



การพัฒนาแอปพลิเคชันติดตามแคลอรีการบริโภคอาหารบนอุปกรณ์พกพา

วีรฤติ ชัยยะสมบูรณ์

สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาวิศวกรรมเว็บและการพัฒนาแอปพลิเคชันบนอุปกรณ์พกพา
วิทยาลัยครีเอทีฟดีไซน์ แอนด์ เอ็นเตอร์เทนเมนต์เทคโนโลยี
มหาวิทยาลัยธุรกิจบัณฑิต
ปีการศึกษา 2566

THE DEVELOPMENT OF CALORIES TRACKER
APPLICATION ON MOBILE DEVICE

WEERAWUT CHAIYASOMBOON

A Thematic Paper Submitted in Partial Fulfillment of the
Requirements for the Degree of Master of Science Program
Department of Web Engineering and Mobile Application
Development, College of Creative and Entertainment Technology
Dhurakij Pundit University
Academic Year 2023



ใบรับรองสารนิพนธ์

วิทยาลัยศรีเอทีพีดีไอซ์ แอนด์ เอ็นเตอร์เทนเมนต์เทคโนโลยี มหาวิทยาลัยธุรกิจบัณฑิต
ปริญญา วิทยาศาสตรมหาบัณฑิต

หัวข้อสารนิพนธ์ การพัฒนาแอปพลิเคชันติดตามเคลอริการบริโภคอาหารบนอุปกรณ์พกพา
เสนอโดย วิรุฒิ ชัยชนะสมบูรณ์
สาขาวิชา วิศวกรรมเว็บและการพัฒนาแอปพลิเคชันบนอุปกรณ์พกพา
อาจารย์ที่ปรึกษา รองศาสตราจารย์ ดร.วราพร จิระพันธุ์ทอง

ได้พิจารณาเห็นชอบโดยคณะกรรมการสอบสารนิพนธ์แล้ว


.....ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.เจมจมา ชองรตถณี)


.....กรรมการและอาจารย์ที่ปรึกษาสารนิพนธ์
(รองศาสตราจารย์ ดร.วราพร จิระพันธุ์ทอง)


.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.วิลาวัลย์ อินทร์ชำนาญ)

วิทยาลัยศรีเอทีพีดีไอซ์ แอนด์ เอ็นเตอร์เทนเมนต์เทคโนโลยี


.....คณบดี
(ผู้ช่วยศาสตราจารย์ ดร.วิลาวัลย์ อินทร์ชำนาญ)
วันที่ ...!... เดือน ...มกราคม พ.ศ. 2560

หัวข้อสารนิพนธ์	การพัฒนาแอปพลิเคชันติดตามแคลอรีการบริโภคอาหารบนอุปกรณ์พกพา
ชื่อผู้เขียน	วีรุฒิ ชัยยะสมบูรณ์
อาจารย์ที่ปรึกษา	รองศาสตราจารย์ ดร.วราพร จิระพันธุ์ทอง
หลักสูตร	วิทยาศาสตรมหาบัณฑิต (วิศวกรรมเว็บและการพัฒนาแอปพลิเคชันบนอุปกรณ์พกพา)
ปีการศึกษา	2566

บทคัดย่อ

ทุกท่านคงเคยได้ยินประโยคที่กล่าวกันว่า “ความไม่มีโรค เป็นลาภอันประเสริฐ” ไม่ว่าจะยากดีมีจน ถ้าไม่ดูแลสุขภาพของตนเองแต่เนิ่น ๆ คนมั่งมีก็อาจจะกลายเป็นคนสิ้นเนื้อประดาตัวก็เป็นได้ การที่เราจะสามารถดูแลสุขภาพของเรานั้น นอกจากจะต้องมีการออกกำลังกายอย่างสม่ำเสมอแล้วนั้น อาหารที่เราบริโภคเข้าไปก็มีส่วนไม่มากนักน้อย ดังประโยคที่กล่าวว่า “You are what you eat” นี่จึงเป็นที่มาของการริเริ่มจัดทำและพัฒนาแอปพลิเคชัน “HealthMe” นี้ขึ้น เพื่อที่ผู้ใช้จะสามารถตรวจสอบรายการอาหารที่บริโภคเข้าไปในแต่ละมื้อ ว่ามีปริมาณแคลอรี ที่มากเกินไปจนจำเป็นจากปริมาณที่ผู้ใช้สามารถบริโภคในแต่ละวันเข้าไปหรือไม่? ซึ่งเป็นการตัดไฟตั้งแต่ต้นลม ซึ่งถ้าผู้ใช้แอปพลิเคชันนี้ ทำควบคู่กับการออกกำลังกายไปด้วย จึงจะเป็นประโยชน์ไม่มากนักน้อยต่อการดูแลสุขภาพของผู้ใช้เอง อนึ่งในแอปพลิเคชันนี้ มีข้อมูลแนะนำอาหารสุขภาพ และวิธีการออกกำลังกายให้ผู้ใช้ได้นำไปใช้ได้ด้วย รวมทั้งผู้ใช้สามารถที่จะดูข้อมูลอาหารที่ผู้ใช้ท่านอื่นบริโภคเข้าไปเพื่อเป็นแนวทางของตนเองได้

การพัฒนาแอปพลิเคชันนี้ใช้ SwiftUI framework ต่อกับฐานข้อมูล Firebase (ใช้ Authentication, Cloud Firestore, และ Cloud Storage) ซึ่งจะสนับสนุนเฉพาะ iPhone ที่ Run บน iOS 16.4 ขึ้นไป ผู้พัฒนานำเสนอแนวคิดในการพัฒนาบน iOS platform และ Firebase โดยมีการแสดงเหตุผลและที่มาที่ไปในบทที่ 2 ส่วนบทที่ 3 เป็นการนำเสนอ Requirement Analysis ได้นำเสนอความต้องการของระบบทั้ง Functional Requirement และ Non-Functional Requirement แสดง Use Case Diagram/Use Case description, Activity Diagram, Class Diagram, ER Diagram เพื่ออธิบายระบบที่จะสร้างขึ้น รวมทั้งมีการออกแบบ UX/UI ของระบบโดยใช้ Figma

ต่อเมื่อพัฒนาแอปพลิเคชันแล้วเสร็จ ในบทที่ 4 ก็นำเสนอ UI ที่ได้สร้างขึ้น โดยแสดงเป็นแผนภาพอธิบายส่วนต่างๆในหน้า UI นั้น ๆ รวมทั้งมีการนำเสนอฐานข้อมูลที่ได้สร้างขึ้นบน Firebase ตามที่ได้ออกแบบไว้ หลังจากนั้นมีการเขียน Test Cases และทำการทดสอบระบบตาม Test Cases ที่สร้างขึ้น ซึ่งได้มีการสรุปผลทุก Test Cases ไว้ด้วยแล้ว นอกจากนี้ ยังมีการอัด VDO Demo พร้อมแบบสำรวจ (Google Form) ส่งให้ผู้ใช้เพื่อขอความเห็นในด้าน UI และ Function การใช้งานของแอปพลิเคชันที่ได้พัฒนาขึ้นนี้ ท้ายสุด มีการสรุปผลการวิจัย และข้อเสนอแนะในบทที่ 5

คำสำคัญ: แคลอรี, สุขภาพ, ออกกำลังกาย, แอปพลิเคชันมือถือ, ไอโฟน, iOS, Firebase



Thematic Paper Title	THE DEVELOPMENT OF CALORIES TRACKER APPLICATION ON MOBILE DEVICE
Author	Weerawut Chaiyasomboon
Thematic Paper Advisor	Associate Professor Waraporn Jirapanthong, Ph.D.
Program	Master of Science (Web Engineering and Mobile Application Development)
Academic Year	2023

ABSTRACT

Everyone used to heard “No illness is the best luck”, whatever rich or poor, you maybe poor one day if you got sick and need to spend all your money to cure yourself. So, you need to take care yourself beforehand by be good healthy you need to do some exercise, however the food you eat is also important as saying “You are what you eat”. This is the first reason we are developing this app, “HealthMe” app. This app can help users tracked calories what they are eating compare to calories/day they can consume. So, if users use this app together with doing exercise it will be the best outcome. Additionally, this app also has recommended healthy food and also recommended exercises workout, so users can follow them.

The development of this app we are using SwiftUI framework connecting with Firebase (Authentication/Cloud Firestore/Cloud Storage), support on all iPhone running iOS16.4 and above. There are the reasons for building on iOS platform and Firebase on Chapter 2.

On Chapter 3 is the Requirement Analysis, we present both Functional and Non-Functional requirements, together with Use case diagram/Use case description, Activity Diagram, Class Diagram, and ER Diagram to explain about this app. Before end we also show UX/UI we did with Figma.

After development was done, we showing all UI with explanation about components on Chapter 4, including Firebase implementation followed the design. Then we develop the Test Cases and test the app followed them and concluded the results of testing. Furthermore, we record VDO demo of the app and send to users with questionnaire about UI and Functions of the app. Finally, Chapter 5 concluded the development with additional recommendation to do next.

Keywords: Calories, Health, Exercise, Mobile application, iPhone, iOS, Firebase



กิตติกรรมประกาศ

สารนิพนธ์เรื่องนี้ “การพัฒนาแอปพลิเคชันติดตามแคลอรีการบริโภคอาหารบนอุปกรณ์พกพา” จะไม่สามารถสำเร็จลุล่วงถ้าไม่ได้ครูบาอาจารย์ทั้งของมหาวิทยาลัยธุรกิจบัณฑิตเอง และทั้งครูบาอาจารย์ตามคลิปสอนต่างๆตามสื่ออินเทอร์เน็ต ทั้งหมดได้ประสิทธิประสาทผู้เขียนให้มีความรู้มาทำการพัฒนาแอปพลิเคชันนี้ได้ลุล่วงไปได้อย่างสมบูรณ์

ขอขอบพระคุณรองศาสตราจารย์ ดร.วราพร จิระพันธุ์ทอง อาจารย์ที่ปรึกษาสารนิพนธ์นี้ ได้กรุณาแนะนำจนผู้เขียนสามารถทำสารนิพนธ์นี้ได้สำเร็จลุล่วงได้ด้วยดี ขอขอบคุณเพื่อน ๆ ทุกท่านที่ให้คำแนะนำช่วยเหลือ

สุดท้ายขอขอบพระคุณคุณพ่อและคุณแม่ของผู้เขียน ที่ทำให้ผู้เขียนมาเรียนได้ตรงจุดนี้เป็นอย่างดี

วีรจุมิ ชัยยะสมบูรณ์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ฌ
สารบัญภาพ.....	ฎ
บทที่	
1. บทนำ.....	1
1.1 ที่มาและความสำคัญ.....	1
1.2 ปัญหา.....	1
1.3 วัตถุประสงค์.....	1
1.4 แผนการดำเนินงาน.....	2
2. แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง.....	3
2.1 การพัฒนาแอปพลิเคชันบนอุปกรณ์มือถือในปัจจุบัน.....	3
2.2 iOS platform.....	4
2.3 Native (SwiftUI) เปรียบเทียบ Cross platform.....	5
2.4 SwiftUI เทียบกับ UIKit.....	5
2.5 Firebase platform.....	5
2.6 แอปพลิเคชันที่เกี่ยวข้อง (พัฒนาโดยผู้พัฒนาคนไทยที่มีใน App Store).....	6
2.7 สรุปรุ่นมือถือที่ใช้ในการพัฒนา.....	8
2.8 ทฤษฎี UX/UI.....	8
3. ระเบียบวิธีวิจัย.....	9
3.1 ความต้องการของระบบ.....	9
3.2 Use cases.....	10
3.3 Activity Diagram.....	17
3.4 Class Diagram.....	19
3.5 ER Diagram.....	21
3.6 UX/UI Design.....	23

สารบัญ (ต่อ)

บทที่	หน้า
4. ผลการวิจัย.....	28
4.1 User Interface สำหรับแอปพลิเคชันที่พัฒนาสำเร็จ.....	28
4.2 แสดงฐานข้อมูล Firebase สำหรับแอปพลิเคชันที่พัฒนาสำเร็จ.....	34
4.3 แสดงการทดสอบระบบของแอปพลิเคชันที่พัฒนาสำเร็จ.....	39
4.4 แสดงผลการสำรวจความพึงพอใจของผู้ใช้.....	79
5. สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ.....	84
5.1 สรุปผลการวิจัยและอภิปราย.....	84
5.2 ข้อเสนอแนะ.....	84
บรรณานุกรม.....	86
ประวัติผู้เขียน.....	88

สารบัญตาราง

ตารางที่	หน้า
3.1 ความต้องการตามการใช้งาน (Functional requirement).....	9
3.2 Use case description UC01.....	11
3.3 Use case description UC02.....	12
3.4 Use case description UC03.....	12
3.5 Use case description UC04.....	13
3.6 Use case description UC05.....	13
3.7 Use case description UC06.....	14
3.8 Use case description UC07.....	14
3.9 Use case description UC08.....	15
3.10 Use case description UC09.....	15
3.11 Use case description UC010.....	16
3.12 Use case description UC011.....	16
3.13 Use case description UC012.....	17
3.14 รายละเอียดแต่ละ Class.....	20
3.15 รายละเอียดแต่ละ Entity.....	22
4.1 รายละเอียด Test cases ทั้งหมด.....	39
4.2 TC01 การลงทะเบียนผู้ใช้ใหม่ (Create new user account) – ใส่ข้อมูลครบและ..... ถูกต้อง	43
4.3 TC02 การลงทะเบียนผู้ใช้ใหม่ (Create new user account) – ใส่ข้อมูลไม่ครบ.....	44
4.4 TC03 การลงทะเบียนผู้ใช้ใหม่ (Create new user account) – ใส่ข้อมูลไม่ถูกต้อง.....	46
4.5 TC04 การเข้าใช้งาน (Login) – ใส่ข้อมูลครบและถูกต้อง.....	47
4.6 TC05 การเข้าใช้งาน (Login) – ใส่ข้อมูลไม่ครบ.....	48
4.7 TC06 การเข้าใช้งาน (Login) – ใส่ข้อมูลไม่ถูกต้อง.....	49
4.8 TC07 เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) – ใส่ข้อมูลครบ.....	51
4.9 TC08 เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) – ใส่ข้อมูลไม่ครบ..	52
4.10 TC09 เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) – โดยกดปุ่ม..... Search	53

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
4.11 TC10 แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) – ใส่ข้อมูลครบ...	54
4.12 TC11 แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) – ใส่ข้อมูลไม่ครบ	55
4.13 TC12 แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) – โดยกดปุ่ม..... Search	56
4.14 TC13 ลบข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Delete food calories info).....	57
4.15 TC14 แสดงข้อมูลอาหารที่เพิ่มไว้แล้วในหน้า Calories Tracker view.....	58
4.16 TC15 แสดงข้อมูลแคลอรีรวมแต่ละวันย้อนหลัง 7 วัน.....	59
4.17 TC16 แสดงข้อมูลแคลอรีและอาหารที่ผู้ใช้ทั้งหมดเพิ่มไว้ในระบบ.....	61
4.18 TC17 แสดงข้อมูลอาหารสุขภาพที่แนะนำ พร้อมรายละเอียด.....	62
4.19 TC18 แสดงข้อมูลการออกกำลังกายที่แนะนำ พร้อมรายละเอียด.....	63
4.20 TC19 แสดงข้อมูลผู้ใช้ (Profile view).....	64
4.21 TC20 แสดงหน้า Edit Profile view – ใส่ข้อมูลครบและถูกต้อง.....	65
4.22 TC21 แสดงหน้า Edit Profile view – ใส่ข้อมูลไม่ครบ.....	66
4.23 TC22 แสดงหน้า Edit Profile view – ใส่ข้อมูลไม่ถูกต้อง.....	67
4.24 TC23 แสดงลำดับผู้ใช้ตามแต้มสะสม (Show Leaderboard).....	68
4.25 TC24 แสดงหน้าอาหารที่ผู้ใช้แต่ละคนเพิ่มไว้.....	69
4.26 TC25 ทดสอบใส่ค่าน้ำหนักที่ไม่สามารถแปลงเป็นตัวเลขได้ ในหน้า..... CreateAccountView	70
4.27 TC26 ทดสอบใส่ค่าความสูงที่ไม่สามารถแปลงเป็นตัวเลขได้ ในหน้า..... CreateAccountView	71
4.28 TC27 ทดสอบใส่ค่า Username เป็นภาษาไทยในหน้า CreateAccountView.....	72
4.29 TC28 ทดสอบใส่ค่า Calories field ที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า..... AddCaloriesView	74
4.30 TC29 ทดสอบใส่ค่า Calories field ที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า..... EditCaloriesView	75

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
4.31 TC30 ทดสอบใส่ค่า Name field เป็นภาษาไทยในหน้า AddCaloriesView.....	76
4.32 TC31 ทดสอบใส่ค่า Name field เป็นภาษาไทยในหน้า EditCaloriesView.....	77
4.33 TC32 ทดสอบเลือกอาหารภาษาไทยในหน้า Select your calories โดยกดปุ่ม..... Search ในหน้า AddCaloriesView หรือ EditCaloriesView	78
4.34 TC33 Sign out ออกจากระบบ.....	79
4.35 ข้อเสนอแนะ หรือความเห็นอื่นๆ.....	82

สารบัญภาพ

ภาพที่	หน้า
1.1 แผนการดำเนินงาน.....	2
2.1 Global mobile app revenue (2021-2025).....	3
2.2 แคลอรี ไดอารี่ แอป.....	6
2.3 CalTracker - สมุดบันทึกแคลอรี แอป.....	7
2.4 จด Calorie แอป.....	7
2.5 ตารางเปรียบเทียบ Functions.....	8
3.1 Use case diagram.....	10
3.2 Activity diagram.....	17
3.3 Class diagram.....	19
3.4 ER diagram.....	21
3.5 Customer journey.....	23
3.6 Login screen and Register screen.....	24
3.7 Calories Tracker screen and Total screen.....	24
3.8 Add Food screen and Edit Food screen.....	25
3.9 Food Menu screen and Food Detail screen.....	25
3.10 Exercises screen and Exercise Detail screen.....	26
3.11 Profile screen and Edit Profile screen.....	26
3.12 Leaderboard screen and Redeem screen.....	27
4.1 Login view.....	28
4.2 Create Account view.....	29
4.3 Calories Tracker view.....	29
4.4 Add Calories view/Select your calories view.....	30
4.5 Calories Tracker view/Edit Calories view.....	30
4.6 All Food Calories view/Calories Stat view.....	31
4.7 Food Menu view/Food Menu Detail view.....	31
4.8 Food Menu Detail view (continue)	32
4.9 Exercise Menu view/Exercise Detail View	32
4.10 Profile view.....	33

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
4.11 Edit Profile view.....	33
4.12 Leaderboard view/Food of user view.....	34
4.13 Firebase Authentication(Email/Password)	35
4.14 exercises collection.....	35
4.15 foodCalValue collection.....	36
4.16 foodCalories collection.....	37
4.17 foods collection.....	37
4.18 users collection.....	38
4.19 Cloud Storage.....	39
4.20 เพศของผู้ตอบแบบสอบถาม.....	80
4.21 อายุของผู้ตอบแบบสอบถาม.....	80
4.22 ความพึงพอใจต่อ User Interface.....	81
4.23 ความพึงพอใจต่อ Functions ที่มี.....	81
4.24 ตัวอย่างข้อเสนอแนะ หรือความเห็นอื่นๆ.....	83
4.25 ตัวอย่างข้อเสนอแนะ หรือความเห็นอื่น ๆ (ต่อ)	83

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ทุกท่านคงเคยได้ยินประโยคที่กล่าวกันว่า “ความไม่มีโรค เป็นลาภอันประเสริฐ” ไม่ว่าจะยากดีมีจน ถ้าไม่ดูแลสุขภาพของตนเองแต่เนิ่น ๆ คนมั่งมีก็อาจจะกลายเป็นคนสิ้นเนื้อประดาตัวก็เป็นได้ การที่เราจะสามารถดูแลร่างกายของเรานั้น นอกจากจะต้องมีการออกกำลังกายอย่างสม่ำเสมอแล้วนั้น อาหารที่เราบริโภคเข้าไปก็มีส่วนไม่มากก็น้อย ดังประโยคที่กล่าวว่า “You are what you eat” นี่จึงเป็นที่มาของการริเริ่มจัดทำและพัฒนาแอปพลิเคชัน “HealthMe” นี้ขึ้น เพื่อให้ผู้ใช้จะสามารถตรวจสอบรายการอาหารที่บริโภคเข้าไปในแต่ละมื้อ ว่ามีปริมาณแคลอรี ที่มากเกินไปจนเกินความจำเป็นจากปริมาณที่ผู้ใช้สามารถบริโภคในแต่ละวันเข้าไปหรือไม่? ซึ่งเป็นการตัดไฟตั้งแต่ต้นลม ซึ่งถ้าผู้ใช้แอปพลิเคชันนี้ ทำควบคู่กับการออกกำลังกายไปด้วย จึงจะเป็นประโยชน์ไม่มากก็น้อยต่อการดูแลสุขภาพของผู้ใช้เอง อนึ่งในแอปพลิเคชันนี้ มีข้อมูลแนะนำอาหารสุขภาพ และวิธีการออกกำลังกายให้ผู้ใช้ได้นำไปใช้ได้ด้วย รวมทั้งผู้ใช้สามารถที่จะดูข้อมูลอาหารที่ผู้ใช้งานอื่นบริโภคเข้าไปเพื่อเป็นแนวทางของตนเองได้

1.2 ปัญหา

1.2.1 มีเครื่องมือน้อยที่ใช้ในการติดตามการบริโภคอาหารเข้าไปในแต่ละมื้อว่ามีแคลอรี มากเกินความจำเป็นของผู้ใช้ในแต่ละวันหรือไม่? อย่างไร?

1.2.2 เพื่อความสะดวกของผู้ใช้ในการใช้งาน ควรมีเครื่องมือแบบนี้เป็นแอปพลิเคชันในมือถือ ผู้ใช้จะสามารถใช้งานได้ตลอดเวลา

1.2.3 แอปพลิเคชันที่ใช้งานง่าย สามารถใช้งานโดยผู้ทุพพลภาพ มีไม่มาก

1.2.4 แอปพลิเคชันที่ไม่มีค่าใช้จ่าย และมี Features ครบตามความต้องการผู้ใช้น้อย ส่วนใหญ่เป็นแบบ In-app purchase

1.2.5 ไม่มีที่รวบรวมอาหารที่ดีต่อสุขภาพ และวิธีการออกกำลังกายที่เหมาะสม

1.2.6 ไม่มีกลุ่มที่ผู้ใช้ที่มีความสนใจด้านการรักษาสุขภาพสามารถเข้าไปแชร์ข้อมูลอาหารที่ตนบริโภคเข้าไปในแต่ละมื้อ

1.3 วัตถุประสงค์

1.3.1 พัฒนาเครื่องมือในการติดตามการบริโภคอาหารเข้าไปในแต่ละมื้อว่ามีแคลอรี มากเกินความจำเป็นของผู้ใช้ในแต่ละวันหรือไม่? อย่างไร?

1.3.2 พัฒนาเครื่องมือในในรูปแบบแอปพลิเคชันมือถือ โดยต้องสามารถใช้งานง่ายได้กับผู้ใช้ทุกระดับ และเป็นแอปพลิเคชันที่ใช้งานได้แบบไม่มีค่าใช้จ่าย

1.3.3 แอปพลิเคชันนี้สามารถแนะนำรายการอาหารสุขภาพ และแนะนำวิธีในการออกกำลังกายแบบต่างๆให้ผู้ใช้ได้

1.3.4 ผู้ใช้สามารถดูข้อมูลอาหารของผู้ใช้ท่านอื่นที่บริโภคไปได้

1.3.5 แอปพลิเคชันสามารถเก็บข้อมูลผู้ใช้ในฐานะข้อมูลกลาง ซึ่งผู้ใช้สามารถเข้าถึงข้อมูลของตนเองได้ โดยข้อมูลไม่มีการสูญหาย ถ้ามีการเปลี่ยนอุปกรณ์มือถือหรือใช้ในอุปกรณ์อีกเครื่อง

1.4 แผนการดำเนินงาน

แผนการดำเนินงาน

ลำดับ	รายการ	สิงหาคม 2566				กันยายน 2566				ตุลาคม 2566				พฤศจิกายน 2566			
		W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4
1	Requirement analysis																
2	UX/UI implementation																
3	System design																
4	Coding																
5	Testing																

รูปที่ 1.1 แผนการดำเนินงาน

บทที่ 2

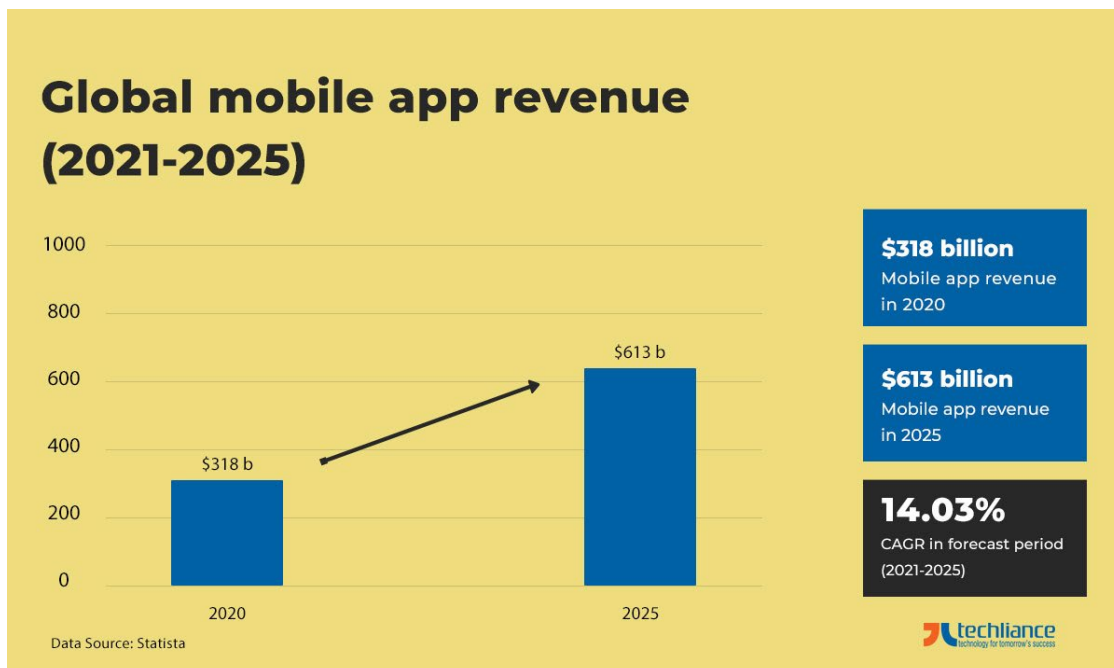
แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง

การพัฒนาแอปพลิเคชันในงานวิจัยนี้ จะพัฒนาสำหรับอุปกรณ์มือถือคือ iPhone (iOS platform) เป็นหลัก (โดยใช้ SwiftUI framework ในการพัฒนา) โดยจะมีการต่อกับฐานข้อมูล Firebase ซึ่งในบทนี้ ผู้เขียนจะแสดงเหตุผลว่าทำไมต้องพัฒนาใน Platform นี้ รวมถึงแสดงแอปพลิเคชันที่มี Functions คล้ายกันของผู้พัฒนาในไทยที่มีอยู่ใน Apple App Store พอเป็นแนวทางในการพัฒนาแอปพลิเคชันนี้ต่อไป

อย่างไรก็ตาม Apple platform นั้นประกอบไปด้วย iPadOS และ macOS ด้วย นอกเหนือจาก iOS ที่เราจะทำการพัฒนาในงานวิจัยนี้ ซึ่งที่ผู้เขียนเลือกที่จะพัฒนาแอปพลิเคชันนี้บน iOS platform เป็นเบื้องต้นนั้น เพราะเมื่อเราพัฒนาบน iOS เรียบร้อยแล้วนั้น ก็ไม่ยากที่จะพัฒนาต่อไปยังบน iPadOS และ macOS เนื่องจากว่า SwiftUI framework ที่เราจะใช้ในการพัฒนานั้นสามารถพัฒนาต่อยอดเป็น Apple Cross Platform Device ได้ไม่ยาก

2.1 การพัฒนาแอปพลิเคชันบนอุปกรณ์มือถือในปัจจุบัน

ในปัจจุบันมี Mobile Operating System ในตลาด ใหญ่ ๆ อยู่ 2 ค่าย คือ Android Platform และ iOS Platform และถ้าพิจารณาจากภาพที่ 2.1 ด้านล่าง จะเห็นว่ารายรับของ Mobile App จะมีมูลค่า \$613 billion ในปี 2025 นี้จึงเป็นอีกเหตุผลที่ผู้เขียนเลือกที่จะพัฒนาแอปพลิเคชันนี้ในอุปกรณ์พกพาหรือมือถือนั่นเอง (นอกจาก Use cases ที่มีแสดงว่าควรพัฒนาเป็นแอปพลิเคชันมือถือ ซึ่งจะได้กล่าวถึงในบทที่ 3)



รูปที่ 2.1 Global mobile app revenue (2021-2025) [1]

สำหรับเครื่องมือและภาษาที่นิยมใช้ในปัจจุบันมีดังต่อไปนี้

- (1) Android Native platform พัฒนาโดยใช้ภาษา Kotlin/Java
- (2) iOS Native platform พัฒนาโดยใช้ภาษา Swift/Objective-c
- (3) Cross platform พัฒนาโดยใช้ภาษา React-Native
- (4) Cross platform พัฒนาโดยใช้ภาษา Flutter(Dart)
- (5) Cross platform อื่นๆ เช่น Xamarin, Ionic, etc.

