

ปริญญา尼พนธ์

แอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกซ่อมจำออนไลน์

Car Maintenance and Memos Online Application

นาย Jarvis Dech

ชาย gwad

นางสาวชนกมล

อินทร์สุริยวงศ์

นายสิทธินนท์

ตดิยะจินดา

ปริญญา尼พนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ครุศาสตร์อุตสาหกรรมบัณฑิต (ครุศาสตร์วิศวกรรม)

สาขาวิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรมและเทคโนโลยี

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2563

ปริญญา尼พนธ์

แอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์

Car Maintenance and Memos Online Application

นาย Jarvis Dech

ชาຍกวด

นางสาวชนกมล

อินทรสุริยวงศ์

นายสิทธินนท์

ตดิยะจินดา

ปริญญา尼พนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ครุศาสตร์อุตสาหกรรมบัณฑิต (ครุศาสตร์วิศวกรรม)

สาขาวิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรมและเทคโนโลยี

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2563

ปริญญา妮พนธ์

เรื่อง แอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์

Car Maintenance and Memos Online Application

วัตถุประสงค์

- เพื่อศึกษาการสร้างและพัฒนาแอปพลิเคชันบนสมาร์ทโฟนโดยใช้ React Native
- เพื่อออกแบบระบบโครงสร้างแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์
- เพื่อสร้างแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์โดยใช้ React Native
- เพื่อทดสอบข้อผิดพลาดการทำงานและระบบของแอปพลิเคชัน
- เพื่อนำแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์ไปใช้งานได้จริง

ประโยชน์ที่คาดว่าจะได้รับ

- เข้าใจหลักการการสร้างและพัฒนาแอปพลิเคชันบนสมาร์ทโฟน โดยใช้ React Native
- ได้ออกแบบระบบโครงสร้างแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์
- ได้สร้างแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์จำได้โดยใช้ React Native
- ได้ทดสอบข้อผิดพลาดการทำงานและระบบของแอปพลิเคชันได้
- ได้นำแอปพลิเคชันแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์ไปใช้งานได้จริง

ชื่อหัวข้อ	แอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์	
นักศึกษา	นายJarut Dech	ชาญกาด
	นางสาวอรอนงค์ อินทรสุริยวงศ์	
	นายสิทธินนท์ ตติยะจินดา	
อาจารย์ที่ปรึกษา	พศ.สุระชัย	พิมพ์สาลี
ปริญญา	ครุศาสตร์อุตสาหกรรมบัณฑิต	
หลักสูตร	ครุศาสตร์วิศวกรรม	
ปีการศึกษา	2563	

บทคัดย่อ

ปริญนานิพนธ์ฉบับนี้นำเสนอแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์ ซึ่งใช้รีแอค เนทีฟในการพัฒนาแอปพลิเคชัน โดยแอปพลิเคชันดังกล่าวสร้างขึ้นมาเพื่อลดปัญหาการดูแลที่ไม่ทั่วถึงของรถยนต์ซึ่งอาจจะทำให้ส่วนที่ไม่ได้รับการดูแลเกิดการเสียหายและลดปัญหาการดูแลบำรุงรักษารถยนต์ที่ทับซ้อนกับผู้ที่ใช้รถร่วมกัน โดยแอปพลิเคชันจะมีฟังก์ชันการทำงานอยู่ 2 หมวดหมู่ ได้แก่ หมวดบันทึกการซ่อมบำรุงรักษารถยนต์และหมวดบันทึกช่วยจำ โดยทั้ง 2 หมวดนี้สามารถทำการเชื่อมโยงร่วมกับผู้ใช้งานอื่นได้เพื่อใช้บันทึกการบำรุงรักษารถยนต์และเชื่อข้อมูลการบำรุงรักษารถยนต์ให้ผู้ใช้อื่นทราบถึงประวัติการบำรุงรักษา

Thesis Title	Car Maintenance and Memos Online Application	
Students	Mr. Jarudat	Chaikuad
	Mrs. Thanakamon	Intharasuriyawong
	Mr. Sittinon	Tatiyajinda
Advisor	Asst.Prof. Surachai Pimsalee	
Degree	Bachelor of Science in Industrial Education	
Program in	Engineering Education	
Academic Year	2020	

ABSTRACT

The thesis present construction of Car Maintenance and Memos Online Application which learn about development application on smartphone using react native. This is application made for reduce low maintenance problem, It can damage your car parts. And can reduce the overlap maintenance problem in car use together. In application can divided in 2 categories is car maintenance mode and memos mode. And important part of this application are use together with email.

กิตติกรรมประกาศ

ปริญญา妮พนธ์ฉบับนี้สามารถสำเร็จลุล่วงได้ดีนั้น เนื่องจากความร่วมมือร่วมใจของสมาชิกในกลุ่มทุกท่าน คณะผู้จัดทำข้อขอบพระคุณท่าน ผศ.สุระชัย พิมพ์สาลี และอาจารย์ประจำสาขาวิชาครุศาสตร์วิศวกรรมทุกท่าน เป็นอย่างมากที่ได้กรุณาให้ความช่วยเหลือ ให้ความรู้ในการสร้างแบบโลเคชันและจัดทำเล่มปริญญา妮พนธ์

ขอกราบขอบพระคุณบิดา มารดา และผู้มีพระคุณสำหรับคณะจัดทำที่ได้ทำการสนับสนุนสิ่งต่างๆ ทุกสิ่งทุกอย่างด้านการศึกษาจนมาถึงปัจจุบัน และสุดท้ายขอบคุณเพื่อนๆ ในชั้นปีที่ 4 แขนงวิชาวิศวกรรมคอมพิวเตอร์ คณะครุศาสตร์อุตสาหกรรมและเทคโนโลยี ปีการศึกษา 2563 ที่ให้การช่วยเหลือและเป็นกำลังใจให้เสมอมา

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VIII
สารบัญรูป	IX
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัจจุบัน	1
1.2 จุดมุ่งหมายของโครงการ	1
1.3 สมมติฐานของการจัดทำโครงการ	2
1.4 ขีดความสามารถของโครง	2
1.5 ขั้นตอนของการทำโครงการ	4
1.6 เนื้อหาโดยสังเขป	4
บทที่ 2 ทฤษฎีและหลักการ	6
2.1 กล่าวนำ	6
2.2 การบำรุงรักษาภารณฑ์	6
2.2.1 การตรวจเช็คสภาพภารณฑ์	6
2.2.2 แบตเตอรี่	7
2.2.3 ระบบเบรก	7
2.2.4 หัวเทียน	7
2.3 บันทึกช่วยจำ	7
2.4 การออกแบบส่วนติดตอกับผู้ใช้งานแอปพลิเคชัน	8
2.4.1 ความหมายของโมบายแอปพลิเคชัน	8
2.4.2 หลักการออกแบบแอปพลิเคชันแมททีเรียล ดีไซน์ (Material Design)	8
2.5 ฐานข้อมูลไฟร์เบส	14

สารบัญ (ต่อ)

เรื่อง	หน้า
2.5.1 การบริการของไฟร์เบส	14
2.5.2 การเริ่มต้นใช้งานไฟร์เบส	16
2.6 เครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชัน	17
2.6.1 แอนดรอยด์ สตูดิโอ	17
2.6.2 โหนดเจอส	25
2.6.3 อัลโคบี เอ็กซ์ดี	29
2.6.4 วิชาล สตูดิโอ โค้ด	32
2.6.5 รีแอค เนทีฟ	34
บทที่ 3 การออกแบบ การสร้าง และการทำงาน	36
3.1 กล่าวนำ	36
3.2 การออกแบบระบบ	36
3.2.1 แผนภาพสถาปัตยกรรมของแอปพลิเคชัน (Architecture Diagram)	36
3.2.2 แผนภาพแสดงหน้าที่ที่ระบบต้องกระทำ (Use Case Diagram)	38
3.3 การออกแบบฐานข้อมูล	46
3.3.1 ฐานข้อมูลของการรายงานต์	46
3.3.2 ฐานข้อมูลของการบันทึกการเติมเชื้อเพลิง	47
3.3.3 ฐานข้อมูลของการบันทึกการซ่อมบำรุง	47
3.3.4 ฐานข้อมูลของการบันทึกช่วยจำ	48
3.4 การออกแบบหน้าต่างผู้ใช้งาน	48
3.4.1 การบันทึกการบำรุงรักษารายงานต์	48
3.4.2 การบันทึกช่วยจำ	54
3.5 การสร้างแอปพลิเคชัน	56
3.6 การเพิ่มไฟร์เบสลงในแอปพลิเคชัน	57
3.7 การสร้างระบบยืนยันตัว	59
3.7.1 การติดตั้งแพ๊กเกจที่จำเป็นสำหรับการสร้างระบบยืนยันตัวตน	59

สารบัญ (ต่อ)

เรื่อง	หน้า
3.7.2 การเพิ่มระบบบันทึกข้อมูล	59
3.7.3 การเพิ่มหน้าจอสำหรับเข้าสู่ระบบ	61
3.8 การสร้างบันทึกรายการซ่อมบำรุงรถยนต์	62
3.8.1 การสร้างหน้าสำหรับเลือกรายการ	52
3.8.2 การสร้างหน้าสรุปรายละเอียดของรายการรถยนต์	54
3.9 การสร้างบันทึกช่วยจำ	63
3.10 การสร้างคลาวด์ ไฟร์สโตร์เพื่อใช้เป็นฐานข้อมูล	65
3.10.1 การอัปโหลดและดึงข้อมูลในฐานข้อมูล	67
บทที่ 4 การทดลองและผลการทดลอง	69
4.1 กล่าวนำ	69
4.2 ทดสอบการทำงานของแอปพลิเคชัน	69
4.2.1 หน้าจอเข้าสู่ระบบ	70
4.2.2 หน้าจอรายกิจกรรมยานยนต์	71
4.2.3 หน้าจอเพิ่มรายการรถยนต์	72
4.2.4 หน้าจอแสดงรายละเอียดรถยนต์	73
4.2.5 หน้าจอแสดงสถิติการเงินของรถยนต์	74
4.2.6 หน้าจอแสดงรายการบันทึกการซ่อมบำรุง	75
4.2.7 หน้าจอแสดงรายละเอียดบันทึกรายการซ่อมบำรุง	76
4.2.8 หน้าจอแสดงรายการบันทึกเชื้อเพลิง	77
4.2.9 หน้าจอเพิ่มรายการบันทึกรายการซ่อมบำรุง	78
4.2.10 หน้าจอเพิ่มรายการบันทึกเชื้อเพลิง	79
4.2.11 หน้าจอแชร์ข้อมูลรถ	80
4.2.12 หน้าจอแสดงรายการบันทึกช่วยจำ	81
4.2.13 หน้าจอเพิ่มรายการบันทึกช่วยจำ	84
4.2.14 หน้าจอแสดงรายละเอียดบันทึกช่วยจำ	85

สารบัญ (ต่อ)

เรื่อง	หน้า
4.2.15 หน้าจอแชร์ข้อมูลบันทึกช่วยจำ	86
4.3 ผลการทดสอบของแอปพลิเคชัน	87
บทที่ 5 บทสรุป	89
5.1 สรุป	89
5.2 ปัญหาและวิธีการแก้ไข	89
5.3 แนวทางการพัฒนา	90
บรรณานุกรม	78
ภาคผนวก ก ผังงาน	94
ภาคผนวก ข ฐานข้อมูล	98
ภาคผนวก ค แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์	102
ภาคผนวก ง รหัสต้นฉบับของโปรแกรม	107
ภาคผนวก จ คู่มือการใช้งาน	119
ประวัติผู้แต่ง	135

สารบัญตาราง

ตารางที่	หน้า
3.1 รายละเอียดของยูสเคส เข้าสู่ระบบ	39
3.2 รายละเอียดของยูสเคส เพิ่มรายการรถยนต์	40
3.3 รายละเอียดของยูสเคส บันทึกประวัติการซ้อมบำรุงรถยนต์	41
3.4 รายละเอียดของยูสเคส บันทึกระยะทางสะสมในการเดินทาง	42
3.5 รายละเอียดของยูสเคส ผู้ใช้สามารถบันทึกค่าใช้จ่าย	43
3.6 รายละเอียดของยูสเคส บันทึกช่วยจำ	44
3.7 รายละเอียดของยูสเคส ใช้งานร่วมกัน	45
3.8 ตารางจำลองข้อมูลของรายการรถยนต์	46
3.9 ตารางจำลองข้อมูลของรายการบันทึกการเติมเชื้อเพลิง	47
3.10 ตารางจำลองข้อมูลของข้อมูลของรายการบันทึกการซ้อมบำรุง	47
3.13 ตารางจำลองข้อมูลของบันทึกช่วยจำ	48
4.1 การทดสอบเข้าสู่ระบบ	87
4.2 การทดสอบการใช้งานในหมวดการบำรุงรักษารถยนต์	87
4.3 การทดสอบการใช้งานในหมวดของการบันทึกช่วยจำ	88

สารบัญรูป

รูปที่	หน้า
2.1 ทีอป แอป บาร์	8
2.2 อเลิฟ ไดอะล็อก	10
2.3 ชนิดของปุ่ม	10
2.4 เท็กซ์ ฟิลต์	12
2.5 การ์ด	13
2.6 ปุ่มลอย	14
2.7 การเพิ่มโครงการของไฟร์เบส	16
2.8 การสร้างโครงของไฟร์เบส	17
2.9 หน้าดาวน์โหลดแอนดรอยด์ สตูดิโอ	18
2.10 หน้าติดตั้งแอนดรอยด์ สตูดิโอ	18
2.11 หน้าเริ่มต้นของแอนดรอยด์ สตูดิโอ	19
2.12 หน้าการสร้างอีเมลเตอร์	19
2.13 เลือกอุปกรณ์สำหรับอีเมลเตอร์	20
2.14 เลือกรุ่นของแอนดรอยด์	20
2.15 หน้าต่างดาวน์โหลดเจตีเค	21
2.16 หน้าต่างการติดตั้งเจตีเค	21
2.17 หน้าต่างการกำหนดเส้นทางให้ให้แอนดรอยด์ สตูดิโอและเจตีเค	22
2.18 หน้าต่างการกำหนดเส้นทางชิสเต้มวาระเยี่ยเบิลเจตีเค	22
2.19 หน้าต่างกำหนดตัวแปรให้แอนดรอยด์	23
2.20 หน้าต่างการกำหนดชื่อตัวแปรให้เจตีเค	23
2.21 หน้าต่างการแก้ไขเส้นทางให้เจตีเค	24
2.22 หน้าต่างการกำหนดเส้นทางสภาพแวดล้อมให้เจตีเค	25
2.23 หน้าต่างดาวน์โหลดหนนดเจอส	26
2.24 หน้าต่างการติดตั้งหนนดเจอส	26
2.25 หน้าต่างข้อกำหนดในการติดตั้งหนนดเจอส	27
2.26 หน้าต่างการเลือกที่อยู่การติดตั้งหนนดเจอส	27

สารบัญรูป (ต่อ)

รูปที่	หน้า
2.27 หน้าต่างการติดตั้งแบบเลือกเอง	28
2.28 หน้าต่างยืนยันการติดตั้งโหนดเจeos	28
2.29 การเลือกขนาดของพื้นที่งาน	30
2.30 โหมดการทำงานของอะโดบี เอ็กซ์ดี	30
2.31 เครื่องมือในอะโดบี เอ็กซ์ดี	31
2.32 ตัวอย่างการเขื่อมหน้า	32
2.33 หน้าดาวน์โหลดวิชวล สตูดิโอ โค้ด	33
2.34 หน้าต่างข้อกำหนดในการติดตั้งวิชวล สตูดิโอ โค้ด	33
2.35 หน้าติดตั้งรีแอค เนทีฟ ชีแอล็อก	35
3.1 แผนภาพสถาปัตยกรรมของแอปพลิเคชัน	36
3.2 แผนภาพผู้ใช้ส่วนตัวของแอปพลิเคชันบันทึกการบำรุงรักษาโดยนักเขียนและบันทึกช่วยจำออนไลน์	38
3.3 หน้าแดชบอร์ดการซ่อมบำรุงรักษา	49
3.4 หน้าจอค่าใช้จ่ายรวมของรักษา	50
3.5 หน้าจอเพิ่มรายการบันทึกการซ่อมบำรุง	51
3.6 หน้าประวัติรายการบันทึกการซ่อมบำรุง	52
3.7 หน้ารายละเอียดรายการการบันทึกการซ่อมบำรุง	53
3.8 หน้ารายการบันทึกช่วยจำ	54
3.9 หน้าจอเพิ่มรายการบันทึกความจำ	55
3.10 การสร้างแอปพลิเคชันโดยใช้คำสั่ง \$ react-native init Project	56
3.11 หน้าอีเมลเตอร์ (Emulator)	56
3.12 หน้าต่างไฟร์เบสคอนโซล	57
3.13 หน้าการลงทะเบียนแอปพลิเคชันในไฟร์เบส	57
3.14 หน้าดาวน์โหลดกูเกิลเซอร์วิส	58
3.15 หน้าเส้นทาง android -> app	58
3.16 โค้ดเพิ่มปลั๊กอินไฟร์เบส	59
3.17 การตั้งค่าระบบยืนยันตัวตน	60

สารบัญรูป (ต่อ)

หัวข้อ	หน้า
3.18 เพิ่มฟังก์ชันการยืนยันตัวตน	60
3.19 สร้างไฟล์เดอร์ screen	61
3.20 เขียนฟังก์ชันเข้าสู่ระบบ	61
3.21 การสร้างหน้าสำหรับเลือกรายการรายนั้น	62
3.22 การกำหนดแบบความก้าวหน้าแบบง่ายๆ	63
3.23 การสร้างรายการบันทึกเชือเพลิง	63
3.24 การสร้างฟังก์ชันคำนวณวันสิ้นสุดภาค	64
3.25 การสร้างช่องป้อนข้อมูลสำหรับบันทึกช่วยจำ	64
3.26 การสร้างหน้าสำหรับแสดงบันทึก	65
3.27 การสร้างฟังก์ชันในการค้นหา	65
3.28 หน้าต่างการสร้างคลาวด์ ไฟร์สโตร์	66
3.29 หน้าต่างการเลือกใหม่ของการสร้างคลาวด์ ไฟร์สโตร์	66
3.30 หน้าฐานข้อมูลคลาวด์ ไฟร์สโตร์	67
3.31 ฟังก์ชันในการอัปโหลดข้อมูล โดยใช้เมธอด .add()	67
3.32 หน้าฐานข้อมูลคลาวด์ ไฟร์สโตร์	68
3.33 ฟังก์ชันในการดึงข้อมูล โดยใช้เมธอด .where() และใช้เมธอด .get()	68
4.1 หน้าจอเข้าสู่ระบบ	70
4.2 หน้าจอรายการรายนั้น	71
4.3 หน้าจอเพิ่มรายการรายนั้น	72
4.4 หน้าจอแสดงรายละเอียดรายนั้น	73
4.5 หน้าจอแสดงสถิติการเงินของรายนั้น	74
4.6 หน้าจอแสดงรายการบันทึกการซ่อมบำรุง	75
4.7 หน้าจอแสดงรายละเอียดบันทึกการซ่อมบำรุง	76
4.8 หน้าจอแสดงรายการบันทึกเชือเพลิง	77
4.9 หน้าจอเพิ่มรายการการซ่อมบำรุง	78
4.10 หน้าจอเพิ่มรายการบันทึกเชือเพลิง	79

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.11 หน้าจอแชร์ข้อมูลรถ	80
4.12 หน้าแสดงรายการบันทึกช่วยจำ	81
4.13 รูปแบบการค้นหาบันทึกช่วยจำ	82
4.14 หน้าจอเพิ่มรายการบันทึกช่วยจำ	84
4.15 หน้าจอแสดงรายละเอียดบันทึกช่วยจำ	85
4.16 หน้าจอแชร์ข้อมูลบันทึกช่วยจำ	86

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

รายงานต์เป็นสิ่งหนึ่งที่ในหลายครอบครัวมีการใช้ร่วมกันระหว่างสมาชิกภายในครอบครัว ทำให้ในการบำรุงรักษารายงานต์อาจเกิดการบำรุงรักษารายงานต์ทับช้อนกันหรือมีการบำรุงรักษารายงานต์ไม่ทั่วถึง เนื่องจากไม่ได้มีการบันทึกประวัติการดูแลรักษารายงานต์

ซึ่งในการบำรุงรักษารายงานต์ ผู้ใช้รายงานต์มักจะนำรายงานต์ไปบำรุงรักษาเมื่อครบกำหนดรายปี หรือตามระยะเวลาแต่ก็ยังมีผู้ใช้อีกหลายคนที่นำรถไปซ่อมบำรุงเฉพาะส่วน และเมื่อไม่ได้มีการบันทึกประวัติการซ่อมบำรุงรักษารายงานต์อาจจะทำให้ผู้ใช้มีการบำรุงรักษารายงานต์ไม่ทั่วถึง ทำให้ในส่วนที่ไม่ได้บำรุงรักษาเกิดการสึกหรอจากการใช้งานตามระยะเวลาหรือระยะเวลาซึ่งอาจทำให้ผู้ใช้เกิดอันตรายและเกิดค่าใช้จ่ายโดยไม่จำเป็นได้หากไม่มีการบำรุงรักษา

1.2 จุดมุ่งหมายของโครงการ

- เพื่อศึกษาการสร้างและพัฒนาแอปพลิเคชันบนสมาร์ทโฟนโดยใช้รีแอค เนทีฟ (React Native)
- เพื่อออกแบบระบบโครงสร้างแอปพลิเคชันบันทึกการบำรุงรักษารายงานต์และบันทึกช่วยจำออนไลน์
- เพื่อสร้างแอปพลิเคชันบันทึกการบำรุงรักษารายงานต์และบันทึกช่วยจำออนไลน์โดยใช้รีแอค เนทีฟ
- เพื่อทดสอบข้อผิดพลาดการทำงานและระบบของแอปพลิเคชัน
- เพื่อนำแอปพลิเคชันบันทึกการบำรุงรักษารายงานต์และบันทึกช่วยจำออนไลน์ไปใช้งานได้จริง

1.3 สมมติฐานของการจัดทำโครงงาน

แอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกซ่อมบำรุงออนไลน์สามารถใช้งานได้จริงและสามารถซ่อมแก้ปัญหาในการบันทึกการใช้งานบำรุงรักษารถยนต์และบันทึกซ่อมบำรุงได้

1.4 ขีดความสามารถของโครงงาน

1.4.1 การเก็บข้อมูลการบำรุงรักษารถยนต์

1.4.1.1 รายการรถยนต์

1. สามารถเพิ่ม และลบรายการรถยนต์ที่ต้องการได้
2. สามารถบันทึกรายละเอียดได้ดังนี้

2.1 ภาพประกอบ

2.2 ยี่ห้อรถยนต์

2.3 ทะเบียนรถยนต์

2.4 ประวัติกัยรถยนต์

2.5 ภาษีรถยนต์

1.4.1.2 บันทึกประวัติการซ่อมบำรุง สามารถบันทึกรายละเอียดได้ดังนี้

1. วัน เดือน ปี
2. ระยะทางสะสม
3. ประเภทการซ่อมบำรุง
4. ศูนย์บริการ
5. ค่าใช้จ่ายในการซ่อมบำรุง
6. แนบใบเสร็จ

1.4.1.3 บันทึกระยะทางสะสมในการเดินทาง

1. สามารถบันทึกระยะทางสะสมการเดินทางด้วยการระบุระยะทางที่เดินทาง
2. บันทึกค่าใช้จ่ายในการซ่อมบำรุงรถยนต์

3. ค่าใช้จ่ายเชื้อเพลิง

3.1 ระยะทางสะสม

3.3 ราคา

3.4 วันที่

4. ค่าภาษี

4.1 วันที่

4.2 จำนวนเงิน

5. ค่าผ่อนรถ (กรณีผ่อนรถ) มีรายละเอียดทั้งหมดดังนี้

5.1 ค่างวด

5.2 ระยะงวด

5.3 เงินที่ชำระ

6. รายการการแจ้งเตือน มีรายละเอียดทั้งหมดดังนี้

6.1 แบตเตอรี่

6.2 วันจ่ายภาษี

1.4.2 บันทึกช่วยจำ

1.4.2.1 สามารถสร้างบันทึกช่วยจำ

1.4.2.2 สามารถແນບຮູບພາບປະກອບໃນແຕ່ລະບັນທຶກໄດ້

1.4.2.3 สามารถค้นหาบันทึกช่วยจำທີ່ຕ້ອງການໄດ້

1.4.3 ผู้ใช้งาน

1.4.3.1 สามารถเข้าสู่ระบบผ่านakuเกิล แอ็คเคนต์ (Google Account) และเฟซบຸກ แอ็คเคนต์ (Facebook Account)

1.4.3.2 สามารถสร้างบันทึกการบำรุงรักษารถยนต์หรือบันทึกช่วยจำแบบใช้งานร่วมกันได้สูงสุด

5 แอ็คเคนต์

1.4.3.3 สามารถเข้าถึงบันทึกที่ต้องการได้ด้วยการเชิญจากเจ้าของบันทึกการบำรุงรักษาโดยนั่นหรือบันทึกช่วยจำ

1.5 ขั้นตอนของการทำโครงการ

โครงการนี้เป็นการจัดทำแอปพลิเคชันบนสมาร์ทโฟนโดยแอปพลิเคชันเกี่ยวกับการบำรุงรักษาโดยนั่นและบันทึกช่วยจำ ในระยะแรกจะทำการวิเคราะห์ความต้องการของระบบและศึกษาการพัฒนาแอปพลิเคชันด้วยรีแอค เนทีฟ ศึกษาข้อมูลเกี่ยวกับการบำรุงรักษาโดยนั่นและการจดบันทึกรวมทั้งทำการศึกษาแอปพลิเคชันเกี่ยวกับการบำรุงรักษาโดยนั่นและการจดบันทึกที่ให้บริการอยู่ในเพลย์สโตร์ (Play Store) จากนั้นจึงทำการศึกษาหลักการออกแบบควบคู่กับการออกแบบแอปพลิเคชันในส่วนของผู้ใช้ เมื่อทำการศึกษาและออกแบบอยู่ในระดับที่ต้องการจะทำการเริ่มต้นการพัฒนาแอปพลิเคชันพร้อมกับการทดสอบควบคู่กันไปและทำการสรุปผลต่อไป

1.6 เนื้อหาโดยสังเขป

เนื้อหาภายในปริญนานิพนธ์ฉบับนี้แบ่งออกเป็นบทต่าง ๆ เพื่อความสะดวกต่อการศึกษา และทำความเข้าใจในแต่ละบทจะประกอบไปด้วยเนื้อหาดังต่อไปนี้

บทที่ 1 กล่าวถึงความเป็นมาความสำคัญของปริญนานิพนธ์ ขีดความสามารถของโครงการ และเนื้อหานอกบทต่าง ๆ โดยสังเขป

บทที่ 2 ประกอบด้วย ทฤษฎีต่าง ๆ เกี่ยวกับข้อมูลเกี่ยวกับรถยนต์ ได้แก่ การบำรุงรักษาโดยนั่น การตรวจเช็คสภาพรถยนต์ ข้อมูลเกี่ยวกับบันทึก การออกแบบส่วนติดต่อกับผู้ใช้งานแอปพลิเคชัน หลักการออกแบบแอปพลิเคชัน ฐานข้อมูลไฟร์เบส (Firebase Database) บริการของไฟร์เบส เครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชัน ได้แก่ แอนดรอยด์สตูดิโอ (Android Studio) โนนดเจอส (Node.js) อะดobe อีเอกซ์ดี (Adobe XD) วิชวลสตูดิโอโคด (Visual Studio Code) รีแอค เนทีฟ

บทที่ 3 กล่าวถึงเนื้อหาที่เกี่ยวข้องกับแผนผังการทำงานการออกแบบ การสร้าง และการทำงานของแอปพลิเคชันบันทึกการบำรุงรักษาโดยนั่นและบันทึกช่วยจำออนไลน์ ได้แบ่งออกเป็น 5 ส่วน ได้แก่

การออกแบบระบบ การออกแบบระบบฐานข้อมูล การออกแบบหน้าต่างผู้ใช้งาน การติดตั้งเครื่องมือสำหรับการพัฒนาโมบายแอปพลิเคชันด้วยรีแอค เนทีฟ และการสร้างและการทำงานของแอปพลิเคชัน

บทที่ 4 บทนี้จะกล่าวถึงการทดลองและผลการทดลองของแอปพลิเคชันบันทึกการบำรุงรักษา รถยนต์และบันทึกช่วยจำออนไลน์โดยจะมีการทดสอบการทำงานของแอปพลิเคชัน และผลการทดสอบการทำงานของแอปพลิเคชัน

บทที่ 5 เป็นบทสรุปการจัดทำโครงงาน ปัญหาที่เกิดขึ้นและแนวทางการแก้ไขรวมทั้งแนวทางการพัฒนา

ภาคผนวก ก จะกล่าวถึงผังงานทั้งหมดภายในแอปพลิเคชันตั้งแต่เริ่มต้นการทำงานจนการจบการทำงานของแอปพลิเคชัน

ภาคผนวก ข ฐานข้อมูลจะกล่าวถึงโครงสร้างข้อมูลที่ใช้ภายในแอปพลิเคชันโดยจะมีข้อมูลรายการรถยนต์ ข้อมูลรายการบันทึกการเติมเชื้อเพลิง ข้อมูลรายการบันทึกการซ่อมบำรุง และข้อมูลบันทึกช่วยจำ

ภาคผนวก ค แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์จะกล่าวถึงแผนภาพการทำงานแบบลำดับปฏิสัมพันธ์ในแต่ละยูสเคส โดยจะมีเข้าสู่ระบบ เพิ่มรายการรถยนต์ บันทึกประวัติการซ่อมบำรุง รถยนต์ บันทึกระยะเวลาทางสะสมในการเดินทาง บันทึกค่าใช้จ่าย บันทึกช่วยจำ และใช้งานร่วมกัน

ภาคผนวก ง รหัสต้นฉบับของโปรแกรม จะกล่าวถึงรหัสต้นฉบับของโปรแกรมในส่วนของระบบยืนยันตัวตน และระบบบันทึกช่วยจำ

ภาคผนวก จ คู่มือการใช้งานจะกล่าวถึงวิธีการใช้งานแอปพลิเคชันเบื้องต้น รวมถึงรายละเอียดในแต่ละหน้าของแอปพลิเคชัน

บทที่ 2

ทฤษฎีและหลักการ

2.1 กล่าวนำ

เนื้อหาของปริญณานิพนธ์ในบทนี้เป็นทฤษฎีและหลักการที่จะนำมาใช้ประกอบการสร้างแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์ มีทฤษฎีที่เกี่ยวข้องและจำเป็นต่อการพัฒนาแอปพลิเคชัน ทั้งด้านเทคนิคและการวางแผนระบบฐานข้อมูล โดยจะมีรายนั้น บันทึกช่วยจำการออกแบบส่วนติดต่อกับผู้ใช้งานโดยเบยแอปพลิเคชัน ฐานข้อมูลไฟร์เบส และเครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชัน

2.2 การบำรุงรักษารถยนต์

การบำรุงรักษารถยนต์ตามระยะเวลาถือเป็นเรื่องจำเป็น โดยทางผู้ผลิตรถยนต์ได้กำหนดไว้ 2 ลักษณะด้วยกัน คือ ระยะเวลา (นับตั้งแต่วันที่ออกรถ) และระยะเวลา (ที่ว่างใช้งานไปแล้ว) หากมีการใช้รถน้อย ให้คำนวณจากระยะเวลา แต่ถ้าใช้รถยนต์เป็นปกติมาก ก็ให้คิดจากระยะทางปัจจุบัน (บนมาตรฐาน) หรือแล้วแต่ระยะเดินทางก็ได้ การนำรถเข้าตรวจสอบเช็คตามระยะเวลาที่กำหนด จะเกิดประโยชน์อย่างมากแก่ผู้ใช้รถ ซึ่งส่วนต่าง ๆ ย่อมมีการซ่อม เปลี่ยน หรือบำรุงรักษาตามกำหนด ก็จะส่งผลให้มีการใช้งานได้ยาวนาน ซึ่งช่วยประหยัดในด้านต่าง ๆ เป็นอย่างดี และไม่จำเป็นที่จะต้องซ่อมบ่อย เพื่อรักษาพร้อมใช้งานอยู่ตลอดเวลา รวมถึงสมรรถนะสูงสุดด้วย ดังนั้นควรนำรถยนต์เข้าเช็คระยะที่ศูนย์บริการตามที่กำหนด

2.2.1 การตรวจสอบสภาพรถยนต์

2.2.1.1 ยางรถยนต์

ยางรถยนต์ มีหน้าที่ในการเป็นตัวกลางถ่ายทอดพลังงานขับเคลื่อน และการหยุดรถ การรับน้ำหนักรถยนต์ และน้ำหนักบรรทุก ลดแรงกระแทกและสั่นสะเทือนจากพื้นถนน และทำให้รถเปลี่ยนทิศทางได้ตามต้องการโดยอายุการใช้งานของยางรถยนต์ในปัจจุบันมีอายุการใช้งานเฉลี่ย 30,000 - 40,000 กิโลเมตร หรือ 2 ปี

2.2.1.2 น้ำมันเครื่อง

น้ำมันเครื่อง มีหน้าที่ในการหล่อลื่นซึ่งส่วนต่างๆ ภายในเครื่องยนต์ เพื่อลดแรงเสียดทาน ช่วยล้างสิ่งสกปรกจากการเผาไหม้ และป้องการเกิดการสึกหรอ สนิม หรือการกัดกร่อนของซึ่งส่วนเครื่องยนต์ ในการเปลี่ยนถ่านน้ำมันเครื่อง จะพิจารณาตามชนิดของน้ำมันเครื่องและแต่ละชนิด ระยะ

เครื่องยนต์ ในการเปลี่ยนถ่ายน้ำมันเครื่อง จะพิจารณาตามชนิดของน้ำมันเครื่องและแต่ละชนิด ระยะการใช้งานจะไม่เท่ากัน โดยพิจารณาดังนี้

1. น้ำมันสังเคราะห์ มีอายุการใช้งานเฉลี่ยตามระยะทางทั้งหมด 15,000-20,000 กิโลเมตร
2. น้ำมันกึ่งสังเคราะห์ มีอายุการใช้งานเฉลี่ยตามระยะทางทั้งหมด 10,000-15,000 กิโลเมตร
3. น้ำมันรرمดา มีอายุการใช้งานเฉลี่ยตามระยะทางทั้งหมด 7,000-7,500 กิโลเมตร

2.2.2 แบตเตอรี่

แบตเตอรี่ ภายในรถยนต์นั้นมีหน้าที่ได้ในการจ่ายกระแสไฟกับมอเตอร์สตาร์ทที่ทำหน้าที่ในติดเครื่องยนต์ครั้งแรก และช่วยในการจ่ายกระแสไฟไปใช้อุปกรณ์อิเล็กทรอนิกส์ต่างๆภายในรถยนต์ เช่น ไฟรถยนต์ วิทยุ แอร์ฯ โดยอายุการใช้งานในแบตเตอรี่ทุกประเภทจะมีอายุการใช้เฉลี่ย 20 เดือน หรือ 1 ปี 8 เดือน

2.2.3 ระบบเบรก

ระบบเบรก ทำหน้าที่หลักในการชะลอความเร็วของรถยนต์ ภายในระบบเบรกที่ผู้ใช้งานต้องดูแล 2 อย่างด้วยกัน ดังนี้

2.2.3.1 ajanเบรก

อายุการใช้ของajanเบรก สามารถดูได้ด้วยสายตา หากajanเบรกไม่เรียบและมีร่องรอย แสดงว่าajanเบรกต้องได้รับการเปลี่ยน เพื่อความปลอดภัยของผู้ใช้งาน

2.2.3.2 ผ้าเบรก

จะมีอายุการใช้งานเฉลี่ย 30,000 – 50,000 กิโลเมตร

2.2.4 หัวเทียน

หัวเทียนมีหน้าที่ในการจุดระเบิดในห้องเผาไหม้ซึ่งหากมีการทำงานที่ผิดปกติหรือเสื่อมสภาพอาจทำให้ประสิทธิภาพในการจุดระเบิดนั้นอยู่ด้วยจะมีผลให้เครื่องยนต์ติดก้มไม่สามารถเพิ่มอัตราเร่งได้เท่าที่ควรในเวลาที่ผู้ขับเหยียบคันเร่ง โดยผู้ขับซึ่ควรทำการเปลี่ยนหัวเทียนเมื่อระยะครบ 100,000 กิโลเมตร

2.3 บันทึกช่วยจำ

คำว่า บันทึก ในพจนานุกรม ฉบับราชบัณฑิตสถาน พ.ศ. 2554 กล่าวไว้ว่า บันทึก หมายถึง จดข้อความเพื่อช่วยความทรงจำหรือเพื่อเป็นหลักฐาน จดหรือถ่ายทำไว้เพื่อช่วยความจำหรือเพื่อเป็นหลักฐาน เช่น บันทึกรายงานการประชุม บันทึกภาพ บันทึกเสียง จดย่อฯ ไว้เพื่อให้รู้เรื่องเดิม

บันทึกช่วยจำเป็นสิ่งที่ช่วยให้ผู้ใช้งานจดจำข้อมูลที่จำเป็นต่างๆของผู้ใช้งาน ป้องการการลืมจากการจดจำด้วยสมอง ช่วยลดระยะเวลาเดือนความจำด้วยสมอง บันทึกช่วยจำจะได้ผลได้ดีที่สุดก็ต่อเมื่อผู้ใช้งานทำการจดวันที่พร้อมกับเนื้อหา และที่สำคัญที่สุดผู้ใช้งานควรที่จะเปิดบันทึกช่วยจำเป็นประจำ ผู้ใช้งานจะสามารถจดจำสิ่งที่ต้องจำได้ทันทีหากผู้ใช้มีการพกบันทึกช่วยจำติดตัว การเปิดบันทึกช่วยจำเป็นประจำและการพกบันทึกช่วยจำติดตัวเพื่อช่วยป้องกันความผิดพลาดในการเตือนความจำและช่วยในการเตือนความจำให้มีประโยชน์สูงสุด

