

ระบบติดตามผู้ป่วยโควิด-19 ภายในโรงพยาบาลสนาม
COVID-19 PATIENT TRACKING SYSTEM IN FIELD HOSPITAL



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบติดตามผู้ป่วยโควิด-19 ภายในโรงพยาบาลสนาม
COVID-19 PATIENT TRACKING SYSTEM IN FIELD HOSPITAL



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2564

ภาควิชาวิศวกรรมโทรคมนาคม


คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบติดตามผู้ป่วยโควิด-19 ภายในโรงพยาบาลสนาม

COVID-19 PATIENT TRACKING SYSTEM IN FIELD HOSPITAL

ผู้จัดทำ

- | | | |
|------------------|------|----------|
| 1. นางสาวนิตา | นำพา | 61010473 |
| 2. นายธาดา | คิม | 61010503 |
| 3. นางสาวนภาพรรณ | ถมมา | 61010551 |


.....
(ผศ.ดร.พิชญ์ สุพรรณกุล)

อาจารย์ที่ปรึกษา


.....
(ผศ.มนต์ชัย แซ่มซอย)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การดำเนินโครงการเรื่อง การประยุกต์ใช้งานเทคโนโลยีปัญญาประดิษฐ์สำหรับการระบุตำแหน่งของผู้ป่วยในโรงพยาบาลสนาม จะไม่สามารถสำเร็จลุล่วงไปได้ด้วยดี หากไม่ได้รับความช่วยเหลือและความอนุเคราะห์อย่างดียิ่งจาก ผศ.ดร.พิชญ์ สุพรรณกุล อาจารย์ที่ปรึกษาโครงการ ผศ.มนต์ชัย แซ่มซ้อย อาจารย์ที่ปรึกษาโครงการร่วม ที่กรุณาให้คำแนะนำ คำปรึกษา และแนวทางการแก้ไขปัญหาที่เป็นประโยชน์ต่อการศึกษา ค้นคว้าวิจัยให้โครงการนี้สำเร็จสมบูรณ์ยิ่งขึ้น รวมถึงสนับสนุนสถานที่ เครื่องมือ และอุปกรณ์ต่าง ๆ ที่ใช้ระหว่างการจัดทำโครงการ

ขอขอบคุณคณาจารย์และเจ้าหน้าที่ประจำภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่าน ที่ได้อบรมสั่งสอน ประสิทธิ์ประสาทวิชา ความรู้ และประสบการณ์ให้แก่ผู้จัดทำขอขอบคุณเพื่อน ๆ พี่ ๆ น้อง ๆ ทุกคน ที่คอยให้ความช่วยเหลือและสนับสนุนโดยตลอด จนทำให้โครงการชิ้นนี้สำเร็จลุล่วงมาได้

และอันดับสุดท้ายขอกราบขอบพระคุณ บิดา มารดา และครอบครัว ที่ให้ความรัก ความห่วงใย และเป็นกำลังใจที่สำคัญเสมอมาและที่สำคัญคือสนับสนุนให้โอกาสทางด้านการศึกษานี้ อันมีค่ายิ่งแก่ผู้จัดทำ

นางสาวธนิศา	นำพา
นายธาดา	คิม
นางสาวนภาวรณ	ธมมา
	ผู้จัดทำ

ระบบติดตามผู้ป่วยโควิด-19 ภายในโรงพยาบาลสนาม
COVID-19 PATIENT TRACKING SYSTEM IN FIELD HOSPITAL

โดย	นางสาวธนิศา	นำพา	61010473
	นายธาดา	คิม	61010503
	นางสาวนภาพรรณ	ถมมา	61010551

อาจารย์ที่ปรึกษา ผศ.ดร.พิชญ์ สุพรรณกุล
อาจารย์ที่ปรึกษาร่วม ผศ.มนต์ชัย แซ่ม้อย

บทคัดย่อ

จากสถานการณ์การแพร่ระบาดของเชื้อไวรัสโคโรนา (COVID-19) ในปัจจุบันทำให้มีผู้ป่วย COVID-19 เพิ่มขึ้นอย่างรวดเร็วเป็นจำนวนมาก จนเกินขีดความสามารถของโรงพยาบาลที่จะรับดูแลรักษาผู้ป่วย ดังนั้นจำเป็นต้องจัดหาสถานที่ในการดูแลรักษาสำหรับรองรับผู้ป่วย เรียกว่า “โรงพยาบาลสนาม” ที่สามารถติดตามผู้ป่วยไม่ให้หลบหนีออกจากโรงพยาบาลสนามเพื่อป้องกันการแพร่เชื้อไปยังภายนอก วัตถุประสงค์ของโครงการนี้คือเพื่อช่วยเหลือบรรเทาภาระงานของบุคลากรทางการแพทย์ เพิ่มระยะห่างระหว่างบุคลากรทางการแพทย์กับผู้ป่วย และการติดตามผู้ป่วยไม่ให้ออกจากพื้นที่ ได้ใช้เทคโนโลยีปีคอนในการระบุตำแหน่งของผู้ป่วยว่ามีการออกจากโรงพยาบาลสนามหรือไม่ ระบบติดตามผู้ป่วย COVID-19 ภายในโรงพยาบาลสนามประกอบด้วย 3 ส่วนคือ ฐานข้อมูลสำหรับเก็บข้อมูลส่วนตัวของผู้ป่วย ส่วนเทคโนโลยีปีคอน และแอปพลิเคชันสำหรับผู้ป่วยและผู้ดูแลระบบ ติดตั้งปีคอนจำนวน 4 ตัวในพื้นที่โรงพยาบาลสนาม พัฒนาแอปพลิเคชันสำหรับผู้ป่วยเพื่อรับข้อมูลของผู้ป่วยและสัญญาณบลูทูธจากปีคอนเพื่อเก็บไว้ในฐานข้อมูล จากนั้นส่งข้อมูลมายังแอปพลิเคชันสำหรับผู้ดูแลระบบ เมื่อพบว่าผู้ป่วยออกจากพื้นที่จะมีการแสดงผลแจ้งเตือนไปยังผู้ดูแลระบบหรือพยาบาลที่ดูแลได้ทราบแบบเรียลไทม์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSTRACT

Due to the current situation of the coronavirus disease (COVID-19) epidemic, the number of COVID-19 patients is increasing rapidly that it exceeds the hospital's ability to take care of patients. Therefore, it is necessary to provide a place to care for patients, called "field hospital", where can track patients not to leave the field hospital to prevent the spread of infection to the outside. The objective of this project is to help alleviate the workload of healthcare professionals, increase the distance between medical personnel and patients and track patients not to leave the area. Beacon technology is used to determine the location of the patient whether they are left from the field hospital or not. The COVID-19 patient tracking system in the field hospital consists of 3 parts: a database for collecting personal data of patients; beacon technology part and application for patients and administrator. There are 4 beacons, which are installed in the field hospital. An application for patients is developed to receive patient data and Bluetooth signal from beacons to store in a database and then transmit the information to an application for administrator. When a patient is found leaving the area, a notification will be displayed to the administrator or nurse in real time.

สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	III
สารบัญรูป	VI
สารบัญตาราง	XI
บทที่ 1	
บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงการ	1
บทที่ 2	
ทฤษฎีและหลักการที่เกี่ยวข้อง	2
2.1 การแพร่ระบาดของโรคโควิด 19	2
2.2 ศึกษาเทคโนโลยีปีคอน	11
2.3 ศึกษาภาษาจาวา Java	13
2.4 ศึกษาการใช้โปรแกรม	14
2.5 ศึกษาการทำแจ้งเตือน (Notification in Android)	18
2.6 ศึกษา Database	25
2.7 ศึกษาการใช้ Firebase	26
2.8 ศึกษา Firebase Authentication	27
2.9 ศึกษา Realtime Database	28
2.10 การศึกษาทำ Run Background	30
2.11 Received Signal Strength (RSSI)	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า	
บทที่ 3	การออกแบบและการจัดทำปฏิญญานิพนธ์	40
	3.1 การออกแบบ	40
	3.2 เครื่องมือที่ใช้ในการทดลอง	67
	3.3 การจัดเก็บผลการทดลอง	69
บทที่ 4	ผลการทดลอง	78
	4.1 การทดสอบการค้นหาสัญญาณบีคอน	78
	4.2 การทดสอบการเชื่อมต่อของแอปพลิเคชันกับฐานข้อมูล	82
	4.3 การทดสอบการเชื่อมต่อของบีคอนกับแอปพลิเคชัน	87
	4.4 การทดสอบการเชื่อมต่อของ GPS กับแอปพลิเคชัน	89
บทที่ 5	สรุปผลและข้อเสนอแนะ	92
	5.1 สรุปผล	92
	5.2 ข้อเสนอแนะ	92
บรรณานุกรม		93
ภาคผนวก ก	การสร้างฐานข้อมูลและการเชื่อมต่อกับแอปพลิเคชัน	96
ภาคผนวก ข	ชุดคำสั่งการทำงานส่วนบีคอนและระบุตำแหน่ง	102
ภาคผนวก ค	วิธีการใช้งานแอปพลิเคชัน	117

สารบัญรูป

รูปที่	หน้า	
1.1	บล็อกไดอะแกรมของระบบ	2
2.1	จุดเสี่ยงในที่สาธารณะ	4
2.2	ความแตกต่างระหว่างโรงพยาบาลกับโรงพยาบาลสนาม	7
2.3	ระดับอาการของผู้ป่วยโควิด	10
2.4	บล็อกรุ่น RL-NRF51822	11
2.5	โลโก้ของภาษา Java	13
2.6	โปรแกรม Android Studio	14
2.7	หน้า Create New Project	15
2.8	รูปแบบ Application	15
2.9	ตั้งชื่อโครงการ	16
2.10	หน้าต่างของโปรแกรม Android Studio	16
2.11	หน้าต่างแสดงไฟล์ข้อมูล	17
2.12	การทำงานของ Notification บนแอนดรอยด์	18
2.13	การทำงานภายในการแจ้งเตือน	19
2.14	แถบการแจ้งเตือนต่าง ๆ	19
2.15	แถบแจ้งเตือนการเวอร์ชัน API	20
2.16	รูปแบบในการแสดงผลของ Notification	20
2.17	โปรแกรมการสร้าง Channel Notification	21
2.18	สร้าง Channel Group	22
2.19	ตัวอย่างหน้าจอแสดงการแจ้งเตือน	22
2.20	ชุดคำสั่ง	23
2.21	โปรแกรมการกำหนดหน้าต่างการแจ้งเตือน	23
2.22	ตัวอย่างหน้าจอแสดงการแจ้งเตือน	24
2.23	ชุดคำสั่ง	25
2.24	การเข้าสู่ระบบรูปแบบต่าง ๆ ของ FIREBASE AUTHENTICATION	27

สารบัญรูป (ต่อ)

รูปที่	หน้า
2.25 Authentication SDK	27
2.26 ตัวอย่างแอปพลิเคชัน	30
2.27 ศูนย์กลางในการทำงาน	31
2.28 ศูนย์กลางในการทำงาน	33
2.29 รูปแบบข้อมูล ibeacon	34
2.30 โครงสร้างแพ็คเกจ	35
3.1 ภาพรวมของการทำงานของแอปพลิเคชัน	40
3.2 บล็อกไดอะแกรมแสดงการทำงานโดยรวมในส่วนของผู้ใช้งาน	41
3.3 ภาพรวมการทำงานในส่วนผู้ดูแล (บุคลากรทางการแพทย์)	42
3.4 การจัดวางโครงสร้างหน้า splash screen	43
3.5 ลำดับการทำงานของหน้าล็อกอินและลงทะเบียน	44
3.6 จัดวางโครงสร้างหน้าล็อกอิน	44
3.7 เปิดใช้งาน sign in เป็น Email/password	45
3.8 ตรวจสอบข้อมูล e-mail ผู้ใช้งาน	45
3.9 จัดวางโครงสร้างหน้าลงทะเบียน	46
3.10 ข้อมูลผู้ป่วย (user) ในฐานข้อมูล	47
3.11 โครงสร้างหน้า user home	47
3.12 แผนผังการทำงานของหน้ากรอกอาการ	48
3.13 โครงสร้างของหน้ากรอกอุณหภูมิ	49
3.14 โครงสร้างของหน้ากรอกอาการ	49
3.15 ลำดับการทำงานของหน้าสวัสดิการ	50
3.16 จัดวางโครงสร้างหน้าสวัสดิการ	50
3.17 ลำดับการทำงานของหน้าติดต่อเจ้าหน้าที่	51
3.18 จัดวางโครงสร้างหน้าติดต่อเจ้าหน้าที่	51
3.19 ชุดคำสั่งหน้าติดต่อเจ้าหน้าที่	52

สารบัญญรูป (ต่อ)

รูปที่	หน้า
3.20 คำสั่งหน้า MANIFEST เพื่ออนุญาตให้เข้าถึงการโทรออก	52
3.21 ลำดับการทำงานหน้าสแกน	53
3.22 จัดวางโครงสร้างของหน้าสแกน	54
3.23 จัดวางโครงสร้างของหน้ายืนยันตำแหน่ง	54
3.24 ลำดับการทำงานหน้าแก้ไขโปรไฟล์	55
3.25 จัดวางโครงสร้างของหน้าโปรไฟล์	56
3.26 โครงสร้างหน้า admin home	57
3.27 ลำดับการทำงานของหน้าแสดงรายชื่อคนออกจากพื้นที่	58
3.28 โครงสร้างหน้าแสดงรายชื่อคนออกจากพื้นที่	59
3.29 ลำดับการทำงานของหน้าดูตำแหน่งของผู้ป่วย	60
3.30 จัดวางโครงสร้างหน้า Home ของหน้าดูตำแหน่งผู้ป่วย	60
3.31 จัดวางโครงสร้างหน้าดูตำแหน่งผู้ป่วยภายในห้อง (Indoor)	61
3.32 จัดวางโครงสร้างหน้าดูตำแหน่งผู้ป่วยภายนอกห้อง (Outdoor) ลำดับการทำงาน	61
3.33 ทำงานของหน้าแสดงข้อมูลผู้ป่วย	62
3.34 จัดวางโครงสร้างหน้าแสดงข้อมูลผู้ป่วย	63
3.35 หน้า manifest การขออนุญาตการเข้าถึง	64
3.36 กำหนดชื่อให้ mac address ของปีคอนแต่ละตัวและค่าที่ส่งเข้า firebase	64
3.37 ลำดับการทำงานของปีคอนในแอปพลิเคชัน	65
3.38 แผนผังบ้านและตำแหน่งการวัด	66
3.39 Ibeacon	67
3.40 Ibeacon eddystone	68
3.41 โทรศัพท์ OPPO RENO 2F	68
3.42 LMS ของ RSS ตามระยะทางจากปีคอนตัวที่ 1	71
3.43 LMS ของ RSS ตามระยะทางจากปีคอนตัวที่ 2	72
3.44 LMS ของ RSS ตามระยะทางจากปีคอนตัวที่ 3	72

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.45 LMS ของ RSS ตามระยะทางจากปีคอนตัวที่ 4	73
3.46 LMS ของ RSS ตามระยะทางปีคอน BLE ทุกตัว	74
3.47 แผนผังห้องที่แบ่งเป็นโซนต่าง ๆ	75
3.48 จุดติดปีคอนในห้อง 201 Room	75
3.49 จุดติดปีคอนในห้อง 202 Room	76
3.50 จุดติดปีคอนในห้อง 203 Room	76
3.51 จุดติดปีคอนในห้อง Exit	77
4.1 แอปพลิเคชัน BLE scanner	78
4.2 กราฟแสดง LRM ของ RSSI ของปีคอนตัวที่ 1	79
4.3 กราฟแสดง LRM ของ RSSI ของปีคอนตัวที่ 2	80
4.4 กราฟแสดง LRM ของ RSSI ของปีคอนตัวที่ 3	80
4.5 กราฟแสดง LRM ของ RSSI ของปีคอนตัวที่ 4	81
4.6 กราฟแสดง LRM ของ RSSI ของปีคอนทั้ง 4 ตัว	81
4.7 แสดงหน้าแอปพลิเคชันข้อมูลการลงทะเบียนของผู้ป่วย	83
4.8 ข้อมูลประวัติส่วนตัวของผู้ใช้งานในฐานะข้อมูล	83
4.9 แสดงหน้าแอปพลิเคชันสำหรับดูข้อมูลผู้ป่วย	84
4.10 แสดงหน้าแอปพลิเคชันสำหรับดูผู้ป่วยรายบุคคล	84
4.11 ข้อมูลที่เข้าฐานข้อมูล	85
4.12 ข้อมูลการกรอกอาการเข้าฐานข้อมูลและแสดงข้อมูลในแอปพลิเคชัน	85
4.13 ข้อมูลการกรอกประวัติการเข้าฐานข้อมูลและแสดงข้อมูลในแอปพลิเคชัน	86
4.14 ปุ่มนี้ติดต่อเจ้าหน้าที่	86
4.15 หน้าสแกนหาสัญญาณของปีคอน	87
4.16 หน้าเลือกดูตำแหน่ง Indoor และ Outdoor ผู้ป่วย	88
4.17 หน้าแสดงรายชื่อและตำแหน่งห้องของผู้ป่วย	88
4.18 หน้าแสดงแผนผังโรงพยาบาลสนาม	89

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.19 หน้าส่งข้อมูล location ของผู้ใช้งาน	90
4.20 แสดงค่า LAT , LON บนแถบแจ้งเตือนของมือถือ	90
4.21 หน้าต่างของแอดมิน ที่แสดง USER ID ที่ออกจากบริเวณพื้นที่	91
4.22 หน้า MAP แสดงตำแหน่งที่อยู่ของ USER ID ที่ออกจากพื้นที่	91



สารบัญตาราง

ตารางที่		หน้า
2.1	ข้อมูลรายละเอียดปีคอนรุ่น RL-NRF51822	12
2.2	ความแตกต่างระหว่าง BEACON กับ WI-FI	33
2.3	Path Loss Exponent	39



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันจากสถานการณ์การแพร่ระบาดของเชื้อไวรัสโคโรนา (Covid-19) ทำให้มีผู้ป่วย Covid-19 เป็นจำนวนมากและเพิ่มขึ้นอย่างรวดเร็ว จนเกินขีดความสามารถของโรงพยาบาลที่จะรับดูแลรักษาผู้ป่วยไว้ได้จึงต้องจัดหาสถานที่ในการดูแลรักษาสำหรับรองรับผู้ป่วย เรียกว่า “โรงพยาบาลสนาม”

“โรงพยาบาลสนาม” หมายถึง สถานที่ที่ให้การดูแลรักษาพยาบาล รองรับผู้ป่วยที่เกินขีดความสามารถของโรงพยาบาลในการรับดูแลรักษา ทั้งนี้การจัดตั้งจะตั้งนอกสถานพยาบาลขึ้นกับการดำเนินการของหน่วยงานที่รับผิดชอบในพื้นที่ เช่น วัด โรงเรียน โรงยิม หรือ หอประชุมขนาดใหญ่ เป็นต้น

เนื่องจากโรงพยาบาลสนามจำเป็นต้องดูแลผู้ป่วยเป็นจำนวนมาก โครงการนี้จึงมีแนวคิดเกี่ยวกับระบบที่สามารถช่วยเหลือบุคลากรทางการแพทย์ได้ เพื่อเป็นการลดภาระงาน และเว้นระยะห่างระหว่างบุคลากรทางการแพทย์กับผู้ป่วย คือ ระบบติดตามผู้ป่วยโควิด-19 ภายในโรงพยาบาลสนามขึ้นมา โดยสร้างและออกแบบแอปพลิเคชันสำหรับทั้งผู้ป่วยและบุคลากรทางการแพทย์ และใช้เทคโนโลยีบีคอนในการระบุตำแหน่งของผู้ป่วยว่ามีอาการออกจากโรงพยาบาลสนามหรือไม่ ซึ่งจะมีการแสดงผลแจ้งเตือนไปยังแอดมินหรือพยาบาลที่ดูแลได้ทราบแบบเรียลไทม์

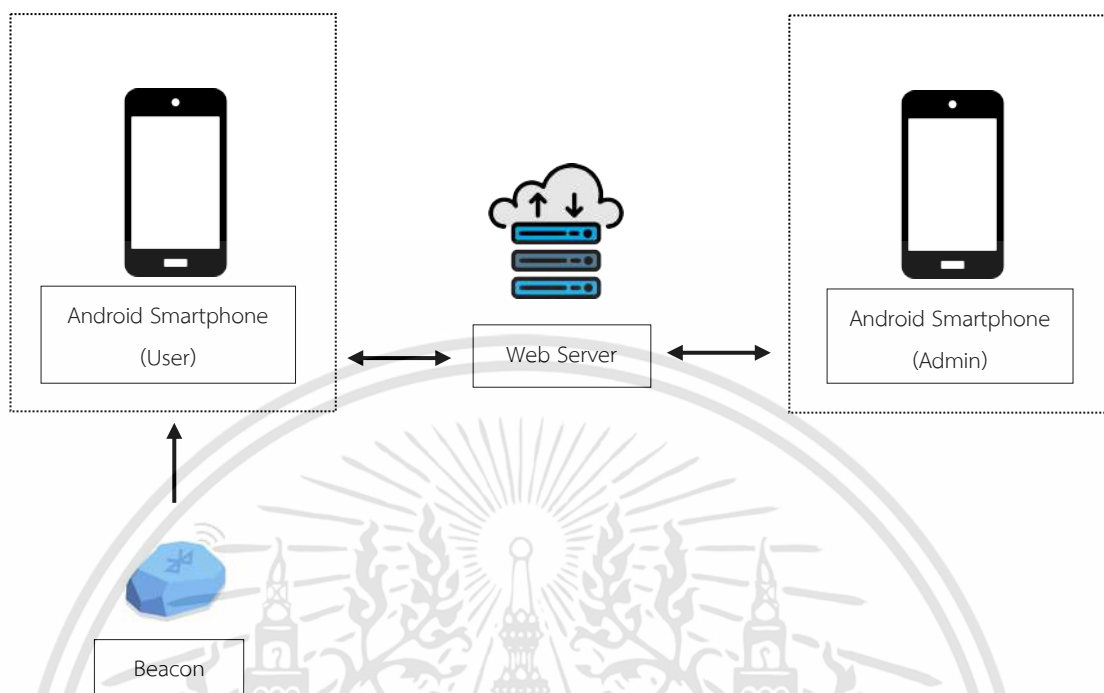
1.2 วัตถุประสงค์

- 1) เพื่อป้องกันผู้ป่วยหรือบุคคลที่กักตัวออกนอกพื้นที่ที่กำหนดไว้
- 2) เพื่อเว้นระยะห่างระหว่างผู้ป่วยกับบุคลากรทางการแพทย์
- 3) เพื่อศึกษาการทำงานแอปพลิเคชันที่สามารถเชื่อมต่อกับเทคโนโลยีบีคอนสำหรับโรงพยาบาลสนาม

1.3 ขอบเขตของปริญญานิพนธ์

- 1) ระบบสามารถตรวจจับตำแหน่งของผู้ใช้งาน (ผู้ป่วย) ว่าอยู่ในพื้นที่ที่กำหนดโดยจะใช้ตัวบีคอนในการส่งค่าสัญญาณและใช้ตัวแอปพลิเคชันรับค่าสัญญาณจากนั้นนำไปเข้าสู่ฐานข้อมูลแล้วประมวลผลเพื่อแสดงออกไป ดังรูปภาพที่ 1.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.1 บล็อกไดอะแกรมของระบบ

2) ระบบสามารถแจ้งเตือนไปยังผู้ดูแล (บุคลากรทางการแพทย์) หากมีบุคคลออกนอกพื้นที่ที่กำหนดโดยจะใช้การตรวจสอบสัญญาณบีคอนที่ได้รับแล้วนำมาวิเคราะห์ให้ตรวจจับว่าถ้ามีค่า distance ตามที่กำหนดไว้ และแจ้งเตือนไปยังทางผู้ดูแลได้รับทราบ

3) ระบบสามารถแสดงข้อมูลของผู้ใช้ให้แก่ผู้ดูแลได้ เช่น ผู้ใช้มีความต้องการสิ่งของหรือยาแก้ปวด ฯลฯ โดยจะให้ผู้ใช้เลือกว่าต้องการอะไรบ้างซึ่งทางผู้ดูแลจะได้รับข้อมูลเหล่านี้และนำสิ่งของไปแจกจ่ายให้กับผู้ใช้

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 การแพร่ระบาดของโรคโควิด 19

โรคโควิด-19 (Covid-19) เป็นโรคอุบัติใหม่ที่มีการแพร่ระบาดใหญ่เป็นวงกว้าง ปัจจุบันทั่วโลก มีผู้ป่วยโรคโควิด-19 แล้วกว่า 100 ล้านราย และเสียชีวิตกว่า 2 ล้านราย (ร้อยละ 2) สำหรับประเทศไทย แม้จะมีการป้องกันและควบคุมโรคโควิด-19 อย่างเข้มแข็งก็พบการติดเชื้อสูง สะสมถึงกว่า 2,107,674 ราย (ข้อมูลวันที่ 4 ธันวาคม 2564)

2.1.1 โรคโควิด-19 คือ

โรคโควิด-19 เกิดจากการติดเชื้อไวรัสโคโรนา 2019 (Coronavirus 2019) หรือชื่อทางวิทยาศาสตร์คือ SARS-CoV-2 เริ่มระบาดครั้งแรกที่มณฑลหูเป่ย์ สาธารณรัฐประชาชนจีน โครงสร้างทางพันธุกรรมคล้ายคลึงกับโรคซาร์สที่แพร่ระบาดใน พ.ศ. 2545 และโรคเมอร์ส (MERS) ที่แพร่ระบาดใน พ.ศ. 2555

2.1.2 การแพร่กระจายเชื้อ

2.1.2.1 ทางละอองฝอย ผ่านการไอจามรดกันในระยะ 1-2 เมตร

2.1.2.2 ทางสัมผัส ผ่านการสัมผัสสิ่งคัดหลั่งของผู้ป่วยติดเชื้อโควิด-19 เช่น น้ำมูก น้ำลาย ทั้งโดยตรง หรือทางอ้อมผ่านพื้นผิวสัมผัสที่ปนเปื้อนเชื้อ เช่น โต๊ะ ลูกบิด ประตู ราวจับ สำหรับสิ่งคัดหลั่งอื่น เช่น อุจจาระ พบว่ามีโอกาสแพร่เชื้อได้น้อยมาก

2.1.2.3 ทางอากาศ พบในบางกรณี เช่น ผู้ป่วยไอบวม ใกล้ชิดผู้ป่วย ในสถานที่ที่อากาศถ่ายเทไม่สะดวก การทำหัตถการที่ก่อให้เกิดการฟุ้งกระจายของละอองฝอยขนาดเล็ก ได้แก่ การดูดเสมหะ การใส่ท่อช่วยหายใจ เป็นต้น

2.1.3 อาการ

ผู้ป่วยเด็กติดเชื้อส่วนใหญ่กว่าร้อยละ 90 มีอาการไม่รุนแรง มักพบอาการเพียงเล็กน้อย เช่น ไข้ ไอ ปวดกล้ามเนื้อ ผู้ป่วยเด็กเพียงร้อยละ 5 เท่านั้นที่มีอาการรุนแรงหรือวิกฤติ เช่น ปอดอักเสบรุนแรง ระบบหายใจหรือระบบไหลเวียนโลหิตล้มเหลว รวมถึงภาวะอวัยวะหลายระบบในเด็ก

ผู้ป่วยเด็กสามารถพบติดเชื้อแต่ไม่มีอาการได้ประมาณร้อยละ 4 ภาวะแทรกซ้อนพบได้น้อย มักพบในผู้ป่วยกลุ่มเสี่ยงสูง เช่น เด็กเล็กอายุน้อยกว่า 1 ปีผู้ป่วยที่มีโรคประจำตัว เช่น โรคหัวใจและหลอดเลือด โรคไต โรคปอดเรื้อรัง หรือภาวะภูมิคุ้มกันบกพร่อง

2.1.4 การป้องกันติดเชื้อและแพร่เชื้อโควิด-19

2.1.4.1 สวมหน้ากากอนามัยหรือหน้ากากผ้า ยกเว้นในเด็กเล็กอายุน้อยกว่า 1 ปี

2.1.4.2 ล้างมือบ่อย ๆ ด้วยเจลแอลกอฮอล์ หรือหากเห็นว่ามือสกปรกก็ควรล้างด้วยน้ำกับสบู่เป็นระยะเวลา 20 วินาที

2.1.4.3 เวลาไอ จาม ควรใช้กระดาษทิชชูปิดปากและจมูก หลังจากนั้นนำไปทิ้งใน ถังขยะที่มีฝาปิด

2.1.4.3 ข้อศอกและต้นแขนด้านในแทน หลังจากนั้นล้างมือให้สะอาด

2.1.4.3 เว้นระยะห่างทางสังคม อย่างน้อย 1-2 เมตร

2.1.4.3 หลีกเลี่ยงการสัมผัสตา จมูก และปาก หากยังไม่ได้ล้างมือ

2.1.4.3 หลีกเลี่ยงการใช้สิ่งของร่วมกัน

2.1.4.3 หลีกเลี่ยงใกล้ชิดกับคนที่มีอาการป่วย

2.1.4.3 งดหอมแก้มหรือสัมผัสใกล้ชิดกับเด็กโดยไม่จำเป็น

2.1.4.3 แม้จะเคยติดเชื้อโควิด-19 มาแล้ว ก็ควรป้องกันการติดเชื้อและแพร่เชื้อโควิด-19 ด้วยเช่นกัน โดยจุดเสี่ยงต่าง ๆ จะแสดงดังรูปที่ 2.1



รูปที่ 2.1 จุดเสี่ยงในที่สาธารณะ [1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 การรักษาตัวในโรงพยาบาลสนาม

2.1.5.1 ขั้นตอนการเข้ารับการรักษาในโรงพยาบาลสนาม

- 1) คัดเลือกผู้ป่วยเพื่อเข้ารับการรักษาตามเกณฑ์ที่กำหนด
- 2.) เตรียมความพร้อมผู้ป่วย โดยชี้แจงแนวทางการดูแลและการปฏิบัติตัวเมื่อเข้าพัก
- 3) เพื่อเข้าพักรักษาตัวลงทะเลเบียน
- 4) รับคำแนะนำในการปฏิบัติตัวระหว่างเข้าพักและลงนามในใบยินยอมการเข้าพักโรงพยาบาลสนาม
- 5) เข้าพักตามพื้นที่ที่กำหนดไว้
- 6) ผู้ป่วยจะได้รับการประเมินภาวะสุขภาพจากทีมเจ้าหน้าที่เพื่อติดตามอาการทุกวัน
- 7) เจ้าหน้าที่จะรวบรวมข้อมูลสุขภาพของผู้ป่วยเพื่อพิจารณาอาการในแต่ละวัน
- 8) ผู้ป่วยที่อยู่ครบตามวันที่กำหนดเจ้าหน้าที่จะแจ้งและจำหน่ายผู้ป่วยหลังจากได้รับรายงานข้อมูลสุขภาพประจำวัน
- 9) เจ้าหน้าที่จะทำการลงทะเบียนจำหน่ายผู้ป่วยและผู้ป่วยลงนามรับทราบข้อแนะนำในการปฏิบัติตัวหลังจำหน่ายกลับบ้าน ขั้นตอนของแต่ละโรงพยาบาลสนามอาจมีการปรับเปลี่ยนตามความเหมาะสม

2.1.5.2 ความแตกต่าง “โรงพยาบาลสนาม” กับ “Hospital”

ข้อมูลจากกรมสนับสนุนบริการสุขภาพ ให้คำจำกัดความว่า “โรงพยาบาลสนาม” หมายถึง สถานที่ที่ให้การดูแลรักษาพยาบาลในกลุ่มผู้ป่วยไม่แสดงอาการหรือมีอาการน้อย กรณีการระบาดโรคติดเชื้อไวรัสโคโรนา 2019 หรือโควิด-19 ซึ่งเกินศักยภาพการจัดระบบบริการในโรงพยาบาล ทั้งนี้การจัดตั้งจะตั้งนอกสถานพยาบาล ขึ้นอยู่กับการดำเนินการของหน่วยงานที่รับผิดชอบในพื้นที่ เช่น วัด โรงเรียน มหาวิทยาลัย โรงแรม หรือหอประชุมขนาดใหญ่ เป็นต้น

โดยวัตถุประสงค์ของการจัดตั้งโรงพยาบาลสนามในกรณีการระบาดของโควิด-19 ในวงกว้าง คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. จัดตั้งระบบบริการการดูแลรักษาพยาบาล ผู้ป่วยที่เข้าข่าย PUI (Patient Under Investigation: ผู้ป่วยที่ต้องดำเนินการสอบสวนโรค) ในกรณีที่เกิดขีดความสามารถของโรงพยาบาล ตั้งแต่การคัดกรองผู้ป่วย การให้การวินิจฉัยดูแลรักษาแบบ One Stop Services

2. สามารถรับผู้ป่วยยืนยันโควิด-19 ไว้ดูแลรักษาแบบผู้ป่วยใน และให้การดูแลผู้ป่วยที่อาการไม่รุนแรง

3. ใช้เป็นสถานที่เพื่อดูแลรักษาผู้ป่วยที่มีอาการคงที่ และได้รับการส่งตัวมาจากโรงพยาบาลซึ่งดูแลผู้ป่วยในภาวะเฉียบพลัน หรือวิกฤต จนดีขึ้นแล้ว

สำหรับการจัดตั้งโรงพยาบาลสนามแต่ละแห่ง เป็นไปตามมาตรฐานที่กำหนด เพื่อให้มีความปลอดภัยเป็นอันดับแรก ภายใต้การทำงานร่วมกันของกระทรวงสาธารณสุข ได้แก่ กรมสนับสนุนบริการสุขภาพ ผู้เชี่ยวชาญทั้งด้านระบาดวิทยา ด้านการแพทย์ กับฝ่ายปกครอง ท้องถิ่นต่าง ๆ และมีนายแพทย์สาธารณสุขจังหวัดเป็นผู้อนุญาตให้จัดตั้งได้ ด้วยความเห็นชอบของคณะกรรมการโรคติดต่อจังหวัด ซึ่งมีการจัดโซนนิ่งให้ห่างจากชุมชน อาคารมีระบบจัดการน้ำเสีย การจัดการอากาศ การไหลเวียนของอากาศ มีบุคลากรทางการแพทย์ทำงานประจำและต่อเนื่อง เป็นต้น

ส่วน Hospitel หรือสถานพยาบาลผู้ป่วยเฉพาะกิจ มีที่มาจากการรวมคำว่า Hospital ที่แปลว่า โรงพยาบาล และ Hotel ที่แปลว่า โรงแรม เข้าด้วยกัน จากการปรับเปลี่ยนโรงแรมให้เป็นสถานพยาบาลชั่วคราว โดย Hospitel เหมาะสำหรับผู้ป่วยโควิด-19 ที่ต้องการความสะดวกสบาย เป็นส่วนตัวมากกว่าโรงพยาบาลสนาม รองรับผู้ป่วยที่มีอาการน้อยหรือไม่รุนแรง ปรับเปลี่ยนโรงแรมให้เป็นพื้นที่เฝ้าระวังอาการ เพื่อให้โรงพยาบาลสามารถรองรับผู้ป่วยที่มีอาการหนักได้อย่างเต็มที่และมีประสิทธิภาพสูงสุด

ตามหลักเกณฑ์ การปรับใช้โรงแรม เป็น “Hospitel” จะต้องเป็นโรงแรมที่มีห้องพักมากกว่า 30 ห้องขึ้นไป, เหมาะสำหรับผู้ป่วยที่ได้รับการรักษาในโรงพยาบาลมาแล้วไม่น้อยกว่า 5-7 วัน และมีผลภาพถ่ายรังสีปอดคงที่, ผู้ป่วยยินดีให้ความร่วมมือ สามารถสื่อสารได้รู้เรื่อง ดูแลตนเองได้ดี ไม่ก้าวร้าว และไม่มีความเสี่ยงทางจิตเวช, ผู้ป่วยไม่มีไข้ กรณีมีโรคประจำตัว ต้องควบคุมอาการได้ดี, ต้องจัดยาให้พร้อม สำหรับให้ผู้ป่วยรับประทานด้วยตัวเองจนครบตามแผนการรักษาของแพทย์ และโรงพยาบาลต้นทางต้องยินดียินดีรับผู้ป่วยกลับเข้ารับรักษาที่โรงพยาบาล หากผู้ป่วยมีอาการเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยมาตรฐานของ Hospitel คือ ทีมแพทย์จะดำเนินการตรวจและบันทึกอาการผู้ป่วยทุกวันผ่านเทเลเมดิซีน (Telemedicine) หรือไลน์ (Line) กลุ่ม, ทุก Hospitel จะต้องมีแพทย์ 1 คนประจำ มีพยาบาลอัตราส่วน 20 ต่อ 1 พยาบาล มีการตรวจคนไข้ผ่านเทเลเมดิซีน และแอปพลิเคชันไลน์ มีการเยี่ยมผู้ป่วยทุกวัน พร้อมทั้งมีเครื่องมือพื้นฐานต่าง ๆ เช่น เครื่องวัดอุณหภูมิ เครื่องวัดระดับความเข้มข้นออกซิเจนในเลือด และเครื่องเอกซเรย์ เป็นต้น ที่สำคัญ คือ ผู้ป่วยทุกราย ต้องผ่านการตรวจคัดกรองจากโรงพยาบาลหลักก่อน ข้อกำหนดหลักๆ เช่น อายุไม่เกิน 50 ปี ไม่มีโรคประจำตัวอื่น ไม่ใช่เด็ก ไม่ตั้งครรภ์ ไม่มีเอกซเรย์ที่เปลี่ยนแปลง จึงจะสามารถย้ายเข้ามาได้ และในกรณีที่ผู้ป่วยมีอาการมากขึ้นก็จะย้ายไปยังโรงพยาบาลหลักได้ทันที แสดงดังรูปที่ 2.2



รูปที่ 2.2 ความแตกต่างระหว่างโรงพยาบาลกับโรงพยาบาลสนาม [2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6 การรักษาตัวแบบ Home Isolation

คือแนวทางปฏิบัติเมื่ออยู่ในสภาวะการแพร่เชื้อรุนแรงจนเตียงไม่เพียงพอต่อการรักษา ทำให้ผู้ป่วยโควิด-19 ที่ไม่ค่อยมีอาการสามารถพักรักษาตัวที่บ้าน หรือสถานที่เฉพาะที่ ถูกจัดไว้ให้แทนการรักษาตัวที่โรงพยาบาลโดยจะได้รับความช่วยเหลือและการติดตามอาการจากแพทย์อยู่เสมอ โดยปกติแล้วผู้ที่ทำ Home Isolation ได้จะต้องมีเกณฑ์ ดังนี้

ผู้ป่วยโควิด-19 ที่รักษาตัวแล้ว : สำหรับผู้ที่รักษาตัวประมาณ 7-10 วันในโรงพยาบาล และแพทย์วินิจฉัยว่าสามารถรักษาตัวต่อที่บ้านได้

ผู้ป่วยโควิด-19 ที่กำลังรอเตียง : สามารถทำ Home Isolation ได้หากผ่านการวินิจฉัยจากแพทย์ตามความเหมาะสม หรือผู้ป่วยสีเขียว คือมีอาการไข้ ไอ เจ็บคอ มีน้ำมูก ไม่ได้กลิ่น ไม่รับรส มีผื่น และถ่ายเหลว แต่ไม่มีอาการหายใจเร็ว หายใจลำบาก ปอดไม่อักเสบ และระดับออกซิเจนไม่ต่ำกว่า 96%

2.1.6.1 คุณสมบัติของผู้ที่สามารถ Home Isolation ได้

- 1) อายุไม่เกิน 60 ปี
- 2) สุขภาพแข็งแรง มีอาการน้อยหรือไม่มีอาการ
- 3) ไม่มีภาวะน้ำหนักเกินมาตรฐาน
- 4) อาศัยอยู่คนเดียว หรือมีผู้อาศัยร่วมด้วยไม่เกิน 1 คน
- 5) ต้องไม่มีโรคประจำตัว ได้แก่ โรคไต โรคหัวใจ-หลอดเลือด