แต่สำหรับงานวิจัยนี้จะทำการพัฒนาบน iOS platform โดยใช้ SwiftUI framework ดังที่กล่าวไปข้างต้น โดยจะแสดงผลสนับสนุนในหัวข้อต่อ ๆ ไป

2.2 iOS platform [2]

2.2.1 ผลตอบแทนการลงทุน (ROI – Return On Investment) – Apple เป็นผู้นำในเรื่องลูกค้าที่ยอมจ่ายซื้อ App มากกว่า Android platform

2.2.2 การพัฒนา App มีประสิทธิภาพมากกว่า – ผู้พัฒนาสำหรับ Android platform ต้องคำนึงถึงอุปกรณ์ Android ที่มีหลากหลาย แต่สำหรับ Apple platform ผู้พัฒนาเพียงพัฒนาสำหรับ iPhone และ iPad เท่านั้น นอกจากนี้ตั้งแต่ปี 2020, Apple ผลิต Mac ที่เป็น Apple Silicon chip ซึ่งทำให้ App native iOS สามารถทำงานบน Mac ได้ด้วย ซึ่งทำให้การพัฒนา App ใน Apple platform มีต้นทุนต่ำและใช้เวลาน้อยกว่าถ้าเทียบกับ Android platform ที่มีความหลากหลายของอุปกรณ์

2.2.3 ความปลอดภัย - มีการพยากรณ์ว่าภายในปี 2021 ภัยทางไซเบอร์ (Cybercrime) จะมีมูลค่าถึง 6 พันล้านดอลลาร์ ต่อปี ซึ่งทำให้ผู้ใช้ระวังและกังวลเรื่องความปลอดภัยใน App ที่เรียกใช้ ข้อมูลต่างๆเกี่ยวกับการจ่ายเงินและข้อมูลส่วนบุคคล สำหรับเรื่องเหล่านี้ Apple ขึ้นชื่อมาแต่เนิ่นนานเรื่องความปลอดภัยในอุปกรณ์ต่างๆของตน รวมทั้งผู้พัฒนา App ใน Apple platform ต้องมีการส่ง App ที่ตนพัฒนาเพื่อ Review ก่อน ส่งถึงมือผู้ใช้ ทำให้ผู้ใช้เชื่อมั่นใน App ต่างๆของ Apple platform มากกว่า

2.2.4 การใช้งานและประสบการณ์ผู้ใช้ – Apple มีข้อกำหนดด้านรูปแบบของ App มากกว่า Android ซึ่งทำให้รูปแบบส่วนต่อกับผู้ใช้มีคุณภาพมากกว่า ซึ่ง iOS platform ทำให้ผู้ใช้งานมั่นใจได้ว่า App จะใช้งานได้ง่ายและสร้างประสบการณ์ให้ผู้ใช้ได้เป็นอย่างดี ซึ่งทำให้บริษัทหลายแห่งทำ App ใน Apple platform เพื่อเป็นหน้าเป็นตาให้กับบริษัท

2.2.5 ชื่อเสียงของ Apple – Apple เป็นบริษัทที่มีชื่อเสียงระดับโลก ทำให้ Apps ใน App Store มีชื่อเสียงอยู่ในระดับสูงเมื่อเทียบกับ Android apps, iPhone รุ่นแรกทำให้เกิดอุตสาหกรรมใหม่สำหรับ Smartphone รวมทั้งตอนที่ Apple นำเสนอ iPad ทำให้มีการผลิต Tablets ต่างๆออกมาแข่ง จึงกล่าวได้ว่า Apple มีชื่อเสียงในด้านนวัตกรรม ซึ่งผู้ใช้งานทั้งหมดทราบดีสำหรับสิ่งเหล่านี้

2.3 Native (SwiftUI) เปรียบเทียบ Cross platform [3]

2.3.1 มีสมรรถนะดีกว่า - เนื่องด้วยเราพัฒนาด้วย Native language (Swift) และ Native framework (SwiftUI) ทำให้ App มีสมรรถนะดีกว่า เร็วกว่า และตอบสนองต่อผู้ใช้ได้ดีกว่า การพัฒนาด้วย Cross platform ที่ต้องแปลง codes เป็น Native

2.3.2 มีความปลอดภัยดีกว่า - เพราะ Native app เข้าถึง Built-in security features

2.3.3 UX มีคุณภาพ - Native app ให้ประสบการณ์กับผู้ใช้ได้ดีกว่า เพราะ UI/UX ทำตาม Design guidelines จาก Native OS platform ซึ่งทำให้ App มีความเป็นธรรมชาติในการใช้งานได้มากกว่า

2.3.4 สามารถเข้าถึงทุกส่วนของ Device - Native app สามารถเข้าถึง Hardware/Services ทุกส่วนของ Device ทำให้ App มี Features ทุกอย่างที่มี ซึ่งเพิ่มประสบการณ์ของผู้ใช้

2.3.5 มีข้อผิดพลาดที่น้อย - Native app สร้างด้วย Native SDKs ทำให้การพัฒนามีข้อผิดพลาดที่น้อยกว่า ผู้พัฒนาจะได้รับ SDKs ที่มีการแก้ไขและ/หรือเพิ่มคุณสมบัติได้ทันที

2.4 SwiftUI เทียบกับ UIKit [4]

ก่อนหน้า SwiftUI framework, การพัฒนาแอปพลิเคชันบน iOS Platform ใช้ UIKit framework ในการพัฒนา และต่อไปนี้เป็นเหตุผลที่ควรเลือกใช้ SwiftUI framework over UIKit framework

2.4.1 ผู้พัฒนาแอปด้วย SwiftUI สามารถพัฒนาแอปพลิเคชันได้อย่างง่ายและรวดเร็วมากกว่า UIKit เนื่องด้วย SwiftUI เป็นแบบ Declarative syntax ผู้พัฒนาสามารถกำหนด UI ได้เลย โดยไม่ต้องเขียน Code แบบเป็นลำดับขั้นเหมือน UIKit

2.4.2 SwiftUI เป็น Apple Cross Platform Devices ผู้พัฒนาสามารถพัฒนาแอปพลิเคชันได้ทุก Apple Devices ทำให้ลดเวลาและค่าใช้จ่ายในการพัฒนา

2.4.3 อย่างไรก็ตาม ข้อจำกัดของ SwiftUI คือสนับสนุนได้ตั้งแต่ iOS 13 ขึ้นไป และรวมทั้ง SwiftUI เป็น framework ที่ยังคงใหม่อยู่ ยังไม่ Maturity เหมือน UIKit แต่อย่างไรก็ตามเราสามารถ Integrate UIKit view to SwiftUI ได้

2.5 Firebase platform [5]

2.5.1 เป็น NoSQL Database (Firestore database) ซึ่งมีคุณสมบัติที่สามารถเก็บข้อมูลได้จำนวนมาก และยังคงมีสมรรถนะที่ดี รวมทั้งมีคุณสมบัติยืดหยุ่นและสามารถขยายฐานข้อมูลได้ (Flexible and Scalable)

2.5.2 พัฒนาได้อย่างรวดเร็ว - Firebase มีคุณสมบัติของ Backend อยู่ในตัว ทำให้เราสามารถเชื่อมต่อ Frontend ไปที่ Firebase ได้โดยตรง ทำให้ลดเวลาในการพัฒนาอย่างมาก รวมทั้งไม่ต้องมีการบำรุงรักษาทางด้าน Backend

2.5.3 การปรับเปลี่ยนตามผู้ใช้งาน (Personalization) - Firebase สามารถเพิ่ม Google Analytics ซึ่งสามารถวิเคราะห์และ เก็บข้อมูลของผู้ใช้ ทำให้สามารถสร้าง App ที่ตอบสนองความต้องการของผู้ใช้ได้ดีขึ้น

2.5.4 ค่ำค่า (Cost-Effectiveness) – เพราะ Firebase ทำให้มีการบำรุงรักษาด้าน Frontend ไม่ต้องมีการพัฒนาและบำรุงรักษาด้าน Backend ทำให้ App สามารถออกสู่ตลาดได้เร็ว นอกจากจะลดบุคคลากรด้าน Backend แล้วนั้น ยังลดข้อผิดพลาดต่างๆ (Bugs) ในการพัฒนาด้วย

2.5.5 มีชุมชนผู้ใช้งานขนาดใหญ่ และมี Technical documents ที่เพียงพอ

2.5.6 ง่ายต่อการบูรณาการ (Ease of integration) – Firebase สามารถสร้าง App ได้โดยง่าย มีการ configuration/setup ที่ง่าย

2.5.7 Push notifications – Firebase cloud messaging ทำให้ผู้พัฒนาสามารถทำ Push notification ส่งให้ผู้ใช้ทราบถึงสิทธิประโยชน์/Promotions ต่างๆ ที่มีในเวลานั้นๆ

2.6 แอปพลิเคชันที่เกี่ยวข้อง (พัฒนาโดยผู้พัฒนาคนไทยที่มีใน App Store)

2.6.1 แคลอรี่ ไดอารี่

แคลอรี่ ไดอารี่ พัฒนาโดย Dimo Co., Ltd

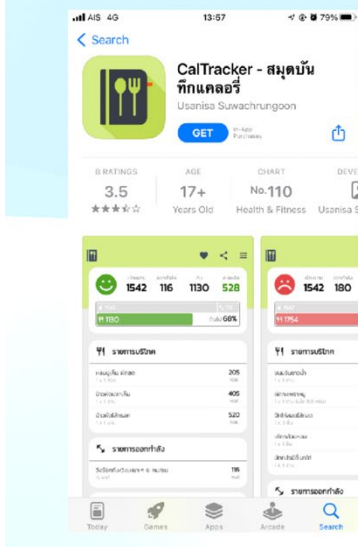


- ช่วยแนะนำและบันทึกปริมาณแคลอรี่ที่คุณบริโภค โดยจะแนะนำปริมาณแคลอรี่ที่ควรบริโภคต่อวัน
- มีกราฟแสดงปริมาณแคลอรี่ที่บริโภค
- สามารถค้นหาอาหารพร้อมแสดงปริมาณแคลอรี่
- สามารถเพิ่มเพื่อน
- รองรับการสแกนบาร์โค้ดเพื่อดูโภชนาการจากผลิตภัณฑ์อาหาร (ต้องสมัครสมาชิก)
- สามารถเชื่อมต่อกับแอปสุขภาพเพื่อการบันทึกและอัปเดตน้ำหนัก รวมถึงเชื่อมข้อมูลการก้าวเดิน (ต้องสมัครสมาชิก)
- มีบทความแนะนำด้านสุขภาพ (ต้องสมัครสมาชิก)
- รวมท่ากายบริหารต่างๆ (ต้องสมัครสมาชิก)

รูปที่ 2.2 แคลอรี่ ไดอารี่ แอป

2.6.2 CalTracker – สมุดบันทึกแคลอรี

CalTracker - สมุดบันทึกแคลอรี พัฒนาโดย Usanisa Suwachrunoon

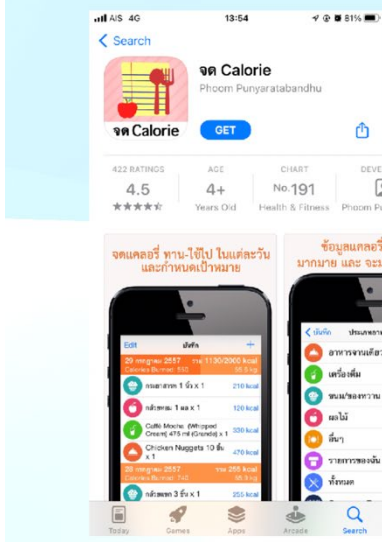


- มีกำหนดเป้าหมายแคลอรีต่อวัน
- สามารถค้นหาเมนูพร้อมปริมาณแคลอรี (บางเมนูต้องสมัครสมาชิก)
- สามารถแสดงผลแบบกราฟ
- สามารถบันทึกเมนูอาหารเป็นรูปภาพ (บางเมนูต้องสมัครสมาชิก)
- สามารถบันทึกรายการการออกกำลังกาย (บางเมนูต้องสมัครสมาชิก)
- สามารถแชร์ข้อมูลบันทึกไปกับ App อื่นๆ ได้
- สามารถเชื่อมต่อไปกับ Apple Health app ได้ (ต้องสมัครสมาชิก)

รูปที่ 2.3 CalTracker – สมุดบันทึกแคลอรี แอป

2.6.3 จด Calorie

จด Calorie พัฒนาโดย Phoom Punyaratabandhu



- ใน App มี Banner โฆษณา
- มีการบันทึก(สามารถแก้ไขได้)รายการอาหารพร้อมแคลอรีตาม List รายการที่มีใน App
- มีการค้นหาข้อมูลรายการอาหารจากการ Scan Barcode/QR
- มีการแสดงผลปริมาณ Calorie ด้วยกราฟ
- มี BMR/TDEE calculator
- มี Calculator คำนวณพลังงานที่ใช้ไปในกิจกรรมต่างๆ (มีกิจกรรมจำกัด)

รูปที่ 2.4 จด Calorie แอป

2.6.4 ตารางเปรียบเทียบ

ตารางเปรียบเทียบ Functions				
Functions	HealthMe	แคลอรีไดอารี่	CalTracker	จุด Calorie
• สามารถสมัครสมาชิกได้	X	X	X	
• สามารถคำนวณ BMI/Health info	X	X	X	X
• นำเข้ารูปอาหาร และใส่ข้อมูลโภชนาการ	X		X	
• Track ปริมาณ Calories ที่บริโภคเทียบกับที่แนะนำได้	X	X	X	
• แสดง Graph Calories ที่บริโภคไป และ Graph Calories 7 วันที่บริโภคไป	X	X	X	X
• แสดงรายการอาหารสุขภาพที่แนะนำพร้อมรายละเอียด	X	X		
• แสดงรายการออกกำลังกายที่แนะนำพร้อมรายละเอียด	X	X		
• แสดงรายละเอียดของผู้ใช้/แก้ไขได้	X	X		
• แสดงการสะสมแต้มของผู้ใช้/Leaderboard การสะสมแต้มแลกของรางวัล	X			
• ราคา	Free	In-App purchase	In-App purchase	Ad Banner

รูปที่ 2.5 ตารางเปรียบเทียบ Functions

2.7 สรุปเครื่องมือที่ใช้ในการพัฒนา

- (1) ใช้ SwiftUI framework ในการพัฒนา
- (2) ใช้ภาษา Swift ในการพัฒนา
- (3) ใช้ Xcode เป็น IDE ในการพัฒนา
- (4) ใช้ Firebase เป็นระบบฐานข้อมูล (Cloud Firestore/Cloud Storage) รวมทั้งระบบ Firebase Authentication (โดยใช้ Email/Password)

2.8 ทฤษฎี UX/UI

การออกแบบส่วนต่อประสานกับผู้ใช้และประสบการณ์ของผู้ใช้ (UX/UI) คือการออกแบบภาพรวมของผู้ใช้ต่อระบบที่ออกแบบ ว่าผู้ใช้มีความรู้สึกอย่างไรต่อระบบที่ออกแบบไว้ ซึ่งก็คือ UX design โดย Interface ของระบบว่ามีหน้าตาอย่างไร และมี Function อย่งไรนั่นก็คือ UI Design

โดยการจะได้มาซึ่ง UX/UI เราต้องมีทำการวิจัยเกี่ยวกับผู้ใช้ นำมาซึ่งการกำหนดคุณสมบัติของผู้ใช้ (Persona), ความคาดหวังของผู้ใช้ (User Expectation), โครงสร้างของผลิตภัณฑ์ (Product structure) จนนำมาถึงการทำ Prototyping/Wireframe

ในงานวิจัยนี้จะมีรายละเอียดของออกแบบ UX/UI ในบทที่ 3 หัวข้อ 3.6

บทที่ 3 ระเบียบวิธีวิจัย

ในบทนี้จะเป็นกระบวนการหาความต้องการของระบบ (Requirement analysis) โดยเราจะใช้ UML diagrams (Object-oriented modeling) เป็นหลักในการอธิบายระบบที่ผู้เขียนพัฒนา (ระบบที่ผู้เขียนพัฒนาได้ถูกตั้งชื่อว่า HealthMe App)

ซึ่งเราจะแสดงความต้องการของระบบ, Use case diagram, Use case description, Activity diagram, Class diagram, และ ER diagram. รวมทั้งจะแสดงการออกแบบ User Experience/User Interface (UX/UI) อีกด้วย

3.1 ความต้องการของระบบ

ความต้องการมี 2 แบบ คือ ความต้องการตามการใช้งาน (Functional Requirement) ซึ่งก็คือความต้องการที่ระบบที่เราพัฒนาต้องทำได้ มีทั้งหมด 9 ข้อ (ตามตารางที่ 3.1) และความต้องการที่ไม่ใช่ตามการใช้งาน (Non-Functional Requirement) ซึ่งเป็นความต้องการที่สนับสนุนการทำงานของระบบที่เราพัฒนา มีทั้งหมด 2 ข้อ

3.1.1 ความต้องการตามการใช้งาน (Functional Requirement)

ตารางที่ 3.1 ความต้องการตามการใช้งาน (Functional Requirement)

Requirement ID	Description
1	สามารถสมัครสมาชิกได้
2	สามารถคำนวณ BMI/BMR ได้
3	นำเข้ารูปอาหาร และใส่ข้อมูลที่เกี่ยวข้อง (สามารถแก้ไขได้ภายหลัง หรือลบออกได้)
4	สามารถติดตาม (Track) ปริมาณแคลอรีที่บริโภคไป เทียบกับที่แนะนำได้
5	แสดงกราฟปริมาณแคลอรีที่บริโภคไป และกราฟปริมาณแคลอรีที่บริโภคไป 7 วันย้อนหลัง
6	แสดงรายการอาหารสุขภาพที่แนะนำพร้อมรายละเอียด
7	แสดงรายการการออกกำลังกายที่แนะนำพร้อมรายละเอียด
8	แสดงรายละเอียดของผู้ใช้/แก้ไขได้
9	แสดงการสะสมแต้มของผู้ใช้ (Leaderboard)
10	แสดงรายการอาหารของผู้ใช้ทุกคนที่ใส่เข้ามาในระบบ

3.1.2 ความต้องการที่ไม่ใช่ตามการใช้งาน (Non-Functional Requirement)

- (1) แอปทำงานในระบบ iOS platform (support iPhone only)
- (2) เชื่อมต่อระบบ Firebase (Authentication/Cloud Firestore/Storage)

3.2 Use cases

3.2.1 Use case diagram คือ แผนภาพแสดงว่าผู้ใช้สามารถทำอะไรได้บ้างกับระบบที่เราพัฒนา ซึ่งในกรณีนี้ มีทั้งหมด 12 Use cases ดังภาพที่ 3.1 ด้านล่าง



รูปที่ 3.1 Use case diagram

3.2.2 Use case description จะเป็นการบรรยายรายละเอียดของแต่ละ Use case ที่มี ซึ่งในกรณีนี้มีทั้งหมด 12 Use cases (ตามตารางที่ 3.2 ถึง ตารางที่ 3.13)

ตารางที่ 3.2 Use case description UC01

Use case number	UC01
Use case name	ลงทะเบียน (Register)
Pre-condition	ผู้ใช้ที่ยังไม่เคยลงทะเบียน
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ผู้ใช้งาน Download App 2. App แสดงหน้า Login และผู้ใช้ยังไม่เคยลงทะเบียนมาก่อน ให้กดปุ่ม Sign Up 3. App แสดงหน้า Create Account 4. ผู้ใช้กรอกข้อมูลตาม Fields ต่างๆ (App มีการ Validating fields ต่างๆ จะแสดงให้ผู้ใช้ทราบ) 5. ผู้ใช้กดปุ่ม Sign Up ระบบจะสร้างข้อมูลของผู้ใช้ ซึ่งรวมทั้งระบบ Authentication (ใช้ Email/Password ของผู้ใช้) และข้อมูลของผู้ใช้ โดยระบบจะมีการคำนวณค่า BMI/BMR แล้วเก็บในฐานข้อมูล Firebase 6. App จะนำผู้ใช้เข้าสู่หน้าหลักของ App โดยอัตโนมัติ (หน้า Calories Tracker) 7. เมื่อผู้ใช้กลับมาใช้ App อีกครั้ง ระบบจะยังคงให้ผู้ใช้เข้าใช้งานหน้าหลักได้เลย (ระบบมีการเก็บ user session ไว้) นอกจากนี้ผู้ใช้ได้ทำการ Sign Out ออกจากระบบ (ปุ่ม Sign Out อยู่ในหน้า Profile) กรณีนี้ App จะนำผู้ใช้ไปสู่หน้า Login ซึ่งผู้ใช้จะต้องทำการ Login ด้วย Email/Password ที่ลงทะเบียนไว้

ตารางที่ 3.3 Use case description UC02

Use case number	UC02
Use case name	เพิ่มข้อมูลอาหารที่รับประทาน (Add Food info)
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ผู้ใช้กดปุ่ม Add ที่หน้า Calories Tracker 2. App แสดงหน้า Add Food Info 3. ผู้ใช้ใส่ข้อมูลอาหารที่ได้รับประทานไป 4. ผู้ใช้กดปุ่ม Add 5. App จะแสดง Food Calories Card view ในหน้า Calories Tracker แสดงเรียงตามเวลาที่ใส่ข้อมูลไป รวมทั้งแสดงปริมาณแคลอรีที่บริโภคในแต่ละวัน และ App ยังแสดงเป็น Progress Ring สำหรับข้อมูลแคลอรีสำหรับวันนี้ (วันที่ปัจจุบัน)

ตารางที่ 3.4 Use case description UC03

Use case number	UC03
Use case name	แก้ไขข้อมูลอาหารที่รับประทานที่เพิ่มไปแล้ว (Edit Food info)
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ผู้ใช้กดที่ Food Calories Card view ในหน้า Calories Tracker ที่เพิ่มข้อมูลไว้ และต้องการแก้ไข 2. App จะแสดง Confirm dialog ผู้ใช้กดปุ่ม Edit Food Calories จะแสดงหน้า Edit Food Info 3. ผู้ใช้ใส่ข้อมูลที่ต้องการแก้ไข กดปุ่ม Save 4. ข้อมูลใน Food Calories Card view ในหน้า Calories Tracker จะถูก Update ตามที่ผู้ใช้แก้ไข

ตารางที่ 3.5 Use case description UC04

Use case number	UC04
Use case name	ลบข้อมูลอาหารที่รับประทานที่เพิ่มไปแล้ว (Delete Food)
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ผู้ใช้กดที่ Food Calories Card view ในหน้า Calories Tracker ที่เพิ่มข้อมูลไว้ และต้องการลบ 2. App จะแสดง Confirm dialog ผู้ใช้กดปุ่ม Delete Food Calories 3. ข้อมูลใน Food Calories Card view ในหน้า Calories Tracker จะถูกลบออกไปตามที่ต้องการ

ตารางที่ 3.6 Use case description UC05

Use case number	UC05
Use case name	แสดงข้อมูลอาหารที่ Track Calories
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ในหน้า Calories Tracker จะแสดง Progress Ring เป็นข้อมูลสำหรับอาหารที่บริโภควันนี้ (วันที่ปัจจุบัน) 2. และแสดง Food Calories Card view ตามวันเวลาที่เพิ่มข้อมูลเข้าไป (เรียงตามวันที่ปัจจุบัน ไล่ถอยหลังไปอดีต)

ตารางที่ 3.7 Use case description UC06

Use case number	UC06
Use case name	แสดงข้อมูลแคลอรีรวมในแต่ละวัน ย้อนหลัง 7 วันแบบกราฟแท่ง (Show Total Graph)
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ในหน้า Calories Tracker ผู้ใช้กดปุ่ม Sum 2. App แสดงหน้า Calories Stat ซึ่งจะแสดงเป็นกราฟแท่ง 7 วันย้อนหลัง รวมทั้งแสดงวันที่ผู้ใช้ทำได้ดีที่สุด (แคลอรีรวม น้อยที่สุด) และแสดงวันที่ผู้ใช้ทำได้แย่ที่สุด (แคลอรีรวมมากที่สุด)

ตารางที่ 3.8 Use case description UC07

Use case number	UC07
Use case name	แสดงข้อมูลแคลอรีและอาหารที่ผู้ใช้ทั้งหมดได้ทำการเพิ่มไว้ในระบบ (Show Food For All Users)
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ในหน้า Calories Tracker ผู้ใช้กดปุ่ม Food 2. App แสดงหน้า Food Calories Card View ที่ผู้ใช้แต่ละคน Add ไว้ (แสดงด้วยว่าเป็นผู้ใช้งานไหนเพิ่มข้อมูลนี้) เรียงตามวันที่ปัจจุบันที่สุด และย้อนหลังไปในอดีต

ตารางที่ 3.9 Use case description UC08

Use case number	UC08
Use case name	แสดงข้อมูลอาหารสุขภาพที่แนะนำ (Show Healthy Food List/Detail)
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ผู้ใช้กด Tab Food Menu ของ App ในหน้า Main 2. จะแสดง Food Menu Card ซึ่งแสดงรายการข้อมูลอาหารที่แนะนำ โดยผู้ใช้สามารถกด Icon Info, App จะแสดงรายละเอียดเพิ่มเติมสำหรับอาหารสุขภาพดังกล่าว 3. ในหน้า Detail ผู้ใช้สามารถกด Link ไปยัง Web site ที่มีข้อมูลเพิ่มเติมของอาหารสุขภาพตัวนั้นได้ (ใน App มี Built-in Web Browser)