2.4 การออกแบบส่วนติดต่อกับผู้ใช้ของแอปพลิเคชัน

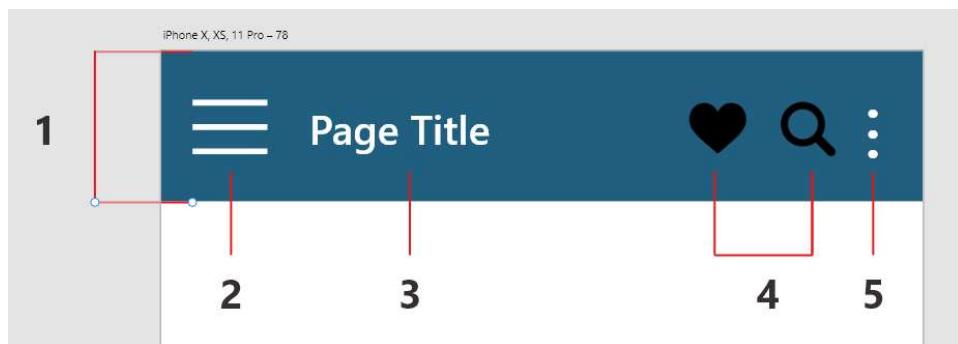
2.4.1 ความหมายของโมบายแอปพลิเคชัน

โมบายแอปพลิเคชันเป็นโปรแกรมประยุกต์สำหรับสมาร์ทโฟนในยุคปัจจุบัน โดยมีไว้เพื่อตอบสนองความต้องการของผู้บริโภคในหลายๆ ด้าน เช่น ด้านการสื่อสาร ด้านการศึกษา ด้านธุรกิจ หรือด้านผู้บริโภคเอง โดยในปัจจุบันจะเห็นได้ว่ามีหลายแอปพลิเคชันที่เป็นนิยมอย่างมาก ตัวอย่างเช่น ไลน์ (Line) เพชบุ๊ก อินสตาแกรม (Instagram) ฯลฯ

2.4.2 หลักการออกแบบแอปพลิเคชันแมททีเรียล ดีไซน์ (Material Design)

2.4.2.1 ท็อป แอป บาร์ (Top App Bars)

เป็นแบบด้านบนที่แสดงเนื้อหาที่เกี่ยวกับการดำเนินการในหน้าปัจจุบัน



รูปที่ 2.1 ท็อป แอป บาร์

1. คอนเทนเนอร์ (Container) ทำหน้าที่เป็นกรอบให้กับส่วนประกอบต่างๆ
2. เนวิกे�เตอร์ ไอคอน (Navigation icon) เป็นไอคอนสำหรับการนำทาง ซึ่งจะอยู่ทางซ้ายสุดซึ่งไม่จำเป็นต้องมี

3. หัวข้อ (Title) ใช้อธิบายว่าผู้ใช้งานกำลังอยู่ในส่วนใดหรือหน้าใด ซึ่งจะอยู่ที่ทางขวา ถัดมาจากเนวิเกเตอร์ ไอคอนซึ่งไม่จำเป็นต้องมี
4. แอคชั่น ไอเทม (Action items) ตำแหน่งของไอคอนที่ใช้งานมากที่สุดจะอยู่ทางซ้าย น้อยจะอยู่ทางขวา หากมีไอคอนที่จะใช้งานมากแต่ไม่เพียงพอในพื้นที่ให้นำไปไว้อยู่ในโอลเวอร์ฟลู เมนู (Overflow menu) โดยที่แอ็คชัน ไอเทม (Action items) จะอยู่ทางขวา ไม่จำเป็นต้องมี
5. โอลเวอร์ฟลู เมนู หากมีจะอยู่ทางขวาสุด ไม่จำเป็นต้องมี

2.4.2.2 แคบเมนูซ่อนได้ (Navigation drawer)

เป็นการนำทางพาไปยังจุดหมายต่างๆ ของแอป ซึ่งสามารถอยู่บนหน้าจอได้แบบถาวรหรือควบคุมได้ด้วยไอคอน โดยแบบเมนูซ่อนได้หมายความกับแอปพลิเคชัน

1. ที่ใชในการนำทางที่มี 5 หรือมากกว่าขึ้นไป
2. ใช้กับแอปที่มี 2 ชั้นหรือมากกว่า
3. ใช้นำไปยังหน้าที่ต้องการเข้าถึงได้อย่างรวดเร็ว

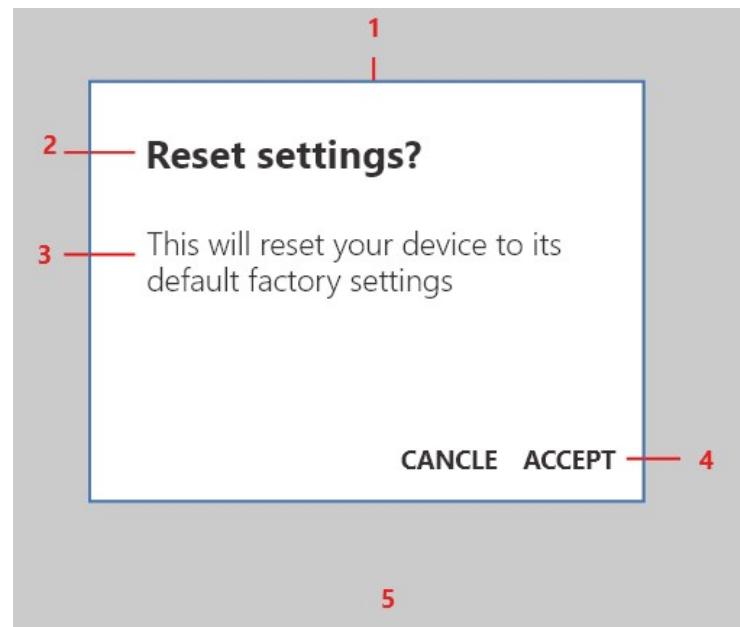
2.4.2.3 ชีเลคชัน คอนโทรล (Selection controls)

1. เช็คบ็อกซ์ (Checkboxes) จะอนุญาตให้ผู้ใช้สามารถเลือกหลายช่อง หากมีเช็คบ็อกซ์ที่เป็นหัวข้อใหญ่ออกจากกรุ๊ปเลือก เช็คบ็อกซ์ที่เป็นหัวข้อย่อยในหัวข้อใหญ่นั้นจะถูกนำออกจากการกรุ๊ปเลือกด้วยเช่นกัน
2. ปุ่มแบบเรดิโอ (Radio buttons) ใช้ปุ่มตัวเลือกแทนการทำเป็นปุ่มเครื่องหมายหากเลือกได้เพียงช่องเดียว หากตัวเลือกหัวข้อใหญ่มีหัวข้อย่อย หัวข้อใหญ่จะทำการแสดงหัวข้อย่อยก็ต่อเมื่อถูกเลือกเท่านั้น และจะแสดงแบบแนวตั้ง เพื่อประหยัดพื้นที่

2.4.2.4 ไดอะล็อก (Dialog)

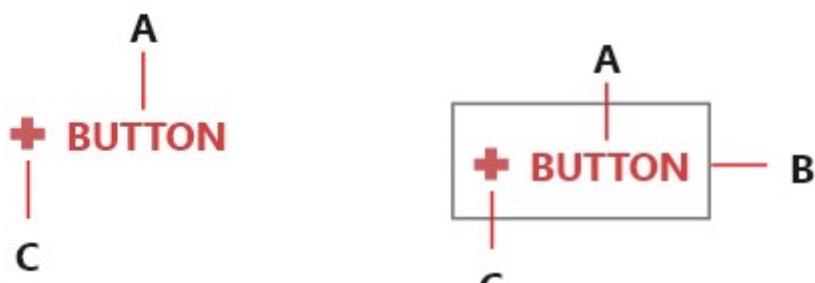
จะเน้นการให้ความสนใจแก่ผู้ใช้ เพื่อที่ให้แนใจว่าต้องการกราฟิกจริง โดยจะแสดงผลเป็นแบบเด้ง (Pop-up) ขึ้นกลางหน้าจอ ภายในไดอะล็อกจะกล่าวถึงข้อมูลที่ตรงไปตรงมา และจะแสดงขึ้นต่อเมื่อผู้ใช้ได้กระทำการเปลี่ยนแปลงสิ่งใดสิ่งหนึ่ง

1. อเลิท (Alert) จะเป็นไดอะล็อกที่มีการแจ้งขึ้นมาว่าต้องการยืนยันหรือไม่ โดยจะมีปุ่มไม่และใช่มาให้เลือก ซึ่งมีส่วนประกอบดังรูปที่ 2.2 อเลิท ไดอะล็อก (Alert Dialog)



รูปที่ 2.2 dialogue ไดอะล็อก

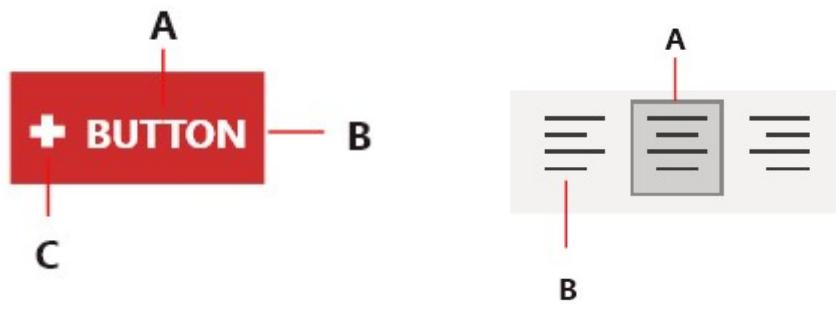
- 1.1 คอนเทนเนอร์ (Container) ทำหน้าที่เป็นกรอบให้กับส่วนประกอบต่างๆ
 - 1.2 หัวข้อ dialogue และหัวข้อที่ต้องการให้แสดง (ไม่จำเป็นต้องมี)
 - 1.3 รายละเอียด จะแสดงรายละเอียดที่ต้องการแจ้งเตือน
 - 1.4 ปุ่ม จะแสดงตัวเลือกให้แก่ผู้ใช้
 - 1.5 สคริม (Scrim) เป็นหน้าจอแสดงผลก่อนที่จะเกิดการแจ้ง dialogue ไดอะล็อก
- 2.4.2.5 ปุ่มการออกแบบปุ่มที่ดีควรมีลักษณะต่างๆ ดังรูปที่ 2.3



ก. ปุ่มตัวอักษร

ข. ปุ่มเฉพาะกรอบ

รูปที่ 2.3 ชนิดของปุ่ม



ค. ปุ่มสีเหลืองผืนผ้า

๙. ปุ่มเฉพาะกรอบสีเหลืองจัตุรัส

รูปที่ 2.3 (ต่อ) ชนิดของปุ่ม

1. ปุ่มตัวอักษร
 - 1.1 A คือ ตัวอักษร
 - 1.2 C คือ ไอคอน (ไม่จำเป็นต้องมี)
 2. ปุ่มเฉพาะกรอบ
 - 2.1 A คือ ตัวอักษร
 - 2.2 B คือ ค่อนเทนเนอร์
 - 2.3 C คือ ไอคอน (ไม่จำเป็นต้องมี)
 3. ปุ่มสีเหลี่ยมผืนผ้า
 - 3.1 A คือ ตัวอักษร
 - 3.2 B คือ ค่อนเทนเนอร์
 - 3.3 C คือ ไอคอน (ไม่จำเป็นต้องมี)
 4. ปุ่มเฉพาะกรอบสีเหลี่ยมจัตุรัส
 - 4.1 A คือ ตัวอักษร
 - 4.2 B คือ ไอคอน (ไม่จำเป็นต้องมี)

2.4.2.6 รายการ (Menus)

รายการที่ดีจะต้องเปิดง่าย มีหัวข้อที่ผู้ใช้ต้องการ และดูง่าย โดยรายการจะมีทั้งหมด 2 ชนิด คือ รายการแบบไม่แสดงรายการกำลังใช้งาน และรายการแบบแสดงรายการที่กำลังใช้งาน

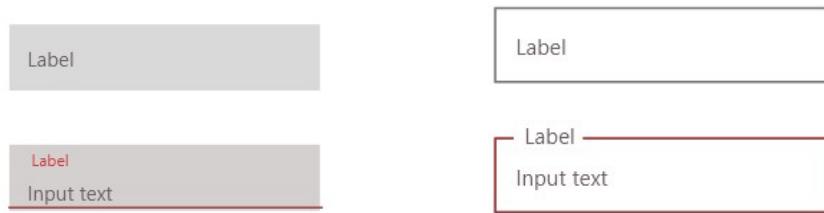
1 การจัดทำรายภาร รายภารลําดับ เลื่อนหน้าสุดของรายภาร (II) หน้างานเดียวที่แสดงผล

1. การจัดวางรายการ รายการจะขึ้นอยู่หน้าสุดของยูทูบ (UI) ทั้งหมดที่แสดงอยู่

2. ความสูงของรายการ รายการจะแสดงสูงสุดต้องเหลือพื้นที่แอปพลิเคชันได้อย่างน้อย 1 ตาราง
3. ตำแหน่งของรายการ ตำแหน่งจะต้องแสดงอยู่ในส่วนของมุม และรายการย่ออยู่ จะต้องแสดงในด้านที่อยู่ตรงข้ามกับตำแหน่งปุ่มรายการหลัก

2.4.2.7 เท็กซ์ พิลต์ (Text fields)

เท็กซ์ พิลต์ใช้เพื่อให้เกิดความชัดเจนให้การตอบโต้โดยเท็กซ์ พิลต์มี 2 แบบ คือ พิลต์ เท็กซ์ พิลต์ (Filled text fields) และเออต์ไลน์ เท็กซ์พิลต์ (Outline text fields)



ก. พิลต์ เท็กซ์ พิลต์

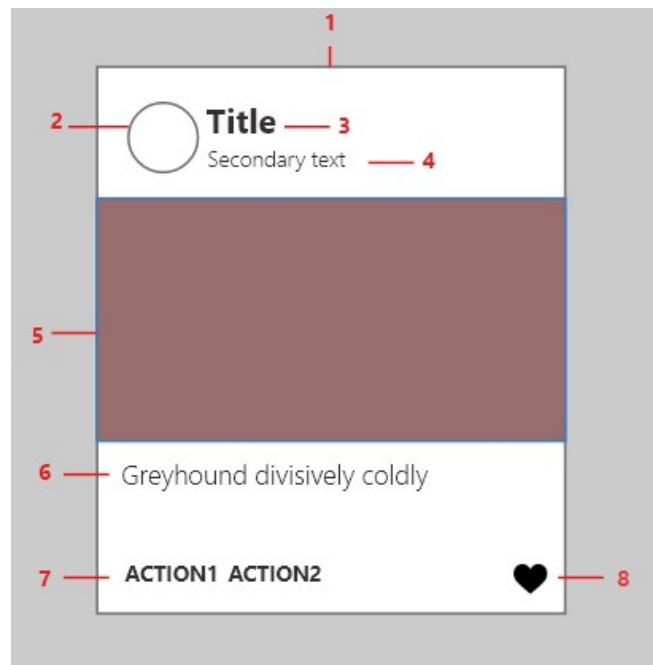
ข. เออต์ไลน์ เท็กซ์พิลต์

รูปที่ 2.4 เท็กซ์ พิลต์

เท็กซ์ พิลต์ 2 แบบมีฟังก์ชันการทำงานที่เหมือนกัน การใช้งานจึงขึ้นอยู่กับสไตร์เพียงอย่างเดียวโดยในการเลือกใช้จะขึ้นอยู่กับความเหมาะสมสมกับสไตร์ของแอปพลิเคชัน ความเอื้อต่อหน้าต่าง ติดต่อผู้ใช้งานที่สุดและมีความแตกต่างจากส่วนประกอบส่วนอื่นๆ เช่น ปุ่ม และเนื้อหาโดยรอบ

2.4.2.8 การ์ด (Card)

การ์ดคือส่วนที่ใช้ในการแสดงผลเนื้อหาและปุ่มต่างๆภายในหัวข้อเดียวกันโดยการออกแบบจะมีส่วนประกอบดังนี้

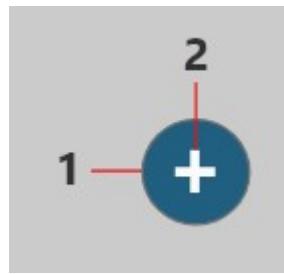


รูปที่ 2.5 การ์ด

1. คอนเทนเนอร์ ใช้สำหรับเป็นขอบเขตในการวางส่วนประกอบภายในทั้งหมด
2. ภาพย่อ (Thumbnail) สามารถใช้รูป โลโก้หรือภาพตัวแทนได้เช่นไม่จำเป็นต้องมี
3. หัวข้อหลัก (Header Text) ใช้สำหรับแสดงหัวข้อหลักที่ต้องการแสดงซึ่งไม่จำเป็นต้องมี
4. หัวข้อย่อย (Subhead) ใช้สำหรับแสดงหัวข้อย่อยที่ต้องการซึ่งไม่จำเป็นต้องมี
5. มีเดีย (Media) ใช้สำหรับแสดงรูปภาพ ภาพถ่ายได้เช่นไม่จำเป็นต้องมี
6. คำอธิบายเพิ่มเติม (Supporting text) ใช้สำหรับเพิ่มคำอธิบายสำหรับการ์ดได้เช่นไม่จำเป็นต้องมี
7. ปุ่ม ใช้สำหรับเพิ่มปุ่มที่ต้องนำทางไปที่อื่นได้เช่นไม่จำเป็นต้องมี
8. ไอคอน ใช้สำหรับเพิ่มไอคอนที่ต้องการนำเสนอทางไปยังที่อื่นได้เช่นไม่จำเป็นต้องมี

2.4.2.9 ปุ่มลอย (floating action button)

ปุ่มลอยจะปรากฏอยู่บนหน้าเนื้อหาของหน้าจอทั้งหมดโดยทว่าไปแล้วจะเป็นรูปวงกลมโดยจะมีไอคอนอยู่ตรงกลางโดยปุ่มลอยจะมีส่วนประกอบดังนี้



รูปที่ 2.6 ปุ่มลอย

1. ค่อนเทนเนอร์
2. ไอคอน

2.5 ฐานข้อมูลไฟร์เบส

ไฟร์เบสถูกสร้างขึ้นจากคุณสมบัติเสริมว่านักพัฒนาสามารถผสมและจับคู่เพื่อให้ดีพอ กับความต้องการของตน บริษัทก่อตั้งขึ้นในปี พ.ศ. 2554 โดยแอนดรูว์และเจมส์ เทมป์ลิน ไฟร์เบส เป็นฐานข้อมูลแบบเรียลไทม์ ซึ่งมีช่องทางการเข้ามาร่วมต่อที่ช่วยให้นักพัฒนาในการจัดเก็บและทำให้ข้อมูล ตรงกัน โดยกูเกิลได้ซื้อกิจการไฟล์เบสและมีการพัฒนาจากบริการหลังบ้าน (Backend) เก็บข้อมูล อย่างเดียว มาเป็นแพลตฟอร์มครบวงจรสำหรับนักพัฒนาแอปพลิเคชัน รองรับบริการแทนทุกอย่างที่ นักพัฒนาแอปพลิเคชันต้องใช้งาน

ฐานข้อมูลไฟร์เบส คือบริการฐานข้อมูลแบบโนอีสควิเอล (NoSQL) ที่จัดเก็บในรูปแบบ เอกสาร (Document) และเป็นแพลตฟอร์มที่รวมเครื่องมือต่างๆ สำหรับการจัดการในส่วนของ บริการหลังบ้าน ซึ่งทำให้สามารถสร้างแอปพลิเคชันบนมือถือได้อย่างมีประสิทธิภาพ

2.5.1 การบริการของไฟร์เบส

2.5.1.1 การยืนยันตัวตน (Authentication)

การยืนยันตัวตนคือบริการตรวจสอบผู้ใช้โดยสามารถตรวจสอบได้หลายวิธี ได้แก่ อีเมล/รหัสผ่าน เบอร์โทรศัพท์ บัญชีกูเกิล เพชบุ๊ค ทวิตเตอร์ (Twitter) กิตฮับ (GitHub) ยะหู (Yahoo) ไมโครซอฟต์ (Microsoft) และเกม เช็นเตอร์ (Game Center) โดยมีฐานข้อมูลเป็นของตัวเองไม่ต้อง

สร้างใหม่หรือออกแบบวิธีการเก็บ สามารถเช็คได้ว่าสมัครสมาชิกด้วยวิธีใด และสามารถเช็คเวลาการเข้าใช้งานล่าสุดได้

2.5.1.2 คลาวด์ ไฟร์สโตร์ (Cloud Firestore)

คลาวด์ ไฟร์สโตร์คือเรียลไทม์ดาต้าเบสรุ่นใหม่ของไฟร์เบส ซึ่งเก็บข้อมูลในรูปแบบโนอสคิว แอลที่จัดเก็บในรูปแบบเอกสาร ใช้วิธีการเก็บข้อมูลแบบเจสัน ทรี (JSON Tree) ขนาดใหญ่และสามารถซิงค์ข้อมูลของผู้ใช้ได้แบบเรียลไทม์ คลาวด์ ไฟร์สโตร์ถูกพัฒนาอยู่บนกูเกิล คลาวด์ แพลตฟอร์ม (Google Cloud Platform) ทำให้สามารถเชื่อมต่อサービスต่างๆ ของกูเกิลและไฟร์เบส ได้อย่างมีประสิทธิภาพ

2.5.1.3 ฐานข้อมูลแบบเรียลไทม์ (Realtime Database)

ฐานข้อมูลแบบเรียลไทม์คือบริการฐานข้อมูลแบบโนอสคิวแอลใช้วิธีการเก็บข้อมูลแบบเจสัน ทรี ขนาดใหญ่และสามารถซิงค์ข้อมูลของผู้ใช้ได้แบบเรียลไทม์

2.5.1.4 คลาวด์ สตอเรจ (Cloud Storage)

คลาวด์ สตอเรจคือบริการเก็บไฟล์รูปภาพ ไฟล์เสียง วิดีโอ เพื่อใช้งานบนแอปพลิเคชัน

2.5.1.5 โฮสติ้ง (Hosting)

เป็นบริการฝากไฟล์ (Static) เช่น เอชทีเอ็มแอล (HTML) ซีอีเอสเอส (CSS) จา瓦สคริปต์ (JavaScript) (ไม่รองรับพีเอชพี (PHP)) เพื่อให้คนอื่นๆ สามารถเข้าใช้งานเว็บไซต์ได้

2.5.1.6 คลาวด์ พังก์ชัน (Cloud Functions)

เป็นบริการสำหรับดีเพลโลย พังก์ชัน (Deploy Function) ที่พัฒนาด้วยจาวาสคริปต์หรือ ไทร์สคริปต์ (TypeScript) เพื่อทำงานตามทริกเกอร์ (Trigger) ที่เกิดขึ้นบนไฟร์เบส

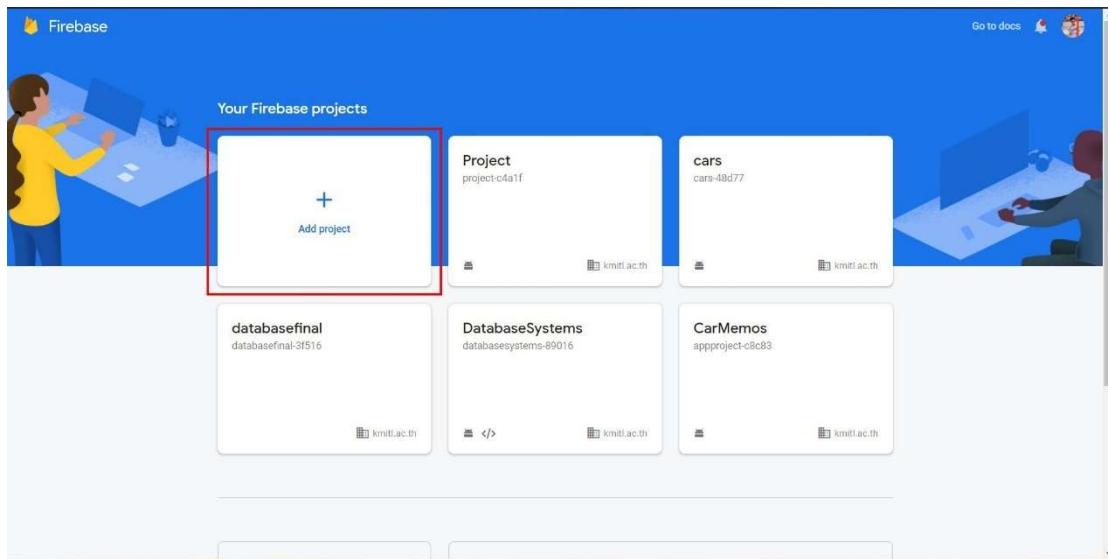
2.5.1.7 การพัฒนาธุรกิจ (Grow your business) มีทั้งหมด 7 ได้แก่

1. อินแอปเมจเจจิ้ง (In-App Messaging) คือบริการแสดงข้อความแบบเด้งภายในแอปพลิเคชัน
2. กูเกิล อนาไลน์ติกส์ (Google Analytics) คือบริการแสดงข้อมูลสถิติต่าง ๆ เพื่อวิเคราะห์กลุ่มเป้าหมาย หรือรับทราบพฤติกรรมของผู้ใช้งานต่าง ๆ
3. พรีดิกชัน (Predictions) คือบริการวิเคราะห์ข้อมูลการใช้งานแอปพลิเคชัน
4. คลาวด์ เมสเจจิ้ง (Cloud Messaging) คือบริการส่งการแจ้งเตือนไปยังแอปพลิเคชัน
5. รีโมท คอนฟิก (Remote Config) คือความสามารถที่จะเปลี่ยนลักษณะการทำงานและลักษณะที่ปรากฏของแอปพลิเคชันได้ทันทีจากหน้าเว็บไฟร์เบสโดยไม่ต้องรอการอนุมัติจากแอปสโตร์ (App Store)

6. ไดนามิก ลิงค์ (Dynamic Links) คือลิงค์เชื่อมโยงไปยังแอปพลิเคชัน ใช้สำหรับแสดงบนหน้าเว็บเพื่อให้ผู้ใช้งานติดตั้งแอปพลิเคชันผ่านลิงค์
7. เอปี เทสติ้ง (A/B Testing) คือความสามารถในการแสดงผลแอปหลายรูปแบบ เพื่อทดสอบการแสดงผลหรือการทำงาน

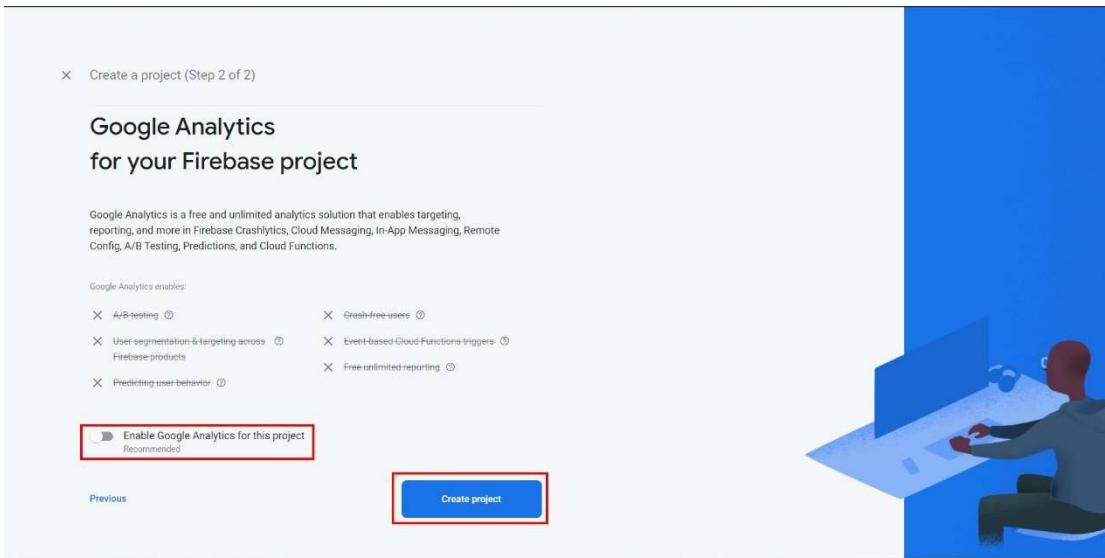
2.5.2 การเริ่มต้นใช้งานไฟร์เบส

1. เปิดเว็บไซต์ <https://console.firebaseio.google.com/>
2. เข้าสู่ระบบโดยใช้บัญชีกิจกรรม
3. คลิกที่ปุ่ม Add project ถ้าใช้แบบพรีสามารถสร้างโครงการได้ 5 โครงการเท่านั้น



รูปที่ 2.7 การเพิ่มโครงการของไฟร์เบส

4. ตั้งชื่อโครงการและคลิกที่ปุ่ม Continue
5. ปิด Google Analytics และคลิกที่ปุ่ม Create project



รูปที่ 2.8 การสร้างโครงของไฟร์เบส

2.6 เครื่องมือที่ใช้ในการพัฒนาแอปพลิเคชัน

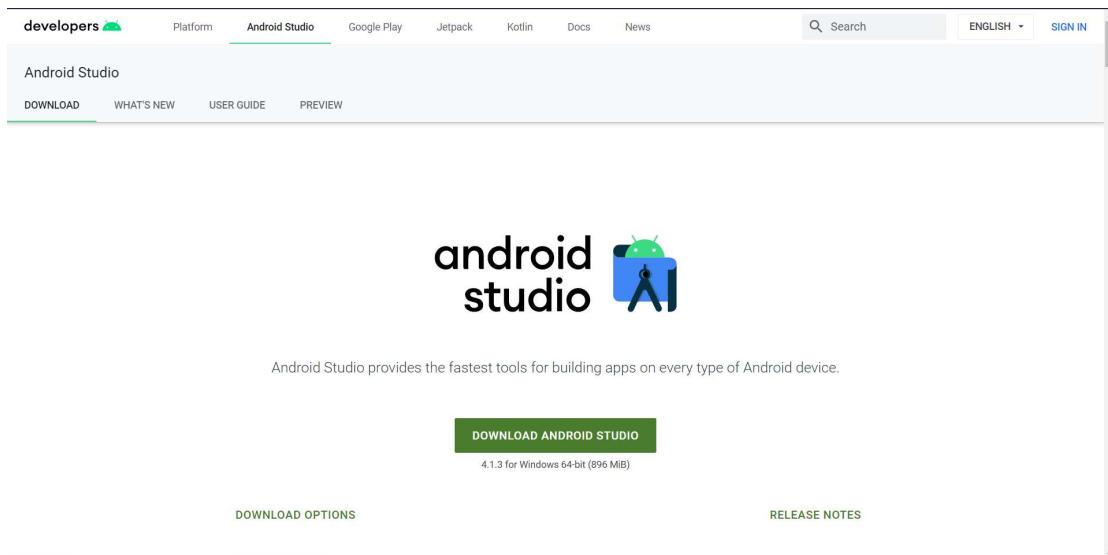
2.6.1 แอนดรอยด์ สตูดิโอ

แอนดรอยด์ สตูดิโอ เป็นเครื่องมือพัฒนา (IDE : Integrated Development Environment) เป็นซอฟต์แวร์สำหรับการทำแอปพลิเคชันบนแอนดรอยด์แบบเนทีฟ (Native) คือเป็นการทำงานที่ออกแบบมาเฉพาะสำหรับแอปพลิเคชันบนแอนดรอยด์ ทำให้มีการทำงานที่รวดเร็วกว่าแบบไฮบริด (Hybrid) ที่จะมีการทำงานที่ซ้ำกัน และมีการดึงประสิทธิภาพของเครื่องออกมากได้ทั้งหมด ทั้ง เช่นเซอร์และระบบกราฟิก ช่วยให้สามารถแสดงตัวอย่างแอปพลิเคชัน ในมุมมองที่แตกต่างกัน บน สมาร์ทโฟนแต่ละรุ่น สามารถแสดงผลได้ทันทีโดยไม่ต้องทำการรันแอปพลิเคชันบน อีミュเลเตอร์ (Emulator) การเขียนจะใช้ภาษา จาวา (Java) หรือ kotlin (Kotlin) โดยภาษาจาวาจะเป็นภาษา หลัก ส่วน kotlin จะเป็นภาษาที่ทางกูเกิล กำลังนำมาใช้ใหม่ เพื่อทดแทนข้อด้อยของจาวาในบางเรื่อง

2.6.1.1 การติดตั้งแอนดรอยด์ สตูดิโอ

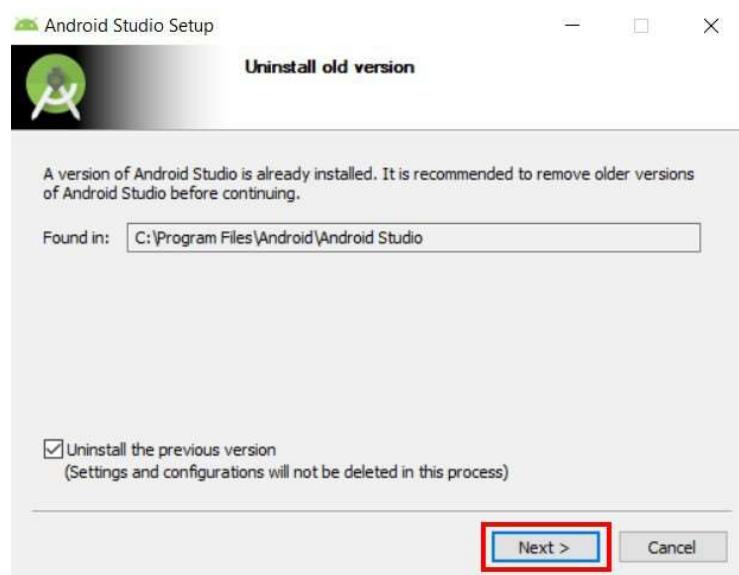
- สามารถดาวน์โหลดได้ที่

<http://developer.android.com/sdk/installing/studio.html> จากนั้นคลิกที่ DOWNLOAD ANDROID STUDIO



รูปที่ 2.9 หน้าดาวน์โหลดแอปพลิเคชัน Android Studio

2. เมื่อดาวน์โหลดเสร็จแล้ว ดับเบิลคลิกที่ไฟล์ คลิกที่ปุ่ม Next

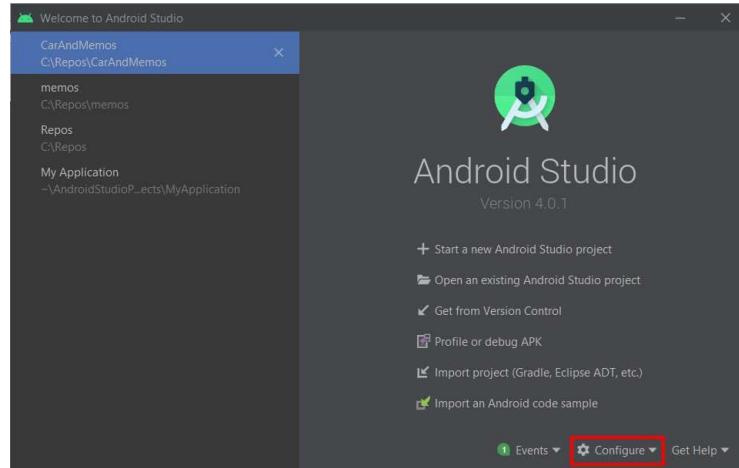


รูปที่ 2.10 หน้าติดตั้งแอปพลิเคชัน Android Studio

3. เมื่อติดตั้งเสร็จแล้วกด Close

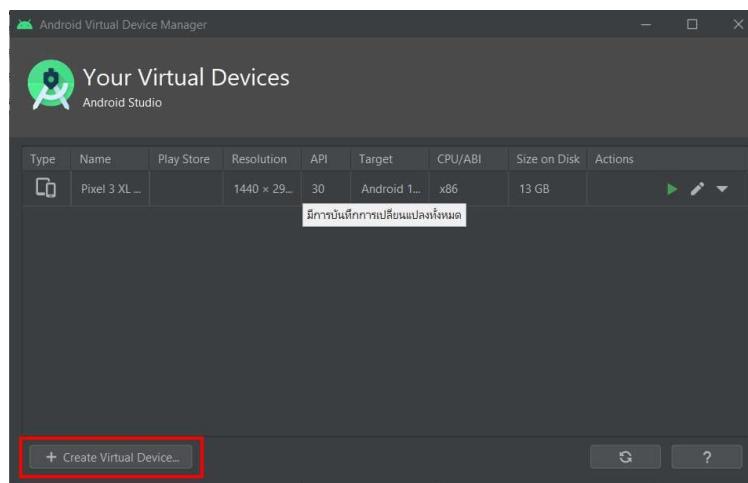
2.6.1.2 การสร้างอีมูเลเตอร์ เพื่อใช้ในการพัฒนาแอปพลิเคชัน

1. เปิดโปรแกรมแอนดรอยด์ สตูดิโอ คลิกที่ Configure เลือก AVD Manager



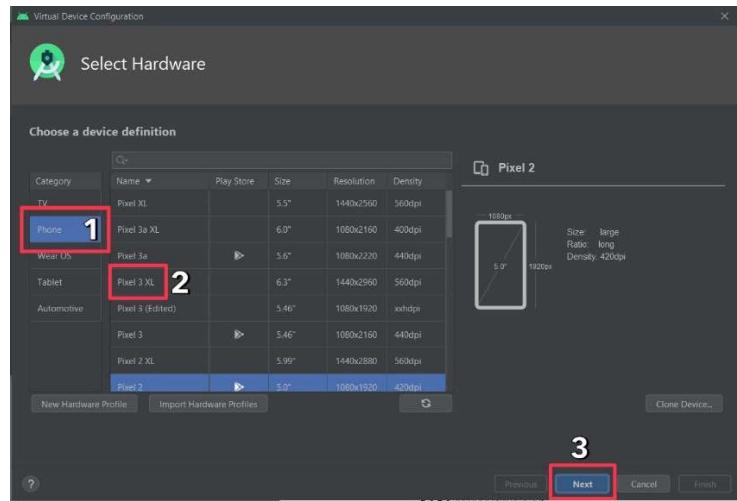
รูปที่ 2.11 หน้าเริ่มต้นของแอนดรอยด์ สตูดิโอ