โรคเบาหวาน โรคหลอดเลือดสมอง และโรคปอด

2.1.6.2 เป็นผู้ป่วย Home Isolation ต้องปฏิบัติตัวอย่างไร

- 1) กักตัวอยู่ในพื้นที่ห้ามออกจากที่พัก และห้ามให้ผู้อื่นเข้าเยี่ยม
- 2) หากมีคนอยู่ในบ้านต้องรักษาระยะห่างไม่ต่ำกว่า 2 เมตร และ

สวมหน้ากากอนามัยตลอดเวลา

3) ห้ามใช้สิ่งของร่วมกับผู้อื่น และแยกห้องน้ำใช้หากสามารถทำได้ ไม่ทานอาหารร่วมกับผู้อื่น แยกซักเสื้อผ้ากับผู้อื่น

- 4) ล้างมือด้วยสบู่หรือเจลแอลกอฮอล์เป็นประจำ และแยกขยะ

มัดปากถุงให้แน่น

2.1.6.3 ทำ Home Isolation การติดต่อสังเกตอาการควรทำอย่างไร

กรอกข้อมูลเพื่อลงทะเบียนที่ <https://crmsup.nhso.go.th/> หรือแอดไลน์ @nhso และกดลงทะเบียนในระบบดูแลที่บ้าน Home Isolation และรอเจ้าหน้าที่ ยืนยันข้อมูล เมื่อได้รับการยืนยันแล้วจะได้รับข้อมูล ได้แก่ Video Call ประเมินอาการ 2 ครั้งต่อวัน / อาหาร 3 มื้อ / ยาพื้นฐานและยาฟ้าทะลายโจร / เครื่องวัดออกซิเจนปลายนิ้ว / ปรอทวัตช์แบบ ดิจิตอล / การประสานงานรับการรักษากรณีอาการหนักขึ้น

2.1.7 การรักษาตามอาการผู้ป่วย

2.1.7.2 อาการของผู้ป่วยสีเขียว

ผู้ป่วยสีเขียวคือกลุ่มที่มีอาการเบาคล้ายไข้หวัด หรือไม่มีอาการเลย มีไข้อุณหภูมิ 37.5 องศาเซลเซียสขึ้นไป ไม่รับรส ไม่รับกลิ่น ไอ มีน้ำมูก เจ็บคอ ตาแดง มีผื่น และถ่ายเหลว

1) การรักษาผู้ป่วยสีเขียว

เนื่องจากอาการที่ไม่รุนแรงเมื่อเทียบกับผู้ป่วยสีเหลือง และสีแดงจึงไม่จำเป็นต้องทำการรักษาที่โรงพยาบาล

Home Isolation : การรักษาด้วยการกักตัวอยู่ที่บ้านโดยจะได้รับการดูแลพร้อมอุปกรณ์ขั้นพื้นฐาน

Hospital : การเข้ารับรักษาตัวที่ Hospital ต้องใช้ใบยืนยันผลการตรวจแบบ RT-PCR (เข้ารับการตรวจได้หากมีผลการตรวจ ATK เป็นบวก) โดยจะได้รับการดูแลจากแพทย์ตลอด 24 ชั่วโมง

2.1.7.2 อาการของผู้ป่วยสีเหลือง

ผู้ป่วยสีเหลืองคือกลุ่มที่มีอาการเสียงรุนแรง หรือมีโรครวม เวียนหัว อ่อนเพลีย ไอแล้วมีอาการเหนื่อย แน่นหน้าอก หายใจลำบาก ขับถ่ายเหลว 3 ครั้งต่อวัน ขึ้นไป อ่อนเพลีย ปอดอักเสบ

1) การรักษาผู้ป่วยสีเหลือง

เนื่องจากอาจเกิดอาการแทรกซ้อนจากโรคประจำตัว หรือ 7 โรคกลุ่มเสี่ยง และมีอาการที่รุนแรงกว่าผู้ป่วยสีเขียวจึงไม่เหมาะที่จะทำการกักตัวรักษาที่บ้านแบบ Home Isolation ได้ หรือไม่สามารถรักษาที่ Hospital ได้ ควรเข้ารับการรักษาตัวที่โรงพยาบาล

2.1.7.3 อาการของผู้ป่วยสีแดง

ผู้ป่วยสีแดงคือกลุ่มที่มีอาการรุนแรงต้องรีบเข้ารับการรักษาตัวโดยเร็ว ระบบหายใจมีปัญหารุนแรงทำให้หายใจลำบาก หอบเหนื่อย หากเอกซเรย์จะพบปอดอักเสบรุนแรง เกิดภาวะปอดบวมจากการเปลี่ยนแปลงความอึดตัวของเลือดน้อยกว่า 96 % หรือลดลง 3 % จากค่าที่วัดได้ในตอนแรก แน่นหน้าอกตลอดเวลา และหายใจเจ็บหน้าอก ตอบสนองช้าหรือไม่รู้สึกรู้ตัว

1) การรักษาผู้ป่วยสีแดง

อาการของผู้ป่วยสีแดงที่รุนแรงต้องอยู่ภายใต้การดูแลของแพทย์ในโรงพยาบาลอย่างใกล้ชิดที่ห้อง ICU โดยผู้ป่วยจะได้รับเครื่องช่วยหายใจ แนะนำระหว่างรอเตียงควรรนอนคว่ำ หากหายใจไม่ออกให้นอนตะแคงประมาณ 45 องศา หากเป็นสตรีตั้งครรภ์ให้นอนตะแคงซ้ายเพื่อให้เลือดไหลเวียนได้ดี พยายามขยับขาป้องกันลิ่มเลือด และดื่มน้ำในปริมาณมาก ทานอาหารที่มีประโยชน์ให้เพียงพอต่อความต้องการของร่างกาย

2.1.8 ผู้ป่วยโควิด-19 ที่สามารถกลับบ้านได้

2.1.8.1 อุณหภูมิวัดได้ไม่เกิน 37.8 องศาเซลเซียสต่อเนื่อง 1-2 วัน

2.1.8.2 มีอัตราการหายใจในระดับปกติไม่เกิน 20 ครั้งต่อนาที

2.1.8.3 ภาพรังสีปอดแสดงให้เห็นว่าปอดไม่ได้มีอาการแย่งลง

2.1.8.4 ความอึดตัวของออกซิเจนต้องมากกว่า 96 % เมื่อไม่ได้ออกแรง

หากพบว่าตนเองมีอาการที่เสี่ยงควรรักษาตัวตามความรุนแรงของอาการ เมื่อได้ออกจากโรงพยาบาลแล้วยังต้องให้ความสำคัญกับการดูแลตนเองและกักตัวตามคำแนะนำของแพทย์



รูปที่ 2.3 ระดับอาการของผู้ป่วยโควิด [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ศึกษาเทคโนโลยีบีคอน

บีคอน (Beacon) คือ อุปกรณ์ขนาดเล็กที่ถูกติดตั้งในสมาร์ทโฟน (Smart Phone) รุ่นใหม่ ๆ หรือสมาร์ตดีไวซ์ (Smart Device) รุ่นใหม่ๆ และอีกแบบ คือตัวบีคอนเฉพาะ ที่ไม่ได้อยู่ในโทรศัพท์ จะเป็นอุปกรณ์ IoT หรือ Internet of Things ที่ทำงานต่างหาก การทำงานของบีคอนคือรับปล่อยสัญญาณผ่าน Bluetooth พลังงานต่ำ มีระยะส่งสัญญาณสูงไกลสุด 200 เมตร ความสามารถเด่นของบีคอนคือ การให้ข้อมูลตำแหน่งระยะใกล้ที่แม่นยำ

2.2.1 หลักการทำงานของบีคอน

การทำงานของบีคอน หลัก ๆ จะมีหน้าที่ส่งสัญญาณออกมาอย่างเดี่ยวส่วนตัว แอปพลิเคชันมือถือนั้น เมื่อได้รับสัญญาณจากบีคอน จะนำข้อมูลที่ถูกส่งออกมาประมวลผล เพื่อตีความหาระยะห่างระหว่างมือถือกับตัวบีคอน ในระยะ 0 – 70 เมตร เพื่อให้ง่ายในการพัฒนา แอปพลิเคชันตัว APIs ของ Estimote สามารถนำระยะห่างที่หน่วยเป็นเมตรนั้นมา แบ่งเป็น 3 ระยะที่เข้าใจง่าย คือ ใกล้มาก (Immediate: 0-20 cm) ใกล้ (Near: 20 cm – 2 m) ไกล (Far: 2 – 70 m) ขึ้นอยู่กับนักพัฒนาจะนำระยะทั้ง 3 ไปกำหนดให้แอปแสดงข้อมูลอะไรแก่ผู้ใช้ เช่น ถ้าผู้ใช้อยู่ไกล ให้แจ้งเตือนว่าเข้ามาในรัศมีแล้ว หรือถ้าผู้ใช้ใกล้ก็ให้แสดงข้อมูลของวัตถุนั้น เป็นต้น

2.2.1.1 ชนิดบีคอนที่ใช้ในการจัดทำโครงการ

บีคอนที่ใช้ในการจัดทำโครงการเป็นรุ่น RL-NRF51822 ดังแสดงในรูปที่ 2.4 และมีข้อมูลรายละเอียด แสดงไว้ในตารางที่ 2.1



รูปที่ 2.4 บีคอน รุ่น RL-NRF51822 [4]

ลักษณะของบีคอนรุ่น NRF51822 บีคอนโมดูลแบบพกพา iBeacon: ULTRA LOW Power ชุดชิป NRF51822 และ Leverage BLE 4.0 ตัวสีขาวเป็นแบบพกพาได้, เป็นฮาร์ดแวร์และเฟิร์มแวร์ที่ถูกต้องตามข้อกำหนดไว้ใช้สำหรับเชิงพาณิชย์และใช้สำหรับในอาคาร โมดูล Location-

Based Service NRF51822 Beacon, สัญญาณคลื่นวิทยุ 2.4 GHz และสามารถตรวจจับสัญญาณ โดยการติดตั้งแอปพลิเคชันสำหรับโทรศัพท์มือถือ interpreted by IOS และ Android BLE

2.2.1.2 ข้อดี

- 1) อย่างรวดเร็ว TO ปรับแต่งปลอกมากตำราคา
- 2) UUID, Major, Minor อื่น ๆ พารามิเตอร์เปลี่ยนโดยลูกค้า
- 3) iBeacon บาร์รหัสและ Easy สำหรับรับรู้
- 4) ใช้ Nordic's NRF51822 ชิพ Bluetooth LOW Energy
- 5) สนับสนุน IOS7.0, Android 4.3 ขึ้นไป
- 6) Effected ช่วง: 50M
- 7) อายุการใช้งานแบตเตอรี่มากกว่า 1 ปี

ตารางที่ 2.1 ข้อมูลรายละเอียดปีคอนรุ่น RL-NRF51822 [4]

RL-NRF51822	
แรงดันไฟฟ้า	2.0-3.6V (DC)
ความถี่ในการทำงาน	2402 - 2483.5MHz
แฟลช (KB)	256/128
Output Power	-30dBm ~ 4dBm
รับความไว	-97dBm (HIGH GAIN โหมด)
TX Current	10.5mA
RX Current	9.7mA
สแตนด์บายปัจจุบัน	2.6uA
เสาอากาศรูปแบบ	เสาอากาศ PCB
ช่วง	1S
Power	4dBm
ส่งระยะทาง	> 50M (0dBm)
ขนาด	25 มม.* 4.6 มม
SENSOR	ตัวเลือก (SHT3X/KX022)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ศึกษาภาษาจาวา Java

เป็นภาษาเขียนโปรแกรมเพื่อวัตถุประสงค์ทั่วไป โดยสามารถทำงานได้พร้อมกัน เป็นภาษาที่สร้างมาจากคลาส และสนับสนุนการเขียนโปรแกรมแบบออบเจ็ค (Object) ถูกออกแบบมาให้พร้อมสำหรับการใช้งาน โดยมีเมธอดและคลาสช่วยอำนวยความสะดวกมากมาย ภาษา Java นั้นมีความตั้งใจที่จะทำให้ให้นักพัฒนา ออกแบบและพัฒนาโปรแกรมน้อยลง นั่นคือการเขียนเพียงครั้งเดียว แต่นำไปใช้งานได้ทุกที่หรือทุกแพลตฟอร์ม แอปพลิเคชันของภาษา Java นั้นโดยปกติแล้วจะคอมไพล์เป็น bytecode ที่สามารถรันได้ใน Java virtual machine (JVM) ขึ้นอยู่กับสถาปัตยกรรมของคอมพิวเตอร์นั้น ๆ และใน ปี 2016 Java เป็นภาษาที่ได้รับ ความนิยมและใช้มากที่สุดในโลก โดยเฉพาะการใช้พัฒนาเว็บแอปพลิเคชัน ภาษา Java นั้นพัฒนาโดย James Gosling ที่บริษัท Sun Microsystems (ปัจจุบันถูกซื้อกิจการเป็น Oracle Corporation) และเปิดตัวครั้งแรก เมื่อปี 1995 โดยภาษานั้นได้รับรูปแบบจากภาษา C และ C++ แต่ภาษา Java ถือว่าเป็นภาษาระดับสูงกว่า ภาษาทั้งสอง ต้นกำเนิดการอ้างอิงในการพัฒนาของ Java compiler virtual machines และคลาสไลบรารี (Class Library) ในตอนต้นนั้นถูกเผยแพร่โดย Sun ภายใต้ลิขสิทธิ์ที่เหมาะสม ใน May 2007 ในการปฏิบัติตามกับข้อระบุของ Java Community Process Sun ได้จดทะเบียนใหม่ในเทคโนโลยีของ Java เป็นจำนวนมากภายใต้ GNU General Public License และการพัฒนาเทคโนโลยีอื่น ๆ เช่น GNU Compiler for Java (bytecode compiler) GNU Classpath (ไลบรารีมาตรฐาน) และ IcedTea-Web เวอร์ชันล่าสุดของ Java คือเวอร์ชัน 8 ที่สนับสนุนโดย Oracle ถึงแม้ในเวอร์ชันก่อนหน้าได้สนับสนุนโดย Oracle และบริษัทอื่นโดยโลโก้ของภาษา Java จะเป็นดังแสดงในรูปที่ 2.5



รูปที่ 2.5 โลโก้ของภาษา Java [5]

หนึ่งของวัตถุประสงค์ในการออกแบบภาษา Java คือให้มันสามารถพกพาได้ ซึ่งหมายความว่าโปรแกรมที่เขียนในภาษา Java จะต้องสามารถรันได้กับฮาร์ดแวร์และระบบปฏิบัติการที่ได้รับการสนับสนุนจาก Java Runtime ซึ่งนี่บรรลุผลได้โดยการคอมไพล์โค้ดของ Java ให้อยู่ในรูปแบบการแสดงผลระดับกลางที่เรียกว่า Java bytecode แทนการแปลงไปยังภาษาเครื่องโดยตรง คำสั่งของ Java bytecode นั้นคล้ายคลึงกับ ภาษาเครื่อง แต่มันจะทำงานโดย virtual machine (VM) ที่เขียนเฉพาะสำหรับฮาร์ดแวร์แต่ละประเภท ซึ่ง โดยปกติผู้ใช้ทั่วไปจะใช้ Java Runtime Environment (JRE) ที่ติดตั้งในเครื่องของพวกเขาสำหรับรัน Standalone Java applications หรือในเว็บเบราว์เซอร์ (Web Browser) สำหรับ Java applets นอกจากนี้ ภาษา Java ยังมีไลบรารีมาตรฐานในการพัฒนาแอปพลิเคชันในเรื่อง Graphics threading และ networking หลังจากที่คุณเรียนจบบทเรียนนี้แล้ว คุณจะเข้าใจพื้นฐานการเขียนโปรแกรมภาษา Java ทั้งหมด ตั้งแต่พื้นฐานไปจนถึงขั้นสูง ในเรื่องของการเขียนโปรแกรมเชิงวัตถุ ในบทเรียนของแต่ละตอนจะมีตัวอย่างของ โปรแกรมสำหรับตอนนั้น ๆ เพื่อช่วยให้คุณเข้าใจมากขึ้น และคุณสามารถนำความรู้เหล่านี้ไปต่อยอดในขั้นสูง ต่อไปในการเรียนรู้ภาษา Java เช่น Java GUI Java Graphics การพัฒนาเว็บไซต์ Java applet หรือ JSP หรือเน็ตเวิร์ค

2.4 ศึกษาการใช้โปรแกรม

Android studio 3.2.1 การใช้งานโปรแกรมเบื้องต้น โปรแกรมที่ใช้ในการจัดทำโครงการนี้คือโปรแกรม Android Studio ที่ใช้ภาษา Java ในการเขียนดังแสดงในรูปที่ 2.6

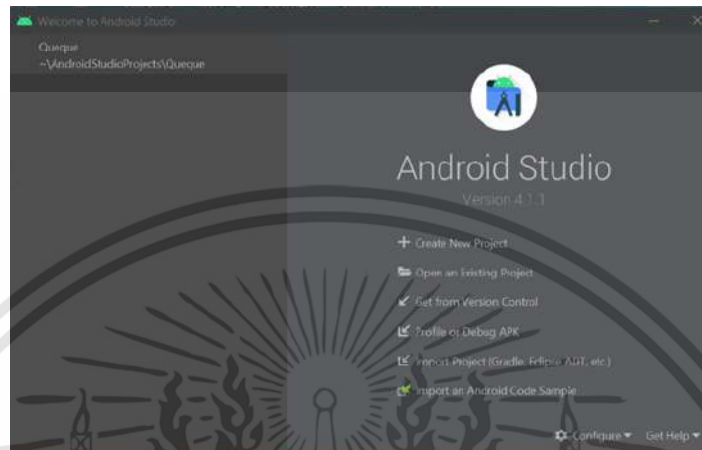


รูปที่ 2.6 โปรแกรม Android Studio [7]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

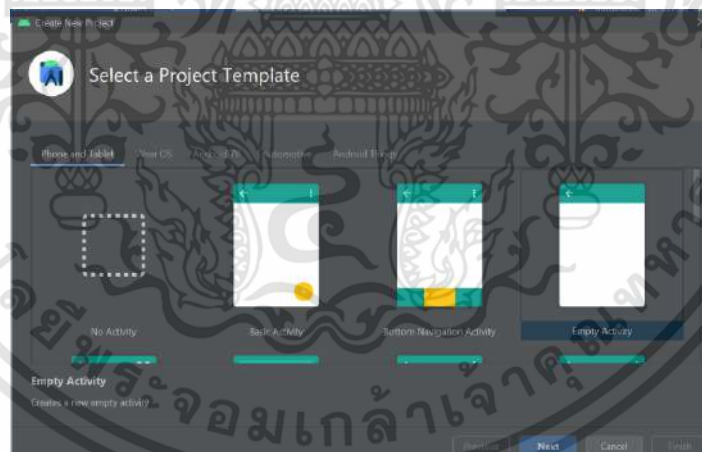
2.4.1. ทำการติดตั้งโปรแกรม Android Studio

2.4.2. เมื่อเราเปิดโปรแกรมจะขึ้นดังรูปที่ 2.7 ให้เลือก Create New Project



รูปที่ 2.7 หน้า Create New Project [8]

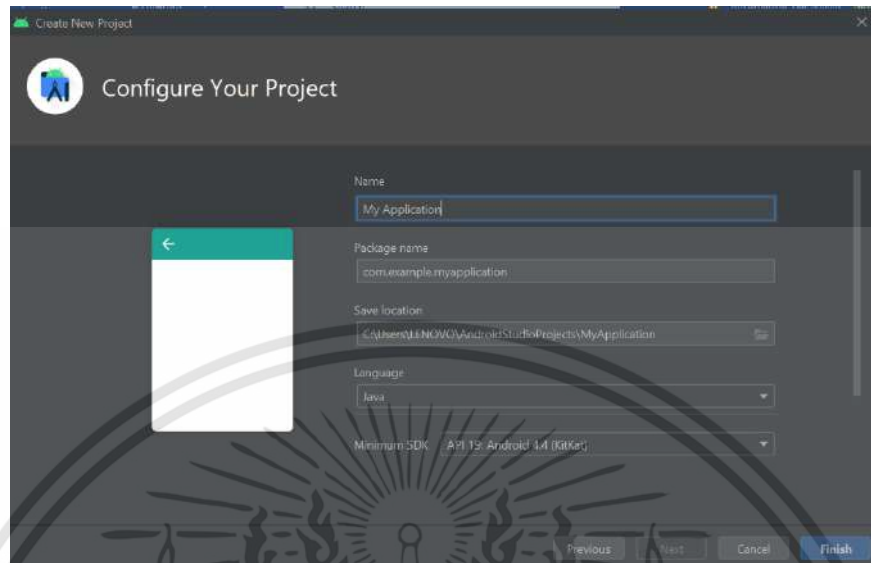
2.4.3. ให้เราเลือกรูปแบบของแอปพลิเคชันตามที่เราต้องการ ดังรูปที่ 2.8



รูปที่ 2.8 รูปแบบ Application [8]

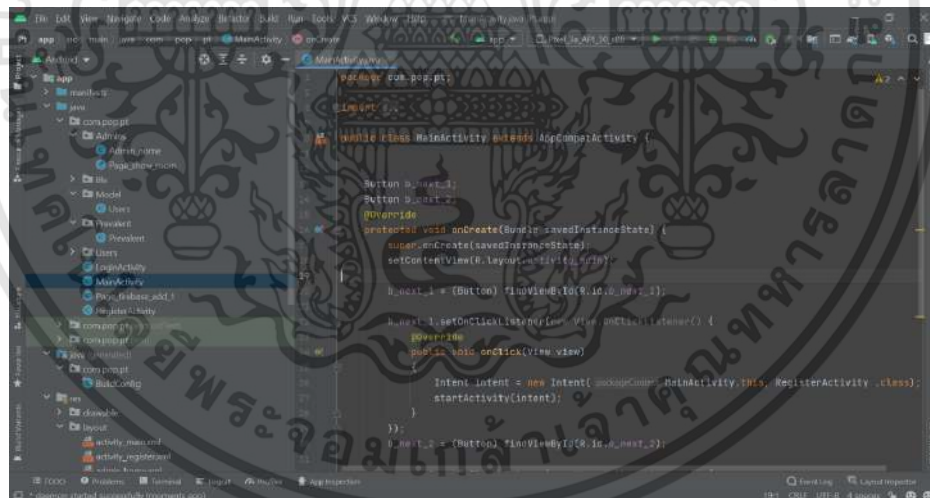
2.4.4. การตั้งชื่อให้โครงงานและกรอกรายละเอียดตามต้องการโดยตั้งค่าดังรูป 2.9 โดย Language เป็นการตั้งค่าภาษาที่ใช้ในการเขียน และ Minimum SDK เป็นค่าที่บอกว่า Android ชนิดไหนสามารถใช้งานได้บ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 ตั้งชื่อโครงการ [8]

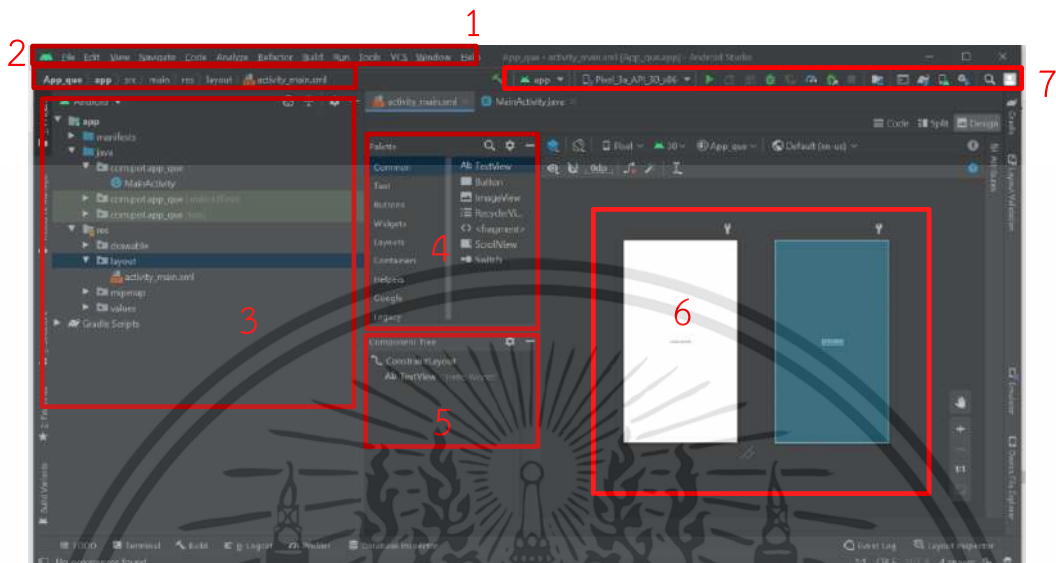
2.4.5. เมื่อทำการตั้งค่าเรียบร้อยแล้วดังรูปที่ 2.10



รูปที่ 2.10 หน้าต่างของโปรแกรม Android Studio [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.6. หน้าต่างแสดงไฟล์ข้อมูลที่อยู่ในโครงการ ดังรูปที่ 2.11



รูปที่ 2.11 หน้าต่างแสดงไฟล์ข้อมูล [9]

แถบเมนูต่าง ๆ

- 2.4.1.1 แถบเครื่องมือต่าง ๆ (Toolbar) และคำสั่งในโปรแกรม
- 2.4.1.2 แถบการนำทาง (Navigation bar) เป็นแถบแสดงทิศทางที่ใช้ในการทำทางไปยัง Project ทำให้ง่ายเพื่อการแก้ไข
- 2.4.1.3 หน้าต่างแสดงไฟล์ข้อมูลที่อยู่ในโปรเจค
- 2.4.1.4 แถบวิดเจ็ต (Widget bar) เป็นส่วนที่เก็บ widget ประเภทต่าง ๆ ที่ใช้ในการออกแบบส่วนประกอบต่าง ๆ ของหน้าจอที่แสดงผล
- 2.4.1.5 โครงสร้างแบบลำดับขั้น (Component tree) แสดงผลโครงสร้างปัจจุบันที่ทำงานอยู่
- 2.4.1.6 หน้าต่างการแก้ไข (Editor window) เป็นส่วนแสดงผลจำลองที่เราได้ออกแบบแอปพลิเคชันไว้จะแสดงผลมายังหน้านี้ โดยจะแสดงอยู่ 2 โหมด คือ Design และ Text
- 2.4.1.7 แถบควบคุมการทดสอบ เป็นส่วนที่ใช้ควบคุมการทดสอบหรือหยุดการรันโปรแกรม

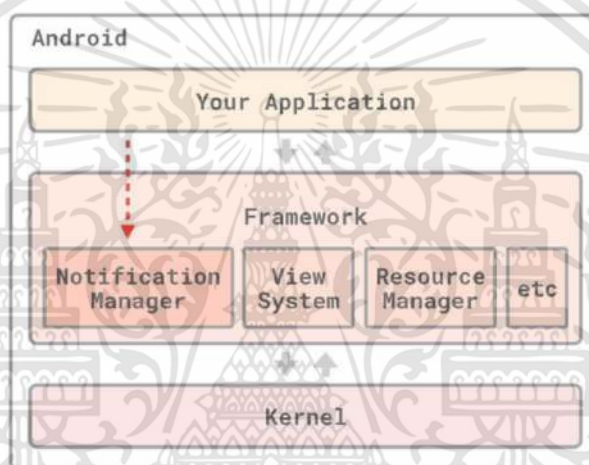
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 ศึกษาการทำแจ้งเตือน (Notification in Android)

การแจ้งเตือนคือข้อความที่ Android แสดงภายนอก UI ของแอปเพื่อให้การแจ้งเตือน การสื่อสารจากบุคคลอื่น หรือข้อมูลอื่น ๆ ในเวลาที่เหมาะสมจากแอปพลิเคชันของคุณแก่ผู้ใช้ ผู้ใช้สามารถแตะการแจ้งเตือนเพื่อเปิดแอปพลิเคชันของคุณหรือดำเนินการจากการแจ้งเตือนได้โดยตรง

2.5.1 การทำงานของ Notification บนแอนดรอยด์

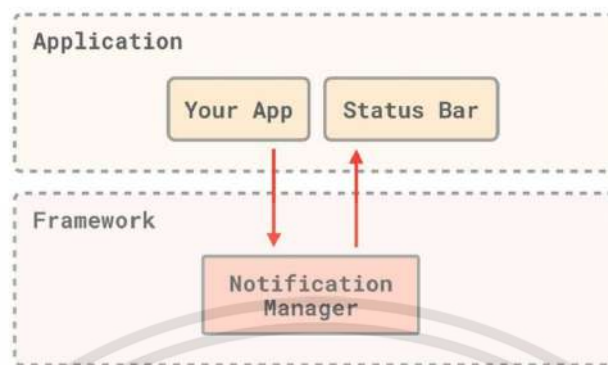
การทำงานจะทำงานอยู่บน Android Framework ซึ่งเป็นหัวใจหลักในการทำงานของระบบแอนดรอยด์ที่มีไว้ให้แอปต่าง ๆ เรียกใช้งานดังแสดงในรูปที่ 2.12



รูปที่ 2.12 การทำงานของ Notification บนแอนดรอยด์ [12]

ซึ่งใน Android Framework ก็จะมีการแบ่งการทำงานแยกกันออกไปตามหน้าที่ และส่วนที่ทำหน้าที่เกี่ยวกับระบบ Notification ก็จะมีชื่อเรียกว่า Notification Manager และ Status Bar ของแอนดรอยด์ที่ทำหน้าที่แสดง Notification จากแอปต่าง ๆ ก็จะได้รับข้อมูลมาจาก Notification Manager ดังนั้นเมื่อใดก็ตามที่แอปพลิเคชันในเครื่องส่ง Notification ไปให้ Notification Manager ก็จะถูกส่งไปเพื่อแสดงใน Notification Drawer ที่อยู่ใน Status Bar ดังแสดงในรูปที่ 2.13

ด้วยเหตุนี้ จึงเป็นที่มาว่านักพัฒนาต้องการแสดงข้อความแบบ Notification ให้กับผู้ใช้ จะต้องเรียกคำสั่งผ่าน Notification Manager อยู่เสมอ



รูปที่ 2.13 การทำงานภายในการแจ้งเตือน [12]

2.5.2 ความแตกต่างของ Notification ในแต่ละเวอร์ชัน (Version) / ยี่ห้อ
 แถบการแจ้งเตือนของ Android เวอร์ชันต่าง ๆ จะเป็นดังแสดงในรูปที่ 2.14



รูปที่ 2.14 แถบการแจ้งเตือนต่าง ๆ [12]

เมื่อแอนดรอยด์มีการปล่อยอัปเดต (Update) เวอร์ชันใหม่ ๆ ก็ต้องมาพร้อมกับการเปลี่ยนแปลง UI ซึ่งรวมไปถึง Notification Drawer เช่นกัน ดังนั้นหากมองข้ามตรงจุดนี้ไป อาจจะทำให้ Notification ของแอปพลิเคชันแสดงผลได้ไม่เหมาะสมกับเวอร์ชันนั้น ๆ

แต่ Notification Drawer บนแอนดรอยด์ในเวอร์ชันหลัง ๆ เริ่มมีความแตกต่างกัน น้อยมาก รวมถึงฟีเจอร์ที่มีให้เรียกใช้งานเช่นกัน และนอกจากเวอร์ชันแล้วอุปกรณ์แอนดรอยด์ที่มียี่ห้อต่างกันก็อาจจะทำให้ UI ของ Notification Drawer แตกต่างเช่นกันดังแสดงในรูปที่ 2.15

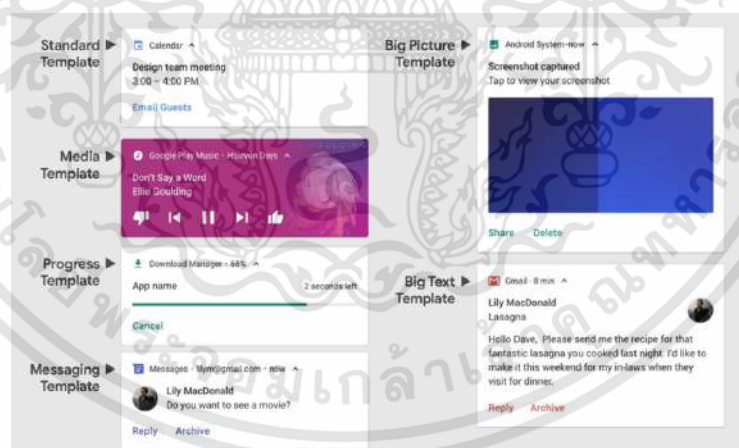
แต่ UI ของแต่ละยี่ห้อก็จะทำตาม UI Guideline Standard ที่ทางแอนดรอยด์กำหนดไว้เช่นกัน เพื่อป้องกันการแสดงผล Notification ที่ไม่เหมาะสม



รูปที่ 2.15 แถบแจ้งเตือนการเวอร์ชัน API [12]

2.5.3 Notification Template

รูปแบบในการแสดงผลของ Notification จะขึ้นอยู่กับโค้ด (Code) ซึ่งแอนดรอยด์ได้เตรียมคำสั่งง่าย ๆ เพื่อแสดงผล Notification ในรูปแบบต่าง ๆ โดยเรียกว่า Template เพื่อให้ นักพัฒนาสามารถนำ Template ไปใช้งานให้ตรงกับความต้องการได้ง่ายขึ้น ซึ่งมีทั้งหมด 6 Template ดังแสดงในรูปที่ 2.16



รูปที่ 2.16 รูปแบบในการแสดงผลของ Notification [12]

โดยทั้งหมดนี้สามารถกำหนดได้ผ่านโค้ดทั้งหมด และยังสามารถแก้ไขบางส่วนเพิ่มเติมได้ตามที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4 คำสั่งพื้นฐานของ Notification

Notification Channel และ Notification Channel Group เป็นสิ่งที่เพิ่มเข้ามาใน Android 8.0 Oreo (API 26) ขึ้นไป โดยบังคับให้นักพัฒนาต้องสร้าง Channel ขึ้นมาสำหรับ Notification ทุกตัวที่จะแสดงในแอปพลิเคชัน ถ้ามี Channel จำนวนมาก และจะแบ่งกลุ่มให้กับ Channel ก็ให้สร้าง Channel Group (สามารถสร้าง Channel ที่ไม่มี Channel Group ได้) โดยคำสั่งที่ใช้ในการสร้าง Channel จะมีลักษณะดังแสดงในรูปที่ 2.17

```

fun initNotificationChannel(context: Context) {
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        val id = "new_promotion"
        val name = "new_promotion"
        val importance = NotificationManager.IMPORTANCE_HIGH
        val channel = NotificationChannel(id, name, importance)
        val manager = context.getSystemService(Context.NOTIFICATION_SERVICE) as NotificationManager
        manager.createNotificationChannel(channel)
    }
}

```

รูปที่ 2.17 โปรแกรมการสร้าง Channel Notification [13]

2.5.4.1 ส่วนประกอบของ Channel

- 1) Channel ID : ID ของ Channel โดยห้ามซ้ำกับ Channel ตัวอื่นที่อยู่ในแอปพลิเคชันเดียวกัน
- 2) Channel Name : ชื่อของ Channel ที่ใช้แสดงให้ผู้ใช้เห็นที่ หน้าเมนู Notification ของ App Info
- 3) Importance : ระดับความสำคัญของ Notification ที่อยู่ใน Channel นี้ ซึ่งจะมีผลต่อเงื่อนไขในการแสดงผลของ Notification ซึ่งระบบแอนดรอยด์จะใช้แทน การกำหนด Priority ของ Notification ในเวอร์ชันเก่า

2.5.4.2 คำสั่งนี้ใช้เฉพาะ Android 8.0 Oreo (API 26) ขึ้นไปเท่านั้น

เนื่องจากคำสั่งดังกล่าวถูกเพิ่มเข้ามาตั้งแต่ Android 8.0 Oreo (API 26) ถ้าแอปรองรับเวอร์ชันที่ต่ำกว่านั้นจะต้องตรวจสอบเวอร์ชันแอนดรอยด์ของเครื่องที่เรียกใช้งาน แอปพลิเคชันด้วยเสมอ ในกรณีที่ต้องการสร้าง Channel Group ด้วยก็จะต้องเพิ่มคำสั่งเข้ามา ดังรูปที่ 2.18

```

fun initNotificationChannel(context: Context) {
    if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        val groupId = "announcement"
        val groupName = "Announcement"
        val group = NotificationChannelGroup(groupId, groupName)

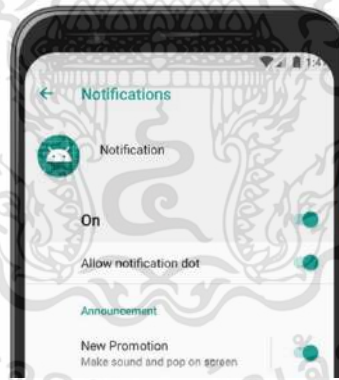
        val channelId = "new_promotion"
        val channelName = "New Promotion"
        val channel = NotificationChannel(channelId, channelName, NotificationManager.IMPORTANCE_HIGH)
        channel.group = groupId

        val manager = context.getSystemService(context.NOTIFICATION_SERVICE) as NotificationManager
        manager.createNotificationChannelGroup(group)
        manager.createNotificationChannel(channel)
    }
}

```

รูปที่ 2.18 สร้าง Channel Group [14]

การสร้าง Channel Group ก็คล้ายกับการสร้าง Channel ที่ต้องกำหนดชื่อและ ID ให้กับ Channel Group ด้วย และต้องกำหนด Channel Group ให้กับ Channel โดยใช้ ID ของ Channel Group ด้วย เมื่อสร้าง Channel และ Channel Group เสร็จแล้วก็ให้เพิ่มเข้าไปใน Notification Manager ทั้งคู่ โดยจะต้องเริ่มทำการสร้าง Channel Group ก่อนสร้าง Channel เสมอ ไม่งั้นจะทำให้แอปพลิเคชันพังเพราะหา Channel Group ไม่เจอได้และตัวอย่างหน้าจอ แสดงการแจ้งเตือน ดังแสดงในรูปที่ 2.19



รูปที่ 2.19 ตัวอย่างหน้าจอแสดงการแจ้งเตือน [14]

เพียงเท่านี้ก็ทำให้แอปพลิเคชันมี Channel และ Channel Group แล้ว โดยสามารถตรวจสอบได้จากหน้าเมนู Notification ใน App Info

2.5.4.3 การสร้าง Notification Channel และ Notification Channel Group

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) ให้สร้างแค่ตอนเปิดแอปพลิเคชันครั้งแรกเท่านั้น เพราะว่าเมื่อทำการสร้างขึ้นมาแล้ว ก็จะอยู่ติดแอปพลิเคชันตลอดไป จนกว่าจะลบแอปพลิเคชัน (Uninstall) หรือลบข้อมูลในแอปพลิเคชัน (Clear Data) เพราะว่าการสร้างแค่ครั้งเดียวก็เพียงพอ นักพัฒนาจึงใส่คำสั่งไว้ในคลาส Application เพื่อให้ทำการสร้างทันทีที่เปิดแอปพลิเคชันขึ้นมาดังแสดงในรูปที่ 2.20

```
// AwesomeApplication.kt
class AwesomeApplication : Application() {
    override fun onCreate() {
        /* ... */
        initNotificationChannel(applicationContext)
    }

    private fun initNotificationChannel(context: Context) {
        /* ... */
    }
}
```

รูปที่ 2.20 ชุดคำสั่ง [14]

2) โดย Channel และ Channel Group สามารถสร้างทับของเดิมได้ ดังนั้น นักพัฒนาส่วนใหญ่จะไม่สนว่ามันจะถูกสร้างซ้ำอีกครั้งในตอนทีคลาส Application ถูกสร้างขึ้น

