงานได้ตามที่คาดหวังไว้



รูปที่ 4.5 ตัวอย่างการใช้งานเมื่อเลือกตัวเลือกที่ 1





รูปที่ 4.6 ตัวอย่างการใช้งานเมื่อเลือกตัวเลือกที่ 2



รูปที่ 4.7 ตัวอย่างการใช้งานเมื่อเลือกตัวเลือกที่ 3



## บทที่ 5

### สรุปผลรายงานและข้อเสนอแนะ

#### 5.1 สรุปผลโครงการ

จากการสร้าง Line Chat bot for searching manual สรุปผลการดำเนินงาน ได้ดังนี้

##### 5.1.1 ข้อจำกัดหรือปัญหาของโครงการ

5.1.1.1 สรุปผลการใช้งานไลน์บอทการใช้งานเป็นไปตามวัตถุประสงค์ตามที่ตั้งไว้ ไลน์บอทสามารถตอบสนองได้อย่างรวดเร็วมีความเสถียรแสดงข้อมูลครบตามต้องการ

5.1.1.2 ข้อจำกัดของไลน์บอทในการลงเนื้อหาคู่มือ จะต้องลงเป็นรูปภาพไม่สามารถลงเป็นไฟล์ได้หากลงเป็นไฟล์จะไม่สามารถเรียกใช้งานบอทได้

5.1.1.3 สรุปผลการใช้งานไลน์บอทสามารถใช้งานได้ดีส่งข้อมูลได้ตรงกับคำถาม สามารถแยกหมวดหมู่ของคู่มือได้ตามความต้องการของผู้ใช้งาน ภาพในการส่งถูกต้องและมองเห็นได้อย่างชัดเจน

5.1.1.4 ข้อจำกัดของไลน์บอทตัวนี้ผู้ที่ใช้จะต้องสแกนคิวอาร์โค้ดเท่านั้นถึงจะสามารถเรียกใช้งานบอทได้

##### 5.1.2 ข้อเสนอแนะ

5.1.2.1 ข้อเสนอแนะหากต้องการส่งเป็นวิดีโอและเสียงจะต้องมีการพัฒนาตัวบอทเพิ่ม

5.1.2.2 ข้อเสนอแนะการเรียกใช้งานคู่มือควรเรียกใช้ให้ถูกต้องเพื่อให้บอทไลน์ตอบสนองได้อย่างมีประสิทธิภาพและตรงความต้องการของผู้ใช้งาน

## 5.2 สรุปผลการปฏิบัติงานสหกิจศึกษา

จากการปฏิบัติงานสหกิจศึกษาใน บริษัท พีไฟฟ์ แมเนจเม้นท์ จำกัด ซึ่งมีผลการปฏิบัติงานสหกิจศึกษาดังต่อไปนี้

### 5.2.1 ข้อดีของการปฏิบัติงานสหกิจศึกษา

5.2.1.1 ข้อดีในการปฏิบัติงานสหกิจศึกษา ผู้จัดทำได้รับความรู้เกี่ยวกับด้านระบบ IT เกี่ยวกับอุปกรณ์ Hardware และ Software ได้มีการลงมือทำงานจริงของทางบริษัท ได้ทักษะของการซ่อมอุปกรณ์ Hardware หลายชนิด, การติดตั้ง, การเช็คอุปกรณ์, การซ่อมตัว Software รวมถึงทักษะการติดต่อสื่อสารกับทางผู้รับบริการ

### 5.2.2 ปัญหาที่พบของการปฏิบัติงานสหกิจศึกษา

5.2.2.1 ปัญหาที่พบของการปฏิบัติสหกิจศึกษา สิ่งที่ได้ศึกษามาจากห้องเรียนส่วนใหญ่หลายวิชาใช้กับการทำงานในชีวิตจริงได้บ้างไม่ได้บ้างจึงต้องศึกษาเพิ่มเติมและปรับตัวให้เข้ากับที่ทำงาน

