

แอปพลิเคชันเวิร์คเอาท์

Work Out Application



นายณัฐพงศ์ พรหมศรี 6004800057

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์

มหาวิทยาลัยสยาม

ปีการศึกษา 2564

หัวข้อปริญญานิพนธ์

แอปพลิเคชันเวิร์คเอาท์

Work Out Application

หน่วยกิตของปริญญานิพนธ์

3 หน่วยกิต

รายชื่อคณะผู้จัดทำ

นาย ณัฏฐพงษ์ พรหมสร 6004800057

อาจารย์ที่ปรึกษา

อาจารย์เอก บำรุงศรี

ระดับการศึกษา

วิทยาศาสตรบัณฑิต

ภาควิชา

วิทยาการคอมพิวเตอร์

ปีการศึกษา

2564

อนุมัติให้ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์

คณะกรรมการสอบปริญญานิพนธ์

.....ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.ภูชงค์ อุทโยภาส)

.....กรรมการ
(อาจารย์ธนาภรณ์ รอดชีวิต)

.....อาจารย์ที่ปรึกษา
(อาจารย์เอก บำรุงศรี)

หัวข้อปริญญานิพนธ์	แอปพลิเคชันเวิร์คเอาท์
หน่วยกิตของปริญญานิพนธ์	3 หน่วยกิต
รายชื่อผู้จัดทำ	นายณัฐพงศ์ พรหมสร 6004800057
อาจารย์ที่ปรึกษา	อาจารย์เอก บำรุงศรี
ระดับการศึกษา	วิทยาศาสตรบัณฑิต
ภาควิชา	วิทยาการคอมพิวเตอร์
ปีการศึกษา	2564

บทคัดย่อ

ปริญญานิพนธ์นี้มีวัตถุประสงค์ในการจัดทำเพื่อพัฒนาแอปพลิเคชันเวิร์คเอาท์ออกกำลังกายเองได้ที่บ้านอย่างถูกต้อง เพื่อช่วยให้ผู้ใช้ที่ไม่มีความรู้เทคนิคในการออกกำลังกาย ได้มีความรู้ความเข้าใจในการออกกำลังกายมากยิ่งขึ้น และเลือกกินอาหารที่มีประโยชน์ตามที่แนะนำเพื่อเสริมสร้างกล้ามเนื้อได้ดียิ่งขึ้น ผู้ใช้สามารถออกกำลังกายตามตารางที่แนะนำ เพื่อแบ่งสัดส่วนการออกกำลังกายในแต่ละวัน และเทียบสัดส่วนก่อนออกกำลังกายและหลังออกกำลังกาย ว่ามีการเปลี่ยนแปลงของร่างกายมากน้อยเพียงใด ออกกำลังกายในส่วนที่ต้องการหรือส่วนต่างๆเพิ่มเติมเพื่อเพิ่มแรงจูงใจในการออกกำลังกายจึงมีระบบคะแนนในการออกกำลังกายแต่ละส่วน และยังสามารถนำคะแนนที่ได้จากการออกกำลังกายนำมาแลกซื้อสินค้า แนะนำชุดรหัส ที่ได้จากการซื้อสินค้า นำมาแลกสินค้ากับทางร้านที่ร่วมรายการ และพัฒนาซอฟต์แวร์ตามหลักการทฤษฎีแอปพลิเคชัน ด้วยภาษา Flutter, Dart และ Java ด้วยโปรแกรม Visual Studio Code จัดการฐานข้อมูลด้วย Firebase

คำสำคัญ: ออกกำลังกาย, แอปพลิเคชัน, ฟลัตเตอร์

Project Title Work Out Application

Project Credits 3 Units

Candidates Mr. Natthapong Promsorn 6004800057

Advisor Mr. Eak Bamrungsi

Program Bachelor of Science

Field of study Computer Science

Academic year 2021

Abstract

The goal of this project was to create a work out application that allows exercise to be done correctly at home. Also to assist users who are unfamiliar with exercise techniques in gaining a better grasp of exercise and choosing nutritious foods, as indicated, in order to build muscles more effectively. The recommended exercise schedule can be followed by users. Users can divide the daily exercise proportion and compare the proportions before and after exercise to see if there is a difference in the body. There is a point system for each workout to boost exercise motivation, as well as the points earned from the workout can be traded for things; Introduce a series of codes gained from product purchases to exchange products with participating stores; and construct the application based on application theory with Visual Studio Code's Flutter, Dart, and Java languages; manage databases with Firebase.

Keywords: exercise/application/Flutter

Approved by

.....

Approved by

.....

กิตติกรรมประกาศ

(Acknowledgment)

การจัดทำปริญญานิพนธ์ฉบับนี้สำเร็จได้นั้น คณะผู้จัดทำได้รับความกรุณาจากอาจารย์ผู้สอนทุกท่านที่ให้ข้อมูลต่างๆ ส่งผลให้คณะผู้จัดทำได้รับความรู้และประสบการณ์ต่างๆ ที่มีค่ามากมายสำหรับปริญญานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดีจากความร่วมมือและสนับสนุนจาก ดังนี้

1. อาจารย์เอก บำรุงศรี อาจารย์ที่ปรึกษา

ผู้จัดทำใคร่ขอขอบพระคุณคณะกรรมการสอบปริญญานิพนธ์ ที่ได้ให้คำแนะนำสำคัญ เพื่อให้การสอบปริญญานิพนธ์ฉบับนี้ และผู้มีส่วนร่วมทุกท่าน รวมทั้งผู้ที่ไม่ได้กล่าวนาม ที่มีส่วนร่วมในการให้ข้อมูลให้ความช่วยเหลือ และเป็นທີ່ปรึกษาให้คำแนะนำต่างๆ จนทำให้งานทุกอย่างประสบความสำเร็จไปด้วยดี และทำรายงานฉบับนี้จนเสร็จสมบูรณ์ ซึ่งผู้จัดทำขอขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้ด้วย

ผู้จัดทำ

นายณัฐพงศ์ พรหมศรี

สารบัญ

หน้า

บทคัดย่อ.....	ก
Abstract	ข
กิตติกรรมประกาศ.....	ค
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของปริญญานิพนธ์	1
1.3 ขอบเขตปริญญานิพนธ์.....	1
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 ขั้นตอนและวิธีการดำเนินงานปริญญานิพนธ์	2
1.6 แผนและระยะเวลาในการดำเนินปริญญานิพนธ์	5
1.7 อุปกรณ์และเครื่องมือที่ใช้ในการพัฒนา.....	5
1.8 อุปกรณ์และเครื่องมือที่รองรับระบบ	5
บทที่ 2 การทบทวนวรรณกรรมและเอกสารที่เกี่ยวข้อง	
2.1 Cross Platform	6
2.2 Flutter	8
2.3 Dart	11
2.4 Firebase	12
บทที่ 3 การวิเคราะห์และออกแบบระบบ	
3.1 รายละเอียดของปริญญานิพนธ์.....	14
3.2 แสดงฟังก์ชันการทำงานของระบบด้วย Use Case Diagram	15
3.3 คำอธิบายรายละเอียดของยูสเคส Use Case Description	16
3.4 Sequence Diagram	23
3.5 Class Diagram.....	32
3.6 โครงสร้างของข้อมูลใน Firebase	49
บทที่ 4 รายละเอียดปริญญานิพนธ์	
4.1 การออกแบบส่วนติดต่อผู้ใช้	52
บทที่ 5 สรุปผลและข้อเสนอแนะ	
5.1 สรุปผลปริญญานิพนธ์.....	61
5.2 ข้อดีของระบบ	61

สารบัญ(ต่อ)

หน้า

5.3 ข้อเสนอแนะ.....	61
บรรณานุกรม.....	62



สารบัญตาราง

หน้า

ตารางที่ 1.1 แผนและระยะเวลาในการดำเนินงานปริญญานิพนธ์	5
ตารางที่ 3.1 แสดงรายละเอียดของ Use case Login	16
ตารางที่ 3.2 แสดงรายละเอียดของ Use case Register	17
ตารางที่ 3.3 แสดงรายละเอียดของ Use case Proportion	18
ตารางที่ 3.4 แสดงรายละเอียดของ Use case Program workout	19
ตารางที่ 3.5 แสดงรายละเอียดของ Use case calendar	19
ตารางที่ 3.6 แสดงรายละเอียดของ Use case food	20
ตารางที่ 3.7 แสดงรายละเอียดของ Use case shop	20
ตารางที่ 3.8 แสดงรายละเอียดของ Use case profile	21
ตารางที่ 3.9 แสดงรายละเอียดของ Use case menu workout	22
ตารางที่ 3.10 Calendar Data	50
ตารางที่ 3.11 Proportion	50
ตารางที่ 3.12 scoreabs	50
ตารางที่ 3.13 scoreback	50
ตารางที่ 3.14 scorebiceps	51
ตารางที่ 3.15 scorechat	51
ตารางที่ 3.16 scorequads	51
ตารางที่ 3.17 scoreticeps	51

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 Dart Platform	12
รูปที่ 3.1 Use Case Diagram ของแอปพลิเคชัน Workout	15
รูปที่ 3.2 Sequence diagram : Login	23
รูปที่ 3.3 Sequence diagram : Register	24
รูปที่ 3.4 Sequence diagram : Proportion	25
รูปที่ 3.5 Sequence diagram : Program workout	26
รูปที่ 3.6 Sequence diagram : Calendar	27
รูปที่ 3.7 Sequence diagram : Food	28
รูปที่ 3.8 Sequence diagram : Show	29
รูปที่ 3.9 Sequence diagram : Profile	30
รูปที่ 3.10 Sequence diagram : Menu workout	31
รูปที่ 3.11 แผนภาพ Class Diagram workout	32
รูปที่ 3.12 โครงสร้างของข้อมูล (Firebase)	49
รูปที่ 4.1 หน้า Welcome	52
รูปที่ 4.2 หน้าเข้าสู่ระบบแอปพลิเคชัน	53
รูปที่ 4.3 แสดงหน้าลงทะเบียนผู้ใช้ใหม่.....	54
รูปที่ 4.4 แสดงหน้าหมวดหมู่การออกกำลังกาย	55
รูปที่ 4.5 แสดงหน้าตารางการออกกำลังกาย.....	56
รูปที่ 4.6 แสดงหน้าการสอนออกกำลังกาย.....	57
รูปที่ 4.7 แสดงหน้าการแนะนำคะแนนมาแลกสินค้า	58
รูปที่ 4.8 แสดงหน้าปฏิทินและกิจกรรม	59
รูปที่ 4.9 แสดงหน้ารายการอาหารแนะนำ.....	60

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

เพื่อช่วยให้คนที่ต้องการออกกำลังกาย ได้ใช้โปรแกรมเวิร์คเอาท์ช่วยในการออกกำลังกายอย่างถูกวิธีมากยิ่งขึ้น เพราะคนที่ต้องการออกกำลังกายในปัจจุบัน ยังไม่รู้เทคนิคและวิธีการออกกำลังกายที่ถูกต้อง จึงทำให้ผู้ที่ต้องการออกกำลังกายไม่มีความกล้าที่จะออกกำลังกาย เพราะท่าออกกำลังกายแบบผิดๆ ไปแล้ว ก็จะไม่เห็นผลการเปลี่ยนแปลงหรือพัฒนาของร่างกาย แล้วยังทำให้ร่างกายได้รับความบาดเจ็บ เนื่องจากการออกกำลังกายแบบผิดๆ และผู้ชายก็จะไม่อยากออกกำลังกายอีก จึงเป็นที่มาในการพัฒนาแอปพลิเคชัน เวิร์คเอาท์ เพื่อช่วยให้ผู้ใช้นั้น สามารถออกกำลังกายได้อย่างถูกต้องมากยิ่งขึ้น ช่วยให้ผู้ชายนั้นได้มีความรู้ ถึงเทคนิคการออกกำลังกายที่ถูกต้อง เพื่อพัฒนา ร่างกายให้มีรูปลักษณ์ที่ดีมากยิ่งขึ้น และยังเลือกกินอาหารที่มีประโยชน์ต่อร่างกายเพื่อช่วยเสริมสร้างกล้ามเนื้อให้ดียิ่งขึ้น ผู้ชายจะสามารถดูตารางการออกกำลังกาย เพื่อแบ่งสัดส่วนในการออกกำลังกายแต่ละวัน ในแต่ละหมวดหมู่การออกกำลังกายจะแบ่งเป็นหกถ้ำในการออกสตาร์ต ส่วน แต่ละท่าจะมีรูปประกอบและคำอธิบายในการออกกำลังกายเพื่อช่วยให้ผู้ออกกำลังกายสามารถออกกำลังกายได้อย่างถูกต้องตามวิธีมากยิ่งขึ้น ส่วนเรื่องอาหารที่ต้องกินเพื่อช่วยในการเสริมสร้างกล้ามเนื้อผู้ใ้สามารถเลือกซื้อตามรายการที่แนะนำได้ และผู้ใ้ยังสามารถก็สัดส่วนก่อนออกกำลังกาย และหลังออกกำลังกาย เพื่อเทียบสัดส่วนก่อนและหลังว่ามีการเปลี่ยนแปลงของร่างกายมากน้อยเพียงใด ผู้ใ้จะได้มุ่งเน้นการออกกำลังกายในส่วนมากยิ่งขึ้น หรือส่วนต่างๆ เพิ่มเติม เพื่อให้ร่างกายพัฒนาเติบโตได้ตามที่ต้องการ เพื่อเพิ่มแรงจูงใจในการออกกำลังกายให้ผู้ใช้ ออกกำลังกายมากยิ่งขึ้น จึงมีระบบคะแนนในการออกกำลังกายแต่ละส่วน เพื่อให้ผู้ใ้จะนำคะแนนที่ได้จากการออกกำลังกายนั้นไปแลกซื้อสินค้า เมื่อผู้ชายนำคะแนนไปแลกสินค้าจะได้ชุด โค้ชรหัส เพื่อนำไปแลกสินค้ากับทางร้านค้าที่ร่วมรายการ

1.2 วัตถุประสงค์ของปริญญานิพนธ์

พัฒนาแอปพลิเคชันเวิร์คเอาท์

1.3 ขอบเขตของปริญญานิพนธ์

1.3.1 การจัดการข้อมูลพื้นฐาน

1.3.1.1 ตรวจสอบ E-mail และ Password

1.3.1.2 ตรวจสอบสัดส่วนก่อนออกกำลังกาย

1.3.1.3 ตรวจสอบสัดส่วนหลังออกกำลังกาย

1.3.1.4 ตรวจสอบคะแนนที่ได้จากการออกกำลังกาย

1.3.2 ข้อมูลวิธีการออกกำลังกาย

1.3.2.1 ออกกำลังกายตามภาพ

1.3.2.2 จัดท่าทางตามคำอธิบาย

1.3.3 ข้อมูลการเลือกกินอาหาร

1.3.3.1 เลือกกินอาหารตามที่แนะนำ

1.3.3.2 เลือกซื้ออาหารตามที่ต้องการในรายการที่แนะนำ

1.3.4 ข้อมูลผู้ใช้

1.3.4.1 ตรวจสอบ รอบแขน/รอบอก/รอบเอว/รอบขา

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 ช่วยอำนวยความสะดวกให้แก่ผู้ที่ใช้งานระบบออกกำลังกาย

1.4.2 สามารถออกกำลังกายได้อย่างถูกวิธีมากยิ่งขึ้น

1.4.3 มีความมั่นใจในรูปร่างมากขึ้น

1.5 ขั้นตอนและวิธีการดำเนินงานปริญญานิพนธ์

1.5.1 การรวบรวมความต้องการและศึกษาข้อมูลของโครงการ (Detailed Study)

รวบรวมความต้องการโดยมีการประชุมกันระหว่างคณะผู้จัดทำและอาจารย์ที่ปรึกษา ตลอดจนการใช้แอปพลิเคชันออกกำลังกายต่างๆ และนำข้อดี ข้อเสีย ของแต่ละแอปพลิเคชันมาวิเคราะห์ในการออกแบบระบบ ว่ามีความสามารถทำอะไรบ้างขอบเขตของการทำงานเป็นอย่างไร รวบรวมปัญหาที่ได้รับจากการปฏิบัติงานเพื่อนำไปแก้ไขและปรับปรุงให้ระบบมีประสิทธิภาพการทำงานที่ดีขึ้น

1.5.2 การวิเคราะห์ระบบงาน (System Analysis)