2.5.4.4 สร้างข้อมูลที่ต้องใช้ในการแสดง Notification

ข้อมูลที่จำเป็นต้องมีใน Notification จะมีอยู่ 5 อย่าง คือ Context, Channel ID, Small Icon, Content Title และ Content Text ดังแสดงในรูปที่ 2.21 และตัวอย่างหน้าจอแสดงการแจ้งเตือน ดังแสดงในรูปที่ 2.22

```
val context: Context = /* ... */
val channelId = "new_promotion"
val title = "Are you looking for awesome Android article?"
val text = "why don't you look at Sleeping For Less website. There's a lot of interest article about Android a"
val notification = NotificationCompat.Builder(context, channelId).apply {
    setSmallIcon(R.drawable.ic_notification)
    setContentTitle(title)
    setContentText(text)
}.build()
```

รูปที่ 2.21 โปรแกรมการกำหนดหน้าต่างการแจ้งเตือน [14]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) Channel ID : ID ของ Notification Channel ที่เคยสร้างไว้แล้ว ถ้าเป็น Channel ID ไม่เคยถูกสร้างขึ้นมาก่อนก็จะทำให้แอปพลิเคชันพังทันที
- 3) Small Icon : สำหรับกำหนด Drawable Resource เพื่อที่จะแสดงเป็นภาพไอคอนตอนแสดง Notification
- 4) Content Title : สำหรับกำหนดชื่อหัวข้อของ Notification นั้น ๆ เพื่อให้ผู้ใช้รู้ว่า เป็น Notification ในเรื่องอะไร
- 5) Content Text : สำหรับกำหนดรายละเอียด Notification เพื่อให้ผู้ใช้เห็นเนื้อหาจาก Notification นั้น ๆ



รูปที่ 2.22 ตัวอย่างหน้าจอแสดงการแจ้งเตือน [14]

ในความจริงแล้วไม่จำเป็นต้องกำหนด Content Title และ Content Text ก็ยังสามารถสร้าง Notification ได้ แต่โดยปกติแล้ว Notification ที่ดีควรมีทั้ง Content Title และ Content Text เพื่อแสดงข้อมูลให้ผู้ใช้เห็นได้มากที่สุด ซึ่งจะช่วยให้ผู้ใช้สนใจใน Notification นั้น ๆ มากขึ้น

2.5.4.4 แสดง Notification ด้วย Notification Manager โดยใช้ชุดคำสั่ง
ดังแสดงในรูปที่ 2.23

เมื่อเตรียมข้อมูลสำหรับ Notification เสร็จเรียบร้อยแล้ว ก็ถึงเวลาส่งข้อมูล Notification ให้ Notification Manager เพื่อแสดงให้ผู้ใช้เห็นด้วยคำสั่งแบบนี้

```

val id = 0
val notification: Notification = /* ... */
val manager = context.getSystemService(Context.NOTIFICATION_SERVICE) as NotificationManager
manager.notify(id, notification)

```

รูปที่ 2.23 ชุดคำสั่ง [14]

นอกจากข้อมูลของ Notification แล้ว ยังมีสิ่งอีกอย่างที่นักพัฒนาต้องกำหนดทุกครั้ง เวลาสั่งให้ Notification Manager แสดง Notification นั่นก็คือ Notification ID

2.5.4.5 Notification ID

ID ของ Notification จะต้องเป็น Unique ID ที่ไม่ซ้ำกับ Notification ตัวอื่น ๆ ที่อยู่ในแอปพลิเคชันเดียวกัน กำหนดค่าเป็นตัวเลขได้สูงสุด 4 หลัก (ตั้งแต่ 0 ถึง 9999) ถ้ามี Notification หลายตัวแต่กำหนด ID ซ้ำกัน จะทำให้ Notification ตัวล่าสุดไปแสดงทับตัวที่แสดงก่อนหน้าทันที แต่เมื่อกำหนดเป็นคนละ ID ก็จะมี Notification คนละตัวกัน ทำให้ระบบแสดง Notification แยกกัน ทั้งนี้ก็เพราะว่านักพัฒนาสามารถอัปเดตข้อมูลที่แสดงใน Notification ใหม่ได้ ถึงแม้ว่าจะแสดงอยู่ก็ตาม โดยอ้างอิงจาก ID ของ Notification ตัวนั้น

2.6 ศึกษา Database

Database หรือ ฐานข้อมูล คือ กลุ่มของข้อมูลที่ถูกเก็บรวบรวมไว้ โดยมีความสัมพันธ์ซึ่งกันและกัน โดยไม่ได้บังคับว่าข้อมูลทั้งหมดนี้จะต้องเก็บไว้ในแฟ้มข้อมูลเดียวกันหรือแยกเก็บหลาย ๆ แฟ้มข้อมูล

ระบบฐานข้อมูล (Database System) คือ ระบบที่รวบรวมข้อมูลต่าง ๆ ที่เกี่ยวข้องกันเข้าไว้ด้วยกันอย่างมีระบบมีความสัมพันธ์ระหว่างข้อมูลต่าง ๆ ที่ชัดเจน ในระบบฐานข้อมูลจะประกอบด้วยแฟ้มข้อมูลหลายแฟ้มที่มีข้อมูล เกี่ยวข้องสัมพันธ์กันเข้าไว้ด้วยกันอย่างเป็นระบบและเปิดโอกาสให้ผู้ใช้สามารถใช้งานและดูแลรักษาป้องกันข้อมูลเหล่านี้ ได้อย่างมีประสิทธิภาพ โดยมีซอฟต์แวร์ที่เปรียบเสมือนสื่อกลางระหว่างผู้ใช้และโปรแกรมต่าง ๆ ที่เกี่ยวข้องกับการใช้ฐานข้อมูล เรียกว่า ระบบจัดการฐานข้อมูล หรือ DBMS (data base management system) มีหน้าที่ช่วยให้ผู้ใช้เข้าถึงข้อมูลได้ง่ายสะดวกและมีประสิทธิภาพ การเข้าถึงข้อมูลของผู้ใช้อาจเป็นการสร้าง

ฐานข้อมูล การแก้ไขฐานข้อมูล หรือการตั้งคำถามเพื่อให้ได้ข้อมูลมา โดยผู้ใช้ไม่จำเป็นต้องรับรู้เกี่ยวกับรายละเอียดภายในโครงสร้างของฐานข้อมูล

2.6.1 ประโยชน์ของฐานข้อมูล

2.6.1.1 ลดการเก็บข้อมูลที่ซ้ำซ้อน ข้อมูลบางชุดที่อยู่ในรูปของแฟ้มข้อมูล อาจมีปรากฏอยู่หลาย ๆ แห่ง เพราะมีผู้ใช้ข้อมูลชุดนี้หลายคน เมื่อใช้ระบบฐานข้อมูลแล้วจะช่วยให้ ความซ้ำซ้อนของข้อมูลลดน้อยลง

2.6.1.2 รักษาความถูกต้องของข้อมูล เนื่องจากฐานข้อมูลมีเพียงฐานข้อมูล เดียว ในกรณีที่มีข้อมูลชุดเดียวกันปรากฏอยู่หลายแห่งในฐานข้อมูล ข้อมูลเหล่านี้จะต้องตรงกัน ถ้า มีการแก้ไขข้อมูลนี้ทุก ๆ แห่งที่ข้อมูลปรากฏอยู่จะแก้ไขให้ถูกต้องตามกันหมดโดยอัตโนมัติด้วย ระบบจัดการฐานข้อมูล

2.6.1.3 การป้องกันและรักษาความปลอดภัยให้กับข้อมูลทำได้อย่างสะดวก การป้องกันและรักษาความปลอดภัยกับข้อมูลระบบฐานข้อมูลจะให้เฉพาะผู้ที่เกี่ยวข้องเท่านั้นซึ่ง ก่อให้เกิดความปลอดภัย(security) ของข้อมูลด้วย

2.7 ศึกษาการใช้ Firebase

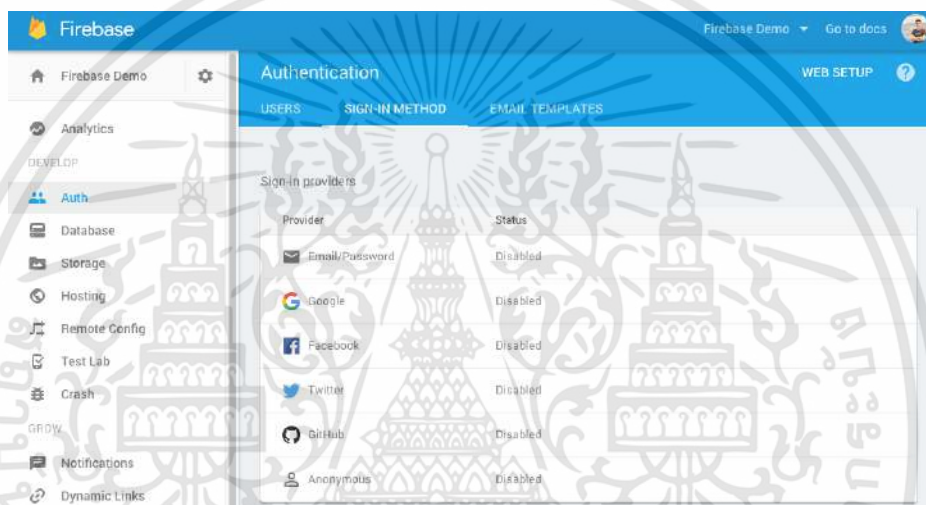
Firebase คือระบบโครงสร้างพื้นฐานที่จำเป็นสำหรับการพัฒนาแอปพลิเคชัน โดยที่ Firebase เองถูกออกแบบมาให้สามารถทำงานได้ในหลากหลาย Platform ทั้ง Web Application, Mobile Application ที่สามารถใช้งานได้ทั้งในระบบปฏิบัติการ iOS และระบบปฏิบัติการ Android

โดยทั่วไปแล้ว โครงสร้างพื้นฐานที่จำเป็นในการพัฒนาแอปพลิเคชัน (Application Infrastructure) จำเป็นจะต้องมีอะไร Firebase ถูกออกแบบมาเพื่อสิ่งนั้น และเตรียมโครงสร้าง นั้นให้นักพัฒนาสามารถนำไปใช้งานโดยไม่ต้องพัฒนาโครงสร้างนั้น ๆ ขึ้นมาใหม่ทุกครั้ง ตัวอย่าง เช่น Analytics เก็บสถิติ, Authentication ยืนยันตัวตน, Realtime Database เก็บข้อมูลแบบ เรียลไทม์, Cloud Messaging สำหรับการส่งข้อความ และอื่น ๆ อีกมากมาย เหล่านี้ล้วนเป็น โครงสร้างพื้นฐานที่จำเป็นในการพัฒนาแอปพลิเคชัน หากต้องพัฒนาขึ้นมาใหม่ทั้งหมด จำเป็นต้อง ใช้เวลาในการพัฒนาที่มากขึ้น รวมไปถึงการพัฒนาต่อยอด ก็จะสามารถพัฒนาต่อยอดได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 ศึกษา Firebase Authentication

Firebase Authentication จะเป็นบริการที่เข้ามาจัดการ backend ทั้งหมด ได้แก่ การลงทะเบียน (register), การเข้าสู่ระบบ (sign-in), การเปลี่ยนรหัสผ่าน (reset password), โดยจะมี SDK ให้ทั้ง Android, iOS และ Web นำไปติดตั้งและใช้งาน ซึ่งรองรับการเข้าสู่ระบบหลากหลายรูปแบบทั้งจาก social network, Email และ Password ของผู้ใช้งาน หรือแบบไม่ระบุตัวตน (Anonymous) แสดงดังรูปที่ 2.24



รูปที่ 2.24 การเข้าสู่ระบบรูปแบบต่าง ๆ ของ Firebase Authentication [16]

Authentication SDK ใน build.gradle ของ app-level ที่ใช้ในการติดตั้งและใช้งาน แสดงดังรูปที่ 2.25

```
dependencies {
    compile 'com.google.firebase:firebase-auth:11.8.0'
}
```

รูปที่ 2.25 Authentication SDK [16]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 ศึกษา Realtime Database

Firebase Realtime Database เป็น NoSQL cloud database ที่เก็บข้อมูลในรูปแบบของ JSON และมีการ sync ข้อมูลแบบ Realtime กับทุก devices ที่เชื่อมต่อแบบอัตโนมัติในเสี้ยววินาที รองรับการทำงานเมื่อ offline (ข้อมูลจะถูกเก็บไว้ใน local จนกระทั่งกลับมา online ก็จะทำให้ทำการ sync ข้อมูลให้อัตโนมัติ) รวมถึงมี Security Rules ให้เราสามารถออกแบบเงื่อนไขการเข้าถึงข้อมูลทั้งการ read และ write ได้ตั้งใจ ทั้ง Android, iOS และ Web

ทำการสร้างโครงการใน firebase เนื่องจาก firebase ถูกออกแบบมาให้เป็น API และ Cloud Storage สำหรับพัฒนา Realtime Application รองรับหลาย Platform ทั้ง IOS App, Android App, Web App และในโครงการนี้ต้องการใช้ข้อมูลแบบ Realtime ในการจัดเก็บข้อมูลของผู้ใช้งาน โดยกรอกรายละเอียดในการสร้างโครงการ โดยตั้งชื่อโครงการเป็น “I’m-ready”

2.9.1 ความสำคัญที่สำคัญ

2.9.1.1 เรียลไทม์ แทนที่จะใช้คำขอ HTTP ทั่วไปฐานข้อมูลเรียลไทม์ของ Firebase จะใช้การซิงโครไนซ์ (Synchronize) ข้อมูลทุกครั้งที่มีข้อมูลมีการเปลี่ยนแปลงอุปกรณ์ที่เชื่อมต่อจะได้รับการอัปเดตนั้นภายในมิลลิวินาที

2.9.1.2 ออฟไลน์ (Offline) แอปพลิเคชัน Firebase ยังคงตอบสนองแม้ในขณะที่ออฟไลน์เนื่องจาก Firebase Realtime Database SDK ยังคงเก็บข้อมูลของคุณไว้ในดิสก์ (Disk) เมื่อสร้างการเชื่อมต่อใหม่แล้วอุปกรณ์ไคลเอนต์ (Client) จะได้รับการเปลี่ยนแปลงที่พลาดไปโดยซิงโครไนซ์กับสถานะเซิร์ฟเวอร์ (Server) ปัจจุบัน

2.9.1.3 เข้าถึงได้จากอุปกรณ์ไคลเอนต์ ฐานข้อมูลเรียลไทม์ (real-time) ของ Firebase สามารถเข้าถึงได้โดยตรงจากอุปกรณ์เคลื่อนที่หรือเว็บเบราว์เซอร์ ไม่จำเป็นต้องมีแอปพลิเคชันเซิร์ฟเวอร์ การรักษาความปลอดภัยและการตรวจสอบข้อมูลมีให้ใช้งานผ่านกฎความปลอดภัยฐานข้อมูลแบบเรียลไทม์ของ Firebase ซึ่งเป็นไปตามนิพจน์ที่ ดำเนินการเมื่อข้อมูลถูกอ่านหรือเขียน

2.9.1.4 ปรับขนาดในหลายฐานข้อมูล ด้วยฐานข้อมูลเรียลไทม์ของ Firebase ในแผนการกำหนดราคาของ Blaze คุณสามารถรองรับความต้องการข้อมูลของแอปพลิเคชันได้ตามขนาดโดยการแบ่งข้อมูลของคุณไปยังอินสแตนซ์ฐานข้อมูลหลายอินสแตนซ์ (Instance) ในโครงการ Firebase เดียวกัน ปรับปรุงการตรวจสอบสิทธิ์ด้วย Firebase Authentication ในโครงการของคุณและตรวจสอบสิทธิ์ผู้ใช้ในอินสแตนซ์ฐานข้อมูลของคุณ

ควบคุมการเข้าถึงข้อมูลในแต่ละฐานข้อมูลด้วยกฎ Firebase Realtime Database ที่กำหนดเอง สำหรับแต่ละอินสแตนซ์ฐานข้อมูล

2.9.2 การทำงาน

ฐานข้อมูลเรียลไทม์ของ Firebase ช่วยให้คุณสามารถสร้างแอปพลิเคชันที่ทำงานร่วมกันได้อย่างสมบูรณ์โดยอนุญาตให้เข้าถึงฐานข้อมูลได้โดยตรงจากโค้ดฝั่งไคลเอนต์ ข้อมูลจะยังคงอยู่ในเครื่องและแม้ในขณะที่ออฟไลน์เหตุการณ์แบบเรียลไทม์ก็ยังคงเริ่มทำงานต่อไปทำให้ผู้ใช้ปลายทางได้รับประสบการณ์ที่ตอบสนอง เมื่ออุปกรณ์ได้รับการเชื่อมต่อฐานข้อมูลเรียลไทม์จะซิงโครไนซ์การเปลี่ยนแปลงข้อมูลในเครื่องกับการอัปเดตระยะไกลที่เกิดขึ้นในขณะที่ไคลเอนต์ออฟไลน์โดยจะรวมข้อขัดแย้งใด ๆ โดยอัตโนมัติ

ฐานข้อมูลแบบเรียลไทม์มีภาษาทวิที่ใช้ฟังก์ชันที่ยืดหยุ่นซึ่งเรียกว่ากฎความปลอดภัยของฐานข้อมูลแบบเรียลไทม์ของ Firebase เพื่อกำหนดโครงสร้างข้อมูลของคุณและเวลาที่สามารถอ่านหรือเขียนข้อมูลได้ เมื่อผสานรวมกับ Firebase Authentication นักพัฒนาสามารถกำหนดได้ว่าใครมีสิทธิ์เข้าถึงข้อมูลใดบ้างและจะเข้าถึงได้อย่างไร Realtime Database เป็นฐานข้อมูล NoSQL และด้วยเหตุนี้จึงมีการปรับแต่งและการทำงานที่แตกต่างกันเมื่อเทียบกับฐานข้อมูลเชิงสัมพันธ์ Realtime Database API ได้รับการออกแบบมาเพื่อให้สามารถดำเนินการได้อย่างรวดเร็วเท่านั้น สิ่งนี้ช่วยให้คุณสร้างประสบการณ์เรียลไทม์ที่ยืดหยุ่นซึ่งสามารถให้บริการผู้ใช้หลายล้านคนโดยไม่กระทบต่อการตอบสนอง ด้วยเหตุนี้จึงเป็นสิ่งสำคัญที่จะต้องพิจารณาว่าผู้ใช้ต้องเข้าถึงข้อมูลของคุณอย่างไรจากนั้นจึง จัดโครงสร้างให้เหมาะสม

2.9.3 การจัดเก็บข้อมูลประเภทอื่น

2.9.3.1 Cloud Firestore เป็นฐานข้อมูลที่ยืดหยุ่นและปรับขนาดได้สำหรับการพัฒนาอุปกรณ์เคลื่อนที่เว็บและเซิร์ฟเวอร์จาก Firebase และ Google Cloud Platform หากต้องการเรียนรู้เพิ่มเติมเกี่ยวกับความแตกต่างระหว่างตัวเลือกฐานข้อมูลโปรดดูที่เลือกฐานข้อมูล: Cloud Firestore หรือ Realtime Database

2.9.3.2 Firebase Remote Config จัดเก็บคู่มือ ค่าที่นักพัฒนาระบุไว้เพื่อเปลี่ยนพฤติกรรมและรูปลักษณ์ของแอปพลิเคชันโดยไม่ต้องให้ผู้ใช้ดาวน์โหลดอัปเดต

2.9.3.3 Firebase Hosting โฮสต์ HTML, CSS และ JavaScript สำหรับเว็บไซต์ของคุณตลอดจนเนื้อหาอื่น ๆ ที่นักพัฒนาจัดทำให้ เช่น กราฟิกแบบอักษรและไอคอน

2.9.3.4 Cloud Storage จัดเก็บไฟล์เช่นรูปภาพวิดีโอและเสียงตลอดจนเนื้อหาอื่น ๆ ที่ผู้ใช้สร้างขึ้น

2.10 การศึกษาทำ Run background

Service ในแอนดรอยด์ Service เป็น 1 ใน 4 Component พื้นฐานของแอปพลิเคชัน (มี Activity, Service, Broadcast Receiver และ Content Provider) โดยมีหน้าที่ทำงานบางอย่างที่ใช้เวลานานและไม่จำเป็นต้องแสดง UI ให้ผู้ใช้เห็นตลอดเวลา

แอปพลิเคชันจำพวก Music Player ที่สามารถเล่นเพลงได้ถึงแม้ว่าผู้ใช้จะย่อหรือปิดแอปพลิเคชันแล้วก็ตาม แสดงดังรูปที่ 2.26 หรือแอปพลิเคชันจำพวก Running Tracker ที่สามารถรู้ตำแหน่งของผู้ใช้ได้ตลอดเวลาถึงแม้ว่าจะไม่ได้เปิดแอปฯอยู่ ก็เพราะว่าใช้ Service เข้ามาช่วยนั่นเอง ซึ่ง Service นั้นสามารถทำงานอยู่เบื้องหลัง (Background) ได้ โดยไม่ต้องแสดง UI ให้ผู้ใช้เห็นตลอดเวลาแบบ Activity

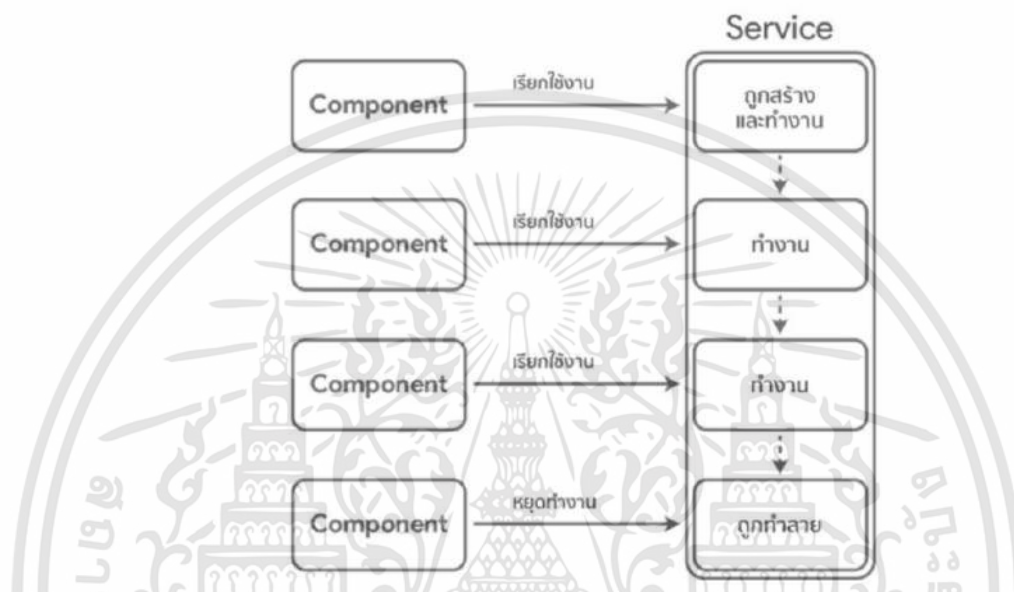


รูปที่ 2.26 ตัวอย่างแอปพลิเคชัน [17]

Service จะถูกเรียกใช้งานจากที่ไหนก็ได้ที่มี Context ให้ใช้งานหรือเป็น Component ที่สืบทอดมาจาก Context ซึ่งในบทความนี้จะเรียกฝั่งที่เรียกใช้งาน Service ว่า Component

Service ตัวหนึ่งจะถูกสร้างขึ้นมาเพียงครั้งเดียวเท่านั้น แต่จะเรียกใช้งานกี่ครั้งก็ได้ จนกว่าจะหยุดทำงานหรือถูกทำลาย Service จะต่างจาก Activity ตรงที่ว่า Service ตัวหนึ่งจะสามารถสร้างขึ้นมาได้แค่ตัวเดียวเท่านั้น ไม่สามารถสร้างตัวเดิมซ้ำ ๆ ได้เหมือนอย่าง Activity ดังรูปที่ 2.27

นั่นหมายความว่าตอนที่ Component เรียกใช้งาน Service ครั้งแรกสุด มันจะสร้าง Service ขึ้นมาแล้วเริ่มทำงาน แต่เมื่อเรียกใช้งานครั้งต่อไปก็จะเริ่มทำงานจาก Service ตัวที่เคยสร้างไว้ทันที ไม่มีการสร้างขึ้นใหม่ เว้นแต่ว่ามันจะถูกทำลายทิ้ง ถึงจะสร้างขึ้นใหม่อีกครั้ง



รูปที่ 2.27 ศูนย์กลางในการทำงาน [17]

ดังนั้น Service จึงเหมาะแก่การออกแบบให้เป็นศูนย์กลางในการทำงานที่ต้องจำเป็นต้องอยู่เบื้องหลัง โดยที่ไม่ได้สนใจว่าผู้ใช้จะทำอะไรอยู่หรือเปิดหน้าต่างไหนอยู่ เช่น Service สำหรับส่งข้อมูลขึ้น Service หรือ Sync ข้อมูลลงใน Local Database เป็นระยะ ๆ เพื่อให้ User จะได้ไม่ต้องมานั่งรอทุกครั้งเมื่อเปิดแอปพลิเคชันขึ้นมา

2.10.1 Service แบ่งออกเป็น 3 ประเภท

2.10.1.1 Background Service

เป็น Service ที่จะทำงานโดยไม่มีการแสดงผลหรือแจ้งเตือนให้ผู้ใช้ทราบ ซึ่งเป็นรูปแบบพื้นฐานที่สุดของ Service เหมาะกับการทำงานบางอย่างที่ต้องใช้เวลานาน แต่ไม่จำเป็นต้องแสดงให้ผู้ใช้เห็นที่กำลังทำงานอยู่หรือทำงานเสร็จแล้ว

2.10.1.2 Foreground Service

เป็น Service ที่จะทำงานบางอย่างที่ใช้เวลานาน ๆ แต่จะแสดงการทำงานหรือแจ้งเตือนให้ผู้ใช้เห็นผ่าน Notification ใน Status Bar

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.1.3 Bound Service

เป็น Service ที่ทำงานโดยสามารถรับส่งข้อมูลระหว่าง Component ที่เรียกใช้งาน Service ได้ตลอดเวลาที่ Service ทำงาน โดยจะมีตัวกลางในการสื่อสารระหว่างทั้งสองฝั่งที่เรียกว่า I Binder โดย Service แบบนี้จะทำงานได้นานเท่าที่ Component นั้น ๆ ยังทำงานอยู่

2.11 Received Signal Strength (RSSI)

2.11.1 BLUETOOTH

เทคโนโลยี LAN ไร้สายที่ออกแบบมาเพื่อเชื่อมต่ออุปกรณ์ต่าง ๆ ทำหน้าที่เมื่ออยู่ใกล้กัน เปรียบเสมือนการสร้างเครือข่ายส่วนบุคคล (PAN)

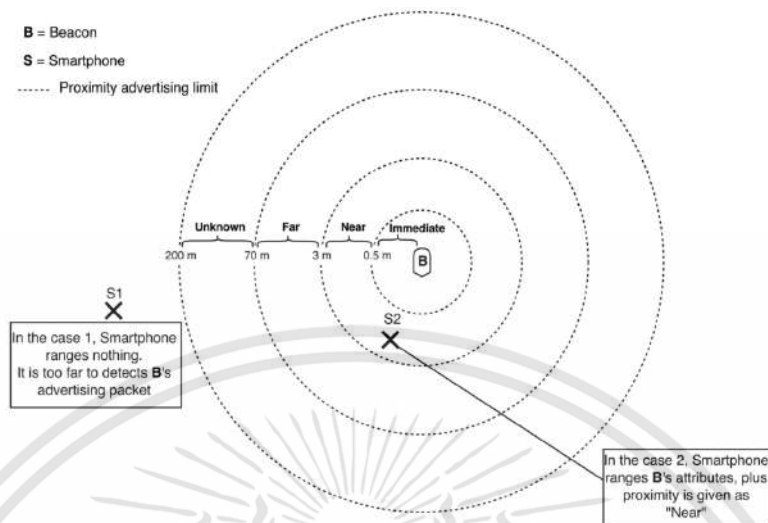
บริหารจัดการโดย Bluetooth Special Interest Group (SIG) IEEE กำหนดมาตรฐาน Bluetooth เป็น IEEE 802.15.1 แต่ไม่ดูแลรักษาอีกต่อไปมาตรฐาน

Bluetooth LAN เป็นเครือข่ายเฉพาะกิจ แกดเจ็ต พบกันและสร้างเครือข่ายที่เรียกว่า piconet

การใช้คลื่นวิทยุ UHF ความยาวคลื่นสั้นในแถบ ISM ตั้งแต่ 2.4 GHz ถึง 2.485 GHz

2.11.1.1 IBEACON

แนวคิดบีดคอนที่ใช้ Bluetooth เวอร์ชันของ Apple ซึ่งช่วยให้อุปกรณ์ Bluetooth ออกอากาศหรือรับสิ่งเล็ก ๆ ได้และขึ้นส่วนข้อมูลคงที่ในระยะทางสั้น ๆ ประกอบด้วย 2 ส่วน คือ ผู้ประกาศ (อุปกรณ์บีดคอน) และเครื่องรับ (แอปสมาร์ตโฟน) ผู้ประกาศข่าวอยู่เสมอโฆษณา "ฉันอยู่ที่นี้และ ID ของฉันคือ ..." ในขณะที่ผู้รับตรวจพบแพ็คเกจวิทยุ Bluetooth เหล่านี้และทำทุกอย่างต้องทำตามระยะใกล้หรือไกล ดังรูปที่ 2.28 ความแตกต่างระหว่าง Beacon กับ Wi-Fi แสดงดังตารางที่ 2.2



รูปที่ 2.28 ศูนย์กลางในการทำงาน [18]

ตารางที่ 2.2 ความแตกต่างระหว่าง Beacon กับ Wi-Fi [19]

	Beacon	Wi-Fi
พลังงาน	Beacon ใช้แบตเตอรี่ในการทำงาน	Wi-Fi ใช้ไฟ AC ในการทำงาน ไม่ใช่แบตเตอรี่
ขอบเขต	มีช่วงถึง 230 ฟุต หรือ 70 เมตร	มีช่วงถึง 330 ฟุต หรือ 100 เมตร
ความแม่นยำ	มีความแม่นยำต่ำถึง 1 เมตร	มีความแม่นยำสูง (1-5 เมตร)
ค่าใช้จ่าย	ราคาถูก	ค่อนข้างแพง
สำหรับ	เหมาะสำหรับใช้ในร่มหรือในอาคาร	เหมาะสำหรับใช้ในร่มหรือในอาคาร

2.11.2 เทคโนโลยีและวิธีการ

2.10.2.1 iBeacon Apple เปิดตัว iBeacon ซึ่งเป็นเครื่องหมายการค้าที่มีเครื่องหมายการค้า เทคโนโลยี ในปี 2013 Summer Worldwide Developers การประชุมโปรโตคอล iBeacon ถือเป็นตัวถัดไปขั้นตอนการพัฒนาในระดับหลัง Near Field Communication (NFC) หรือเทคโนโลยี QR Code iBeacon ใช้ BLE (บลูทูธพลังงานต่ำ) มาตรฐานซึ่งเป็นส่วนหนึ่งของ Bluetooth เทคโนโลยี 4.0 มันขึ้นอยู่กับ การตรวจจับความใกล้ชิด BLE โดยเผยแพร่ ID ที่ไม่ซ้ำกันในระดับสากลนี้เป็นเอกลักษณ์สากลประกอบด้วยตัวแปร UUID คู่หลักและคู่รอง

UUID: นี้คือสตริงขนาด 16 ไบต์ที่ใช้เพื่อแยกความแตกต่างของขนาดใหญ่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่ม beacon ที่เกี่ยวข้อง

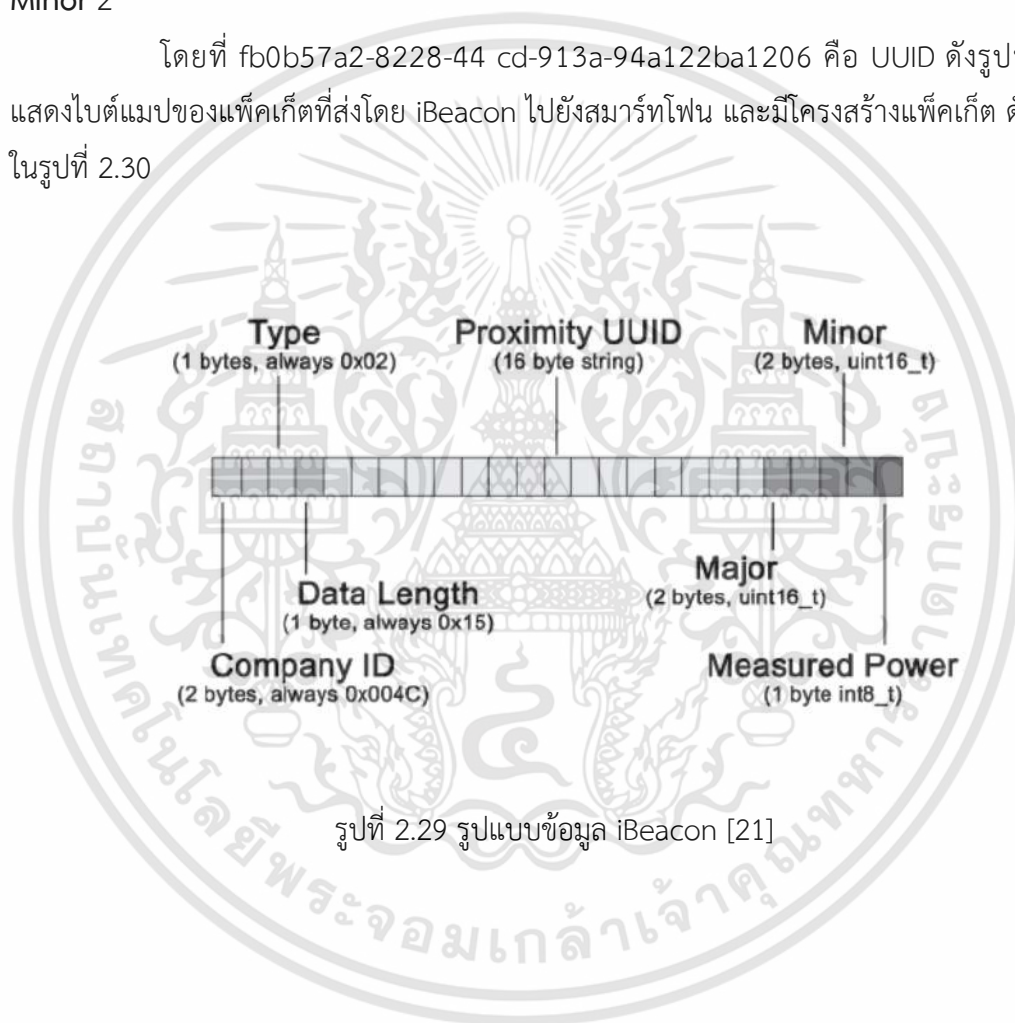
Major: นี่คือนามสกุล 2 ไบต์ที่ใช้เพื่อแยกความแตกต่างที่เล็กกว่าชุดย่อยของ beacon ภายในกลุ่มใหญ่

Minor: นี่คือนามสกุล 2 ไบต์ที่ใช้ระบุตัวบุคคล beacon

ตัวอย่างเฟรม iBeacon: fb0b57a2-8228-44 cd-913a-94a122ba1206 Major 1

Minor 2

โดยที่ fb0b57a2-8228-44 cd-913a-94a122ba1206 คือ UUID ดังรูปที่ 2.29 แสดงไบต์แมปของแพ็คเกจที่ส่งโดย iBeacon ไปยังสมาร์ทโฟน และมีโครงสร้างแพ็คเกจ ดังแสดงในรูปที่ 2.30



รูปที่ 2.29 รูปแบบข้อมูล iBeacon [21]

Byte 0-2: Standard BLE Flags

Byte 0: Length : 0x02
 Byte 1: Type: 0x01 (Flags)
 Byte 2: Value: 0x06 (Typical Flags)

Byte 3-29: Apple Defined iBeacon Data

Byte 3: Length: 0x1a
 Byte 4: Type: 0xff (Custom Manufacturer Packet)
 Byte 5-6: Manufacturer ID : 0x4c00 (Apple)
 Byte 7: SubType: 0x2 (iBeacon)
 Byte 8: SubType Length: 0x15
 Byte 9-24: Proximity UUID
 Byte 25-26: Major
 Byte 27-28: Minor
 Byte 29: Signal Power

รูปที่ 2.30 โครงสร้างแพ็คเกจ [21]

2.11.3 กรณีการใช้งาน

การคำนวณระยะทางโดยใช้ RSSI สำหรับการคำนวณระยะทางสูตรคือ

$$d = 10^{\left(\frac{T_x \text{ power} - \text{RSSI}}{20}\right)} \quad (2.1)$$

โดย

d คือ ระยะทาง

$T_x \text{ power}$ คือ กำลังในการส่งสัญญาณจากตัวบีมอน

การอ่าน RSSI ไม่เสถียรและขึ้นอยู่กับสิ่งแวดล้อมและปัจจัยอื่น ๆ ดังนั้นจึงให้น้อยลงผลลัพธ์ที่แม่นยำ อุปสรรคระหว่างบีมอนกับมือถือของลูกค้าถือมือถือในมุมที่ต่างกันและมือถือรุ่นต่าง ๆ (ที่มีเสาอากาศต่างกัน) คือ สาเหตุหลักที่ทำให้การอ่าน RSSI เปลี่ยนไปและความแม่นยำเพราะคำนวณระยะทางได้ไม่ดี

2.12 จำลองช่องสัญญาณ (Channel Model)

ทำแบบจำลองและหาคุณสมบัติเชิงสถิติของช่องสัญญาณ บีมอน BLE สภาวะแวดล้อมภายในอาคารในโครงการนี้ได้ใช้ LRM ทำแบบจำลองหาความสัมพันธ์ระหว่าง RSS ของบีมอนแต่ละตัวตามระยะทางระหว่างบีมอนกับโทรศัพท์มือถือที่ใช้วัด โดยที่ LRM จะมพื้นฐานอยู่บน

สูตรการส่งผ่านของฟรีส สำหรับสภาวะแวดล้อมใด ๆ สามารถกำหนดความสัมพันธ์ระหว่าง RSS กับระยะทางได้เป็น

$$P_r = 10n \log(d) + P_{ref} \quad (2.2)$$

เมื่อ P_r เป็น RSS

d เป็น ระยะทาง n

P_{ref} เป็น RSS ที่ระยะอ้างอิง 1 m

จุดประสงค์ในการทำแบบจำลองคือต้องการหาพารามิเตอร์ n และ P_{ref} ที่ทำให้เกิดค่าผิดพลาดน้อยสุด ซึ่งแบบจำลองนี้จะนำไปใช้ประมาณหาระยะทางจาก RSS สำหรับการหาตำแหน่งต่อไป ขั้นแรกจะต้องวัด RSS ที่ระยะทางที่กำหนดโดยกำหนดให้ที่ระยะทางที่ i d_i วัด RSS ได้ $P_{r,i}$ ทำการวัดทั้งหมด N ตำแหน่ง จะได้ค่าผิดพลาดของ RSS กำลังสองทั้งหมดเป็น

$$\sum_{i=2}^N E_i^2 = \sum_{i=2}^N [P_{r,i} + 10n \log(d_i) - P_{ref}]^2 \quad (2.3)$$

จากสมการที่ 2.3 ใช้วิธีค่าผิดพลาดกำลังสองน้อยสุดหา พารามิเตอร์ n และ P_{ref} ได้จาก

$$\begin{bmatrix} -10n \\ P_{ref} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^N [\log(d_i)]^2 & \sum_{i=1}^N \log(d_i) \\ \sum_{i=1}^N \log(d_i) & N \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^N \log(d_i) P_{r,i} \\ \sum_{i=1}^N P_{r,i} \end{bmatrix} \quad (2.4)$$