### 5.2.3 ข้อเสนอแนะ

5.2.3.1 ข้อเสนอแนะในการปฏิบัติสหกิจศึกษา ในการไปปฏิบัติงานสหกิจศึกษา ควรเตรียมตัวไปให้พร้อม โดยสอบถามพนักงานที่ปรึกษาว่าควรศึกษาในเรื่องใดบ้างไปก่อนเพื่อจะได้ปฏิบัติหน้าที่ได้อย่างคล่องแคล่วและเป็นการเรียนรู้ระบบงานได้อย่างรวดเร็ว เนื่องจากช่วงโรคระบาดโควิด-19 งานเพิ่มขึ้นเป็นจำนวนมากทำให้พนักงานที่ปรึกษาอาจไม่มีเวลาดูแลได้เต็มที่

## บรรณานุกรม

บริษัท วินัส ซัพพลาย จำกัด. (2562). การสร้าง LINE Bot โดยใช้ Dialogflow. เข้าถึงได้จาก

<https://www.venussolutions.com/การสร้าง-line-bot-โดยใช้-dialogflow-ตอนที่-1>

CHATCOMPOSE. (2563). วิธีสร้างแชทบอทสำหรับ LINE. เข้าถึงได้จาก

<https://www.chatcompose.com/th/intro.html>

Line Official Manager. (2554). Line Official. เข้าถึงได้จาก

<https://account.line.biz/login?redirectUri=https%3A%2F%2Fmanager.line.biz%2Faccount%2F%40123azmrt>

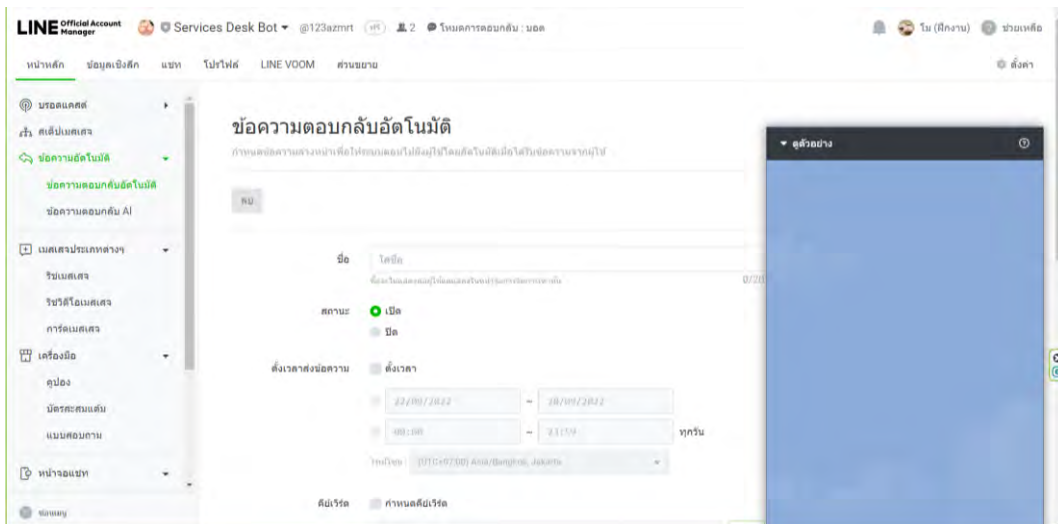
Nattapon Sirikamonnet. (2561). สร้าง LINE BOT กันเถอะ (เริ่มต้น + reply message). เข้าถึงได้จาก

<https://medium.com/@nattaponsirikamonnet/สร้าง-bot-ด้วย-line-messaging-api-d7de644ac892>

