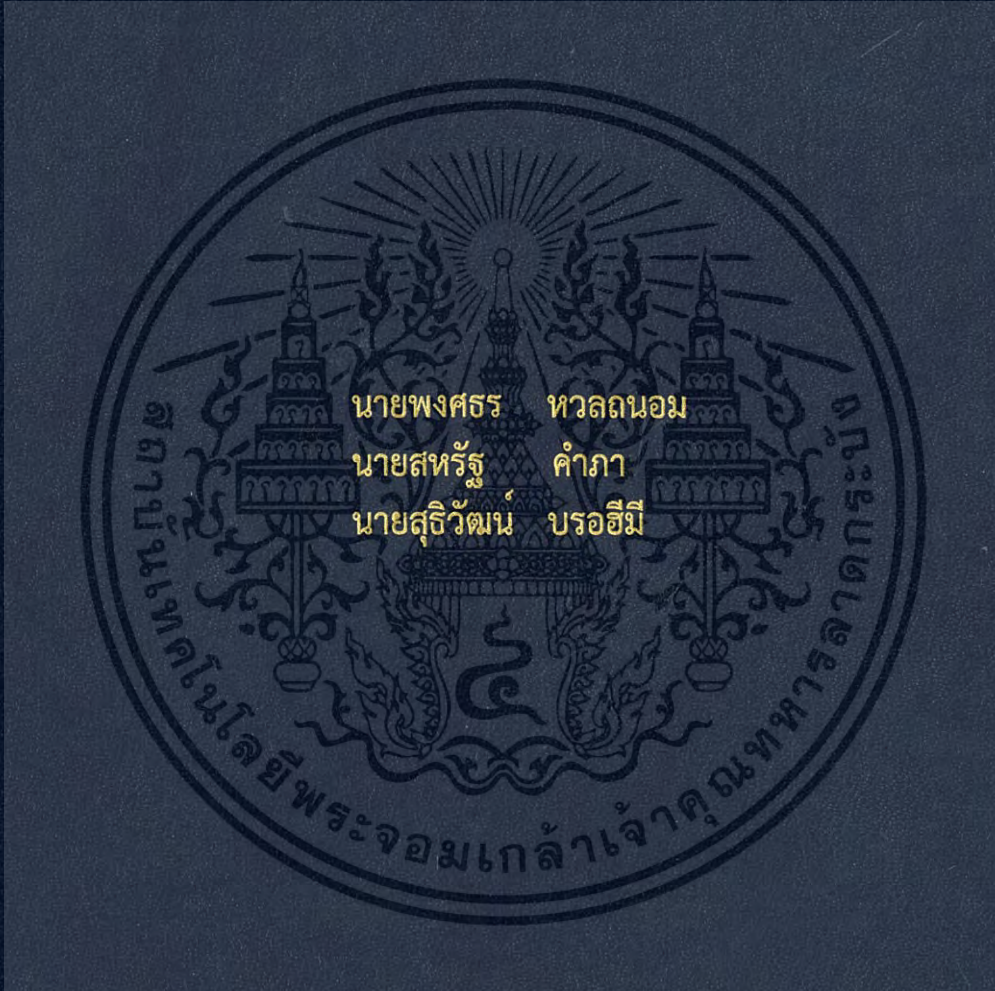


## ฟิตเกมมิ่ง 2 : เกมสุขภาพบนฟิตบิต

### FITGAMING 2 : A HEALTH GAME ON FITBIT



ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)  
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2558

ฟิตเกมมิ่ง 2 : เกมสุขภาพบนฟิตบิต

FITGAMING 2 : A HEALTH GAME ON FITBIT



T149241



เลขหมู่..... 2008  
เลขทะเบียน..... 149241  
วันเดือนปี..... 29 ส.ค. 2561

44879074

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)  
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2558

# FITGAMING 2 : A HEALTH GAME ON FITBIT



A SPECIAL PROBLEM SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENT FOR  
THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE)  
DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG  
ACADEMIC YEAR 2015

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



|                    |  |         |              |          |
|--------------------|--|---------|--------------|----------|
| หัวข้อโครงการพิเศษ | ฟิตเกมมิ่ง 2 : เกมสุขภาพบนฟิตบิต               |         |              |          |
|                    | FitGaming 2 : A Health Game on Fitbit          |         |              |          |
| ชื่อนักศึกษา       | นายพงศธร                                       | หวลถนอม | รหัสนักศึกษา | 55050383 |
|                    | นายสหรัฐ                                       | คำภา    | รหัสนักศึกษา | 55050490 |
|                    | นายสุธิวัฒน์                                   | บรอฮีมี | รหัสนักศึกษา | 55050507 |
| ปริญญา             | วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)        |         |              |          |
| ภาควิชา            | วิทยาการคอมพิวเตอร์                            |         |              |          |
| คณะ                | วิทยาศาสตร์                                    |         |              |          |
| มหาวิทยาลัย        | สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง |         |              |          |
| ปีการศึกษา         | 2558   |         |              |          |
| อาจารย์ที่ปรึกษา   | ดร. รุ่งรัตน์ เวียงศรีพนาวัลย์                 |         |              |          |

### บทคัดย่อ

ปัญหาพิเศษนี้จัดทำขึ้นเพื่อพัฒนาแอปพลิเคชันเกมบนแอนดรอยด์ชื่อฟิตเกมมิ่ง 2 (FitGaming 2) ซึ่งมีวัตถุประสงค์หลักในการนำข้อมูลการออกกำลังกายจากสายรัดข้อมือฟิตบิตมาใช้ในการเล่นเกมเพื่อเพิ่มความท้าทาย โดยการนำค่าที่ได้จากการออกกำลังกายมาแปรเปลี่ยนเป็นพลังงานเพื่อใช้เล่นเกมในแต่ละจังหวัด ภายในเกมมีการอ้างอิงสถานที่ท่องเที่ยวในประเทศไทยตามภูมิภาคและจังหวัด โดยประกอบด้วย 6 ภูมิภาค คือภาคเหนือ ภาคตะวันออกเฉียงเหนือ ภาคกลาง ภาคตะวันตก ภาคตะวันออก และภาคใต้ โดยในแต่ละภูมิภาคจะมีด่านให้เล่น 4 จังหวัด ผู้เล่นจะต้องเล่นเกมวิ่งผ่านสถานที่สำคัญของจังหวัดนั้นและกระโดดข้ามสิ่งกีดขวางเพื่อเอาชนะอุปสรรคและเก็บคะแนนเพื่อไปให้ถึงจุดหมายของด่านนั้น ก่อนที่จะผ่านไปยังด่านถัดไป ความยากในแต่ละด่านจะเพิ่มขึ้นเรื่อย ๆ ตามลำดับของด่านและเวลาในการออกกำลังกาย มีการแข่งขันกับเพื่อนที่ใช้ฟิตบิตด้วยกันและแสดงผลคะแนนตามลำดับเพื่อเพิ่มความท้าทาย มีระบบบันทึกการนอนหลับและแสดงผลการนอนหลับในรูปแบบกราฟแท่ง แอปพลิเคชันจะร้องขอข้อมูลผ่านทาง API ของเว็บไซต์ของฟิตบิต (Fitbit.com) โดยมีการใช้ OAuth 2.0 เฟรมเวิร์ก ในการพิสูจน์ตัวตน PHP และ MySQL ในการพัฒนาเซิร์ฟเวอร์ฟิตเกมมิ่ง 2 แอนดรอยด์สตูดิโอ 2.12 และเฟรมเวิร์กของ LibGDX ในการพัฒนาเกม

คำสำคัญ : เกม ฟิตบิต ฟิตบิตเอพีไอ ลิบจีดีเอ็กซ์ แอนดรอยด์ โอออธ 2.0

|                      |   |
|----------------------|---|
| <b>Title</b>         | FitGaming 2 : A Health Game on Fitbit   |
| <b>Students</b>      | Mr.Pongsatorn Hualtanom Student ID 55050383<br>Mr.Saharat Kampa Student ID 55050490<br>Mr.Sutiwat Barohimee Student ID 55050507 |
| <b>Degree</b>        | Bachelor of Science (Computer Science)  |
| <b>Department</b>    | Computer Science  |
| <b>Faculty</b>       | Science   |
| <b>University</b>    | King Mongkut's Institute of Technology Ladkrabang (KMITL)   |
| <b>Academic Year</b> | 2015  |
| <b>Advisor</b>       | Dr.Rungrat Wiangsripanawan  |

### Abstract

This special problem presents an android game application called “FitGaming 2”. Unlike most of Fitbit applications which simply display the fitness information, FitGaming 2 uses the fitness information as the energy source to play the adventure game. The game consists of six sections according to the location in Thailand: Northern, Northeastern, Central, Western, Eastern and Southern. There are four provinces in each section. In each province, the player has to control the character to jump over various obstacles and run across the famous places in that province to the destination. The energy for jumping and running are from the exercise information. To play the next province, the player has to complete the assigned fitness task. Otherwise, he cannot play although there is energy left. The tasks are gradually increased their difficulties according to the number of the provinces he has played. The game provides the view score menu so that the player can see his score compared with others. FitGaming 2 also allows its player to view his exercise and sleep information. It displays the sleep information as the bar graph. FitGaming 2 uses Fitbit API to retrieve information from Fitbit server, OAuth 2.0 Framework to authenticate the application to Fitbit, PHP and MySQL to implement FitGaming 2 Server, Android Studio 2.12 and LibGDX Framework to develop the game.

**Keywords :** Android, Game, Fitbit, Fitbit API, LibGDX, OAuth 2.0

## กิตติกรรมประกาศ

การจัดทำปัญหาพิเศษหัวข้อ พิตบิตเกมเพื่อสุขภาพสำเร็จลุล่วงไปได้ด้วยดีเนื่องจากผู้จัดทำได้รับความช่วยเหลือจากบุคคลผู้มีพระคุณหลายท่าน ดังนี้

ขอขอบพระคุณ บิดา มารดา และสมาชิกในครอบครัวทุกคน ผู้ซึ่งอยู่เบื้องหลัง คอยอบรมสั่งสอน ให้กำลังใจตลอดมาในการจัดทำปัญหาพิเศษ และความอบอุ่นเสมอมา

ขอขอบพระคุณ ดร.รุ่งรัตน์ เวียงศรีพนาวลัย อาจารย์ที่ปรึกษาในการทำหัวข้อปัญหาพิเศษนี้ ที่ได้ให้คำปรึกษาและคำแนะนำในการแก้ปัญหาต่าง ๆ รวมถึงการตรวจสอบและแก้ไขการเขียนหัวข้อปัญหาพิเศษนี้อย่างละเอียด และที่สำคัญที่สุดอาจารย์ให้กำลังใจให้มีแรงสู้กับปัญหาต่าง ๆ ที่เกิดขึ้นอย่างเต็มที่

รศ.ดร.จิรพร วีระพันธุ์ และ ผศ.ดร.วรางคณา กัมปาน ประธานกรรมการและกรรมการสอบปัญหาพิเศษที่ช่วยชี้แนะปัญหาและแนวทางในการแก้ปัญหา ซึ่งก่อให้เกิดประโยชน์ในการจัดทำและปรับปรุงปัญหาพิเศษ

เพื่อน ๆ พี่ ๆ น้อง ๆ ทั้งในและนอกสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่คอยให้คำชี้แนะและให้กำลังใจเพื่อฝ่าฟันอุปสรรคต่าง ๆ ในการทำปัญหาพิเศษ

สุดท้ายนี้ขอขอบพระคุณคณาจารย์ในภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ ซึ่งได้ให้ความรู้ทางวิชาการ จนกระทั่งผู้จัดทำพอมีความสามารถที่จะดำเนินปัญหาพิเศษลุล่วงได้เช่นนี้ ขอขอบพระคุณทุกท่านจากใจจริง

พงศธร หวลถนอม

สหรัฐ คำภา

สุธิวัฒน์ บรอฮิม

# สารบัญ

|   | หน้า     |
|---|----------|
| บทคัดย่อภาษาไทย .....   | ก        |
| บทคัดย่อภาษาอังกฤษ .....  | ข        |
| กิตติกรรมประกาศ .....   | ค        |
| สารบัญ .....  | ง        |
| สารบัญตาราง .....   | ช        |
| สารบัญรูป .....   | ซ        |
| <b>บทที่ 1 บทนำ</b> .....   | <b>1</b> |
| 1.1 ความเป็นมาและความสำคัญ .....                                  | 1        |
| 1.2 วัตถุประสงค์ในการทำปัญหาพิเศษ .....                           | 1        |
| 1.3 ขอบเขตปัญหาพิเศษ .....  | 1        |
| 1.4 ประโยชน์ที่ได้ .....  | 1        |
| 1.5 ขั้นตอนการดำเนินงาน .....                                     | 2        |
| 1.6 อุปกรณ์ที่ใช้ .....   | 2        |
| <b>บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง</b> .....                | <b>3</b> |
| 2.1 วิดีโอเกม .....   | 3        |
| 2.2 การวัดดัชนีมวลกาย .....                                       | 4        |
| 2.2.1 วิธีคำนวณดัชนีมวลกาย Body Mass Index (BMI) .....            | 4        |
| 2.2.2 อัตราการเผาผลาญพลังงาน Basal Metabolic Rate (BMR) .....     | 5        |
| 2.2.3 วิธีคำนวณการเผาผลาญพลังงาน Basal Metabolic Rate (BMR) ..... | 5        |
| 2.3 อัตราการเต้นของหัวใจ .....                                    | 6        |
| 2.4 อุปกรณ์สวมใส่ Wearable Device .....                           | 7        |
| 2.4.1 ฟิตบิต (Fitbit) .....                                       | 7        |
| 2.4.2 ประเภทของฟิตบิต .....                                       | 7        |
| 2.5 การเข้าถึง Fitbit API .....                                   | 9        |
| 2.5.1 การได้รับความยินยอม .....                                   | 9        |
| 2.5.2 หน้าการในการให้สิทธิ์ .....                                 | 10       |
| 2.5.3 Authorization Code Grant Flow .....                         | 11       |
| 2.5.4 Implicit Grant Flow .....                                   | 14       |
| 2.5.5 การกำหนดข้อมูล .....  | 16       |
| 2.5.6 โหมดการเข้าถึงแอปพลิเคชัน .....                             | 17       |
| 2.5.7 การตอบกลับของ URIs .....                                    | 17       |
| 2.5.8 แอปพลิเคชันที่พัฒนาโดยใช้ Fitbit API .....                  | 18       |
| 2.5.8.1 Fitbit .....  | 18       |
| 2.5.8.2 FitGaming .....   | 19       |
| 2.5.8.3 FitRPG .....  | 20       |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

|   | หน้า      |
|---|-----------|
| 2.6 มายเอสคิวแอล(MySQL).....                                      | 21        |
| 2.7 เว็บเซิร์ฟเวอร์ (Web server) .....                            | 21        |
| 2.7.1 XAMPP.....  | 21        |
| 2.7.2 แอปเซิร์ฟ(AppServ) .....                                    | 22        |
| 2.7.3 อาปาเช(Apache) .....  | 22        |
| 2.7.4 พีเอชพี (PHP).....  | 22        |
| 2.8 เจสัน(JSON) .....   | 23        |
| 2.9 Libgdx .....  | 24        |
| <b>บทที่ 3 การวิเคราะห์ระบบและออกแบบระบบ.....</b>                 | <b>25</b> |
| 3.1 สถาปัตยกรรมระบบ .....   | 25        |
| 3.2 การวิเคราะห์เกม.....  | 26        |
| 3.2.1 Functional Requirement .....                                | 26        |
| 3.2.2 Non-Functional Requirement .....                            | 26        |
| 3.2.3 Use Case Diagram.....                                       | 27        |
| 3.2.4 Activity Diagram .....                                      | 28        |
| 3.2.5 Sequence Diagram .....                                      | 37        |
| 3.3 การออกแบบฐานข้อมูล.....                                       | 42        |
| 3.4 การออกแบบแอปพลิเคชัน.....                                     | 43        |
| 3.5 การพัฒนาแอปพลิเคชัน .....                                     | 46        |
| 3.6 การพัฒนาแอปพลิเคชันในส่วนของแอนดรอยด์ .....                   | 46        |
| 3.6.1 การตั้ง Url ของแอนดรอยด์ เพื่อรับข้อมูลจากเว็บพีดีบีต ..... | 46        |
| 3.6.2 สร้างฟังก์ชันการติดต่อ Fitbit API ด้วย OAuth 2.0.....       | 47        |
| 3.6.2.1 การลงทะเบียนแอปพลิเคชันฝั่งพีดีบีตเซิร์ฟเวอร์ .....       | 48        |
| 3.6.3 แอปพลิเคชันฝั่งไคลเอนต์ (Application on Client-Side).....   | 50        |
| 3.7 การพัฒนาแอปพลิเคชันในส่วนของเกม โดยใช้ Libgdx.....            | 55        |
| 3.7.1 ทำการตั้งค่าคลาสหลักเพื่อรองรับการเปลี่ยนหน้าจอหลายจอ ..... | 55        |
| 3.7.2 การใช้คลาส Texture.....                                     | 56        |
| 3.7.3 การเขียนตัวอักษร .....                                      | 58        |
| 3.7.4 การใช้คลาส Button.....                                      | 58        |
| 3.7.5 การใช้คลาส Window .....                                     | 61        |
| 3.7.6 การเขียนโค้ดในส่วนของ LevelChoose.....                      | 61        |
| 3.7.7 การทำตัวละคร Player .....                                   | 67        |
| 3.7.8 การทำสิ่งกีดขวางให้ชนผู้เล่น.....                           | 71        |

## สารบัญ (ต่อ)

|   | หน้า      |
|---|-----------|
| 3.7.9 การแสดงค่าพลังและการคิดคำนวณ.....                           | 75        |
| 3.7.10 การคำนวณคะแนน และระบบตอนจบ.....                            | 76        |
| 3.7.11 การส่งข้อมูลการนอนหลับไปให้พีตบิต.....                     | 77        |
| 3.7.12 การดึงข้อมูลการนอนหลับและการแสดงด้วยกราฟ .....             | 77        |
| 3.7.13 การดึงข้อมูลคะแนนจาก Web Server โดยใช้ JSON Format .....   | 80        |
| <b>บทที่ 4 ผลลัพธ์ของระบบ.....</b>                                | <b>83</b> |
| 4.1 ผลลัพธ์ของแอปพลิเคชัน .....                                   | 83        |
| 4.2 ฟังก์ชันของแอปพลิเคชัน.....                                   | 83        |
| <b>บทที่ 5 บทสรุปและข้อเสนอแนะ .....</b>                          | <b>89</b> |
| 5.1 บทสรุป.....   | 89        |
| 5.2 ข้อจำกัดการพัฒนาแอปพลิเคชัน .....                             | 89        |
| เอกสารอ้างอิง .....   | 91        |
| ภาคผนวก.....  | 94        |
| ภาคผนวก ก การติดตั้งโปรแกรมในการพัฒนาแอปพลิเคชัน.....             | 95        |
| ก.1 การติดตั้งไฟล์ apk.....                                       | 95        |
| ก.2 การติดตั้ง XAMPP .....  | 99        |
| ก.3 การติดตั้ง Android Studio .....                               | 103       |
| ก.4 การติดตั้ง LibGDX.....  | 107       |
| ภาคผนวก ข คู่มือการใช้งานแอปพลิเคชัน .....                        | 110       |
| ข.1 ขั้นตอนการใช้งานแอปพลิเคชัน.....                              | 110       |
| ข.1.1 ลงทะเบียน.....  | 110       |
| ข.1.2 หน้าจอเลือกภาคและด่าน .....                                 | 110       |
| ข.1.3 หน้าจอภายใน Town.....                                       | 111       |
| ข.1.4 หน้าจอ Play Game .....                                      | 111       |
| ข.1.5 หน้าจอ Sleep Room .....                                     | 112       |
| ข.1.6 หน้าจอ Sleep View.....                                      | 112       |
| ข.1.7 หน้าจอ Profile.....   | 113       |
| ข.1.8 หน้าจอ View Score.....                                      | 113       |
| ภาคผนวก ค ตัวอย่างข้อมูลสถานที่ท่องเที่ยวจากจังหวัดในภาคกลาง..... | 114       |
| ค.1 จังหวัดกรุงเทพมหานคร.....                                     | 114       |
| ค.2 จังหวัดพระนครศรีอยุธยา.....                                   | 116       |
| ค.3 จังหวัดเพชรบุรี.....  | 118       |

## สารบัญตาราง

| ตารางที่   | หน้า |
|--|------|
| 2.1 เปรียบเทียบความสามารถของอุปกรณ์วัดการทำกิจกรรมต่าง ๆ ของร่างกาย..... | 9    |
| 2.2 พารามิเตอร์ URI .....  | 10   |
| 2.3 ขอบเขตของแอปพลิเคชัน .....   | 16   |
| 3.1 ตาราง profile บนเซิร์ฟเวอร์.....                                     | 44   |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

| รูปที่ |  | หน้า |
|--------|--|------|
| 2.1    | การเดินของหัวใจกับการออกกำลังกาย .....                                 | 7    |
| 2.2    | พิตบิตเฟล็กซ์ .....  | 7    |
| 2.3    | พิตบิตชาร์จ .....  | 8    |
| 2.4    | พิตบิตชาร์จเฮทอาร์ .....   | 8    |
| 2.5    | พิตบิตเซอร์จ .....   | 9    |
| 2.6    | แสดงถึงการใช้ OAuth 2.0 Authorization Code Grant Flow .....            | 13   |
| 2.7    | แสดงถึงการใช้ OAuth 2.0 Implicit Grant Flow .....                      | 15   |
| 2.8    | แอปพลิเคชัน Fitbit.....  | 18   |
| 2.9    | แอปพลิเคชัน FitGaming .....  | 19   |
| 2.10   | แอปพลิเคชัน FitRPG .....   | 20   |
| 3.1    | สถาปัตยกรรมของระบบ .....   | 25   |
| 3.2    | Use Case Diagram ของเกมส์บนแอนดรอยด์ .....                             | 27   |
| 3.3    | Activity Diagram ของ Sign in .....                                     | 28   |
| 3.4    | Activity Diagram ของ levelChoose เลือกภาค .....                        | 30   |
| 3.5    | Activity Diagram ของ levelChoose เลือกเมือง.....                       | 31   |
| 3.6    | Activity Diagram ของ town หน้าจอของเมือง .....                         | 32   |
| 3.7    | Activity Diagram ของ PlayGame หน้าเล่นเกมส์ .....                      | 33   |
| 3.8    | Activity Diagram ของ SleepRoom ห้องนอนหลับ .....                       | 34   |
| 3.9    | Activity Diagram ของ SleepView แสดงข้อมูลการนอนหลับ .....              | 34   |
| 3.10   | Activity Diagram ของ profile หน้าจอแสดงข้อมูลของตนเอง .....            | 35   |
| 3.11   | Activity Diagram ของ ViewScore หน้าจอแสดงคะแนนของตัวเองและเพื่อน ..... | 35   |
| 3.12   | Sequence Diagram Sign in .....   | 36   |
| 3.13   | Sequence Diagram LevelChoose.....                                      | 37   |
| 3.14   | Sequence Diagram Town ขณะอยู่ในแต่ละจังหวัด .....                      | 38   |
| 3.15   | Sequence Diagram การเล่นเกมในแต่ละจังหวัด .....                        | 39   |
| 3.16   | Sequence Diagram การดูข้อมูลโปรไฟล์ .....                              | 40   |
| 3.17   | Sequence Diagram ในหน้าหลักของ Sleep Room.....                         | 40   |
| 3.18   | Sequence Diagram การดูข้อมูลการนอน .....                               | 41   |
| 3.19   | Sequence Diagram การดูคะแนน .....                                      | 41   |
| 3.20   | ภาพหน้าจอ Sign in.....   | 43   |
| 3.21   | ภาพหน้าจอ เลือกภาค .....   | 43   |
| 3.22   | ภาพหน้าจอ เมือง.....   | 44   |
| 3.23   | ภาพหน้าจอ ห้องนอน .....  | 44   |
| 3.24   | ภาพหน้าจอดูเวลาการนอนรายสัปดาห์ .....                                  | 44   |
| 3.25   | ภาพหน้าจอการเล่นเกม .....  | 45   |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

| รูปที่ | หน้า   |
|--------|--|
| 3.26   | ภาพหน้าจอ Profile..... 45  |
| 3.27   | ภาพหน้าจอแสดงคะแนนเพื่อน..... 46                                   |
| 3.28   | ภาพหน้าจอแสดงโปรไฟล์เพื่อน..... 46                                 |
| 3.29   | การเพิ่มสิทธิ์การใช้งานอินเทอร์เน็ตในแอนดรอยด์แอปพลิเคชัน..... 46  |
| 3.30   | โค้ดกำหนดที่อยู่ข้อมูลของแอนดรอยด์..... 47                         |
| 3.31   | หน้าจอสมัครสมาชิก..... 48  |
| 3.32   | เข้าสู่ระบบสำหรับการพัฒนาแอปพลิเคชันพีตบิต..... 48                 |
| 3.33   | การลงทะเบียนแอปพลิเคชันพีตบิต..... 49                              |
| 3.34   | หน้าจอแสดงพารามิเตอร์หลังจากที่ลงทะเบียนเรียบร้อยแล้ว..... 50      |
| 3.35   | การเปิดหน้าเว็บเพื่อไปเรียกขอข้อมูลพีตบิต..... 50                  |
| 3.36   | โค้ดรับข้อมูล code จากที่เว็บพีตบิตส่งมา..... 51                   |
| 3.37   | การส่งข้อมูลที่ได้จากพีตบิตและข้อมูลเข้ารหัสไปยังคลาสถัดไป..... 51 |
| 3.38   | การรับข้อมูลและตั้งค่าส่งไปยังคลาส Libgdx..... 51                  |
| 3.39   | เมธอดการเรียก Https Request (1)..... 52                            |
| 3.40   | เมธอดการเรียก Https Request (2)..... 52                            |
| 3.41   | การเรียกใช้เมธอด Https ในการติดต่อขอโทเค็น..... 53                 |
| 3.42   | การดึงข้อมูลจากก้อน JSON ตามรูปแบบพีตบิต..... 53                   |
| 3.43   | ตัวอย่างการดึงข้อมูลพีตบิตส่วนตัวของผู้ใช้..... 54                 |
| 3.44   | ตัวอย่างการดึงข้อมูลโปรไฟล์ของผู้ใช้งานจาก JSON..... 54            |
| 3.45   | เมธอดการตัดการเชื่อมต่อ..... 54                                    |
| 3.46   | คลาสหลัก Fitgaming extends คลาส Game..... 55                       |
| 3.47   | โค้ดที่ทำให้รองรับหน้าจอตามคลาสที่เปลี่ยน..... 55                  |
| 3.48   | การตั้งค่าคลาสหลักในส่วนของ create..... 55                         |
| 3.49   | การเขียนโค้ดเปลี่ยนหน้าจอของคลาสหลัก..... 56                       |
| 3.50   | การเขียนเปลี่ยนหน้าจอของคลาส Screen อื่นๆ..... 56                  |
| 3.51   | การตั้งค่าคลาสในส่วนตอนที่สร้างคลาส..... 56                        |
| 3.52   | การตั้งค่ากล้องในส่วนของ render..... 56                            |
| 3.53   | การประกาศภาพ Texture..... 57                                       |
| 3.54   | ตัวอย่างการวาด Texture ใน render..... 57                           |
| 3.55   | การเขียนโค้ดตัวอักษร..... 58                                       |
| 3.56   | รูปที่กดปุ่มแล้ว และรูปที่ยังไม่ได้กดปุ่ม..... 58                  |
| 3.57   | ไฟล์ที่ได้มาจากโปรแกรม gdx-texture..... 59                         |
| 3.58   | โค้ดในไฟล์ .atlas..... 59  |
| 3.59   | เขียนไฟล์ JSON สำหรับปุ่ม..... 59                                  |
| 3.60   | ประกาศ Skin Stage และตั้งค่าให้ Stage สามารถกดได้..... 60          |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

| รูปที่   | หน้า |
|--|------|
| 3.61 โค้ดการเขียนปุ่ม Button.....  | 60   |
| 3.62 การใช้พื้นที่ Stage ให้แสดงรูป Button.....                          | 60   |
| 3.63 ไฟล์ JSON ของ Window.....   | 61   |
| 3.64 โค้ดในการเขียนคลาส Window.....                                      | 61   |
| 3.65 โค้ดประกาศตัวแปรกดปุ่ม ภาพ และการกดปุ่มของคลาส Stage ที่ประกาศ..... | 62   |
| 3.66 โค้ดการตั้งค่าหน้าจอด้วย OthographicCamera.....                     | 62   |
| 3.67 ส่วน render ในหน้าจอของ TownChoose SectionChoose.....               | 63   |
| 3.68 ประกาศสร้างปุ่มเลือกภาพ ทั้ง 6 ภาค (1).....                         | 64   |
| 3.69 ประกาศสร้างปุ่มเลือกภาพ ทั้ง 6 ภาค (2).....                         | 65   |
| 3.70 สร้างปุ่มเลือกจังหวัดของภาคกลาง (1).....                            | 65   |
| 3.71 สร้างปุ่มเลือกจังหวัดของภาคกลาง (2).....                            | 66   |
| 3.72 ขั้นตอนการทำภาพแอนิเมชันในแต่ละเฟรม.....                            | 67   |
| 3.73 รูปภาพทั้ง 16 เฟรม.....   | 68   |
| 3.74 สไปรท์ชีทสำหรับการเคลื่อนไหวของตัวละคร.....                         | 68   |
| 3.75 โค้ดการใช้คลาส Animation.....                                       | 68   |
| 3.76 โค้ดในการกำหนดตำแหน่งตัวละครและการเคลื่อนไหวของตัวละคร.....         | 69   |
| 3.77 โค้ดในการลงภาพเคลื่อนไหวของ Player.....                             | 69   |
| 3.78 การคำนวณการเคลื่อนไหวของตัวละคร กล้อง และเส้นชัย.....               | 70   |
| 3.79 โค้ดในการใส่ปุ่มกระโดดและวิธีการคำนวณ.....                          | 71   |
| 3.80 โค้ดในการเพิ่มลดค่ากระโดด และการลดเกินขีดจำกัด.....                 | 71   |
| 3.81 โค้ดการประกาศรูปภาพ Texture ของ อุปสรรค.....                        | 72   |
| 3.82 โค้ดกำหนดตำแหน่งและชนิดของอุปสรรค (1).....                          | 72   |
| 3.83 โค้ดกำหนดตำแหน่งและชนิดของอุปสรรค (2).....                          | 73   |
| 3.84 โค้ดตัวอย่างในการวาดสิ่งกีดขวาง.....                                | 73   |
| 3.85 โค้ดในการชนของตัวละครและสิ่งกีดขวาง (1).....                        | 74   |
| 3.86 โค้ดในการชนของตัวละครและสิ่งกีดขวาง (2).....                        | 74   |
| 3.87 เมธอดเมื่อตัวละครได้ชนกับสิ่งกีดขวาง.....                           | 75   |
| 3.88 การแสดงค่าพลังงานที่ลดต่อเนื่อง.....                                | 75   |
| 3.89 โค้ดคำนวณค่าลดพลังงาน.....  | 75   |
| 3.90 โค้ดแสดงเงื่อนไขเมื่อแพ้และชนะ.....                                 | 76   |
| 3.91 เมธอดแสดงหน้าจอคะแนนของเกมที่ใช้เล่น.....                           | 77   |
| 3.92 โค้ดในการส่งข้อมูลการนอนหลับด้วย Https Request.....                 | 77   |
| 3.93 การดึงข้อมูลการนอนหลับโดยเรียกใช้เมธอด Https Request 7 วัน.....     | 78   |
| 3.94 การดึงข้อมูลการนอนหลับจาก JSON และบันทึกลง Preference.....          | 79   |
| 3.95 การดึงข้อมูลออกจาก Preference และดึง JSON มาตั้งกราฟ.....           | 80   |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

| รูปที่  | หน้า |
|---|------|
| 3.96 การดึงคะแนนจาก Webserver โดยใช้เมธอด httpRequest .....           | 81   |
| 3.97 โค้ดการสร้างหน้าจอสถาปัตยกรรมของ ViewScore (1) .....             | 81   |
| 3.98 โค้ดการสร้างหน้าจอสถาปัตยกรรมของ ViewScore (2) .....             | 82   |
| 4.1 หน้าจอการยืนยันตัวตนผ่านเว็บไซต์พีดีบีทีก่อนเข้าเกม .....         | 84   |
| 4.2 หน้าจอเลือกภาคในแต่ละด่าน (จังหวัด).....                          | 85   |
| 4.3 หน้าจอ Town .....   | 85   |
| 4.4 หน้าจอ Play Game .....  | 86   |
| 4.5 หน้าจอ Sleep Room .....   | 86   |
| 4.6 หน้าจอ Sleep View .....   | 87   |
| 4.7 หน้าจอ Profile .....  | 87   |
| 4.8 หน้าจอ View Score.....  | 88   |
| ก.1 หน้าจอไฟล์ที่เชื่อมต่อกับคอมพิวเตอร์.....                         | 95   |
| ก.2 หน้าจอ Internal Storage จากอุปกรณ์ที่เชื่อมต่อกับคอมพิวเตอร์..... | 95   |
| ก.3 หน้าจอการใส่ไฟล์ .apk ลงเครื่อง.....                              | 96   |
| ก.4 หน้าจอการเปิดการใช้งาน Unknown Sources (1).....                   | 96   |
| ก.5 หน้าจอการเปิดการใช้งาน Unknown Sources (2).....                   | 97   |
| ก.6 หน้าจอการเปิดการใช้งาน Unknown Sources (3).....                   | 97   |
| ก.7 หน้าจอการค้นหาไฟล์ .apk เพื่อเตรียมทำการติดตั้ง .....             | 98   |
| ก.8 หน้าจอการติดตั้งไฟล์ apk .....                                    | 98   |
| ก.9 หน้าจอก่อนเปิดแอปพลิเคชันหลังติดตั้งเสร็จ .....                   | 99   |
| ก.10 หน้าจอเริ่มต้นการติดตั้ง XAMPP .....                             | 99   |
| ก.11 หน้าจอการติดตั้ง Component ต่างๆ.....                            | 100  |
| ก.12 หน้าจอกำหนด Path ของโปรแกรม .....                                | 100  |
| ก.13 หน้าจอเตรียมพร้อมที่จะติดตั้งโปรแกรม XAMPP .....                 | 101  |
| ก.14 หน้าจอของโปรแกรม XAMPP ที่กำลังถูกติดตั้ง.....                   | 101  |
| ก.15 หน้าจอการแจ้งเตือนของ Firewall.....                              | 102  |
| ก.16 หน้าจอการติดตั้งโปรแกรม XAMPP เสร็จสิ้น .....                    | 102  |
| ก.17 หน้าจอดาวน์โหลด Android Studio.....                              | 103  |
| ก.18 หน้าจอยอมรับข้อตกลงในการใช้งาน.....                              | 103  |
| ก.19 หน้าจอติดตั้งโปรแกรม Android Studio .....                        | 104  |
| ก.20 หน้าจอการติดตั้ง Component ต่างๆ.....                            | 104  |
| ก.21 หน้าจอข้อตกลงการใช้งานของ Android Studio .....                   | 105  |
| ก.22 หน้าจอกำหนด Path ของโปรแกรม .....                                | 105  |
| ก.23 หน้าจอกำหนด Start Menu Folder และเตรียมพร้อมการติดตั้ง.....      | 106  |
| ก.24 หน้าจอโปรแกรม Android Studio ที่กำลังถูกติดตั้ง .....            | 106  |

## สารบัญรูป (ต่อ)

| รูปที่  | หน้า |
|---|------|
| ก.25 หน้าจอการติดตั้งโปรแกรม Android Studio เสร็จสิ้น.....        | 107  |
| ก.26 หน้าจอดาวน์โหลด LibGDX .....                                 | 107  |
| ก.27 หน้าจอโปรแกรม LibGDX.....                                    | 108  |
| ก.28 หน้าจอดาวน์โหลดไฟล์ในโปรแกรม LibGDX.....                     | 108  |
| ก.29 หน้าจอของไฟล์ต่างๆที่ผ่านการดาวน์โหลดจากโปรแกรม LibGDX ..... | 109  |
| ข.1 หน้าจอ Log in ของ Fitbit ก่อนเข้าเกม .....                    | 111  |
| ข.2 หน้าจอเลือกภาคและด่าน (จังหวัด).....                          | 112  |
| ข.3 หน้าจอ Town .....   | 112  |
| ข.4 หน้าจอ Play Game .....  | 113  |
| ข.5 หน้าจอ Sleep Room.....  | 113  |
| ข.6 หน้าจอ Sleep View .....                                       | 114  |
| ข.7 หน้าจอ Profile .....  | 114  |
| ข.8 หน้าจอ View Score.....  | 115  |



# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันคนไทยส่วนมากเริ่มให้ความสนใจกับการดูแลสุขภาพของตนเองมากขึ้น มีการนำอุปกรณ์สวมใส่ประเภท wearable device ที่สามารถเชื่อมต่อและอัปเดตข้อมูลการออกกำลังกายเข้ากับสมาร์ทโฟน เพื่อบันทึกข้อมูลการออกกำลังกายและประเมินสุขภาพของตนเองในรูปแบบที่ผู้ใช้ทั่วไปสามารถเข้าใจได้ง่าย ฟิตบิต (Fitbit) เป็นหนึ่งใน wearable device ที่มีผู้ใช้งานค่อนข้างมาก เนื่องจากมีหลายรุ่นหลายราคา ผู้ใช้สามารถเลือกซื้อได้ตามฟังก์ชันที่ต้องการ ในส่วนของการพัฒนาแอปพลิเคชัน ทางฟิตบิตได้จัดทำ Fitbit API ไว้ให้สามารถดึงข้อมูลได้โดยง่ายและมีข้อมูลให้ใช้หลากหลาย ทั้งในส่วนของ การออกกำลังกาย รับประทานอาหาร หรือแม้แต่การนอนหลับ อย่างไรก็ตามแอปพลิเคชันฟิตบิตที่มีอยู่หลากหลายมักจะแสดงผลข้อมูลเท่านั้น ไม่เป็นที่น่าดึงดูดสำหรับบุคคลทั่วไปที่ไม่ได้ออกกำลังกายอย่างสม่ำเสมอ ขาดความท้าทายในการใช้งานระยะยาว ผู้จัดทำจึงเกิดแนวคิดในการนำข้อมูลการออกกำลังกายของฟิตบิตมาใช้ในการเล่นเกมเพื่อให้ผู้ออกกำลังกายมีแรงจูงใจในการออกกำลังกายอย่างต่อเนื่องและสม่ำเสมอ

เกมบนฟิตบิตในปัญหาพิเศษนี้เป็นเวอร์ชันที่ 2 โดยมีการพัฒนาเพิ่มจากเวอร์ชันแรกในส่วนของเกม จากเดิมเป็นเกมผจญภัยแบบผู้เล่นต้องปฏิบัติตามภารกิจออกกำลังกายให้สำเร็จจึงจะสามารถเลื่อนไปเล่นด่านถัดไปได้ เป็นเกมผจญภัยที่ผู้เล่นต้องปฏิบัติตามภารกิจเพื่อนำเคล็ดลับที่ได้จากการออกกำลังกายไปเป็นพลังงานในการเล่นเกมกระโดดข้ามสิ่งกีดขวางโดยฉากในเกมจะเป็นสถานที่ท่องเที่ยวของแต่ละจังหวัด และในการเข้าเล่นด่านถัดไปนั้น นอกจากผู้เล่นจะต้องเล่นผ่านด่านปัจจุบันแล้ว ผู้เล่นจะต้องปฏิบัติตามภารกิจการออกกำลังกายตามที่กำหนดจึงจะสามารถเล่นได้ เกมจะมีภารกิจที่ยากขึ้นตามลำดับ โดยภารกิจในเกมด่านที่สูงขึ้นไปนั้นจะมีการวางแผนให้สอดคล้องกับระยะเวลาออกกำลังกาย เช่น ถ้าไม่ทำการออกกำลังกายในวันนี้อีกก่อน จำนวน 20 นาที จะไม่สามารถเล่นด่านนี้ได้ เป็นต้น

### 1.2 วัตถุประสงค์ในการทำปัญหาพิเศษ

- 1) เพื่อศึกษาการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการ Android
- 2) เพื่อศึกษาการใช้ API (Application Programming Interface) ของ Fitbit
- 3) เพิ่มความน่าสนใจและสนุกสนานในการใช้สายรัดข้อมือ Fitbit
- 4) เพื่อแนะนำสถานที่ท่องเที่ยวในประเทศไทย

### 1.3 ขอบเขตของปัญหาพิเศษ

- 1) เกมนี้สามารถเล่นได้บนระบบปฏิบัติการแอนดรอยด์เท่านั้น
- 2) ก่อนเล่นเกมผู้ใช้จะต้องลงทะเบียนกับ Fitbit ก่อน โดยสามารถลงทะเบียนคอมพิวเตอร์หรือผ่าน Fitbit app ในระบบปฏิบัติการแอนดรอยด์
- 3) ในการเล่นเกมผู้ใช้ต้องสวมและเชื่อมต่อสายรัดข้อมือฟิตบิต
- 4) ต้องเข้าเล่นเกมด้วยบัญชีของฟิตบิต
- 5) ชื่อด่านในเกมจะเป็นชื่อในภูมิภาคของประเทศไทย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 1.4 ประโยชน์ที่ได้

- 1) ผู้ใช้ได้แรงบันดาลใจกับการออกกำลังกายมากยิ่งขึ้น
- 2) ผู้ใช้มีความสนุกสนานไปกับการเล่นแอปพลิเคชัน
- 3) ผู้ใช้มีสุขภาพที่ดีขึ้นจากการออกกำลังกาย
- 4) ผู้พัฒนาได้เรียนรู้วิธีการพัฒนาแอปพลิเคชัน Fitbit และการส่งผ่านข้อมูลระหว่าง Fitbit และแอปพลิเคชัน

#### 1.5 ขั้นตอนการดำเนินงาน

- 1) ศึกษาวิธีการทำแอปพลิเคชันของระบบปฏิบัติการ Android และเครื่องมืออุปกรณ์ที่เกี่ยวข้อง
- 2) ออกแบบและวิเคราะห์การทำเกม
- 3) รวบรวมข้อมูลที่เป็น ทำการออกแบบหน้าจอและ systemboard ของเกม
- 4) ดำเนินการเขียนโค้ดเกม
- 5) ทดสอบความถูกต้องของแอปพลิเคชัน
- 6) ทำเอกสารคู่มือการใช้งาน

#### 1.6 อุปกรณ์ที่ใช้

##### ฮาร์ดแวร์ (Hardware)

- 1) สายรัดข้อมือสุขภาพ Fitbit Charge HR
- 2) โน้ตบุ๊ก ยี่ห้อ Acer รุ่น Aspire V3-4710-53214G75Makk
  1. หน่วยประมวลผล Intel(R) Core(TM) i5-3210M ความเร็ว 2.5 กิกะเฮิร์ตซ์
  2. RAM ความจุ 8 กิกะไบต์
  3. การ์ดจอ NVIDIA GeForceGT 630M
- 3) Samsung GALAXY Note 3

##### ซอฟต์แวร์ (Software)

- 1) Android Studio 2.12
- 2) Photoshop CS6 Extended
- 3) Gdx-texturepacker.jar

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# ทฤษฎีที่เกี่ยวข้อง

### 2.1 วิดีโอเกม

วิดีโอเกม (Video game) [1] [2] คือเกมอิเล็กทรอนิกส์ที่มนุษย์ใช้ส่วนต่อประสานกับผู้ใช้ (user interface) สร้างผลป้อนกลับไปยังอุปกรณ์วิดีโอ คำว่า วิดีโอ ในวิดีโอเกม แต่เดิมหมายถึง อุปกรณ์แสดงภาพแบบแรสเตอร์แต่ปัจจุบันสามารถใช้เรียกอุปกรณ์แสดงภาพใด ๆ ก็ได้ที่สร้างภาพสองมิติหรือสามมิติขึ้นมา ตัวอย่างเช่นคอมพิวเตอร์ส่วนบุคคล และเครื่องเล่นวิดีโอเกม อุปกรณ์เหล่านี้เป็นไปได้ตั้งแต่เมนเฟรมคอมพิวเตอร์ ไปจนถึงอุปกรณ์มือถือขนาดเล็ก ประเภทและชนิดของเกมมีดังนี้ [3]

- เกมเลียนแบบหรือการจำลอง (Simulation Games) ซึ่งเป็นเกมที่พยายามเลียนแบบเหตุการณ์จริง เพื่อพัฒนาทักษะของผู้เล่น เช่น การฝึกบินจำลอง การขับรถจำลอง
- เกมแอคชั่นแบบ FPS (Action First Person Shooters Games) เป็นเกม ยิงปืนที่ผู้เล่นเป็นตัวละครไอ้ผู้ร้าย ไปจนถึงสัตว์ประหลาดต่างๆตามระดับการเล่น มีทั้งเล่นแบบคนเดียวและเล่นเป็นกลุ่ม
- เกมผจญภัย (Adventure Games) มีวัตถุประสงค์ของเกมเพื่อทำภารกิจให้สำเร็จในดินแดนที่สร้างขึ้น ต้องแก้ไขปัญหาหรือหาสิ่งจำเป็นในระดับของเกมที่แตกต่างกันไป
- เกม RPG (Role-Playing) เป็นเกมที่ผู้เล่นสามารถสร้างหรือเลือก ลักษณะของตัวละครให้ตรงกับความต้องการของตัวเอง แล้วเล่นไปตามเนื้อเรื่องของเกม
- เกมต่อสู้ (Fighting Game) เป็นเกมต่อสู้กันโดยมีตัวละครต่อสู้กันแบบตัวต่อตัว ด้วยเทคนิคการต่อสู้เฉพาะตัวของผู้เล่น
- เกมวางแผน (Strategy Games) เกมที่ใช้ความคิด นำกลยุทธ์มาใช้เพื่อเอาชนะ เกมมีเรื่องราวเป็นนิทาน หรือตำนาน มีตัวละครนำและการผูกเรื่องเข้ากับการต่อสู้และวางแผนในเกม
- เกมปริศนา (Puzzle Game) เกมแก้ปัญหาให้ลุล่วงตามจุดประสงค์หลักของเกม
- เกมกีฬาและการแข่งขัน (Sport & Racing Games) วัตถุประสงค์ของเกมเพื่อการเป็นที่หนึ่งของการแข่งขัน
- เกมการศึกษา วัตถุประสงค์เพื่อให้ได้ความรู้และความเพลิดเพลิน
- เกมแนวดนตรีหรือการจับจังหวะ (Music Game) คือเกมที่ผู้เล่นต้องใช้เสียงเพลงในการเล่นด้านต่างๆให้ชนะ ซึ่งผู้เล่นจะต้องกดปุ่มให้ถูกต้องหรือตรงจังหวะหรือตรงตำแหน่ง โดยใช้เสียงเพลงเป็นตัวบอกเวลาที่จะต้องกด
- ปาร์ตี้เกม (Party Game) คือเกมที่มีการบรรจุเกมย่อยๆ มากมายเอาไว้ โดยในแต่ละเกมย่อยจะมีกฎและกติกาที่ต่างกันออกไป โดยผู้เล่นจะต้องเข้าไปเล่นในเกมย่อยนั้นๆ และหาทางแข่งขันกับผู้เล่นอื่น ๆ ให้ชนะ (ทั้งคอมพิวเตอร์และผู้เล่นที่เป็นมนุษย์ด้วย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กันเอง) จุดขายของปาร์ตี้เกมคือการเล่นเป็นหมู่คณะ ซึ่งจะสร้างความบันเทิงได้มากกว่าการเล่นคนเดียว