นำข้อมูลต่างๆที่ได้รวบรวมมานำมาวิเคราะห์และวางแผนปฏิบัติงานและกำหนดขอบเขตที่ได้จากการรวบรวมข้อมูลเพื่อทำการออกแบบระบบเพื่อให้ตอบสนองต่อความต้องการของผู้ใช้ อย่างสมบูรณ์

1.5.3 การออกแบบระบบงาน (System Design)

ออกแบบสถาปัตยกรรม (Architecture Design) สถาปัตยกรรมที่ใช้ในการพัฒนา โครงการ เป็นแบบไคลเอนต์/เซิร์ฟเวอร์

ออกแบบฐานข้อมูล (Database Design) ฐานข้อมูลของระบบมีปริมาณของข้อมูลจำนวนมาก การออกแบบฐานข้อมูลจึงมีความซับซ้อน การออกแบบฐานข้อมูลต้องใช้เวลาในการดำเนินการพอสมควรและจำเป็นต้องออกแบบ อย่างเหมาะสมเพื่อตอบสนองความต้องการของผู้ใช้และความรวดเร็วในการประมวลผลข้อมูล ฐานข้อมูลที่เลือกใช้เป็นฐานข้อมูลเชิงสัมพันธ์ ซึ่งออกแบบโดยยึดหลักกฎความบูรณาภาพเพื่อลดความซ้ำซ้อนของข้อมูลดังนี้

กฎความบูรณาภาพของเอนทิตี (Entity Integrity Rule) คือการระบุไว้ว่าแอททริบิวต์ใดที่เป็นคีย์หลัก ข้อมูลในแอททริบิวต์นั้นจะเป็นค่าว่าง (Null) ไม่ได้ ความหมายของ การเป็นค่าว่างไม่ได้ (Not Null) หมายความว่า ข้อมูลของแอททริบิวต์ที่เป็นคีย์หลักจะไม่ทราบค่าที่แน่นอนหรือไม่มีค่าไม่ได้

กฎความบูรณาภาพของการอ้างอิง (Referential Integrity Rule) การ อ้างอิงข้อมูลระหว่างรีเลชันในฐานข้อมูลเชิงสัมพันธ์จะใช้คีย์นอกของรีเลชันหนึ่งไปตรวจสอบกับค่าของแอททริบิวต์ที่เป็นคีย์หลักของอีกรีเลชันหนึ่งเพื่อเรียกดูข้อมูลอื่น ๆ ที่ เกี่ยวข้องหรือค่าของคีย์นอกจะต้องอ้างอิงให้ตรงกับค่าของคีย์หลักได้จึงจะสามารถเชื่อมโยง ข้อมูลระหว่างสองรีเลชันได้สำหรับคีย์นอกจะมีค่าว่างได้หรือไม่ขึ้นอยู่กับกฎเกณฑ์การออกแบบฐานข้อมูล

ออกแบบส่วนติดต่อกับผู้ใช้ (User Interface Design) ส่วนติดต่อกับผู้ใช้หรือหน้าจอ แต่ละหน้าจอของแอปพลิเคชันโครงสร้างการทำงานของสถาปัตยกรรม Android ได้มีการออกแบบการพัฒนาแอปพลิเคชันเพื่อแบ่งออกเป็นสองส่วนคือ ส่วนของการทำงานหลักและส่วน ติดต่อกับผู้ใช้โดยส่วนติดต่อกับผู้ใช้พัฒนาโดยใช้ภาษา XML (Extensible Markup Language) ทำให้สะดวกต่อการจัดการแอปพลิเคชันจากนั้นจึงเขียนชุดคำสั่งในการติดต่อตัวควบคุม (Controller) ต่าง ๆ ในส่วนติดต่อกับผู้ใช้

1.5.4 การพัฒนาระบบ (System Develop)

ขั้นตอนในการพัฒนาระบบเป็นการนำข้อมูลทั้งหมดที่ได้วิเคราะห์และออกแบบไว้มาทำการสร้างและเขียนชุดคำสั่งด้วยโปรแกรม Android Studio โปรแกรม Microsoft Visual Studio Code สำหรับใช้ในการเขียนชุดคำสั่งสำหรับเว็บไซต์ของผู้ดูแลระบบและโปรแกรม Netbeans ide 8.2 สำหรับเขียนชุดคำสั่งของเว็บเซิร์ฟเวอร์เพื่อใช้ในการติดต่อกับฐานข้อมูลโดยใช้โปรแกรม MySQL Workbench

1.5.5 การทดสอบระบบ (System Testing)

Unit Testing ตรวจสอบความผิดพลาดของแต่ละฟังก์ชันการทำงาน ตรวจสอบโดยผู้จัดทำเอง เป็นการทดสอบแอปพลิเคชันในระดับ Function Call เพื่อเป็นการยืนยันการทำงานของระดับย่อยที่สุดของแอปพลิเคชันว่าทำงานได้อย่างถูกต้อง

Integration Testing ทดสอบการทำงานของแอปพลิเคชัน โดยทดสอบการตอบสนองของแต่ละความต้องการของอาจารย์ที่ปรึกษา แต่ละหน้าของแอปพลิเคชันว่ามีการทำงานที่ สมบูรณ์ และถูกต้องโดยทำการเชื่อมต่อส่วนย่อย ๆ ของ Module นำมาประกอบกันเป็นแอปพลิเคชัน

System Testing ทดสอบการเชื่อมต่อของแอปพลิเคชันกับ Server โดยการ เชื่อมต่อผ่านเครือข่ายอินเทอร์เน็ตและทดสอบการเชื่อมต่อแอปพลิเคชันกับ Server ผ่านเครือข่ายโทรศัพท์มือถือและผ่านเครือข่ายอินเทอร์เน็ตไร้สาย

Acceptant Testing นำแอปพลิเคชันไปให้ผู้ใช้กลุ่มลูกค้า ทดลองใช้แอปพลิเคชันเพื่อตรวจสอบว่าแอปพลิเคชันสามารถทำงานได้ตรงกับความต้องการของผู้ใช้หรือไม่พร้อมกันรับข้อเสนอแนะมาปรับแก้ไขให้ถูกต้อง

Usability Testing หลังจากการพัฒนาแอปพลิเคชันเสร็จสิ้นสมบูรณ์แล้ว นำแอปพลิเคชันให้ผู้เชี่ยวชาญทำการทดสอบโดยให้อาจารย์ที่ปรึกษาโครงการ อาจารย์จรรยา แหยมเจริญ ทดสอบการใช้ของแอปพลิเคชันเพื่อบอกปัญหาของแอปพลิเคชันและเสนอแนะแนวทางในการแก้ไข

1.5.6 จัดทำเอกสารประกอบปริญญานิพนธ์ (Documentation)

จัดทำเอกสารประกอบปริญญานิพนธ์แนวทางในการจัดทำปริญญานิพนธ์วิธีการและขั้นตอน การดำเนินปริญญานิพนธ์เพื่อแสดงรายละเอียดการพัฒนาแอปพลิเคชันละเป็นคู่มือในการใช้งาน แอปพลิเคชันอีกทั้งยังเป็นเอกสารสำหรับการนำแอปพลิเคชันไปพัฒนาต่อในอนาคต

1.6 แผนและระยะเวลาดำเนินงานปริญญานิพนธ์

ตารางที่ 1.1 แผนและระยะเวลาในการดำเนินงานปริญญานิพนธ์

ขั้นตอนในการดำเนินงาน	2563					2564			
	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.
1. รวบรวมความต้องการ			←→						
2. วิเคราะห์ระบบ				←→					
3. ออกแบบระบบ				←→					
4. พัฒนาระบบ						←→		→	
5. ทดสอบระบบ								←→	→
6. จัดทำเอกสาร								←→	→

1.7 อุปกรณ์และเครื่องมือที่ใช้ในการพัฒนา

1.7.1 ฮาร์ดแวร์

เครื่องคอมพิวเตอร์โน้ตบุ๊ก Acer Intel core i7-8700

1.7.2 ซอฟต์แวร์

- ระบบปฏิบัติการ Microsoft Windows 10
- โปรแกรม Android Studio
- โปรแกรม Adobe Illustrator
- โปรแกรม Visual Studio Code
- โปรแกรม Microsoft Office 2010
- โปรแกรม Google Chrome

1.8 อุปกรณ์และเครื่องมือที่รองรับ

1.8.1 ฮาร์ดแวร์

โทรศัพท์มือถือแอนดรอยด์ทุกรุ่นที่มีขนาดหน้าจอ 7 นิ้วลงไป

1.8.2 ซอฟต์แวร์

ระบบปฏิบัติการแอนดรอยด์ Android (OS) ตั้งแต่เวอร์ชัน 4.4 (Kitkat) เป็นต้นไป

บทที่ 2

การทบทวนวรรณกรรมที่เกี่ยวข้อง

สุขภาพของคนเป็นสิ่งสำคัญ ทำให้คนเริ่มหันมาใส่ใจกับสุขภาพกันมากขึ้นและแอปพลิเคชันออกกำลังกายในปัจจุบันได้รับความนิยมเป็นอย่างมาก เพราะแอปพลิเคชันออกกำลังกายมีฟังก์ชันที่สามารถช่วยให้เราไปถึงเป้าหมายได้เร็วขึ้น เช่น การมีหุ่นที่ดี และสุขภาพที่แข็งแรง และการออกกำลังกายของแต่ละคนมีวัตถุประสงค์ที่ไม่เหมือนกัน บางคนออกกำลังกายเพื่อลดน้ำหนัก บางคนออกกำลังกายเพื่อสุขภาพ และบางคนออกกำลังกายเพื่อสร้างกล้ามเนื้อ โดยการจัดทำแอปพลิเคชันแอปพลิเคชันแนะนำเทคนิคในการดูแลสุขภาพ ซึ่งผู้ใช้สามารถบรรลุตามเป้าหมายหรือจุดประสงค์ของการดูแลสุขภาพและสร้างกล้ามเนื้อได้ โดยการพัฒนาโปรแกรมเสริมสร้างกล้ามเนื้อ เพื่อเป็นการเพิ่มประสิทธิภาพในการดำเนินงานนี้ให้ดียิ่งขึ้น แอปพลิเคชันนี้เป็นระบบโมบาย แอปพลิเคชันครอสแพลตฟอร์ม(Cross - Platform) เพื่อที่จะรันแอปพลิเคชันได้ทั้งระบบ IOS และทั้ง Android โดยใช้ Firebase ในการจัดเก็บข้อมูล

2.1 Cross Platform¹

Cross platform คือระบบปฏิบัติการที่สามารถทำงานได้หลายแพลตฟอร์ม Cross platform เรียกกันอีกอย่างว่า Hybrid Mobile App เช่น ซอฟต์แวร์สามารถรันในระบบ windows บนสถาปัตยกรรม x86 และ Mac OS X บน PowerPC แพลตฟอร์ม

2.1.1 รูปแบบสำคัญของ Cross Platform ใน Java

การทำงานข้ามระบบของ java คือการสร้างสิ่งที่เรียกว่า Runtime ที่จำเป็นต้องติดตั้งลงในเครื่องคอมพิวเตอร์ก่อน ถึงจะเรียกใช้งานโปรแกรมที่เขียนด้วยภาษา Java ได้

ดังนั้น Runtime จะทำหน้าที่เหมือนเป็น “ล่ามท้องถิ่น” ที่ประสานการทำงานระหว่างภาษา java กับระบบปฏิบัติการบนเครื่องเป้าหมาย Java จึงเป็นเทคโนโลยีหนึ่งที่ทำให้แนวทางการทำงานข้ามระบบเป็นจริง ได้ การเรียนรู้วิธีใช้ภาษา java ก็สามารถพัฒนาโปรแกรมใช้งานระบบต่างๆได้ครบตามที่ติดตั้ง Java Runtime ลงไป

2.1.2 Mobile Application : เป้าหมายท้าทายของ Cross Platform

ความแตกต่างหนึ่งระหว่างโปรแกรมติดตั้งใช้งานดั้งเดิม กับเว็บแอปพลิเคชันคือตอนเปิดใช้งาน ซึ่งแบบโปรแกรมดั้งเดิมจะเร็วกว่ามาก เพราะทั้งหมดอยู่บนเครื่องคอมพิวเตอร์อยู่แล้ว ไม่ต้องต่ออินเทอร์เน็ต เพื่อเรียกไฟล์จากผู้ให้บริการมาอีก

¹ <https://nextflow.in.th/2016/10-year-cross-platform-application-development-thai>

ด้วยเหตุผลทางธุรกิจ ได้ถูกบังคับให้เลือกฝ่ายอีกครั้งว่าจะไปทาง Android ใช้ภาษา Java เวอร์ชันปรับแต่ง หรือไปทาง iOS ใช้ภาษา Objective-C หรือ Swift โดยจะเรียกการพัฒนาโมบายล์ แอปพลิเคชันตามเครื่องมือ และภาษาที่เจ้าของระบบกำหนดมาว่า เนทีฟแอป (Native Application Development)

2.1.3 Cross Platform Native Application อีกแนวทางเลือก

ในช่วงยุคที่ 1 ของ Hybrid App ก็มีนักพัฒนาบางกลุ่มคิดว่า บางทีอาจจะมีวิธีอื่นนอกเหนือจากการใช้เทคโนโลยีเว็บก็ได้โดยมีแนวคิดแตกออกเป็น 2 ทางดังนี้

1. น่าจะมีระบบสักอย่างที่สามารถแปลงโค้ดเป็นโมบายแอปพลิเคชันในแต่ละระบบได้
2. น่าจะมีระบบสักอย่างที่สามารถควบคุมโมบายล์แอปพลิเคชันในแต่ละระบบได้

นั่นคือแนวคิดนี้ไม่เลือกที่จะใช้เทคโนโลยีเว็บเป็นตัวแสดงหน้าแอป แบบเดียวกับ Hybrid App แต่เลือกที่จะหาทางควบคุมการพัฒนาให้ได้ผลลัพธ์เป็น Native App ในรูปแบบต่างๆ เช่น สำหรับ Xamarin.Forms ใช้วิธีการแปลงระบบ C# ของตัวเองให้ทำงานได้กับระบบ Native รวมถึงการสร้างส่วนติดต่อผู้ใช้แบบ Native UI จาก XAML หรือจะเป็น React Native ของ Facebook ที่ใช้แนวคิดส่วนติดต่อ ผู้ใช้บนเว็บของตัวเอง มาใช้สร้าง และควบคุมการทำงานของส่วนติดต่อผู้ใช้ Native UI บนระบบ iOS หรือ Android สำหรับ React Native พักหนึ่ง มันจะง่ายและใช้เวลาน้อยกว่าปัจจุบันในการนำมาใช้งานในโปรเจกต์จริง

และในช่วงที่ Cross Platform Native Application ขึ้นมาเป็นตัวเลือกอีกตัว นอกจาก Hybrid Application คือยุคที่ 3 ของ Cross Platform ฝั่ง Mobile Application ในปัจจุบัน

2.1.4 จุดอ่อนของ Cross Platform Native App

เช่นเดียวกับเทคโนโลยี Hybrid App ทาง Cross Platform Native App ก็มีสิ่งที่ต้องแลกมา และจากการนำ Xamarin และ React Native มาใช้งานในหลายๆ โปรเจกต์จุดอ่อนของมันคือ

1. ความซับซ้อนของตัวเฟรมเวิร์คเอง
2. การสนับสนุนต่อเนื่องของผู้สร้าง
3. ปัญหาเรื่องความเข้ากันไม่ได้ในระบบต่างๆ

2.2 Flutter²

เมื่อปลายเดือนกุมภาพันธ์ ปี 2018 google เปิดตัว flutter framework แบบ beta เวอร์ชันแรก นี่เป็น เครื่องมือ การสร้าง native interface สำหรับ android และ iOS และในเดือนสิงหาคม ปี 2019 ก็ได้ออกเวอร์ชัน 1.7.8 เป็นเวอร์ชันล่าสุด ณ ตอนนี