ค่าผิดพลาดที่เกิดขึ้นนี้เป็นผลกระทบโดยตรงจากการจางหายหลายวิถี โดยพิจารณาให้ค่าผิดพลาดที่ระยะทาง i ใด ๆ E_i เป็นตัวแปรสุ่ม E

$$E_i = P_{r,i} + 10n \log(d_i) - P_{ref} \quad (2.5)$$

$$E = \{E_1, E_2, \dots, E_N\} \quad (2.6)$$

หลังจากนั้นจะหาคุณสมบัติเชิงสถิติในเทอมของ PDF และ CDF ของค่าผิดพลาด หรือผลกระทบของการจางหายหลายวิถีโดยที่ PDF และ CDF มีนิยามเป็นสมการที่ (2.9)

$$f_E(E_i) = \frac{d}{dE_i} P\{E \leq E_i\} \quad (2.8)$$

$$F_E(E_i) = P\{E \leq E_i\} + \int_{-\infty}^{E_i} f_E(\tau) d\tau \quad (2.9)$$

เมื่อ f_E เป็น PDF F_E เป็น CDF และ P เป็นความน่าจะเป็นในบทความนี้ได้เทียบคุณสมบัติเชิงสถิติของผลกระทบจางหายวิธีนี้เทียบกับแบบจำลองสถิติแบบเกาส์ซึ่งมีนิยามของ PDF และ CDF เป็น

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/(2\sigma^2)} \quad (2.10)$$

$$F_X(x) = Q\left(\frac{x-\mu}{\sigma}\right) \quad (2.11)$$

เมื่อ X เป็นตัวแปรสุ่ม x เป็นค่าตัวแปรสุ่ม μ เป็นค่าเฉลี่ยของตัวแปรสุ่ม σ เป็นส่วนเบี่ยงเบนมาตรฐานของตัวแปรสุ่ม และฟังก์ชัน Q มีนิยามเป็น

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-t^2/2} dt \quad (2.12)$$

2.13 Path Loss Model

2.13.1 Power

การสูญเสียกำลังสัญญาณในอากาศ (Free Space Path loss) ซึ่งเกิดจากการที่คลื่นแพร่กระจายออกไปมูกว้าง ทำให้ความเข้มของกำลังคลื่นสัญญาณอ่อนตัวลงตามระยะทางที่เพิ่มขึ้นแบบกำลังสอง มีความสัมพันธ์ดังสมการที่ 2.13

$$\frac{P_r}{P_t} = G_t G_r \left[\frac{\lambda}{4\pi d} \right]^2 \quad (2.13)$$

โดยที่

P_r คือ กำลังของสัญญาณที่รับได้ หน่วยเป็นเดซิเบล (dBm)

P_t คือ กำลังของสัญญาณที่ส่งออก หน่วยเป็นเดซิเบล (dBm)

G_t คือ อัตราการขยายของสายอากาศส่ง หน่วยเป็นเดซิเบล (dBi)

G_r คือ อัตราการขยายของสายอากาศรับ หน่วยเป็นเดซิเบล (dBi)

λ คือ ค่าความยาวคลื่น โดย $\lambda = \frac{c}{f}$

โดยที่ c คือความเร็วแสง (3×10^8 m/s)

f คือ ความถี่ของสัญญาณ หน่วยเป็นเฮิร์ต (Hz)

d คือ ระยะห่างระหว่างภาคส่งและภาครับ มีหน่วยเป็นเมตร (m)

ค่าการสูญเสียกำลังของสัญญาณในอากาศโดยทั่วไปคิดค่าในหน่วยเดซิเบล (dB) จะได้
ดังสมการที่ 2.14

$$L_{dB} = 10 \log(P_t) - 10 \log(P_r) \quad (2.14)$$

ในการส่งข้อมูลแบบไร้สาย ค่าความเข้มของสัญญาณจะถูกลดทอนไปตามระยะทาง
เมื่อรับสัญญาณเข้ามาต่ำและระยะทางระหว่างเครื่องรับ และเครื่องส่งเพิ่มขึ้น ถ้าไม่มีสิ่งกีดขวางใน
สภาพแวดล้อมจะมีเพียงการลดทอนในอากาศ ดังสมการที่ 2.15 หากถูกลดทอนจากสิ่งกีดขวางใน
สภาพแวดล้อม (Path Loss Exponent) จะได้ดังสมการที่ 2.16 ซึ่งค่าของ Path Loss Exponent
ในสภาพแวดล้อมต่าง ๆ แสดงดังตารางที่ 2.3

$$L_p = d^2 f^2 \quad (2.15)$$

โดยที่

L_p คือ การลดทอน หน่วยเป็นเดซิเบล (dB)

d คือ ระยะทาง มีหน่วยเป็นเมตร (m)

f คือ ความถี่ หน่วยเป็นเฮิร์ต (Hz)

$$P_r = P_t K \left(\frac{d_r}{d}\right)^\gamma \quad (2.16)$$

โดยที่

K คือ ค่าคงที่ Factor ของ $\frac{P_r(d_r)}{P_t}$

d_r คือ ระยะอ้างอิง

γ คือ Path Loss Exponent

ตารางที่ 2.3 Path Loss Exponent [24]

Environment	Path Loss Exponent, n
Free Space	2
Urban area cellular	2.7 - 3.5
Shadowed urban cellular	3 - 5
In building line-of-sight	1.6 - 1.8
Obstructed in building	4 - 6
Obstructed in factories	2 - 3

2.13.2 RSSI

เป็นการวัดความแรงหรือความเข้มของสัญญาณตัวรับ(พลังงานเป็นตัวสำคัญ) หน่วยเป็นเดซิเบล (dBm) ยิ่งติดลบน้อยยิ่งดี

- 1) สัญญาณมีความแรงระดับ -50 dBm ลงมา แสดงว่าเราอยู่ใกล้เสาส่งสัญญาณ มีสัญญาณแรงมาก
- 2) สัญญาณมีความแรงระดับ -60 ถึง -75 dBm. แสดงว่าสัญญาณแรง
- 3) สัญญาณมีความแรงระดับที่ตัวเลขค่า เพิ่มขึ้นไปเรื่อย ๆ คุณภาพความแรงของสัญญาณก็จะลดลง จนถึงประมาณ -110 dBm เป็นต้นไป หมายความว่าคุณภาพสัญญาณต่ำมาก

เมื่อวัดประมาณค่า RSSI จากนั้นจะถูกแปลงเป็นระยะทาง โดยมีความสัมพันธ์ระหว่างระยะทางกับกำลังงานตัวรับ ดังสมการที่ 2.5

$$P_r(d)[dBm] = P_r(1m) - 10n \times \log_{10}(d) \quad (2.17)$$

โดยที่

$P_r(d)$ คือ กำลังงาน RSSI ที่รับได้ระยะทาง d ในหน่วย dBm

$P_r(1m)$ คือ กำลังงาน RSSI ที่รับได้ใน 1 เมตร

n คือ Path Loss Exponent

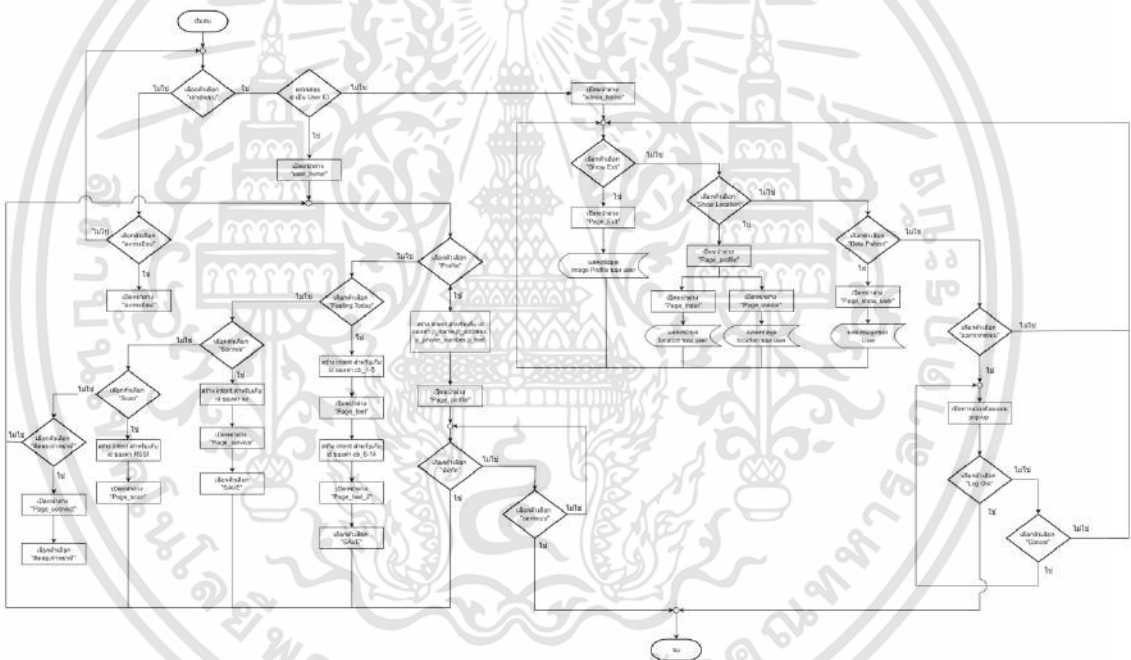
บทที่ 3

การออกแบบและการจัดทำปฏิญญานิพนธ์

3.1 การออกแบบ

3.1.1 ออกแบบการทำงานโดยรวมของแอปพลิเคชัน

ออกแบบการทำงานของแอปพลิเคชัน โดยแบ่งการทำงานออกเป็นสองส่วนหลักๆ คือ ส่วนของผู้ใช้งาน (ผู้ป่วย) และส่วนของผู้ดูแล (บุคลากรทางการแพทย์) โดยภาพรวมการทำงานของแอปพลิเคชัน แสดงดังรูปที่ 3.1

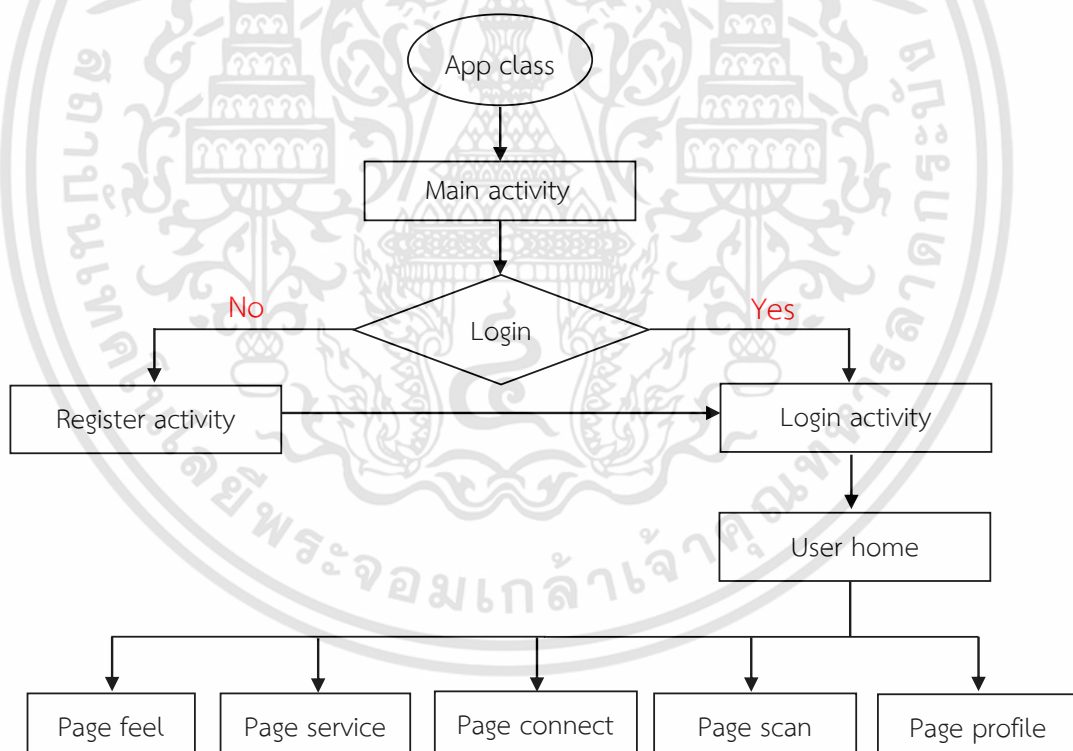


รูปที่ 3.1 ภาพรวมของการทำงานของแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.1 การทำงานในส่วนของผู้ใช้งาน (ผู้ป่วย)

สำหรับการทำงานในส่วนผู้ใช้งาน (ผู้ป่วย) เมื่อเปิดใช้งานแอปพลิเคชัน จะเริ่มค้นหาสัญญาณบิคอน เมื่อกดปุ่มสแกน แล้วแสดงตำแหน่งที่พบในหน้าสแกน เมื่อตรวจสอบแล้วตรงกับตำแหน่งที่เราอยู่ จะให้กดยืนยันตำแหน่งอีกครั้ง ในกรณีที่ไม่มีพบสัญญาณของตัวบิคอนก็ไม่สามารถยืนยันตำแหน่งได้ หลังจากที่ยืนยันตำแหน่งแล้วผู้ใช้งานสามารถออกจากแอปพลิเคชัน และมีการส่งค่า RSS และค่า Distance ของผู้ใช้งานเข้าฐานข้อมูลตลอดเวลา จนกว่าจะมีการออกนอกพื้นที่ที่กำหนดหรือหยุดการเชื่อมต่อสัญญาณจากตัวบิคอน หรือตรวจพบเจอสัญญาณบิคอนที่อยู่บริเวณทางออก จะมีการแจ้งเตือนให้ผู้ใช้งานเข้ากลับสู่พื้นที่กักตัว พร้อมทั้งมีฟังก์ชันเสริมที่อำนวยความสะดวกเพื่อลดภาระหน้าที่แก่ผู้ดูแล (บุคลากรทางการแพทย์) ได้แก่ หน้ากรอกรายการรายวัน หน้าสวัสดีการ และหน้าติดต่อเจ้าหน้าที่ โดยจะแสดงภาพรวมของระบบ ดังรูปที่ 3.2



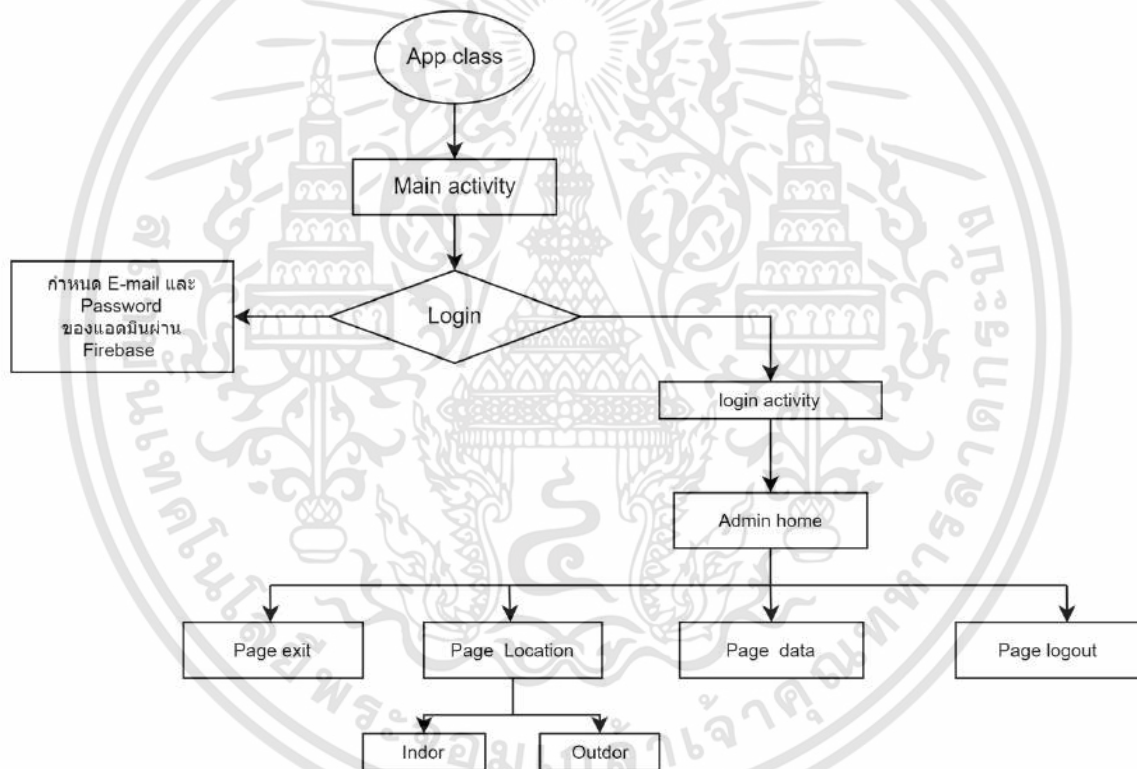
รูปที่ 3.2 บล็อกไดอะแกรมแสดงการทำงานโดยรวมในส่วนของผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.2 การทำงานในส่วนของผู้ดูแล (บุคลากรทางการแพทย์)

สำหรับการทำงานในส่วนผู้ดูแล (บุคลากรทางการแพทย์)

เมื่อออกแบบระบบให้สามารถแจ้งเตือนไปยังผู้ดูแล (บุคลากรทางการแพทย์) หากมีบุคคลออกนอกพื้นที่ที่กำหนดโดยจะใช้การตรวจสอบสัญญาณบีคอนที่ได้รับแล้วนำมาวิเคราะห์ให้ตรวจจับว่าถ้ามีค่า distance ตามที่กำหนดจะให้แจ้งเตือนไปยังทางผู้ดูแลได้รับทราบ โดยในส่วนของผู้ดูแล (บุคลากรทางการแพทย์) จะประกอบไปด้วยหน้าต่าง ๆ ได้แก่ หน้าดูตำแหน่งของผู้ป่วย หน้าดูข้อมูลของผู้ป่วยที่กรอกอาการรายวัน และหน้าออกจากระบบ ดังรูปภาพที่ 3.2 ภาพรวมการทำงาน ในส่วนผู้ดูแล (บุคลากรทางการแพทย์) แสดงดังรูป 3.3



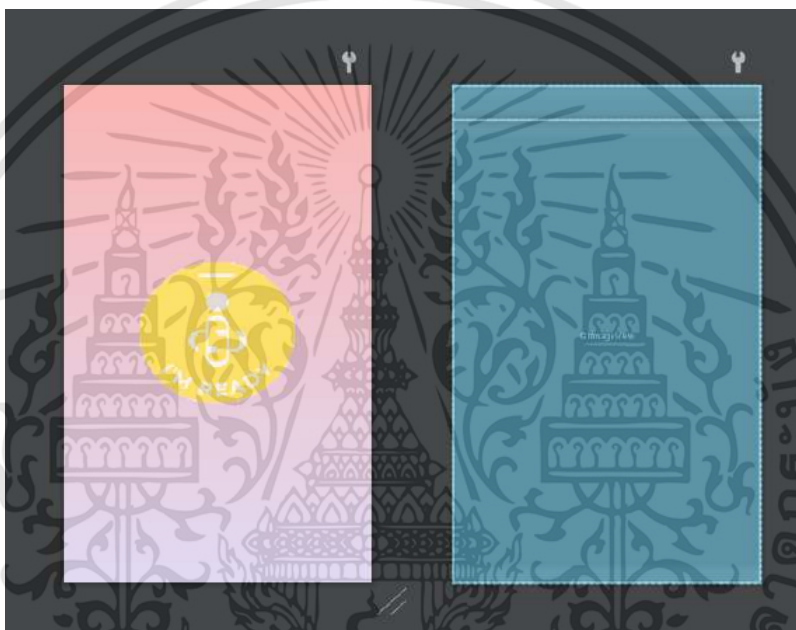
รูปที่ 3.3 ภาพรวมการทำงานในส่วนผู้ดูแล (บุคลากรทางการแพทย์)

3.1.2 ออกแบบหน้าต่างการทำงานของแอปพลิเคชัน

แต่ละหน้าต่างออกแบบและเขียนโปรแกรมสำหรับแอปพลิเคชันมี 2 ส่วน ได้แก่ ส่วนของผู้ป่วย (User) และส่วนของแอดมิน (Admin) การทำงานดังนี้

3.1.2.1 ออกแบบการทำงานของแอปพลิเคชันในส่วนของผู้ป่วย (User)

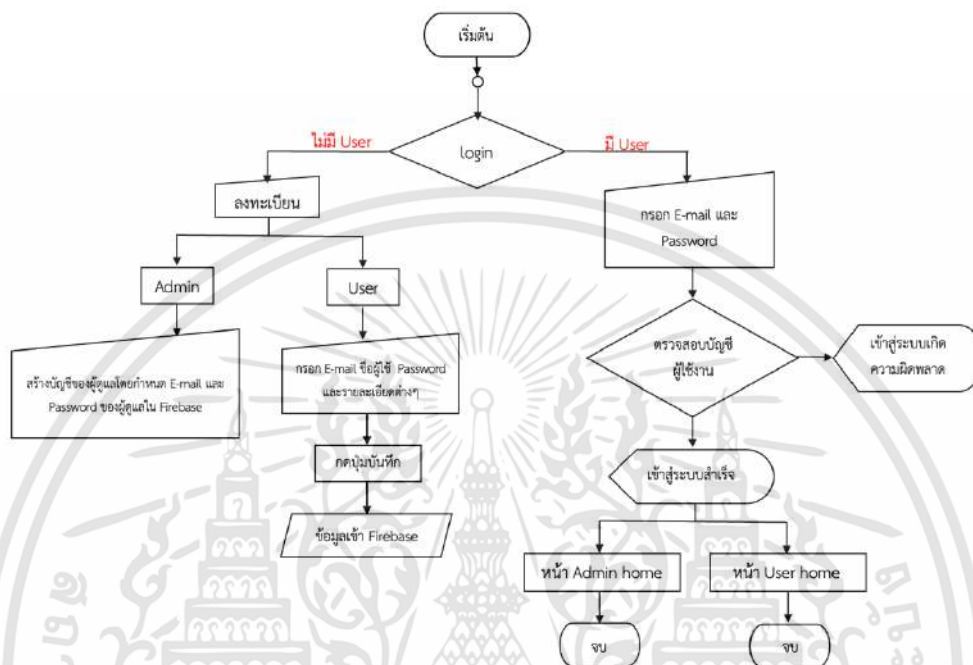
1) หน้า Splash screen เป็นหน้าแรกของแอปพลิเคชัน และแสดงโลโก้แอปพลิเคชัน ในส่วนนี้ยังไม่มีการทำงานของระบบ การจัดวางโครงสร้างของหน้าดังรูปที่ 3.1



รูปที่ 3.4 การจัดวางโครงสร้างหน้า Splash screen

2) หน้า Login หลังจากหน้าแสดงโลโก้ของแอปพลิเคชัน (Splash screen) หน้าถัดไปคือ หน้า Login เพื่อให้ผู้ใช้งานเข้าสู่ระบบ ถ้าหากยังไม่มีบัญชีผู้ใช้งานให้กดปุ่มลงทะเบียนเพื่อเข้าสู่ระบบ โดยมีลำดับการทำงานตามผังงานดังรูปที่ 3.4 และจัดวางโครงสร้างดังรูปที่ 3.5 ผู้ใช้งานจะ login ด้วยอีเมลกับรหัสผ่าน ใช้ Firebase Authentication ในการจัดการสามารถเปิดใช้งาน Authentication Sign-in method ได้หลังจากสร้าง Firebase Project หรือใช้งาน Project เดิมที่มีอยู่แล้ว ในหน้า Console ให้ไปที่ Authentication → Sign-in method แล้วกำหนดเปิดใช้งานการ Sign in ซึ่งเลือกเปิดใช้งาน Email/Password ดังรูปที่ 3.6 โดยตรวจสอบข้อมูลอีเมลผู้ใช้งานได้ใน Firebase Console ดังรูปที่ 3.7 หากตรงกันระบบ

จะสามารถเข้าสู่หน้าต่อไปได้และแสดง Toast “เข้าสู่ระบบสำเร็จ” หาก E-mail และ Password ไม่ตรงกับข้อมูลที่อยู่ใน Firebase ดังรูปที่จะแสดง Toast “เข้าสู่ระบบเกิดความผิดพลาด”

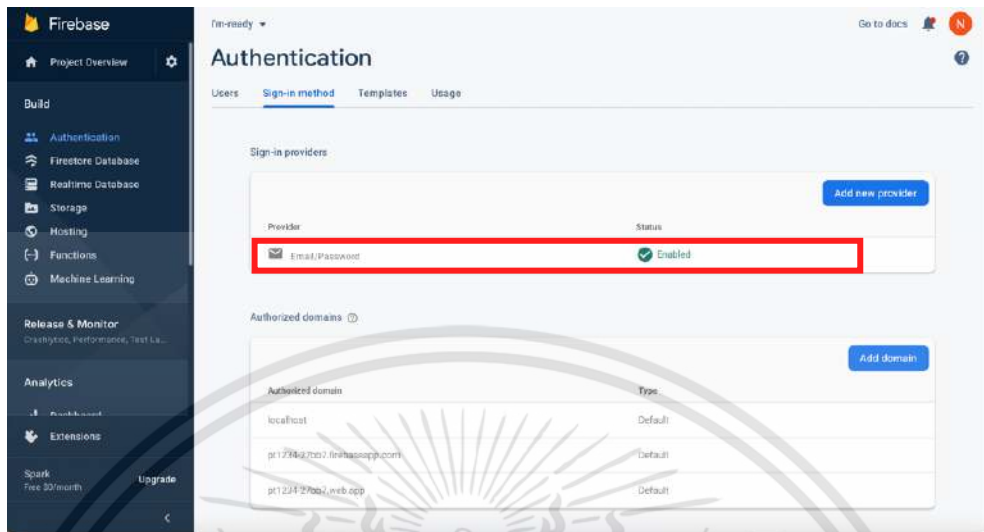


รูปที่ 3.5 ลำดับการทำงานของหน้าล็อกอินและลงทะเบียน

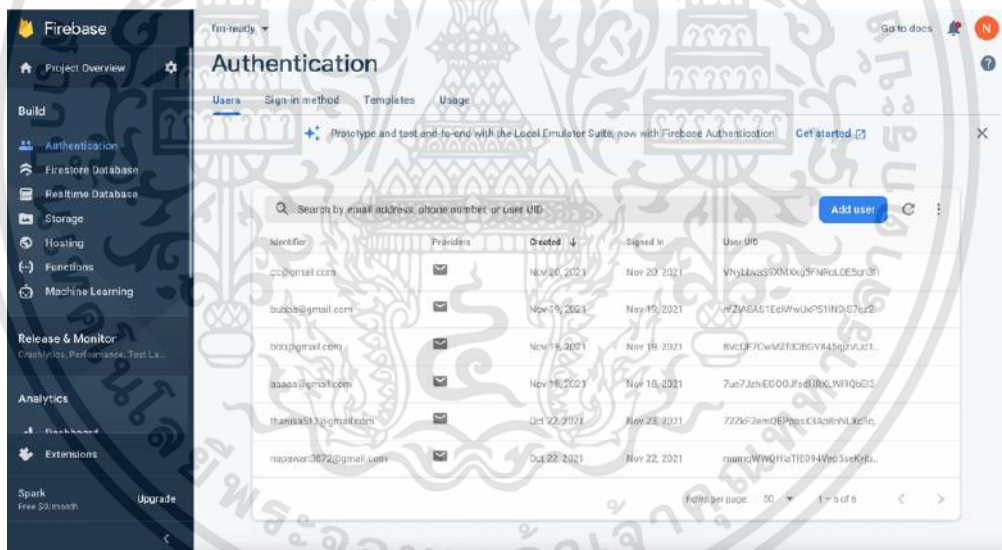


รูปที่ 3.6 จักรวาลโครงสร้างหน้าล็อกอิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 เปิดใช้งาน Sign in เป็น Email/Password



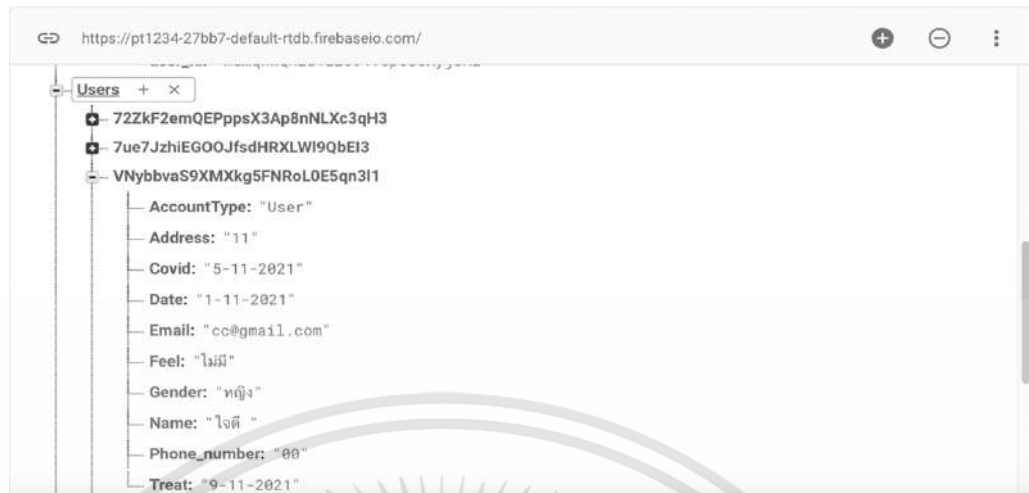
รูปที่ 3.8 ตรวจสอบข้อมูล E-mail ผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) หน้า Register ออกแบบหน้าลงทะเบียนสำหรับผู้ที่ยังไม่มีบัญชี โดยสมัครด้วย E-mail โดยใช้ Edit Text สำหรับใส่ข้อความหรือแก้ไขข้อความเพื่อเป็นช่องกรอกรายละเอียดที่ระบุไว้และใช้ Button สำหรับเป็นปุ่มลงทะเบียน เมื่อลงทะเบียนเสร็จสิ้นให้กดปุ่มบันทึก ซึ่งข้อมูลที่ลงทะเบียนจะถูกส่งมายังระบบฐานข้อมูล Firebase ที่ชื่อฟิลด์ว่า Users ดังรูปที่ 3.10 จากนั้นจะกลับมายังหน้าล็อกอินใหม่เพื่อให้กรอก E-mail และ Password ระบบจะนำ E-mail และ Password ที่กรอกมาเทียบกับข้อมูลที่ตอนลงทะเบียนเพื่อเทียบ E-mail และ Password ว่าตรงกับข้อมูลที่อยู่ใน Firebase หรือไม่ โดยมีลำดับการทำงานตามผังงานดังรูปที่ 3.4 โดยจัดวางโครงสร้างดังรูปที่ 3.9

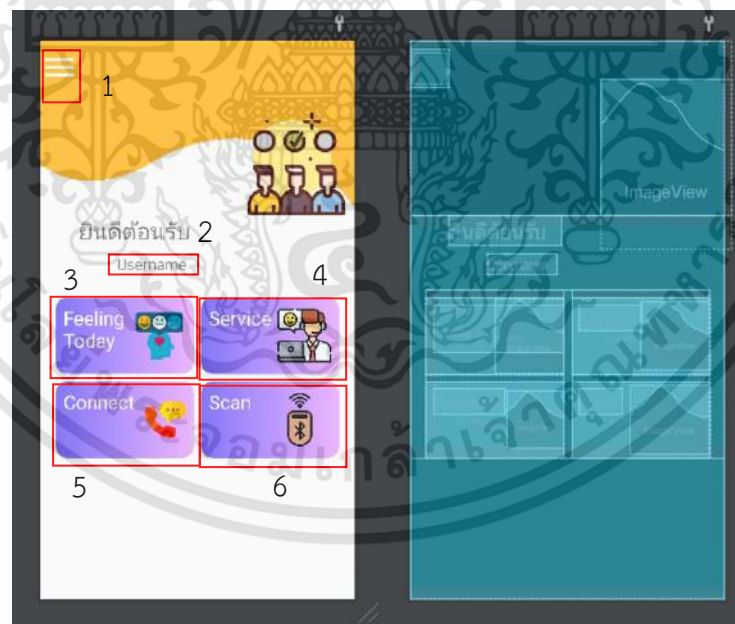
รูปที่ 3.9 จัดวางโครงสร้างหน้าลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 ข้อมูลผู้ป่วย (User) ในฐานข้อมูล

4) หน้า User Home เมื่อลงทะเบียนมีบัญชีผู้ใช้งาน และเข้าสู่ระบบเรียบร้อยแล้ว จะเข้าสู่หน้า User Home ดังรูปที่ 3.7 หน้านี้จะแสดงเมนูต่าง ๆ ของแอปพลิเคชัน โดยแต่ละเมนูแสดงปุ่ม Profile ชื่อผู้ใช้งาน, ปุ่มกรอกอาการรายวัน, ปุ่มสวัสดีการ, ปุ่มติดต่อเจ้าหน้าที่ และปุ่มสแกน เมื่อมีการแสดงโดยจัดวางโครงสร้างดังรูปที่ 3.11

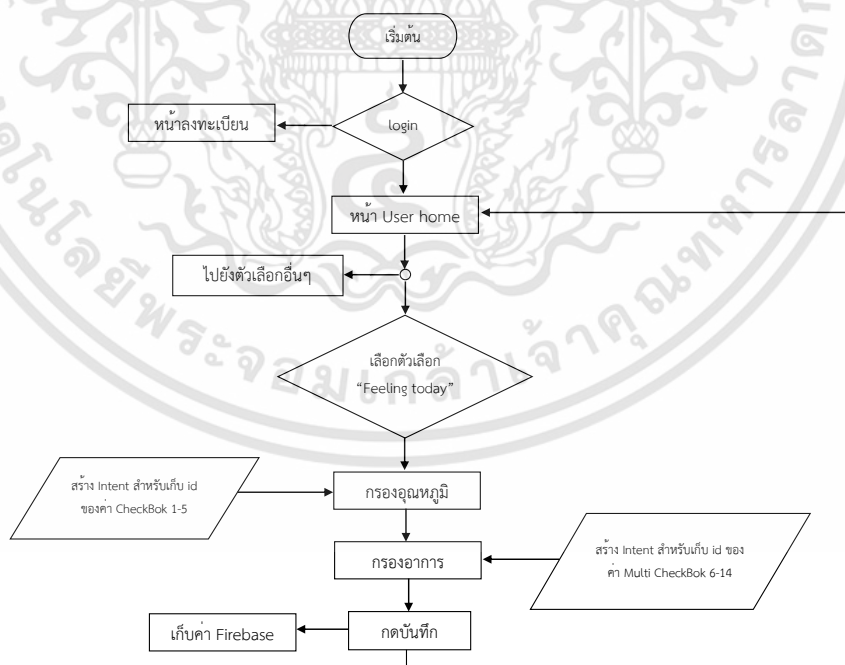


รูปที่ 3.11 โครงสร้างหน้า User home

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

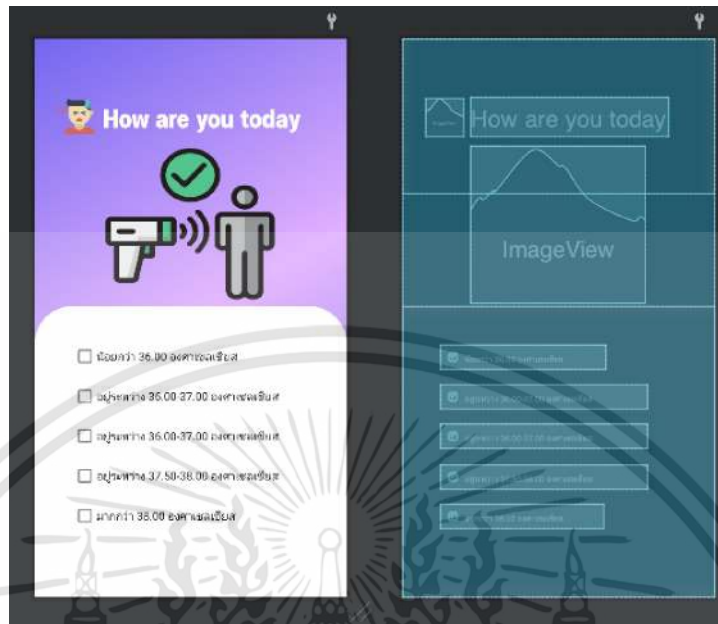
- หมายเหตุ
1. ปุ่มของหน้า Profile
 2. ส่วนที่แสดงชื่อของ User
 3. ปุ่มของหน้ากรอกอาการรายวัน
 4. ปุ่มของหน้าสวัสดิการ
 5. ปุ่มติดต่อเจ้าหน้าที่
 6. ปุ่มของหน้าสแกนหาตัวปีคอนและระบุตำแหน่ง

5) หน้ากรอกอาการรายวัน ออกแบบหน้านี้เพื่ออำนวยความสะดวกแก่ผู้ใช้งาน (ผู้ป่วย) และผู้ดูแล (บุคลากรทางการแพทย์) โดยหน้านี้จะใช้คำสั่ง 2 คำสั่ง คือ ส่วนที่กรอกอุณหภูมิร่างกาย จะใช้คำสั่ง CheckBox โดยจะเลือกได้แค่ 1 ตัวเลือกเท่านั้น การสร้างคำสั่งจะใช้ตรรกศาสตร์มาประยุกต์ใช้ดังรูปที่ 3.15 โดยจะแสดงคำสั่งดังรูปที่ ส่วนกรอกอาการต่าง ๆ จะใช้เป็นคำสั่ง Multi CheckBox สามารถเลือกได้มากกว่า 1 ตัวเลือก โดยจะแสดงคำสั่งดังรูปที่ เมื่อทำการกรอกอาการรายวันเรียบร้อยแล้วจะมีการบันทึกไปยัง Firebase เพื่อนำมาแสดงผลในการเรียกดูข้อมูลในแอปพลิเคชันของผู้ดูแล (บุคลากรทางการแพทย์) โดยมีลำดับการทำงานตามผังงานดังรูปที่ 3.13 โดยจัดวางโครงสร้างดังรูปที่ 3.14 และ รูปที่ 3.15



รูปที่ 3.12 แพนผังการทำงานของหน้ากรอกอาการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



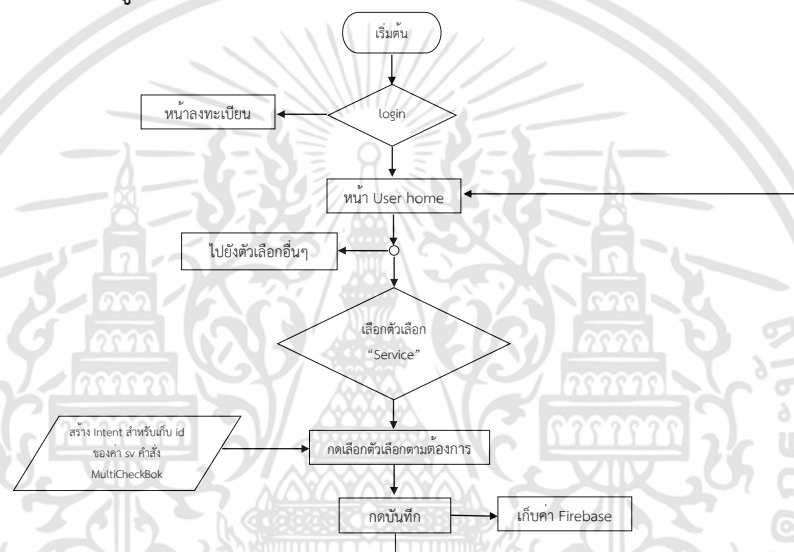
รูปที่ 3.13 โครงสร้างของหน้ากรอกอุณหภูมิ



รูปที่ 3.14 โครงสร้างของหน้ากรอกอาการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) หน้าสวัสดิการ ออกแบบหน้าสวัสดิการเพื่อให้ผู้ป่วยเข้าถึงสวัสดิการของรัฐบาลที่ช่วยเหลือผู้ป่วยโควิด-19 โดยจะรวบรวมข่าวของเครื่องใช้ที่จำเป็นที่รัฐบาลมีแจกและเมื่อเรากดสิ่งของที่อยากได้หรืออื่น ๆ จะเก็บข้อมูลไว้ที่ Firebase และจะแสดงผลไปยังหน้าข้อมูลผู้ป่วยของแอปพลิเคชันของผู้ดูแล เมื่อผู้ดูแลได้รับข้อมูลก็จะนำมาให้ผู้ป่วยอีกที่บริเวณกักตัวของแต่ละบุคคล โดยการเลือกสวัสดิการจะใช้คำสั่ง Multi CheckBok ให้เลือกตัวเลือกได้มากกว่า 1 ตัวเลือกและอื่น ๆ ใช้คำสั่ง Edittext โดยมีลำดับการทำงานตามผังงานดังรูปที่ 3.15 และจัดวางโครงสร้างดังรูปที่ 3.16



