

การพัฒนาคุณลักษณะการสนทนาในแอปพลิเคชัน
มือถือสำหรับผู้ให้บริการ (Consumer) ของ
แพลตฟอร์มคลาวด์คิทเชน

Chat features for consumer mobile application
development of Cloud Kitchen platform



สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2565

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CHAT FEATURES FOR CONSUMER MOBILE
APPLICATION DEVELOPMENT OF
CLOUD KITCHEN PLATFORM



A COOPERATIVE EDUCATION SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENT FOR
THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE)
DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2022

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อสหกิจศึกษา การพัฒนาคุณลักษณะการสนทนาในแอปพลิเคชันมือถือ
สำหรับผู้ใช้บริการ (Consumer) ของแพลตฟอร์มครัวคลาวด์คิทเช่น
CHAT FEATURES FOR CONSUMER MOBILE
APPLICATION DEVELOPMENT OF CLOUD KITCHEN
PLATFORM

ชื่อนักศึกษา นายธนวิชญ์ อยู่ทุก รหัสนักศึกษา 62050173



ปริญญา วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)

ภาควิชา วิทยาการคอมพิวเตอร์

ปีการศึกษา 2565

อาจารย์ที่ปรึกษา ผศ.ดร.อินทราพร อรัณยนาค

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้
สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วิทยาการ
คอมพิวเตอร์) ประจำปีการศึกษา 2565

คณะกรรมการสอบ	ลายมือชื่อ
ผศ.ดร.อินทราพร อรัณยนาค กรรมการและอาจารย์ที่ปรึกษา	
ผศ.ดร.ปีตมา เจริญพร กรรมการ	

ลิขสิทธิ์ของคณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อสหกิจศึกษา	การพัฒนาคุณลักษณะการสนทนาในแอปพลิเคชันมือถือ สำหรับผู้ให้บริการ (Consumer) ของแพลตฟอร์มคลาวด์คิทเช่น CHAT FEATURES FOR CONSUMER MOBILE APPLICATION DEVELOPMENT OF CLOUD KITCHEN PLATFORM
ชื่อนักศึกษา	นายธนวิษณุ อยู่พุก รหัสนักศึกษา 62050173
ปริญญา	วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา	วิทยาการคอมพิวเตอร์
ปีการศึกษา	2565
อาจารย์ที่ปรึกษา	ผศ.ดร.อินทราพร อรัณยະนาค

บทคัดย่อ

โครงการสหกิจศึกษานี้ จัดทำขึ้นเพื่อวัตถุประสงค์ในการร่วมพัฒนาแอปพลิเคชันของแพลตฟอร์มคลาวด์คิทเช่น (Cloud Kitchen Platform) ซึ่งเป็นแพลตฟอร์มสำหรับจัดการการสั่งซื้อ การขาย และการจัดส่งอาหาร โดยมีเป้าหมายเพื่ออำนวยความสะดวกให้แก่ผู้ให้บริการ ที่มีความต้องการสั่งอาหารโดยไม่สะดวกที่จะไปทานที่ร้านหรือต้องการประหยัดเวลาที่ใช้ในการเดินทาง อีกทั้งยังอำนวยความสะดวกให้แก่ผู้ประกอบการร้านค้าที่ต้องการขายอาหารบนแพลตฟอร์มออนไลน์ ให้มีทางเลือกในการขายอาหารได้มากยิ่งขึ้น จากการร่วมพัฒนา ผู้จัดทำได้ทำการศึกษา พัฒนา และ ทดสอบ คุณลักษณะการสนทนาบนแอปพลิเคชันสำหรับผู้ให้บริการ (Chat Feature for Consumer Application) ซึ่งทำงานอยู่บนระบบปฏิบัติการแอนดรอยด์ (Android Operating System) โดยเป็นส่วนหนึ่งของระบบการสนทนา (Chat System) บนแพลตฟอร์มคลาวด์คิทเช่น (Cloud Kitchen Platform) เพื่อเพิ่มช่องทางในการติดต่อสื่อสารระหว่างผู้ให้บริการกับผู้ประกอบการร้านค้า ผ่านข้อความตัวอักษรและรูปภาพ ซึ่งจะช่วยให้แพลตฟอร์มคลาวด์คิทเช่นและผู้ประกอบการร้านค้า สามารถยกระดับการบริการให้ดีและมีความเป็นมืออาชีพมากยิ่งขึ้น

คำสำคัญ : แพลตฟอร์มคลาวด์คิทเช่น, คุณลักษณะการสนทนาบนแอปพลิเคชันสำหรับผู้ให้บริการ, ระบบปฏิบัติการแอนดรอยด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	CHAT FEATURES FOR CONSUMER MOBILE APPLICATION DEVELOPMENT OF CLOUD KITCHEN PLATFORM
Students	Mr. Thanawich Yoopuk Student ID 62050173
Degree	Bachelor of Science (Computer Science)
Department	Computer Science
School	Science
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)
Academic Year	2022
Advisor	Asst.Prof.Inthraporn Aranyanak, Ph.D.

Abstract

This cooperative education project Prepared for the purpose of co-developing an application of the Cloud Kitchen Platform, which is a platform for managing orders, sales and food delivery. with the goal of facilitating service users who want to order food without being inconvenient to eat at the restaurant or want to save time spent traveling It also facilitates shop operators who want to sell food on online platforms. To have more options to sell food. from co-development the organizer has studied, developed and tested the Chat Feature for Consumer Application which runs on Android Operating system. It is part of the Chat System on the Cloud Kitchen Platform to increase communication channels between service users and shop operators. through text messages and images This will enable the cloud kitchen platform and merchant operators to enhance their services to be better and more professional.

Keywords : Cloud Kitchen Platform, Chat Feature for Consumer Application, Chat System, Android Operating Syste

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำโครงการสหกิจศึกษาในหัวข้อเรื่อง การพัฒนาคุณลักษณะการสนทนาในแอปพลิเคชันมือถือสำหรับผู้ให้บริการ (Consumer) ของ แพลตฟอร์มคลาวด์คิดเช่น สามารถดำเนินการจนประสบความสำเร็จลุล่วงไปได้ด้วยดี เนื่องจากได้รับความกรุณา ความอนุเคราะห์ และการสนับสนุนจากบุคคลหลายท่าน ผู้จัดทำขอขอบคุณบุคคลต่างๆ ที่ได้ให้ความอนุเคราะห์ และกรุณาเสียสละเวลาในการให้คำแนะนำปรึกษาโดยตลอด อันได้แก่

ครอบครัวของผู้จัดทำอันได้แก่ บิดา มารดา ที่คอยให้การสนับสนุนในด้านการศึกษา รวมไปถึงปัจจัยด้านอื่น ๆ ทั้งทางด้านกายภาพและทางด้านจิตใจ ส่งผลให้การทำงานสามารถผ่านลุล่วงไปได้ด้วยดี

ผศ.ดร.อินทราพร อรัณยธนา และ ผศ.ดร.ปัทมา เจริญพร ที่ได้กรุณาเสียเวลาให้คำแนะนำ คำปรึกษา ข้อเสนอแนะ และวิธีแก้ไขปัญหาต่าง ๆ ที่เกิดขึ้นในตลอดระยะเวลาการจัดทำโครงการสหกิจศึกษานี้ อีกทั้งยังคอยติดตามความคืบหน้าของการดำเนินงาน และช่วยตรวจสอบแก้ไขให้โครงการสหกิจศึกษานี้เพื่อให้บรรลุผลสำเร็จไปได้ด้วยดี

บริษัทและพี่เลี้ยงทุกท่าน ที่ได้มอบโอกาสและประสบการณ์อันดีตลอดการเป็นนักศึกษาฝึกงานในตำแหน่งผู้พัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ รวมทั้งให้ความรู้และประสบการณ์เกี่ยวกับการทำงานภายในองค์กร และความรู้ทางด้านวิทยาการคอมพิวเตอร์ ที่สามารถนำมาใช้งานได้จริงในการจัดทำโครงการสหกิจศึกษานี้ ทำให้โครงการสหกิจศึกษานี้สามารถสำเร็จได้อย่างราบรื่น

คณะอาจารย์ภาควิชาวิทยาการคอมพิวเตอร์ และภาควิชาอื่น ๆ ที่ได้อบรมและให้ความรู้ความสามารถ และแบ่งปันประสบการณ์ในการทำงานและการใช้ชีวิตต่าง ๆ ตลอดระยะเวลาสี่ปีที่ได้ศึกษา ณ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

บุคคลท่านอื่นๆ ที่ไม่ได้กล่าวถึงไว้ ณ ที่นี้ ขอขอบคุณทุกท่าน ที่ได้สละเวลามาช่วยเหลือให้คำปรึกษาและอนุเคราะห์ และเป็นกำลังใจ ในการจัดทำโครงการสหกิจศึกษานี้ ให้สำเร็จไปได้ด้วยดี

สุดท้ายนี้ ผู้จัดทำหวังว่าโครงการสหกิจศึกษานี้จะมีประโยชน์อยู่ไม่น้อยแก่หน่วยงานที่เกี่ยวข้อง และผู้ที่ต้องการศึกษาต่อไป

ธนวิษณุ อยู่พุก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ช
สารบัญรูป.....	ซ
คำย่อ/สัญลักษณ์ (ถ้ามี).....	ญ
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 วัตถุประสงค์.....	1
1.3 ขอบเขต.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	3
2.1 ทฤษฎีที่เกี่ยวข้อง.....	3
2.1.1 Agile Software Development.....	3
2.1.2 แนวทางการพัฒนาโปรแกรมในสถาปัตยกรรมรูปแบบ MVVM.....	4
2.1.2 รูปแบบโครงสร้างของโปรเจคแบบ Architecture Components.....	5
2.1.3 Android Activity Lifecycle.....	5
2.1.4 การทดสอบระบบสำหรับแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์.....	6
2.2 เทคโนโลยีที่เกี่ยวข้อง.....	7
2.2.1 ระบบปฏิบัติการแอนดรอยด์ (Android operating system).....	7
2.2.2 Android Studio.....	9
2.2.3 Kotlin for Android.....	9
2.2.4 Figma.....	10
2.2.5 Sourcetrue.....	10
2.2.6 Postman.....	10
2.2.7 การแลกเปลี่ยนข้อมูลผ่าน RESTful Web Service.....	10
2.2.8 การแลกเปลี่ยนข้อมูลผ่าน Socket.IO.....	11
2.2.9 Koin Injection Framework.....	11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มาไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 3 วิธีการดำเนินงานวิจัย	12
3.1 การศึกษาข้อมูลและวิเคราะห์ระบบงานเดิมของแพลตฟอร์มคลาวด์คิทเช่น	12
3.1.1 การศึกษากระบวนการทำงานของบริษัท	12
3.1.2 การศึกษาการเขียนโปรแกรมด้วยภาษา Kotlin บนระบบปฏิบัติการ Android และการใช้งานโปรแกรม Android Studio	12
3.1.3 การศึกษาโครงสร้างและสถาปัตยกรรมซอฟต์แวร์ของแอปพลิเคชัน สำหรับผู้ใช้งาน.....	13
3.2 การศึกษาและวิเคราะห์ความต้องการของผู้ใช้และความต้องการของระบบ	16
3.3 การออกแบบระบบ	17
3.3.1 แผนภาพการแสดงผลการทำงานของผู้ใช้ระบบ (Use Case Diagram) ..	17
3.3.2 แผนผังการทำงานแบบลำดับปฏิสัมพันธ์ (Sequence Diagram)	21
บทที่ 4 ผลการวิจัยและการอภิปรายผล	29
4.1 ส่วนต่อประสานผู้ใช้ (User Interface).....	29
4.1.1 การแสดงประวัติการสนทนา	29
4.1.2 การส่งข้อความตัวอักษร.....	32
4.1.3 การรับข้อความตัวอักษร	34
4.1.4 การส่งข้อความรูปภาพ.....	35
4.1.5 การรับข้อความรูปภาพ	37
4.1.6 การปรับวิธูรูปแสดงเต็มหน้าจอ.....	38
4.1.7 การดูสถานะการอ่าน	39
4.1.8 การแจ้งเตือนข้อความใหม่	40
4.2 การทดสอบระบบซอฟต์แวร์ (Software Testing).....	42
4.2.1 การทดสอบคลาส ViewModel	42
4.2.2 การทดสอบคลาส Adapter	43
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	46
5.1 สรุปผลการวิจัย.....	46
5.2 ข้อเสนอแนะ	46
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า เอกสารอ้างอิง	48
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้	

สารบัญ (ต่อ)

	หน้า
ภาคผนวก.....	50
ภาคผนวก ก.....	51
ภาคผนวก ข.....	52



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 ชื่อและเวอร์ชันของระบบปฏิบัติการแอนดรอยด์	8
3.1 คำอธิบาย Use Case Description ของดูประวัติการสนทนา	18
3.2 คำอธิบาย Use Case Description ของส่งข้อความตัวอักษร.....	18
3.3 คำอธิบาย Use Case Description ของส่งข้อความรูปภาพ.....	19
3.4 คำอธิบาย Use Case Description ของฟรีวีวรูปภาพ	19
3.5 คำอธิบาย Use Case Description ของดูสถานะการอ่าน	19
3.6 คำอธิบาย Use Case Description ของแจ้งเตือนข้อความ.....	20
3.7 คำอธิบาย Use Case ของส่งข้อความใหม่อีกครั้ง.....	20
4.1 กรณีการทดสอบ (Test Case) ของคลาส ViewModel	42
4.1 กรณีการทดสอบ (Test Case) ของคลาส Adapter.....	43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 แผนภาพแสดงตัวอย่างการทำงานโดยใช้แนวคิด Agile Software Development	3
2.2 ภาพตัวอย่างการทำงานของ Android Activity Lifecycle	6
2.3 แผนภาพพีระมิดของการทดสอบระบบ	6
2.4 การกระจายตัวการใช้งานในแต่ละเวอร์ชันของระบบปฏิบัติการแอนดรอยด์	8
2.5 ตัวอย่างส่วนต่อประสานผู้ใช้และอุปกรณ์จำลองของโปรแกรม Android Studio	9
3.1 แผนภาพโครงสร้างของแอปพลิเคชันสำหรับผู้ใช้งาน	14
3.2 แผนภาพแสดงการทำงานของผู้ใช้ระบบสารสนเทศ	17
3.3 แผนภาพ Sequence Diagram ของการดูประวัติการสนทนา	21
3.4 แผนภาพ Sequence Diagram ของการส่งข้อความตัวอักษร	22
3.5 แผนภาพ Sequence Diagram ของการส่งข้อความรูปภาพ	23
3.6 แผนภาพ Sequence Diagram ของการส่งข้อความใหม่อีกครั้ง	24
3.7 แผนภาพ Sequence Diagram ของการส่งข้อความรูปภาพใหม่อีกครั้ง	25
3.8 แผนภาพ Sequence Diagram ของการระบุสถานะการอ่าน	26
3.9 แผนภาพ Sequence Diagram ของการแจ้งเตือนข้อความใหม่	27
3.10 แผนภาพ Sequence Diagram ของการแสดงปุ่มเข้าสู่ห้องสนทนาพร้อมด้วยจุดสีแดง....	28
4.1 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนา กรณีคำสั่งซื้ออยู่ภายในระยะเวลา 24 ชั่วโมงและยังไม่มีประวัติการสนทนา	29
4.2 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนา กรณีคำสั่งซื้ออยู่ภายในระยะเวลา 24 ชั่วโมงและมีประวัติการสนทนา	30
4.3 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนา กรณีคำสั่งซื้ออยู่เกินระยะเวลา 24 ชั่วโมง .	31
4.4 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการส่งข้อความตัวอักษร.....	32
4.5 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการส่งข้อความตัวอักษรสำเร็จ	32
4.6 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการส่งข้อความตัวอักษรไม่สำเร็จ	33
4.7 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการรับข้อความตัวอักษร	34
4.8 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการส่งข้อความตัวอักษรสำเร็จ	35
4.9 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการส่งข้อความรูปภาพไม่สำเร็จ	36
4.10 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการรับข้อความรูปภาพ	37
4.11 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการพรีวิวแสดงรูปภาพเต็มหน้าจอ	38
4.12 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อผู้ประกอบการร้านค้าได้อ่าน ข้อความของผู้ใช้บริการแล้ว	39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
4.13 ตัวอย่างส่วนต่อประสานผู้ใช้ของระบบเมื่อมีการแจ้งเตือนข้อความใหม่.....	40
4.14 ตัวอย่างส่วนต่อประสานผู้ใช้ของระบบเมื่อมีจัดกลุ่มการแจ้งเตือน	40
4.15 ตัวอย่างส่วนต่อประสานผู้ใช้ของหน้าต่างติดตามสถานะคำสั่งซื้อเมื่อมีการแจ้งเตือน ข้อความใหม่	41



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากบริษัทแห่งหนึ่งได้มีการจัดทำแพลตฟอร์มคลาวด์คิทเชน (Cloud Kitchen Platform) ซึ่งเป็นกลุ่มของแอปพลิเคชันสำหรับการสั่งอาหาร เพื่ออำนวยความสะดวกให้แก่ผู้ใช้บริการ ที่มีความต้องการสั่งอาหารโดยไม่สะดวกที่จะไปทานที่ร้านหรือต้องการประหยัดเวลาที่ใช้ในการเดินทาง อีกทั้งยังอำนวยความสะดวกให้แก่ผู้ประกอบการร้านค้าที่ต้องการขายอาหารบนแพลตฟอร์มออนไลน์ ให้มีทางเลือกในการขายอาหารได้มากยิ่งขึ้น รวมถึงมีการจัดหาบริการจัดส่งอาหารสำหรับผู้ประกอบการร้านค้าที่ไม่มีผู้จัดส่งด้วยตัวเอง โดยแอปพลิเคชันทั้งหมดจะมีด้วยกัน 3 แอปพลิเคชัน ได้แก่ 1) แอปพลิเคชันสำหรับผู้ใช้บริการ (Consumer) ซึ่งมีไว้สำหรับผู้ใช้บริการที่ต้องการสั่งอาหารตามร้านอาหารต่างๆ 2) แอปพลิเคชันสำหรับผู้ประกอบการร้านค้า (Merchant) ซึ่งสร้างขึ้นเพื่อให้ผู้ประกอบการร้านค้าที่ใช้บริการ สามารถสร้างหน้าร้าน เมนูต่างๆ ให้ผู้ใช้บริการได้เลือกซื้อ รวมถึงจัดหาบริการจัดส่งให้แก่ผู้ประกอบการร้านค้า 3) แอปพลิเคชันสำหรับผู้จัดส่ง (Raider) มีไว้สำหรับให้ผู้จัดส่งอาหารรับงานจัดส่งอาหาร ติดต่อกับผู้ประกอบการร้านค้า และแสดงเส้นทางในการจัดส่งอาหารให้แก่ลูกค้า

อย่างไรก็ตาม Cloud Kitchen Platform นี้ยังขาดในส่วนของคุณลักษณะการสนทนาภายในแอปพลิเคชันสำหรับผู้ใช้บริการ (Consumer) และแอปพลิเคชันสำหรับผู้ประกอบการร้านค้า (Merchant) เพื่อเพิ่มช่องทางที่ใช้ในการติดต่อสื่อสารเพื่อแจ้งรายละเอียดเกี่ยวกับรายการเมนูอาหารที่ทางผู้ใช้บริการได้ทำการสั่งซื้อกับทางผู้ประกอบการร้านค้า

ดังนั้นผู้จัดทำจึงเล็งเห็นความสำคัญที่จะต้องมีการพัฒนาคุณลักษณะการสนทนาภายในแอปพลิเคชันสำหรับผู้ใช้บริการ (Consumer) เพื่อให้ผู้ใช้บริการสามารถติดต่อสื่อสารและแจ้งรายละเอียดเกี่ยวกับรายการเมนูอาหารที่ทางผู้ใช้บริการได้ทำการสั่งซื้อกับทางผู้ประกอบการร้านค้าได้ง่ายมากยิ่งขึ้น ซึ่งจะทำให้แพลตฟอร์มคลาวด์คิทเชนและผู้ประกอบการร้านค้าสามารถยกระดับการบริการให้ดีขึ้นและมีความเป็นมืออาชีพมากยิ่งขึ้น

