



การพัฒนาแอปพลิเคชันติดตามแคลอรีการบริโภคอาหารบนอุปกรณ์พกพา
The Development of Calories Tracker Application on Mobile Device

วีรวุฒิ ชัยยะสมบูรณ์ และ วราพร จิระพันธุ์ทอง*

Weerawut Chaiyasomboon and Waraporn Jirapanthong*

วิทยาลัยครีเอทีฟดีไซน์ แอนด์ เอ็นเตอร์เทนเมนต์เทคโนโลยี, มหาวิทยาลัยธุรกิจบัณฑิตย์

College of Creative Design and Entertainment Technology, Dhurakij Pundit University

Received: June 22, 2024; Revised: June 24, 2023; Accepted: June 28, 2024; Published: June 30, 2024

ABSTRACT – According to the study on health of Thai people [1], it is found that one third of Thai people are overweight and one tenth of Thais are considered obese. Moreover, the research [1] claimed that during 1991-2009, the rate of Thais aged 15 years and over was overweight double (from 17.2 percent to 34.7 percent of all Thais) and considered obese increased almost three times (from 3.2 percent to 1 percent). The problem of overnutrition, especially in early childhood and school-aged children, is an important health issue that all parties must be aware of and cooperate to solve. This research aims to initiate and develop a tool that can support Thais to take care of their health sustainably and effectively. In this research, the researchers presented the development of the application named “HealthMe” The objectives are to monitor the nutrition and calories which a user consumes at each meal and assist the user to control the amount of consuming calories each day. Consumption data is recorded and analyzed along with other health data. The data can be then analyzed along with exercise data to analyze in-depth and broad health information. In this article, the design and development of the application are presented. It can also provide recommendation about healthy food and exercise methods that are appropriate to the user's health information. There is also a community of users using this application to take care of their health and food consumption. They can communicate and share information about nutrients intake to guide one's own diet. The application is developed using the SwiftUI framework connected to the Firebase database (using Authentication, Cloud Firestore, and Cloud Storage), which is only supported on iPhones running on iOS 16.4 or later. The authors propose the development of App on iOS platform. and Firebase. Additionally, we collected and analyzed functional and quality requirements from 78 Thai people aged 18-65 years. Our work include the design of user experience and user interface (UX/UI). The experimental results are claimed that the accuracy in term of functions was 100 percent and the user satisfaction was 93 percent.

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

บทคัดย่อ -- จากการศึกษาเบื้องต้นของสุขภาพประชาชนในประเทศไทย [1] พบว่าจำนวนหนึ่งในสามของคนไทยมีภาวะน้ำหนักเกินและจำนวนหนึ่งในสิบของคนไทยจัดอยู่ในภาวะอ้วน จากงานวิจัย [1] พบว่า ช่วงปี 2534-2552 คนไทยที่มีอายุตั้งแต่ 15 ปีขึ้นไป มีอัตราของภาวะน้ำหนักเกินเพิ่มขึ้น 2 เท่า (จากร้อยละ 17.2 เป็นร้อยละ 34.7ของคนไทยทั้งหมด) และภาวะอ้วนเพิ่มขึ้นเกือบ 3 เท่า (จากร้อยละ 3.2 เป็นร้อยละ 1) ทั้งนี้ปัญหาโภชนาการเกิน โดยเฉพาะในเด็กปฐมวัยและเด็กวัยเรียน เป็นวาระทางสุขภาพสำคัญที่ทุกฝ่าย ต้องตระหนักและร่วมมือแก้ไข งานวิจัยนี้มีวัตถุประสงค์ริเริ่มจัดทำและพัฒนาเครื่องมือหรืออุปกรณ์ที่สามารถสนับสนุนให้คนไทยสามารถดูแลสุขภาพได้อย่างยั่งยืนและมีประสิทธิภาพ ในงานวิจัยนี้นักวิจัยได้

*Corresponding Author: waraporn.jir@dpu.ac.th

นำเสนอการออกแบบและการพัฒนาแอปพลิเคชัน ชื่อว่า “HealthMe” โดยมีวัตถุประสงค์เพื่อสนับสนุนให้ผู้ใช้สามารถตรวจสอบรายการอาหารที่บริโภคเข้าไปในแต่ละมื้อ ว่ามีปริมาณแคลอรี ที่มากเกินไปจนเกินความจำเป็นจากปริมาณที่ผู้ใช้สามารถบริโภคในแต่ละวันเข้าไป ข้อมูลของการบริโภคจะถูกบันทึกและวิเคราะห์ที่ประกอบด้วยข้อมูลด้านสุขภาพอื่น ๆ อีกต่อไปได้ และสามารถนำข้อมูลจากแอปพลิเคชันไปวิเคราะห์ควบคู่กับข้อมูลการออกกำลังกายเพื่อวิเคราะห์ข้อมูลสุขภาพเชิงลึกและกว้าง และเป็นประโยชน์ต่อการดูแลสุขภาพของผู้ใช้ต่อไปได้ ในบทความนี้ได้นำเสนอการออกแบบและพัฒนา แอปพลิเคชันที่ตรวจสอบปริมาณสารอาหารและแคลอรีที่ผู้ใช้ได้บริโภคในแต่ละวัน พร้อมทั้งแนะนำอาหารสุขภาพ และวิธีการออกกำลังกายที่เหมาะสมกับข้อมูลสุขภาพของผู้ใช้ นอกจากนี้ยังมีชุมชนของผู้ใช้งานแอปพลิเคชันนี้เพื่อดูแลสุขภาพการบริโภคอาหาร สามารถให้ข้อมูลเชิงวัดผลเปรียบเทียบของการบริโภคสารอาหารเพื่อเป็นแนวทางของตนเองได้ 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 นักวิจัยได้รวบรวมและวิเคราะห์ความต้องการทั้งในด้านฟังก์ชันและคุณภาพจากกลุ่มตัวอย่างจำนวน 78 คน ที่มีอายุในช่วง 18-65 ปี รวมทั้งมีการออกแบบส่วนติดต่อประสานและประสบการณ์ของผู้ใช้ ผลการทดสอบโดยกลุ่มตัวอย่างได้ พบว่าความถูกต้องของฟังก์ชันเป็นร้อยละ 100 และความพึงพอใจในการใช้งานของผู้ใช้คิดเป็นร้อยละ 93