ตารางที่ 3.10 Use case description UC09

Use case number	UC09
Use case name	แสดงข้อมูลการออกกำลังกายที่แนะนำ (Show Exercise List/Detail)
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ผู้ใช้กด Tab Exercise Menu ของ App ในหน้า Main 2. จะแสดง Exercise Card ซึ่งแสดงรายการข้อมูลการออกกำลังกายที่แนะนำ โดยผู้ใช้สามารถกด Icon Info, App จะแสดงรายละเอียดเพิ่มเติมสำหรับการออกกำลังกายดังกล่าว 3. ในหน้า Detail ผู้ใช้สามารถกด Link ไปยัง Web site ที่มีข้อมูลเพิ่มเติมของการออกกำลังกายนั้นได้ (ใน App มี Built-in Web Browser)

ตารางที่ 3.11 Use case description UC10

Use case number	UC10
Use case name	แสดงข้อมูลผู้ใช้/แก้ไข (Show/Edit User Profile View)
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ผู้ใช้กด Tab Profile ของ App ในหน้า Main 2. จะแสดง Profile View ของผู้ใช้ที่ Login อยู่ 3. ผู้ใช้สามารถกดปุ่ม Edit เพิ่มแก้ไขข้อมูลที่ผู้ใช้ต้องการได้ ซึ่งถ้าผู้ใช้กดปุ่ม Save, App จะแสดงหน้า Profile View ที่มีข้อมูลที่ผู้ใช้แก้ไข Updated ทันที

ตารางที่ 3.12 Use case description UC11

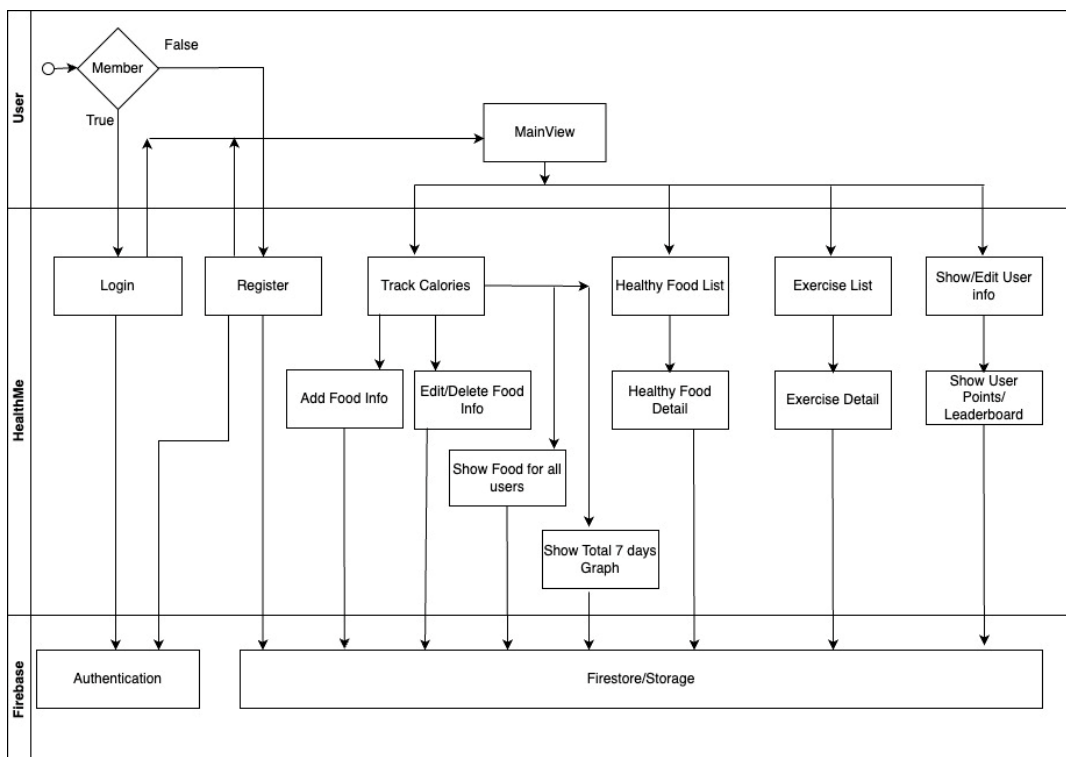
Use case number	UC11
Use case name	แสดงลำดับของผู้ใช้ทั้งหมดที่มีแต้มสะสมเรียงจากมากไปน้อย (Show Leaderboard) – แต้มสะสมจะ + 10 points ทุกครั้งที่ผู้ใช้งานเพิ่มอาหารเข้ามาในระบบ
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ในหน้า Profile View ผู้ใช้สามารถกดปุ่ม Leaderboard 2. จะแสดงหน้า Leaderboard แสดงรายการของผู้ใช้ในระบบทั้งหมด เรียงตามจากแต้มสะสมมากไปน้อย (สำหรับผู้ใช้ปัจจุบันที่ Login อยู่ จะแสดงพื้นหลังสีเขียว เพื่อให้เห็นได้ชัดเจนว่าผู้ใช้คนนั้นอยู่ในลำดับที่เท่าไร) 3. ถ้ากดที่ตัว Leaderboard Card View ของแต่ละผู้ใช้ จะแสดงหน้าอาหารทั้งหมดที่ผู้ใช้คนนั้นเพิ่มเข้าสู่ระบบ

ตารางที่ 3.13 Use case description UC12

Use case number	UC12
Use case name	การเข้าใช้งาน (Login with Email/Password)
Pre-condition	ผู้ใช้ที่ลงทะเบียนแล้ว (มี Email/Password)
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1. ผู้ใช้ไปที่หน้า Login 2. กรอก Email/Password และกดปุ่ม OK 3. ระบบจะตรวจสอบ Firebase Authentication ที่ลงทะเบียนไว้ 4. ถูกต้องจะนำผู้ใช้ไปหน้าเข้าสู่หน้าหลักของ App โดยอัตโนมัติ (หน้า Calories Tracker) 5. ไม่ถูกต้องจะมี Alert view แจ้ง

3.3 Activity Diagram

เป็นการแสดง Work flow ของระบบที่ผู้ใช้มีการตอบสนองด้วย ซึ่งภาพที่ 3.2 แสดงองค์รวมทั้งหมดของระบบ



รูปที่ 3.2 Activity Diagram

ซึ่งมีรายละเอียดของ Work flow ดังต่อไปนี้

3.3.1 ผู้ใช้เข้าใช้แอปครั้งแรก จะแสดงหน้า Login ผู้ใช้ยังไม่มี Account ก็ให้กดไปหน้าลงทะเบียน (Register) ซึ่งก็คือ Use case description UC01 จากนั้นระบบจะเข้าสู่ระบบโดยอัตโนมัติ (Authentication) นำผู้ใช้เข้าสู่ Main View

3.3.2 หน้า Main view จะเป็นแบบ Tab view ให้ผู้ใช้เลือกได้ 4 Views ได้แก่ Calories Tracker view, Food list view, Exercise list view, และ Profile view

3.3.3 ในหน้า Calories Tracker view ผู้ใช้สามารถกดปุ่ม Add food calories ระบบจะนำผู้ใช้ไปสู่หน้า Add food info view ผู้ใช้กรอกรายละเอียด (Use case description UC02) ระบบจะบันทึกข้อมูลลง Firebase และแสดง Food Calories Card view ในหน้า Calories Tracker view

3.3.4 ในหน้า Calories Tracker view ผู้ใช้สามารถกดที่ Food Calories Card view ระบบจะแสดง Confirmation dialog ผู้ใช้เลือก Edit Food Info (Use case description UC03) ระบบจะนำไปสู่หน้า Edit Food Info ผู้ใช้กรอกข้อมูลที่ต้องการแก้ไข กด Save ระบบจะ Update ข้อมูลใน Firebase

3.3.5 ในหน้า Calories Tracker view ผู้ใช้สามารถกดที่ Food Calories Card view ระบบจะแสดง Confirmation dialog ผู้ใช้เลือก Delete Food (Use case description UC04) ระบบจะลบข้อมูลใน Firebase

3.3.6 ในหน้า Calories Tracker view จะมีแสดง Progress Ring view แสดงปริมาณแคลอรีที่บริโภคในวันนี้ (วันปัจจุบัน) และแสดง Food Calories card view เรียงตามวันที่ที่เพิ่มเข้าไปในระบบ รวมทั้งแสดงปริมาณแคลอรีที่บริโภคไปในแต่ละวัน (ตรงกับ Use case description UC05)

3.3.7 ในหน้า Calories Tracker view ผู้ใช้กดปุ่ม Sum ระบบจะนำผู้ใช้ไปหน้า Calories Stat ซึ่งจะแสดงกราฟแท่งแสดงปริมาณแคลอรีในแต่ละวันที่บริโภคไปย้อนหลัง 7 วัน รวมทั้งแสดงวันที่ดีที่สุด (วันที่ปริมาณแคลอรีน้อยที่สุด) และวันที่แย่ที่สุด (วันที่ปริมาณแคลอรีมากที่สุด) (ตรงกับ Use case description UC06)

3.3.8 ในหน้า Calories Tracker view ผู้ใช้กดปุ่ม Food ระบบจะนำผู้ใช้ไปหน้าแสดงรายการอาหารที่ผู้ใช้ทั้งหมดเพิ่มเข้าไปในระบบ เรียงตามลำดับวันที่เพิ่มข้อมูลเข้าไปที่หลัง แล้วย้อนหลังไปในอดีตตามลำดับ (ตรงกับ Use case description UC07)

3.3.9 ผู้ใช้กด Tab view Food menu ระบบจะนำไปสู่หน้า Healthy food menu list ผู้ใช้สามารถกดปุ่ม Info icon จะแสดงหน้ารายละเอียดของ Food menu นั้น ซึ่งผู้ใช้ยังสามารถกด Link ไปยัง Web site ที่มีข้อมูลเพิ่มเติมเกี่ยวกับ Food menu นั้น (ตรงกับ Use case description UC08)

3.3.10 ผู้ใช้กด Tab view Exercise menu ระบบจะนำไปสู่หน้า Exercise menu list ผู้ใช้สามารถกดปุ่ม Info icon จะแสดงหน้ารายละเอียดของ Exercise menu นั้น ซึ่งผู้ใช้ยังสามารถกด Link ไปยัง Web site ที่มีข้อมูลเพิ่มเติมเกี่ยวกับ Exercise menu นั้น (ตรงกับ Use case description UC09)

3.3.11 ผู้ใช้กด Tab view Profile view จะแสดงรายละเอียดของผู้ใช้ที่ Login อยู่ ซึ่งผู้ใช้สามารถ

กดปุ่ม Edit ระบบจะนำไปสู่หน้า Edit Profile view ผู้ใช้แก้ไขข้อมูลที่ต้องการแล้วกดปุ่ม Save ระบบจะบันทึกข้อมูลที่แก้ไขลง Firebase และแสดงข้อมูลที่ Update บน Profile view (ตรงกับ Use case description UC10)

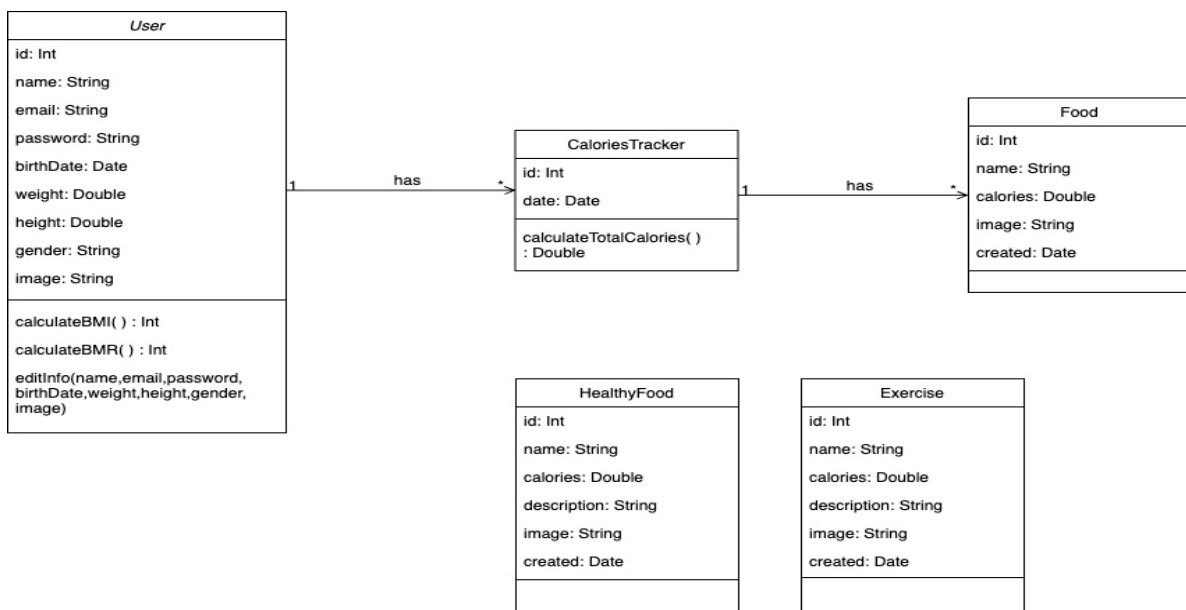
3.3.12 ในหน้า Profile view ผู้ใช้สามารถกดปุ่ม Trophy ระบบจะนำผู้ใช้ไปสู่หน้า Leaderboard ซึ่งจะแสดงรายการผู้ใช้ที่มีลำดับคะแนนที่ได้รับสูงสุด เรียงไปคะแนนน้อยที่สุด โดยสำหรับผู้ใช้ที่ Login อยู่ นั้น จะมีการแสดงเป็นพื้นสีเขียว เพื่อให้ผู้ใช้สังเกตเห็นได้ว่าตนเองอยู่ลำดับที่เท่าไร และในหน้า Leaderboard นี้ ถ้าผู้ใช้กดที่ card ของผู้ใช้คนใด ระบบจะนำไปสู่หน้ารายการอาหารที่ผู้ใช้ท่านนั้นเพิ่มเข้าสู่ระบบทั้งหมด (ตรงกับ Use case description UC11)

3.4 Class Diagram

เป็นการแสดงโครงสร้างของระบบ ตามแผนภาพที่ 3.3 ประกอบไปด้วย

- (1) User class
- (2) CaloriesTracker class
- (3) Food class
- (4) HealthyFood class
- (5) Exercise class

โดย User class มีความสัมพันธ์แบบ One-to-Many กับ CaloriesTracker class ซึ่งมีความสัมพันธ์แบบ One-to-Many กับ Food class ส่วน HealthyFood class กับ Exercise class ไม่มีความสัมพันธ์กับ class ใด ๆ เพราะเป็นเพียงแสดงข้อมูลในระบบเท่านั้น



รูปที่ 3.3 Class Diagram

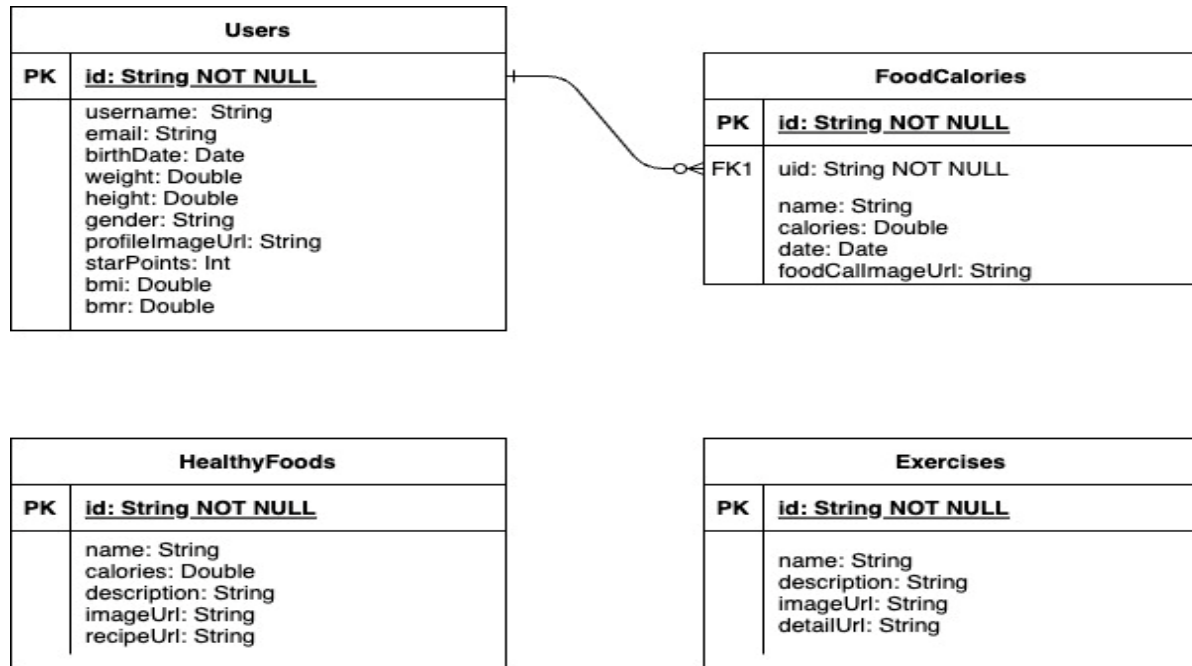
โดยตารางที่ 3.14 แสดงรายละเอียดของแต่ละ Class

ตารางที่ 3.14 รายละเอียดแต่ละ Class

Class	Attribute/Type	Method
User	id/Integer name/String email/String password/String birthDate/Date weight/Double height/Double gender/String image/String	calculateBMI() : Integer calculateBMR() : Integer editInfo(name,email,password, birthdate,weight,height,gender,image)
CaloriesTracker	id/Integer date/Date	calculateTotalCalories() : Double
Food	id/Integer name/String calories/Double image/String created/Date	
HealthyFood	id/Integer name/String calories/Double description/String image/String created/Date	
Exercise	id/Integer name/String calories/Double description/String image/String created/Date	

3.5 ER Diagram

ER diagram หรือ Entity Relationship diagram แสดงโครงสร้างของ Database โดยแสดงเป็น Diagram ดังภาพที่ 3.4 ซึ่งจะเห็นได้ว่า Users Entity มีความสัมพันธ์แบบ One-to-Many กับ FoodCalories Entity สำหรับ HealthyFoods Entity กับ Exercises Entity ไม่มีความสัมพันธ์กับ Entity ใด เนื่องมาจากการแสดงข้อมูลในระบบเท่านั้น



รูปที่ 3.4 ER Diagram

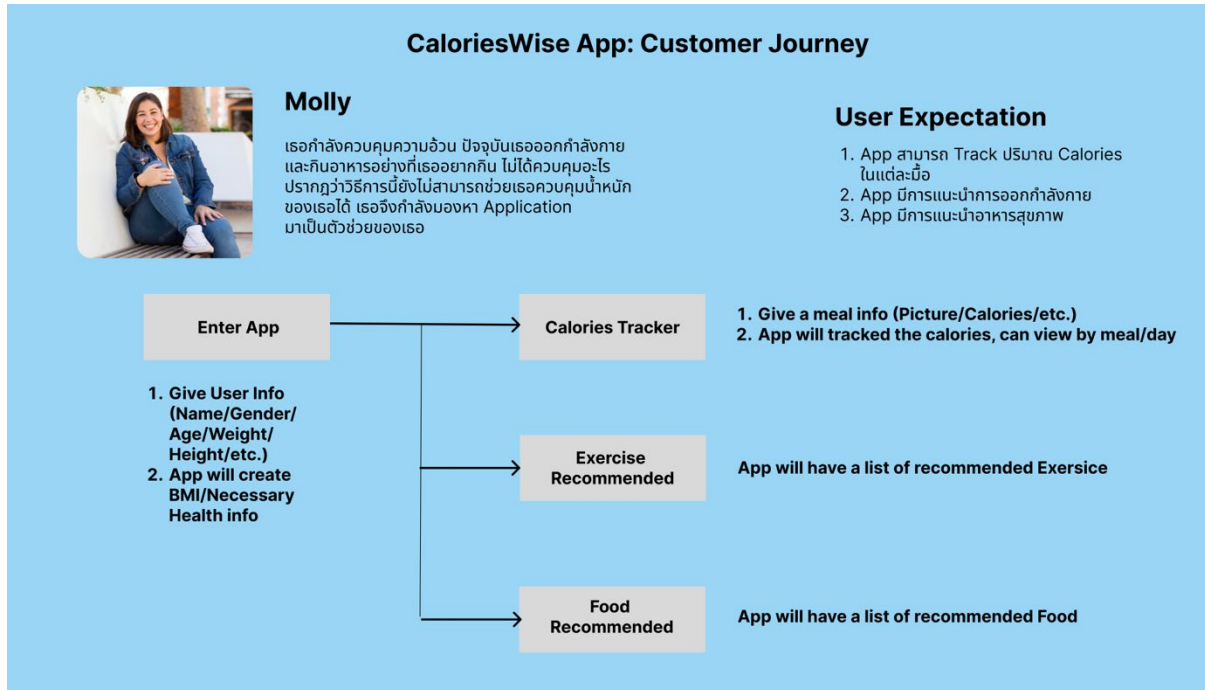
ส่วนการแสดงรายละเอียดของแต่ละ Entity แสดงตามตารางที่ 3.15

ตารางที่ 3.15 รายละเอียดแต่ละ Entity

Entity	Attribute/Type
Users	PK id: String NOT NULL username: String email: String birthDate: Date weight: Double height: Double gender: String profileImageUrl: String starPoints: Integer bmi: Double bmr: Double
FoodCalories	PK id: String NOT NULL FK uid: String NOT NULL name: String calories: Double date: Date foodCallImageUrl: String
HealthyFoods	PK id: String NOT NULL name: String calories: Double description: String imageUrl: String recipeUrl: String
Exercises	PK id: String NOT NULL name: String calories: Double description: String imageUrl: String detailUrl: String

3.6 UX/UI design

ในเริ่มแรกผู้เขียน มีการทำ Customer Journey ของ App มีการ Setup Persona ของผู้ใช้ App และ Expectation ของผู้ใช้ ดังแสดงรายละเอียด ตามภาพที่ 3.5 ด้านล่าง (ในขณะนั้นตั้งชื่อว่า CaloriesWise App และมาเปลี่ยนชื่อในภายหลังเป็น HealthMe App)



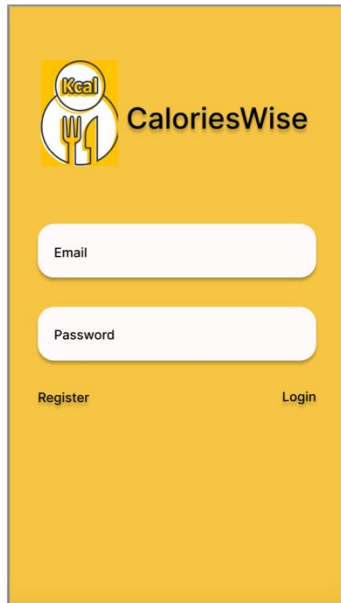
รูปที่ 3.5 Customer Journey

Persona สำหรับ App นี้ที่วิเคราะห์ไว้นั้นคือ เป็นผู้หญิงวัยทำงาน ที่มีรูปร่างเจ้าเนื้อ แต่ไม่อ้วน ซึ่งต้องการควบคุมน้ำหนัก ซึ่งเธอมีการออกกำลังกายเป็นประจำ แต่ไม่ได้มีการควบคุมอาหารแต่อย่างใด ซึ่งวิธีการออกกำลังกายอย่างเดียวของเธอไม่สามารถทำให้เธอควบคุมน้ำหนักได้ เธอจึงต้องการหาตัวช่วยโดยการหาเครื่องมือให้เธอช่วยในการควบคุมปริมาณอาหารที่เธอบริโภคเข้าไป จึงเกิดเป็นแนวคิดของ App นี้ขึ้นเอง

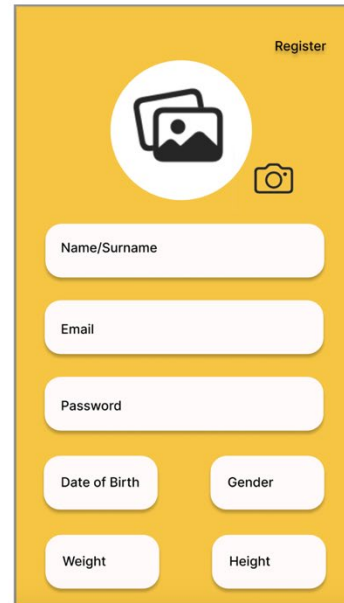
โดย Expectation ของ App นี้คือ

- (1) App สามารถ Track ปริมาณแคลอรี ที่บริโภคอาหารเข้าไปในแต่ละมื้อ
- (2) App มีการแนะนำการออกกำลังกาย
- (3) App มีการแนะนำอาหารสุขภาพ

ส่วน UI design ในเบื้องต้น ได้ทำการออกแบบโดยใช้ Figma ซึ่งมีแนวคิดคร่าวๆ ของ App คือ เป็นแอปที่เป็นกันเองไม่เป็นทางการมากนัก ผู้ใช้ใช้แล้วควรจะรู้สึกสนุกกับการใช้ และอยากกลับมาใช้แอปอยู่ตลอด จึงออกแบบแอปโดยใช้ 3 สี คือ สีเหลือง สีขาว และสีดำ เป็นโทนสีหลักของแอป ตามภาพที่ 3.6 ถึง 3.12 เป็น UI ที่ออกแบบไว้เบื้องต้นใน Figma