1.2 วัตถุประสงค์ของงานวิจัย

1) เพื่อพัฒนาคุณลักษณะในการสนทนา (Chat) ในแอปพลิเคชันมือถือสำหรับผู้ใช้บริการ (Consumer)

2) เพื่อเพิ่มช่องทางติดต่อสื่อสารระหว่างผู้ใช้บริการ (Consumer) กับผู้ประกอบการร้านค้า (Merchant) ให้สะดวกมากยิ่งขึ้น

ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้นที่ ไม่มีเหตุแห่งข้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 3) เพื่อเพิ่มความถูกต้องของรายละเอียดต่างๆ ในรายการอาหารที่ผู้ให้บริการ (Consumer) ได้สั่งกับทางผู้ประกอบการร้านค้า (Merchant)
- 4) เพื่อทดสอบการทำงานของคุณลักษณะในการสนทนา (Chat) ในแอปพลิเคชันมือถือสำหรับผู้ให้บริการ

1.3 ขอบเขตของงานวิจัย

- 1) แอปพลิเคชันมือถือจะทำงานบนระบบปฏิบัติการแอนดรอยด์ (Android) เท่านั้น
- 2) แอปพลิเคชันมือถือทำงานผ่านเครือข่ายอินเทอร์เน็ต
- 3) แอปพลิเคชันมือถือต้องมีการขอใช้ตำแหน่งที่ตั้งของผู้ให้บริการ
- 4) เป็นแอปพลิเคชันสำหรับผู้ให้บริการ (Consumer) เท่านั้น

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ได้รับคุณลักษณะในการสนทนาสำหรับแพลตฟอร์มคลาวด์คิดเช่น
- 2) ผู้ใช้งาน (Consumer) กับผู้ประกอบการร้านค้า (Merchant) สามารถติดต่อสื่อสารกันได้สะดวกมากยิ่งขึ้น
- 3) เพิ่มความถูกต้องของรายละเอียดต่างๆ ในรายการอาหารที่ผู้ใช้งาน (Consumer) ได้สั่งกับทางผู้ประกอบการร้านค้า (Merchant)
- 4) การทำงานของคุณลักษณะการสนทนาสามารถทำงานได้อย่างไม่มีข้อผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

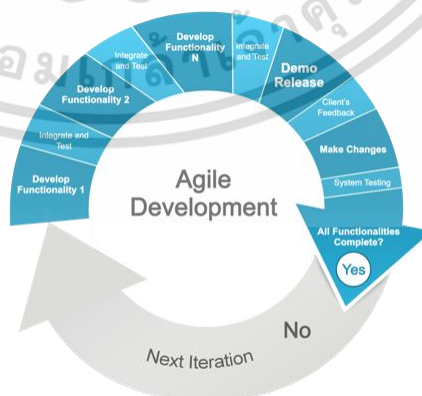
การจัดทำรูปแบบสหกิจศึกษานี้ เป็นการพัฒนาคุณลักษณะสำหรับการสนทนาในแอปพลิเคชันมือถือสำหรับผู้ให้บริการ (Consumer) บนแพลตฟอร์มคลาวด์คิพ เช่น จึงจำเป็นต้องมีการอาศัยความรู้และความเข้าใจเกี่ยวกับการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ เพื่อใช้ในกระบวนการพัฒนา โดยได้มีการศึกษาจากเอกสาร งานวิจัย และบทความต่างๆ ในหัวข้อที่เกี่ยวข้องกับการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ โดยมีหัวข้อและรายละเอียดดังต่อไปนี้

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 Agile Software Development

Agile เป็นแนวทางและวิธีการพัฒนาซอฟต์แวร์ที่กำลังได้รับความนิยมในปัจจุบัน เป็นการพัฒนาแนวคิดในการทำงาน โดยให้ความสำคัญกับเรื่องของ บุคคล การสื่อสาร และแนวทางที่จะนำไปใช้ในการพัฒนาซอฟต์แวร์

วิธีการแบบ Agile เน้นไปยังการปรับปรุงซอฟต์แวร์อย่างรวดเร็ว หรือเน้นการพัฒนาเพื่อรองรับการเปลี่ยนแปลงในอนาคต โดยได้ทำการแบ่งงานให้เป็นงานเล็กๆ และมีการวางแผนในช่วงสั้นๆ โดยได้แบ่งการทำงานออกเป็นช่วงสั้นๆ ในแต่ละช่วงการทำงานจะใช้รูปแบบวงจรการพัฒนาซอฟต์แวร์เต็มรูปแบบ ซึ่งรวมถึงการวางแผน การเก็บข้อมูลความต้องการ การออกแบบซอฟต์แวร์ การพัฒนา และการทดสอบซอฟต์แวร์ และในเมื่อหมดช่วงการทำงานนั้นๆ ทีมผู้พัฒนาจะแสดงผลงานที่พัฒนาในช่วงสั้นๆ ให้ลูกค้าได้รับทราบ



รูปที่ 2.1 แผนภาพแสดงตัวอย่างการทำงานโดยใช้แนวคิด Agile Software Development เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ที่มา : จาก <https://reg7.pwa.co.th/kmr7/?p=453> ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทั่วไปแล้ว Agile ไม่มีการแบ่งงานและกำหนดเวลาที่ตายตัว แต่จะขึ้นอยู่กับแต่ละองค์กรที่นำมาปรับใช้ แต่ต้องอยู่ในหลักการและแนวทาง 4 ข้อสำคัญดังต่อไปนี้

1) เน้นการสื่อสารและการปรับเปลี่ยน มากกว่าการยึดถือในแนวทางและวิธีการ เช่น การให้โปรแกรมเมอร์สามารถพูดคุยกับลูกค้าได้โดยตรงโดยไม่ต้องผ่านเจ้าของโปรเจค (Project Owner) เพื่อความเข้าใจที่มากขึ้น

2) เน้นที่ผลงานที่ทำสำเร็จมากกว่ารายงานและเอกสารต่างๆ ในที่นี้หมายถึง การสร้างซอฟต์แวร์ที่ใช้งานจริงให้ได้ดีที่สุด

3) ให้ความสำคัญกับการติดต่อสื่อสารระหว่างผู้พัฒนาและลูกค้า เพื่อเก็บความต้องการและข้อชี้แนะจากลูกค้าและผู้ใช้งาน เพื่อนำมาพัฒนาซอฟต์แวร์ให้เป็นไปในแนวทางเดียวกันกับความต้องการ

4) ให้ความสำคัญกับการปรับเปลี่ยนและยืดหยุ่นแผนงาน พร้อมรับการปรับปรุงและแก้ไขตลอดเวลา เพื่อให้ขั้นตอนการพัฒนาซอฟต์แวร์เป็นไปได้อย่างราบรื่น

2.1.2 แนวทางการพัฒนาโปรแกรมในสถาปัตยกรรมรูปแบบ MVVM

เป็นรูปแบบของสถาปัตยกรรมซอฟต์แวร์ที่ได้รับการยอมรับในอุตสาหกรรมซอฟต์แวร์ในปัจจุบัน โดยเป็นรูปแบบของสถาปัตยกรรมที่สามารถทดแทนจุดด้อยของสถาปัตยกรรมแบบ MVC และ MVP ได้ทั้งหมด โดยสถาปัตยกรรมแบบ MVVM แนะนำให้มีการแยกตรรกะการนำเสนอข้อมูล (View or UI Logic) ออกจากตรรกะทางธุรกิจของแอปพลิเคชัน (Business Logic) โดยสถาปัตยกรรมแบบ MVVM จะแบ่งลำดับชั้นการพัฒนาออกเป็น 3 ชั้น ได้แก่

2.1.1.1 Model

Model เป็นลำดับชั้นที่มีหน้าที่รับผิดชอบในการจัดการข้อมูลเป็นหลัก โดย Model จะทำงานร่วมกับ ViewModel เพื่อทำการรับและบันทึกข้อมูล

2.1.1.2 View

จุดประสงค์ของลำดับชั้นนี้คือมีหน้าที่แจ้ง ViewModel เกี่ยวกับการกระทำของผู้ใช้ และทำการสังเกต (Observe) ViewModel เพื่อแสดงข้อมูลที่เปลี่ยนแปลงไป โดยในลำดับชั้นนี้จะไม่มีการปรากฏตรรกะของแอปพลิเคชันใดๆ

2.1.1.3 ViewModel

เป็นลำดับชั้นที่สร้างขึ้นเพื่อนำมาใช้เกี่ยวกับตรรกะทางธุรกิจ และการรับส่งข้อมูลระหว่างลำดับชั้น View และ Model โดยมีการเก็บสถานะและคงอยู่แม้ผ่านกลางเปลี่ยนแปลงการกำหนดค่าต่างๆ ทำให้ View ไม่จำเป็นต้องดึงข้อมูลอีกครั้งระหว่างกิจกรรมต่างๆ หรือตามการเปลี่ยนแปลงการกำหนดค่า เช่น เมื่อมีการหมุนหน้าจอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการศึกษาระดับปริญญาโทขึ้นไปเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของสถาปัตยกรรมแบบ MVVM มีความคล้ายคลึงกันบางประการกับรูปแบบของสถาปัตยกรรมแบบ MVP โดย ViewModel มีบทบาทคล้ายคลึงกับ Presenter อย่างไรก็ตาม ข้อเสียของสถาปัตยกรรมแบบ MVP ได้รับการแก้ไขโดย MVVM ดังต่อไปนี้

- 1) ViewModel ไม่มีการอ้างอิงใดๆ ต่อ View
- 2) ความสัมพันธ์ระหว่าง View และ ViewModel เป็นแบบ Many to 1
- 3) ไม่ต้องมีคำสั่งกระตุ้นในการอัปเดต View

2.1.3 รูปแบบโครงสร้างของโปรเจกต์แบบ Architecture Components

Architecture Components คือกลุ่มก้อน Library ใหม่จาก Google ซึ่งประกอบไปด้วย Room, LiveData, Lifecycle และ ViewModel ซึ่ง Library ต่างๆเหล่านี้จะช่วยจัดการเรื่อง Lifecycle ช่วยจัดการเรื่องการเชื่อมต่อกับ Database ให้ง่ายขึ้น ช่วยแยก app ให้เป็นส่วนๆ ง่ายต่อการ test ช่วยจัดการ error ที่เกิดขึ้นบ่อยๆ และช่วยลดความซ้ำในการเขียนโค้ด โดยมีแนวคิดสำคัญคือการแยกชั้น Layer การทำงานของแอปพลิเคชันให้โค้ดแต่ละส่วนแยกชั้นออกจากกันอย่างชัดเจน และแต่ละชั้นจะอ้างอิงถึงชั้นที่ติดกับตนเองเท่านั้น

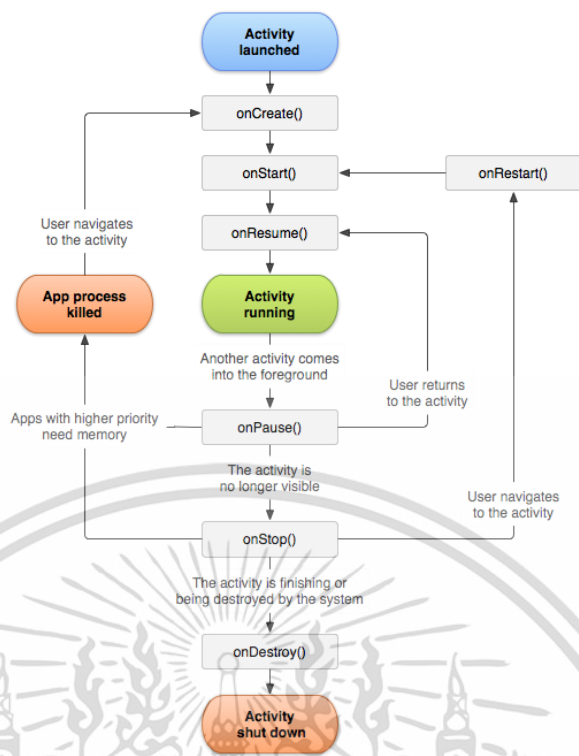
Architecture Components จะแบ่งโค้ดส่วนต่างๆออกเป็นทั้งหมด 3 ชั้นและแต่ละชั้นจะมีส่วนการทำงานดังต่อไปนี้

- 1) Presentation Layer เป็นชั้นบนสุดที่จะติดต่อกับผู้ใช้ รับข้อมูลจากชั้น Domain Layer หรือชั้น Data Layer ขึ้นมาเพื่อแสดงผลแก่ผู้ใช้ อีกทั้งทำหน้าที่ประมวล View Logic และ Business Logic รวมถึงส่งข้อมูลที่ประมวลแล้วส่งกลับไปยังชั้น Domain Layer
- 2) Domain Layer เป็นชั้นที่จะทำหน้าที่เกี่ยวกับ Data Logic โดยจะทำการจัดรูปแบบข้อมูลที่ได้จากชั้น Data Layer ให้อยู่ในรูปแบบที่สามารถใช้แสดงผลได้ก่อนจะส่งไปยังชั้น Presentation Layer และในทางกลับกันก็ต้องจัดการแปลงรูปข้อมูลที่ได้รับจากผู้ใช้ผ่านทาง Presentation Layer ให้สามารถส่งกลับไปยัง Data Layer ได้อีกด้วย
- 3) Data Layer เป็นชั้นที่ของข้อมูล มีหน้าที่จัดหาข้อมูลจากแหล่งข้อมูลต่างๆ ทั้งจาก Local Storage ภายในอุปกรณ์ และจาก Network / Service เพื่อส่งไปแสดงผลบนชั้น Presentation Layer รวมถึงส่งข้อมูลที่ได้จากชั้น Presentation Layer ไปยัง Local Storage หรือ Service ต่างๆ

2.1.4 Android Activity Lifecycle

Android Activity Lifecycle คือ วัฏจักรการทำงานของแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ โดยจะมีการแจ้งกลับจากระบบว่าตอนนี้แอปพลิเคชันอยู่ในสถานะใดในวัฏจักรการทำงานเพื่อกำหนดกิจกรรมที่จะทำในช่วงสถานะนั้นๆ

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์หรือการสงวนเพื่อสิทธิของเจ้าของเท่านั้น เมื่ออนุญาตให้ไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

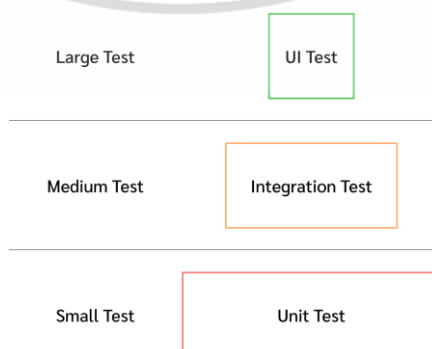


รูปที่ 2.2 ภาพตัวอย่างการทำงานของ Android Activity Lifecycle

ที่มา : <https://developer.android.com/guide/components/activities/activity-lifecycle>

2.1.5 การทดสอบระบบสำหรับแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์

ในกระบวนการการพัฒนาแอปพลิเคชันหนึ่ง ๆ นั้นจะต้องมีการทดสอบการทำงานของแอปพลิเคชันอยู่เรื่อย ๆ เพื่อให้การทำงานของแอปพลิเคชันมีข้อผิดพลาดน้อยที่สุด ซึ่งการทำทดสอบแบบอัตโนมัติจะช่วยให้ผู้พัฒนาประหยัดทั้งเวลาในการทดสอบได้มาก นอกจากนี้ยังสามารถใช้ทดสอบการทำงานในกรณีต่าง ๆ ที่อาจจะไม่สะดวกในการทดสอบด้วยตนเองอีกด้วย โดยการทำการทดสอบแบบทดสอบแบบอัตโนมัติสำหรับแอปพลิเคชันที่ทำงานบนระบบปฏิบัติการแอนดรอยด์นั้นจะแบ่งออกได้เป็น 3 ระดับดังรูปที่ 2.3



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 2.3 แผนภาพพีระมิดของการทดสอบระบบ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่เปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) Small Test มักที่รู้จักในชื่อ Unit Test เป็นการทดสอบการทำงานของชุดคำสั่งในคลาสใดคลาสหนึ่ง เพื่อให้มั่นใจว่าชุดคำสั่งหรือฟังก์ชันต่าง ๆ สามารถทำงานได้ถูกต้องตามหน้าที่ของคลาสนั้นๆ โดยจะเป็นการเขียนชุดคำสั่งขึ้นมาชุดหนึ่งสำหรับการทดสอบแบบอัตโนมัติกับคลาสนั้นๆ ชุดคำสั่งดังกล่าวจะทำการส่งข้อมูลเข้าไปยังคลาสโดยตรง และเปรียบเทียบผลลัพธ์ที่ได้จากคลาสนั้นว่าเป็นไปตามคาดหวังหรือไม่

2) Medium Test หรือที่เรียกว่า Integration Test เป็นการทดสอบการทำงานของโค้ดหลายส่วนร่วมกัน เพื่อให้มั่นใจว่าชุดคำสั่งที่อยู่ต่างคลาสนั้น สามารถทำงานร่วมกันได้อย่างถูกต้อง เช่น การทำงานร่วมกันระหว่างส่วน View กับ ViewModel หรือการทำงานร่วมกันระหว่างส่วนต่าง ๆ ใน Data Layer โดยจะเป็นการเขียนชุดคำสั่งขึ้นมาชุดหนึ่งสำหรับการทดสอบแบบอัตโนมัติกับคลาสนั้นๆ ชุดคำสั่งดังกล่าวจะทำการป้อนข้อมูลเข้าไปยังคลาสใดคลาสหนึ่ง และรอดูผลลัพธ์ที่ได้จากกระบวนการทำงานของคลาสนั้นๆ ว่าเป็นไปตามที่คาดหวังหรือไม่

3) Large Test หรือ UI Test เป็นการทดสอบการทำงานของชุดคำสั่งทั้งหมด โดยจำลองสถานการณ์ต่าง ๆ ให้เหมือนกับการใช้งานจริงของผู้ใช้งาน เพื่อให้มั่นใจว่าแอปพลิเคชันของเราสามารถทำงานได้ตามที่คาดหวังไว้

ซึ่งโดยพื้นฐานของการเขียนการทดสอบแล้ว นักพัฒนาควรเน้นไปที่เขียน Small Test เป็นหลัก และทำการทดสอบ Medium Test และ Large Test ตามกันไป เพราะหากแต่ละคลาสสามารถทำหน้าที่ของตนเองได้สมบูรณ์แล้ว การทำงานในภาพรวมก็จะสมบูรณ์ไปด้วย

2.2 เทคโนโลยีที่เกี่ยวข้อง

2.2.1 ระบบปฏิบัติการแอนดรอยด์ (Android operating system)

ระบบปฏิบัติการแอนดรอยด์ (Android operating system) คือ ระบบปฏิบัติการหรือแพลตฟอร์มที่ถูกพัฒนาขึ้นบนพื้นฐานของระบบปฏิบัติการลินุกซ์ (Linux) เพื่อใช้ในการควบคุมการทำงานบนอุปกรณ์อิเล็กทรอนิกส์ต่างๆ โดยเน้นการทำงานบนโทรศัพท์มือถือและอุปกรณ์พกพา ถูกพัฒนาขึ้นโดยบริษัทแอนดรอยด์ (Android, Inc.) ก่อนที่จะถูกบริษัทกูเกิ้ล (Google Inc.) เข้าซื้อ และปรับให้เป็นระบบปฏิบัติการแบบเปิดเผยซอฟต์แวร์ต้นฉบับ (Open Source) โดยเปิดให้ผู้พัฒนาอื่นสามารถนำไปปรับปรุง แก้ไข และจำหน่ายได้ โดยต้องอยู่ในเงื่อนไขที่ทางกูเกิ้ลนั้นได้กำหนดไว้

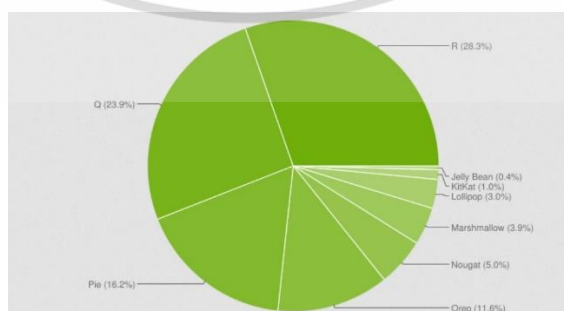
ในปัจจุบัน ระบบปฏิบัติการแอนดรอยด์ได้มีการพัฒนามาหลายหลายเวอร์ชัน โดยมีชื่อและเวอร์ชันดังตารางที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 ชื่อและเวอร์ชันของระบบปฏิบัติการแอนดรอยด์