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

1. บทนำ

1.1 ที่มาและความสำคัญ

จากงานวิจัย [2] พบว่า ความอ้วนมีแนวโน้มแปรผันตามระดับการพัฒนาประเทศและคุณภาพชีวิต ความหนาแน่นของประชากรที่อยู่ในภาวะน้ำหนักเกินและอ้วนของคนไทยกำลังเพิ่มขึ้นอย่างรวดเร็ว โดยที่หญิงไทยมีแนวโน้มจะอ้วนมากกว่าชายไทยในหลายตัวชี้วัด หากเปรียบเทียบในระดับภูมิภาค ปัจจุบันคนไทยอ้วนมากเป็นอันดับ 2 ใน 10 ประเทศอาเซียน รองจากประเทศมาเลเซีย ซึ่งหากแยกดูตามเพศ ชายไทยอ้วนเป็นอันดับที่ 4 ขณะที่หญิงไทย อ้วนเป็นอันดับที่ 2 งานวิจัยนี้มีวัตถุประสงค์ริเริ่มจัดทำและพัฒนาเครื่องมือหรืออุปกรณ์ที่สามารถสนับสนุนให้คนไทยสามารถดูแลสุขภาพได้อย่างยั่งยืนและมีประสิทธิภาพ ในงานวิจัยนี้นักวิจัยได้นำเสนอการออกแบบและการพัฒนาแอปพลิเคชัน ชื่อว่า “HealthMe” เพื่อที่ผู้ใช้จะสามารถตรวจสอบรายการอาหารที่บริโภคเข้าไปในแต่ละมื้อ ว่ามีปริมาณแคลอรี ที่มากเกินไปจนเกินความจำเป็นจากปริมาณที่ผู้ใช้สามารถบริโภคในแต่ละวันเข้าไปหรือไม่ และให้ข้อมูลสุขภาพด้านการออกกำลังกาย การติดต่อระหว่างชุมชนเพื่อเรียนรู้ร่วมกันและแนวทางปฏิบัติของผู้ใช้ จากการศึกษาเบื้องต้นพบว่า ในปัจจุบัน

- มีเครื่องมือไม่เพียงพอใช้ในการติดตามการบริโภคอาหารเข้าไปในแต่ละมื้อว่ามีแคลอรี มากเกินความจำเป็นของผู้ใช้ในแต่ละวันหรือไม่และอย่างไร
- ผู้ใช้ต้องการเครื่องมือที่สะดวกต่อการใช้งาน ผู้ใช้จะยังสามารถใช้งานได้ตลอดเวลา
- แอปพลิเคชันที่มีอยู่มีความซับซ้อนและรองรับภาษาอังกฤษเป็นส่วนใหญ่
- แอปพลิเคชันที่มีอยู่มีค่าใช้จ่ายในการใช้หรือติดตั้ง
- แอปพลิเคชันที่ใช้งานได้แบบไม่เสียค่าใช้จ่าย ขาดฟังก์ชันตามความต้องการผู้ใช้ ส่วนใหญ่จำเป็นต้องติดตั้งเป็นแบบ In-app purchase
- ไม่มีแอปพลิเคชันที่รวบรวมข้อมูลอาหารในท้องถิ่นของคนไทยที่ติดต่อสุขภาพ และวิธีการออกกำลังกายที่เหมาะสม
- ไม่มีแอปพลิเคชันที่กลุ่มผู้ใช้ที่มีความสนใจด้านการรักษาสุขภาพสามารถเข้าไปแชร์ข้อมูลอาหารที่ตนบริโภคเข้าไปในแต่ละมื้อ

1.2 วัตถุประสงค์

- ศึกษาความต้องการ วิเคราะห์และพัฒนาเครื่องมือในการติดตามการบริโภคอาหารเข้าไปในแต่ละมื้อว่ามีแคลอรี

มากขึ้นความจำเป็นของผู้ใช้ในแต่ละวันหรือไม่และอย่างไร โดยต้องสามารถใช้งานง่ายได้กับผู้ใช้ทุกวัย และเป็นแอปพลิเคชันที่ใช้งานได้แบบไม่มีค่าใช้จ่าย

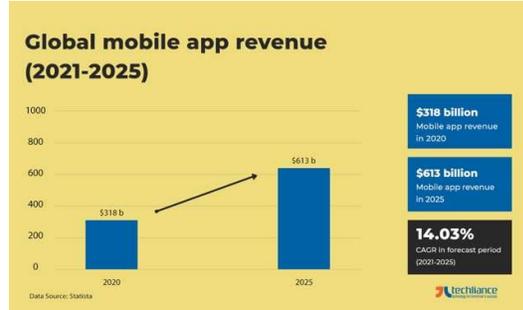
- พัฒนาแอปพลิเคชันที่สามารถแนะนำรายการอาหารสุขภาพ และแนะนำวิธีการออกกำลังกายแบบต่างๆให้ผู้ใช้ได้
- พัฒนา แอปพลิเคชันที่สามารถเก็บข้อมูลผู้ใช้ในฐานข้อมูลกลาง ซึ่งผู้ใช้สามารถเข้าถึงข้อมูลของตนเองได้โดยข้อมูลไม่มีการสูญหาย ถ้ามีการเปลี่ยนอุปกรณ์มือถือหรือเข้าใช้ในอุปกรณ์เครื่อง และทำให้ผู้ใช้สามารถดูข้อมูลอาหารของผู้ใช้ท่านอื่นที่บริโภคไปได้

2. แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง

การพัฒนาแอปพลิเคชันในงานวิจัยนี้ จะพัฒนาสำหรับอุปกรณ์มือถือ iPhone (iOS platform) เป็นหลัก (โดยใช้ SwiftUI framework ในการพัฒนา) โดยจะมีการต่อกับฐานข้อมูล Firebase ทั้งนี้จากการศึกษาแนวคิดและทฤษฎีที่เกี่ยวข้อง ผู้วิจัยพบว่า Apple platform นั้นประกอบไปด้วย iPadOS และ macOS ด้วย งานวิจัยนี้ออกแบบและพัฒนาแอปพลิเคชันบนระบบปฏิบัติการ iOS เพื่อรองรับการใช้งานแอปพลิเคชันทั้ง บนอุปกรณ์อื่นได้แก่ iPadOS และ macOS เนื่องจาก SwiftUI framework ที่ใช้ในการพัฒนานั้นสามารถพัฒนาต่อยอดเป็น Apple Cross Platform Device ได้ ทั้งนี้แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้องมีดังนี้

2.1 การพัฒนาแอปพลิเคชันบนอุปกรณ์มือถือในปัจจุบัน

ในปัจจุบันมี Mobile Operating System ในตลาดอยู่ 2 ค่าย คือ Android Platform และ iOS Platform และจากรูปที่ 1 จะเห็นว่ารายรับของ Mobile App จะมีมูลค่า \$613 billion ในปี ค.ศ. 2025 นี้จึงเป็นอีกเหตุผลที่ผู้วิจัยเลือกที่จะพัฒนาแอปพลิเคชันนี้ในอุปกรณ์พกพาหรือมือถือ



รูปที่ 1. แสดง Global mobile app revenue (2021-2025) [3].

สำหรับเครื่องมือและภาษาที่นิยมใช้ในปัจจุบันมีดังต่อไปนี้

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

2.2 iOS platform

จากงานศึกษา [4] ได้นำเสนอจุดเด่นของ iOS platform โดยสรุปได้ดังนี้

- ผลตอบแทนการลงทุน (ROI – Return On Investment) – Apple เป็นผู้นำในเรื่องลูกค้าที่ยอมจ่ายซื้อ App มากกว่า Android platform
- การพัฒนา App มีประสิทธิภาพมากกว่า – ผู้พัฒนาสำหรับ Android platform ต้องคำนึงถึงอุปกรณ์ Android ที่มีหลากหลาย แต่สำหรับ Apple platform ผู้พัฒนาเพียงพัฒนาสำหรับ iPhone และ iPad เท่านั้น นอกจากนี้ตั้งแต่ปี ค.ศ. 2020, Apple ผลิต Mac ที่เป็น Apple Silicon chip ซึ่งทำให้ App native iOS สามารถทำงานบน Mac ได้ด้วย ซึ่งทำให้การพัฒนา App ใน Apple platform มีต้นทุนต่ำและใช้เวลาน้อยกว่าถ้าเทียบกับ Android platform ที่มีความหลากหลายของอุปกรณ์
- ความปลอดภัย - มีการพยากรณ์ว่าภายในปี ค.ศ. 2021 ภัยทางไซเบอร์ (Cybercrime) จะมีมูลค่าถึง 6 พันล้านดอลลาร์ต่อปี ซึ่งทำให้ผู้ใช้ระวังและกังวลเรื่องความปลอดภัยใน App ที่เรียกใช้ ข้อมูลต่างๆเกี่ยวกับการจ่ายเงินและข้อมูลส่วนบุคคล สำหรับเรื่องเหล่านี้ Apple ขึ้นชื่อมาแต่เนิ่นนาน เรื่องความปลอดภัยในอุปกรณ์ต่างๆของคน รวมทั้ง

ผู้พัฒนา App ใน Apple platform ต้องมีการส่ง App ที่ตนพัฒนาเพื่อ Review ก่อน ส่งถึงมือผู้ใช้ ทำให้ผู้ใช้เชื่อมั่นใน App ต่างๆของ Apple platform มากกว่า

- การใช้งานและประสบการณ์ผู้ใช้ – Apple มีข้อกำหนดด้านรูปแบบของ App มากกว่า Android ซึ่งทำให้รูปแบบส่วนต่อกับผู้ใช้มีคุณภาพมากกว่า ซึ่ง iOS platform ทำให้ผู้ใช้งานมั่นใจได้ว่า App จะใช้งานได้ง่ายและสร้างประสบการณ์ให้ผู้ใช้ได้เป็นอย่างดี ซึ่งทำให้บริษัทหลายแห่งทำ App ใน Apple platform เพื่อเป็นหน้าเป็นตาให้กับบริษัท
- ชื่อเสียงของ Apple – Apple เป็นบริษัทที่มีชื่อเสียงระดับโลก ทำให้ Apps ใน App Store มีชื่อเสียงอยู่ในระดับสูงเมื่อเทียบกับ Android apps, iPhone รุ่นแรกทำให้เกิดอุตสาหกรรมใหม่สำหรับ Smartphone รวมทั้งตอนที่ Apple นำเสนอ iPad ทำให้มีการผลิต Tablets ต่างๆ แข่งขัน จึงกล่าวได้ว่า Apple มีชื่อเสียงในด้านนวัตกรรม ซึ่งผู้ใช้งานทั้งหมดทราบดีสำหรับสิ่งเหล่านี้

2.3 Native (SwiftUI) เปรียบเทียบ Cross platform

จากการศึกษาเปรียบเทียบระหว่างการพัฒนาแอปพลิเคชันแบบ Native (SwiftUI) และ cross platform [4][5][6] สามารถสรุปได้ดังนี้