Framework สามารถพัฒนา native app แบบ cross-platform ได้ การรวบรวมข้อมูล Native ให้ตัว ประมวลผล ARM นั้น ช่วยให้การพัฒนาง่ายขึ้น มีการ render ที่ง่ายขึ้น พวกชุดเครื่องมือ, widgets, และ features ก็มีตัวช่วยให้ process นั้นง่ายขึ้นเช่นกัน แล้วมีอะไรเพิ่มเติมอีกล่ะ นอกจากการเป็น Framework แบบ open source ก็อย่างที่บอกไปที่ว่ามันเป็นแบบ open-source Developer ที่ไหนก็สามารถ แก้ไขงานใน Frame work ที่ GitHub ได้แบบ real-time ดังนั้น Developer จึงมีส่วนช่วยพัฒนาให้ Flutter นั้นดีขึ้น

ในด้านของ features นั้น หนึ่งใน feature ที่น่าสนใจที่สุดเลยก็คือ ปุ่ม Hot reload ถ้าคุณกดปุ่มนี้ การเปลี่ยนแปลงทั้งหมดที่เขียนภายใน code จะปรากฏขึ้นที่หน้าจอทันทีที่เราจะสามารถปรับแต่ง ตัว UI ได้อย่างง่ายดาย แต่มันก็มี surprise อย่างหนึ่งที่ไม่เหมือน framework อื่นๆของ google คือ framework ตัวนี้ใช้ Dart ในการพัฒนาแอป

2.2.1 Flutter เวอร์ชันล่าสุดมีอะไร

- **DevTools** เป็นเครื่องมือที่ใช้แก้ bug ได้ดีใน Flutter และ Dart ผู้พัฒนาสามารถวิเคราะห์ layout และ state ในแอปได้ สามารถแก้ปัญหาเกี่ยวกับความจำ ใน Flutter app ได้ หรือว่าปัญหา command line ใน Dart และแก้ไข bug อื่นๆอีกมากมาย
- **Widget ใหม่** google ให้ความสนใจกับ widget อย่างมาก เพื่อให้ผู้ใช้ ใช้งานได้อย่างมีประสิทธิภาพ Flutter เวอร์ชันล่าสุด ได้สร้าง Slider ตัวใหม่มา เพื่อสร้าง range ที่ดูดี ตัวอย่างเช่นเมื่อผู้ใช้ต้องการจะเปลี่ยนช่วงราคาในแอป mCommerce พวกเขาจึงต้องใช้ widget ที่ดึงดูดความสนใจได้
- **Plugin** ที่สามารถซื้อในแอปได้เลย ตอนนี้ มีการเพิ่ม plugin ใหม่ใน Flutter ที่สามารถใช้ซื้อสิ่งต่างๆได้ใน แอป
- **รองรับ AndroidX google** ได้มีการ support Android มากขึ้น โดยการนำ Android x เข้ามาใช้ ทำให้การสร้างแอปมีประสิทธิภาพมากขึ้น
- **รองรับ Android bundles** ในเดือนสิงหาคม ปี 2019 แอปทั้งหมดที่สร้างสำหรับ Android pie ต้องรองรับในทั้งเวอร์ชัน 32-bit and 64-bit ใน Google play store

² <https://www.thaiprogrammer.org/2019/11/flutter-framework/>

- **Fonts ใหม่** UI ในเวอร์ชันล่าสุดนั้น มีการออกแบบตัวพิมพ์ใหม่ที่เพิ่มตัวเลขและตัวอักษร ใหม่ๆ หลาย style ขึ้นมา การ Interaction กับ fonts ใน iOS ก็ได้รับการพัฒนาเช่นกัน

2.2.2 ทำไมถึงต้องใช้ Flutter

Flutter สามารถจัดการไฟล์ที่ไม่เป็นระเบียบได้ในโหมด Development การรวบรวม Jit ช่วยให้การ พัฒนาและการแก้ bug นั้นเร็วขึ้น นอกจากนั้น Framework ตัวนี้ยังรองรับ plugins หลากๆ อย่างๆ เช่น Firebase หรือ SQLite Pub.dev จะช่วยหา plugins ที่คุณต้องการ Flutter ยังมี ความคล้ายคลึงกับองค์ประกอบของภาษา ในการเขียน โปรแกรมมาตรฐานอีกด้วย

อีกเหตุผลหนึ่งที่ใช้ Flutter ก็คือ Developer จะสามารถทำงานได้ทั้งใน Web และ Desktop แต่ลองคิดภาพดูว่าทีมของเราจะสร้าง app ที่สามารถรันได้ทั้งใน Android, iOS, Web, MacOS, Linux, และ Windows

2.2.3 การทำงานของ Flutter

แอปที่ใช้ Flutter สร้างสามารถใช้ได้ใน iOS, Android, พร้อมกับ Desktop และใน Website ได้ แต่ tool ที่ใช้ทำมีลักษณะที่แตกต่างจาก UI framework ทั่วไปสำหรับระบบปฏิบัติการทั้งสอง ในความเป็นจริงแล้ว แอปใน Web และ Desktop ยังเร็วเกินไปที่นำมาใช้ ดังนั้น Flutter ก็ยังคงได้รับการพัฒนาต่อไป

Flutter ไม่ได้ compiled โดยตรงไปที่ Android หรือ iOS เลย แอปจะเปิดขึ้นได้ด้วยการ ทำงานร่วมกัน ของ rendering engine สร้างจาก C++ และ ใช้ Flutter Design UI ที่สร้างจาก Dart ไฟล์ทั้งหมดที่ถูกสร้าง ด้วยวิธีนี้จะอยู่ในแอปแต่ละแอป และในแต่ละแอปก็จะมี SDK assemblies สำหรับ platform โดยเฉพาะ

อาจส่งผลต่อขนาดของแอปที่สร้างจาก Flutter โดยที่มันอาจจะกินพื้นที่เยอะกว่าแอปปกติ อย่างไรก็ตามประสิทธิภาพของแอปดังกล่าวนี้ดีกว่าแอปทั่วไป การ render อาจจะทำให้เร็วขึ้นสูง ถึง 120 FPS เลยทีเดียว

2.2.3 ข้อดีของ Flutter

- **Hot Reload Feature** นี้ นอกจากจะช่วยให้ Developer ทำงานเร็วขึ้น มันเป็น เครื่องมือที่ฉลาด ถ้าคุณต้องการ จะเปลี่ยนบางอย่างใน Data ของคุณในการสร้าง app interface คุณสามารถเปลี่ยนแปลงภายใน code ขณะที่ Software กำลังดำเนินการอยู่ การ เปลี่ยนแปลงทุกอย่างก็จะปรากฏให้เห็นในแอปเลย ราวกับว่าคุณกำลังสร้าง website ใน การเปิดใช้งาน เพียงแค่กดปุ่ม Hot reload

- **การพัฒนาที่รวดเร็ว** นอกจากจากปุ่ม Hot reload แล้วยังมี widgets, ตัวควบคุม และ libraries จำนวนมากที่คุณสามารถปรับแต่งได้เองตามที่คุณต้องการ และกระบวนการพัฒนา UI ของมันยังมีความรวดเร็วมากอีกด้วย

- **Screen reader** เป็น Software ชนิดหนึ่งสำหรับ Developer ที่มีปัญหาเกี่ยวกับตา หรือว่าการมองเห็น มันช่วยโดยการบอกด้วยเสียงหรือว่าใช้ signals

- **การ REndeR** ที่รวดเร็ว เนื่องจากมีการ Render ที่เร็ว จึงสามารถสร้าง รูปและ โมเดล 2D ได้เร็วขึ้น

- **Cross-platform** Flutter ทำให้เราสามารถสร้าง UI ที่ใช้ได้ ใน iOS และ Android ด้วยความช่วยเหลือของหลายๆเครื่องมือ อย่างเช่น intelliJ, Android Studio และ Visual Studio Code

- **Open source และฟรี** สิ่งเดียวที่ Developers ควรจะมีคือ ความต้องการในการพัฒนา แอป เพราะ Flutter นั้นไม่มีข้อจำกัดในการใช้ ไม่มีค่าใช้จ่ายเพิ่มเติม ทุกคนสามารถใช้งานได้เลย

- **Themes** Themes และ Flutter UI ที่หลากหลาย ทำให้ Developers สามารถสร้าง ทิมที่ไม่เหมือนกันให้กับ Android และ iOS

2.2.4 ข้อเสียของ Flutter

- **รองรับแค่ใน Mobile** ตอนนี้ Flutter ทำงานได้แค่ในแอปมือถือเท่านั้น ยังไม่มี web browser ที่รองรับ แต่มันก็เป็นเรื่องของเวลา เดี่ยวข้อเสียนี้จะหายไปไม่ช้า แต่ก็ยังต้องรอต่อไป

- **ไม่ค่อยมีคนนิยมใช้ Dart** ไม่เหมือนกับพวก Java หรือ Kotlin ซึ่งเป็นภาษาส่วนใหญ่ที่คนนิยมใช้พัฒนาแอปกัน Dart ยังไม่มีความนิยมเท่าไร นั่นคือเหตุผลที่ Developers ส่วนใหญ่ภาษาใหม่ๆ เพราะมีคนที่ใช้ Java เป็นประจำอยู่แล้ว และ UI Frame ที่มีอยู่หลายหลายก็เพียงพอสำหรับพวกเขาอยู่แล้ว อย่างไรก็ตาม Dart นั้นมีข้อดีมากกว่า Java ในการพัฒนาแอป แต่ก็ยังน้อยกว่า Kotlin ในขณะเดียวกัน ฟังก์ชัน ของ Flutter เช่น Ho Reload จะไม่ทำงานถ้าหากไม่มี Dart

- **ไม่รองรับอุปกรณ์ทั้งหมด** ตัวอย่างเช่น แอปที่สร้างไม่สามารถใช้ได้ ใน 32-bit iOS ดังนั้น iOS รุ่นแรกๆทั้งหมดเช่น iPhone 5 ลงไป จึงไม่ใช่กลุ่มที่ตลาดที่เป็นเป้าหมายในการสร้างแอปขึ้นมา

● มี **libraries** ที่จำกัด เนื่องจาก Flutter เป็น Framework ที่ยังใหม่อยู่ จึงยังไม่ค่อยมี libraries ที่เป็นประโยชน์อย่างเช่นภาษาอื่นๆ นั่นจึงเป็นเหตุผลว่าทำไม Developers จำเป็นต้องสร้าง libraries ส่วนใหญ่ด้วยตัวเอง มันก็เลยต้องเสียเวลาไปมากกว่าเดิม

2.3 Dart³

Dart เป็นภาษา Programming ตัวหนึ่งซึ่ง เหล่า Developers ที่ Google และ ที่อื่น ๆ ใช้ เพื่อสร้าง applications ที่มีคุณภาพสูง

- Designed by Lars Bak and Kasper Lund
- Developed by Google

First appeared: 2011 หรือเมื่อ 6 ปีที่แล้ว แต่เพิ่งจะ release เมื่อ 2017 นี้เอง หลังจากนั้นก็เริ่มมีการ release version อื่น ๆ มากขึ้น

2.3.1 Productive

Dart มี syntax ที่ค่อนข้างเคลียร์ไม่ซับซ้อน เหมาะสำหรับพัฒนา web app, android และ iOS

2.3.2 Fast

Dart ช่วยเพิ่มประสิทธิภาพในการรวบรวมข้อมูล เพื่อให้ได้ประสิทธิภาพสูง และ คุณ สามารถเริ่มต้นได้อย่างรวดเร็วทั้งงานบนมือถือ และ บนเว็บไซต์

2.3.3 Portable

Dart compiles ARM และ x86 code ทำให้ source ที่เขียนจาก Dart สามารถรันแบบ natively บน android, iOS และ อื่น ๆ ได้ สำหรับ web apps Dart จะเปลี่ยนเป็น JS ให้ด้วย

2.3.4 Approachable

Dart language มีไวยากรณ์ คล้าย ๆ กับพวก C++, C# หรือ Java ถ้าคุณพอจะมีความรู้ภาษา ที่กล่าวมาข้างต้นอยู่แล้วล่ะก็ คุณจะใช้เวลาในการเรียนรู้ Dart ไม่นาน

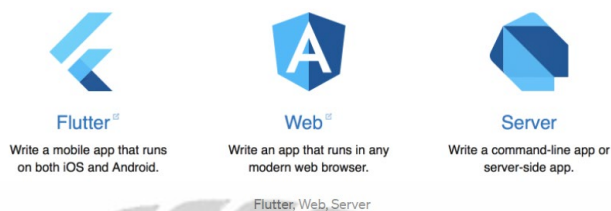
2.3.5 Platform

Dart มีอยู่ด้วยกัน 3 Platform ใหญ่ ๆ

- Flutter สำหรับเขียน mobile app ใช้งานได้ทั้ง iOS และ Android
- Web สำหรับเขียน web application

³ <https://iamgique.medium.com/dart-flutter-is-f61778caa6a0>

- Server สำหรับเขียนฝั่ง Server side



รูปที่ 2.1 Dart Platform

2.4 Firebase⁴

Firebase คือ Project ที่ถูกออกแบบมาให้เป็น API และ Cloud Storage สำหรับพัฒนา Realtime Application รองรับหลาย Platform ทั้ง iOS App, Android App, Web App

Firebase ถูกสร้างขึ้นจากคุณสมบัติเสริมว่านักพัฒนาสามารถผสมและจับคู่เพื่อให้พอดีกับความต้องการของตน บริษัท ก่อตั้งขึ้นในปี 2011 โดยแอนดรูว์และเจมส์ เทมปลิน สินค้าเริ่มต้น Firebase ป็นฐานข้อมูลเรียลไทม์ซึ่งมี API ที่ช่วยให้นักพัฒนาในการจัดเก็บและชิงค์ข้อมูล โดย Google Firebase 2.0 ถูกเปิดตัวขึ้นได้ซื้อกิจการ Firebase และมีการพัฒนาให้สามารถ จากบริการ backend เก็บข้อมูลอย่างเดียว มาเป็น แพลตฟอร์ม ครบวงจรสำหรับนักพัฒนาแอป รองรับบริการแทบทุกอย่างที่นักพัฒนาแอปต้องใช้งาน

2.4.1 บริการของ Firebase

- **Firebase Analytics** บริการวิเคราะห์ข้อมูล ดึงเทคโนโลยีมาจาก Google Analytics แคมเปญเปิดให้ใช้ฟรีแบบไม่จำกัดปริมาณข้อมูลใดๆ
- **Firebase Cloud Messaging (FCM)** ระบบส่งข้อความแจ้งเตือน ใช้งานฟรีไม่จำกัดปริมาณข้อความ
- **Firebase Storage** บริการพื้นที่เก็บข้อมูล เอาไว้เก็บภาพ วิดีโอ หรือไฟล์ขนาดใหญ่จากแอปของผู้ใช้ สร้างอยู่บน Google Cloud Storage
- **Firebase Remote Config** ตัวช่วยอัปเดตคอนฟิกของแอป สำหรับปรับแต่งค่าต่างๆ ในแอปจากระยะไกล (เช่น เกมที่อยากปรับสมดุลของเกมตลอดเวลา) สามารถใช้ร่วมกับ Firebase Analytics เพื่อกำหนดผู้ใช้งานแยกเป็นกลุ่มๆ ได้

⁴ <https://www.mindphp.com/คู่มือ/73-คืออะไร/3921-what-is-firebase-backend.html>

- **Firebase Crash Reporting** ตัวรายงานการแครชของแอป รองรับทั้ง iOS และ Android
- **Firebase Test Lab for Android** บริการทดสอบแอปบนฮาร์ดแวร์จริง
- **Firebase Notifications** เป็นคอนโซลสำหรับนักพัฒนา เพื่อยิงข้อความผ่าน FCM ไปยังผู้ใช้ สำหรับโปรโมทหรือกระตุ้นให้ผู้ใช้กลับมาเปิดแอปของเรา (เช่น แจกของในเกม)
- **Firebase Dynamic Links** บริการ URL กลางที่สามารถชี้ทางไปยังเพจต่างๆ แปรผันตามอุปกรณ์หรือคุณสมบัติของผู้ใช้ (เช่น แต่ละประเทศกดลิงก์เดียวกัน เข้าคนละเพจกัน)
- **Firebase Invites** ระบบเชิญเพื่อนมาใช้แอป มีฟีเจอร์ referral คนชวนได้สิทธิประโยชน์
- **Firebase App Indexing** เปลี่ยนชื่อมาจาก Google App Indexing ที่ช่วยให้ Google Search ค้นเจอเนื้อหาภายในแอป