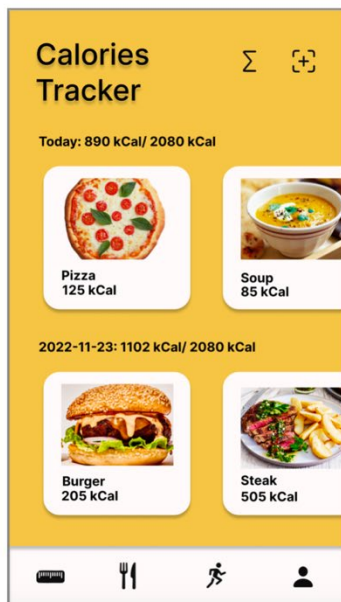


Login Screen



Register Screen

รูปที่ 3.6 Login screen and Register screen

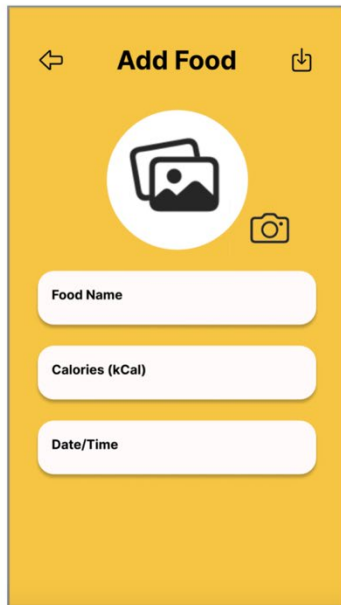


Calories Tracker Screen

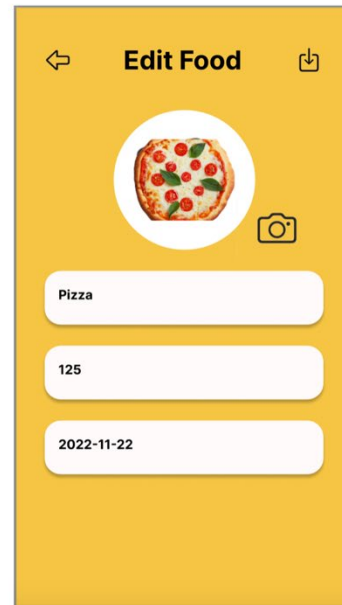


Total Screen

รูปที่ 3.7 Calories Tracker screen and Total screen

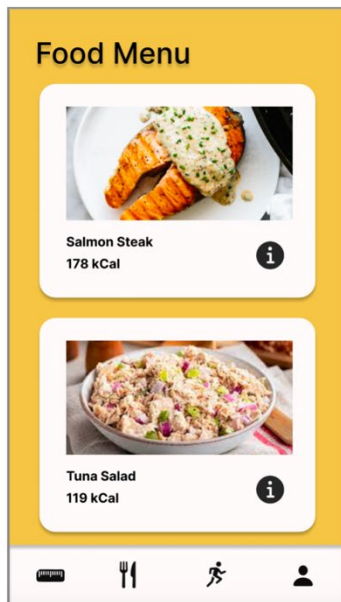


Add Food Screen

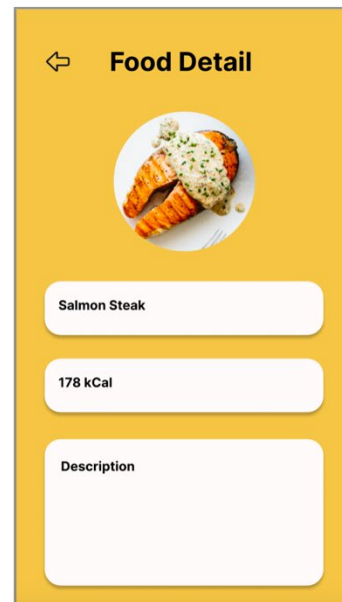


Edit Food Screen

รูปที่ 3.8 Add Food screen and Edit Food screen

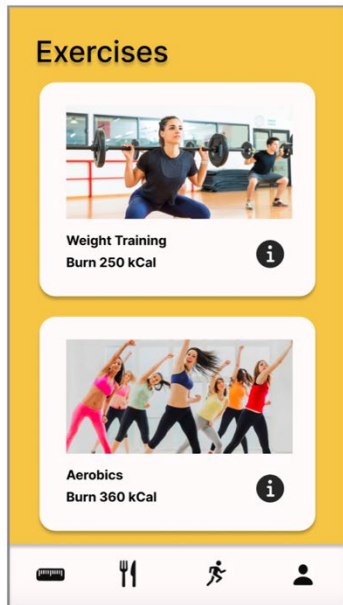


Food Menu Screen

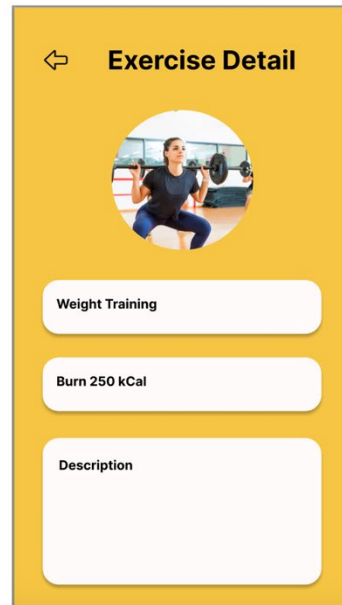


Food Detail Screen

รูปที่ 3.9 Food Menu screen and Food Detail screen

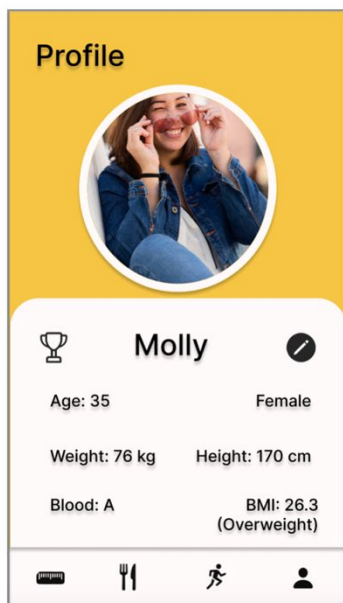


Exercises Screen

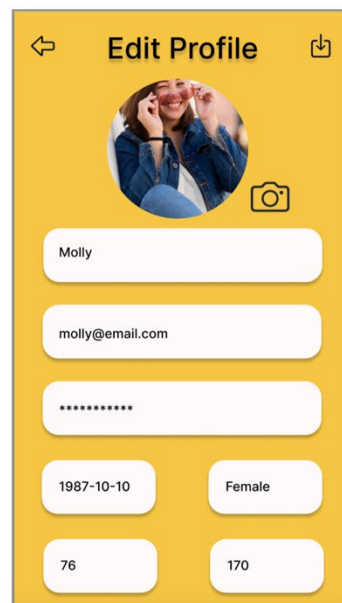


Exercise Detail Screen

รูปที่ 3.10 Exercises screen and Exercise Detail screen

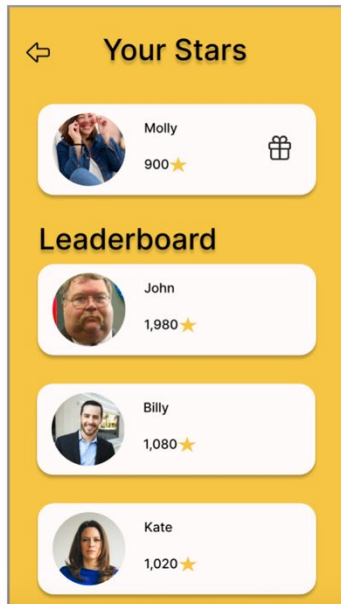


Profile Screen

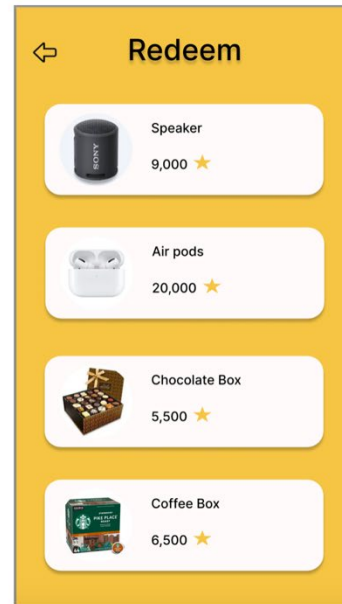


Edit Profile Screen

รูปที่ 3.11 Profile screen and Edit Profile screen



Leaderboard Screen



Redeem Screen

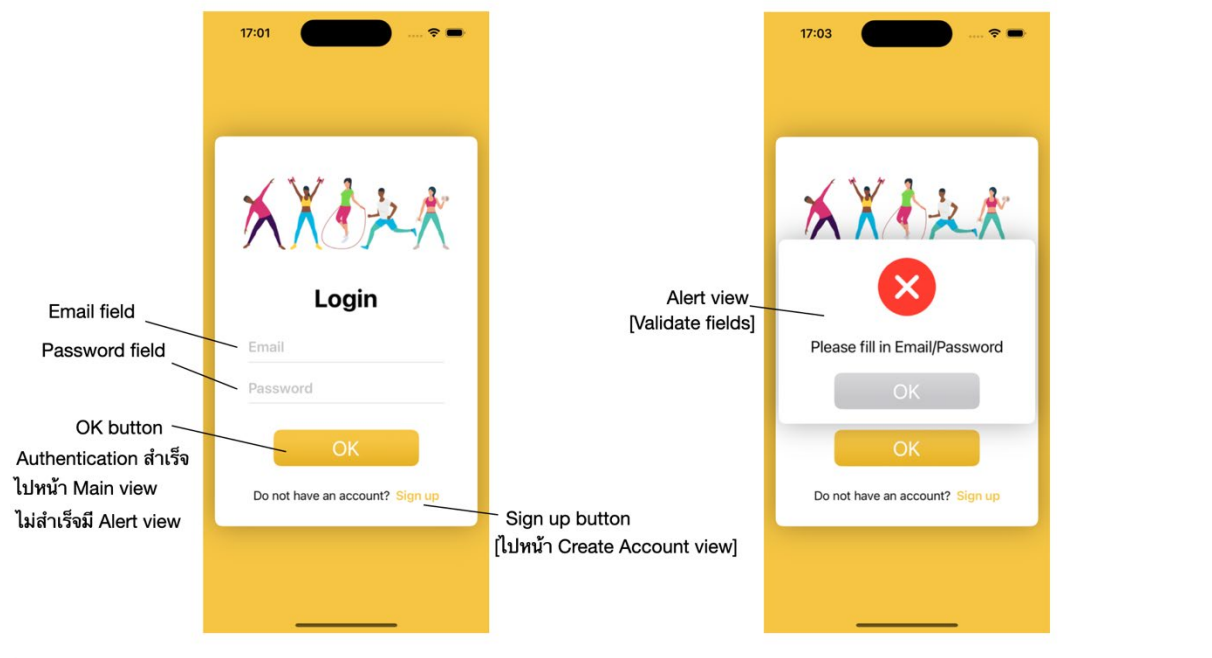
รูปที่ 3.12 Leaderboard screen and Redeem screen

บทที่ 4 ผลการวิจัย

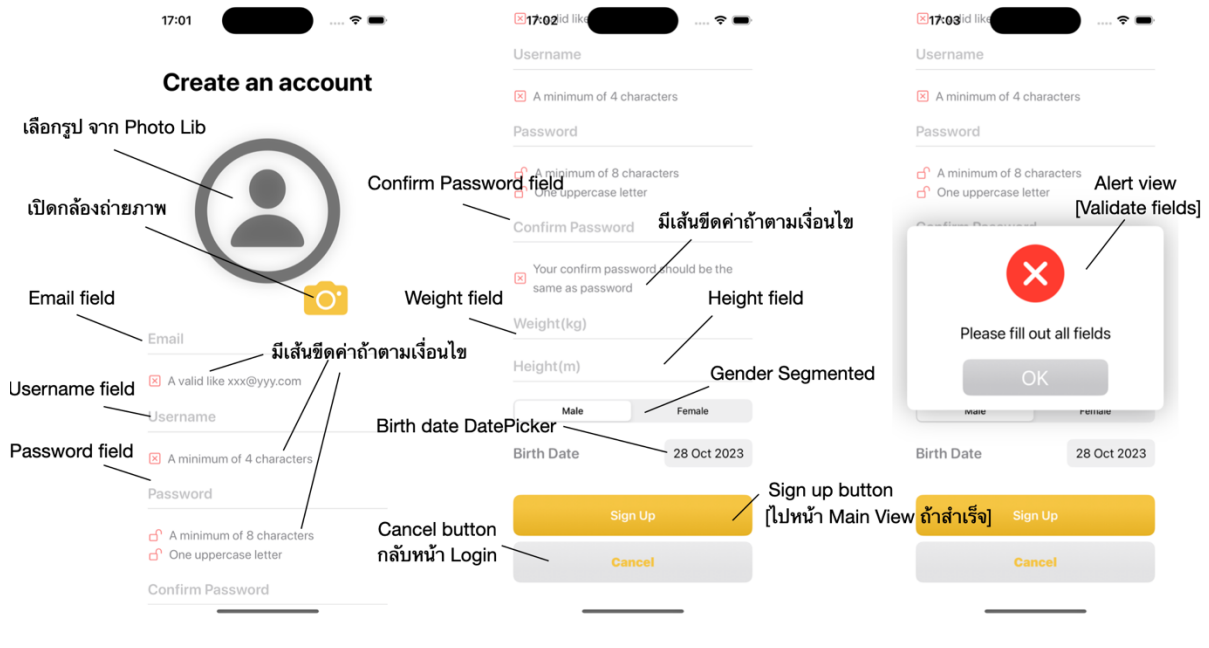
ในบทนี้จะแสดงผลของการพัฒนาแอปพลิเคชัน โดยแสดง User Interface ของแอปพลิเคชันที่พัฒนา แสดงฐานข้อมูล Firebase ที่สร้างขึ้น และแสดงผลการทดสอบระบบที่พัฒนาขึ้น

4.1 User Interface สำหรับแอปพลิเคชันที่พัฒนาสำเร็จ

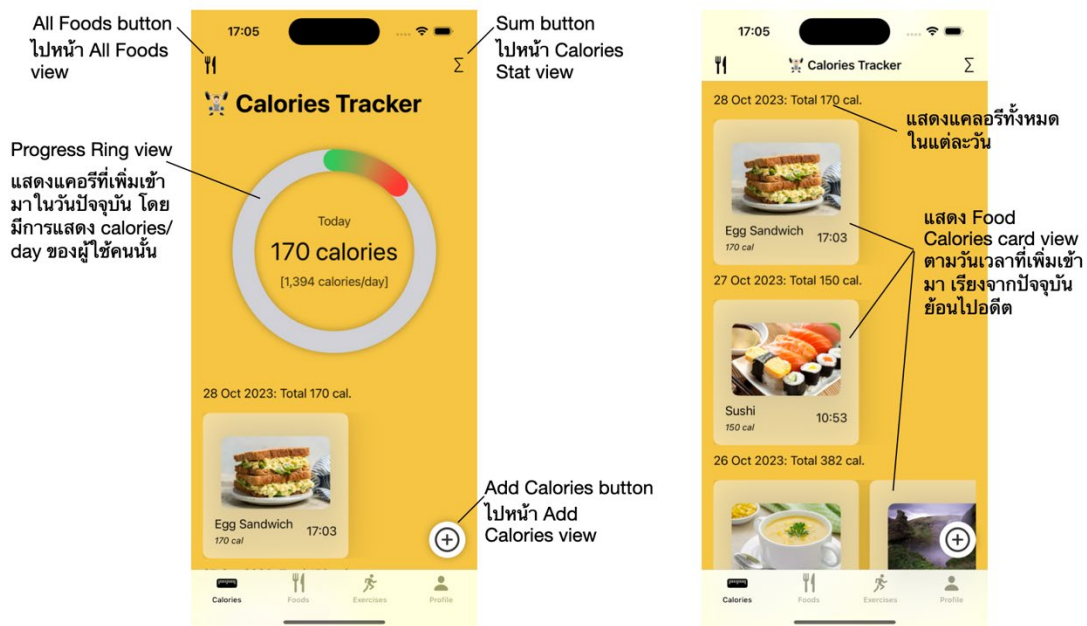
แสดง User Interface ของแอปพลิเคชันที่พัฒนาสำเร็จ ตามภาพที่ 4.1 ถึง ภาพที่ 4.12



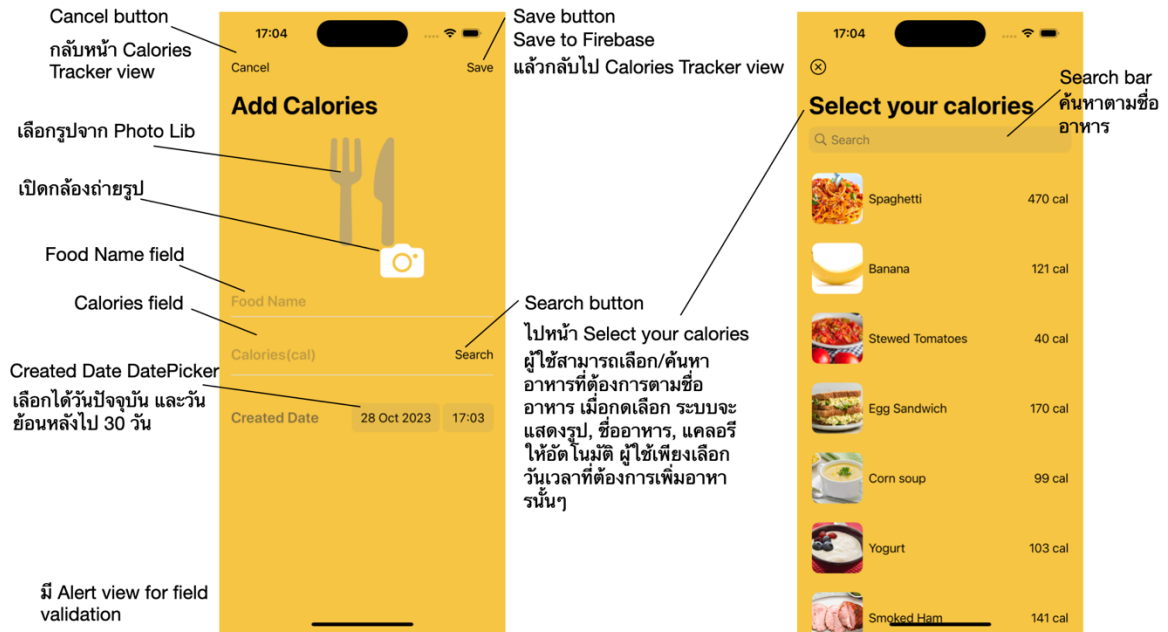
รูปที่ 4.1 Login view



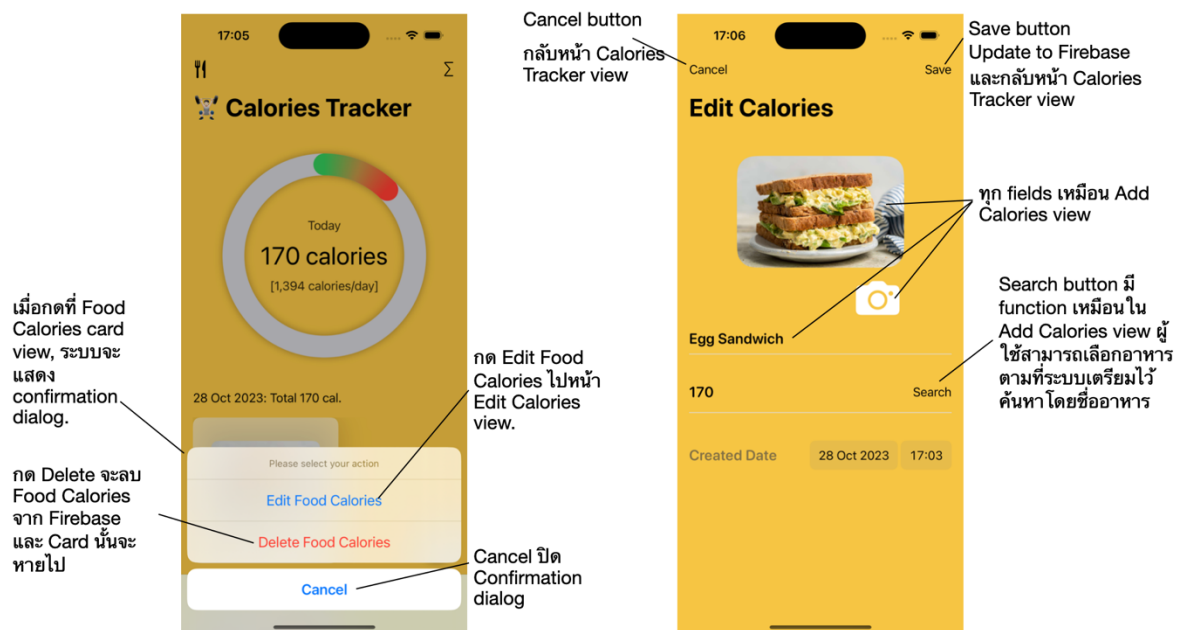
รูปที่ 4.2 Create Account view



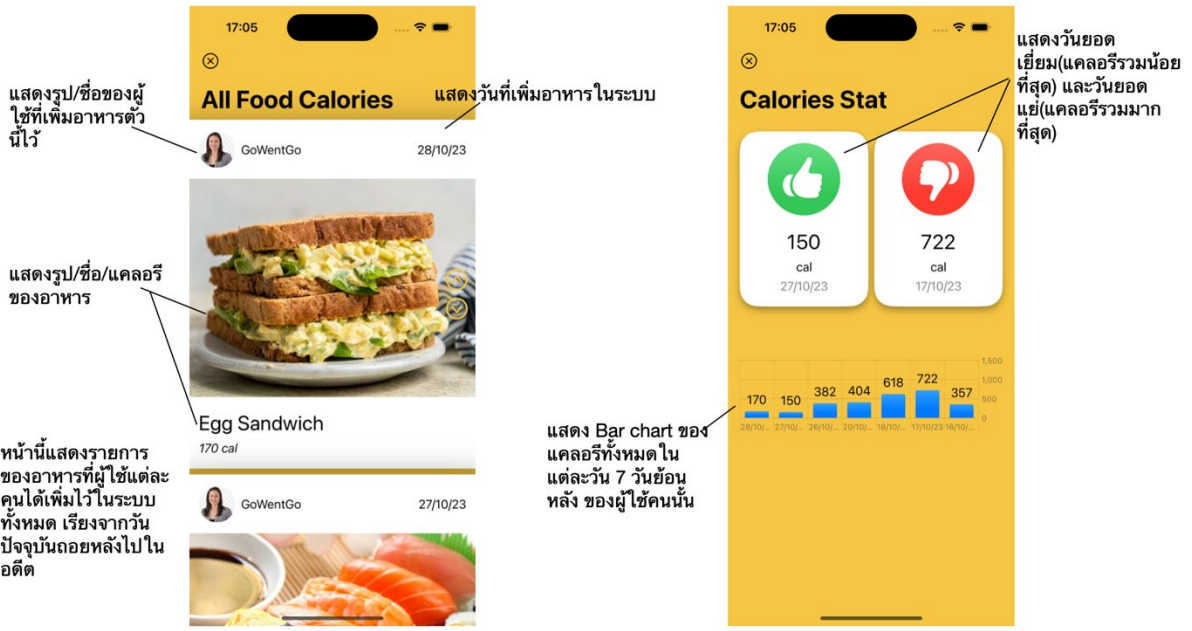
รูปที่ 4.3 Calories Tracker view



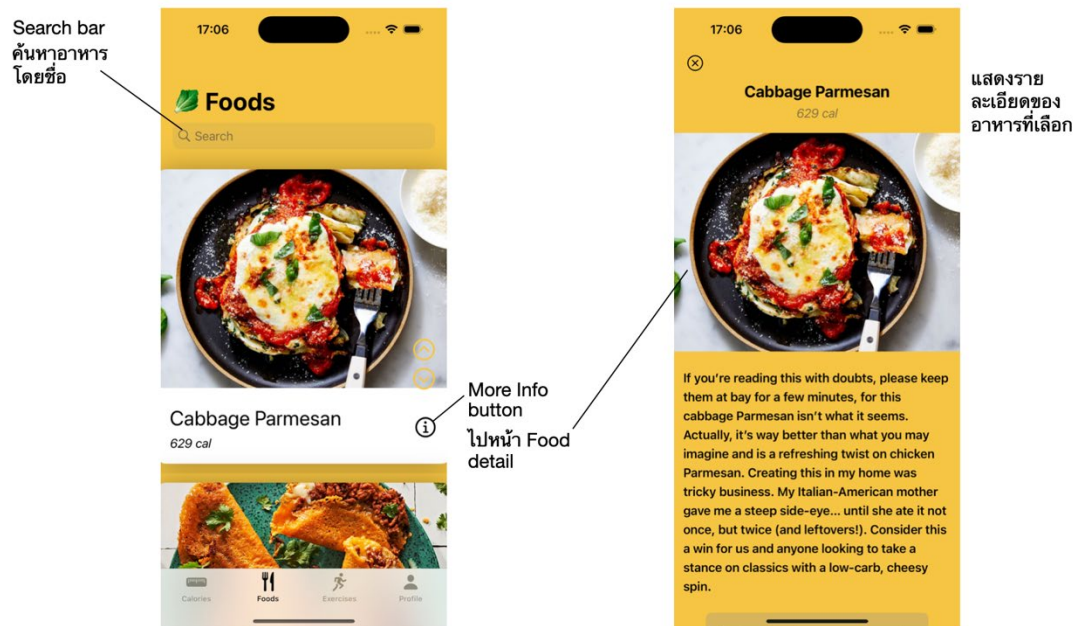
รูปที่ 4.4 Add Calories view/Select your calories view



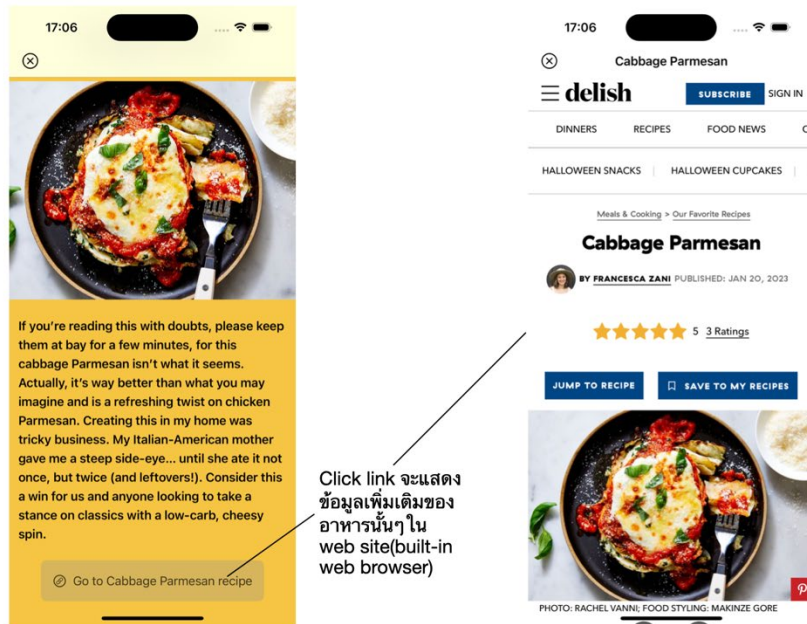
รูปที่ 4.5 Calories Tracker view/Edit Calories view



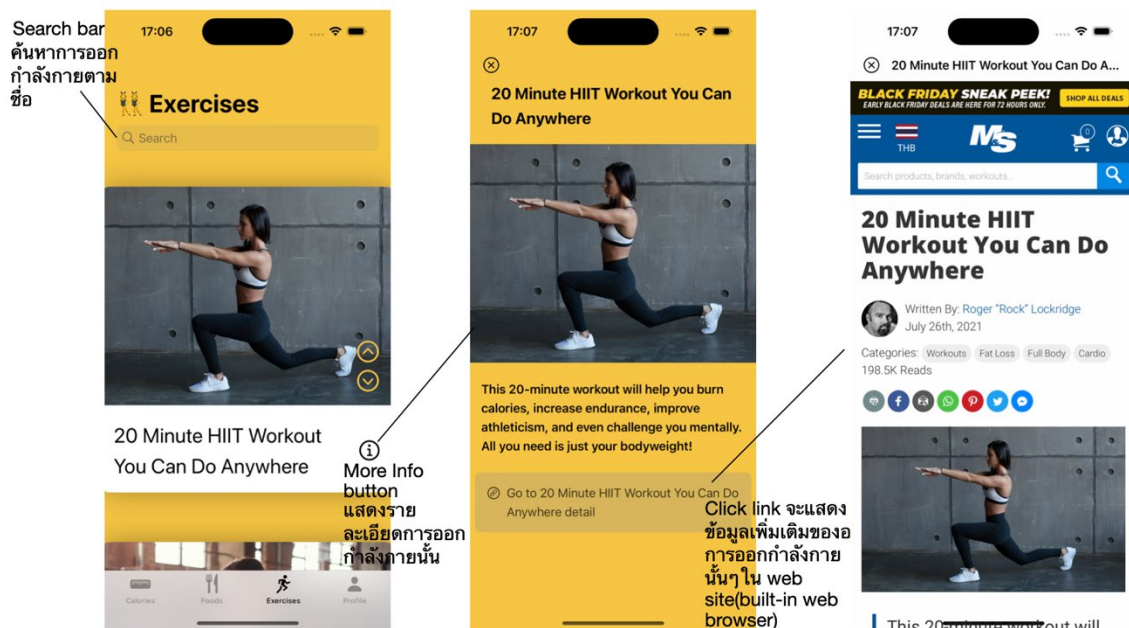
รูปที่ 4.6 All Food Calories view/Calories Stat view