- เกมอาเขต (Arcade Game) คือเกมที่ถูกสร้างมาให้กับเครื่องเกมตู้ โดยส่วนมากเกมประเภทนี้มักจะใช้เวลาจบไม่นาน (ไม่เกิน 30 นาที หรือ ไม่เกิน 1 ชั่วโมง) เน้นความเรียบง่ายของตัวเกม มักมีเวลาจำกัดในการเล่นและมักจะไม่มีกัณฑ์ความก้าวหน้าในการเล่น เกมจะบันทึกเพียงคะแนนสูงสุดเท่านั้น เกมประเภทนี้มักมีความท้าทายของระดับความยากง่ายดึงดูดใจให้ผู้เล่นกลับมาเล่นซ้ำ

## 2.2 การวัดดัชนีมวลร่างกาย

การวัดดัชนีมวลร่างกาย Body Mass Index (BMI) [8] คือ อัตราส่วนระหว่างน้ำหนักต่อส่วนสูง ที่ใช้บ่งบอกว่าอ้วนหรือผอม ในผู้ใหญ่ตั้งแต่อายุ 20 ปีขึ้นไป ความสำคัญของการรู้ค่าดัชนีมวลร่างกาย เพื่อดูอัตราการเสี่ยงต่อการเกิดโรคต่าง ๆ ถ้าค่าที่คำนวณได้ มากหรือน้อยเกินไป เพราะถ้าเป็นโรคอ้วนแล้ว จะมีภาวะเสี่ยงต่อการเป็นโรคความดันโลหิตสูง โรคเบาหวาน โรคหัวใจขาดเลือด และโรคนิ่วในถุงน้ำดี แต่ในขณะเดียวกัน ผู้ที่ผอมเกินไป ก็จะมีเสี่ยงต่อการติดเชื้อ ประสิทธิภาพในการทำงานของร่างกายลดลง ดังนั้นควรรักษาระดับน้ำหนักให้อยู่ในเกณฑ์ปกติ

### 2.2.1 วิธีคำนวณดัชนีมวลร่างกาย Body Mass Index (BMI)

สูตรคำนวณดัชนีมวลร่างกายคือ [ดัชนีมวลร่างกาย = น้ำหนักตัว / ความสูง (เมตร) ยกกำลังสอง]

- ดัชนีมวลร่างกายเท่ากับ 40 หรือมากกว่านี้: โรคอ้วนขั้นสูงสุด
- 35.0 - 39.9: โรคอ้วนระดับ 2 เสี่ยงต่อการเกิดโรคที่มากับความอ้วน หากมีเส้นรอบเอวมากกว่าเกณฑ์ปกติจะเสี่ยงต่อการเกิดโรคสูง ต้องควบคุมอาหาร และออกกำลังกายอย่างจริงจัง
- 28.5 - 34.9: โรคอ้วนระดับ 1 และหากมีเส้นรอบเอวมากกว่า 90 ซม. (ชาย) 80 ซม.(หญิง) จะมีโอกาสเกิดโรคความดัน เบาหวานสูง จำเป็นต้องควบคุมอาหาร และออกกำลังกาย
- 23.5 - 28.4: น้ำหนักเกิน หากมีกรรมพันธุ์เป็นโรคเบาหวานหรือไขมันในเลือดสูงต้องพยายามลดน้ำหนักให้ดัชนีมวลร่างกายต่ำกว่า
- 23.5 - 23.4: น้ำหนักปกติ และมีปริมาณไขมันอยู่ในเกณฑ์ปกติ มักจะไม่ค่อยมีโรคภัย อุบัติการณ์ของโรคเบาหวาน ความดันโลหิตสูงต่ำกว่าผู้ที่อ้วนกว่านี้
- น้อยกว่า 18.5: น้ำหนักน้อยเกินไป ซึ่งอาจจะเกิดจากนักกีฬาที่ออกกำลังกายมาก และได้รับสารอาหารไม่เพียงพอ วิธีแก้ไขต้องรับประทานอาหารที่มีคุณภาพ และมีปริมาณพลังงานเพียงพอ และออกกำลังกายอย่างเหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2.2 อัตราการเผาผลาญพลังงาน Basal Metabolic Rate (BMR)

Basal Metabolic Rate (BMR) คือ อัตราการความต้องการเผาผลาญของร่างกายในชีวิตประจำวัน หรือจำนวนแคลอรีขั้นต่ำที่ต้องการใช้ในชีวิตแต่ละวัน ดังนั้นการคำนวณ BMR จะช่วยให้คุณคำนวณปริมาณแคลอรีที่ใช้ต่อวันเพื่อรักษาน้ำหนักปัจจุบันได้ และเมื่ออายุมากขึ้นเราจะควบคุมน้ำหนักได้ยากขึ้น เพราะ BMR เราลดลง การอดอาหารก็เป็นสาเหตุหนึ่งที่ทำให้ BMR ลดลง วิธีป้องกันคือ "หมั่นออกกำลังกาย" เพื่อเพิ่มประสิทธิภาพของการเผาผลาญ ซึ่งจะช่วยให้ BMR ไม่ลดลงเร็วเกินไป

BMR (Basal Metabolic Rate) พลังงานที่จำเป็นพื้นฐานในการมีชีวิต - กิโลแคลอรี

TDEE (Total Daily Energy Expenditure) พลังงานที่คุณใช้ในแต่ละวัน - กิโลแคลอรี

## 2.2.3 วิธีคำนวณการเผาผลาญพลังงาน Basal Metabolic Rate (BMR)

สูตรคำนวณอัตราการเผาผลาญของร่างกายในชีวิตประจำวันคือ

- สำหรับผู้ชาย :  $BMR = 66 + (13.7 \times \text{น้ำหนักตัวเป็น กก.}) + (5 \times \text{ส่วนสูงเป็น ซม.}) - (6.8 \times \text{อายุ})$
- สำหรับผู้หญิง :  $BMR = 665 + (9.6 \times \text{น้ำหนักตัวเป็น กก.}) + (1.8 \times \text{ส่วนสูงเป็น ซม.}) - (4.7 \times \text{อายุ})$

จะสังเกตได้ว่าน้ำหนัก ส่วนสูงและอายุมีผลต่อการเผาผลาญพลังงาน เมื่อหาค่า BMR (Basal Metabolic Rate) มาแล้วก็จะสามารถรู้ได้ว่าเรามีการการเผาผลาญพลังงานโดยไม่ทำกิจกรรมอะไรเลยเท่าไร แต่หากมีกิจกรรมอย่างออกกำลังกายจะมีการเผาผลาญพลังงานโดยคำนวณได้ดังนี้

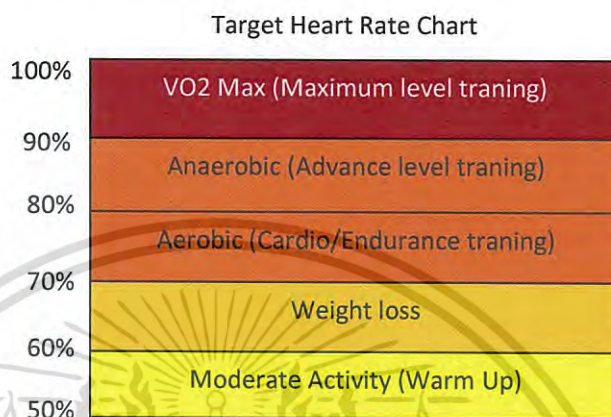
"การเผาผลาญพลังงานโดยปกติ =  $BMR \times \text{ตัวแปร}$ "

โดยตัวแปรของเราจะขึ้นอยู่กับกิจกรรมออกกำลังกายดังนี้

- นั่งทำงานอยู่กับที่ และไม่ได้ออกกำลังกายเลย =  $BMR \times 1.2$
- ออกกำลังกายหรือเล่นกีฬาเล็กน้อย ประมาณอาทิตย์ละ 1-3 วัน =  $BMR \times 1.375$
- ออกกำลังกายหรือเล่นกีฬานานกลาง ประมาณอาทิตย์ละ 3-5 วัน =  $BMR \times 1.55$
- ออกกำลังกายหรือเล่นกีฬาอย่างหนัก ประมาณอาทิตย์ละ 6-7 วัน =  $BMR \times 1.725$
- ออกกำลังกายหรือเล่นกีฬาอย่างหนักทุกวันเข้าเย็น =  $BMR \times 1.9$

## 2.3 อัตราการเต้นของหัวใจ

การเต้นของหัวใจ บ่งบอกว่าการออกกำลังกายในขณะนั้นอยู่ที่ความหนักระดับใดนั่นเอง ถ้าหากออกกำลังกายเกินกว่า Maximum heart rate ของหัวใจ อาจจะทำให้ช็อค หรือเสียชีวิตได้ Maximum heart rate คิดง่าย ๆ คือเท่ากับ 220 – อายุ ตัวอย่าง นาย A อายุ 20 :  $220-20=200$  จังหวะหัวใจใช้หน่วย bpm หรือ Beats per minute



รูปที่ 2.1 การเต้นของหัวใจกับการออกกำลังกาย

- Zone 1 50%-60% เป็นระดับที่ง่าย เป็นช่วงของการอุ่นเครื่อง การเผาผลาญมีเพียงเล็กน้อย และการเผาผลาญมาจากไขมันเพียงเล็กน้อยเท่านั้น
- Zone 2 60%-70% เริ่มมีความหนักมากขึ้น บางครั้งอาจเรียกว่า fat burning zone เชื่อว่าเป็นช่วงที่มีการเผาผลาญไขมันสูงมากขึ้นในช่วงนี้
- Zone 3 70%-80% เป็นระดับเรียกว่า cardio zone การเผาผลาญแคลอรีต่อชั่วโมงจะเพิ่มมากขึ้น และพลังงานครึ่งหนึ่งจะมาจากคาร์โบไฮเดรต และอีกครึ่งหนึ่งมาจากไขมัน
- Zone 4 80%-90% เป็นระดับที่เข้าสู่ anaerobic แคลอรีที่ถูกนำมาใช้จะเพิ่มมากขึ้น แต่ส่วนใหญ่จะเป็นพลังงานที่มาจากคาร์โบไฮเดรตมากกว่าไขมัน
- Zone 5 90%-100% เป็นระดับที่หนักมาก การเต้นของหัวใจอาจจะขึ้นไปถึงระดับ maximum heart rate พลังงานที่ใช้ได้จากคาร์โบไฮเดรต

ถ้าต้องการออกกำลังกายเพื่อรักษาความฟิตของร่างกาย คุณอาจจะใช้ช่วงของ 60-80% อัตราการเต้นของหัวใจควรอยู่ที่ 120-160 bpm

ถ้าต้องการออกกำลังกายเพื่อลดน้ำหนัก และเพิ่มความฟิตของร่างกาย อาจจะใช้ช่วง 70-90% อัตราการเต้นของหัวใจควรอยู่ที่ 140-180 bpm

สำหรับนักกีฬา หรือผู้ที่ออกกำลังกายสม่ำเสมอมากกว่า 6 เดือนอาจจะใช้ช่วง 80-100% อัตราการเต้นของหัวใจควรอยู่ที่ 160-200 bpm

## 2.4 อุปกรณ์สวมใส่ Wearable Device

Wearable Device [4] คือ อุปกรณ์สวมใส่ที่เข้ามามีบทบาทและเป็นอุปกรณ์ที่เข้ามาเกี่ยวข้องกับชีวิตประจำวัน ที่ไม่ใช่เป็นเพียงแค่เครื่องประดับที่ติดตัวแต่สามารถทำงานด้วยระบบคอมพิวเตอร์และเข้าถึงข้อมูลอินเทอร์เน็ต และ สมาร์ทโฟนได้ดีด้วย ซึ่งเป็นแรงผลักดันมาจากบรรดา Gadget ทั้งหลาย เช่นอุปกรณ์สวมใส่อัจฉริยะต่างๆ ที่กำลังได้รับความนิยมในช่วงนี้ เช่น Activity Trackers, Running Watches, Wearable Cameras, Smart Tracking และ Smart Watch เป็นต้น ซึ่งเมื่อนำมาสวมใส่แล้วจะมีความสามารถที่หลากหลาย ไม่ว่าจะเป็นในด้านการดูแลสุขภาพ เช่น

- สามารถนับก้าวเดินของว่าวันหนึ่งๆ เดินไปทั้งหมดกี่ก้าวแล้ว
- สามารถตรวจสอบก้าวที่วิ่งได้ เมื่อวิ่งออกกำลังกายและตรวจสอบว่าได้เผาผลาญไปกี่แคลอรี
- สามารถตั้งเพื่อตรวจสอบการพักผ่อนนอนหลับของเราได้ ว่านอนหลับลึกไม่ลึกแค่ไหน
- สามารถตั้งเป็นนาฬิกาปลุกตอนเช้าได้โดยการสั่นที่ข้อมือ
- สามารถตรวจสอบอาหารที่กินเข้าไปนั้นก็แคลอรี
- สามารถนำอุปกรณ์ที่มีความสามารถดังกล่าวมาเชื่อมกับที่เสียบหูฟังสมาร์ทโฟนเพื่อนำข้อมูลต่างๆ ของกิจกรรมในแต่ละวันมา Sync เก็บไว้ และนำมาแสดงผลในแอปพลิเคชันเพื่อเก็บไว้เป็นสถิติและแบ่งปันใน Social Network เพื่อให้ได้เห็นและติดตามได้เช่นเดียวกัน

### 2.4.1 ฟิตบิต (Fitbit)

Fitbit [5] เป็นอุปกรณ์วัดความแอคทีฟ หรือ Activity tracker เป็น Wearable Device ที่สามารถช่วยลดน้ำหนัก เผาผลาญไขมัน และเป็น Wearable Tech สำหรับออกกำลังกายอีกมากมายและสามารถนับก้าวที่เดิน แคลอรีที่เบิร์น เวลาที่ใช้ ตั้งเป้าหมายการออกกำลังกาย บันทึกข้อมูลได้ถึง 7 วัน ความแม่นยำสูงถึง 99% และสามารถ ติดตามผลลัพธ์การแอคทีฟในแต่ละวัน แซร์ข้อมูลผ่านโลกออนไลน์

### 2.4.2 ประเภทของ Fitbit

- ฟิตบิตเฟล็กซ์ (Fitbit Flex) สามารถเก็บสถิติการเดินก้าว ระยะทางที่เคลื่อนที่ ความก้าวหน้าของการเดินในแต่ละวัน พฤติกรรมการนอน เก็บการกินอาหาร และบอกอัตราการเผาผลาญ มีแบตเตอรี่ใช้ได้ถึง 5 วัน



รูปที่ 2.2 ฟิตบิตเฟล็กซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟิตบิตชาร์จ (Fitbit Charge) สามารถเก็บสถิติการเดินก้าว ระยะทางที่เคลื่อนที่ ความก้าวหน้าของการเดินในแต่ละวัน พฤติกรรมการนอน เก็บการกินอาหาร บอกอัตราการเผาผลาญของแคลอรี แจ็งเตือนสายที่โทรเข้ามาได้



รูปที่ 2.3 ฟิตบิตชาร์จ

- ฟิตบิตชาร์จเอชอาร์ (Fitbit Charge HR) สามารถเก็บสถิติการเดินก้าว ระยะทางที่เคลื่อนที่ ความก้าวหน้าของการเดินในแต่ละวัน พฤติกรรมการนอน เก็บการกินอาหาร บอกอัตราการเผาผลาญของแคลอรี แจ็งเตือนสายที่โทรเข้ามาได้ และสามารถวัดการเต้นของหัวใจเพื่อวัดอัตราการเผาผลาญของแคลอรีได้ดียิ่งขึ้น



รูปที่ 2.4 ฟิตบิตชาร์จเฮทอาร์

- ฟิตบิตเซอร์จ (Fitbit Surge) จะมีตัว GPS มาช่วยในการติดตามการก้าวเดิน ปีน ป่ายได้ดีขึ้น สามารถตรวจจับการเต้นของหัวใจได้ วัดการออกกำลังกายได้ทั้งวัน มีแบตเตอรี่ใช้ได้ถึง 7 วัน สามารถทำการแจ็งเตือนข้อความที่เข้ามาหรือเล่น เพลงได้ ตรวจจับการนอนหลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ฟิตบิตเซิร์จ

ตารางที่ 2.1 เปรียบเทียบความสามารถของอุปกรณ์วัดการทำกิจกรรมต่าง ๆ ของร่างกาย

|                          | ฟิตบิตเฟล็กซ์ | ฟิตบิตชาร์จ | ฟิตบิตชาร์จเฮท<br>อาร์ | ฟิตบิตเซิร์จ |
|--------------------------|---------------|-------------|------------------------|--------------|
| นับก้าวเดิน,เผาผลาญไขมัน | ✓             | ✓           | ✓                      | ✓            |
| นาฬิกา                   | -             | ✓           | ✓                      | ✓            |
| ระยะเวลาในการนอน         | ✓             | ✓           | ✓                      | ✓            |
| จับการนอนแบบอัตโนมัติ    | -             | ✓           | ✓                      | ✓            |
| การขึ้นบันได,การปีนป่าย  | -             | ✓           | ✓                      | ✓            |
| วัดจังหวะการเต้นของหัวใจ | -             | -           | ✓                      | ✓            |
| ดูหมายเลขโทรเข้า         | -             | ✓           | ✓                      | ✓            |
| การแจ้งเตือนข้อความ      | -             | -           | -                      | ✓            |
| ควบคุมเครื่องเล่นเพลง    | -             | -           | -                      | ✓            |
| ติดตาม GPS               | -             | -           | -                      | ✓            |

(ที่มา : [www.fitbit.com/th/compare](http://www.fitbit.com/th/compare))

## 2.5 การเข้าถึง Fitbit API

Fitbit [7] ใช้ OAuth 2.0 สำหรับผู้ใช้ที่ได้รับอนุญาต และ

### 2.5.1 การได้รับความยินยอม

Fitbit รองรับ Authorization Code Grant และ Implicit Grant Flows ตามที่กำหนดไว้ใน RFC 6749 Authorization Code Grant Flow เป็นที่แนะนำสำหรับการใช้งานที่มีบริการเว็บ การไหลนี้ต้องมีการสื่อสารเซิร์ฟเวอร์ไปยังเซิร์ฟเวอร์โดยใช้แอปพลิเคชันของลูกค้ำที่เป็นความลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อประกอบการพิจารณาการรักษาความปลอดภัยในหน้าเว็บอนุญาต OAuth 2.0 จะต้องเสนอในมุมมองที่เบราว์เซอร์จัดไว้ให้ ผู้ใช้ Fitbit สามารถยืนยันว่าพวกเขาได้รับการตรวจสอบกับเว็บไซต์ Fitbit.com ของแท้ ถ้าพวกเขามีเครื่องมือที่เบราว์เซอร์มีให้ เช่นแถบ URL และข้อมูลใบรับรอง Transport Layer Security (TLS)

สำหรับ Native Applications ในที่นี้หมายถึงหน้าอนุญาตต้องเปิดด้วยเบราว์เซอร์เริ่มต้น Native Applications สามารถใช้รูปแบบ URL ที่กำหนดเองแบบ Callback URIs เพื่อเปลี่ยนเส้นทางผู้ใช้กลับมาจากเบราว์เซอร์ สู่การร้องขอของแอปพลิเคชันที่ได้รับอนุญาต

แอปพลิเคชัน Android อาจจะใช้ Chrome Custom Tabs แทนการสลับแอปพลิเคชันในเบราว์เซอร์เริ่มต้น การใช้ WebView เป็นสิ่งต้องห้ามสำหรับการใช้งานเว็บไม่ควรใช้ iframe การใช้งานเว็บอาจใช้หน้าต่างแบบ Pop-Up トラバイトที่แถบ URL สามารถมองเห็นได้

## 2.5.2 หน้าการในการให้สิทธิ์

เปลี่ยนเส้นทางผู้ใช้ไปยังหน้าการในการให้สิทธิ์ Fitbit OAuth 2.0 ที่มีพารามิเตอร์ต่อไปนี้

### ตารางที่ 2.2 พารามิเตอร์ URI

| พารามิเตอร์ URI                      | รายละเอียด  |
|--------------------------------------|---|
| client_id<br>จำเป็นต้องใส่ใน URI     | นี่คือ Fitbit API แอป ID จากการตั้งค่าผ่าน dev.Fitbit.com<br>ประเภท : String  |
| response_type<br>จำเป็นต้องใส่ใน URI | code สำหรับ Authorization Code Grant Flow<br>token สำหรับ Implicit Grant Flow<br>ประเภท : String  |
| Scope<br>จำเป็นต้องใส่ใน URI         | รายชื่อที่คั่นด้วยช่องว่างของสิทธิ์ที่ร้องขอ<br>ประเภท : String   |
| redirect_uri<br>จำเป็นต้องใส่ใน URI  | กรณีที่ Fitbit ควรส่งผู้ใช้หลังจากผู้ใช้มอบสิทธิ์หรือปฏิเสธการได้รับความยินยอมเป็นทางเลือก, ถ้าได้ระบุไว้เพียงหนึ่ง Callback URI สำหรับแอปพลิเคชันในการตั้งค่าใน dev.Fitbit.com อย่างอื่น<br>จำเป็นต้องมี<br>ประเภท : URI |

ตารางที่ 2.2 พารามิเตอร์ URI (ต่อ)

| พารามิเตอร์ URI | รายละเอียด  |
|-----------------|---|
| expires_in      | ใช้สำหรับ Implicit Grant Flow เท่านั้น เลือกอายุการใช้งานการเข้าถึงโทเค็นที่ต้องการก่อน ค่าตั้งต้นที่ 86400 (86400 สำหรับ 1 วัน 604800 สำหรับ 1 สัปดาห์ และ 2592000 สำหรับ 30 วัน)<br>ประเภท : String   |
| prompt          | โดยระบุ ถ้าต้องการที่จะบังคับให้รับรองความถูกต้อง Fitbit หรือหน้าอนุญาต OAuth 2.0 จะแสดงขึ้น none สำหรับการทำงานเริ่มต้น login เพื่อบังคับให้ผู้ใช้เข้าสู่ระบบ consent เพื่อบังคับให้ผู้ใช้อนุมัติแอปพลิเคชันอีกครั้งเป็นทางเลือก<br>ประเภท : String  |
| state<br>แนะนำ  | ให้สถานะใด ๆ ที่อาจจะมีประโยชน์ในการใช้แอปพลิเคชัน เมื่อผู้ใช้ถูกนำกลับไปสู่แอปพลิเคชัน พารามิเตอร์นี้จะถูกเพิ่มเข้าไปในการเปลี่ยนเส้นทางของ URI ตามแอปพลิเคชันของคุณที่ระบุ Fitbit ขอแนะนำรวมทั้งโทเค็นที่ต่อต้านการปลอมแปลงในพารามิเตอร์นี้ และยืนยันความคุ้มค่าในการเปลี่ยนเส้นทางที่จะลดการต่อต้านการปลอมแปลงคำขอข้ามเว็บไซต์ (CSRF)<br>เป็นทางเลือก<br>ประเภท : String |

ค่าพารามิเตอร์ทั้งหมดจะต้องเข้ารหัส URI

(ที่มา : [dev.fitbit.com/docs/oauth2](https://dev.fitbit.com/docs/oauth2))

### 2.5.3 Authorization Code Grant Flow

Fitbit ตามข้อกำหนดของ OAuth 2.0 Authorization Code Grant ที่ระบุไว้ใน RFC 6749. Fitbit ขอแนะนำให้นักพัฒนาตรวจสอบข้อกำหนดและใช้ไลบรารีในสไลด์เอนท์แบบ OAuth 2 สำหรับภาษาเขียนโปรแกรมที่นักพัฒนาใช้

Authorization Code Grant Flow มีขั้นตอนดังต่อไปนี้

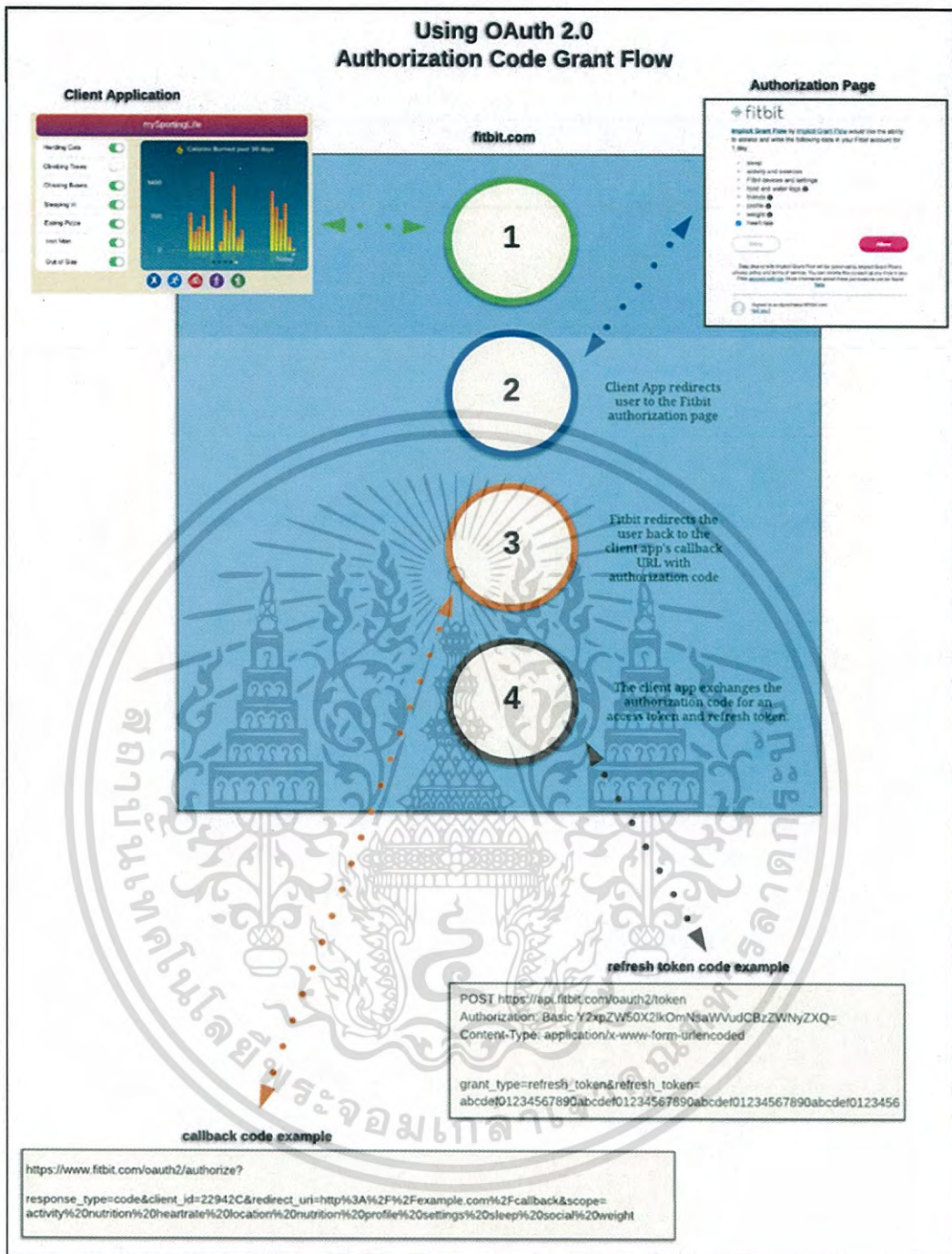
- 1) แอปพลิเคชันของนักพัฒนาจะเปลี่ยนเส้นทางของผู้ใช้ไปยังหน้าการให้สิทธิ์ของ Fitbit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) ในการรับความยินยอมจากผู้ใช้ Fitbit จะเปลี่ยนเส้นทางผู้ใช้กลับไปยัง Application Callback URL ด้วยรหัสที่ใช้ในการอนุญาต เป็นพารามิเตอร์แบบ URL
- 3) แอปพลิเคชันของนักพัฒนาจะแลกเปลี่ยนรหัสที่ใช้ในการอนุญาต สำหรับ Access Token และ Refresh Token
- 4) แอปพลิเคชันของนักพัฒนา จะกักเก็บ Access Token และ Refresh Token โดยจะใช้ Access Token เพื่อทำการร้องขอไปยัง Fitbit API โดยจะใช้ Refresh Token เพื่อรับ Access Token แบบใหม่เมื่อ Access Token เดิมหมดอายุลง โดยปราศจากการแจ้งเตือนซ้ำเพื่อให้ผู้ใช้รับทราบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 แสดงถึงการใช OAuth 2.0 Authorization Code Grant Flow

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5.4 Implicit Grant Flow

Fitbit ตามข้อกำหนดของ OAuth 2.0 Implicit Code Grant ที่ระบุไว้ใน RFC 6749. Fitbit ขอแนะนำให้นักพัฒนาตรวจสอบข้อกำหนดและใช้ไลบรารีในส่วนโคลเอนท์แบบ OAuth 2 สำหรับภาษาเขียนโปรแกรมที่นักพัฒนาใช้

Implicit Code Grant Flow มีขั้นตอนดังต่อไปนี้

- 1) แอปพลิเคชันของนักพัฒนาจะเปลี่ยนเส้นทางของผู้ใช้ไปยังหน้าการให้สิทธิ์ของ Fitbit
- 2) ในการรับความยินยอมจากผู้ใช้ Fitbit จะเปลี่ยนเส้นทางผู้ใช้กลับไปยัง Application Callback URL ด้วย Access Token โดยเป็น URL แบบแบ่งส่วน
- 3) แอปพลิเคชันของนักพัฒนา จะกักเก็บ Access Token โดยจะใช้ Access Token เพื่อทำการร้องขอไปยัง Fitbit API





Access Token มีทั้ง 1 วัน 1 สัปดาห์ และ 30 วัน แอปพลิเคชันสามารถเลือกอายุการใช้งานของ Token ได้ล่วงหน้า แต่สุดท้าย ผู้ใช้จะตัดสินใจเอง

ถ้าแอปพลิเคชันใช้ Implicit Grant Flow ส่งผู้ใช้ไปยังหน้าการอนุมัติ ก่อนที่ Access Token ก่อนหน้านี้อาจหมดอายุลง ผู้ใช้จะไม่ได้รับการแจ้งเตือนแต่ขอบเขตการใช้งานจะเพิ่มขึ้น ผู้ใช้จะถูกเปลี่ยนเส้นทางทันทีเพื่อไปยังแอปพลิเคชันด้วยกับ Access Token

### 2.5.5 การกำหนดข้อมูล

แอปพลิเคชันจะต้องร้องขอคำอนุญาตเพื่อทรัพยากรที่ตั้งใจจะเข้าถึงและแก้ไข OAuth 2.0 อ้างอิงคำอนุญาตเหล่านี้เป็นการกำหนดข้อมูล ที่ปลายทางของ Fitbit API ทั้งหมดจำเป็นต้องกำหนดข้อมูลมากกว่าหนึ่งอย่าง ซึ่งได้ระบุไว้ในด้านล่าง

แอปพลิเคชันจะต้องระบุรายชื่อของขอบเขตเมื่อเปลี่ยนเส้นทางผู้ใช้ไปยังหน้าการอนุญาต โทเค็นการเข้าถึงที่ออกมาจะมีขอบเขตแอปพลิเคชันที่ร้องขอ

#### ตารางที่ 2.3 ขอบเขตของแอปพลิเคชัน

| ข้อมูลกิจกรรมใน Fitbit | คำอธิบาย   |
|------------------------|--|
| activity               | การกำหนดข้อมูลกิจกรรมนั้นรวมถึงข้อมูลกิจกรรมและข้อมูลบันทึกการออกกำลังกายที่เกี่ยวข้องกัน เช่น การก้าวเดิน ระยะทาง แคลอรีที่เผาผลาญ และเวลาที่ใช้งาน |
| heartrate              | การกำหนดข้อมูลการเต้นของหัวใจนั้นรวมถึงข้อมูลการเต้นของหัวใจอย่างต่อเนื่องและการวิเคราะห์ที่เกี่ยวข้อง   |
| location               | การกำหนดข้อมูลสถานที่รวมถึง GPS และข้อมูลสถานที่อื่นๆ  |
| Nutrition              | การกำหนดข้อมูลของโภชนาการรวมถึงแคลอรีที่ได้รับและโภชนาการที่มีคุณสมบัติที่เกี่ยวข้องกัน เช่น ข้อมูลบันทึกอาหารและน้ำ เป้าหมายและแผน                  |
| profile                | การกำหนดข้อมูลข้อมูลส่วนตัวคือข้อมูลพื้นฐานของผู้ใช้   |
| Settings               | การกำหนดข้อมูลการตั้งค่ารวมถึงบัญชีผู้ใช้และการตั้งค่าอุปกรณ์ อย่างเช่น สัญญาณเตือน  |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 ขอบเขตของแอปพลิเคชัน (ต่อ)

| ข้อมูลกิจกรรมใน Fitbit | คำอธิบาย   |
|------------------------|--|
| Sleep                  | การกำหนดข้อมูลการนอนหลับนั้นรวมถึงข้อมูลบันทึกการนอนหลับ และการวิเคราะห์การนอนหลับที่เกี่ยวข้อง                              |
| Social                 | การกำหนดข้อมูลของสังคมนั้นรวมถึง ข้อมูลที่เกี่ยวข้องกับเพื่อน อย่างเช่น รายชื่อเพื่อน การเชิญชวน และลีดเดอร์บอร์ด            |
| Weight                 | การกำหนดข้อมูลของน้ำหนักนั้นรวมถึงน้ำหนัก และข้อมูลที่เกี่ยวข้อง อย่างเช่น ดัชนีมวลกาย เปอร์เซ็นต์ไขมันในร่างกาย และเป้าหมาย |

ค่าพารามิเตอร์ทั้งหมดต้องเข้ารหัส URI

(ที่มา : [dev.fitbit.com/docs/oauth2](https://dev.fitbit.com/docs/oauth2))

จาก the OAuth 2.0 Beta Program ข้อมูล activity nutrition profile settings sleep social และ weight นั้นต้องการกำหนดข้อมูลและมีค่าเท่ากับการเข้าถึง OAuth 1.0a ถ้าไม่ได้รับอนุญาตกำหนดข้อมูลเหล่านี้ มันจะไปอยู่ที่ส่วนท้ายของรายชื่อที่ระบุ Fitbit ยังทำการปรับปรุงปลายทางที่ยังมีอยู่ทั้งหมดเพื่อสนับสนุนตัวกำหนดข้อมูลที่เกี่ยวข้อง ตัวกำหนดข้อมูลจะถูกลบออกจากรายการที่ต้องการเมื่อปลายทางที่เกี่ยวข้องนั้นได้ทำการปรับปรุง การกำหนดข้อมูลทั้งหมดจะเป็นตัวเลือกในอนาคต

2.5.6 โหมดการเข้าถึงแอปพลิเคชัน

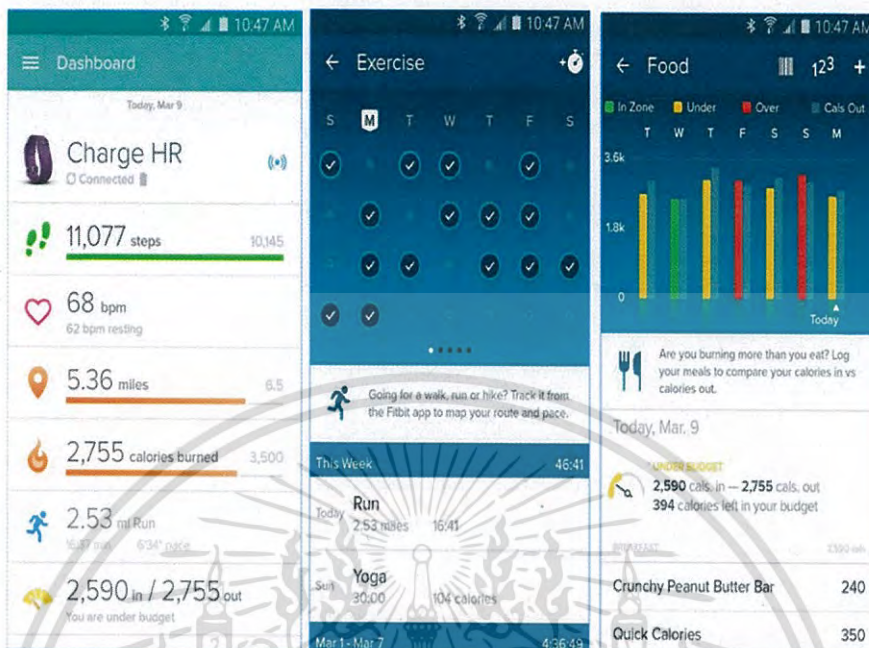
แอปพลิเคชันต่างๆปฏิบัติการมีตัวอ่านข้อมูลอย่างเดียวหรือการจัดการทั้งอ่านและทั้งเขียนข้อมูล นี่เป็นการตั้งค่าในระดับแอปพลิเคชันและมีผลในทุกๆการกำหนดข้อมูล ซึ่งสามารถทำการแก้ไขได้ในการตั้งค่าแอปพลิเคชันบน dev.Fitbit.com

2.5.7 การตอบกลับของ URIs

เมื่อผู้ใช้ยินยอมหรือห้ามการเข้าถึงจากหน้าจอรหัสสิทธิ์ URIs จะเปลี่ยนเส้นทางกลับไปที่แอปพลิเคชัน ซึ่งจะสามารถระบุการตอบกลับ URI ได้ในรูปแบบได้ในการตั้งค่าแอปพลิเคชันบน dev.Fitbit.com แอปพลิเคชันอาจจะมีการตอบกลับของ URIs ที่ลงทะเบียนแล้วหลายตัวที่พืทบิทแนะนำว่าให้ระบุการตอบกลับ URI ที่ตั้งใจ ให้เป็น redirect\_uri เมื่อส่งผู้ใช้ไปยังหน้ารหัสสิทธิ์

## 2.5.8 แอปพลิเคชันที่พัฒนาโดยใช้ Fitbit API

### 2.5.8.1 Fitbit



รูปที่ 2.8 แอปพลิเคชัน Fitbit

แอปพลิเคชัน Fitbit ถูกออกแบบมาเพื่อทำงานร่วมกับการติดตามกิจกรรม ของฟิตบิต ให้รองรับกิจกรรมต่างๆ ได้ดีขึ้นผ่านสมาร์ทโฟน

- เชื่อมข้อมูลแบบไร้สาย ระหว่างฟิตบิตกับอุปกรณ์ Android ของผู้ใช้
- ติดตามเป้าหมายประจำวันของผู้ใช้และความคืบหน้าสำหรับการก้าวเดิน, ระยะทาง, การเผาผลาญแคลอรีและอื่น ๆ ตลอดเวลา
- ใช้จีพีเอสในการติดตามการวิ่ง, การเดินและการปั่นเขาของผู้ใช้
- เก็บรวบรวมกิจกรรมต่างๆ เช่น โยคะและการออกกำลังกายกลางแจ้ง
- บันทึก รายการอาหารที่รับประทานเพื่อดูจำนวนแคลอรีที่คุณรับประทานในแต่ละวันและการจัดการน้ำหนักของคุณตลอดเวลา
- ความท้าทายและเปรียบเทียบสถิติในชีวิตประจำวันหรือรายสัปดาห์กับเพื่อน
- ติดตามอัตราการเต้นของหัวใจทุกวัน กับ ฟิตบิตชาร์จเฮทอาร์และฟิตบิตเซิร์จ

#### ข้อดีของแอปพลิเคชัน Fitbit

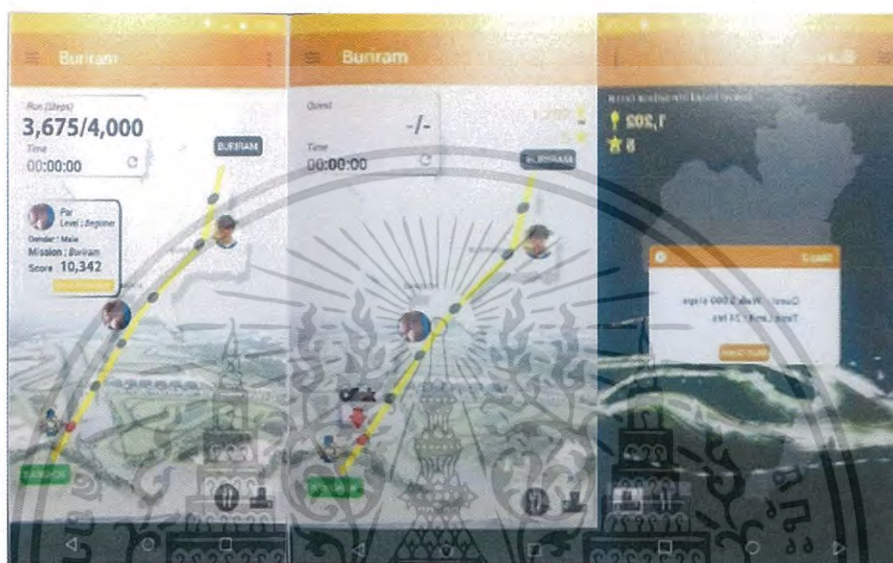
- 1) มีการเชื่อมต่อกับเพื่อนที่ใช้ ฟิตบิตด้วยกัน
- 2) ใช้การติดต่อระหว่างแอปกับตัวฟิตบิตแบบตามเวลาจริง
- 3) สามารถส่งข้อความพูดคุยกับเพื่อนผ่านแอปพลิเคชัน
- 4) ใช้ได้กับระบบปฏิบัติการแอนดรอยด์ได้ทุกรุ่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ข้อจำกัดของแอปพลิเคชัน Fitbit

- 1) การทำหายเพื่อนยังไม่เป็นที่น่าสนใจ
- 2) รายการอาหารในแอปพลิเคชันนี้ มีเพียงภูมิภาคแถบยุโรปเท่านั้น
- 3) ใช้ได้กับอุปกรณ์ฟิตบิตเพียงอย่างเดียว

#### 2.5.8.2 FitGaming



รูปที่ 2.9 แอปพลิเคชัน FitGaming

FitGaming [15] เป็นแอปที่นำข้อมูลการก้าวเดินจากอุปกรณ์ Fitbit มาใช้ในการเล่นเกม โดยจะต้องก้าวเดินให้ได้จำนวนที่กำหนดในแต่ละด่าน ซึ่งจะมีการรวมด่านหลาย ๆ ด่านเข้าเป็นภูมิภาค (ภาคเหนือ, ภาคตะวันออกเฉียงเหนือ, ภาคกลาง, ภาคใต้) โดยมีการอธิบายถึงวัฒนธรรมในแต่ละภาคด้วย อีกทั้งยังมีการรวบรวมรายชื่ออาหารไทยไว้ในแอปนี้ นอกจากนี้ยังเชื่อมต่อในหมู่สังคมเพื่อนที่ใช้ Fitbit ด้วยกันได้อีกด้วย

#### ข้อดีของ FitGaming

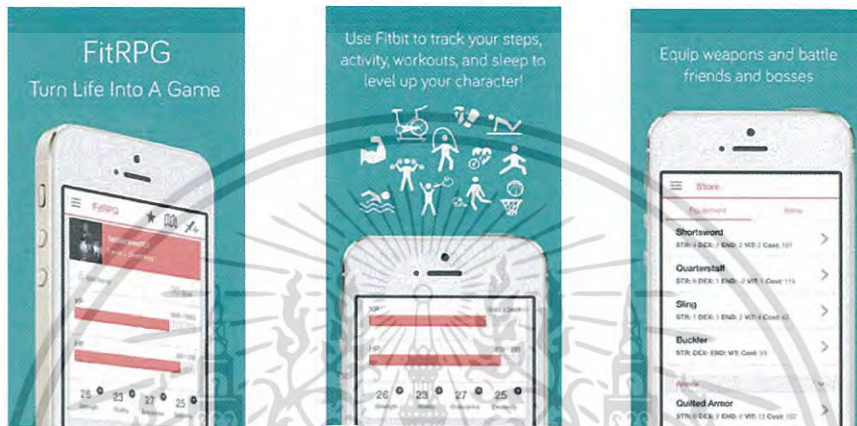
- 1) เมนูเกมที่ไม่ซับซ้อน แม้เริ่มเล่นก็สามารถเข้าใจได้ง่าย
- 2) ระบบการเล่นที่ง่ายมาก เหมาะสำหรับผู้เล่นทุกวัย
- 3) มีกราฟิกที่ชวนให้ดูน่าเล่น
- 4) มีการเชื่อมต่อกับเพื่อนที่ใช้ Fitbit ด้วยกัน
- 5) มีข้อมูลของอาหารไทยและสถานที่ท่องเที่ยวไทย ในขณะที่แอปพลิเคชันเกมอื่น ๆ ที่ใช้ฟิตบิตในการเล่น ไม่มีแบบนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ข้อจำกัดของ FitGaming

- 1) ด้วยระบบการเล่นที่ง่ายมาก ความท้าทายในการเล่นระยะยาวจึงลดลง
- 2) ข้อมูลอาหารไทยในแอปที่มีอยู่ กลับไม่ได้ใช้ในการเล่นหรือคำนวณค่าต่างๆ ภายในเกม ทั้งที่ควรจะเป็น
- 3) ไม่รองรับ ระบบปฏิบัติการแอนดรอยด์รุ่นที่ต่ำกว่า 5.0

#### 2.5.8.3 FitRPG



รูปที่ 2.10 แอปพลิเคชัน FitRPG

FitRPG [6] เป็นแอปที่จะเปลี่ยนข้อมูลการออกกำลังกายของผู้เล่น ให้เป็นค่าความแข็งแรงของตัวละคร, ค่า HP, ค่าความอดทน, ค่าความชำนาญและค่าประสบการณ์ ในภารกิจหรือออกกำลังกายหรือมีส่วนร่วมในการต่อสู้เพื่อเพิ่มระดับ นอนเพื่อฟื้นฟูค่า HP ของผู้เล่น เข้าสู่ระบบการออกกำลังกายเพื่อเพิ่มความแข็งแรงและค่าความชำนาญของผู้เล่น เพิ่มขึ้นตอนของผู้เล่นเพื่อปรับปรุงค่าประสบการณ์และค่าความอดทนของผู้เล่นเอง ใช้เวลาออกกำลังกายไปอีกระดับหนึ่งอย่างแท้จริง

- การแข่งขัน : ต่อสู้กับเพื่อนของผู้เล่น Fitbit จะขโมยทองและค่าประสบการณ์ของเพื่อนผู้เล่น ผู้เล่นส่วนใหญ่ที่แข็งแรงมักจะเป็นผู้ชนะ โดยดูวิธีการจัดอันดับกับผู้เล่นคนอื่นๆ ในตารางผู้นำ
- แรงจูงใจในการเล่น : ในภารกิจสำหรับผู้เล่นคนเดียว มีแบบจับเวลาไปจนถึงการวิ่งแบบมาราธอน ถ้าคุณชนะและประสบความสำเร็จจะได้รับทองและค่าประสบการณ์เป็นของรางวัล และจะสูญเสียทองถ้าคุณล้มเหลว
- ความอยากที่จะเล่นเกม FitRPG : การเล่นอย่างต่อเนื่องกับต่อสู้กับบอสเพื่อเพิ่มความสามรถ และความพยายามที่จะไต่อันดับขึ้น ทำให้ผู้เล่นมีความอยากที่จะเล่น FitRPG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ใช้ทองของผู้เล่นซื้ออาวุธและอุปกรณ์อื่นๆ ใช้ยาเพื่อที่จะคืนค่า HP และอาวุธเพื่อเพิ่มโอกาสของผู้เล่นที่จะชนะการต่อสู้กับเพื่อนของผู้เล่นเอง
- เข้าสู่ระบบด้วยบัญชี Fitbit ของผู้เล่นที่มีอยู่ และเริ่มเล่นได้ทันที
- สามารถใช้ร่วมกับ Jawbone และอุปกรณ์วัดการออกกำลังกายแบบอื่น ๆ ได้

#### ข้อดีของแอป FitRPG

- 1) เมื่อมีบัญชีผู้ใช้ของ Fitbit แล้ว สามารถเข้าเล่นได้ทันที
- 2) เป็นแอปพลิเคชันที่ไม่มีโฆษณา
- 3) ใช้ร่วมกับอุปกรณ์ออกกำลังกายอื่นๆ ได้ด้วย เช่น Jawbone
- 4) มีภารกิจในเกมให้เล่นอย่างหลากหลาย
- 5) ใช้แอปพลิเคชันนี้บนระบบปฏิบัติการ Android ได้ตั้งแต่เวอร์ชัน 2.3.3 ขึ้นไป