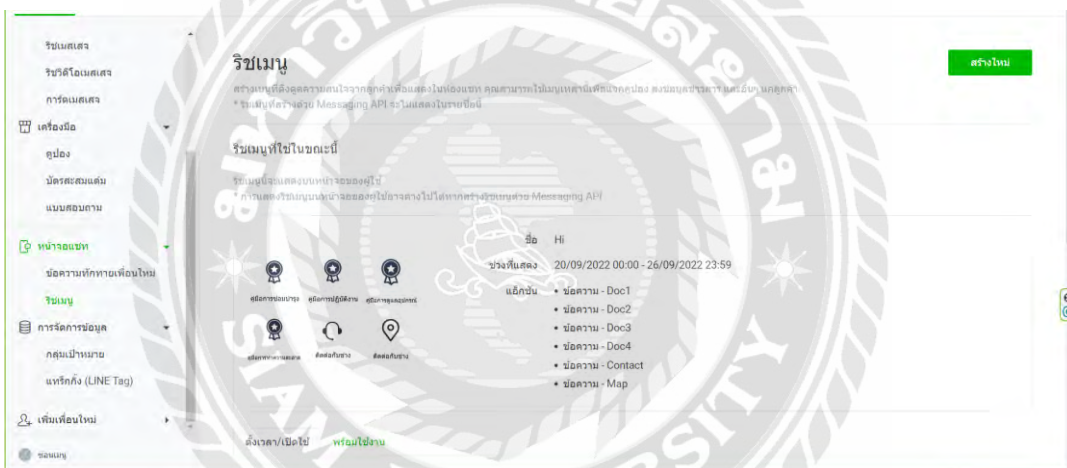




ภาคผนวก ก  
ภาพของการสร้างบ่อท



รูปที่ ก.1 รูประหว่างการสร้างบอท



รูปที่ ก.2 รูประหว่างการสร้างบอท

```

1 from flask import Flask, request, abort
2 import requests
3 import json
4 app = Flask(__name__)
5 @app.route('/', methods=['POST', 'GET'])
6 def webhook():
7     if request.method == 'POST':
8         payload = request.json
9         reply_token = payload['events'][0]['replyToken']
10        print(reply_token)
11        message = payload['events'][0]['message']['text']
12        print(message)
13        if 'Doc4' in message:
14            reply_message = '1. พบหน้าจาวามเมสเสจได้ครบ/ไม่ครบ?'
15            reply_message(reply_token, reply_message, 'Aapsr11t6zWtTzE@PLUX90mTTha55fEz04ClaxefZvqoRjYLda+T9FH191E8wqJqAD01E2hwApKcQd11V74z')
16            return request.json, 200
17        else:
18            abort(400)
19
20 def ReplyMessage(reply_token, TextMessage, Line_Acees_Token):
21     LINE_API = 'https://api.line.me/v2/bot/message/reply'
22     Authorization = 'Bearer {}'.format(Line_Acees_Token)
23     print(Authorization)
24     headers = {
25         'Content-Type': 'application/json; charset=UTF-8',
26         'Authorization': Authorization

```

Windows PowerShell  
Copyright (c) Microsoft Corporation. All rights reserved.  
Try the new cross-platform PowerShell <https://aka.ms/pscore6>  
PS C:\Users\lasus\pboom> Sapling