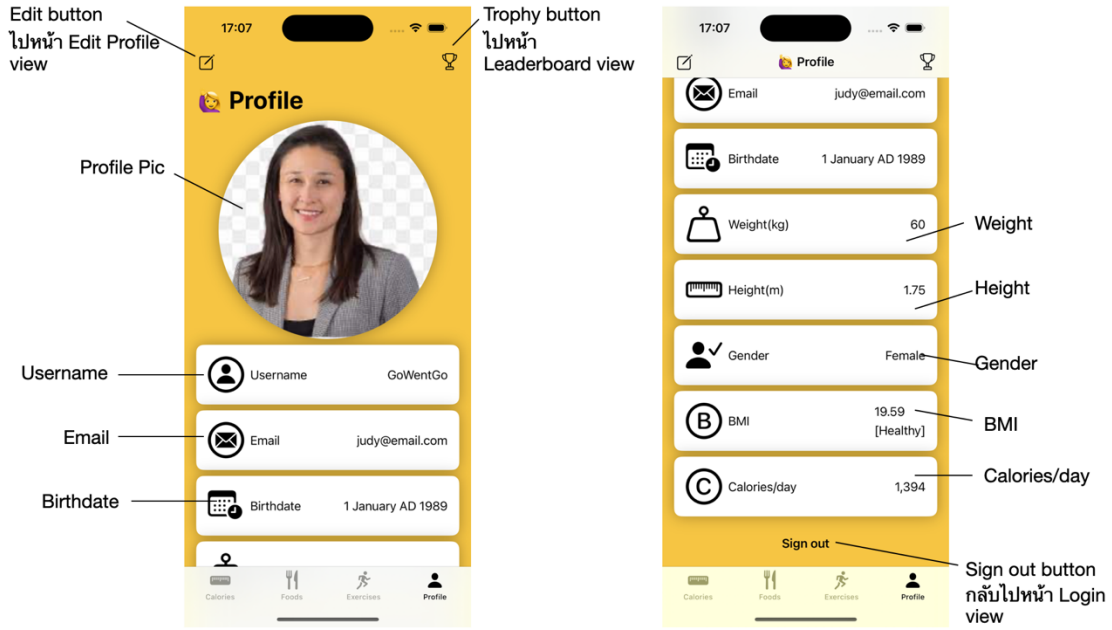
รูปที่ 4.7 Food Menu view/Food Menu Detail view



รูปที่ 4.8 Food Menu Detail view (continue)



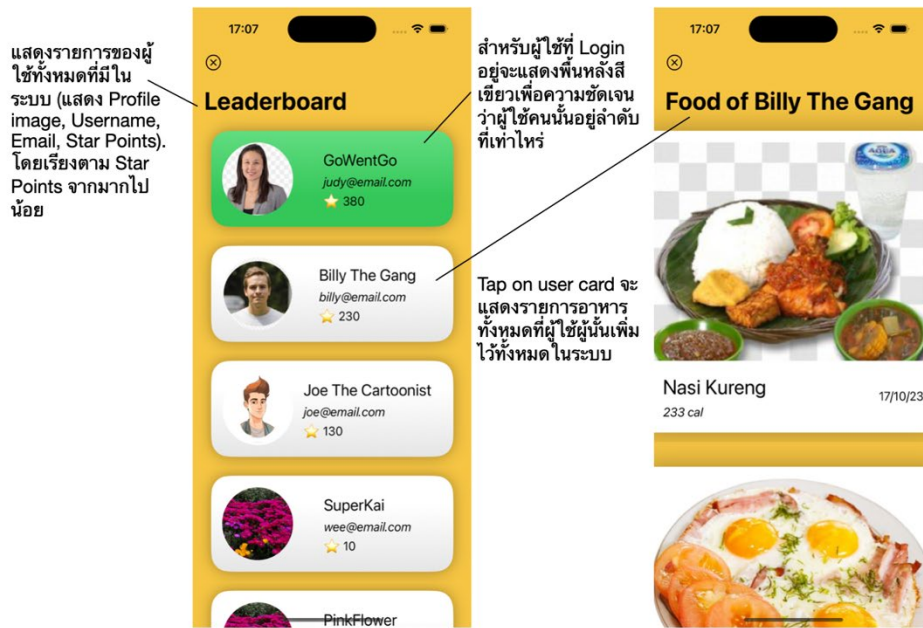
รูปที่ 4.9 Exercise Menu view/Exercise Detail view



รูปที่ 4.10 Profile view



รูปที่ 4.11 Edit Profile view



รูปที่ 4.12 Leaderboard view/Food of user view

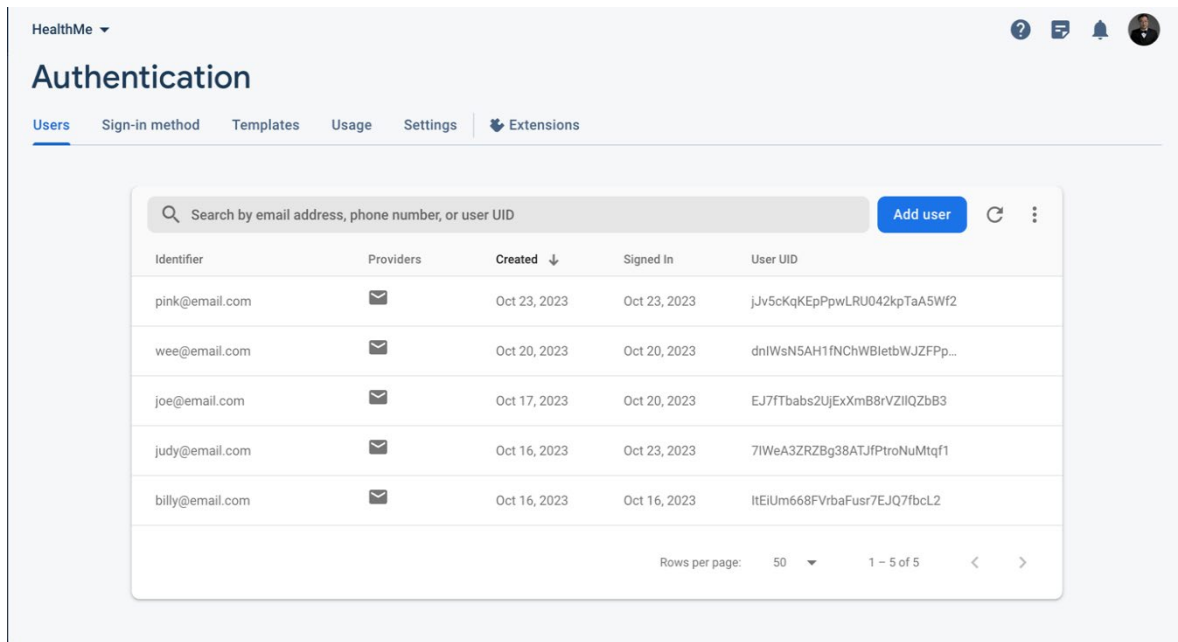
4.2 แสดงฐานข้อมูล Firebase สำหรับแอปพลิเคชันที่พัฒนาสำเร็จ

สำหรับ Firebase ผู้พัฒนาใช้งานในแอปพลิเคชันอยู่ 3 ส่วนคือ

- (1) Authentication with Email/Password
- (2) Cloud Firestore เก็บข้อมูลเป็น Collections
- (3) Cloud Storage ใช้เก็บข้อมูลรูปภาพต่างๆ แล้วเก็บ Link ของรูปภาพนั้นไว้ที่ Cloud

Firestore

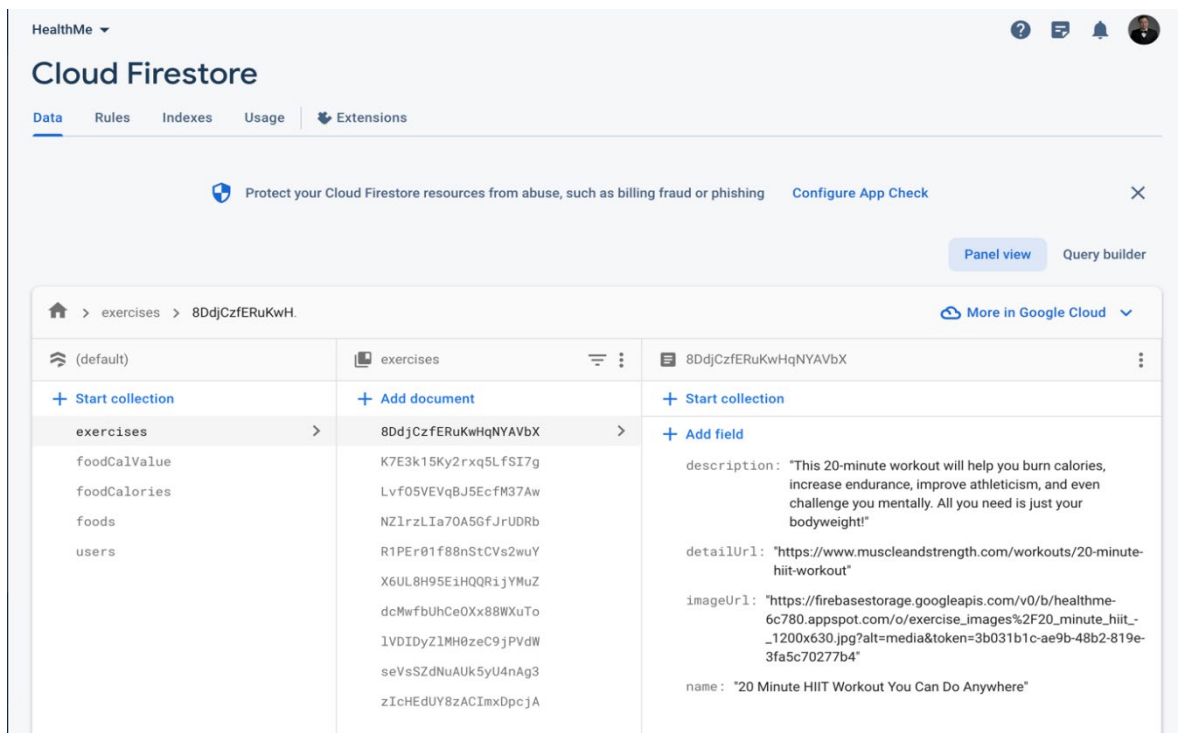
4.2.1 Authentication with Email/Password



รูปที่ 4.13 Firebase Authentication (Email/Password)

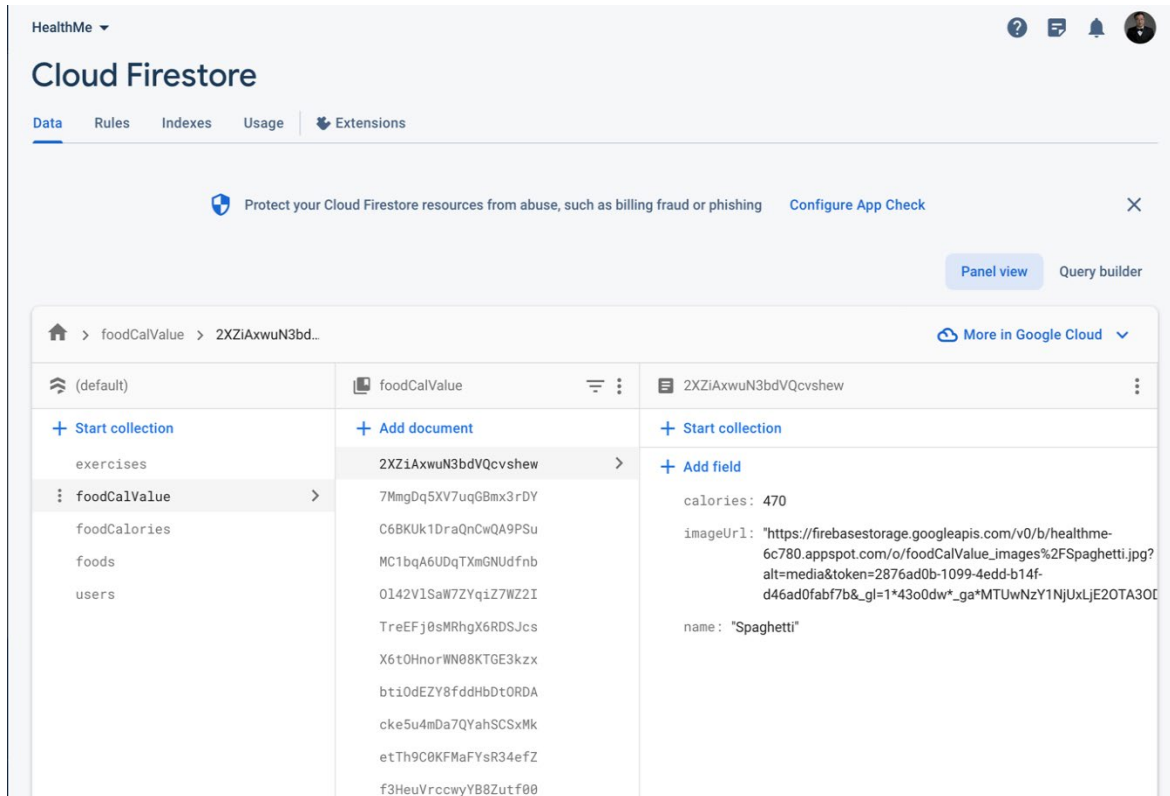
4.2.2 Cloud Firestore เก็บข้อมูลเป็น Collections โดยเก็บข้อมูลตาม Data Model ที่ได้ออกแบบไว้ตามตารางที่ 3.14 ซึ่งได้แก่ Collection ดังต่อไปนี้

(1) exercises collection ตามภาพที่ 4.14 (ตรงกับ Exercises Entity ในตารางที่ 3.14)



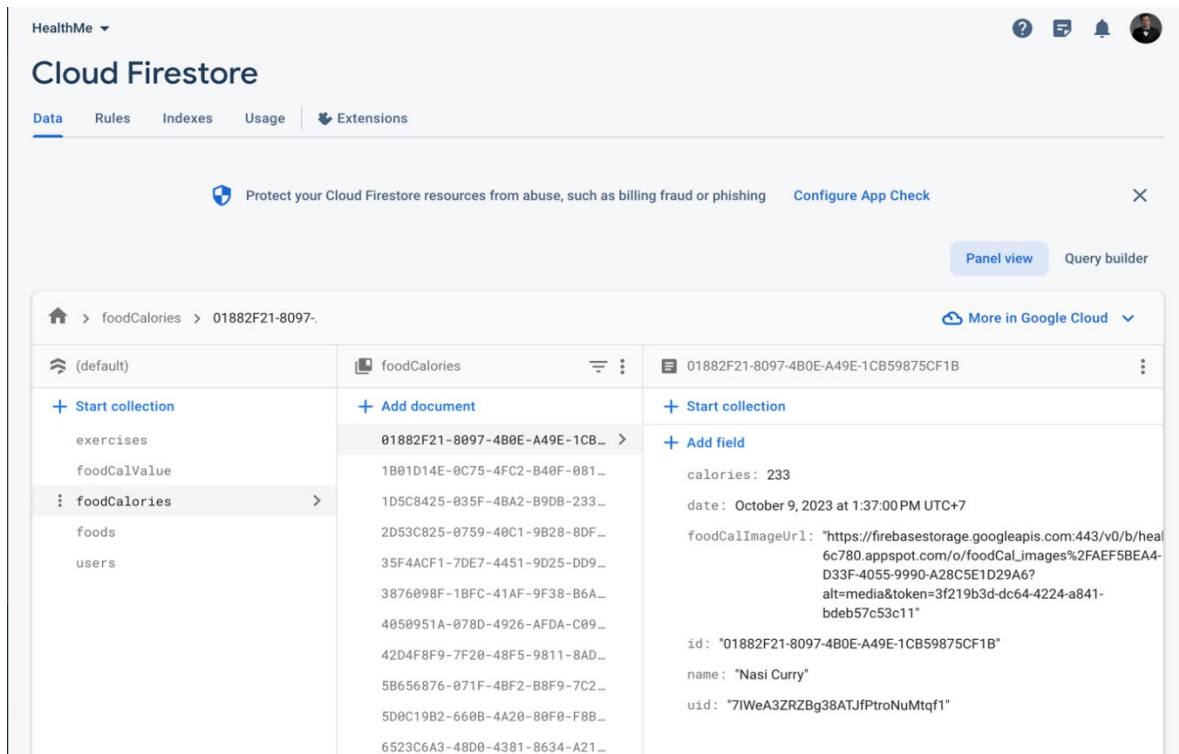
รูปที่ 4.14 exercises collection

(2) foodCalValue collection ตามภาพที่ 4.15(Collection นี้ใช้กับ Predefined Food Calories Value -Search button ใน Add Calories view/Edit Calories view)



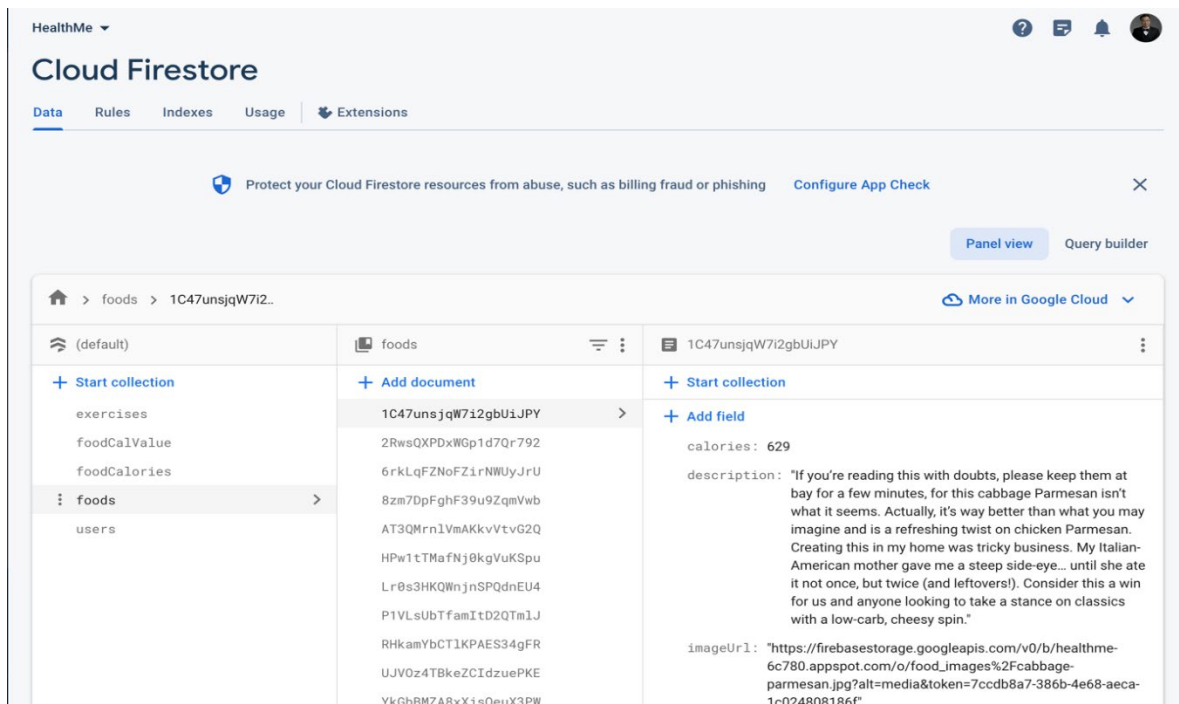
รูปที่ 4.15 foodCalValue collection

(3) foodCalories collection ตามภาพที่ 4.16(ตรงกับ FoodCalories Entity ในตารางที่ 3.14)



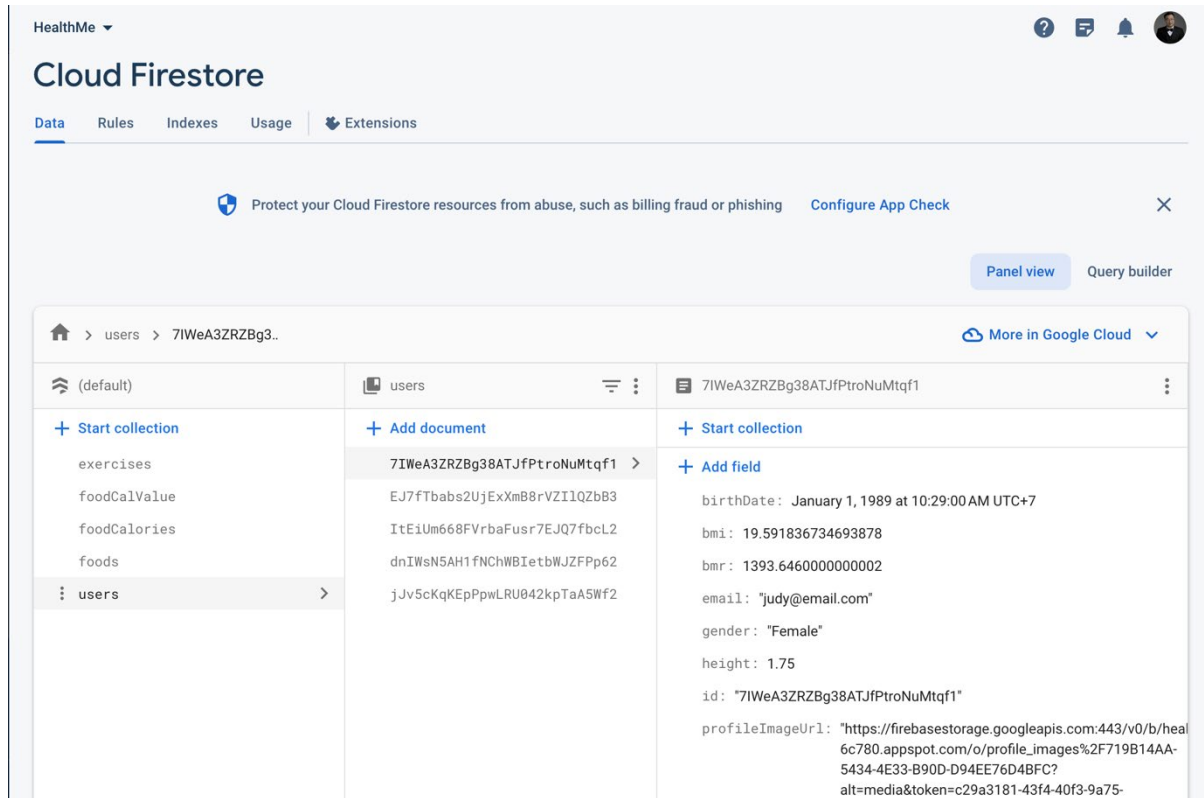
รูปที่ 4.16 foodCalories collection

(4) foods collection ตามภาพที่ 4.17 (ตรงกับ HealthyFoods Entity ในตารางที่ 3.14)



รูปที่ 4.17 foods collection

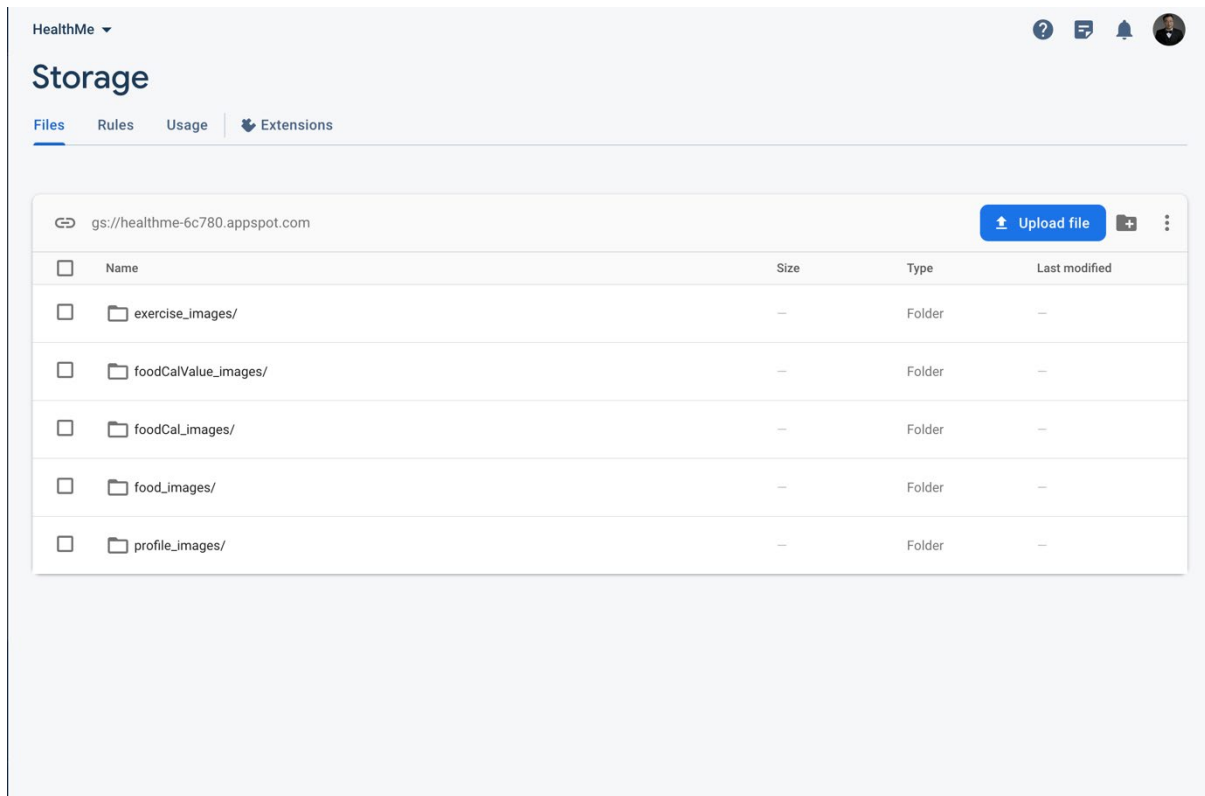
(5) users collection ตามภาพที่ 4.18 (ตรงกับ Users Entity ในตารางที่ 3.14)



รูปที่ 4.18 users collection

4.2.3 Cloud Storage เก็บข้อมูลรูปภาพต่าง ๆ ตาม Folder ดังต่อไปนี้

- (1) exercise_images เก็บรูปสำหรับ Exercise menu view
- (2) foodCalValue_images เก็บรูปสำหรับ Predefined Food Calories Value view (Search button ใน Add Calories view/Edit Calories view)
- (3) foodCal_images เก็บรูปสำหรับ Food Calories card view
- (4) food_images เก็บรูปสำหรับ Food menu view
- (5) profile_images เก็บรูปสำหรับ Profile view



รูปที่ 4.19 Cloud Storage

4.3 แสดงการทดสอบระบบของแอปพลิเคชันที่พัฒนาสำเร็จ

ในการทดสอบ ผู้เขียนได้ทำการแยกเป็น 33 Test cases ตามตารางที่ 4.1 ด้านล่าง

ตารางที่ 4.1 รายละเอียด Test cases ทั้งหมด

Test case number	Test case title	Related Use case
TC01	การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลครบและถูกต้อง	UC01 ลงทะเบียน (Register)
TC02	การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลไม่ครบ	UC01 ลงทะเบียน (Register)
TC03	การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลไม่ถูกต้อง	UC01 ลงทะเบียน (Register)
TC04	การเข้าใช้งาน (Login) - ใส่ข้อมูลครบและถูกต้อง	UC12 การเข้าใช้งาน (Login with Email/Password)
TC05	การเข้าใช้งาน (Login) - ใส่ข้อมูลไม่ครบ	UC12 การเข้าใช้งาน (Login with Email/Password)

ตารางที่ 4.1 (ต่อ)