#### ข้อจำกัดของแอป FitRPG

- 1) ใช้ตัวหนังสือเดินเกมเป็นหลัก ตามสไตล์เกมแนว RPG จึงเข้าใจยากสำหรับผู้ที่หัดเล่น
- 2) ไม่มีกราฟิกใดๆ จึงไม่น่าดึงดูดให้ชวนเล่น

## 2.6 มายเอสคิวแอล (MySQL)

MySQL [9] เป็นซอฟต์แวร์โอเพนซอร์สใช้จัดการฐานข้อมูลเชิงสัมพันธ์ โดยใช้ภาษา SQL ที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่น เช่น การทำงานร่วมกับเครื่องบริการเว็บ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษา PHP ภาษา ASP.NET ภาษา JSP หรือทำงานร่วมกับโปรแกรมประยุกต์ เช่น ภาษาวิซวลเบสิกดอทเน็ต (VB.NET) ภาษาจาวา (JAVA) ภาษาซีชาร์ป (C#) โปรแกรมถูกออกแบบมาให้สามารถทำงานได้บนระบบปฏิบัติการที่หลากหลาย

## 2.7 เว็บเซิร์ฟเวอร์ (Web server)

เว็บเซิร์ฟเวอร์ (Web server) [10] คือเครื่องคอมพิวเตอร์ ที่ทำการติดตั้งซอฟต์แวร์ที่เกี่ยวข้องกับเซิร์ฟเวอร์ที่ทำหน้าที่บริการและส่งข้อมูลให้กับไคลเอนต์ ที่ทำการร้องขอข้อมูลเว็บเพจ โดยเว็บเซิร์ฟเวอร์จะทำการตอบรับ (Response) เพื่อส่งข้อมูลหน้าเว็บเพจที่ไคลเอนต์ร้องขอกลับไปผ่านพอร์ต 80 (http) หรือ พอร์ต 443 (https) โดยจะมีชุดโปรแกรมที่ใช้พัฒนาแอปพลิเคชันดังนี้

### 2.7.1 XAMPP

XAMPP [13] คือ ชุดโปรแกรมที่เอาไว้จำลองเว็บเซิร์ฟเวอร์เพื่อทดสอบสคริปต์หรือเว็บเพจในเครื่อง โดยที่ไม่ต้องเชื่อมต่ออินเทอร์เน็ตและไม่ต้องมีค่าใช้จ่าย ง่ายต่อการ

ติดตั้ง เพราะได้รวม Apache ทำงานเป็นเว็บเซิร์ฟเวอร์ MySQL ระบบฐานข้อมูล สนับสนุนฐานข้อมูล MySQL และ SQLite PHP สำหรับการพัฒนาเว็บแอปพลิเคชัน phpMyAdmin ใช้จัดการและเชื่อมต่อฐานข้อมูลทำให้มีความสะดวกต่อการใช้งาน

### 2.7.2 อาปาเช (Apache)

อาปาเช (Apache) [11] คือ ซอฟต์แวร์สำหรับเปิดให้บริการเซิร์ฟเวอร์บนโปรโตคอล HTTP สามารถทำงานได้บนหลายระบบปฏิบัติการ ได้รับการพัฒนาจาก HTTPD Web Server ทำหน้าที่ในการจัดเก็บโฮมเพจและส่งโฮมเพจไปยังเบราว์เซอร์ที่มีการร้องขอมายังเว็บเซิร์ฟเวอร์ที่เก็บโฮมเพจนั้นอยู่ อาปาเช เป็นซอฟต์แวร์แบบโอเพนซอร์สที่อนุญาตให้บุคคลทั่วไปสามารถเข้ามาร่วมพัฒนาส่วนต่างๆของอาปาเชได้ ซึ่งทำให้มีการสร้างโมดูลที่มีประโยชน์มากมาย เช่น โมดูลที่ใช้กับ ภาษา perl ,python และphp (mod\_perl ,mod\_python ,mod\_php) ตามลำดับ เป็นต้น ซึ่งทำให้อาปาเชสามารถใช้ประโยชน์และทำงานร่วมกับภาษาอื่นแทนที่จะเป็นเพียงเซิร์ฟเวอร์ที่ให้บริการ HTML เพียงอย่างเดียว นอกจากนี้ อาปาเชเองยังมีความสามารถด้านอื่นๆด้วย เช่น การยืนยันตัวบุคคล (mod\_auth, mod\_access, mod\_digest) หรือเพิ่มความปลอดภัยในการสื่อสารผ่านโปรโตคอล https (mod\_ssl) และยังมีโมดูลอื่นๆ ที่ได้รับความนิยมใช้เช่น mod\_vhost ทำให้สามารถสร้างโฮสต์เสมือนภายในเครื่องเดียวกันได้ หรือ mod\_rewrite ซึ่งเป็นเครื่องมือที่จะช่วยให้ url ของเว็บนั้นอ่านง่ายขึ้น

### 2.7.3 พีเอชพี (PHP)

ภาษาพีเอชพี [12] เป็นโปรแกรมประเภทสคริปต์ (script) ที่ใช้ในเว็บเซิร์ฟเวอร์ (Server-Side script language หรือ HTML-embedded scripting language) ทุกครั้งก่อนที่เครื่องคอมพิวเตอร์ซึ่งให้บริการเป็นเว็บเซิร์ฟเวอร์จะส่งหน้าเว็บเพจที่เขียนด้วยพีเอชพีมาให้ฝั่งไคลเอนต์ (เบราว์เซอร์) จะต้องมีการประมวลผลตามคำสั่งที่มีอยู่ในสคริปต์ให้เสร็จก่อนแล้วถึงส่งผลลัพธ์ที่ได้ให้กับฝั่งไคลเอนต์ที่เรียกไปยังหน้าเว็บเพจนั้น ผลลัพธ์ที่ได้นั้นคือเว็บเพจที่ผู้ใช้เห็นนั่นเอง ดังนั้นพีเอชพีจัดเป็นเครื่องมือที่สำคัญชนิดหนึ่งที่ช่วยให้สามารถสร้าง Dynamic Web Pages (เว็บเพจที่มีการโต้ตอบกับผู้ใช้) ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น

การแสดงผลของพีเอชพี จะปรากฏในลักษณะเอชทีเอ็มแอลซึ่งจะไม่แสดงคำสั่งที่ผู้ใช้เขียน ซึ่งเป็นลักษณะเด่นที่พีเอชพีแตกต่างจากภาษาอื่นในลักษณะไคลเอนต์ไซด์สคริปต์ เช่น ภาษาจาวาสคริปต์ที่ผู้ชมเว็บไซต์สามารถอ่าน ดู และคัดลอก คำสั่งไปใช้เอง

## 2.8 เจสัน (JSON)

JSON หรือ Java Script Object Notation [14] เป็นวิธีการที่ทำให้ JavaScript แลกเปลี่ยนข้อมูลกับ Server ได้อย่างง่ายดาย JSON ถูกสร้างขึ้นจากชุดข้อมูลของ literal object notation ใน JavaScript JSON จะใช้ [] แทน array และใช้ {} แทน hash (หรือ associate array) แต่ละสมาชิกคั่นด้วย comma (,) และแต่ละ ชื่อสมาชิกคั่นด้วย colon (:)

### 2.8.1 JSON สามารถสร้างได้ 2 รูปแบบ คือ

- 1) การจัดเก็บในชุดข้อมูลที่มีชื่อข้อมูลและข้อมูลคู่กัน ในภาษาต่างๆ ข้อมูลจะจัดอยู่ในรูปแบบ ของ Object, record, struct, dictionary, hash table, keyed list หรือ associative array
- 2) ลำดับของค่าข้อมูล ในภาษาโปรแกรมส่วนใหญ่ จะจัดอยู่ในรูปแบบของ array, vector, list หรือ sequence

ใน JSON มีหลายมีรูปแบบต่างๆเหล่านี้ (JSON format)

- 1) Object เป็นชุดของข้อมูลที่มีชื่อข้อมูลและค่าของข้อมูลนั้นคู่กัน ซึ่งจะถูกเริ่มต้นด้วยเครื่องหมาย { และจะปิดท้ายข้อมูลด้วยเครื่องหมาย } ข้อมูลแต่ละค่าจะมีเครื่องหมาย : กำกับระหว่างชื่อข้อมูลกับค่าของข้อมูล และแต่ละข้อมูลจะมีเครื่องหมาย , คั่น
- 2) Array เป็นลำดับของข้อมูล ซึ่งจะถูกเริ่มต้นด้วยเครื่องหมาย [ และจะจบด้วยเครื่องหมาย ] แต่ละค่าของข้อมูลจะถูกคั่นด้วยเครื่องหมาย ,
- 3) Value เป็น String ที่อยู่ในเครื่องหมาย “ ” หรือตัวเลข หรือค่าทางตรรกศาสตร์ true, false หรือค่า null หรือ object หรือ array ซึ่งโครงสร้างสามารถวางซ้อนกันได้
- 4) String เป็นลำดับของตัวอักษรตั้งแต่ 0 ตัวอักษรหรือมากกว่า ซึ่งอยู่ภายใต้เครื่องหมาย “ ” และจะใช้เครื่องหมาย ในการใส่เครื่องหมายกำกับต่างๆ ซึ่งจะมีลักษณะคล้ายกับ String ในภาษา C หรือภาษา Java
- 5) Number นั้นมีความคล้ายคลึงกับ Number ในภาษา C หรือภาษา Java อย่างมาก แต่จะไม่สามารถใช้เลขฐาน 8 กับเลขฐาน 16 ได้
- 6) ช่องว่าง (Whitespace) สามารถที่จะใส่ไว้ระหว่างสัญลักษณ์ต่างๆได้ ยกเว้นรายละเอียดซึ่งเข้ารหัสที่สมบูรณ์ในการบรรยายภาษาต่างๆ

### 2.8.2 โครงสร้างของ JSON

JSON นั้นใช้ลักษณะภาษาของ JavaScript แต่ไม่ถูกมองว่าเป็นภาษาโปรแกรม กลับถูกมองว่าเป็นภาษาในการแลกเปลี่ยนข้อมูลมากกว่า ในปัจจุบันมีไลบรารีของภาษาโปรแกรมอื่นๆ ที่ใช้ประมวลผลข้อมูลในรูปแบบ JSON มากมาย

### ตัวอย่างโค้ดของ JSON 2.1

```
[
  {"firstname":"name","lastname":"name"},
  {"firstname":"name1", "lastname":"name2"}
]
```

JSON นั้นยังสามารถจัดเก็บข้อมูลที่เป็นลักษณะของ Master-Detail ได้อีก

### ตัวอย่างโค้ดของ JSON 2.2

```
[
  {"firstname" : "name",
    "lastname": "name",
    "address" : [
      {
        "address1" : "address",
        "province" : "bangkok",
        "country" : "Thailand"
      }
    ]
  }
]
```

## 2.9 LibGDX

LibGDX [32,33] คือ เฟรมเวิร์กที่ใช้ในการพัฒนาเกมที่สามารถเล่นได้ทุกแพลตฟอร์ม ซึ่งแพลตฟอร์มที่รองรับในปัจจุบันคือ วินโดวส์ ลินุกซ์ แมคโอเอสทีเอ็น แอนดรอยด์ (ตั้งแต่เวอร์ชัน 2.2 ขึ้นไป) แบล็คเบอร์รี่ ไอโอเอส จาวาแอปเพล็ตที่ลง JVM และจาวาสคริปต์ หรือ WebGL

แอปพลิเคชันที่พัฒนาขึ้นโดยใช้ LibGDX นำไปใช้งานในแพลตฟอร์มต่างๆ โดยไม่จำเป็นต้องพัฒนาแอปพลิเคชันขึ้นมาใหม่หรือแก้ไข Code เพื่อให้เหมาะกับ แพลตฟอร์มนั้น ซึ่งทำให้ใช้เวลาในการพัฒนาน้อยลง

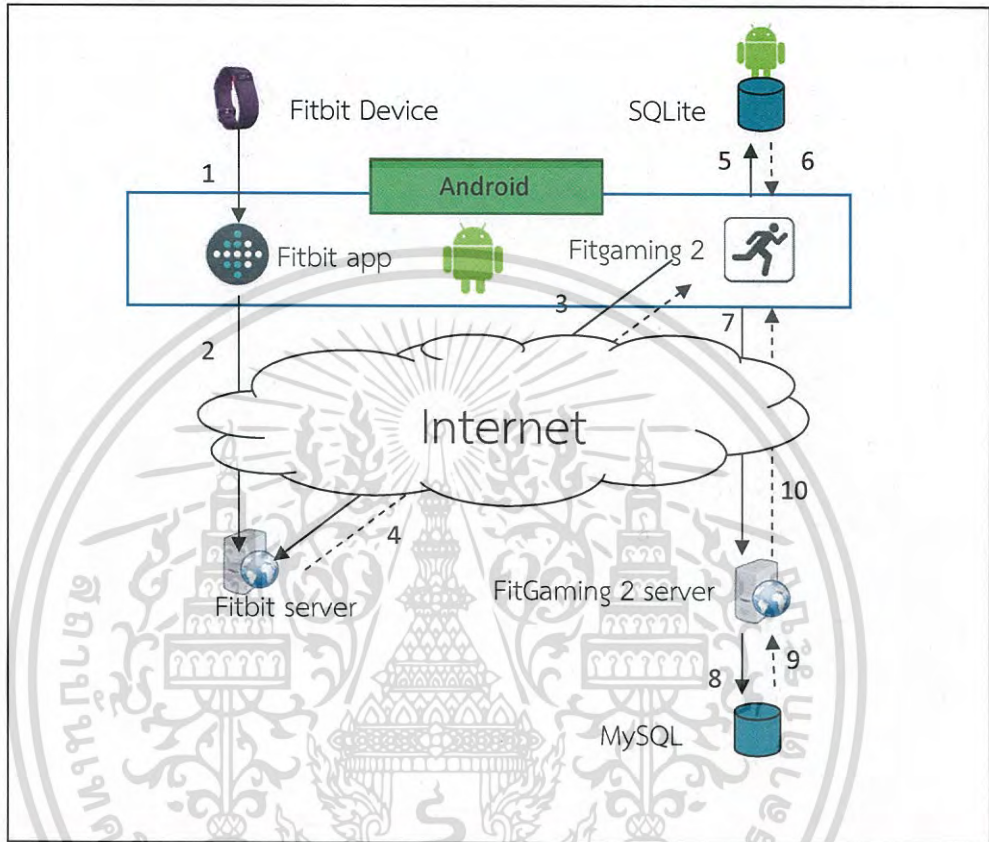
LibGDX มีความสามารถในการจัดการกับไฟล์ ระบบอุปกรณ์ Input ระบบเสียง และ ระบบ Interface ของ OpenGL ES 2.0 ได้ถูกรวมไว้เป็น API เพื่อให้สะดวกแก่การนำไปพัฒนาแอปพลิเคชันได้ง่ายมากขึ้น เช่น การ Render ข้อความ และ Sprite, การสร้าง User Interface, การเล่นเสียง Effect และ เพลงประกอบ การคำนวณพีชคณิตและตรีโกณมิติ การส่งข้อมูลแบบ JSON และ XML เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การวิเคราะห์และออกแบบเกม

#### 3.1 สถาปัตยกรรมระบบ



รูปที่ 3.1 สถาปัตยกรรมของระบบ

จากรูปที่ 3.1 การพัฒนาแอปพลิเคชันมีขั้นตอนการทำงานของระบบตามลำดับหมายเลขดังนี้

- 1) อุปกรณ์สายรัดข้อมือฟิตบิตแอปพลิเคชันบนสมาร์ตโฟนผ่านบลูทูธ
- 2) ฟิตบิตแอปพลิเคชันอัปเดตข้อมูลไปยังฟิตบิตเซิร์ฟเวอร์ผ่านเครือข่ายอินเทอร์เน็ต
- 3) ฟิตเกมมิ่ง 2 แอปพลิเคชันร้องขอข้อมูลของผู้ใช้ฟิตบิตและยืนยันตัวตนของแอปพลิเคชันโดยผ่าน OAuth 2.0 framework
- 4) ฟิตบิตเซิร์ฟเวอร์ส่งข้อมูลที่ร้องขอกลับมาที่แอปพลิเคชันฟิตเกมมิ่ง 2
- 5) แอปพลิเคชันฟิตเกมมิ่ง 2 ส่งข้อมูลไปเก็บลงฐานข้อมูล SQLite
- 6) เมื่อเกิดการร้องขอข้อมูลจากฐานข้อมูล SQLite จะส่งข้อมูลกลับไปยังฟิตเกมมิ่ง 2
- 7) แอปพลิเคชันฟิตเกมมิ่ง 2 ติดต่อกับฟิตเกมมิ่ง 2 เซิร์ฟเวอร์เพื่อเก็บข้อมูล
- 8) ฟิตเกมมิ่ง 2 เซิร์ฟเวอร์ส่งข้อมูลไปเก็บลงฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 9) เมื่อ แอปพลิเคชันพิตเกมมิ่ง 2 ร้องขอข้อมูลจากฐานข้อมูล ฐานข้อมูลจะส่งข้อมูลไปยังพิตบิตเกมมิ่ง 2 เซิร์ฟเวอร์
- 10) แอปพลิเคชันพิตเกมมิ่ง 2 เซิร์ฟเวอร์จะส่งข้อมูลกลับมาที่แอปพลิเคชันพิตเกมมิ่ง 2

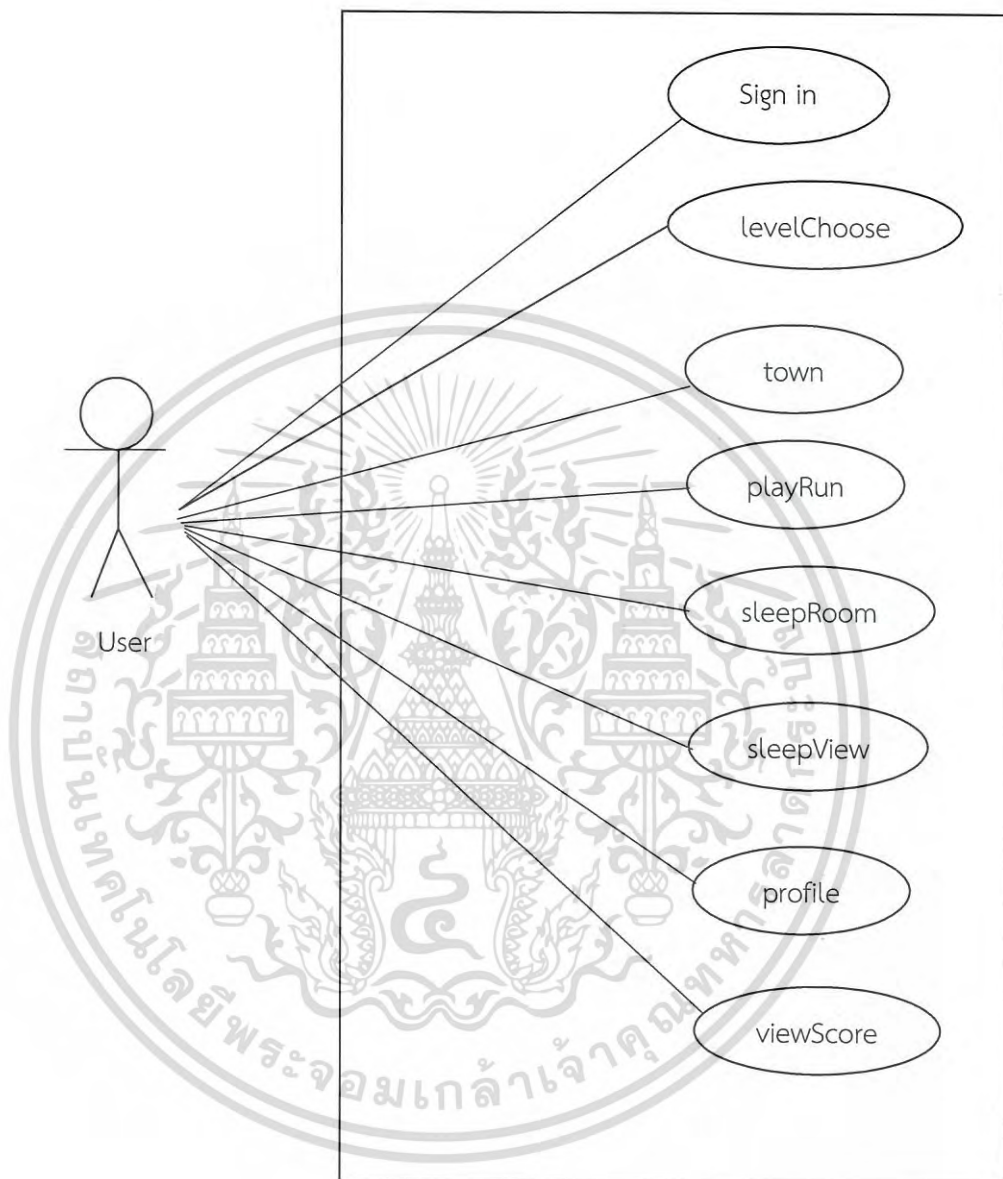
## 3.2 การวิเคราะห์เกม

### 3.2.1 Functional Requirement

- เกมแอนดรอยด์จะเป็นแบบออนไลน์
- ผู้เล่นเกมนี้ต้องลงทะเบียนและเป็นสมาชิกของพิตบิต
- ผู้เล่นต้องใส่ข้อมูลตามความเป็นจริงเพื่อให้ได้ตัวละครที่สมจริงที่สุด
- สามารถแสดงรายชื่อและคะแนนของเพื่อนที่เข้ามาเล่นเกมนี้ได้
- GameScreen เป็นฟังก์ชันเข้าไปในหน้าเกม
- Sleep View เป็นฟังก์ชันแสดงข้อมูลการนอนหลับ
- Level Choose เป็นฟังก์ชันในการเลือกด่านของเกม
- Profile เป็นฟังก์ชันแสดงข้อมูลส่วนตัวพิตบิตของผู้เล่น
- Show Result เป็นฟังก์ชันแสดงคะแนนที่ได้จากการเล่นเกม

### 3.2.3 Use Case Diagram

จากการวิเคราะห์ความต้องการของระบบใน 3.2.1 และ 3.2.2 จึงนำมาเขียน Use Case Diagram เพื่อแสดงระบบโดยรวมได้ในรูป 3.2

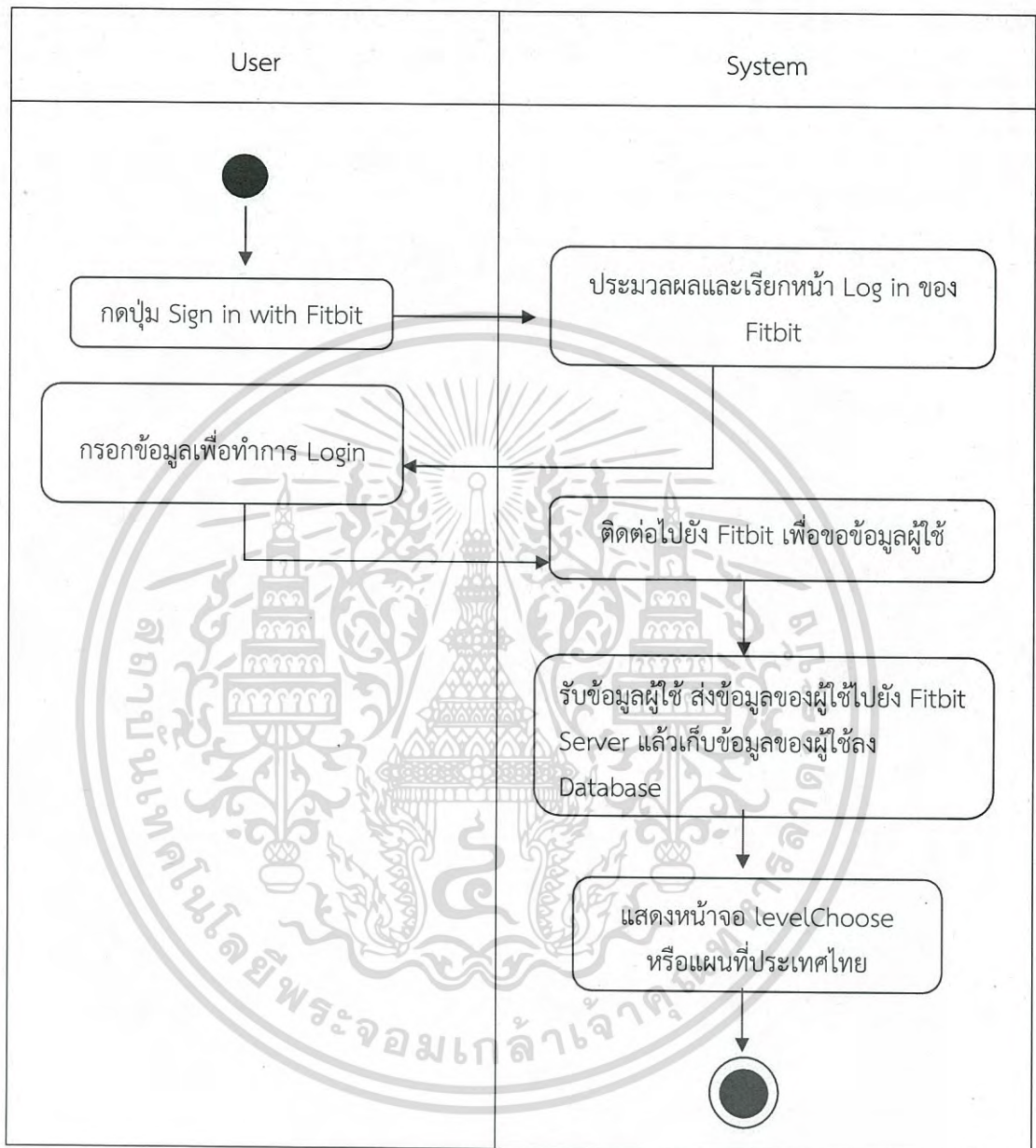


รูปที่ 3.2 Use Case Diagram ของเกมสับสนแอนดรอยด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.4 Activity Diagram

Activity Diagram ในหัวข้อ Sign in จะแสดงเมื่อเข้าแอปพลิเคชันเป็นครั้งแรก มี Activity ของหน้าจอ Sign in ดังรูปที่ 3.3



รูปที่ 3.3 Activity Diagram ของ Sign in

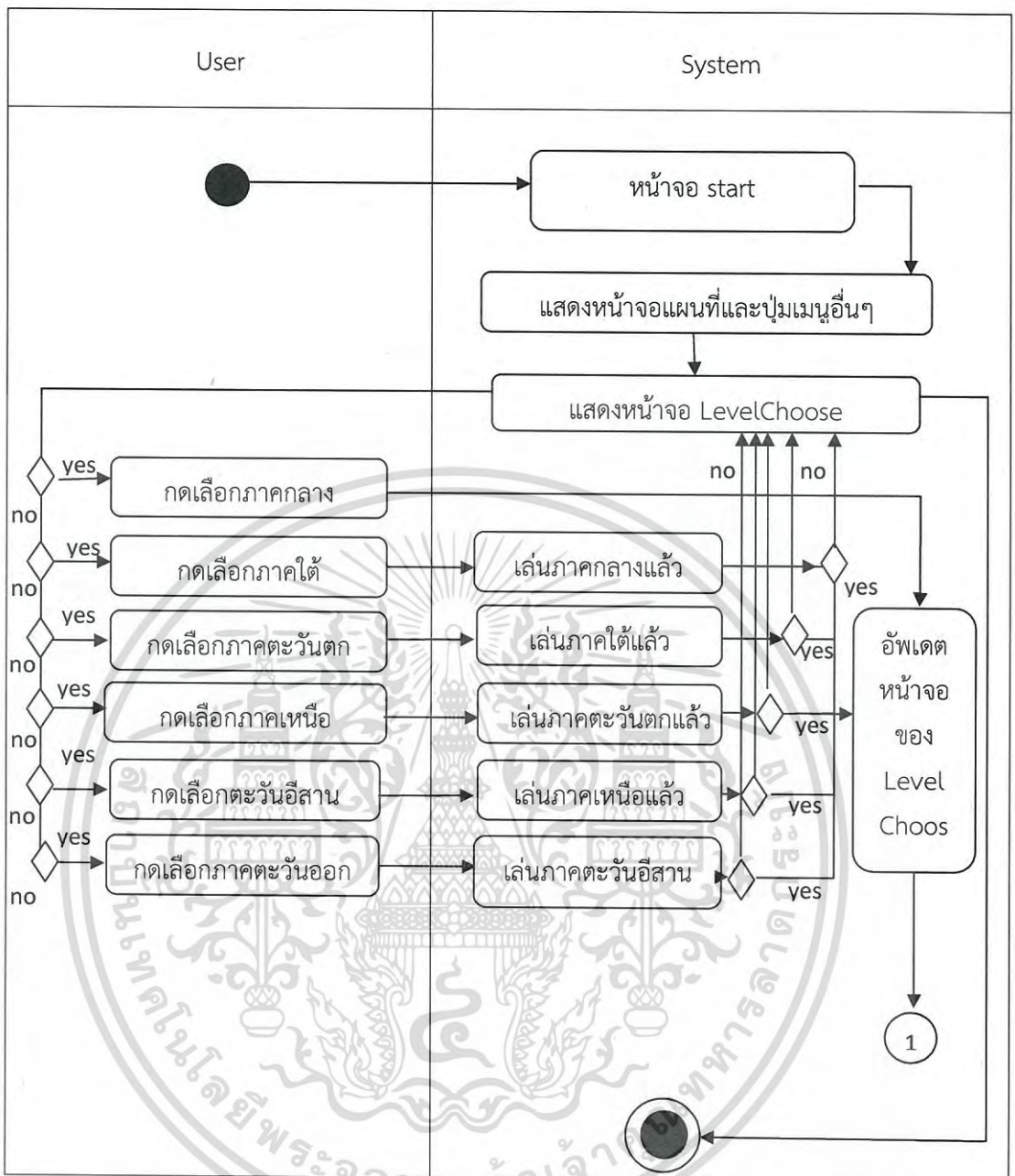
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ Sign in เรียบร้อยแล้วก็จะเข้าสู่หน้าจอของเกม โดยหน้าจอเกมจะมี Activity Diagram 11 ไดอะแกรม

หน้าจอเลือกเข้าด่าน

- Activity Diagram levelChoose เป็นหน้าจอเลือกภาคในประเทศไทยและเลือกเมือง ดังรูปที่ 3.4 และ 3.5
- Activity Diagram town เป็นหน้าจอเมือง ดังรูปที่ 3.6
- Activity Diagram PlayGame เป็นหน้าจอเมือง ดังรูปที่ 3.7
- Activity Diagram sleepRoom หน้าจอพักผ่อน ดังรูปที่ 3.8
- Activity Diagram sleepView หน้าจอแสดงข้อมูลการนอนหลับ ดังรูปที่ 3.9
- Activity Diagram profile หน้าจอแสดงข้อมูลของตนเอง ดังรูปที่ 3.10
- Activity Diagram viewscore หน้าจอแสดงคะแนนของตัวเองและเพื่อน ดังรูปที่ 3.11

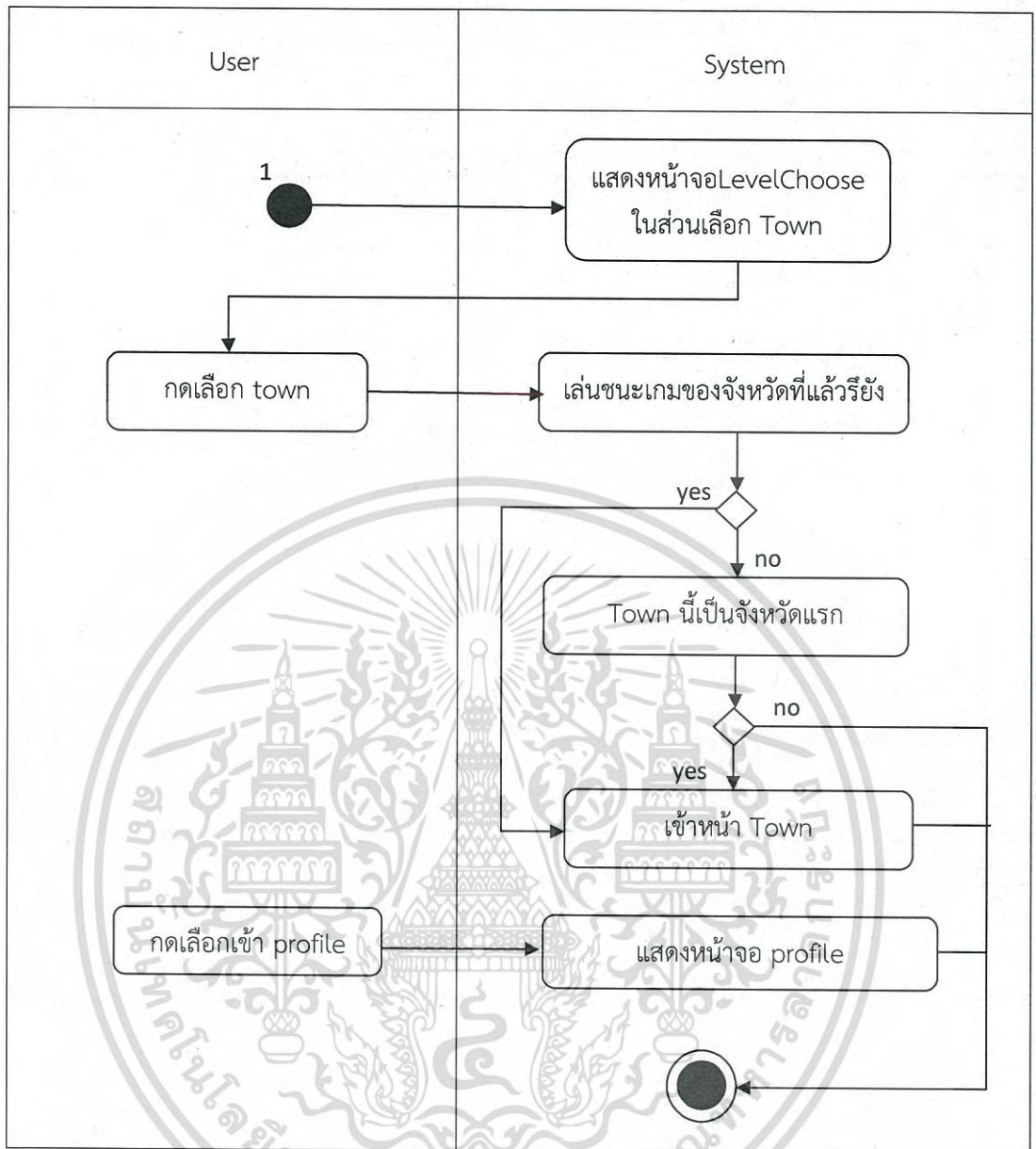




รูปที่ 3.4 Activity Diagram ของ LevelChoose เลือกภาค

จากรูปที่ 3.4 ในส่วนของการอัปเดตหน้าจอ Level Choose จะอัปเดตกลุ่มเมืองที่อยู่ในภาคที่เลือกไป ซึ่งจำนวนจะต่างกันไปในแต่ละภาค

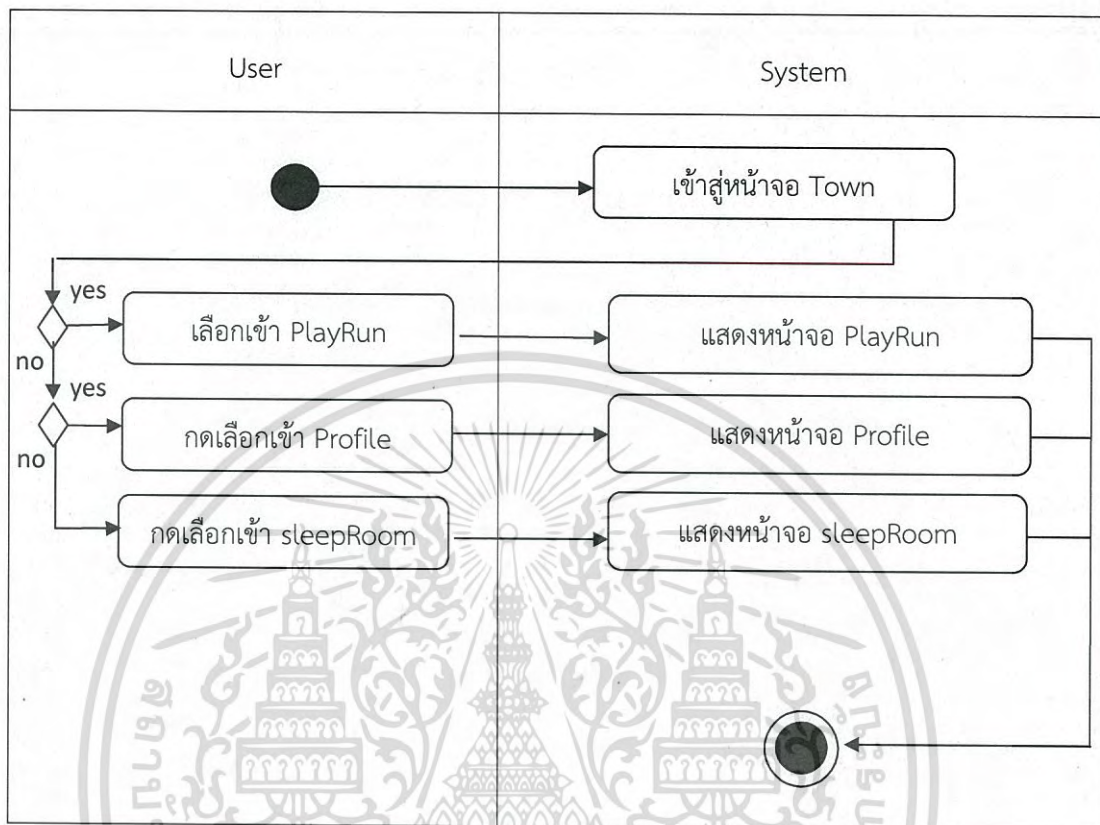
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 Activity Diagram ของ LevelChoose เลือกเมือง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

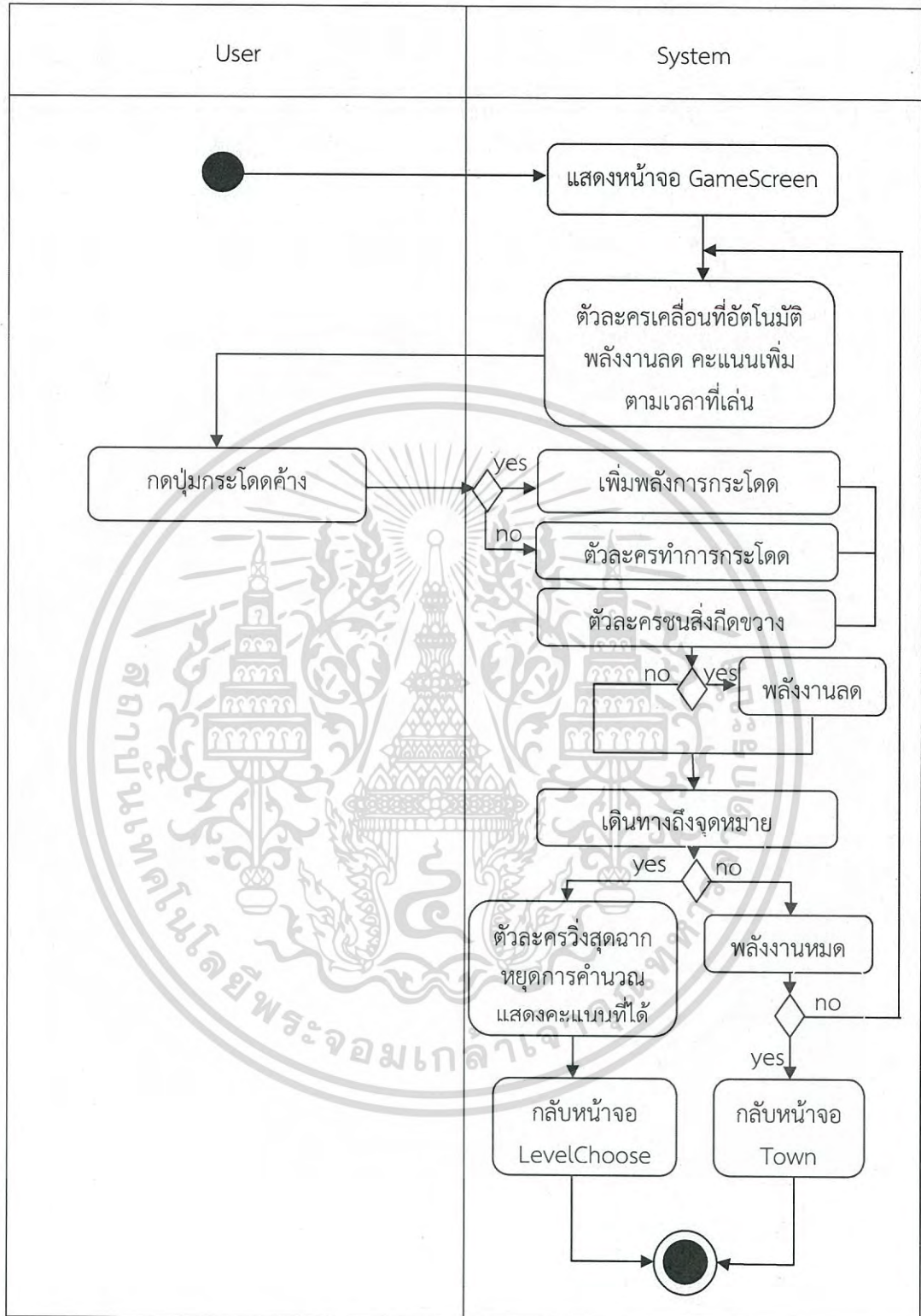
พอเข้าสู่หน้าจอ town โดยเข้าจาก levelChoose ซึ่งในหน้าจอนี้ก็จะสามารถเข้า  
ได้อีก 7 รายการ โดย foodCourt Shop playRun Profile sleepRoom back และ Log  
out ได้ตามรูปที่ 3.6



รูปที่ 3.6 Activity Diagram ของ town หน้าจอของเมือง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

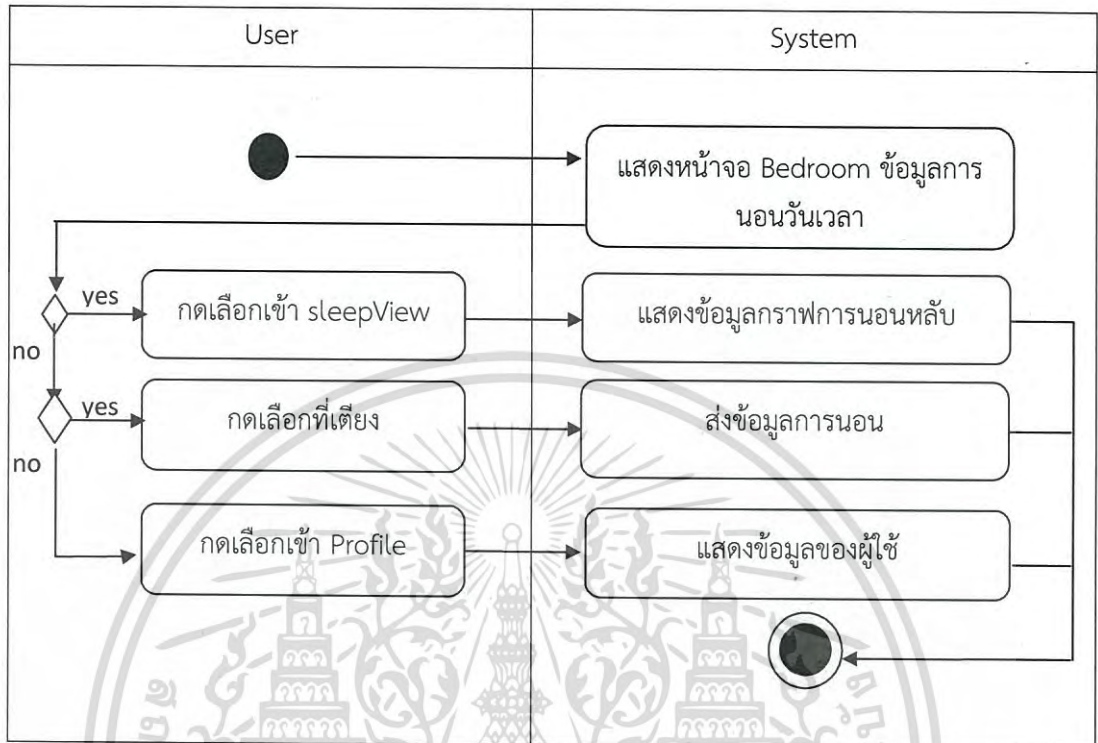
พอเข้าสู่หน้าจอ GameScreen จากหน้า Town โดยหน้านี้จะมีการกระทำคือ



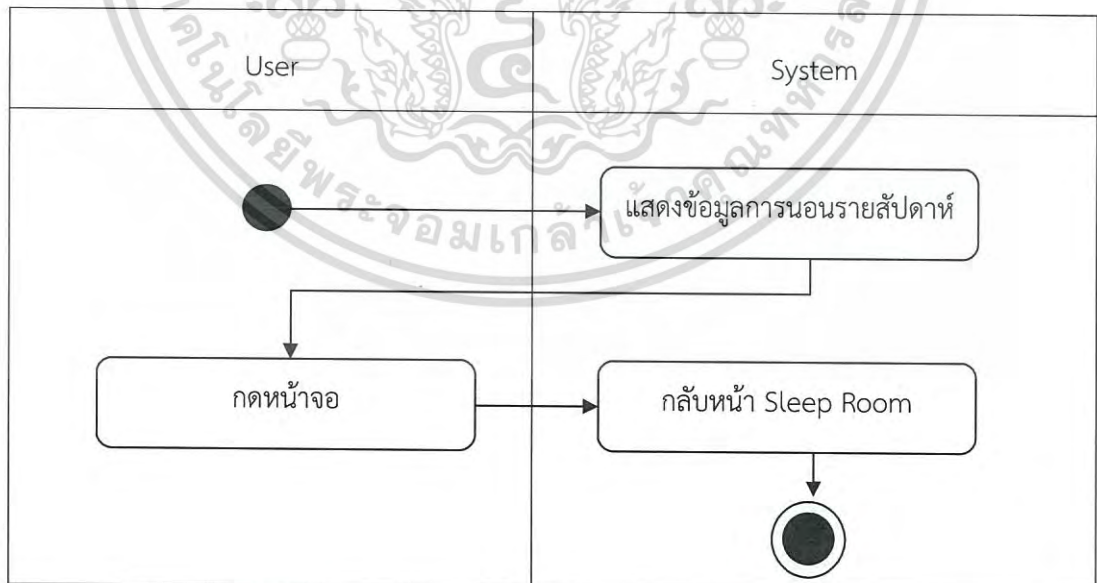
รูปที่ 3.7 Activity Diagram ของ PlayGame หน้าเล่นเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าจอ sleepRoom สามารถเข้าได้ที่ town ซึ่งในหน้าต่างของ sleepRoom จะมีการกระทำได้อยู่ 2 อย่าง คือแสดงกราฟการนอนหลับ และเข้าสู่สถานะการนอนตามรูปที่ 3.8