บทที่ 3

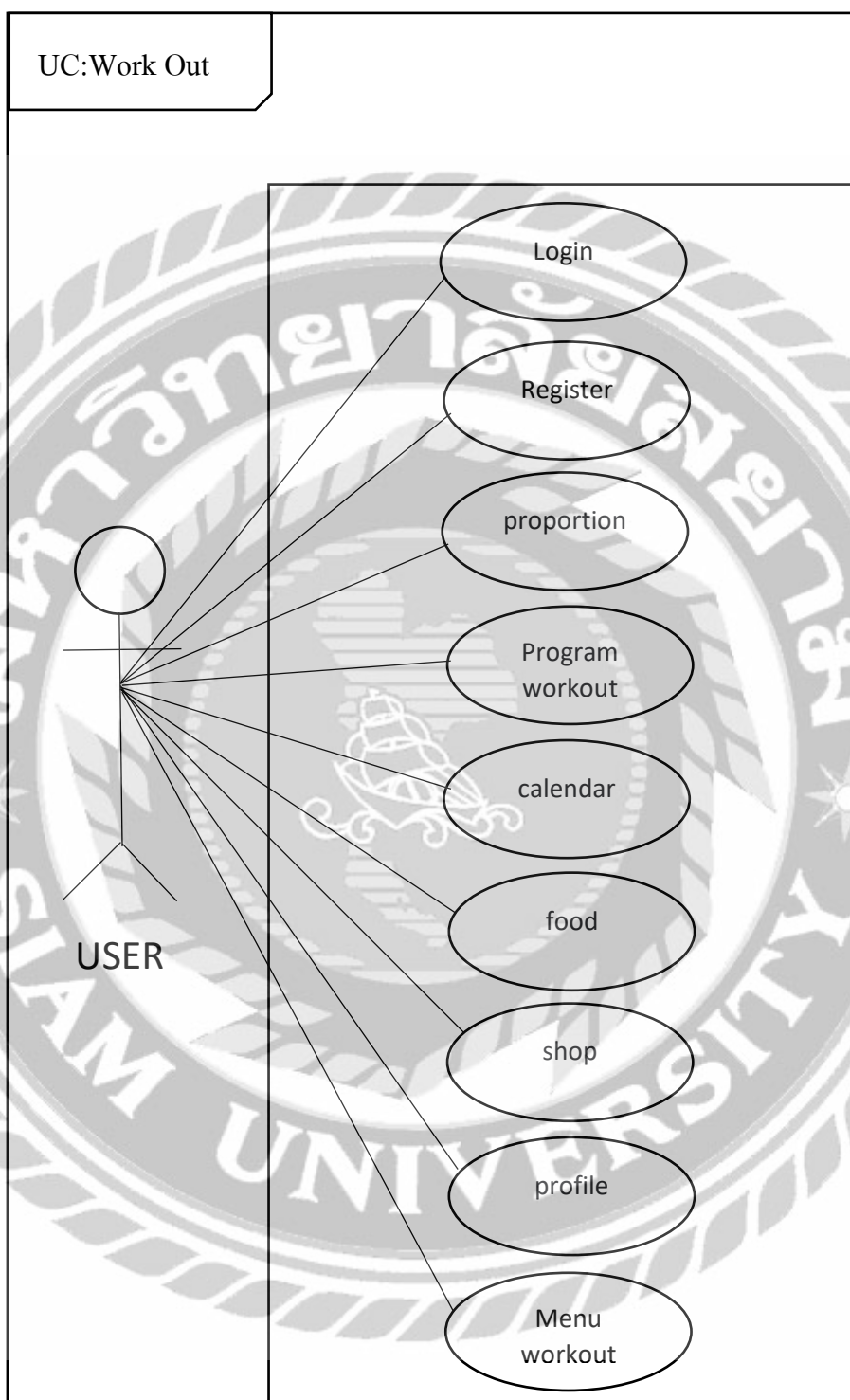
การวิเคราะห์และออกแบบระบบ

3.1 รายละเอียดของปริญญานิพนธ์

แอปพลิเคชันเวิร์คเอาท์ เป็นแอปพลิเคชันช่วยออกกำลังกายสำหรับคนที่คิดจะออกกำลังกายสร้างกล้ามเนื้อ โคนสามารถใช้แอปพลิเคชัน workout ช่วยสอนท่าออกกำลังกายอย่างถูกต้อง และทำตามตารางการออกกำลังกายได้อย่างเหมาะสม โดยผู้ใช้งานสามารถลงทะเบียนใช้งานแอปพลิเคชันได้ โดยผู้ใช้งานจะต้องกรอกสัดส่วนก่อนเริ่มออกกำลังกาย และกรอกสัดส่วนที่ต้องการให้เพิ่มขึ้น โดยผู้ใช้งานสามารถดูตารางการออกกำลังกาย อาหารที่ควรกินสำหรับสร้างกล้ามเนื้อ และหมวดการออกกำลังกายแต่ละส่วนได้ และหลังออกกำลังกายไปเรียบร้อยแล้วผู้ใช้งานสามารถไปกรอกสัดส่วนอัพเดทได้ เพื่อเทียบสัดส่วนก่อนและหลังออกกำลังกายว่ามีการเป็นปรังหรือไม่

ดังนั้นการพัฒนาระบบ จะประกอบด้วยกรอกราคาสัดส่วนของผู้ใช้งาน โดยผู้ใช้งานจะต้องลงทะเบียนเข้าใช้งานก่อนโดยใช้เพียงอีเมล (Email Address) และรหัสผ่าน (Password) สำหรับครั้งแรกของการเข้าใช้งานแอปพลิเคชันผู้ใช้งานจะต้องทำการกรอกข้อมูลสัดส่วนร่างกายก่อนออกกำลังกาย และกรอกสัดส่วนที่ต้องการ หลังจากการออกกำลังกายไปเรียบร้อยแล้วผู้ใช้งานสามารถกรอกสัดส่วน หลัง ออกกำลังกายได้ แอปพลิเคชันพัฒนาโดยใช้ Flutter ในการสร้าง Native Interface สำหรับแอนดรอยด์และไอโอเอส ข้อมูลของผู้ใช้งานจะถูกจัดเก็บลงใน Firebase และเขียนชุดคำสั่งด้วยภาษา Dart เครื่องมือที่ใช้คือ Visual Studio Code

3.2 แสดงฟังก์ชันการทำงานของระบบด้วย Use Case Diagram



รูปที่ 3.1 Use Case Diagram ของแอปพลิเคชัน Workout

3.3 คำอธิบายรายละเอียดของยูสเคส Use Case Description

ตารางที่ 3.1 แสดงรายละเอียดของ Use case Login

Use Case Name	LOGIN
Use Case ID	UC1
Brief Description	สำหรับยืนยันตัวตน โดยใช้อีเมลและรหัสผ่าน เพื่อเข้าใช้งานแอปพลิเคชัน
Primary Actors	User
Secondary Actors	-
Preconditions	จะต้องทำการลงทะเบียนก่อนเข้าใช้งานระบบ
Main Flow	• ยูสเคสจะเริ่มขึ้นเมื่อผู้ใช้เข้าสู่ระบบแอปพลิเคชัน • ระบบแสดง Field สำหรับป้อนข้อมูลเพื่อเข้าสู่ระบบ โดยใช้อีเมลและรหัสผ่าน • ถ้าผู้ใช้ป้อนข้อมูล ชื่อผู้ใช้ และ รหัสผ่าน ถูกต้อง 2.1.1 ระบบแสดงหน้า Feed หลัก • ถ้าผู้ใช้ป้อนข้อมูล อีเมล และ รหัสผ่าน ไม่ถูกต้อง 3.2.1 ระบบแสดง ข้อความ ไม่สามารถเข้าสู่ระบบได้ 3.2.2 ผู้ใช้ต้องป้อนข้อมูล เพื่อเข้าสู่ระบบอีกครั้ง
Post Condition	เมื่อเข้าสู่ระบบสำเร็จ ระบบแสดงหน้าหน้า Feed หลัก
Alternative Flows	-

ตารางที่ 3.2 แสดงรายละเอียดของ Use case Register

Use Case Name	REGISTER
Use Case ID	UC2
Brief Description	สำหรับลงทะเบียน เข้าใช้งานแอปพลิเคชัน
Primary Actors	User
Secondary Actors	-
Preconditions	-
Main Flow	• ยูสเคสจะเริ่มขึ้นเมื่อผู้ใช้ทำการลงทะเบียนข้อมูลผู้ใช้ • ผู้ใช้งานป้อนข้อมูลสำหรับลงทะเบียนผู้ใช้งาน ดังนี้ อีเมล , รหัสผ่าน • ถ้าผู้ใช้งานป้อนข้อมูลครบตามที่กำหนด • ระบบแสดงหน้า proportion เพื่อให้ผู้ใช้ทำการกรอกสัดส่วนร่างกาย • ถ้าผู้ใช้งานป้อนข้อมูลไม่ถูกต้องตามที่กำหนด 2.2.1 ระบบแสดงข้อความให้ผู้ใช้งานป้อนข้อมูลให้ถูกต้องตามที่กำหนด • ผู้ใช้งานทำการลงทะเบียนข้อมูลผู้ใช้งาน จัดเก็บข้อมูลใน Firebase ที่มีชื่อว่า UserData โดยจัดเก็บข้อมูล email, proportion
Post Condition	-
Alternative Flows	-

ตารางที่ 3.3 แสดงรายละเอียดของ Use case Proportion

Use Case Name	PROPORTION
Use Case ID	UC3
Brief Description	สำหรับกรอกข้อมูลสัดส่วน
Primary Actors	User
Secondary Actors	-
Preconditions	-
Main Flow	• ยูสเคสจะเริ่มเข้าไปหน้าหลักได้หลังกรอกข้อมูลสัดส่วนแล้ว • ผู้ใช้งานป้อนข้อมูลสัดส่วน ดังนี้ ชื่อ, ส่วนสูง, น้ำหนัก, อายุ, รอบหน้าอก, รอบแขน, รอบเอว, รอบขา • ถ้าผู้ใช้งานป้อนข้อมูลครบตามที่กำหนด • ระบบแสดงหน้า workout เพื่อให้ผู้ใช้ทำการเลือกเมนูอื่นๆต่อตามต้องการ • ถ้าผู้ใช้ป้อนข้อมูลไม่ถูกต้องตามที่กำหนด 2.2.1 ระบบแสดงข้อความให้ผู้ใช้งานป้อนข้อมูลให้ถูกต้องตามที่กำหนด ผู้ใช้งานทำการลงทะเบียนข้อมูลผู้ใช้งาน จัดเก็บข้อมูลใน Firebase ที่มีชื่อว่า UserData โดยจัดเก็บข้อมูล name, height, weight, age, chest, aroundthear, aroundtheleg, waistline
Post Condition	-
Alternative Flows	-

ตารางที่ 3.4 แสดงรายละเอียดของ Use case Program workout

Use Case Name	PROGRAM WORKOUT
Use Case ID	UC4
Brief Description	สำหรับดูตารางออกกำลังกาย
Primary Actors	User
Secondary Actors	-
Preconditions	-
Main Flow	ยูสเคสจะเข้าไปหน้า program workout เพื่อดูตารางการออกกำลังกาย
Post Condition	-
Alternative Flows	-

ตารางที่ 3.5 แสดงรายละเอียดของ Use case calendar

Use Case Name	CALENDAR
Use Case ID	UC5
Brief Description	สำหรับบันทึกกิจกรรมที่ต้องการ
Primary Actors	User
Secondary Actors	-
Preconditions	-
Main Flow	ยูสเคสจะเข้าไปหน้า calendar เพื่อบันทึกกิจกรรมต่างๆ
Post Condition	-
Alternative Flows	-

ตารางที่ 3.6 แสดงรายละเอียดของ Use case food

Use Case Name	FOOD
Use Case ID	UC6
Brief Description	สำหรับดูอาหารแนะนำที่ควรกิน
Primary Actors	User
Secondary Actors	-
Preconditions	-
Main Flow	ยูสเคสจะเข้าไปหน้า food เพื่อดูอาหารที่ควรกิน
Post Condition	-
Alternative Flows	-

ตารางที่ 3.7 แสดงรายละเอียดของ Use case shop

Use Case Name	SHOP
Use Case ID	UC7
Brief Description	สำหรับเลือกซื้อสินค้า
Primary Actors	User
Secondary Actors	-
Preconditions	-
Main Flow	• ยูสเคสจะเข้าไปหน้า shop เพื่อเลือกซื้อสินค้า • ผู้ใช้งานเลือกซื้อสินค้า • กดซื้อสินค้า • ระบบจะทำการหักคะแนนออกตามราคาสินค้า • ระบบจะทำการส่งรหัสสินค้าให้ผู้ใช้
Post Condition	-
Alternative Flows	-

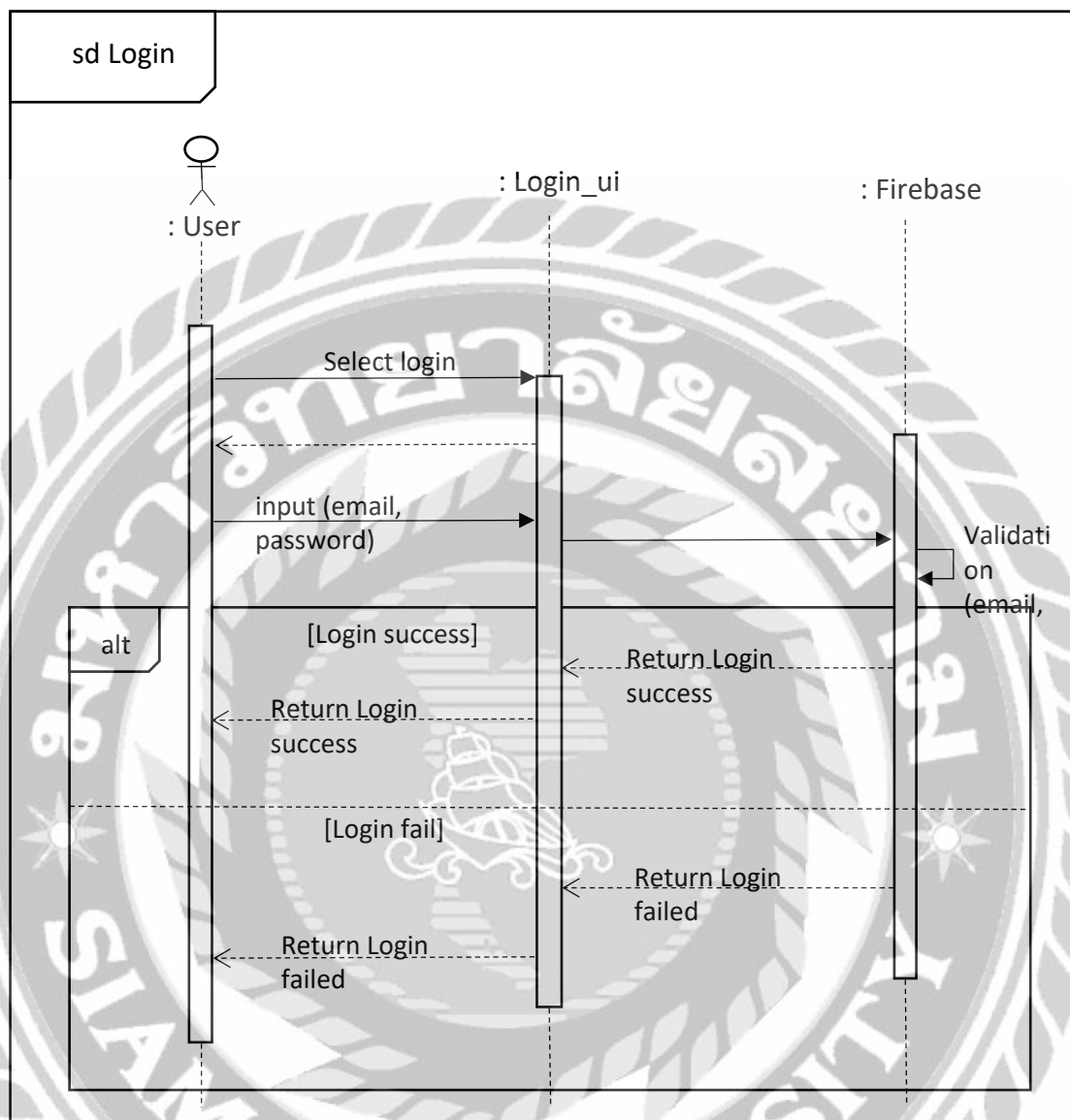
ตารางที่ 3.8 แสดงรายละเอียดของ Use case profile