Test case number	Test case title	Related Use case
TC06	การใช้งาน (Login) - ใส่ข้อมูลไม่ถูกต้อง	UC12 การเข้าใช้งาน (Login with Email/ Password)
TC07	เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) - ใส่ข้อมูลครบ	UC02 เพิ่มข้อมูลอาหารที่รับประทาน (Add food info)
TC08	เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) - ใส่ข้อมูลไม่ครบ	UC02 เพิ่มข้อมูลอาหารที่รับประทาน (Add food info)
TC09	เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) - โดยกดปุ่ม Search	UC02 เพิ่มข้อมูลอาหารที่รับประทาน (Add food info)
TC10	แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) - ใส่ข้อมูลครบ	UC03 แก้ไขข้อมูลอาหารที่รับประทานที่เพิ่มไปแล้ว (Edit Food info)
TC11	แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) - ใส่ข้อมูลไม่ครบ	UC03 แก้ไขข้อมูลอาหารที่รับประทานที่เพิ่มไปแล้ว (Edit Food info)
TC12	แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) - โดยกดปุ่ม Search	UC03 แก้ไขข้อมูลอาหารที่รับประทานที่เพิ่มไปแล้ว (Edit Food info)
TC13	ลบข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Delete food calories info)	UC04 ลบข้อมูลอาหารที่รับประทานที่เพิ่มไปแล้ว (Delete Food)
TC14	แสดงข้อมูลอาหารที่เพิ่มไว้แล้วในหน้า Calories Tracker view	UC05 แสดงข้อมูลอาหารที่ Track Calories
TC15	แสดงข้อมูลแคลอรีรวมแต่ละวัน ย้อนหลัง 7 วัน	UC06 แสดงข้อมูลแคลอรีรวมในแต่ละวัน ย้อนหลัง 7 วันแบบกราฟแท่ง (Show Total Graph)
TC16	แสดงข้อมูลแคลอรีและอาหารที่ผู้ใช้ทั้งหมดเพิ่มไว้ในระบบ	UC07 แสดงข้อมูลแคลอรีและอาหารที่ผู้ใช้ทั้งหมดได้ทำการเพิ่มไว้ในระบบ (Show Food For All Users)

ตาราง 4.1 (ต่อ)

Test case number	Test case title	Related Use case
TC17	แสดงข้อมูลอาหารสุขภาพที่แนะนำ พร้อมรายละเอียด	UC08 แสดงข้อมูลอาหารสุขภาพที่แนะนำ (Show Healthy Food List/Detail)
TC18	แสดงข้อมูลการออกกำลังกายที่แนะนำ พร้อมรายละเอียด	UC09 แสดงข้อมูลการออกกำลังกายที่แนะนำ (Show Exercise List/Detail)
TC19	แสดงข้อมูลผู้ใช้ (Profile view)	UC10 แสดงข้อมูลผู้ใช้/แก้ไข (Show/Edit User Profile View)
TC20	แสดงหน้า Edit Profile view - ใส่ข้อมูลครบและถูกต้อง	UC10 แสดงข้อมูลผู้ใช้/แก้ไข (Show/Edit User Profile View)
TC21	แสดงหน้า Edit Profile view - ใส่ข้อมูลไม่ครบ	UC10 แสดงข้อมูลผู้ใช้/แก้ไข (Show/Edit User Profile View)
TC22	แสดงหน้า Edit Profile view - ใส่ข้อมูลไม่ถูกต้อง	UC10 แสดงข้อมูลผู้ใช้/แก้ไข (Show/Edit User Profile View)
TC23	แสดงลำดับผู้ใช้ตามแต้มสะสม (Show Leaderboard)	UC11 แสดงลำดับของผู้ใช้ทั้งหมดที่มีแต้มสะสมเรียงจากมากไปน้อย (Show Leaderboard)
TC24	แสดงหน้าอาหารที่ผู้ใช้แต่ละคนเพิ่มไว้	UC11 แสดงลำดับของผู้ใช้ทั้งหมดที่มีแต้มสะสมเรียงจากมากไปน้อย (Show Leaderboard)
TC25	ทดสอบใส่ค่าน้ำหนักที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า CreateAccountView	UC01 ลงทะเบียน (Register)
TC26	ทดสอบใส่ค่าส่วนสูงที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า CreateAccountView	UC01 ลงทะเบียน (Register)
TC27	ทดสอบใส่ค่า Username เป็นภาษาไทยในหน้า CreateAccountView	UC01 ลงทะเบียน (Register)
TC28	ทดสอบใส่ค่า Calories field ที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า AddCaloriesView	UC02 เพิ่มข้อมูลอาหารที่รับประทาน (Add food info)

ตารางที่ 4.1 (ต่อ)

Test case number	Test case title	Related Use case
TC29	ทดสอบใส่ค่า Calories field ที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า EditCaloriesView	UC03 แก้ไขข้อมูลอาหารที่รับประทานที่เพิ่มไปแล้ว (Edit Food info)
TC30	ทดสอบใส่ค่า Name field เป็นภาษาไทยในหน้า AddCaloriesView	UC02 เพิ่มข้อมูลอาหารที่รับประทาน (Add food info)
TC31	ทดสอบใส่ค่า Name field เป็นภาษาไทยในหน้า EditCaloriesView	UC03 แก้ไขข้อมูลอาหารที่รับประทานที่เพิ่มไปแล้ว (Edit Food info)
TC32	ทดสอบเลือกอาหารภาษาไทยในหน้า Select your calories โดยกดปุ่ม Search ในหน้า AddCaloriesView หรือ EditCaloriesView [แสดงตัวอย่างที่หน้า AddCaloriesView]	UC02 เพิ่มข้อมูลอาหารที่รับประทาน (Add food info) และ UC03 แก้ไขข้อมูลอาหารที่รับประทานที่เพิ่มไปแล้ว (Edit Food info)
TC33	Sign out ออกจากระบบ	UC12 การเข้าใช้งาน (Login with Email/Password)

และตารางที่ 4.2 ถึง ตารางที่ 4.34 ต่อไปนี้ คือรายละเอียดและผลของแต่ละ Test case โดยทั้งหมดทำบน Test Environment คือ iPhone 15 Pro iOS 17.0.1 (Simulator) แต่ระบบที่รองรับคือ iOS 16.4 ขึ้นไป

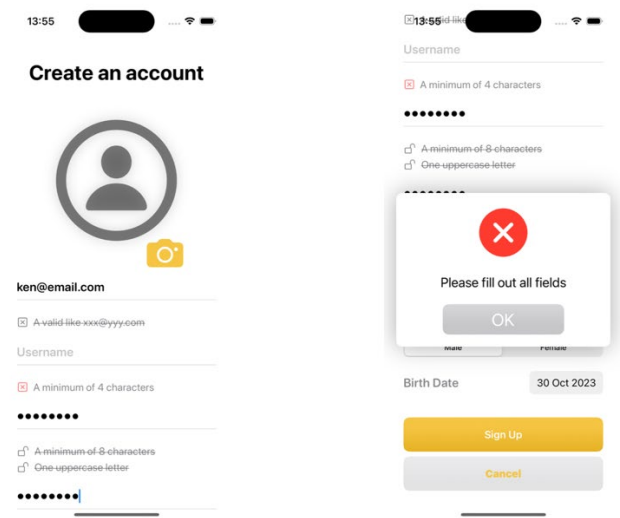
ตารางที่ 4.2 TC01 การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลครบและถูกต้อง

Test case number	TC01
Test case title	การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลครบและถูกต้อง
Precondition	Download App
Test steps	<p>(4) กดปุ่ม Sign up ในหน้า Login</p> <p>(5) ระบบนำสู่หน้า Create user account</p> <p>(6) ผู้ใช้กรอกข้อมูลครบถ้วน และถูกต้อง</p> <p>(7) ระบบ Create user account ใน Firebase Authentication และ user ใน users collection ใน Cloud Firestore</p> <p>(8) App นำผู้ใช้ไปหน้า Main view (Calories Tracker view)</p>
Test data	<ol style="list-style-type: none"> 1. Profile image 2. Email (A valid like xxx@yyy.com): ken@email.com 3. Username (Minimum 4 characters): Ken The Rock 4. Password (Minimum 8 characters/One uppercase letter): 1234567A 5. Confirm Password (Same as Password): 1234567A 6. Weight(kg) (if not fill, default 50 kg): 66 7. Height(m) (if not fill, default 1.65 m): 1.75 8. Choose between Male or Female: Male 9. Birthdate (default is Today): Oct 11, 2000
Expected result	ระบบสร้าง User ใน Firebase Authentication/Cloud Firestore และนำผู้ใช้เข้าสู่หน้า Main view (Calories Tracker View)
Actual result	ระบบสร้าง User ใน Firebase Authentication/Cloud Firestore และนำผู้ใช้เข้าสู่หน้า Main view (Calories Tracker View)
Status	Passed
Result Screens	

ตารางที่ 4.3 TC02 การลงทะเบียนผู้ใช้ใหม่ (Create new user account) – ใส่ข้อมูลไม่ครบ

Test case number	TC02
Test case title	การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลไม่ครบ
Precondition	Download App
Test steps	<ol style="list-style-type: none"> 1. กดปุ่ม Sign up ในหน้า Login 2. ระบบนำสู่หน้า Create user account 3. ผู้ใช้กรอกข้อมูลไม่ครบถ้วน 4. ระบบแสดง Alert view แจ้งให้ผู้ใช้ทราบว่าใส่ข้อมูลไม่ครบ
Test data	<ol style="list-style-type: none"> 1. Profile image – ไม่เลือก (จะไม่มีรูปในระบบ แต่ไม่กำหนดว่าต้องใส่ มาใส่ทีหลังใน Edit Profile view ได้) 2. Email (A valid like xxx@yyy.com): ken@email.com 3. Username (Minimum 4 characters): ไม่ใส่ 4. Password (Minimum 8 characters/One uppercase letter): 1234567A 5. Confirm Password (Same as Password): 1234567A 6. Weight(kg) (if not fill, default 50 kg): ไม่ใส่

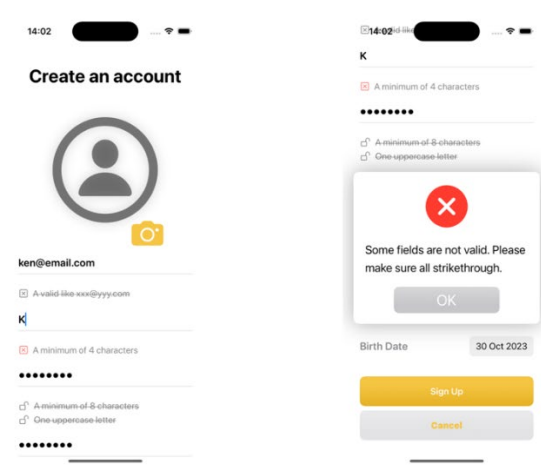
ตารางที่ 4.3 (ต่อ)

Test case number	TC02
Test case title	การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลไม่ครบ
Test data	<p>7. Height(m) (if not fill, default 1.65 m): ไม่ใส่</p> <p>8. Choose between Male or Female: Male</p> <p>9. Birthdate (default is Today): ไม่เลือก</p>
Expected result	ระบบแสดง Alert view ว่าใส่ข้อมูลไม่ครบ
Actual result	ระบบแสดง Alert view ว่าใส่ข้อมูลไม่ครบ
Status	Passed
Result Screens	 <p>The image shows two screenshots of a mobile application's 'Create an account' screen. The left screenshot displays the registration form with fields for email (ken@email.com), password, and birth date (30 Oct 2023). The right screenshot shows the same form with a red error message overlay that says 'Please fill out all fields'.</p>

ตารางที่ 4.4 TC03 การลงทะเบียนผู้ใช้ใหม่ (Create new user account) – ใส่ข้อมูลไม่ถูกต้อง

Test case number	TC03
Test case title	การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลไม่ถูกต้อง
Precondition	Download App
Test steps	<ol style="list-style-type: none"> 1. กดปุ่ม Sign up ในหน้า Login 2. ระบบนำสู่หน้า Create user account 3. ผู้ใช้กรอกข้อมูลไม่ถูกต้อง 4. ระบบแสดง Alert view แจ้งให้ผู้ใช้ทราบว่าใส่ข้อมูลไม่ถูกต้อง
Test data	<ol style="list-style-type: none"> 1. Profile image – ไม่เลือก (จะไม่มีรูปในระบบ แต่ไม่กำหนดว่าต้องใส่ มาใส่ทีหลังใน Edit Profile view ได้) 2. Email (A valid like xxx@yyy.com): ken@email.com 3. Username (Minimum 4 characters): K 4. Password (Minimum 8 characters/One uppercase letter): 1234567A 5. Confirm Password (Same as Password): 1234567A 6. Weight(kg) (if not fill, default 50 kg): ไม่ใส่ 7. Height(m) (if not fill, default 1.65 m): ไม่ใส่ 8. Choose between Male or Female: Male 9. Birthdate (default is Today): ไม่เลือก
Expected result	ระบบแสดง Alert view ว่าใส่ข้อมูลไม่ถูกต้อง
Actual result	ระบบแสดง Alert view ว่าใส่ข้อมูลไม่ถูกต้อง
Status	Passed

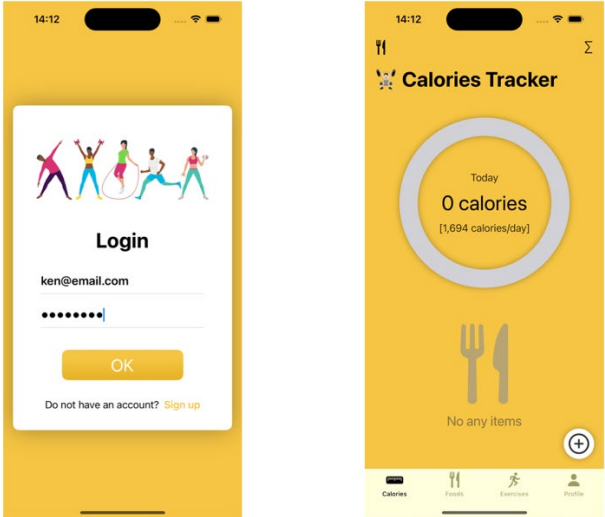
ตารางที่ 4.4 (ต่อ)

Test case number	TC03
Test case title	การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลไม่ถูกต้อง
Result Screens	

ตารางที่ 4.5 TC04 การเข้าใช้งาน (Login) – ใส่ข้อมูลครบและถูกต้อง

Test case number	TC04
Test case title	การเข้าใช้งาน (Login) - ใส่ข้อมูลครบและถูกต้อง
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. เปิดแอปจะเจอหน้า Login 2. กรอก Email/Password ที่ถูกต้อง 3. กดปุ่ม OK ระบบจะตรวจสอบ Firebase Authentication 4. ถ้าถูกต้องระบบจะนำไปสู่หน้า Main view (Calories Tracker view)
Test data	<ol style="list-style-type: none"> 1. Email: ken@email.com 2. Password: 1234567A
Expected result	ระบบตรวจสอบกับ Firebase Authentication และระบบจะแสดงหน้า Main view (Calories Tracker view)

ตารางที่ 4.5 (ต่อ)

Test case number	TC04
Test case title	การเข้าใช้งาน (Login) - ใส่ข้อมูลครบและถูกต้อง
Actual result	ระบบตรวจสอบกับ Firebase Authentication และระบบจะแสดงหน้า Main view (Calories Tracker view)
Status	Passed
Result Screens	

ตารางที่ 4.6 TC05 การเข้าใช้งาน (Login) – ใส่ข้อมูลไม่ครบ

Test case number	TC05
Test case title	การเข้าใช้งาน (Login) - ใส่ข้อมูลไม่ครบ
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. เปิดแอปจะเจอหน้า Login 2. กรอก Email แต่ไม่กรอก Password 3. กดปุ่ม OK 4. ระบบแสดง Alert view ว่ากรอกข้อมูลไม่ครบ
Test data	<ol style="list-style-type: none"> 5. Email: ken@email.com 6. Password: ไม่กรอก

ตารางที่ 4.6 (ต่อ)

Test case number	TC05
Test case title	การเข้าใช้งาน (Login) - ใส่ข้อมูลไม่ครบ
Expected result	ระบบแสดง Alert view ว่ากรอกข้อมูลไม่ครบ
Actual result	ระบบแสดง Alert view ว่ากรอกข้อมูลไม่ครบ
Status	Passed
Result Screens	

ตารางที่ 4.7 TC06 การเข้าใช้งาน (Login) – ใส่ข้อมูลไม่ถูกต้อง

Test case number	TC06
Test case title	การเข้าใช้งาน (Login) - ใส่ข้อมูลไม่ถูกต้อง
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. เปิดแอปจะเจอหน้า Login 2. กรอก Email กรอก Password 3. กดปุ่ม OK 4. ระบบแสดง Alert view ว่ากรอกข้อมูลไม่ครบ

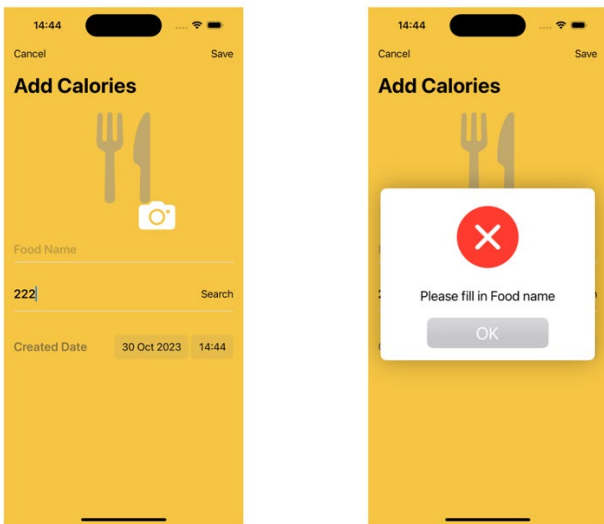
ตารางที่ 4.7 (ต่อ)

Test case number	TC06
Test case title	การเข้าใช้งาน (Login) - ใส่ข้อมูลไม่ถูกต้อง
Test data	5. Email: ken@email.com 6. Password: 12345
Expected result	ระบบจะตรวจสอบกับ Firebase Authentication และแสดง Alert view ว่าข้อมูลไม่ถูกต้อง
Actual result	แสดง Alert view ว่าข้อมูลไม่ถูกต้อง
Status	Passed
Result Screens	

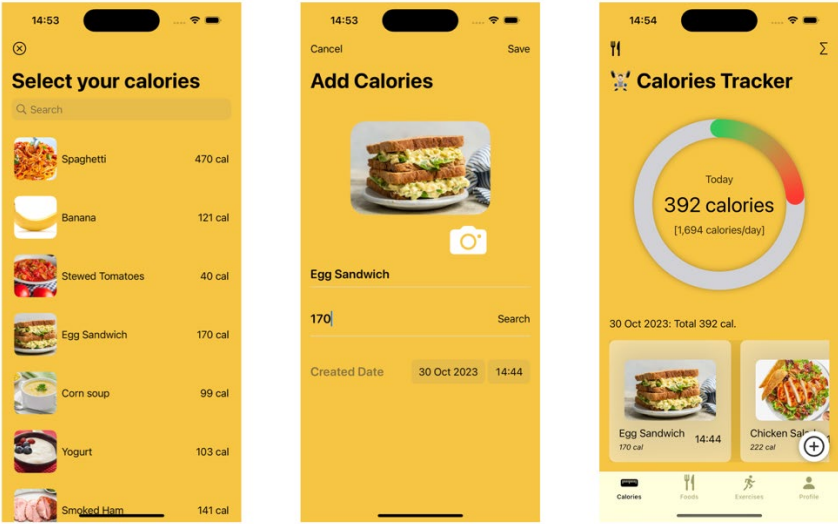
ตารางที่ 4.8 TC07 เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) – ใส่ข้อมูลครบ

Test case number	TC07
Test case title	เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) - ใส่ข้อมูลครบ
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. หน้า Calories Tracker view กดปุ่ม Add (Plus) 2. ระบบนำไปสู่หน้า Add Calories 3. กรอกข้อมูลครบถ้วน 4. กดปุ่ม Save 5. ระบบจะแสดง Food Calories Card view สำหรับอาหารที่เพิ่มเข้าไป ในหน้า Calories Tracker view
Test data	<ol style="list-style-type: none"> 1. รูปอาหาร 2. ชื่ออาหาร: Chicken Salad 3. ปริมาณแคลอรี: 222 cal 4. วันเวลา: 30 Oct 2023 14.28
Expected result	ระบบจะเพิ่มข้อมูลอาหารไปใน Cloud Firestore และจะแสดง Food Calories Card view สำหรับอาหารที่เพิ่มเข้าไป ในหน้า Calories Tracker view
Actual result	แสดง Food Calories Card view สำหรับอาหารที่เพิ่มเข้าไป ในหน้า Calories Tracker view
Status	Passed
Result Screens	

ตารางที่ 4.9 TC08 เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) – ใส่ข้อมูลไม่ครบ

Test case number	TC08
Test case title	เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) – ใส่ข้อมูลไม่ครบ
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. หน้า Calories Tracker view กดปุ่ม Add (Plus) 2. ระบบนำไปสู่หน้า Add Calories 3. กรอกข้อมูลไม่ครบ 4. กดปุ่ม Save 5. ระบบจะแสดง Alert view ว่าใส่ข้อมูลไม่ครบ
Test data	<ol style="list-style-type: none"> 1. รูปอาหาร (ไม่ใส่ก็ได้ มาใส่ตอน Edit Calories ก็ได้) 2. ชื่ออาหาร: ไม่ใส่ 3. ปริมาณแคลอรี: 222 cal 4. วันเวลา: เวลาปัจจุบัน (ค่า Default)
Expected result	ระบบแสดง Alert view ว่าใส่ข้อมูลไม่ครบ
Actual result	ระบบแสดง Alert view ว่าใส่ข้อมูลไม่ครบ
Status	Passed
Result Screens	

ตารางที่ 4.10 TC09 เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) – โดยกดปุ่ม Search

Test case number	TC09
Test case title	เพิ่มข้อมูลอาหารและรายละเอียด (Add food calories info) – โดยกดปุ่ม Search
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. หน้า Calories Tracker view กดปุ่ม Add (Plus) 2. ระบบนำไปสู่หน้า Add Calories 3. กดปุ่ม Search ระบบจะนำสู่หน้า Predefined Food Calories (มีทั้งรูป/ชื่ออาหาร/แคลอรี) ผู้ใช้สามารถเลือกจากรายการได้เลย 4. ระบบจะกลับมาหน้า Add Calories โดย filled ข้อมูลตามอาหารที่เลือกมา 5. กดปุ่ม Save 6. ระบบจะแสดง Food Calories Card view สำหรับอาหารที่เพิ่มเข้าไป ในหน้า Calories Tracker view
Test data	1. เลือกหาอาหารในหน้า Search ข้อมูลจะ Fill out ในหน้า Add Calories อัตโนมัติ
Expected result	ระบบแสดงอาหารที่เลือกจากการ Search โดยข้อมูลจะ Filled out ในหน้า Add Calories อัตโนมัติ กดปุ่ม Save ระบบจะแสดง Food Calories Card view สำหรับอาหารที่เพิ่มเข้าไป ในหน้า Calories Tracker view
Actual result	Filled out ข้อมูลในหน้า Add Calories อัตโนมัติ กดปุ่ม Save ระบบจะแสดง Food Calories Card view สำหรับอาหารที่เพิ่มเข้าไป ในหน้า Calories Tracker view
Status	Passed
Result Screens	

ตารางที่ 4.11 TC10 แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) – ใส่ข้อมูลครบ

Test case number	TC10
Test case title	แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) - ใส่ข้อมูลครบ
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. หน้า Calories Tracker view กดที่ Food Calories card view ที่ต้องการจะแก้ไข 2. ระบบแสดง Confirm Dialog เลือก Edit Food Calories 3. ระบบแสดงหน้า Edit Calories โดย Filled out ข้อมูลเดิมก่อนจะแก้ไข 4. แก้ไข Fields ที่ต้องการ 5. กด Save 6. ระบบจะแสดง Food Calories Card view สำหรับอาหารที่แก้ไขไป ในหน้า Calories Tracker view
Test data	แก้ไข Chicken salad เปลี่ยนค่าแคลอรี เป็น 255 cal
Expected result	ระบบจะแก้ไขแคลอรีของ Chicken salad เป็น 255 cal แล้วแสดง Food Calories card view ที่แก้ไขแล้วในหน้า Calories Tracker view
Actual result	ระบบแก้ไขแคลอรีของ Chicken salad เป็น 255 cal แล้วแสดง Food Calories card view ที่แก้ไขแล้วในหน้า Calories Tracker view
Status	Passed
Result Screens	

ตารางที่ 4.12 TC11 แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) – ใส่ข้อมูลไม่ครบ

Test case number	TC11
Test case title	แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) - ใส่ข้อมูลไม่ครบ
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. หน้า Calories Tracker view กดที่ Food Calories card view ที่ต้องการจะแก้ไข 2. ระบบแสดง Confirm Dialog เลือก Edit Food Calories 3. ระบบแสดงหน้า Edit Calories โดย Filled out ข้อมูลเดิมก่อนจะแก้ไข 4. ไม่ใส่ชื่ออาหารใน Name field 5. กด Save 6. ระบบจะแสดง Alert view แจ้งว่าใส่ข้อมูลไม่ครบ
Test data	แก้ไข Chicken salad เอาชื่ออาหารออก
Expected result	ระบบจะแสดง Alert view แจ้งว่าใส่ข้อมูลไม่ครบ
Actual result	ระบบจะแสดง Alert view แจ้งว่าใส่ข้อมูลไม่ครบ
Status	Passed
Result Screens	

ตารางที่ 4.13 TC12 แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) - โดยกดปุ่ม Search

Test case number	TC12
Test case title	แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) - โดยกดปุ่ม Search
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. หน้า Calories Tracker view กดที่ Food Calories card view ที่ต้องการจะแก้ไข 2. ระบบแสดง Confirm Dialog เลือก Edit Food Calories 3. ระบบแสดงหน้า Edit Calories โดย Filled out ข้อมูลเดิมก่อนจะแก้ไข 4. กดปุ่ม Search ระบบจะนำสู่หน้า Predefined Food Calories (มีทั้งรูป/ชื่ออาหาร/แคลอรี) ผู้ใช้สามารถเลือกจากรายการได้เลย 5. กดเลือกอาหารที่ต้องการ ระบบจะกลับสู่หน้า Edit Calories โดย Filled out ทุก Fields ตามที่เลือก 6. กดปุ่ม Save 7. ระบบจะแสดง Food Calories Card view สำหรับอาหารที่แก้ไขไป ในหน้า Calories Tracker view
Test data	แก้ไข Chicken salad กดปุ่ม Search เลือก Spaghetti
Expected result	ระบบแสดงอาหารที่เลือกจากการ Search โดยข้อมูลจะ Filled out ในหน้า Edit Calories อัตโนมัติ กดปุ่ม Save ระบบจะแสดง Food Calories Card view สำหรับอาหารที่ได้แก้ไขเข้าไป ในหน้า Calories Tracker view
Actual result	ระบบแสดงอาหารที่แก้ไขแสดงเป็น Food Calories Card view สำหรับอาหารที่ได้แก้ไขเข้าไป ในหน้า Calories Tracker view
Status	Passed

ตารางที่ 4.13 (ต่อ)

Test case number	TC12
Test case title	แก้ไขข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Edit food calories info) - โดยกดปุ่ม Search
Result Screens	

ตารางที่ 4.14 TC13 ลบข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Delete food calories info)