เวอร์ชัน	ชื่อ	ปีที่เปิดตัว
1.0 – 1.1	Base	2008
1.5	Cupcake	2009
1.6	Donut	2009
2.0 – 2.1	Éclair	2009
2.2 – 2.2.3	Froyo	2010
2.3 – 2.3.7	Gingerbread	2010
3.0 – 3.2.6	Honeycomb	2011
4.0 – 4.0.4	Ice Cream Sandwich	2011
4.1 – 4.3.1	Jelly Bean	2012
4.4 – 4.4.4	KitKat	2013
5.0 – 5.1.1	Lollipop	2014
6.0 – 6.0.1	Marshmallow	2015
7.0 – 7.1.2	Nougat	2016
8.0 – 8.1	Oreo	2017
9	Pie	2018
10	Q	2019
11	Red Velvet Cake	2020
12	Snow Cone	2021
13	Tiramisu	2022

โดยมีการกระจายตัวการใช้งานในแต่ละเวอร์ชันดังรูปที่ 2.4



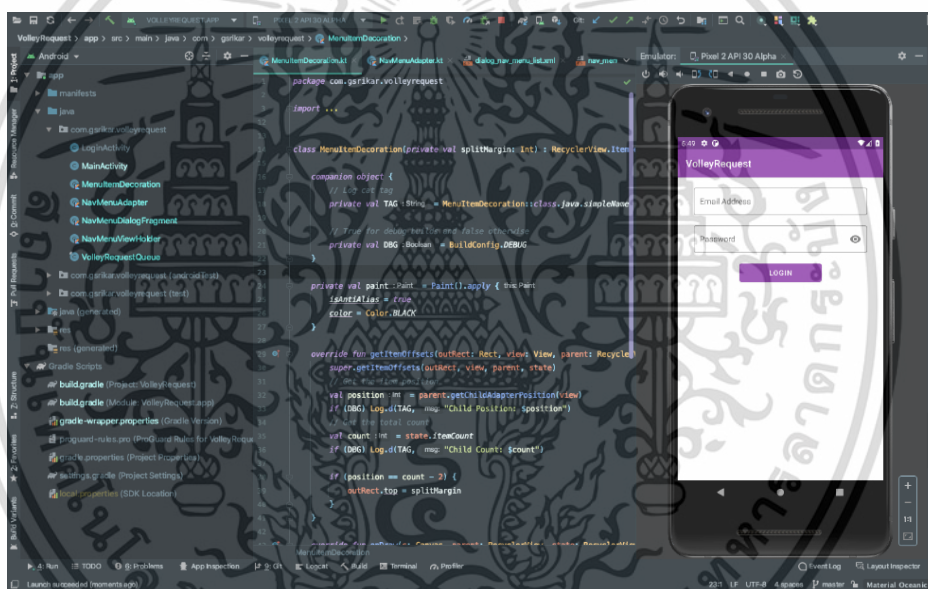
รูปที่ 2.4 การกระจายตัวการใช้งานในแต่ละเวอร์ชันของระบบปฏิบัติการแอนดรอยด์

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การเขียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่เชิงพาณิชย์ การค้า
 ที่มา : <https://techweez.com/2022/05/21/most-widely-used-android-version/>

การพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ จำเป็นที่จะต้องรองรับการทำงานบนอุปกรณ์ที่มีขนาดหน้าจอและความละเอียดแตกต่างกันเป็นอย่างมาก ดังนั้นการพัฒนาแอปพลิเคชันแอนดรอยด์จึงจำเป็นต้องศึกษารูปแบบในการพัฒนาและการทำงานของเครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชันเป็นอย่างดี

2.2.2 Android Studio

Android Studio เป็น IDE Tools ที่พัฒนาโดย Google .Inc ใช้สำหรับการเขียนแอปพลิเคชันบนระบบปฏิบัติการ Android โดยเฉพาะ โดยสามารถเขียนแอปพลิเคชันได้สำหรับหลายอุปกรณ์ เช่น สมาร์ทโฟน แท็บเล็ต สมาร์ททีวี สมาร์ทวอร์ช และอื่นๆ โดย Android Studio สามารถพัฒนาแอปพลิเคชันได้ทุกระบวนการ เช่น การให้คำสั่ง การออกแบบส่วนต่อประสานผู้ใช้ การติดตั้งแอปพลิเคชันบนอุปกรณ์จำลอง การทดสอบต่างๆ เป็นต้น



รูปที่ 2.5 ตัวอย่างส่วนต่อประสานผู้ใช้และอุปกรณ์จำลองของโปรแกรม Android Studio
ที่มา : <https://www.gsrikar.com/2020/09/run-emulator-directly-in-android-studio.html>

2.2.3 Kotlin for Android

Kotlin คือ ภาษาทางโปรแกรมที่กำลังได้รับความนิยมในปัจจุบัน โดยเฉพาะกับการใช้ในการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ เนื่องจากทาง Google .Inc ได้ประกาศให้ Kotlin เป็นภาษาหลักในการพัฒนาแอปพลิเคชันแอนดรอยด์เมื่อปี 2019

ภาษา Kotlin ได้ถูกพัฒนามาจากภาษา Java โดยบริษัท JetBrains โดยมีจุดเด่นหลักๆ ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) Expressive and concise (ชัดเจนและรัดกุม) : ภาษา Kotlin ออกแบบมาให้เขียนโค้ดน้อยที่สุด ลดการเขียนโค้ดที่ไม่จำเป็นออก ทำให้เข้าใจความหมายของโค้ดได้ง่าย และปรับปรุงแก้ไขโค้ดได้ง่ายมากยิ่งขึ้น

2) Safer code (ความปลอดภัย) : Kotlin เป็นภาษาที่ออกแบบเพื่อหลีกเลี่ยงการสร้าง error จากการเขียนโปรแกรม เช่น NullPointerExceptions ในโค้ดเพื่อทำให้เกิดปัญหาน้อยลงในแอปพลิเคชัน

3) Interoperable (ทำงานร่วมกับ Java ได้) : Kotlin ถูกออกแบบมาให้สามารถทำงานร่วมกับ Java ได้ 100% ทำให้สามารถเติมโค้ดภาษา Kotlin ในโปรเจกต์ที่เป็น ภาษา Java ได้ นอกจากนี้ยังมีเครื่องมือในการช่วยเปลี่ยนจากภาษา Java เป็น ภาษา Kotlin ด้วย

4) Structured Concurrency (ทำงานพร้อมกัน) : ด้วยความสามารถของ Coroutines ใน Kotlin จะช่วยให้การเขียน Code แบบ Asynchronous Code สามารถอยู่ในรูปของ Blocking Code ได้ โดยจะช่วยในเรื่องของ Task Management (เช่น การเรียกใช้งานสัญญาณโทรศัพท์หรือการเข้าใช้งาน Database) การใช้ Coroutines เป็นสิ่งที่แนะนำสำหรับการเขียนโปรแกรมแบบ Asynchronous บน Android

2.2.4 Figma

Figma คือ เครื่องมือออกแบบเว็บไซต์ แอปพลิเคชัน โลโก้ และอื่นๆ ผ่านการใช้พีเจอร์ต่าง ๆ ซึ่งมีจุดเด่นอยู่ที่การใช้งานบนได้ทุกระบบปฏิบัติการ และยังมี Community ที่ผู้ใช้สามารถแชร์ไฟล์งาน Prototype หรือ Plug-in ต่าง ๆ แล้วนำไปปรับใช้กับงานของตัวเองได้มากกว่านั้น ผู้ใช้งานสามารถทำงานเป็นทีมได้อย่างมีประสิทธิภาพ เนื่องจากการทำงานบนเว็บเบราว์เซอร์มีการอัปเดตแบบ Real-time ทำให้ในทีมสามารถพูดคุย และปรับแก้งานได้ทันทีเมื่อมีปัญหาระหว่างการทำโปรเจกต์ หรือมีไอเดียเพิ่มเติม

2.2.5 Sourcetree

Sourcetree คือ โปรแกรมสำหรับจัดการ Source code ต่างๆ บนระบบ GIT ผ่าน GUI แทนที่การใช้คำสั่งของ Command line เพื่อให้ง่ายต่อการใช้งานของนักพัฒนา

2.2.6 Postman

Postman คือเครื่องมือสำหรับการพัฒนาและทดสอบ API Service โดยสามารถจำลอง Data, Header รวมถึงบันทึก Collection ที่ใช้ในการทดสอบไว้ เพื่อทดสอบว่าระบบสามารถตอบกลับมาได้ถูกต้องหรือไม่

2.2.7 การแลกเปลี่ยนข้อมูลผ่าน RESTful Web Service

RESTful คือรูปแบบของ Web Service ที่ใช้ REST Architecture Style โดย RESTful อนุญาตให้ระบบ Request และเข้าถึง Resource บนเว็บโดยใช้ชุดคำสั่งที่กำหนดเอาไว้

ล่วงหน้า โดยที่การโต้ตอบของระบบที่ใช้ REST จะอยู่บนพื้นฐานของ Hypertext Transfer Protocol (HTTP). Request จะส่งคำขอไปยัง URI ที่กำหนดและรับ response กลับมาเป็น Payload ในแบบ HTML, XML, JSON หรือ format อื่น ๆ โดยประกอบไปด้วย Client หรือ ผู้ที่จะเข้ามาข้อมูลและ Server หรือผู้เป็นผู้ให้บริการข้อมูล

2.1.8 การแลกเปลี่ยนข้อมูลผ่าน Socket.IO

Socket.IO เป็น Library ที่ใช้สำหรับการแลกเปลี่ยนข้อมูลที่มีเวลาแฝงในการส่งข้อมูลต่ำ และเป็นการแลกเปลี่ยนแบบสองทิศทาง คือ ผู้ใช้สามารถเป็นทั้งผู้รับและผู้ส่งได้ในเวลาเดียวกัน โดยจะสร้างขึ้นบนโพรโทคอล WebSocket และกำหนดการรับส่งข้อมูลผ่านอีเวนต์ (Event)

2.1.9 Koin Injection Framework

Koin คือ Dependency Injection framework แบบ Lightweight สำหรับ Kotlin DSL ช่วยจัดการเรื่องการสร้าง Instant ต่างๆที่ต้องใช้งาน โดยไม่ generate code ขึ้นภายหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีการดำเนินงานวิจัย

ในการพัฒนาคุณลักษณะการสนทนาภายในแอปพลิเคชันสำหรับผู้ให้บริการ (Consumer) ของแพลตฟอร์มคลาวด์คิทเช่น เป็นการพัฒนาเพื่อเพิ่มช่องทางการติดต่อสื่อสารระหว่างผู้ให้บริการกับผู้ประกอบการร้านค้า (Merchant) โดยคุณลักษณะการสนทนาจะอยู่ในรูปแบบของการรับส่งข้อความและรูปภาพต่างๆ ระหว่างผู้ให้บริการกับทางผู้ประกอบการร้านค้า เพื่อให้ทางผู้ให้บริการสามารถติดต่อสื่อสารและแจ้งรายละเอียดเกี่ยวกับรายการเมนูอาหารที่สั่งซื้อได้ง่ายมากยิ่งขึ้น โดยมีขั้นตอนวิธีการดำเนินงานดังต่อไปนี้

3.1 การศึกษาข้อมูลและวิเคราะห์ระบบงานเดิมของแพลตฟอร์มคลาวด์คิทเช่น

3.1.1 การศึกษากระบวนการทำงานของบริษัท

จากการศึกษากระบวนการทำงานของบริษัทพบว่าทางบริษัทได้นำแนวทาง Agile Software Development มาใช้ในการทำงานและพัฒนาระบบ โดยจะมีกรอบการวางแผนและปฏิบัติงานภายในระยะเวลาสองสัปดาห์

3.1.2 การศึกษาการเขียนโปรแกรมด้วยภาษา Kotlin บนระบบปฏิบัติการ Android และการใช้งานโปรแกรม Android Studio

3.1.2.1 การสร้างโปรเจกการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์

การพัฒนาแอปพลิเคชันเพื่อใช้งานบนระบบปฏิบัติการแอนดรอยด์มีความจำเป็นต้องใช้ Android SDK (Android Software Development Kit) ในการพัฒนา โดย Android SDK จะประกอบไปด้วยเครื่องมือต่างๆ ที่ช่วยในการพัฒนาแอปพลิเคชัน ชุดคำสั่งและไลบรารีที่จำเป็นต่อการพัฒนา รวมไปถึงโปรแกรมจำลองอุปกรณ์ที่ใช้ระบบปฏิบัติการ Android (Android Emulator) ซึ่งช่วยให้ผู้พัฒนาสามารถทดสอบการทำงานต่างๆ ของแอปพลิเคชันในสภาพแวดล้อมเสมือนได้ก่อนที่จะนำไปติดตั้ง ทดสอบ และทำงานบนอุปกรณ์ที่ใช้ระบบปฏิบัติการแอนดรอยด์จริง

การเริ่มต้นสร้างแอปพลิเคชันแอนดรอยด์แอปพลิเคชันหนึ่งนั้น จะทำได้โดยการสร้างโปรเจกของตัวแอปพลิเคชันที่ต้องการพัฒนาขึ้นมา โดยจะมีการสร้างโครงสร้างพื้นฐานของแอปพลิเคชันที่ประกอบไปด้วยไฟล์ทั้งหมด 3 ไฟล์ ดังต่อไปนี้

1) Manifest เป็นไฟล์ที่ใช้สำหรับใช้ในการกำหนดค่าต่างๆ ของแอปพลิเคชัน เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า กำหนดชื่อแอปพลิเคชัน กำหนดการแสดงผลขั้นพื้นฐาน กำหนดหน้าหลักหรือหน้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรกที่จะแสดงผลเมื่อแอปพลิเคชันถูกเรียกใช้งาน การขอ Permission เพื่อขอเข้าถึงทรัพยากรต่างๆ จากในอุปกรณ์ เป็นต้น

2) Java Class หรือ Kotlin Class (.java หรือ .kt) เป็นไฟล์ที่บรรจุชุดคำสั่งและตรรกะเกี่ยวกับการทำงานของแอปพลิเคชัน เป็นส่วนหลักที่ใช้เพื่อกำหนดการทำงานของแอปพลิเคชัน

3) Resource (.xml) เป็นไฟล์ที่ใช้สำหรับเก็บทรัพยากรต่างๆ ก่อนที่จะถูก Java Class หรือ Kotlin Class เรียกใช้ เช่น รูปแบบของส่วนต่อประสานผู้ใช้ รูปภาพ ค่าสี รูปแบบของตัวอักษร เป็นต้น

3.1.2.2 การสร้างส่วนต่อประสานผู้ใช้และกำหนดการทำงาน

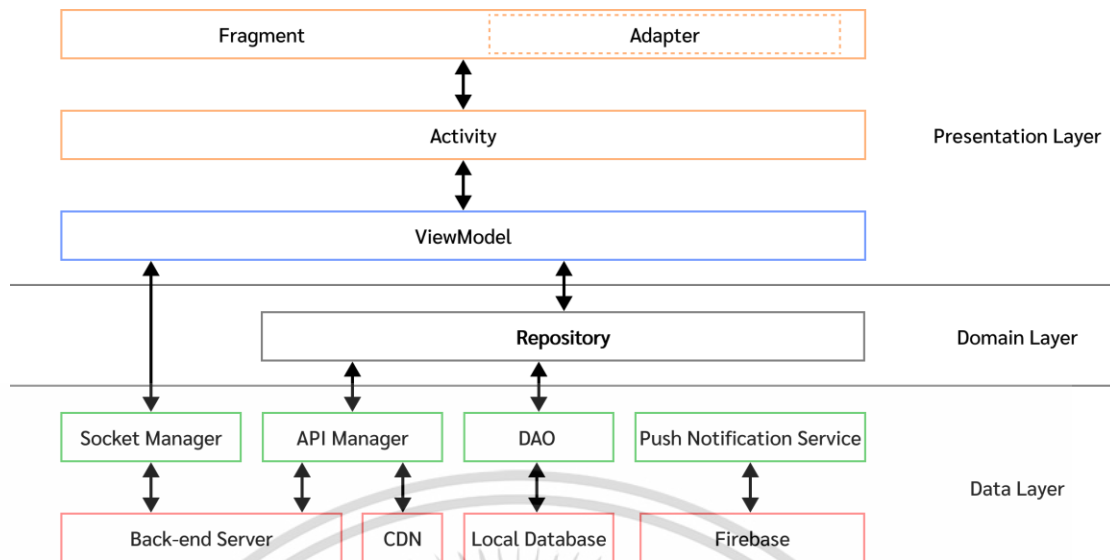
ส่วนต่อประสานผู้ใช้ (User Interface) ของแอปพลิเคชันแอนดรอยด์ เป็นการทำงานร่วมกันระหว่างสองส่วน คือ ส่วนของโครงสร้างเลย์เอาต์ที่อยู่ในรูปแบบของ Resource ไฟล์ และส่วนของ Java Class หรือ Kotlin Class ที่มีหน้าที่เรียกใช้และแสดงผลเลย์เอาต์ รวมไปถึงการกำหนดการแสดงผลและตรรกะในการตอบโต้กับผู้ใช้งาน

3.1.2.2 การแสดงผลแบบรายการ

การแสดงผลแบบรายการในแอปพลิเคชันแอนดรอยด์ ทำได้โดยการสร้างอะแดปเตอร์ (Adapter) ซึ่งเป็น Class ที่ทำงานร่วมกับส่วนต่อประสาน โดยมีหน้าที่จัดการแสดงผลรายการของข้อมูลที่มีลักษณะเป็นการทำซ้ำและเรียงเป็นแถวติดต่อกัน กล่าวคือมีหน้าที่ในการนำเลย์เอาต์มาจัดเรียงบนหน้าจอและกำหนดการแสดงผลของรายการเหล่านั้น

3.1.3 การศึกษาโครงสร้างและสถาปัตยกรรมซอฟต์แวร์ของแอปพลิเคชันสำหรับผู้ใช้งาน

เนื่องจากการพัฒนาคุณลักษณะการสนทนาในแอปพลิเคชันมือถือสำหรับผู้ให้บริการ (Consumer) เป็นการพัฒนาคุณลักษณะการทำงานใหม่บนพื้นฐานของแอปพลิเคชันสำหรับผู้ให้บริการที่มีอยู่เดิม จึงทำให้ผู้จัดทำต้องศึกษาโครงสร้างและสถาปัตยกรรมซอฟต์แวร์ของแอปพลิเคชัน เพื่อช่วยในการออกแบบและพัฒนาให้คุณลักษณะการสนทนานั้นสมบูรณ์และสามารถทำงานได้อย่างเต็มประสิทธิภาพ



รูปที่ 3.1 แผนภาพโครงสร้างของแอปพลิเคชันสำหรับผู้ใช้งาน

จากการศึกษาพบว่าแอปพลิเคชันสำหรับผู้ให้บริการบนแพลตฟอร์มคลาวด์คิทเช่นนี้ ถูกพัฒนาบนพื้นฐานของสถาปัตยกรรมรูปแบบ MVVM ตามแนวคิดของ Architecture Components โดยมีองค์ประกอบในแต่ละชั้นและการทำงานดังต่อไปนี้

3.1.3.1 Data Layer

เป็นชั้นที่ทำหน้าที่จัดการดูแลเกี่ยวกับ Data Source หรือข้อมูลต่างๆ ประกอบด้วย

1) Back-end Server

ระบบหลังบ้านของแพลตฟอร์มคลาวด์คิทเช่น ทำหน้าที่ประมวลผลการทำงาน รวมไปถึงเก็บและรับส่งข้อมูลทั้งหมด ที่ใช้งานในแอปพลิเคชัน

2) CDN Service (Content Delivery Network Service)

เป็นบริการที่ใช้รับฝากรูปบนอินเทอร์เน็ต โดยสามารถอัปโหลดรูปภาพที่ต้องการ จะใช้ขึ้นไปไว้บนเซิร์ฟเวอร์และโหลดภาพมาใช้ผ่าน URL ของรูปภาพได้