2. คลิกที่ Create Virtual Device



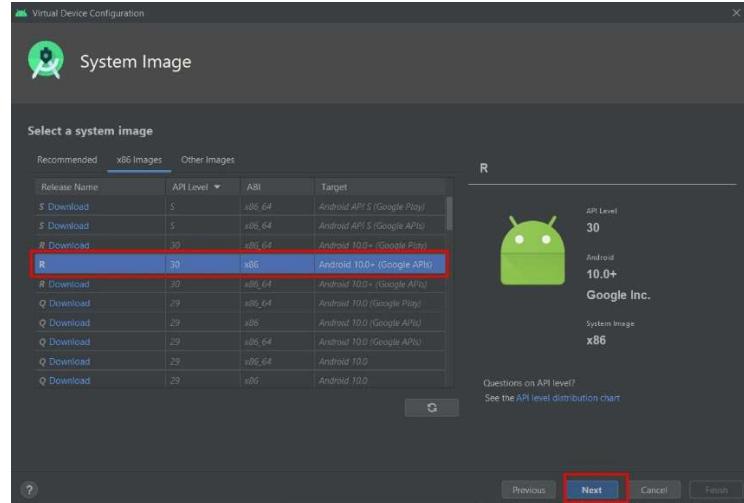
รูปที่ 2.12 หน้าการสร้างอีมูเลเตอร์

3. เลือกชนิดของอีมูลเลเตอร์ จากนั้นกด Next



รูปที่ 2.13 เลือกอุปกรณ์สำหรับอีเมลเตอร์

4. เลือกระบบปฏิบัติการแอนดรอยด์ที่ต้องการใช้ในการพัฒนาแอปพลิเคชัน จากนั้นกด Next



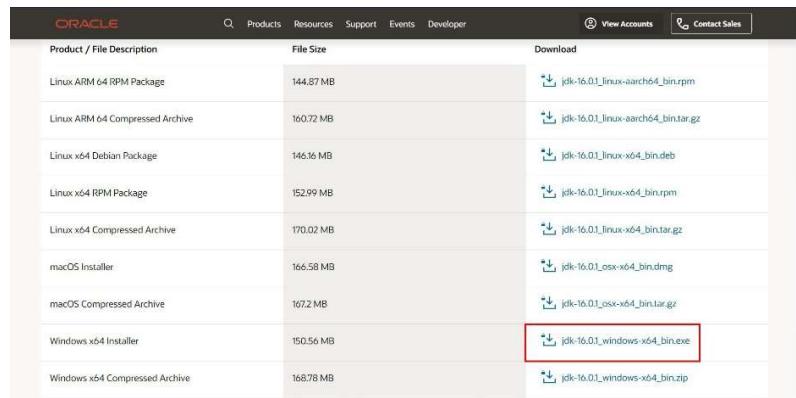
รูปที่ 2.14 เลือกรุ่นของแอนดรอยด์

5. เมื่อดาวน์โหลดและติดตั้งเสร็จเรียบร้อยแล้ว คลิกที่ Finish

2.6.1.3 การดาวน์โหลดและติดตั้งเจดีเค (JDK)

- สามารถดาวน์โหลดเจดีเคเวอร์ชันล่าสุดได้จาก

<https://www.oracle.com/java/technologies/javase-jdk16-downloads.html>



Product / File Description	File Size	Download
Linux ARM 64 RPM Package	144.87 MB	jdk-16.0.1_linux-aarch64_bin.rpm
Linux ARM 64 Compressed Archive	160.72 MB	jdk-16.0.1_linux-aarch64_bin.tar.gz
Linux x64 Debian Package	146.36 MB	jdk-16.0.1_linux-x64_bin.deb
Linux x64 RPM Package	152.99 MB	jdk-16.0.1_linux-x64_bin.rpm
Linux x64 Compressed Archive	170.02 MB	jdk-16.0.1_linux-x64_bin.tar.gz
macOS Installer	166.58 MB	jdk-16.0.1_osx-x64_bin.dmg
macOS Compressed Archive	167.2 MB	jdk-16.0.1_osx-x64_bin.tar.gz
Windows x64 Installer	150.56 MB	jdk-16.0.1_windows-x64_bin.exe
Windows x64 Compressed Archive	168.78 MB	jdk-16.0.1_windows-x64_bin.zip

รูปที่ 2.15 หน้าต่างดาวน์โหลดเจดีเค

- เมื่อดาวน์โหลดเสร็จแล้ว ให้ดับเบิลคลิกไฟล์เพื่อทำการติดตั้ง คลิกที่ Install

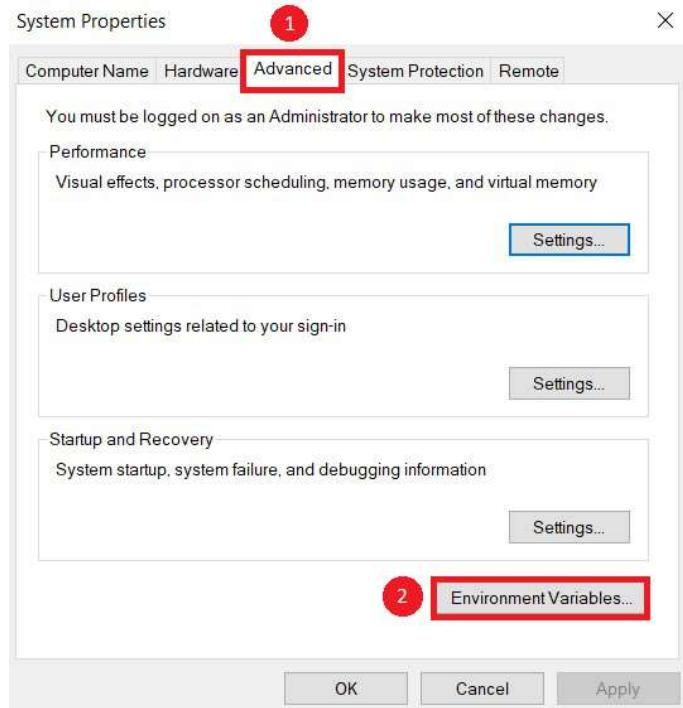


รูปที่ 2.16 หน้าต่างการติดตั้งเจดีเค

- เมื่อติดตั้งเสร็จแล้วให้กด Close

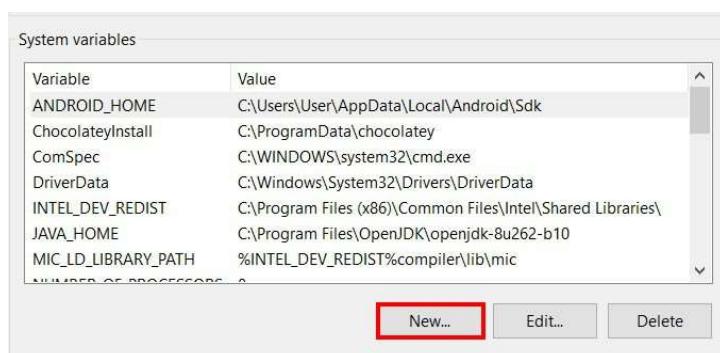
2.6.1.4 การกำหนดเส้นทาง (Part) ให้แอนดรอยด์ สตูดิโอและเจดีเค

1. คลิกขวาที่พีซีเครื่องนี้ (This pc) เลือกคุณสมบัติ (Properties)
2. เลือกตัวเลือกขั้นสูง (Advanced system settings)
3. ไปที่แท็บ Advanced ให้คลิกเข้าไปที่ Environment Variables...



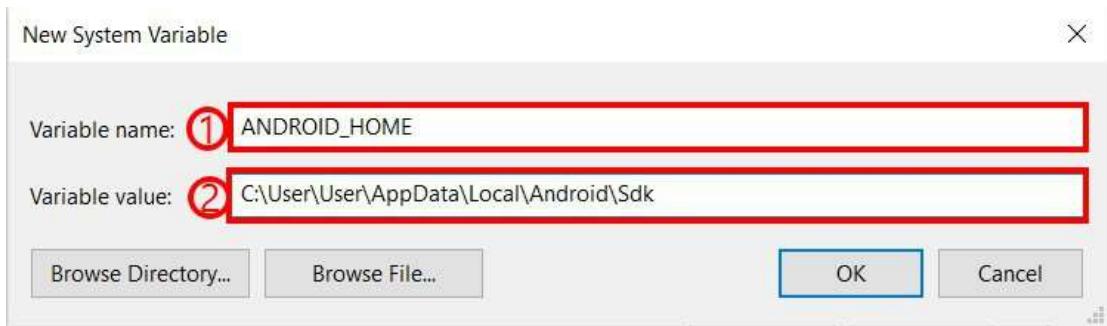
รูปที่ 2.17 หน้าต่างการกำหนดเส้นทางให้ให้แอนดรอยด์ สตูดิโอและเจดีเค

4. คลิกปุ่ม New... ตรงส่วน System variables



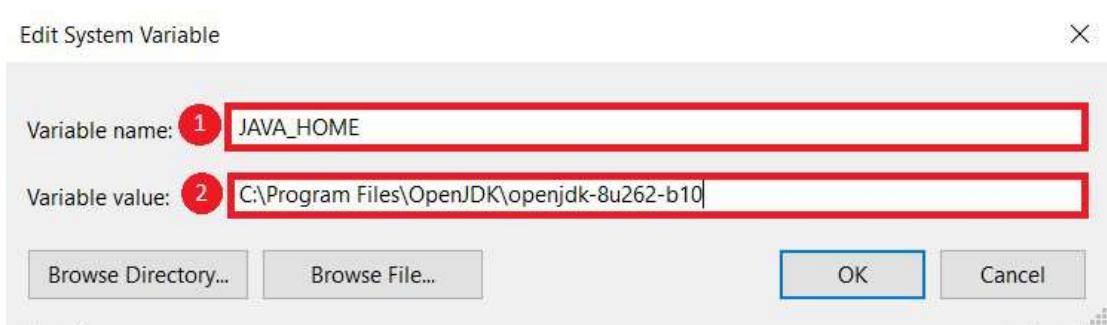
รูปที่ 2.18 หน้าต่างการกำหนดเส้นทางชิสเต็มว่าเรียເອເປີລເຈົ້າ

5. กำหนด Variable name เป็น ANDROID_HOME และกำหนด Variable value เป็นเส้นทางที่อยู่ของแอนดรอยด์อสเดี๋ย จากนั้นคลิกปุ่ม OK



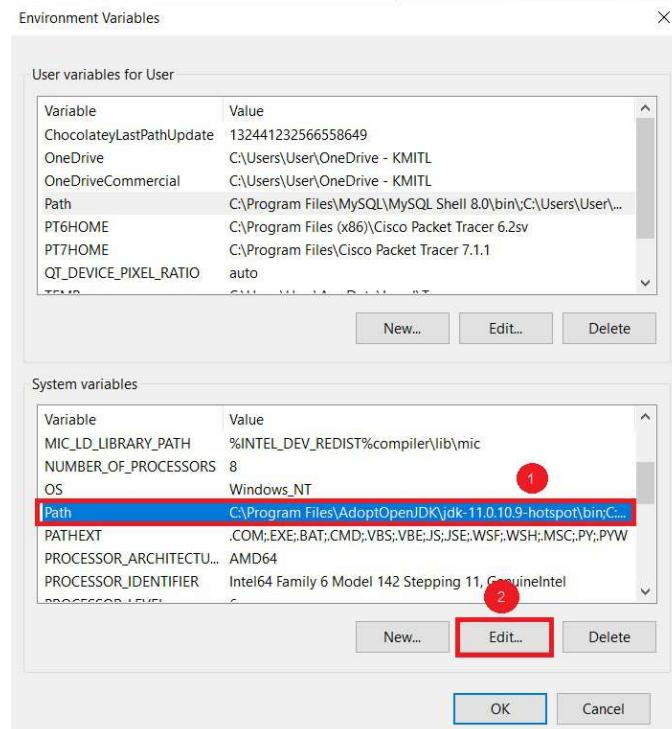
รูปที่ 2.19 หน้าต่างกำหนดตัวแปรให้แอนดรอยด์

6. คลิกที่ปุ่ม New... ตรงส่วน System variables กำหนด Variable name เป็น JAVA_HOME และกำหนด Variable value เป็นเส้นทางที่ตั้งเจดีจากนั้นคลิกปุ่ม OK



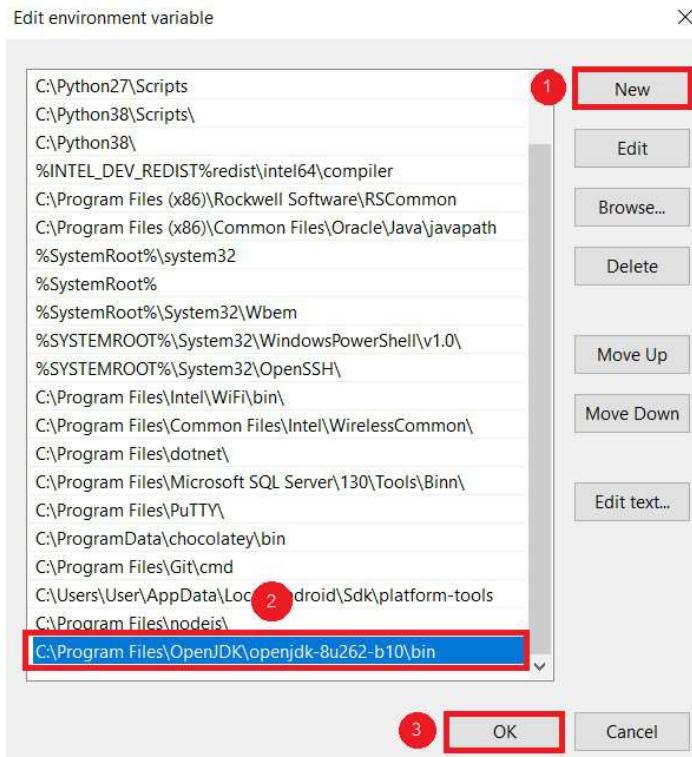
รูปที่ 2.20 หน้าต่างการกำหนดชื่อตัวแปรให้เจดีคิ

7. คลิกไปที่เส้นทางจากนั้นคลิกปุ่ม Edit...



รูปที่ 2.21 หน้าต่างการแก้ไขเส้นทางให้เจดีเค

8. คลิกไปที่ New... เข้าไปในส่วนที่ติดตั้งเจดีเคแล้วเข้าไปในโฟลเดอร์ bin และคัดลอกเส้นทางมาใส่ จากนั้นกด OK



รูปที่ 2.22 หน้าต่างการกำหนดเส้นทางสภาพแวดล้อมให้เจดีเค

2.6.2 โหนดเจอส

โหนดเจอส คือ จา瓦สคริปต์ รันท์ไทม์ (JavaScript Runtime) ที่ถูกสร้างโดยโครม วีแอปด จา瓦สคริปต์ เอนจีน (Chrome V8 JavaScript Engine) ที่ทำงานอยู่ผ่านบริการหลังบ้านเป็นเว็บ เซิร์ฟเวอร์ (Web Server) จากเดิมที่เคยอยู่ผ่านหน้าบ้าน (Frontend) ทำหน้าที่ร่วมกับกับเซิร์ฟเวอร์ แอล ใบปัจุบันมีผู้พัฒนาจำนวนมากเป็นโหนดเจอสและยังเป็นครอสแพลตฟอร์ม (Cross Platform Environment) และโอเพ่นซอร์ส (Open Source)

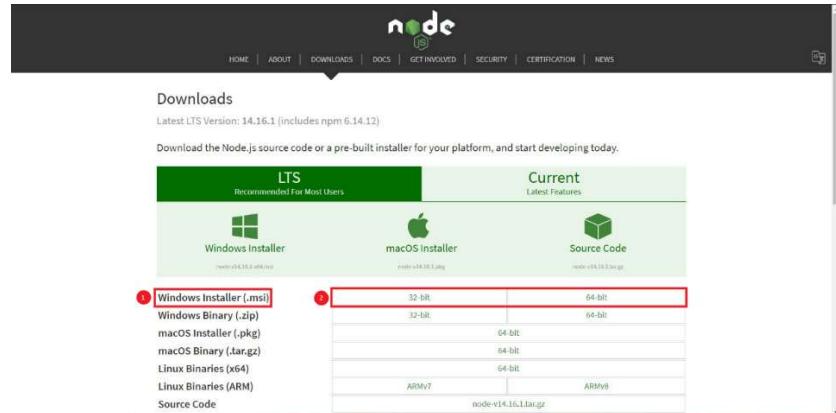
2.6.2.1 การติดตั้งโหนดเจอส

การติดตั้งโหนดเจอสเพื่อที่จะเรียกใช้แพ็กเกจของรีแอค เนทีฟผ่านคำสั่งเอ็นพีเอ็ม (npm)

- ดาวน์โหลดโหนดเจอส สำหรับวินโดว์สามารถดาวน์โหลดได้ที่

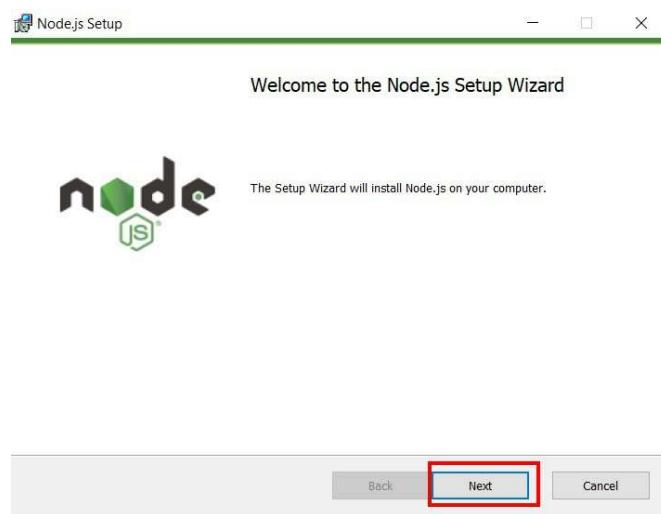
<https://nodejs.org/en/download/> จากนั้นไปที่ Windows Installer (.msi)

แล้วเลือก Windows ให้ตรงตามสเปคเครื่องคอมพิวเตอร์



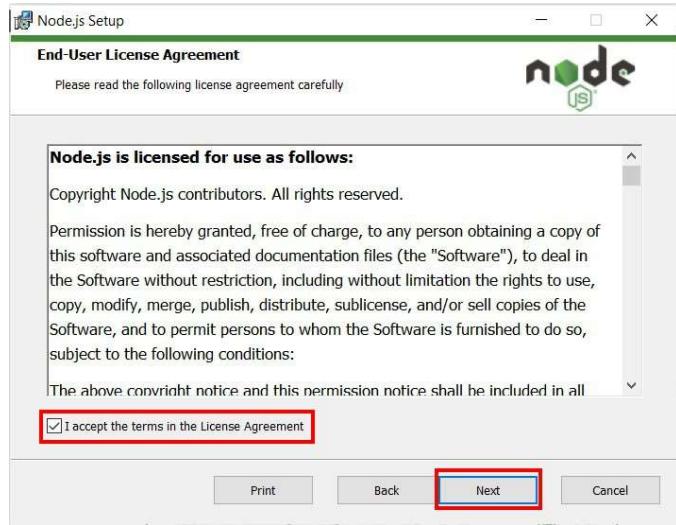
รูปที่ 2.23 หน้าต่างดาวน์โหลดโหนดเจอส

2. เมื่อดาวน์โหลดเสร็จแล้วให้ดับเบิลคลิกที่ไฟล์ node-v14.16.1-x64.msi จากนั้นกด Next



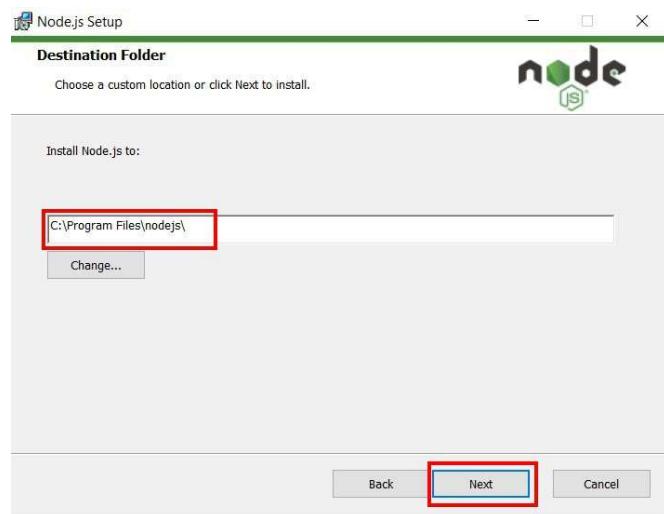
รูปที่ 2.24 หน้าต่างการติดตั้งโหนดเจอส

3. เลือก I accept the terms in the License Agreement จากนั้นกดปุ่ม Next



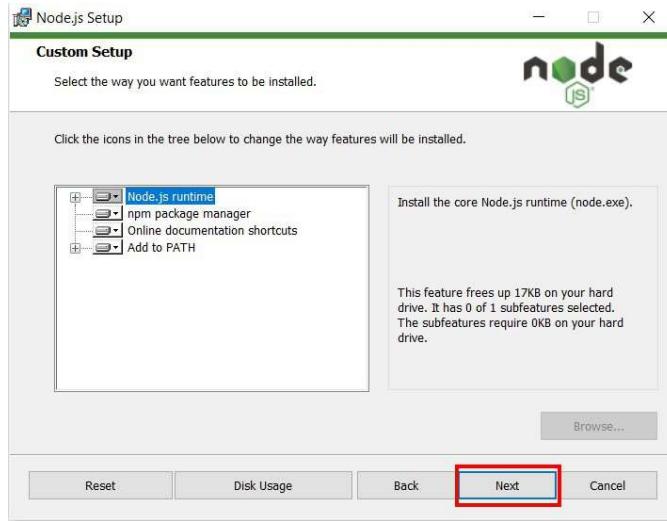
รูปที่ 2.25 หน้าต่างข้อกำหนดในการติดตั้งnodejs

4. เลือกตำแหน่งที่ต้องการติดตั้ง จากนั้นกดปุ่ม Next



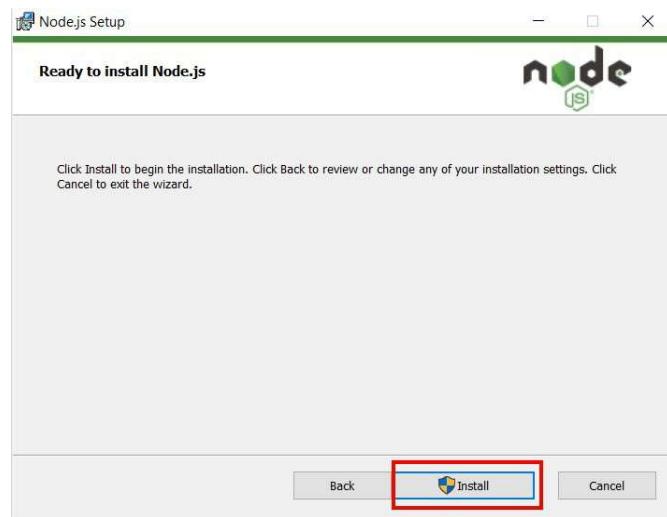
รูปที่ 2.26 หน้าต่างการเลือกที่อยู่การติดตั้งnodejs

5. เลือกคุณสมบัติที่ต้องการติดตั้ง จากนั้นคลิกที่ปุ่ม Next



รูปที่ 2.27 หน้าต่างการติดตั้งแบบเลือกเอง

6. คลิกที่ปุ่ม Install เพื่อทำการติดตั้ง



รูปที่ 2.28 หน้าต่างยืนยันการติดตั้งโนนดเจอส

7. เมื่อติดตั้งเสร็จเรียบร้อยแล้วให้คลิกที่ปุ่ม Finish

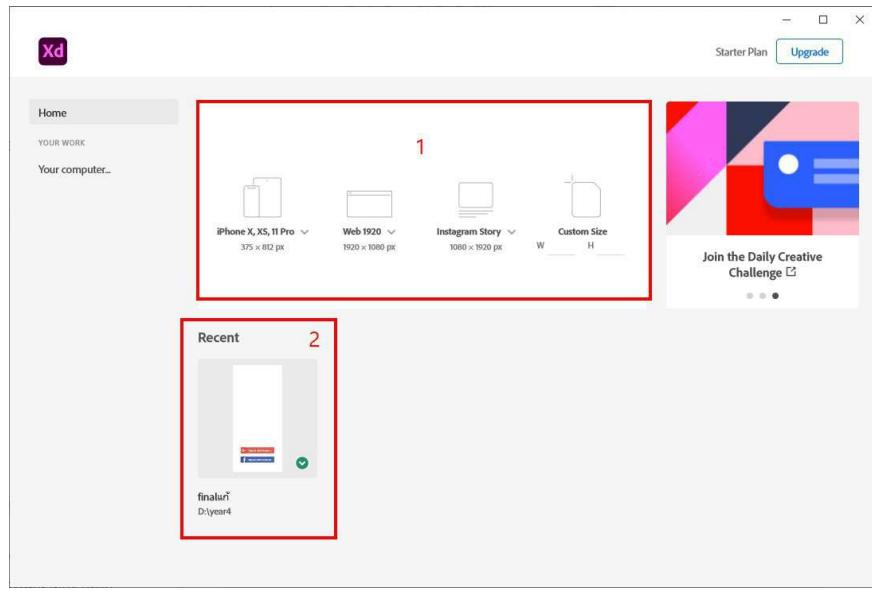
2.6.3 อะโดบี เอ็กซ์ดี

อะโดบี เอ็กซ์ดีหรืออะโดบี เอ็กซ์พีเรียนซ์ ดีไซน์ (Adobe Experience Design) เป็นเครื่องมือที่ใช้สำหรับการออกแบบการทำงานในส่วนประสบการณ์ผู้ใช้งาน สามารถใช้งานได้ฟรีและรองรับทั้งในวินโดวส์ (Windows) และแมคโออีส (MacOS)

ชิ่งอะโดบี เอ็กซ์ดีมีฟีเจอร์ทั้งการออกแบบหน้าตา การเชื่อมหน้า และการส่งต่องานชิ่งในการออกแบบจะมีเครื่องมือที่ช่วยในการออกแบบทำให้สะดวกและรวดเร็วมากขึ้น ในการออกแบบสามารถเลือกขนาดของงานมีให้เลือกหลากหลายหรือจะกำหนดขนาดเองก็สามารถทำได้ มีญี่โอดิต (UI Kit) ซึ่งสามารถใช้ได้ฟรีซึ่งมีหลากหลายแบบให้เลือกใช้มี แอสเซ็ต (Assets) เป็นพังก์ชันในการเก็บส่วนประกอบหรือสไตร์ที่มีการใช้งานชำนาญ นำมาใช้งานได้มี ปลั๊กอิน (Plugin) ในการทำงานมากมายมีตัวต้นแบบ (Prototype) ในการเชื่อมหน้ากันทำให้ในการจำลองการใช้งานแอปพลิเคชันได้สมมุติจริงและสามารถแชร์ไฟล์ร่วมกันทำงานกับทีมได้ผ่านอะโดบี คลาวด์ (Adobe Cloud) โดยสามารถดูประวัติการแก้ไขได้สูงสุด 30 วัน

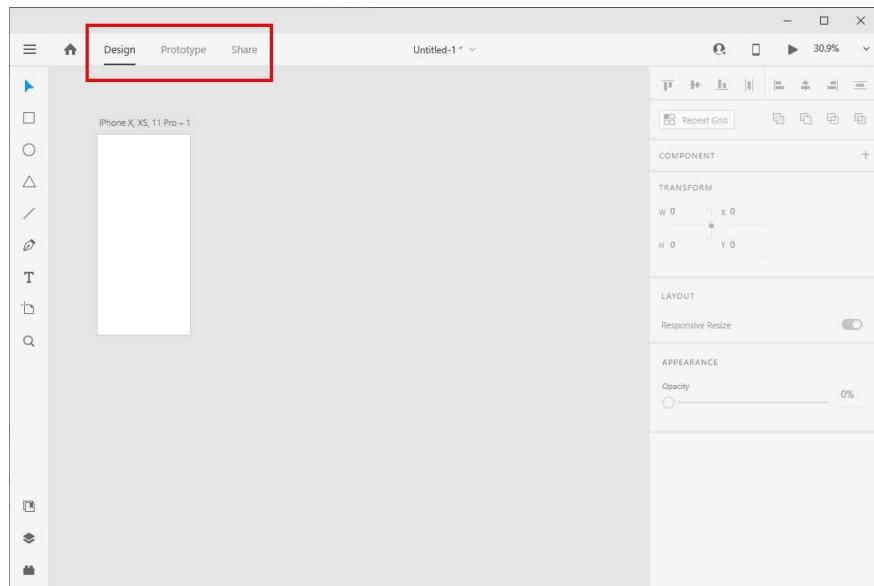
2.6.3.1 เริ่มต้นในอะโดบี เอ็กซ์ดี

1. เลือกขนาดของพื้นที่ในการทำงาน ผู้ใช้งานจะสามารถเลือกขนาดของพื้นที่งานได้โดยจะมีขนาดที่โปรแกรมได้ระบุมาให้กับขนาดที่ผู้ใช้งานสามารถกำหนดขึ้นมาได้ด้วยตนเอง โดยการเลือกขนาดจะอยู่ในจุดที่ 1 ของรูปที่ 2.29 และสามารถเปิดไฟล์งานที่ผู้ใช้ได้ทำการใช้งานไปล่าสุดได้ เมื่อผู้ใช้งานได้ทำการเลือกไฟล์งาน โปรแกรมจะทำการเปิดไฟล์งาน ผู้ใช้งานจะสามารถเปิดไฟล์ที่เคยใช้งานล่าสุดได้ในจุดที่ 2 ของรูปที่ 2.29



รูปที่ 2.29 การเลือกขนาดของพื้นที่งาน

2. เมื่อทำการเลือกขนาดพื้นที่งาน โปรแกรมจะนำทางผู้ใช้งานมาสู่พื้นที่ในการทำงานภายในหน้าจอจะแบ่งพังก์ชันการทำงานออกเป็น 3 โหมด คือ โหมดออกแบบ (Design) โหมดเชื่อมประสาน (Prototype) และการแชร์ ดังรูปที่ 2.30



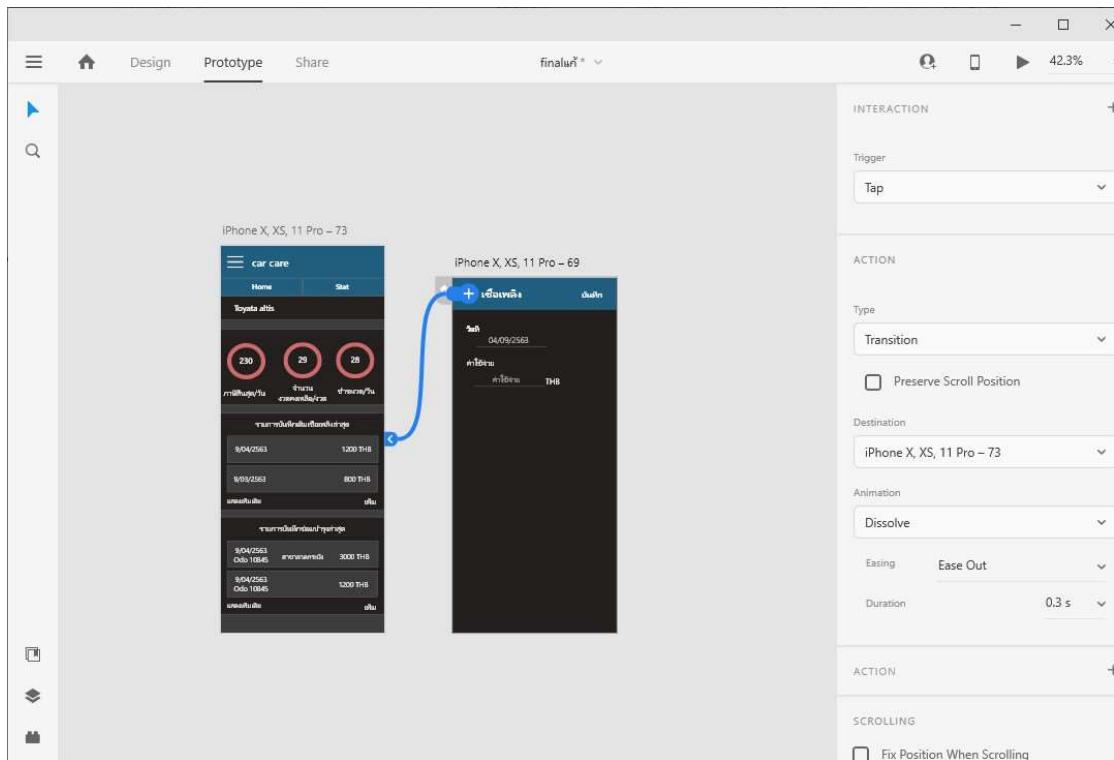
รูปที่ 2.30 โหมดการทำงานของอะโดบี เอ็กซ์ดี

3. ในหน้าจอของໂທມດອກແບບ ຜູ້ໃຊ້ຈານສາມາຮັດໃຫ້ເຄີ່ງມືອຕ່າງທີ່ໂປຣແກຣມມີໄວ້ໃຫ້ເດີ ດັ່ງນີ້
- 3.1 ລູກศຽເລືອກ ໃຫ້ໃນການຈັບວັດຖຸຕ່າງໆ
 - 3.2 ສື່ເໝີຍມ ໃຫ້ສໍາຫັບກາຣວາດສື່ເໝີຍມ
 - 3.3 ວົກລົມ ໃຫ້ສໍາຫັບກາຣວາດຮູປ່ງກລມ
 - 3.4 ສາມເໝີຍມ ໃຫ້ສໍາຫັບກາຣວາດຮູປ່ງສາມເໝີຍມ
 - 3.5 ເສັ້ນຕຽງ ໃຫ້ສໍາຫັບກາຣວາດເສັ້ນ
 - 3.6 ປາກກາ ໃຫ້ໃນກາຣວາດເສັ້ນ ວັດຖາມທີ່ຜູ້ໃຊ້ຈານຕ້ອງກາຣ
 - 3.7 ເທິກ່ານ ໃຫ້ໃນກາຣສ້າງຕ້ວອັກຊະຣ
 - 3.8 ອາຮົກທອບອົບ (Artboard) ໃຫ້ກາຣເພີ່ມພື້ນທີ່ງານ
 - 3.9 ແວ່ນຂໍາຍາຍ ໃຫ້ໃນກາຣຂໍາຍາຍເຂົ້າ-ອອກ



ຮູບທີ 2.31 ເຄີ່ງມືອໃນອະໂດປີ ເວັກຊົດ

4. การทำเชื่อมหน้า (Prototyping) เป็นการตั้งค่าการเชื่อมหน้าซึ่งในการเชื่อมหน้าจะทำได้ก็ต่อเมื่อมีการสร้างหน้าอย่างน้อย 2 หน้าโดยมีวิธีการเชื่อมหน้าดังนี้
- 4.1 เลือกวัตถุที่ต้องการเมื่อกดให้เปลี่ยนหน้า
 - 4.2 ดึงเส้นจากวัตถุไปยังหน้าที่ต้องการเชื่อม



รูปที่ 2.32 ตัวอย่างการเชื่อมหน้า

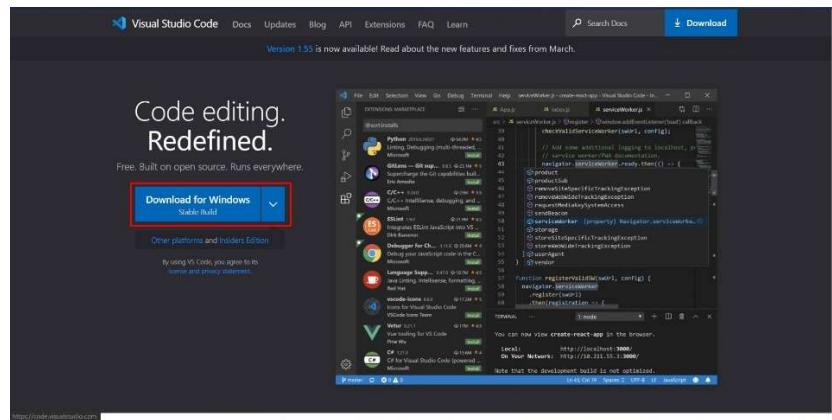
2.6.4 วิชาล สตูดิโอ โค้ด

วิชาล สตูดิโอ โค้ดเป็นโปรแกรมแก๊กโค้ด (Code Editor) ที่ใช้ในการแก้ไขและปรับแต่งโค้ดโดยมาจากการค่ายไมโครซอฟท์ ที่มีการพัฒนาออกมาในรูปแบบของโอเพ่นซอร์สจึงสามารถนำมาใช้งานได้แบบฟรีวิชาล สตูดิโอ โค้ด เหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานกับแพลตฟอร์มมีการรองรับการใช้งานทั้งบนวินโดวส์ แมคโออีส และลีนุก (Linux) มีการสนับสนุนทั้งภาษาจาวาสคริปต์ ไทร์สคริปต์ และโหนดเจอสามารถเชื่อมต่อกับกิต (Git) สามารถนำมาใช้งานได้ง่ายไม่ซับซ้อน มีเครื่องมือส่วนขยายต่าง ๆ ให้เลือกใช้อย่างมาก สามารถเปิดใช้งานภาษาอื่น ๆ ทั้ง ภาษาซีพลัสพลัส (C++) ซีชาร์ป(C#) จาวา ไฟรอน (Python) พีเอชพี หรือโก (Go)