- มีสมรรถนะดีกว่า - เนื่องด้วยเราพัฒนาด้วย Native language (Swift) และ Native framework (SwiftUI) ทำให้ App มีสมรรถนะดีกว่า เร็วกว่า และตอบสนองต่อผู้ใช้ได้ดีกว่า การพัฒนาด้วย Cross platform ที่ต้องแปลง codes เป็น Native
- มีความปลอดภัยดีกว่า - เพราะ Native app เข้าถึง Built-in security features
- UX มีคุณภาพ – Native app ให้ประสบการณ์กับผู้ใช้ได้ดีกว่า เพราะ UI/UX ทำตาม Design guidelines จาก Native OS platform ซึ่งทำให้ App มีความเป็นธรรมชาติในการใช้งานได้มากกว่า
- สามารถเข้าถึงทุกส่วนของ Device – Native app สามารถเข้าถึง Hardware/Services ทุกส่วนของ Device ทำให้ App มี Features ทุกอย่างที่มี ซึ่งเพิ่มประสบการณ์ของผู้ใช้
- มีข้อผิดพลาดที่น้อย – Native app สร้างด้วย Native SDKs ทำให้การพัฒนาที่มีข้อผิดพลาดที่น้อยกว่า ผู้พัฒนาจะได้รับ SDKs ที่มีการแก้ไขและ/หรือเพิ่มคุณสมบัติได้ทันที

2.4 SwiftUI เทียบกับ UIKit

ก่อนหน้านี้ SwiftUI framework, การพัฒนาแอปพลิเคชันบน iOS Platform ใช้ UIKit framework [6]ในการพัฒนา และต่อไปนี่คือเหตุผลที่ควรเลือกใช้ SwiftUI framework over UIKit framework ผู้พัฒนาแอปด้วย SwiftUI สามารถพัฒนาแอปพลิเคชันได้อย่างง่ายและรวดเร็วมากกว่า UIKit เนื่องด้วย SwiftUI เป็นแบบ Declarative syntax ผู้พัฒนาสามารถกำหนด UI ได้เลย โดยไม่ต้องเขียน Code แบบเป็นลำดับขั้นเหมือน UIKit

- SwiftUI เป็น Apple Cross Platform Devices ผู้พัฒนาสามารถพัฒนาแอปพลิเคชันได้ทุก Apple Devices ทำให้ลดเวลาและค่าใช้จ่ายในการพัฒนา
- 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 [7]

Firebase [7] เป็น NoSQL Database (Firestore database) ซึ่งมีคุณสมบัติที่สามารถเก็บข้อมูลได้จำนวนมากและยังคงมีสมรรถนะที่ดี รวมทั้งมีคุณสมบัติยืดหยุ่นและสามารถขยายฐานข้อมูลได้ (Flexible and Scalable) นักวิจัยได้ทดสอบใช้งานพบว่า แอปพลิเคชันที่ใช้ร่วมกับ Firebase สามารถพัฒนาได้อย่างรวดเร็ว – Firebase มีคุณสมบัติของ Backend อยู่ในตัว ทำให้สามารถเชื่อมต่อ Frontend ไปที่ Firebase ได้โดยตรง ทำให้ลดเวลาในการพัฒนาอย่างมาก รวมทั้งไม่ต้องการบำรุงรักษาทางด้าน Backend

Firebase สามารถเพิ่ม Google Analytics ซึ่งสามารถวิเคราะห์และ เก็บข้อมูลของผู้ใช้ ทำให้สามารถสร้าง App ที่ตอบสนองความต้องการของผู้ใช้ได้ดีขึ้น Firebase ทำให้มีการบำรุงรักษาทางด้าน Frontend ไม่ต้องการพัฒนาและบำรุงรักษาทางด้าน Backend ทำให้ App สามารถออกสู่ตลาดได้เร็ว นอกจากจะลดบุคลากรด้าน Backend แล้วนั้น ยังลดข้อผิดพลาดต่างๆ (Bugs) ในการพัฒนาด้วย

นอกจากนี้ยังมีชุมชนผู้ใช้งานขนาดใหญ่ และมี Technical documents ที่เพียงพอ และง่ายต่อการบูรณาการ (Ease of integration) – Firebase สามารถสร้าง App ได้โดยง่าย มีการ configuration/setup ที่ง่าย และ Firebase ยังมีคุณสมบัติของ Push notifications หรือ Firebase cloud messaging ทำให้ผู้พัฒนา

สามารถทำ Push notification ส่งให้ผู้ใช้ทราบถึงสิทธิประโยชน์/
Promotions ต่างๆ ที่มีในเวลานั้นๆ

2.6 ทฤษฎี UX/UI

การออกแบบส่วนต่อประสานกับผู้ใช้และประสบการณ์ของผู้ใช้ (UX/UI) คือการออกแบบภาพรวมของผู้ใช้ต่อระบบที่ออกแบบว่าผู้ใช้มีความรู้สึกรู้เข้าใจต่อระบบที่ออกแบบไว้ ซึ่งก็คือ UX design โดย Interface ของระบบว่ามีหน้าตาอย่างไร และมี Function ใดๆ ในนั้นก็คือ UI Design โดยการจะได้มาซึ่ง UX/UI เราต้องมีการวิจัยเกี่ยวกับผู้ใช้ นำมาซึ่งการกำหนดคุณสมบัติของผู้ใช้ (Persona), ความคาดหวังของผู้ใช้ (User Expectation), โครงสร้างของผลิตภัณฑ์ (Product structure) จนนำมาถึงการทำ Prototyping/Wireframe

3. ระเบียบวิธีวิจัย

จากกลุ่มตัวอย่างของงานวิจัยจำนวน 78 คน ที่มีอายุในช่วง 18-65 ปี ได้เข้าร่วมการทดสอบการใช้งานแอปพลิเคชัน (HealthMe App) และได้ประเมินผลการใช้งานในด้านความถูกต้องของฟังก์ชันที่รองรับตามวัตถุประสงค์ที่ได้ออกแบบไว้สำหรับงานวิจัยชิ้นนี้ รวมถึงประเมินความพึงพอใจของการใช้งานแอปพลิเคชัน โดยจากกระบวนการระบุความต้องการของระบบ (Requirement analysis) ผ่านการสัมภาษณ์และแบบสอบถาม รวมถึงการศึกษางานวิจัยที่เกี่ยวข้อง นักวิจัยได้วิเคราะห์และออกแบบฟังก์ชันของแอปพลิเคชัน โดยแสดงในรูปแบบของ use case diagram และคำอธิบายรายละเอียด (Use case description) กิจกรรมของการใช้งานแอปพลิเคชันในรูปแบบ Activity diagram และโครงสร้างของโปรแกรมและฐานข้อมูลในรูปแบบ (Class diagram) และ ER diagram รวมทั้งจะแสดงการออกแบบส่วนต่อประสานและประสบการณ์ของผู้ใช้