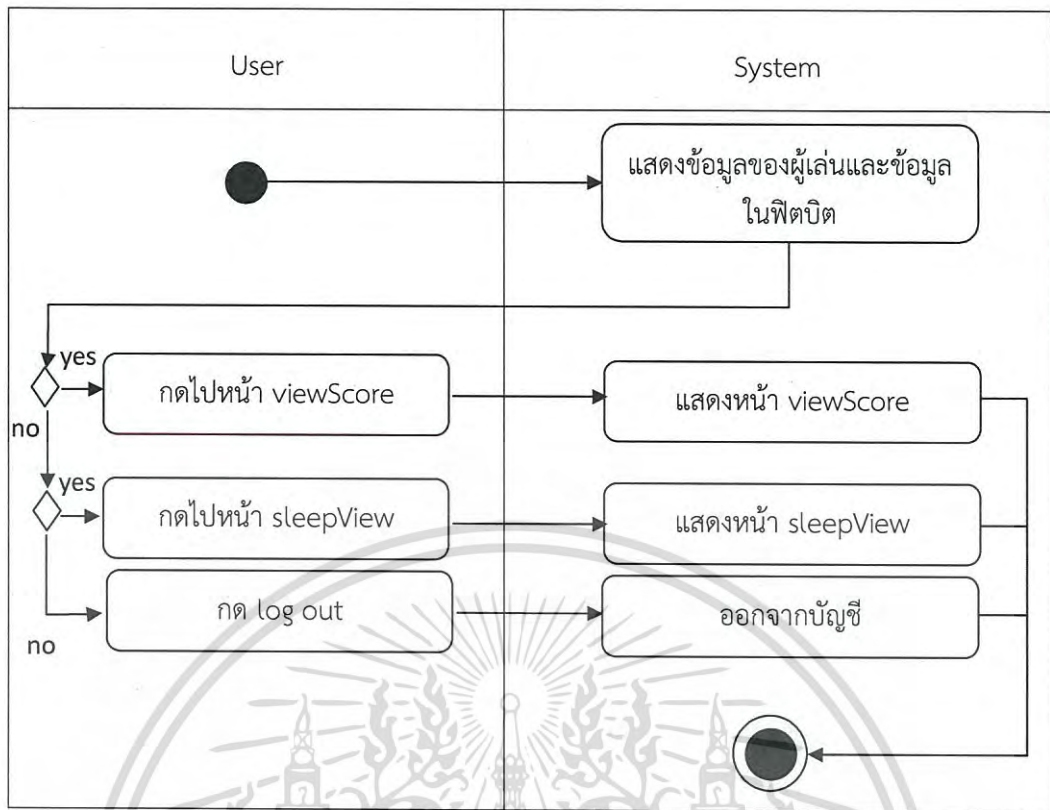


รูปที่ 3.8 Activity Diagram ของ SleepRoom ห้องนอนหลับ  
 หน้าจอของ sleepView สามารถเข้าได้ทางหน้าจอ profile และ sleepRoom

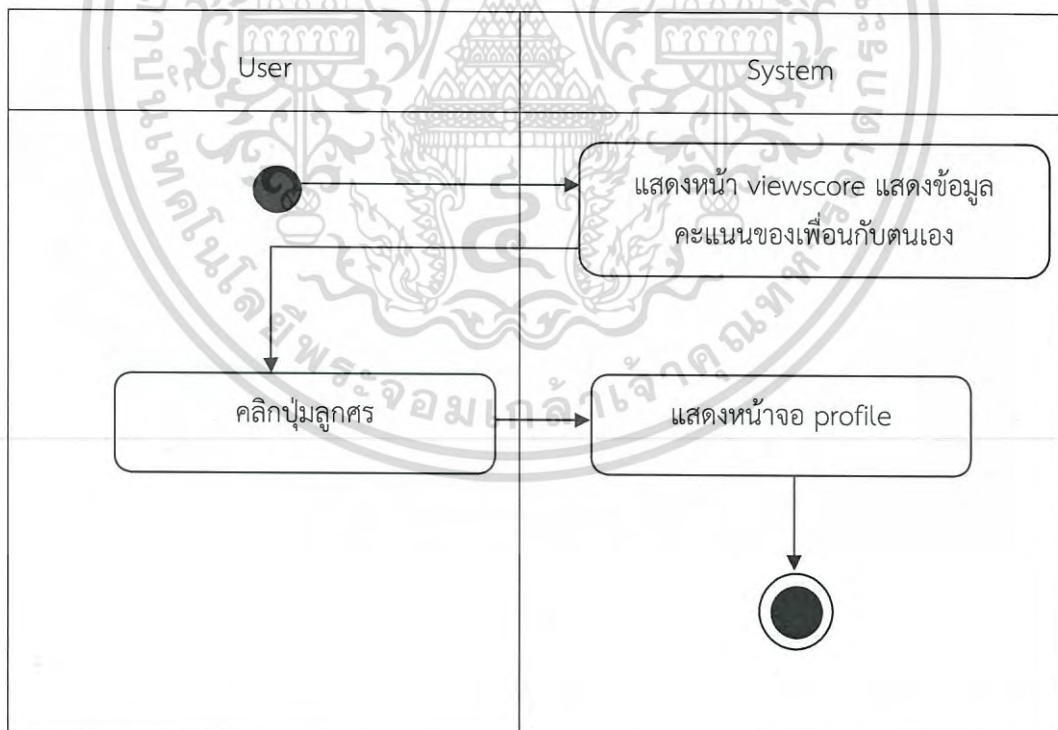


รูปที่ 3.9 Activity Diagram ของ sleepView แสดงข้อมูลการนอนหลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 Activity Diagram ของ profile หน้าจอแสดงข้อมูลของตนเอง



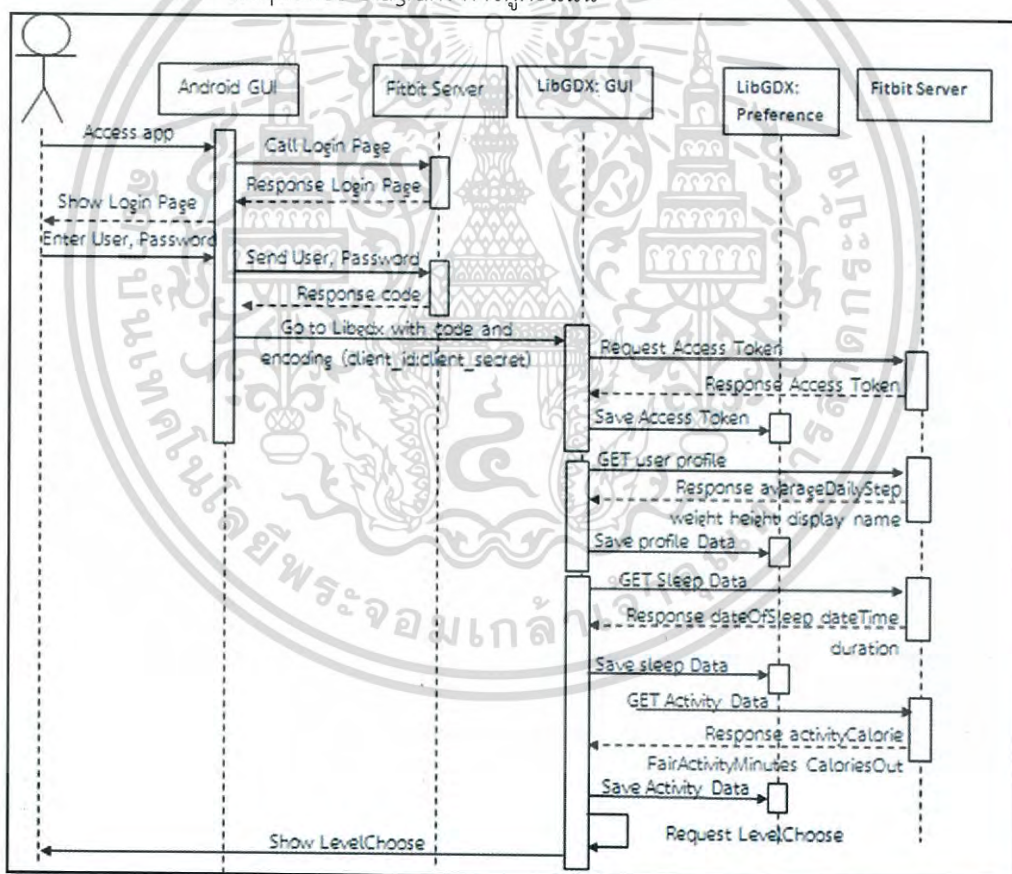
รูปที่ 3.11 Activity Diagram ของ ViewScore หน้าจอแสดงคะแนนของตัวเองและเพื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.5 Sequence Diagram

เป็นการออกแบบขั้นตอนการทำงานของแอปพลิเคชันเป็นแบบลำดับชั้น ในที่นี้จะพูดถึงส่วนการทำงานระหว่างผู้ใช้กับระบบของเกมแอปพลิเคชัน โดยมี Sequence Diagram ของแอปพลิเคชันดังนี้

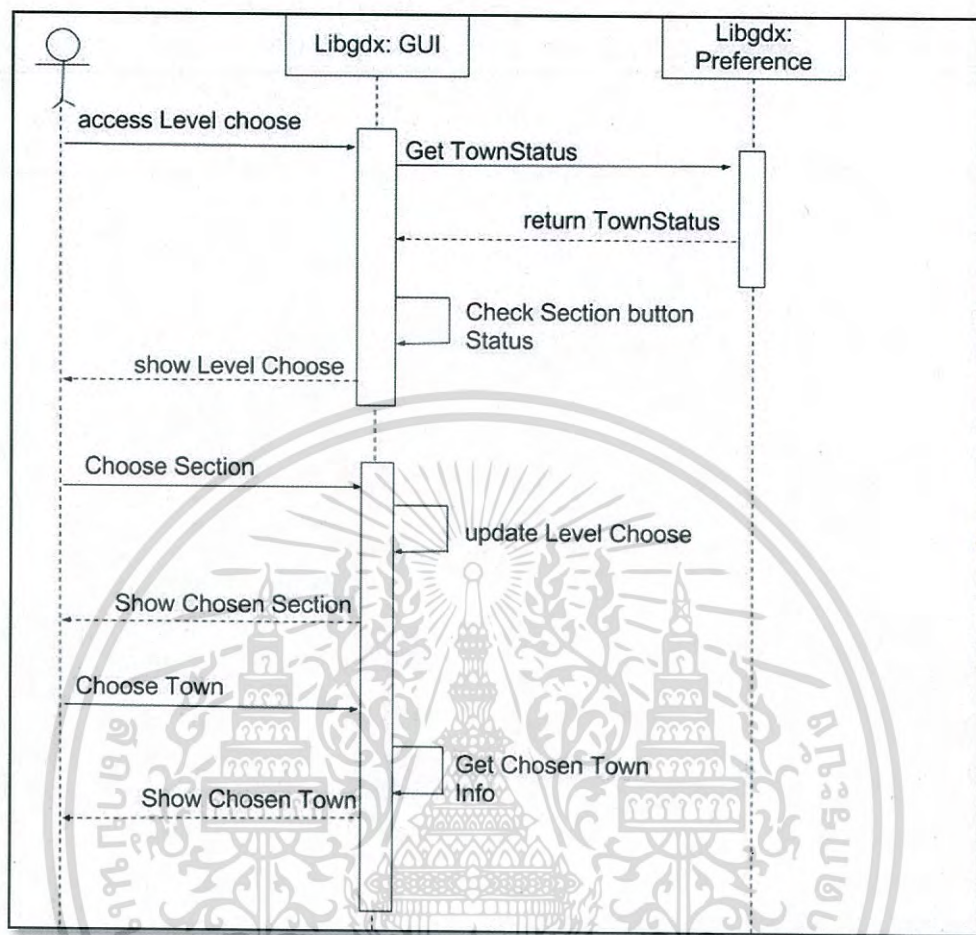
- Sequence Diagram Sign in
- Sequence Diagram การเลือกภาค
- Sequence Diagram การเลือกเมือง
- Sequence Diagram ขณะอยู่ในแต่ละเมือง
- Sequence Diagram การเล่นเกมแต่ละจังหวัด
- Sequence Diagram การดูข้อมูลของตัวเอง
- Sequence Diagram ในหน้าหลักของ Sleep Log
- Sequence Diagram การดูข้อมูลการนอน
- Sequence Diagram การดูคะแนน



รูปที่ 3.12 Sequence Diagram Sign in

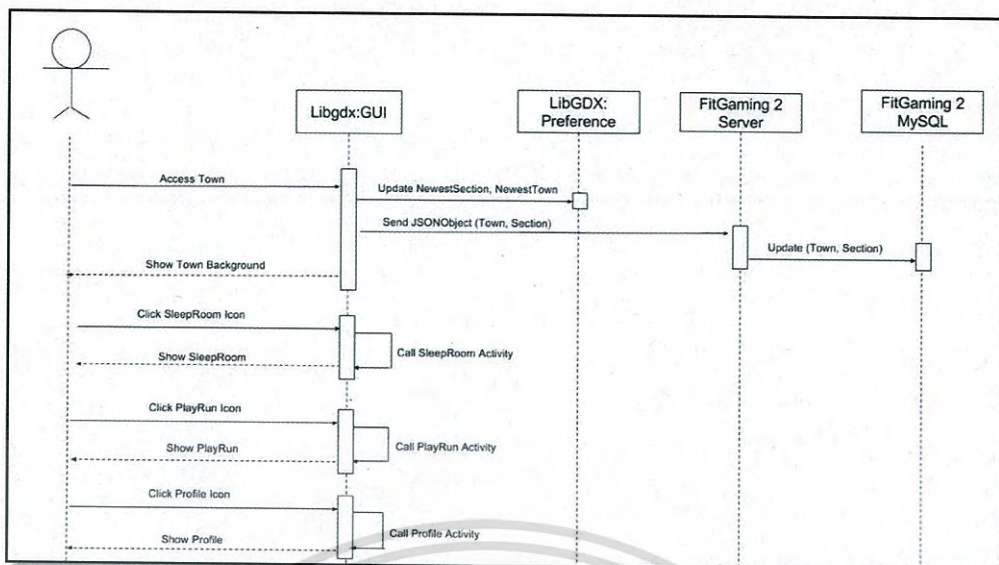
รูปที่ 3.12 อธิบายการทำงานของการทำงานการติดต่อกับฟิตบิตเซอร์ฟเวอร์ ซึ่งจะเป็นการติดต่อระหว่างแอนดรอยด์กับฟิตบิตเซอร์ฟเวอร์ โดยจะติดต่อกับฟิตบิตเซอร์ฟเวอร์เพื่อร้องขอ คีย์ต่าง ๆ ในการใช้ยืนยันตัวตนเพื่อนำมาใช้ร้องขอข้อมูลต่าง ๆ จากฟิตบิตเซอร์ฟเวอร์ เมื่อได้

ข้อมูลโปรไฟล์ ข้อมูลการออกกำลังกาย และข้อมูลจากฟิตเนสเซ็นเซอร์ จะทำการเก็บเข้ามาในตัว Preference



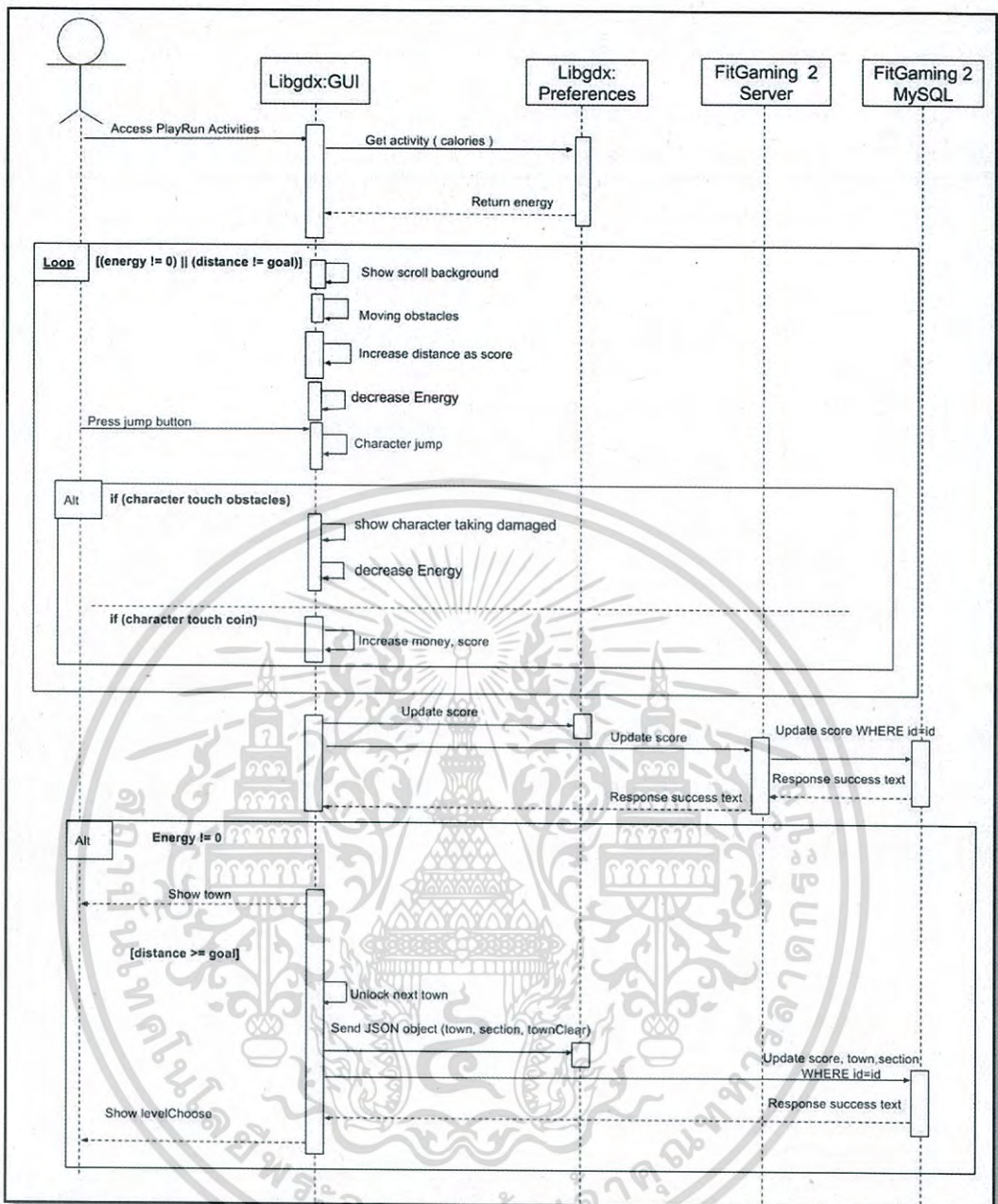
รูปที่ 3.13 Sequence Diagram LevelChoose

รูปที่ 3.13 อธิบายการทำงานของการเล่นเลือกภาคและเมืองในเกม โดยจะตรวจการผ่านด่านจากข้อมูลที่เก็บไว้ใน Preference ถ้าด่านได้ถูกเล่นไปแล้ว จะทำการปลดล็อกด่านถัดไปได้ และในการเล่นภาคถัดไปจะสามารถเล่นได้ โดยจะต้องผ่านทุกด่านของภาคที่แล้ว (ยกเว้นภาคแรก)



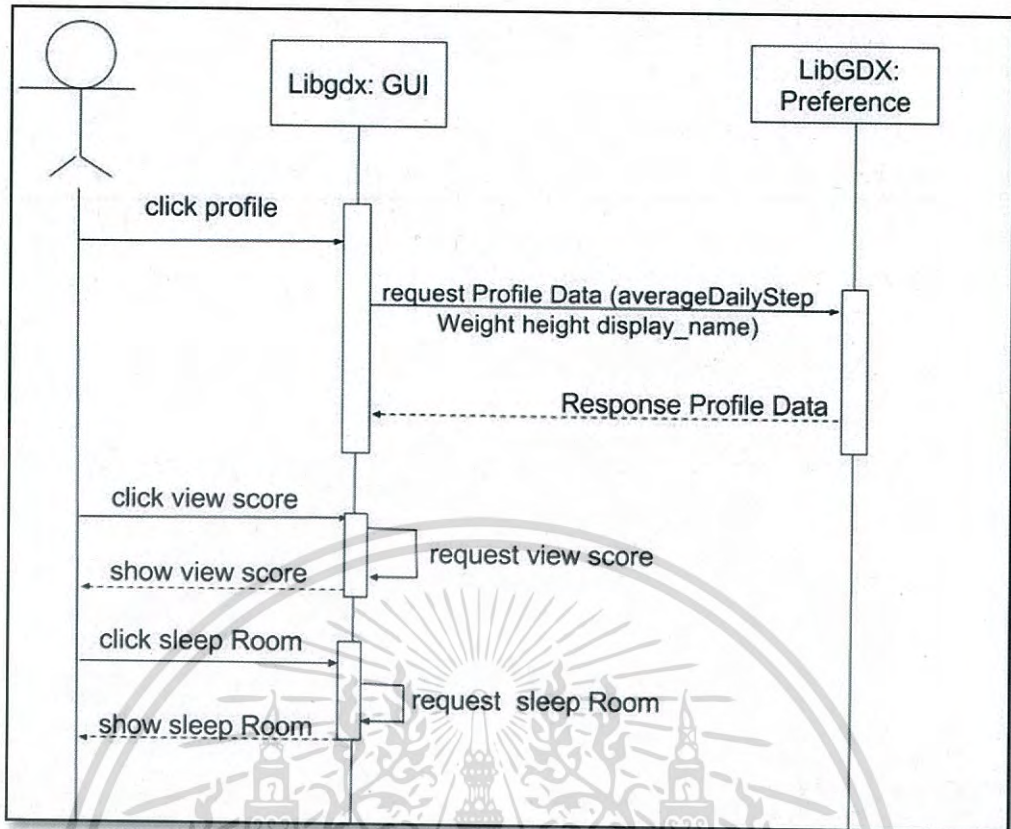
รูปที่ 3.14 Sequence Diagram Town ขณะอยู่ในแต่ละจังหวัด

รูปที่ 3.14 อธิบายการทำงานขณะอยู่ในแต่ละจังหวัด จะมีปุ่ม Play Game, Sleep Log และ Profile ให้กด นอกจากนี้จะมีการอัปเดตภาพพื้นหลังในแต่ละเมืองตามจังหวัดนั้นๆ อีกด้วย



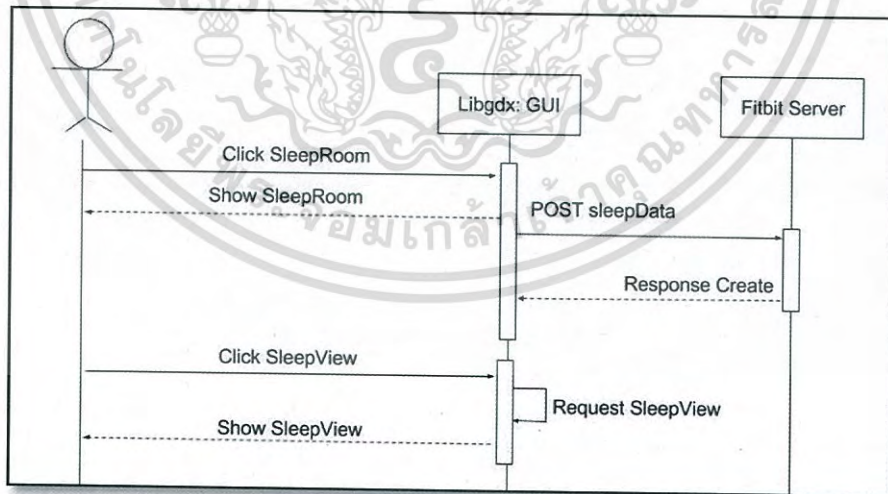
รูปที่ 3.15 Sequence Diagram การเล่นเกมในแต่ละด่าน

รูปที่ 3.15 อธิบายการทำงานของการเล่นเกมในแต่ละด่าน หากตัวละครโดนสิ่งกีดขวาง ค่าพลังจะลดลง ถ้าหากค่าพลังลดลงจนหมดโดยที่ยังไม่เข้าเส้นชัยจะถือว่าแพ้ในด่านนั้น ตัวละครก็สามารถสะสมเหรียญในแต่ละด่าน และถ้าหากตัวละครวิ่งเข้าเส้นชัยจะถือว่าชนะในด่านนั้น และผ่านไปสู่ด่านต่อไป



รูปที่ 3.16 Sequence Diagram การดูข้อมูลโปรไฟล์

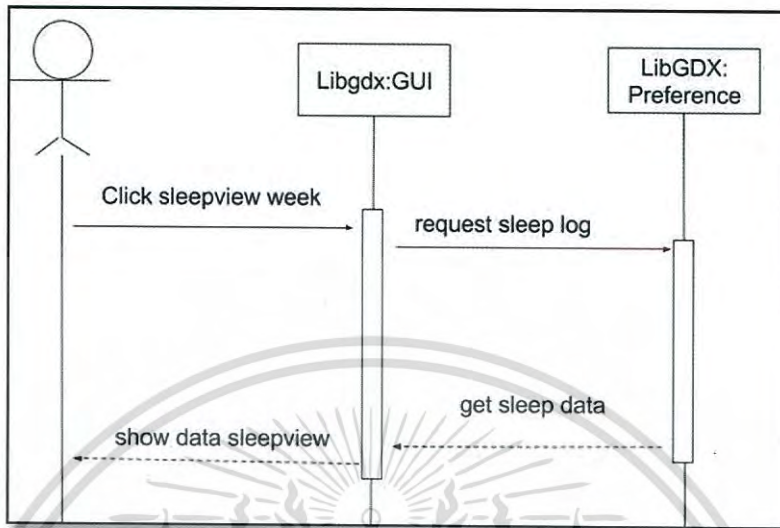
รูปที่ 3.16 อธิบายการทำงานของการทำงานของการดูข้อมูลของตัวเอง โดยแอปพลิเคชันจะดึงข้อมูลของผู้ใช้ทั้งหมดมาจาก Preference



รูปที่ 3.17 Sequence Diagram ในหน้าหลักของ Sleep Room

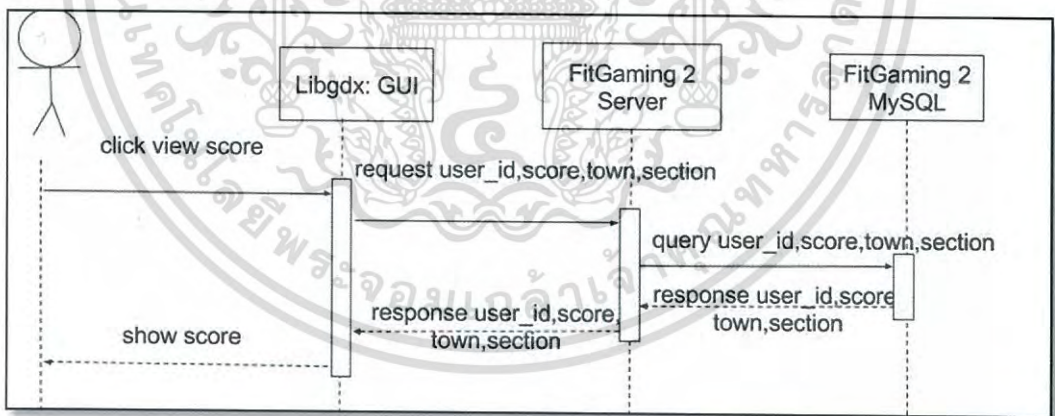
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.17 อธิบายการทำงานในหน้าหลักของ Sleep View เมื่อผู้ใช้กดเข้าสู่หน้า Sleep View จะมีการดึงข้อมูลการนอนของผู้ใช้มารอล่วงหน้า และจะโหลดข้อมูลแบบเต็มเมื่อผู้ใช้กดเข้าไปดูข้อมูลการนอน



รูปที่ 3.18 Sequence Diagram การดูข้อมูลการนอน

รูปที่ 3.18 อธิบายการทำงานของ การดูข้อมูลการนอน ซึ่งเป็นข้อมูลการนอนในระยะเวลา 1 สัปดาห์ ในรูปแบบของกราฟแท่ง โดยจะดึงข้อมูลมาจาก Preference ที่ได้เก็บข้อมูลการนอนกลับไว้แล้ว



รูปที่ 3.19 Sequence Diagram การดูคะแนน

รูปที่ 3.19 อธิบายการทำงานของ การดูคะแนน โดยแอปพลิเคชันจะดึงข้อมูลคะแนนของเพื่อนจาก MySQL จาก FitGaming2Server อยู่ในรูปของตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 : ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 การออกแบบฐานข้อมูล

จากการวิเคราะห์พบว่าเกมบนพีดีบีดีมีตารางข้อมูลเก็บข้อมูลในเซิร์ฟเวอร์ 1 ตาราง คือ

- 1) ตาราง profile รายละเอียดในตาราง 3.1

ตารางที่ 3.1 ตาราง profile บนเซิร์ฟเวอร์

| ชื่อฟิลด์   | ประเภทข้อมูล | รายละเอียด                |
|-------------|--------------|---------------------------|
| Id          | VARCHAR(20)  | เก็บรหัสประจำตัวผู้เล่น   |
| Displayname | VARCHAR(20)  | เก็บชื่อของผู้เล่น        |
| Region      | VARCHAR(20)  | เก็บภูมิภาคของผู้เล่น     |
| Mission     | VARCHAR(10)  | เก็บด่านล่าสุดของผู้เล่น  |
| Score       | INT(50)      | เก็บคะแนนล่าสุดของผู้เล่น |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

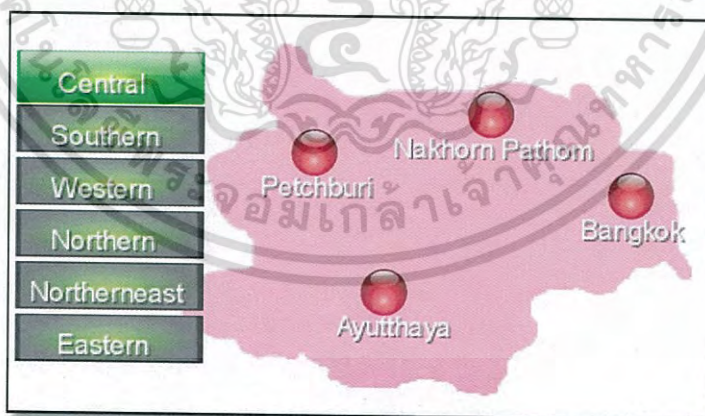
### 3.4 การออกแบบแอปพลิเคชัน

ในการออกแบบแอปพลิเคชัน FitGaming2 จะมีฟังก์ชันการทำงานหลักๆ ดังนี้ ฟังก์ชันยืนยันตัวตน โปรไฟล์ของผู้ใช้ โปรไฟล์เพื่อน ฟังก์ชันการเล่นเกม ฟังก์ชันคะแนน ฟังก์ชันอาหาร และฟังก์ชันการนอน



รูปที่ 3.20 ภาพหน้าจอ Sign in

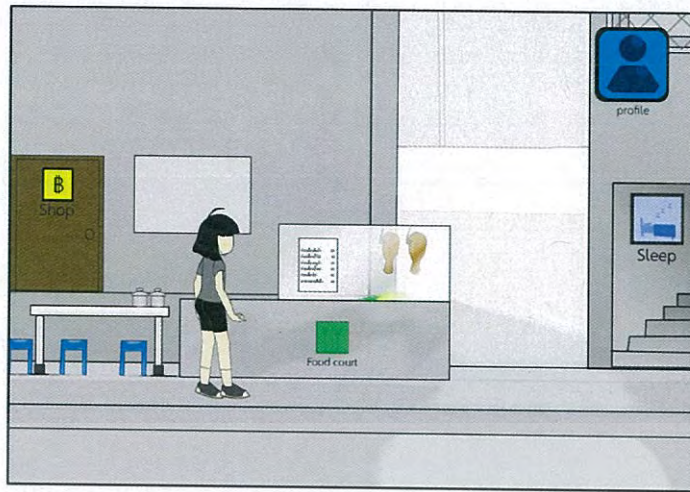
ในการเข้าใช้แอปพลิเคชัน ในครั้งแรกผู้ใช้ต้องทำการยืนยันตัวตน และต้องขอเข้าถึงข้อมูลพิตบิตของผู้ใช้ อย่างในรูปที่ 3.20



รูปที่ 3.21 ภาพหน้าจอ เลือกภาค

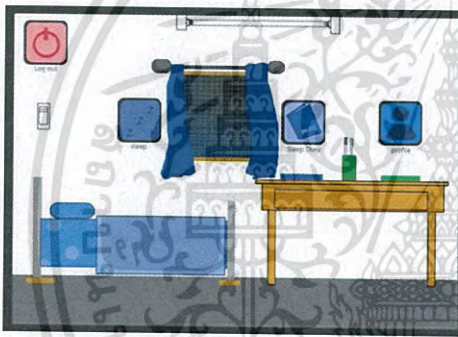
ในการออกแบบหน้าจอการเลือกภาค ในหน้าจอจะแสดงแผนที่ประเทศไทยที่แบ่งภาคแล้ว และจุดที่จะเลือกไปเล่นในภาค ดังในรูปที่ 3.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

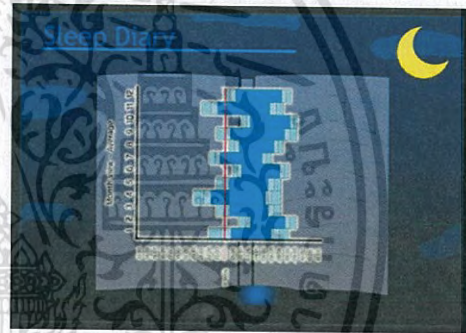


รูปที่ 3.22 ภาพหน้าจอเมือง

ในการออกแบบหน้าจอเมือง ก็จะมีหน้าต่างห้องให้เลือก เพื่อไปยังห้องนั้นๆ ดังนี้ ห้องนอน ห้องเกม และโปรไฟล์ ซึ่งในหน้าจอก็จะแสดงถึงภาคบรรยากาศในเมืองนั้นๆ ดังรูปที่ 3.22



รูปที่ 3.23 ภาพหน้าจอ ห้องนอน



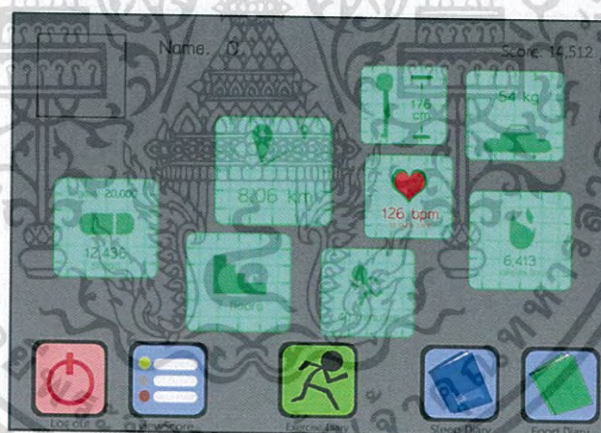
รูปที่ 3.24 ภาพหน้าจอ ดูเวลาการนอนรายสัปดาห์

ในการออกแบบหน้าจอห้องนอน ภายในหน้านี้ที่แสดงในรูป 3.23 สามารถเลือกเข้าไปดูว่า ในแต่ละคืนผู้ใช้ ใช้เวลานานกี่ชั่วโมง และเวลานอนเฉลี่ยของผู้ใช้เป็นเท่าใดในรายสัปดาห์ดังในรูป 3.24



รูปที่ 3.25 ภาพหน้าจอเล่นเกม

ในการออกแบบหน้าจอเล่นเกม หลังจากเข้าสู่หน้าจอเล่นเกมผู้ใช้งานมีเวลาเตรียมตัว 3 วินาที หลังจากเริ่มเกม ตัวละครจะวิ่งไปในฉาก ระหว่างทางจะมีเหรียญให้เก็บสะสม และผู้เล่นต้องคอยหลบหลีกอุปสรรคต่างๆ เพื่อป้องกันไม่ให้ค่าพลังงานของผู้ใช้หมดก่อนที่จะบรรลุเป้าหมาย ซึ่งเวลาที่บรรลุในเป้าหมายและค่าพลังงานเห็นได้ในรูปที่ 3.25



รูปที่ 3.26 ภาพหน้าจอ Profile

หน้าจอข้อมูลผู้ใช้ของแอปพลิเคชันจะแสดงข้อมูลต่างๆของผู้ใช้ คือ ชื่อ คะแนน ข้อมูลการออกกำลังกาย และสามารถเชื่อมโยงไปหน้าจออื่นๆอย่าง หน้าจอดูคะแนน หน้าจอดูการนอนหลับ อย่างในรูป 3.26



รูปที่ 3.27 ภาพหน้าจอแสดงคะแนนเพื่อน



รูปที่ 3.28 ภาพหน้าจอแสดงโปรไฟล์เพื่อน

หน้าจอคะแนน ในหน้าจอนี้ผู้ใช้สามารถคะแนนของเพื่อนที่ใช้แอปพลิเคชันเดียวกันดังรูป 3.27 สามารถดูโปรไฟล์เพื่อนได้ในรูปที่ 3.28

### 3.5 การพัฒนาแอปพลิเคชัน

ในการพัฒนาแอปพลิเคชัน จะต้องทำการสร้างโปรเจกต์ผ่านตัวติดตั้งของ LibGDX สามารถดูได้จากภาคผนวกในการติดตั้ง โครงสร้างของโปรเจกต์ที่ได้มาจะมีทั้งหมด 5 ส่วน คือ ส่วนของแอนดรอยด์ ไอโอเอส เว็บ คอมพิวเตอร์ และ ส่วนของตัวเกมที่สร้างด้วย LibGDX ในโปรเจกต์นี้ผู้พัฒนาเลือกที่จะใช้ในส่วนของแอนดรอยด์

ในการพัฒนาแอปพลิเคชัน จะแบ่งการออกเป็นสองส่วน คือการสร้างในส่วนของแอนดรอยด์ และในส่วนของส่วน core ซึ่งในส่วนนี้จะใช้คลาสของ LibGDX

### 3.6 การพัฒนาแอปพลิเคชันในส่วนของแอนดรอยด์

#### 3.6.1 ตั้ง url ของแอนดรอยด์ เพื่อรับข้อมูลจากเว็บฟิตบิต

การพัฒนาแอปพลิเคชันที่เชื่อมต่อกับฟิตบิตบนระบบปฏิบัติการแอนดรอยด์นั้นต้องทำการกำหนดสิทธิการใช้งานอินเทอร์เน็ตสำหรับเชื่อมต่อฟิตบิตก่อน ดังรูปที่ 3.29 และ ตั้งที่อยู่ให้ข้อมูล URI จากฟิตบิตเข้าแอนดรอยด์โดยการใส่คำสั่งดังรูปที่ 3.30

```
<uses-sdk android:minSdkVersion="16" android:targetSdkVersion="23" />
<uses-permission android:name="android.permission.INTERNET"></uses-permission>

<application
    android:allowBackup="true"
    android:icon="@drawable/ic_launcher"
    android:label="Fitgaming2alpha"
    android:theme="@style/GdxTheme" >
    <activity
        android:name="fit.main.AndroidLauncher"
```

รูปที่ 3.29 การเพิ่มสิทธิการใช้งานอินเทอร์เน็ต ในแอนดรอยด์แอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<application
    android:allowBackup="true"
    android:icon="@drawable/ic_launcher"
    android:label="Fitgaming2alpha"
    android:theme="@style/CdxTheme" >
    <activity
        android:name="fit.main.AndroidLauncher"

        android:screenOrientation="landscape"
        android:configChanges="keyboard|keyboardHidden|orientation|screenSize">
        <intent-filter android:label="Fitgaming2alpha" >
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>

        <intent-filter >
            <action android:name="android.intent.action.VIEW" />
            <category android:name="android.intent.category.DEFAULT" />
            <category android:name="android.intent.category.BROWSABLE" />
            <data android:scheme="fit" android:host="gaming" ></data>
        </intent-filter>
    </activity>

```

รูปที่ 3.30 โค้ดกำหนดที่อยู่ข้อมูลของแอนดรอยด์

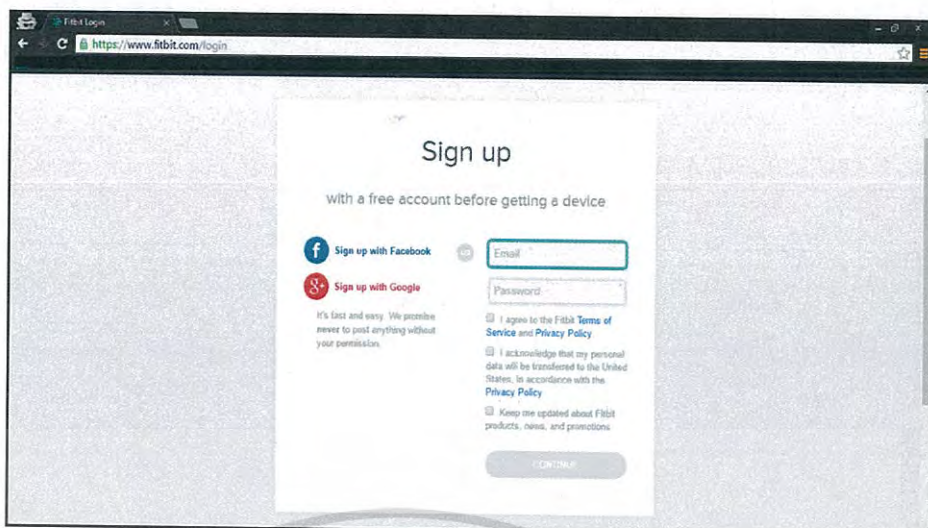
```

<action android:name="android.intent.action.VIEW" />
เป็นคำสั่งข้อมูลที่ใช้แอปพลิเคชันสามารถดูได้
<category android:name="android.intent.category.DEFAULT" />
ถ้าจะรับค่า intent จำเป็นต้องใส่คำสั่งนี้
<category android:name="android.intent.category.BROWSABLE" />
เป็นคำสั่งสามารถทำให้เปิดเว็บเบราว์เซอร์ได้เพื่อที่จะแสดงข้อมูล
<data android:scheme="fit" android:host="gaming" ></data>
เป็นคำสั่งที่อยู่ของข้อมูลที่เว็บพิตบิตจะส่งเข้ามา โดยการกำหนด Scheme

```

### 3.6.2 สร้างฟังก์ชันการติดต่อ Fitbit API ด้วย OAuth 2.0

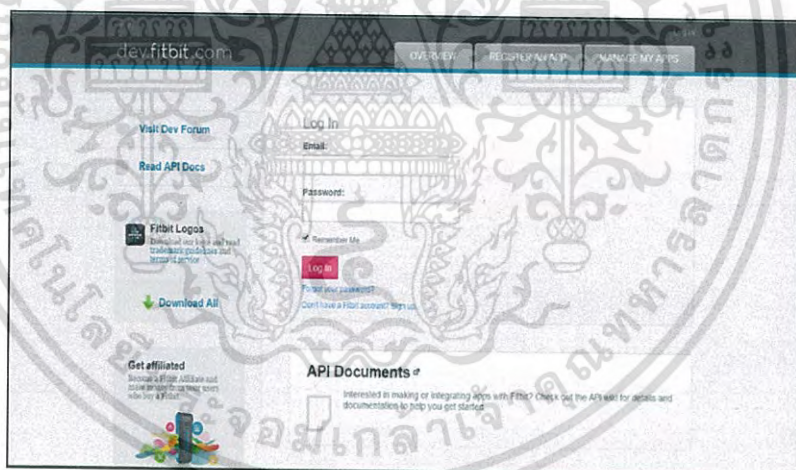
OAuth 2.0 เป็นการติดต่อของ Fitbit โดยใช้ Https Request ในการดึงข้อมูลตามที่ Fitbit API ยอมให้เรียกใช้ ซึ่งจะต้องทำการสมัครสมาชิกของ Fitbit เสียก่อน ซึ่งสามารถเข้าไปสมัครสมาชิกได้ที่ <https://www.fitbit.com/signup> ดังที่แสดงในรูปที่ 3.31



รูปที่ 3.31 หน้าจอสมัครสมาชิก

### 3.6.2.1 การลงทะเบียนแอปพลิเคชัน ฟิตบิตเซิร์ฟเวอร์

ในการพัฒนาแอปพลิเคชันที่ติดต่อกับฟิตบิตนั้นจำเป็นต้องยืนยันตัวตนสำหรับแอปพลิเคชันผ่าน `dev.fitbit.com/login` กรณีลงทะเบียน Fitbit.com เรียบร้อยแล้ว เพื่อเข้าสู่หน้าลงทะเบียนแอปพลิเคชัน



รูปที่ 3.32 หน้าจอเข้าสู่ระบบการพัฒนาแอปพลิเคชันฟิตบิต

เมื่อทำการเข้าสู่ระบบ จากนั้นให้เลือกเมนู Register an app เพื่อทำการสร้างแอปพลิเคชัน โดยประเภทของแอปพลิเคชันคือ Browser ดังรูปที่ 3.33 ในช่องที่ไม่มีเครื่องหมายสามารถใส่ชื่ออะไรก็ได้ลงไป

The screenshot shows a web form titled "Register an application". On the left, there are links for "Visit Dev Forum", "Read API Docs", "Fitbit Logos", "Download All", and "Get affiliated". The main form fields are:
 

- Application Name \* (text input)
- Description \* (text area)
- Application Website \* (text input)
- Organization \* (text input)
- Organization Website \* (text input)
- OAuth 2.0 Application Type \* (radio buttons for Server, Client, Personal)
- Callback URL \* (text input)
- Default Access Type \* (radio buttons for Read & Write, Read-Only)

 There are red error icons next to the Application Website, Organization, Callback URL, and Read-Only options. At the bottom, there is a link "Add a subscriber".

รูปที่ 3.33 การลงทะเบียนแอปพลิเคชันฟิตบิต

### OAuth 2.0 Application Type

ให้เลือกส่วนของ Client เพราะแอปพลิเคชันที่จะทำการดึงข้อมูล Fitbit เข้าแอนดรอยด์เพื่อให้ในข้อมูลเหล่านั้นไปใช้ต่อในตัวแอปพลิเคชัน

### Default Access Type

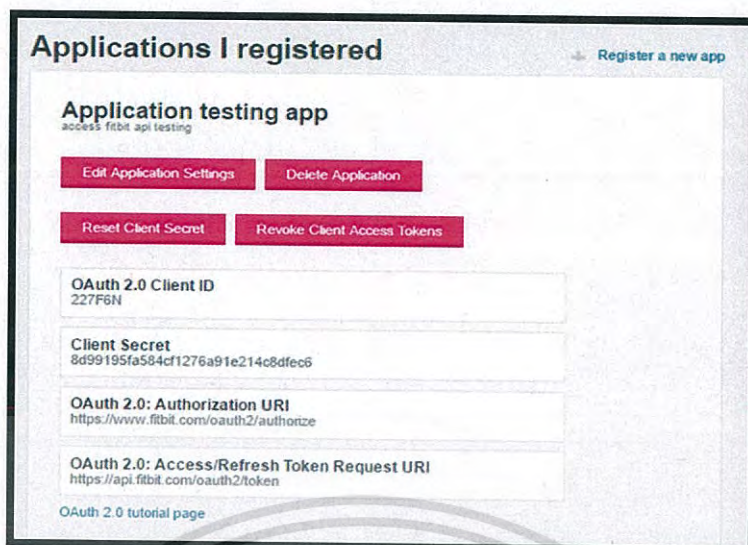
เป็นส่วนกำหนดการเข้าถึงข้อมูล ซึ่งในการพัฒนาแอปพลิเคชันนี้ต้องการทั้งเขียนและอ่านข้อมูล

### Callback Url

จะเป็นที่อยู่ Url ที่ Fitbit จะส่งข้อมูลกลับมาหาแอนดรอยด์ หรือตัว Custom Scheme ที่ตั้งไว้ในโปรเจ็คแอนดรอยด์ ซึ่งในกรณีนี้คือ fit://gaming

หลังจากการลงทะเบียนแล้ว เราก็จะได้พารามิเตอร์ไว้สำหรับติดต่อส่งข้อมูลระหว่างแอปพลิเคชันและFitbit Server ผ่าน OAuth 2.0 ดังรูปที่ 3.22 ได้แก่

- OAuth 2.0 Client ID
- Client Secret
- OAuth 2.0: Authorization URI
- OAuth 2.0: Access/Refresh Token Request URI



รูปที่ 3.34 หน้าจอแสดงพารามิเตอร์หลังจากที่ลงทะเบียนเรียบร้อยแล้ว

### 3.6.3 แอปพลิเคชันฝั่งไคลเอนต์ (Application on Client-Side)

#### 1) แอปพลิเคชันเชื่อมต่อกับฟิตบิตบนระบบปฏิบัติการแอนดรอยด์

ในการพัฒนาแอปพลิเคชันที่เชื่อมต่อกับฟิตบิตบนระบบปฏิบัติการแอนดรอยด์นั้นจะต้องเรียกใช้ intent เปิดหน้าเว็บ Fitbit เพื่อขอ code มาในการขอ Access Token โดย url ที่ร้องขอนั้น จะต้องทำการเขียนตามที่ฟิตบิตกำหนดไว้ โดยมี <https://www.fitbit.com/oauth2/authorize?> นำหน้าและด้านหลังตามด้วยค่าพารามิเตอร์ที่ฟิตบิตกำหนดให้ใส่ซึ่งได้มาจากค่าพารามิเตอร์หลังจากการลงทะเบียนแอปพลิเคชันและการตั้งค่า url ของแอนดรอยด์ ดังรูปที่ 3.35 แล้ว ข้อมูลจะถูกส่งกลับมาในรูปแบบของ JSON ซึ่งจะสามารถเขียนรับข้อมูลนำไปใช้ได้ดังรูปที่ 3.36