Use Case Name	PROFILE
Use Case ID	UC8
Brief Description	สำหรับกรอกสัดส่วนหลังออกกำลังกาย
Primary Actors	User
Secondary Actors	-
Preconditions	-
Main Flow	• ยูสเคสจะเข้าไปหน้า profile เพื่อกรอกค่าสัดส่วนหลังออกกำลังกาย • ผู้ใช้งานป้อนข้อมูลสัดส่วน ดังนี้ รอบอก, รอบแขน, รอบเอว, รอบขา • ถ้าผู้ใช้งานป้อนข้อมูลสัดส่วนแล้วกดปุ่ม cal • ระบบจะทำการเทียบสัดส่วนก่อนและหลังออกกำลังกาย • ถ้าผู้ใช้งานป้อนข้อมูลสัดส่วนแล้วกดปุ่ม edit • ระบบจะทำการบันทึกสัดส่วนใหม่
Post Condition	ระบบทำการแก้ไขข้อมูลของผู้ใช้งาน
Alternative Flows	-

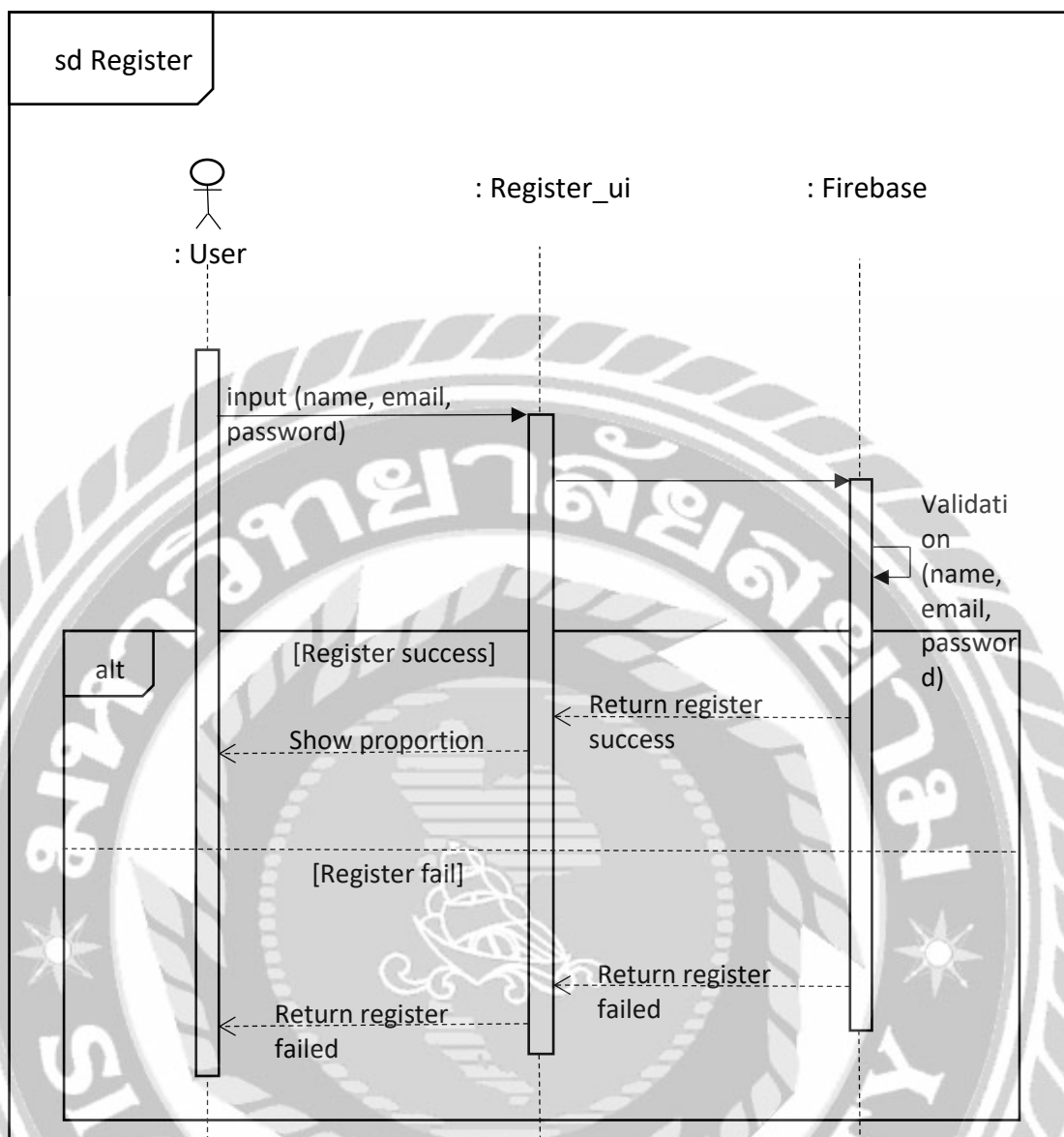
ตารางที่ 3.9 แสดงรายละเอียดของ Use case menu workout

Use Case Name	MENUWORKOUT
Use Case ID	UC9
Brief Description	สำหรับเลือกเมนูออกกำลังกาย
Primary Actors	User
Secondary Actors	-
Preconditions	-
Main Flow	• ยูสเคสจะเข้าไปหน้า menu workout เพื่อเลือกเมนูออกกำลังกาย • ผู้ใช้งานทำการเลือกเมนูออกกำลังกาย • ผู้ใช้งานกด start เริ่มออกกำลังกาย • ผู้ใช้งานกด next เพื่อไปทำต่อไป • เมื่อผู้ใช้งานออกกำลังกายครบทุกท่าแล้วกดปุ่ม Return to home จะกลับไปหน้า menu workout
Post Condition	-
Alternative Flows	-