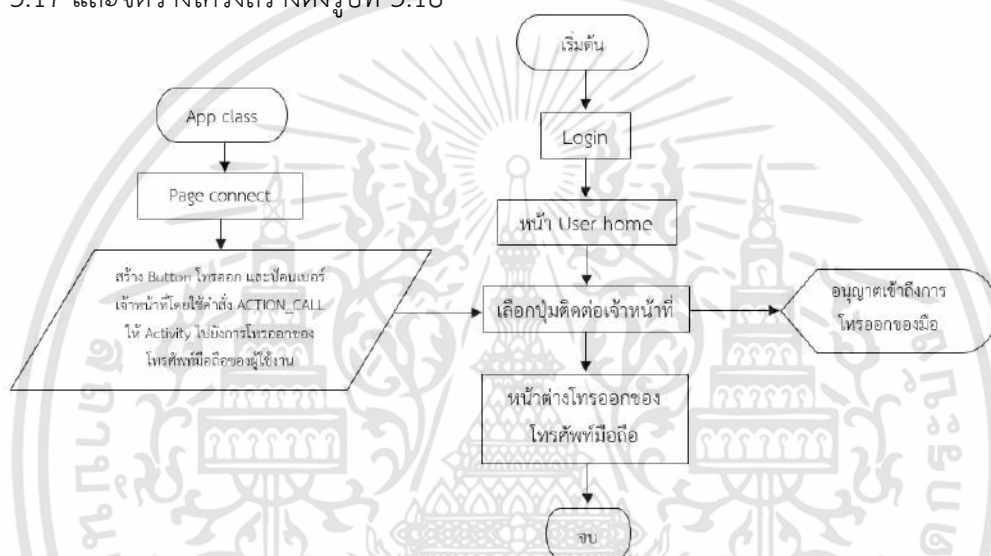
รูปที่ 3.15 ลำดับการทำงานของหน้าสวัสดิการ



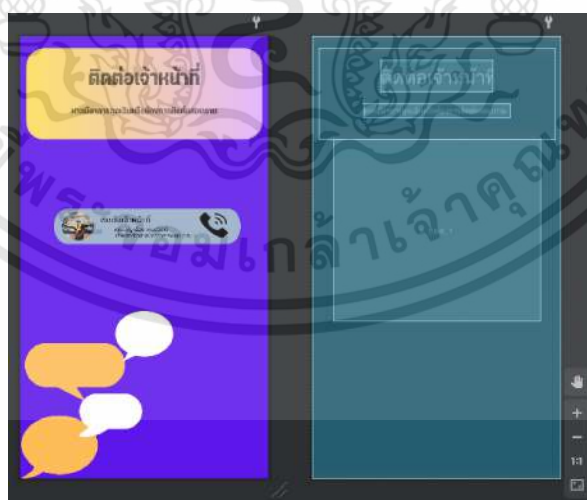
รูปที่ 3.16 จัดวางโครงสร้างหน้าสวัสดิการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) หน้าลงติดต่อเจ้าหน้าที่ ออกแบบหน้าติดต่อสำหรับการ
 ผู้ใช้งานต้องการติดต่อเจ้าหน้าที่โดยตรง เมื่อเข้ามายังหน้า User home แล้วให้กดปุ่มติดต่อ
 เจ้าหน้าที่ แล้วจะแสดงหน้าต่างที่มีปุ่ม ติดต่อเจ้าหน้าที่ เมื่อเรากดปุ่มนี้ จะไปยังหน้าต่างโทรออก
 ของโทรศัพท์มือถือของผู้ใช้งาน ปุ่มที่ใช้ Button สำหรับกดโทรออก และกำหนดเบอร์ของ
 เจ้าหน้าที่ โดยใช้คำสั่ง Intent intent = new Intent(Intent.ACTION_CALL, Uri.parse
 ("tel:" + " เบอร์เจ้าหน้าที่ ")); startActivity(intent); โดยมีลำดับการทำงานตามผังงานดังรูปที่
 3.17 และจัดวางโครงสร้างดังรูปที่ 3.18



รูปที่ 3.17 ลำดับการทำงานของหน้าติดต่อเจ้าหน้าที่



รูปที่ 3.18 จัดวางโครงสร้างหน้าติดต่อเจ้าหน้าที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

@Override
public void onClick(View view) {

    Intent intent = new Intent(Intent.ACTION_CALL, Uri.parse("tel:" + "0920654512"));
    startActivity(intent);
}
});

```

รูปที่ 3.19 ชุดคำสั่งหน้าต่างติดต่อเจ้าหน้าที่

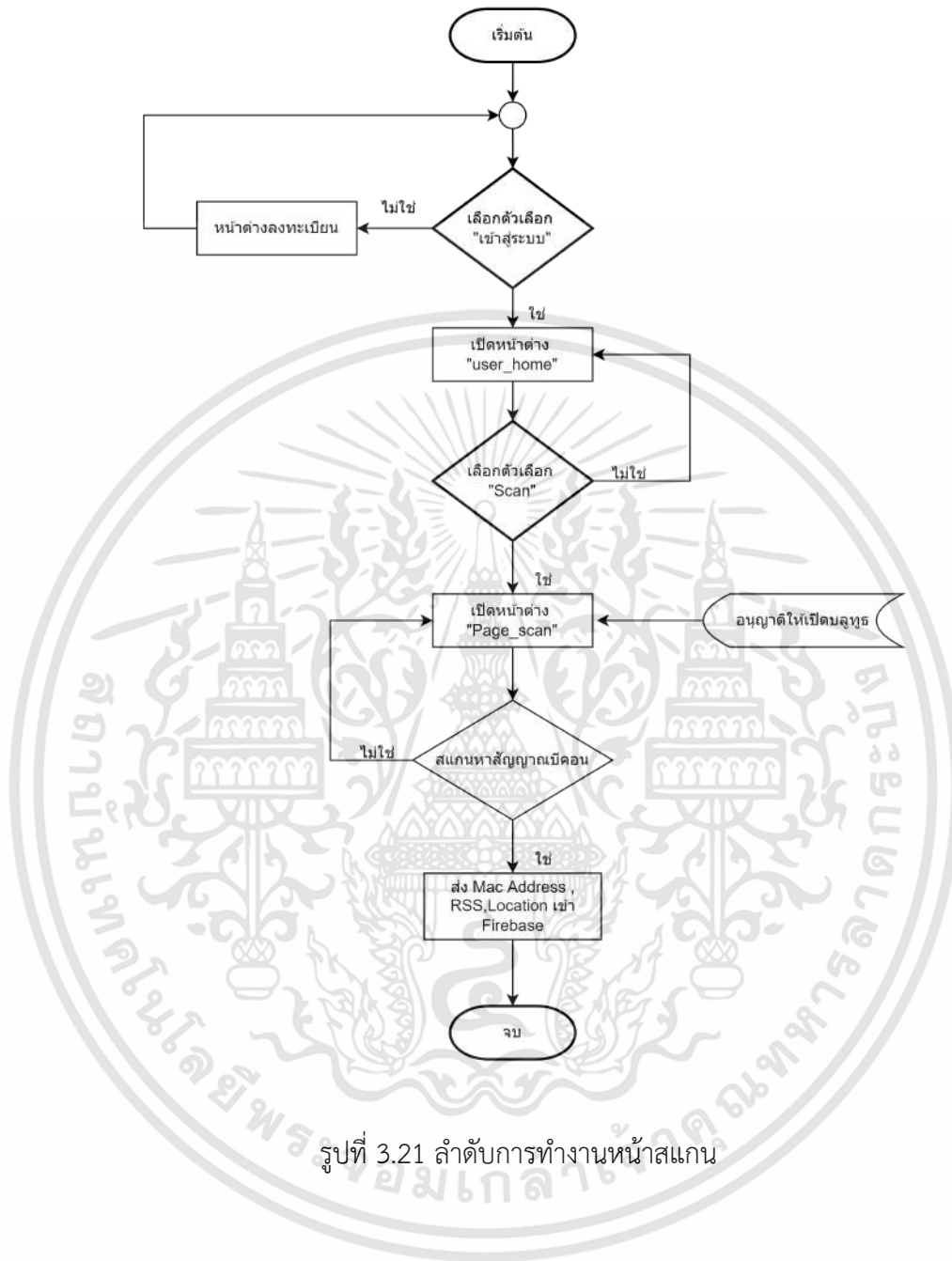
```

<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.STORAGE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.ACTIVITY_RECOGNITION" />
<uses-permission android:name="android.permission.CALL_PHONE" />

```

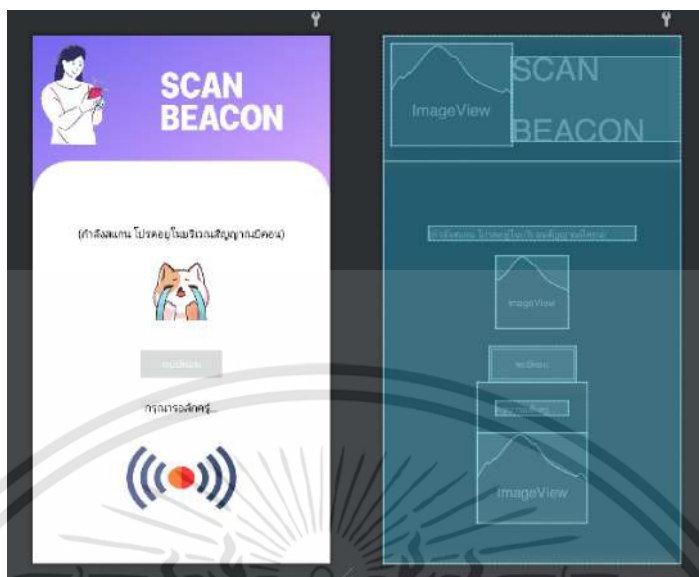
รูปที่ 3.20 คำสั่งหน้า Manifest เพื่ออนุญาตเข้าถึงการโทรออก

8) หน้าสแกน ออกแบบหน้าสแกนเพื่อค้นหาสัญญาณบลูทูธ จากตัวปีคอน เพื่อระบุตำแหน่งห้องหรือตำแหน่งภายในอาคาร โดยจะอธิบายการทำงานของปีคอน แบบละเอียดในหัวข้อที่ 3.1.3 การออกแบบการทำงานของปีคอนในแอปพลิเคชัน โดยหน้านี้จะมี การขออนุญาตจากโทรศัพท์มือถือให้เปิดใช้บลูทูธ ใช้ในคำสั่งอนุญาตการเข้าถึงในคลาส Manifest โดยการทำงานของหน้านี้จะประกอบไปด้วยคำสั่ง การสแกนหา Mac Address ของปีคอน, การดึงค่า RSS มาเปรียบเทียบเพื่อหาตัวปีคอนที่อยู่ใกล้ที่สุด เพื่อระบุตำแหน่งของผู้ใช้งาน และการทำงานแบบ BackgroundService โดยมีลำดับการทำงานตามผังงานดังรูปที่ 3.21 และ ที่จัดวาง โครงสร้างดังรูป 3.22 และรูปที่ 3.23



รูปที่ 3.21 ลำดับการทำงานหน้าสแกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



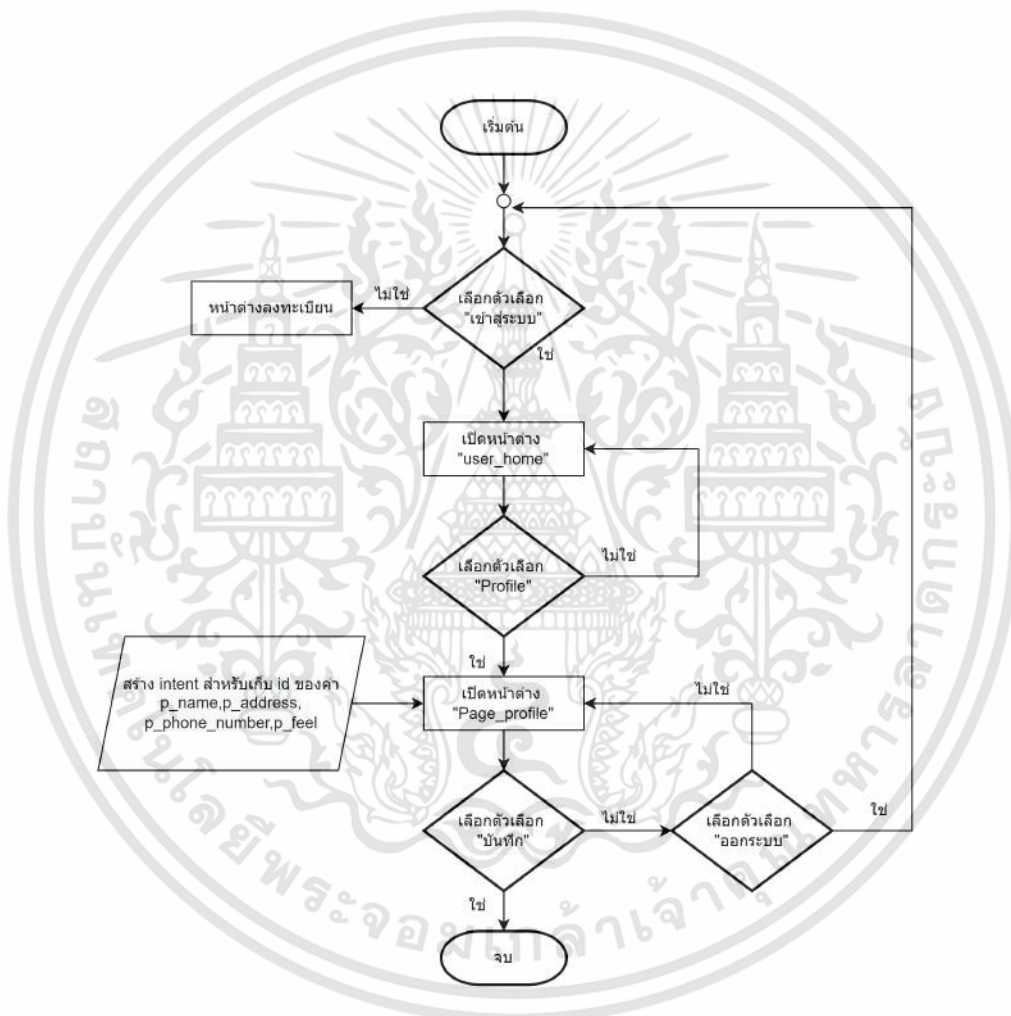
รูปที่ 3.22 จักรวาลโครงสร้างของหน้าสแกน



รูปที่ 3.23 จักรวาลโครงสร้างของหน้าสแกน

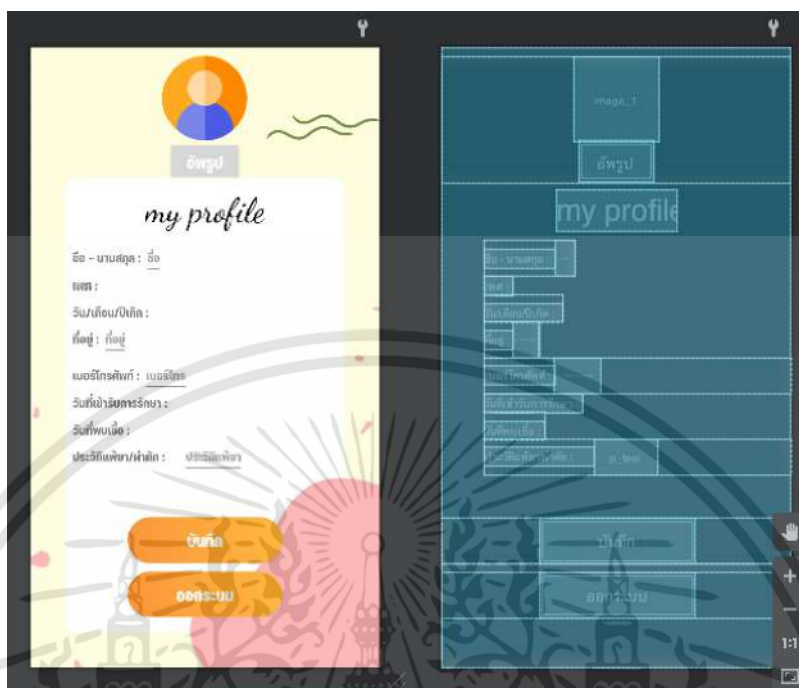
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) หน้าแก้ไข Profile ออกแบบหน้าแก้ไขเพื่อให้ผู้ใช้งานสามารถแก้ไขข้อมูลส่วนตัวได้ หากมีข้อผิดพลาดด้านข้อมูลการลงทะเบียนก็สามารถอัปเดตข้อมูลของตนเองได้ โดยใช้ใช้คำสั่ง Editttext ในการแก้ไข เมื่อแก้ไขเสร็จสิ้นก็ทำการกดปุ่มบันทึกข้อมูลและทำการส่งข้อมูลเข้า Firebase และหน้านี้ยังมีปุ่มออกจากระบบอีกด้วย โดยมีลำดับการทำงานตามผังงานดังรูปที่ 3.24 และจัดวางโครงสร้างดังรูปที่ 3.25



รูปที่ 3.24 ลำดับการทำงานหน้าแก้ไขโปรไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

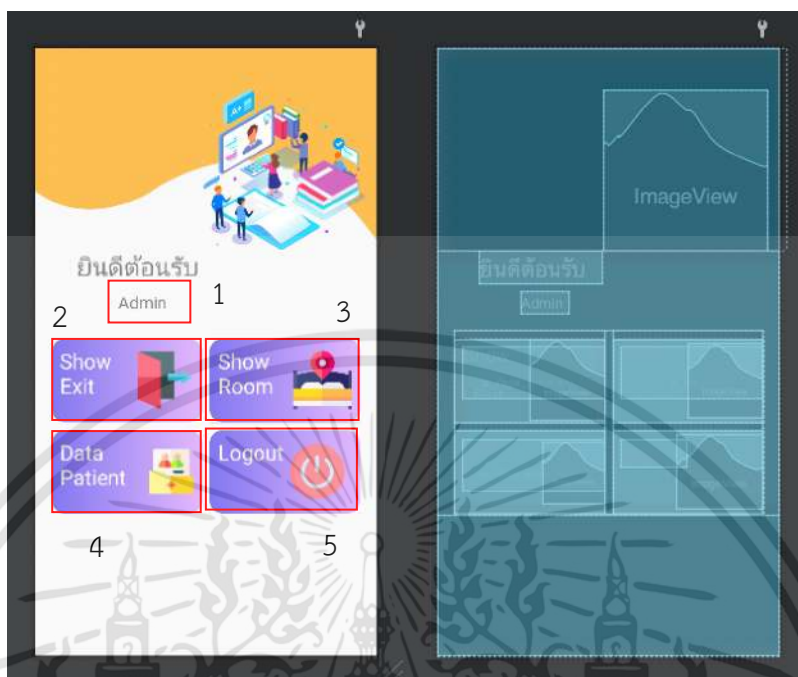


รูปที่ 3.25 จัตุวางโครงสร้างของหน้าโปรไฟล์

3.1.2.2 ออกแบบการทำงานของแอปพลิเคชันในส่วนของผู้ดูแล (Admin)

การทำงานในส่วนนี้จะแสดงการดึงข้อมูลมาแสดงในตัวแอปพลิเคชันเป็นหลัก อันดับแรกเราต้องกำหนด E-mail และ Password ของผู้ดูแล (Admin) ใน Firebase ให้เป็นประเภท Admin โดยจะประกอบด้วยหน้าต่าง ๆ ดังนี้

1) หน้า Admin Home เมื่อลงทะเบียนมีบัญชีผู้ใช้งาน และเข้าสู่ระบบเรียบร้อยแล้ว จะเข้าสู่หน้า Admin Home หน้านี้จะแสดงเมนูต่าง ๆ ของแอปพลิเคชันโดยแต่ละเมนูแสดงปุ่มแสดงรายชื่อคนออกจากพื้นที่, ปุ่มแสดงตำแหน่งของของผู้ใช้งาน, ปุ่มแสดงข้อมูลของผู้ป่วย และปุ่มออกจากระบบ โดยจัดวางโครงสร้างดังรูปที่ 3.26

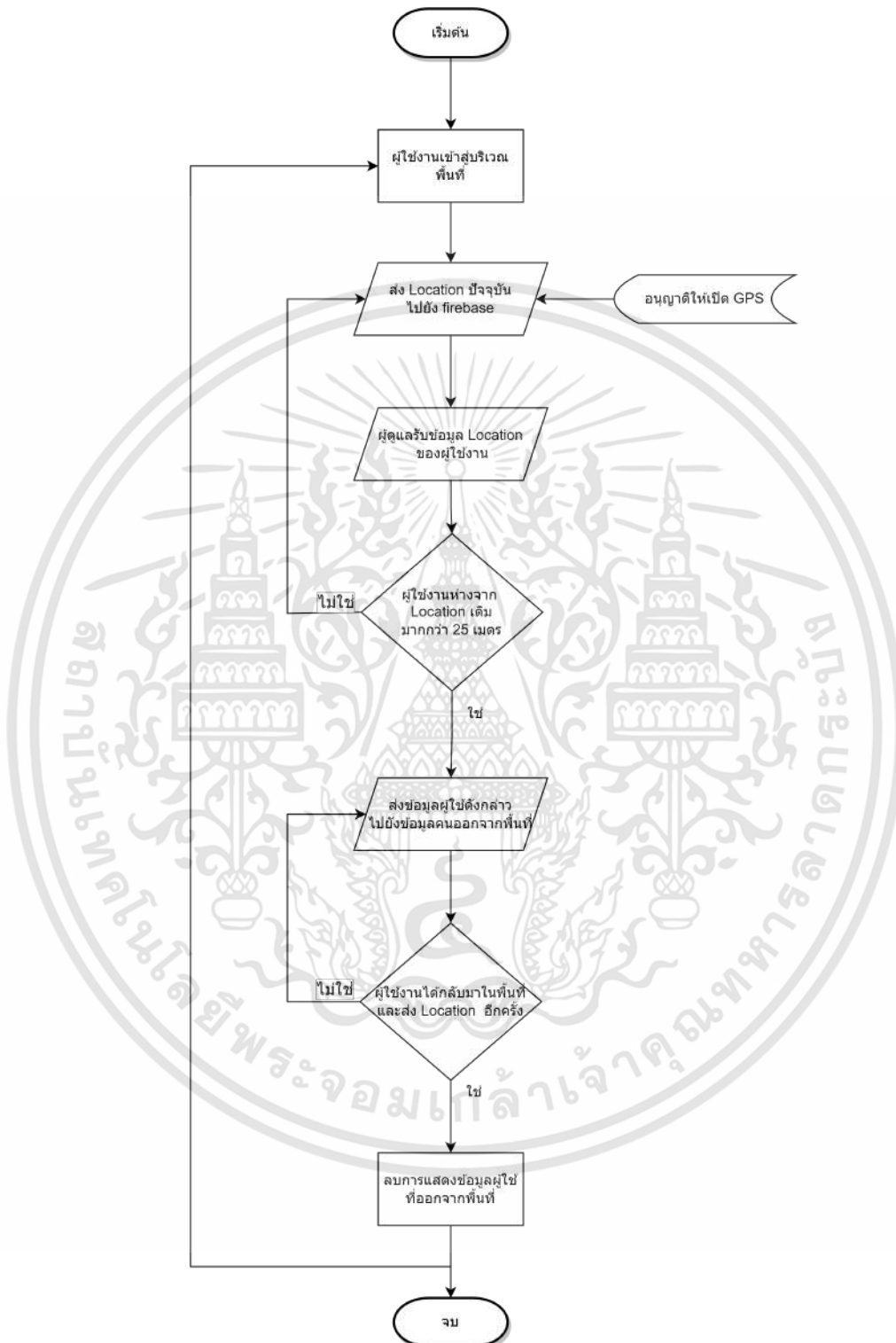


รูปที่ 3.26 โครงสร้างหน้า Admin home

หมายเหตุ 1) แสดงชื่อของผู้ดูแล

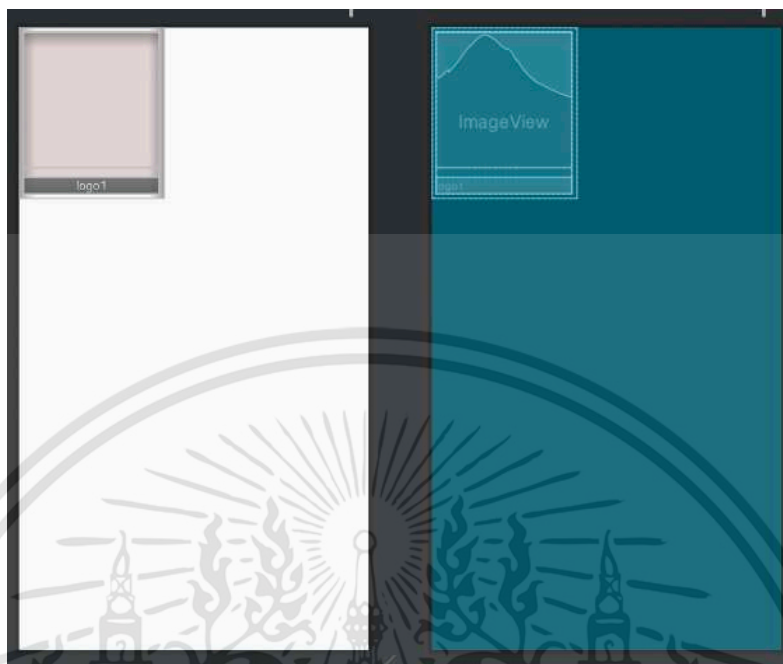
- 2) ปุ่มของหน้าแสดงรายชื่อคนออกจากพื้นที่
- 3) ปุ่มของหน้าแสดงตำแหน่งของผู้ป่วย
- 4) ปุ่มของหน้าแสดงข้อมูลผู้ป่วย
- 5) ปุ่มออกจากระบบ

2) หน้าแสดงรายชื่อคนที่ออกจากพื้นที่ ออกแบบหน้านี้เพื่อดูรายชื่อคนที่ออกจากพื้นที่ ถ้ามีผู้ป่วย (User) ออกนอกพื้นที่ที่เรากำหนด จะแสดงรายชื่อไปยังหน้านี้เพื่อให้ผู้ดูแล (Admin) ตรวจสอบ ถ้าผู้ป่วยกลับเข้าพื้นที่แล้วทำการแสกนหาบุคคลใหม่ รายชื่อของผู้ใช้งานก็จะถูกลบโดยอัตโนมัติ โดยจะใช้คำสั่งการรับค่ามาจาก Firebase โดยข้อมูลจะถูกดึงมาจาก child ที่ชื่อว่า exit1 นำมาแสดง โดยมีลำดับการทำงานตามผังงานดังรูปที่ 3.27 และจัดวางโครงสร้างดังรูปที่ 3.28



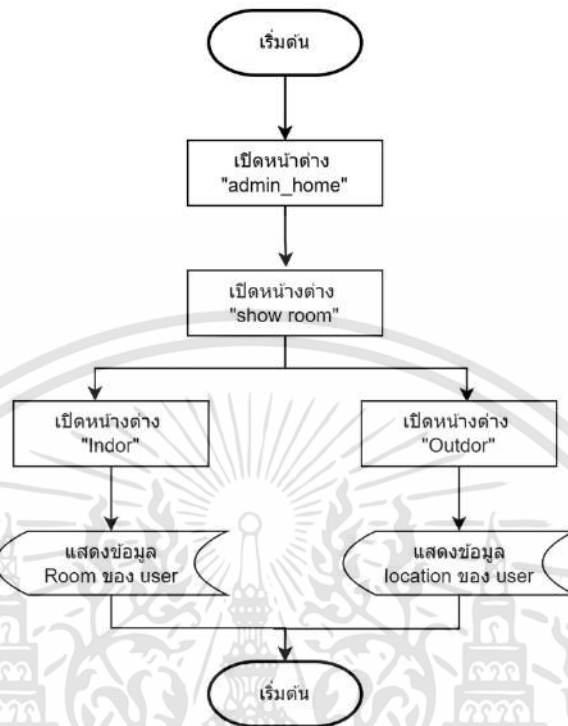
รูปที่ 3.27 ลำดับการทำงานของหน้าแสดงรายชื่อคนออกจากพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

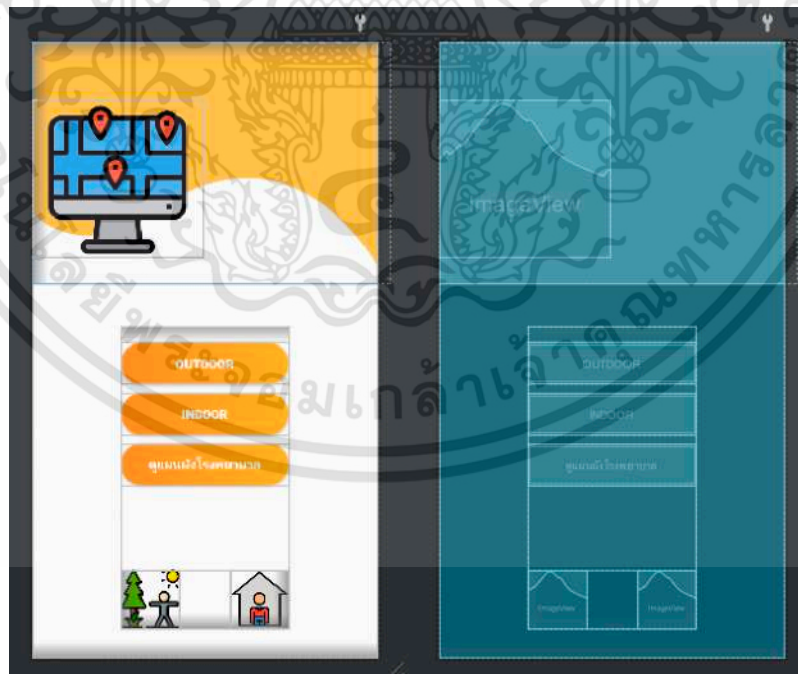


รูปที่ 3.28 โครงสร้างหน้าแสดงรายชื่อคนออกจากพื้นที่

3) หน้าตำแหน่งของผู้ป่วย (User) ออกแบบหน้าเพื่อให้ผู้ดูแล (Admin) สามารถดูตำแหน่งผู้ป่วยได้แบบเรียลไทม์และยังสามารถดูทั้งภายในอาคารและนอกอาคารได้ โดยส่วนที่ระบุตำแหน่งภายในอาคาร มาจากการรับสัญญาณบีกอนแต่ละตัว โดยแต่ละตัวจะระบุเลขห้องตามเลข Mac Address แล้วหมุดของผู้ป่วยจะแสดงตามหมายเลขห้องตามที่ได้กำหนดชื่อไว้ ถ้าผู้ป่วยออกนอกพื้นที่ก็จะแสดงหมุดที่บริเวณทางออก และถ้าผู้ป่วยกลับเข้าพื้นที่อีกครั้งก็จะแสดงหมุดตามหมายเลขห้องเหมือนเดิม ส่วนด้านภายนอกอาคารจะใช้ตัว Google Map เข้ามาทำงานร่วมด้วย เพื่อให้ผู้ดูแลสามารถตรวจสอบได้แม่นยำมากขึ้น โดยจะกำหนดเงื่อนไขให้ Google Map ก็ต่อเมื่อมีการรับสัญญาณจากบีกอนก่อน โดยมีลำดับการทำงานตามผังงานดังรูปที่ 3.29 และจัดวางโครงสร้างดังรูปที่ 3.30 , รูปที่ 3.31 และ รูปที่ 3.32

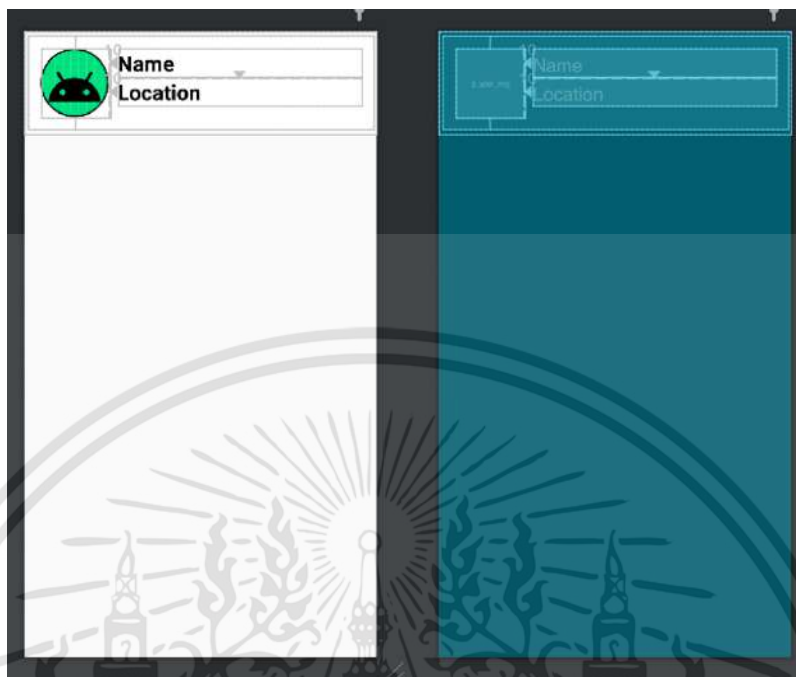


รูปที่ 3.29 ลำดับการทำงานของหน้าดูตำแหน่งของผู้ป่วย

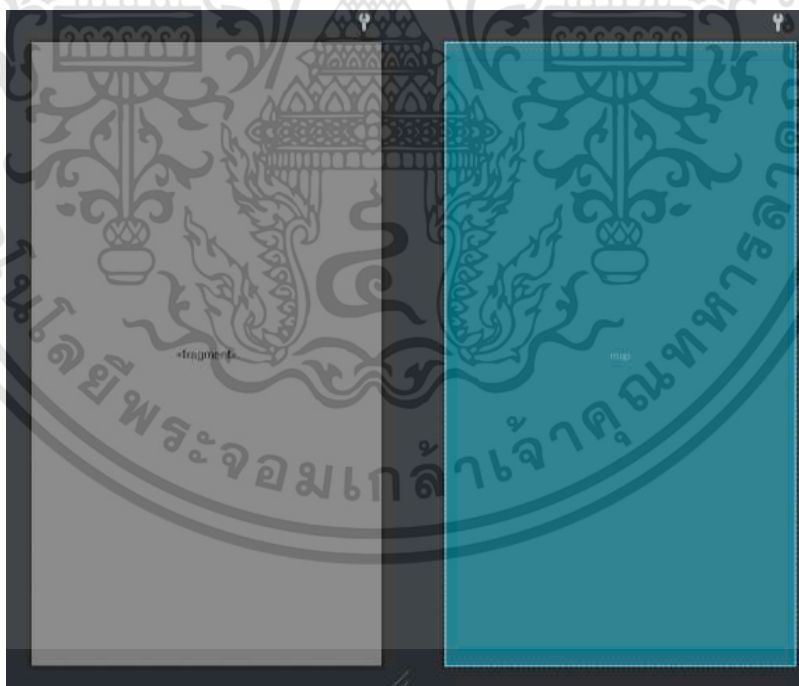


รูปที่ 3.30 จัดวางโครงสร้างหน้า home ของหน้าดูตำแหน่งผู้ป่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



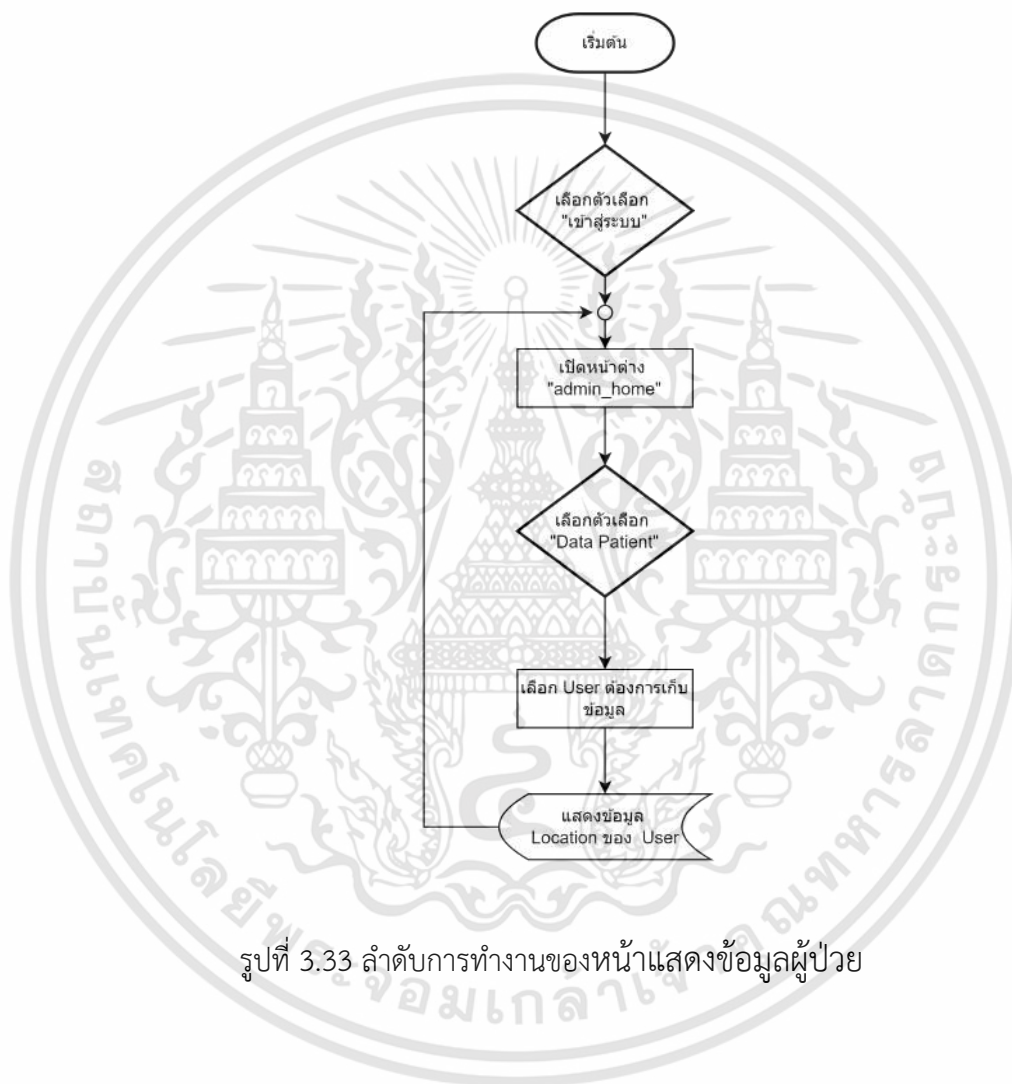
รูปที่ 3.31 จัดวางโครงสร้างหน้าดูตำแหน่งผู้ช่วยภายในห้อง (Indoor)



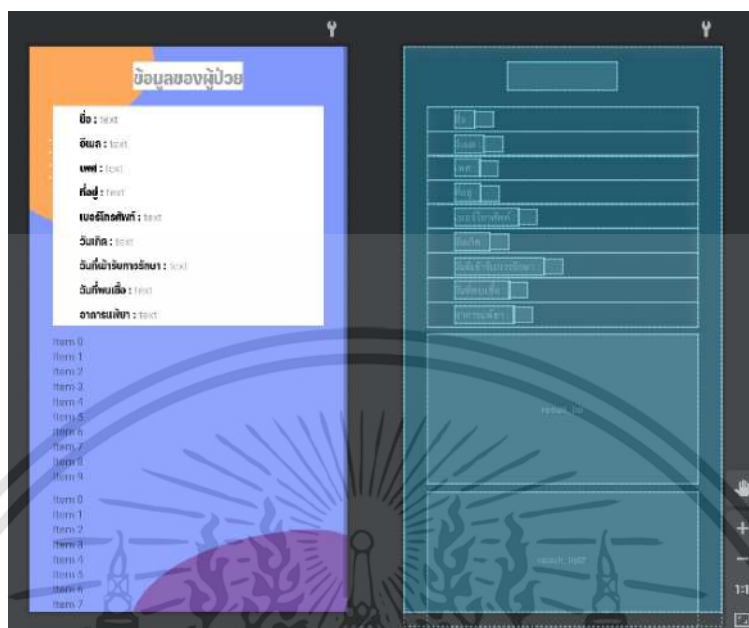
รูปที่ 3.32 จัดวางโครงสร้างหน้าดูตำแหน่งผู้ช่วยภายนอกห้อง (Outdoor)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) หน้าแสดงข้อมูลผู้ป่วย ออกแบบหน้านี้เพื่อรับข้อมูลที่มาจากปุ่มแก้ไข Profile , ปุ่มกรอกอาการรายวัน และปุ่มสวัสดิการ จะนำมาแสดงในหน้านี้ โดยจะดึงข้อมูลมาจาก child User , child Feel และ child service โดยใช้คำสั่ง getIntent(); โดยมีลำดับการทำงานตามผังงานดังรูปที่ 3.33 และจัดวางโครงสร้างดังรูปที่ 3.34



รูปที่ 3.33 ลำดับการทำงานของหน้าแสดงข้อมูลผู้ป่วย



รูปที่ 3.34 จัดวางโครงสร้างหน้าแสดงข้อมูลผู้ป่วย

3.1.3 การออกแบบการทำงานของปีคอนในแอปพลิเคชัน

การทำงานของปีคอนในแอปพลิเคชัน เขียนโปรแกรมให้แอปพลิเคชันสามารถเชื่อมต่อกับตัวปีคอน ทำการขออนุญาตเปิดบลูทูธก่อนโดยใช้เมธอด `startActivityForResult(enableBtIntent, REQUEST_ENABLE_BT)` และเพิ่มคำสั่งในหน้า Manifest ดังรูปที่ 3.35 จากนั้นจะรับค่าสัญญาณ (RSS) ไปประมวลผล เมื่อเรากดปุ่มแสดงเพื่อหาสัญญาณบลูทูธของตัวปีคอน เมื่อเจอสัญญาณของตัวปีคอนที่เรากำหนดไว้ (Mac Address) จากนั้นกำหนดเงื่อนไขว่า ถ้าเจอตัวที่ใกล้ที่สุดให้ส่งข้อมูลเข้า Firebase ได้แก่ ชื่อ Mac Address โดยตั้งชื่อ Mac Address ให้ตรงกับสถานที่อยู่ดังแสดงดังรูปที่ 3.36 โดยวิเคราะห์ผลจากการทดสอบหากำลังรับระหว่างปีคอนกับโทรศัพท์มือถือ ซึ่งผลการทดลองจะอยู่ในหัวข้อการจัดเก็บผลการทดลอง โดยจะออกแบบลำดับการทำงานตามแผนผังดังรูปที่ 3.37