2.6.4.1 การติดตั้งวิชาล สตูดิโอ โค้ด แก๊งโค้ด

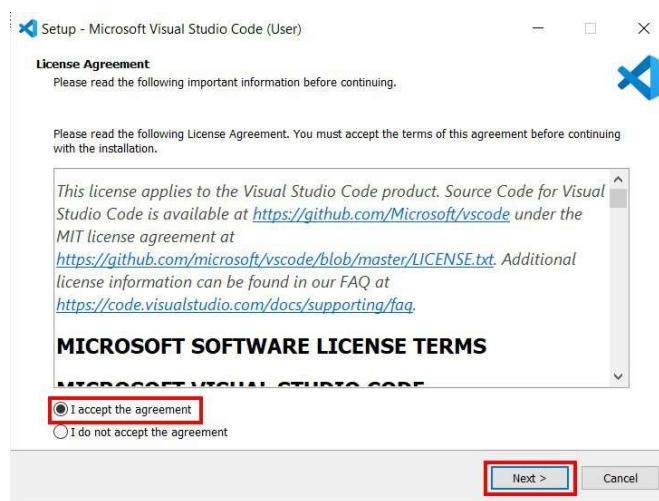
ติดตั้งวิชาล สตูดิโอ โค้ด เพื่อใช้เป็นโปรแกรมแก๊งโค้ดในการพัฒนาแอปพลิเคชัน

- สามารถดาวน์โหลดวิชาล สตูดิโอ โค้ด ได้ที่ <https://code.visualstudio.com/>
คลิกที่ Download for Windows



รูปที่ 2.33 หน้าดาวน์โหลดวิชาล สตูดิโอ โค้ด

- เมื่อดาวน์โหลดเสร็จแล้ว ดับเบิลคลิกที่ไฟล์ VSCodeUserSetup-x64-1.55.2 เลือก I accept agreement และคลิกที่ปุ่ม Next



รูปที่ 2.34 หน้าต่างข้อกำหนดในการติดตั้งวิชาล สตูดิโอ โค้ด

3. เลือกตำแหน่งที่ต้องการติดตั้งวิชวล สตูดิโอ โคด์ จากนั้นกดปุ่ม Next
4. เมื่อติดตั้งเสร็จแล้ว คลิกที่ Finish

2.6.5 รีแอค เนทีฟ

การสร้างแอปพลิเคชัน แบบครอสแพลตฟอร์มกำลังเป็นที่นิยมด้วยเฟรมเวิร์ค (Framework) ที่สามารถสร้างเนทีฟ แอป (Native App) โดยไม่ต้องใช้ภาษาswift (Swift) สำหรับไอโอเอช (iOS) หรือjava สำหรับแอนดรอยด์ (Android)

รีแอค เนทีฟเป็น จา瓦สคริปต์ เฟรมเวิร์ค (JavaScript Framework) สำหรับพัฒนาแอป พลิเคชันบนสมาร์ทโฟนที่สร้างโดยบริษัทเฟซบุ๊ก โดยเริ่มแรกถูกพัฒนาขึ้นเพื่อแก้ปัญหาในการใช้งานเฟซบุ๊กบนสมาร์ทโฟนที่สามารถใช้งานบนเว็บбраузரได้แต่มีความช้าและไม่เป็นมิตรต่อผู้ใช้งานรีแอค เนทีฟจึงถูกพัฒนามาเพื่อแก้ไขปัญหานี้และพัฒนาขึ้นมาจนเป็นเฟรมเวิร์คที่สามารถพัฒนาแบบครอสแพลตฟอร์มได้ โดยการนำส่วนประกอบ (Component) ของรีแอค เนทีฟไปแสดงผล เนทีฟ ยู ไอ (Native UI) บนไอโอเอชและแอนดรอยด์ทำให้นักพัฒนาที่สร้างเว็บด้วยรีแอคอยู่แล้วสามารถแปลงโครงการให้สามารถใช้งานบนมือถือได้

ข้อดีของรีแอค เนทีฟเป็นการใช้งานข้ามแพลตฟอร์ม เริ่มแรกคิดคันเพื่อการพัฒนาไอโอเอช, รีแอค เนทีฟ เฟรมเวิร์ค เน้นถึงความนิยมและความสำเร็จ ดังนั้นเฟซบุ๊กจึงตัดสินใจที่จะให้การสนับสนุนแพลตฟอร์มแอนดรอยด์ในภายหลัง ดังนั้นแอปพลิเคชันรีแอค เนทีฟสามารถพัฒนาผ่านการใช้โค้ดเบส (Codebase) เดียวสำหรับทั้งสองแพลตฟอร์มพร้อมกันทำให้มีประสิทธิภาพในการทำงานสูงและประหยัดพลังงาน ของการใช้รีแอค เนทีฟ เฟรมเวิร์คร่วมถึงการพัฒนาที่รวดเร็วขึ้น ความสามารถในการนำองค์ประกอบกลับมาใช้ใหม่และง่ายต่อการดูแลรักษา ดังนั้นอุตสาหกรรมหรือบุคคลที่ว่าไบในปัจจุบันจึงนิยมใช้รีแอค เนทีฟเพื่อสร้างแอปพลิเคชันบนมือถือ

ข้อเสียรีแอค เนทีฟต้องการปรับบางส่วนจากแพ็คเกจส่วนกลาง กรอบการทำงานของรีแอค (React) นั้นไม่สมบูรณ์แบบในตอนนี้และมีข้อบกพร่องบางอย่างที่ยังไม่ได้รับการแก้ไขโดยตรงที่กำหนดเองบางส่วนในเฟรมเวิร์คหายไปทำให้เสียเวลาในการสร้างโมดูล

2.6.5.1 ภาษา (Language)

รีแอค เนทีฟถูกเขียนทั้งหมดด้วยจาวาสคริปต์เป็นข้อดีสำหรับรีแอค เนทีฟเนื่องจากจาวасคริปต์จัดเป็นหนึ่งในภาษาที่ได้รับความนิยมและผู้คนนิยมใช้เขียนโปรแกรมมากที่สุดในโลก การหากนักพัฒนาสำหรับจาวาสคริปต์เป็นเรื่องที่ง่าย ดังนั้นการเรียนรู้รีแอค เนทีฟจึงเป็นเรื่องค่อนข้างง่าย สำหรับผู้พัฒนาจาวาสคริปต์

2.6.5.2 บริษัทที่สนับสนุน (Corporate Backing)

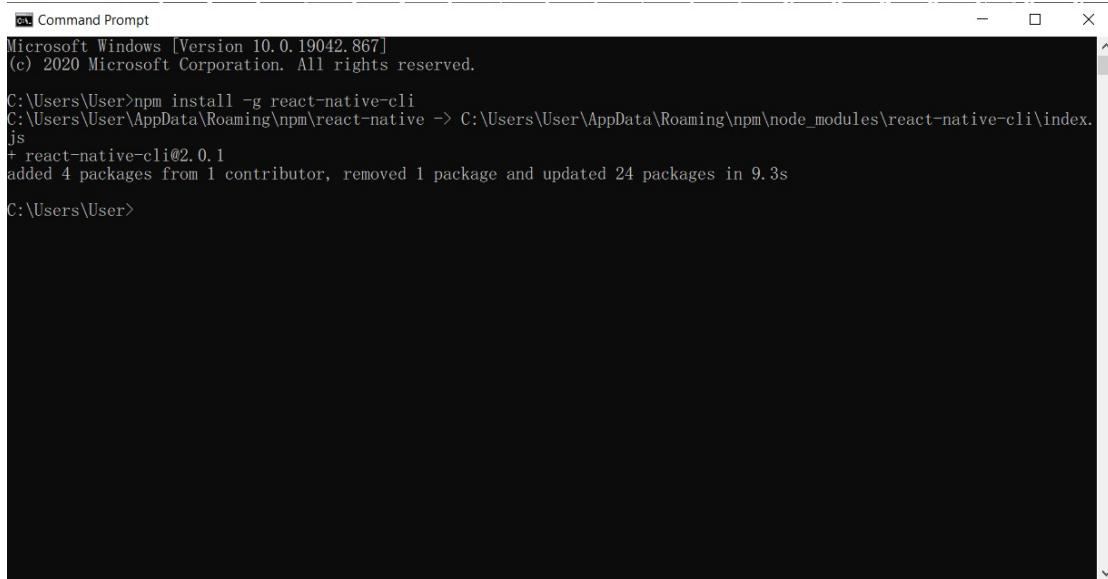
รีแอค เนทีฟถูกพัฒนาโดยเฟซบุ๊กและได้เปิดเผยแพร่รหัสต้นฉบับ (Source code) ในปี 2015 เพื่อให้มีวิศวกรคุณภาพที่ทำงานเกี่ยวกับรีแอค เนทีฟและได้พัฒนาให้ดีขึ้นอย่างต่อเนื่อง เนื่องจากว่าเฟซบุ๊กได้ลงทุนไปกับรีแอค เนทีฟอย่างมาก ส่วนมากจึงถูกเขียนขึ้นมาด้วยรีแอค เนทีฟทำให้รีแอค เนทีฟได้รับความไว้วางใจและความนิยมอย่างมากในช่วงหลายปีที่ผ่านมา

2.6.5.3 การนำมาใช้ (Adoption)

รีแอค เนทีฟถูกใช้นำไปอย่างกว้างขวาง ความนิยมบางส่วนอาจจะมาจากรีแอค เนทีฟทุกวันนี้ ผลิตภัณฑ์อย่าง เฟซบุ๊ก อินสตาแกรม อูเบอร์ อีท (Uber Eats) เทสลา (Tesla) บลูมเบิร์ก (Bloomberg) และอีกมากมายใช้รีแอค เนทีฟในการสร้างแอปพลิเคชัน

2.6.5.4 การติดตั้งรีแอค เนทีฟ ชีแอลไอ (React Native CLI)

1. เปิดคอมมานด์พร้อมท์ (Command Prompt)
2. พิมพ์คำสั่ง \$ npm install -g react-native-cli



```
Microsoft Windows [Version 10.0.19042.867]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\User>npm install -g react-native-cli
C:\Users\User\AppData\Roaming\npm\react-native -> C:\Users\User\AppData\Roaming\npm\node_modules\react-native\index.js
+ react-native-cli@2.0.1
added 4 packages from 1 contributor, removed 1 package and updated 24 packages in 9.3s

C:\Users\User>
```

รูปที่ 2.35 หน้าติดตั้งรีแอค เนทีฟ ชีแอลไอ

บทที่ 3

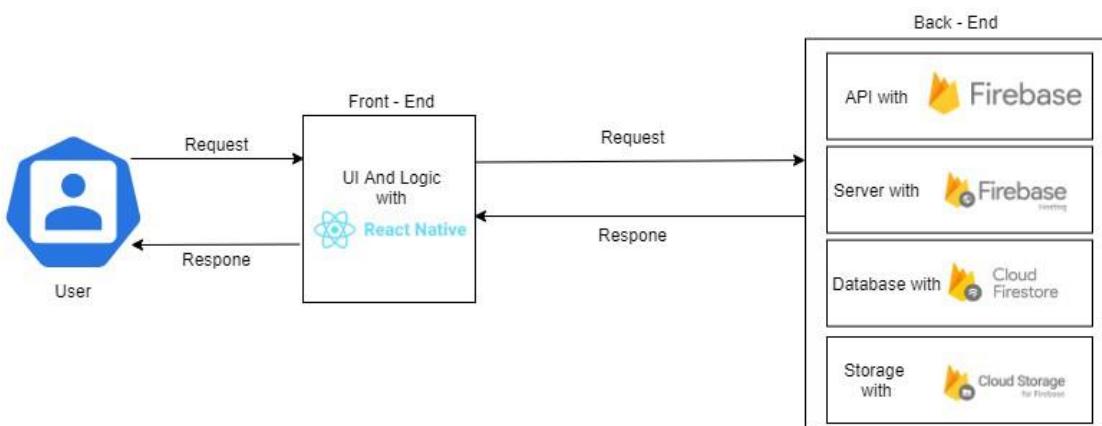
การออกแบบ การสร้าง และการทำงาน

3.1 กล่าวว่า

เนื้อหาของปริญณานิพนธ์นี้จะกล่าวถึงในส่วนของการออกแบบ การสร้าง และการทำงานของแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์ โดยประกอบไปด้วย การออกแบบระบบ การออกแบบระบบฐานข้อมูล การออกแบบหน้าต่างผู้ใช้งาน การติดตั้งเครื่องมือสำหรับการพัฒนาโดยใช้อุปกรณ์เดียว เช่น เน็ตฟิล และการสร้างและการทำงานของแอปพลิเคชัน โดยการออกแบบ การสร้าง และการทำงานของแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์จะแบ่งออกเป็น 3 ส่วน ได้แก่ ส่วนของผู้ใช้งาน ส่วนติดต่อผู้ใช้งาน (User Interface and Logic) และส่วนฐานข้อมูลไฟร์เบส โดยการสร้างและการทำงานแอปพลิเคชันบันทึกการซ่อมบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์ ใช้ไฟร์เบสเป็นส่วนสำหรับการจัดการข้อมูลที่ได้รับมาจากส่วนติดต่อผู้ใช้งาน

3.2 การออกแบบระบบ

3.2.1 แผนภาพสถาปัตยกรรมของแอปพลิเคชัน (Architecture Diagram)



รูปที่ 3.1 แผนภาพสถาปัตยกรรมของแอปพลิเคชัน

สถาบันกรรมของแอปพลิเคชันจะประกอบด้วย 3 ส่วนดังนี้

3.2.1.1 ส่วนของผู้ใช้งาน

ส่วนของผู้ใช้งาน เป็นส่วนการเริ่มต้นการส่งคำขอและเป็นส่วนสุดท้ายในการรับการตอบสนองจากแอปพลิเคชัน

3.2.1.2 ส่วนติดต่อผู้ใช้งาน

ส่วนติดต่อผู้ใช้ เป็นส่วนสำหรับการแสดงผลแอปพลิเคชันที่ทำการตอบรับคำขอจากส่วนของผู้ใช้งาน และแสดงผลกลับไปที่ผู้ใช้งาน ในขณะเดียวกันก็เป็นส่วนที่ติดต่อกับฝั่งเซิร์ฟเวอร์ (Server - Side) เพื่อใช้ในการจัดเก็บข้อมูลลงในฐานข้อมูล

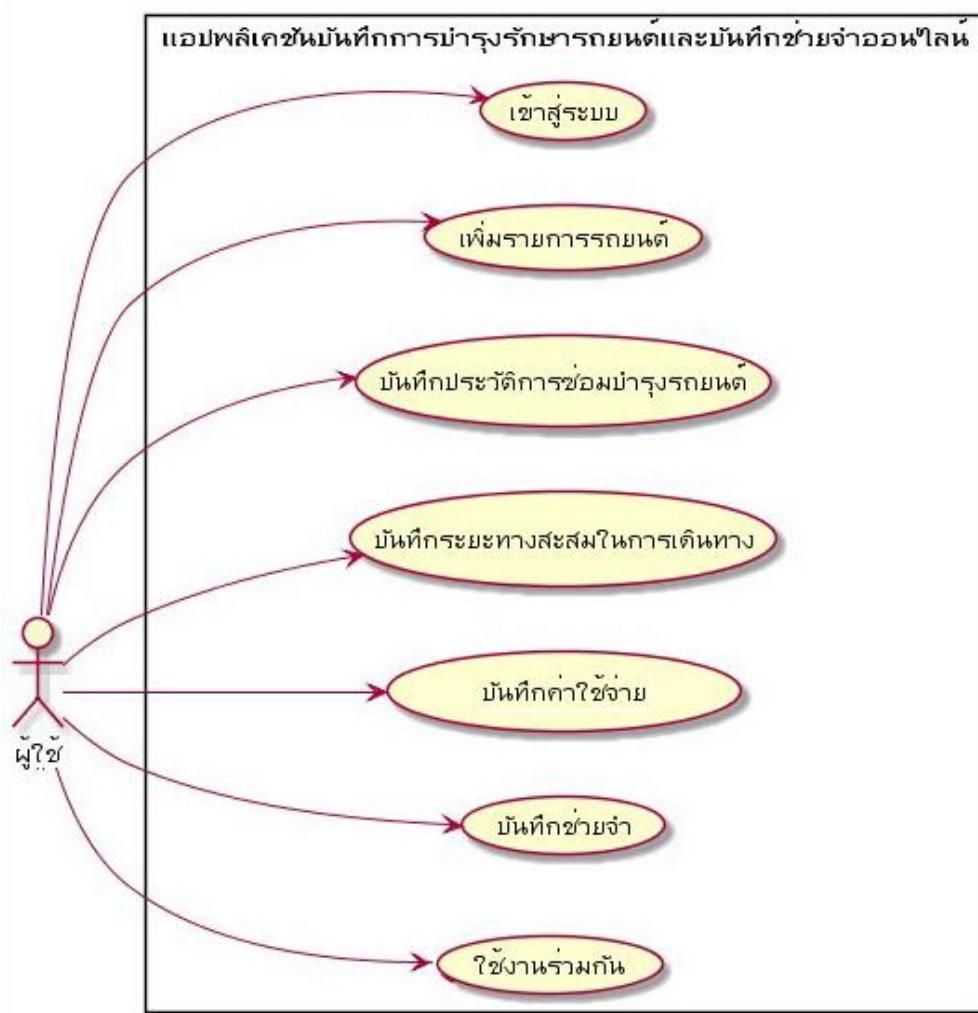
3.2.1.3 ส่วนของฐานข้อมูลเฟร์เบส

ส่วนของติดต่อผู้ใช้ เป็นส่วนสำหรับการจัดการข้อมูลที่ได้รับมาจากส่วนของติดต่อผู้ใช้งาน โดย ซึ่งแอปพลิเคชันจะใช้บริการของเฟร์เบส 2 บริการคือ

1. คลาวด์ ไฟร์สโตร์ เป็นบริการฐานข้อมูลในเฟร์เบส ฐานข้อมูลนี้จัดเก็บข้อมูลในรูปแบบของเจสัน (JSON) ใช้สำหรับเก็บข้อมูลบันทึกรายการซ้อมบำรุง บันทึกช่วยจำ
2. คลาวด์ สตอเรจ เป็นบริการเก็บข้อมูลสเตติกไฟล์สำหรับการเก็บรูปภาพในส่วนบันทึกช่วยจำ และรูปใบเสร็จในส่วนเพิ่มรายการบันทึกการซ้อมบำรุง

3.2.2 แผนภาพแสดงหน้าที่ที่ระบบต้องกระทำ (Use Case Diagram)

หน้าที่ที่ระบบของแอปพลิเคชันการบำรุงรักษารถยนต์และบันทึกซ่อมจاؤนไลน์ต้องกระทำ จะมีอยู่ด้วยกันทั้งหมด 7 ยูสเคส (Use Case) คือ เข้าสู่ระบบ เพิ่มรายการรถยนต์ บันทึกประวัติการซ่อมบำรุงรถยนต์ บันทึกระยะทางสะสมในการเดินทาง บันทึกค่าใช้จ่าย บันทึกซ่อมจำ และใช้งานร่วมกันดังรูปที่ 3.2



รูปที่ 3.2 แผนภาพยูสเคสของแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกซ่อมจاؤนไลน์

3.2.2.1 รายละเอียดการทำงานของแต่ละยูสเคส

ตารางที่ 3.1 รายละเอียดของยูสเคส เข้าสู่ระบบ

Use Case No. :	1	Use Case Name :	เข้าสู่ระบบ
Brief Description :	สำหรับผู้ใช้ที่มีบัญชีกิจกรรมและเพชบັກเพื่อเข้าสู่ระบบใช้งานแอปพลิเคชัน		
Actors :	ผู้ใช้		
Related Use Case :	-		
Stakeholders :	-		
Pre - Conditions :	ผู้ใช้ต้องมีบัญชีกิจกรรมหรือเพชบັກ		
Post - Conditions :	ผู้ใช้เป็นสมาชิกในระบบ มีข้อมูลผู้ใช้งานในฐานข้อมูล		
Flow of Events :	ผู้ใช้	ระบบ	
	1. ผู้ใช้เลือกวิธีเข้าสู่ระบบ	1. ระบบแสดงตัวเลือกเพื่อให้เลือกวิธีเข้าสู่ระบบ 2. ระบบตรวจสอบการยืนยันตัวตนและแสดงหน้าจอเมนูหลักของแอปพลิเคชัน	
Exception Conditions :	ผู้ใช้ไม่มีบัญชีกิจกรรมหรือเพชบັກ		

ตารางที่ 3.2 รายละเอียดของยูสเคส เพิ่มรายการถนนต์

Use Case No. :	2	Use Case Name :	เพิ่มรายการถนนต์
Brief Description :	สำหรับผู้ใช้ที่ต้องการเพิ่ม แก้ไข หรือลบรายการถนนต์		
Actors :	ผู้ใช้งาน		
Related Use Case :	เข้าสู่ระบบ		
Stakeholders :	-		
Pre - Conditions :	ผู้ใช้ต้องสู่เข้าระบบ		
Post - Conditions :	มีรายการถนนต์เพิ่มเข้ามาในแอปพลิเคชัน		
Flow of Events :	ผู้ใช้	ระบบ	
	1. ผู้ใช้กดปุ่มเพิ่มรายการถนน 2. ผู้ใช้เพิ่ม ภาพประกอบ ยี่ห้อ ทะเบียนรถ วันหมดประกันภัย รถ วันหมดภาษี ในกรณีผ่อน ผ่อนรถ สามารถเพิ่มวันที่ชำระ งวด จำนวนงวด ค่างวด และ เพิ่มการชำระข้อมูลให้กับผู้ใช้อื่น	1. ระบบแสดงหน้าจอสำหรับ บันทึก 2. ระบบบันทึกข้อมูลลงบน ฐานข้อมูล และแสดงผลใน หน้าจอรายการถนน	
Exception Conditions :	-		

ตารางที่ 3.3 รายละเอียดของยูสเคส บันทึกประวัติการซ้อมบำรุงรักษา

Use Case No. :	3	Use Case Name :	บันทึกประวัติการซ้อมบำรุงรักษา
Brief Description :	สำหรับผู้ใช้ที่ต้องการบันทึกประวัติการซ้อมบำรุงรักษาโดยนํา		
Actors :	ผู้ใช้งาน		
Related Use Case :	เข้าสู่ระบบ,เพิ่มรายการรักษา		
Stakeholders :	-		
Pre - Conditions :	ผู้ใช้ต้องเข้าสู่ระบบ		
Post - Conditions :	มีข้อมูลประวัติการบำรุงรักษาโดยนําเข้ามาในแอปพลิเคชัน		
Flow of Events :	ผู้ใช้	ระบบ	
	1. ผู้ใช้สามารถเพิ่มประวัติการซ้อมบำรุงรักษาโดยนํา	1. ระบบแสดงหน้าจอสำหรับบันทึกประวัติการซ้อมบำรุงรักษาโดยนํา	
	2. ผู้ใช้เพิ่ม วันเดือนปี ระยะเวลา สถานที่ และรายละเอียดของผู้รักษา	2. ระบบบันทึกข้อมูลลงบนฐานข้อมูล และแสดงผลในประวัติการซ้อมบำรุงรักษาโดยนํา	
Exception Conditions :	-		

ตารางที่ 3.4 รายละเอียดของยูสเคส บันทึกระยะทางสะสมในการเดินทาง

Use Case No. :	4	Use Case Name :	บันทึกระยะทางสะสมในการเดินทาง
Brief Description :	สำหรับผู้ใช้ที่ต้องการบันทึกระยะทางสะสมในการเดินทาง		
Actors :	ผู้ใช้งาน		
Related Use Case :	เข้าสู่ระบบ เพิ่มรายการรถยนต์ และบันทึกประวัติการซ้อมบำรุงรักษณ์		
Stakeholders :	-		
Pre - Conditions :	ผู้ใช้ต้องเข้าสู่ระบบ		
Post - Conditions :	มีข้อมูลระยะทางสะสมในการเดินทางเข้ามาในแอปพลิเคชัน		
Flow of Events :	ผู้ใช้	ระบบ	
	1. ผู้ใช้สามารถบันทึกระยะทางสะสมในการเดินทาง	1. ระบบแสดงหน้าจอสำหรับระยะทางสะสมในการเดินทาง	
		2. ระบบบันทึกข้อมูลลงบนฐานข้อมูล และแสดงผลในระยะทางสะสมในการเดินทาง	
Exception Conditions :	-		

ตารางที่ 3.5 รายละเอียดของยูสเคส ผู้ใช้สามารถบันทึกค่าใช้จ่าย

Use Case No. :	5	Use Case Name :	ผู้ใช้สามารถบันทึกค่าใช้จ่าย				
Brief Description :	สำหรับผู้ใช้ที่ต้องการบันทึกค่าใช้จ่าย						
Actors :	ผู้ใช้งาน						
Related Use Case :	เข้าสู่ระบบ เพิ่มรายการรายนั้น และบันทึกประวัติการซ้อมบำรุง รายนั้น						
Stakeholders :	-						
Pre - Conditions :	ผู้ใช้ต้องเข้าสู่ระบบ						
Post - Conditions :	มีข้อมูลค่าใช้จ่ายเข้ามาในแอปพลิเคชัน						
Flow of Events :	<table border="1"> <tr> <td>ผู้ใช้</td> <td>ระบบ</td> </tr> <tr> <td>1. ผู้ใช้สามารถเพิ่มบันทึกค่าใช้จ่าย</td> <td>1. ระบบแสดงหน้าจอสำหรับบันทึกค่าใช้จ่าย 2. ระบบบันทึกข้อมูลลงบนฐานข้อมูล และแสดงผลในบันทึกค่าใช้จ่าย</td> </tr> </table>	ผู้ใช้	ระบบ	1. ผู้ใช้สามารถเพิ่มบันทึกค่าใช้จ่าย	1. ระบบแสดงหน้าจอสำหรับบันทึกค่าใช้จ่าย 2. ระบบบันทึกข้อมูลลงบนฐานข้อมูล และแสดงผลในบันทึกค่าใช้จ่าย		
ผู้ใช้	ระบบ						
1. ผู้ใช้สามารถเพิ่มบันทึกค่าใช้จ่าย	1. ระบบแสดงหน้าจอสำหรับบันทึกค่าใช้จ่าย 2. ระบบบันทึกข้อมูลลงบนฐานข้อมูล และแสดงผลในบันทึกค่าใช้จ่าย						
Exception Conditions :	-						

ตารางที่ 3.6 รายละเอียดของยูสเคส บันทึกช่วยจำ

Use Case No. :	6	Use Case Name :	บันทึกช่วยจำ				
Brief Description :	สำหรับผู้ใช้ที่ต้องการสร้างบันทึกช่วยจำ						
Actors :	ผู้ใช้งาน						
Related Use Case :	เข้าสู่ระบบ						
Stakeholders :	-						
Pre - Conditions :	ผู้ใช้ต้องเข้าสู่ระบบ						
Post - Conditions :	มีข้อมูลบันทึกช่วยจำเข้ามาในแอปพลิเคชัน						
Flow of Events :	<table border="1"> <tr> <td>ผู้ใช้</td> <td>ระบบ</td> </tr> <tr> <td> 1. ผู้ใช้สามารถเพิ่ม แก้ไข ลบ บันทึกช่วยจำ 2. ผู้ใช้สามารถแนบรูปภาพ ประกอบ </td> <td> 1. ระบบแสดงหน้าจอสำหรับ บันทึกช่วยจำ 2. ระบบบันทึกข้อมูลลงบน ฐานข้อมูล และแสดงผลใน บันทึกช่วยจำ </td> </tr> </table>	ผู้ใช้	ระบบ	1. ผู้ใช้สามารถเพิ่ม แก้ไข ลบ บันทึกช่วยจำ 2. ผู้ใช้สามารถแนบรูปภาพ ประกอบ	1. ระบบแสดงหน้าจอสำหรับ บันทึกช่วยจำ 2. ระบบบันทึกข้อมูลลงบน ฐานข้อมูล และแสดงผลใน บันทึกช่วยจำ		
ผู้ใช้	ระบบ						
1. ผู้ใช้สามารถเพิ่ม แก้ไข ลบ บันทึกช่วยจำ 2. ผู้ใช้สามารถแนบรูปภาพ ประกอบ	1. ระบบแสดงหน้าจอสำหรับ บันทึกช่วยจำ 2. ระบบบันทึกข้อมูลลงบน ฐานข้อมูล และแสดงผลใน บันทึกช่วยจำ						
Exception Conditions :	-						

ตารางที่ 3.7 รายละเอียดของยูสเคส ใช้งานร่วมกัน

Use Case No. :	9	Use Case Name :	ใช้งานร่วมกัน
Brief Description :	สำหรับผู้ใช้ที่ต้องการใช้งานแอปพลิเคชันร่วมกัน		
Actors :	ผู้ใช้งาน		
Related Use Case :	เข้าสู่ระบบ เพิ่มรายการรถยนต์ และบันทึกช่วยจำ		
Stakeholders :	-		
Pre - Conditions :	ผู้ใช้ต้องเข้าสู่ระบบ		
Post - Conditions :	ผู้ใช้งานสามารถใช้งานแอปพลิเคชันร่วมกัน		
Flow of Events :	ผู้ใช้	ระบบ	
	1. ผู้ใช้ระบุอีเมลที่ต้องการใช้งานร่วมกัน	1. ระบบจะแสดงข้อมูลสำหรับผู้ใช้งานร่วมกัน	
Exception Conditions :	-		

3.3 การออกแบบระบบฐานข้อมูล

ฐานข้อมูลของแอปพลิเคชันบันทึกการซ่อมบำรุงรักษาระยนต์และบันทึกช่วยจำออนไลน์ มีการออกแบบฐานข้อมูล แบ่งออกเป็น 4 ฐานข้อมูล ดังนี้ ฐานข้อมูลของรายการรถยนต์ ฐานข้อมูลของรายการบันทึกการเติมเชื้อเพลิง ฐานข้อมูลของบันทึกการซ่อมบำรุง และฐานข้อมูลของรายการบันทึกช่วยจำ

3.3.1 ฐานข้อมูลของรายการรถยนต์

ฐานข้อมูลของรายการรถยนต์ ใช้สำหรับเก็บข้อมูล แบรนด์รถยนต์ ทะเบียนรถยนต์ วันหมดภาษี วันหมดประกันภัย วันที่ชำระงวด จำนวนงวด ค่างวด และเก็บอีเมลของผู้ใช้งานเพื่อใช้สำหรับเข้าถึงข้อมูลของแต่ละรายการรถยนต์และนำมาแสดงผล และเก็บข้อมูลอีเมลของผู้ที่ต้องการให้ใช้งานร่วมกัน

ตารางที่ 3.8 ตารางฐานข้อมูลของรายการรถยนต์

ลำดับ	ชื่อ	ประเภท	ข้อมูลที่เก็บ
1	Brand	String	เก็บข้อมูลแบรนด์รถยนต์
2	Registration	String	เก็บข้อมูลทะเบียนรถยนต์
3	InsuranceDate	String	เก็บข้อมูลวันหมดประกันภัย
4	TaxDate	String	เก็บข้อมูลวันหมดอายุภาษี
5	FirstPayment	String	เก็บข้อมูลวันที่ชำระงวด
6	Repayment	Number	เก็บข้อมูลจำนวนงวด
7	Installment	Number	เก็บข้อมูลค่างวด
8	SharedEmail	Array	เก็บข้อมูลอีเมลของผู้ที่ต้องการให้ใช้งานร่วมกัน

3.3.2 ฐานข้อมูลของรายการบันทึกการเติมเชื้อเพลิง

ฐานข้อมูลของรายการบันทึกการเติมเชื้อเพลิง ใช้สำหรับเก็บข้อมูลค่าใช้จ่ายในการเติมเชื้อเพลิง ข้อมูลระยะทาง และเก็บข้อมูลทะเบียนรถเพื่อใช้สำหรับเข้าถึงข้อมูลของรายการรถยนต์และนำมาแสดงผล

ตารางที่ 3.9 ตารางฐานข้อมูลของรายการบันทึกการเติมเชื้อเพลิง

ลำดับ	ชื่อ	ประเภท	ข้อมูลที่เก็บ
1	Registration	String	เก็บข้อมูลทะเบียนรถยนต์
2	Miles	Number	เก็บข้อมูลระยะทาง
3	Cost	Number	เก็บข้อมูลค่าใช้จ่ายในการเติมเชื้อเพลิง

3.3.3 ฐานข้อมูลของรายการบันทึกการซ่อมบำรุง

ฐานข้อมูลของรายการบันทึกการซ่อมบำรุง ใช้สำหรับเก็บข้อมูลรายการซ่อมบำรุง ผู้ให้บริการ ข้อมูลรายละเอียดเพิ่มเติม เก็บข้อมูลระยะทาง และเก็บข้อมูลทะเบียนรถ เพื่อใช้สำหรับเข้าถึงข้อมูลของรายการรถยนต์และนำมาแสดงผล

ตารางที่ 3.10 ตารางฐานข้อมูลของรายการบันทึกการซ่อมบำรุง

ลำดับ	ชื่อ	ประเภท	ข้อมูลที่เก็บ
1	Registration	String	เก็บข้อมูลทะเบียนรถยนต์
2	Miles	Number	เก็บข้อมูลระยะทาง
3	Service	Array	เก็บข้อมูลรายการซ่อมบำรุง
4	Servicer	String	เก็บข้อมูลผู้ให้บริการ
5	Note	String	เก็บข้อมูลรายละเอียดเพิ่มเติม
6	ImgReceipt	String	เก็บข้อมูล URL ของรูปภาพ

3.3.4 ฐานข้อมูลของรายการบันทึกช่วยจำ

ฐานข้อมูลของรายการบันทึกช่วยจำ ใช้สำหรับเก็บข้อมูล ซึ่งเรื่องที่บันทึก รายละเอียดที่บันทึก วันเวลาที่บันทึก และเก็บอีเมลของผู้ใช้งาน เพื่อใช้สำหรับเข้าถึงข้อมูลของแต่ละบันทึกมาและนำมาแสดงผล และเก็บข้อมูลอีเมลของผู้ที่ต้องการให้ใช้งานร่วมกัน

ตารางที่ 3.11 ตารางฐานข้อมูลของบันทึกช่วยจำ

ลำดับ	ชื่อ	ประเภท	ข้อมูลที่เก็บ
1	Email	String	เก็บข้อมูลอีเมลของผู้ใช้งาน
2	Date	TimeStamp	เก็บข้อมูลวันที่
3	Title	String	เก็บข้อมูลชื่อเรื่องที่บันทึก
4	Detail	String	เก็บข้อมูลรายละเอียดที่บันทึก
5	SharedEmail	String	เก็บข้อมูลอีเมลของผู้ที่ต้องการให้ใช้งานร่วมกัน

3.4 การออกแบบหน้าต่างผู้ใช้งาน

การออกแบบหน้าต่างผู้ใช้งานได้นำหลักการการออกแบบมาจากการแมททีเรียล ดีไซน์ทั้งการวางแผนสร้างของวัตถุในแต่ละหน้าต่างผู้ใช้งาน รวมถึงรูปแบบความเหมาะสมของวัตถุในการวางโดยในแอปพลิเคชันจะประกอบด้วยไปด้วย 2 ส่วนหลักคือส่วนของการใช้งานในด้านการบันทึกการบันทึกและการรายงานต์และส่วนของการบันทึกช่วยจำ

3.4.1 การบันทึกการบันทึกและการรายงานต์

ในส่วนของการออกแบบบันทึกการบันทึกการบันทึกและการรายงานต์จะมีหน้าหลักในหมวดนี้อยู่ด้วยกัน 5 หน้าดังนี้

3.4.1.1 หน้าแดชบอร์ด (Dashboard)

หน้าแดชบอร์ดการซ่อมบำรุงรถยนต์จะแสดงรายละเอียดในการซ่อมบำรุงต่างๆที่ได้ทำการซ่อมบำรุงล่าสุด 3 รายการ แสดงรายละเอียดการเติมเชื้อเพลิงที่ได้ทำการล่าสุด 3 รายการ และแสดงวันที่ในการชำรุดภายนอก ประกัน และวันผ่อนชำระเงิน โดยในการแสดงวันที่ในการชำรุดภายนอก อายุของเบรก อายุของแบตเตอรี่ จะแสดงเป็นแบบความก้าวหน้า (Progressive) เพื่อให้ทำการมองเห็นได้ชัด เนื่องจากส่วนนี้เป็นส่วนที่นานครั้งจะมีการบำรุงรักษาจึงนำมาแสดงในส่วนของแบบความก้าวหน้า เพื่อให้เห็นได้อย่างชัดเจน ในส่วนของรายการการเติมเชื้อเพลิงและการซ่อมบำรุงมีการจะทำการว่างรายการการเติมเชื้อเพลิงไว้ก่อนเนื่องจากมีการใช้งานที่บ่อย



รูปที่ 3.3 หน้าแดชบอร์ดการซ่อมบำรุงรถยนต์

3.4.1.2 หน้าจอค่าใช้จ่ายรวมของรถยนต์

หน้าจอค่าใช้จ่ายรวมของรถยนต์เป็นหน้าจอสำหรับการแสดงข้อมูลจำนวนเงินที่ได้ทำการเติมเข้าสู่เพลิงและค่าใช้จ่ายของรถในการบำรุงรักษาโดยจะใช้ข้อมูลที่ได้มีการเก็บรายละเอียดไว้คำนวณเป็นค่าใช้จ่ายของรถยนต์โดยด้านบนจะมีการกำกับด้วยยี่ห้อของรถว่าเป็นคันใด ในส่วนลัดลงมาจะเป็นรายละเอียดค่าใช้จ่ายในแต่ละรายการและด้านล่างสุดจะเป็นการรวมค่าใช้จ่ายทั้งหมดของรถยนต์ โดยในแต่ละรายการบันทึกจะมีเส้นใต้แยกไว้แต่ละรายการเพื่อให้สามารถมองได้สะดวก โดยหัวข้อแต่ละรายการจะอยู่ด้านซ้ายของหน้าจอและจำนวนเงินจะอยู่ด้านขวาของหน้าจอ