```
private String client_id = "227F6N"; //
private String client_secret = "8d99195fa584cf1276a91e214c8dfec6";
private String hostapp = "fit://gaming";

private String urlA = "https://www.fitbit.com/oauth2/authorize?"
+ "client_id="+client_id
+ "&response_type=code"
+ "&redirect_uri="+hostapp
+ "&scope=activity%20heartrate%20location%20nutrition%20profile%20settings%20sleep%20social%20weight";

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    Intent i = new Intent(Intent.ACTION_VIEW,
        Uri.parse(urlA));
    startActivity(i);
}
```

รูปที่ 3.35 การเปิดหน้าเว็บเพื่อไปเรียกขอข้อมูลฟิตบิต

```

@Override
protected void onStart() {
    super.onStart();
    Uri data = getIntent().getData();
    if (data != null) {
        startGame(data.getQuery());
    }
}

```

รูปที่ 3.36 โค้ดรับข้อมูล code จากที่เว็บพีตบิตส่งมา

หลังจากได้รับข้อมูล code ที่ส่งมาแล้ว จะต้องส่งค่า code และค่าเข้ารหัสของ client\_id:client\_secret ตามที่พีตบิตกำหนดลงไปใน LibGDX เพื่อนำไปร้องขอข้อมูลพีตบิตเพื่อไปใช้งานต่อดังรูปที่ 3.37 และ 3.38

```

String authorizationBase64Encode = Base64.encode(client_id+":"+client_secret);
private void startGame(String code){
    Intent g = new Intent(this, goToLibgdx.class);
    g.putExtra("code",code);
    g.putExtra("encode", authorizationBase64Encode);
    startActivity(g);
}

private String Base64encode(String input) {
    // String to be encoded with Base64
    Log.v(tag, "input encode: " + input);
    // Sending side
    byte[] data = null;
    try {
        data = input.getBytes("UTF-8");
    } catch (UnsupportedEncodingException e1) {
        e1.printStackTrace();
    }
    String base64 = Base64.encodeToString(data, Base64.DEFAULT);
    Log.v(tag, "Base64 encode: " + base64);
    return base64;
}

```

รูปที่ 3.37 การส่งข้อมูลที่ได้จากพีตบิตและข้อมูลที่เข้ารหัสไปยังคลาสถัดไป

```

public class goToLibgdx extends com.badlogic.gdx.backends.android.AndroidApplication {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        Intent intent = getIntent();
        String code = intent.getExtras().getString("code");
        String encode = intent.getExtras().getString("encode");

        AndroidApplicationConfiguration config = new AndroidApplicationConfiguration();
        setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_LANDSCAPE);

        initialize(new Fitgaming(new DatabaseAndroid(this.getBaseContext()),code,encode), config);
    }
}

```

รูปที่ 3.38 การรับข้อมูลและตั้งค่า ส่งไปยังคลาส LibGDX

## 2) การขอข้อมูลจากฟิตบิต

ในการขอข้อมูลจากฟิตบิต ซึ่งจะต้องใช้ Oauth 2.0 ซึ่งเป็นการใช้ Https Request ในการติดต่อ ดังรูปที่ 3.39 และ 3.40

```
//for fitbit
public InputStream requestHttps(String url,String requestMethod
                               ,String requestHeader1,String requestParameter1
                               ,String requestHeader2,String requestParameter2
                               ,String urlParameter) {

    InputStream getData = null;

    try {
        Gdx.app.log(tag, "start try");
        urlx = new URL(url);
        Gdx.app.log(tag, "URL set");
        connection = (HttpsURLConnection) urlx.openConnection();
        Gdx.app.log(tag, "pass HttpURLConnection.. setting connection");

        connection.setDoInput(true);
        connection.setRequestMethod(requestMethod);
        if(requestHeader1 != null) {
            connection.setRequestProperty(requestHeader1, requestParameter1);
        }
        if(requestHeader2 != null) {
            connection.setRequestProperty(requestHeader2, requestParameter2);
        }

        Gdx.app.log(tag, "set Request");
        Gdx.app.log(tag, "connect");
        if(urlParameter != null) {
            connection.setDoOutput(true);
            OutputStreamWriter wr = new OutputStreamWriter(connection.getOutputStream());
            Gdx.app.log(tag, "OutputStream Connect");
            wr.write(urlParameter);
            wr.flush();
            wr.close();
        }

        Gdx.app.log(tag, "RequestOut");

        int response = connection.getResponseCode();
        String responseS = connection.getResponseMessage();
        String D = Integer.toString(response);

        if(response ==200) {
            getData = connection.getInputStream();
            Gdx.app.log(tag, "getData InputStream");
        }
    }
}
```

รูปที่ 3.39 เมธอดการเรียก Https Request (1)

```
else{
    getData = connection.getErrorStream();
    Gdx.app.log(tag, "Error input");

    BufferedReader reader = new BufferedReader(new InputStreamReader(getData));
    StringBuilder result = new StringBuilder();
    String line;
    while ((line = reader.readLine()) != null) {
        result.append(line);
    }
    String responseThing = result.toString();
    Gdx.app.log(tag, "Response Error");
    //Log.v(tag, "Response List");
    Gdx.app.log(tag, "respon :" + responseThing);

    //refresh token for fitbit
    Gdx.app.log(tag, "refresh true? "+refreshToken);
    if(!refreshToken){
        refreshToken = true;
        getRefreshToken();
    }
}
```

รูปที่ 3.40 เมธอดการเรียก Https Request(2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3) การขอ Access Token และ Refresh Token

จากข้อมูลที่ได้ในการลงทะเบียนแอปพลิเคชัน จะสามารถขอ Token ได้ที่ <https://api.fitbit.com/oauth2/token> โดยการ เรียกใช้เมธอด Https Request ที่สร้างไว้ โดยที่ urlParameter ที่ส่งไปจะต้องเขียนตามรูปแบบที่กำหนดไว้ ดังใน รูปที่ 3.41

```
tring urlParameter = "client_id="+client_id+"&grant_type=authorization_code" +
    "&redirect_uri="+hostapp;

if(!token) {
    Gdx.app.log(tag, "do Token");
    InputStream data = requestHttps(url, "POST"
        , "Authorization", "Basic " + getAutherEncoded
        , "Content-Type", "application/x-www-form-urlencoded"
        , urlParameter + "&" + code);
```

รูปที่ 3.41 การเรียกใช้เมธอด Https ในการติดต่อขอโทเค็น

เมื่อได้รับ JSON แล้ว ก็จะทำให้การดึงออกมาเพื่อไปใช้ดึงข้อมูลจากฟิตบิตไปใช้ในการเล่นเกมโดยใช้ JsonValue ซึ่งเป็นของ LibGDX โดยค่าที่ดึงจะต้องดึงตามตัวแปรที่ฟิตบิตใส่ไว้ดังในรูปที่ 3.42

```
if(!token) {
    Gdx.app.log(tag, "do Token");
    InputStream data = requestHttps(url, "POST"
        , "Authorization", "Basic " + getAutherEncoded
        , "Content-Type", "application/x-www-form-urlencoded"
        , urlParameter + "&" + code);

    JsonValue v = new JsonReader().parse(data);
    access_token = v.getString("access_token");
    expires_in = v.getInt("expires_in");
    refresh_token = v.getString("refresh_token");
    token_type = v.getString("token_type");
    user_id = v.getString("user_id");
```

รูปที่ 3.42 การดึงข้อมูลจากก้อน JSON ตามรูปแบบฟิตบิต

### 4) การขอข้อมูลต่างๆของผู้ใช้

ในการพัฒนาแอปพลิเคชันฟิตเกมมิ่ง 2 ได้มีการร้องขอข้อมูลจากฟิตบิต 3 ประเภทคือ

- Get User Info : ข้อมูลส่วนตัวของผู้ใช้จากฟิตบิต  
<https://api.fitbit.com/1/user/-/profile.JSON>
- Get Activities : ข้อมูลการออกกำลังกายจากฟิตบิต  
[https://api.fitbit.com/1/user/-/activities/date/"date".JSON](https://api.fitbit.com/1/user/-/activities/date/)

- Get Sleep data : ข้อมูลเวลาการนอนจากฟิตบิต  
[https://api.fitbit.com/1/user/-/sleep/date/"date".JSON](https://api.fitbit.com/1/user/-/sleep/date/)  
 “date” ใน url นี้หมายถึงเวลาที่ผู้พัฒนาต้องการดึงข้อมูล

โดยนำ URL ของข้อมูลที่ต้องการนำมาใช้ในการพัฒนาแอปพลิเคชันส่งเป็นพารามิเตอร์ให้ เมธอด requestHttps โดยเมธอดนี้จะร้องขอข้อมูลตาม url เฮดเดอร์รีควีส และพารามิเตอร์ที่ฟิตบิตได้กำหนดไว้ และดึงค่าตามรูปแบบ JSON ที่ฟิตบิตกำหนด โดยใช้คลาส JsonValue ไว้ไปเก็บใน Preference เพื่อนำไปโหลดใช้ในภายหลัง ดังในรูปที่ 3.43 และ 3.44

```
public void getUserData(){
    Gdx.app.log(tag, "getUserData");
    preferences = Gdx.app.getPreferences(ProfilePreference.PREFERENCE_NAME);

    InputStream getUserData = requestHttps("https://api.fitbit.com/1/user/-/profile.json",
        "GET", "Authorization", token_type+ " "+access_token, null, null, null);

    JsonValue v = new JsonReader().parse(getUserData);
    JsonValue vv = v.get("user");

    Gdx.app.log(tag, "All data:" + vv.toString());
    Gdx.app.log(tag, "ADStep:" + vv.getInt("averageDailySteps"));
    Gdx.app.log(tag, "weight:" + vv.getString("weight"));
    Gdx.app.log(tag, "height:" + vv.getString("height"));

    Gdx.app.log(tag, "weightunit:" + vv.getString("weightunit"));
    Gdx.app.log(tag, "heightunit:" + vv.getString("heightunit"));
}
```

รูปที่ 3.43 ตัวอย่างการดึงข้อมูลฟิตบิตส่วนตัวของผู้ใช้

```
preferences.putInteger(ProfilePreference.averageDailySteps, vv.getInt("averageDailySteps"));
preferences.putString(ProfilePreference.avatar, vv.getString("avatar"));
preferences.putString(ProfilePreference.avatar150, vv.getString("avatar150"));
preferences.putInteger(ProfilePreference.weight, vv.getInt("weight"));
preferences.putInteger(ProfilePreference.height, vv.getInt("height"));
preferences.putString(ProfilePreference.weightunit, vv.getString("weightunit"));
preferences.putString(ProfilePreference.heightunit, vv.getString("heightunit"));
preferences.putString(ProfilePreference.gender, vv.getString("gender"));
preferences.putString(ProfilePreference.displayName, vv.getString("displayName"));
preferences.flush();
```

รูปที่ 3.44 โค้ดดึงข้อมูลโปรไฟล์ของผู้ใช้จาก JSON

หลังจากทำการดึงข้อมูลแล้วก็จะต้องทำการตัดการเชื่อมต่อของฟิตบิตที่ยังค้างไว้ โดยใช้เมธอดดังรูปที่ 3.45 ถ้าไม่ได้ทำส่วนนี้การต่อข้อมูลจะทับถมกัน

```
public void disconnect(){
    if(connection != null){
        connection.disconnect();
    }
}
```

รูปที่ 3.45 เมธอดการตัดการเชื่อมต่อ

### 3.7 การพัฒนาแอปพลิเคชันในส่วนของเกม โดยใช้ LibGDX

#### 3.7.1 ทำการตั้งค่าคลาสหลักเพื่อรองรับการเปลี่ยนหน้าจอหลายจอ

ในการพัฒนาเกม Fitgaming 2 หน้าจอจะต้องทำการเรียกใช้หน้าจอหลายหน้าจอ เพื่อแสดงข้อมูลต่าง ๆ เช่น หน้าจอเลือกกระดับ เลือกเมือง หน้าจอเกม หน้าจอกราฟดูการนอนหลับ เป็นต้น ดังนั้นจึงจำเป็นต้องใช้คลาสหนึ่งเป็นหน้าจอหลัก โดยให้คลาสหลัก extends คลาส Game และคลาสหน้าจออื่น implements คลาส Screen ซึ่งทั้งสองคลาส เป็นคลาสที่อยู่ใน LibGDX

คลาสหลัก(คลาสหลักในที่นี้คือ Fitgaming) จะต้องมีเมธอดในการรองรับหน้าจออื่น ในเวลาเปลี่ยนหน้าจอ ดังในรูปที่ 3.46 ถึง 3.48

```
public class Fitgaming extends Game {
    public SpriteBatch batch;
    public Texture img;
    public BitmapFont font;
    private Preferences preferences;
    Boolean token;
    boolean refreshToken = false;
```

รูปที่ 3.46 คลาสหลัก Fitgaming extends คลาส Game

```
@Override
public Screen getScreen() {
    return super.getScreen();
}

@Override
public void render() {
    super.render();
}
```

รูปที่ 3.47 โค้ดที่ทำให้รองรับหน้าจอตามคลาสที่เปลี่ยน

```
@Override
public void create () {

    batch = new SpriteBatch();
    font = new BitmapFont();
```

รูปที่ 3.48 การตั้งค่าคลาสหลักใส่ส่วนของ create

ในการเปลี่ยนหน้าจอจากหน้าจอหลักไปหน้าจออื่นสามารถทำได้ด้วยโค้ด this.setScreen (new ชื่อคลาสที่จะไป(this)); ดังรูปที่ 3.49 ส่วนหน้าจออื่นที่ implement ด้วย Screen ต้องเขียนตัวแปรที่อ้างอิงคลาสหลัก จากนั้นก็เขียนโค้ดเปลี่ยนหน้าจอที่อยู่ใน คลาสหลัก โดยพารามิเตอร์ข้างในต้องใส่ตัวคลาสหลักลงไปด้วย ดังรูปที่ 3.50

```
@Override
public void create () {
    this.setScreen(new Profile(this));
}
```

รูปที่ 3.49 การเขียนโค้ดเปลี่ยนหน้าจอของคลาสหลัก

```
Fitgaming game;

public Levelchoose(Fitgaming gam) {
    this.game = gam;

    game.setScreen(new Profile(game));
}
```

รูปที่ 3.50 การเขียนเปลี่ยนหน้าจอของคลาส Screen อื่นๆ

### 3.7.2 การใช้คลาส Texture

ก่อนหน้าที่จะใช้คลาส Texture ได้ จะต้องทำการประกาศและตั้งค่า OrthographicCamera ก่อนเพื่อแสดงหน้าจอในส่วนตอนที่ render() ทำงาน ซึ่งตัวโค้ดสามารถเขียนได้ดังรูปที่ 3.51 และ 3.52 โดยตั้งค่าให้หน้ากล้องเท่ากับขนาดหน้าจอ

```
public LevelChoose(Fitgaming gam) {
    this.game = gam;
    camera = new OrthographicCamera();
    configureCamera();
}

public void configureCamera(){
    camera = new OrthographicCamera();
    if(Gdx.graphics.getHeight() < game.agdxgraphicGetWidth()) {
        //camera.setToOrtho(false, 800, 800 * Gdx.graphics.getWidth() / Gdx.graphics.getHeight());
        //camera.setToOrtho(false, 800, 800 * Gdx.graphics.getWidth() / Gdx.graphics.getHeight());
        camera.setToOrtho(false, game.agdxgraphicGetWidth(), Gdx.graphics.getHeight());
    }
    else{
        //camera.setToOrtho(false, 800 * Gdx.graphics.getWidth() / Gdx.graphics.getHeight(), 800);
        //camera.setToOrtho(false, 800 * Gdx.graphics.getWidth() / Gdx.graphics.getHeight(), 800);
        camera.setToOrtho(false, Gdx.graphics.getHeight(), game.agdxgraphicGetWidth());
    }
}
}
```

รูปที่ 3.51 การตั้งค่ากล้องในส่วนตอนที่สร้างคลาส

```
@Override
public void render(float delta) {

    Gdx.gl.glClearColor(0.9f, 0.9f, 0.9f, 1);
    Gdx.gl.glClear(GL20.GL_COLOR_BUFFER_BIT | GL20.GL_DEPTH_BUFFER_BIT);
    Gdx.gl.glEnable(GL20.GL_BLEND);

    camera.update();
    game.batch.setProjectionMatrix(camera.combined);
}
```

รูปที่ 3.52 การตั้งค่ากล้องในส่วนของ render

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การลงภาพพื้นหลังนั้นต้องใช้ฟังก์ชัน Texture ที่ LibGDX มีมาใช้ โดยการประกาศตัวแปร Texture ไว้ด้านนอกและสร้าง Texture หลังจากคลาสมีการถูกเรียกโดยพารามิเตอร์ของ Texture จะส่งไปตามเส้นทางของ asset ที่อยู่ในตัวโปรเจกของแอนดรอยด์ ดังรูปที่ 3.53

```
Texture rb;
Fitgaming game;

public LevelChoose(Fitgaming gam) {
    Lib.gam = gam;

    rb = new texture(gdx.files.Internal("Levelchoose/red_status.png"));
}
```

รูปที่ 3.53 การประกาศภาพ Texture

จากนั้นใช้โค้ดวาดรูป Texture ดังรูปที่ 3.54 ลงในเมธอด render() ที่จะเป็นเมธอดวนลูบหลังจากเมธอด create() ได้สิ้นสุดลง และจะทำการวนลูบจนกระทั่งออกคลาส โดยที่โค้ดจะเริ่มต้นวาดด้วยที่ batch.begin() จนถึง batch.end()

```
@Override
public void render(float delta) {
    town(level);
    Gdx.gl.glClear(GL20.GL_COLOR_BUFFER_BIT | GL20.GL_DEPTH_BUFFER_BIT);
    camera.update();
    game.batch.setProjectionMatrix(camera.combined);

    game.batch.begin();
    game.font.getData().setScale(s);
    game.batch.draw(bgRegionTown, 0, 0, Gdx.graphics.getWidth(), Gdx.graphics.getHeight());
    game.batch.end();
}
```

รูปที่ 3.54 ตัวอย่างการวาด Texture ใน render()

ซึ่งโค้ดของ batch draw จะส่งค่าพารามิเตอร์ตามลำดับดังนี้

Batch.draw (ตัวแปร Texture, ตำแหน่ง x, ตำแหน่ง y, ความกว้างรูป, ความสูงรูป)

ซึ่งความกว้างของรูปและความสูงของรูป สามารถปรับเปลี่ยนได้โดยไม่จำเป็น ต้องตามขนาดของรูปภาพจริงๆ ดังนั้น ถ้าเป็นการทำภาพพื้นหลัง ตำแหน่ง x และ y จะเป็น 0 และความยาวและกว้างของรูปเท่ากับความกว้างและความยาวของหน้าจอ ซึ่งสามารถเขียนโค้ดได้ดังนี้

Gdx.graphics.getWidth() เป็นโค้ดที่คืนค่าความกว้างของหน้าจอ

Gdx.graphics.getHeight() เป็นโค้ดที่คืนค่าความสูงของหน้าจอ

### 3.7.3 การเขียนตัวอักษร

โค้ดในการเขียนตัวอักษรเขียนลงในเมธอด render ซึ่งจะต้องเขียนอยู่ระหว่าง batch.begin() และ batch.end() เหมือนกับคลาส Texture ดังในรูปที่ 3.55 โดยที่โค้ด game.font.getData().setScale(5) เป็นตัวปรับขนาดฟอนต์ทั้งหมด เพราะเป็นตัวแปรที่อ้างอิงจากตัวเดียวกัน

```

@Override
public void render(float delta) {
    town(level);
    Gdx.gl.glClear(GL20.GL_COLOR_BUFFER_BIT | GL20.GL_DEPTH_BUFFER_BIT);

    camera.update();
    game.batch.setProjectionMatrix(camera.combined);

    game.batch.begin();
    game.font.getData().setScale(5);

    if(game.sectionClear == 1) {
        game.batch.draw(bgRegionTown, 0, 0, Gdx.graphics.getWidth(), Gdx.graphics.getHeight());

        TownChooseButtons();

        game.font.draw(game.batch, "Bangkok", 82 * (Gdx.graphics.getWidth() / 100), (Gdx.graphics.getHeight() / 2)); //1675, 600
        game.font.draw(game.batch, "Petchburi", 36 * (Gdx.graphics.getWidth() / 100), 3 * (Gdx.graphics.getHeight() / 5)); //825
        game.font.draw(game.batch, "Ayutthaya", 47 * (Gdx.graphics.getWidth() / 100), (Gdx.graphics.getHeight() / 4)); //1275, 3
        game.font.draw(game.batch, "Nakhorn Pathom", 55 * (Gdx.graphics.getWidth() / 100), 7 * (Gdx.graphics.getHeight() / 10));

    }

    game.batch.end();
}

```

รูปที่ 3.55 การเขียนโค้ดตัวอักษร

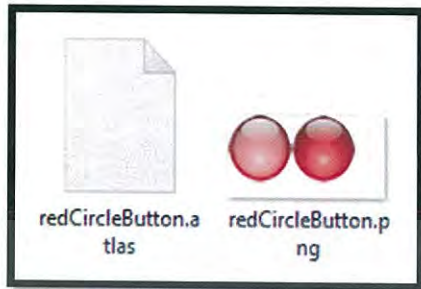
### 3.7.4 การใช้คลาส Button

ในการทำคลาส Button จะต้องมีไฟล์ 3 ไฟล์ที่จะต้องทำก่อน คือไฟล์ JSON เพื่อทำ Skin ของปุ่ม Button ไฟล์ตัวเชื่อมที่ JSON จะดึงภาพมาใช้ และไฟล์ spritesheet ที่ไฟล์ตัวเชื่อมจะดึงมาใช้ จะต้องมีการเคลื่อนไหวของปุ่มก่อน ทั้งภาพปุ่มตอนที่ยังไม่กดและกดแล้วเพื่อให้ผู้ใช้แอปพลิเคชันได้รู้สึกถึงการตอบสนอง ซึ่งการสร้างปุ่มนั้นทั้งสองภาพจะต้องมีขนาดเท่ากัน ดังรูปที่ 3.56



รูปที่ 3.56 รูปที่กดปุ่มแล้ว และ รูปที่ยังไม่กดปุ่ม

จากนั้นใช้โปรแกรม gdx-texturepacker.jar มารวมภาพ เพื่อให้ได้ไฟล์สองไฟล์ ดังรูปที่ 3.57 โดยที่ภาพในไฟล์ .atlas จะเก็บชื่อไฟล์ภาพที่รวมไว้ ดังภาพที่ 3.58 จึงเขียนโค้ด JSON ของปุ่มดังภาพที่ 3.59 โดยที่ชื่อของไฟล์ JSON และ atlas ต้องชื่อเหมือนกัน



```

1 |
2 | redCircleButton.png
3 | format: RGBA8888
4 | filter: Nearest,Nearest
5 | repeat: none
6 | red_circle_button_down
7 | rotate: false
8 | xy: 1, 1
9 | size: 50, 50
10 | orig: 50, 50
11 | offset: 0, 0
12 | index: -1
13 | red_circle_button_up
14 | rotate: false
15 | xy: 53, 1
16 | size: 50, 50
17 | orig: 50, 50
18 | offset: 0, 0
19 | index: -1

```

รูปที่ 3.57 ไฟล์ที่ได้มาจากโปรแกรม gdx-Texture

รูปที่ 3.58 โค้ดในไฟล์ .atlas

```

1 | {
2 |   com.badlogic.gdx.graphics.g2d.BitmapFont: { default-font: { file: font.fnt } },
3 |   com.badlogic.gdx.graphics.Color: {
4 |     green: { a: 1, b: 0, g: 1, r: 0 },
5 |     white: { a: 1, b: 1, g: 1, r: 1 },
6 |     red: { a: 1, b: 0, g: 0, r: 1 },
7 |     black: { a: 1, b: 0, g: 0, r: 0 }
8 |   },
9 |   com.badlogic.gdx.scenes.scene2d.ui.Buttons$ButtonStyle: {
10 |     default: { down:red_circle_button_down, up:red_circle_button_up },
11 |     toggle: { down: red_circle_button_down, checked: red_circle_button_down, up: red_circle_button_up }
12 |   },
13 |   com.badlogic.gdx.scenes.scene2d.ui.TextButtons$TextButtonStyle: {
14 |     default: { down: red_circle_button_down, up: red_circle_button_up, font: default-font, fontColor: black },
15 |     toggle: { down: red_circle_button_down, up: red_circle_button_up,
16 |       checked: red_circle_button_down, font: default-font, fontColor: black, downFontColor: white }
17 |   }
18 | }

```

รูปที่ 3.59 ไฟล์ JSON สำหรับปุ่ม

เมื่อได้ครบทั้ง 3 ไฟล์แล้วกลับมาที่ส่วนการทำปุ่มของโปรแกรม ใช้คลาส Stage เพื่อทำพื้นที่รองรับของปุ่ม และใช้คลาส Skin โดยให้พารามิเตอร์เป็นเส้นทางของไฟล์ JSON ของปุ่ม ดังรูปที่ 3.60 ถึง 3.62

```

Fitgaming game;
public LevelChoose(Fitgaming gam) {
    this.game = gam;

    section = new Stage();
    town = new Stage();

    redCircleButton = new Skin(Gdx.files.internal("ButtonsJson/redCircleButton.json"));
    darkGreenSquareButton = new Skin(Gdx.files.internal("ButtonsJson/darkGreenSquareButton.json"));
    greenSquareButton = new Skin(Gdx.files.internal("ButtonsJson/greenSquareButton.json"));
    greenCircleButton = new Skin(Gdx.files.internal("ButtonsJson/greenCircleButton.json"));
    blackCircleButton = new Skin(Gdx.files.internal("ButtonsJson/blackCircleButton.json"));

    rb = new Texture(Gdx.files.internal("LevelChoose/red_status.png"));

    bgTown = new Texture(Gdx.files.internal("LevelChoose/central.png"));
    bgRegionTown = new TextureRegion(bgTown, 0, 0, bgTown.getWidth(), bgTown.getHeight());

    sq = new Texture(Gdx.files.internal("LevelChoose/square_green_glossy.png"));
    sectionLock = new Texture(Gdx.files.internal("LevelChoose/square_black_locked.png"));

    camera = new OrthographicCamera();
    configureCamera();

    ButtonsForSectionChoose();
    InputMultiplexer inputMultiplexer = new InputMultiplexer();
    inputMultiplexer.addProcessor(section);
    inputMultiplexer.addProcessor(town);
    Gdx.input.setInputProcessor(inputMultiplexer);
}

```

รูปที่ 3.60 ประกาศ Skin Stage และตั้งค่าให้ stage สามารถกดได้

```

private void ButtonsForSectionChoose(){
    section1 = new Button(greenSquareButton);
    //section1.add("Central");
    section1.setWidth(sq.getWidth() / 2);
    section1.setHeight(sq.getHeight() / 7);
    section1.setPosition(game.adxgraphic.getWidth() / 100, (game.adxgraphic.getHeight() - 100) - (sq.getHeight() / 7));
    section1.addListener(new ClickListener() {
        @Override
        public void touchup(InputEvent event, float x, float y, int pointer, int button) {
            Gdx.app.log("button", "touchup");
            town.clear();
            game.sectionClear = 1;
            super.touchup(event, x, y, pointer, button);
        }
    });
    section.addActor(section1);
}

```

รูปที่ 3.61 โค้ดเขียนปุ่ม Button

```

@Override
public void render(float delta) {
    town(level);
    Gdx.gl.glClearColor(0.9f, 0.9f, 0.9f, 1);
    Gdx.gl.glClear(GL20.GL_COLOR_BUFFER_BIT | GL20.GL_DEPTH_BUFFER_BIT);
    Gdx.gl.glEnable(GL20.GL_BLEND);

    camera.update();
    game.batch.setProjectionMatrix(camera.combined);

    game.batch.begin();
    stage.draw();
    forButtons.draw();
}

```

รูปที่ 3.62 การใช้พื้นที่ Stage ให้แสดงรูป Button

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.7.5 การใช้คลาส Window

ในการใช้คลาส Window จะต้องทำการใช้ไฟล์ที่เก็บข้อมูลในรูปแบบ JSON (.json) ไฟล์ atlas และไฟล์รวมภาพ เหมือนกับ Button แต่โค้ดใน JSON จะเขียนแตกต่างกัน ดังรูปที่ 3.63 จากนั้นจะสามารถเขียนโค้ด Window ได้ดังรูปที่ 3.64 และ Add เข้าตัวแปรจากคลาส Stage

```
{
  com.badlogic.gdx.graphics.g2d.BitmapFont: { default-font: { file: font.fnt } },
  com.badlogic.gdx.graphics.Color: {
    green: { a: 1, b: 0, g: 1, r: 0 },
    white: { a: 1, b: 1, g: 1, r: 1 },
    red: { a: 1, b: 0, g: 0, r: 1 },
    black: { a: 1, b: 0, g: 0, r: 0 }
  },
  com.badlogic.gdx.scenes.scene2d.ui.Window$WindowStyle: {
    default: { titleFont: default-font, background: Viewscore_Window, titleFontColor: white },
    dialog: { titleFont: default-font, background: _Window, titleFontColor: white, stageBackground: Viewscore_Window }
  },
  com.badlogic.gdx.scenes.scene2d.ui.List$ListStyle: {
    default: { fontColorUnselected: white, selection: _Window, fontColorSelected: white }
  },
  com.badlogic.gdx.scenes.scene2d.ui.Label$LabelStyle: {
    default: { font: default-font, fontColor: white }
  }
}
```

รูปที่ 3.63 ไฟล์ JSON ของ Window

```
Skin skin = new Skin(Gdx.files.internal("skinpack/skinpack.json"));
Window window = new Window("SleepView", skin);
window.setPosition(3 * game.aGdxgraphicGetWidth / 30
, 116 * (game.aGdxgraphicGetHeight / 1000) * (7-day)); //first 3+
window.setWidth((float) hours * game.aGdxgraphicGetWidth / 30);
window.setHeight(game.aGdxgraphicGetHeight / 8);
window.top().left().padLeft(30).padTop(50).padRight(30).padBottom(50).align(3);
stage.addActor(window);
```

รูปที่ 3.64 โค้ดการเขียนคลาส Window

### 3.7.6 การเขียนโค้ดในส่วนของ LevelChoose

โครงสร้างในการทำหน้าที่เลือกภาพและเลือกจังหวัดจะแบ่งเป็นสองส่วน ส่วนแรกคือส่วนของการประกาศตัวแปรเวลาสร้างคลาส แล้วส่วนของการตั้งค่า Orthographic Camera แล้วปุ่มที่ใช้กด ดังรูปที่ 3.65 และ 3.66 และส่วนของ render ที่ทำให้ภาพเคลื่อนไหวได้ต่อเนื่อง ดังรูป 3.67 ซึ่งในส่วนของปุ่มนั้นจะมีการทำงานเป็นสองขั้นตอนคือ การเลือกภาคก่อน และเลือกจังหวัดหลังจาก โดยที่จะมีการกำหนดตามค่าของตัวแปรภาค ดังรูปที่ 3.68 กับ 3.69 ในการเลือกจังหวัดก็จะมีเงื่อนไขใน ดังรูปที่ 3.70 กับ 3.71

```

Fitgaming game;
public LevelChoose(Fitgaming gam) {
    this.game = gam;

    section = new Stage();
    town = new Stage();

    redCircleButton = new Skin(Gdx.files.internal("ButtonsJson/redCircleButton.json"));
    darkGreenSquareButton = new Skin(Gdx.files.internal("ButtonsJson/darkGreenSquareButton.json"));
    greenSquareButton = new Skin(Gdx.files.internal("ButtonsJson/greenSquareButton.json"));
    greenCircleButton = new Skin(Gdx.files.internal("ButtonsJson/greenCircleButton.json"));
    blackCircleButton = new Skin(Gdx.files.internal("ButtonsJson/blackCircleButton.json"));

    rb = new Texture(Gdx.files.internal("LevelChoose/red_status.png"));

    bgTown = new Texture(Gdx.files.internal("LevelChoose/central.png"));
    bgRegionTown = new TextureRegion(bgTown, 0, 0, bgTown.getWidth(), bgTown.getHeight());

    sq = new Texture(Gdx.files.internal("LevelChoose/square_green_glossy.png"));
    sectionLock = new Texture(Gdx.files.internal("LevelChoose/square_black_locked.png"));

    camera = new OrthographicCamera();
    configureCamera();

    ButtonsForSectionChoose();
    InputMultiplexer inputMultiplexer = new InputMultiplexer();
    inputMultiplexer.addProcessor(section);
    inputMultiplexer.addProcessor(town);
    Gdx.input.setInputProcessor(inputMultiplexer);
}

```

รูปที่ 3.65 โค้ดประกาศการกวดปุ่ม ภาพ และการกวดปุ่มของคลาส Stage

```

public void configureCamera(){
    if(Gdx.graphics.getHeight() < Gdx.graphics.getWidth()) {
        //camera.setToOrtho(false, 800, 800 * Gdx.graphics.getWidth() / Gdx.graphics.getHeight());
        //camera.setToOrtho(false, 600, 800 * Gdx.graphics.getWidth() / Gdx.graphics.getHeight());
        camera.setToOrtho(false, Gdx.graphics.getWidth(), Gdx.graphics.getHeight());
    }
    else{
        //camera.setToOrtho(false, 800 * Gdx.graphics.getWidth() / Gdx.graphics.getHeight(), 800);
        //camera.setToOrtho(false, 800 * Gdx.graphics.getWidth() / Gdx.graphics.getHeight(), 600);
        camera.setToOrtho(false, Gdx.graphics.getHeight(), Gdx.graphics.getWidth());
    }
}

```

รูปที่ 3.66 โค้ดตั้งค่าขนาดหน้าจอ ด้วย OrthographicCamera

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

@Override
public void render(float delta) {
    town.level();
    Gdx.gl.glClearColor(GL20.GL_COLOR_BUFFER_BIT | GL20.GL_DEPTH_BUFFER_BIT);

    camera.update();
    game.batch.setProjectionMatrix(camera.combined);

    game.batch.begin();
    game.font.getData().setScale(5);

    if (game.sectionClear == 1) {
        game.batch.draw(bgRegionTown, 0, 0, Gdx.graphics.getWidth(), Gdx.graphics.getHeight());

        TownChooseButtons();
        game.font.draw(game.batch, "Bangkok", 82 * (Gdx.graphics.getWidth() / 100), (Gdx.graphics.getHeight() / 2)); //1675, 600
        game.font.draw(game.batch, "Petchburi", 36 * (Gdx.graphics.getWidth() / 100), 3 * (Gdx.graphics.getHeight() / 5)); //825, 800
        game.font.draw(game.batch, "Ayutthaya", 47 * (Gdx.graphics.getWidth() / 100), (Gdx.graphics.getHeight() / 4)); //1275, 300
        game.font.draw(game.batch, "Makhorn Pathon", 55 * (Gdx.graphics.getWidth() / 100), 7 * (Gdx.graphics.getHeight() / 10)); //608, 350
    }

    game.batch.end();

    section.draw();
    section.act();

    town.draw();

    //town.act();
    game.batch.begin();
    game.font.draw(game.batch, "Central", 7 * (Gdx.graphics.getWidth() / 100), (game.ogdxgraphicHeight-100) - sq.getHeight() / 7 * sq.getHeight() / 12);
    game.font.draw(game.batch, "Southern", 6 * (Gdx.graphics.getWidth() / 100), (game.ogdxgraphicHeight-100) - (2 * sq.getHeight() / 7) * sq.getHeight() / 12);
    game.font.draw(game.batch, "Western", 6 * (Gdx.graphics.getWidth() / 100), (game.ogdxgraphicHeight-100) - (3 * sq.getHeight() / 7) * sq.getHeight() / 12);
    game.font.draw(game.batch, "Northern", 8 * (Gdx.graphics.getWidth() / 100), (game.ogdxgraphicHeight-100) - (4 * sq.getHeight() / 7) * sq.getHeight() / 12);
    game.font.draw(game.batch, "Northeast", 3 * (Gdx.graphics.getWidth() / 100), (game.ogdxgraphicHeight-100) - (5 * sq.getHeight() / 7) * sq.getHeight() / 12);
    game.font.draw(game.batch, "Eastern", 7 * (Gdx.graphics.getWidth() / 100), (game.ogdxgraphicHeight-100) - (6 * sq.getHeight() / 7) * sq.getHeight() / 12);

    game.batch.end();
}

```

รูปที่ 3.67 การ render ในหน้าจอของ LevelChoose

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Button section1,section2,section3,section4,section5,section6;
private void ButtonsForSectionChoose(){

    section1 = new Button(greenSquareButton);
    //section1.add("Central");
    section1.setWidth(sq.getWidth() / 2);
    section1.setHeight(sq.getHeight() / 7);
    section1.setPosition(game.gdxgraphicGetWidth / 100, (game.gdxgraphicGetHeight-100) - (sq.getHeight() / 7));
    section1.addListener(new ClickListener() {
        @Override
        public void touchUp(InputEvent event, float x, float y, int pointer, int button) {
            Gdx.app.log("button", "touchUp");
            town.clear();
            game.sectionClear =1;
            super.touchUp(event, x, y, pointer, button);
        }
    });
    section.addActor(section1);

    if(section ==1) {
        section2 = new Button(greenSquareButton);
    }else{
        section2 = new Button(darkGreenSquareButton);
    }
    section2.setWidth(sq.getWidth() / 2);
    section2.setHeight(sq.getHeight() / 7);
    section2.setPosition(game.gdxgraphicGetWidth / 100, (game.gdxgraphicGetHeight-100) - (2*sq.getHeight() / 7));
    section2.addListener(new ClickListener() {
        @Override
        public void touchUp(InputEvent event, float x, float y, int pointer, int button) {
            Gdx.app.log("button", "touchUp");
            //pressed = false;
            town.clear();
            game.sectionClear =2;
            super.touchUp(event, x, y, pointer, button);
        }
    });
    section.addActor(section2);

    if(section >=2) {
        section3 = new Button(greenSquareButton);
    }
    }else{
        section3 = new Button(darkGreenSquareButton);
    }
    section3.setWidth(sq.getWidth() / 2);
    section3.setHeight(sq.getHeight() / 7);
    section3.setPosition(game.gdxgraphicGetWidth / 100, (game.gdxgraphicGetHeight-100) - (3*sq.getHeight() / 7));
    section3.addListener(new ClickListener() {
        @Override
        public void touchUp(InputEvent event, float x, float y, int pointer, int button) {
            Gdx.app.log("button", "touchUp");
            //pressed = false;
            town.clear();
            game.sectionClear =3;
            super.touchUp(event, x, y, pointer, button);
        }
    });
    section.addActor(section3);
}

```

รูปที่ 3.68 ประกาศสร้างปุ่มเลือกภาค ทั้ง 6 ภาค (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

section3.setWidth(sq.getWidth() / 2);
section3.setHeight(sq.getHeight() / 7);
section3.setPosition(game.adxgraphicGetWidth / 100, (game.adxgraphicGetHeight-100) - (3*sq.getHeight() / 7));
section3.addListener(new ClickListener() {

@Override
public void touchUp(InputEvent event, float x, float y, int pointer, int button) {
    Gdx.app.log("button", "touchup");
    //pressed = false;
    town.clear();
    game.sectionClear = 3;
    super.touchUp(event, x, y, pointer, button);
}
});
section.addActor(section3);

if(section >=3) {
} else {
    section4 = new Button(darkGreenSquareButton);
}

if(section >=4) {
    section5 = new Button(greenSquareButton);
} else {
    section5 = new Button(darkGreenSquareButton);
}

section.addActor(section5);

if(section >=5) {
} else {
    section6 = new Button(darkGreenSquareButton);
}

section6.setWidth(sq.getWidth() / 2);
section6.setHeight(sq.getHeight() / 7);
section6.setPosition(game.adxgraphicGetWidth / 100, (game.adxgraphicGetHeight-100) - (6*sq.getHeight() / 7));
section6.addListener(new ClickListener() {

@Override
public void touchUp(InputEvent event, float x, float y, int pointer, int button) {
    Gdx.app.log("button", "touchup");
    //pressed = false;
    town.clear();
    game.sectionClear = 6;
    super.touchUp(event, x, y, pointer, button);
}
});
section.addActor(section6);

```

รูปที่ 3.69 ประกาศสร้างปุ่มเลือกภาค ทั้ง 6 ภาค (2)

```

Preferences preference;
Button town1,town2,town3,town4;
private void TownChooseButtons(){
    //skin = new Skin(Gdx.files.internal(""));
    //Dialog tell = new Dialog("Can't Access this level",skin);
    preference = Gdx.app.getPreferences(ActiviyPreference.PREFERENCE_NAME);
    if(game.townNo<=0) {
        town1 = new Button(greencircleButton);
    }
    else {
        town1 = new Button(blackCircleButton);
    }
    town1.setWidth(150);
    town1.setHeight(150);
    town1.setPosition(85*(game.adxgraphicGetWidth / 100),
        (Gdx.graphics.getHeight() / 2));
    town1.addListener(new ClickListener() {

@Override
public boolean touchDown(InputEvent event, float x, float y, int pointer, int button) {
    Gdx.app.log("button", "touchup");
    //pressed = false;
    Gdx.app.log("level", "a");
    game.setScreen(new Town(game,0));
    return super.touchDown(event, x, y, pointer, button);
}

});

if(game.townNo<=0) {
    town2 = new Button(greencircleButton);
}
else {
    town2 = new Button(blackCircleButton);
}
}

```

รูปที่ 3.70 สร้างปุ่มเลือกจังหวัด ของภาคกลาง (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(game.townNo<=1) {
    town2 = new Button(greenCircleButton);
}
else{
    town2 = new Button(blackCircleButton);
}
town2.setWidth(150);
town2.setHeight(150);
town2.setPosition(40 * (game.aGdxgraphicGetWidth / 100),
    3*(Gdx.graphics.getHeight() / 5));
town2.addListener(new ClickListener() {
    @Override
    public boolean touchDown(InputEvent event, float x, float y, int pointer, int button) {
        Gdx.app.log("button", "touchUp");
        Gdx.app.log("level", "1");
        if (preference.getInteger(ActivityResult.fairlyActiveMinutes) > 15) {
            game.setScreen(new Town(game, 1));
        }else{
        }
        super.touchUp(event, x, y, pointer, button);
        //pressed = true;

        return super.touchDown(event, x, y, pointer, button);
    }
});

if(game.townNo<=2) {
    town3 = new Button(greenCircleButton);
}
else{
    town3 = new Button(blackCircleButton);
}
town3.setWidth(150);
town3.setHeight(150);
town3.setPosition((game.aGdxgraphicGetWidth / 2),
    (Gdx.graphics.getHeight() / 4));
town3.addListener(new ClickListener() {
    @Override
    public boolean touchDown(InputEvent event, float x, float y, int pointer, int button) {
        Gdx.app.log("button", "touchUp");
        //pressed = false;
        Gdx.app.log("level", "2");
        game.setScreen(new Town(game,2));
        super.touchUp(event, x, y, pointer, button);
        return super.touchDown(event, x, y, pointer, button);
    }
});

if(game.townNo<=3) {
    town4 = new Button(greenCircleButton);
}
else{
    town4 = new Button(blackCircleButton);
}
town4.setWidth(150);
town4.setHeight(150);
town4.setPosition(65 * (game.aGdxgraphicGetWidth / 100),
    7*(Gdx.graphics.getHeight() / 10));
town4.addListener(new ClickListener() {
    @Override
    public boolean touchDown(InputEvent event, float x, float y, int pointer, int button) {
        Gdx.app.log("button", "touchDown");
        Gdx.app.log("level", "3");
        game.setScreen(new Town(game,3));
        return super.touchDown(event, x, y, pointer, button);
    }
});

if(game.sectionClear ==1) {
    town.addActor(town1);
    town.addActor(town2);
    town.addActor(town3);
    town.addActor(town4);
}

```

รูปที่ 3.71 สร้างปุ่มเลือกจังหวัด ของภาคกลาง (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.7.7 การทำตัวละคร Player

การทำตัวละครจะมีขั้นตอนใหญ่ๆดังนี้

- 1) การวาดภาพตัวละครวิ่ง
- 2) โฉดภาพเคลื่อนไหวลงในเกม
- 3) สร้างพื้นที่สัมผัสของตัวละคร
- 4) ทำการกระโดดของตัวละคร

#### 1) การวาดภาพตัวละครวิ่ง

ในการทำตัวละครวิ่ง จะต้องทำภาพเคลื่อนไหวที่ใช้ในเกม (Sprite) และนำภาพเหล่านั้นมารวมกันเป็นกลุ่มของภาพ (Sprite sheet) ที่จะเป็นภาพการเคลื่อนไหวของตัวละครที่จะวิ่งในแต่ละเฟรม ซึ่งในแต่ละเฟรมต้องมีขนาดเท่ากัน และพื้นหลังของตัวละครต้องโปร่งใส เพื่อให้ไม่ให้อาภาพพื้นหลังตัวละครไปซ้อนทับกับภาพพื้นหลังของเกม ตัวสไปรท์ซิทควรจะมีภาพครบทุกช่อง ไม่ควรมีส่วนไหนขาดหาย โดยวิธีการทำในแต่ละภาพมีดังนี้

- วาดภาพตัวละครการเคลื่อนไหวครั้งแรก บนพื้นหลังใส
- ลงสีให้เรียบร้อย
- ทำภาพใสเพื่อให้ภาพในเฟรมการเคลื่อนไหวของตัวละครถัดมาวาดได้ง่ายขึ้น
- ขึ้นอีก Layer ใหม่ เพื่อการวาดภาพใหม่

ทำซ้ำตามขั้นตอนไปจนกระทั่งได้ตามภาพที่ต้องการ ดังรูปที่ 3.72 โดยที่เฟรมสุดท้ายของการเคลื่อนไหว ต้องทำให้สอดคล้องกับภาพเฟรมแรก เพื่อให้ภาพเคลื่อนไหวสามารถเคลื่อนที่ไปมาอย่างต่อเนื่องและลื่นไหล



รูปที่ 3.72 ขั้นตอนการทำภาพแอนิเมชันในแต่ละเฟรม

ภาพตัวละครไหวของตัวละครในเกมนี้ในแต่ละภาพจะเป็น ดังรูปที่ 3.73 จากนั้นทำการคัดและนำมารวมเป็นภาพเดียวกันโดยเรียงให้เป็นขนาด 3x4 จะได้สไปรท์ซิทดังรูปที่ 3.74



รูปที่ 3.73 รูปภาพทั้ง 16 เฟรม



รูปที่ 3.74 สไปรท์ซีทสำหรับการเคลื่อนไหวของตัวละคร

2) โค้ดภาพเคลื่อนไหวลงในเกม

ในส่วนของการโค้ด จะต้องเขียนตามการตัดแบ่งการวิ่งของตัวละคร ตัวสไปรท์ซีท ตามแนวตั้งและแนวนอน และนำมาใส่อาร์เรย์ของ TextureRegion จากนั้นจึงนำมาใช้ในคลาส Animation ดังรูปที่ 3.75