3) Firebase

เป็นแพลตฟอร์มที่ให้บริการเกี่ยวกับระบบหลังบ้าน (Back-end as a Service) ที่ จะทำการส่งข้อความแจ้งเตือนแบบข้ามแพลตฟอร์ม (Cross platform messaging)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) Socket Manager

เป็นคลาสที่ทำหน้าที่เกี่ยวกับการจัดการส่งและการรับ Event จากฝั่งเซิร์ฟเวอร์ เพื่อใช้งานในการอัปเดตสถานะของคำสั่งซื้อ เช่น แสดงสถานะว่าผู้ประกอบการร้านค้ากำลังปรุงอาหาร แสดงสถานะว่าอาหารกำลังถูกจัดส่ง เป็นต้น

5) API Manager

เป็นคลาสที่จัดการการทำงานเกี่ยวกับการสร้างและการส่ง Request การรับ Response และข้อมูลจากทางฝั่งเซิร์ฟเวอร์หรือเซอร์วิสอื่นๆ โดยจะใช้ในการขอข้อมูลทั่วไปที่มีการใช้งานในแอปพลิเคชัน เช่น การขอข้อมูลคำสั่งซื้อ เป็นต้น

6) DAO (Data Access Object)

เป็นคลาส Kotlin ที่ทำหน้าที่ในการกำหนดคำสั่งจัดการข้อมูลภายในฐานข้อมูล Local Database เป็นคลาสที่มีฟังก์ชันที่บรรจุคำสั่ง SQL ต่าง ๆ ไว้ให้คลาสอื่น ๆ เรียกใช้งานเพื่อจัดการกับข้อมูลต่าง ๆ ในฐานข้อมูล

7) Push Notification Service

เป็นคลาสที่ใช้รับข้อมูลเกี่ยวกับการแจ้งเตือนจาก Firebase Cloud Message และทำหน้าที่แจ้งเตือนบนอุปกรณ์ โดยใช้ในการแสดงการแจ้งเตือนเมื่อสถานะต่างๆ ของคำสั่งซื้อเปลี่ยนแปลงไป เช่น การแจ้งเตือนว่าร้านค้ากำลังปรุงอาหาร การแจ้งเตือนว่าคำสั่งซื้อถูกยกเลิก เป็นต้น

3.1.3.2 Domain Layer

เป็นชั้นที่ทำหน้าที่จัดการดูแลเกี่ยวกับตรรกะทางข้อมูลต่างๆ ของแอปพลิเคชัน

1) Repository

เป็นคลาส Kotlin ที่ทำหน้าที่เป็นศูนย์กลางรวมข้อมูลต่างๆ จากแหล่งข้อมูลทั้ง API Manager และ Local Database โดยในแง่ของการแลกเปลี่ยนข้อมูลผ่าน API Manager นั้น Repository จะทำแปลงข้อมูลที่ได้จาก Response เป็นข้อมูลที่สามารถใช้งานได้ ก่อนจะส่งไปยัง ViewModel ต่อไป ส่วนในแง่ของ Local Database เมื่อ ViewModel ต้องการเก็บข้อมูลใดๆ ก็สามารถส่งต่อให้ Repository จัดการต่อด้วยการแปลงข้อมูลดังกล่าวเป็นสามารถบรรจุลงในฐานข้อมูล SQL ได้ก่อนทำการบันทึกข้อมูลดังกล่าวต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2.3 Presentation Layer

ชั้นของส่วนที่ทำหน้าที่เกี่ยวกับตรรกะการตอบโต้กับผู้ใช้และ Business Logic

1) ViewModel

เป็นตัวกลางระหว่างส่วนของข้อมูลและส่วนต่อประสานผู้ใช้ ทำหน้าที่รับส่งข้อมูลระหว่างส่วนของข้อมูลและส่วนต่อประสานผู้ใช้

2) Activity

เป็นส่วนหลักของ View ซึ่งจะเป็นตัวกำหนดวงจรชีวิตต่างๆ เพื่อกำหนดการทำงานในวงจรชีวิต เช่น การเชื่อมต่อ Socket เป็นต้น

3) Fragment

เป็นส่วนของ View ที่ทำงานอยู่บน Activity อีกชั้นหนึ่ง ทำหน้าที่แสดงผลหน้าจอตอบโต้กับผู้ใช้เมื่อกดปุ่มต่างๆ รวมถึงการรับข้อมูลจากผู้ใช้

4) Adapter

เป็นคลาสที่มีหน้าที่กำหนดการแสดงผลของ View ใน recyclerView ว่าให้มีการแสดงผลในรูปแบบใด เช่น เมื่อเป็นประเภทข้อความของผู้ใช้บริการ ให้ข้อความแสดงผลที่ทางด้านขวา

3.2 การศึกษาและวิเคราะห์ความต้องการของผู้ใช้และความต้องการของระบบ

ในการพัฒนาคุณลักษณะการสนทนาภายในแอปพลิเคชันสำหรับผู้ใช้บริการของแพลตฟอร์มคลาวด์คิดเช่น มีการศึกษาพฤติกรรมของผู้ใช้บริการในการใช้งานแอปพลิเคชันที่มีคุณลักษณะการสนทนาและข้อจำกัดต่างๆ ของทางแอปพลิเคชันสำหรับผู้ใช้งานและระบบโดยรวมของแพลตฟอร์มคลาวด์คิดเช่น จึงสามารถสรุปเป็นความต้องการในการพัฒนาคุณลักษณะการสนทนาได้ดังนี้

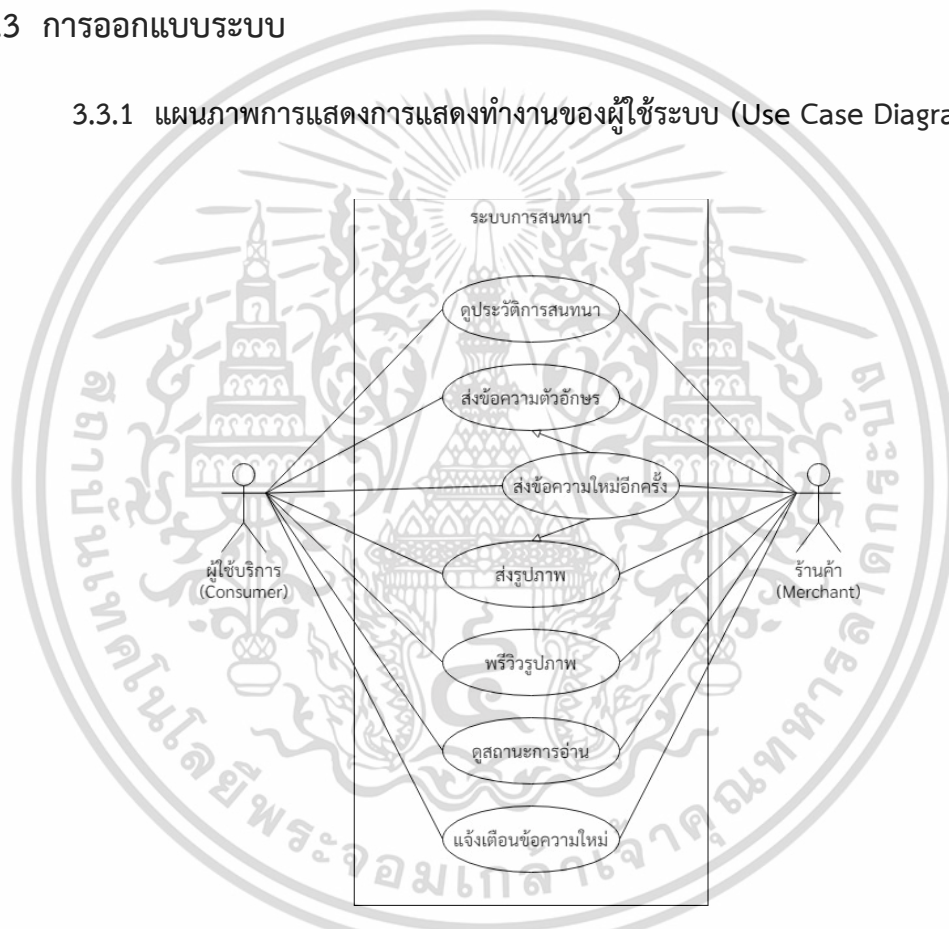
- 1) คุณลักษณะการสนทนาสามารถ รับ/ส่ง ข้อความตัวอักษรได้
- 2) คุณลักษณะการสนทนาสามารถ รับ/ส่ง รูปภาพได้
- 3) คุณลักษณะการสนทนาสามารถแสดงรูปภาพขนาดใหญ่เมื่อผู้ใช้งานกดได้
- 4) คุณลักษณะการสนทนาสามารถดูประวัติการสนทนาที่ผ่านมาได้
- 5) เมื่อคู่สนทนาอ่านข้อความแล้ว จะแสดงข้อความแสดงสถานะ "อ่านแล้ว" ด้านข้างข้อความ
- 6) ในกรณีที่มีข้อความส่งไม่สำเร็จ ให้เก็บไว้ที่ Local Database และสามารถกดปุ่มเพื่อส่งข้อความใหม่อีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 7) คุณลักษณะการสนทนาจะมีการแจ้งเตือนผ่าน Notification เมื่อมีข้อความสนทนาใหม่เข้ามา
- 8) การแจ้งเตือนใน Notification ต้องมีการจัดกลุ่มไว้ตามรายการการคำสั่งซื้อ
- 9) เมื่อผู้ใช้กดที่ Notification สามารถพาไปยังหน้าการสนทนาของคำสั่งซื้อนั้นได้
- 10) เมื่อมีข้อความใหม่และผู้ใช้ยังไม่ได้เปิดเข้ามาอ่านข้อความ ปุ่มแชทที่หน้าต่างๆ จะแสดงจุดสีแดง
- 11) ห้องแชทจะเข้าสู่ Empty Stage เมื่อเวลาผ่านไป 24 ชั่วโมงหลังจากส่งออเดอร์

3.3 การออกแบบระบบ

3.3.1 แผนภาพการแสดงผลการทำงานของผู้ใช้ระบบ (Use Case Diagram)



รูปที่ 3.2 แผนภาพแสดงการทำงานของใช้ระบบการสนทนา

การสนทนาจะเริ่มขึ้นเมื่อทางผู้ให้บริการสั่งซื้ออาหารกับทางผู้ประกอบการร้านค้า โดยผู้ให้บริการสามารถเข้าไปยังห้องสนทนา ได้ทันที ซึ่งแต่ละคำสั่งซื้อจะมีห้องสนทนาเป็นของตัวเอง เมื่อผู้ให้บริการเข้าไปยังห้องสนทนาแล้ว ผู้ให้บริการจะเห็นประวัติการสนทนายระหว่างผู้ให้บริการกับทางผู้ประกอบการร้านค้า (กรณีที่มีประวัติการสนทนา) ซึ่งผู้ให้บริการจะสามารถสนทนากับทางผู้ประกอบการร้านค้าได้ทั้งการส่งตัวอักษรและรูปภาพ เอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อข้อความถูกส่งสำเร็จ ผู้ประกอบการร้านค้าจะได้รับข้อความนั้นๆ หากข้อความดังกล่าวเป็นข้อความรูปภาพ จะแสดงรูปภาพในห้องสนทนา ซึ่งทั้งผู้ใช้บริการและผู้ประกอบการร้านค้าสามารถกดที่รูปภาพเพื่อดูรูปภาพขนาดใหญ่แบบเต็มจอได้

ในกรณีที่ผู้ใช้บริการมีการส่งข้อความและเกิดเหตุสัญญาณอินเทอร์เน็ตขัดข้อง ทำให้ส่งข้อความได้ไม่สำเร็จ ข้อความดังกล่าวจะถูกเก็บลงในฐานข้อมูลบนอุปกรณ์ เมื่อสัญญาณอินเทอร์เน็ตกลับมาใช้งานได้ ผู้ใช้บริการจะสามารถกดส่งข้อความใหม่อีกครั้งได้

ผู้ใช้บริการยังสามารถทราบได้อีกว่าข้อความที่ส่งออกไปนั้นถูกอ่านโดยผู้ประกอบการร้านค้า หรือยังไม่ถูกอ่าน โดยเมื่อผู้ประกอบการร้านค้าได้ทำการเปิดอ่านข้อความจากทางผู้ใช้บริการแล้ว ข้อความดังกล่าวจะปรากฏอักษรบ่งบอกสถานะว่าข้อความนั้นถูกเปิดอ่านแล้วด้านข้างข้อความนั้นๆ

นอกจากนี้ เมื่อผู้ใช้บริการได้รับข้อความใหม่จากทางผู้ประกอบการร้านค้า ผู้ใช้บริการจะได้รับการแจ้งเตือนแบบ Push Notification เพื่อแจ้งให้ผู้ใช้บริการทราบว่ามีความใหม่ของผู้ประกอบการร้านค้าส่งเข้ามา

ตารางที่ 3.1 คำอธิบาย Use Case Description ของดูประวัติการสนทนา

Name	ดูประวัติการสนทนา
Description	ผู้ใช้บริการสามารถดูประวัติการสนทนายระหว่างผู้ใช้บริการกับทางผู้ประกอบการร้านค้า โดยประวัติการสนทนาจะเป็นประวัติการสนทนาของห้องแชทนั้นๆ
Input	
Output	ประวัติการสนทนา
Condition	ผู้ใช้บริการจะสามารถดูประวัติการสนทนาได้ เฉพาะคำสั่งซื้อที่ทำการสั่งซื้อไม่เกิน 24 ชั่วโมง

ตารางที่ 3.2 คำอธิบาย Use Case Description ของส่งข้อความตัวอักษร

Name	ส่งข้อความตัวอักษร
Description	ผู้ใช้บริการสามารถส่งข้อความที่เป็นตัวอักษร และอีโมจิ (Emoji) ไปยังผู้ประกอบการร้านค้าได้
Input	ข้อความตัวอักษร และอีโมจิ (Emoji)
Output	เวลาที่ข้อความถูกส่งได้สำเร็จ
Condition	ข้อความตัวอักษรที่ลูกค้าผู้ใช้บริการทำการส่งไปต้องไม่เกิน 128 ตัวอักษรและจะต้องมีการเชื่อมต่ออินเทอร์เน็ต

ตารางที่ 3.3 คำอธิบาย Use Case Description ของส่งข้อความรูปภาพ

Name	ส่งข้อความรูปภาพ
Description	ผู้ใช้บริการสามารถส่งข้อความที่เป็นรูปภาพไปยังผู้ประกอบการร้านค้าได้
Input	รูปภาพ
Output	เวลาที่ข้อความถูกส่งได้สำเร็จ
Condition	ส่งรูปภาพได้ครั้งละหนึ่งรูปภาพและจะต้องมีการเชื่อมต่ออินเทอร์เน็ต

ตารางที่ 3.4 คำอธิบาย Use Case Description ของพรีวิวรูปภาพ

Name	พรีวิวรูปภาพ
Description	ผู้ใช้บริการสามารถกดที่ข้อความรูปภาพเพื่อดูรูปภาพแบบเต็มจอได้
Input	
Output	แสดงหน้าพรีวิวรูปภาพที่มีขนาดเต็มจอ

ตารางที่ 3.5 คำอธิบาย Use Case Description ของดูสถานะการอ่าน

Name	การดูสถานะการอ่าน
Description	ผู้ใช้บริการสามารถตรวจสอบสถานะการอ่านของข้อความที่ได้ทำการส่งไปยังผู้ประกอบการร้านค้าได้ โดยหากลูกค้าได้ทำการเข้าสู่ห้องสนทนาและเปิดอ่านข้อความแล้ว ข้อความที่ถูกส่งจากผู้ใช้บริการจะมีสถานะขึ้นให้รู้ว่าข้อความดังกล่าวถูกอ่านโดยผู้ประกอบการร้านค้าแล้ว
Input	
Output	สถานะการอ่านของข้อความ
Condition	กรณีที่ข้อความล่าสุดขึ้นสถานะว่าอ่านแล้ว ข้อความก่อนหน้าทั้งหมดจะต้องมีสถานะว่าอ่านแล้วเช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.6 คำอธิบาย Use Case Description ของแจ้งเตือนข้อความ

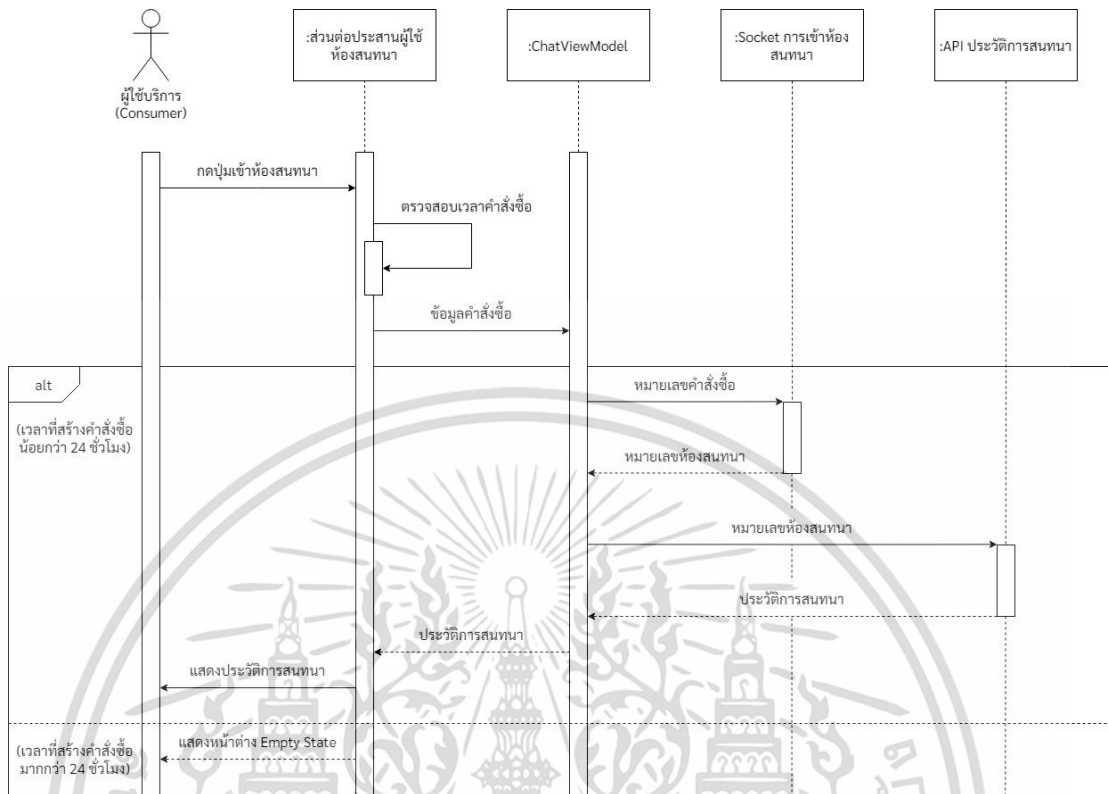
Name	แจ้งเตือนข้อความ
Description	เมื่อผู้ประกอบการร้านค้าทำการส่งข้อความไปยังผู้ใช้บริการสำเร็จ ทางผู้ใช้บริการจะได้รับการแจ้งเตือนข้อความ (notification) โดยผู้ใช้บริการ สามารถกด notification เพื่อไปยังห้องแชทนั้นได้
Input	
Output	แสดงผลแจ้งเตือนข้อความว่าได้รับข้อความใหม่

ตารางที่ 3.7 คำอธิบาย Use Case ของส่งข้อความใหม่อีกครั้ง

Name	ส่งข้อความใหม่อีกครั้ง
Description	มีการส่งข้อความและเกิดเหตุสัญญาณอินเทอร์เน็ตขัดข้อง ทำให้ส่งข้อความได้ไม่สำเร็จ ข้อความดังกล่าวจะถูกเก็บลงในฐานข้อมูลบนอุปกรณ์ เมื่อสัญญาณอินเทอร์เน็ตกลับมาใช้งานได้ ผู้ใช้บริการจะสามารถกดส่งข้อความใหม่อีกครั้งได้
Input	
Output	เวลาที่ข้อความถูกส่งได้สำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 แผนผังการทำงานแบบลำดับปฏิสัมพันธ์ (Sequence Diagram)