```

<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-feature android:name="android.hardware.bluetooth_le" android:required="true"/>
<activity android:name=".Page_profile_2"/>
<service android:name=".Ble.BackgroundService"/>

```

รูปที่ 3.35 หน้า Manifest การขออนุญาตการเข้าถึง

เมทีอด Activity การทำงานของปีคอนในแอปพลิเคชัน

หมายเหตุ

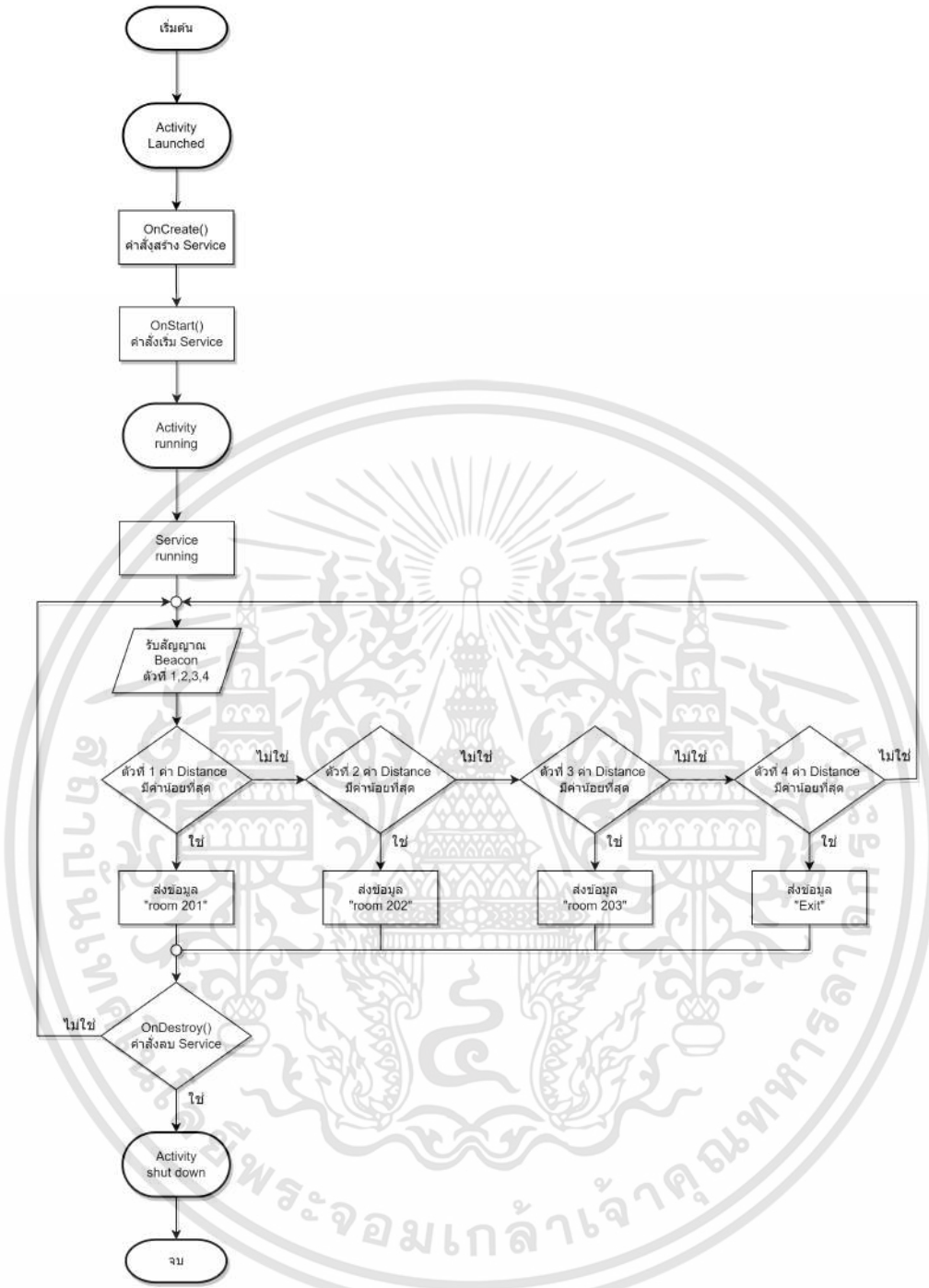
- 1) เมทีอดการทำงานของบลูทูธของโทรศัพท์มือถือ
- 2) เมทีอดการทำงานแบบ BackgroundService

```

if (addr.equals("DA:E8:55:21:A9:B6")){
    dataSnapshot.getRef().child("address1").setValue(addr);
    dataSnapshot.getRef().child("location1").setValue("201 room");
    dataSnapshot.getRef().child("rssi1").setValue(rssi);
    dataSnapshot.getRef().child("distance1").setValue(distance);
    dataSnapshot.getRef().child("user_id").setValue(s_user_id);
}

```

รูปที่ 3.36 กำหนดชื่อให้ Mac Address ของปีคอนแต่ละตัวและค่าที่ส่งเข้า Firebase

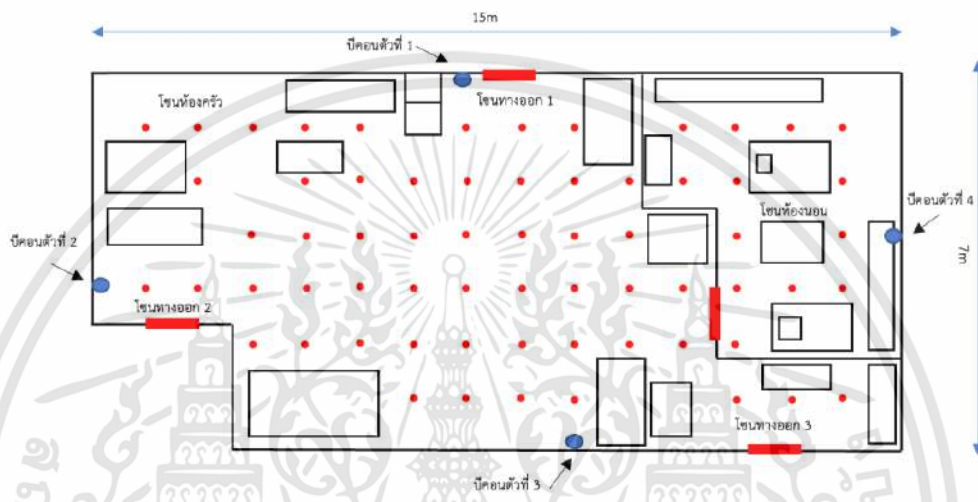


รูปที่ 3.37 ลำดับการทำงานของปีคอนในแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4 การออกแบบการวางตำแหน่งปีคอนในโรงพยาบาลสนาม

โครงการนี้ได้ออกแบบการติดตั้งปีคอน 4 ตัวโดยให้ทั้ง 4 ตัวมีสถานะแวดล้อมที่อยู่ภายในอาคารเหมือนกันแต่จะติดตั้งคนละมุม เพื่อให้มีความแตกต่างด้านกายภาพที่แตกต่างกันเล็กน้อยโดยมีการวางแผนผังบ้านดังรูปที่ 3.38



รูปที่ 3.38 แผนผังบ้านและตำแหน่งการวัด

หมายเหตุ

- 1) ปีคอนตัวที่ 1 ติดบริเวณบริเวณโถงทางเดินและประกอบไปด้วย ตู้เสื้อผ้า อุปกรณ์ไฟฟ้า โต๊ะ เก้าอี้ และผนังบาง
- 2) ปีคอนตัวที่ 2 ติดบริเวณห้องครัวที่ประกอบไปด้วย โต๊ะ เก้าอี้ เครื่องครัว และโถงทางเดิน
- 3) ปีคอนตัวที่ 3 ติดบริเวณโถงทางเดินอีกฝั่งตรงข้ามกับปีคอนตัวที่ 2 ประกอบไปด้วย โถงทางเดิน ตู้ และผนังบาง
- 4) ปีคอนตัวที่ 4 ติดบริเวณห้องนอน ประกอบไปด้วย เตียง 2 ตัว ประตู ตู้เสื้อผ้า และผนังบาง

จากการทำแบบจำลองการวัด และพิจารณาสถานะแวดล้อมภายในบ้านมีขนาดกว้าง 7 เมตร ยาว 15 เมตร ภายในบ้านประกอบไปด้วยห้อง 3 ห้อง ได้แก่ ห้องนอน 2 ห้อง และห้องครัว 1 ห้อง ซึ่งภายในบ้านมีโต๊ะ ตู้ และเตียงเป็นจำนวนมาก ดังที่แสดงในรูปที่ 3.38 ในแผนผังบ้านและตำแหน่งการวัด ได้กำหนดตำแหน่งในการวัดในแนวด้านกว้าง 6 ตำแหน่ง ด้านยาว 14

ตำแหน่ง แต่ละตำแหน่งในแนวด้านกว้างและแนวด้านยาวห่างกัน 1 m ตำแหน่งที่สามารถวัดได้มีทั้งหมด 64 ตำแหน่ง (เป็นตำแหน่งทางเดินที่ไม่มีสิ่งกีดขวางเพื่ออำนวยความสะดวกในการทำงานและความรวดเร็ว) และได้ติดตั้งบีมคอน BLE จำนวน 4 ตัวสูงจากพื้น 2 m อยู่ที่ผนังด้านหน้า ด้านซ้าย ด้านขวาและด้านหลังตามลำดับ

3.2 เครื่องมือที่ใช้ในการทดลอง

ในโครงการนี้ มีเครื่องมือและอุปกรณ์ ดังนี้

3.2.1 บีมคอน

บีมคอนเป็นอุปกรณ์ที่ใช้ปล่อยสัญญาณในลักษณะของบลูทูธและสามารถปล่อยสัญญาณได้ทุกที่ แม้อยู่ในจุดอับสัญญาณ เพียงแต่อยู่ในบริเวณที่อุปกรณ์บีมคอนตั้งอยู่ บลูทูธที่ Beacon ใช้จะเป็นบลูทูธ 4.0 (Bluetooth Low Energy) โดยในโครงการนี้ใช้บีมคอนทั้งหมด 4 ตัว Ibeacon Mac Address คือ DA :E8:55:21:A9:B6 (1), F7:72:B7:A4:48:10 (2), FB:7C:06:0E:94:EA (3) และ CD:55:4F:CF:DE:E3 (4) (Eddystone) ตามลำดับ ดังรูปที่ 3.39 และรูปที่ 3.40



รูปที่ 3.39 Ibeacon



รูปที่ 3.40 Ibeacon Eddystone

3.2.2 โทรศัพท์มือถือ OPPO reno 2f

ในการทดลองใช้แอปพลิเคชัน จะใช้โทรศัพท์ OPPO reno 2f ที่เป็นระบบปฏิบัติการแอนดรอยด์ ในการทดสอบการทำงานของแอปพลิเคชัน ดังรูปที่ 3.41



รูปที่ 3.41 โทรศัพท์ OPPO reno 2f

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การจัดเก็บผลการทดลอง

3.3.1 จำลองช่องสัญญาณ (Channel Model)

ทำแบบจำลองและหาคุณสมบัติเชิงสถิติของช่องสัญญาณ บีคอน BLE สภาวะแวดล้อมภายในอาคาร คือ บ้านของผู้จัดทำเพื่อได้มาซึ่งความหลากหลายทางกายภาพ ที่ประกอบไปด้วย โต้ะเก้าอี้จำนวนมาก , ตู้ และเตียงโดยติดตั้งบีคอน BLE ทั้งหมด 4 ตัว ใช้โทรศัพท์มือถือวัด RSS ของบีคอน BLE แต่ละตัว สำหรับการทำให้แบบจำลองของช่องสัญญาณ บีคอน BLE ได้แสดงแบบจำลองรีเกรสชันเชิงเส้น (Linear Regression Model: LRM) ของ RSS ตามระยะทาง หลังจากนั้นหาคุณสมบัติเชิงสถิติของช่องสัญญาณ BLE ในเทอมของฟังก์ชันความหนาแน่นความน่าจะเป็น (Probability Density Function: PDF) และฟังก์ชันการแจกแจงสะสม (Cumulative Distribution Function: CDF) ของ RSS ผลที่ได้เหล่านี้สามารถนำไปใช้สำหรับหาตำแหน่งภายในอาคารที่มีพื้นฐานอยู่บน RSS โดยจะพิจารณาของบีคอนที่ละตัวตามพื้นที่ที่คณะผู้จัดทำแบ่งตามพื้นที่ที่กำหนดและทำการจำลองช่องสัญญาณบีคอน

ในโครงการนี้ได้ใช้ LRM ทำแบบจำลองหาความสัมพันธ์ระหว่าง RSS ของ บีคอนแต่ละตัวตามระยะทางระหว่างบีคอนกับโทรศัพท์มือถือที่ใช้วัด โดยที่ LRM จะมีพื้นฐานอยู่บน สูตรการส่งผ่านของฟรีส (5) สำหรับสภาวะแวดล้อมใด ๆ สามารถกำหนดความสัมพันธ์ระหว่าง RSS กับระยะทางได้เป็น (6)

$$P_r = 10n\log(d) + P_{ref} \quad (3.1)$$

เมื่อ P_r เป็น RSS
 d เป็นระยะทาง n
 P_{ref} เป็น RSS ที่ระยะอ้างอิง 1 m

จุดประสงค์ในการทำให้แบบจำลองคือต้องการหาพารามิเตอร์ n และ P_{ref} ที่ทำให้เกิดค่าผิดพลาดน้อยสุด ซึ่งแบบจำลองนี้จะนำไปใช้ประมาณหาระยะทางจาก RSS สำหรับการหาตำแหน่งต่อไป ขั้นแรกจะต้องวัด RSS ที่ระยะทางที่กำหนดโดยกำหนดให้ที่ระยะทางที่ i d_i วัด RSS ได้ $P_{r,i}$ ทำการวัดทั้งหมด N ตำแหน่ง จะได้ค่าผิดพลาดของ RSS กำลังสองทั้งหมดเป็น

$$\sum_{i=2}^N Ei^2 = \sum_{i=2}^N [P_{r,i} + 10n\log(d_i) - P_{ref}]^2 \quad (3.2)$$

จากสมการ (2) ใช้วิธีค่าผิดพลาดกำลังสองน้อยสุดหา หาพารามิเตอร์ n และ P_{ref} ได้จาก

$$\begin{bmatrix} -10n \\ P_{ref} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^N [\log(d_i)]^2 & \sum_{i=1}^N \log(d_i) \\ \sum_{i=1}^N \log(d_i) & N \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^N \log(d_i) P_{r,i} \\ \sum_{i=1}^N P_{r,i} \end{bmatrix} \quad (3.3)$$

ค่าผิดพลาดที่เกิดขึ้นนี้เป็นผลกระทบโดยตรงจากการจางหายหลายวิถี โดยพิจารณาให้ค่าผิดพลาดที่ระยะทาง i ไต่ ๆ E_i เป็นตัวแปรสุ่ม E

$$E_i = P_{r,i} + 10n \log(d_i) - P_{ref} \quad (3.4)$$

$$E = \{E_1, E_2, \dots, E_N\} \quad (3.5)$$

หลังจากนั้นจะหาคุณสมบัติเชิงสถิติในเทอมของ PDF และ CDF ของค่าผิดพลาดหรือผลกระทบของการจางหายหลายวิถีโดยที่ PDF และ CDF มีนิยามเป็น

$$f_E(E_i) = \frac{d}{dE_i} P\{E \leq E_i\} \quad (3.6)$$

$$F_E(E_i) = P\{E \leq E_i\} + \int_{-\infty}^{E_i} f_E(\tau) d\tau \quad (3.7)$$

เมื่อ f_E เป็น PDF F_E เป็น CDF และ P เป็นความน่าจะเป็นในบทความนี้ได้เทียบคุณสมบัติเชิงสถิติของผลกระทบจางหายวิถีนี้เทียบกับแบบจำลองสถิติแบบเกาส์ซึ่งมีนิยามของ PDF และ CDF เป็น

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/(2\sigma^2)} \quad (3.8)$$

$$F_X(x) = Q\left(\frac{x-\mu}{\sigma}\right) \quad (3.9)$$

เมื่อ X เป็นตัวแปรสุ่ม x เป็นค่าตัวแปรสุ่ม μ เป็นค่าเฉลี่ยของตัวแปรสุ่ม σ เป็นส่วนเบี่ยงเบนมาตรฐานของตัวแปรสุ่ม และฟังก์ชัน Q มีนิยามเป็น

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\tau^2/2} d\tau \quad (3.10)$$

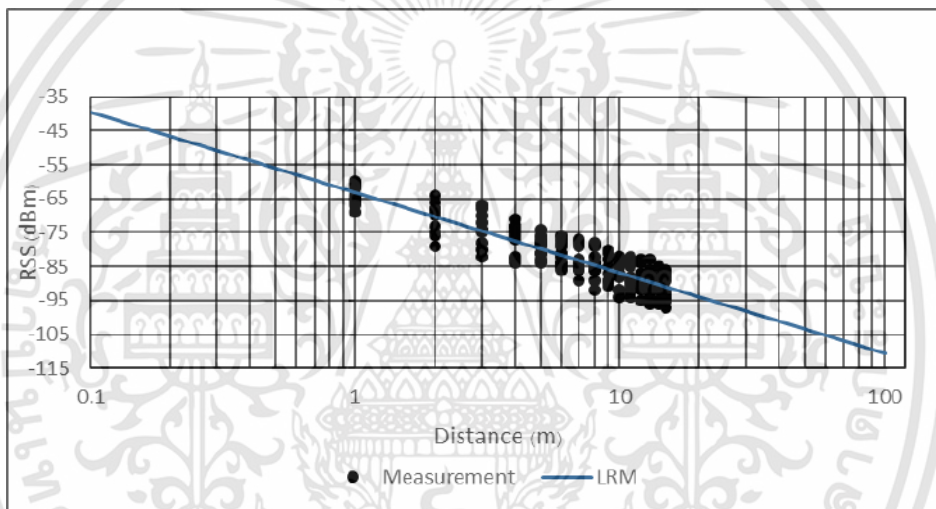
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทำแบบจำลองของช่องสัญญาณบิคอน BLE ได้แสดงแบบจำลองรีเกรสชันเชิงเส้น (Linear Regression Model: LRM) ของ RSS ตามระยะทาง และความสัมพันธ์ระหว่าง RSS ของบิคอนแต่ละตัวตามระยะทางระหว่างบิคอนกับโทรศัพท์มือถือที่ใช้วัด ได้สมการ LRM ดังนี้

ตัวที่ 1 ได้สมการ คือ

$$P_r = -10(2.36)\log(d) - 63.29 \quad (3.11)$$

โดย LRM ของ RSS ตามระยะทางจากบิคอนตัวที่ 1 แสดงดังรูปที่ 3.42

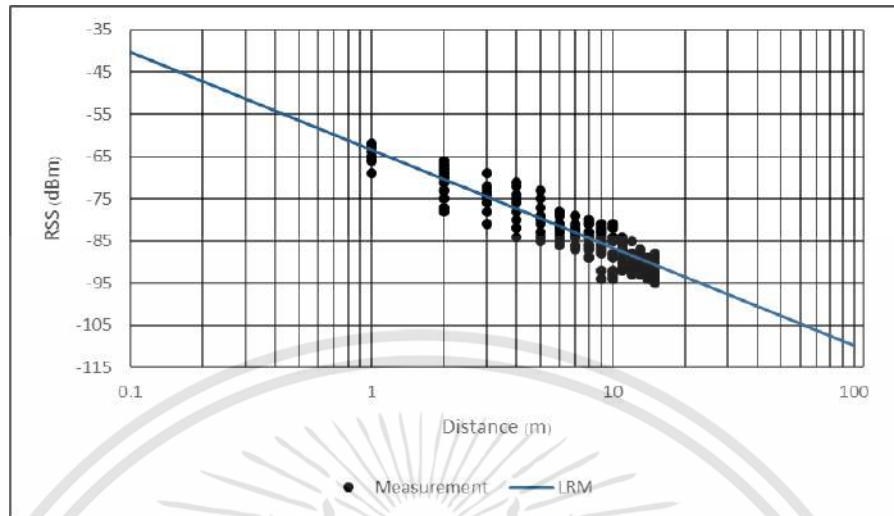


รูปที่ 3.42 LRM ของ RSS ตามระยะทางจากบิคอนตัวที่ 1

ตัวที่ 2 ได้สมการ คือ

$$P_r = -10(2.32)\log(d) - 64.00 \quad (3.12)$$

โดย LRM ของ RSS ตามระยะทางจากบิคอนตัวที่ 2 แสดงดังรูปที่ 3.43

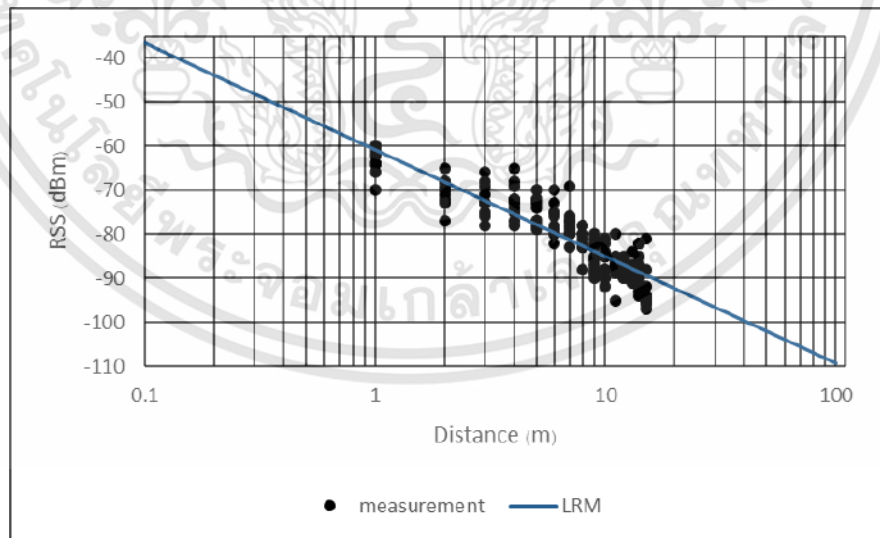


รูปที่ 3.43 LRM ของ RSS ตามระยะทางจากบีกอนตัวที่ 2

ตัวที่ 3 ได้สมการ คือ

$$P_r = -10(2.42)\log(d) - 60.82 \quad (3.13)$$

โดย LRM ของ RSS ตามระยะทางจากบีกอนตัวที่ 3 แสดงดังรูปที่ 3.44



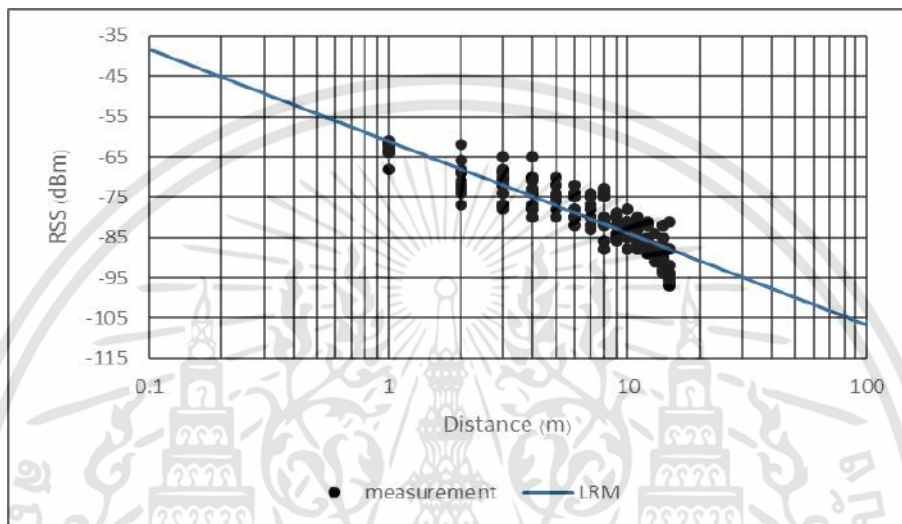
รูปที่ 3.44 LRM ของ RSS ตามระยะทางจากบีกอนตัวที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวที่ 4 ได้สมการ คือ

$$P_r = -10(2.28)\log(d) - 61.12 \quad (3.14)$$

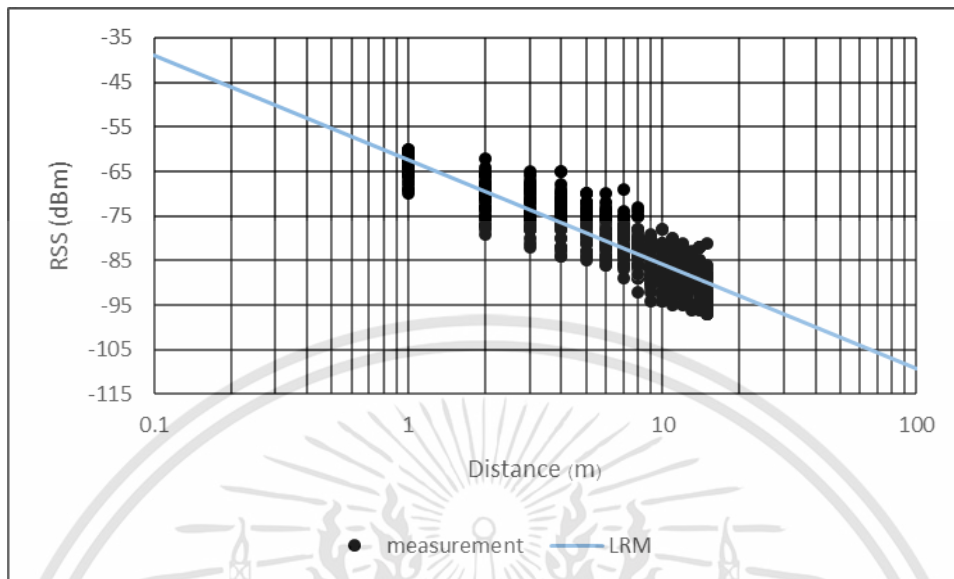
โดย LRM ของ RSS ตามระยะทางจากปีคอนตัวที่ 4 แสดงดังรูปที่ 3.45



รูปที่ 3.45 LRM ของ RSS ตามระยะทางจากปีคอนตัวที่ 4

จากนั้นได้ทำการหา LRM ของ RSS ที่ วัดได้จากปีคอน BLE แต่ละตัวตามระยะทางระหว่างปีคอน BLE กับโทรศัพท์มือถือ โดยใช้ข้อมูล RSS 600 ข้อมูล ค่าพารามิเตอร์ LRM คือ เลขยกกำลังการสูญเสียเชิงวิถีของปีคอน BLE ตัวที่ 1, 2, 3 และ 4 มีค่าเป็น $n = 2.36, 2.32, 2.42$ และ 2.28 ตามลำดับ และ RSS ที่ระยะอ้างอิง 1 m ของปีคอน BLE ตัวที่ 1, 2, 3 และ 4 มีค่าเป็น $P_{ref} = -63.29$ dBm, -64.00 dBm, -60.82 dBm และ -61.12 dBm ตามลำดับ

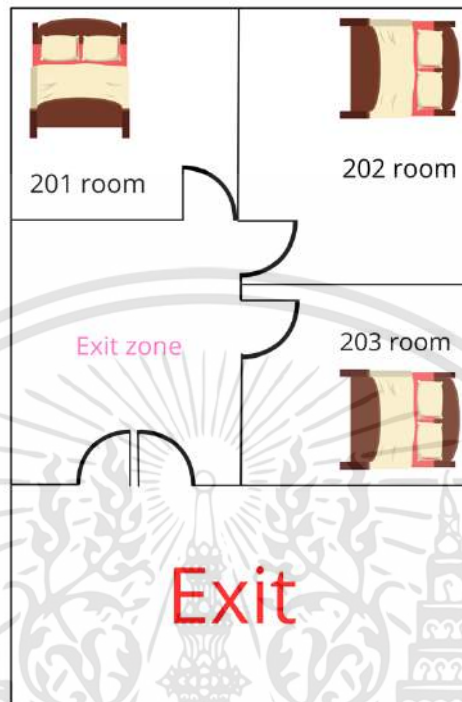
จากการบันทึกผลที่ได้จะเห็นว่าพารามิเตอร์ของ LRM จากปีคอน BLE แต่ละตัวแตกต่างกันไม่มาก ดังนั้น จึงนำข้อมูล RSS ที่วัดได้จากปีคอน BLE แต่ละตัวรวมกันทั้งหมดเป็น 600 ข้อมูล สร้าง LRM ของ RSS ตามระยะทางระหว่างปีคอน BLE กับโทรศัพท์มือถือผลที่ได้แสดงไว้ในรูปที่ 3.46 ซึ่งพารามิเตอร์ของ LRM คือเลขยกกำลังการสูญเสียเชิงวิถี $n = 2.34$ และ RSS ที่ระยะอ้างอิง 1 m $P_{ref} = -62.31$ dBm



รูปที่ 3.46 LRM ของ RSS ตามระยะทางปีคอน BLE ทุกตัว

3.3.2 การทดลองการเชื่อมต่อระหว่างแอปพลิเคชันกับสัญญาณปีคอน

จากการจำลองช่องสัญญาณ ได้ทำการทดสอบเดินเพื่อทดสอบว่าแอปพลิเคชันสามารถตรวจจับสัญญาณเจอหรือไม่และได้ติดตั้ง ตัวปีคอนทั้ง 4 ตัวพร้อมทั้งกำหนดชื่อของ Mac Address ให้แก่ตัวปีคอนและทำการเดิน จากห้องหนึ่งไปอีกอย่างห้องหนึ่ง โดยสถานที่ที่ใช้ในการทดลองคือ ตึกภาควิชาโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง โดยจะแบ่งโซนตามแผนผังดังรูปที่ 3.47 ทั้งนี้ต้องติดตั้งให้เหมาะสมกับสถานที่ที่ด้วยว่าควรวางตำแหน่งปีคอนตำแหน่งใดจึงออกแบบตำแหน่งวางปีคอนให้เหมาะสมกับสถานที่ที่ใช้การทดสอบและทำการติดปีคอนไว้ตามจุดต่าง ๆ ดังรูป 3.48 , 3.49 , 3.50 และรูป 3.51



รูปที่ 3.47 แผนผังห้องที่แบ่งเป็นโซนต่าง ๆ

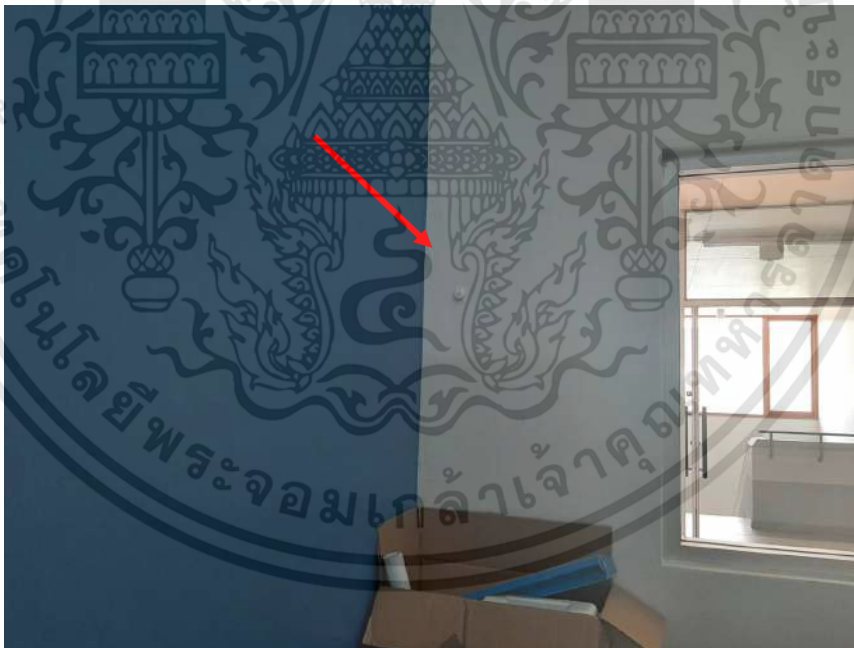


รูปที่ 3.48 จุดติดบีกอนในห้อง 201 room

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.49 จุดติดปีกอนในห้อง 202 room



รูปที่ 3.50 จุดติดปีกอนในห้อง 203 room

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.51 จุดติดบีคอนในห้อง Exit

3.3.1 การทดสอบการเชื่อมต่อของแอปพลิเคชันเข้าฐานข้อมูล

เริ่มจากการสร้างแอปพลิเคชันที่เชื่อมต่อกับฐานข้อมูล โดยส่วนที่เชื่อมกับฐานข้อมูล ได้แก่

- 1) ส่วนหน้าลงทะเบียนเพื่อเข้าสู่ระบบ ทำการป้อนข้อมูลของผู้ใช้งาน จากนั้นจะบันทึกข้อมูลดังกล่าวที่ฐานข้อมูล firebase ในฟิลด์ Users
- 2) ส่วนหน้า Login เพื่อเข้าสู่ระบบ จะดึงข้อมูลมาจากฐานข้อมูล firebase เพื่อเข้าสู่ระบบ โดยมาจากข้อมูลที่ผู้ใช้งานทำการลงทะเบียนไว้แล้ว
- 3) ส่วนหน้าสวัสดิการ และหน้าอาการรายวัน เมื่อเลือกหรือกรอกข้อมูลจะบันทึกที่ฐานข้อมูล firebase โดยเมื่อมีการเลือกหรือกรอกข้อมูลใหม่ก็จะมีการอัปเดตข้อมูลที่ฐานข้อมูล firebase
- 4) ส่วนหน้าดูข้อมูลของผู้ป่วย เป็นส่วนของแอดมิน จะดึงข้อมูลของผู้ป่วยแต่ละคนมาจากฐานข้อมูล firebase มาแสดงยังแอปพลิเคชัน
- 5) การทดสอบการตรวจจับตำแหน่งของผู้ป่วย โดยทำการเปิดอุปกรณ์บีคอน เพื่อปล่อยสัญญาณ และเปิดแอปพลิเคชันในโทรศัพท์มือถือ จากนั้นทำการเปิดลูทูลเพื่อทดสอบว่า ออกจากบริเวณโรงพยาบาลสนามหรือไม่

บทที่ 4

ผลการทดลอง

4.1 การทดสอบการค้นหาสัญญาณบีคอน

ทำการค้นหาสัญญาณบีคอนโดยใช้แอปพลิเคชัน BLE Scanner โดยกดเปิดอุปกรณ์บีคอน และเปิดแอปพลิเคชัน BLE Scanner (ต้องเปิดบลูทูธของโทรศัพท์มือถือที่เปิดแอปพลิเคชันนี้ด้วย) เพื่อค้นหาสัญญาณของอุปกรณ์บีคอน พบว่าแอปพลิเคชัน BLE Scanner แสดงข้อมูลของบีคอน ดังนี้ 1.UUID 2.Major 3.Minor 4.Address ดังแสดงในรูปที่ 4.1

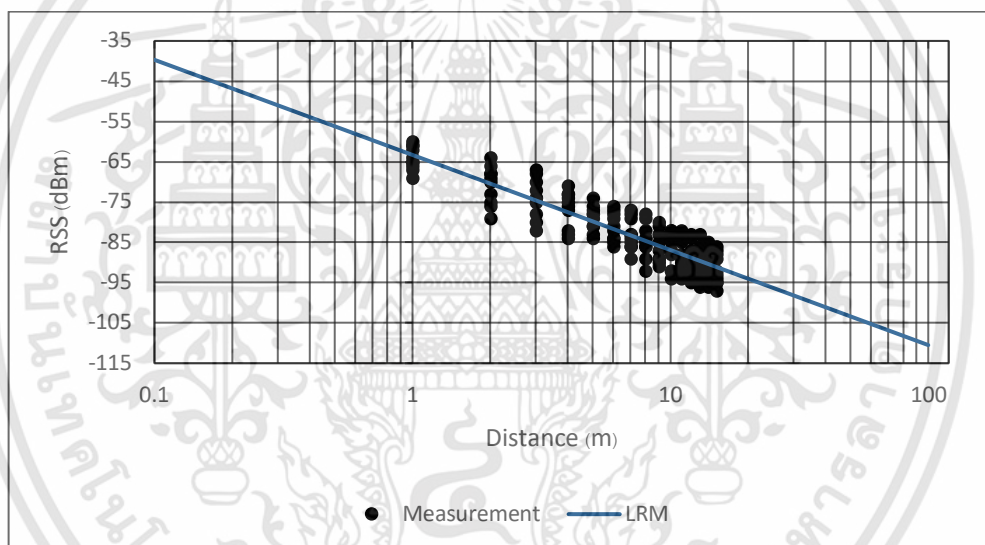


รูปที่ 4.1 แอปพลิเคชัน BLE Scanner

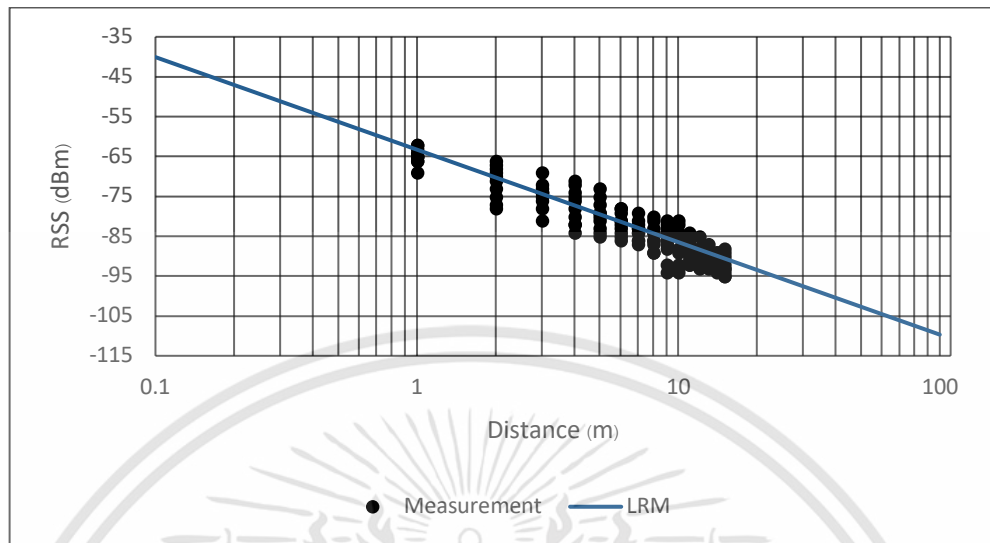
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลทดสอบโดยใช้ปีคอน 4 ตัว วัดค่าความแรงเมื่อวัดประมาณค่า RSSI และนำมาหาค่าเฉลี่ยของแต่ละจุดและคำนวณระยะจริงและระยะทดลอง

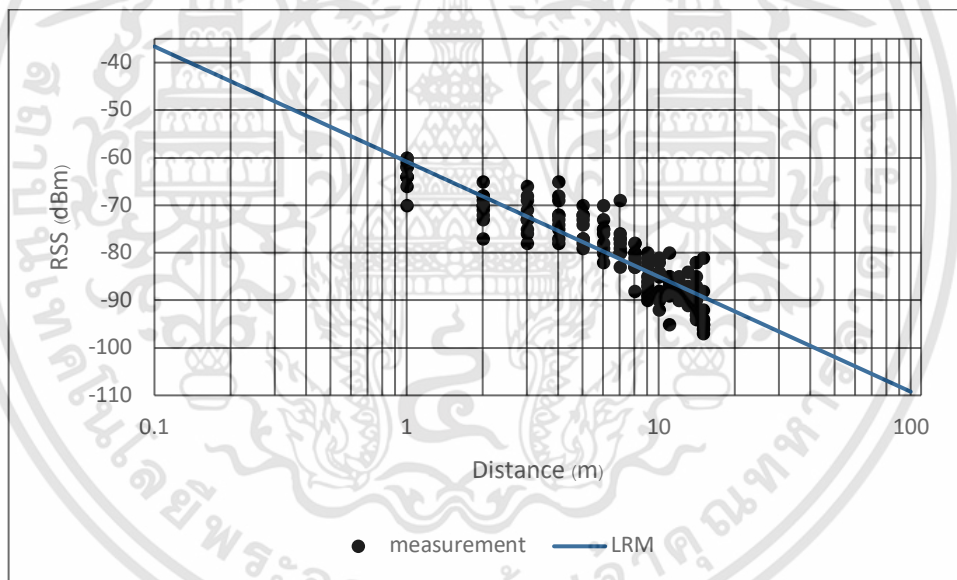
จากนั้นนำค่าเฉลี่ยมาแสดงแบบจำลองรีเกรสชันเชิงเส้น (Linear Regression Model : LRM) เลขยกกำลังการสูญเสียเชิงวิถีของปีคอนตัวที่ 1, 2, 3 และ 4 คือ $n = 2.36$, 2.32 , 2.42 และ 2.28 ตามลำดับและ RSSI ที่ระยะอ้างอิง 1 เมตรของปีคอน BLE ตัวที่ 1, 2, 3 และ 4 คือ $P_{ref} = -63.29$, -64.00 , -60.82 และ -61.12 ตามลำดับเพื่อหาความสัมพันธ์ระหว่าง RSS ของปีคอนแต่ละตัวตามระยะทางระหว่างปีคอน BLE กับโทรศัพท์มือถือและได้แสดงบนกราฟ LRM ของปีคอนตัวที่ 1 ดังรูปที่ 4.2 กราฟ LRM ของปีคอนตัวที่ 2 ดังรูปที่ 4.3 กราฟ LRM ของปีคอนตัวที่ 3 ดังรูปที่ 4.4 และกราฟ LRM ของปีคอนตัวที่ 4 ดังรูปที่ 4.5



รูปที่ 4.2 กราฟแสดง LRM ของ RSSI ของปีคอนตัวที่ 1

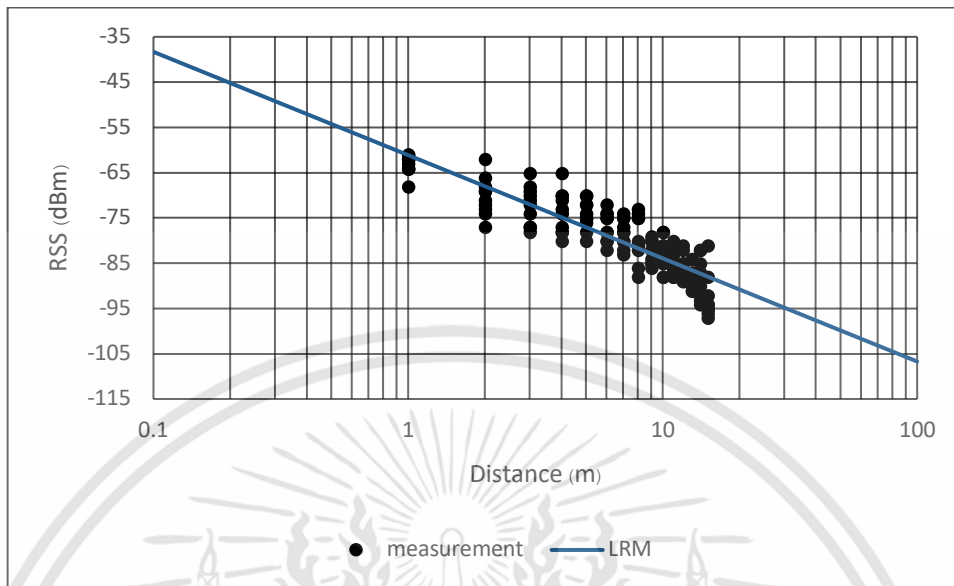


รูปที่ 4.3 กราฟแสดง LRM ของ RSSI ของปีคอนตัวที่ 2



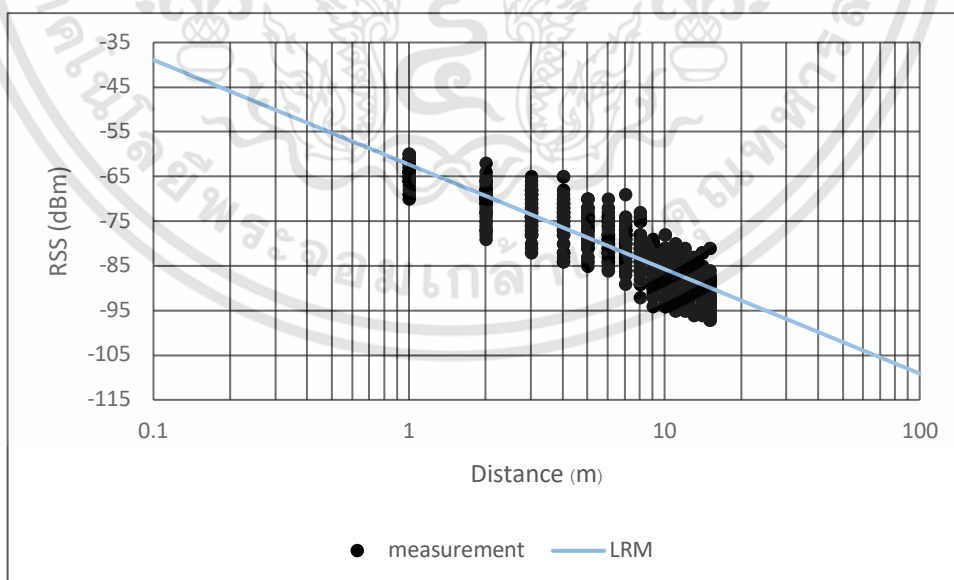
รูปที่ 4.4 กราฟแสดง LRM ของ RSSI ของปีคอนตัวที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 กราฟแสดง LRM ของ RSSI ของปีคอนตัวที่ 4

จากการที่นำค่าเฉลี่ยค่า RSSI ของปีคอนทั้ง 4 ตัว มาพล็อตกราฟเพื่อหาแนวโน้ม LRM ของปีคอน พบว่าค่า LRM ของปีคอน BLE ทั้ง 4 ตัวนั้นมีค่าต่างกันไม่มาก หลังจากนั้นนำข้อมูลของปีคอน BLE ทั้ง 4 ตัวมารวมกันซึ่งจะได้ค่าเลขยกกำลังการสูญเสียเชิงวิถี คือ $n=2.34$ และ RSSI ที่ระยะอ้างอิง 1 เมตรของปีคอน BLE รวมทั้ง 4 ตัว คือ $P_{ref} = -62.31$ จะได้กราฟ LRM ของปีคอน BLE รวมทั้ง 4 ตัว ดังรูปที่ 4.6



รูปที่ 4.6 กราฟแสดง LRM ของ RSSI ของปีคอนทั้ง 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นจะถูกแปลงเป็นเมื่อพิจารณาความสัมพันธ์ตามสมการ Path loss ในสมการที่ 2.2 จะได้ สมการ Path loss สำหรับสัญญาณปีคอนดังนี้

$$P_r(d)[dBm] = P_{ref}(1m) - 10n \times \log_{10}(d) \quad (4.1)$$

$$\text{จะได้} \quad P_r(d)[dBm] = (-62.31) - 10(2.34) \times \log_{10}(d) \quad (4.2)$$

โดยที่ $P_r(d)$ คือ กำลังงาน RSSI ที่รับได้ระยะทาง d ในหน่วย dBm

$P_r(1m)$ คือ กำลังงาน RSSI ที่รับได้ใน 1 เมตร

n คือ Path Loss Exponent

จากนั้นทำการหาค่าพารามิเตอร์ n และ P_{ref} ที่ทำให้เกิดความผิดพลาดน้อยที่สุด ซึ่งจะสามารถประมาณหาระยะทางจาก RSSI สำหรับการหาตำแหน่งต่อไป ค่าผิดพลาดที่เกิดขึ้นนี้เป็นผลกระทบโดยตรงจากการจางหายหลายวิถี โดยพิจารณาให้ค่าผิดพลาดที่ระยะทาง i ใด ๆ และให้ E_i เป็นตัวแปรสุ่ม E

$$E_i = P_{r,i} + 10n \log(d_i) - P_{ref} \quad (4.3)$$

$$E = \{E_1, E_2, \dots, E_N\} \quad (4.4)$$

เมื่อนำค่าผิดพลาดนิยามให้เป็นผลกระทบของการจางหายหลายวิถี ซึ่งข้อมูลผลกระทบของการจางหายหลายวิถีมีค่าเฉลี่ย $\mu = 0$ dB และค่าส่วนเบี่ยงเบนมาตรฐานเป็น $\sigma = 8.69$ dB

4.2 การทดสอบการเชื่อมต่อของแอปพลิเคชันกับฐานข้อมูล

ผลการทดสอบการเชื่อมต่อระหว่างแอปพลิเคชันกับฐานข้อมูล เมื่อผู้ใช้งานกรอกประวัติส่วนตัว แล้วข้อมูลส่วนตัว จะไปแสดงในฐานข้อมูล

4.2.1 การลงทะเบียนเพื่อเข้าสู่ระบบ

ทำการออกแบบหน้าของแอปพลิเคชันสำหรับลงทะเบียน โดยสามารถกรอกชื่อ-นามสกุล เบอร์โทรศัพท์ รหัสผ่าน อายุ เพศ ที่อยู่ วัน/เดือน/ปีเกิด วันที่เข้ารับการรักษา วันที่พบเชื้อ ประวัติแพ้ยา/ผ่าตัด ดังรูปที่ 4.7 โดยเมื่อผู้ใช้งานกรอกประวัติตามนี้ แล้วข้อมูลดังกล่าวจะไปแสดงผลในฐานข้อมูลดังรูปที่ 4.8 และการเรียกดูข้อมูลของผู้ป่วยโดยแอปพลิเคชัน (แอดมิน) ดังรูปที่ 4.9 โดยที่สามารถดูข้อมูลผู้ป่วยรายบุคคลได้อีกด้วย ดังรูปที่ 4.10

REGISTER

ชื่อ - นามสกุล Name

อีเมล Email

รหัสผ่าน Password

ยืนยันรหัสผ่าน Confirm password

เพศ ชาย หญิง

ที่อยู่ Address

เบอร์โทรศัพท์ Phone number

วัน/เดือน/ปีเกิด วัน/เดือน/ปี เลือก

วันที่เข้ารับการตรวจ วัน/เดือน/ปี เลือก

วันนัดหมาย วัน/เดือน/ปี เลือก

ประวัติแพ้ยา/ผ่าตัด Drug allergy/Surgery

ลงทะเบียน