Test case number	TC13
Test case title	ลบข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Delete food calories info)
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. หน้า Calories Tracker view กดที่ Food Calories card view ที่ต้องการจะแก้ไข 2. ระบบแสดง Confirm Dialog เลือก Delete Food Calories 3. ระบบจะทำการลบเอกสารใน Cloud Firestore 4. ระบบจะแสดง Calories Tracker view ที่ไม่มี Card view ของ Food ที่ลบทิ้งไป
Test data	เลือกลบ Spaghetti Food Calories card view
Expected result	ระบบจะแสดง Calories Tracker view ที่ไม่มี Card view ของ Food ที่ลบทิ้งไป
Actual result	ระบบแสดง Calories Tracker view ที่ไม่มี Card view ของ Food ที่ลบทิ้งไป

ตารางที่ 4.14 (ต่อ)

Test case number	TC13
Test case title	ลบข้อมูลอาหารที่เพิ่มเข้าไปแล้ว (Delete food calories info)
Status	Passed
Result Screens	

ตารางที่ 4.15 TC14 แสดงข้อมูลอาหารที่เพิ่มไว้แล้วในหน้า Calories Tracker view

Test case number	TC14
Test case title	แสดงข้อมูลอาหารที่เพิ่มไว้แล้วในหน้า Calories Tracker view
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> กด Tab view หน้า Calories Tracker view จะมีแสดง Progress Ring view แสดงปริมาณ Calories ของอาหารที่เพิ่มเข้าไปในวันนี้ (วันปัจจุบัน) โดยมีปริมาณแคลอรีที่ผู้ใช้ผู้นั้นสามารถบริโภคได้ในแต่ละวัน จะมีแสดง Food Calories card view ของอาหารที่เพิ่มเข้าไปของผู้ใช้ท่านนี้ทั้งหมด โดยเรียงตามวันเวลาจากวันปัจจุบัน ไปไปในอดีต โดยในแต่ละวัน จะมีการแสดงปริมาณแคลอรีที่บริโภคทั้งหมดในวันนั้นที่เพิ่มไว้ในระบบ
Test data	ไม่มี

ตารางที่ 4.15 (ต่อ)

Test case number	TC14
Test case title	แสดงข้อมูลอาหารที่เพิ่มไว้แล้วในหน้า Calories Tracker view
Expected result	ระบบจะแสดง Calories Tracker view ซึ่งมี Progress View และ Food Calories card view สำหรับอาหารที่เพิ่มไว้ในแต่ละวัน
Actual result	ระบบแสดง Calories Tracker view ซึ่งมี Progress View และ Food Calories card view สำหรับอาหารที่เพิ่มไว้ในแต่ละวัน
Status	Passed
Result Screens	

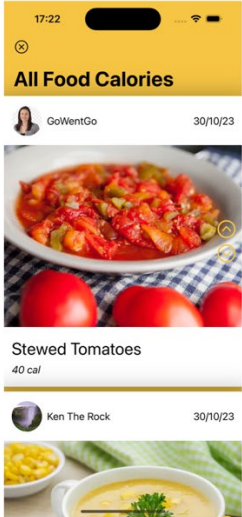
ตารางที่ 4.16 TC15 แสดงข้อมูลแคลอรีรวมแต่ละวัน ย้อนหลัง 7 วัน

Test case number	TC15
Test case title	แสดงข้อมูลแคลอรีรวมแต่ละวัน ย้อนหลัง 7 วัน
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. กด Tab view หน้า Calories Tracker view 2. กดปุ่ม Sum 3. ระบบจะแสดงหน้า Calories Stat 4. ในหน้านี้จะแสดงข้อมูลแคลอรีทั้งหมดในแต่ละวันของอาหารที่เพิ่มไว้ 7 วันย้อนหลัง เป็นกราฟแท่ง

ตารางที่ 4.16 (ต่อ)

Test case number	TC15
Test case title	แสดงข้อมูลแคลอรีรวมแต่ละวัน ย้อนหลัง 7 วัน
Test steps	5. ในหน้านี้ยังแสดงวันที่ยอดเยี่ยม(ปริมาณแคลอรีน้อยที่สุด) และวันที่ยอดแย่(ปริมาณแคลอรีมากที่สุด)
Test data	ไม่มี
Expected result	ระบบจะแสดง หน้า Calories Stat แสดงกราฟแท่ง 7 วันย้อนหลัง และแสดงวันยอดเยี่ยมและยอดแย่
Actual result	ระบบแสดง หน้า Calories Stat แสดงกราฟแท่ง 7 วันย้อนหลัง และแสดงวันยอดเยี่ยมและยอดแย่
Status	Passed
Result Screens	

ตารางที่ 4.17 TC16 แสดงข้อมูลแคลอรีและอาหารที่ผู้ใช้ทั้งหมดเพิ่มไว้ในระบบ

Test case number	TC16
Test case title	แสดงข้อมูลแคลอรีและอาหารที่ผู้ใช้ทั้งหมดเพิ่มไว้ในระบบ
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. กด Tab view หน้า Calories Tracker view 2. กดปุ่ม Food 3. ระบบจะแสดงหน้า All Food Calories added by users 4. ในหน้านี้จะแสดงข้อมูลแคลอรีของอาหารที่แต่ผู้ใช้แต่ละคนได้เพิ่มไว้ เรียงตามวันเวลาปัจจุบัน ไล่ย้อนหลังไปในอดีต
Test data	ไม่มี
Expected result	ระบบจะแสดง หน้า All Food Calories added by users เรียงตามวันเวลาปัจจุบัน ไล่ย้อนหลังไปในอดีต
Actual result	ระบบแสดง หน้า All Food Calories added by users เรียงตามวันเวลาปัจจุบัน ไล่ย้อนหลังไปในอดีต
Status	Passed
Result Screens	 

ตารางที่ 4.18 TC17 แสดงข้อมูลอาหารสุขภาพที่แนะนำ พร้อมรายละเอียด

Test case number	TC17
Test case title	แสดงข้อมูลอาหารสุขภาพที่แนะนำ พร้อมรายละเอียด
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. กด Tab view หน้า Food Menu view 2. จะแสดงรายการอาหารที่แนะนำของระบบ (ดึงข้อมูลจาก Cloud Firestore, foods collection) 3. ผู้ใช้สามารถกดที่ปุ่ม Info ระบบจะแสดง Food Menu Detail view จะมีข้อมูลเพิ่มเติมของอาหารที่แนะนำนั้น 4. ในหน้า Detail view สามารถกด Link ระบบจะแสดง Web site ที่มีข้อมูลเพิ่มเติมของอาหารนั้น (มี Built-in web browser)
Test data	ไม่มี
Expected result	ระบบจะแสดง Food Menu view/Food Menu Detail view/Web site แสดงข้อมูลเพิ่มเติม
Actual result	ระบบแสดง Food Menu view/Food Menu Detail view/Web site แสดงข้อมูลเพิ่มเติม
Status	Passed
Result Screens	  

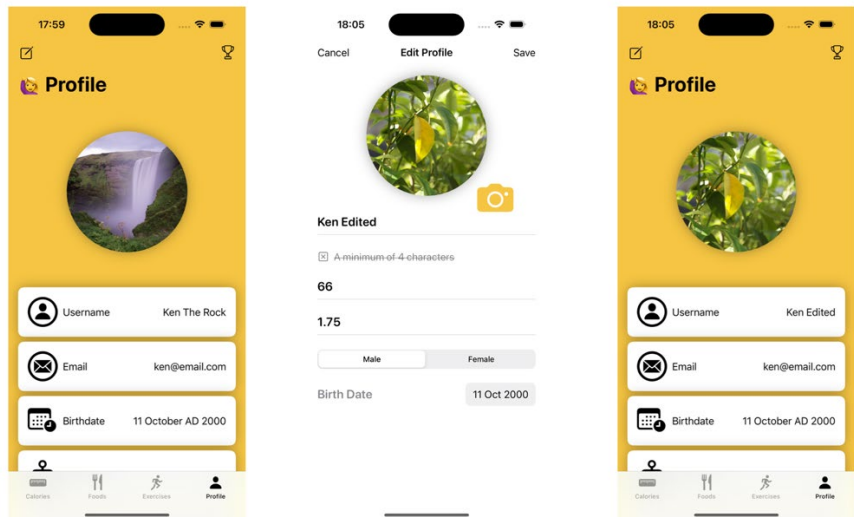
ตารางที่ 4.19 TC18 แสดงข้อมูลการออกกำลังกายที่แนะนำ พร้อมรายละเอียด

Test case number	TC18
Test case title	แสดงข้อมูลการออกกำลังกายที่แนะนำ พร้อมรายละเอียด
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. กด Tab view หน้า Exercises view 2. จะแสดงรายการออกกำลังกายที่แนะนำของระบบ (ดึงข้อมูลจาก Cloud Firestore, exercises collection) 3. ผู้ใช้สามารถกดที่ปุ่ม Info ระบบจะแสดง Exercise Detail view จะมีข้อมูลเพิ่มเติมของการออกกำลังกายที่แนะนำนั้น 4. ในหน้า Detail view สามารถกด Link ระบบจะแสดง Web site ที่มีข้อมูลเพิ่มเติมของการออกกำลังกายนั้น (มี Built-in web browser)
Test data	ไม่มี
Expected result	ระบบจะแสดง Exercises view/Exercise Detail view/Web site แสดงข้อมูลเพิ่มเติม
Actual result	ระบบแสดง Exercises view/Exercise Detail view/Web site แสดงข้อมูลเพิ่มเติม
Status	Passed
Result Screens	 <p>The screenshots show the app's interface for displaying exercise information. The first screenshot shows the 'Exercises' tab with a search bar and a list of exercises, including a '20 Minute HIIT Workout You Can Do Anywhere'. The second screenshot shows the 'Exercise Detail view' for the same workout, featuring a description: 'This 20-minute workout will help you burn calories, increase endurance, improve athleticism, and even challenge you mentally. All you need is just your bodyweight!' and a button to 'Go to 20 Minute HIIT Workout You Can Do Anywhere detail'. The third screenshot shows the 'Web site' with a 'BLACK FRIDAY SNEAK PEEK!' banner, the MS logo, and the same exercise title and description as the detail view.</p>

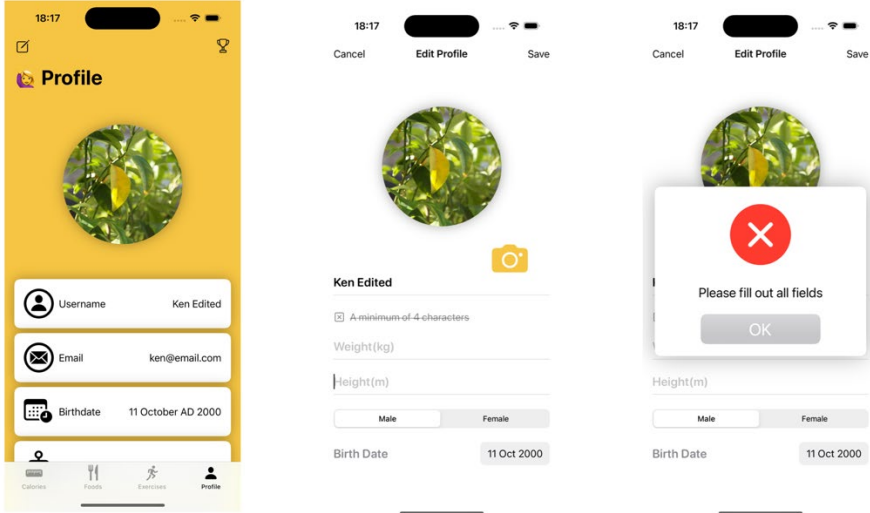
ตารางที่ 4.20 TC19 แสดงข้อมูลผู้ใช้ (Profile view)

Test case number	TC19
Test case title	แสดงข้อมูลผู้ใช้ (Profile view)
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> กด Tab view หน้า Profile view จะแสดงข้อมูลของผู้ใช้คนนั้น (ที่ Login อยู่) จาก Cloud Firestore, users collection โดยมีปุ่ม Sign out เพื่อออกจากระบบอยู่ด้านล่างด้วย
Test data	ไม่มี
Expected result	ระบบจะแสดง Profile view ของผู้ใช้ที่ Login อยู่โดยจะแสดงProfile image/Username/Email/Birthdate/Weight(kg)/Height(m)/Gender/BMI/Calories per day
Actual result	ระบบแสดง Profile view ของผู้ใช้ที่ Login อยู่โดยจะแสดงProfile image/Username/Email/Birthdate/Weight(kg)/Height(m)/Gender/BMI/Calories per day
Status	Passed
Result Screens	

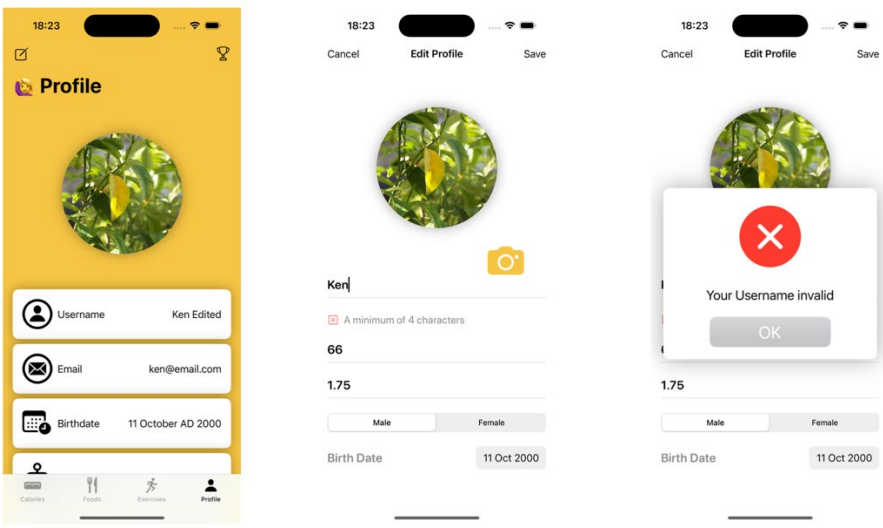
ตารางที่ 4.21 TC20 แสดงหน้า Edit Profile view – ใส่ข้อมูลครบและถูกต้อง

Test case number	TC20
Test case title	แสดงหน้า Edit Profile view - ใส่ข้อมูลครบและถูกต้อง
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. กด Edit หน้า Profile view 2. ระบบจะแสดงหน้า Edit Profile view พร้อม Filled out ข้อมูลเดิม 3. ผู้ใช้สามารถแก้ไข Profile image/ Username/ Weight/ Height/ Gender/ Birthdate 4. เมื่อกด Save ระบบจะ Update user document นั้นใน Cloud Firestore, users collection 5. ระบบจะแสดง Profile view พร้อม Updated ข้อมูลของผู้ใช้นั้นที่แก้ไขไป
Test data	<ol style="list-style-type: none"> 1. แก้ไข Username เป็น Ken Edited (เดิม Ken The Rock) 2. เปลี่ยนรูป Profile image
Expected result	ระบบจะแสดง Profile view ของผู้ใช้ที่แก้ไขข้อมูล
Actual result	ระบบแสดง Profile view ของผู้ใช้ที่แก้ไขข้อมูล
Status	Passed
Result Screens	

ตารางที่ 4.22 TC21 แสดงหน้า Edit Profile view – ใส่ข้อมูลไม่ครบ

Test case number	TC21
Test case title	แสดงหน้า Edit Profile view - ใส่ข้อมูลไม่ครบ
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. กด Edit หน้า Profile view 2. ระบบจะแสดงหน้า Edit Profile view พร้อม Filled out ข้อมูลเดิม 3. ผู้ใช้สามารถแก้ไข Profile image/ Username/ Weight/ Height/ Gender/ Birthdate 4. ลองไม่ใส่ค่า Weight/ Height 5. ระบบจะแสดง Alert view ว่าใส่ข้อมูลไม่ครบ
Test data	1. ไม่ใส่ค่า Weight/ Height
Expected result	ระบบจะแสดง Alert view แสดงว่าใส่ข้อมูลไม่ครบ
Actual result	ระบบแสดง Alert view แสดงว่าใส่ข้อมูลไม่ครบ
Status	Passed
Result Screens	

ตารางที่ 4.23 TC22 แสดงหน้า Edit Profile view - ใส่ข้อมูลไม่ถูกต้อง

Test case number	TC22
Test case title	แสดงหน้า Edit Profile view - ใส่ข้อมูลไม่ถูกต้อง
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. กด Edit หน้า Profile view 2. ระบบจะแสดงหน้า Edit Profile view พร้อม Filled out ข้อมูลเดิม 3. ผู้ใช้สามารถแก้ไข Profile image/ Username/ Weight/ Height/ Gender/ Birthdate 4. ลองใส่ค่า Username น้อยกว่า 4 Characters 5. ระบบจะแสดง Alert view ว่าใส่ข้อมูลไม่ถูกต้อง
Test data	1. ใส่ค่า Username เป็น Ken
Expected result	ระบบจะแสดง Alert view แสดงว่าใส่ข้อมูลไม่ถูกต้อง
Actual result	ระบบแสดง Alert view แสดงว่าใส่ข้อมูลไม่ถูกต้อง
Status	Passed
Result Screens	

ตารางที่ 4.24 TC23 แสดงลำดับผู้ใช้ตามแต้มสะสม (Show Leaderboard)

Test case number	TC23
Test case title	แสดงลำดับผู้ใช้ตามแต้มสะสม (Show Leaderboard)
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. กดปุ่ม Trophy หน้า Profile view 2. ระบบจะแสดงหน้า Leaderboard ที่แสดงรายการของผู้ใช้ที่มีในระบบทั้งหมด (ดึงข้อมูลจาก Cloud Firestore, users collection) โดยเรียงตาม Star Points ของแต่ละผู้ใช้จากมากไปน้อย 3. โดยมีพื้นหลังสีเขียวสำหรับผู้ใช้ปัจจุบันที่ Login อยู่ เพื่อความชัดเจนว่าผู้ใช้ปัจจุบันอยู่ในตำแหน่งใด
Test data	ไม่มี
Expected result	ระบบจะแสดงหน้า Leaderboard ที่แสดงรายการของผู้ใช้ที่มีในระบบทั้งหมด โดยเรียงตาม Star Points ของแต่ละผู้ใช้จากมากไปน้อย
Actual result	ระบบแสดงหน้า Leaderboard ที่แสดงรายการของผู้ใช้ที่มีในระบบทั้งหมด โดยเรียงตาม Star Points ของแต่ละผู้ใช้จากมากไปน้อย
Status	Passed
Result Screens	

ตารางที่ 4.25 TC24 แสดงหน้าอาหารที่ผู้ใช้แต่ละคนเพิ่มไว้

Test case number	TC24
Test case title	แสดงหน้าอาหารที่ผู้ใช้แต่ละคนเพิ่มไว้
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. ในหน้า Leaderboard 2. กดที่ Card ของผู้ใช้นั้นใด ระบบจะแสดงรายการอาหารที่ผู้ใช้คนนั้นเพิ่มเข้ามาในระบบทั้งหมด
Test data	ไม่มี
Expected result	ระบบจะแสดง รายการอาหารที่ผู้ใช้คนนั้นเพิ่มเข้ามาในระบบทั้งหมด
Actual result	ระบบแสดง รายการอาหารที่ผู้ใช้คนนั้นเพิ่มเข้ามาในระบบทั้งหมด
Status	Passed
Result Screens	

ตารางที่ 4.26 TC25 ทดสอบใส่ค่าน้ำหนักที่ไม่สามารถแปลงเป็นตัวเลขได้ ในหน้า CreateAccountView

Test case number	TC25
Test case title	ทดสอบใส่ค่าน้ำหนักที่ไม่สามารถแปลงเป็นตัวเลขได้ ในหน้า CreateAccountView
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. ไปที่หน้า CreateAccountView 2. ใส่ข้อมูลใน Fields ต่างๆให้ถูกต้อง ยกเว้น Weight filed ให้ใส่ข้อมูลผิด 3. กดปุ่ม SignUp
Test data	<ol style="list-style-type: none"> 1.เลือกรูป Profile 2. Email: test@email.com 3. Username: The Test 4. Password: 1234567A 5. Confirm Password: 1234567A 6. Weight: 77..7. 7. Height: 1.65 8. Gender: Male 9. Birthdate: Now
Expected result	ระบบจะแสดง UIAlertView แจ้งว่า “Weight or Height invalid, please check”
Actual result	ระบบจะแสดง UIAlertView แจ้งว่า “Weight or Height invalid, please check”
Status	Passed
Result Screens	

ตารางที่ 4.27 TC26 ทดสอบใส่ค่าความสูงที่ไม่สามารถแปลงเป็นตัวเลขได้ ในหน้า CreateAccountView

Test case number	TC26
Test case title	ทดสอบใส่ค่าส่วนสูงที่ไม่สามารถแปลงเป็นตัวเลขได้ ในหน้า CreateAccountView
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. ไปที่หน้า CreateAccountView 2. ใส่ข้อมูลใน Fields ต่างๆให้ถูกต้อง ยกเว้น Weight filed ให้ใส่ข้อมูลผิด 3. กดปุ่ม SignUp
Test data	<ol style="list-style-type: none"> 1. เลือกรูป Profile 2. Email: test@email.com 3. Username: The Test 4. Password: 1234567A 5. Confirm Password: 1234567A 6. Weight: 77.7 7. Height: 1.65.. 8. Gender: Male 9. Birthdate: Now
Expected result	ระบบจะแสดง AlertView แจ้งว่า “Weight or Height invalid, please check”
Actual result	ระบบจะแสดง AlertView แจ้งว่า “Weight or Height invalid, please check”
Status	Passed

ตารางที่ 4.27 (ต่อ)

Test case number	TC26
Test case title	ทดสอบใส่ค่าส่วนสูงที่ไม่สามารถแปลงเป็นตัวเลขได้ ในหน้า CreateAccountView
Result Screens	

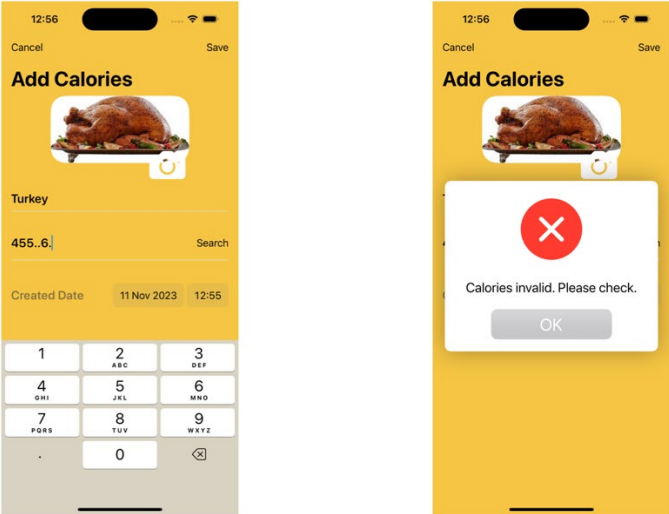
ตารางที่ 4.28 TC27 ทดสอบใส่ค่า Username เป็นภาษาไทยในหน้า CreateAccountView

Test case number	TC27
Test case title	ทดสอบใส่ค่า Username เป็นภาษาไทยในหน้า CreateAccountView
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. ไปที่หน้า CreateAccountView 2. ใส่ข้อมูลใน Fields ต่างๆให้ถูกต้อง โดย Username ให้ใส่เป็นภาษาไทย 3. กดปุ่ม SignUp 4. ไปตรวจสอบที่หน้า ProfileView
Test data	<ol style="list-style-type: none"> 1. เลือกรูป Profile 2. Email: test@email.com 3. Username: ทดสอบนะจ๊ะ 4. Password: 1234567A 5. Confirm Password: 1234567A

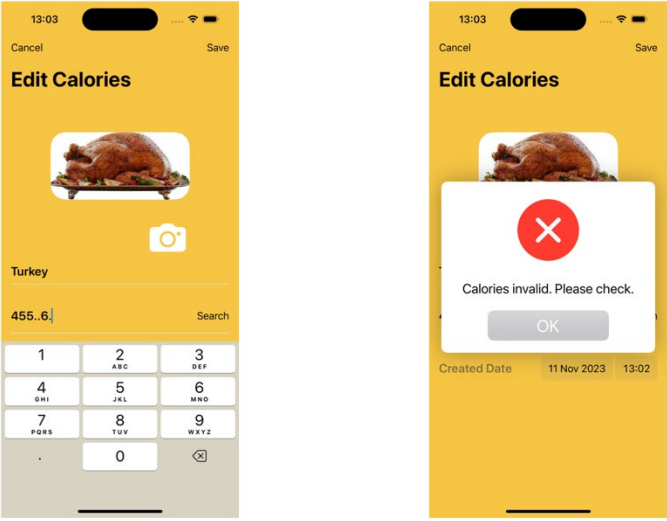
ตารางที่ 4.28 (ต่อ)

Test case number	TC27
Test case title	ทดสอบใส่ค่า Username เป็นภาษาไทยในหน้า CreateAccountView
Test data	<p>6. Weight: 77.7</p> <p>7. Height: 1.65</p> <p>8. Gender: Male</p> <p>9. Birthdate: Now</p>
Expected result	ในหน้า ProfileView, Username จะแสดงเป็น “ทดสอบนะจ๊ะ”
Actual result	ในหน้า ProfileView, Username จะแสดงเป็น “ทดสอบนะจ๊ะ”
Status	Passed
Result Screens	

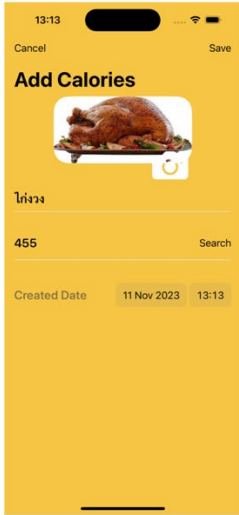

ตารางที่ 4.29 TC28 ทดสอบใส่ค่า Calories field ที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า AddCaloriesView

Test case number	TC28
Test case title	ทดสอบใส่ค่า Calories field ที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า AddCaloriesView
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. ไปที่หน้า AddCaloriesView 2. เลือกรูปภาพ,ใส่ชื่ออาหาร,CreateDate ให้ถูกต้อง 3. ใส่ ค่า Calories ไม่ถูกต้อง 4. กด Save
Test data	<ol style="list-style-type: none"> 1. เลือกรูปภาพ 2. Name: Turkey 3. Calories: 455..6. 4. CreateDate: Now
Expected result	ระบบจะแสดง UIAlertView แจ้งว่า “Calories invalid. Please check.”
Actual result	ระบบจะแสดง UIAlertView แจ้งว่า “Calories invalid. Please check.”
Status	Passed
Result Screens	

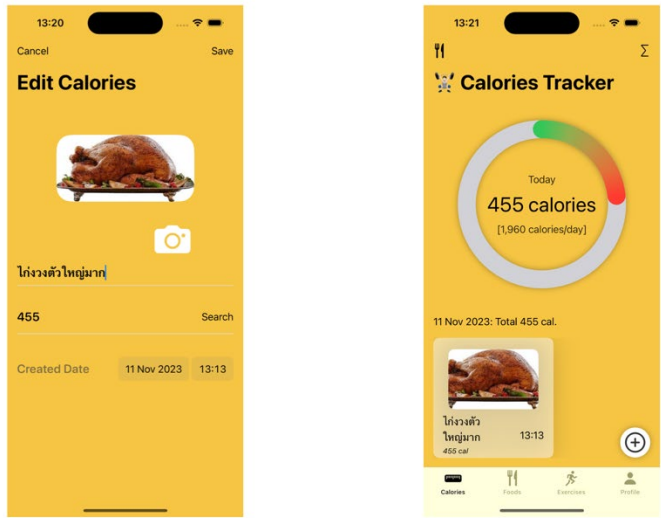
ตารางที่ 4.30 TC29 ทดสอบใส่ค่า Calories field ที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า EditCaloriesView