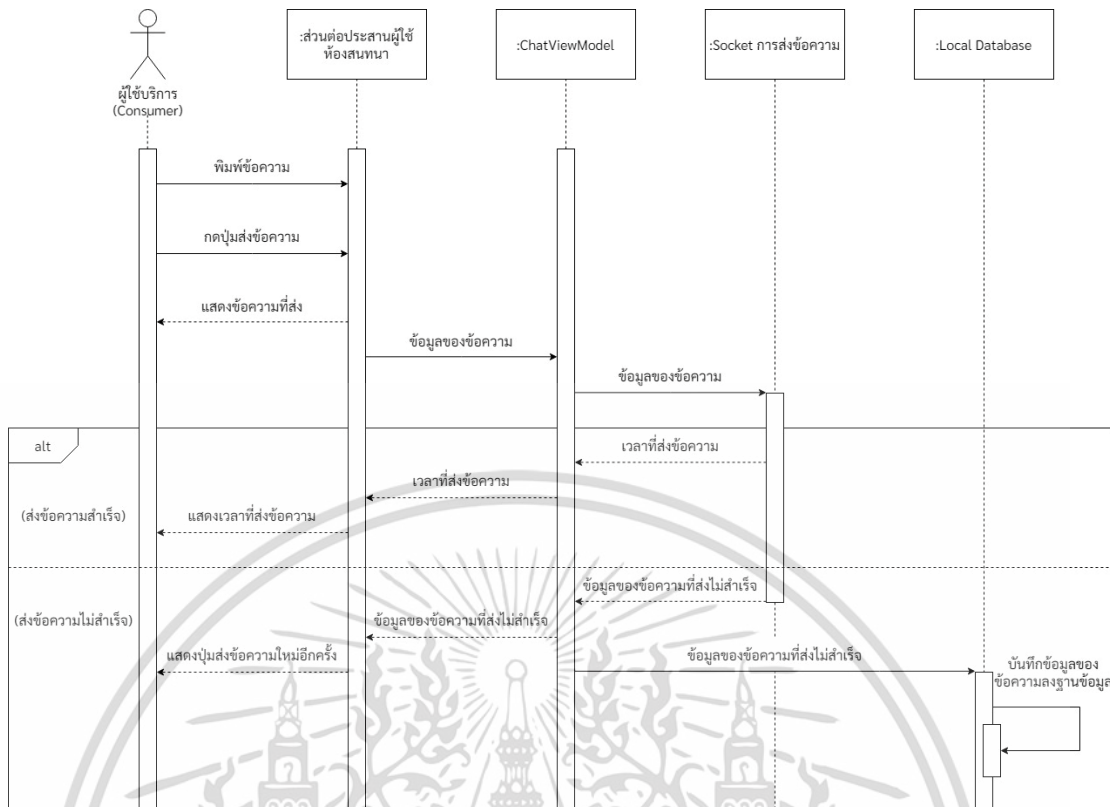
รูปที่ 3.3 แผนภาพ Sequence Diagram ของการดูประวัติการสนทนา

เริ่มต้นเมื่อผู้ให้บริการทำการกดปุ่มเพื่อเข้าสู่ห้องสนทนาผ่านหน้าสถานะคำสั่งซื้อ เมื่อเข้าไปที่หน้าการสนทนาแล้ว จะทำการตรวจสอบวันและเวลาในการสั่งซื้อ หากระยะเวลาที่ทำการสั่งซื้อยังไม่เกิน 24 ชั่วโมง ChatViewModel จะทำการส่งเลขที่คำสั่งซื้อดังกล่าวไปยังเซิร์ฟเวอร์ผ่าน Socket เพื่อขอหมายเลขของห้องสนทนาของคำสั่งซื้อ กลับมาใช้ในการขอประวัติการสนทนาจาก API

เมื่อ API ได้ทำการส่งข้อมูลประวัติการสนทนากลับมา ประวัติการสนทนาจะถูกนำมาแสดงผลในห้องสนทนานั้น

แต่ถ้าหากระยะเวลาในการสั่งซื้อของคำสั่งซื้อนั้นๆ มีระยะเวลาที่ทำการสั่งซื้อเกิน 24 ชั่วโมงแล้ว Fragment จะทำการแสดงหน้าจอ Empty State ซึ่งผู้ให้บริการจะไม่สามารถดูประวัติการสนทนาที่ผ่านมาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

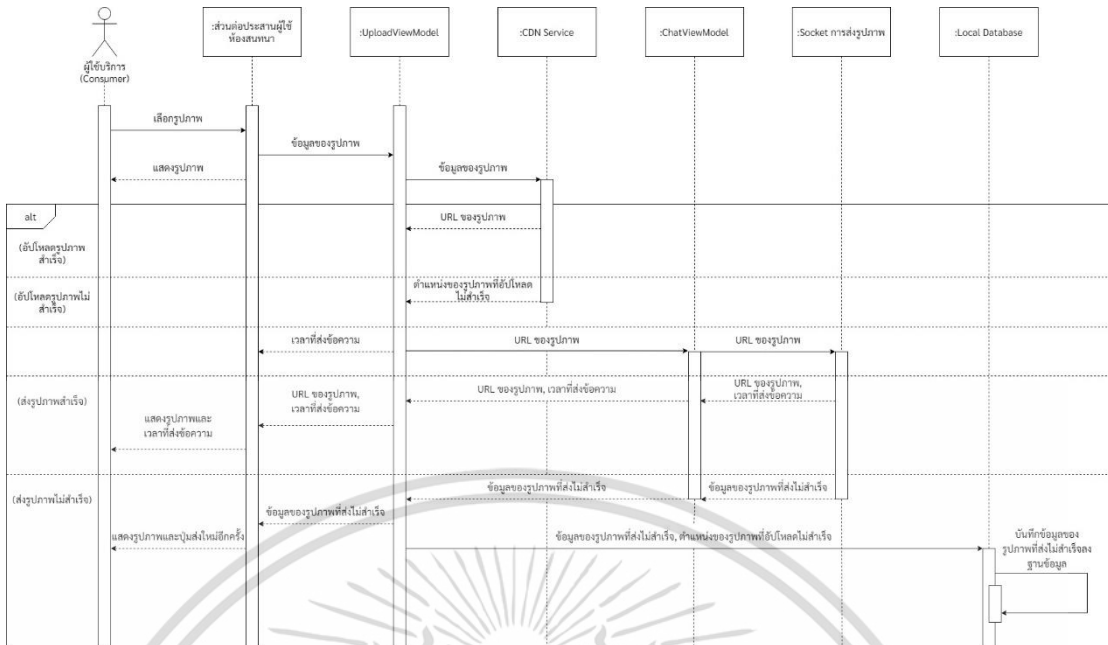


รูปที่ 3.4 แผนภาพ Sequence Diagram ของการส่งข้อความตัวอักษร

ผู้ใช้บริการสามารถกรอกข้อความและกดปุ่มเพื่อส่งข้อความไปยังผู้ประกอบการร้านค้าได้ โดยเมื่อกดปุ่มส่งข้อความแล้ว ห้องสนทนาจะทำการแสดงผลข้อความและส่งข้อความดังกล่าวไปยัง ChatViewModel โดยภายใน ChatViewModel จะทำการสร้างข้อมูลของข้อความ และทำการส่งข้อความดังกล่าวไปยังผู้ประกอบการร้านค้าผ่าน Socket และเมื่อข้อความถูกส่งสำเร็จ ทางเซิร์ฟเวอร์จะทำการคืนค่าของเวลาที่ส่งข้อความกลับมาเพื่อแสดงผลในห้องสนทนาของผู้ใช้บริการ

ในกรณีที่ส่งข้อความไม่สำเร็จเนื่องจากอินเทอร์เน็ตขัดข้อง Socket จะทำการคืนค่ากลับมาให้ทราบว่าส่งข้อความล้มเหลว และแสดงผลปุ่มส่งใหม่อีกครั้งข้างข้อความดังกล่าวในห้องสนทนา โดยข้อมูลของข้อความที่ส่งไม่สำเร็จจะถูก ChatViewModel ส่งไปบันทึกยัง Local Database เพื่อให้ข้อมูลดังกล่าวไม่หายไปเมื่อผู้ใช้บริการปิดห้องสนทนาลงและเก็บไว้ใช้ในการส่งข้อความใหม่ในครั้งต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



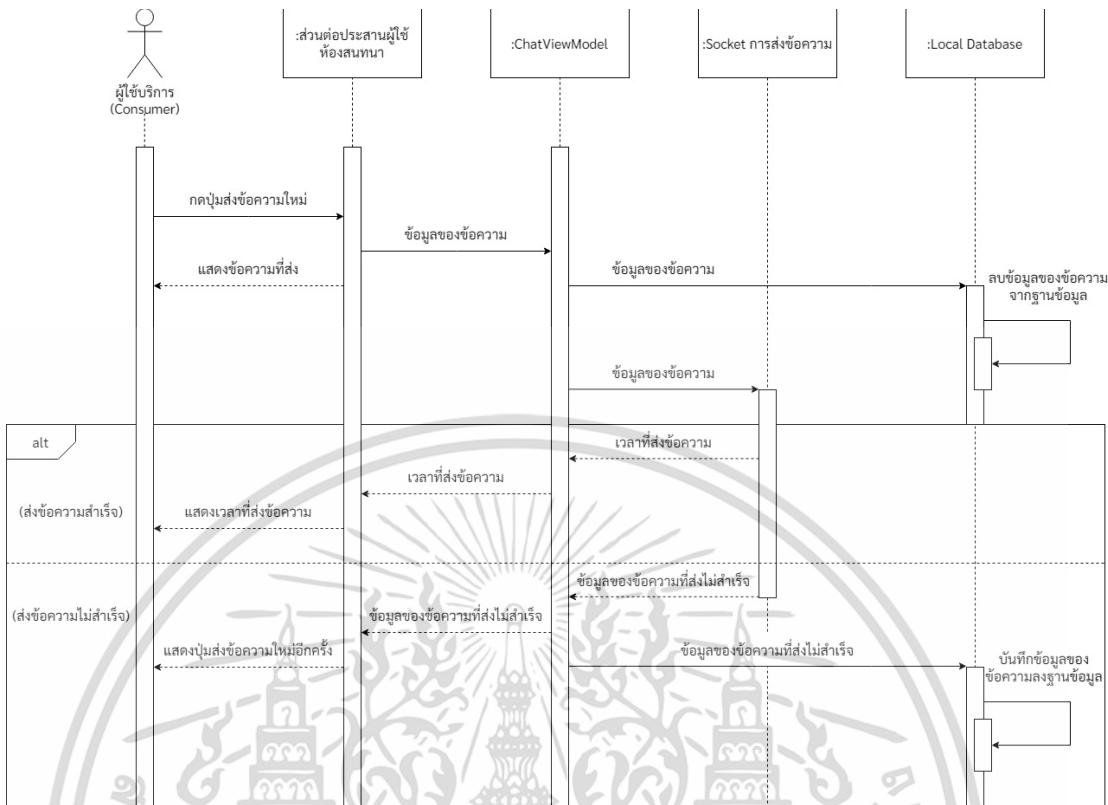
รูปที่ 3.5 แผนภาพ Sequence Diagram ของการส่งข้อความรูปภาพ

ผู้ให้บริการสามารถกดปุ่มส่งรูปภาพและเลือกรูปภาพที่ต้องการจะส่งไปยังผู้ประกอบการร้านค้าได้ โดยเมื่อกดเลือกรูปภาพแล้ว ห้องสนทนาจะทำการแสดงผลรูปภาพนั้นและส่งรูปภาพดังกล่าวไปยัง UploadViewModel เพื่อทำการอัปโหลดรูปภาพนั้นขึ้นไปยัง CDN Service โดยเมื่อทำการอัปโหลดสำเร็จ ทาง CDN Service จะคืนค่า URL ของรูปภาพดังกล่าวกลับมา จากนั้น ChatViewModel จะทำการสร้างข้อมูลของข้อความรูปภาพนั้นและส่งข้อความรูปภาพดังกล่าวไปยังผู้ประกอบการร้านค้าผ่าน Socket และเมื่อข้อความถูกส่งสำเร็จ ทางเซิร์ฟเวอร์จะทำการคืนค่าของเวลาที่ส่งข้อความกลับมาเพื่อแสดงผลในห้องสนทนาของผู้ให้บริการ

ในกรณีที่อัปโหลดรูปภาพไม่สำเร็จ CDN Service จะทำการคืนค่ากลับมาให้ทราบว่าอัปโหลดรูปภาพล้มเหลว จากนั้นห้องสนทนาจะแสดงผลปุ่มส่งใหม่อีกครั้งข้างข้อความดังกล่าวในห้องสนทนาและทำการส่งตำแหน่งของไฟล์รูปบนอุปกรณ์ให้ ChatViewModel เพื่อส่งไปบันทึกยัง Local Database ให้พร้อมสำหรับการอัปโหลดใหม่อีกครั้งเมื่อผู้ให้บริการกดปุ่มส่งรูปภาพใหม่

ในกรณีที่ส่งอัปโหลดรูปภาพสำเร็จแล้ว แต่ส่งรูปภาพไม่สำเร็จเนื่องจากอินเทอร์เน็ตขัดข้อง Socket จะทำการคืนค่ากลับมาให้ทราบว่าส่งข้อความล้มเหลว และแสดงผลปุ่มส่งใหม่อีกครั้งข้างข้อความดังกล่าวในห้องสนทนา โดยข้อมูลของข้อความที่ส่งไม่สำเร็จจะถูก ChatViewModel ส่งไปบันทึกยัง Local Database เพื่อให้ข้อมูลดังกล่าวไม่หายไปเมื่อ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อเผยแพร่ความรู้เท่านั้น ไม่สามารถนำข้อมูลไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

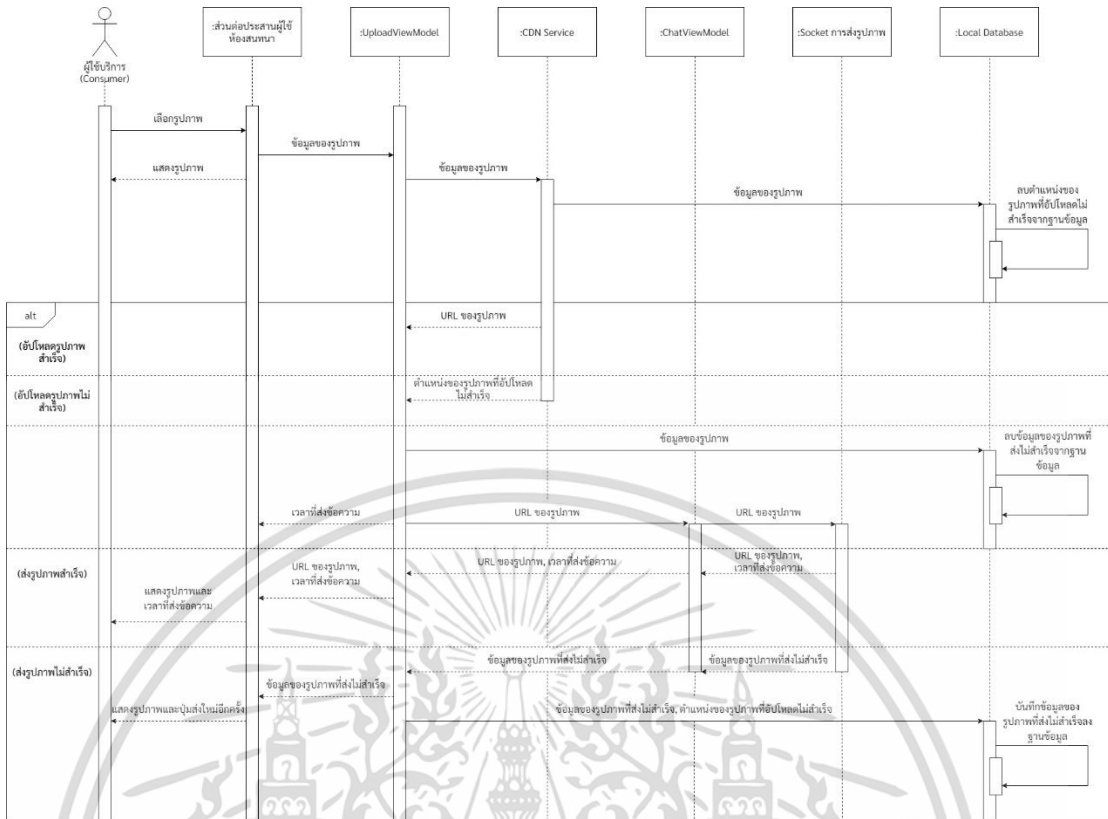


รูปที่ 3.6 แผนภาพ Sequence Diagram ของการส่งข้อความใหม่อีกครั้ง

เมื่อมีข้อความที่ถูกส่งไม่สำเร็จและทางผู้ให้บริการได้กดปุ่มส่งข้อความใหม่อีกครั้ง ChatViewModel จะส่งข้อมูลของข้อความนั้นไปยัง Local Database เพื่อให้ Local Database ค้นหาและดึงข้อมูลของข้อความดังกล่าวมาใช้ จากนั้น Local Database จะลบข้อมูลของข้อความดังกล่าว หลังจากได้ข้อมูลมาจาก Local Database แล้ว ChatViewModel จะทำการส่งข้อความใหม่อีกครั้งผ่าน Socket และเมื่อข้อความถูกส่งสำเร็จ ทางเซิร์ฟเวอร์จะทำการคืนค่าของเวลาที่ส่งข้อความกลับมาเพื่อแสดงผลในห้องสนทนาของผู้ใช้บริการ

ในกรณีที่ส่งข้อความไม่สำเร็จอีกครั้ง Socket จะทำการคืนค่ากลับมาให้ทราบว่าส่งข้อความล้มเหลว และแสดงผลปุ่มส่งใหม่อีกครั้งข้างข้อความดังกล่าวในห้องสนทนา โดยข้อมูลของข้อความที่ส่งไม่สำเร็จจะถูก ChatViewModel ส่งไปบันทึกยัง Local Database เพื่อให้ข้อมูลดังกล่าวไม่หายไปเมื่อผู้ใช้บริการปิดห้องสนทนาลงและเก็บไว้ใช้ในการส่งข้อความใหม่ในครั้งต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



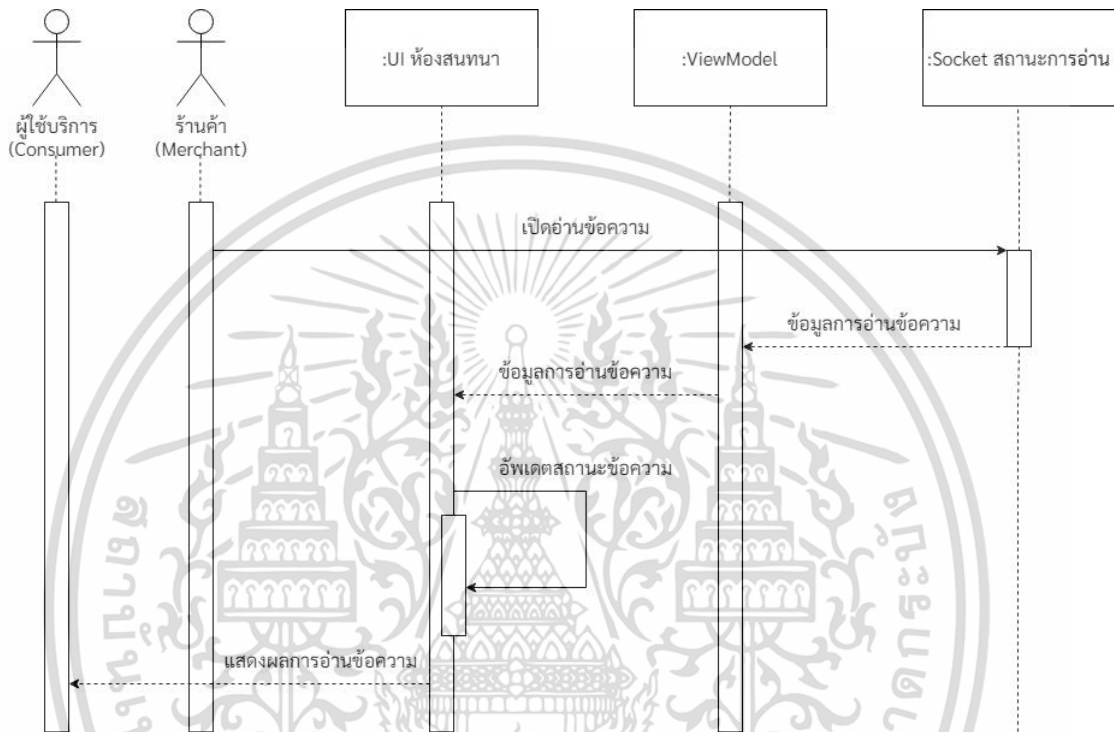
รูปที่ 3.7 แผนภาพ Sequence Diagram ของการส่งข้อความรูปภาพใหม่อีกครั้ง