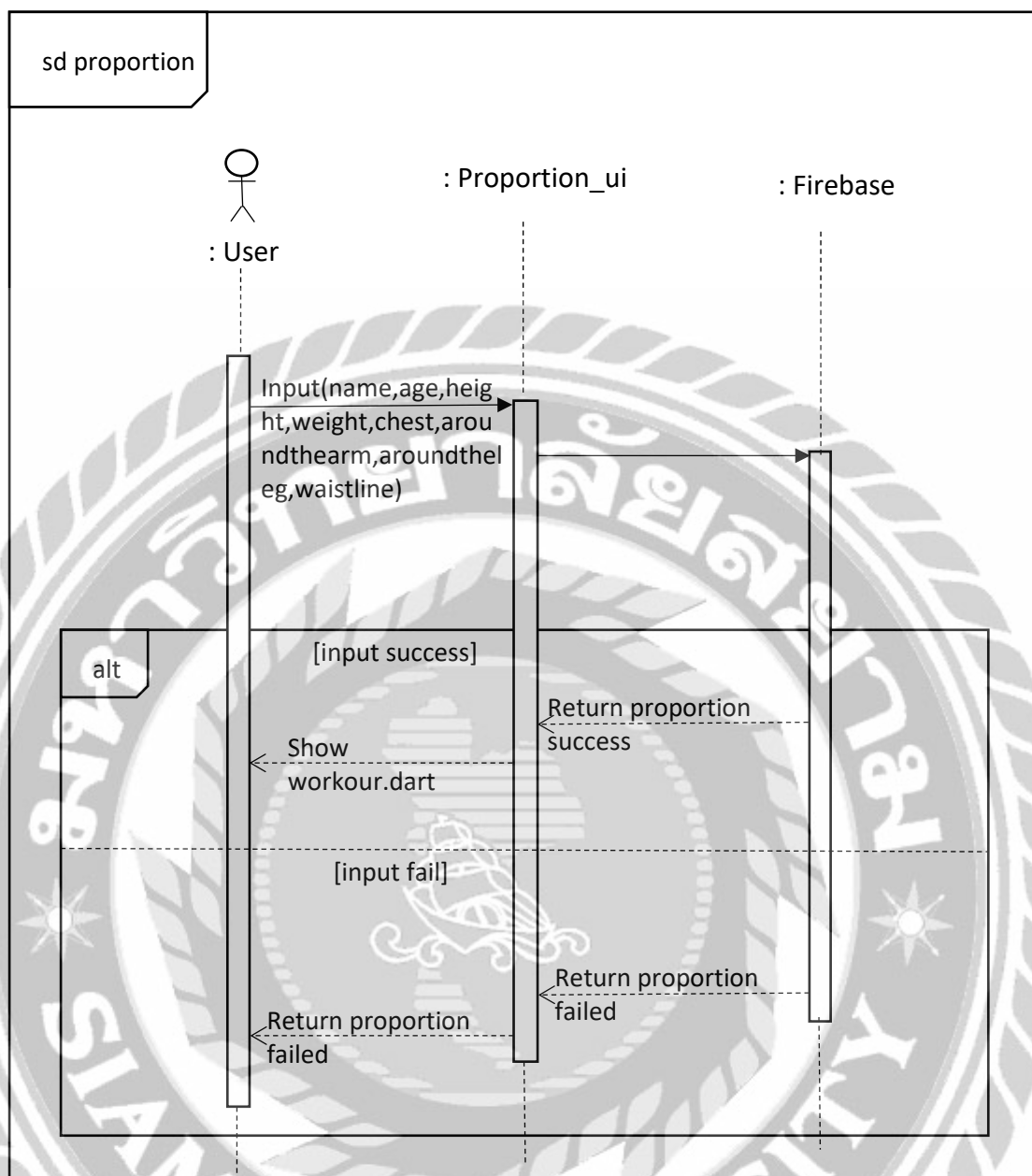
3.4 Sequence Diagram



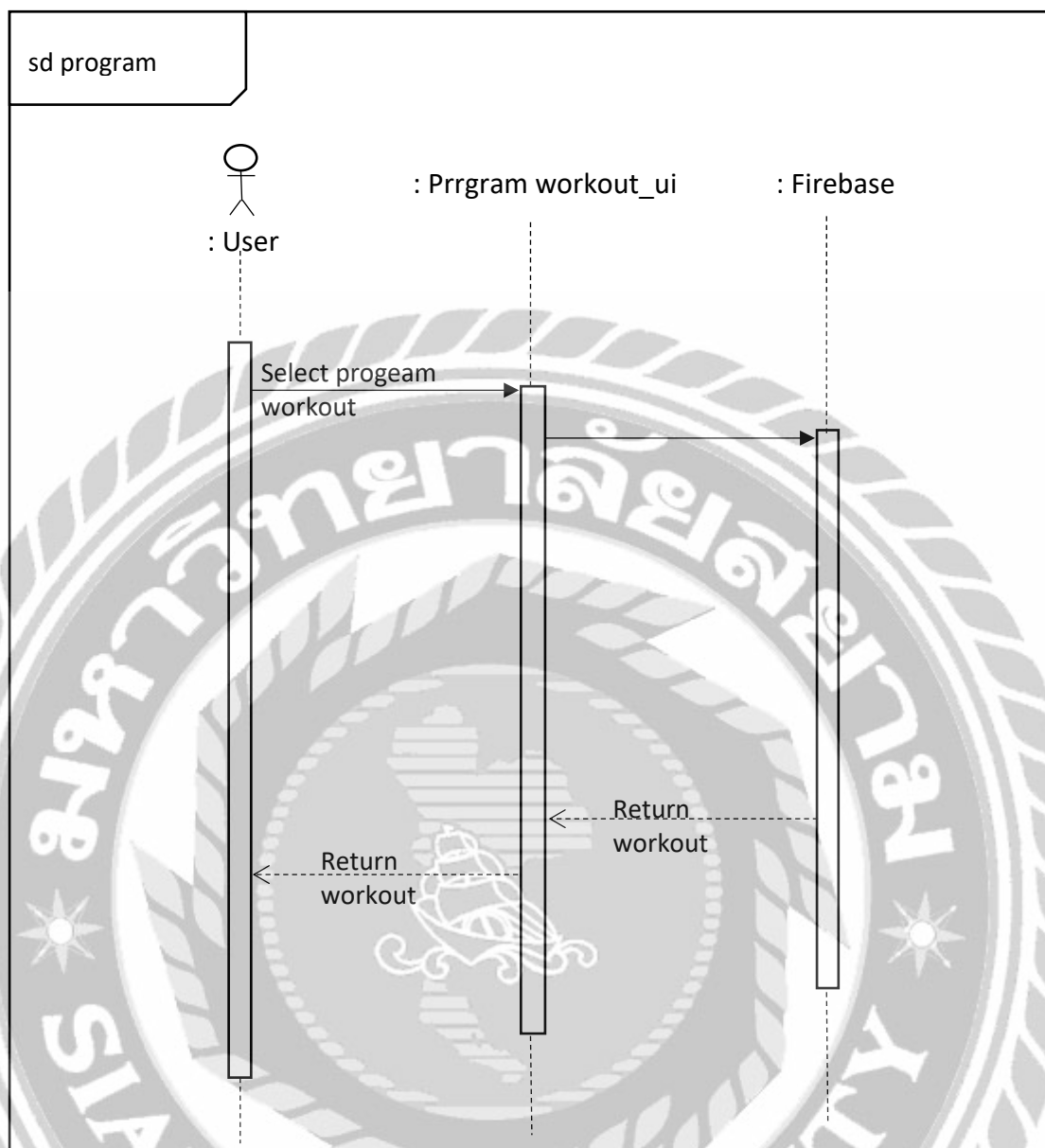
รูปที่ 3.2 Sequence diagram : Login



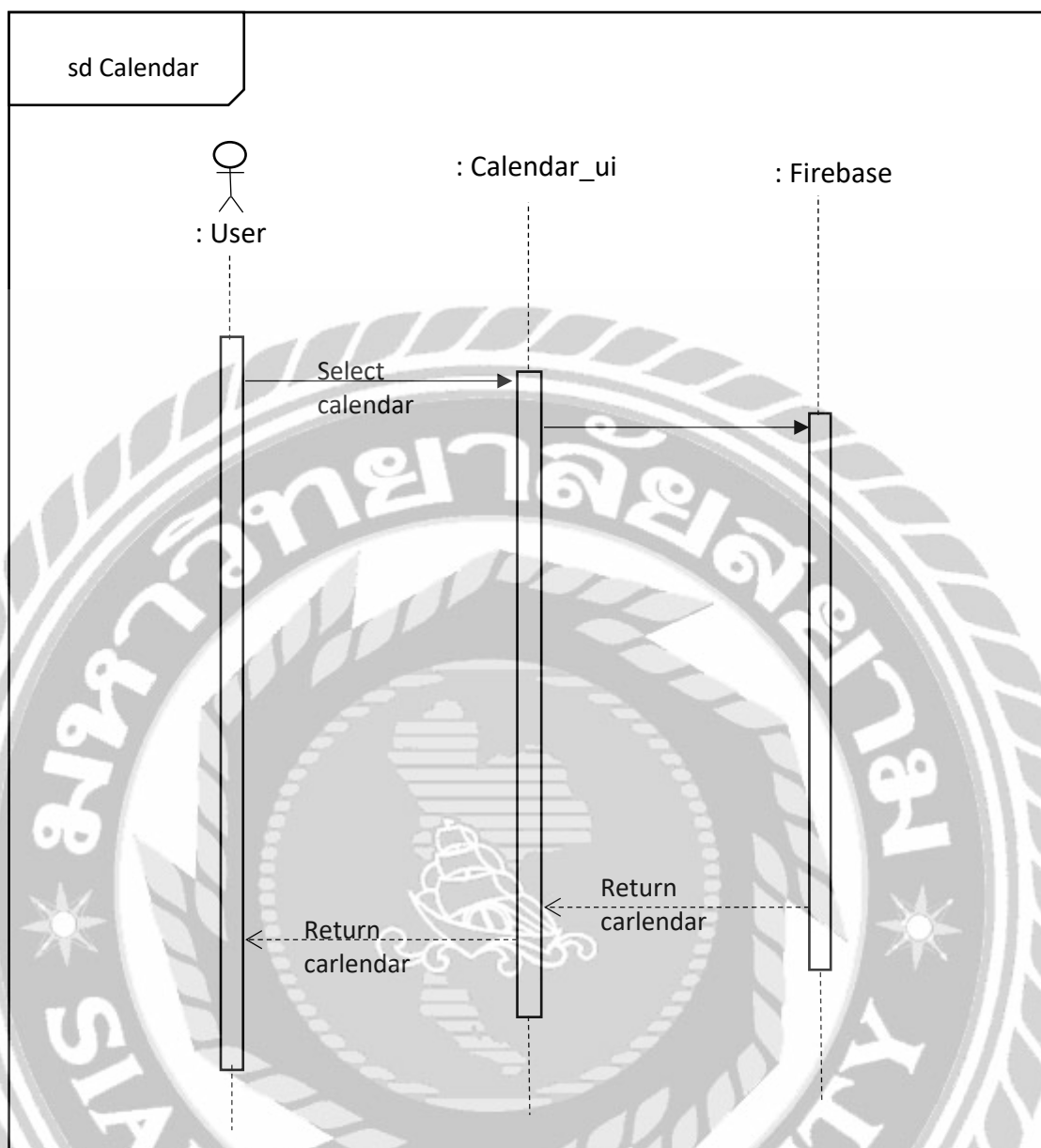
รูปที่ 3.3 Sequence diagram : Register



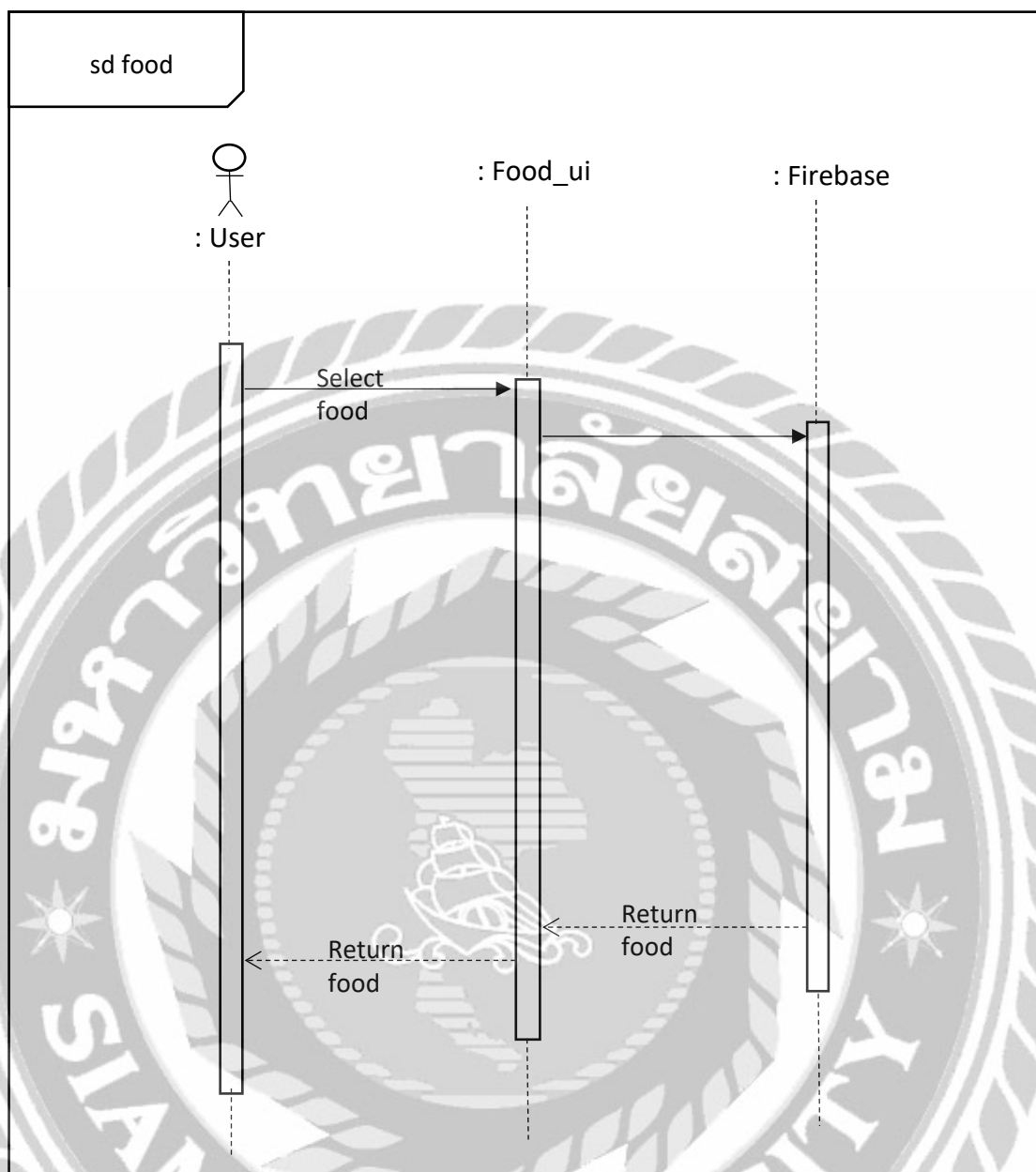
รูปที่ 3.4 Sequence diagram : Proportion



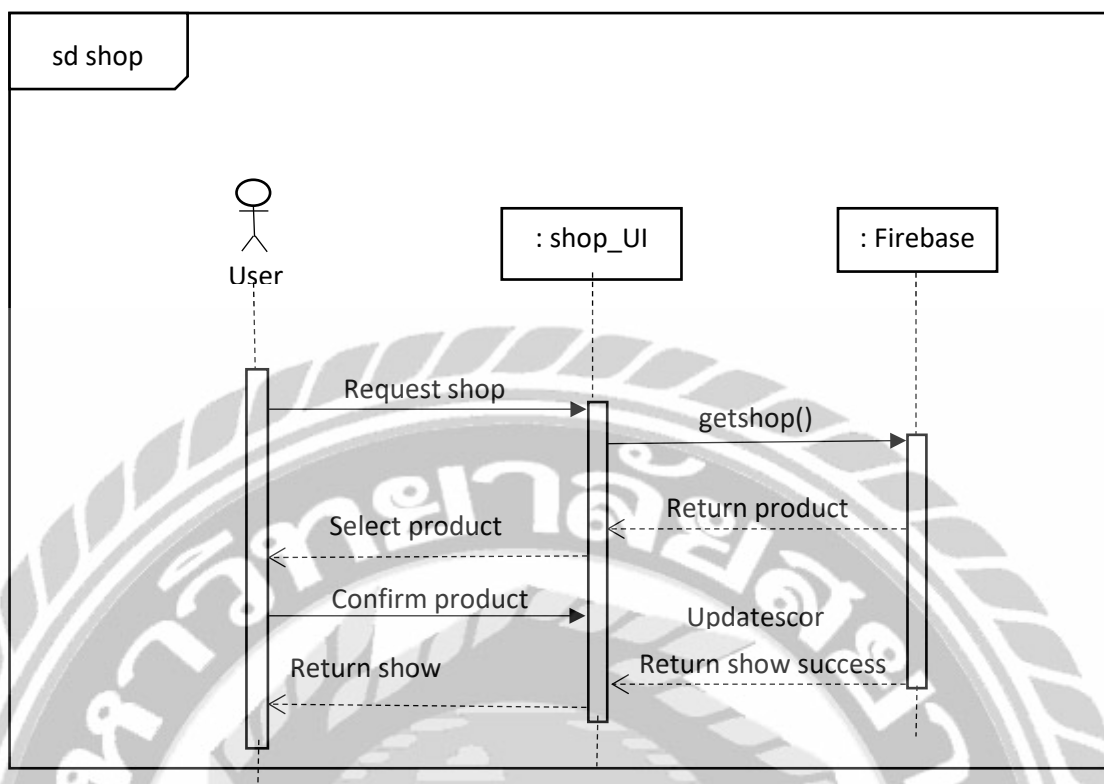
รูปที่ 3.5 Sequence diagram : Program workout