รูปที่ 3.4 หน้าจอค่าใช้จ่ายรวมของรถยนต์

3.4.1.3 หน้าการบันทึกรายละเอียด

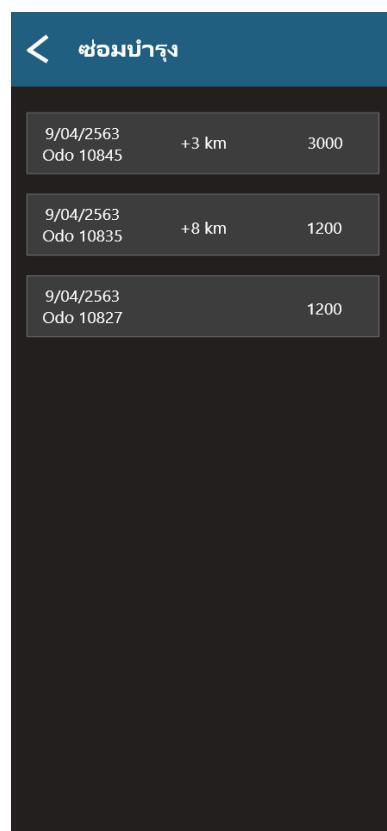
หน้าจอสำหรับการเพิ่มรายการการรายละเอียดในรถ ในการออกแบบจะมีชื่อหน้าการบันทึกข้อมูลไว้ในส่วนของท็อป แอป บาร์นอกจากชื่อหน้าต่างรายละเอียดการบันทึกแล้วยังมีปุ่มน้ำทางในการย้อนกลับไปยังหน้าเดขาดบอร์ดซึ่งจะอยู่ทางด้านซ้ายและปุ่มการบันทึกการเพิ่มรายละเอียดอยู่ด้านขวาเมื่อ โดยด้านล่างของท็อป แอป บาร์จะเป็นรายละเอียดในการบันทึกโดยหัวข้อบันทึกแต่ละรายการจะอยู่ด้านบนของช่องสำหรับกรอกข้อมูล ในการแนบไฟล์จะมีปุ่มเป็นเครื่องหมายบวก และข้อความกำกับการแนบไฟล์



รูปที่ 3.5 หน้าจอเพิ่มรายการบันทึกการซ่อมบำรุง

3.4.1.4 หน้าการแสดงประวัติรายบันทึกการซ่อมบำรุง

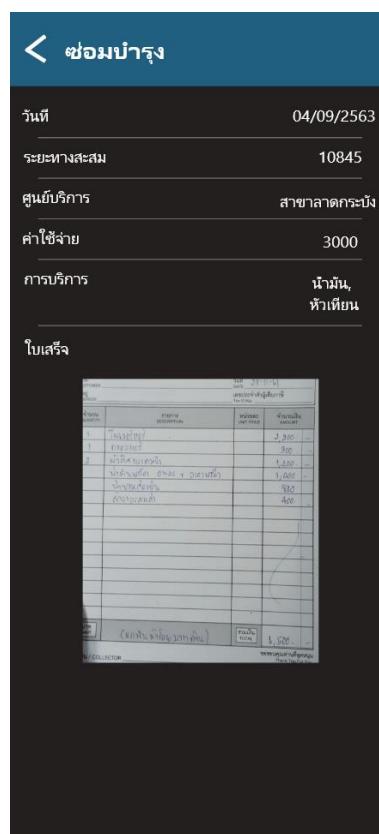
หน้าแสดงประวัติรายบันทึก โดยในการออกแบบจะมีชื่อหน้าประวัติรายบันทึกไว้ในส่วนของที่อป แอป บาร์นอกจากชื่อหน้าต่างรายละเอียดการบันทึกแล้วยังมีปุ่มน้ำทางในการย้อนกลับไปยังหน้าแดชบอร์ดซึ่งจะอยู่ทางด้านซ้าย โดยด้านล่างถัดมาจากการที่อป แอป บาร์จะเป็นในส่วนของประวัติการบันทึก โดยแต่ละรายการบันทึกจะแสดงอยู่บนการ์ดซึ่งบนการ์ดจะแสดงข้อมูลวันที่ในการบันทึก ราคา และระยะทางสะสมซึ่งการ์ดจะเรียงลำดับจากวันที่ล่าสุดอยู่ด้านบน



รูปที่ 3.6 หน้าประวัติรายบันทึกการซ่อมบำรุง

3.4.1.5 หน้าการแสดงรายละเอียดรายการการบันทึกการซ่อมบำรุง

ในการออกแบบจะมีชื่อหน้ารายละเอียดรายการการบันทึกไว้ในส่วนของท็อป แอป บาร์ นอกจากชื่อหน้าต่างรายละเอียดการบันทึกแล้วยังมีปุ่มน้ำทางในการย้อนกลับไปยังหน้าแดชบอร์ดซึ่งจะอยู่ทางด้านซ้าย โดยด้านล่างถัดมาจากการท็อป แอป บาร์จะเป็นรายละเอียดในการบันทึกที่ได้ทำการบันทึกไว้ในการบันทึกซึ่งหากไม่ได้มีรายการใดจะเป็นการเว้นว่างไว้โดยในแต่ละรายการของรายละเอียดการบันทึกจะมีเส้นด้านล่างแบ่งระหว่างรายการเพื่อให้สามารถอ่านได้ง่ายและสะดวก โดยในแต่ละหัวข้อการบันทึกจะมีข้อความบอกรายการบันทึกอยู่ด้านซ้ายและมีรายละเอียดอยู่ด้านขวา และในรายการการแนะนำไปเสริจจะมีชื่อรายการอยู่ด้านบนและรูปไปเสริจจะอยู่ด้านล่าง



รูปที่ 3.7 หน้ารายละเอียดรายการการบันทึกการซ่อมบำรุง

3.4.2 การบันทึกช่วยจำ

การออกแบบการบันทึกช่วยจำจะมีหน้าหลักอยู่ในหมวดนี้ด้วยกัน 2 หน้าดังนี้

3.4.2.1 หน้าแสดงรายการบันทึกช่วยจำ

หน้าแสดงรายการบันทึกช่วยจำ ในการออกแบบจะมีชื่อหน้าแสดงรายการบันทึกช่วยจำไว้ในส่วนของท็อป แอป บาร์นอกจากชื่อหน้าต่างรายละเอียดการบันทึกแล้วยังมีปุ่มแฮมเบอร์เกอร์ (Hamburger Button) ทางด้วยซ้ายและมีปุ่มสำหรับการเพิ่มบันทึกช่วยจำด้านขวาโดยปุ่มการเพิ่มบันทึกช่วยจำจะมีสัญลักษณ์เป็นรูปเครื่องหมายบวก โดยถัดมาจากการส่วนของท็อป แอป บาร์จะเป็นแถบสำหรับการค้นหน้าข้อมูลและถัดลงมาจากการค้นหาจะเป็นการดแต่ละรายการการบันทึกช่วยจำโดยแต่ละการดจะแสดงหมวดหมู่ ชื่อเรื่อง และรายละเอียดบางส่วนที่ได้ทำการบันทึกโดยแต่ละการดในแต่ละหมวดจะมีสีที่ต่างกันซึ่งจะแบ่งออกเป็น 3 หมวดหมู่ได้แก่ สีเขียวหมวดการศึกษา สีชมพูหมวดครอบครัว และสีฟ้าหมวดทั่วไป



รูปที่ 3.8 หน้ารายการบันทึกช่วยจำ

3.4.2.2 หน้าจอเพิ่มรายการบันทึกความจำ

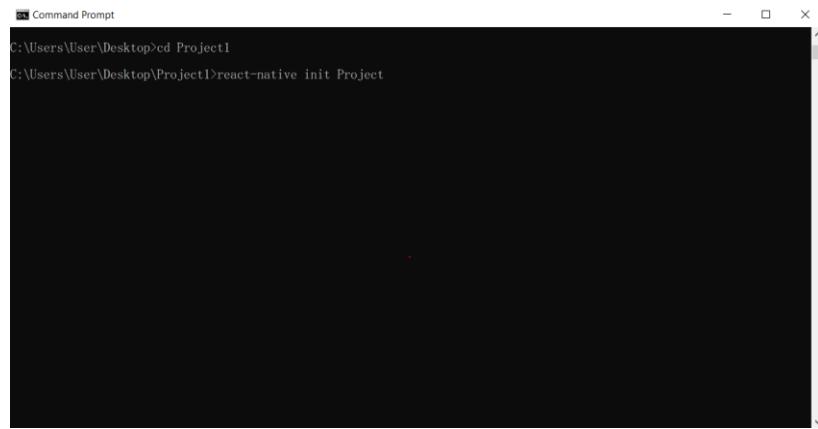
หน้าจอสำหรับการเพิ่มรายการบันทึกความจำ ในส่วนของท็อป แอป บาร์ปุ่มน้ำทางในการย้อนกลับไปยังหน้าแสดงรายการบันทึกช่วยจำซึ่งจะอยู่ทางด้านซ้าย ตัดมาจากส่วนของท็อป แอป บาร์จะเป็นช่องสำหรับการใส่ชื่อหัวข้อการบันทึกและถัดลงมาจะเป็นรายละเอียดของการบันทึกซึ่งก็จะมีเส้นแบ่งระหว่างรายการบันทึกแต่ละช่อง โดยในรายการบันทึกจะมีการชื่อหัวข้อการบันทึกหมวดหมู่ และช่องการใส่รายละเอียดของการบันทึก และปุ่มล่างขวาของหน้าจอจะมีสัญลักษณ์เป็นรูปเครื่องหมายบวกคือปุ่มสำหรับกดเพื่อเพิ่มรูปภาพในบันทึกช่วยความจำ



รูปที่ 3.9 หน้าจอเพิ่มรายการบันทึกความจำ

3.5 การสร้างแอปพลิเคชัน

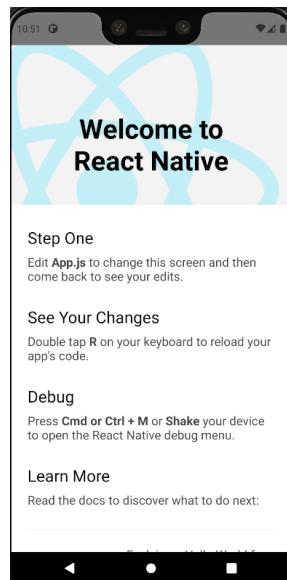
1. เปิดคอมมานด์พร้อมท์
2. ใช้คำสั่ง cd ไปยังเส้นทางที่ต้องการสร้างแอปพลิเคชัน
3. สร้างแอปพลิเคชัน โดยใช้คำสั่ง \$ react-native init Project



```
C:\Users\User\Desktop>cd Project1
C:\Users\User\Desktop\Project1>react-native init Project
```

รูปที่ 3.10 การสร้างแอปพลิเคชันโดยใช้คำสั่ง \$ react-native init Project

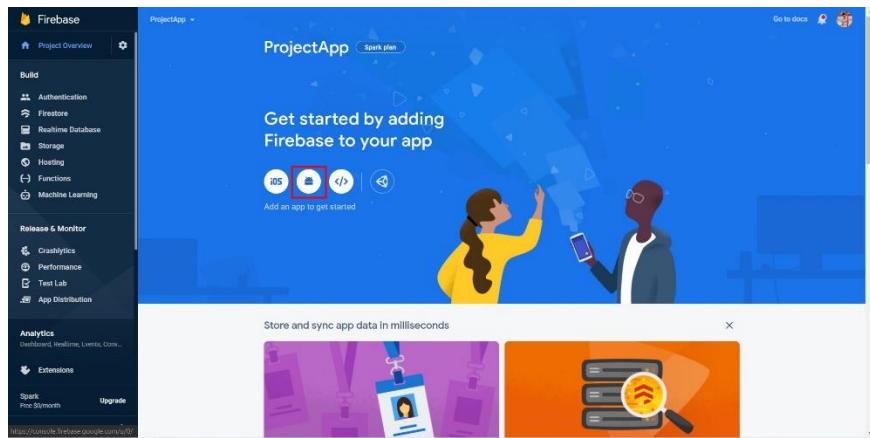
4. เริ่มพัฒนาแอปพลิเคชัน โดยใช้คำสั่ง \$ react-native run-android



รูปที่ 3.11 หน้าอิมเมเตอร์ (Emulator)

3.6 การเพิ่มไฟร์เบสลงในแอปพลิเคชัน

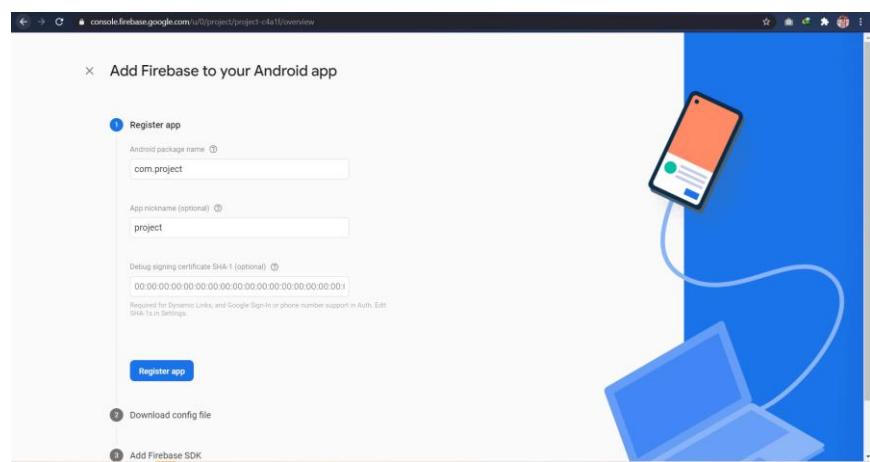
1. ไปที่ไฟร์เบสคอนโซล เลือกเพิ่มแอปเปิ้ลยังแอนดรอยด์



รูปที่ 3.12 หน้าต่างไฟร์เบสคอนโซล

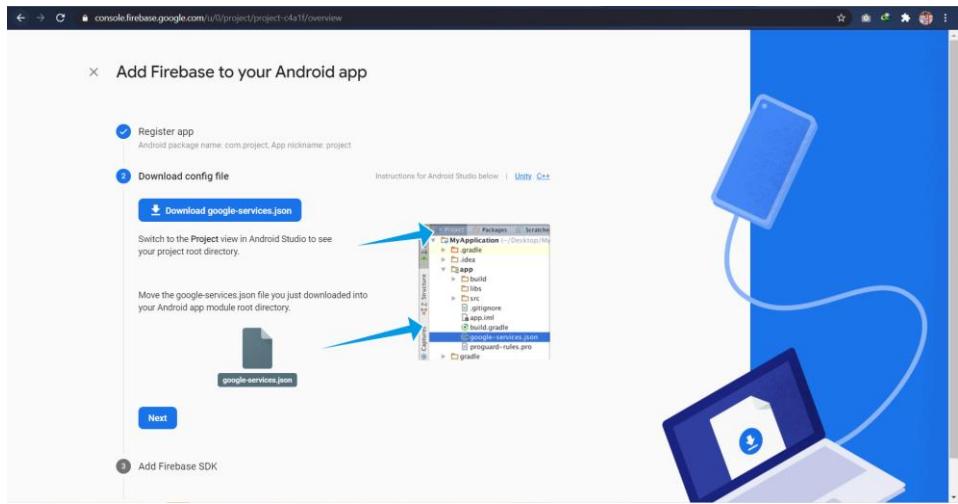
2. เลือก Add Firebase to your Android app

3. ลงทะเบียนแอปพลิเคชัน



รูปที่ 3.13 ทำการลงทะเบียนแอปพลิเคชันในไฟร์เบส

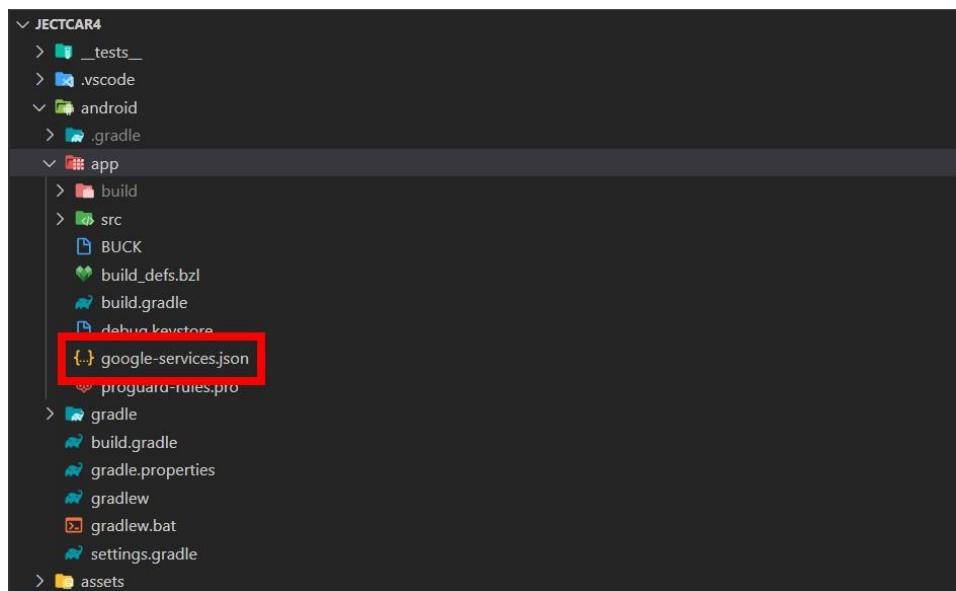
5. ดาวน์โหลดกุเกิลเชอร์วิส



รูปที่ 3.14 หน้าดาวน์โหลดกูเกิลเซอร์วิส

6. เพิ่มไฟร์เบสເສເດີເຄ

7. นำไฟล์กูเกิลເຊ່ອງວິສໄປໄວ້ທີ່ເສັ້ນທາງ android -> app



รูปที่ 3.15 หน้าເສັ້ນທາງ android -> app

8. เพิ่ມປລັກອິນໄຟຣີບສ ໂດຍໄປທີ່ພາທ android/build.gradle . ໂດຍເພີ່ມປລັກອິນ

apply plugin: 'com.google.gms.google-services'

```

buildscript {
    ext {
        googlePlayServicesVersion = "+" // default: "+"
        firebaseMessagingVersion = "+" // default: "+"
        buildToolsVersion = "29.0.2"
        minSdkVersion = 16
        compileSdkVersion = 29
        targetSdkVersion = 29
        supportLibVersion = "23.1.1" // default: 23.1.1
        googlePlayServicesAuthVersion = "16.0.1"
    }
    repositories {
        google()
        jcenter()
    }
    dependencies {
        classpath("com.android.tools.build:gradle:3.5.3")
        classpath 'com.google.gms:google-services:4.3.3'
    }
}

```

รูปที่ 3.16 โค้ดเพิ่มปลั๊กอินไฟร์เบส

3.7 การสร้างระบบยืนยันตัวตน

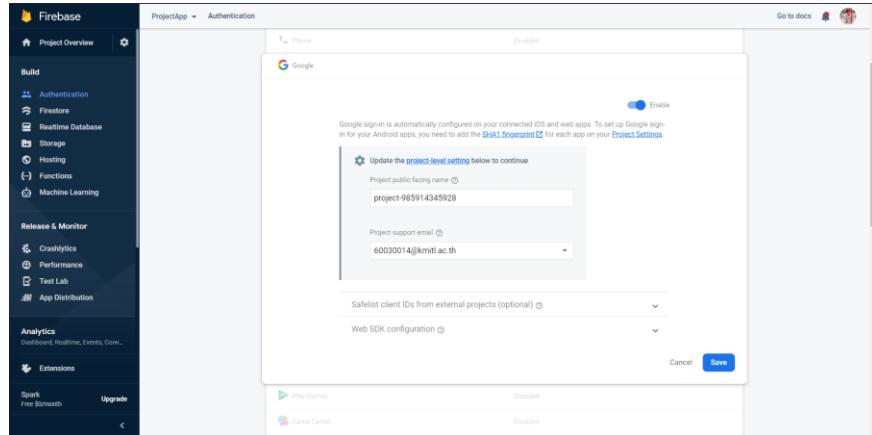
การยืนยันตัวตนเป็นส่วนสำคัญของแอปพลิเคชัน เพื่อความปลอดภัยของผู้ใช้งาน โดยไฟร์เบสเป็นโซลูชันเบื้องตนที่ถูกเก็บไว้ให้สำหรับการพัฒนาแอปพลิเคชัน

3.7.1 การติดตั้งแพ็กเกจที่จำเป็นสำหรับการสร้างระบบยืนยันตัวตน

1. เปิดคอมมานด์พร้อมที่
2. ใช้คำสั่ง `cd` ไปยังเส้นทางที่อยู่ของแอปพลิเคชัน
3. ติดตั้งแพ็กเกจ โดยใช้คำสั่ง `npm install --save @react-native-firebase/app @react-native-firebase/auth @react-native-community/google-signin @react-native-community/masked-view`

3.7.2. การเพิ่มระบบยืนยันตัวตน

1. เลือกเมนู Authentication ในไฟร์เบสคอนโซล
2. เปิด Google for the sign-in provider. จากนั้นกด Save



รูปที่ 3.17 การตั้งค่าระบบยืนยันตัวตน

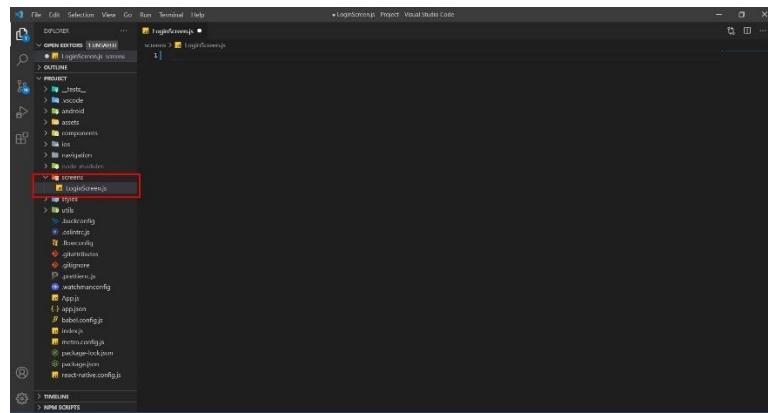
3. เปิด Facebook for the sign-in provider. จากนั้นกด Save
4. ไปที่วิชวล ส托ดิโอ โค้ดเพิ่มไฟล์ AuthProvider.js
5. เพิ่มฟังก์ชันของระบบยืนยันตัวตนในไฟล์ AuthProvider.js โดยการสร้าง AuthContext.Provider ดังรูปที่ 3.18

```
<AuthContext.Provider
  value={{
    user,
    setUser,
    googleLogin: async () => {
      try {
        const { idToken } = await GoogleSignIn.signIn();
        const googleCredential = auth.GoogleAuthProvider.credential(idToken);
        await auth().signInWithCredential(googleCredential);
      } catch(error) {
        console.log({error});
      }
    },
    fbLogin: async () => {
      try {
        const result = await LoginManager.logInWithPermissions(['public_profile', 'email']);
        if (result.isCancelled) {
          throw 'User cancelled the login process';
        }
        const data = await AccessToken.getCurrentAccessToken();
        if (!data) {
          throw 'Something went wrong obtaining access token';
        }
        const facebookCredential = auth.FacebookAuthProvider.credential(data.accessToken);
        await auth().signInWithCredential(facebookCredential);
      } catch(error) {
        console.log({error});
      }
    },
  }}>
{children}
</AuthContext.Provider>
```

รูปที่ 3.18 เพิ่มฟังก์ชันการยืนยันตัวตน

3.7.3 การเพิ่มหน้าจอสำหรับเข้าสู่ระบบ

1. สร้างไฟล์เดอร์ชื่อ screen หลังจากนั้นสร้างไฟล์ขึ้นมาใหม่ชื่อ LoginScreen.js



รูปที่ 3.19 สร้างไฟล์เดอร์ screen

2. เขียนฟังก์ชันสำหรับการสร้างหน้าเข้าสู่ระบบ โดยนำเข้า View, Image, ScrollView มาจากรีแอค เนทีฟ และนำเข้า AuthContext มาจาก AuthProvider ดัง code ในรูป 3.20

```

LoginScreen.js • build.gradle ...app • build.gradle android • AuthProvider.android.js •
screens > LoginScreen.js > ...
1 | import React, { useContext, useState } from 'react';
2 | import { View, Image, Platform, StyleSheet, ScrollView } from 'react-native';
3 | import SocialButton from '../components/SocialButton';
4 | import { AuthContext } from '../navigation/AuthProvider';
5 |
6 | const LoginScreen = () => {
7 |   const { googleLogin, fbLogin } = useContext(AuthContext);
8 |   return (
9 |     <ScrollView contentContainerStyle={styles.container}>
10 |       <Image
11 |         source={require('../assets/def.jpg')}
12 |         style={styles.logo}
13 |       />
14 |       {Platform.OS === 'android' ? (
15 |         <View>
16 |           <SocialButton
17 |             buttonTitle="Sign In with Facebook"
18 |             btnType="facebook"
19 |             color="#4A867A"
20 |             backgroundColor="#e6eaef"
21 |             onPress={() => fbLogin()}
22 |           />
23 |           <SocialButton
24 |             buttonTitle="Sign In with Google"
25 |             btnType="google"
26 |             color="#D4E4D4"
27 |             backgroundColor="#f5e7ea"
28 |             onPress={() => googleLogin()}
29 |           />
30 |         </View>
31 |       ) : null}
32 |       </ScrollView>
33 |   );
34 | }
35 | export default LoginScreen;

```

รูปที่ 3.20 เขียนฟังก์ชันเข้าสู่ระบบ

3.8 การสร้างบันทึกรายการซ่อมบำรุงรถยนต์

3.8.1 การสร้างหน้าสำหรับเลือกรายการรถยนต์

1. ไปยังโฟลเดอร์ screen สร้างไฟล์ใหม่ชื่อ LoginScreen.js
2. สร้างรายการรถยนต์โดยใช้ FlatList เป็นตัวแสดงรายการรถยนต์ โดยมีรายละเอียดการสร้างดังรูปที่ 3.21

```
import React, { useEffect, useState, useContext, useCallback } from 'react';
import {
  ScrollView,
  FlatList,
  SafeAreaView,
  RefreshControl,
} from 'react-native';
import CarCard from '../components/CarCard';
import { Container } from '../styles/HomeCar';
const carList = (props) => {

  return (
    <SafeAreaView style={{ flex: 1, }}>
      <Container>
        <ScrollView>
          <FlatList
            data={carlist}
            renderItem={({ item }) => (
              <CarCard item={item} onDelete={handleDelete} parentProps={props} />
            )}
            keyExtractor={(item) => item.id}
            ListHeaderComponent={ListHeader}
            ListFooterComponent={ListHeader}
            showsVerticalScrollIndicator={false}
          />
        </ScrollView>
      </Container>
    </SafeAreaView>
  );
};

export default carList;
```

รูปที่ 3.21 การสร้างหน้าสำหรับเลือกรายการรถยนต์

3.8.2 การสร้างหน้าสรุประยะລະເວີດຂອງรายการรถยนຕໍ

1. ไปยังโฟลเดอร์ screen สร้างไฟล์ใหม่ชื่อ carDetail.js
2. สร้าง progress wheel โดยใช้ AnimatedCircularProgress โดยนำเข้ามาจาก react-native-circular-progress และใช้โค้ด ดังรูปที่ 3.22

```

<AnimatedCircularProgress style={styles.pro1}
| size={95}
| width={17}
| fill={[itemDate.diff * 100 / 365]}
| tintColor="green"
| onAnimationComplete={() => console.log('onAnimationComplete')}
| backgroundColor="red" >
{
  (fill) => (
    <Text>
      {itemDate.diff} Day
    </Text>
  )
}
</AnimatedCircularProgress>

```

รูปที่ 3.22 การกำหนดแบบความก้าวหน้าแบบวงล้อ

- สร้างรายการบันทึกเชือเพลิงและบันทึกการซ่อมบำรุงโดยใช้ Card Text และ FlatList เป็นตัวแสดงข้อมูล โดยนำเข้ามาจากการรีแอค เนทีฟและสร้างโดยใช้โค้ด ดังรูปที่

```

<Card style={styles.Gas}>
  <Text style={styles.TextCen}>Recent Fuel Logs</Text>
  <FlatList
    data={gas}
    numColumns={1}
    keyExtractor={(item) => item.id.toString()}
    renderItem={({ item }) =>
      <GasCard item={item} parentProps={props} />
    }
    keyExtractor={(item) => item.id}
    ListHeaderComponent={ListHeader}
    ListFooterComponent={ListHeader}
    showsVerticalScrollIndicator={false}
  </FlatList>
</Card>

```

รูปที่ 3.23 การสร้างรายการบันทึกเชือเพลิง

- สร้างฟังก์ชันการคำนวนวันสิ้นอายุภาษีรถยนต์ โดยใช้ วันสิ้นอายุของภาษี ลบด้วยวันปัจุบันแล้วผลลัพธ์ออกมาเป็นวัน มีรูปแบบการเขียนดังนี้

```

querySnapshot.forEach((doc) => {
  const {
    Brand,
    CarRegistration,
    img,
    Insurance,
    Tax,
    DateFirst,
    Repayment,
    Installment,
  } = doc.data();
  list.push({
    id: doc.id,
    Brand,
    CarRegistration,
    img,
    Insurance,
    Tax,
    DateFirst,
    Repayment,
    Installment,
  });
});
setCar(list);
setInsuranceDate({ ...list, diff: moment(item.Insurance).diff(moment(), "days") })

```

รูปที่ 3.24 การสร้างฟังก์ชันคำนวณวันสิ้นสุดภาคี

3.9 การสร้างบันทึกช่วยจำ

- สร้างหน้าสำหรับเพิ่มบันทึกโดยช่วยจำ โดยใช้ TextInput และ Dropdown มีรูปแบบการเขียนได้ดังรูปที่

```

<TextInput style={styles.textTitle}
placeholder="TITLE..."
value={title}
onChangeText={(content) => settitle(content)}
/>
<Dropdown style={styles.dropdown}
icon='chevron-down'
iconColor="#000"
label='category'
baseColor="#fff"
labelFontSize='18'
data={data}
value={category}
onChangeText={(content) => setCategory(content)}
/>
<TextInput style={styles.textDescription}
underlineColorAndroid="transparent"
placeholder="DESCRIPTION..."
multiline={true}
value={memos}
onChangeText={(content) => setMemos(content)}
/>

```

รูปที่ 3.25 การสร้างช่องป้อนข้อมูลสำหรับบันทึกช่วยจำ

2. สร้างหน้าสำหรับแสดงบันทึก โดยใช้ FlatList เป็นตัวแสดงข้อมูลและ StatusBar เป็นตัวค้นหาบันทึก มีรูปแบบการเขียนโดยดังรูปที่

```
<View style={styles.parentView}>
  <StatusBar backgroundColor="white" barStyle="dark-content" />
  <TextInput
    style={styles.search}
    placeholder="search..."
    value={search}
    underlineColorAndroid="transparent"
    onChangeText={(text) => searchFilter(text)}
  />
  <FlatList
    style={styles.flatList}
    data={filteredData}
    numColumns={2}
    keyExtractor={(item, index) => index.toString()}
    //ItemSeparatorComponent = {ItemSeparatorView}
    renderItem={({ item }) =>
      <MemosCard item={item} onDelete={handleDelete} parentProps={props} />
    }
  }
  keyExtractor={(item) => item.id}
  ListHeaderComponent={ListHeader}
  ListFooterComponent={ListHeader}
  showsVerticalScrollIndicator={false}
  refreshControl={[
    <RefreshControl refreshing={refreshing} onRefresh={onRefresh} />
  ]}
  />
</View>
```

รูปที่ 3.26 การสร้างหน้าสำหรับแสดงบันทึก

3. สร้างฟังก์ชันในการค้นหาโดยใช้ filter เป็นตัวกรองข้อมูลมีรูปแบบการเขียนดังรูปที่

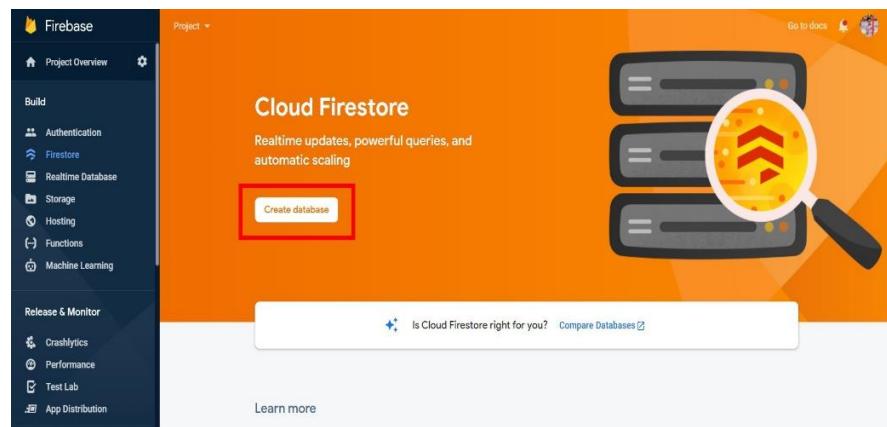
```
const searchFilter = (text) => {
  if (text) {
    const newData = masterData.filter((item) => {
      const itemData = item.Title ? item.Title.toUpperCase()
        : ''.toUpperCase();
      const textData = text.toUpperCase();
      const itemCat = item.category ? item.category.toUpperCase(): '';
      const itemDetail = item.MemosDetails ? item.MemosDetails.toUpperCase(): '';
      return itemData.indexOf(textData) > -1 || itemCat.indexOf(textData) >-1 || itemDetail.indexOf(textData) >-1;
    });
    setfilteredData(newData);
    setsearch(text);
  } else {
    setfilteredData(masterData);
    setsearch(text);
  }
};
```

รูปที่ 3.27 การสร้างฟังก์ชันในการค้นหา

3.10 การสร้างคลาวด์ ไฟร์สโตร์เพื่อใช้เป็นฐานข้อมูล

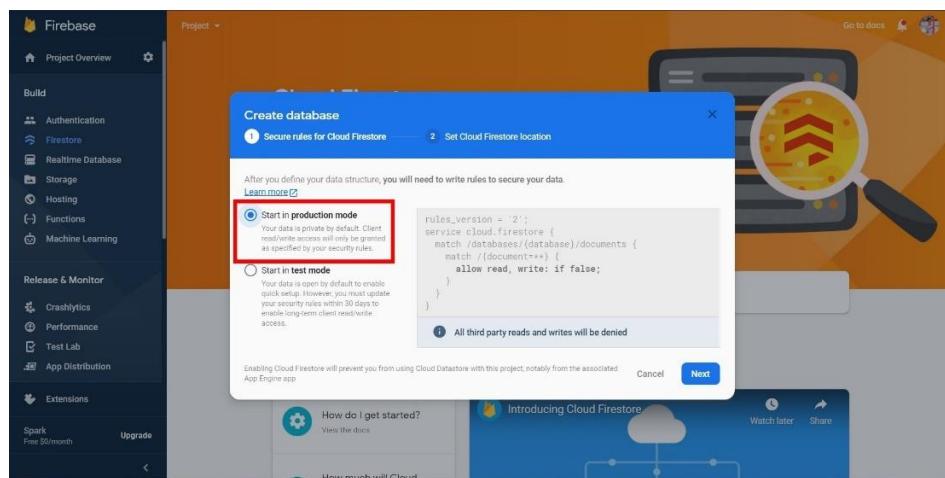
1. เริ่มต้นสร้างคลาวด์ ไฟร์สโตร์ และเลือก Create database โดยไปที่เว็บไซต์

<https://console.firebaseio.google.com/firestor>



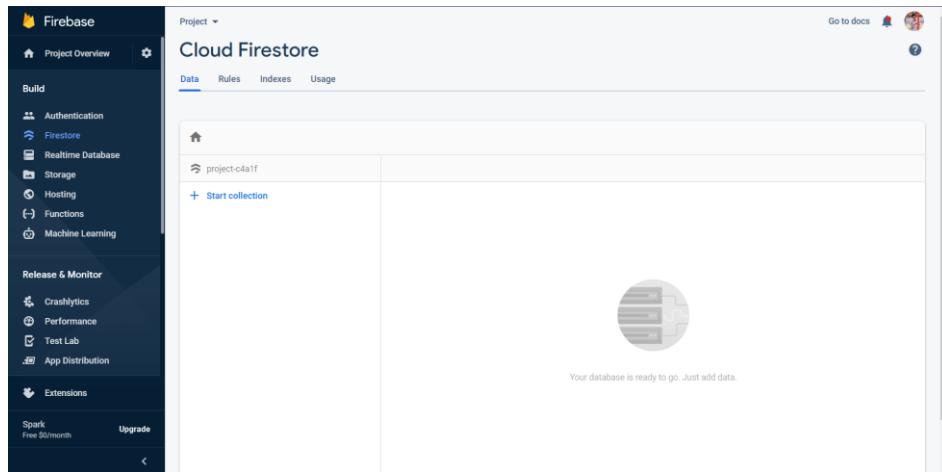
รูปที่ 3.28 หน้าต่างการสร้างคลาวด์ ไฟร์สโตร์

2. เลือก Strat in production mode



รูปที่ 3.29 หน้าต่างการเลือกโหมดการสร้างคลาวด์ ไฟร์สโตร์

3. เริ่มต้นใช้งานคลาวด์ ไฟร์สโตร์



รูปที่ 3.30 หน้าจูนข้อมูลคลาวด์ ไฟร์สโตร์

3.10.1 การอัปโหลดและดึงข้อมูลในฐานข้อมูล

3.10.1.1 การอัปโหลดข้อมูลไปเก็บในฐานข้อมูล

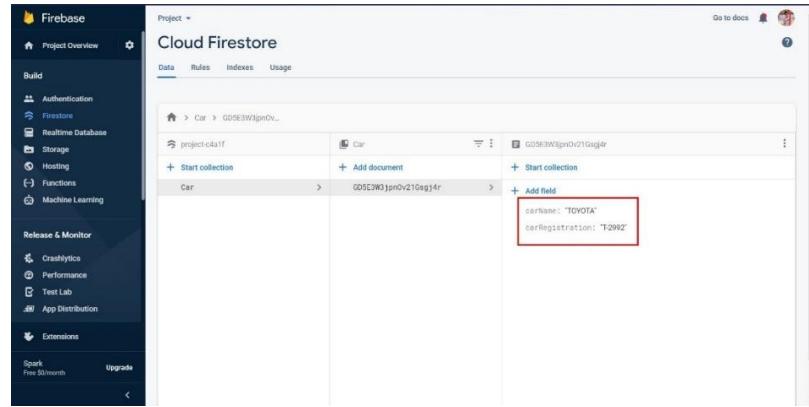
1. นำเข้าไฟร์เบสหมายังหน้าสกินที่ต้องการอัปโหลดข้อมูลขึ้นไปยังฐานข้อมูล
2. เขียนฟังก์ชันในการอัปโหลดข้อมูล โดยใช้เมธอด (Method) .add() ดังในรูปที่