3.1 ความต้องการของระบบ

ความต้องการมี 2 แบบ คือ ความต้องการตามการใช้งาน (Functional Requirement) ซึ่งก็คือความต้องการที่ระบบที่เราพัฒนาต้องทำได้ มีทั้งหมด 9 ข้อ (ตามตารางที่ 1) และความต้องการที่ไม่ใช่ตามการใช้งาน (Non-Functional Requirement) ซึ่งเป็นความต้องการที่สนับสนุนการทำงานของระบบที่เราพัฒนา มีทั้งหมด 2 ข้อ

ตารางที่ 1. แสดงความต้องการตามการใช้งาน (Functional Requirement)

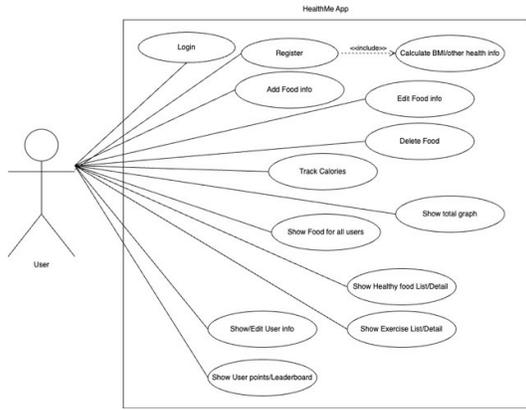
Requirement ID	Description
1	สามารถสมัครสมาชิกได้
2	สามารถคำนวณ BMI/BMR ได้
3	นำเข้ารูปอาหาร และใส่ข้อมูลที่เกี่ยวข้อง (สามารถแก้ไขได้ภายหลัง หรือลบออกได้)
4	สามารถติดตาม (Track) ปริมาณแคลอรีที่บริโภคไป เทียบกับที่แนะนำได้
5	แสดงกราฟปริมาณแคลอรีที่บริโภคไป และกราฟปริมาณแคลอรีที่บริโภคไป 7 วัน ย้อนหลัง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

สำหรับความต้องการที่ไม่ใช่ตามการใช้งาน (Non-Functional Requirement) ได้แก่

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

3.2 Use cases

Use case diagram คือ แผนภาพแสดงว่าผู้ใช้สามารถทำอะไรได้บ้างกับแอปพลิเคชัน ซึ่งในกรณีนี้ มีทั้งหมด 12 Use cases ดังรูปที่ 2 ด้านล่าง



รูปที่ 2. แสดง Use case diagram.

Use case description จะเป็นการบรรยายรายละเอียดของแต่ละ Use case ที่มี ซึ่งในกรณีนี้ มีทั้งหมด 12 Use cases (แต่จะแสดงตัวอย่างรายละเอียดให้ดูตามตารางที่ 2

ตารางที่ 2 แสดง Use case description UC01.

Use case number	UC01
Use case name	ลงทะเบียน (Register)
Pre-condition	ผู้ที่ยังไม่เคยลงทะเบียน
Primary Actor	ผู้ใช้งานแอป
Scenario	<ol style="list-style-type: none"> 1.ผู้ใช้งาน Download App 2.App แสดงหน้า Login และผู้ที่ยังไม่เคยลงทะเบียนมาก่อน ให้กดปุ่ม Sign Up 3.App แสดงหน้า Create Account

4.ผู้ใช้อกรอกข้อมูลตาม Fields ต่างๆ (App มีการ Validating fields ต่างๆ จะแสดงให้ผู้ใช้ทราบ)

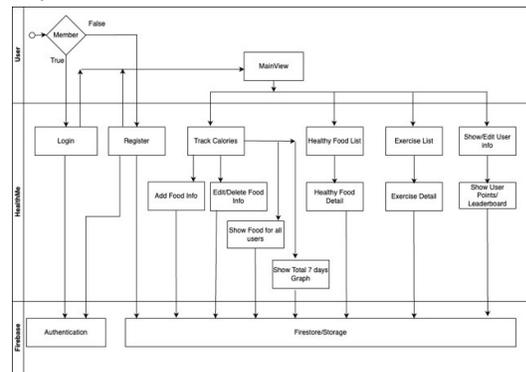
5.ผู้ใช้กดปุ่ม Sign Up ระบบจะสร้างข้อมูลของผู้ใช้ ซึ่งรวมทั้งระบบ Authentication (ใช้ Email/Password ของผู้ใช้) และข้อมูลของผู้ใช้ โดยระบบจะมีการคำนวณค่า BMI/BMR แล้วเก็บในฐานข้อมูล Firebase

6.App จะนำผู้ใช้เข้าสู่หน้าหลักของ App โดยอัตโนมัติ (หน้า Calories Tracker)

7.เมื่อผู้ใช้กลับมาใช้ App อีกครั้ง ระบบจะยังคงให้ผู้ใช้เข้าใช้งานหน้าหลักได้เลย (ระบบมีการเก็บ user session ไว้) นอกจากผู้ใช้ได้ทำการ Sign Out ออกจากระบบ (ปุ่ม Sign Out อยู่ในหน้า Profile) กรณีนี้ App จะนำผู้ใช้ไปสู่นำหน้า Login ซึ่งผู้ใช้จะต้องทำการ Login ด้วย Email/Password ที่ลงทะเบียนไว้