Test case number	TC29
Test case title	ทดสอบใส่ค่า Calories field ที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า EditCaloriesView
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. ไปที่หน้า EditCaloriesView 2. ใช้รูปภาพอาหารก่อน Edit, ใช้ชื่ออาหารก่อน Edit, ใช้ CreatedDate ก่อน Edit 3. ใส่ ค่า Calories ไม่ถูกต้อง 4. กด Save
Test data	<ol style="list-style-type: none"> 1. ใช้รูปภาพอาหารก่อน Edit 2. Name: ใช้ชื่ออาหารก่อน Edit 3. Calories: 455..6. 4. CreatedDate: ใช้ CreatedDate ก่อน Edit
Expected result	ระบบจะแสดง AlertView แจ้งว่า “Calories invalid. Please check.”
Actual result	ระบบจะแสดง AlertView แจ้งว่า “Calories invalid. Please check.”
Status	Passed
Result Screens	

ตารางที่ 4.31 TC30 ทดสอบใส่ค่า Name field เป็นภาษาไทยในหน้า AddCaloriesView

Test case number	TC30
Test case title	ทดสอบใส่ค่า Name field เป็นภาษาไทยในหน้า AddCaloriesView
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. ไปที่หน้า AddCaloriesView 2. เลือกรูปภาพ,ใส่ค่า Calories,CreateDate ให้ถูกต้อง 3. ใส่ชื่ออาหารเป็นภาษาไทย 4. กด Save
Test data	<ol style="list-style-type: none"> 1. เลือกรูปอาหาร 2. Name: ไก่จวง 3. Calories: 455 4. CreatedDate: Now
Expected result	หน้า CaloriesTrackerView ใน FoodCardView จะแสดงชื่ออาหารเป็นภาษาไทย ในที่นี้คือ “ไก่จวง”
Actual result	หน้า CaloriesTrackerView ใน FoodCardView จะแสดงชื่ออาหารเป็นภาษาไทย ในที่นี้คือ “ไก่จวง”
Status	Passed
Result Screens	 

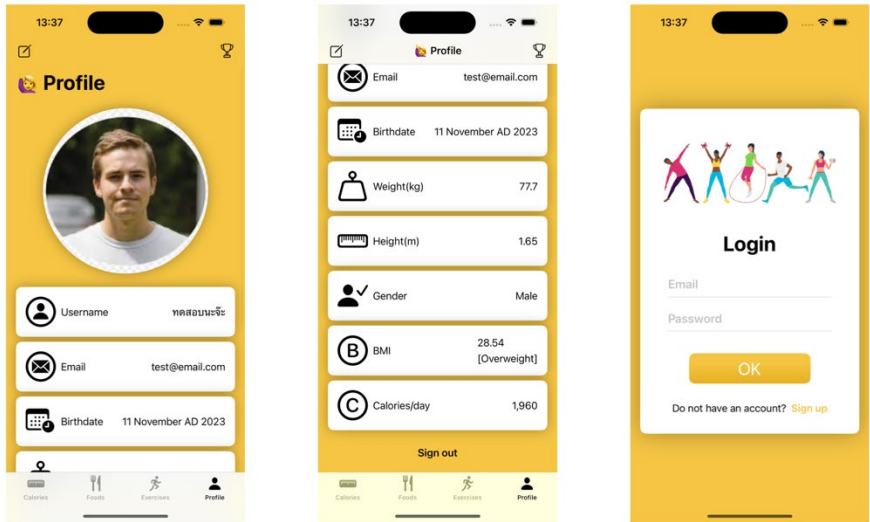
ตารางที่ 4.32 TC31 ทดสอบใส่ค่า Name field เป็นภาษาไทยในหน้า EditCaloriesView

Test case number	TC31
Test case title	ทดสอบใส่ค่า Name field เป็นภาษาไทยในหน้า EditCaloriesView
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. ไปที่หน้า EditCaloriesView 2. เลือกรูปภาพ,ใส่ค่า Calories,CreateDate ให้ถูกต้อง 3. ใส่ชื่ออาหารเป็นภาษาไทย 4. กด Save
Test data	<ol style="list-style-type: none"> 1. ใช้รูปอาหารก่อน Edit 2. Name: ไก่วงตัวใหญ่มาก 3. Calories: ใช้ค่าก่อน Edit 4. CreatedDate: ใช้ค่าก่อน Edit
Expected result	หน้า CaloriesTrackerView ใน FoodCardView จะแสดงชื่ออาหารเป็นภาษาไทย ในที่นี้คือ “ไก่วงตัวใหญ่มาก”
Actual result	หน้า CaloriesTrackerView ใน FoodCardView จะแสดงชื่ออาหารเป็นภาษาไทย ในที่นี้คือ “ไก่วงตัวใหญ่มาก”
Status	Passed
Result Screens	

ตารางที่ 4.33 TC32 ทดสอบเลือกอาหารภาษาไทยในหน้า Select your calories โดยกดปุ่ม Search ในหน้า AddCaloriesView หรือ EditCaloriesView

Test case number	TC32
Test case title	ทดสอบเลือกอาหารภาษาไทยในหน้า Select your calories โดยกดปุ่ม Search ในหน้า AddCaloriesView หรือ EditCaloriesView [แสดงตัวอย่างที่หน้า AddCaloriesView]
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	<ol style="list-style-type: none"> 1. ไปหน้า AddCaloriesView 2. กดปุ่ม Search 3. เลือกอาหารที่แสดงเป็นภาษาไทย 4. ระบบ Fill ข้อมูลในหน้า AddCaloriesView 5. กดปุ่ม Save
Test data	เลือก “ข้าวหมูแดง”
Expected result	หน้า CaloriesTrackerView ใน FoodCardView จะแสดงอาหารที่เลือก (ข้าวหมูแดง)
Actual result	หน้า CaloriesTrackerView ใน FoodCardView จะแสดงอาหารที่เลือก (ข้าวหมูแดง)
Status	Passed
Result Screens	

ตารางที่ 4.34 TC33 Sign out ออกจากระบบ

Test case number	TC33
Test case title	Sign out ออกจากระบบ
Precondition	ผู้ใช้เคยลงทะเบียนแล้วในระบบ
Test steps	1. ไปหน้า ProfileView 2. กดปุ่ม SignOut
Test data	ไม่มี
Expected result	ระบบจะ Sign out ผู้ใช้คนนั้นออกจากระบบ และจะแสดงหน้า LoginView
Actual result	ระบบจะ Sign out ผู้ใช้คนนั้นออกจากระบบ และจะแสดงหน้า LoginView
Status	Passed
Result Screens	 <p>The image shows three screenshots of a mobile application interface. The first screenshot shows the 'Profile' page with a user's profile picture, name (ทดสอบนะจ๊ะ), email (test@email.com), and birthdate (11 November AD 2023). The second screenshot shows the same profile page with a 'Sign out' button at the bottom. The third screenshot shows the 'Login' page with fields for 'Email' and 'Password', an 'OK' button, and a 'Sign up' link for users who do not have an account.</p>

4.4 แสดงผลการสำรวจความพึงพอใจของผู้ใช้

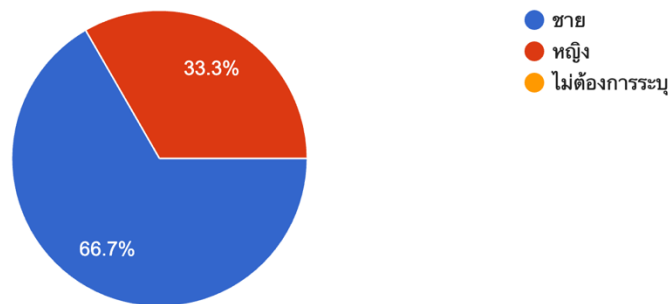
ผู้พัฒนามีการอัด VDO Demo ส่งให้ผู้ใช่ โดยมีการส่งแบบสำรวจให้ผู้ใช่แสดงความคิดเห็นใน Google Form โดยมีคำถามดังต่อไปนี้

- (1) เพศของผู้ตอบแบบสอบถาม - มีตัวเลือก (ชาย, หญิง, ไม่ระบุ)
- (2) อายุของผู้ตอบแบบสอบถาม - เป็น Field ให้เติม

- (3) ความพึงพอใจต่อ User Interface - มีตัวเลือก (น้อยที่สุด, น้อย, ปานกลาง, มาก, มากที่สุด)
 - (4) ความพึงพอใจต่อ Functions - มีตัวเลือก (น้อยที่สุด, น้อย, ปานกลาง, มาก, มากที่สุด)
 - (5) ข้อเสนอแนะ หรือความเห็นอื่นๆ - เป็น Field ให้เติม
- โดยมีผลจากแบบสอบถาม (ล่าสุดวันที่ 11/11/2566) ดังต่อไปนี้

1. เพศของผู้ตอบแบบสอบถาม: ชาย 66.7%, หญิง 33.3%

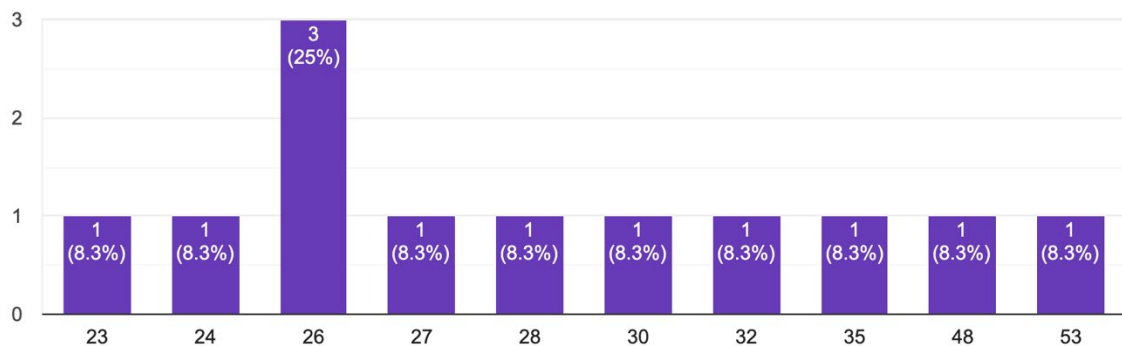
เพศของผู้ตอบแบบสอบถาม
12 responses



รูปที่ 4.20 เพศของผู้ตอบแบบสอบถาม

2. อายุของผู้ตอบแบบสอบถาม

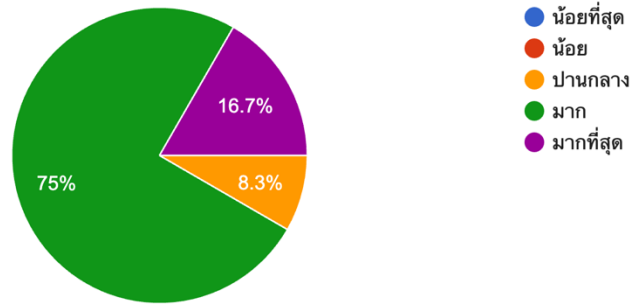
อายุของผู้ตอบแบบสอบถาม
12 responses



รูปที่ 4.21 อายุของผู้ตอบแบบสอบถาม

3. ความพึงพอใจต่อ User Interface: 16.7% มากที่สุด, 75% มาก, 8.3% ปานกลาง

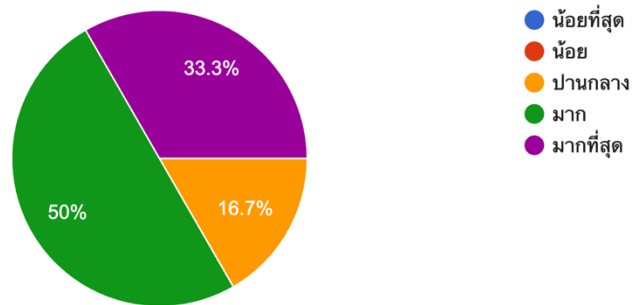
ความพึงพอใจต่อ User Interface
12 responses



รูปที่ 4.22 ความพึงพอใจต่อ User Interface

4. ความพึงพอใจต่อ Functions: 33.3% มากที่สุด, 50% มาก, 16.7% ปานกลาง

ความพึงพอใจต่อ Functions ที่มี
12 responses



รูปที่ 4.23 ความพึงพอใจต่อ Functions ที่มี

5. ข้อเสนอแนะ หรือความเห็นอื่น ๆ โดยมีการเสนอแนะ หรือความเห็นอื่นๆ สรุปได้ดังตารางที่ 4.34

ตารางที่ 4.35 ข้อเสนอแนะ หรือความเห็นอื่นๆ

ความเห็นที่ 1	ทำได้ดีมากครับ
ความเห็นที่ 2	ใน part UI นะครับ หลายๆหน้าคือดูดีเลยครับ สีขาวตัดสีส้มคือสวยเลย แต่บางหน้ากลับกลายเป็นไม่มีสีขามาตัด (เช่นหน้า add calories ส่วนตัวมองว่าถ้าทำสี input ให้ขาวตัดไว้อาจจะดูดีขึ้นครับ) ส่วนฟังก์ชัน ด้วยความที่ว่าดูอย่างเดียว ไม่ได้ไปชนเลยไม่มั่นใจ แต่โดยรวมดูครบถ้วนดีครับ
ความเห็นที่ 3	เยี่ยมมาก
ความเห็นที่ 4	เรื่องของขนาดความสมดุล การจัดวางองค์ประกอบของ ปุ่ม ตัวอักษรและสัญลักษณ์ ไอคอน สี ของแอปพลิเคชัน
ความเห็นที่ 5	<ol style="list-style-type: none"> 1. Calories Tracker รูปด้านขวาถูกตัดไป ค่ะ 2. น่าจะเพิ่ม ตั้ง set ค่า น้ำหนักที่ต้องการลด 3. แนะนำปริมาณ Calories อาหารที่ควรบริโภคในแต่ละวัน หลังจาก set ค่า น้ำหนักที่ต้องการลด
ความเห็นที่ 6	<p>ตัวแอปสวยงามดูใช้งานง่ายและเข้าใจได้ง่ายค่า ฟีเจอร์ต่างๆ เมนูที่ใส่ไว้ และลิงการออกกำลังกายดูสะดวกต่อผู้ใช้งานค่ะ น่าใช้งานเลยคะ</p> <p>แอปมีความคิดเห็นเพิ่มเติมหน่อยคะ ส่วน UXUI</p> <ol style="list-style-type: none"> 1. ในส่วนของ register ตอนกรอกข้อมูล error ที่ pop up ขึ้นมาดีแล้วค่า แต่ wording ‘strikethrough’ ไม่ค่อยพบเท่าไร ถ้าเป็น Please fill all required field ก็ทำให้ ... เป็นสีแดงอาจจะเห็นได้ง่ายขึ้นนะคะ 2. ข้อความ BMI ชิดขวาเหมือนเพื่อนๆ แถวอื่นดีไหมคะ ถ้าได้บรรทัดเดียวจะสวยกว่าเดิมนิดนึงค่า 3. หน้า Add calories ปุ่ม search แยกออกมาจากเส้น Calories (fill in line) ดีไหมคะ เมื่อ user คิดว่าหาแต่ cal จากเมนูที่เขาใส่ชื่อข้างบนอะค่า <p>มีคำถามเผื่ออีกนิดนึง ถ้าเกิดใส่แคลเกินวงแหวน จะมีแจ้งเตือนไหมคะ หรือจะแสดงผลยังไง</p> <p>ขอชื่นชมผลงานอีกครั้งค่ะ หวังว่าจะได้ลองใช้แอปจริงบน app store นะคะ)</p>

ข้อเสนอแนะ หรือความเห็นอื่นๆ

6 responses

ทำได้ดีมากครับ

ใน part UI นะครับ หลายๆหน้าคือดูดีเลยครับ สีขาวตัดสีส้มคือสวยเลย แต่บางหน้ากลับกลายเป็นไม่มีสีขามาตัด (เช่น หน้า add calories ส่วนตัวมองว่าถ้าทำสี input ให้ขาวตัดไว้อาจจะดูดีขึ้นครับ) ส่วนฟังก์ชัน ด้วยความที่ว่าดูอย่างเดียว ไม่ได้ไปชนเลยไม่มั่นใจ แต่โดยรวมดูครบถ้วนดีครับ

เยี่ยมมาก

เรื่องของขนาดความสมดุล การจัดวางองค์ประกอบของ ปุ่ม ตัวอักษรและสัญลักษณ์ ไอคอน สี ของแอปพลิเคชัน

1. Calories Tracker รูปด้านขวาถูกตัดไป ค่ะ
2. น่าจะเพิ่ม ตั้ง set ค่า น้ำหนักที่ต้องการลด
3. แนะนำปริมาณ Calories อาหารที่ควรบริโภคในแต่ละวัน หลังจาก set ตำน้าหนักที่ต้องการลด

ตัวแอปสวยงามดูใช้งานง่ายและเข้าใจได้ง่ายค่า
ฟีเจอร์ต่างๆ เมนูที่ใส่ไว้ และลิงการออกกำลังกายดูสะดวกต่อผู้ใช้งานค่ะ นำ ใช้งานเลยคะ

แอปมีความคิดเห็นเพิ่มเติมหน่อยคะ ส่วน UXUI

รูปที่ 4.24 ตัวอย่างข้อเสนอแนะ หรือความเห็นอื่นๆ

ข้อเสนอแนะ หรือความเห็นอื่นๆ

6 responses

1. Calories Tracker รูปด้านขวาถูกตัดไป ค่ะ
2. น่าจะเพิ่ม ตั้ง set ค่า น้ำหนักที่ต้องการลด
3. แนะนำปริมาณ Calories อาหารที่ควรบริโภคในแต่ละวัน หลังจาก set ตำน้าหนักที่ต้องการลด

ตัวแอปสวยงามดูใช้งานง่ายและเข้าใจได้ง่ายค่า
ฟีเจอร์ต่างๆ เมนูที่ใส่ไว้ และลิงการออกกำลังกายดูสะดวกต่อผู้ใช้งานค่ะ นำ ใช้งานเลยคะ

แอปมีความคิดเห็นเพิ่มเติมหน่อยคะ ส่วน UXUI

1. ในส่วนของ register ตอนกรอกข้อมูล error ที่ pop up ขึ้นมาดีแล้วค่า แต่ wording 'strikethrough' ไม่ค่อยพบเท่าไร ถ้าเป็น Please fill all required field กับทำให้ ... เป็นสีแดงอาจจะเห็นได้ง่ายขึ้นนะคะ
2. ข้อความ BMI ชิดขวาเหมือนเพื่อนๆ แลอื่นดีไหมคะ ถ้าได้บรรทัดเดียวจะสวยกว่าเดิมนิดนึงค่า
3. หน้า Add calories ปุ่ม search แยกออกมาจากเส้น Calories (fill in line) ดีไหมคะ เมื่อ user คิดว่าหาแต่ cal จากเมนูที่เขาใส่ชื่อข้างบนอะค่า

มีคำถามเมื่ออีกนิดนึง ถ้าเกิดใส่แคลเกินวงแหวน จะมีแจ้งเตือนไหมคะ หรือจะแสดงผลยังง

ขอชื่นชมผลงานอีกครั้งคะ หวังว่าจะได้ลอง ใช้แอปจริงบน app store นะคะ :)

รูปที่ 4.25 ตัวอย่างข้อเสนอแนะ หรือความเห็นอื่น ๆ (ต่อ)

บทที่ 5

สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ

ในบทนี้จะทำการสรุปผลการพัฒนาแอปพลิเคชันติดตามแคลอรีการบริโภคอาหารบนอุปกรณ์พกพา รวมทั้งอภิปรายและข้อเสนอแนะ

5.1 สรุปผลการวิจัยและอภิปราย

ผู้เขียนได้ทำการพัฒนาแอปพลิเคชันติดตามแคลอรีการบริโภคอาหารบนอุปกรณ์พกพา โดยใช้ SwiftUI framework ต่อกับฐานข้อมูล Firebase (ใช้ในส่วนของ Authentication ด้วย Email/Password, Cloud Firestore, และ Cloud Storage) ซึ่งแอปพลิเคชันนี้รองรับบนอุปกรณ์ iPhone iOS 16.4 ขึ้นไป ซึ่งหลังจากพัฒนาแล้วเสร็จ ก็ตรวจสอบกลับได้ว่าตรงกับวัตถุประสงค์ทั้งหมดในข้อ 1.3 และมี Function ครบถ้วนตามความต้องการของระบบในข้อ 3.1 ซึ่งแอปพลิเคชันนี้มีทั้งหมด 12 Use cases ตามหัวข้อ 3.2 (มีแสดง Use case description ตามตารางที่ 3.2 ถึง ตารางที่ 3.13) มีการแสดง Flow ของแอปในหัวข้อ 3.3 Activity Diagram รวมทั้ง Class Diagram หัวข้อ 3.4 และ ER Diagram หัวข้อ 3.5 และมีการแสดง UX/UI ที่มีการออกแบบไว้ด้วยเครื่องมือ Figma ในหัวข้อ 3.6

สำหรับแอปที่พัฒนาสำเร็จ มีการแสดง UI ที่พัฒนาขึ้นในหัวข้อ 4.1 ซึ่งมีการอธิบายว่ามีส่วนประกอบอะไรบ้างในแต่ละหน้าของ UI รวมทั้งในหัวข้อ 4.2 มีการแสดงฐานข้อมูล Firebase ที่พัฒนาขึ้นตามที่มีการออกแบบไว้ (Authentication ด้วย Email/Password, เก็บรูปภาพใน Cloud Storage, และเก็บฐานข้อมูลแบบ Document เป็น Collection ใน Cloud Firestore) ส่วนในหัวข้อ 4.3 มีการแสดง Test cases ทั้งหมด 24 Test cases โดยมีการเทียบว่าตรงกับ Use case ใดในตารางที่ 4.1 และผลการทดสอบทั้งหมดตามตารางที่ 4.2 ถึง ตารางที่ 4.25 และในหัวข้อ 4.4 มีการสรุปผลการสำรวจกับผู้ใช้ โดยการบันทึก VDO Demo ให้ผู้ใช้ดู และให้ผู้ใช้แสดงความเห็นในแบบสำรวจที่ทำขึ้นด้วย Google Form โดยความพึงพอใจต่อ User Interface - 20% พอใจมากที่สุด 80% พอใจมาก 10% พอใจปานกลาง และความพึงพอใจต่อ Function ที่มี - 40% พอใจมากที่สุด 40% พอใจมาก 20% พอใจปานกลาง

สรุปแอปพลิเคชันที่พัฒนาขึ้นเป็นไปตามความต้องการที่กำหนดไว้ ทั้งด้าน Function การใช้งาน และ User Interface แต่ควรที่จะต้องเพิ่มเติม Function ต่างๆมากขึ้นด้วย รวมทั้งเพิ่มเติมอื่นๆ ดังจะแสดงในหัวข้อต่อไป

5.2 ข้อเสนอแนะ

หลังจากพัฒนาแอปพลิเคชันนี้แล้วเสร็จ ปรากฏว่ามีข้อเสนอแนะเพื่อปรับปรุง และเป็นข้อควรแนะนำให้เห็นว่าจะมีการพัฒนาแอปพลิเคชันนี้ให้ดียิ่งขึ้นต่อไปอย่างไร

5.2.1 พัฒนาเพิ่มเติมในส่วน User Interface ให้รองรับ iPad ด้วย

5.2.2 การทดสอบแอปพลิเคชันควรส่งเป็นตัวอย่างให้ผู้ใช้งานได้ใช้จริง ซึ่งทำได้โดยใช้ Test Flight ซึ่งต้องมีการสมัคร Apple Developer Account และทำการส่งแอปให้ Apple Review จากนั้นสามารถส่งเป็น Link ให้ผู้ใช้งานได้ทดสอบเป็นเวลา 90 วัน

5.2.3 พัฒนาแอปพลิเคชันรองรับ Android platform โดยอาจใช้การพัฒนาแบบ Native (Kotlin) หรือ Cross platform โดยใช้ Flutter หรือ React-Native ซึ่งฐานข้อมูลสามารถใช้ฐานข้อมูล Firebase เดียวกันกับที่ได้พัฒนาขึ้นมาได้เลย

5.2.4 เพิ่ม Function ต่างๆเข้าไปในแอปพลิเคชันให้มากขึ้น เช่น สามารถตั้งเป้าหมายการบริโภคแคลอรีโดยผู้ใช้, สร้างปุ่ม Like ที่เมนูอาหารที่แต่ละผู้ใช้เพิ่มเข้ามา, เพิ่ม Function ลืม Password, เพิ่ม Function การแสดง Badge เมื่อมีข้อมูลใหม่ในระบบ(ในหน้า Food Menu view และ Exercise Menu view), เพิ่ม Function การแลกของรางวัล เป็นต้น

5.2.5 เนื่องด้วยผู้พัฒนาต้องการให้แอปพลิเคชันนี้ใช้งานได้ฟรี แต่อย่างไรก็ตามก็มีค่าใช้จ่ายที่ต้องจ่าย เช่น ค่าเช่า Firebase ดังนั้นจึงต้องมีการหารายได้ให้แอปพลิเคชัน โดยผู้พัฒนาคิดว่าควรจะไปติดต่อร่วมงานกับร้านขายอาหารสุขภาพ ให้เขานำเมนูของเขามาแสดงในรายการอาหารที่แนะนำโดยระบบ (หน้า Food Menu view) ซึ่งอาจจะมีช่องทางให้ผู้ใช้งานได้สั่งอาหารนั้นๆได้ รวมทั้งควรจะไปติดต่อผู้แนะนำการออกกำลังกายต่างๆให้มาทำ Content การออกกำลังกายต่างๆ และอาจมีช่องทางให้ผู้ติดต่อเพื่อเข้า Courses ต่างๆที่นำเสนอ และอาจหาช่องทางในการทำ In-App Ads (เช่น AdMob) เป็นต้น

5.2.6 ควรจะมีการทดสอบแอปพลิเคชันเพิ่มเติมได้แก่ Unit test, UI Test, และ Performance Test ซึ่ง Apple มีจัดไว้ให้ใน XCTest framework (ซึ่งผู้พัฒนากำลังพยายามศึกษาเพิ่มเติมอยู่ แต่ไม่สามารถทำในงานวิจัยนี้ได้ทัน)

บรรณานุกรม

บรรณานุกรม

- [1] “*Best Mobile App Development Tools, Frameworks & Platforms,*” Techliance Blog [Online]. <https://blog.techliance.com/mobile-app-development-tools/>. (Accessed: Oct 27, 2023).
- [2] “*Why Your Business Should Develop An iOS App,*” Quantilus innovation [Online]. <https://quantilus.com/article/why-your-business-should-develop-an-ios-app/>. (Accessed: Oct 27, 2023).
- [3] J. Schmitt, “*Native vs cross-platform mobile app development,*” Circleci Blog [Online]. <https://circleci.com/blog/native-vs-cross-platform-mobile-dev/>. (Accessed: Oct 27, 2023).
- [4] M. Pratap, “*SwiftUI vs UIKit: Which one to choose for your iOS app,*” Supersourcing [Online]. <https://supersourcing.com/blog/?s=SwiftUI+vs+UIKit%3A+Which+one+to+choose+for+your+iOS+app>. (Accessed: Oct 27, 2023).
- [5] B. Guriev, “*Firebase Pros and Cons: When you should and shouldn’t use Firebase,*” OSDB [Online]. <https://osdb.io/firebase-pros-and-cons-when-you-should-and-shouldnt-use-firebase-osdb/>. (Accessed: Oct 27, 2023).