3.46

```
screens > AddMemos.js > CarScreen > submitCar
1 import React, { useState } from 'react';
2 import { View, TextInput, Button } from 'react-native';
3 import firestore from '@react-native-firebase/firestore';
4
5 const CarScreen = () => {
6   const [car, setCar] = useState();
7   const submitCar = async () => {
8     firestore()
9       .collection(['Car'])
10      .add({
11        carName: 'TOYOTA',
12        carRegistration: 'T-2992'
13      })
14      .then(() => {
15        console.log('Car');
16        setMemos(null);
17      })
18      .catch((error) => {
19        console.log(`Something went wrong with added Car to firestore: ${error}`);
20      });
21    };
22    return (
23      <View>
24        <TextInput
25          value={car}
26          onChangeText={(content) => setCar(content)}
27        />
28        <Button onPress={submitCar}/>
29      </View>
30    );
31  };
32  export default CarScreen;
```

รูปที่ 3.31 ฟังก์ชันในการอัปโหลดข้อมูล โดยใช้เมธอด .add()

3. ໄປທີ່ຮູ້ຈານຂໍ້ມູນຄລາວດ് ໄຟຣັສໂຕຣ



ຮູບທີ 3.32 ມີຫຼາຍຸງຂໍ້ມູນຄລາວດ് ໄຟຣັສໂຕຣ

3.10.1.2 ການດຶງຂໍ້ມູນໃນຮູ້ຈານຂໍ້ມູນມາຍັງແອປພລິເຄີ້ນ

1. ນຳເຂົາໄຟຣັບສາມາຍັງໜ້າສກືນທີ່ຕ້ອງການດຶງຂໍ້ມູນໃນຮູ້ຈານຂໍ້ມູນມາຍັງແອປພລິເຄີ້ນ
2. ເຂີຍນິ້ນັກໆໃນການດຶງຂໍ້ມູນ ໂດຍໃໝ່ເມດຮອດ .where() ໄປຢັງຄອລເລັກຊັ້ນທີ່ຕ້ອງການດຶງ
ຂໍ້ມູນ ແລະໃໝ່ເມດຕອດ .get() ໃນການດຶງຂໍ້ມູນ ດັ່ງໃນຮູບທີ 3.48

```

1 import React, { useState } from 'react';
2 import { View, TextInput, Button } from 'react-native';
3 import firestore from '@react-native-firebase/firestore';
4
5 const CarScreen = () => {
6   const [car, setCar] = useState();
7   const fetchCar = async () => {
8     const list = [];
9     await firestore()
10       .collection('car')
11       .where("carName", "==", 'TOYOTA')
12       .get()
13       .then((querySnapshot) => {
14         querySnapshot.forEach(doc) => {
15           const [
16             carName,
17             carRegistration,
18           ] = doc.data();
19           list.push({
20             carName: carName,
21             carRegistration: carRegistration,
22           });
23         });
24       });
25   };
26   useEffect(() => {
27     fetchCar();
28   }, []);
29   return (
30     <View>
31       <Text>{car}</Text>
32     </View>
33   );
34 };
35 export default CarScreen;
36

```

ຮູບທີ 3.33 ພິ່ນົກໆໃນການດຶງຂໍ້ມູນ ໂດຍໃໝ່ເມດຮອດ .where() ແລະໃໝ່ເມດຕອດ .get()

บทที่ 4

การทดลองและผลการทดลอง

4.1 กล่าวนำ

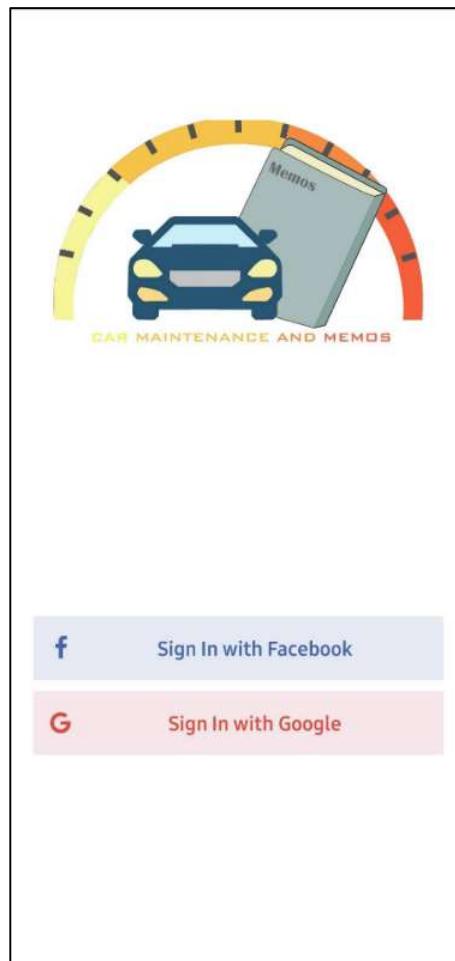
ในบทนี้จะกล่าวถึงการทดลองและผลการทดลองของแอปพลิเคชันบันทึกการบำรุงรักษารถยนต์ และบันทึกช่วยวิ่งจำออนไลน์โดยจะมีการทดสอบทั้งหมด 4 การทดสอบ คือ การแสดงผลในแต่ละหน้าจอ แอปพลิเคชัน การเข้าสู่ระบบ การใช้งานในหมวดการบำรุงรักษารถยนต์ และการใช้งานในหมวดบันทึกช่วยจำ

4.2 ทดสอบการทำงานของแอปพลิเคชัน

การทดสอบแอปพลิเคชันจะทำการทดสอบทั้งหมด 4 การทดสอบไปพร้อมกันในทุกการทดสอบ โดยจะเริ่มจากการเปิดในแต่ละหน้าจอของแอปพลิเคชันเพื่อทดสอบการแสดงผล และจะทำการทดสอบในทุกฟังก์ชันการใช้งาน ดังนี้

4.2.1 หน้าจอเข้าสู่ระบบ

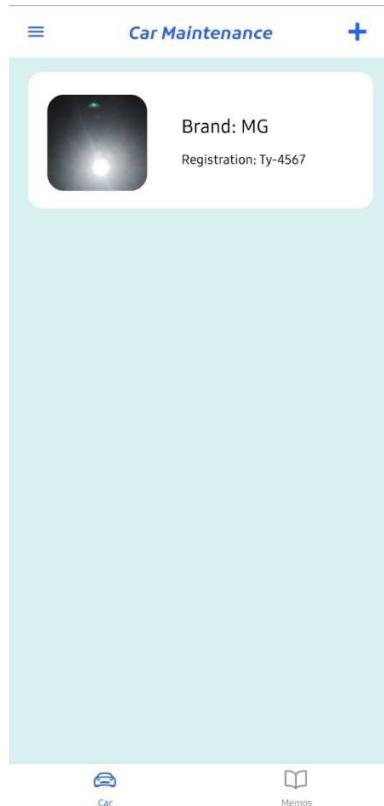
เมื่อทำการเข้าสู่ระบบแอปพลิเคชันการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์ ผู้ใช้งานจะต้องทำการยืนยันตัวตนผ่านระบบเฟซบุ๊กหรืออภิเกิลแอ็คเคนต์เพื่อเข้าใช้งานฟังก์ชันในแอปพลิเคชัน



รูปที่ 4.1 หน้าจอเข้าสู่ระบบ

4.2.2 หน้าจอรายการรถยนต์

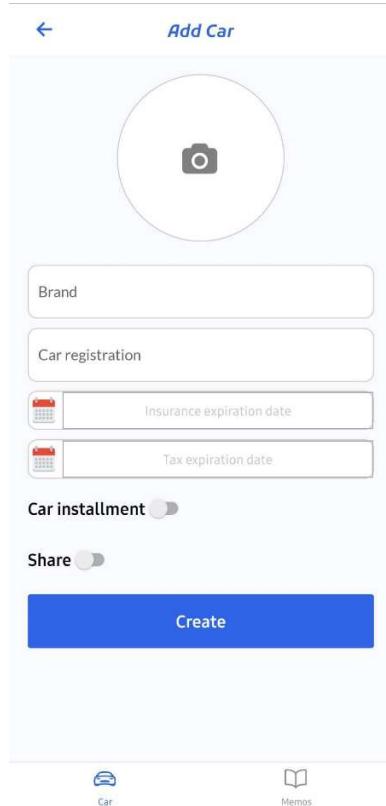
เมื่อผู้ใช้ทำการยืนยันตัวตัวเรียบร้อยแล้ว หน้าแรกของแอปพลิเคชันจะเป็นหน้าแสดงรายการรถยนต์ของผู้ใช้ โดยที่ผู้ใช้สามารถทำการเพิ่มรายการรถด้วยการกดปุ่มเครื่องหมายบวกด้านขวาบนได้ ซึ่งรายการของรถจะแสดงอยู่ด้านล่างท็อป บาร์ดังรูปที่ 4.2



รูปที่ 4.2 หน้าจอรายการรถยนต์

4.2.3 หน้าจอเพิ่มรายการรถยนต์

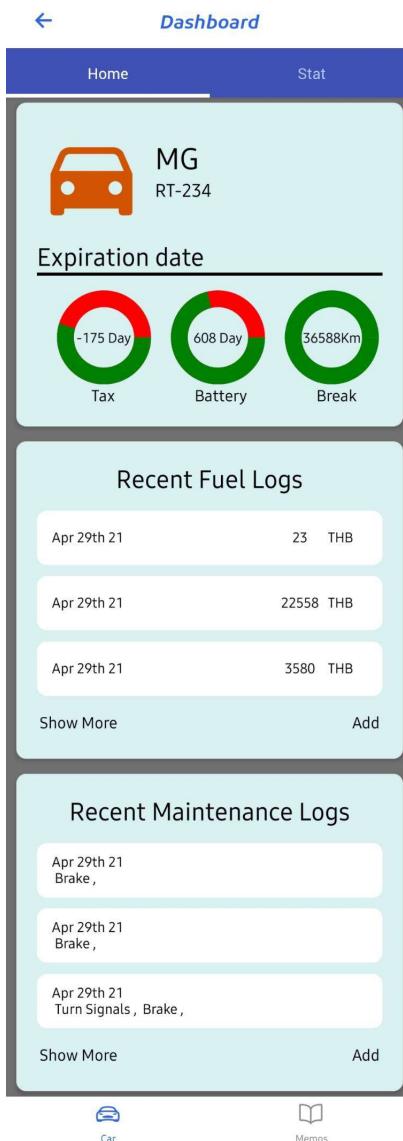
เมื่อผู้ใช้ทำการกดเพิ่มรถหน้ารายการของรถนั้น จะแสดงหน้าให้กรอกรายละเอียดของรถซึ่งประกอบไปด้วยรูป ยี่ห้อ เลขทะเบียนรถ วันสิ้นสุดอายุประกันรถยนต์ วันสิ้นสุดภาษีรถยนต์ กรณีการผ่อนชำระ และการเช็คข้อมูลรถร่วมกับผู้ใช้งานดังรูปที่ 4.3



รูปที่ 4.3 หน้าจอเพิ่มรายการรถยนต์

4.2.4 หน้าจอแสดงรายละเอียดรถยนต์

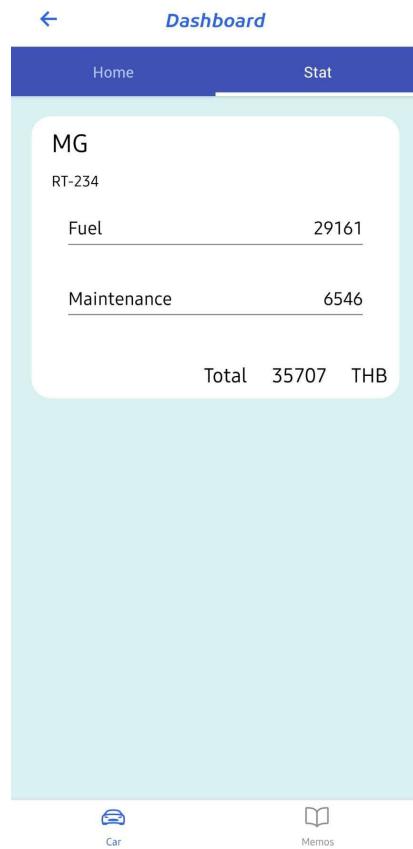
เมื่อผู้ใช้ทำการเลือกรายการรถจากหน้ารายการรถ ข้อมูลของรถจะแสดงดังรูป โดยจะประกอบไปด้วยรายการเติมเชื้อเพลิงล่าสุด รายการซ่อมบำรุงล่าสุดรวมถึงແບຄວາມກ້າວໜ້າในการแสดงรายการของภาษี ແບຕເຕອຣີ ແລະ ເບຣກດັ່ງນີ້



ຮູບທີ 4.4 หน้าจอแสดงรายละเอียดรถยนต์

4.2.5 หน้าจอแสดงค่าใช้จ่ายรวมของรถยนต์

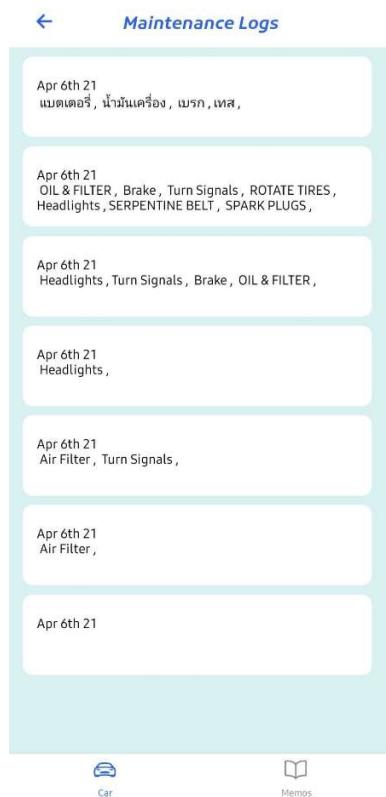
เมื่อผู้ใช้งานบันทึกหน้าค่าใช้จ่ายรวมของรถยนต์หน้านี้จะแสดงจำนวนเงินทั้งหมดที่ใช้ในการซ่อมบำรุง และเติมเชื้อเพลิงดังรูปที่ 4.5



รูปที่ 4.5 หน้าค่าใช้จ่ายรวมของรถยนต์

4.2.6 หน้าจอแสดงรายการบันทึกการซ่อมบำรุง

เมื่อผู้ใช้กดเพิ่มเติมในรายการการซ่อมบำรุงในหน้าจอแสดงรายการจะแสดงรายละเอียด จะแสดงรายการบันทึกการซ่อมบำรุงทั้งหมดที่ผู้ใช้ได้ทำการบันทึกโดยจะแสดงจากลำดับที่มีการบันทึกล่าสุดอยู่ด้านบนดังรูปที่ 4.6



รูปที่ 4.6 หน้าจอแสดงรายการบันทึกการซ่อมบำรุง

4.2.7 หน้าจอแสดงรายละเอียดบันทึกการซ่อมบำรุง

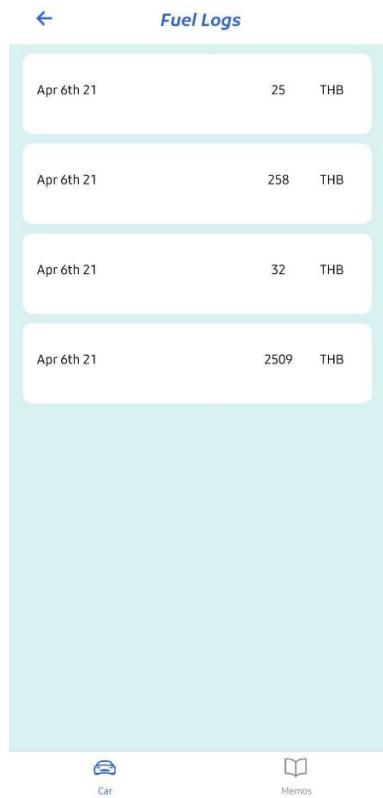
เมื่อผู้ใช้ทำการเลือกเข้าไปในรายการบันทึกการซ่อมบำรุงรายการได้รายการหนึ่งในหน้าจอแสดงรายการบันทึกการซ่อมบำรุง ระบบจะนำผู้ใช้งานมาที่หน้าจอแสดงรายละเอียดบันทึกการซ่อมบำรุงที่ผู้ใช้ได้เลือกไว้ดังรูปที่ 4.7



รูปที่ 4.7 หน้าจอแสดงรายละเอียดบันทึกการซ่อมบำรุง

4.2.8 หน้าจอแสดงรายการบันทึกเชื้อเพลิง

เมื่อผู้ใช้กดเพิ่มเติมในรายการเติมเชื้อเพลิงในหน้าจอแสดงรายการเบื้องต้น จะแสดงรายการบันทึกการเติมเชื้อเพลิงทั้งหมดที่ผู้ใช้ได้ทำการบันทึกดังรูปที่ 4.8



รูปที่ 4.8 หน้าจอแสดงรายการบันทึกเชื้อเพลิง

4.2.9 หน้าจอเพิ่มรายการบันทึกรายการซ่อมบำรุง

เมื่อผู้ใช้กดเพิ่มในรายการการซ่อมบำรุงในหน้าจอแสดงรายละเอียด จะแสดงหน้าในการให้ผู้ใช้ทำการเพิ่มรายการการซ่อมบำรุงโดยจะประกอบไปด้วยไมล์สะสม ราคา ผู้บริการ รายการรับบริการ หมายเหตุ และใบเสร็จดังรูปที่ 4.9

The screenshot shows a mobile application interface for adding a service. At the top is a back arrow and the text 'Add Service'. The main area contains several input fields and dropdowns:

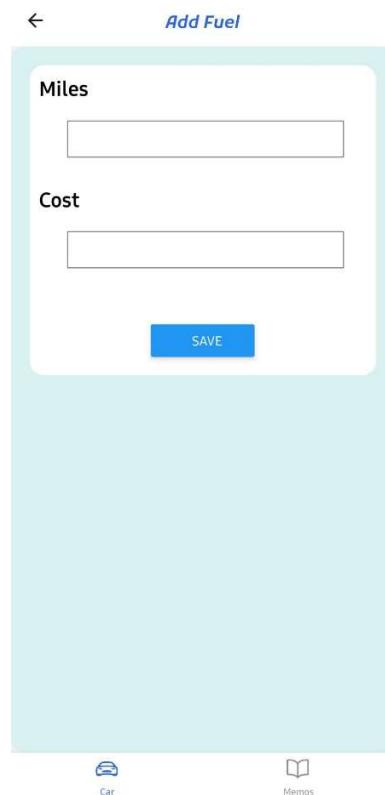
- Miles:** A text input field containing '258456'.
- Cost:** An empty text input field.
- Services:** A dropdown menu labeled 'Pick Items'.
- Servicer:** An empty text input field.
- Note:** An empty text input field.
- Add Receipt:** A blue button.
- SAVE:** A blue button at the bottom.

Below the form are two navigation icons: a car icon labeled 'Car' and a memo icon labeled 'Memos'.

รูปที่ 4.9 หน้าจอเพิ่มรายการการซ่อมบำรุง

4.2.10 หน้าจอเพิ่มรายการบันทึกเชื้อเพลิง

เมื่อผู้ใช้กดเพิ่มในรายการเติมเชื้อเพลิงในหน้าจอแสดงรายละเอียด จะแสดงหน้าในการให้ผู้ใช้ทำการเพิ่มรายการเติมเชื้อเพลิงโดยจะประกอบไปด้วยไมล์สะสมและราคัดังรูปที่ 4.10



รูปที่ 4.10 หน้าจอเพิ่มรายการบันทึกเชื้อเพลิง

4.2.11 หน้าจอแชร์ข้อมูลรถ

เมื่อผู้ใช้เลือกไปยังดรอเวอร์ เนวิเกชันแล้วเลือก Shared with me จะแสดงหน้าที่ผู้ใช้อื่นได้ทำ การแชร์รุ่นมา;yังผู้ใช้ดังรูปที่ 4.11



รูปที่ 4.11 หน้าจอแชร์ข้อมูลรถ

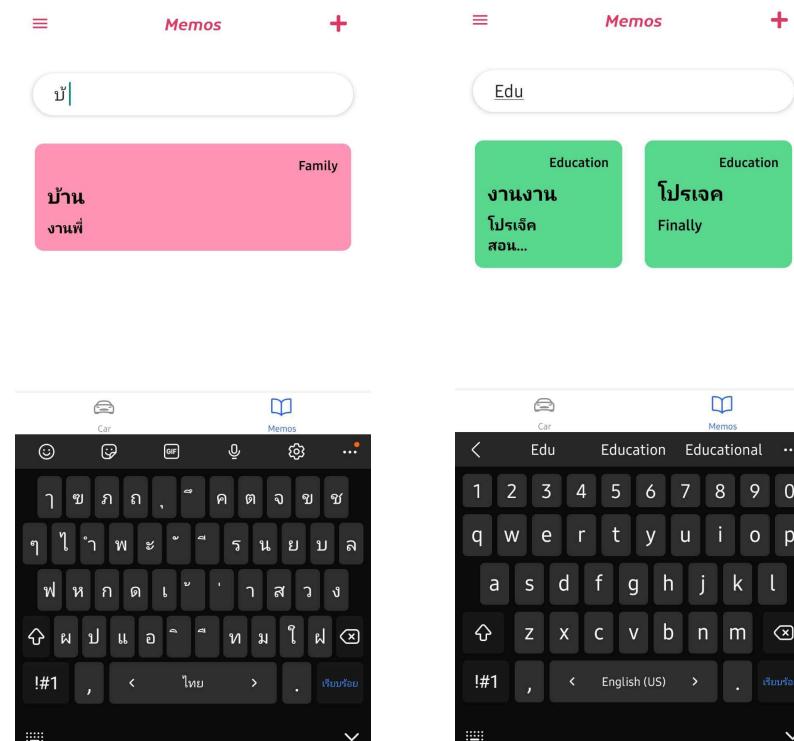
4.2.12 หน้าจอแสดงรายการบันทึกช่วยจำ

เมื่อผู้ใช้ทำการกดไอคอนรูปสมุดบันทึกที่บนothtom เนวิเกชันจะแสดงรายการบันทึกช่วยจำโดยผู้ใช้สามารถทำการกดดูและเพิ่มรายการการบันทึกช่วยจำในหน้านี้ได้ โดยแต่ละรายการบันทึกจะมีสีแต่ละหมวดหมู่กำกับดังรูปที่ 4.12



รูปที่ 4.12 หน้าแสดงรายการบันทึกช่วยจำ

โดยในหน้าแสดงรายการบันทึกช่วยจำจะสามารถทำการค้นหาบันทึกช่วยจำได้ โดยสามารถทำ การค้นหาได้ 3 รูปแบบคือ การค้นหาด้วยชื่อหัวข้อการบันทึก การค้นหาด้วยหมวดหมู่ และการค้นหาด้วยรายละเอียดของการบันทึกตั้งรูปที่ 4.13



ก. การค้นหาด้วยหัวข้อการบันทึก

ข. การค้นหาด้วยหมวดหมู่

รูปที่ 4.13 รูปแบบการค้นหาบันทึกช่วยจำ

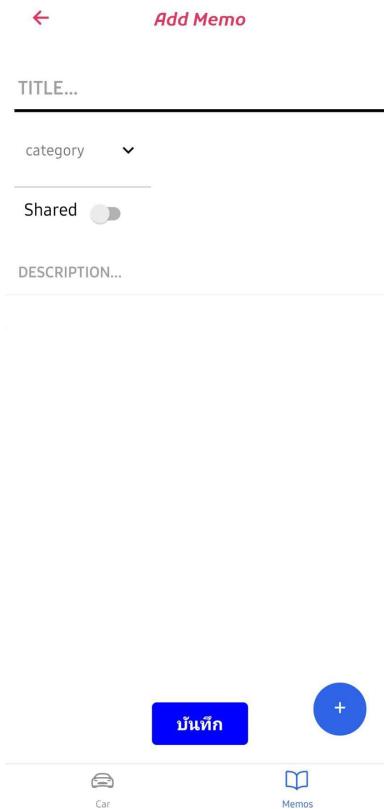


ค. การค้นหาด้วยรายละเอียด

รูปที่ 4.13 (ต่อ) รูปแบบการค้นหาบันทึกช่วยจำ

4.2.13 หน้าจอเพิ่มรายการบันทึกช่วยจำ

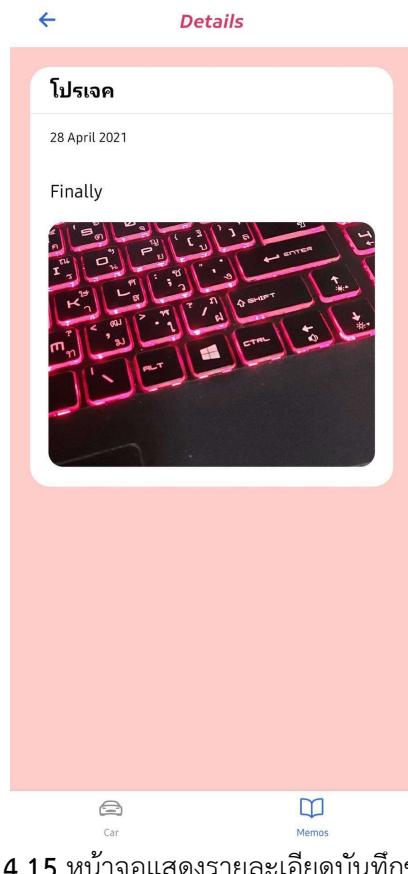
เมื่อผู้ใช้ทำการกดแอลกอชัน บัทตอนบนหน้าแสดงรายการบันทึกช่วยจำ จะเข้าไปยังรายการการบันทึกความจำโดยประกอบไปด้วยหัวข้อเรื่อง หมวดหมู่ ปุ่มการใช้งานร่วมกัน รายละเอียด และรูปดังรูปที่ 4.14



รูปที่ 4.14 หน้าจอเพิ่มรายการบันทึกช่วยจำ

4.2.14 หน้าจอแสดงรายละเอียดบันทึกช่วยจำ

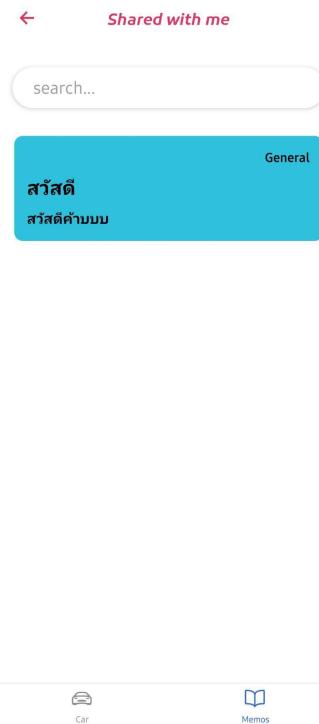
เมื่อผู้ใช้ทำการเลือกรายการบันทึกช่วยจำจากหน้าจอแสดงรายการบันทึกช่วยจำจะแสดงรายละเอียดของการบันทึกโดยจะแสดงหัวข้อการบันทึก วันที่บันทึก รายละเอียดการบันทึก และแสดงรูปภาพหากได้มีการบันทึกรูปดังรูปที่ 4.15



รูปที่ 4.15 หน้าจอแสดงรายละเอียดบันทึกช่วยจำ

4.2.15 หน้าจอแชร์ข้อมูลบันทึกช่วยจำ

เมื่อผู้ใช้เลือกไปยังครอเวอร์ เนวิเกชันแล้วเลือกแชร์ข้อมูลบันทึกช่วยจำจะแสดงหน้าที่ผู้ใช้อื่นได้ทำการแชร์บันทึกมายังผู้ใช้ดังรูปที่ 4.16



รูปที่ 4.16 หน้าจอแชร์ข้อมูลบันทึกช่วยจำ

4.3 ผลการทดสอบของแอปพลิเคชัน

ในการทดสอบของแอปพลิเคชันด้วยอุปกรณ์ตัวอย่างที่มีคุณสมบัติ ดังนี้

Samsung A71 ระบบปฏิบัติการ Android 10

RAM : 8 GB, ROM : 128 GB, CPU : Snapdragon 730

Battery : 4,500 mAh

จากการทดสอบการทำงานของแอปพลิเคชันในพื้นที่ขั้นการทำงาน ดังนี้

ตารางที่ 4.1 การทดสอบเข้าสู่ระบบ

ข้อการทดสอบ : การเข้าสู่ระบบ		
ขั้นตอนการทำงาน	ผลที่คาดหวัง	ผลการทดสอบ
การเข้าสู่ระบบด้วยเฟชบุ๊ก	สามารถเข้าสู่ระบบด้วยเฟชบุ๊กได้	ผ่าน
การเข้าสู่ระบบด้วยกุกเกิล	สามารถเข้าสู่ระบบด้วยกุกเกิลได้	ผ่าน

ตารางที่ 4.2 การทดสอบใช้งานในหมวดการบำรุงรักษาภายนต์

ข้อการทดสอบ : การใช้งานส่วนของการบำรุงรักษาภายนต์		
ขั้นตอนการทำงาน	ผลที่คาดหวัง	ผลการทดสอบ
เพิ่มรายการรยนต์	สามารถเพิ่มรายการรยนต์ได้	ผ่าน
ลบรายการรยนต์	สามารถลบรายการรยนต์ได้	ผ่าน
ใช้งานร่วมกับผู้อื่นผ่านการกรอกอีเมล	สามารถเพิ่มผู้ใช้งานร่วมกันได้สูงสุด 5 คน	ผ่าน
เพิ่มรายการการบันทึกการซ่อมบำรุง	สามารถเพิ่มข้อมูลการซ่อมบำรุงได้	ผ่าน
ลบรายการการบันทึกการซ่อมบำรุง	สามารถลบข้อมูลการซ่อมบำรุงได้	ผ่าน

ชื่อการทดสอบ : การใช้งานส่วนของการบำรุงรักษาภารณฑ์		
ขั้นตอนการทำงาน	ผลที่คาดหวัง	ผลการทดสอบ
แสดงรายละเอียดการซ่อมบำรุง	สามารถแสดงรายละเอียดการซ่อมบำรุงที่ได้ทำการบันทึกไว้	ผ่าน
เพิ่มรายการบันทึกเขื้อเพลิง	สามารถเพิ่มข้อมูลเขื้อเพลิงได้	ผ่าน
ลบรายการบันทึกเขื้อเพลิง	สามารถลบข้อมูลเขื้อเพลิงได้	ผ่าน
แสดงรายการนับถอยหลังบนแดบความคีบหน้า	สามารถนับถอยหลังได้	ผ่าน
การแจ้งเตือน	สามารถแจ้งเตือนได้	ไม่ผ่าน
การนับค่าใช้จ่ายทั้งหมดต่อคืน	สามารถรวมค่าใช้จ่ายได้	ผ่าน

ตารางที่ 4.3 การทดสอบใช้งานในหมวดของการบันทึกช่วยจำ

ชื่อการทดสอบ : การใช้งานส่วนของบันทึกช่วยจำ		
ขั้นตอนการทำงาน	ผลที่คาดหวัง	ผลการทดสอบ
เพิ่มบันทึกช่วยจำ	สามารถเพิ่มบันทึกช่วยจำได้	ผ่าน
แยกหมวดของบันทึกช่วยจำตามหมวดสี	สามารถแยกหมวดหมุนของบันทึกช่วยจำด้วยสีได้	ผ่าน
ใช้งานบันทึกช่วยจำร่วมกับผู้อื่นผ่านอีเมลได้	สามารถเพิ่มผู้ใช้งานร่วมกันได้	ผ่าน
ค้นหาบันทึกช่วยจำ	สามารถค้นหาบันทึกช่วยจำด้วยชื่อหมวดหมู่ และรายละเอียดการบันทึกได้	ผ่าน

บทที่ 5

บทสรุป

5.1 สรุป

แอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำออนไลน์ เป็นแอปพลิเคชันที่ใช้ในการช่วยเหลือผู้ใช้ข้อมูลในการจดจำเกี่ยวกับการบำรุงรักษารถยนต์โดยภายในแอปพลิเคชันจะแบ่งหมวดการใช้งานเป็น 2 หมวด คือ หมวดการบำรุงรักษารถยนต์ และหมวดบันทึกช่วยจำ ผู้ใช้งานจะสามารถใช้หมวดการบำรุงรักษารถยนต์ในการบันทึกรายการรถยนต์แต่ละคันที่มี และรถแต่ละคันที่ถูกบันทึกจะสามารถบันทึกการซ่อมบำรุงและบันทึกการซ่อมเชื่อเพลิงได้ เมื่อผู้ใช้ทำการบันทึกระบบจะสามารถคำนวณค่าใช้จ่าย อายุการใช้งานแบบเต็อร์ เบรกและสามารถแสดงผลแบบนับถอยหลังเป็นจำนวนวันให้แก่ผู้ใช้ได้ ส่วนบันทึกช่วยจำผู้ใช้งานสามารถจดบันทึกทั่วไปในการเตือนความจำต่างๆในชีวิตประจำวันและในแต่ละบันทึกจะสามารถแนบรูปภาพสิ่งที่ต้องการเพื่อช่วยในการเตือนความจำให้ชัดเจนขึ้นได้อีกด้วย และอีกหนึ่งจุดเด่นของแอปพลิเคชันคือสามารถใช้งานร่วมกันได้โดยจะสามารถรองรับผู้ใช้งานได้สูงสุด 5 คนต่อรายการบันทึก

5.2 ปัญหาและวิธีการแก้ไข

จากการดำเนินการสร้างและทดสอบโครงการนี้ปรากฏว่ามีปัญหาเกิดขึ้นหลายประการ ซึ่งสามารถสรุปได้ดังนี้

1. ไม่สามารถใช้งานไลบรารีชื่อ react-navigator/drawer จากปกติที่ใช้งานได้ วิธีการแก้ไข ลบโมดูลทั้งหมด ทำการล้างแคช (Cache) และทำการลงแพ็กเกจใหม่
2. ปัญหาการใช้งานอีमูเลเตอร์เนื่องจากเวอร์ชันของเอสดีเคที่ติดตั้งภายในกลุ่มมีเวอร์ชันที่แตกต่างกัน วิธีการแก้ไข อัพเดทเวอร์ชันของเอสดีเคให้ตรงกัน
3. ปัญหาการใช้งานอีมูเลเตอร์เนื่องจากอีมูเลเตอร์ขึ้นสถานะว่าอฟไลน์ วิธีการแก้ไข ทำการรีสตาร์ทอีมูเลเตอร์

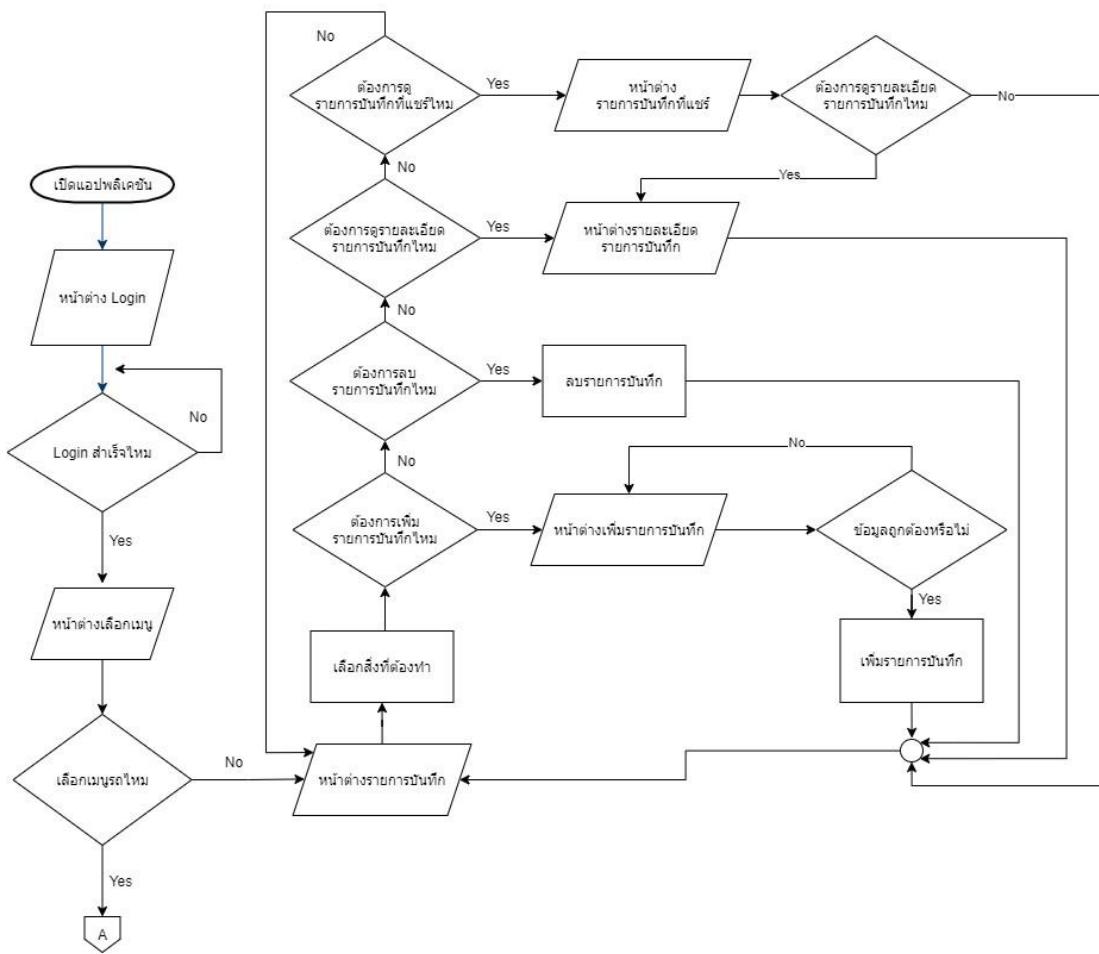
4. ไม่สามารถเรียกใช้งาน Card จากไลบรารีชื่อ native-base ได้
วิธีการแก้ไข อัพเดทเวอร์ชัน native-base
5. ปัญหาการใช้งานไม่สามารถเรียกอีมูเลเตอร์ได้ในผู้ใช้งานซีพียูของเอเอ็มดี (AMD)
วิธีการแก้ไข ทำการเปิด SVM ใน BIOS ก่อนการติดตั้งไดรเวอร์ Hypervisor ที่เป็นไดรเวอร์สำหรับการใช้งานอีมูเลเตอร์ของผู้ใช้งานซีพียูของเอเอ็มดี
6. ไม่สามารถเพิ่มอุปกรณ์อีมูเลเตอร์ในแอนดรอยด์ สตูดิโอได้
วิธีการแก้ไข ถอนการติดตั้งแอนดรอยด์ สตูดิโอแล้วทำการติดตั้งใหม่
7. ไม่สามารถเขียนโค้ดแบบ Class Component ร่วมกับ Functional Component ได้
วิธีการแก้ไข เปลี่ยนการเขียนเป็นแบบ Functional Component เพื่อให้โค้ดของスマาร์ตโฟนสามารถใช้งานร่วมกันได้
8. ไม่สามารถใช้ เอปีโอ กีด (API Key) เชื่อมต่อกับเฟร์เบส
วิธีการแก้ไข ใช้กุเกิลเซอร์วิส แทนการเขียน เอปีโอ กีด
9. แอปพลิเคชันไม่สามารถแสดงผล Responsive กับสมาร์ทโฟนทุกรุ่นได้
วิธีการแก้ไข ใช้ไลบรารีในการช่วยจัดการที่ชื่อว่า react-native-responsive-screens
10. ไม่สามารถใช้งานระบบบันทึกตัวตน 2 รูปแบบพร้อมกันได้
วิธีการแก้ไข เพิ่มแพ็กเกจ multidex ลงใน dependencies ของแอนดรอยด์