```
private void SpriteSettoAnimation(Texture spritesheet){
    runSheet = spritesheet;
    TextureRegion[][] tmp = TextureRegion.split(runSheet, runSheet.getWidth()/FRAME_COLUMNS, runSheet.getHeight()/FRAME_ROWS);
    runFrames = new TextureRegion[FRAME_COLUMNS*FRAME_ROWS]; //TextureRegion[]
    int index = 0;
    for(int i = 0; i<FRAME_ROWS ; i++){
        for ( int j = 0; j<FRAME_COLUMNS ; j++){
            runFrames[index++] = tmp[i][j];
        }
    }
    runAnimation = new Animation(0.25f,runFrames);
}
```

รูปที่ 3.75 โค้ดการใช้คลาส Animation

3) สร้างพื้นที่สัมผัสของตัวละคร

หลังจากประกาศภาพเคลื่อนไหวแล้ว ต้องทำการกำหนดขอบเขตพื้นที่สัมผัสของตัวละครในการชนกับสิ่งกีดขวาง โดยใช้ Rectangle ของ LibGDX กำหนดตำแหน่งและขนาดของภาพ และใช้ Vector2 เพื่อกำหนดความเร็วในการวิ่งและความสูงของการกระโดดของตัวละครในขณะที่ไม่ได้ทำการกระโดดข้ามสิ่งกีดขวาง ดังรูปที่ 3.76

```
private void resetGame(){
    configureCamera();
    manPosition = new Rectangle(200,DEFAULT_HEIGHT,200,250);
    manvelocity = new Vector2(500,0);
}
```

รูปที่ 3.76 โค้ดในการกำหนดตำแหน่งตัวละคร และ การเคลื่อนไหวของตัวละคร

เมื่อกำหนดตำแหน่งตัวละคร แล้วทำการวาดตัวแอนิเมชันนี้ลงในเมธอด render โดยตั้งตำแหน่งที่วาดไว้ที่เดียวกันกับ Rectangle ที่กำหนดไว้ โดยให้ตัว time เป็นเวลาที่ผ่านไปเพื่อให้แอนิเมชันได้เคลื่อนไหวไปตามเวลา และการแสดงตัวละครบนหน้าจอ จะต้องใช้เมธอด render ซึ่งมีการใช้คลาส OrthographicCamera ในการแสดงหน้าจอ โดยตำแหน่งของตัวละครจะอยู่บน Rectangle ที่กำหนดไว้ตั้งแต่ต้น และให้ time เป็นช่วงเวลาที่ใช้ในการเคลื่อนไหวระหว่างเฟรม ซึ่งจะเคลื่อนไหวตลอดเวลา ดังรูปที่ 3.77

```
@Override
public void render(float delta) {
    town(level);
    Gdx.gl.glClearColor(0.9f, 0.9f, 0.9f, 1);
    Gdx.gl.glClear(GL20.GL_COLOR_BUFFER_BIT | GL20.GL_DEPTH_BUFFER_BIT);
    Gdx.gl.glEnable(GL20.GL_BLEND);

    camera.update();
    game.batch.setProjectionMatrix(camera.combined);

    game.batch.begin();
    game.batch.draw(runAnimation.getKeyFrame(time, true), manPosition.x, manPosition.y - DEFAULT_HEIGHT + movePosition, manPosition.width, manPosition.height);
    game.batch.end();
}
```

รูปที่ 3.77 โค้ดในการลงภาพเคลื่อนไหวของ Player

หลังจากการวางภาพตัวละครแล้ว ตัวละครต้องเคลื่อนไหว เช่นเดียวกับตัวกล้อง หรือ คลาส OrthographicCamera เพื่อให้ภาพหน้าจอติดตามตัวละคร เมื่อตัวละครไปถึงเป้าหมายที่กำหนดไว้ กล้องจะทำการหยุดและให้ตัวละครเคลื่อนไหวจนหลุดหน้าจอและทำการโชว์หน้าต่างหน้าคะแนน ซึ่งเขียนโค้ดได้ดังรูปที่ 3.78 และต้องใส่โค้ดลงในส่วนของ render เพื่อให้เกิดการทำงานตลอดเวลา

```

if(manPosition.x >= distance_goal){
    winstatement(true);
}

time += Gdx.graphics.getDeltaTime();

manPosition.x += manVelocity.x * Gdx.graphics.getDeltaTime();
manPosition.y += manVelocity.y * Gdx.graphics.getDeltaTime();
manVelocity.y -= 1000 * Gdx.graphics.getDeltaTime();

if(!win) {
    move = manPosition.x - 100;
    //score calculated
    thisGameScore += (manPosition.x/1000);
    if(thisGameScore >= 100){
        scoreInGame ++1;
        thisGameScore = 0;
    }

    if(sEnergy > 0) {
        sEnergy -- 0.1;
    }
    else{
        winstatement(false);
    }

    if (Gdx.graphics.getWidth() < distance_goal) {
        camera.translate(manVelocity.x * Gdx.graphics.getDeltaTime(), 0);
    }
}
else{
    if(Gdx.input.isTouched()) {
        game.callBack();
        //game.touchUp
        game.setScreen(new LevelChoose(game));
    }
}

stage.draw();
forButtons.draw();

```

รูปที่ 3.78 การคำนวณการเคลื่อนไหวกของตัวละคร กล้อง และเส้นชัย

#### 4) การกระโดดของตัวละคร

หลังจากทำให้ตัวละครเคลื่อนไหวกเองอัตโนมัติแล้ว จะต้องทำให้ตัวละครกระโดดได้ตามที่ผู้เล่นกดด้วย โดยระบบการกระโดดจะทำงานเมื่อตัวละครอยู่กับพื้น และเวลากดปุ่ม จะทำการสะสมพลังการกระโดด และเวลาปล่อยปุ่มจะทำการกระโดด ดังรูปที่ 3.79 ซึ่งส่วนนี้ต้องใส่ในคอนสตรัคเตอร์ หรือ create ของคลาส

เมื่อเขียนปุ่มกระโดดขึ้นแล้ว จะต้องเขียนตำแหน่งตอนลงให้ตัวละครลงด้วย เพื่อให้การกระโดดของตัวละครนั้นสมจริงดังรูปที่ 3.80 ถ้าในส่วนนี้ไม่ได้ถูกเขียนตัวละครจะลอยอยู่กลางอากาศ

```

bJump = new Button(skinButtonJump);
bJump.setWidth(game.aGdxgraphicGetWidth / 13);
bJump.setHeight(game.aGdxgraphicGetHeight / 5);
bJump.setPosition(game.aGdxgraphicGetWidth - bJump.getWidth(), 0);
bJump.addListener(new ClickListener() {

@Override
public void touchDragged(InputEvent event, float x, float y, int pointer) {
    timeTouch += 15;
    Gdx.app.log("touch Dragged", Float.toString(timeTouch));
    super.touchDragged(event, x, y, pointer);
}

@Override
public void touchUp(InputEvent event, float x, float y, int pointer, int button) {
    msg = "touch up";
    Gdx.app.log("ooo", msg);
    if (timeTouch > jumpMax) {
        timeTouch = jumpMax;
    }
    if (manPosition.y == 0) {
        manVelocity.y = jumpDefault + timeTouch; //Default = 200
        if (manVelocity.y <= 0) {
            manVelocity.y = jumpDefault;
            Gdx.app.log("Jump Power Adjust", "Adjust");
        }
    }
    Gdx.app.log("Jump Power", Float.toString(manVelocity.y));
    timeTouch = 0;

    super.touchUp(event, x, y, pointer, button);
}
});

```

รูปที่ 3.79 โค้ดในการใส่ปุ่มกระโดดและวิธีการคำนวณ

```

manPosition.y += manVelocity.y * Gdx.graphics.getDeltaTime();
manVelocity.y -= 1000 * Gdx.graphics.getDeltaTime();

if (manPosition.y < 0) {
    manPosition.y = 0;
    manVelocity.y = 0;
}

```

รูปที่ 3.80 โค้ดในการเพิ่มลดค่ากระโดด และการลดเกินขีดจำกัด

### 3.7.8 การทำสิ่งกีดขวางให้ชนผู้เล่น

ในการทำสิ่งกีดขวางให้ชนกับผู้เล่นนั้น จะต้องกำหนดภาพและตำแหน่งของสิ่งกีดขวางก่อน โดยที่ตำแหน่งของสิ่งกีดขวางจะต้องทำเป็น ArrayList เพื่อให้รองรับคลาส Rectangle หลายตัว ซึ่งจะเป็นตัวแทนของอุปสรรคต่างๆ ดังรูปที่ 3.81 ซึ่งในส่วนนี้จะใส่ในคอนสตรัคเตอร์ หรือ create ของคลาส

```

void CreateLevel(int level){
    Texture coin = new Texture("game/coin.png");
    coinRegion = new TextureRegion(coin,0,0, coin.getWidth(),coin.getHeight());
    coinPosition = new ArrayList<Rectangle>();
    int x = 1800; //coin start point

    Texture rock = new Texture("game/rock.png");
    rockRegion = new TextureRegion(rock,0,0,rock.getWidth(),rock.getHeight());
    rockPosition = new ArrayList<Rectangle>();
    int r = 2000; //rock start point

    Texture plate = new Texture("game/Plat.png");
    pRegion = new TextureRegion(plate,0,0,plate.getWidth(),plate.getHeight());
    pPosition = new ArrayList<Rectangle>();

    Texture peel = new Texture("game/bananapeel.png");
    peelRegion = new TextureRegion(peel,0,0,peel.getWidth(),peel.getHeight());
    peelPosition = new ArrayList<Rectangle>();

    Texture drain = new Texture("game/drainpipe.png");
    drainRegion = new TextureRegion(drain,0,0,drain.getWidth(),drain.getHeight());
    drainPosition = new ArrayList<Rectangle>();

    Texture cone = new Texture("game/cone.png");
    coneRegion = new TextureRegion(cone,0,0,cone.getWidth(),cone.getHeight());
    conePosition = new ArrayList<Rectangle>();
}

```

รูปที่ 3.81 โค้ดการประกาศรูปภาพ Texture ของอุปสรรค

การวางตำแหน่งของอุปสรรคนั้น จะต้องทำการตั้งค่าสุ่ม เพื่อไม่ให้แต่ละด่านมีความเหมือนกันและซ้ำซาก โดยให้แต่ละอุปสรรคเกิดขึ้นได้ตามช่วงเวลาที่กำหนด และสุ่มตัวอุปสรรคที่จะเกิดขึ้นด้วยโอกาสความน่าจะเป็น ดังรูปที่ 3.82 และ 3.83

```

if(level ==0) //distance goal = 10000
{
    for(int start = 1200; start<distance_goal; start = start+500) {
        int chance = new Random().nextInt(40);
        if (chance < 15) {
            coinPosition.add(new Rectangle(x, DEFAULT_HEIGHT, coinRegion.getWidth(), coinRegion.getHeight())); //set movePosition
        } else if (chance > 15 && chance < 30) {
            rockPosition.add(new Rectangle(start, DEFAULT_HEIGHT, rock.getWidth(), rock.getHeight()));
        }
    }
}
else if(level ==1) {
    for(int start = 1000; start<distance_goal; start = start+500) {
        int chance = new Random().nextInt(70);
        if (chance < 15) {
            coinPosition.add(new Rectangle(start, DEFAULT_HEIGHT, coinRegion.getWidth(),
            coinRegion.getHeight())); //set movePosition
            if(new Random().nextBoolean()){
                coinPosition.add(new Rectangle(start+coin.getWidth(), DEFAULT_HEIGHT,
                coinRegion.getWidth(), coinRegion.getHeight())); //set movePosition
            }
        } else if (chance > 20 && chance < 60) {
            rockPosition.add(new Rectangle(start, DEFAULT_HEIGHT, rock.getWidth(), rock.getHeight()));
            if (new Random().nextBoolean()) {
                rockPosition.add(new Rectangle(start + rock.getWidth(), DEFAULT_HEIGHT, rock.getWidth(), rock.getHeight())); //set movePosition
            }
        } else if (chance > 60 && chance < 70) {
            peelPosition.add(new Rectangle(start, DEFAULT_HEIGHT, peel.getWidth() / 2, peel.getHeight() / 2));
        }
    }
}
}

```

รูปที่ 3.82 โค้ดกำหนดตำแหน่งและชนิดของอุปสรรค (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

else if(level==2){
    for(int start = 1000; start<=distance_goal; start = start+1500) {
        int chance = new Random().nextInt(70);
        if (chance < 15) {
            coinPosition.add(new Rectangle(start, DEFAULT_HEIGHT, coinRegion.getRegionWidth(), coinRegion.getRegionHeight())); //set movePosition
            if( new Random().nextBoolean()){
                coinPosition.add(new Rectangle(start-coin.getWidth(), DEFAULT_HEIGHT
                    , coinRegion.getRegionWidth(), coinRegion.getRegionHeight())); //set movePosition
            }
        } else if (chance > 20 && chance < 40) {
            rockPosition.add(new Rectangle(start, DEFAULT_HEIGHT, rock.getWidth(), rock.getHeight()));
            if (new Random().nextBoolean()) {
                rockPosition.add(new Rectangle(start + rock.getWidth(), DEFAULT_HEIGHT, rock.getWidth(), rock.getHeight())); //set movePosition
            }
        } else if (chance > 40 && chance < 70) {
            peelPosition.add(new Rectangle(start, DEFAULT_HEIGHT, peel.getWidth() / 2, peel.getHeight() / 2));
        }
    }
}
}

```

รูปที่ 3.83 โค้ดกำหนดตำแหน่งและชนิดของอุปสรรค (2)

หลังจากกำหนดตำแหน่งที่อยู่ของสิ่งกีดขวางแล้ว จะต้องทำการวาดของสิ่งกีดขวางแต่ละตัวด้วย เพื่อให้ผู้เล่นได้เห็นว่ามีของสิ่งกีดขวางนั้นอยู่ตรงไหน ดังรูปที่ 3.84

```

void drawObstacle(){
    for (Rectangle p : pPosition) {
        game.batch.draw(pRegion, p.x, p.y, p.width, p.height);
    }

    //coin
    for (Rectangle c : coinPosition) {
        game.batch.draw(coinRegion, c.x, c.y, c.width, c.height);
    }

    //rock
    for (Rectangle r : rockPosition) {
        game.batch.draw(rockRegion, r.x, r.y, r.width, r.height);
    }

    for (Rectangle pe : peelPosition) {
        game.batch.draw(peelRegion, pe.x, pe.y, pe.width, pe.height);
    }

    for (Rectangle dr : drainPosition) {
        game.batch.draw(drainRegion, dr.x, dr.y, dr.width, dr.height);
    }

    for (Rectangle co : conePosition) {
        game.batch.draw(coneRegion, co.x, co.y, co.width, co.height);
    } //draw movePosition must be the same as coin movePosition
}

```

รูปที่ 3.84 โค้ดในการวาดสิ่งกีดขวางทั้งหมด

จะทำเขียนให้ตัวละคร Player ชนกับสิ่งกีดขวางได้ โดยให้สิ่งกีดขวางแต่ละอันจะทำความเสียหายหรือหน้าที่ที่ต่างกันไป โดยโค้ดสามารถเขียนได้ดังรูปที่ 3.85 และ 3.86 โดยตอนที่ความเสียหายเกิดขึ้น จะให้ตัวละครหรือ Player มีความสามารถอมตะชั่วคราว โดยโค้ดสามารถเขียนได้ ดังรูปที่ 3.87

```

/** weird collision**/
private void eventCollision(){
//Overlap coin & Player
for (Rectangle c : coinPosition) {
if (c.overlaps(manPosition))
&& c.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())) < 90) {
//collision sound_coin.play();
game.coin += 1;
c.x = new Random().nextFloat();
}
}

//Overlap rock & Player
for (Rectangle r : rockPosition) {
if (r.overlaps(manPosition))
&& r.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())) < 110){
Gdx.app.log("rock", "center : " + r.getCenter(new Vector2()).toString());
Gdx.app.log("manPosition", "center : " + manPosition.getCenter(new Vector2()).toString());
Gdx.app.log("dst", "center : " + r.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())));
//collision_r.play();
setDamage(120);
//sEnergy -= 120;
//winStatement(false);
}
}

for (Rectangle p : pPosition) {
if (p.overlaps(manPosition) &&
p.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())) < p.getWidth()/2 ||
p.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())) < p.getHeight()/2){
//collision_r.play();
Gdx.app.log("plate", "center" + p.getCenter(new Vector2()).toString());
Gdx.app.log("manPosition", "center" + manPosition.getCenter(new Vector2()).toString());
Gdx.app.log("dst", "center : " + p.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())));
setDamage(150);
}
}
}

```

รูปที่ 3.85 โค้ดในการชนของตัวละครและสิ่งกีดขวาง (1)

```

//overlap banana peel
for (Rectangle pe : peelPosition) {
if (pe.overlaps(manPosition) &&
pe.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())) < 130){
//manPosition.getCenter(new Vector2()).dst(pe.getCenter(new Vector2())) < 60);
Gdx.app.log("peel", "center" + pe.getCenter(new Vector2()).toString());
Gdx.app.log("manPosition", "center" + manPosition.getCenter(new Vector2()).toString());
Gdx.app.log("dst", "center : " + pe.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())));
setDamage(40);
//sEnergy -= 40;
}
}

//overlap drain pipe
for (Rectangle dr : drainPosition) {
if (dr.overlaps(manPosition) && dr.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())) < 165) {
Gdx.app.log("drain", "center" + dr.getCenter(new Vector2()).toString());
Gdx.app.log("manPosition", "center" + manPosition.getCenter(new Vector2()).toString());
Gdx.app.log("dst", "center : " + dr.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())));
winStatement(false);
}
}

//overlap cone
for (Rectangle co : conePosition) {
if (co.overlaps(manPosition) && co.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())) < 150) {
Gdx.app.log("cone", "center" + co.getCenter(new Vector2()).toString());
Gdx.app.log("manPosition", "center" + manPosition.getCenter(new Vector2()).toString());
Gdx.app.log("dst", "center : " + co.getCenter(new Vector2()).dst(manPosition.getCenter(new Vector2())));
setDamage(200);
//sEnergy -= 200;
}
}
}
}

```

รูปที่ 3.86 โค้ดในการชนของตัวละครและสิ่งกีดขวาง (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

boolean damage;
private void setDamage(int damageSet){
    if(!damage){
        Gdx.app.log("damage",Integer.toString(damageSet));
        SpriteSetToAnimation(new Texture(Gdx.files.internal("game/small_player_pack3x4_2_damage.png")));
        sEnergy -= damageSet;
        damage = true;
    }
    else{
        Timer.schedule(new Timer.Task() {
            @Override
            public void run() {
                SpriteSetToAnimation(new Texture(Gdx.files.internal("game/small_player_pack3x4_2.png")));
                damage = false;
            }
        },3);
    }
}
}

```

รูปที่ 3.87 เมธอดเมื่อตัวละครได้ชนกับสิ่งกีดขวาง

### 3.7.9 การแสดงค่าพลังและการคิดคำนวณ

เมื่อค่าพลังงานชีวิตของผู้เล่นหมดจะแพ้เกมนั้นในทันที ซึ่งค่านั้นจะดึงมาจากค่าแคลอรีพลังงานเผาผลาญของร่างกายที่พิถีพิถันส่งมาให้ โดยการแสดงภาพค่าพลังงานที่ลดลงจะสามารถเขียนได้ ดังรูปที่ 3.88 ซึ่งจะต้องเขียนในระหว่าง batch.begin() กับ batch.end() และโค้ดวิธีการคิดตัวเลขค่าพลังงาน ดังรูปที่ 3.89 โดยที่ทั้งสองส่วนนี้เขียนในตัว render()

```

move = manPosition.x - 180;
//energy
game.font.getData().setScale(6);
game.font.draw(game.batch, (int)sEnergy+"/"+energyMax,10+ move,
(game.aGdxgraphicGetHeight - (game.aGdxgraphicGetHeight / 6)));
game.batch.draw(energyImg, 10 + move
,game.aGdxgraphicGetHeight - (game.aGdxgraphicGetHeight / 6),
(sEnergy/energyMax) * (game.aGdxgraphicGetWidth - 200),40);

```

รูปที่ 3.88 การแสดงค่าพลังงานที่ลดต่อเนื่อง

```

if(sEnergy > 0) {
    sEnergy -= 0.1;
}
else{
    winStatement(false);
}

```

รูปที่ 3.89 โค้ดคำนวณค่าลดพลังงาน

### 3.7.10 การคำนวณคะแนน และระบบตอนจบ

การคิดคะแนนเกมส์ จะคิดตามจำนวนก้าวของตัวละครตามที่กำหนด และให้หยุดคิดคำนวณคะแนนและค่าพลังงานเมื่อจบฉาก พร้อมทั้งหยุดการเคลื่อนไหวอื่นอย่างแถบพลังงานกับการเคลื่อนไหวของคลาสกลิ้ง ดังรูปที่ 3.90 และแสดงหน้าต่างแสดงคะแนนที่ได้และทั้งหมด ดังรูปที่ 3.91

```

if(!win) {
    move = manPosition.x - 180;
    //score calculated
    thisGamScore += (manPosition.x/1000);
    if(thisGamScore >= 100){
        scoreIngame ++;
        thisGamScore = 0;
    }
}

if(sEnergy > 0) {
    sEnergy -= 0.1;
}
else{
    winstatement(false);
}

if (Gdx.graphics.getWidth() < distance_goal) {
    camera.translate(manvelocity.x * Gdx.graphics.getDeltaTime(), 0);
}
else{
    if(Gdx.input.isTouched()) {
        game.callback();
        // game.townNO--s;
        game.setScreen(new LevelChoose(game));
    }
}

boolean once = false;
void winstatement(boolean win)
{
    //while(loop) {
        if (win) {
            if(!once) {
                updateData();
                scoreboard();
                once = true;
            }
            this.win = true;
        }
        else {
            resetGame();
            configureCamera();
            game.setScreen(new Town(game, level));
        }
    }
}
//}
}

```

รูปที่ 3.90 โค้ดแสดงเงื่อนไขเมื่อแพ้และชนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void scoreboard(){
    skin = new Skin(Gdx.files.internal("skinpack/skinpack.json"));
    skin.getFont("default-font").getData().setScale(3);
    score = new Window("Score", skin);
    score.setWidth(game.aGdxgraphicGetWidth/3);
    score.setHeight(game.aGdxgraphicGetHeight/3);
    score.setPosition((game.aGdxgraphicGetWidth/2)-(game.aGdxgraphicGetWidth/(3*2))
        ,(game.aGdxgraphicGetHeight/2)-(game.aGdxgraphicGetHeight/(3*2)));
    score.add("Total Score: "+game.score);
    score.row();
    score.add("Score: "+scoreIngame);
    stage.addActor(score);
}

```

รูปที่ 3.91 เมธอดแสดงหน้าจอบันทึกคะแนนของเกมทีเล่น

### 3.7.11 การส่งข้อมูลการนอนหลับไปให้ฟิตบิต

การส่งข้อมูลการนอนหลับไปให้ฟิตบิต จะทำการใช้เมธอด https Request ในการ POST ตามข้อมูลที่ฟิตบิตกำหนดไว้ โดยเว็บ https ของฟิตบิตที่จะร้องขอไป ต้อง POST ข้อมูล วันที่ เวลาและระยะในการนอนหลับ ดังรูปที่ 3.92

```

private void sendDataSleep(long duration){
    Calendar c = Calendar.getInstance();
    SimpleDateFormat d = new SimpleDateFormat("yyyy-MM-dd");
    String date = d.format(c.getTime());
    d = new SimpleDateFormat("HH:mm");
    String time = d.format(c.getTime());

    Gdx.app.log(CLASS_NAME, "Date : "+date);
    Gdx.app.log(CLASS_NAME, "Time : "+time);
    String urlParameterE = null;
    try {
        urlParameterE = URLEncoder.encode("startTime", "UTF-8") + "=" + URLEncoder.encode(time, "UTF-8");
        urlParameterE += "&" + URLEncoder.encode("duration", "UTF-8") + "=" + URLEncoder.encode("22000000", "UTF-8");
        urlParameterE += "&" + URLEncoder.encode("date", "UTF-8") + "=" + URLEncoder.encode(date, "UTF-8");
    }
    catch (UnsupportedEncodingException e){
        Gdx.app.log(CLASS_NAME, "unsupport E");
    }

    game.requestHttps("https://api.fitbit.com/1/user/" + game.user_id + "/sleep.json"
        , "POST", "Authorization", game.token_type + " " + game.access_token
        , "Content-Type", "application/x-www-form-urlencoded", urlParameterE);
    game.disconnect();
}

```

รูปที่ 3.92 โค้ดการส่งข้อมูลการนอนหลับด้วย https request

### 3.7.12 การดึงข้อมูลการนอนหลับและการแสดงด้วยกราฟ

การดูข้อมูลการนอนหลับ การทำงานจะให้ข้อมูลใน 1 สัปดาห์นับจากวันปัจจุบัน โดยต้องดึงข้อมูลการนอนหลับจากเมธอด HttpsRequest ดังรูปที่ 3.93

```

private void getSleepDataMultiplyTime(int times) {
    Gdx.app.log(tag, "loopgetSleepData");
    int day = 0;
    int loops = times;

    preferences = Gdx.app.getPreferences(SleepPreference.PREFERENCE_NAME);

    while (day < loops) {
        getSleepData(day);
        day++;
    }

    Gdx.app.log(tag, "finished get sleepdata");
}

void getSleepData(int day){
    Gdx.app.log(tag, "getSleepData");
    Calendar c = Calendar.getInstance();

    c.add(Calendar.DATE, -day);
    SimpleDateFormat d = new SimpleDateFormat("yyyy-MM-dd");
    String date = d.format(c.getTime());

    InputStream getSleepDataOnce = requestHttps("https://api.fitbit.com/1/user/" + user_id + "/sleep/date/" + date + ".json"
        , "GET", "Authorization", "Basic " + getAutherEncoded, "Authorization", token_type + " " + access_token, null);
}

```

รูปที่ 3.93 การดึงข้อมูลการนอนหลับโดยเรียกใช้เมธอด HttpRequest 7 วัน

จากนั้นจะทำการดึงข้อมูลที่ต้องการจากข้อมูลในรูปแบบ JSON และบันทึกลงไปที่ Preference โดยถ้าข้อมูลว่างจะบันทึกข้อมูลคัมมีลงไปเพื่อไม่ให้ตัวข้อมูลว่าง ดังโค้ดที่แสดงในรูปที่ 3.94 และโค้ดแสดงการดึงออกมาใช้ในการเขียนเป็นกราฟแท่งเป็นดังรูปที่ 3.95



```

SleepData sleepData = new SleepData();
try {
    JsonValue v = new JsonReader().parse(getSleepDataOnce);
    if(v.isNull()){
        Gdx.app.log(tag, "v is null" );
    }
    Gdx.app.log(tag, "A1:" + v.toString());
    Gdx.app.log(tag, "A2:" + v.child().toString());
    JsonValue vv = v.get("sleep");
    Gdx.app.log(tag, "A3:" + vv.toString());

    Gdx.app.log(tag, "B1:" + vv.child().getString("dateOfSleep").toString());
    String dateOfSleep = vv.child().getString("dateOfSleep").toString();

    Gdx.app.log(tag, "E1:" + vv.child().get("minuteData").child().getString("dateTime"));
    String time = vv.child().get("minuteData").child().getString("dateTime");

    Gdx.app.log(tag, "B2:" + vv.child().getString("duration").toString());
    int duration = vv.child().getInt("duration");

    preferences.putString(SleepPreference.DATE,dateOfSleep);
    //preferences.putString(SleepPreference.TIME,time)

    sleepData.setDate(dateOfSleep);
    sleepData.setTime(time);
    sleepData.setDuration(duration);

    Json json = new Json();
    json.toJson(sleepData);

    Gdx.app.log("First", json.toJson(sleepData));
    preferences.putString(Integer.toString(day), json.prettyPrint(sleepData));
    preferences.flush();

    Gdx.app.log("FirstP", json.prettyPrint(sleepData));
} catch (NullPointerException e) {
    Gdx.app.log(tag,"Null pointer Exception");

    Gdx.app.log(tag, "null");

    sleepData.setDate(date);
    sleepData.setTime("1");
    sleepData.setDuration(1);

    json json = new Json();
    json.toJson(sleepData);

    Gdx.app.log("First", json.toJson(sleepData));
    Gdx.app.log("FirstP", json.prettyPrint(sleepData));
    preferences.putString(Integer.toString(day), json.prettyPrint(sleepData));
    preferences.flush();
}
finally {
    disconnect();
}

```

รูปที่ 3.94 การดึงข้อมูลการนอนหลับจาก JSON และบันทึกลง Preference

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกฏนำไปใช้

```

Preferences shared =.gdx.app.getPreferences(SleepPreference.PREFERENCE_NAME)
for(int day=0;day<=7;day++){
    JsonValue jsonValue = new JsonReader()
        .parse(shared.getString(Integer.toString(day)).toString());
    date = jsonValue.getString('day')
    time = jsonValue.getString('time')
    int duration = jsonValue.getInt('duration');
    duration = ((duration/1000)/60); //millisec to minute
    if(duration>=100) {
        hours = duration/60;
    } else {
        hours = 1;
    }
    sleep += hours
    skin = new skin(Gdx.files.internal("skinpack/skinpack.json"));
    Window window = new window("sleepview", skin);
    window.setPosition(3 * game.aGdxgraphicGetWidth/30
        ,116 * (game.aGdxgraphicGetHeight/1000)*(7-day));
    window.setWidth((float)hours*game.aGdxgraphicGetWidth/30);
    window.setHeight(game.aGdxgraphicGetHeight/8);
    window.top().left().padLeft(30).padTop(50).padRight(30).padBottom(50).align(3);
    stage.addActor(window);}
}

```

รูปที่ 3.95 การดึงข้อมูลออกจาก Preference และดึง JSON มาตั้งกราฟ

### 3.7.12 การดึงข้อมูลคะแนนจาก Web Server โดยใช้ JSON Format

JSON (JavaScript Object Notation) เป็นฟอร์แมตที่ใช้ในการรับส่งข้อมูลสำหรับ JavaScript ข้อมูลที่เข้ารหัสเป็น JSON โดยแบ่งการพิจารณาได้ดังนี้

- ส่วนการทำหน้าที่ Encode JSON จาก Web Server (JSON Parser)
- ส่วนการสร้างและใช้งาน JSON Object

#### ส่วนการทำหน้าที่ Encode JSON จาก Web Server

ข้อมูลเว็บเซิร์ฟเวอร์ที่ส่งมาในลักษณะ JSON นั้น ได้รับการแปลง (Encode) ให้อยู่ในรูปแบบของ JSON เรียบร้อยแล้ว ดังนั้นเมื่อแอนดรอยด์อ่านข้อมูลจะต้องทำการแปลงกลับ (Decode)

#### ส่วนการสร้างและใช้งาน JSON

การดึงข้อมูลของเซิร์ฟเวอร์ที่ทำการดึงข้อมูล JSON จาก Web Server จะใช้เมธอด HttpRequest โดยวิธีการ POST ข้อมูลรหัสผู้ใช้ที่ดึงมาจากพิตบิต ดังรูปที่ 3.96 และทำการดึงค่าข้อมูลออกจากรูปแบบ JSON มาสร้างเป็นตารางด้วยคลาส Window ดังรูปที่ 3.97 และ 3.98

```

InputStream getScore(){
String tag = "ViewScore";
Preferences preferences = Gdx.app.getPreferences(ProfilePreference.PREFERENCE_NAME);

UserData userData = new UserData();
userData.setUserID(game.user_id);
userData.setUsername(preferences.getString(ProfilePreference.displayName));

Json json = new Json();
String postData = json.toJson(userData);

String urlParameter = "userID="+game.user_id+"&userName="+preferences.getString(ProfilePreference.displayName);

InputStream getScore = game.requestHttp("http://161.246.60.238/htdocs/showScore.php", "POST",
null,null,null,null,urlParameter);

return getScore;
}

```

รูปที่ 3.96 การดึงคะแนนจาก Webserver โดยใช้เมธอด Https Request

```

InputStream scoreJson = getScore();
JsonValue jsonValue = new JsonReader().parse(scoreJson);
jsonValue = jsonValue.child();
Gdx.app.log(tag, "11; "+jsonValue.toString());
String idUser = null, nameUser = null, score = null, section = null, town = null;
for (int i = 1; i<=4; i++) {
try{
nameUser = jsonValue.getString("nameUser");
Gdx.app.log(tag, "json : "+jsonValue.getString("nameUser"));
score = jsonValue.getString("score");
Gdx.app.log(tag, "json : "+jsonValue.getString("score"));
section = jsonValue.getString("section");
Gdx.app.log(tag, "json : "+jsonValue.getString("town"));
town = jsonValue.getString("town");

if(!jsonValue.child().next().isNull()){
jsonValue = jsonValue.next();
Gdx.app.log(tag, "jsonChild : "+jsonValue.next());
}
}catch (NullPointerException e){
Gdx.app.log(tag, "Null error", e.getCause());
}

windows = new Skin(Gdx.files.internal("windowJson/window.json"));
windows.getFont("default-font").getData().setScale(3);

if(i=4) {
window = new Window(" ", windows); //other
}
else{
window = new Window(" ", windows); //user
}
window.setPosition(Gdx.graphics.getWidth()/8, Gdx.graphics.getHeight()/8*i);
window.setWidth(4*game.sGdxgraphicGetWidth/5);
window.setHeight(200);

window.top().left().padLeft(30).padTop(50).padRight(30).padBottom(50);

window.add(nameUser).expandX().left();
window.add(score).expandX().left();
window.add(section).expandX().left();
window.add(town).expandX().left();

```

รูปที่ 3.97 โค้ดการสร้างหน้าต่างแสดงคะแนนของ ViewScore (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

window.add(town).expandX().left();
window.setTouchable(Touchable.disabled);
window.setMovable(false);
//window.row();

stage.addActor(window);