3.3 Activity Diagram

เป็นการแสดง Work flow ของระบบที่ผู้ใช้มีการตอบสนองด้วย ซึ่งรูปที่ 3 แสดงองค์รวมทั้งหมดของระบบ



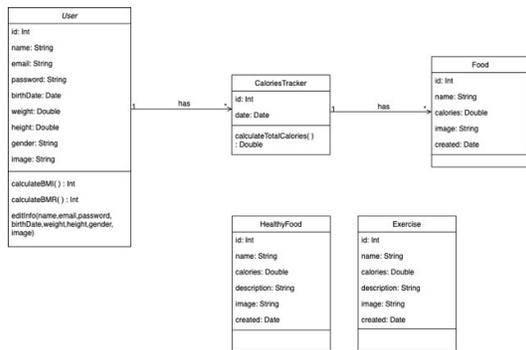
รูปที่ 3. แสดง Activity Diagram

3.4 Class Diagram

เป็นการแสดงโครงสร้างของระบบ ตามแผนรูปที่ 4 ประกอบไปด้วย

- User class
- CaloriesTracker class
- Food class
- HealthyFood class
- Exercise class

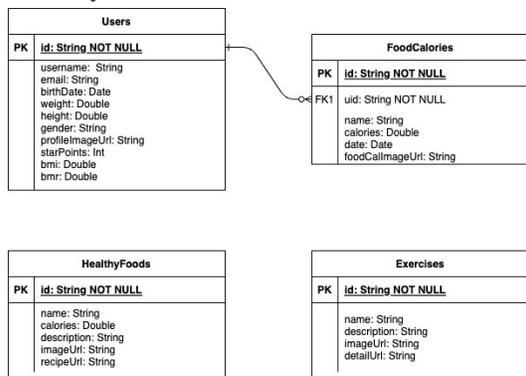
โดย User class มีความสัมพันธ์แบบ One-to-Many กับ CaloriesTracker class ซึ่งมีความสัมพันธ์แบบ One-to-Many กับ Food class ส่วน HealthyFood class กับ Exercise class ไม่มีความสัมพันธ์กับ class ใดๆ เพราะเป็นเพียงแสดงข้อมูลในระบบเท่านั้น



รูปที่ 4. แสดง Class Diagram.

3.5 ER Diagram

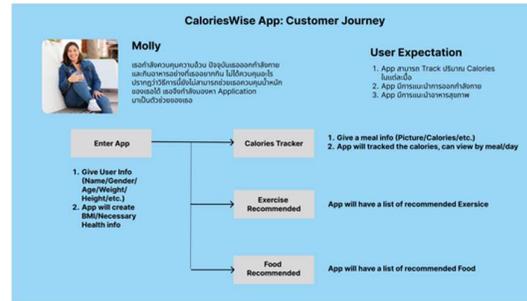
ER diagram หรือ Entity Relationship diagram แสดงโครงสร้างของ Database โดยแสดงเป็น Diagram ดังรูปที่ 5 ซึ่งจะเห็นได้ว่า Users Entity มีความสัมพันธ์แบบ One-to-Many กับ FoodCalories Entity สำหรับ HealthyFoods Entity กับ Exercises Entity ไม่มีความสัมพันธ์กับ Entity ใด เนื่องจากเป็นการแสดงข้อมูลในระบบเท่านั้น



รูปที่ 5. แสดง ER Diagram

3.6 UX/UI design

ในเบื้องต้นนักวิจัยได้ทำ Customer Journey ของ App มีการ Setup Persona ของผู้ใช้ App และ Expectation ของผู้ใช้ ดังแสดงรายละเอียด ตามรูปที่ 6 ด้านล่าง (ในขณะนั้นตั้งชื่อว่า CaloriesWise App และมาเปลี่ยนชื่อในภายหลังเป็น HealthMe App)



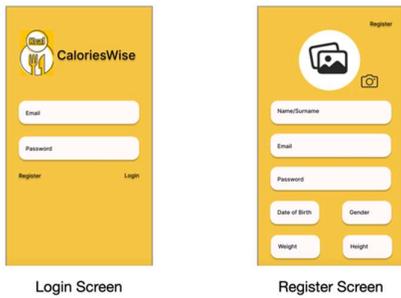
รูปที่ 6. แสดง Customer Journey

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

โดย Expectation ของ App นี้ก็คือ

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

ส่วน UI design ในเบื้องต้น ได้ทำการออกแบบโดยใช้ Figma ซึ่งมีแนวคิดคร่าวๆ ของ App คือ เป็นแอปที่เป็นกันเองไม่ทางการมากนัก ผู้ใช้แล้วควรจะรู้สึกสนุกกับการใช้และอยากกลับมาใช้แอปอยู่ตลอด จึงออกแบบแอปโดยใช้ 3 สี คือ สีเหลือง สีขาว และสีดำ เป็นโทนสีหลักของแอป ตามรูปที่ 7 แสดงตัวอย่าง UI ที่ออกแบบไว้เบื้องต้น ใน Figma



รูปที่ 7. แสดง Login screen and Register screen

3.7 เครื่องมือที่ใช้ในการพัฒนา

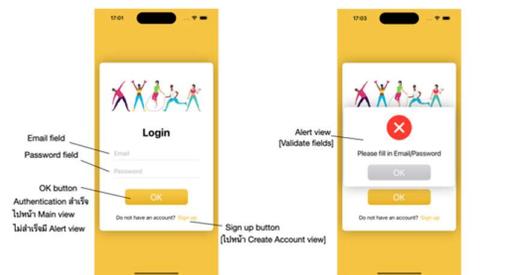
- ใช้ SwiftUI framework ในการพัฒนา
- ใช้ภาษา Swift ในการพัฒนา
- ใช้ Xcode เป็น IDE ในการพัฒนา
- ใช้ Firebase เป็นระบบฐานข้อมูล (CloudFirestore/Cloud Storage) รวมทั้งระบบ Firebase Authentication (โดยใช้ Email/Password)