5.3 แนวทางการพัฒนา

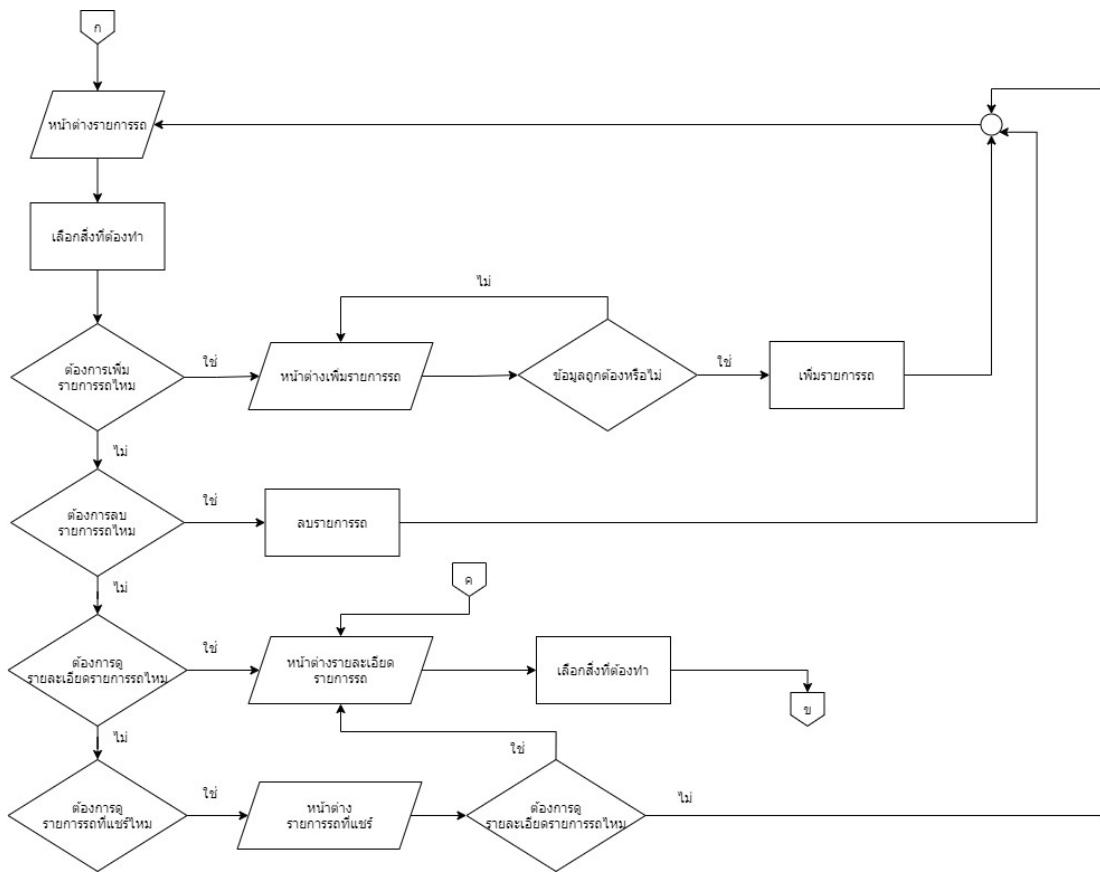
1. พัฒนาให้สามารถใช้ GPS ติดตาม เพื่อบันทึกระยะทางได้อัตโนมัติ
2. พัฒนาให้สามารถใช้งานแอปพลิเคชันได้รวดเร็วขึ้น

ภาคผนวก ก

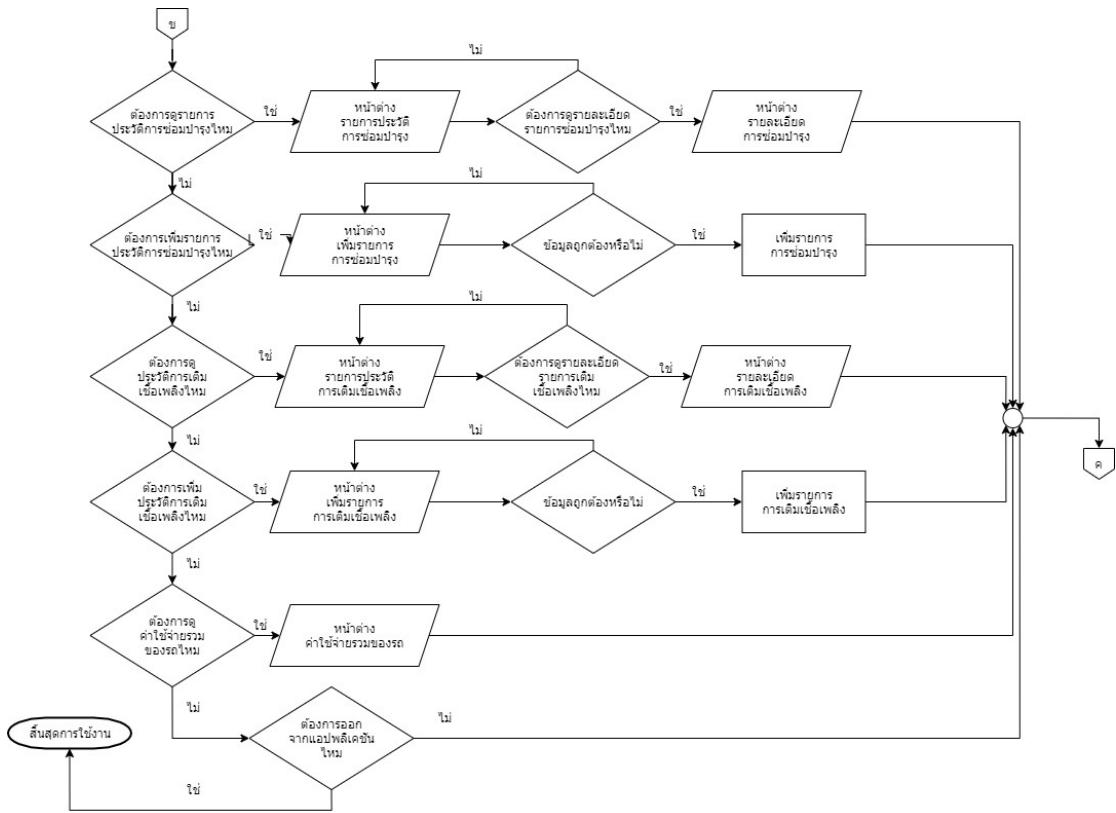
ผังงาน



รูป ก.1 ผังการทำงานของแอปพลิเคชัน



รูป ก.2 (ต่อ) ผังการทำงานของแอปพลิเคชัน



รูป ก.3 (ต่อ) ผังการทำงานของแอปพลิเคชัน

ภาคผนวก ข

ฐานข้อมูล

ตารางที่ ข.1 โครงสร้างข้อมูลรายการรถยนต์

ลำดับ	ชื่อ	ประเภท	ข้อมูลที่เก็บ
1	Brand	String	เก็บข้อมูลแบรนด์รถยนต์
2	Registration	String	เก็บข้อมูลทะเบียนรถยนต์
3	InsuranceDate	String	เก็บข้อมูลวันหมดประกันภัย
4	TaxDate	String	เก็บข้อมูลวันหมดอายุภาษี
5	FirstPayment	String	เก็บข้อมูลวันที่ชำระงวด
6	Repayment	Number	เก็บข้อมูลจำนวนงวด
7	Installment	Number	เก็บข้อมูลค่างวด
8	SharedEmail	Array	เก็บข้อมูลอีเมลของผู้ที่ต้องการให้ใช้งานร่วมกัน

ตารางที่ ข.2 โครงสร้างข้อมูลรายการบันทึกการเติมเชื้อเพลิง

ลำดับ	ชื่อ	ประเภท	ข้อมูลที่เก็บ
1	Registration	String	เก็บข้อมูลทะเบียนรถยนต์
2	Miles	Number	เก็บข้อมูลระยะทาง
3	Cost	Number	เก็บข้อมูลค่าใช้จ่ายในการเติมเชื้อเพลิง

ตารางที่ ข.3 โครงสร้างข้อมูลรายการบันทึกการซ่อมบำรุง

ลำดับ	ชื่อ	ประเภท	ข้อมูลที่เก็บ
1	Registration	String	เก็บข้อมูลทะเบียนรถยนต์
2	Miles	Number	เก็บข้อมูลระยะทาง
3	Service	Array	เก็บข้อมูลรายการซ่อมบำรุง
4	Servicer	String	เก็บข้อมูลผู้ให้บริการ
5	Note	String	เก็บข้อมูลรายละเอียดเพิ่มเติม
6	ImgReceipt	String	เก็บข้อมูล URL ของรูปภาพ

ตารางที่ ข.4 โครงสร้างข้อมูลบันทึกช่วยจำ

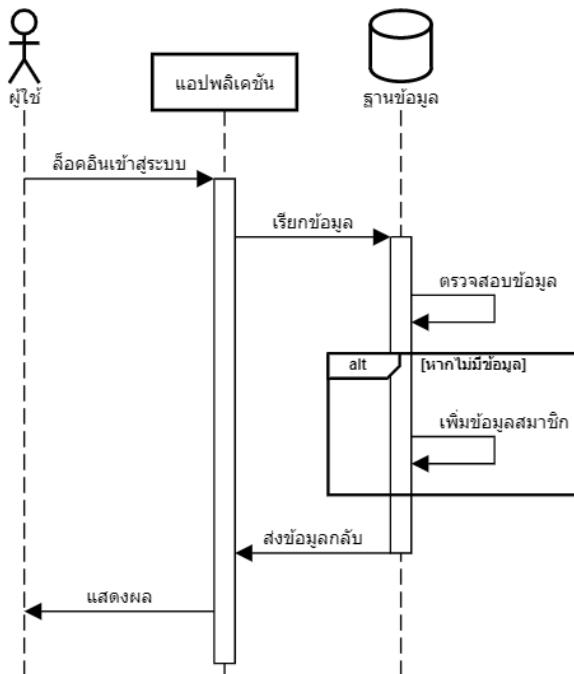
ลำดับ	ชื่อ	ประเภท	ข้อมูลที่เก็บ
1	Email	String	เก็บข้อมูลอีเมลของผู้ใช้งาน
2	Date	TimeStamp	เก็บข้อมูลวันที่
3	Title	String	เก็บข้อมูลชื่อเรื่องที่บันทึก
4	Detail	String	เก็บข้อมูลรายละเอียดที่บันทึก
5	SharedEmail	String	เก็บข้อมูลอีเมลของผู้ที่ต้องการให้ใช้งานร่วมกัน

ภาคผนวก ค

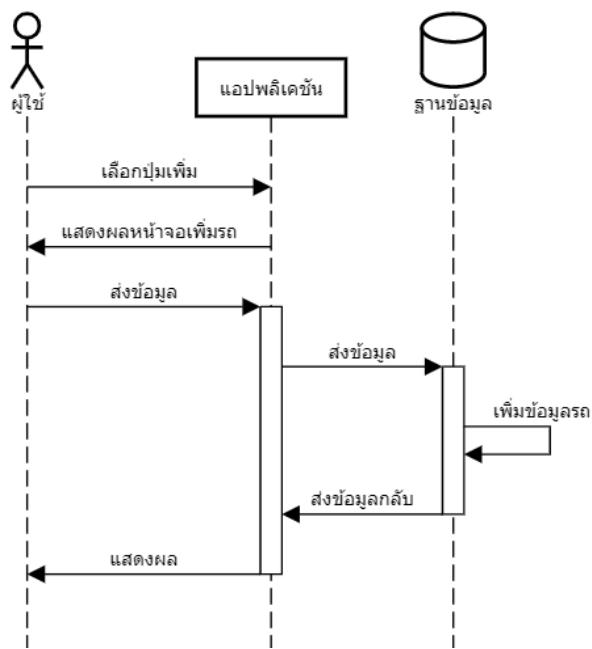
แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์

แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์

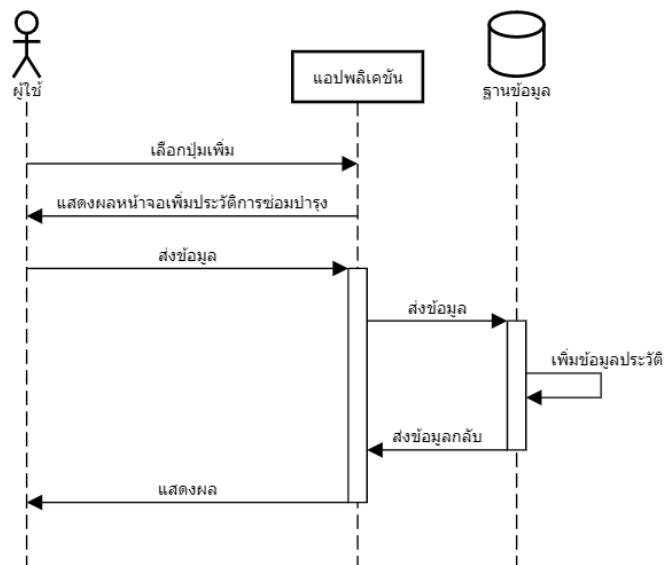
แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์ของแอปพลิเคชันการบำรุงรักษารถยนต์และบันทึกช่วยว่าյจำอ่อนลื่นจะมีด้วยกันทั้งหมด 7 แผนภาพตามในแต่ละยูสเคสของแผนภาพแสดงหน้าที่ที่ระบบต้องกระทำ โดยจะมีเข้าสู่ระบบ เพิ่มรายการรถยนต์ บันทึกประวัติการซ่อมบำรุงรถยนต์ บันทึกระยะเวลาทางสะสมในการเดินทาง บันทึกค่าใช้จ่าย บันทึกช่วยจำ และใช้งานร่วมกัน



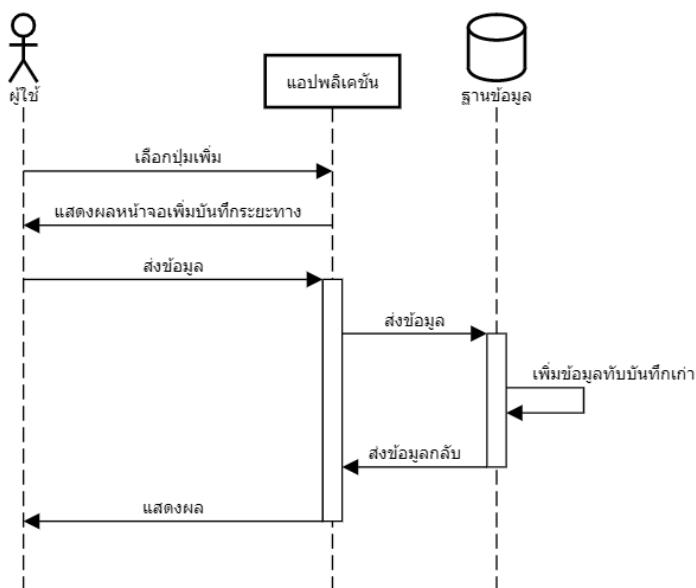
รูปที่ ค.1 แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์ของเข้าสู่ระบบ



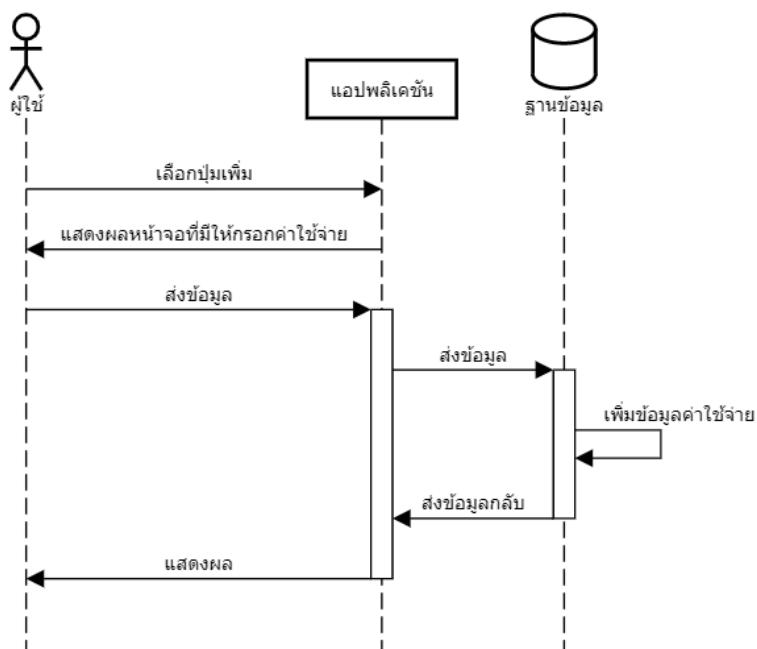
รูปที่ ค.2 แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์ของเพิ่มรายการรถยนต์



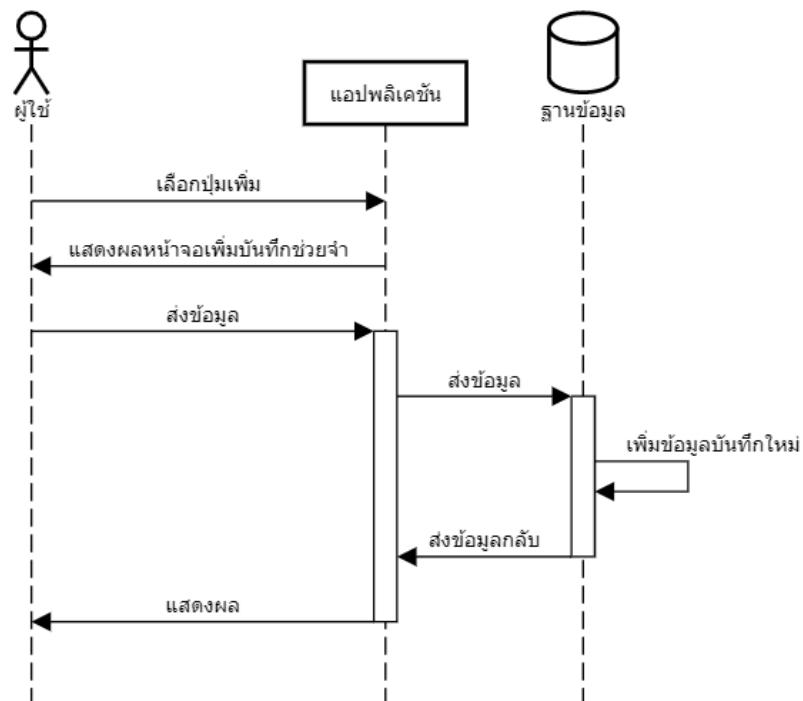
รูปที่ ค.3 แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์ของบันทึกประวัติการซ่อมบำรุงรถยนต์



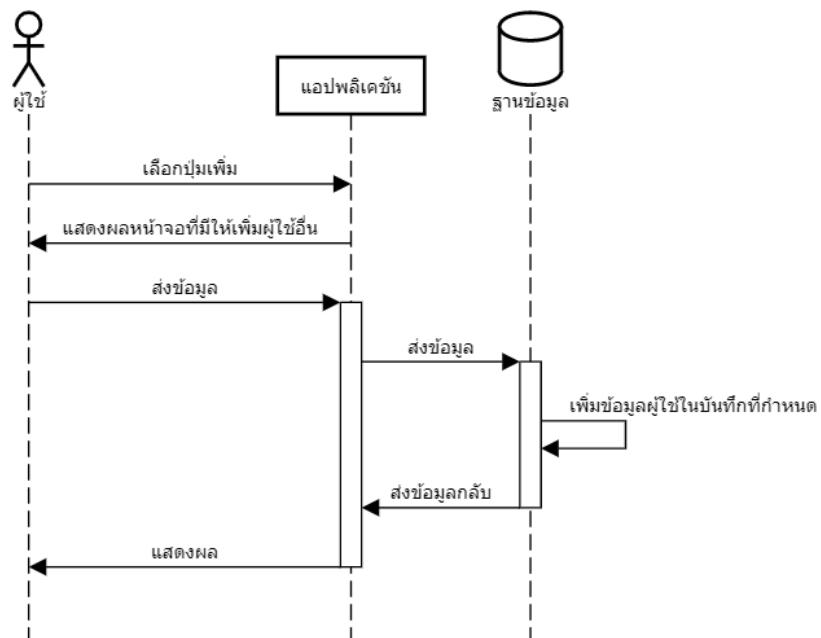
รูปที่ ค.4 แผนภาพการทำงานแบบระบบลำดับปฏิสัมพันธ์ของบันทึก registryทางสะสานในการเดินทาง



รูปที่ ค.5 แผนภาพการทำงานแบบระบบลำดับปฏิสัมพันธ์ของบันทึกที่เก่าใช้จ่าย



รูปที่ ค.6 แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์ของบันທຶກໜ່າຍຈໍາ



รูปที่ ค.7 แผนภาพการทำงานแบบลำดับปฏิสัมพันธ์ของໃໝ່ງນ່ວມກັນ

ภาคผนวก ง
รหัสต้นฉบับของโปรแกรม

ໂປຣແກຣມຄວບຄຸມການທ່າງຈານຂອງຮະບບຍືນຍັນຕົວຕົນ

```

import React, {useContext} from 'react';
import {
  View,
  Image,
  Platform,
  StyleSheet,
  ScrollView
} from 'react-native';
import SocialButton from '../components/SocialButton';
import {AuthContext} from '../navigation/AuthProvider';

const LoginScreen = () => {
  const {googleLogin, fbLogin} = useContext(AuthContext);
  return (
    <ScrollView contentContainerStyle={styles.container}>
      <Image
        source={require('../assets/def.jpg')}
        style={styles.logo}
      />
      {Platform.OS === 'android' ? (
        <View>
          <SocialButton
            buttonTitle="Sign In with Facebook"
            btnType="facebook"
            color="#4867aa"
            backgroundColor="#e6eaf4"
            onPress={() => fbLogin()}
          />
          <SocialButton
            buttonTitle="Sign In with Google"
            btnType="google"
            color="#de4d41"
            backgroundColor="#f5e7ea"
            onPress={() => googleLogin()}
          />
        </View>
      ) : null}
    </ScrollView>
  );
};

export default LoginScreen;

```

โปรแกรมควบคุมการทำงานของระบบบันทึกรายการซ่อมบำรุงของรถยนต์

```

import React, { useState, useContext, useEffect, useCallback } from
"react";
import {
  StyleSheet,
  ScrollView,
  View,
  Text,
  TouchableOpacity,
  FlatList,
  RefreshControl,
} from 'react-native';
import { Card, Tabs, Tab, Icon } from 'native-base';
import { widthPercentageToDP as wp, heightPercentageToDP as hp } from
'react-native-responsive-screen';
import GasCard from '../components/GasCard';
import firestore from '@react-native-firebase/firestore';
import ServiceCard from '../components/ServiceCard';
import moment from 'moment';
import { AnimatedCircularProgress } from 'react-native-circular-
progress';

const DetailsCar = (props) => {
  const { item } = props.route.params
  const [service, setService] = useState(null);
  const [kilo, setKilo] = useState();
  const [gas, setGas] = useState(null);
  const [loading, setLoading] = useState(true);
  const [deleted, setDeleted] = useState(false);
  const [refreshing, setRefreshing] = useState(false);
  const [totalPrice, setTotalPrice] = useState(0)
  const [totalService, setTotalService] = useState(0);
  const [itemDate, setitemDate] = useState({ ...item, diff:
moment(item.Insurance).diff(moment(), "days") });
  const [itemBattery, setitemBattery] = useState({});
  const [Total, setTotal] = useState(0);

  useEffect(() => {
    setTotal(totalPrice + totalService);
  }, [totalPrice, totalService])

  const onRefresh = useCallback(async () => {
    setRefreshing(true);
    fetchGas().then(() => {
      setRefreshing(false);
    });
    fetchService().then(() => {
      setRefreshing(false);
    });
  }, [refreshing]);

  const fetchGas = async () => {
    try {
      const list = [];
      await firestore()

```

```

.collection('Gas')
.where('CarRegistration', '==', item.CarRegistration)
.orderBy('GasDate', 'desc')
.get()
.then((querySnapshot) => {
  querySnapshot.forEach((doc) => {
    const {
      Raka,
      GasDate,
    } = doc.data();
    list.push({
      id: doc.id,
      Raka,
      GasDate,
    });
  });
})
.setTotalPrice(list.reduce((prev, cur) => {
  return prev + Number(cur.Raka)
}, 0))
.setGas(list);
if (loading) {
  setLoading(false);
}
} catch (e) {
  console.log(e);
}
},
useEffect(() => {
  fetchGas();
}, []);
useEffect(() => {
  fetchGas();
  setDeleted(false);
}, [deleted]);
const fetchService = async () => {
  try {
    const list1 = [];
    await firestore()
      .collection('Service')
      .where('CarRegistration', '==', item.CarRegistration)
      .orderBy('ServiceDate', 'desc')
      .get()
      .then((querySnapshot) => {
        console.log('Total : ', querySnapshot.size);
        querySnapshot.forEach((doc) => {
          const {
            Cost,
            Note,
            Receipt,
            Service,
            ServiceDate,
            ServiceProvider,
            TotalKilo,
          } = doc.data();
          list1.push({
            id1: doc.id,
            Cost,
            Note,
            Receipt,
            Service,
            ServiceDate,
            ServiceProvider,
            TotalKilo,
          });
        });
      });
  }
});

```

```

        Cost: Cost,
        Note: Note,
        Receipt: Receipt,
        Service: Service,
        ServiceDate: ServiceDate,
        ServiceProvider: ServiceProvider,
        TotalKilo: TotalKilo,
    );
);
});
);
setTotalService(list1.reduce((ser, vice) => {
    return ser + Number(vice.Cost)
}, 0))
setKilo(list1[0].TotalKilo);
setItemBattery({ ...list1, diff:
moment(list1[0].ServiceDate).add(1, 'Y').add(8, 'M').diff(moment(),
'days') })
setService(list1);
console.log(list1[0].TotalKilo);
if (loading) {
    setLoading(false);
}
} catch (e) {
    console.log(e);
}
};
useEffect(() => {
    fetchService();
}, []);
useEffect(() => {
    fetchService();
    setDeleted(false);
}, [deleted]);
const ListHeader = () => {
    return null;
};
return (
<Tabs>
<Tab heading="Home">
<View style={styles.container}>
<ScrollView
    refreshControl={
        <RefreshControl
            refreshing={refreshing}
            onRefresh={onRefresh}
        />
    }
</Tab>
<Card style={styles.contbox}>
<View style={styles.Card}>
<View style={{ flexDirection: 'row' }}>
<View>
    <Icon style={styles.icon}
        name='car' />
</View>
<View>
    <Text style={styles.brand}>{item.Brand}</Text>
    <Text
        style={styles.CarRegistration}>{item.CarRegistration}</Text>

```

```

        </View>
    </View>
    <View>
        <Text style={styles.dateTax1}>Expiration
date</Text>
    </View>
    <View style={{ flexDirection: 'row' }}>
        <View>
            <AnimatedCircularProgress style={styles.pro1}
                size={95}
                width={17}
                fill={200 * 100 / 365}
                tintColor="green"
                onAnimationComplete={() =>
                    console.log('onAnimationComplete')}
                backgroundColor="red" >
            {
                (fill) => (
                    <Text >
                        {itemDate.diff} Day
                    </Text>
                )
            }
        </AnimatedCircularProgress>
    </View>
    <View>
        <AnimatedCircularProgress style={styles.pro2}
            size={95}
            width={17}
            fill={450 * 100 / 630}
            tintColor="green"
            onAnimationComplete={() =>
                console.log('onAnimationComplete')}
            backgroundColor="red" >
        {
            (fill) => (
                <Text>
                    {itemBatterry.diff} Day
                </Text>
            )
        }
    </AnimatedCircularProgress>
</View>
<View>
    <AnimatedCircularProgress style={styles.pro3}
        size={95}
        width={17}
        fill={kilo * 100 / 60000}
        tintColor="red"
        onAnimationComplete={() =>
            console.log('onAnimationComplete')}
        backgroundColor="green" >
    {
        (fill) => (
            <Text>
                {kilo}Km
            </Text>
        )
    }

```

```

        }
    </AnimatedCircularProgress>
</View>
</View>
<View style={{ flexDirection: 'row' }}>
    <Text style={styles.A}>Tax</Text>
    <Text style={styles.B}>Battery</Text>
    <Text style={styles.C}>Break</Text>
</View>
</View>
</Card>
<Card style={styles.Gas}>
    <Text style={styles.TextCen}>Recent Fuel Logs</Text>
    <FlatList
        data={gas}
        numColumns={1}
        keyExtractor={(item) => item.id.toString()}
        renderItem={({ item }) => (
            <GasCard item={item} parentProps={props} />
        )}
        keyExtractor={(item) => item.id}
        ListHeaderComponent={ListHeader}
        ListFooterComponent={ListHeader}
        showsVerticalScrollIndicator={false}
    />
    <Card style={styles.transparent}>
        <TouchableOpacity>
            <Text style={styles.Text} onPress={() => {
                props.navigation.navigate('GasTotal', { item: item })
            }}>Show More</Text>
        </TouchableOpacity>
        <TouchableOpacity>
            <Text style={styles.Text} onPress={() => {
                props.navigation.navigate('addgas', { item: item })
            }}>Add</Text>
        </TouchableOpacity>
    </Card>
    <Card style={styles.Gas}>
        <Text style={styles.TextCen}>Recent Maintenance Logs</Text>
        <FlatList
            data={service}
            numColumns={1}
            keyExtractor={(item) => item.id.toString()}
            renderItem={({ item }) => (
                <ServiceCard item={item} parentProps={props} />
            )}
            keyExtractor={(item) => item.id}
            ListHeaderComponent={ListHeader}
            ListFooterComponent={ListHeader}
            showsVerticalScrollIndicator={false}
        />
        <Card style={styles.transparent}>
            <TouchableOpacity>
                <Text style={styles.Text} onPress={() => {
                    props.navigation.navigate('ServiceTotal', { item: item })
                }}>Show More</Text>
            </TouchableOpacity>
        </Card>
    </Card>
</View>

```

```

        <TouchableOpacity>
            <Text style={styles.Text} onPress={() => {
props.navigation.navigate('addService', { item: item })
}}>Add</Text>
        </TouchableOpacity>
        </Card>
    </Card>
    <ScrollView>
    </View>
</Tab>
<Tab heading="Stat">
    <View style={styles.Brand}>
        <View style={styles.con2}>
            <View style={styles.CarTopic}>
                <Text style={{ fontSize: 25, color: 'black',
alignSelf: 'flex-start', marginLeft: 20, }}>{item.Brand}</Text>
                <Text style={{ fontSize: 15, color: 'black',
alignSelf: 'flex-start', marginLeft: 20
}}>{item.CarRegistration}</Text>
            </View>
            <View style={styles.Brand2}>
                <View style={styles.CardDetail}>
                    <Text style={{ fontSize: 18, color: 'black'
}}>Fuel</Text>
                    <Text style={{ fontSize: 18, color: 'black'
}}>{totalPrice}</Text>
                </View>
                <View style={styles.CardDetail}>
                    <Text style={{ fontSize: 18, color: 'black'
}}>Maintenance</Text>
                    <Text style={{ fontSize: 18, color: 'black'
}}>{totalService}</Text>
                </View>
                <View style={{ alignSelf: 'flex-end', flexDirection:
'row', fontSize: 20, marginTop: hp('3%') }}>
                    <Text style={{ marginHorizontal: wp('3%'),
fontSize: 20, color: 'black', marginBottom: 10 }}>Total</Text>
                    <Text style={{ marginHorizontal: wp('3%'),
fontSize: 20, color: 'black', marginBottom: 10 }}>{Total}</Text>
                    <Text style={{ marginHorizontal: wp('3%'),
fontSize: 20, color: 'black', marginBottom: 10 }}>THB</Text>
                </View>
            </View>
        </View>
    </Tab>
</Tabs>
);
};

export default DetailsCar;

```

โปรแกรมควบคุมการทำงานของระบบบันทึกช่วยจำ

```

import React, { useEffect, useState, useContext, useCallback } from
'react';
import {
  View,
  ScrollView,
  StyleSheet,
  FlatList,
  SafeAreaView,
  Alert,
  StatusBar,
  TextInput,
  RefreshControl,
} from 'react-native';
import SkeletonPlaceholder from "react-native-skeleton-placeholder";
import MemosCard from '../components/MemosCard';
import storage from '@react-native-firebase/storage';
import firestore from '@react-native-firebase/firestore';
import { AuthContext } from '../navigation/AuthProvider';

const SharedMemos = (props) => {
  const [refreshing, setRefreshing] = useState(false);
  const { user } = useContext(AuthContext);
  const [loading, setLoading] = useState(true);
  const [deleted, setDeleted] = useState(false);
  const [filteredData, setfilteredData] = useState([]);
  const [masterData, setmasterData] = useState([]);
  const [search, setsearch] = useState('');
  const onRefresh = useCallback(async () => {
    setRefreshing(true);
    fetchMemos().then(() => {
      setRefreshing(false);
    });
  }, [refreshing]);
  const fetchMemos = async () => {
    try {
      const list = [];
      await firestore()
        .collection('Memos')
        .where('Email', '==', user.email || 'Name', '==',
user.displayName)
        .orderBy('postTime', 'desc')
        .get()
        .then((querySnapshot) => {
          console.log('Total Memos: ', querySnapshot.size);
          querySnapshot.forEach((doc) => {
            const {
              title,
              memosDetails,
              postImg,
              postTime,
              category,
            } = doc.data();
            list.push({
              title,
              memosDetails,
              postImg,
              postTime,
              category,
            });
          });
        });
    } catch (error) {
      console.error(error);
    }
  };
  useEffect(() => {
    fetchMemos();
  }, []);
  return (
    <SafeAreaView style={styles.container}>
      <FlatList
        data={list}
        keyExtractor={(item) => item.id}
        renderItem={({ item }) =>
          <MemosCard
            item={item}
            onDelete={() => {
              setDeleted(true);
              setTimeout(() => {
                setDeleted(false);
                setmasterData([...masterData, item]);
                setfilteredData([...filteredData, item]);
              }, 500);
            }}
          />
        }
        refreshing={refreshing}
        onRefresh={onRefresh}
      />
    </SafeAreaView>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    padding: 10,
  },
  header: {
    margin: 10,
  },
  input: {
    width: '100%',
    height: 40,
    border: 1px solid #ccc,
    padding: 5,
    margin: 10 0,
  },
  button: {
    width: '100%',
    height: 40,
    background-color: '#007bff',
    color: 'white',
    border: 1px solid #007bff,
    border-radius: 5px,
    padding: 10,
    margin: 10 0,
  },
  buttonText: {
    color: 'white',
  },
  list: {
    width: '100%',
    height: 100px,
    border: 1px solid #ccc,
    padding: 10,
    margin: 10 0,
  },
  footer: {
    margin: 10,
  },
});

```

```

        id: doc.id,
        Title: title,
        MemosDetails: memosDetails,
        postTime: postTime,
        postImg,
        category,
    );
}
);
);
setfilterdData(list.map(l=>{...l,color:l.category=="Education"? "#58D68D":(l.category=="Family"? "#Ff94b6": "#2FC2DF") })));
setmasterData(list.map(l=>{...l,color:l.category=="Education"? "#58D68D":(l.category=="Family"? "#Ff94b6": "#2FC2DF") })));
if (loading) {
    setLoading(false);
}
console.log('Memos: ', filterdData);
} catch (e) {
    console.log(e);
}
};
useEffect(() => {
    fetchMemos();
}, []);
useEffect(() => {
    fetchMemos();
    setDeleted(false);
}, [deleted]);
const handleDelete = (memosId) => {
    Alert.alert(
        'Delete Memos',
        'Are you sure?',
        [
            {
                text: 'Cancel',
                onPress: () => console.log('Cancel Pressed!'),
                style: 'cancel',
            },
            {
                text: 'Confirm',
                onPress: () => deleteMemos(memosId),
            },
        ],
        { cancelable: false },
    );
};
const deleteMemos = (memosId) => {
    console.log('Current Memos Id: ', memosId);
    firestore()
        .collection('Memos')
        .doc(memosId)
        .get()
        .then((documentSnapshot) => {
            if (documentSnapshot.exists) {
                const { memosImg } = documentSnapshot.data();
                if (memosImg != null) {
                    const storageRef = storage().refFromURL(memosImg);
                    const imageRef = storage().ref(storageRef.fullPath);