}

```

รูปที่ 3.98 โค้ดการสร้างหน้าจอแสดงคะแนนของ ViewScore (2)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลลัพธ์ของระบบ

ผลลัพธ์ของแอปพลิเคชัน FitGaming 2 คือ แอปพลิเคชันที่ผู้ใช้สามารถนำข้อมูลจาก Fitbit มาเล่นเกมได้ แอปพลิเคชันแบ่งการทำงานเป็น 3 ส่วนหลัก คือ

- 1) การเรียกดูข้อมูลจาก Fitbit และการเล่นเกม โดยจะนำข้อมูลของการออกกำลังกาย คือ การเผาผลาญของร่างกายมาเป็นพลังงานในการเล่นเกมน
- 2) เกม จะเป็นเกมแนว RPG ผู้เล่นจะต้องตะลุยด่านในแต่ละจังหวัดตามแผนที่ที่เกมมีมาให้ โดยผู้เล่นจะต้องวิ่งเก็บเหรียญและกระโดดข้ามสิ่งกีดขวางเพื่อผ่านด่านให้ได้ หากผ่านด่านได้จะสามารถเข้าเล่นด่านถัดไปได้ หากผ่านด่านครบทุกด่านในแต่ละภูมิภาคจะสามารถเข้าเล่นภูมิภาคถัดไปได้ แต่ถ้าผู้เล่นไม่ผ่านด่านนั้น ๆ ผู้เล่นจะต้องเริ่มเล่นใหม่ใน Town ของด่านล่าสุด
- 3) ในแต่ละวันต้องออกกำลังกายอย่างน้อยเท่ากับที่แอปพลิเคชันกำหนด จึงจะสามารถเล่นเกมในด่านถัดไปได้ โดยแอปพลิเคชันจะกำหนดระยะเวลาในการออกกำลังกายนานขึ้นในแต่ละด่าน ผู้ใช้สามารถดูผลคะแนนของตนเองเทียบกับเพื่อนบนฟิตบิตที่เล่นเกมนี้ด้วยกันได้

ในการพัฒนาแอปพลิเคชัน FitGaming 2 ฟังก์ชันการทำงานหลักของแอปพลิเคชันมีดังต่อไปนี้

- ฟังก์ชันยืนยันตัวตนผ่านเว็บไซต์ฟิตบิต มีหน้าจอดังรูปที่ 4.1
- Level Choose แสดงด่านในเกม ในรูปแบบของภูมิภาคในประเทศไทย มีหน้าจอดังรูปที่ 4.2
- Town แสดงหน้าจอของแต่ละจังหวัด มีหน้าจอดังรูปที่ 4.3
- Play Game ฟังก์ชันการเล่นเกม มีหน้าจอดังรูปที่ 4.4
- Sleep Log บันทึกการนอนและแสดงประวัติการนอนภายใน 1 สัปดาห์ ในรูปแบบของกราฟแท่ง มีหน้าจอดังรูปที่ 4.5 และ 4.6
- Profile แสดงข้อมูลของผู้ใช้แอปพลิเคชัน มีหน้าจอดังรูปที่ 4.7
- View Score ตารางแสดงคะแนนของผู้ใช้และเปรียบเทียบกับคะแนนของเพื่อน ๆ มีหน้าจอดังรูปที่ 4.8

แอปพลิเคชัน FitGaming 2 เป็นแอปพลิเคชันที่มีความสามารถดังต่อไปนี้

- ผู้ใช้สามารถนำข้อมูลการออกกำลังกายจากสายรัดข้อมือฟิตบิตมาใช้ในการเล่นเกม โดยทำการเชื่อมสายรัดข้อมือฟิตบิตกับสมาร์ทโฟนผ่านทางบลูทูธ
- ผู้ใช้จะมีเพื่อนในเกมโดยแอปพลิเคชันจะนำเพื่อนของผู้ใช้มาจากรายชื่อเพื่อนในแอปพลิเคชันฟิตบิต

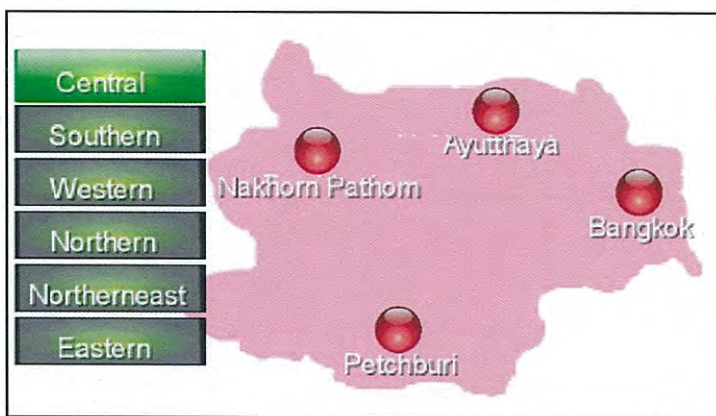
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ผู้ใช้สามารถดูข้อมูลโปรไฟล์ของตนเองและของเพื่อนได้
- ในการเล่นเกมจะมีด่านทั้ง 6 ภาค ได้แก่ Central, Southern, Western, Northern, Northerneast, Eastern
- แต่ละภาคจะมีด่านให้เลือกทั้ง 4 จังหวัด
- เมื่อเข้าไปในหน้า Town ของแต่ละจังหวัด จะพบกับภาพพื้นหลังของสถานที่สำคัญในจังหวัดนั้น ๆ และมีเมนู Sleep Log และ Play Game
- เมื่อผู้ใช้เลือก Play Game โดยผู้เล่นจะต้องวิ่งเก็บเหรียญและกระโดดข้ามสิ่งกีดขวางเพื่อผ่านด่านให้ได้ หากผ่านด่านได้สำเร็จ จะไปสู่จังหวัดต่อไป แต่ถ้าไม่ผ่าน จะกลับมาเริ่มใหม่ที่หน้า Town ของจังหวัดล่าสุด
- ถ้าผู้ใช้ผ่านครบทุกจังหวัดในภูมิภาคนั้น ๆ ผู้ใช้จะได้เข้าสู่ภูมิภาคต่อไป
- ผู้ใช้สามารถดูคะแนนของเพื่อนคนอื่น ๆ ได้



รูปที่ 4.1 หน้าจอยืนยันตัวตนผ่านเว็บไซต์ Fitbit ก่อนเข้าเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 หน้าจอเลือกภาคและด้าน (จังหวัด)

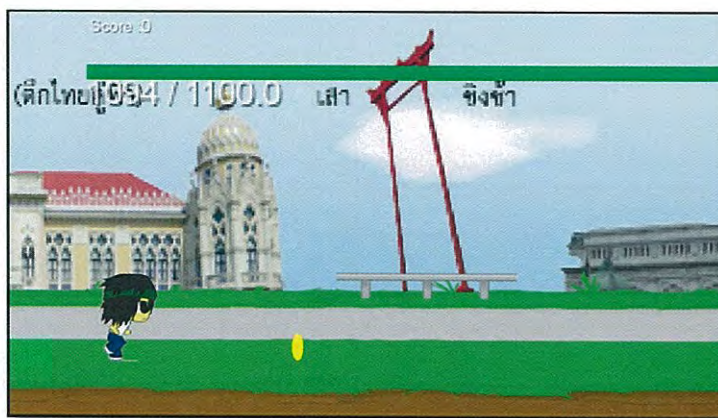
รูปที่ 4.2 แสดงหน้าจอเลือกภาคและด้าน (จังหวัด) ซึ่งประกอบไปด้วยปุ่มเลือกภูมิภาคต่างๆ ในประเทศไทย และด้านในแต่ละภาคที่มีภาคละ 4 จังหวัด



รูปที่ 4.3 หน้าจอ Town

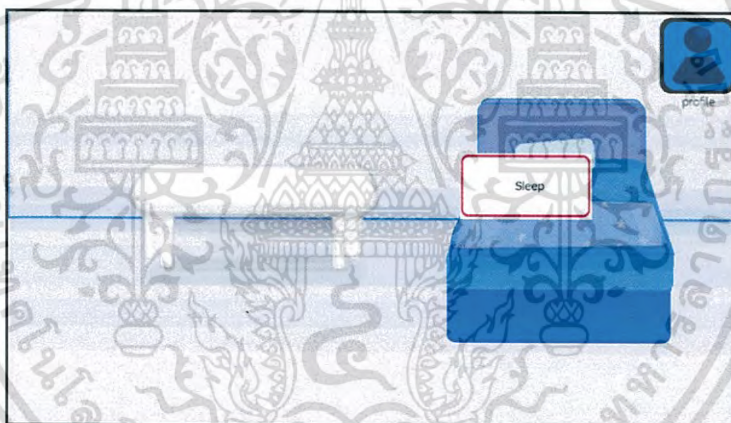
รูปที่ 4.3 แสดงหน้าจอ Town โดยที่หน้าจอ Town ในแต่ละจังหวัดนั้นจะมีภาพพื้นหลังที่เป็นสถานที่สำคัญในจังหวัดนั้น ๆ รวมถึงปุ่ม Play Game, Sleep Log และ Profile

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



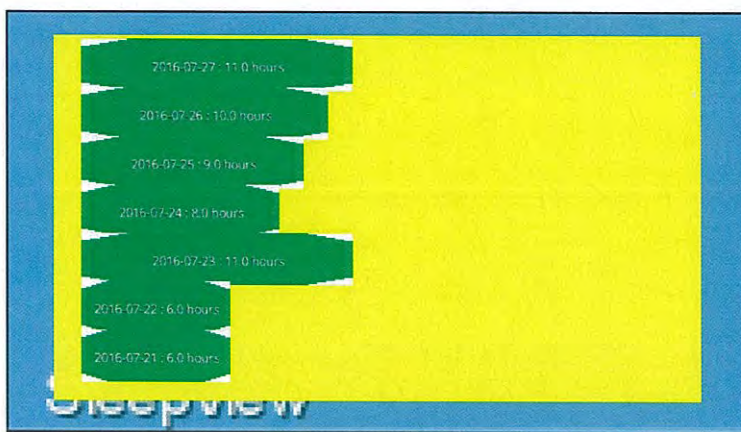
รูปที่ 4.4 หน้าจอ Play Game

รูปที่ 4.4 แสดงหน้าจอ Play Game ซึ่งประกอบไปด้วยตัวละครในเกมที่กำลังวิ่งอยู่ โดยมีเหรียญและอุปสรรคต่าง ๆ ค่อย ๆ ปรากฏขึ้นมาตามทาง ภาพพื้นหลังเป็นสถานที่สำคัญในจังหวัดนั้น ๆ โดยมีจังหวัดละ 5 สถานที่ มีแถบระยะทางคอยบอกว่าวิ่งไปได้ระยะทางเท่าใด มี Score ที่บอกค่าคะแนน และมีตัวเลขบอกค่าพลังงาน



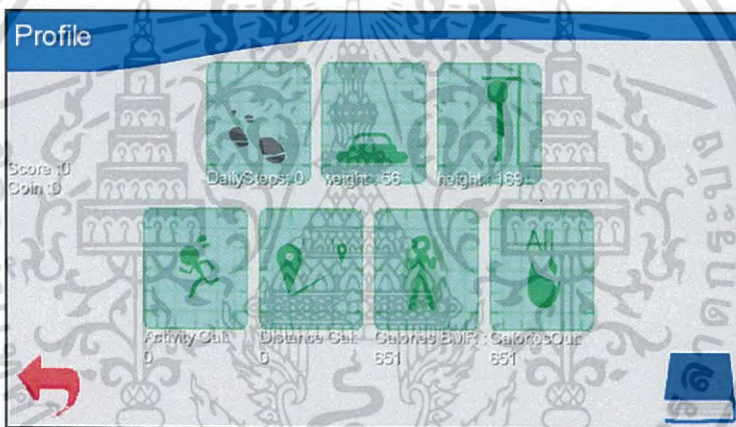
รูปที่ 4.5 หน้าจอ Sleep Room

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 หน้าจอ Sleep View

รูปที่ 4.5 แสดงหน้าจอ Sleep Room ที่มีปุ่ม Sleep อยู่บนรูปเตียงนอน เมื่อกดปุ่ม Sleep ก็จะไปพบกับหน้าจอ Sleep View ซึ่งแสดงข้อมูลการนอนหลับใน 1 สัปดาห์ ในรูปของกราฟแท่ง ดังรูปที่ 4.6



รูปที่ 4.7 หน้าจอ Profile

รูปที่ 4.7 แสดงหน้าจอ Profile โดยจะแสดงประวัติส่วนตัวต่าง ๆ ของผู้เล่น เช่น ก้าวเดินในแต่ละวัน น้ำหนัก ส่วนสูง การเผาผลาญพลังงานจากกิจกรรม การเผาผลาญพลังงานจากก้าวเดิน BMR การเผาผลาญพลังงานทั้งหมด ตามลำดับ

| Name  | Score | Section | Province |
|-------|-------|---------|----------|
| sense | 10    | central | bangkok  |
| aa    | 0     | central | bankok   |
| 111   | 0     | central | bankok   |
| gotty | 3     | central | bangkok  |

รูปที่ 4.8 หน้าจอ View Score

รูปที่ 4.8 แสดงหน้าจอหน้าจอ View Score แสดงคะแนนการเล่นของผู้เล่นและเพื่อน ๆ ที่เล่นเกม FitGaming 2 ด้วยกัน โดยจะสรุปผลเป็นอันดับตารางเรียงตามคะแนน และบอกตำแหน่งของผู้เล่นแต่ละคนว่าอยู่ภาคไหน ด้าน (จังหวัด) อะไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

# บทสรุปและข้อเสนอแนะ

### 5.1 บทสรุป

แอปพลิเคชันพิตเกมมิง 2 มีรูปแบบเป็นเกมแนว RPG โดยให้ผู้เล่นเข้าเล่นในแต่ละด่าน (จังหวัด) โดยการเก็บเหรียญและกระโดดข้ามสิ่งกีดขวางจนกว่าจะเข้าเส้นชัย จึงจะถือว่าผ่านด่าน เมื่อผู้เล่นผ่านในแต่ละด่านได้จะสามารถเข้าไปเล่นในด่านถัดไป ถ้าผู้เล่นสามารถผ่านด่านทุกด่านในภาคนั้น ๆ ผู้เล่นจะสามารถเข้าไปสู่ภาคถัดไปได้

ในแต่ละด่านจะมีภาพพื้นหลังเป็นสถานที่สำคัญที่มีอยู่จริงของจังหวัดนั้น ๆ

นอกจากนี้ ผู้เล่นยังสามารถดูข้อมูลการนอนหลับของตนเองในรูปแบบของกราฟแท่ง และสามารถดูคะแนนของตนเองและเพื่อน ๆ ตามลำดับได้อีกด้วย

### 5.2 ปัญหาและข้อจำกัดการพัฒนาแอปพลิเคชัน

ในการพัฒนาแอปพลิเคชันพิตเกมมิงบนพีดีบีดี บนระบบปฏิบัติการแอนดรอยด์ มีข้อจำกัด ดังนี้

- โปรแกรมเกมที่เชื่อมต่อกับอุปกรณ์สายรัดข้อมือพิตบีดี ไม่สามารถเชื่อมต่อกับอุปกรณ์ได้โดยตรง โดยจะต้องติดต่อผ่านทาง API ของ Fitbit.com ซึ่งข้อมูลที่ได้จะขึ้นอยู่กับสิ่งที่ API มีให้ จึงไม่สามารถนำค่ามาแสดงผลได้ชัดเจนมากนัก
- เมื่อเข้าสู่ส่วน core ของ LibGDX แล้ว คลาส Android บางตัวจะไม่สามารถใช้ได้ เช่น JSONObject Encoding ของ Java และคลาสที่สร้างในส่วนของ Android
- คลาส LibGDX บางตัวต้องใช้โค้ดเฉพาะ เพื่อในการใช้งานคลาสนั้น ๆ เช่น คลาส Button Window
- การวาดรูปตัวสไปรท์ซีทีใช้เวลาาน

### 5.3 ข้อเสนอแนะ

- การดึงข้อมูลแบบที่ใช้มีขีดจำกัดในการดึงข้อมูล โดยเฉพาะตัวโปรไฟล์ของพิตบีดี ควรจะใช้ Subscription ในการดึงข้อมูลแทนซึ่งจะสามารถดึงข้อมูลปัจจุบันได้อย่างต่อเนื่อง เมื่อเวลาที่ข้อมูลมีการเปลี่ยนแปลง
- ในแต่ละเมืองของเกมควรมีเรื่องราวการอธิบายของเมือง ๆ นั้น เช่นการให้ตัวละครในเมืองให้พูดถึงประวัติศาสตร์ที่เกี่ยวข้อง คำพูดให้กำลังใจ หรือให้ NPC พุดขึ้นมา เพื่อให้ผู้ใช้ได้มีความรู้สึกเข้าถึงกับตัวเกมมากยิ่งขึ้น
- เกมควรมีตัวช่วยให้ผู้เล่นข้ามอุปสรรคได้ง่ายขึ้น อย่าง การใช้สิ่งของเพื่อหลบหลีกสิ่งกีดขวาง เพิ่มพลังงานอย่างฉับพลัน เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แอปพลิเคชันมีระบบเพื่อน ดังนั้นควรมีการติดต่อสื่อสารกันระหว่างผู้เล่นและเพื่อน ๆ ผ่านแอปพลิเคชัน (แชท)
- ควรใช้อุปกรณ์สายรัดข้อมือฟิตบิตรุ่นใหม่ มีฟังก์ชันการทำงานมากกว่ารุ่นเก่า และแอปพลิเคชันสามารถออกแบบการเล่นเกมที่หลากหลายกว่า
- ถ้าจะใช้ LibGDX ในการพัฒนาเกม ควรจะใช้เฉพาะตัวเกมเท่านั้น เพราะไม่เหมาะในการทำ Layout



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เอกสารอ้างอิง

- [1] เว็บไซต์สนุกกูรู. (2557). ประเภทของเกม. สืบค้นเมื่อ 12 พฤศจิกายน 2558, จาก <http://guru.sanook.com/27010/>
- [2] เว็บไซต์ไทยวิกิพีเดีย. (2557). วิดีโอเกม. สืบค้นเมื่อ 12 พฤศจิกายน 2558, จาก <https://th.wikipedia.org/วิดีโอเกม>
- [3] orangepeego. (2555). มารู้จักเกมประเภทต่าง ๆ กันเถอะ. สืบค้นเมื่อ 12 พฤศจิกายน 2558, จาก <http://orangepeego.exteen.com/20120602/entry>
- [4] Admin. (2558). Wearable Device. สืบค้นเมื่อ 12 พฤศจิกายน 2558, จาก <http://www.dailytech.in.th/รีวิว-wearable-device-ยอดนิยม/>
- [5] fitbit. (2558). Fitbit Surge. สืบค้นเมื่อ 12 พฤศจิกายน 2558, จาก [www.fitbit.com/th/surge](http://www.fitbit.com/th/surge)
- [6] FatChicken Studios. (2557). FitRPG. สืบค้นเมื่อ 12 พฤศจิกายน 2558, จาก <https://play.google.com/store/apps/details?id=com.fatchickenstudios.fitrpg&hl=th>
- [7] fitbit. (2558). การเข้าถึง Fitbit API. สืบค้นเมื่อ 12 พฤศจิกายน 2558, จาก <https://dev.fitbit.com/docs/oauth2/>
- [8] memo8. (2558). การวัดดัชนีมวลร่างกายและอัตราการเผาผลาญพลังงาน. สืบค้นเมื่อ 13 พฤศจิกายน 2558, จาก <http://kcal.memo8.com/calorie-fat-burn/>
- [9] itgenius. (2557). MySQL คืออะไร. สืบค้นเมื่อ 13 พฤศจิกายน 2558, จาก [http://www.itgenius.co.th/article/\(MySQL\)%20คืออะไร.html](http://www.itgenius.co.th/article/(MySQL)%20คืออะไร.html)
- [10] ไวกพ ตู๋น้อย. (2558). เว็บเซิร์ฟเวอร์. สืบค้นเมื่อ 13 พฤศจิกายน 2558, จาก <http://www.thaiall.com/omni/indexo.html>
- [11] Mindphp. (2557). Apache คืออะไร อปาเซ คือ เว็บเซิร์ฟเวอร์. สืบค้นเมื่อ 13 พฤศจิกายน 2558, จาก <http://www.mindphp.com/คู่มือ/73-คืออะไร/2265-apache-คืออะไร.html>
- [12] น.อ.รุ่งธรรม บัวแดง. (2556). พีเอชพี (PHP) คืออะไร. สืบค้นเมื่อ 13 พฤศจิกายน 2558, จาก <http://www.dstd.mi.th/board/index.php?topic=875.0>
- [13] ninetechno. (2557). XAMPP คืออะไร. สืบค้นเมื่อ 13 พฤศจิกายน 2558, จาก <http://www.ninetechno.com/a/website/873-xampp.html>
- [14] allhandweb. (2556). ทำความรู้จักกับ JSON คืออะไร. สืบค้นเมื่อ 13 พฤศจิกายน 2558, จาก <http://www.allhandweb.com/?page=Blog.ShowBlogDetail&blogID=13>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เอกสารอ้างอิง (ต่อ)

- [15] นพรุจ ศิวะพรชัย, วีระวัฒน์ สุวัฒน์พิเศษ. (2557). **พิตเกมมิ่ง : เกมเพื่อสุขภาพบนพิตบิต.** ปัญหาพิเศษ ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2557
- [16] เว็บไซต์ไทยวิกิพีเดีย. (2558). **เสาชิงช้า.** สืบค้นเมื่อ 5 มิถุนายน 2559, จาก <https://th.wikipedia.org/wiki/เสาชิงช้า>
- [17] เว็บไซต์ไทยวิกิพีเดีย. (2559). **เอเชียทีค เดอะ ริเวอร์ฟรอนท์.** สืบค้นเมื่อ 5 มิถุนายน 2559, จาก [https://th.wikipedia.org/wiki/เอเชียทีค\\_เดอะ\\_ริเวอร์ฟรอนท์](https://th.wikipedia.org/wiki/เอเชียทีค_เดอะ_ริเวอร์ฟรอนท์)
- [18] เว็บไซต์ไทยวิกิพีเดีย. (2559). **ทำเนียบรัฐบาลไทย.** สืบค้นเมื่อ 5 มิถุนายน 2559, จาก <https://th.wikipedia.org/wiki/ทำเนียบรัฐบาลไทย>
- [19] เว็บไซต์ไทยวิกิพีเดีย. (2559). **พระที่นั่งอนันตสมาคม.** สืบค้นเมื่อ 5 มิถุนายน 2559, จาก <https://th.wikipedia.org/wiki/พระที่นั่งอนันตสมาคม>
- [20] เว็บไซต์ไทยวิกิพีเดีย. (2559). **พระบรมมหาราชวัง.** สืบค้นเมื่อ 5 มิถุนายน 2559, จาก <https://th.wikipedia.org/wiki/พระบรมมหาราชวัง>
- [21] เว็บไซต์ไทยวิกิพีเดีย. (2559). **พระราชวังบางปะอิน.** สืบค้นเมื่อ 6 มิถุนายน 2559, จาก <https://th.wikipedia.org/wiki/พระราชวังบางปะอิน>
- [22] เว็บไซต์ไทยวิกิพีเดีย. (2558). **วัดใหญ่ชัยมงคล.** สืบค้นเมื่อ 6 มิถุนายน 2559, จาก <https://th.wikipedia.org/wiki/วัดใหญ่ชัยมงคล>
- [23] เว็บไซต์ไทยวิกิพีเดีย. (2559). **วัดพระศรีสรรเพชญ์.** สืบค้นเมื่อ 6 มิถุนายน 2559, จาก <https://th.wikipedia.org/wiki/วัดพระศรีสรรเพชญ์>
- [24] Ayutthaya.org. (ไม่ปรากฏปีที่พิมพ์). **ศูนย์ศิลปาชีพบางไทร.** สืบค้นเมื่อ 6 มิถุนายน 2559, จาก [http://www.ayutthaya.org/attractions/ayutthaya\\_Bangsai02.html](http://www.ayutthaya.org/attractions/ayutthaya_Bangsai02.html)
- [25] นายอาชีพ. (ไม่ปรากฏปีที่พิมพ์). **ศูนย์ศิลปาชีพบางไทร แหล่งเรียนรู้และสร้างอาชีพที่ยั่งยืน.** สืบค้นเมื่อ 6 มิถุนายน 2559, จาก <https://archeep-center.blogspot.com/2013/04/blog-post.html>
- [26] ท่าอากาศยานดอนเมือง. (2558). **หมู่บ้านญี่ปุ่น (พระนครศรีอยุธยา).** สืบค้นเมื่อ 6 มิถุนายน 2559, จาก <http://donmueangairportthai.com/th/popular-destinations/1727/japanese-village-phra-nakhon-si-ayutthaya>
- [27] www.paiduaykan.com. (ไม่ปรากฏปีที่พิมพ์). **เขาวัง อุทยานประวัติศาสตร์พระนครศรีอยุธยา.** สืบค้นเมื่อ 7 มิถุนายน 2559, จาก <http://www.paiduaykan.com/province/central/phetchaburi/khaowang.html>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เอกสารอ้างอิง (ต่อ)

- [28] กระปุกดอทคอม. (ไม่ปรากฏปีที่พิมพ์). บุกเดี่ยว แบกเป้เที่ยว คาเมล รีพับลิก แลนด์ มาร์กแห่งใหม่ในชะอำ. สืบค้นเมื่อ 7 มิถุนายน 2559, จาก <http://travel.kapook.com/view100252.html>
- [29] Santorini Park Cha-Am. (2558). Santorini Park Cha-Am : AMUSED SHOPPING EXPERIENCE. สืบค้นเมื่อ 7 มิถุนายน 2559, จาก <http://www.santoriniparkchaam.com/index.php?op=page-index&id=1>
- [30] www.paiduaykan.com. (ไม่ปรากฏปีที่พิมพ์). พระราชนิเวศน์มฤคทายวัน. สืบค้นเมื่อ 7 มิถุนายน 2559, จาก [http://www.paiduaykan.com/76\\_province/central/phetchaburi/marukathaywan.html](http://www.paiduaykan.com/76_province/central/phetchaburi/marukathaywan.html)
- [31] การท่องเที่ยวแห่งประเทศไทย. (ไม่ปรากฏปีที่พิมพ์). พระรามราชนิเวศน์. สืบค้นเมื่อ 7 มิถุนายน 2559, จาก [thai.tourismthailand.org/สถานที่ท่องเที่ยว/พระรามราชนิเวศน์—977](http://thai.tourismthailand.org/สถานที่ท่องเที่ยว/พระรามราชนิเวศน์—977)
- [32] Mario Zechner. (2556). LibGDX. สืบค้นเมื่อ 30 มิถุนายน 2559, จาก <https://LibGDX.badlogicgames.com/features.html>
- [33] Zemith Sirimark. (2557). [Libgdx] เริ่มต้นกับ Game Engine. สืบค้นเมื่อ 30 มิถุนายน 2559, Libgdx <http://zemithlab.blogspot.com/2013/08/libgdx-libgdx.html>



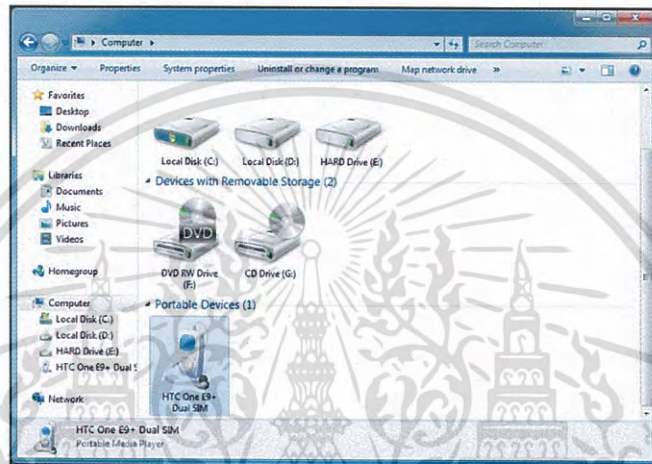
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ก

### การติดตั้งโปรแกรมในการพัฒนาแอปพลิเคชัน

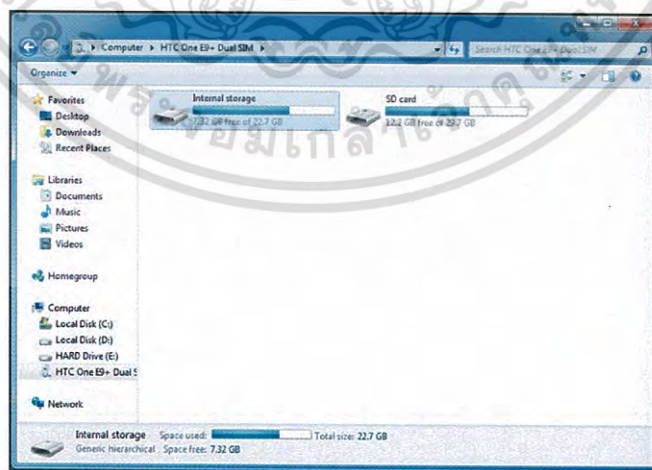
#### ก.1 การติดตั้งไฟล์ apk

1. ทำการเชื่อมต่ออุปกรณ์แอนดรอยด์เข้ากับเครื่องคอมพิวเตอร์ผ่าน USB แล้วทำการคลิก ไอคอนเครื่องที่เชื่อมต่อ



รูปที่ ก.1 หน้าจอไฟล์ที่เชื่อมต่อกับคอมพิวเตอร์

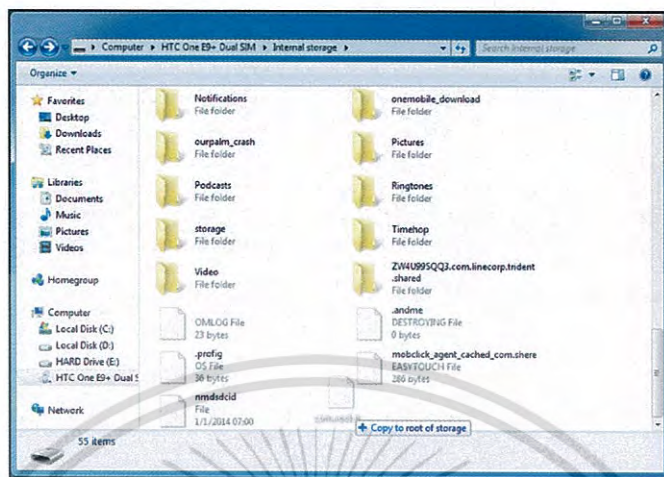
2. คลิกเพื่อเปิด Internal Storage ที่คัดลอกไฟล์เก็บไว้



รูปที่ ก.2 หน้าจอ Internal Storage จากอุปกรณ์ที่เชื่อมต่อกับคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. ลากไฟล์ Android apk ไปใส่ในเครื่องอุปกรณ์แอนดรอยด์



รูปที่ ก.3 หน้าจอการใส่ไฟล์ apk ลงเครื่อง

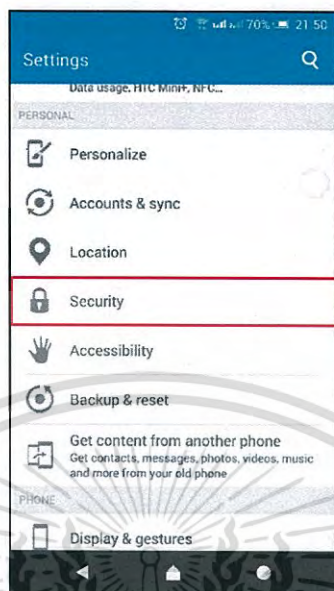
### 4. เปิดการใช้งาน Unknown Sources โดยไปที่เมนู Settings



รูปที่ ก.4 หน้าจอการเปิดใช้งาน Unknown Sources (1)

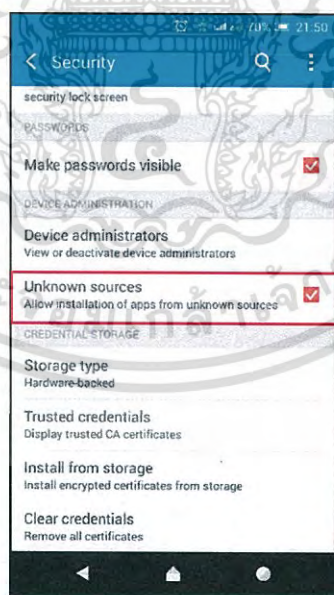
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5. ไปที่หัวข้อ Security



รูปที่ ก.5 หน้าจอการเปิดใช้งาน Unknown Sources (2)

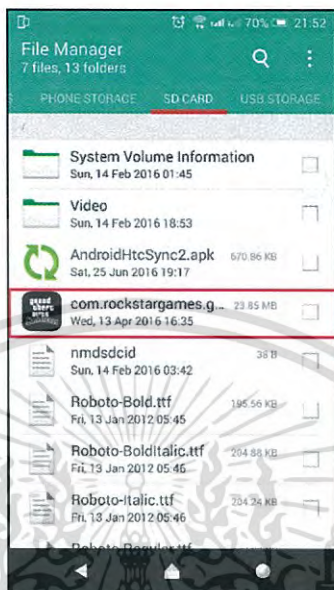
## 6. จะพบคำสั่ง Unknown Sources กดเพื่อเปิดใช้งาน



รูปที่ ก.6 หน้าจอการเปิดใช้งาน Unknown Sources (3)

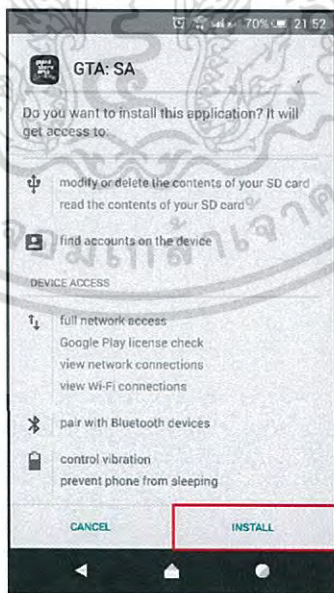
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เข้าไปยังพื้นที่ที่ทำการเก็บไฟล์ Android apk ในตอนแรก โดยใช้แอปพลิเคชันประเภท File Manager เพื่อค้นหาไฟล์ apk



รูปที่ ก.7 หน้าจอการค้นหาไฟล์ apk เพื่อเตรียมทำการติดตั้ง

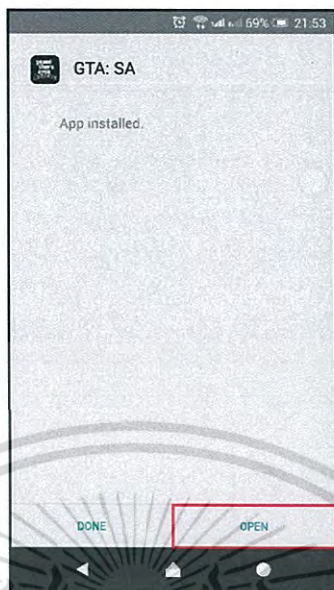
8. คลิกที่ไฟล์ apk จะพบหน้าจอแจ้งรายละเอียดของ Permission ให้คลิกที่ Install แล้วรอจนกว่าแอปพลิเคชันจะถูกติดตั้งเสร็จ



รูปที่ ก.8 หน้าจอการติดตั้งไฟล์ apk

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

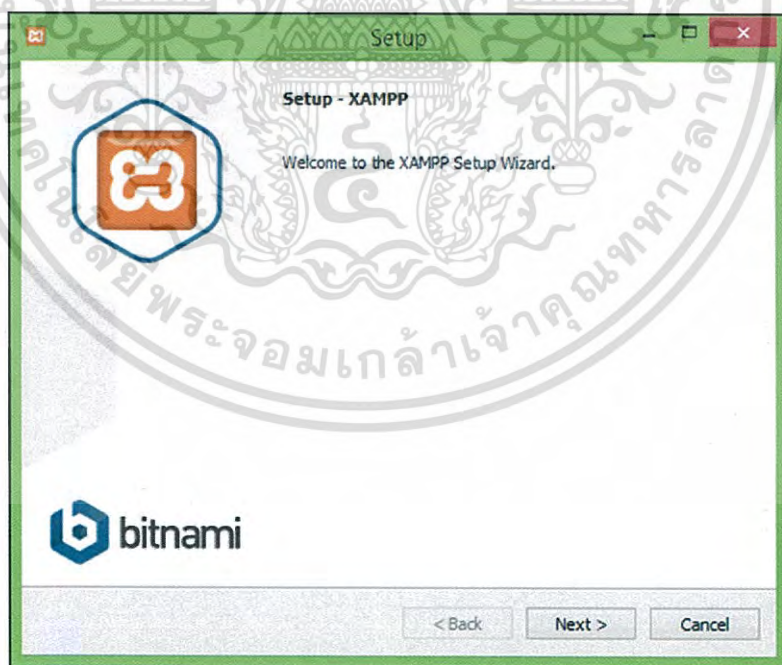
9. เมื่อแอปพลิเคชันถูกติดตั้งเสร็จแล้ว คลิก Open เพื่อเรียกการทำงานของแอปพลิเคชัน



รูปที่ ก.9 หน้าจอก่อนเปิดแอปพลิเคชัน หลังการติดตั้งเสร็จสิ้น

## ก.2 การติดตั้ง XAMPP

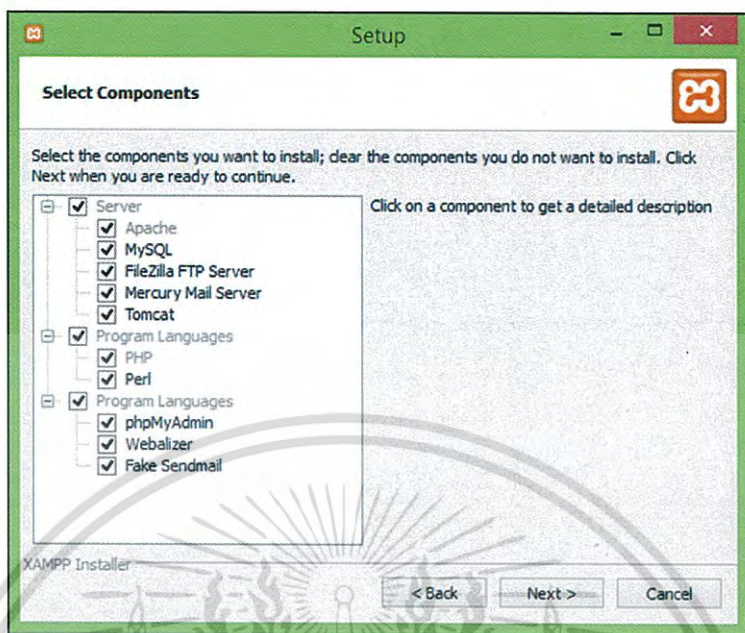
1. เมื่อทำการติดตั้ง จะพบหน้าจอสำหรับติดตั้งโปรแกรม ให้ทำการคลิกที่ปุ่ม Next



รูปที่ ก.10 หน้าจอเริ่มต้นการติดตั้ง XAMPP

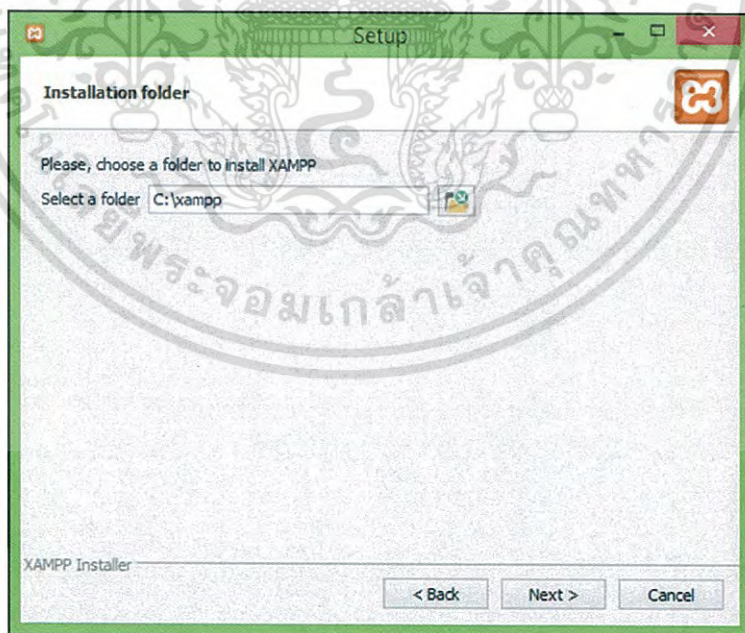
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เลือก Component ทุกตัว แล้วคลิกที่ Next



รูปที่ ก.11 หน้าจอการติดตั้ง Component ต่าง ๆ

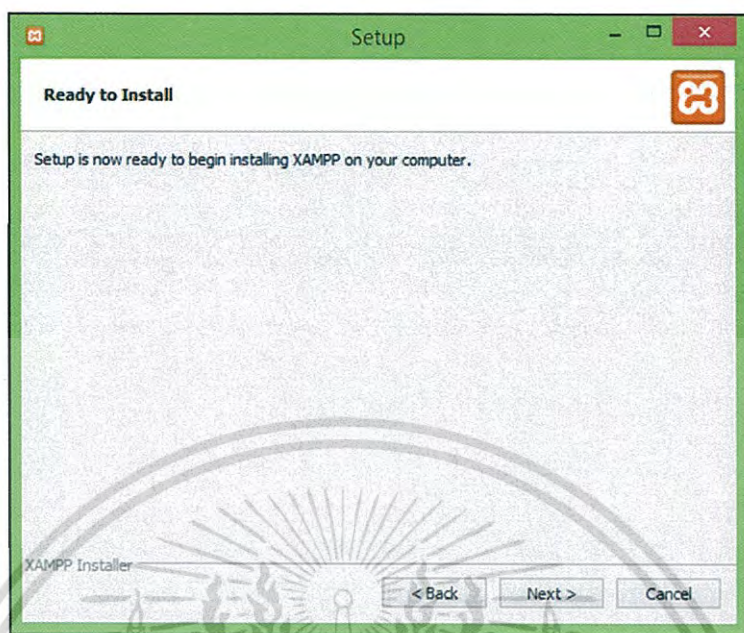
3. กำหนด Path ของโปรแกรมที่จะติดตั้ง แล้วคลิกที่ Next



รูปที่ ก.12 หน้าจอกำหนด Path ของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. คลิกที่ปุ่ม Next เพื่อทำการติดตั้งโปรแกรม XAMPP



รูปที่ ก.13 หน้าจอเตรียมพร้อมที่จะติดตั้งโปรแกรม XAMPP

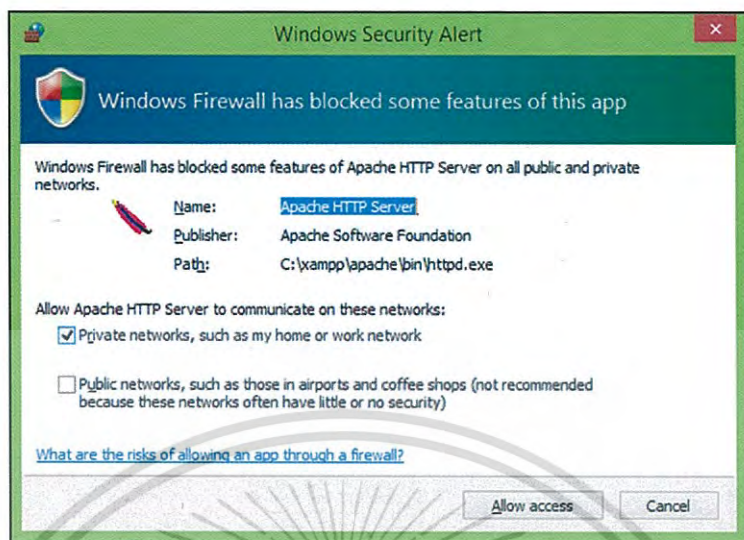
5. รอจนกว่าการติดตั้งจะเสร็จสิ้น



รูปที่ ก.14 หน้าจอของโปรแกรม XAMPP ที่กำลังถูกติดตั้ง

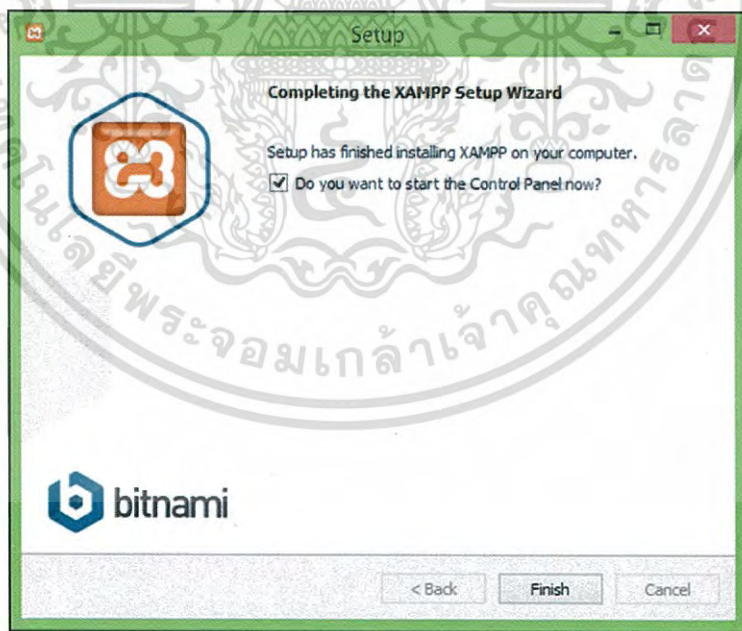
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. เมื่อมีการแจ้งเตือนจาก Firewall ให้คลิก อนุญาต หรือ Allow access (ถ้ามี)



รูปที่ ก.15 หน้าจอการแจ้งเตือนของ Firewall

7. เมื่อโปรแกรม XAMPP ถูกติดตั้งเสร็จเรียบร้อยแล้ว คลิกที่ Finish เพื่อปิดหน้าต่างการติดตั้งโปรแกรม แล้วเปิด XAMPP



รูปที่ ก.16 หน้าจอการติดตั้งโปรแกรม XAMPP เสร็จสิ้น

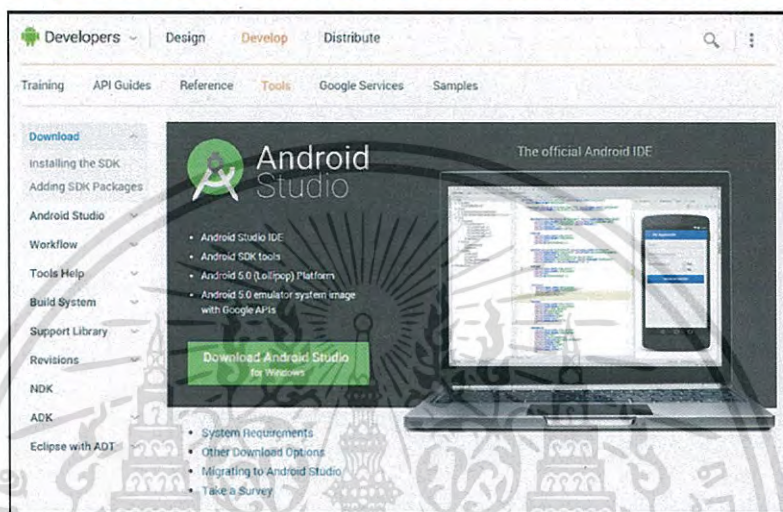
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ก.3 การติดตั้ง Android Studio

ในการติดตั้งและใช้งานโปรแกรม Android Studio จะต้องทำการติดตั้งโปรแกรม Java เสียก่อน โดยสามารถดาวน์โหลดได้ที่ <https://java.com/en/download/>

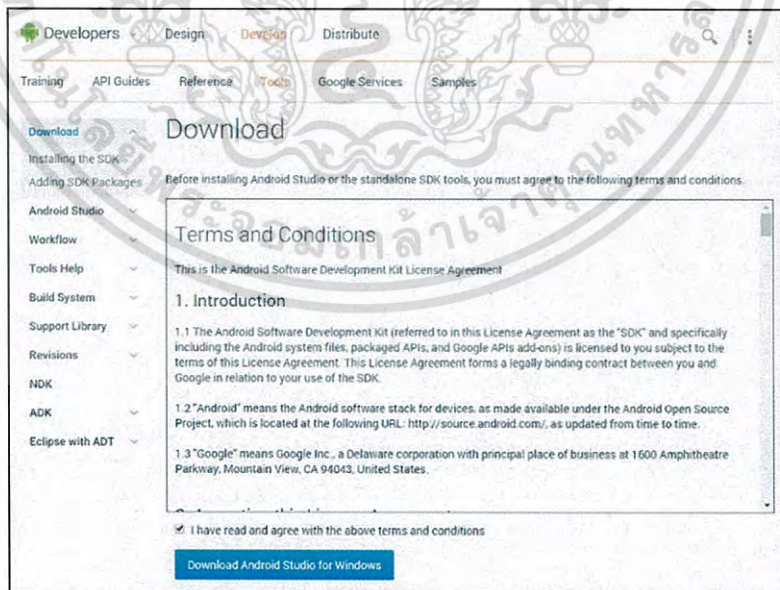
1. ดาวน์โหลด Android Studio ได้จาก

<https://developer.android.com/studio/index.html#downloads>



รูปที่ ก.17 หน้าจอดาวน์โหลด Android Studio

2. คลิกเพื่อยอมรับข้อตกลงการใช้งานเสียก่อน จึงจะกดปุ่มดาวน์โหลดได้



รูปที่ ก.18 หน้าจอยอมรับข้อตกลงการใช้งาน

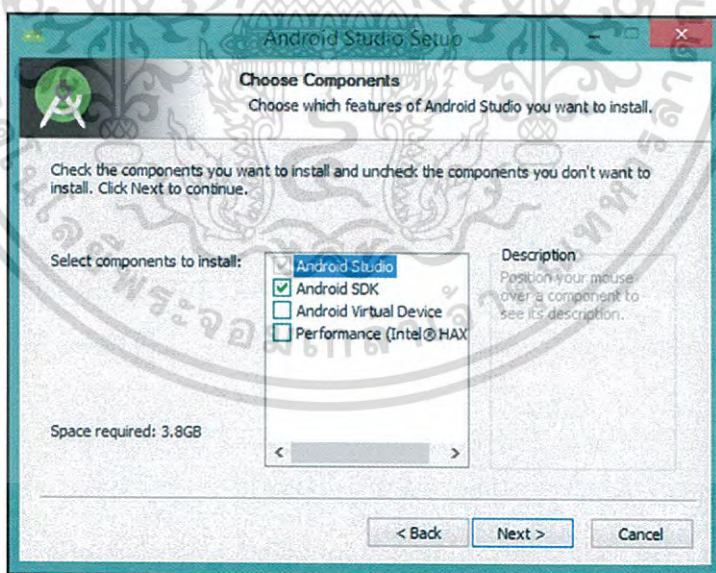
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อทำการดาวน์โหลดเสร็จเรียบร้อยแล้วให้เปิดไฟล์ขึ้นมา จะพบกับหน้า Welcome ให้คลิกที่ Next



รูปที่ ก.19 หน้าจอติดตั้งโปรแกรม Android Studio

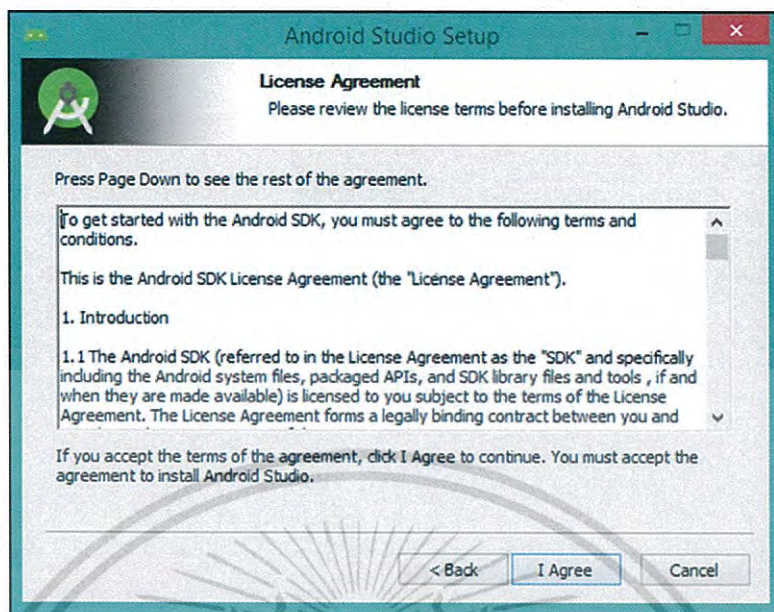
4. คลิกที่ Android SDK ส่วน Android Virtual Device (AVD) กับ Performance (Intel® HAXM) นั้นเป็นเพียงแค่ Emulator และตัวช่วยที่ทำให้ Emulator ทำงานได้ไวขึ้น ผู้ใช้จะเลือกติดตั้งหรือไม่ก็ได้ แล้วคลิกที่ Next



รูปที่ ก.20 หน้าจอการติดตั้ง Component ต่าง ๆ

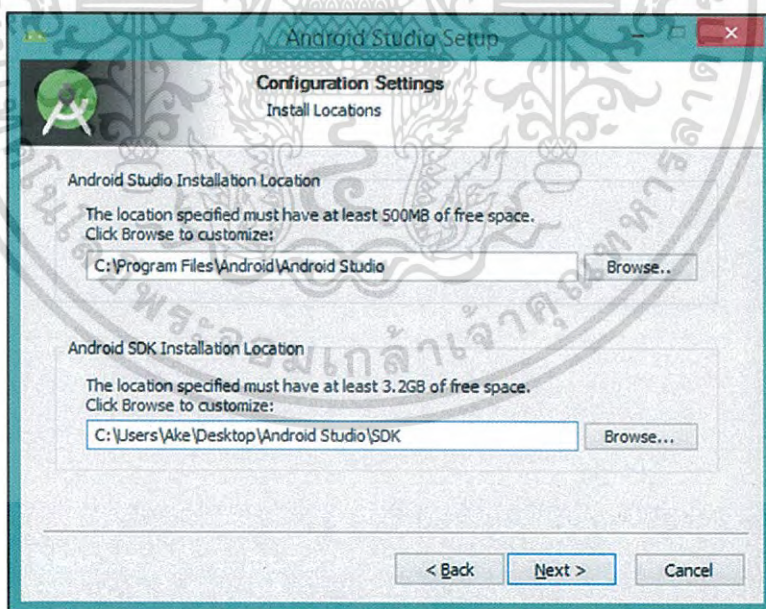
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. จะพบข้อตกลงการใช้งานของ Android Studio ให้คลิกที่ I Agree



รูปที่ ก.21 หน้าจอข้อตกลงการใช้งาน

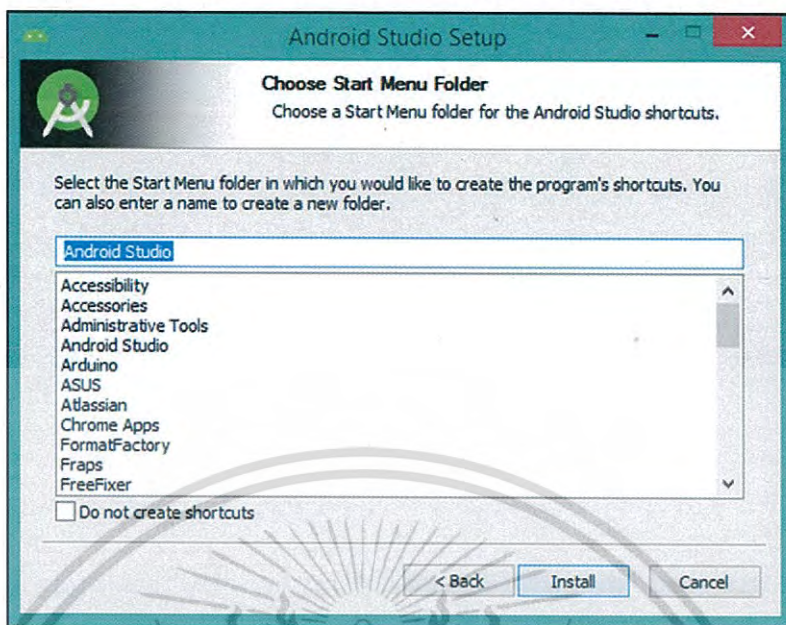
6. กำหนด Path ที่ต้องการติดตั้ง แล้วคลิกที่ Next



รูปที่ ก.22 หน้าจอกำหนด Path ของโปรแกรม

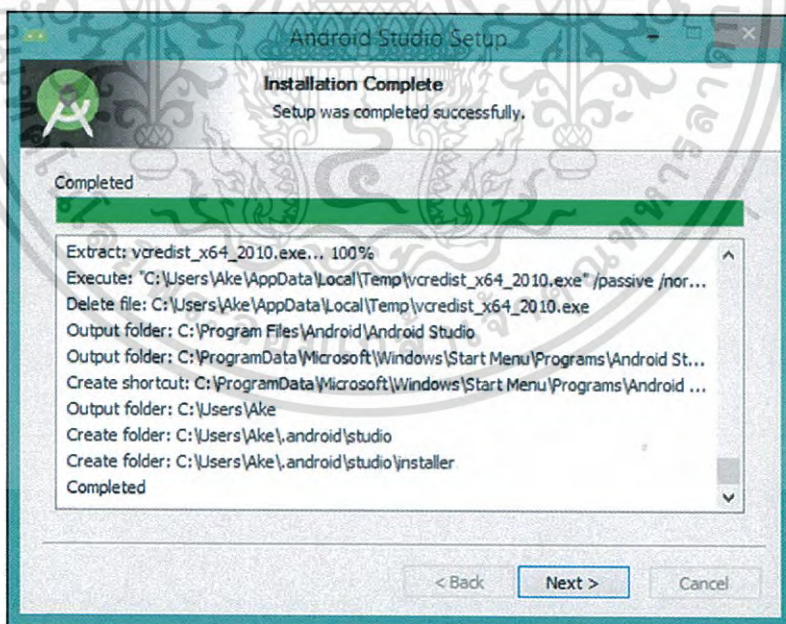
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. กำหนด Start Menu Folder ซึ่งในขั้นตอนนี้ให้คลิกที่ Install เพื่อเริ่มทำการติดตั้ง



รูปที่ ก.23 หน้าจอกำหนด Start Menu Folder และเตรียมพร้อมการติดตั้ง

8. รอให้โปรแกรมทำการติดตั้งจนเสร็จ เมื่อติดตั้งเสร็จแล้วให้คลิกที่ Next



รูปที่ ก.24 หน้าจอโปรแกรม Android Studio ที่กำลังถูกติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

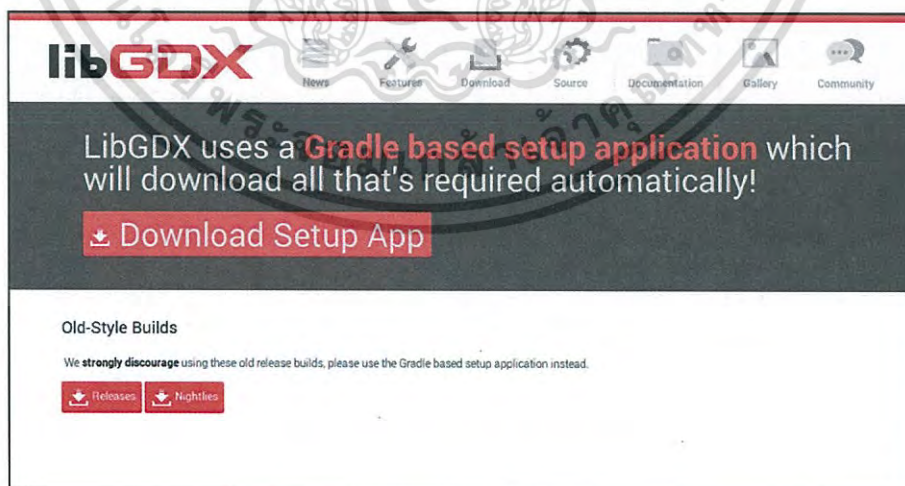
9. เมื่อโปรแกรม Android Studio ถูกติดตั้งเสร็จเรียบร้อยแล้ว คลิกที่ Finish เพื่อปิดหน้าต่างการติดตั้งโปรแกรม แล้วเปิด Android Studio



รูปที่ ก.25 หน้าจอการติดตั้งโปรแกรม Android Studio เสร็จสิ้น

#### ก.4 การติดตั้ง LibGDX

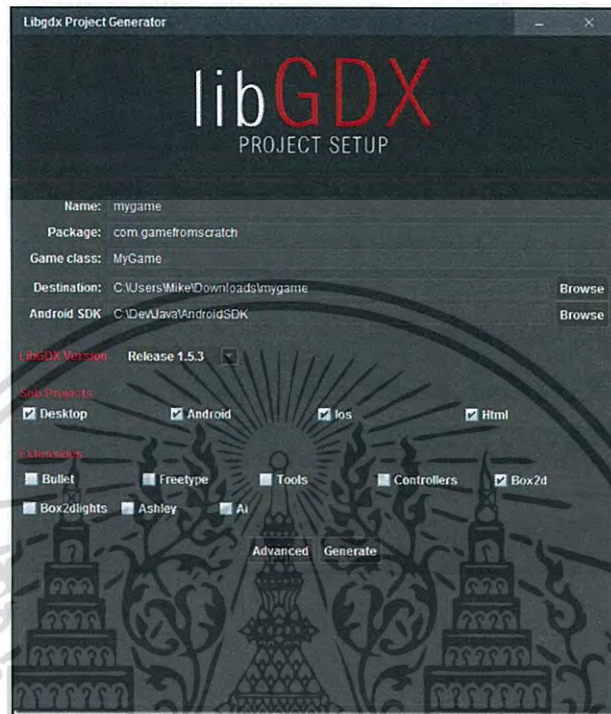
1. ขั้นตอนการติดตั้ง LibGDX ให้เข้าไปยังเว็บไซต์ของ LibGDX ที่ <https://libgdx.badlogicgames.com/> แล้วเลือกดาวน์โหลด



รูปที่ ก.26 หน้าจอดาวน์โหลด LibGDX

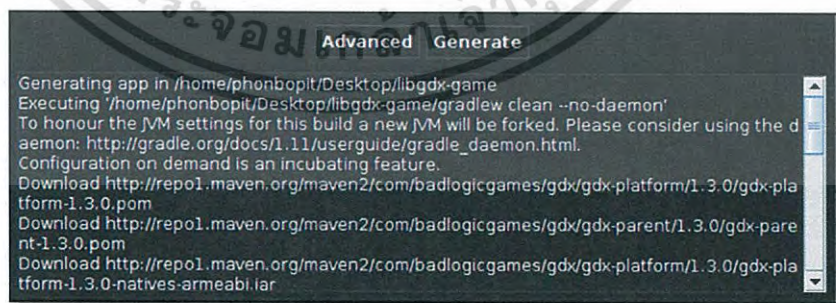
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ทำการดาวน์โหลดตัว GDX-Setup.jar เมื่อดาวน์โหลดเสร็จแล้ว ให้เปิดโปรแกรม แล้วทำการ Setup ตัวโปรแกรมตั้งรูปด้านล่าง หากผู้ใช้ต้องการเขียนโปรแกรมในส่วนของ Android ให้คลิกเลือกเฉพาะ Android (ส่วนนี้คือการตั้งค่า เพื่อให้ LibGDX ทำการ Generate Project)



รูปที่ ก.27 หน้าจอโปรแกรม LibGDX

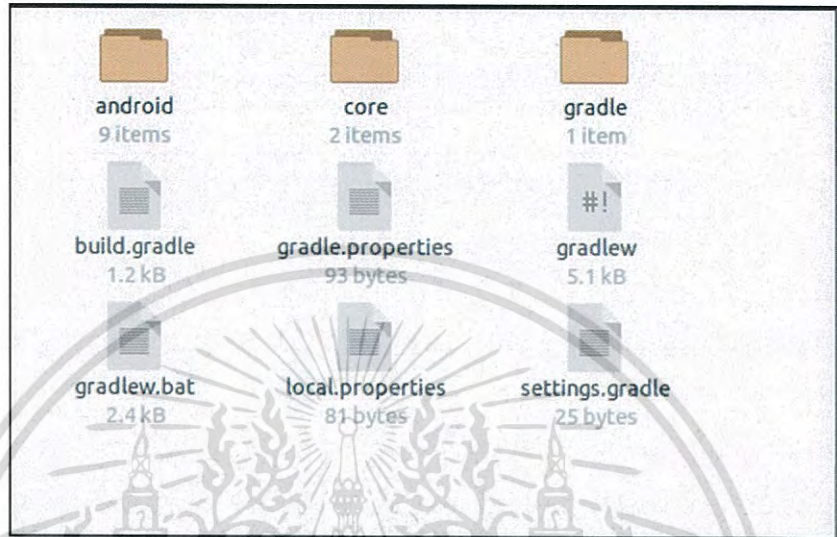
3. โปรแกรม LibGDX-Setup จะทำการ Config และดาวน์โหลดไฟล์ต่าง ๆ ที่จำเป็นในการเขียนโปรแกรมมาให้ทั้งหมด รอจนกว่าการดาวน์โหลดจะเสร็จสิ้น



รูปที่ ก.28 หน้าจอการดาวน์โหลดไฟล์ในโปรแกรม LibGDX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อการดาวน์โหลดไฟล์เสร็จสิ้นแล้ว จะมีข้อความแจ้งเตือน เมื่อสำรวจในโฟลเดอร์ที่เลือก Destination ไว้ตั้งแต่ต้น จะพบโฟลเดอร์ต่าง ๆ ดังรูป ผู้ใช้สามารถนำไฟล์เหล่านี้ไปทำการ Import เข้าสู่โปรแกรม Android Studio ได้



รูปที่ ก.29 หน้าจอของไฟล์ต่าง ๆ ที่ผ่านการดาวน์โหลดจากโปรแกรม LibGDX

## ภาคผนวก ข

### คู่มือการใช้งานแอปพลิเคชัน

#### ข.1 ขั้นตอนการใช้งานแอปพลิเคชัน

##### ข.1.1 ลงทะเบียน

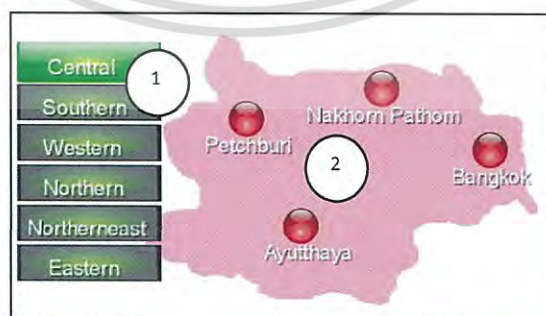
เมื่อผู้ใช้งานมีการเข้าใช้งานครั้งแรก ผู้ใช้ต้องลงทะเบียนกับฟิตบิตเพื่อยืนยันตัวตนและร้องขอข้อมูลจากฟิตบิตเซิร์ฟเวอร์ ซึ่งรูปที่ ข.1 จะแสดงหน้าจอ Log in ของ Fitbit ก่อนเข้าเกม