4. ผลการวิจัย

ในส่วนนี้จะแสดงผลของการพัฒนาแอปพลิเคชัน โดยประยุกต์ใช้แนวทางของการพัฒนาตามกระบวนการซอฟต์แวร์แบบ Agile method และดำเนินการตามมาตรฐาน ISO29110 [8] โดยแสดง User Interface ของแอปพลิเคชันที่พัฒนา ฐานข้อมูล Firebase ที่สร้างขึ้น และผลการทดสอบระบบที่พัฒนาขึ้น

4.1 User Interface

แสดงตัวอย่าง User Interface ของแอปพลิเคชันที่พัฒนาสำเร็จตามรูปที่ 8 และ 9



รูปที่ 8. แสดงหน้าจอ Login view



รูปที่ 9 แสดงหน้าจอ Calories Tracker view

4.2 ฐานข้อมูล Firebase

สำหรับ Firebase ผู้พัฒนาใช้งานในแอปพลิเคชันอยู่ 3 ส่วนคือ

- (1) Authentication with Email/Password
- (2) Cloud Firestore เก็บข้อมูลเป็น Collections โดยเก็บข้อมูลตาม Data Model ที่ได้ออกแบบไว้ (ภาพที่ 3.4 ER Diagram)
- (3) Cloud Storage ใช้เก็บข้อมูลรูปภาพต่างๆ แล้วเก็บ Link ของรูปภาพนั้นไว้ที่ Cloud Firestore

4.3 การทดสอบระบบ

ในการทดสอบ นักวิจัยได้ทำการแยกเป็นกรณีทดสอบทั้งหมด 33 กรณี ดังในแสดงในตารางที่ 3 พบว่ากลุ่มผู้ร่วมทดสอบทั้งหมด สามารถใช้งานตามกรณีทดสอบได้ทุกกรณี และสามารถใช้งานแอปพลิเคชันได้ครบถ้วน ผลการทดสอบใช้งานแอปพลิเคชันตามกรณีพบว่าแอปพลิเคชันสามารถทำงานได้ครบถ้วนและถูกต้องร้อยละ 100

ตารางที่ 3 แสดง รายละเอียด Test cases

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)
TC06	การเข้าใช้งาน (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	แสดงข้อมูลแคลอรีรวมแต่ละวันย้อนหลัง 7 วัน	UC06 แสดงข้อมูลแคลอรีรวมในแต่ละวันย้อนหลัง 7 วันแบบกราฟแท่ง (Show Total Graph)
TC16	แสดงข้อมูลแคลอรีและอาหารที่ผู้ใช้ทั้งหมดเพิ่มไว้ในระบบ	UC07 แสดงข้อมูลแคลอรีและอาหารที่ผู้ใช้ทั้งหมดได้ทำการเพิ่มไว้ในระบบ (Show Food For All Users)
TC17	แสดงข้อมูลอาหารสุขภาพที่แนะนำพร้อมรายละเอียด	UC08 แสดงข้อมูลอาหารสุขภาพที่แนะนำ (Show Healthy Food List/Detail)
TC18	แสดงข้อมูลการออกกำลังกายที่แนะนำพร้อมรายละเอียด	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

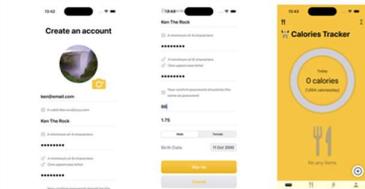
		(Show/Edit User Profile View)
TC22	แสดงหน้า Edit Profile view - ใส่ข้อมูลไม่ถูกต้อง	UC10 แสดงข้อมูลผู้ใช้/แก้ไข (Show/Edit User Profile View)
TC23	แสดงลำดับผู้ใช้ตามแต้มสะสม (Show Leaderboard)	UC11 แสดงลำดับของผู้ใช้ทั้งหมดที่มีแต้มสะสมเรียงจากมากไปน้อย (Show Leaderboard)
TC24	แสดงหน้าอาหารที่ผู้ใช้แต่ละคนเพิ่มไว้	UC11 แสดงลำดับของผู้ใช้ทั้งหมดที่มีแต้มสะสมเรียงจากมากไปน้อย (Show Leaderboard)
TC25	ทดสอบใส่ค่านักที่ ไม่สามารถแปลงเป็นตัวเลขได้ ในหน้า CreateAccountView	UC01 ลงทะเบียน (Register)
TC26	ทดสอบใส่ค่าส่วนสูงที่ ไม่สามารถแปลงเป็นตัวเลขได้ ในหน้า CreateAccountView	UC01 ลงทะเบียน (Register)
TC27	ทดสอบใส่ค่า Username เป็นภาษาไทยในหน้า CreateAccountView	UC01 ลงทะเบียน (Register)
TC28	ทดสอบใส่ค่า Calories field ที่ไม่สามารถแปลงเป็นตัวเลขได้ในหน้า AddCaloriesView	UC02 เพิ่มข้อมูลอาหารที่รับประทาน (Add food info)
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 คือตัวอย่างรายละเอียดและผลของแต่ละ Test case โดยทั้งหมดทำบน Test Environment คือ iPhone 15 Pro iOS 17.0.1 (Simulator) แต่ระบบที่รองรับคือ iOS 16.4 ขึ้นไป

ตารางที่ 4. TC01 การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลครบและถูกต้อง

Test case no.	TC01
Test case title	การลงทะเบียนผู้ใช้ใหม่ (Create new user account) - ใส่ข้อมูลครบและถูกต้อง
Precondition	Download App
Test steps	<ol style="list-style-type: none"> 1. กดปุ่ม Sign up ในหน้า Login 2. ระบบนำสู่หน้า Create user account 3. ผู้ใช้กรอกข้อมูลครบถ้วน และถูกต้อง 4. ระบบ Create user account ใน Firebase Authentication และ user ใน users collection ใน Cloud Firestore