รูปที่ ก.3 รูปการณ์พัฒนาตัวโค้ดของ Line Bot

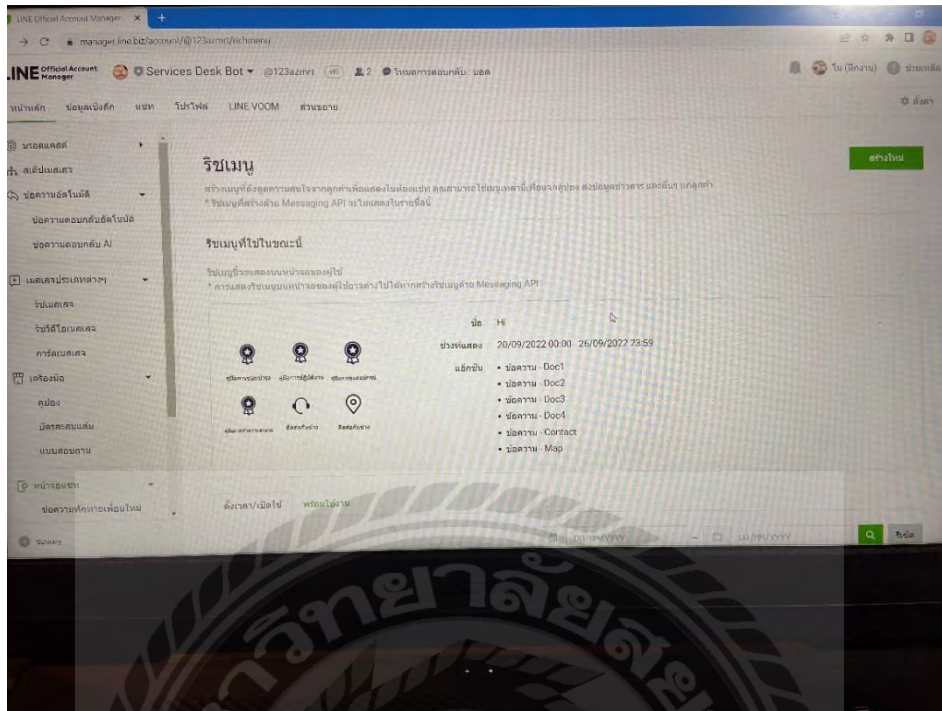


ภาคผนวก ข

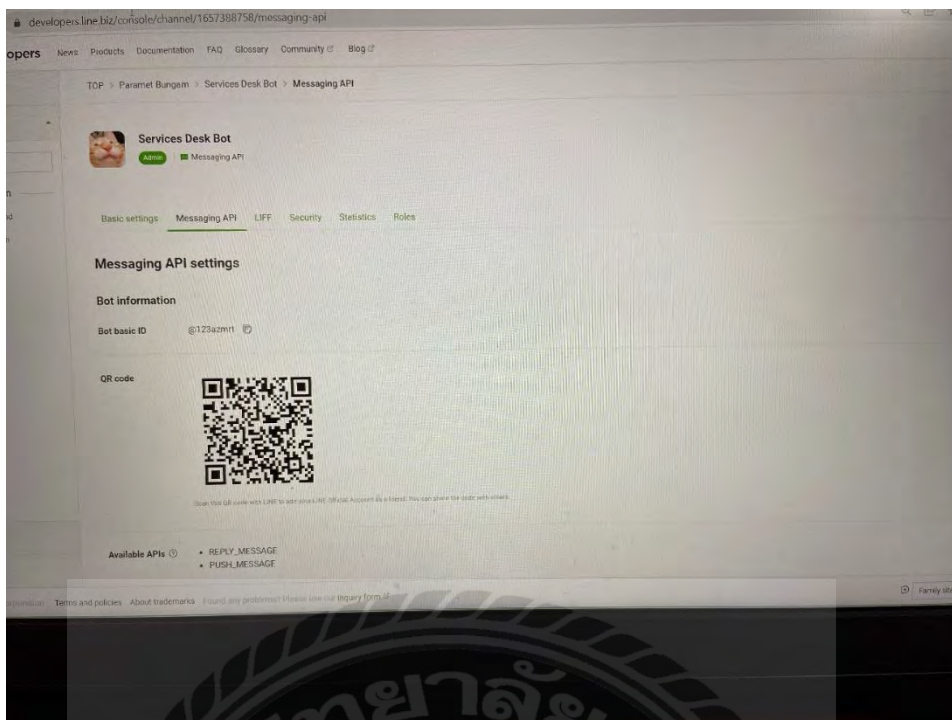
ภาพขณะปฏิบัติงาน



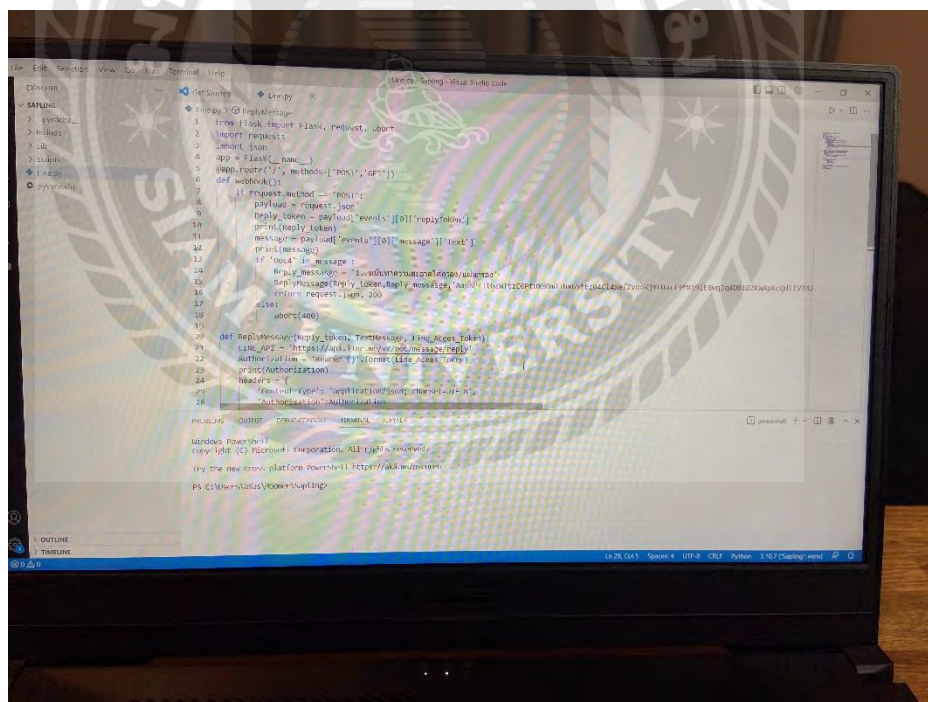




รูปที่ ข.1 ภาพขณะกำลังใช้งานเว็บไซต์ LINE Official Manager



รูปที่ ข.2 ภาพขณะกำลังใช้งานเว็บไซต์ LINE Developer



รูปที่ ข.3 ภาพขณะกำลังพัฒนาโค้ดในโปรแกรม Visual Studio Code

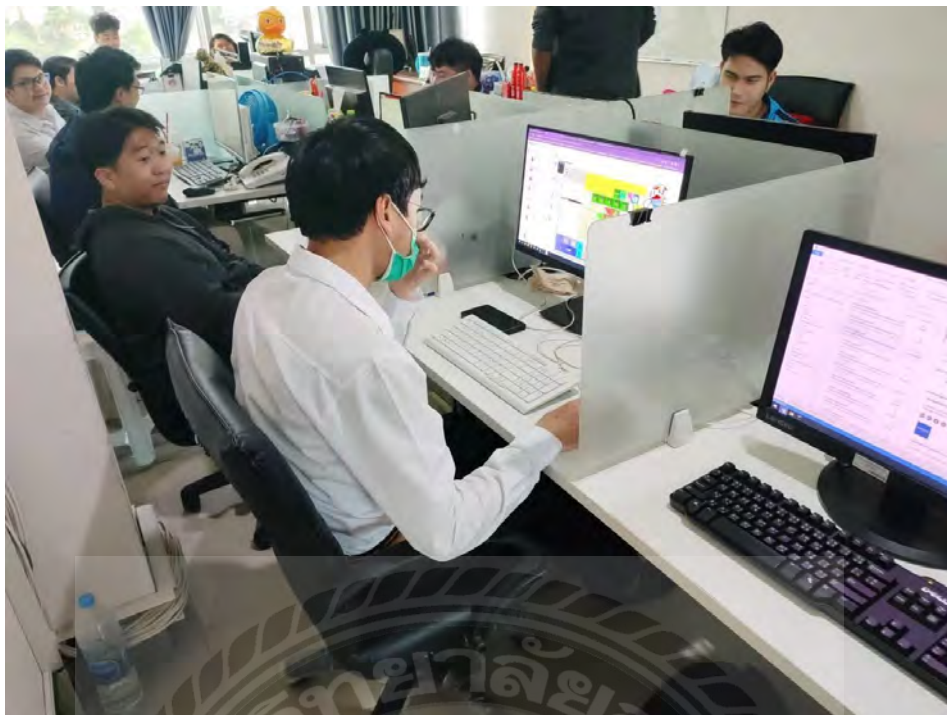




รูปที่ ข.4 ภาพขณะกำลังปฏิบัติงานที่สถานที่ฝึกงาน



รูปที่ ข.5 ภาพขณะกำลังปฏิบัติงานที่สถานที่ฝึกงาน



รูปที่ ข.6 ภาพขณะกำลังปฏิบัติงานที่สถานที่ฝึกงาน



## ประวัติผู้จัดทำ



ชื่อ-สกุล : นาย ปารามศ บัวงาม  
รหัสนักศึกษา : 6204000009  
ภาควิชา : วิศวกรรมคอมพิวเตอร์  
คณะ : วิศวกรรมศาสตร์  
ที่อยู่ปัจจุบัน : 299 ถนนเพชรเกษม แขวงปากคลอง  
ภาษีเจริญ  
เขตภาษีเจริญ กรุงเทพมหานคร 10160



ชื่อ-สกุล : นาย สรวิชญ์ สุขผล  
รหัสนักศึกษา : 6204000012  
ภาควิชา : วิศวกรรมคอมพิวเตอร์  
คณะ : วิศวกรรมศาสตร์  
ที่อยู่ปัจจุบัน : 172/5 หมู่ 5 ตำบลอ้อมน้อย  
อำเภอกระทุ่มแบน  
จังหวัดสมุทรสาคร 74130