รูปที่ ข.1 หน้าจอ Log in ของ Fitbit ก่อนเข้าเกม

##### ข.1.2 หน้าจอเลือกภาคและด่าน (จังหวัด)

เมื่อเข้าระบบมาแล้วจะพบกับหน้าจอเลือกภาคและด่าน แบ่งเป็นปุ่มเลือกภาค (1) และปุ่มเลือกด่าน (2) ตามชื่อภาคและจังหวัดบางส่วนที่มีอยู่จริงในประเทศไทย โดยในครั้งแรกที่เข้าเล่นเกม จะเล่นได้เฉพาะภาคกลาง และเล่นได้เฉพาะด่านกรุงเทพมหานครเท่านั้น หากผู้เล่นชนะด่านใดด่านหนึ่งได้ จะสามารถเล่นด่านอื่น ๆ ต่อไปได้ และถ้าชนะทุกด่านในแต่ละภาคก็สามารถเข้าไปเล่นภาคต่อไปได้ ดังรูปที่ ข.2



รูปที่ ข.2 หน้าจอเลือกภาคและด่าน (จังหวัด)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ข.1.3 หน้าจอภายใน Town

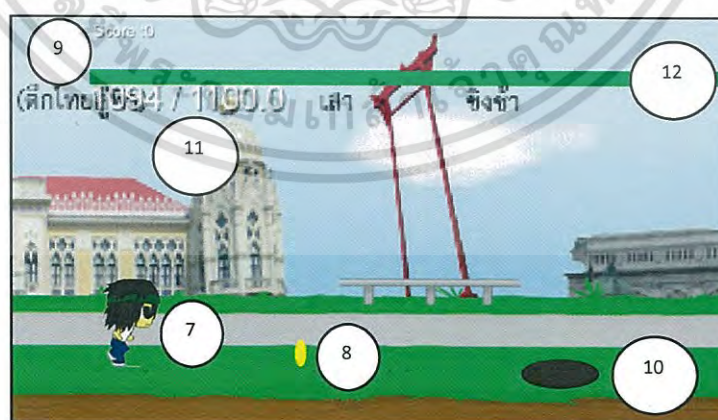
เมื่อกดเข้ามาในด้านใดด้านหนึ่งแล้ว จะพบกับหน้าจอ Town โดยจะมีภาพพื้นหลังเป็นรูปของสถานที่สำคัญในจังหวัดนั้น ๆ โดยทางด้านซ้ายของจอจะมีปุ่ม Play Game (3) เพื่อเข้าสู่การเล่น เกม กับปุ่ม Sleep Log (4) เพื่อดูประวัติการนอน ส่วนทางด้านขวาจะมีปุ่ม Profile (5) เพื่อดูประวัติ ส่วนตัวของผู้เล่น และมีคะแนนกับเหรียญที่เก็บได้ในเกมบอกไว้ด้วย (6) ดังรูปที่ ข.3



รูปที่ ข.3 หน้าจอด้านกรุงเทพมหานคร

### ข.1.4 หน้าจอ Play Game

เมื่อกดปุ่ม Play Game จะเข้าสู่การเล่นเกมที่ ผู้เล่นจะต้องกดส่วนใดส่วนหนึ่งของหน้าจอเพื่อบังคับให้ตัวละคร (7) กระโดด โดยจะมีเหรียญ (8) ให้เก็บเป็นคะแนน (9) และมีสิ่งกีดขวางต่าง ๆ ที่ต้องกระโดดข้าม (10) หากตัวละครโดนสิ่งกีดขวางค่าพลังจะลดลง (11) หากค่าพลังลดลงจนหมดจะถือว่าไม่ผ่านด่านนั้นและต้องเริ่มเกมใหม่ในหน้า Town ของด่านนั้น ๆ แต่ถ้าผู้เล่นผ่านด่านได้สำเร็จโดยการวิ่งจนครบระยะทางที่กำหนด (12) และเข้าสู่เส้นชัย ผู้เล่นจะผ่านเข้าสู่ด่านต่อไป ดังรูปที่ ข.4

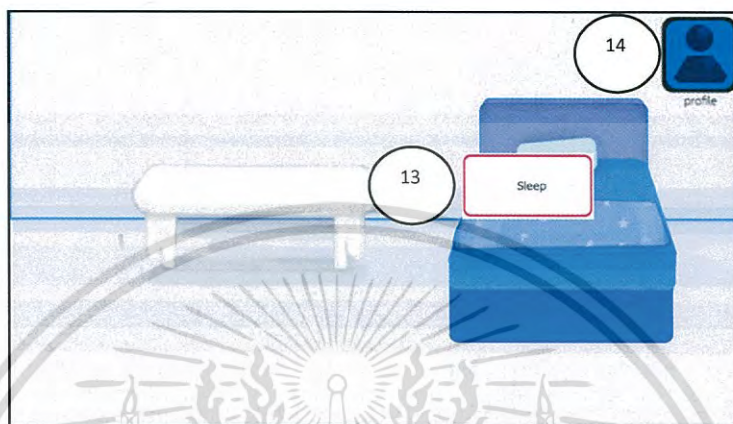


รูปที่ ข.4 หน้าจอ Play Game

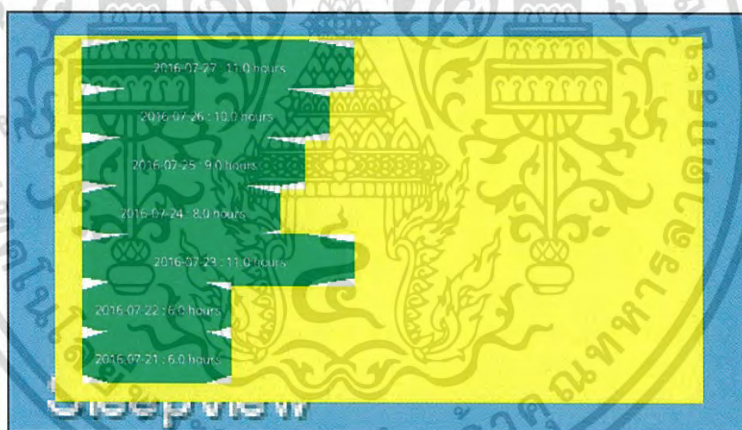
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ข.1.5 หน้าจอ Sleep Room

เมื่อผู้เล่นกดปุ่ม Sleep Log จากในหน้าจอ Town จะพบกับห้อง Sleep Room ก่อน โดยมีปุ่ม Sleep (13) และปุ่ม Profile (14) ดังรูปที่ ข.5 ถ้าผู้เล่นกดปุ่ม Sleep จะเข้าสู่หน้าจอ Sleep View ซึ่งผู้เล่นสามารถดูประวัติการนอนของตัวเองได้ใน 1 สัปดาห์ โดยแสดงผลในรูปแบบกราฟแท่ง ดังรูปที่ ข.6



รูปที่ ข.5 หน้าจอ Sleep Room



รูปที่ ข.6 หน้าจอ Sleep View

### ข.1.6 หน้าจอ Profile

เมื่อผู้เล่นกดปุ่ม Profile จากในหน้า Town จะพบกับหน้าจอ Profile ที่แสดงประวัติส่วนตัวต่าง ๆ ของผู้เล่น เช่น ก้าวเดินในแต่ละวัน น้ำหนัก ส่วนสูง การเผาผลาญพลังงานจากกิจกรรม การเผาผลาญพลังงานจากก้าวเดิน BMR การเผาผลาญพลังงานทั้งหมด ตามลำดับ ดังรูปที่ ข.7



รูปที่ ข.7 หน้าจอ Profile

### ข.1.7 หน้าจอ View Score

เมื่อผู้เล่นกดปุ่มรูปหนังสือทางซ้ายล่างของหน้าจอ จากในหน้า Profile ก็จะพบกับหน้าจอ View Score ที่จะแสดงคะแนนการเล่นของผู้เล่นและเพื่อน ๆ ที่เล่นเกม FitGaming 2 ด้วยกัน โดยจะสรุปผลเป็นอันดับ โดยเรียงตามคะแนน นอกจากนี้ยังบอกตำแหน่งของผู้เล่นว่าอยู่ภาคไหนด้านไหนได้อีกด้วย

| Name  | Score | Section | Province |
|-------|-------|---------|----------|
| sense | 10    | central | bangkok  |
| aa    | 0     | central | bankok   |
| 111   | 0     | central | bankok   |
| gotty | 3     | central | bangkok  |

รูปที่ ข.8 หน้าจอ View Score

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ค

### ข้อมูลสถานที่ท่องเที่ยวจากจังหวัดในภาคกลาง

#### ค.1 กรุงเทพมหานคร

- 1) เสาชิงช้า [16] เป็นสถาปัตยกรรมที่สร้างขึ้นเพื่อใช้ประกอบพิธีโล้ชิงช้า ในพระราชพิธีตรียัมพวาย ตรีปวายของศาสนาพราหมณ์-ฮินดู โดยทั่วไปหมายถึงเสาชิงช้าที่ตั้งอยู่หน้าวัดสุทัศนเทพวราราม และลานหน้าศาลาว่าการกรุงเทพมหานคร (ลานคนเมือง) ใกล้กับเทวสถานโบสถ์พราหมณ์ ในพื้นที่แขวงเสาชิงช้าและแขวงวัดราชบพิธ เขตพระนคร กรุงเทพมหานคร ซึ่งถือเป็นสัญลักษณ์อย่างหนึ่งของกรุงเทพมหานคร แม้พิธีโล้ชิงช้าได้เลิกไปแล้วตั้งแต่สมัยพระบาทสมเด็จพระปกเกล้าเจ้าอยู่หัว รัชกาลที่ 7 ก็ตามนอกจากนี้ในประเทศไทยยังมีเสาชิงช้าอีกแห่งหนึ่ง ตั้งอยู่ที่ หน้าหอพระอิศวร เมืองนครศรีธรรมราช ซึ่งมีการประกอบพิธีโล้ชิงช้ามาแต่โบราณเช่นกัน แต่ได้เลิกไปก่อนที่จะมีการก่อสร้างขึ้นใหม่ในภายหลัง โดยจำลองแบบมาจากเสาชิงช้าที่กรุงเทพมหานคร
- 2) เอเชียติก เดอะ ริเวอร์ฟรอนท์ (อังกฤษ: Asiatique The Riverfront) [17] เดิมเคยเป็นที่ตั้งของท่าเรือและบริษัท อีสต์เอเชียติก บริษัทเดินเรือสัญชาติเดนมาร์ก มาเปิดกิจการในสมัยรัชกาลที่ 5 โดยสร้างเป็นโกดัง โรงเลื่อย และนำเครื่องจักรขนาดใหญ่หลายตัวมาติดตั้งไว้ ในช่วงสงครามมหาเอเชียบูรพา กองทัพญี่ปุ่นได้เข้ามายึดท่าเรือและคลังสินค้าของบริษัท อีสต์เอเชียติก เพื่อใช้เป็นฐานกำลังและคลังแสง และยังปรากฏมาจนถึงปัจจุบัน จนกระทั่งในปี พ.ศ. 2555 ได้มีพัฒนาปรับปรุงเป็นศูนย์การค้าอย่างในปัจจุบัน ได้ชื่อว่าเป็นสถานที่ท่องเที่ยวและช้อปปิ้งที่มีทัศนียภาพที่สวยงาม โดยมีจุดเด่น คือชิงช้าสวรรค์ขนาดใหญ่ "เอเชียติก สกาย" ซึ่งเมื่อขึ้นไปแล้วจะเห็นทัศนียภาพโดยรอบของกรุงเทพมหานครในแบบมุมสูงได้รอบตัว
- 3) ทำเนียบรัฐบาล [18] เป็นสถานที่ราชการสำคัญที่สุดแห่งหนึ่งของประเทศไทย เนื่องจากเป็นสถานที่ทำงานของรัฐบาลไทย นายกรัฐมนตรี รองนายกรัฐมนตรี รัฐมนตรีประจำสำนักนายกรัฐมนตรี ตลอดจนข้าราชการและเจ้าหน้าที่ ในสังกัดสำนักนายกรัฐมนตรี, สถานที่ประชุมคณะรัฐมนตรี, สถานที่ต้อนรับบุคคลสำคัญระดับผู้นำต่างประเทศ ซึ่งมาเยือนประเทศไทย และยังใช้เป็นสถานที่จัดรัฐพิธี เช่น สโมสรสันนิบาต เนื่องในวันเฉลิมพระชนมพรรษาฯ เป็นต้น ตั้งอยู่เลขที่ 1 ถนนพิษณุโลก แขวงดุสิต เขตดุสิต กรุงเทพมหานคร มีเนื้อที่ทั้งสิ้น 27 ไร่ 3 งาน 44 ตารางวา
- 4) พระที่นั่งอนันตสมาคม [19] เดิมเป็นท้องพระโรงของพระราชวังดุสิต ชื่อของพระที่นั่งนี้ นำมาจากพระที่นั่งองค์หนึ่งในพระบรมมหาราชวัง ซึ่งสร้างขึ้นในรัชสมัยพระบาทสมเด็จพระจุลจอมเกล้าเจ้าอยู่หัว (รัชกาลที่ 5) พระที่นั่งองค์นี้ จัดได้ว่า เป็นรัฐสภาแห่งแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของประเทศไทย เพราะว่า ในช่วงที่เพิ่งเปลี่ยนแปลงการปกครองเป็นระบอบประชาธิปไตย อันมีพระมหากษัตริย์เป็นประมุข ได้ใช้พระที่นั่งองค์นี้ในการประชุมสภาผู้แทนราษฎร ก่อนที่จะย้ายมาอยู่ ณ อาคารรัฐสภา ปัจจุบัน ซึ่งอยู่ข้างๆพระที่นั่งองค์นี้ รวมทั้งยังเป็นสถานที่ประกอบพระราชพิธี รัฐพิธี สำคัญๆมากมาย อาทิ รัฐพิธีพระราชทานรัฐธรรมนูญแห่งราชอาณาจักรไทย พระราชพิธีสถาปนา สมเด็จพระบรมโอรสาธิราชฯ สยามมกุฎราชกุมาร พระราชพิธีสมโภชเดือนและขึ้นพระอู่ พระเจ้าหลานเธอ พระองค์เจ้าทีปังกรรัศมีโชติ และการเสด็จออกมหาสมาคมสี่ครั้งในรัชกาลปัจจุบัน

- 5) พระบรมมหาราชวัง (อังกฤษ: Grand Palace) [20] หรือพระราชวังพระนคร เป็นที่ประทับของพระมหากษัตริย์สมัยรัตนโกสินทร์ ตั้งแต่รัชสมัยพระบาทสมเด็จพระพุทธยอดฟ้าจุฬาโลกมหาราช จนถึงรัชสมัยพระบาทสมเด็จพระจุลจอมเกล้าเจ้าอยู่หัว ปัจจุบัน พระบรมมหาราชวังเป็นสถานที่ท่องเที่ยว ที่ได้รับความนิยมมากเป็นอันดับที่ 16 ของโลก โดยมีผู้เข้าเยี่ยมชมในปี พ.ศ. 2549 เป็นจำนวนถึง 8,995,000 คน พระบรมมหาราชวังได้ใช้เป็นที่ประทับและศูนย์กลางการปกครองของพระมหากษัตริย์สมัยรัตนโกสินทร์มาตลอด จนถึงกลางรัชสมัยพระบาทสมเด็จพระจุลจอมเกล้าเจ้าอยู่หัว ในรัชสมัยพระบาทสมเด็จพระมงกุฎเกล้าเจ้าอยู่หัวและ พระบาทสมเด็จพระปกเกล้าเจ้าอยู่หัว ได้เสด็จประทับเพียงครั้งคราว ในรัชสมัย พระบาทสมเด็จพระปรเมนทรมหาอานันทมหิดลได้เสด็จประทับ ณ พระที่นั่งบรมพิมาน ในพระบรมมหาราชวัง กระทั่งเสด็จสวรรคต หลังจากนั้นจนถึงปัจจุบัน มิได้มีพระมหากษัตริย์เสด็จมาประทับในพระบรมมหาราชวังเป็นการถาวรอีก ปัจจุบันพระที่นั่งบรมพิมาน เป็นเขตพระราชฐาน และที่ประทับส่วนพระองค์ใน ทูลกระหม่อมหญิงอุบลรัตนราชกัญญา สิริวัฒนาพรรณวดี ในรัชกาลปัจจุบัน พระบาทสมเด็จพระเจ้าอยู่หัวเสด็จประทับบ้างเป็นครั้งคราว เช่นเวลาซ้อมพระตำหนักจิตรลดารโหฐาน หรือเวลามีการพระราชพิธี เป็นต้น พระบรมมหาราชวังใช้เป็นสถานที่ประกอบพระราชพิธีสำคัญต่าง ๆ ตามพระราชประเพณี, เป็นที่รับแขกเมือง และพระราชอาคันตุกะ รวมทั้งเป็นที่ตั้งพระบรมศพและพระศพของพระบรมวงศานุวงศ์ชั้นสูง ส่วนบริเวณเขตพระราชฐานชั้นนอกได้ใช้เป็นที่ตั้งของสถานที่ราชการ เช่น สำนักพระราชวัง, สำนักราชเลขาธิการและราชบัณฑิตยสถาน และเขตพระราชฐานชั้นในก็ได้เป็นที่อยู่อาศัยของเจ้านายฝ่ายในอีกต่อไป แต่ยังคงเป็นที่ทำการและที่พำนักของข้าราชการสำนักพระราชวัง ฝ่ายพระราชฐานชั้นในบางส่วนซึ่งล้วนเป็นสตรีทั้งสิ้น

## ค.2 จังหวัดพระนครศรีอยุธยา

- 1) พระราชวังบางปะอิน [21] ตั้งอยู่ในตำบลบ้านเลน อำเภอบางปะอิน จังหวัดพระนครศรีอยุธยา อยู่ห่างจากเกาะเมืองลงมาทางทิศใต้ประมาณ 18 กิโลเมตร เป็นพระราชวังโบราณตั้งแต่สมัยกรุงศรีอยุธยา สร้างขึ้นโดยสมเด็จพระเจ้าปราสาททอง เนื่องจากเป็นที่ประสูติของพระองค์ ใช้เป็นสถานที่ที่ทรงใช้ประทับแรม ของพระมหากษัตริย์แห่งกรุงศรีอยุธยา ด้วยเป็นพระราชวังใกล้พระนครนั่นเอง หลังจากการเสียกรุงศรีอยุธยาครั้งที่สอง พระราชวังบางปะอินถูกปล่อยให้รกร้างมาระยะหนึ่ง แต่กลับมาเป็นที่รู้จักอีกครั้งโดยสุนทรภู่ซึ่งได้ตามเสด็จพระบาทสมเด็จพระพุทธยอดฟ้าจุฬาโลกมหาราชไปนมัสการพระพุทธบาทสระบุรีได้ประพันธ์ถึงพระราชวังบางปะอินไว้ในนิราศพระบาท จนกระทั่ง ในรัชสมัยพระบาทสมเด็จพระจอมเกล้าเจ้าอยู่หัว จึงได้เริ่มการบูรณะพระราชวังขึ้น และในรัชสมัยพระบาทสมเด็จพระจุลจอมเกล้าเจ้าอยู่หัว ได้บูรณะครั้งใหญ่ โดยสร้างพระที่นั่ง พระตำหนัก และตำหนักต่าง ๆ ขึ้นมากมายเพื่อใช้เป็นที่พักประทับรับรองพระราชอาคันตุกะ และพระราชทานเลี้ยงในโอกาสต่าง ๆ ปัจจุบันพระราชวังบางปะอินอยู่ในความดูแลของสำนักพระราชวัง และยังเป็นสถานที่แปรพระราชฐานของพระบาทสมเด็จพระเจ้าอยู่หัวและพระบรมวงศานุวงศ์ รวมถึงประกอบพระราชพิธีสังเวยพระปาย แต่ได้เปิดให้ประชาชนและนักท่องเที่ยวเข้าชมได้ โดยต้องแต่งกายให้สุภาพ
- 2) วัดใหญ่ชัยมงคล [22] เดิมชื่อ "วัดป่าแก้ว" หรือ "วัดเจ้าไท" ตั้งอยู่ทางทิศตะวันออกเฉียงใต้ของเกาะพระนคร ปัจจุบันเป็นพื้นที่ตำบลคลองสวนพลู อำเภอพระนครศรีอยุธยา จังหวัดพระนครศรีอยุธยา จุดเด่นของวัดได้แก่เจดีย์องค์ใหญ่ที่เชื่อกันว่า ได้รับการปฏิสังขรณ์ขึ้นใหม่ในสมัยสมเด็จพระนเรศวรมหาราช ที่ภายในได้มีการค้นพบชัยมงคลคาถาบรรจุอยู่ ภายในพระอุโบสถ เป็นที่ประดิษฐานพระพุทธชัยมงคล พระประธานที่เป็นสิ่งศักดิ์สิทธิ์ของวัด นอกจากนี้แล้ว ภายในวัดยังเป็นที่ประดิษฐานศาลสมเด็จพระนเรศวรมหาราช ที่ก่อสร้างแล้วเสร็จในปี พ.ศ. 2544 อีกด้วย
- 3) วัดพระศรีสรรเพชญ์ หรือ วัดพระศรีสรรเพชญ์ [23] เป็นวัดหลวงในพระราชวังโบราณอยุธยา ไม่มีพระสงฆ์จำพรรษา ซึ่งเป็นต้นแบบของ วัดพระศรีรัตนศาสดาราม กรุงเทพมหานคร วัดพระศรีสรรเพชญ์ เดิมในสมัยสมเด็จพระรามาธิบดีที่ 1 ใช้เป็นที่ประทับ ต่อมาสมเด็จพระบรมไตรโลกนาถ ทรงสร้างพระราชมณเฑียรขึ้นใหม่ทางตอนเหนือ แล้วจึงโปรดฯให้ยกเป็นเขตพุทธาวาส เพื่อประกอบพิธีสำคัญต่าง ๆ ของบ้านเมือง จึงเป็นวัดในเขตพระราชวังที่ไม่มีพระสงฆ์จำพรรษา แตกต่างกับวัดมหาธาตุสุโขทัย ที่มีพระสงฆ์จำพรรษา ทั้งวัดมหาธาตุ สุโขทัย, วัดพระศรีสรรเพชญ์ อยุธยา และวัดพระศรีรัตนศาสดาราม ต่างก็ถูกสถาปนาขึ้นในมูลเหตุการสร้างวัดเดียวกันนั้นคือ "สร้างเพื่อเป็นวัดประจำพระราชวัง" ราวรัชสมัยสมเด็จพระเจ้าอยู่หัวบรมโกศ มีการบูรณะปฏิสังขรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัดหลวงแห่งนี้เป็นครั้งแรก ในสมัยพระบาทสมเด็จพระจุลจอมเกล้าเจ้าอยู่หัว พระยาโบราณราชธานินทร์ สมุหเทศาภิบาลมณฑลกรุงเก่าได้ดำเนิน การขุดสมบัติจากกรุ ภายในเจดีย์ พบพระพุทธรูป เครื่องทอง มากมาย และในสมัย จอมพล ป.พิบูลสงคราม ได้มีการบูรณะวัดนี้จนมีสภาพที่เห็นอยู่ในปัจจุบัน

- 4) ศูนย์ศิลปาชีพบางไทร [24] [25] เป็นศูนย์ศิลปาชีพเพื่อมุ่งฝึกอาชีพเกี่ยวกับงานศิลปหัตถกรรมต่างๆ วิชาที่สอนให้แก่เกษตรกรได้แก่ การประดิษฐ์ผลิตภัณฑ์จากเส้นใยพืช, การแกะสลัก, การจักสาน, การทำตุ๊กตา, การทำดอกไม้ประดิษฐ์, การทำเครื่องเรือน, การทอผ้า ผลิตภัณฑ์จากผ้า การย้อมสี, ช่างเชื่อมและเครื่องเคลือบดินเผา โดยผลิตภัณฑ์ที่เสร็จแล้วจะส่งไปจำหน่ายที่ร้านจิตรลดาทุกสาขาทั่วประเทศ เมื่อพ.ศ. 2523 สมเด็จพระนางเจ้าสิริกิติ์ พระบรมราชินีนาถ โปรดเกล้าฯ ให้จัดตั้งศูนย์ฯ แห่งนี้ขึ้นริมฝั่งแม่น้ำเจ้าพระยา ต. ช่างใหญ่ อ. บางไทร ในเขตที่ดินปฏิรูปเพื่อการเกษตรกรรม พื้นที่กว่า 1,000 ไร่ มุ่งฝึกงานช่างฝีมือแบบศิลปะไทยโบราณให้แก่เกษตรกรที่สนใจฝึกอาชีพ เป็นรายได้พิเศษจากช่วงที่ว่างจากงานเกษตร โดยทางศูนย์ฯ มีผู้ชำนาญงานช่างแขนงต่าง ๆ มาฝึกสอน เมื่อสามารถผลิตงานได้แล้ว ศูนย์ฯ จะรับซื้อผลงานไปจำหน่าย เปิดอบรมศิลปาชีพรุ่นแรกเมื่อวันที่ 1 พ.ค. 2524 สมเด็จพระนางเจ้าสิริกิติ์ พระบรมราชินีนาถ เสด็จพระราชดำเนินเปิดศูนย์ศิลปาชีพบางไทรเมื่อวันที่ 7 ธ.ค. 2527 ปัจจุบันมีกว่า 20 แผนก เช่น แผนกตัดเย็บเสื้อผ้า แผนกทอผ้าลายตีนจก แผนกเครื่องเรือนไม้ เป็นต้น อบรมปีละสองรุ่น รุ่นละ 500 คน ใช้เวลาอบรม 6 เดือน โดยให้ที่พัก อาหาร สวัสดิการ และเบี้ยเลี้ยง ผลงานที่ผลิตได้นอกจากจะจำหน่ายที่ศูนย์ฯ แล้ว ยังส่งไปจำหน่ายที่ร้านจิตรลดาซึ่งมีสาขาทั่วประเทศ และส่งออกต่างประเทศด้วย
- 5) หมู่บ้านญี่ปุ่น [26] ตั้งอยู่ที่ตำบลเกาะเรียน เมื่อปลายคริสต์ศตวรรษที่ 16 ชาวต่างประเทศเข้ามาค้าขายใน กรุงศรีอยุธยาจำนวนมากขึ้น ทางการญี่ปุ่นได้อนุญาตให้ชาวญี่ปุ่นเดินเรือออกไปค้าขายกับชาวต่างชาติในบรรดาพวกที่ไปค้าขายมีพวกหนึ่งเดินทางมายังกรุงศรีอยุธยา พระเจ้าแผ่นดินไทยมีพระบรมราชานุญาตให้ชาวญี่ปุ่น มาตั้งหลักแหล่งในกรุงศรีอยุธยาครอบนอกเกาะเมืองเหมือนชาติอื่น ๆ นับตั้งแต่นั้นมาก็มีชาวญี่ปุ่นเข้ามาอาศัยอยู่ในอยุธยามากขึ้น โดยมีหัวหน้าปกครองในกลุ่มตน หัวหน้าชาวญี่ปุ่นในขณะนั้นคือ นากามาเซ ยามาตะ เป็นผู้มีอำนาจและเป็นที่โปรดปรานของ สมเด็จพระเจ้าทรงธรรม จนได้รับแต่งตั้งเป็นออกญาเสนาภิมุขรับราชการ ต่อมาได้รับแต่งตั้งเป็นเจ้าเมืองนครศรีธรรมราชจนสิ้นชีวิต ปัจจุบันสมาคมไทย-ญี่ปุ่นได้สร้าง หุ่นจำลอง นากามาเซ ยามาตะ และจารึกประวัติศาสตร์ความเป็นมาของหมู่บ้านญี่ปุ่นในสมัยกรุงศรีอยุธยามาตั้งไว้ภายในหมู่บ้าน มีอาคารจัดแสดงเรื่องความสัมพันธ์ระหว่างอยุธยากับต่างประเทศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ค.3 จังหวัดเพชรบุรี

- 1) เขาวัง หรือ อุทยานประวัติศาสตร์พระนครคีรี [27] เป็นสถานที่เที่ยวที่สำคัญของเพชรบุรีซึ่งถือได้ว่าเป็นสัญลักษณ์ของเมืองเพชรบุรี เพราะเมื่อเข้าสู่เมืองเพชรบุรี สิ่งที่จะได้เห็นโดดเด่นเห็นมาแต่ไกล คือ เขาวัง ซึ่งมีเจดีย์และอาคารสีขาวสะอาด เขาวังเป็นโบราณสถาน เก้าแก่คูเมืองเพชรบุรี ตั้งอยู่บนยอดเขาสูง 92 เมตร เดิมเรียกว่า เขาสมนหรือเขาคีรี พระบาทสมเด็จพระจอมเกล้าเจ้าอยู่หัว รัชกาลที่ 4 ทรงพอพระราชหฤทัยที่จะสร้างพระราชวังสำหรับเสด็จแปรพระราชฐานขึ้นบนยอดเขาแห่งนี้ จึงโปรดเกล้าฯ ให้เจ้าพระยาเพชรนิสสัย ศรีสวัสดิ์ปลัดเมือง เพชรบุรีเป็นนายงานก่อสร้างจนสำเร็จเรียบร้อยเมื่อปี พ.ศ.2403 ทรงพระราชทานนามว่า พระนครคีรี แต่ชาวเมืองเพชรเรียกกันติดปากว่า เขาวัง สืบมาจนบัดนี้
- 2) คาเมล รีพับลิก (Camel Republic) [28] ตั้งอยู่บนพื้นที่กว่า 35 ไร่ เป็นสถานที่ท่องเที่ยวแห่งใหม่ของเมืองชะอำ จังหวัดเพชรบุรี ในรูปแบบสวนสนุกในสไตล์โมร็อกโก ถูกสร้างขึ้นภายใต้คอนเซ็ปต์อูฐและฤดูร้อน โดดเด่นด้วยสีสันฉูดฉาด ภายในออกแบบตัวอาคารให้เหมือนสถาปัตยกรรมอาคารในแถบตะวันออกกลางสไตล์โมร็อกโก (Moroccan Architecture) ผสมผสานกับศิลปะสีเส้นสดใส ภายในถูกแบ่งออกเป็น 3 ส่วน ได้แก่ "โซนเครื่องเล่น" ที่มีให้เลือกหลากหลายชนิด ซึ่งเป็นนวัตกรรมเครื่องเล่นที่ทันสมัยและมีที่นั่งแห้งเดียวในเอเชีย ต่อมาคือ "โซนสวนสัตว์" เป็นแหล่งรวบรวมสัตว์หายากหลายชนิด และโซนสุดท้าย คือ "ร้านค้า" มีการจำหน่ายสินค้าและของที่ระลึก รวมถึงร้านอาหารอร่อย ๆ มากมาย
- 3) ซานโตรินี่ พาร์ค ชะอำ [29] บริหารงานโดยคุณนัสวีร์ ตันติจิรสกุล ทายาทของคุณสุพจน์ ตันติจิรสกุล เจ้าของธุรกิจพรีเมียม เอาร์ทเล็ท ชะอำ ดำรงตำแหน่ง กรรมการผู้จัดการ และบริหารโครงการภายใต้บริษัท ธิม พลาซ่า ดีวิลอปเม้นท์ จำกัด อีกหนึ่งบริษัทในเครือพีน่า กรุ๊ป กลุ่มบริษัทที่มีธุรกิจหลากหลายและประสบการณ์ด้านค้าปลีกมากกว่า 30 ปี โดยเริ่มต้นจากธุรกิจค้าปลีกสินค้าแฟชั่น หัวหิน-ชะอำเป็นพื้นที่ที่คนไปเที่ยว การตอบโจทย์คนไปเที่ยวได้ต้องมีกิจกรรมให้ทำ มีทั้งช้อปปิ้งและความสนุกสนานผจญภัยลงไปให้ผู้คนระหว่างท่องเที่ยวด้วยเครื่องเล่นที่คิดสรรมาให้ตอบโจทย์ทุกคนในครอบครัว นอกจากนี้ยังมีกิจกรรมการตลาดที่จัดให้มีขึ้นอย่างต่อเนื่องในโครงการ บรรยากาศที่ให้ความสำคัญมากคือคอนเซ็ปต์ซึ่งอิงกับเทรนด์ปัจจุบันที่คนชอบถ่ายรูป ชอบแชร์รูป ดังนั้นคอนเซ็ปต์ของที่นี่ คือทุกมุมต้องถ่ายรูปสวย เพื่อให้ทุกคนรู้สึกมีความสุขและร่วมเป็นส่วนหนึ่งของบรรยากาศ จึงได้ลงทุนกับด้านสถาปัตยกรรมและงานประติมากรรมในพื้นที่ค่อนข้างสูง ให้ความสำคัญการสร้างบรรยากาศโครงการให้เป็นสีฟ้า-ขาวสไตล์เกาะซานโตรินี่ ประเทศกรีซ มาจากคุณสุพจน์ ซึ่งมองว่าแม้พื้นที่ตรงนี้จะไม่ได้ติดทะเล แต่หัวหิน-ชะอำถูกมองว่าเป็นเมืองทะเล ทำให้นักถึงเมืองที่เป็นเกาะที่มีชื่อเสียงและมีคาแร็กเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ชัดเจน คำตอบคือซานโตรินี่ เพราะถ้าเห็นสถาปัตยกรรมของเกาะนี้จะทราบได้ทันทีว่าคือที่ไหน ซานโตรินี่เป็นเมืองเกาะ มีเรื่องปัจจัยของน้ำเข้ามาเกี่ยวข้อง นี่คือสาเหตุว่าทำไมที่นี่จึงมีอะไรที่เกี่ยวกับน้ำค่อนข้างเยอะ ทั้งน้ำตกชั้นบันได น้ำพุ หรือวอเตอร์บอลสำหรับเด็ก มีพื้นที่สีเขียวของต้นไม้ เพราะฉะนั้นบรรยากาศข้างในจึงเพลิดเพลินมากเหมาะสำหรับการพักผ่อนกับกลุ่มเพื่อนและครอบครัว

- 4) พระราชานิเวศน์มฤคทายวัน [30] ตั้งอยู่ในเขตพื้นที่ตำรวจตระเวนชายแดนค่ายพระรามหก ประกอบด้วยหมู่พระที่นั่งใหญ่ 3 องค์ ปลูกเรียงรายไปตามแนวชายหาด ทุกองค์สร้างอย่างแบบยุโรปสิ่งที่น่ายอกย่องของสถานที่แห่งนี้ ได้แก่ ความตั้งใจและความพยายาม ที่จะสร้างที่ประทับให้เหมาะสมกับสภาพภูมิประเทศและภูมิอากาศอย่างแท้จริงโดยปรับให้เข้ากับภูมิอากาศของไทยที่ร้อนชื้นได้เป็นอย่างดี สร้างขึ้นด้วยไม้สักทองทั้งหลัง ตั้งอยู่ท่ามกลางธรรมชาติของพรรณไม้ร่มรื่นและหาดทรายขาวสะอาดตา ผสานเสียงเกลียวคลื่นดังกระทบฝั่งอยู่เป็นระยะ ๆ นับเป็นบรรยากาศที่สงบและรื่นรมย์ดุจตั้งแต่กาลก่อน รูปแบบสถาปัตยกรรมพระราชานิเวศน์มฤคทายวันมีได้ฤกษ์ที่โปร่งโล่งเปิดรับลมทะเลที่พัดเข้าสู่หมู่พระที่นั่งให้ความเย็นสบายตลอดวัน หลังคาทรงปั้นหยาซึ่งกันแดดและฝนได้ดี กระเบื้องมุงหลังคาทำด้วยซีเมนต์เคลือบสีแดง แนวระเบียงเชื่อมองค์พระที่นั่งทั้งสามอย่างฝรั่ง ที่เรียกว่า คัฟเวอร์เวย์ (Cover Way) ให้ความสะดวกสบายและปลอดภัยในการสัญจรไปมา ไม่ต้องเดินขึ้นลงบันไดบ่อยๆ พื้นระเบียงและพระที่นั่งทำด้วยไม้สักลงเงาดูโอ้อ่าสวยงามยิ่ง ส่วนเพดานใช้คานไม้ตัดโค้งบรรจुरะหว่างช่วงเสาทุกช่วงตลอด แนวระเบียง เพิ่มความอ่อนช้อยให้กับพระราชานิเวศน์เช่นเดียวกับสถาปัตยกรรม ยุโรปซึ่งจัดจางหระโค้งของวงกบหน้าต่างตอนบนให้ ความอ่อนหวานและยังคงได้รับความนิยมสืบมาจนถึงปัจจุบัน พระราชานิเวศน์แห่งนี้มีตำนานที่เล่าขานต่อ ๆ สืบเนื่องมาจากเมื่อคราวที่พระนางเจ้าอินทรศักดิศจี ทรงพระครรภ์นั้นองค์พระมหากษัตริราช เจ้าทรงพระเกษมสำราญยิ่งด้วยทรงมุ่งหวังว่าจะทรงมีพระปิโยรสแต่ความหวังทั้งมวลก็สิ้นสลายเมื่อ สมเด็จพระนางเจ้าฯ ไม่สามารถมีพระประสูติกาลได้ยามนั้นพระองค์ท่านทรงอภิบาลพระมหเสีด้วยน้ำพระทัยเป็นห่วงและเศร้าสร้อย ณ พระที่นั่งสมุทรพิมานแห่งนี้ จึงเป็นที่มาของชื่อพระราชวังแห่งความรักและความหวัง พระบาทสมเด็จพระมงกุฎเกล้าเจ้าอยู่หัวได้เสด็จ มาประทับ ณ พระราชานิเวศน์มฤคทายวันพร้อมด้วยสมเด็จพระนางเจ้าสุวัทนาฯ อีกครั้งระหว่างวันที่ 12 เมษายน ถึงวันที่ 20 มิถุนายน 2468 การเสด็จครั้งนี้เสมือน หนึ่งการเสด็จมาเพื่ออำลาพระราชานิเวศน์ที่ทรงรักโดยแท้ เพราะเมื่อเสด็จกลับพระนครแล้วต่อมาไม่นาน ก็เสด็จสวรรคตจากวันนั้นนับเนื่องมาจนถึงปัจจุบัน เป็นเวลาร่วม 70 กว่าปีแล้ว พระราชานิเวศน์ได้ถูกทอดทิ้งจนทรุดโทรมลงเป็นลำดับจนกรมตำรวจ ได้เข้ามาดูแลเป็นการถาวร โดยใช้เป็นที่ตั้งกองกำกับการ 1 กองบังคับการฝึกพิเศษ กองบัญชาการตำรวจตระเวนชายแดน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และได้ทำการบูรณะซ่อมแซม ปลุกต้นไม้เพิ่มเติมจนงดงามร่มรื่น พระราชนิเวศน์มฤคทายวันได้เผยความงามอันล้ำค่าจากงานฝีมือ ของช่างไทยและช่างยุโรปในอดีตผสมผสานมาเป็นงานสถาปัตยกรรมอันงดงามนุ่มนวล เป็นสถานที่ท่องเที่ยวที่น่าไปเยี่ยมชมเยือน และควรแก่การทะนุถนอม หวงแหวนเป็นมรดกล้ำค่าของอนุชนชาวไทยสืบไป

- 5) พระรามราชนิเวศน์ หรือ พระราชวังบ้านปืน หรือ วังบ้านปืน [31] สร้างสมัยของพระบาทสมเด็จพระจุลจอมเกล้าเจ้าอยู่หัว (รัชกาลที่ 5) มีพระราชประสงค์ให้สร้างพระราชวังแบบยุโรป เพื่อใช้สำหรับแปรพระราชฐานในฤดูฝน สร้างแบบสถาปัตยกรรมยุโรป พระราชวังแห่งนี้ถือว่าเป็นพระราชวังที่งดงามแห่งหนึ่งของไทย วังบ้านปืนตั้งอยู่ที่ฝั่งตะวันตกของแม่น้ำเพชรบุรี ที่บ้านปืน อำเภอเมือง จังหวัดเพชรบุรี พระบาทสมเด็จพระจุลจอมเกล้าเจ้าอยู่หัว โปรดเกล้าฯ ให้สมเด็จพระเจ้าลูกยาเธอ เจ้าฟ้ากรมขุนนครสวรรค์วรพินิต ผู้บัญชาการทหารเรือ กับ พระเจ้าน้องยาเธอกรมหลวงดำรงราชานุภาพ (พระยศในขณะนั้น) เป็นแม่กองดำเนินการก่อสร้างพระตำหนัก ถนน และสถานที่ต่างๆ และโปรดเกล้าฯ ให้พระยาสุรินทรภาไชย ผู้ว่าราชการจังหวัดเพชรบุรีเป็นผู้จ่ายเงินสั่งของและเป็นผู้ตรวจการ ทั้งโปรดเกล้าฯ ให้หมิสเตอร์คาล ดอห์ริง นายช่างเยอรมัน เป็นผู้คิดเขียนแบบรูปพระตำหนักตามกระแสพระราชดำริ โดยมี ดอกระเตอร์ควดไบเยอร์ ชาวเยอรมัน เป็นนายช่างก่อสร้าง นายคลูเซอร์ เป็นผู้รับเหมาก่อสร้าง ภายในตำหนักชั้นล่างประกอบด้วย ห้องรอเฝ้า ห้องพระโรงกลาง ห้องเสวย ห้องเครื่อง และห้องเทียบเครื่อง สำหรับชั้นบนประกอบด้วย ห้องพระบรรทมใหญ่ ห้องพระบรรทมพระราชนิ ห่องพระบรรทมเจ้าฟ้า และห้องทรงพระอักษร พระบาทสมเด็จพระจุลจอมเกล้าเจ้าอยู่หัว เสด็จพระราชดำเนินทรงวางศิลาฤกษ์เมื่อวันที่ 19 สิงหาคม พ.ศ. 2453 แต่การก่อสร้างดำเนินได้ไม่นานก็เสด็จสวรรคต พระบาทสมเด็จพระมงกุฎเกล้าเจ้าอยู่หัว จึงโปรดเกล้าฯ ให้ช่างสร้างต่อจนเสร็จใน พ.ศ. 2459 รวมเวลาสร้างเกือบ 7 ปี และโปรดเกล้าฯ ให้มีงานฉลองพระตำหนักวันที่ 13 มิถุนายน พ.ศ. 2461