```

```

        imageRef
          .delete()
          .then(() => {
            console.log(`\$${memosImg} has been deleted
successfully.`);
            deleteFirestoreData(memosId);
          })
          .catch((e) => {
            console.log('Error while deleting the image. ', e);
          });
        // If the post image is not available
      } else {
        deleteFirestoreData(memosId);
      }
    }
  );
};

const deleteFirestoreData = (memosId) => {
  firestore()
    .collection('Memos')
    .doc(memosId)
    .delete()
    .then(() => {
      Alert.alert(
        'Memos deleted!',
        'Your Memos has been deleted successfully!',
      );
      setDeleted(true);
    })
    .catch((e) => console.log('Error deleting posst.', e));
};

const searchFilter = (text) => {
  if (text) {
    const newData = masterData.filter((item) => {
      const itemData = item.Title ? item.Title.toUpperCase()
        : ''.toUpperCase();
      const textData = text.toUpperCase();
      const itemCat = item.category ?
        item.category.toUpperCase(): ''
      const itemdetail = item.MemosDetails ?
        item.MemosDetails.toUpperCase(): ''
      return itemData.indexOf(textData) > -1 ||
        itemCat.indexOf(textData) >-1 || itemdetail.indexOf(textData) >-1;
    });
    setfilterdData(newData);
    setsearch(text);
  } else {
    setfilterdData(masterData);
    setsearch(text);
  }
};
const ListHeader = () => {
  return null;
};
return (
  <SafeAreaView style={{ flex: 1 }}>
    {loading ? (
      <ScrollView

```

```

        style={{ flex: 1 }}
        contentContainerStyle={{ alignItems: 'center' }}>
      <SkeletonPlaceholder>
        <View style={{ marginTop: 10, marginBottom: 30 }}>
          <View
            style={{ marginTop: 6, width: 350, height: 200,
borderRadius: 4 }}>
            />
          </View>
        </SkeletonPlaceholder>
      <SkeletonPlaceholder>
        <View style={{ marginTop: 10, marginBottom: 30 }}>
          <View
            style={{ marginTop: 6, width: 350, height: 200,
borderRadius: 4 }}>
            />
          </View>
        </SkeletonPlaceholder>

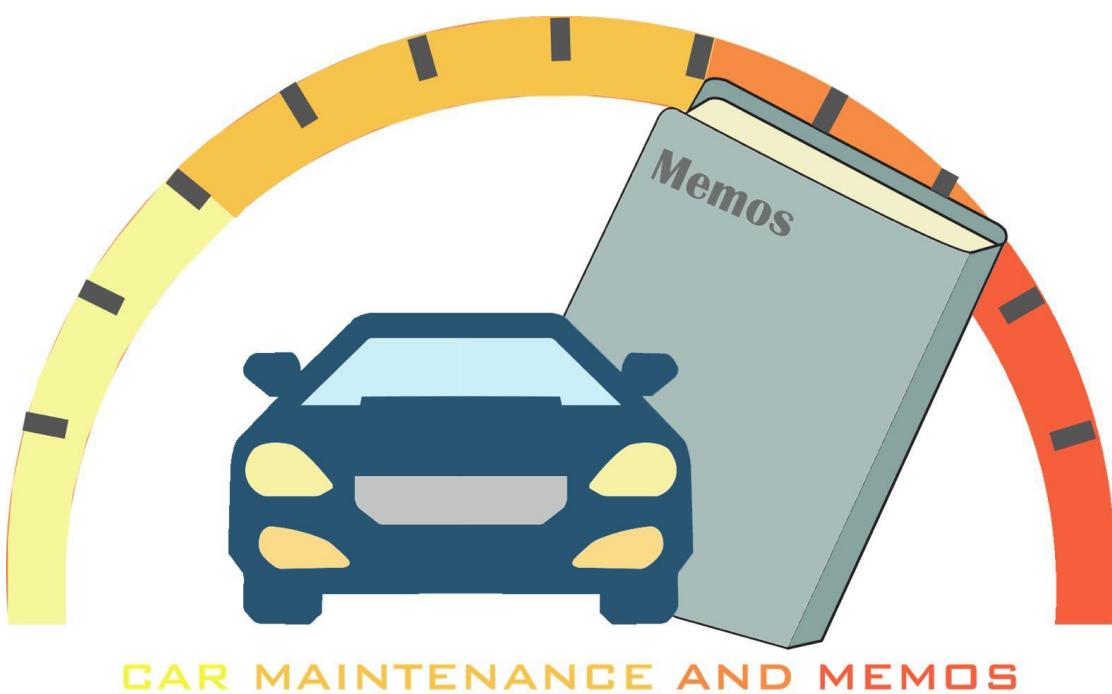
      </ScrollView>
    ) : (
      <View style={styles.parentView}>
        <StatusBar backgroundColor="white" barStyle="dark-
content" />
        <TextInput
          style={styles.search}
          placeholder="search..."
          value={search}
          underlineColorAndroid="transparent"
          onChangeText={(text) => searchFilter(text)}
        />
        <FlatList
          style={styles.flatList}
          data={filteredData}
          numColumns={2}
          keyExtractor={(item, index) => index.toString()}
          //ItemSeparatorComponent = {ItemSeparatorView}
          renderItem={({ item }) => (
            <MemosCard item={item} onDelete={handleDelete}
parentProps={props} />
          )}
          keyExtractor={(item) => item.id}
          ListHeaderComponent={ListHeader}
          ListFooterComponent={ListHeader}
          showsVerticalScrollIndicator={false}
          refreshControl={
            <RefreshControl refreshing={refreshing}
onRefresh={onRefresh} />
          }
        />
      </View>
    ) )
  </SafeAreaView>
);
};

export default SharedMemos;

```

ภาคผนวก จ
คู่มือการใช้งาน

คู่มือการใช้งาน
แอปพลิเคชันบันทึกการบำรุงรักษารถยนต์และบันทึกช่วยจำ
ออนไลน์



สาขาวิชาครุศาสตร์วิศวกรรม
คณะครุศาสตร์อุตสาหกรรมและเทคโนโลยี
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2563

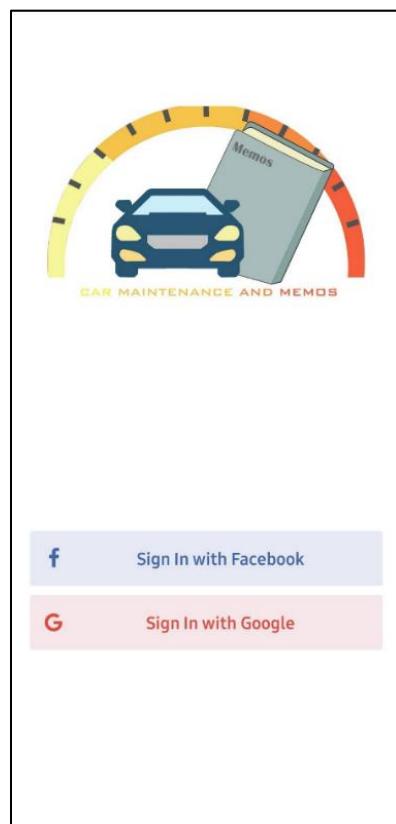
1. คำแนะนำเบื้องต้น

ในการใช้งานแอปพลิเคชันบันทึกการซ่อมบำรุงรถยนต์และบันทึกช่วยจำออนไลน์ ควรศึกษาการใช้งานจากคู่มือการใช้งานก่อนใช้ เพื่อการใช้แอปพลิเคชันให้เกิดประสิทธิภาพและให้เกิดประโยชน์สูงสุดแก่ผู้ใช้งาน

2. หน้าจอการใช้งานและการแสดงผลของผู้ใช้งาน

2.1 หน้าจอเข้าสู่ระบบ

จากภาพที่ 1 ผู้ใช้งานสามารถเข้าสู่ระบบและใช้งานแอปพลิเคชันได้ด้วยบัญชีเฟซบุ๊กและบัญชีกูเกิล หากใช้งานเป็นครั้งแรกระบบจะทำการสมัครสมาชิกให้โดยอัตโนมัติ ดังรูปที่ จ.1



รูปที่ จ.1 หน้าจอเข้าสู่ระบบ

2.2 หน้าจอรายการรักษาดูแล

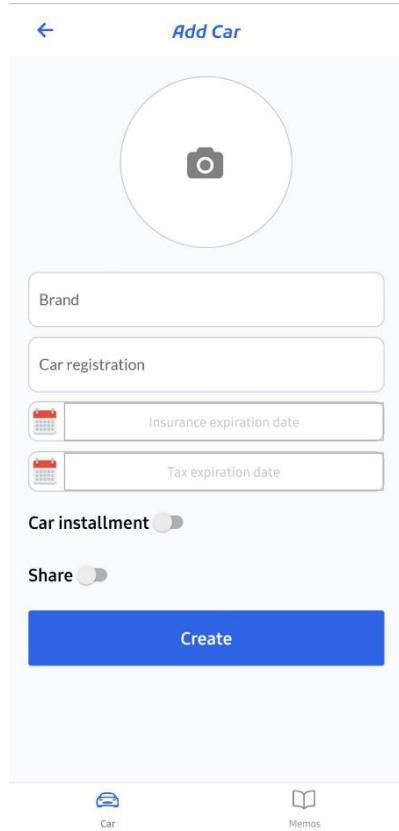
เมื่อผู้ใช้งานทำการเข้าสู่ระบบเข้ามาระบบจะนำผู้ใช้มาสู่หน้าจอแสดงรายการรักษาดูแลโดยจะมีปุ่มเมนูมุมซ้ายบน และปุ่มสำหรับเพิ่มรายการรักษาดูแลที่ด้านล่าง บนซ้าย หากเป็นผู้ใช้งานใหม่จะไม่มีรายการรักษาดูแล ต้องทำการเพิ่มรายการรักษาดูแลก่อน ดังรูปที่ จ.2



รูปที่ จ.2 หน้าจอรายการรักษาดูแล

2.3 หน้าจอเพิ่มรายการรถยนต์

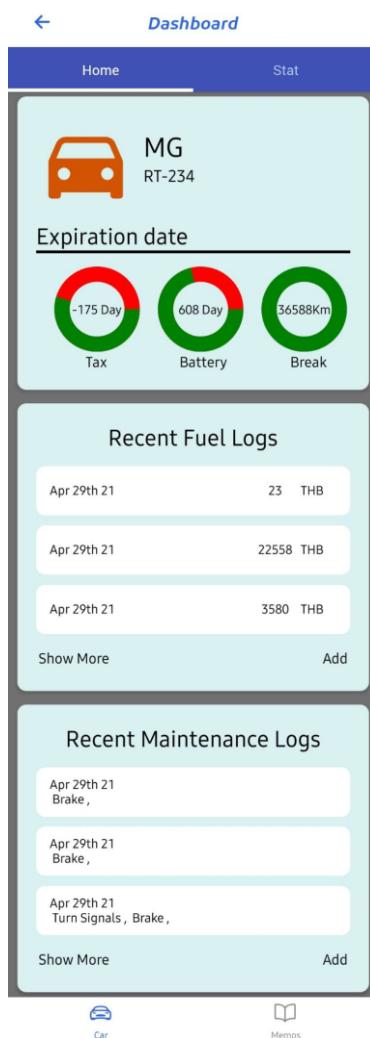
เมื่อทำการกดปุ่มเพิ่มรายการรถยนต์ระบบจะนำทางมาสู่หน้าจอเพิ่มรายการรถยนต์ โดยภายในหน้าจอผู้ใช้งานจะสามารถเพิ่มรายละเอียดยี่ห้อ ทะเบียนรถ วันหมดภาคี วันหมดประกัน รูปภาพ การผ่อนรถ และแชร์ เมื่อผู้ใช้งานกรอกข้อมูลเสร็จสิ้นให้กดที่ปุ่ม Create เพื่อทำการเพิ่มข้อมูลลงไปในฐานข้อมูล ดังรูปที่ จ.3



รูปที่ จ.3 หน้าจอเพิ่มรายการรถยนต์

2.4 หน้าจอแสดงรายละเอียด

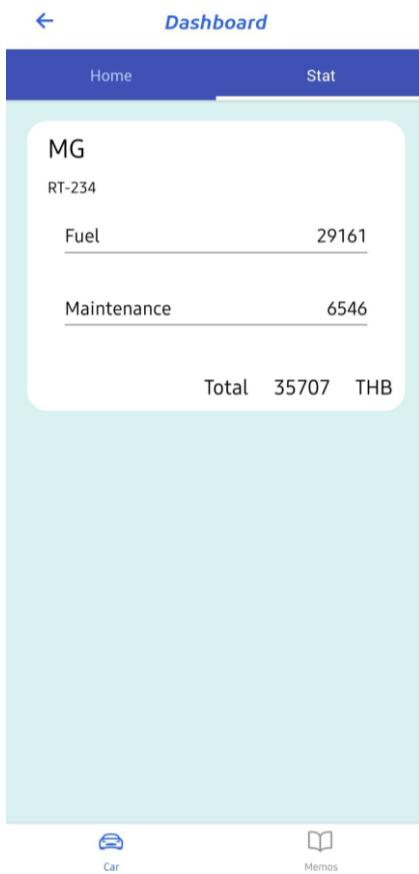
หากผู้ใช้งานมีรถยนต์ที่ได้ทำการเพิ่มลงฐานข้อมูลแล้ว ผู้ใช้สามารถเลือกรถยนต์ที่ต้องการจากหน้าจอรายการรถยนต์เพื่อที่จะเข้าสู่หน้าจอแสดงรายละเอียดรถยนต์ โดยระบบจะแสดงจำนวนวันที่จะหมดอายุของภาษี แบตเตอรี่ และเบรก อีกทั้งยังแสดงรายการประวัติการเติมเชื้อเพลิง และประวัติการซ่อมบำรุงของรถยนต์ที่ผู้ใช้งานได้เลือกไว้ ผู้ใช้งานสามารถเพิ่มรายการเชื้อเพลิง และการซ่อมบำรุงได้ที่ปุ่ม Add ในแต่ละรายการ ดังรูปที่ จ.4



รูปที่ จ.4 หน้าจอแสดงรายละเอียดรถยนต์

2.5 หน้าจอแสดงสถิติการเงินของรถยนต์

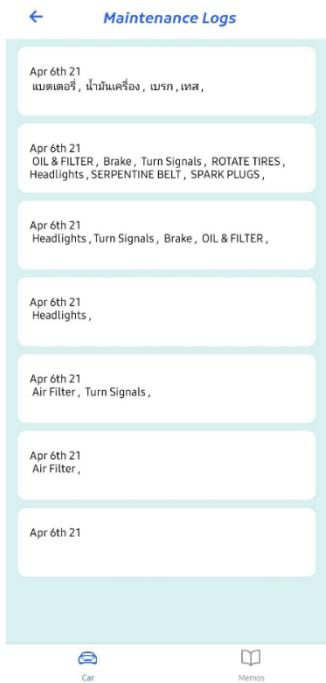
ผู้ใช้งานสามารถเข้ามาสู่หน้าจอนี้ได้โดยการลากนิวจากขวาไปซ้าย ระบบจะนำผู้ใช้งานมาสู่หน้าจอแสดงสถิติการเงินของรถยนต์ โดยระบบจะคำนวณทั้งหมดของรถยนต์ที่ผู้ใช้งานเลือก ดังรูปที่ จ.5



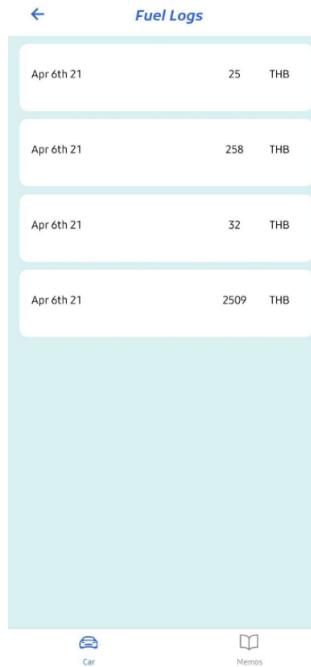
รูปที่ จ.5 หน้าจอแสดงแสดงสถิติการเงินของรถยนต์

2.6 หน้าจอแสดงรายการบันทึกการซ่อมบำรุงและหน้าจอแสดงรายการบันทึกเชื้อเพลิง

ระบบจะนำเข้ามาสู่หน้าจอนี้ เมื่อผู้ใช้งานได้ทำการกดปุ่ม Show more ในแต่ละประวัติรายการโดยจะแสดงข้อมูลวันที่ และบริการที่ใช้ในรายการบันทึกการซ่อมบำรุง และจะแสดงวันที่กับจำนวนเงินในรายการบันทึกเชื้อเพลิง ดังรูปที่ จ.6 และรูปที่ จ.7



รูปที่ จ.6 หน้าจอแสดงแสดงรายการบันทึกซ่อมบำรุง



รูปที่ จ.7 หน้าจอแสดงแสดงรายการบันทึกเชื้อเพลิง

2.7 หน้าจอแสดงรายการเบ็ดเตล็ดบันทึกรายการซ่อมบำรุง

หากผู้ใช้งานทำการเลือกรายการประวัติบันทึกรายการซ่อมบำรุงรายการใดรายการหนึ่ง ระบบจะทำการนำผู้ใช้งานมาสู่หน้าจอแสดงรายการเบ็ดเตล็ดบันทึกรายการซ่อมบำรุงของรายการที่ผู้ใช้งานได้ทำการเลือกโดยจะมีข้อมูลวันที่ ไมล์ ค่าใช้จ่าย บริการที่ใช้ ผู้ให้บริการ บันทึกเพิ่มเติม และใบเสร็จ ดังรูปที่ จ.8



รูปที่ จ.8 หน้าจอแสดงรายละเอียดบันทึกการซ่อมบำรุง

2.8 หน้าจอเพิ่มรายการบันทึกการซ่อมบำรุงและหน้าจอเพิ่มรายการบันทึกเชือเพลิง

เมื่อผู้ใช้งานทำการเลือกปุ่ม Add ในแต่ละบันทึก ระบบจะนำผู้ใช้งานมาสู่หน้าจอเพิ่มรายการบันทึกของแต่ละบันทึก หากผู้ใช้งานเลือกเพิ่มรายการบันทึกการซ่อมบำรุงจะมีข้อมูลให้กรอกประกอบไปด้วยวันที่ ไมล์ ค่าใช้จ่ายบริการที่ใช้ ผู้ให้บริการ บันทึกเพิ่มเติม และใบเสร็จ หากผู้ใช้งานเลือกเพิ่มรายการบันทึกเชือเพลิงจะมีข้อมูลไมล์และค่าใช้จ่ายให้กรอก ดังรูปที่ จ.9 และรูปที่ จ.10

Add Service

Miles

Cost

Services

▼

Servicer

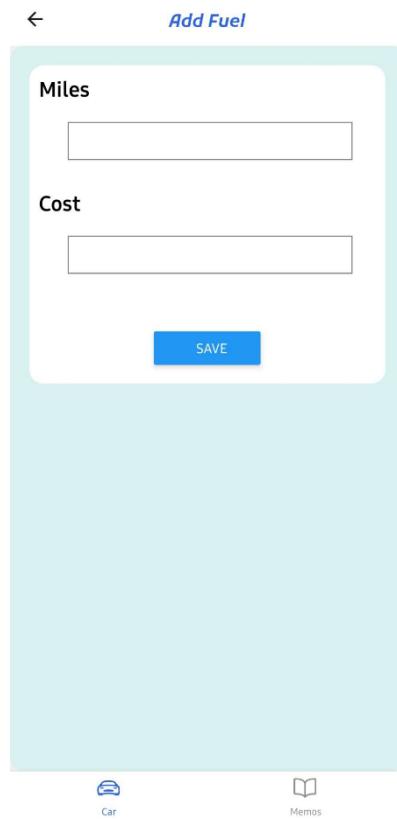
Note

Add Receipt

SAVE

Car Memos

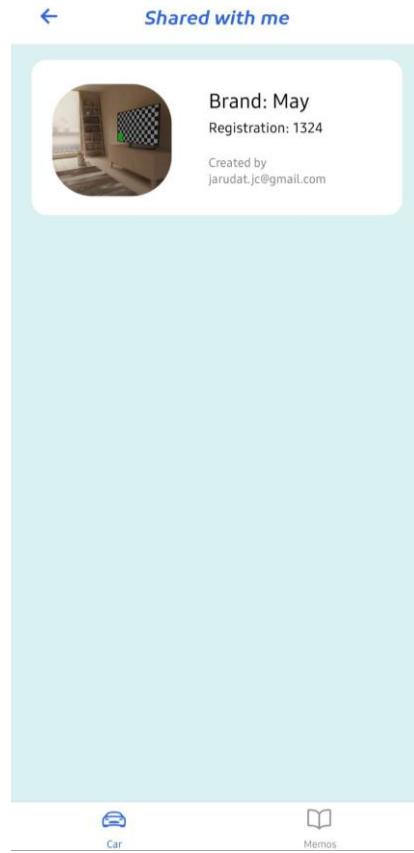
รูปที่ ๑.๙ หน้าจอเพิ่มรายการการซ่อมบำรุง



รูปที่ จ.10 หน้าจอเพิ่มรายการบันทึกเชื้อเพลิง

2.9 หน้าจอแชร์ข้อมูลรถ

หากผู้ใช้งานทำการเลือกเมนูมุมซ้ายบนในหน้าจอรายการรถยนต์ จะมีปุ่ม Shared with me ให้เลือก เมื่อทำการเลือกรอบจะนำผู้ใช้งานมาสู่หน้าจอแชร์ข้อมูลรถ โดยหน้าจอนี้จะแสดงรายการรถยนต์ที่มีการแชร์มาจากผู้ใช้งานอื่น ดังรูปที่ จ.11



รูปที่ จ.11 หน้าจอเพิ่มรายการบันทึกเชื้อเพลิง

2.10 หน้าจอแสดงรายการบันทึกช่วย

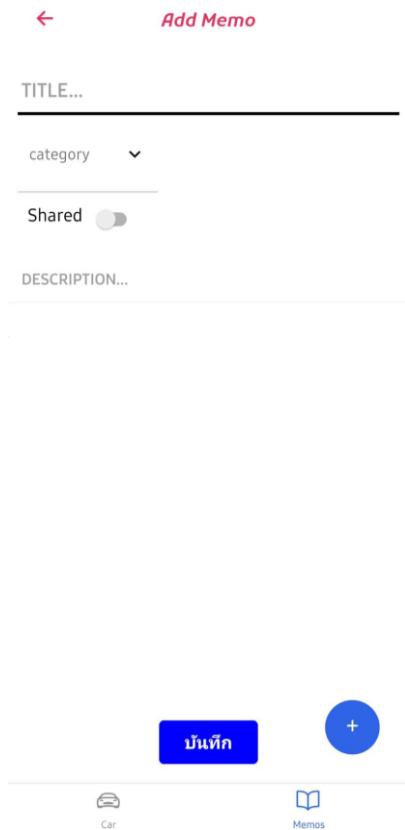
ผู้ใช้งานสามารถมาสู่หน้าจอหนึ่งได้ก็ต่อเมื่อผู้ใช้งานได้ทำการเลือกที่ปุ่ม Memos ในด้านล่างของทุกหน้าจอ โดยภายในหน้าจอจะแสดงรายการบันทึกช่วย จำนวน จำกัด จะมีเมนูบนมุมซ้ายบน และปุ่มเพิ่มที่มุมขวาบน สามารถหารายการบันทึกได้จากช่องค้นหาตรงกลางหน้าจอซึ่งจะสามารถค้นหาได้ตามชื่อหัวข้อบันทึก หมวดหมู่ และรายละเอียดของบันทึก ดังรูปที่ จ.12



รูปที่ จ.12 หน้าแสดงรายการบันทึกช่วยจำ

2.11 หน้าจอเพิ่มรายการบันทึกช่วยจำ

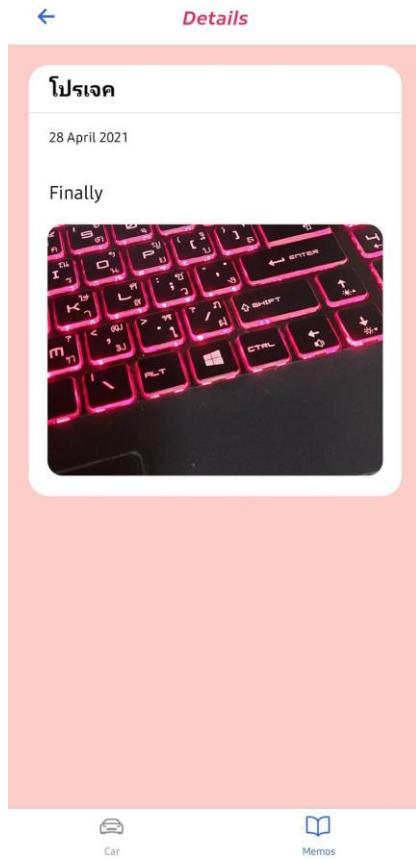
เมื่อผู้ใช้งานต้องการเพิ่มรายการบันทึกช่วยจำ ผู้ใช้งานสามารถเข้าสู่หน้าจอเพิ่มรายการบันทึกช่วยจำได้โดยการเลือกปุ่มเพิ่มที่บนมุมขวาของหน้าจอแสดงรายการบันทึกช่วยจำ โดยรายละเอียดในหน้าจอเพิ่มรายการบันทึกช่วยจำจะมีหัวข้อ หมวดหมู่ คำอธิบายเพิ่มเติม และสามารถเพิ่มรูปภาพได้ที่ปุ่มเพิ่มด้านมุมขวาล่าง ดังรูปที่ จ.13



รูปที่ จ.13 หน้าจอเพิ่มรายการบันทึกช่วยจำ

2.12 หน้าจอแสดงรายละเอียดบันทึกช่วยจำ

หากผู้ใช้งานต้องการดูรายละเอียดของบันทึกช่วยจำ ผู้ใช้งานสามารถทำได้โดยการเลือกรายการบันทึกรายการโดยการแตะรายการหนึ่งในหน้าจอแสดงรายการบันทึกช่วยจำ ระบบจะนำผู้ใช้งานมาสู่หน้าจอแสดงรายละเอียดบันทึกช่วยจำ โดยจะมีหัวข้อ หมวดหมู่ คำอธิบายเพิ่มเติม และรูปภาพ ดังรูปที่ จ.14



รูปที่ จ.14 หน้าจอแสดงรายละเอียดบันทึกช่วยจำ

2.13 หน้าจอแชร์ข้อมูลบันทึกช่วยจำ

ผู้ใช้งานสามารถดูบันทึกช่วยจำที่แชร์ผู้ใช้งานอื่นได้โดยไปที่เมนูมุมซ้ายบนของหน้าจอแสดงรายการบันทึกช่วยจำ จากนั้นให้เลือกที่ "Shared with me" ระบบจะนำผู้ใช้งานมาสู่หน้าจอแชร์ข้อมูลบันทึกช่วยจำที่จะแสดงรายการบันทึกที่ถูกสร้างโดยผู้ใช้งานอื่น ดังรูปที่ จ.15

← Shared with me

search...

General

ສ້າສດີ

ສ້າສດີຄໍານັນ



Car



Memos

ຮູບທີ ຈ.15 ພັນຈອແຊຣ໌ຂໍ້ມູນບັນທຶກຫ່ວຍຈຳ

บรรณานุกรม

ณัฐพล แสนคำ. 2563. **วิธีการใช้งาน Visual Studio Code.** [ระบบออนไลน์]. แหล่งที่มา :

<http://cs.bru.ac.th/สอนวิธีการใช้-visual-studio-code-2/> (20 พฤศจิกายน 2563).

มนัส ช่วยบำรุง. 2562. **หัวเทียนรถยนต์สำคัญอย่างไร .**[ระบบออนไลน์]. แหล่งที่มา :

<https://khaorot.com/car-care-and-maintenance /ไขข้อข้องใจระยะเวลาในการเปลี่ยนหัวเทียน-พร้อมรีวิวหัวเทียนรถยนต์ยี่ห้อไหนดีที่สุด-aid20190605150454175> (20 พฤศจิกายน 2563).

อาคม รวมสุวรรณ. 2560. **ยางกับอายุการใช้งาน .**[ระบบออนไลน์]. แหล่งที่มา :

<https://www.thairath.co.th/content/863012> (20 พฤศจิกายน 2563).

เอกสารชี้.2558. **อายุของแบตเตอรี่.**[ระบบออนไลน์].แหล่งที่มา :

<http://www.chokbuncha.com/เกร็ดความรู้/อายุงานแบตเตอรี่แห่ง-19#:~:text=อายุการใช้งานของ,สั้นลงตามลำดับครับ> (20 พฤศจิกายน 2563).

Android Studio. 2563. **Android Studio.** [ระบบออนไลน์]. แหล่งที่มา :

<https://developer.android.com/studio> (22 พฤศจิกายน 2563).

Champ Milework. 2563. **ดูแลรักษาหัวเทียน .**[ระบบออนไลน์]. แหล่งที่มา :

<https://www.milework.com/blog/spark-plug/> (20 พฤศจิกายน 2563).

designil. 2561. **Adobe XD คืออะไร .**[ระบบออนไลน์]. แหล่งที่มา :

<https://www.designil.com/free-adobe-xd-download.html> (20 พฤศจิกายน 2563).

Firebase. 2563. **Firebase Documentation.** [ระบบออนไลน์]. แหล่งที่มา :

<https://firebase.google.com/docs> (18 พฤศจิกายน 2563).

Four. 2561. **Firebase คืออะไร.** [ระบบออนไลน์]. แหล่งที่มา :

<http://www.mindphp.com/forums/viewtopic.php?f=198&t=44710>

(18 พฤศจิกายน 2563).

Google. ม.ป.ป. **App bars: top .**[ระบบออนไลน์]. แหล่งที่มา :

<https://material.io/components/app-bars-top> (20 พฤศจิกายน 2563).

Google. ม.ป.ป. **Buttons .**[ระบบออนไลน์]. แหล่งที่มา :

<https://material.io/components/buttons> (20 พฤศจิกายน 2563).

Google. ม.ป.ป. **Dialogs** .[ระบบออนไลน์]. แหล่งที่มา :

<https://material.io/components/dialogs> (20 พฤศจิกายน 2563).

Google. ม.ป.ป. **Menus** : top .[ระบบออนไลน์]. แหล่งที่มา :

<https://material.io/components/menus> (20 พฤศจิกายน 2563).

Google. ม.ป.ป. **Navigation drawer** .[ระบบออนไลน์]. แหล่งที่มา :

<https://material.io/components/navigation-drawer> (20 พฤศจิกายน 2563).

Google. ม.ป.ป. **Selection controls** .[ระบบออนไลน์]. แหล่งที่มา :

<https://material.io/components/selection-controls> (20 พฤศจิกายน 2563).

Google. ม.ป.ป. **Text fields** .[ระบบออนไลน์]. แหล่งที่มา :

<https://material.io/components/text-fields> (20 พฤศจิกายน 2563).

Google. ม.ป.ป. **Cards** .[ระบบออนไลน์]. แหล่งที่มา :

<https://material.io/components/cards> (29 พฤศจิกายน 2564).

Google. ม.ป.ป. **Buttons: floating action button** .[ระบบออนไลน์]. แหล่งที่มา :

<https://material.io/components/buttons-floating-action-button#theming> (29 พฤศจิกายน 2564).

Infographic Thailand. 2557. **Adobe XD** โปรแกรมที่นักออกแบบควรมีติดเครื่อง .

[ระบบออนไลน์]. แหล่งที่มา : <http://infographic.in.th/infographic/สอนใช้-adobe-xd-ฉบับมือใหม่หัดดี> (20 พฤศจิกายน 2563).

Inthanon Panyasopa. 2563. **ข้อดีของ โปรแกรม Adobe XD** .[ระบบออนไลน์]. แหล่งที่มา :

<https://www.grappik.com/inside-adobe-xd> (20 พฤศจิกายน 2563).

Inthanon Panyasopa. 2563. **สอนใช้ Adobe XD พื้นฐานโปรแกรมและฟีเจอร์ต่างๆ** .[ระบบออนไลน์]. แหล่งที่มา : <https://www.grappik.com/start-design-with-adobe-xd> (20 พฤศจิกายน 2563).

Maxup420. 2562. **React native หรือ Flutter เราควรเลือกใช้ตัวไหนในการสร้าง**

Mobile app ดี? .[ระบบออนไลน์]. แหล่งที่มา : <https://www.thaiprogrammer.org/2019/11/react-native-vs-flutter/> (18 พฤศจิกายน 2563).

OilSquire. ม.ป.ป. **อายุการใช้งานน้ำมันเครื่อง**.[ระบบออนไลน์]. แหล่งที่มา :

<https://www.oilsquare.com/608933/อายุการใช้งานน้ำมันเครื่อง>

OpenJS Foundation. 2563. **About Node.js** .[ระบบออนไลน์]. แหล่งที่มา :

<https://nodejs.org/en/about> (22 พฤศจิกายน 2563).

- Rachel Nabors. 2563. **React Native Documentation.** [ระบบออนไลน์]. แหล่งที่มา :
<https://reactnative.dev/blog> (18 พฤศจิกายน 2563).
- Softmeli. 2561. **Firebase คืออะไร และมีข้อดีอย่างไรบ้าง.** [ระบบออนไลน์]. แหล่งที่มา :
<https://www.softmelt.com/article.php?id=588> (18 พฤศจิกายน 2563).
- techniccar. 2554. **ระบบเบรก .**[ระบบออนไลน์]. แหล่งที่มา :
<http://www.techniccar.com/braking-system/> (20 พฤศจิกายน 2563).
- Teerasej Jiraphatchandej. 2559. **Adobe XD อีกเครื่องมือ UX แสนสะดวก .**[ระบบออนไลน์]. แหล่งที่มา : <https://nextflow.in.th/2016/adobe-xd-experience-design-thai/#:~:text=Adobe%20XD%20หรือ%20Adobe%20Experience,เหล่า%20Designer%20สนุกสนานกันแน่นอน> (20 พฤศจิกายน 2563).
- TheJuPaNmm. 2554. **ประวัติความเป็นมาของรถยนต์.** [ระบบออนไลน์]. แหล่งที่มา :
<http://g19-53010914125.blogspot.com/2011/11/blog-post.html> (20 พฤศจิกายน 2563).

ประวัติผู้แต่ง



ชื่อ-สกุล

นาย Jarvis เดชา ชาญกวด

วัน เดือน ปีเกิด

9 สิงหาคม พ.ศ. 2541

ภูมิลำเนา

187 หมู่ 12 ตำบลนาเดื่อ อำเภอศรีสังคราม จังหวัดนครพนม 48150

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนบ้านนาเดื่อ จังหวัดนครพนม

มัธยมศึกษาตอนต้น

โรงเรียนนาเดื่อพิทยาคม จังหวัดนครพนม

ประกาศนียบัตรวิชาชีพ

วิทยาลัยเทคโนโลยีอุตสาหกรรมศรีสังคราม มหาวิทยาลัยนครพนม

จังหวัดนครพนม

ปริญญาตรี

ครุศาสตร์อุตสาหกรรมบัณฑิต (ครุศาสตร์วิศวกรรม)

สาขาวิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรมและเทคโนโลยี สาล.

ความสนใจพิเศษ

สื่อบรรเทง

คติพจน์

คนพากลถ้าจริงใจก็ไม่ผิด คบบัณฑิตไม่จริงใจก็เร็ผล

ประวัติผู้แต่ง



ชื่อ-สกุล

นางสาวธนกมล อินทรสุริยวงศ์

วัน เดือน ปีเกิด

9 กุมภาพันธ์ พ.ศ. 2542

ภูมิลำเนา

120/1 หมู่ 3 ตำบลรั้วใหญ่ อำเภอเมือง จังหวัดสุพรรณบุรี 72000

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนเทศบาล 1 วัดประตุสาร จังหวัดสุพรรณบุรี

มัธยมศึกษาตอนต้น

โรงเรียนสังวนหญิง จังหวัดสุพรรณบุรี

มัธยมศึกษาตอนปลาย

โรงเรียนสังวนหญิง จังหวัดสุพรรณบุรี

ปริญญาตรี

ครุศาสตร์อุตสาหกรรมบัณฑิต (ครุศาสตร์วิศวกรรม)

สาขาวิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรมและเทคโนโลยี สาล.

ความสนใจพิเศษ

gap year

คติพจน์

กระปืออยู่ที่ใจ แค่ไม่ฝึกไว้เทียมทาน

ประวัติผู้แต่ง



ชื่อ-สกุล

นายสิทธินนท์ ตติยะจินดา

วัน เดือน ปีเกิด

19 มิถุนายน พ.ศ. 2541

ภูมิลำเนา

77/78 ซอยนวมินทร์ 75 แยก 1 ถนนนวมินทร์ แขวงคลองกุ่ม

เขตบึงกุ่ม กรุงเทพมหานคร 10240

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนวัดคุณวิเศษ

จังหวัดตรัง

มัธยมศึกษาตอนต้น

โรงเรียนมัธยมบ้านบางกะปิ

จังหวัดกรุงเทพมหานคร

มัธยมศึกษาตอนปลาย

โรงเรียนมัธยมบ้านบางกะปิ

จังหวัดกรุงเทพมหานคร

ปริญญาตรี

ครุศาสตร์อุตสาหกรรมบัณฑิต (ครุศาสตร์วิศวกรรม)

สาขาวิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรมและเทคโนโลยี สาขาวิชา

ความสนใจพิเศษ

เกมส์ การ์ตูน

คติพจน์

สายในวันนี้ ไม่ต้องรอวันหน้า