```

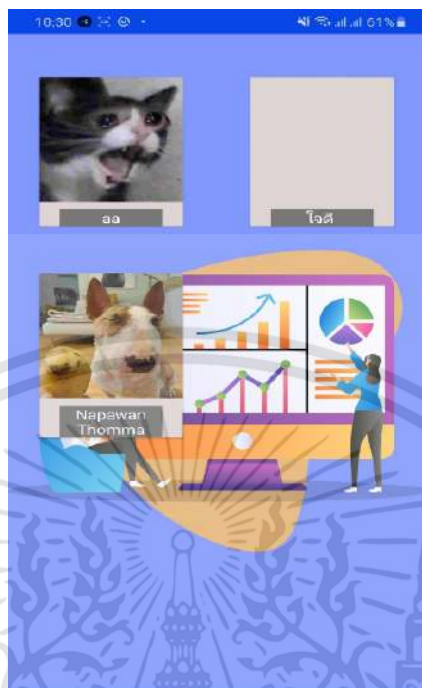
Service
Users
  72ZkF2emQEPPpX3Ap8nNLXc3qH3
  HWkTIVkp1kRSPjQXn6i79xK95ZH2
    AccountType: "User"
    Address: "km11"
    Covid: "2-12-2021"
    Date: "11-12-2000"
    Email: "napawan@gmail.com"
    Feel: "ไม่มี"
    Gender: "ชาย"
    Name: "กันเขียง หนุอ"
    Phone_number: "0611382580"
    Treat: "7-12-2021"
  VNybbvaS9XMXkg5FNRoLOE5qn3I1
  mumqWWQHlaTIE094Vep5seKyjbH2

```

รูปที่ 4.7 แสดงหน้าแอปพลิเคชันข้อมูลการลงทะเบียนของผู้ป่วย

รูปที่ 4.8 ข้อมูลประวัติส่วนตัวของผู้ใช้งานในฐานะข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 แสดงหน้าแอปพลิเคชันสำหรับดูข้อมูลผู้ป่วย

ข้อมูลของผู้ป่วย	
ชื่อ :	aa
อีเมล :	aa@gmail.com
เพศ :	หญิง
ที่อยู่ :	1/5
เบอร์โทรศัพท์ :	00000
วันเกิด :	21-2-2022
วันที่เข้ารับการรักษา :	16-2-2022
วันที่พบเชื้อ :	24-2-2022
อาการแพ้ยา :	-

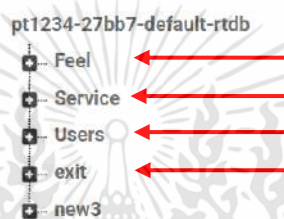
รูปที่ 4.10 แสดงหน้าแอปพลิเคชันสำหรับดูผู้ป่วยรายบุคคล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 การ Login เพื่อเข้าสู่ระบบ

ทำการออกแบบหน้า Login โดยเมื่อผู้ใช้งานมีข้อมูลประวัติส่วนตัวในฐานข้อมูลแล้วทำการเข้าสู่ระบบโดยต้องกรอกอีเมล และรหัสผ่าน เพื่อที่จะสามารถเข้าสู่ระบบได้

4.2.3 การใช้ฟังก์ชันเสริมในแอปพลิเคชัน จะประกอบไปด้วยปุ่มกรอกอุณหภูมิและบันทึกอาการรายวัน ปุ่มสวัสดีการ และปุ่มติดต่อเจ้าหน้าที่ โดยออกแบบเพื่ออำนวยความสะดวกแก่ผู้ป่วยในโรงพยาบาลสนาม เพื่อลดการสัมผัสระหว่างผู้ป่วยกับบุคคล เมื่อผู้ป่วยบันทึกผลในแอปพลิเคชัน ข้อมูลที่บันทึกต้องเข้าฐานข้อมูลดังรูปที่ 4.11 ละแอดมินต้องเรียกดูข้อมูลผ่านได้



รูปที่ 4.11 ข้อมูลที่เข้าฐานข้อมูล

ข้อมูลที่เข้าฐานข้อมูลได้แก่

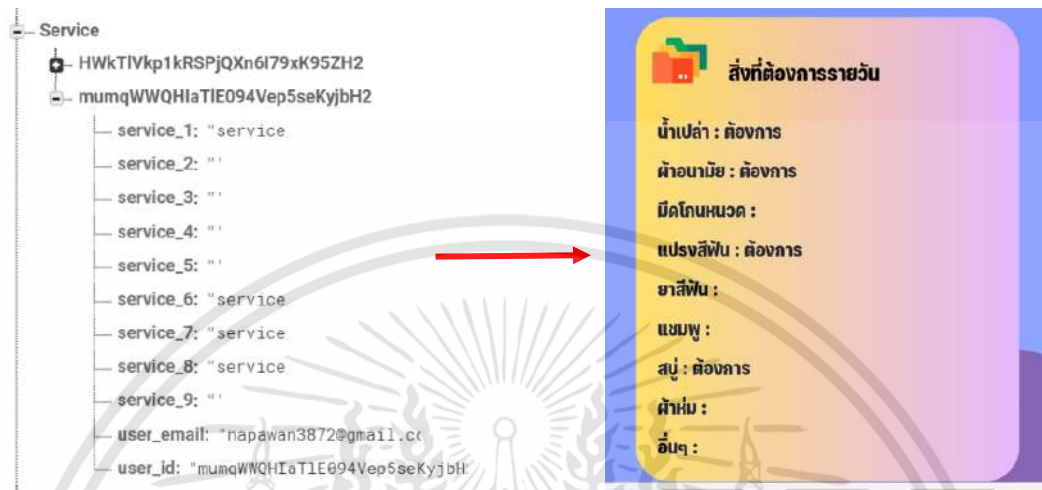
1. Feel อาการและอุณหภูมิของผู้ป่วย
2. Service สวัสดีการ ในกรณีผู้ป่วยต้องการสั่งอำนวยความสะดวกเพิ่มเติม
3. Users เป็นข้อมูลส่วนตัวของผู้ป่วย
4. exit ข้อมูลการเชื่อมต่อปีคอน