เมื่อมีรูปภาพที่ถูกอัปโหลดไม่สำเร็จและทางผู้ให้บริการได้กดปุ่มส่งรูปภาพใหม่อีกครั้ง ChatViewModel จะส่งข้อมูลของข้อความนั้นไปยัง Local Database เพื่อให้ Local Database ค้นหาและดึงข้อมูลของรูปภาพ รวมไปถึงตำแหน่งของรูปภาพดังกล่าวบนอุปกรณ์ จากนั้น Local Database จะลบข้อมูลของรูปภาพดังกล่าว หลังจากได้ข้อมูลมาจาก Local Database แล้ว UploadViewModel จะทำการอัปโหลดรูปภาพดังกล่าวขึ้นไปยัง CDN Service อีกครั้งโดยเมื่อทำการอัปโหลดสำเร็จ ทาง CDN Service จะคืนค่า URL ของรูปภาพดังกล่าวกลับมา จากนั้น ChatViewModel จะทำการสร้างข้อมูลของข้อความรูปภาพนั้น และส่งข้อความรูปภาพดังกล่าวไปยังผู้ประกอบการร้านค้าผ่าน Socket และเมื่อข้อความถูกส่งสำเร็จ ทางเซิร์ฟเวอร์จะทำการคืนค่าของเวลาที่ส่งข้อความกลับมาเพื่อแสดงผลในหน้าจอสนทนาของผู้ใช้บริการ

ในกรณีที่อัปโหลดรูปภาพไม่สำเร็จอีกครั้ง CDN Service จะทำการคืนค่ากลับมาให้ทราบ ว่าอัปโหลดรูปภาพล้มเหลว จากนั้นหน้าจอสนทนาจะแสดงผลปุ่มส่งใหม่อีกครั้งข้างข้อความดังกล่าวในหน้าจอสนทนาและทำการส่งตำแหน่งของไฟล์รูปบนอุปกรณ์ให้ ChatViewModel เพื่อส่งไปยังที่ที่ยัง Local Database ให้พร้อมสำหรับการอัปโหลดใหม่อีกครั้งเมื่อผู้บริการ

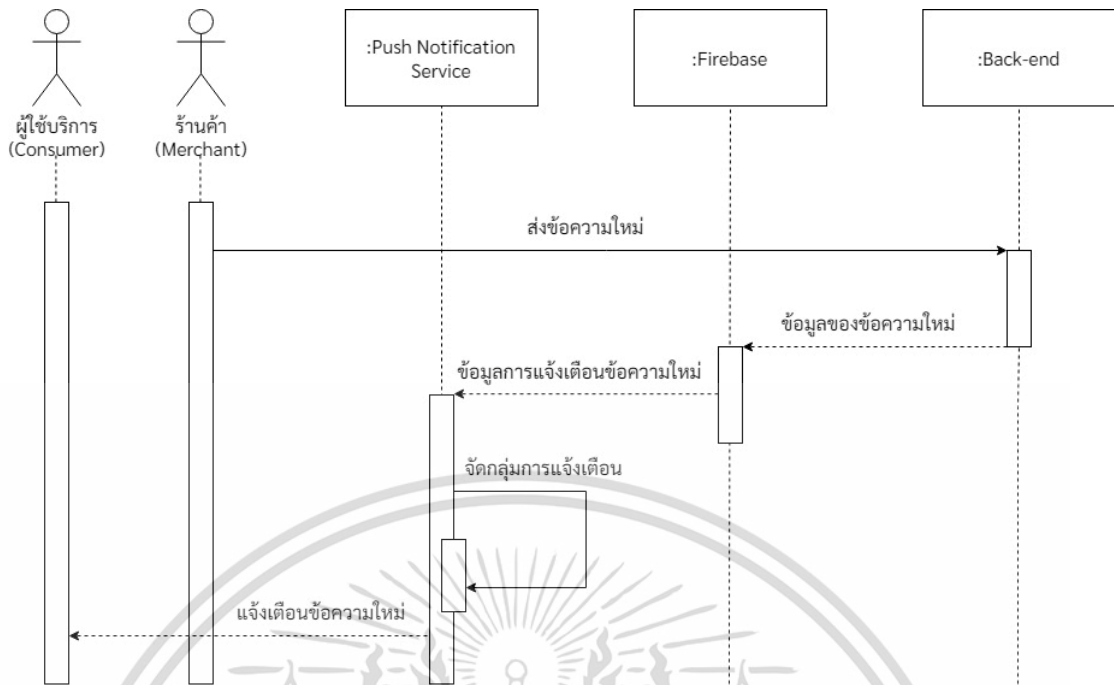
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่ส่งอัปเดตรูปภาพสำเร็จแล้ว แต่ส่งรูปภาพไม่สำเร็จเนื่องจากอินเทอร์เน็ตขัดข้องอีกครั้ง Socket จะทำการคืนค่ากลับมาให้ทราบว่าส่งข้อความล้มเหลว และแสดงผลปุ่มส่งใหม่อีกครั้งข้างข้อความดังกล่าวในห้องสนทนา โดยข้อมูลของข้อความที่ส่งไม่สำเร็จจะถูก ChatViewModel ส่งไปบันทึกยัง Local Database เพื่อให้ข้อมูลดังกล่าวไม่หายไปเมื่อผู้ใช้บริการปิดห้องสนทนาลงและเก็บไว้ใช้ในการส่งรูปภาพใหม่ในครั้งต่อไป



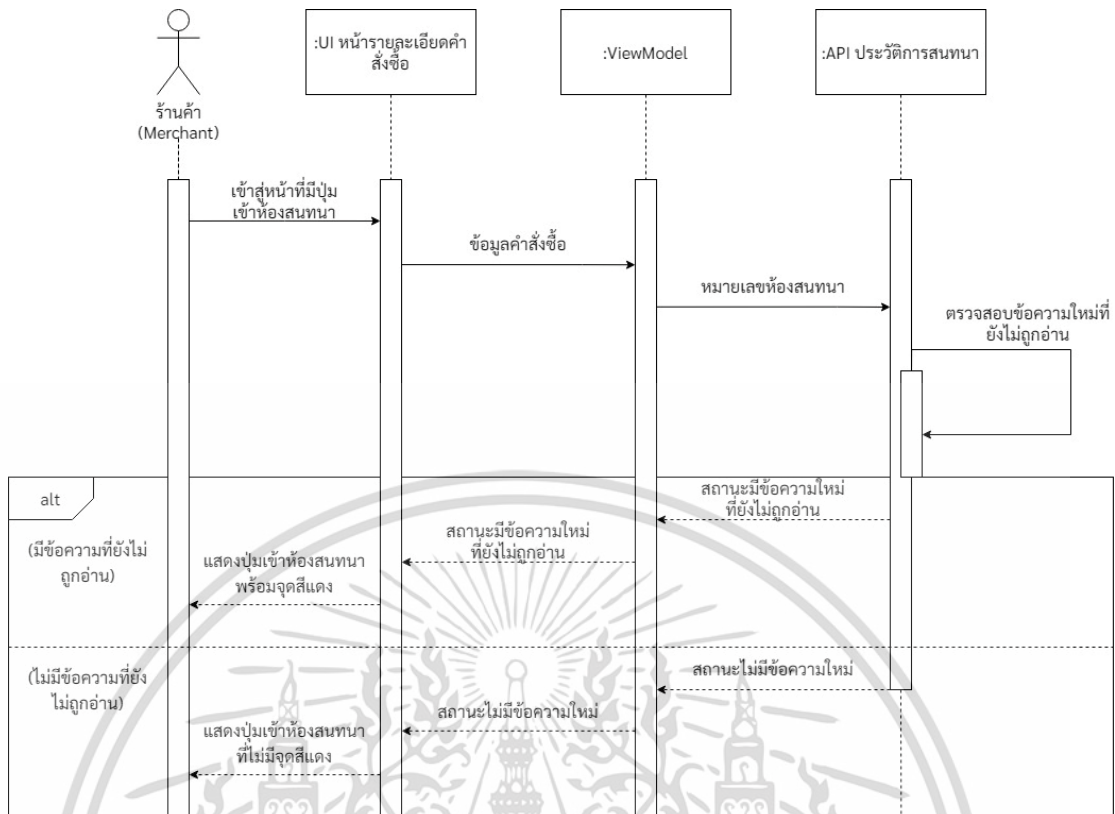
รูปที่ 3.8 แผนภาพ Sequence Diagram ของการระบุสถานะการอ่าน

ผู้ใช้บริการจะสามารถทราบสถานะการอ่านของข้อความที่ส่งไปยังผู้ประกอบการร้านค้าได้ผ่านทางข้อความแสดงสถานะ “อ่านแล้ว” ที่จะขึ้นด้านข้างของข้อความนั้นๆ โดยเมื่อผู้ประกอบการร้านค้าได้อ่านข้อความดังกล่าวด้วยการเข้าสู่ห้องสนทนา ทางแอปพลิเคชันของผู้ประกอบการร้านค้าจะส่งข้อมูลการอ่านข้อความกลับมาผ่าน Socket และ ChatViewModel จะส่งข้อมูลสถานะการอ่านไปยังห้องสนทนาเพื่ออัปเดตสถานะการอ่านของข้อความ



รูปที่ 3.9 แผนภาพ Sequence Diagram ของการแจ้งเตือนข้อความใหม่

ผู้ให้บริการจะได้รับการแจ้งเตือนข้อความใหม่บนอุปกรณ์เมื่อมีข้อความใหม่จากผู้ประกอบการร้านค้าถูกส่งเข้ามาในห้องสนทนา โดยข้อมูลของข้อความดังกล่าวจะถูกส่งจากแอปพลิเคชันสำหรับผู้ประกอบการร้านค้าไปยังเซิร์ฟเวอร์ผ่าน Socket และถูกส่งต่อไปยัง Firebase เพื่อทำการส่งการแจ้งเตือนไปยังอุปกรณ์ของผู้ให้บริการ โดยข้อมูลของข้อความที่ถูกส่งโดยผ่าน Firebase จะเข้าสู่ Push Notification Service ที่ทำการจัดการการแจ้งเตือน โดยจะมีการส่งเสียงแจ้งเตือนและปรากฏข้อความการแจ้งเตือนบนหน้าจอของระบบ ซึ่งการแจ้งเตือนจะประกอบไปด้วยชื่อของผู้ประกอบการร้านค้า และข้อความใหม่ที่ผู้ประกอบการร้านค้าได้ส่งมา นอกจากนี้การแจ้งเตือนข้อความจะถูกจับกลุ่มตามรายการคำสั่งซื้อ กล่าวคือ ถ้ามีคำสั่งซื้อเกิดขึ้นพร้อมกันหลายคำสั่งซื้อ การแจ้งเตือนจะถูกจับกลุ่มการแสดงผลตามคำสั่งซื้อนั้นๆ



รูปที่ 3.10 แผนภาพ Sequence Diagram ของการแสดงปุ่มเข้าสู่ห้องสนทนาพร้อมด้วยจุดสีแดง

เมื่อผู้ใช้บริการเข้าสู่หน้าต่างๆ ของแอปพลิเคชันที่มีปุ่มทางเข้าห้องสนทนาปรากฏอยู่ แอปพลิเคชันจะทำการส่งข้อมูลของคำสั่งซื้อนั้นไปยังเซิร์ฟเวอร์เพื่อทำการตรวจสอบว่าห้องสนทนาของคำสั่งซื้อดังกล่าวมีข้อความใหม่จากผู้ประกอบการร้านค้าที่ยังไม่ได้ถูกอ่านหรือไม่ จากนั้นเซิร์ฟเวอร์จะทำการส่งข้อมูลสถานะข้อความใหม่ของคำสั่งซื้อนั้นกลับมา โดยถ้าหากมีสถานะมีข้อความใหม่เข้ามาและยังไม่ได้ถูกอ่าน ปุ่มทางเข้าห้องสนทนาจะมีการปรากฏจุดสีแดงอยู่ที่มุมด้านขวาบนเพื่อระบุว่าข้อความใหม่เข้ามา แต่ถ้าหากมีสถานะเป็นไม่มีข้อความใหม่เข้ามา จุดสีแดงจะไม่ทำการแสดงผล

บทที่ 4

ผลการวิจัยและอภิปรายผล

ในการพัฒนาคุณลักษณะการสนทนาการสนทนาในแอปพลิเคชันมือถือสำหรับผู้ให้บริการ (Consumer) ของแพลตฟอร์มคลาวด์คิทเช่น โดยดำเนินการพัฒนารวมถึงทำการทดสอบระบบต่างๆ ด้วยภาษา Kotlin บนโปรแกรม Android Studio ซึ่งได้ทำการศึกษาและออกแบบไว้ในบทที่ 3 จะได้ผลลัพธ์ในการพัฒนาออกมาดังนี้

4.1 ส่วนต่อประสานผู้ใช้ (User Interface)

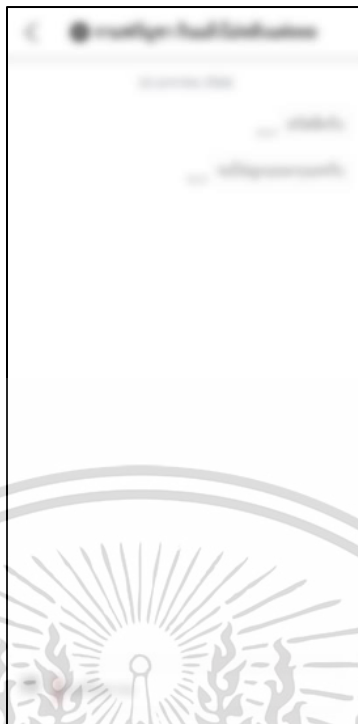
4.1.1 การแสดงประวัติการสนทนา

4.1.1.1 ห้องสนทนากรณีคำสั่งซื้ออยู่ภายในระยะเวลา 24 ชั่วโมง



รูปที่ 4.1 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนา กรณีคำสั่งซื้ออยู่ภายในระยะเวลา 24 ชั่วโมงและยังไม่มีประวัติการสนทนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนา กรณีคำสั่งซื้ออยู่ภายในระยะเวลา 24 ชั่วโมงและมีประวัติการสนทนา

หากคำสั่งซื้ออยู่ภายในระยะเวลา 24 ชั่วโมง เมื่อผู้ใช้บริการกดปุ่มเข้าสู่ห้องสนทนา ผ่านหน้าต่างๆ ในแอปพลิเคชัน ผู้ใช้บริการจะเข้าสู่ห้องสนทนาและสามารถทำการส่งข้อความและรับข้อความจากผู้ประกอบการร้านค้าได้ ซึ่งหน้าห้องสนทนาจะไม่ปรากฏข้อความใดๆ เมื่อเข้าใช้งานในครั้งแรก ดังรูปที่ 4.1 และเมื่อมีการสนทนาภายในห้องสนทนาของคำสั่งซื้อนั้น ห้องสนทนาจะแสดงประวัติการสนทนา โดยจะแสดงเรียงตามเวลาที่ข้อความได้ถูกส่งออกไป โดยข้อความที่ถูกส่งจากผู้ใช้บริการจะแสดงอยู่ทางด้านขวาของหน้าจอห้องสนทนา และข้อความของผู้ประกอบการร้านค้าจะแสดงอยู่ทางด้านซ้ายของหน้าจอห้องสนทนา ดังรูปที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1.2 ห้องสนทนากรณีคำสั่งซื้อเกินระยะเวลา 24 ชั่วโมง



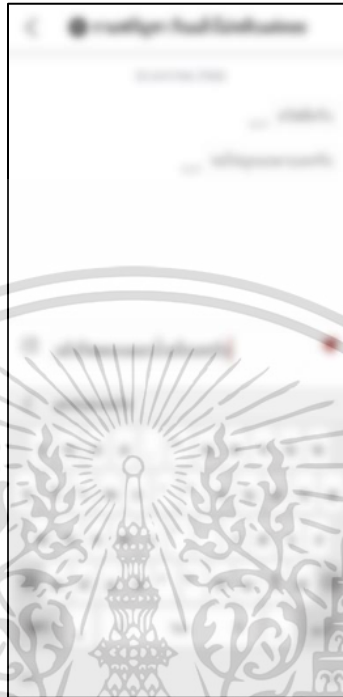
รูปที่ 4.3 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนา กรณีคำสั่งซื้ออยู่เกินระยะเวลา 24 ชั่วโมง

หากคำสั่งซื้อเกินระยะเวลา 24 ชั่วโมง เมื่อผู้ใช้บริการกดปุ่มเข้าห้องสนทนาผ่านหน้าต่างๆ ในแอปพลิเคชัน จะปรากฏหน้า Empty State ซึ่งผู้ใช้บริการจะไม่สามารถส่งข้อความไปยังผู้ประกอบการร้านค้าได้ และไม่สามารถดูประวัติการสนทนาที่ผ่านมาได้ ดังรูปที่ 4.3

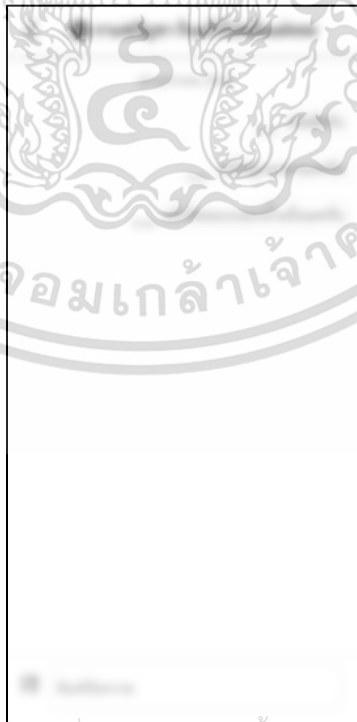
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 การส่งข้อความตัวอักษร

4.1.2.1 กรณีส่งข้อความสำเร็จ



รูปที่ 4.4 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการส่งข้อความตัวอักษร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณี รูปที่ 4.5 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการส่งข้อความตัวอักษรสำเร็จ

เมื่อผู้ใช้บริการต้องการที่จะส่งข้อความตัวอักษรไปยังผู้ประกอบการร้านค้า ผู้ใช้บริการสามารถพิมพ์ข้อความลงในช่องพิมพ์ข้อความ ดังรูปที่ 4.4 และกดปุ่มส่งข้อความเพื่อส่งข้อความนั้นไปยังผู้ประกอบการร้านค้า เมื่อข้อความนั้นถูกส่งได้สำเร็จ จะมีการแสดงเวลาที่ข้อความนั้นได้ถูกส่งทางด้านซ้ายของกล่องข้อความ ดังรูปที่ 4.5

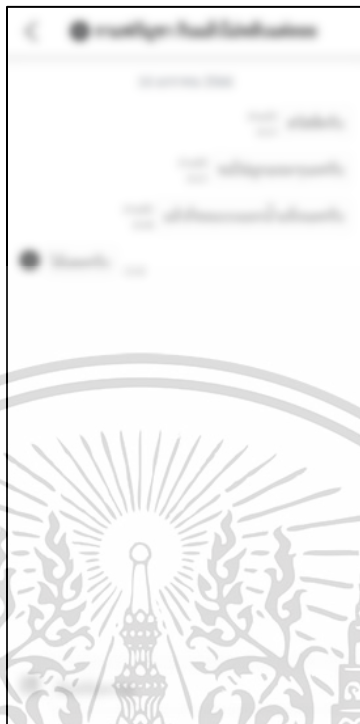
4.1.2.2 กรณีส่งข้อความไม่สำเร็จ



รูปที่ 4.6 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการส่งข้อความตัวอักษรไม่สำเร็จ

เมื่อผู้ใช้บริการต้องการที่จะส่งข้อความตัวอักษรไปยังผู้ประกอบการร้านค้า แล้วข้อความที่ผู้ใช้บริการส่งไปยังผู้ประกอบการร้านค้านั้นส่งได้ไม่สำเร็จ จะมีการแสดงปุ่มส่งใหม่อีกครั้งทางด้านซ้ายของกล่องข้อความ ดังรูปที่ 4.6