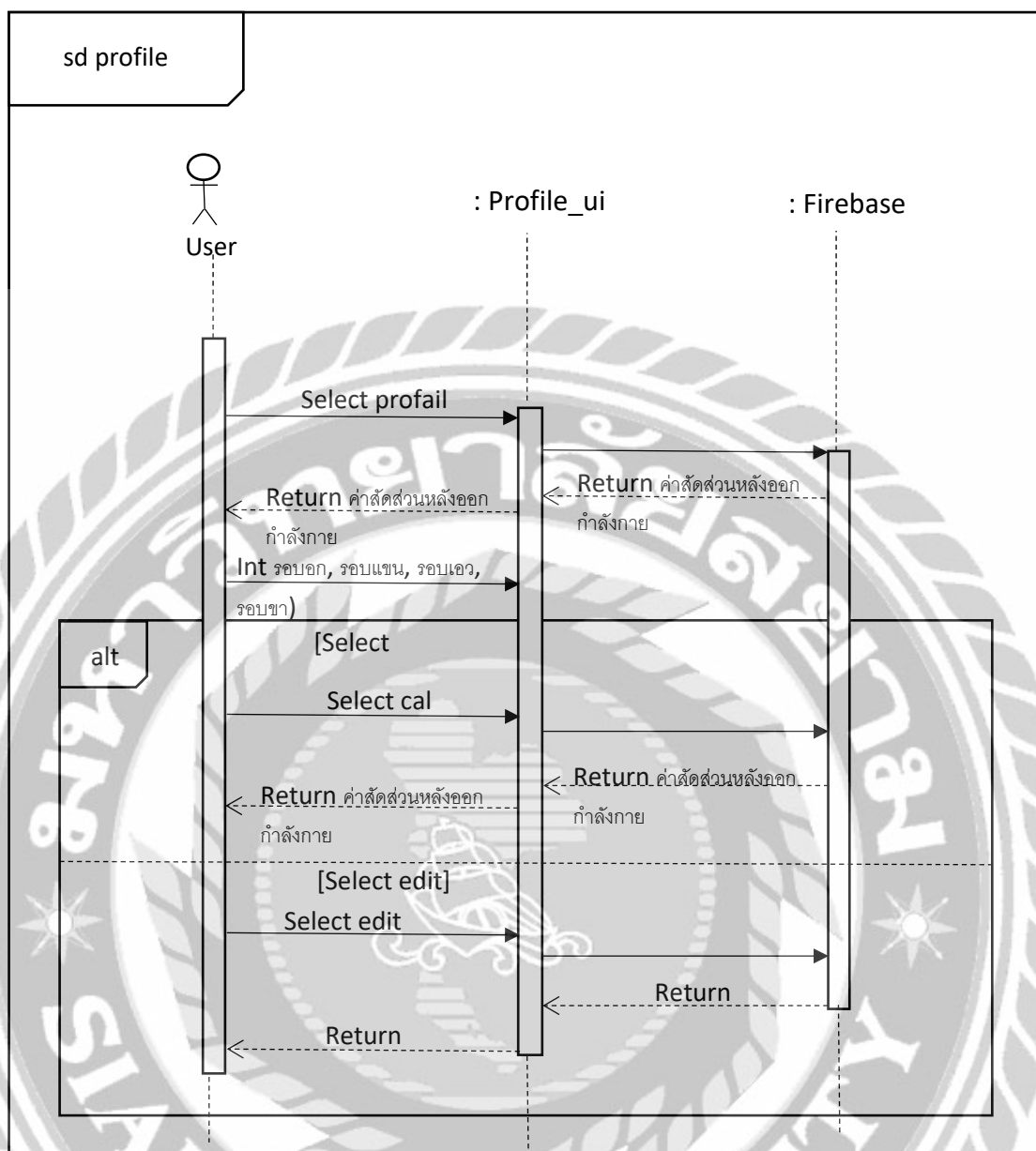
รูปที่ 3.6 Sequence diagram : Calendar



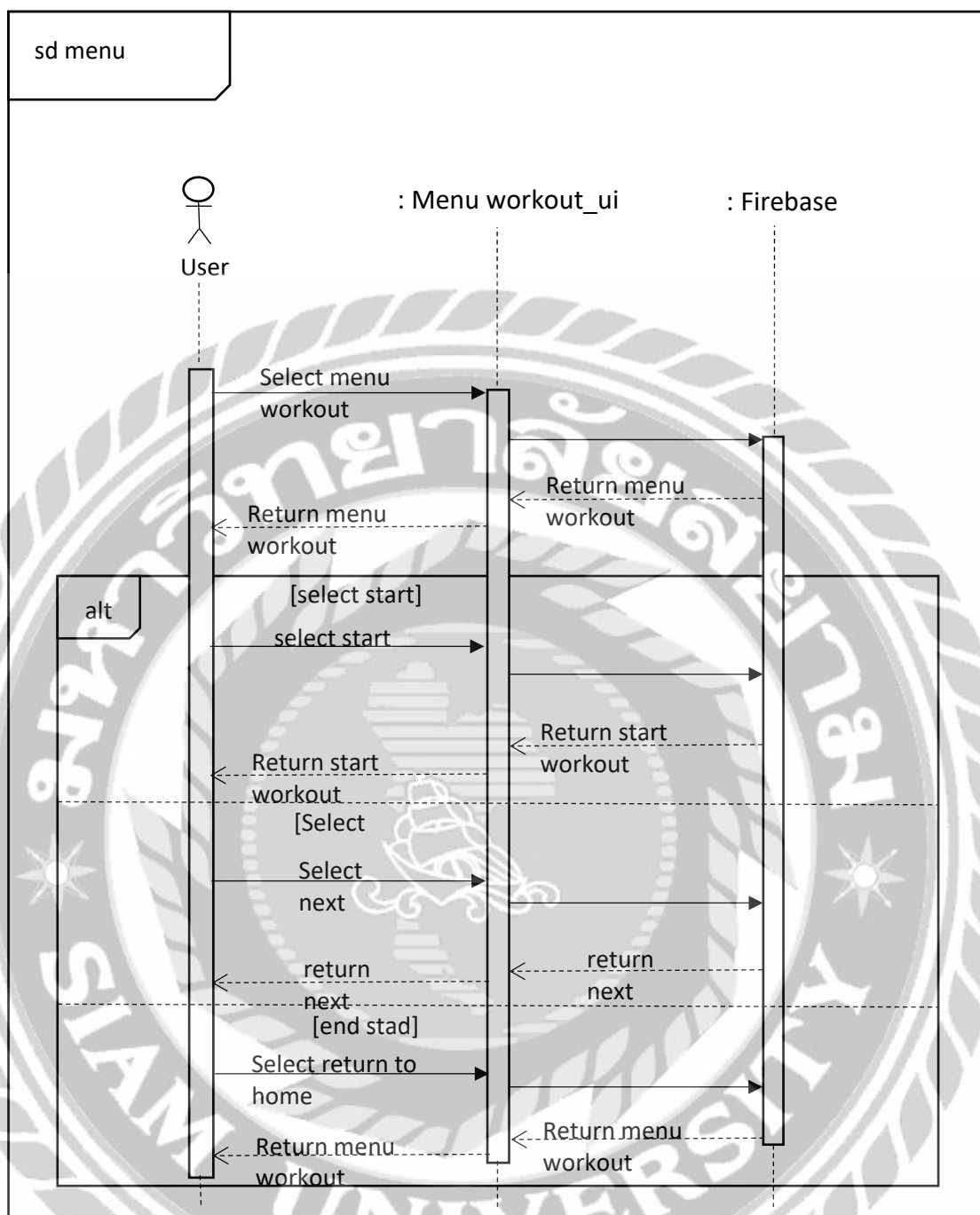
รูปที่ 3.7 Sequence diagram : Food



รูปที่ 3.8 Sequence diagram : Show

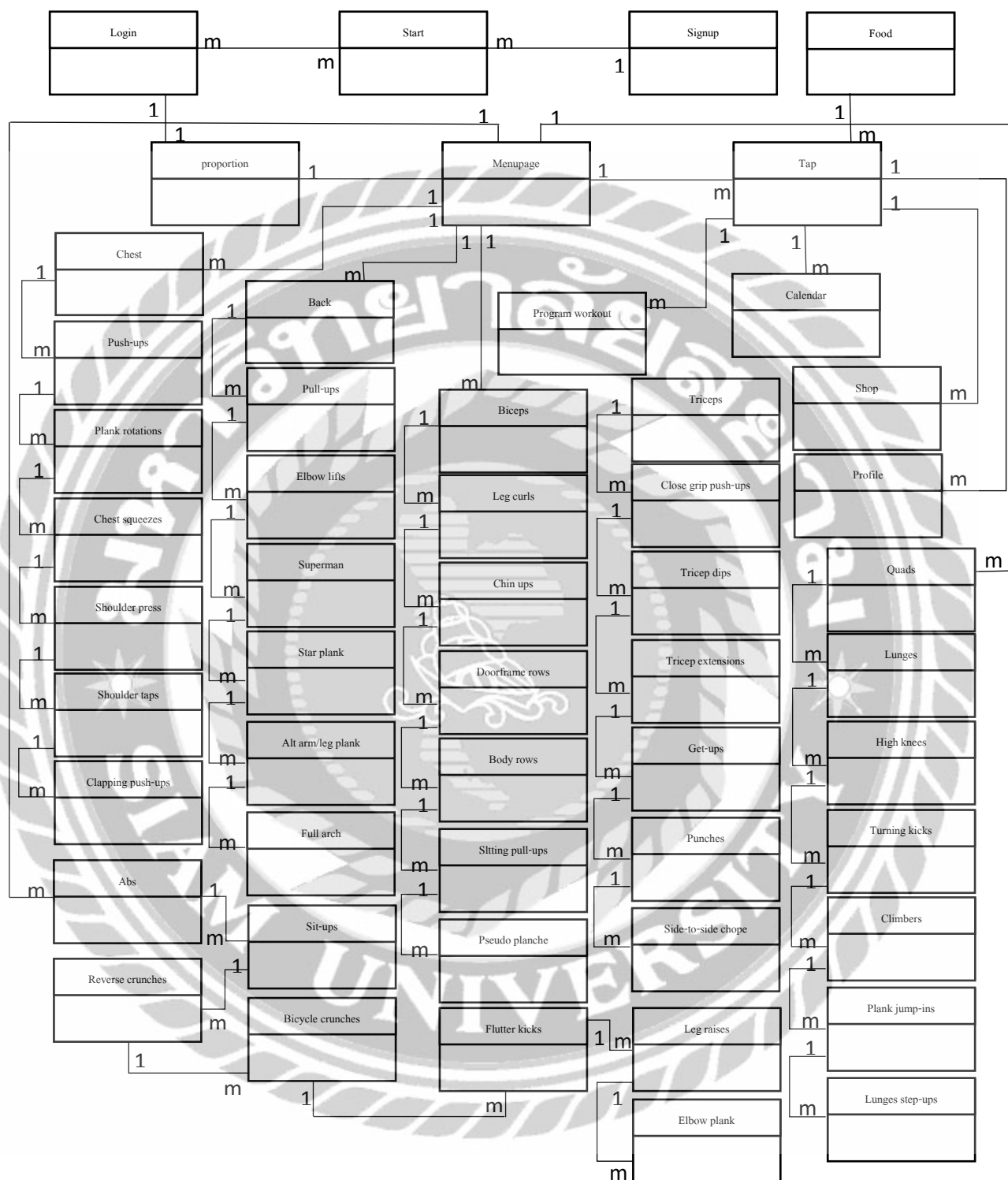


รูปที่ 3.9 Sequence diagram : Profile



รูปที่ 3.10 Sequence diagram : Menu workout

3.5 Class Diagram



รูปที่ 3.11 แผนภาพ Class Diagram workout

Start
-
+ navigateToLogin() + navigateToRegister()

Signup
- name : String - email : String - password : String
+ checkAuthentication() + signUp() + showError()

Login
- email : String - password : String
+ checkAuthentication() + login() + showError() + navigateToSignUp()

Proportion
- name : String - height : String - weight: String - age: String - chest: String - aroundthearm: String - waistline: String

- aroundtheleg: String

+ savespoportion()



Menupage
- time : Timestamp - score1 : var - score2 : var - score3 : var - score4 : var - score5 : var - score6 : var
+ getlikes() + gettoke() + getpoint()

Tap
- user : User - isloggedin : bool
+ checkAuthentification() + getUser() + signOut() + initState()

Program workout
-
+

Calendar
- selectedDay : DateTime - name : String
+ initState() + dispose() + events() + storeNewUser() + create() + showDialog() + fetchEvents() + addEvent() + deleteEvent() + calendar() + eventTitle()
Food
-
+

Shop
- score1 : int - totalscore : int - price : int - getscore1 : int - getscore2 : int - getscore3 : int - getscore4 : int - getscore5 : int - getscore6 : int
+ setinterest() + gettoke() + getscore() + getscore3() + getscore4() + getscore5() + getscore6() + gettotalscore() + initState()

Profile
- dataController : var - chestAF : var - aroundthearmAF : var - waistlineAF : var - aroundthelegAF : var - chest : var - aroundthearm : var - waistline : var - aroundtheleg : var - totalchest : var - totalaroundthearm : var - totalwaistline : var - totalaroundtheleg : var
+ getpro() + initState() + showAlertDialog()
Chest
-
+
Push-ups
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Plank rotations
- timer : Timer - start : int - count : int - totalcount : var
+ startTimer() + dispose()

Chest squeezes
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Shoulder press
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Shoulder taps
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Chapping push-ups
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Back
-
+

Pull-ups
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Elbow lifts
- timer : Timer
- start : int
- count : int
- totalcount : var
+ startTimer()
+ dispose()

Superman
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Strar plank
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Alt arm/leg plank
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Full arch
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Bicipes
-
+

Leg curls
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Chin-ups
- timer : Timer
- start : int
- count : int
- totalcount : var
+ startTimer()
+ dispose()

Doorframe rows
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Body rows
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Sitting pull-ups
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Pseudo planche
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Tricipes
-
+

Close grip push-ups
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Tricipes dips
- timer : Timer - start : int - count : int - totalcount : var
+ startTimer() + dispose()

Triceps extensions
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Get-ups
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Punches
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Side-to-side chops
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Quads
-
+

Lunges
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

High knees
- timer : Timer - start : int - count : int - totalcount : var
+ startTimer() + dispose()

Turning kicks
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Climbers
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Plank jump-ins
- timer : Timer - start : int - count : int
+ startTimer() + dispose()

Lunges step-ups
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Abs
-
+

Sit-ups
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Reverse crunches
- timer : Timer
- start : int
- count : int
- totalcount : var
+ startTimer()
+ dispose()

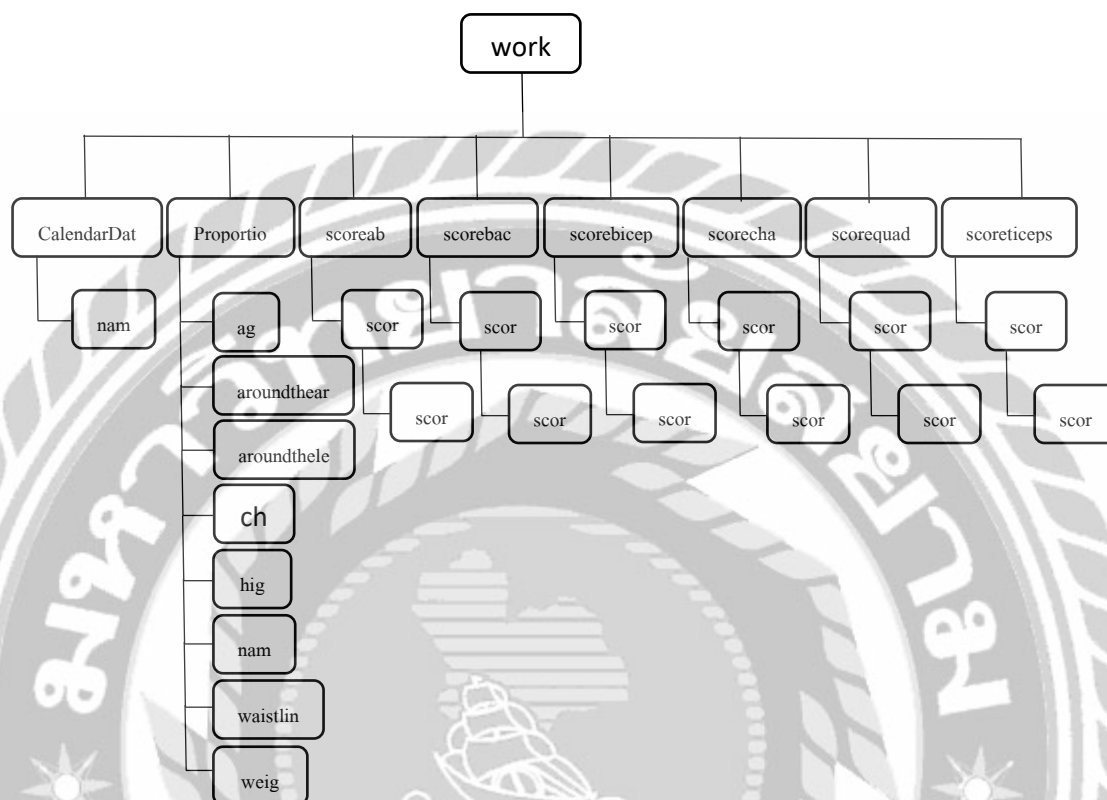
Bicycle crunches
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Flutter kicks
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Leg raises
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

Elbow plank
- timer : Timer
- start : int
- count : int
+ startTimer()
+ dispose()

3.6 โครงสร้างของข้อมูลใน Firebase



รูปที่ 3.12 โครงสร้างของข้อมูล (Firebase)

ตารางที่ 3.10 Calendar Data

name	Description	Data Type
name	บันทึกของ user	String

ตารางที่ 3.11 Proportion

name	Description	Data Type
Age	อายุ	String
Aroundthearm	รอบแขน	String
Aroundtheleg	รอบขา	String
Chest	รอบอก	String
Hight	ส่วนสูง	String
Name	ชื่อ	String
Waistline	รอบเอว	String
Weight	น้ำหนัก	String

ตารางที่ 3.12 scoreabs

name	Description	Data Type
Score	คะแนน	Int
Score	คะแนน	Int

ตารางที่ 3.13 scoreback

name	Description	Data Type
Score	คะแนน	Int
Score	คะแนน	Int

ตารางที่ 3.14 scorebiceps

name	Description	Data Type
Score	คะแนน	Int
Score	คะแนน	Int

ตารางที่ 3.15 scorechat

name	Description	Data Type
Score	คะแนน	Int
Score	คะแนน	Int

ตารางที่ 3.16 scorequads

name	Description	Data Type
Score	คะแนน	Int
Score	คะแนน	Int

ตารางที่ 3.17 scoreticeps

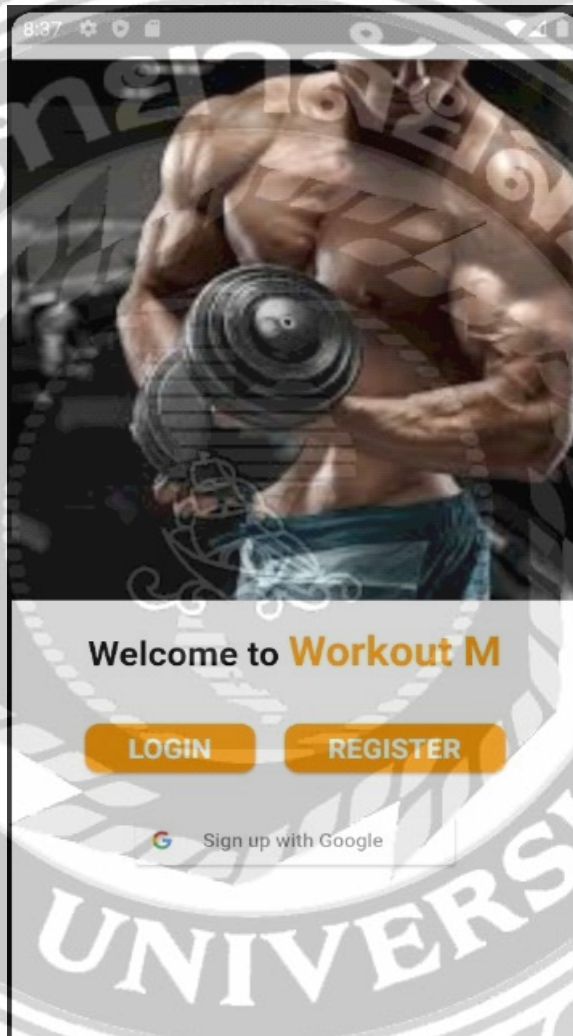
name	Description	Data Type
Score	คะแนน	Int
Score	คะแนน	Int

บทที่ 4

การออกแบบทางกายภาพ

4.1 การออกแบบส่วนติดต่อผู้ใช้งาน (User Interface Design)

แอปพลิเคชัน Workout (M) ออกแบบโดยยึดหลัก UX (User Experience) เป็นหลักโดยคำนึงถึงความพึงพอใจและการใช้งานที่ง่ายต่อผู้ใช้ ด้วยการออกแบบที่เรียบง่าย โดยผู้ใช้งานเป็นได้ทั้งผู้สร้างกิจกรรมและผู้เข้าร่วมกิจกรรม ดังนี้



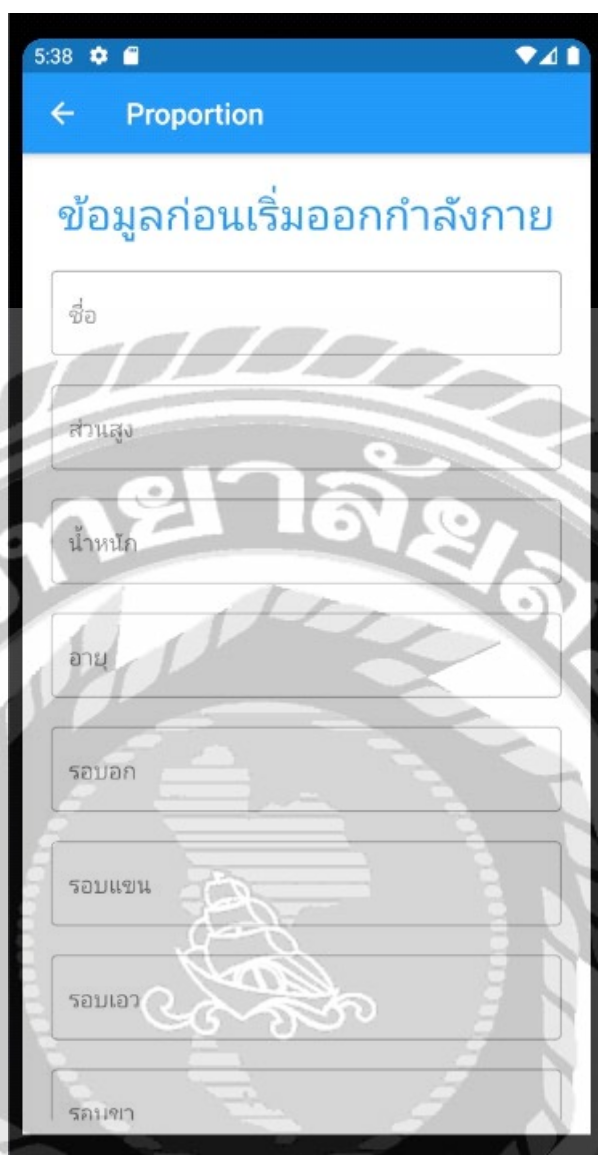
รูปที่ 4.1 หน้า Welcome

จากรูปที่ 4.1 แสดงหน้า Welcome โดยให้ผู้ใช้งานเลือก เข้าสู่ระบบแอปพลิเคชัน หรือลงทะเบียนผู้ใช้ใหม่



รูปที่ 4.2 หน้าเข้าสู่ระบบแอปพลิเคชัน

จากรูปที่ 4.2 แสดงหน้าเข้าสู่ระบบแอปพลิเคชัน โดยให้ผู้ใช้งานกรอกชื่อ อีเมล และรหัสผ่าน เพื่อเข้าสู่ระบบ หรือผู้ใช้งานสามารถเข้าสู่ระบบแอปพลิเคชัน