```
Feel
{
  "mumqWWQHlaTIE094Vep5seKyjbH2": {
    "syntom_1": "",
    "syntom_10": "",
    "syntom_2": "",
    "syntom_3": "syntom_3",
    "syntom_4": "",
    "syntom_5": "",
    "syntom_6": "",
    "syntom_7": "syntom_7",
    "syntom_8": "",
    "syntom_9": "",
    "temp": "temp_1",
    "user_email": "napawan3872@gmail.com",
    "user_id": "mumqWWQHlaTIE094Vep5seKyjbH2"
  }
}
```

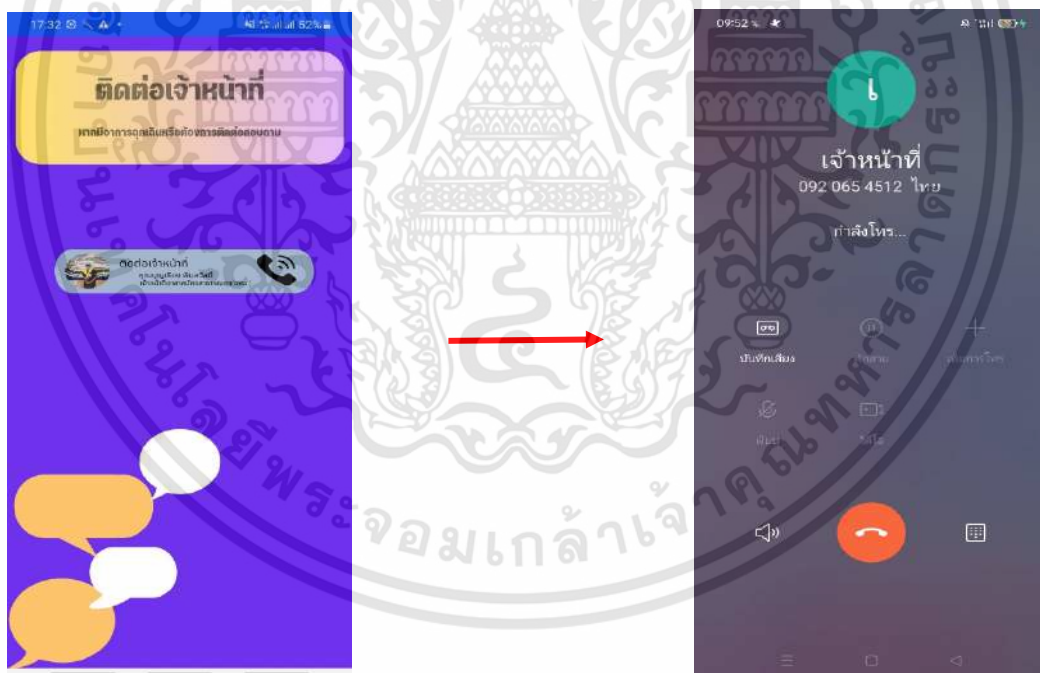


รูปที่ 4.12 ข้อมูลการกรอกอาการเข้าฐานข้อมูลและแสดงข้อมูลในแอปพลิเคชันแอดมิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 ข้อมูลการกรอกสวัสดิการเข้าฐานข้อมูลและแสดงข้อมูลในแอปพลิเคชันแอดมิน

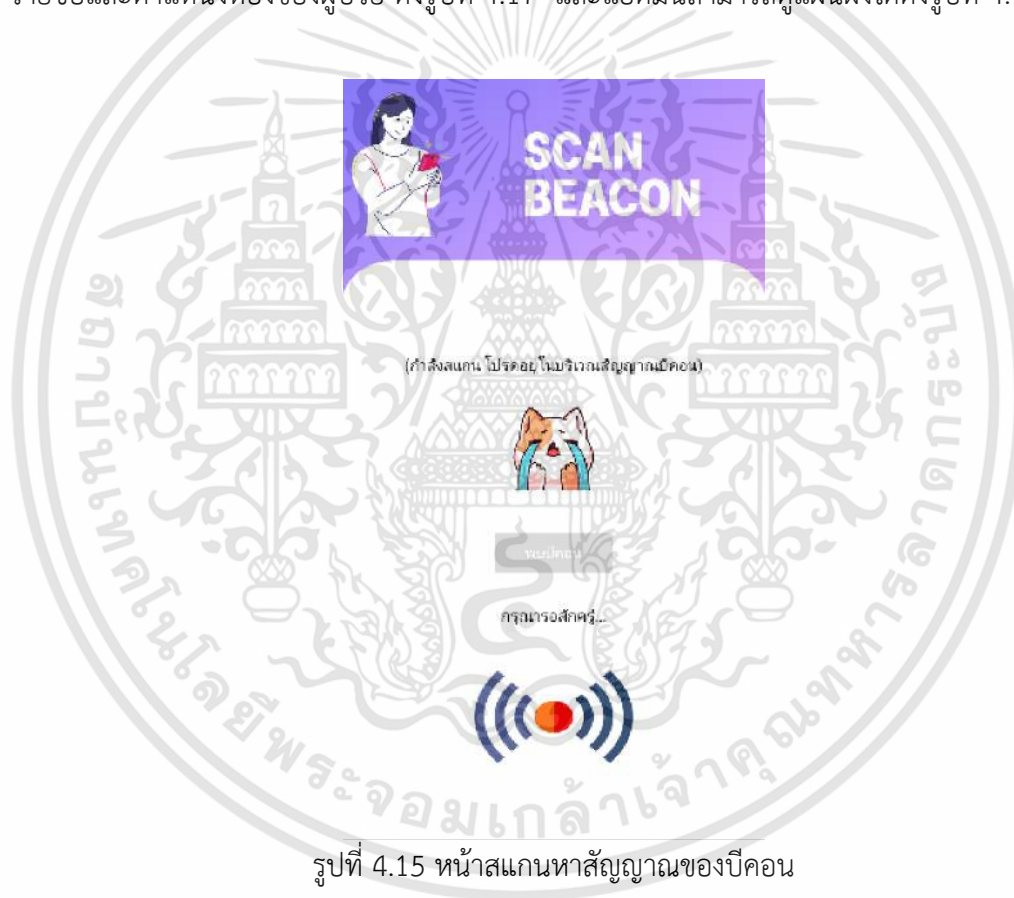


รูปที่ 4.14 ปุ่มนี้ติดต่อเจ้าหน้าที่โดยจะเชื่อมการโทรออกในโทรศัพท์ของผู้ป่วยไม่ต้องเข้าฐานข้อมูล

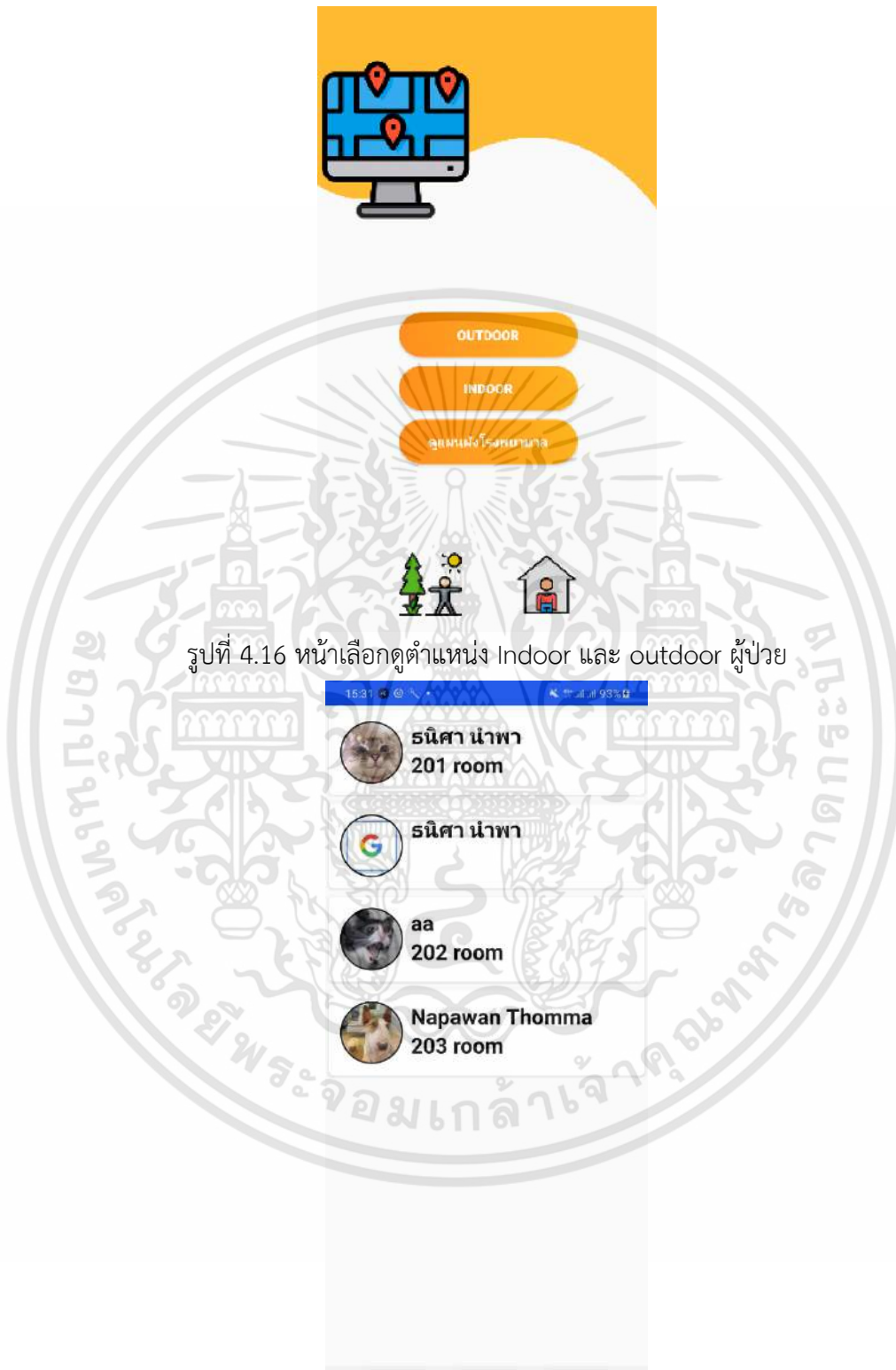
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดสอบการเชื่อมต่อของปีคอนกับแอปพลิเคชัน

การทดสอบการเชื่อมต่อของปีคอนกับแอปพลิเคชัน เพื่อทดสอบว่าสามารถรับสัญญาณของสัญญาณปีคอนได้จริงหรือไม่ และสามารถแสดงตำแหน่งของผู้ป่วยได้หรือไม่ โดยเมื่อผู้ใช้งานทำการกดปุ่มแสกนแล้ว แอปพลิเคชันจะไปยังหน้าสแกนหาสัญญาณของปีคอน ดังรูปที่ 4.15 โดยโทรศัพท์มือถือของผู้ใช้งานต้องเปิดบลูทูธ และอยู่ในบริเวณรัศมีสัญญาณปีคอนที่เรากำหนด โดยจะแบ่งเป็น room 201 , room 202 , room 203 และ Exit ซึ่งแอดมินสามารถเลือกดูตำแหน่ง indoor หรือ outdoor ของผู้ป่วยได้ ดังรูปที่ 4.16 และในส่วนของหน้า Indoor จะบอกรายชื่อและตำแหน่งห้องของผู้ป่วย ดังรูปที่ 4.17 และแอดมินสามารถดูแผนผังได้ดังรูปที่ 4.18



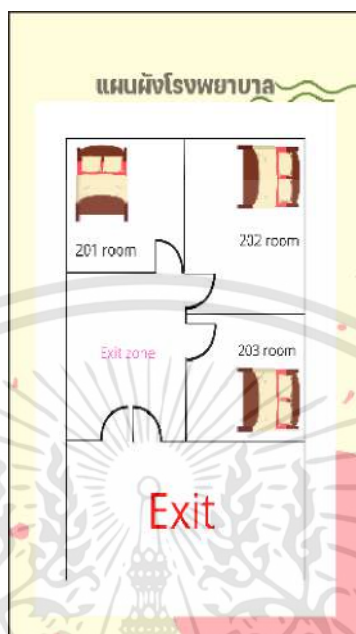
รูปที่ 4.15 หน้าสแกนหาสัญญาณของปีคอน



รูปที่ 4.16 หน้าเลือกดูตำแหน่ง Indoor และ outdoor ผู้ป่วย

รูปที่ 4.17 หน้าแสดงรายชื่อและตำแหน่งห้องของผู้ป่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 หน้าแสดงแผนผังโรงพยาบาลสนาม

4.4 การทดสอบการเชื่อมต่อของ GPS กับแอปพลิเคชัน

การทดสอบการเชื่อมต่อของ GPS กับแอปพลิเคชันนั้น ผู้ใช้งานจะต้องกด”ยืนยันตำแหน่ง” เพื่อส่งข้อมูล Lat , Lon ไปยังฐานข้อมูลให้กับแอดมิน ดังรูปที่ 4.19 และจะมีแถบแจ้งเตือนด้านบนของโทรศัพท์ที่จะแสดงค่า Lat , Lon ของผู้ใช้งานอีกด้วย ดังรูปที่ 4.20 และทางฐานข้อมูลจะส่งข้อมูลผู้ใช้งานที่ออกจากพื้นที่เป็นระยะมากกว่า 25 เมตรเพื่อให้ทางแอดมินได้ทราบว่าผู้ใช้งานที่ออกจากพื้นที่นั้นมีใครบ้าง ดังรูปที่ 4.21 และสามารถเลือกดูตำแหน่ง indoor หรือ outdoor ของผู้ป่วยได้ ดังรูปที่ 4.16 และในส่วนของ outdoor จะแสดงแผนที่ของผู้ใช้งานที่ออกจากพื้นที่นั้นอยู่ตำแหน่งไหนในปัจจุบัน ดังรูปที่ 4.22 โดยผู้ใช้งานสามารถลบข้อมูลผู้ใช้งานที่ออกจากพื้นที่ได้ โดยการกดสแกนป๊อปอัพและยืนยันตำแหน่ง Location สถานที่ที่กำหนดเอาไว้เท่านั้น จึงจะสามารถลบข้อมูลผู้ใช้งานที่ออกจากพื้นที่ได้



รูปที่ 4.19 หน้าส่งข้อมูล Location ของผู้ใช้งาน

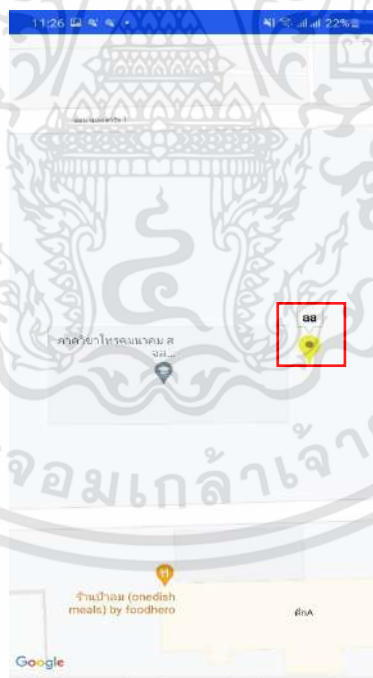


รูปที่ 4.20 แสดงค่า Lat , Lon บนแถบแจ้งเตือนของมือถือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 หน้าต่างของแอดมิน ที่แสดง User id ที่ออกจากบริเวณพื้นที่



รูปที่ 4.22 หน้า map แสดงตำแหน่งที่อยู่ของ User ID ที่ออกจากพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

ในปฏิญญาฉบับนี้มีการออกแบบและสร้างแอปพลิเคชัน เพื่อตรวจสอบบุคคลที่อยู่ในสถานีกักตัวหรือโรงพยาบาลสนาม ซึ่งในปฏิญญาฉบับนี้ได้ดำเนินการมาถึงการทำการตรวจสอบบุคคลโดยแอปพลิเคชันที่สร้างขึ้นในโปรแกรม Android Studio และมีการประยุกต์ใช้ตัวบ่งชี้ในการระบุตำแหน่งภายในอาคาร สามารถดูตำแหน่งของผู้ใช้งาน (ผู้ป่วย) ได้แบบเรียลไทม์ รวมถึงมีฐานข้อมูลที่สามารถตรวจสอบบุคคลที่มีความประสงค์จะออกจากสถานที่ดังกล่าวได้ พร้อมทั้งมีบริการอำนวยความสะดวกความต้องการของผู้ใช้งาน (ผู้ป่วย) รวมถึงการกรอกข้อมูลและอาการรายวันอีกด้วย ทั้งหมดนี้ล้วนเป็นการจัดทำเพื่อให้เกิดการประยุกต์นำความรู้ที่ศึกษาและค้นคว้านำมาใช้สร้างเป็นแอปพลิเคชันที่สามารถนำไปใช้งานจริงและเพื่อใช้เป็นเครื่องมือในการตรวจสอบและอำนวยความสะดวกให้แก่ผู้ใช้งาน (ผู้ป่วย) และ ผู้ดูแล (บุคลากรทางการแพทย์)

5.2 ข้อเสนอแนะ

บิตคอนถูกสัญญาณรบกวนได้ง่าย กำลังที่ได้รับค่อนข้างกว้างต้องการทดสอบซ้ำจุดเดิม ๆ หลายครั้งเพื่อได้ค่าเฉลี่ยของจุด ๆ นั้น

มีปัญหาในเรื่องจากเขียนโปรแกรมเนื่องจากโปรแกรม Android studio นั้นมีการอัปเดตอยู่ตลอด โค้ดบางส่วนนั้นเป็นยังเป็นโค้ดเก่าทำให้เราต้องค้นคว้าจากหลาย ๆ แหล่งที่มีการอัปเดตอยู่เสมอ ทำให้ล่าช้าในการทำงาน แก้ปัญหาโดยการศึกษามาจาก YouTube สอบถามในกลุ่มพูดคุยแลกเปลี่ยนความรู้ในกลุ่มพัฒนาแอปพลิเคชันของ Facebook

บรรณานุกรม

- [1] องค์การบริหารส่วนตำบลดงบัง. “จุดเสี่ยงในที่สาธารณะ...จับแล้วต้องรีบล้าง.”
<http://dongbung-buengkan.go.th/?p=16205>.
- [2] สำนักงานสภานโยบายการอุดมศึกษา วิทยาศาสตร์ วิจัยและนวัตกรรมแห่งชาติ (สอวช.).
“เทียบความแตกต่าง “โรงพยาบาลสนาม” กับ “Hospital” สถานที่ใดเหมาะกับผู้ป่วย
แบบไหน.” <https://www.nxpo.or.th/th/7551/>.
- [3] โรงพยาบาลเพชรเวช. “เช็กอาการผู้ป่วยโควิด-19 สีเขียว สีเหลือง และสีแดง.”
https://www.petcharavejhospital.com/th/Article/article_detail/symptom-level-of-covid-19.
- [4] Alibaba Group. “Taidacent NRF51822 Beacon โมดูลแบบพกพา IoT Ibeacon ฐาน
สถานีอุปกรณ์ Ibeacon ทหมดหมู่ Bluetooth-ใช้งานร่วมกับ Proximity Beacon.”
<https://th.aliexpress.com/item/4000601214651.html>.
- [5] MarcusCode. “ภาษา Java.” <http://marcuscode.com/lang/java>.
- [6] อีรวัดน์ ประกอบผล. *เขียนโปรแกรมเชิงวัตถุภาษา Java ฉบับสมบูรณ์(3)*. กรุงเทพฯ : รีโว่, 2560.
- [7] Amanda Alexander. “Android Studio Arctic Fox (2020.3.1) Stable.”
<https://android-developers.googleblog.com/2021/07/android-studio-arctic-fox-202031-stable.html>.
- [8] Ponglang Petrung. “การเริ่มต้นพัฒนาโปรแกรม Android Studio.”
<https://pongployappdev.medium.com/บทที่-3-การเริ่มต้นพัฒนาโปรแกรม-android-studio-32beda86f397>.
- [9] Computer system department Office of Computer Services, Kasetsart University.
“การออกแบบและกำหนดค่าเลย์เอาต์.”
<http://pirun.ku.ac.th/~faastwc/Android/ponglang/ready/pdf/8%20บทที่-5-การออกแบบและกำหนดค่าเลย์เอาต์-81-97.pdf>.
- [10] Palm. “เริ่มต้นสร้าง Android Application พื้นฐานด้วย Android Studio.”
<https://medium.com/@palmz/เริ่มต้นสร้าง-android-application-พื้นฐานด้วย-android-studio-lab-3sb04-3fda43b07a1>.

- [11] Akexorcist. “รู้จักการใช้งาน Android Studio แบบพื้นฐาน — ตอนที่ 2.”
<https://akexorcist.dev/using-android-studio-2/>.
- [12] Akexorcist. “Notification in Android ตอนที่ 1 เรื่องพื้นฐานของ Notification ที่ควรรู้.”
<https://akexorcist.dev/notification-in-android-part-1/>.
- [13] Akexorcist. “Notification in Android ตอนที่ 5 — Notification Channel.”
<https://akexorcist.dev/notification-in-android-part-5/>.
- [14] Akexorcist. “Notification in Android ตอนที่ 2 — Notification Channel.”
<https://akexorcist.dev/notification-in-android-part-2/>.
- [15] Developers. “Create a Notification.”
<https://developer.android.com/training/notify-user/build-notification>.
- [16] Akexorcist. “Service in Android — [ตอนที่ 1] พื้นฐานของ Service.”
<https://akexorcist.dev/service-in-android-part-1-basic-of-android-service/>.
- [17] Moko technology limited. “What is iBeacon?.”
<https://www.mokoblue.com/a-detailed-guide-to-ibeacon/>.
- [18] Moko technology limited. “All you need to Know about Beacons.”
<https://www.mokoblue.com/th/what-is-beacon/>.
- [19] Komkid. “ไขความลับ Beacons เทคโนโลยี Internet of Things ที่ฮาร์ดแวร์สตาร์ทอัพชอบใช้.” <https://startup.in.th/beacon-internet-of-thing-hardware-startup/#:~:text=Beacons%20%E0%B9%80%E0%B8%9B%E0%B9%87%E0%B8%99%E0%B8%AB%E0%B8%99%E0%B8%B6%E0%B9%88%E0%B8%87%E0%B9%83%E0%B8%99%E0%B8%AD%E0%B8%B8%E0%B8%9B%E0%B8%81%E0%B8%A3%E0%B8%93%E0%B9%8C,%E0%B8%84%E0%B8%A7%E0%B8%A3%E0%B9%81%E0%B8%AA%E0%B8%94%E0%B8%87%E0%B8%82%E0%B9%89%E0%B8%AD%E0%B8%A1%E0%B8%B9%E0%B8%A5%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3%20%E0%B9%81%E0%B8%A5%E0%B8%B0>.
- [20] Nikita Adkar, Akshata Talele, Chinmayee Mundhe, Anita Gunjal. “Bluetooth Beacon Applications in Retail Market.” Department of MAEER’s MIT College of Engineering, Pune, India. 2018.

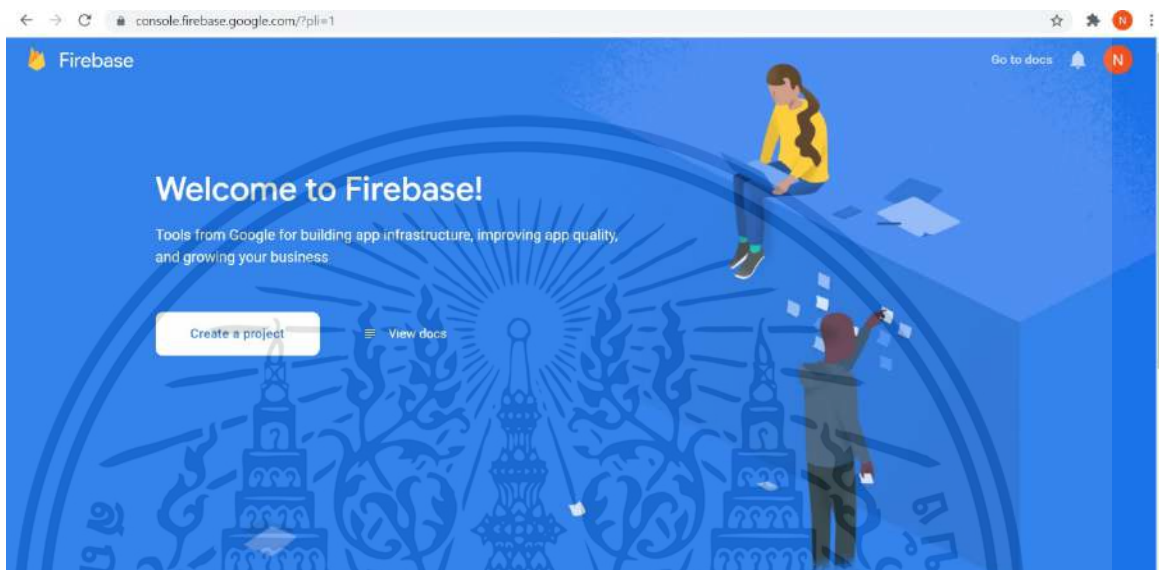
- [21] บริษัท โค้ดบี จำกัด. “การพัฒนาซอฟต์แวร์ เว็บและโมบายเทคโนโลยี.”
<https://www.codebee.co.th/labs/>.
- [22] Jirawatee. “รู้จัก Firebase Realtime Database ตั้งแต่ Zero จนเป็น Hero.”
<https://medium.com/firebasethailand/tagged/rtdb>
- [23] Mohammed S. Salim, Khalil H. Sayidmarie, Abdullah H. Aboud. “Investigation of indoor propagation of WLAN signals.” Department of Information Communication Engineering, College of Electronic Engineering, Ninevah University, 2019.
- [24] ภารตะ เพ็ชศรี . “การวิเคราะห์คุณภาพสัญญาณเพื่อการประเมินจุดบัพของเครือข่าย.” วิทยานิพนธ์ปริญญาวิศวกรรมศาสตรมหาบัณฑิต, สาขาวิศวกรรมอุตสาหกรรม คณะวิศวกรรมศาสตร์, มหาวิทยาลัยสงขลานครินทร์, 2556.
- [25] ธนากร ประสารศรี และคนอื่น ๆ. “การประมาณตำแหน่งของไวเลสเซ็นเซอร์เน็ตเวิร์ค ด้วยวิธีสปริงรีแลกเซชัน.” วิทยานิพนธ์ปริญญาวิศวกรรมศาสตรมหาบัณฑิต, สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์, มหาวิทยาลัยขอนแก่น, 2556.
- [26] Santosh Subedi and Jae-Young Pyun. “Practical Fingerprinting Localization for Indoor Positioning System by Using Beacons.” Department of Information Communication Engineering, Chosun University, 2017.
- [27] Jae Gu Lee , Jin Kim , Seon Woo Lee and Young Woong Ko. “A Location Tracking System using BLE Beacon Exploiting a Double-Gaussian Filter.” Department of Computer Engineering, Smart Computing Lab, Hallym University, Chuncheon, Gangwon, 2017.
- [28] ตฤณ จันทนะ, มนต์ชัย แซ่มซ้อย, พิชญ์ สุพรรณกุล. “การทำแบบจำลองและการหาคุณสมบัติเชิงสถิติของช่องสัญญาณบีกอนที่ใช้ลูทูล่วงงานสำหรับการหาตำแหน่งภายในอาคาร.” สาขาวิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2563.
- [29] Pitchi cha ichi tasho ichi to. “interface android studio.”
https://www.pinterest.com/pit_bright/interface-android-studio/.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

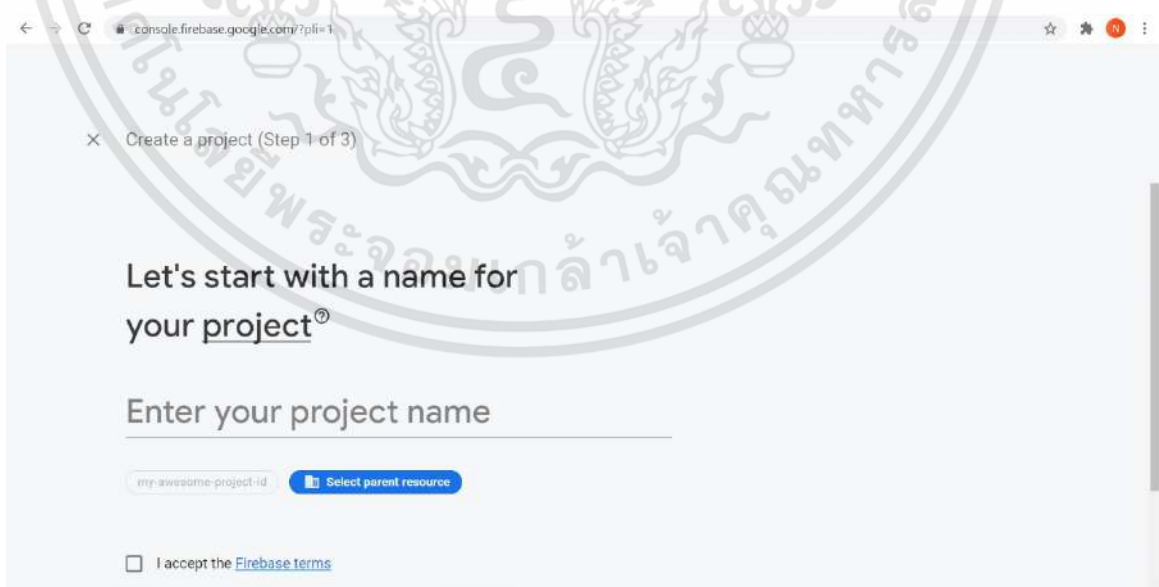
สร้างฐานข้อมูลใน Firebase

1. เริ่มต้นสร้างโครงการโดยเข้าไปที่ <https://firebase.google.com>
2. เข้าสู่ระบบบัญชี google
3. กด Creat Project



รูปที่ 1 เริ่มต้นสร้างโครงการ

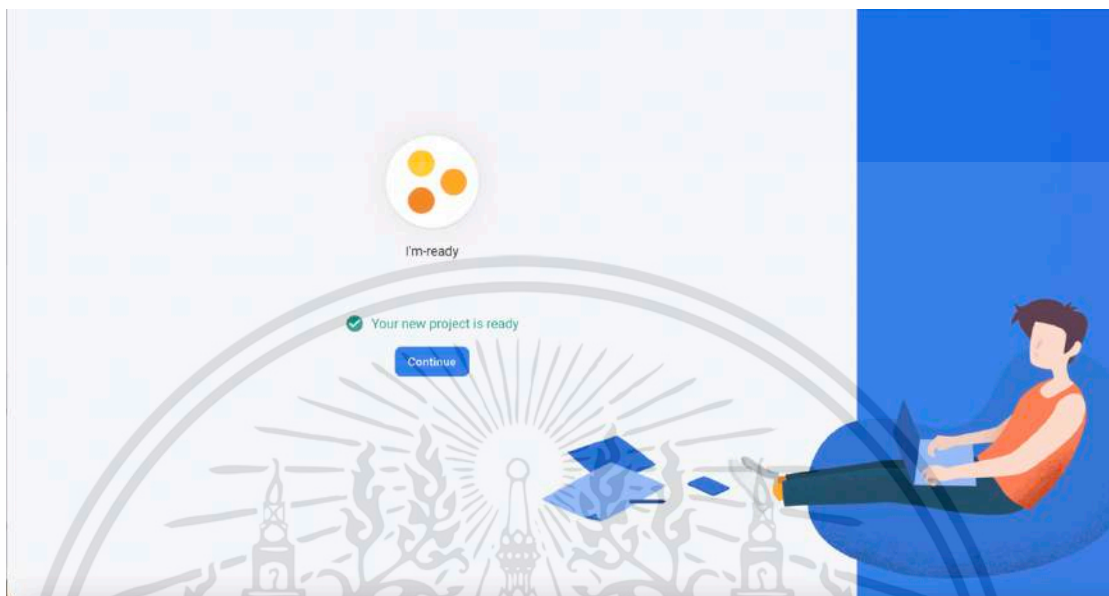
4. ตั้งชื่อโครงการเป็น I'm-ready



รูปที่ 2 กรอกรายละเอียดโครงการ

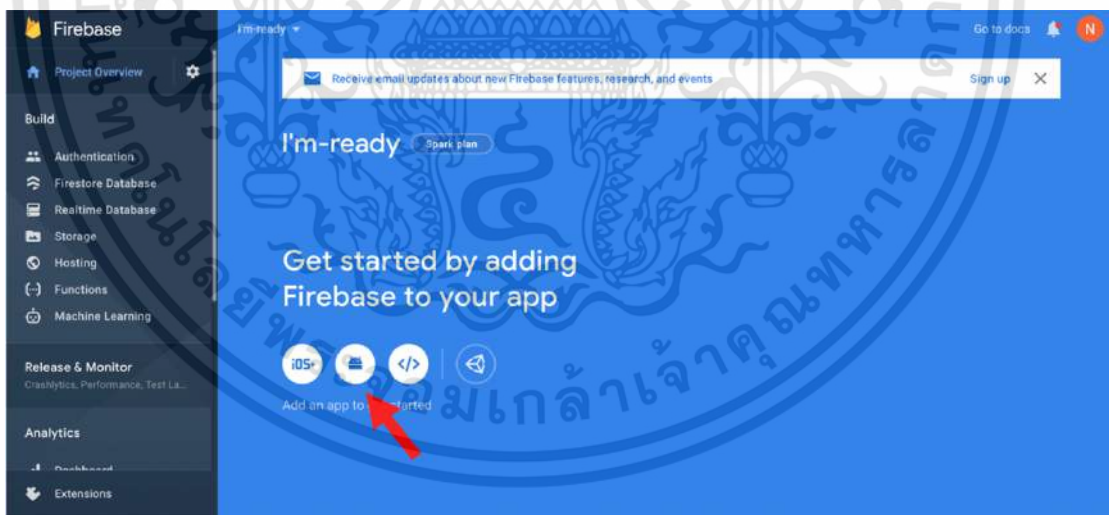
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. กด Continue เพื่อสร้างโครงการ



รูปที่ 3 สร้างโครงการ

6. กดเลือกประเภท Application เป็น android

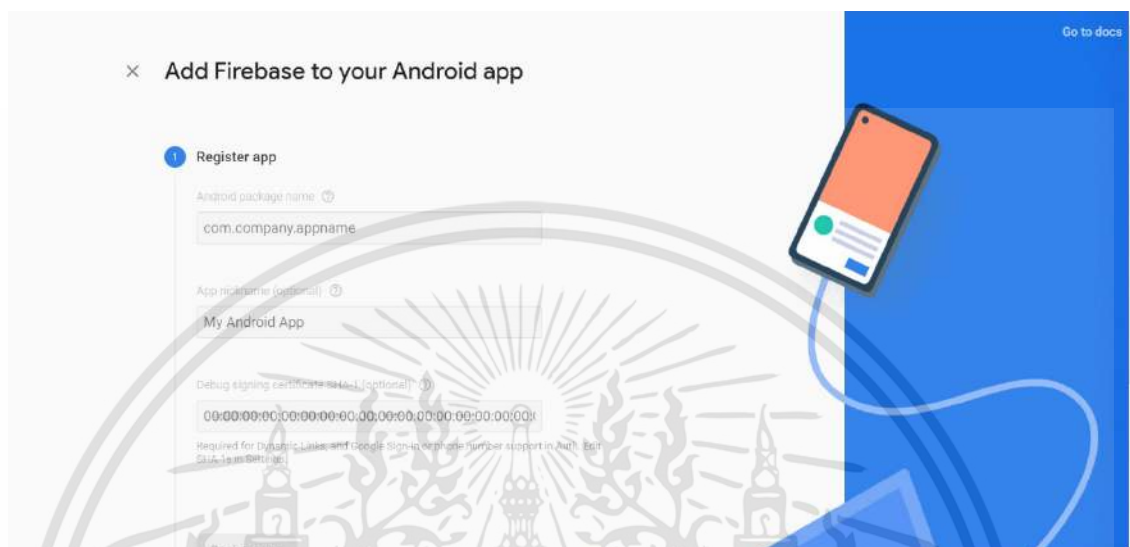


รูปที่ 4 การเลือกประเภทของ Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อ Firebase กับ Application

1. กรอกชื่อ package ของ application



รูปที่ 5 การลงทะเบียนเพื่อเชื่อม Firebase กับ Application

2. ดาวน์โหลดไฟล์ google-service.json แล้วนำไปไว้วางใน folder app ของแอปพลิเคชัน

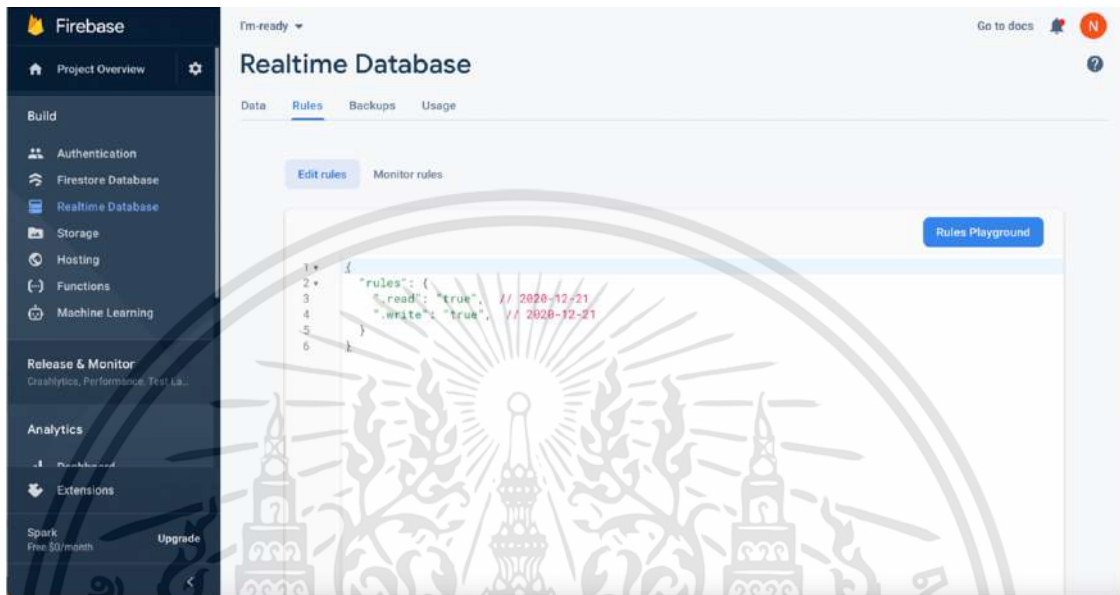


รูปที่ 6 ดาวน์โหลดไฟล์ google-service.json

3. กดถัดไป และทำการ Run รอการเชื่อมต่อ แล้วกดดำเนินการต่อในคอนโซล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

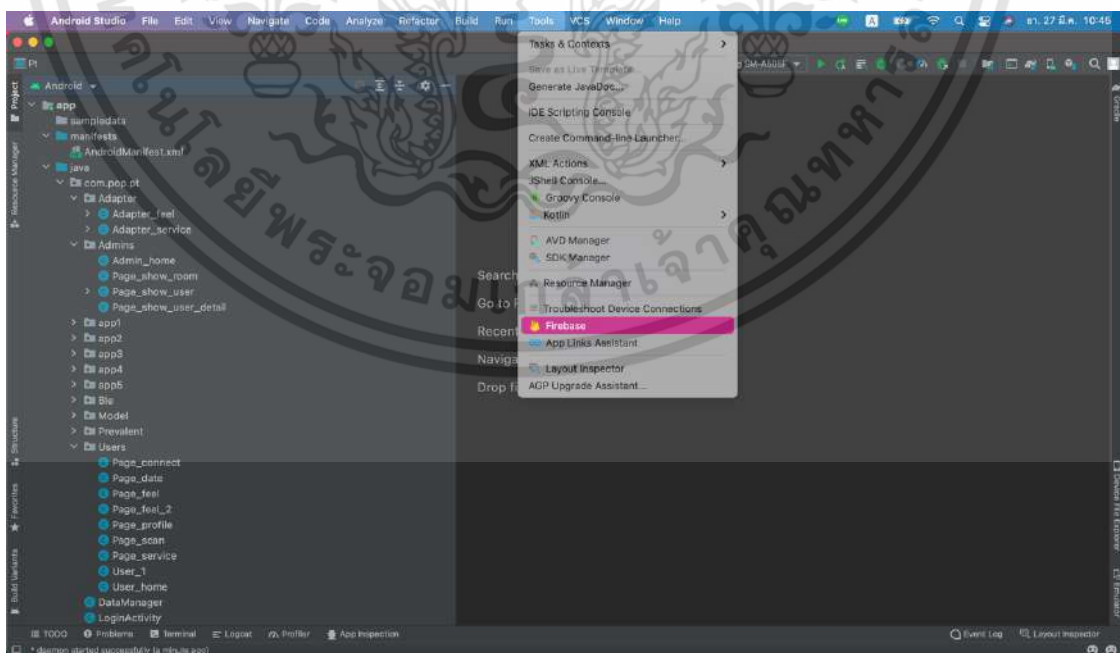
4. ใส่ตัวเลือกของ Realtime Database ทำการแก้ไขโค้ดเพื่อให้สามารถ Read และ Write ได้ และ กด Publish



รูปที่ 7 การตั้งค่าให้สามารถ Read และ Write ได้

การเชื่อมต่อ Application กับ Firebase

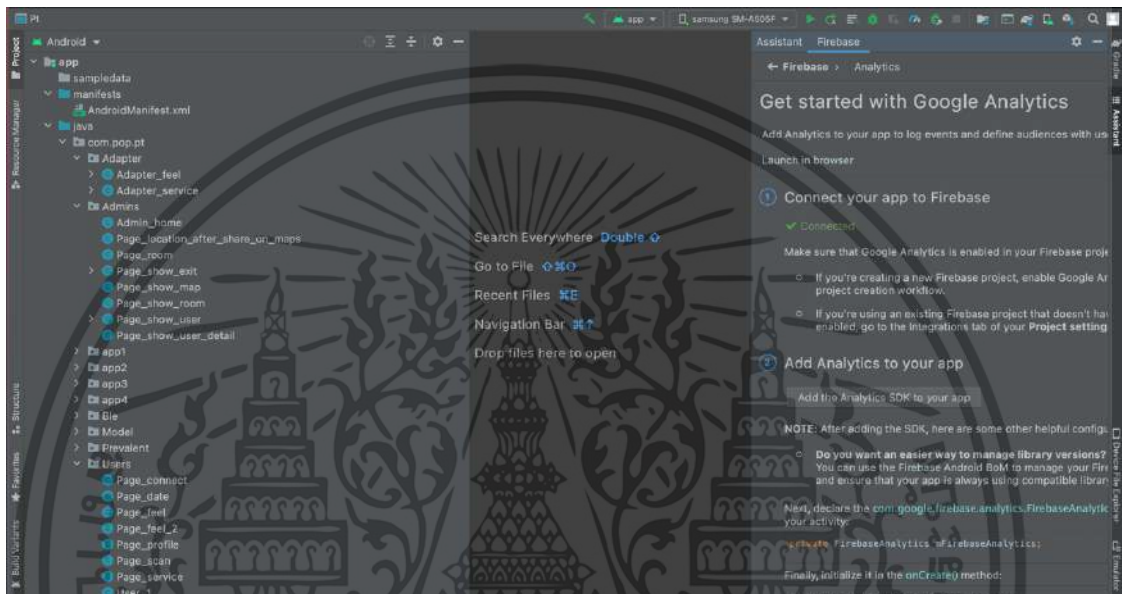
1. กด Tools และเลือก Firebase เพื่อทำการเชื่อมต่อกับ Firebase



รูปที่ 8 หน้าต่างของ Android Studio

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. จากนั้นทำการเพิ่ม Firebase โดยใช้เครื่องมือ Firebase Assistant ในโปรแกรม Android Studio โดยเครื่องมือ Firebase Assistant จะลงทะเบียนแอปพลิเคชันกับโครงการ Firebase และเพิ่มไฟล์ Firebase ปลั๊กอินและการอ้างอิงที่จำเป็นให้กับโครงการ Android ทั้งหมดสามารถทำได้จากภายในโปรแกรม Android Studio



รูปที่ 9 เครื่องมือ Firebase Assistant

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชุดคำสั่งในส่วนบ๊อคอนที่ทำงานในโหมด Background Service

1) การนำเข้า library

```
import android.annotation.SuppressLint;
import android.app.Service;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.le.ScanCallback;
import android.bluetooth.le.ScanResult;
import android.content.*;
import android.os.*;
import android.util.Log;
import android.widget.Toast;
import androidx.annotation.RequiresApi;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import java.util.ArrayList;
import java.util.Collections;
import java.util.HashMap;
import java.util.Iterator;
import java.util.List;
import java.util.Map;
import java.util.concurrent.ThreadLocalRandom;
```

2) การประกาศตัวแปรต่าง ๆ

```
public class BackgroundService extends Service {
    private FirebaseAuth mAuth;
    private BluetoothAdapter mBluetoothAdapter =
        BluetoothAdapter.getDefaultAdapter();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public DatabaseReference mDatabaseRef;
public FirebaseDatabase database;
public DatabaseReference myRef;
public static ArrayList<NSDevice> mDeviceList = new ArrayList<>();
public static HashMap<String, NSDevice> mDeviceMap = new HashMap<>();
private MainAdapter mAdapter;
private String mFilterString = "";
private FirebaseAuth firebaseAuth;
private String s_user_id;
private String name;
public Context context = this;
public Handler handler = null;
public static Runnable runnable = null;

```

3) ส่วนติดต่อกับฐานข้อมูล

```

@Override
public IBinder onBind(Intent intent) {
    return null;
}
@Override
public void onCreate() {
    Toast.makeText(this, "Service created!", Toast.LENGTH_LONG).show();
    database = FirebaseDatabase.getInstance();
    mDatabaseRef = database.getReference();
    scanLeDevice(true);
    firebaseAuth = FirebaseAuth.getInstance();
    try {
        s_user_id = firebaseAuth.getCurrentUser().getUid();
    } catch (NullPointerException e) {
        Toast.makeText(this, "User Id not available", Toast.LENGTH_SHORT).show();
    }
}

```

4) ส่วนรับค่า MAC Address, RSSI และระยะทางตัวที่ใกล้ที่สุด โดยได้กำหนดเงื่อนไขตัว MAC Address ของปีคอนกับ location เพื่อไปอัปเดตยังฐานข้อมูล

```

handler = new Handler();
runnable = new Runnable() {
    public void run() {
        Log.d("run service", "present");
        mDatabaseRef.child("Exit").addListenerForSingleValueEvent(new
ValueEventListener() {
@Override
public void onDataChange(DataSnapshot dataSnapshot) {
    Log.d("service runn :", "Device Size :" + mDeviceList.size());
    if (!mDeviceList.isEmpty()) {
        String addr = mDeviceList.get(0).getAddress();
        int rssi = mDeviceList.get(0).getRssi();
        double distance = RssiUtil.getDistance(rssi);
        Log.d("service runn :", "Send Data to Fire Base :" + mDeviceList.size());
        int randomNum = ThreadLocalRandom.current().nextInt(0, 1000 + 1);
        if (addr.equals("DA:E8:55:21:A9:B6")){
            dataSnapshot.getRef().child("address1").setValue(addr);
            dataSnapshot.getRef().child("location1").setValue("201 room");
            dataSnapshot.getRef().child("rssi1").setValue(rssi);
            dataSnapshot.getRef().child("distance1").setValue(distance);
            dataSnapshot.getRef().child("user_id").setValue(s_user_id);
            dataSnapshot.getRef().child("user_name").setValue(name);
        }else if (addr.equals("CD:55:4F:CF:DE:E3")){
            dataSnapshot.getRef().child("address1").setValue(addr);
            dataSnapshot.getRef().child("location1").setValue("202 room");
            dataSnapshot.getRef().child("rssi1").setValue(rssi);
            dataSnapshot.getRef().child("distance1").setValue(distance);
            dataSnapshot.getRef().child("user_id").setValue(s_user_id);
            dataSnapshot.getRef().child("user_name").setValue(name);
        }
    }
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}else if (addr.equals("F7:72:B7:A4:48:10")){
    dataSnapshot.getRef().child("address1").setValue(addr);
    dataSnapshot.getRef().child("location1").setValue("203 room");
    dataSnapshot.getRef().child("rssi1").setValue(rssi);
    dataSnapshot.getRef().child("distance1").setValue(distance);
    dataSnapshot.getRef().child("user_id").setValue(s_user_id);
    dataSnapshot.getRef().child("user_name").setValue(name);
}else if (addr.equals("FB:7C:06:0E:94:EA")){
    dataSnapshot.getRef().child("address1").setValue(addr);
    dataSnapshot.getRef().child("location1").setValue("exit");
    dataSnapshot.getRef().child("rssi1").setValue(rssi);
    dataSnapshot.getRef().child("distance1").setValue(distance);
    dataSnapshot.getRef().child("user_id").setValue(s_user_id);
    dataSnapshot.getRef().child("user_name").setValue(name);
}
}
refreshDeviceList();
}
@Override
public void onCancelled(DatabaseError databaseError) {
    Log.d("User", databaseError.getMessage());
}
});
handler.postDelayed(runnable, 1000);
}
};
handler.postDelayed(runnable, 1000);
}
@Override
public void onDestroy() {
}

```

```

@Override
public void onStart(Intent intent, int startid) {
}

@SuppressLint("NewApi")
private void scanLeDevice(final boolean start) {
    if (start) {
        if (mBluetoothAdapter != null) {
            mBluetoothAdapter.getBluetoothLeScanner().startScan(mScanCallback);
        }
    } else {
        if (mBluetoothAdapter != null) {
            mBluetoothAdapter.getBluetoothLeScanner().stopScan(mScanCallback);
        }
    }
}
}

```

5) ส่วนสแกนหาและรับอุปกรณ์บลูทูธที่พบ

```

private ScanCallback mScanCallback = new ScanCallback() {
    @RequiresApi(api = Build.VERSION_CODES.LOLLIPOP)
    @Override
    public void onScanResult(int callbackType, ScanResult result) {
        super.onScanResult(callbackType, result);
        String address = result.getDevice().getAddress();
        String name = result.getDevice().getName();
        NSDevice deviceEntity = new NSDevice();
        deviceEntity.setAddress(address);
        deviceEntity.setName(name);
        deviceEntity.setRssi(result.getRssi());

        ////////////////คำสั่งให้แสดงLogcat เพื่อตรวจสอบความถูกต้องของโค้ด////////////////////
        Log.d("run service", "str_new_name : " + result.getDevice().getName());
        Log.d("run service", "address : " + result.getDevice().getAddress());
        Log.d("run service", "str_new_rssi : " + String.valueOf(result.getRssi()));
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        mDeviceMap.put(deviceEntity.getAddress(), deviceEntity);
        int aa = 12;
        double distance = RssiUtil.getDistance(result.getRssi());
        Log.d("run service", "distance: " + String.valueOf(distance));
    }
    @Override
    public void onBatchScanResults(List<ScanResult> results) {
        super.onBatchScanResults(results);
    }
    @Override
    public void onScanFailed(int errorCode) {
        super.onScanFailed(errorCode);
    }
};
public void refreshDeviceList() {
    mDeviceList.clear();
    Iterator iter = mDeviceMap.entrySet().iterator();
    while (iter.hasNext()) {
        Map.Entry entry = (Map.Entry) iter.next();
        NSDevice val = (NSDevice) entry.getValue();
        mDeviceList.add(val);
    }
    Log.d("run service", "mDeviceList size after add : " +
String.valueOf(mDeviceList.size()));
    Collections.sort(mDeviceList, (arg0, arg1) ->
arg1.getRssi().compareTo(arg0.getRssi()));
}
    public static final int REQUEST_ENABLE_BT = 0x001;
}

```

ชุดคำสั่งในส่วนระบุตำแหน่งโดยใช้ Location provider

1) การนำเข้า library

```
import android.Manifest;
import android.content.BroadcastReceiver;
import android.content.ComponentName;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.ServiceConnection;
import android.content.SharedPreferences;
import android.content.pm.PackageManager;
import android.location.Location;
import android.net.Uri;
import android.os.Bundle;
import android.os.IBinder;
import android.preference.PreferenceManager;
import android.provider.Settings;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.localbroadcastmanager.content.LocalBroadcastManager;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DatabaseReference;
```

2) การประกาศตัวแปรต่าง ๆ

```
public class Page_new_3 extends AppCompatActivity implements
    SharedPreferences.OnSharedPreferenceChangeListener {
```

```

private FirebaseAuth firebaseAuth;
private String s_user_id;
private DatabaseReference mDatabase_1;
private static final int REQUEST_PERMISSIONS_REQUEST_CODE = 34;
private MyReceiver myReceiver;
private LocationUpdatesService_2 mService = null;
private boolean mBound = false;
private Button mRequestLocationUpdatesButton;
private Button mRemoveLocationUpdatesButton;
private final ServiceConnection mServiceConnection = new ServiceConnection() {
    @Override
    public void onServiceConnected(ComponentName name, IBinder service) {
        LocationUpdatesService_2.LocalBinder binder =
(LocationUpdatesService_2.LocalBinder) service;
        mService = binder.getService();
        mBound = true;
    }
    @Override
    public void onServiceDisconnected(ComponentName name) {
        mService = null;
        mBound = false;
    }
};
TextView text_1;
TextView text_2;
String s_text_1;
private Button b_action_1;

```

3) ขออัปเดตตำแหน่ง และขออนุญาตเข้าถึงตำแหน่ง

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    myReceiver = new MyReceiver();
    setContentView(R.layout.page_new_3);
    text_1 = (TextView) findViewById(R.id.text_1);
    text_2 = (TextView) findViewById(R.id.text_2);
    s_text_1 = text_1.getText().toString();
    if (s_text_1.matches("")) {
        return;
    }
    if (Utils_2.requestingLocationUpdates(this)) {
        if (!checkPermissions()) {
            requestPermissions();
        }
    }
}
@Override
protected void onStart() {
    super.onStart();
    PreferenceManager.getDefaultSharedPreferences(this)
        .registerOnSharedPreferenceChangeListener(this);
    mRequestLocationUpdatesButton = (Button)
findViewById(R.id.mRequestLocationUpdatesButton);
    mRemoveLocationUpdatesButton = (Button)
findViewById(R.id.mRemoveLocationUpdatesButton);
    b_action_1 = (Button) findViewById(R.id.b_action_1);
    b_action_1.setOnClickListener(new View.OnClickListener() {
@Override
        public void onClick(View v) {

```



```

        super.onPause();
    }
    @Override
    protected void onStop() {
        if (mBound) {
            unbindService(mServiceConnection);
            mBound = false;
        }
        PreferenceManager.getDefaultSharedPreferences(this)
            .unregisterOnSharedPreferenceChangeListener(this);
        super.onStop();
    }
    private boolean checkPermissions() {
        return PackageManager.PERMISSION_GRANTED ==
            ActivityCompat.checkSelfPermission(this,
                Manifest.permission.ACCESS_FINE_LOCATION);
    }
    private void requestPermissions() {
        boolean shouldProvideRationale =
            ActivityCompat.shouldShowRequestPermissionRationale(this,
                Manifest.permission.ACCESS_FINE_LOCATION);
        if (shouldProvideRationale) {
            Snackbar.make(
                findViewById(R.id.activity_main),
                R.string.permission_rationale,
                Snackbar.LENGTH_INDEFINITE)
                .setAction(R.string.ok, new View.OnClickListener() {

```

```

@Override
    public void onClick(View view) {
        ActivityCompat.requestPermissions(Page_new_3.this,
            new String[]{Manifest.permission.ACCESS_FINE_LOCATION},
            REQUEST_PERMISSIONS_REQUEST_CODE);
    }
})
.show();
} else {
    ActivityCompat.requestPermissions(Page_new_3.this,
        new String[]{Manifest.permission.ACCESS_FINE_LOCATION},
        REQUEST_PERMISSIONS_REQUEST_CODE);
}
}
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[]
permissions,
    @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if (requestCode == REQUEST_PERMISSIONS_REQUEST_CODE) {
        if (grantResults.length <= 0) {
        } else if (grantResults[0] == PackageManager.PERMISSION_GRANTED) {
            mService.requestLocationUpdates();
        } else {
            setButtonsState(false);
            Snackbar.make(findViewById(R.id.activity_main),
                R.string.permission_denied_explanation,
                Snackbar.LENGTH_INDEFINITE)
                .setAction(R.string.settings, new View.OnClickListener() {
                    @Override
                    public void onClick(View view) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Intent intent = new Intent();
intent.setAction(
    Settings.ACTION_APPLICATION_DETAILS_SETTINGS);
Uri uri = Uri.fromParts("package",
    getApplicationContext().getPackageName(), null);
intent.setData(uri);
intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
startActivity(intent);
}
})
.show();
}
}
}
private class MyReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        Location location =
intent.getParcelableExtra(LocationUpdatesService_2.EXTRA_LOCATION);
        if (location != null) {
            text_1.setText("" + location.getLatitude());
            text_2.setText("" + location.getLongitude());
            startService(new Intent(Page_new_3.this, MyService.class));
        }
    }
}
@Override
public void onSharedPreferenceChanged(SharedPreferences sharedPreferences,
String s) {
    if (s.equals(Utils_2.KEY_REQUESTING_LOCATION_UPDATES)) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

setButtonsState(sharedPreferences.getBoolean(Utils_2.KEY_REQUESTING_LOCATION_U
PDATES,
                false));
    }
}
private void setButtonsState(boolean requestingLocationUpdates) {
    if (requestingLocationUpdates) {
        mRequestLocationUpdatesButton.setEnabled(false);
        mRemoveLocationUpdatesButton.setEnabled(true);
        b_action_1.setEnabled(true);
    } else {
        mRequestLocationUpdatesButton.setEnabled(true);
        mRemoveLocationUpdatesButton.setEnabled(false);
        b_action_1.setEnabled(false);
    }
}
}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ส่วนของ User หรือผู้ใช้งาน

การลงทะเบียนและ Login เข้าสู่ระบบ

1) เปิดหน้าแอปพลิเคชันกดปุ่มลงทะเบียน (หากผู้ใช้งานยังไม่ได้ลงทะเบียน) เพื่อเข้าสู่หน้าลงทะเบียน

2) ผู้ใช้งานกรอกข้อมูลให้ครบทุกช่อง แล้วกดปุ่ม “ลงทะเบียน”

3) หากผู้ใช้งานมีบัญชีใช้งานอยู่แล้ว ทำการ Login เข้าสู่ระบบ โดยกรอกอีเมลและรหัสผ่านจากนั้นกดปุ่มเข้าสู่ระบบเพื่อเข้าสู่หน้าหลักผู้ใช้งาน

1.1 การแก้ไขข้อมูลส่วนตัว

1) เลือกปุ่มไอคอน 3 ขีดด้านบนมุมซ้ายของหน้าหลักผู้ใช้งาน

2) เลือกแก้ไขตรงช่องที่ต้องการแก้ไขข้อมูลจากนั้นกดบันทึกข้อมูล

3) ผู้ใช้งานสามารถอัปโหลดรูปโปรไฟล์ได้โดยกดที่รูปด้านบน เลือกรูปในแกลอรี และกดอัปรูป

1.2 การหาตำแหน่งของผู้ใช้งาน

การหาตำแหน่งของผู้ใช้งานมี 2 รูปแบบ คือ สแกนหาสัญญาณบีคอน (หาตำแหน่งภายในอาคาร) และโลเคชั่นระบุตำแหน่ง (หาตำแหน่งภายนอกอาคาร)

1) กดปุ่มสแกนที่หน้าหลักผู้ใช้งาน

2) จากนั้นจะมายังหน้าสแกน เมื่อผู้ปวยอยู่ตำแหน่งห้องใดในพื้นที่ที่กำหนดก็จะพบสัญญาณบีคอน และข้อมูลจะถูกส่งมายังหน้าแอปพลิเคชันผู้ดูแล

3) เมื่อพบสัญญาณบีคอน ให้กดปุ่มยืนยันตำแหน่งยังหน้าถัดไป

2. ส่วนของ Admin หรือผู้ดูแล

1) เมื่อเข้าสู่ระบบด้วยบัญชีของผู้ดูแล กดปุ่ม show data patient เพื่อดูข้อมูลทั่วไปของผู้ใช้งาน

2) กดปุ่ม show exit เพื่อดูรายชื่อคนที่ออกนอกพื้นที่ที่กำหนด

3) กดปุ่ม show room จะแบ่งเป็น 3 หน้า ได้แก่ หน้าดูตำแหน่งผู้ใช้งานภายในอาคาร หน้าดูตำแหน่งผู้ใช้งานภายนอกอาคาร และหน้าดูแผนผังของห้อง