4.1.3 การรับข้อความตัวอักษร



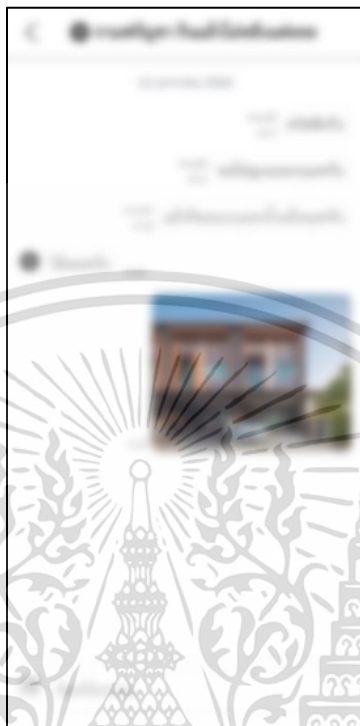
รูปที่ 4.7 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการรับข้อความตัวอักษร

เมื่อผู้ประกอบการร้านค้ามีการส่งข้อความตัวอักษรมายังผู้ใช้บริการ ข้อความที่ส่งมาจะแสดงอยู่ทางด้านซ้ายของหน้าจอห้องสนทนา และแสดงระยะเวลาที่ผู้ประกอบการร้านค้าได้ส่งข้อความที่ด้านขวาของกล่องข้อความที่ผู้ประกอบการร้านค้าได้ส่งมา ดังภาพที่ 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.4 การส่งข้อความรูปภาพ

4.1.4.1 กรณีส่งข้อความสำเร็จ



รูปที่ 4.8 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการส่งข้อความตัวอักษรสำเร็จ

เมื่อผู้ใช้บริการต้องการที่จะส่งข้อความรูปภาพไปยังผู้ประกอบการร้านค้า ผู้ใช้บริการสามารถกดที่ปุ่มเลือกรูปภาพที่อยู่ด้านซ้ายของช่องพิมพ์ข้อความ จากนั้นเลือกรูปภาพที่ต้องการจะส่ง เมื่อข้อความนั้นถูกส่งได้สำเร็จ จะมีการแสดงเวลาที่ข้อความนั้นได้ถูกส่งทางด้านซ้ายของกล่องข้อความ ดังรูปที่ 4.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.4.2 กรณีส่งข้อความไม่สำเร็จ

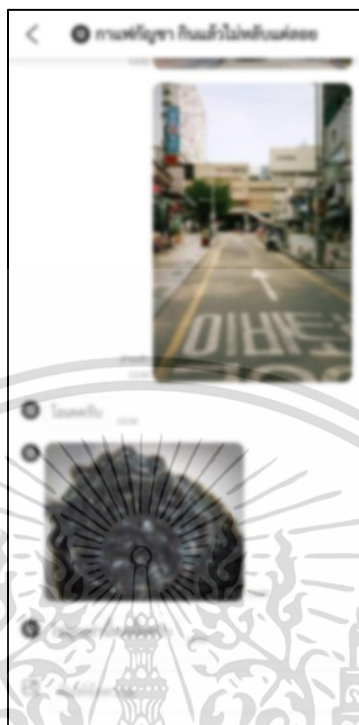


รูปที่ 4.9 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการส่งข้อความรูปภาพไม่สำเร็จ

เมื่อผู้ใช้บริการต้องการที่จะส่งข้อความรูปภาพไปยังผู้ประกอบการร้านค้า แล้วข้อความที่ผู้ใช้บริการส่งไปยังผู้ประกอบการร้านค้านั้นส่งได้ไม่สำเร็จ จะมีการแสดงปุ่มส่งใหม่อีกครั้งทางด้านซ้ายของรูปภาพ ดังรูปที่ 4.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.5 การรับข้อความรูปภาพ

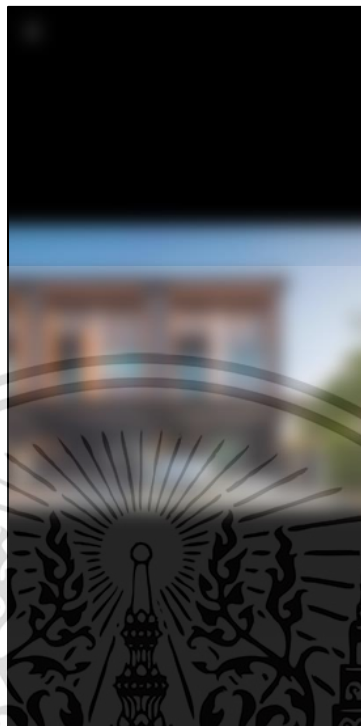


รูปที่ 4.10 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการรับข้อความรูปภาพ

เมื่อผู้ประกอบการร้านค้ามีการส่งข้อความรูปภาพมายังผู้ใช้บริการ รูปภาพที่ส่งมาจะแสดงอยู่ทางด้านซ้ายของหน้าจอห้องสนทนา และแสดงระยะเวลาที่ผู้ประกอบการร้านค้าได้ส่งรูปภาพที่ด้านขวาของรูปภาพที่ผู้ประกอบการร้านค้าได้ส่งมา ดังรูปที่ 4.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.6 การพรีวิวแสดงรูปภาพเต็มหน้าจอ

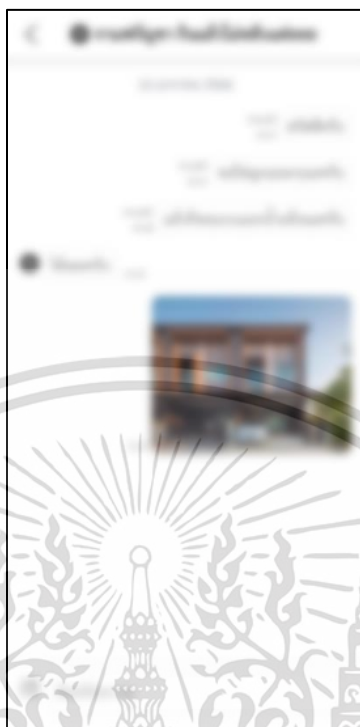


รูปที่ 4.11 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อมีการพรีวิวแสดงรูปภาพเต็มหน้าจอ

เมื่อผู้ใช้บริการหรือผู้ประกอบการร้านค้ามีการส่งข้อความรูปภาพ ผู้ใช้บริการสามารถกดที่รูปภาพเพื่อให้รูปภาพนั้นมีการแสดงภาพเต็มหน้าจอได้ ดังรูปที่ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.7 การดูสถานะการอ่าน



รูปที่ 4.12 ตัวอย่างส่วนต่อประสานผู้ใช้ของห้องสนทนาเมื่อผู้ประกอบการร้านค้าได้อ่านข้อความของผู้ใช้บริการแล้ว

เมื่อผู้ประกอบการร้านค้าได้ทำการเปิดอ่านข้อความที่ผู้ใช้บริการได้ส่งไปแล้ว ข้อความที่ผู้ใช้บริการส่งไปยังผู้ประกอบการร้านค้าจะแสดงสถานะ “อ่านแล้ว” ขึ้นที่ด้านซ้ายของกล่องข้อความนั้นๆ ดังรูปที่ 4.12

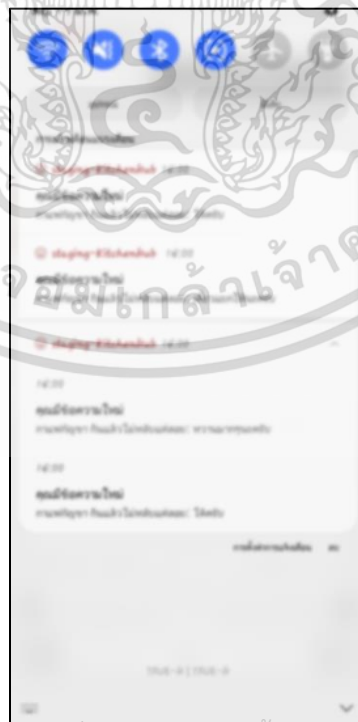
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.8 การแจ้งเตือนข้อความใหม่

4.1.8.1 การแจ้งเตือนข้อความใหม่ผ่านระบบปฏิบัติการแอนดรอยด์



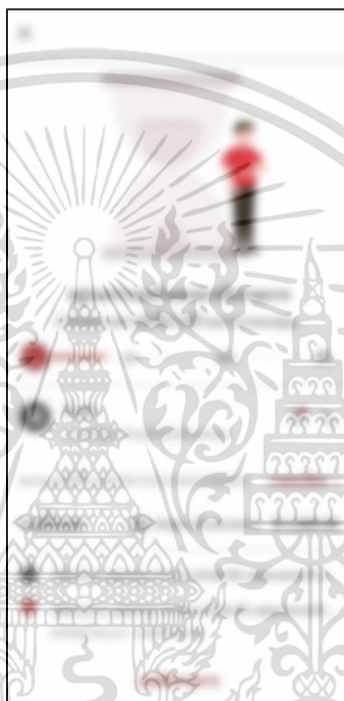
รูปที่ 4.13 ตัวอย่างส่วนต่อประสานผู้ใช้ของระบบเมื่อมีการแจ้งเตือนข้อความใหม่



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้นกรณีที่มีการขออนุญาตจากเจ้าของลิขสิทธิ์ที่เกี่ยวข้อง

เมื่อมีข้อความใหม่จากผู้ประกอบการร้านค้าถูกส่งเข้ามาในห้องสนทนา จะมีการแจ้งเตือนข้อความใหม่ผ่านส่วนต่อประสานผู้ใช้ของระบบ ดังรูปที่ 4.13 และเมื่อมีข้อความใหม่เข้ามาในห้องสนทนาของคำสั่งซื้อเดียวกัน ระบบจะทำการจัดกลุ่มการแจ้งเตือนให้เป็นกลุ่มเดียวกัน แต่ถ้าหากข้อความที่เข้ามาใหม่นั้นเป็นข้อความของคำสั่งซื้ออื่น ระบบจะทำการแยกการแจ้งเตือนข้อความใหม่ตามคำสั่งซื้อนั้นๆ ดังรูปที่ 4.14

4.1.8.2 การแจ้งเตือนข้อความใหม่ภายในแอปพลิเคชัน



รูปที่ 4.15 ตัวอย่างส่วนต่อประสานผู้ใช้ของหน้าต่างติดตามสถานะคำสั่งซื้อเมื่อมีการแจ้งเตือนข้อความใหม่

เมื่อมีข้อความใหม่จากผู้ประกอบการร้านค้าถูกส่งเข้ามาในห้องสนทนา ปุ่มเข้าสู่ห้องสนทนาที่หน้าต่างๆ จะมีการแสดงจุดสีแดงที่ด้านบนขวาของปุ่มเข้าสู่ห้องสนทนา เพื่อเป็นการแจ้งเตือนว่าคำสั่งซื้อนี้มีข้อความใหม่ที่ผู้ใช้บริการยังไม่ได้เปิดอ่าน ดังรูปที่ 4.15

4.2 การทดสอบระบบซอฟต์แวร์ (Software Testing)

หลังจากการพัฒนาคุณลักษณะการสนทนาในแอปพลิเคชันมือถือสำหรับผู้ให้บริการ (Consumer) ของแพลตฟอร์มคลาวด์คิดิเช่นแล้ว ในขั้นตอนถัดไปจะเป็นการทดสอบระบบเพื่อตรวจสอบความถูกต้องในการทำงานของชุดคำสั่งต่างๆ ที่ถูกเขียนขึ้น โดยทางผู้จัดทำได้เลือกทำการทดสอบระบบในระดับของ Small Test หรือ Unit Test ที่จะเป็นการเขียนชุดคำสั่งสำหรับการทดสอบระบบแบบอัตโนมัติกับคลาสใดคลาสหนึ่งที่ต้องการจะทดสอบ โดยชุดคำสั่งดังกล่าวจะทำการสร้างข้อมูลจำลองและส่งข้อมูลเข้าไปยังคลาสนั้นๆ โดยตรง และเปรียบเทียบผลลัพธ์ที่ได้จากการทำงานกับผลลัพธ์ที่คาดหวังของระบบว่ามีค่าถูกต้องหรือไม่

4.2.1 การทดสอบระบบของคลาส ViewModel

การทดสอบระบบในคลาส ViewModel ของแอปพลิเคชันมีความสำคัญเป็นอย่างมาก เนื่องจากภายในคลาส ViewModel มีชุดคำสั่งที่ทำหน้าที่เกี่ยวกับ Business Logic ซึ่งมีความสำคัญต่อการทำงานคุณลักษณะการสนทนา โดยจะมีการทดสอบการทำงานในกรณีต่างๆ ดังต่อไปนี้

ตารางที่ 4.1 กรณีการทดสอบ (Test Case) ของคลาส ViewModel

Test Scenario	Test Case Objective	Test Step	Test Data	Expected Result	Status (Pass/Fail)
การโหลดประวัติการสนทนา	การโหลดประวัติการสนทนาโดยไม่มีข้อผิดพลาด	1) ตั้งค่าหมายเลขห้องสนทนาเป็น 1234 2) เรียกฟังก์ชันที่ทำการขอข้อมูลประวัติการสนทนา	1) หมายเลขห้องสนทนา 1234	มีข้อมูลประวัติการสนทนาในอาเรย์	Pass
การดึงข้อผิดพลาดจากการโหลดประวัติการสนทนา	การโหลดประวัติการสนทนาโดยไม่มีข้อผิดพลาด	1) ตั้งค่าหมายเลขห้องสนทนาเป็น -1234 2) เรียกฟังก์ชันที่ทำการขอประวัติการสนทนา	1) หมายเลขห้องสนทนา - 1234	ค่า Error ที่ระบุสาเหตุของความผิดพลาดมาจากการโหลดประวัติการสนทนา	Pass

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 การทดสอบระบบของคลาส Adapter

การทดสอบระบบในคลาส Adapter ของแอปพลิเคชันมีความสำคัญเป็นอย่างมาก เนื่องจากภายในคลาส Adapter มีชุดคำสั่งที่ทำหน้าที่เกี่ยวกับ View Logic ที่เป็นคำสั่งหลักในการแสดงผลของคุณลักษณะการสนทนา โดยจะมีการทดสอบการทำงานในกรณีต่างๆ ดังต่อไปนี้

ตารางที่ 4.2 กรณีการทดสอบ (Test Case) ของคลาส ViewModel

Test Scenario	Test Case Objective	Test Step	Test Data	Expected Result	Status
การแสดงผลประวัติการสนทนา	การนำข้อมูลประวัติการสนทนา มาแสดงผล โดยที่ข้อความส่งในวันเดียวกัน	1) บรรจุข้อมูลข้อความที่ถูกส่งในวันเดียวกัน 2 ข้อความลงในอาเรย์ 2) เรียกใช้ฟังก์ชันที่นำอาเรย์ที่บรรจุข้อมูลข้อความไปแสดงผลข้อมูลประวัติการสนทนา	1) ข้อมูลข้อความที่ถูกส่งในวันเดียวกัน 2 ข้อความ	1) ข้อมูลประวัติสนทนาที่ประกอบด้วยข้อความทั้งสองพร้อมด้วยส่วนระบุวันที่ส่งข้อความด้านบน ข้อความทั้งสอง	Pass
การแสดงผลประวัติการสนทนา	การนำข้อมูลประวัติการสนทนา มาแสดงผล โดยที่ข้อความส่งคนละวันกัน	1) บรรจุข้อมูลข้อความที่ถูกส่งในวันเดียวกัน 2 ข้อความลงในอาเรย์ 2) เรียกใช้ฟังก์ชันที่นำอาเรย์ที่บรรจุข้อมูลข้อความไปแสดงผลข้อมูลประวัติการสนทนา	1) ข้อมูลข้อความที่ถูกส่งในวันแรก 1) ข้อมูลข้อความที่ถูกส่งในวันที่สอง	1) ข้อมูลประวัติสนทนาที่ประกอบด้วยข้อความทั้งสองพร้อมด้วยส่วนระบุวันที่ด้านบนข้อความที่ส่งในวันแรก และส่วนระบุวันที่ด้านบนข้อความที่ส่งในวันที่สอง	Pass
การแสดงผลข้อความจากผู้ประกอบการร้านค้า	การนำข้อความที่ส่งจากผู้ประกอบการร้านค้า มาแสดงผล โดยข้อความถูกส่งในวันเดียวกัน	1) เรียกใช้ฟังก์ชันในการรับข้อความจากผู้ประกอบการร้านค้า	1) ข้อมูลข้อความจากผู้ประกอบการร้านค้าส่งมา 2 ข้อความในวันเดียวกัน	1) ข้อมูลข้อความที่ประกอบด้วยข้อความทั้งสองจากผู้ประกอบการร้านค้า พร้อมด้วยส่วนระบุ วันที่ส่งข้อความด้านบน ข้อความทั้งสอง	Pass

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ทำซ้ำโดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Test Scenario	Test Case Objective	Test Step	Test Data	Expected Result	Status
การแสดงผลข้อความจากผู้ประกอบการร้านค้า	การนำข้อความที่ส่งจากผู้ประกอบการร้านค้ามาแสดงผล โดยข้อความถูกคนละวันกัน	1) เรียกใช้ฟังก์ชันในการรับข้อความจากผู้ประกอบการร้านค้า	1) ข้อมูลข้อความที่ถูกส่งในวันแรก 2) ข้อมูลข้อความที่ถูกส่งในวันที่สอง	1) ข้อมูลข้อความที่ประกอบด้วยข้อความทั้งสองพร้อมด้วยส่วนระบุวันที่ส่งข้อความด้านบนข้อความที่ส่งในวันแรก และส่วนระบุวันที่ด้านบนข้อความที่ส่งในวันที่สอง	Pass
การแสดงผลข้อความจากผู้ให้บริการ	การนำข้อความที่ส่งจากผู้ให้บริการมาแสดงผล โดยข้อความถูกส่งในวันเดียวกัน	1) เรียกใช้ฟังก์ชันในการรับข้อความจากผู้ให้บริการ	1) ข้อมูลข้อความที่ผู้ใช้บริการส่งมา 2 ข้อความในวันเดียวกัน	1) ข้อมูลข้อความที่ประกอบด้วยข้อความทั้งสองจากผู้ให้บริการ พร้อมด้วยส่วนระบุวันที่ส่งข้อความด้านบนข้อความทั้งสอง	Pass
การแสดงผลข้อความจากผู้ให้บริการ	การนำข้อความที่ส่งจากผู้ให้บริการมาแสดงผล โดยข้อความถูกส่งคนละวันกัน	1) เรียกใช้ฟังก์ชันในการรับข้อความจากผู้ให้บริการ	1) ข้อมูลข้อความที่ถูกส่งในวันแรก 2) ข้อมูลข้อความที่ถูกส่งในวันที่สอง	1) ข้อมูลข้อความที่ประกอบด้วยข้อความทั้งสองพร้อมด้วยส่วนระบุวันที่ส่งข้อความด้านบนข้อความที่ส่งในวันแรก และส่วนระบุวันที่ด้านบนข้อความที่ส่งในวันที่สอง	Pass
การแสดงผลเวลาที่ส่งข้อความ	การแสดงผลเวลาที่ส่งข้อความด้านข้างกล่องข้อความ	1) เรียกใช้ฟังก์ชันที่ใช้ในการรับข้อความ ที่ยังไม่ระบุเวลาการส่งที่สำเร็จจากผู้ให้บริการ 2) เรียกใช้ฟังก์ชันที่ใช้ในการรับข้อความ ที่ระบุเวลาการส่งที่สำเร็จแล้วจากผู้ให้บริการ	1) เวลาที่ส่งข้อความสำเร็จ	1) เวลาที่ส่งข้อความแสดงที่ด้านข้างของกล่องข้อความ	Pass