5:38

← Proportion

ข้อมูลก่อนเริ่มออกกำลังกาย

ชื่อ

ส่วนสูง

น้ำหนัก

อายุ

รอบอก

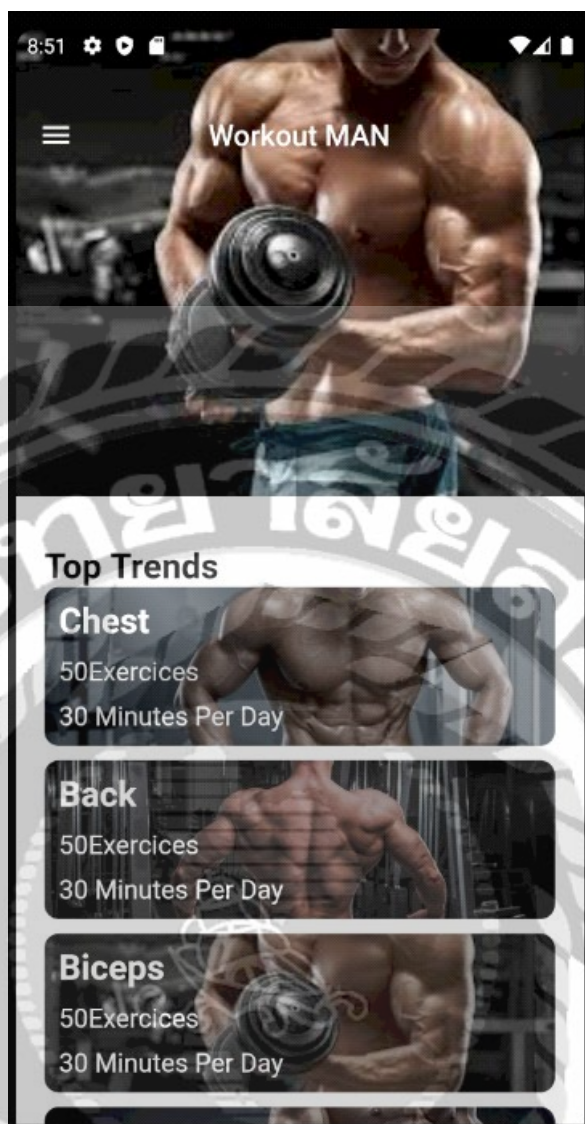
รอบแขน

รอบเอว

รอบขา

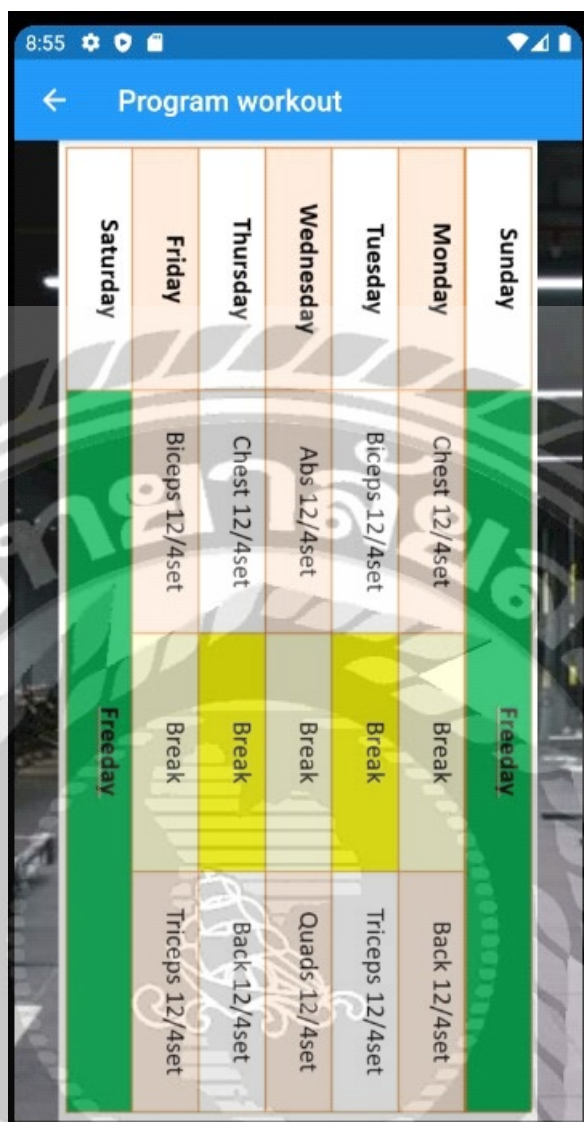
รูปที่ 4.3 แสดงหน้าลงทะเบียนผู้ใช้ใหม่

จากรูปที่ 4.3 แสดงหน้าลงทะเบียนผู้ใช้ใหม่ ที่จะต้องใส่ข้อมูลสัดส่วนร่างกายก่อนออกกำลังกายเมื่อกรอกข้อมูลครบสามารถกดปุ่ม Next Step เพื่อสมัคร ถ้าผู้ใช้งานสมัครสมาชิกสำเร็จระบบจะแสดงหน้า หมดหน้การออกกำลังกายดังรูปที่ 4.4



รูปที่ 4.4 แสดงหน้าหมวดหมู่การออกกำลังกาย

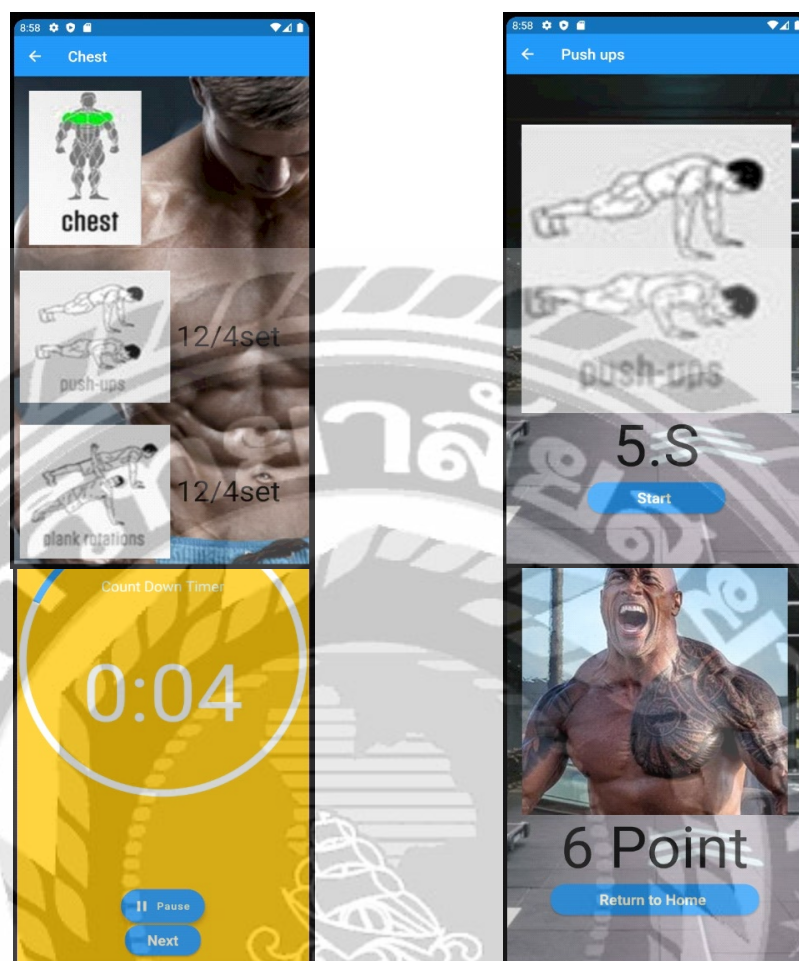
จากรูปที่ 4.4 แสดงหน้าหมวดหมู่การออกกำลังกาย โดยผู้ใช้งานจะต้องเลือกหมวดการออกกำลังกายตามตารางการออกกำลังกายตามภาพที่ 4.5



Day	Workout	Break
Sunday		Freeday
Monday	Chest 12/4set	Break
Tuesday	Biceps 12/4set	Break
Wednesday	Abs 12/4set	Break
Thursday	Chest 12/4set	Break
Friday	Biceps 12/4set	Break
Saturday		Freeday

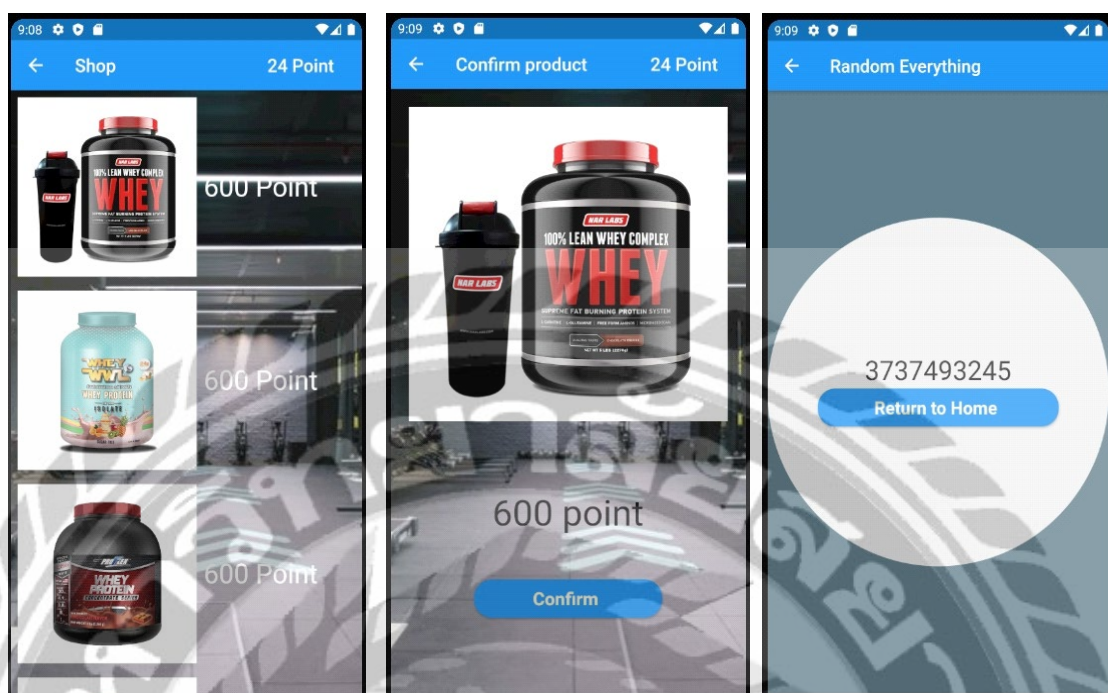
รูปที่ 4.5 แสดงหน้าตารางการออกกำลังกาย

จากรูปที่ 4.5 แสดงหน้าตารางการออกกำลังกาย เมื่อผู้ใช้งานรู้ว่าต้องออกกำลังกายส่วนไหนถึงจะไปเลือกหมวดการออกกำลังกายตามภาพที่ 4.6



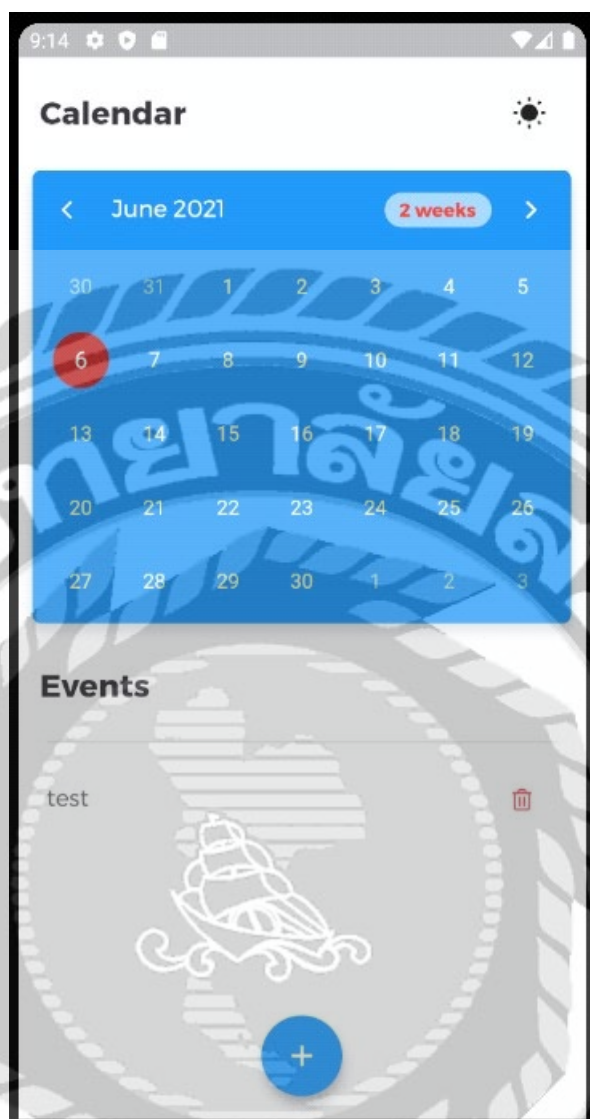
รูปที่ 4.6 แสดงหน้าการสอนออกกำลังกาย

จากรูปที่ 4.6 แสดงหน้าการสอนออกกำลังกาย,เวลาพักและหน้าสรุปคะแนนที่ได้จากการออกกำลังกาย คะแนนที่ได้ก็จะสะสมเพื่อนำไปแลกสินค้าได้ตามภาพ 4.7



รูปที่ 4.7 แสดงหน้าการนำคะแนนมาแลกสินค้า

จากรูปที่ 4.7 แสดงหน้าการนำคะแนนมาแลกสินค้าที่ต้องการและจะได้รหัสเพื่อนำไปแลกสินค้า



รูปที่ 4.8 แสดงหน้าปฏิทินและกิจกรรม

จากรูปที่ 4.8 แสดงหน้าปฏิทินและกิจกรรมที่ผู้ใช้งานเพิ่มขึ้นเอง ผู้ใช้งานสามารถเพิ่มกิจกรรมในวันไหนก็ได้ตามที่ต้องการ



รูปที่ 4.9 แสดงหน้ารายการอาหารแนะนำ

จากรูปที่ 4.9 แสดงหน้ารายการอาหารแนะนำให้กับผู้ใช้งานได้เลือกกินตามต้องการเป็นรายการอาหารที่ช่วยเสริมสร้างกล้ามเนื้อ

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผลปริญญานิพนธ์

แอปพลิเคชันออกกำลังกายเป็นแอปพลิเคชันที่พัฒนาขึ้นมาเพื่อตอบสนองความต้องการของผู้ใช้งาน แอปพลิเคชันนี้จะเป็นตัวช่วยสอนให้ผู้ใช้ออกกำลังกายได้อย่างถูกต้อง และ เลือกกินอาหารที่มีประโยชน์ต่อร่างกายเพื่อช่วยให้ร่างกายสร้างกล้ามเนื้อได้ดีมากยิ่งขึ้น โดยผู้ใช้งานสามารถหาซื้ออาหารได้ตามที่แนะนำ และออกกำลังกายได้ตามตารางที่มีให้ รวมถึงการเทียบค่าสัดส่วนก่อนออกกำลังกายและหลังออกกำลังกายได้อีกด้วยแถมยังมีการเอาคะแนนจากการออกกำลังกายไปแลกซื้อของได้อีกด้วย

5.2 ข้อดีของระบบ

- 5.2.1 ผู้ใช้สามารถออกกำลังกายที่บ้านเองได้ตามตาราง
- 5.2.2 ผู้ใช้สามารถออกกำลังกายได้อย่างถูกวิธีมากยิ่งขึ้น

5.3 ข้อเสนอแนะ

เพื่อให้ระบบมีประสิทธิภาพยิ่งขึ้น ควรมีการพัฒนาฟังก์ชันการทำงานเพิ่ม อาทิเช่น

- 5.3.1 ควรมีระบบจัดการการแลกซื้อของกับทางร้านค้า
- 5.3.2 ควรมีระบบแอดมินสำหรับอัปเดตข้อมูล

บรรณานุกรม

เน็กซ์โฟลว์. (2560). *Cross Platform*. เข้าถึงได้จาก <https://nextflow.in.th/2016/10-year-cross-platform-application-development-thai/>

มายด์พีเอสพี. (2563). *Firebase (ไฟร์เบส) คืออะไร*. เข้าถึงได้จาก <https://www.mindphp.com/คู่มือ/73-คืออะไร/3921-what-is-firebase-backend.html>

สกุล มนธา. (2562). *Dart + Flutter is?*. เข้าถึงได้จาก <https://iamgique.medium.com/dart-flutter-is-f61778caa6a0>

สมาคมโปรแกรมเมอร์ไทย. (2561). *Flutter Framework เครื่องมือที่ประสิทธิภาพในการสร้าง native app แบบหลาย platform*. เข้าถึงได้จาก <https://www.thaiprogrammer.org/2019/11/flutter-framework/>