	5. App นำผู้ใช้ไปหน้า Main view (Calories Tracker view)
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	

4.4 ผลการสำรวจความพึงพอใจของผู้ใช้

นักวิจัยได้ออกแบบการทดสอบ โดยกำหนดให้มีการบันทึกวิธีโอของผู้ร่วมทดสอบขณะที่มีการใช้งานแอปพลิเคชัน และใช้แบบสำรวจประเมินความพึงพอใจพร้อมทั้งแสดงความเห็นใน Google Form โดยสาระสำคัญของแบบสำรวจ ได้แก่ เพศของผู้ร่วมทำการทดสอบและประเมินผล อายุของผู้ร่วมทำการทดสอบและประเมินผล ความพึงพอใจด้านส่วนติดต่อประสานและประสบการณ์ของผู้ใช้ และความพึงพอใจต่อฟังก์ชันของแอปพลิเคชัน ดังแสดงในตารางที่ 5

ตารางที่ 5 แสดงผลการสำรวจความพึงพอใจ

เพศ	ชาย ร้อยละ 66.7	หญิง ร้อยละ 33.3	
อายุของผู้ตอบแบบสอบถาม:	18-35 ร้อยละ 66.7	35-60 ปี ร้อยละ 33.3	
ความพึงพอใจด้านส่วนติดต่อประสานและประสบการณ์ของผู้ใช้	พึงพอใจมากที่สุด ร้อยละ 16.7	พึงพอใจมาก ร้อยละ 75	พึงพอใจปานกลาง ร้อยละ 8.3
ความพึงพอใจต่อฟังก์ชันของแอปพลิเคชัน	พึงพอใจมากที่สุด ร้อยละ 33.3	พึงพอใจมาก ร้อยละ 50	พึงพอใจปานกลาง ร้อยละ 16

5. บทสรุป การอภิปราย และข้อเสนอแนะ

ในงานวิจัยนี้ นักวิจัยได้ศึกษาถึงปัญหาและความต้องการของการมีเครื่องมือสนับสนุนเพื่อส่งเสริมให้คนไทยได้มีสุขภาพที่ดีขึ้น โดยในบทความนี้ได้นำเสนอการวิเคราะห์ ออกแบบและพัฒนาแอปพลิเคชันบนอุปกรณ์พกพาเพื่อติดตามแคลอรีของการบริโภคอาหารของผู้ใช้ โดยใช้ SwiftUI framework ต่อกับฐานข้อมูล Firebase รองรับบนอุปกรณ์ iPhone iOS 16.4 ขึ้นไป โดยมุ่งหวังให้การออกแบบความต้องการและรูปแบบการใช้งานเป็นที่เหมาะสมและสอดคล้องกับการใช้งานของประชากรในไทย มีการเก็บข้อมูลผ่าน Firebase และยืนยันตัวตนในการใช้ด้วยอีเมลและรหัสผ่าน พร้อมทั้งเก็บรูปภาพ ใน Cloud Storage มีการทดสอบตามกรณีทดสอบฟังก์ชัน และสำรวจความพึงพอใจในการใช้งาน

การต่อยอดการดำเนินงานวิจัยจากนี้ จะมีการพัฒนาแอปพลิเคชันนี้โดยจะพัฒนาเพิ่มเติมฟังก์ชันอื่น ๆ เพื่อส่งเสริมสุขภาพ อาทิเช่น สามารถตั้งเป้าหมายการบริโภคแคลอรีโดยผู้ใช้ ผู้ใช้สามารถเมนูได้จากการแสกนภาพ การแลกของรางวัล สำหรับผู้ใช้ในชุมชนที่ทำได้บรรลุเป้าหมาย เป็นต้น นอกจากนี้ในส่วน User Interface ให้รองรับ iPad ด้วย และต่อยอดแอป

พลิกชันเพื่อให้รองรับ Android platform ต่อไป ซึ่งอาจใช้การ พัฒนาแบบ Native (Kotlin) หรือ Cross platform โดยใช้ Flutter หรือ React-Native และสามารถใช้งานข้อมูล Firebase เดียวกัน กับที่ได้พัฒนาไว้แล้ว

Retrieved from <https://li01.tci-thaijo.org/index.php/sehs/article/view/257969>

เอกสารอ้างอิง

- [1] สำนักงานสำรวจสุขภาพประชาชนไทย (สสท.). ม.ป.ป.. รายงานการสำรวจสุขภาพ ประชาชนไทยโดยการตรวจ ร่างกาย ครั้งที่ 4 ปี 2551-2552. นนทบุรี: บริษัท เดอะ กราฟิโก ซิสเต็มส์ จำกัด.
- [2] World Health Organization (WHO). 2013. World Health Statistics 2013. Geneva: WHO.
- [3] Techliance, “Best Mobile App Development Tools, Frameworks & Platforms”, Techliance Blog. <https://blog.techliance.com/mobile-app-development-tools/> (Accessed: Oct 27,2023)
- [4] Quantilus, “Why Your Business Should Develop An iOS App”, Quantilus innovation. <https://quantilus.com/article/why-your-business-should-develop-an-ios-app/> (Accessed: Oct 27,2023)
- [5] Jacob Schmitt, “Native vs cross-platform mobile app development”, Circleci Blog. <https://circleci.com/blog/native-vs-cross-platform-mobile-dev/> (Accessed: Oct 27,2023)
- [6] Mayank Pratap, “SwiftUI vs UIKit: Which one to choose for your iOS app”, Supersourcing. <https://supersourcing.com/blog/swiftui-vs-uikit-which-one-to-choose-for-your-ios-app/#portfolio> (Accessed: Oct 27,2023)
- [7] Bogdan Guriev, “Firebase Pros and Cons: When you should and shouldn’t use Firebase”, OSDB. <https://osdb.io/firebase-pros-and-cons-when-you-should-and-shouldnt-use-firebase-osdb/> (Accessed: Oct 27,2023)
- [8] Jirapanthong, W. (2023). Toward traceability model between ISO/IEC 29110 artefacts and agile process. Science, Engineering and Health Studies, 17, 23040006.