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Test Scenario	Test Case Objective	Test Step	Test Data	Expected Result	Status
การแสดงผลสถานะการอ่าน	การอัปเดตสถานะการอ่านของข้อความที่ถูกส่งจากผู้ให้บริการ	1) เรียกใช้ฟังก์ชันที่ใช้ในการรับข้อความที่ยังไม่ระบุเวลาการส่งที่สำเร็จจากผู้ให้บริการ 2) เรียกใช้ฟังก์ชันที่ใช้ในการรับข้อความที่ระบุเวลาการส่งที่สำเร็จแล้วจากผู้ให้บริการ 3) เรียกใช้ฟังก์ชันที่ใช้ในอัปเดตสถานะการอ่านข้อความที่ส่งจากผู้ให้บริการทั้งหมด	1) เวลาที่ส่งข้อความ	1) สถานะการอ่านแสดงที่ด้านข้างของกล่องข้อความ	Pass
การแสดงผลปุ่มส่งข้อความใหม่อีกครั้ง	การแสดงผลปุ่มส่งข้อความใหม่อีกครั้งเมื่อข้อความของผู้ให้บริการส่งไม่สำเร็จ	1) เรียกใช้ฟังก์ชันที่ใช้ในการรับข้อความที่ยังไม่ระบุเวลาการส่งที่สำเร็จจากผู้ให้บริการ 2) เรียกใช้ฟังก์ชันที่ใช้ในการรับข้อความจากผู้ให้บริการร้านค้าที่ส่งไม่สำเร็จ เพื่อนำมาแสดงผลปุ่มส่งข้อความใหม่อีกครั้ง	1) ข้อความที่ยังไม่ระบุเวลาการส่งที่สำเร็จจากผู้ให้บริการ 2) ข้อมูลข้อความเดิมจากผู้ให้บริการที่ถูกระบุสถานะว่าข้อความดังกล่าวส่งไม่สำเร็จ	1) ข้อมูลข้อความจากผู้ให้บริการที่ถูกระบุสถานะว่าข้อความดังกล่าวส่งไม่สำเร็จ	Pass
การแสดงผลปุ่มส่งข้อความใหม่อีกครั้ง	การแสดงผลข้อความที่ถูกส่งใหม่อีกครั้ง เมื่อข้อความที่ส่งไม่สำเร็จถูกกดปุ่มส่งใหม่อีกครั้ง	1) เรียกใช้ฟังก์ชันที่ใช้ในการรับข้อความที่ยังไม่ระบุเวลาการส่งที่สำเร็จจากผู้ให้บริการ 2) เรียกใช้ฟังก์ชันที่ใช้ในการรับข้อความจากผู้ให้บริการร้านค้าที่ส่งไม่สำเร็จ เพื่อนำมาแสดงผลปุ่มส่งข้อความใหม่อีกครั้ง 3) เรียกใช้ฟังก์ชันที่ใช้ส่งข้อความใหม่อีกครั้ง และลบข้อความเมื่อข้อความที่ส่งไม่สำเร็จ	1) ข้อความที่ยังไม่ระบุเวลาการส่งที่สำเร็จจากผู้ให้บริการ 2) ข้อมูลข้อความเดิมจากผู้ให้บริการที่ถูกระบุสถานะว่าข้อความดังกล่าวส่งไม่สำเร็จ	1) ข้อมูลข้อความจากผู้ให้บริการที่ถูกระบุสถานะว่าข้อความดังกล่าวส่งไม่สำเร็จ	Pass

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

จากการดำเนินงานในการพัฒนาคุณลักษณะการสนทนาในแอปพลิเคชันมือถือสำหรับผู้ให้บริการ (Consumer) ของแพลตฟอร์มคลาวด์คิดเช่น สามารถดำเนินการมาได้จนสำเร็จลุล่วง และได้รับการทดสอบว่าสามารถใช้งานได้จริงและตรงตามวัตถุประสงค์ที่ได้วางไว้ คือ ช่วยในการเพิ่มช่องทางติดต่อสื่อสารระหว่างผู้ให้บริการกับร้านค้าให้สะดวกมากยิ่งขึ้น และเพิ่มความถูกต้องของรายละเอียดต่างๆ ในรายการอาหารที่ผู้ให้บริการได้สั่งกับทางร้านค้า ซึ่งคุณลักษณะการสนทนาบนแอปพลิเคชันสำหรับผู้ให้บริการนั้นเป็นเพียงส่วนหนึ่งของระบบสนทนาของแพลตฟอร์มคลาวด์คิดเช่น โดยจะต้องมีการทำงานร่วมกับอีกสองส่วนสำคัญ คือ ระบบการสนทนาที่ทำงานอยู่บนระบบหลังบ้านบนเซิร์ฟเวอร์ และคุณลักษณะการสนทนาบนแอปพลิเคชันสำหรับผู้ประกอบการร้านค้า ทั้งนี้ระบบการสนทนาดังกล่าวนั้นได้มีการนำไปใช้งานจริงบนแอปพลิเคชัน โดยคุณลักษณะการสนทนาได้ถูกเพิ่มเข้าไปในแอปพลิเคชันสำหรับผู้ให้บริการและแอปพลิเคชันสำหรับผู้ประกอบการร้านค้าเป็นที่เรียบร้อยแล้ว โดยทางผู้ให้บริการและผู้ประกอบการร้านค้าสามารถใช้คุณลักษณะการสนทนาได้ทันทีเมื่อมีการอัปเดตแอปพลิเคชันเป็นเวอร์ชันปัจจุบัน

5.2 ข้อเสนอแนะ

ในการพัฒนาคุณลักษณะการสนทนาบนแอปพลิเคชันสำหรับผู้ให้บริการ ทางผู้จัดทำได้สังเกตเห็นจุดที่ควรนำไปปรับปรุงและพัฒนาต่อไปในอนาคต โดยทางผู้จัดทำได้สรุปเป็นข้อเสนอแนะดังต่อไปนี้

1) เนื่องจากทางผู้จัดทำยังขาดความรู้และประสบการณ์ในการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ ส่งผลให้การพัฒนาคุณลักษณะการสนทนาถูกพัฒนาโดยชุดคำสั่งและรูปแบบการทำงานพื้นฐานทั่วไป ทำให้ยังมีช่องว่างในการพัฒนาอีกมาก ซึ่งสามารถนำไปปรับปรุงประสิทธิภาพการทำงานได้โดยการใช้ชุดคำสั่งขั้นสูงและรูปแบบการทำงานที่เหมาะสมกับการพัฒนาคุณลักษณะการสนทนามากกว่าและทำให้เกิดประสิทธิภาพการทำงานที่ดีมากยิ่งขึ้น

2) เนื่องจากทางผู้จัดทำยังขาดความรู้และประสบการณ์ในการทำการทดสอบแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ ส่งผลให้ไม่มีการทำการทดสอบอัตโนมัติแบบ Unit Test กับส่วนการเอกสารนี้ทำงานต่าง ๆ และส่วนอื่นของแอปพลิเคชัน ในระหว่างการพัฒนาคุณลักษณะการสนทนา เมื่อการไม่มีการพัฒนาใกล้จะเสร็จสิ้นลง ทางผู้จัดทำจึงได้พบกับข้อผิดพลาดจำนวนมากในการทำงาน ทำให้ผู้จัดทำ

ต้องรีบทำการแก้ไขข้อผิดพลาดเหล่านั้นด้วยความเร่งรีบ ซึ่งถ้าหากมีการทดสอบอัตโนมัติแบบ Unit Test เป็นระยะระหว่างการพัฒนา จะช่วยให้ทางผู้จัดทำสามารถพบจุดที่ผิดพลาดได้ตั้งแต่เนิ่น ๆ และทำการแก้ไขได้อย่างทันท่วงที เพื่อไม่ให้เกิดความยากและซับซ้อนในการพัฒนาซอฟต์แวร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] กองเทคโนโลยีสารสนเทศ การประปาส่วนภูมิภาคเขต7. 2559. Agile Method: วิธีการพัฒนาซอฟต์แวร์สำหรับปัจจุบันและอนาคต. [Online]. เข้าถึงได้จาก <https://reg7.pwa.co.th/kmr7/?p=453>
- [2] ณัฐติ แซ่ลื้อ. 2560. MVC MVP MVVM คืออะไร และต่างกันอย่างไร. [Online]. เข้าถึงได้จาก <https://medium.com/@leelorz6/mvc-mvp-mvvm-คืออะไร-และต่างกันอย่างไร-ca16a19631dc>
- [3] ไพจิตร สุขสมบุรณ์. 2563. การออกแบบซอฟต์แวร์. [Online]. เข้าถึงได้จาก <http://paijit.lpru.ac.th/cgi-bin/S5672601/SoftEng8.pdf>
- [4] วชิรชาติ สวัสดิ์วัฒน์ ณ อยุธยา. 2563. Kotlin ภาษาแห่งอนาคตของนักพัฒนา Android. [Online]. เข้าถึงได้จาก <https://blog.skooldio.com/kotlin/>
- [5] วรนาถ ประทุมมณี. 2563. ทำไมภาษา Kotlin ถึงเป็นภาษาหลักในการพัฒนา Android? และ Kotlin จะมาแทนที่ Java หรือไม่?. [Online]. เข้าถึงได้จาก <https://blog.skooldio.com/kotlin-vs-java-android/>
- [6] สกกุล มณฑา. 2562. REST กับ RESTful API ต่างกันนะรู้ยัง. [Online]. เข้าถึงได้จาก <https://iamgique.medium.com/restful-api-กับ-rest-api-ต่างกันนะรู้ยัง-2c70c42990e3>
- [7] สมเกียรติ กิจวงศ์วัฒน์. 2564. มาเขียน Unit Test บน Android Studio กัน. เข้าถึงได้จาก <https://akexorcist.dev/unit-test-in-android-studio/>
- [8] สมเกียรติ กิจวงศ์วัฒน์. 2563. เรื่องที่ควรรู้เกี่ยวกับการเขียนเทสบนแอนดรอยด์. เข้าถึงได้จาก <https://akexorcist.dev/unit-test-in-android-studio/>
- [9] สมเกียรติ กิจวงศ์วัฒน์. 2563. ทำไม App Architecture บนแอนดรอยด์ในยุคนี้ถึงนิยมใช้ Architecture Components?. เข้าถึงได้จาก <https://akexorcist.dev/unit-test-in-android-studio/>
- [10] Google Developer. 2022. The activity lifecycle. [Online]. <https://developer.android.com/guide/components/activities/activity-lifecycle>

เอกสารนี้เป็นเอกสารของมหาวิทยาลัยเทคโนโลยีสุรนารีสงวนลิขสิทธิ์สงวนไว้เพื่อใช้ในการเรียนการสอนและการวิจัยเท่านั้น เมื่อผู้รู้เห็นหาประโยชน์อื่นใดโดยไม่จ่ายค่า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [11] Google Developer. 2023. ViewModel overview. [Online].
<https://developer.android.com/topic/libraries/architecture/viewmodel>
- [12] Risue mishra. 2022. MVVM (Model View ViewModel) Architecture Pattern in Android. [Online]. <https://www.geeksforgeeks.org/mvvm-model-view-viewmodel-architecture-pattern-in-android/>
- [13] Socket.IO. 2022. Introduction. [Online]. <https://socket.io/docs/v4/>
- [14] Wutthichai Eaberb. 2022. สรุปเรื่อง Activity Lifecycle ของการพัฒนา Android app. [Online]. เข้าถึงได้จาก <https://blog.skooldio.com/activity-lifecycle-android/>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



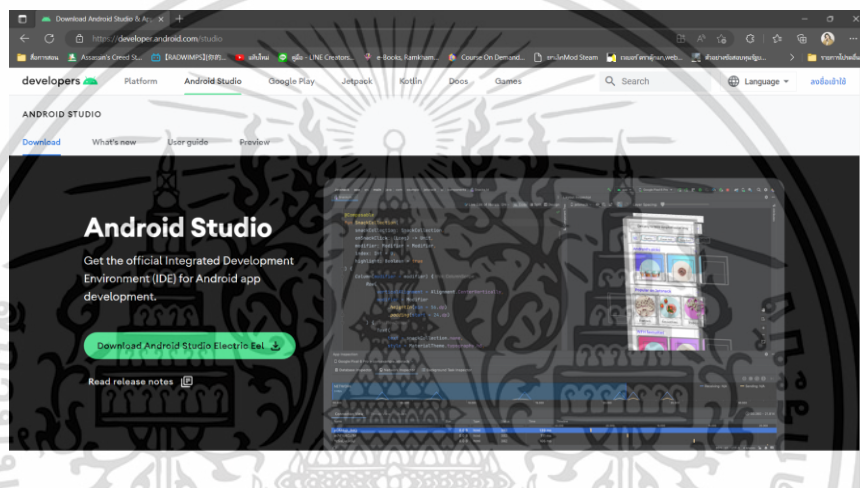
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

การติดตั้งโปรแกรม Android Studio

ก.1 การดาวน์โหลดและติดตั้งโปรแกรม Android Studio

- 1) ในปัจจุบันนี้โปรแกรม Android Studio สามารถถูกดาวน์โหลดและติดตั้งโปรแกรมได้อย่างสะดวก โดยเริ่มต้นด้วยการไปยังเว็บไซต์อย่างเป็นทางการสำหรับการดาวน์โหลด Android Studio ที่เว็บไซต์ <https://developer.android.com/studio> ดังรูปที่ ก.1



รูปที่ ก.1 หน้าเว็บไซต์สำหรับการดาวน์โหลด Android Studio

- 2) กดดาวน์โหลดโปรแกรม Android Studio
- 3) เมื่อเปิดไฟล์ติดตั้งโปรแกรมขึ้นมา จะแสดงหน้าต่างให้กด install เพื่อทำการติดตั้งดังรูปที่ ก.2



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอญูญาติให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีก

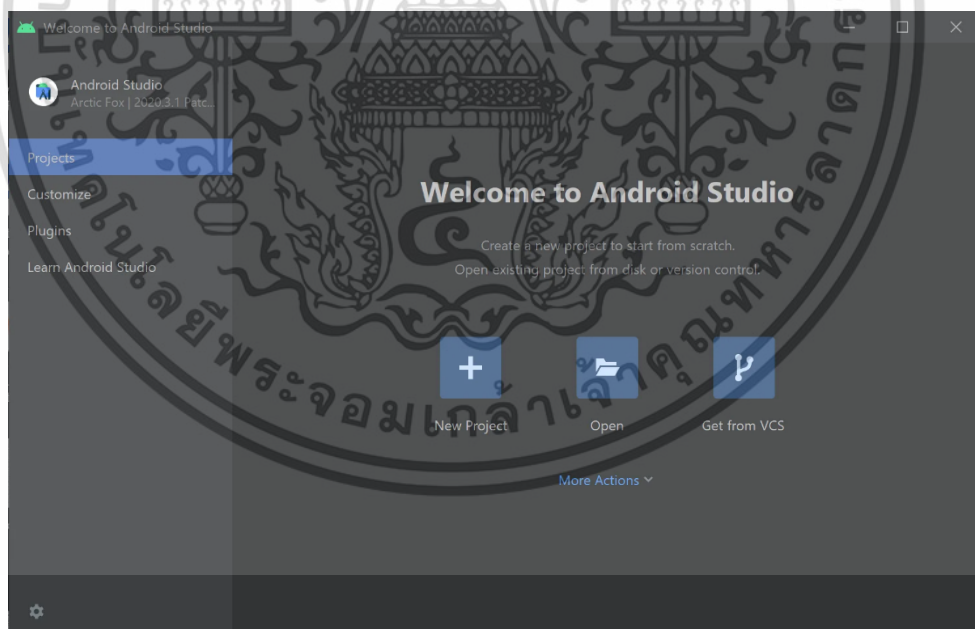
รูปที่ ก.2 หน้าต่างเริ่มต้นการติดตั้งโปรแกรม Android Studio

- 4) เมื่อติดตั้งสำเร็จจะปรากฏปุ่ม Finish เพื่อปิดหน้าต่างการติดตั้งและเริ่มการทำงานของโปรแกรม Android Studio ดังรูปที่ ก.3



รูปที่ ก.3 หน้าต่างเมื่อเสร็จสิ้นการติดตั้งโปรแกรม Android Studio

- 5) เมื่อติดตั้งเสร็จสิ้น จะสามารถเปิดโปรแกรม Android Studio เพื่อสร้าง Project ต่าง ๆ ได้ ดังที่ปรากฏหน้าต่างเริ่มต้นการทำงานของโปรแกรม Android Studio ในรูปที่ ก.4



รูปที่ ก.4 หน้าต่างเริ่มต้นการทำงานของโปรแกรม Android Studio

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



งานทะเบียนคณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

คำรับรองเล่มโครงการพิเศษ/ปัญหาพิเศษ/สหกิจศึกษา

วันที่ 23 เดือน กุมภาพันธ์ พ.ศ. 2566

ข้าพเจ้า นายธนวิษณุ อยู่พัก รหัสประจำตัว 62050173 นักศึกษาหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชา
วิทยาการคอมพิวเตอร์ ภาควิชาวิทยาการคอมพิวเตอร์ ขอรับรองว่าโครงการพิเศษ/ปัญหาพิเศษ/สหกิจศึกษา เรื่อง
ชื่อภาษาไทย การพัฒนาคุณลักษณะการสนทนาในแอปพลิเคชันมือถือสำหรับผู้ให้บริการ (Consumer) ของ
แพลตฟอร์มคลาวด์คิทเชน ชื่อภาษาอังกฤษ CHAT FEATURES FOR CONSUMER MOBILE APPLICATION
DEVELOPMENT OF CLOUD KITCHEN PLATFORM ปีการศึกษา 2565 เป็นผลงานวิจัยที่มีได้คัดลอกหรือละเมิด
ลิขสิทธิ์ของผู้อื่นและได้ผ่านการตรวจสอบความซ้ำซ้อนเรียบร้อยแล้ว และได้แนบเอกสารการตรวจสอบการลอกเลียน
งานวรรณกรรมที่ตรวจสอบจากเล่มโครงการพิเศษ/ปัญหาพิเศษ/สหกิจศึกษาลบสมบูรณ์แล้ว โปรแกรมอักขรวิสุทธิ
0.33%

ลงชื่อ ธนวิษณุ อยู่พัก

(นายธนวิษณุ อยู่พัก)

นักศึกษา

ข้าพเจ้า ผศ.ดร.อินทรภาพ อรัณยนาถ อาจารย์ที่ปรึกษาสหกิจศึกษา ได้ตรวจสอบสหกิจศึกษาของนักศึกษา
ข้างต้นแล้ว ขอรับรองว่าเป็นผลงานวิจัยของนักศึกษาจริงและมีเนื้อหาสมบูรณ์ จึงลงชื่อไว้เป็นหลักฐาน

ลงชื่อ อินทรภาพ อรัณย.ท

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำรับรองเล่มสหกิจศึกษาโดยสถานประกอบการ

วันที่ 10 เดือน มีนาคม พ.ศ. 2566

ข้าพเจ้า นายทรงวุฒิ ชัยประเสริฐศรี ตำแหน่ง Head of Mobile Developer เป็นตัวแทนของสถานประกอบการ บริษัท ฟูลทิม จำกัด ขอรับรองว่า ทางสถานประกอบการได้ตรวจสอบเล่มสหกิจศึกษา เรื่อง การพัฒนาคุณลักษณะการสนทนาในแอปพลิเคชันมือถือสำหรับผู้ให้บริการของแพลตฟอร์มคลาวด์คิทเช่น ของนักศึกษาชื่อ นายธนวิชญ์ อยู่พุก ซึ่งเป็นนักศึกษาภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เรียบร้อยแล้ว และไม่มีส่วนหนึ่งส่วนใดในเล่มสหกิจศึกษาที่มีข้อมูลอ่อนไหว และ/หรือ ข้อมูลอันเป็นความลับอันจะก่อให้เกิดความเสียหายต่อสถานประกอบการ รวมทั้งอนุญาตให้สามารถเผยแพร่ต่อสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังได้ จึงลงชื่อไว้เป็นหลักฐาน

ลงชื่อ **ทรงวุฒิ ชัยประเสริฐศรี**

(ทรงวุฒิ ชัยประเสริฐศรี)

ตัวแทนสถานประกอบการ

ข้าพเจ้า ผศ.ดร.อินทราพร อรัณยะนาค อาจารย์ที่ปรึกษาสหกิจศึกษา ได้ตรวจสอบเล่มสหกิจศึกษาแล้ว และรับทราบว่าสถานประกอบการดำเนินการตรวจสอบเล่มสหกิจศึกษาแล้ว จึงลงชื่อไว้เป็นหลักฐาน

ลงชื่อ **อินทราพร อรัณยะนาค**

(อินทราพร อรัณยะนาค)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้