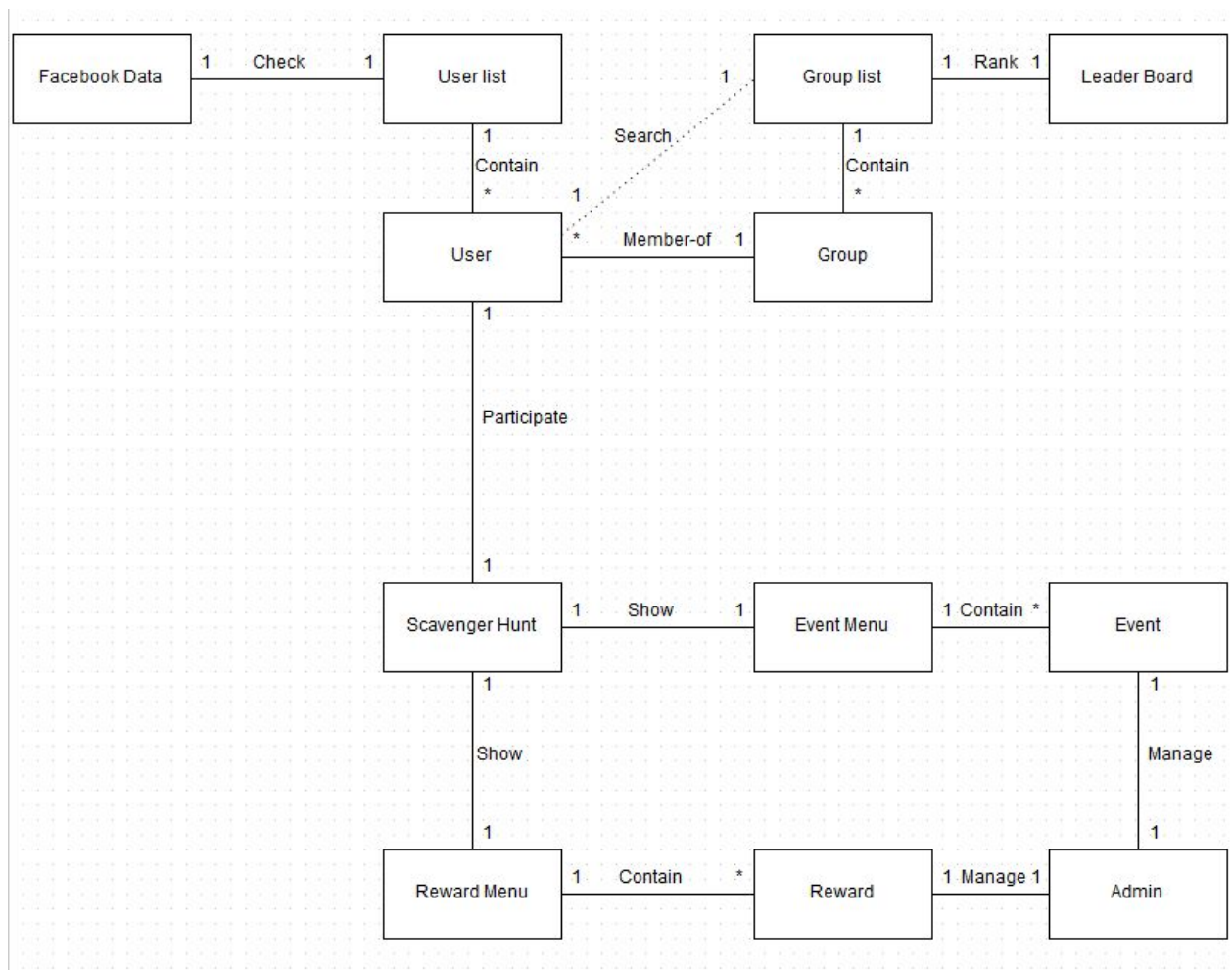


## Use cases

- Invite friends
- Participate in Scavenger hunt
- Share on Facebook
- Manage user profile
- redeem Scavenger hunt's reward
- Manage group profile
- Walk and counted
- View Leaderboard
- Update event(Scavenger hunt)
- Manage Scavenger hunt's reward

\*\*brief format use cases

## Domain model



## Domain concept

### Facebook Data

ข้อมูลส่วนตัวของผู้ใช้ จากแต่ละบัญชี facebook ที่ทำการ log in  
โดยเมื่อ log in ระบบ จะ search และทำการ match กับข้อมูลในระบบ หากมีข้อมูล User อยู่  
ก็จะดึงข้อมูล User เข้ามาใช้ หากไม่พบข้อมูลในระบบ จะถือเป็นการ Log in ครั้งแรก แล้วระบบ  
จะทำการสร้างข้อมูล User ขึ้นใหม่ พร้อมทั้ง match เข้ากับข้อมูล facebook ที่ได้รับมา

### Userlist

รายชื่อของผู้ใช้งานทั้งหมดที่มีในระบบ

### User

ข้อมูลของผู้ใช้งานแต่ละคน โดยจะเก็บรายละเอียดต่างๆ เช่น ชื่อ, Facebook ID Number,  
จำนวนก้าวที่นับได้, แด้มพิเศษ มี 2 ระดับคือ Leader ซึ่งเป็นหัวหน้ากลุ่ม  
สามารถจัดการข้อมูลเกี่ยวกับกลุ่มได้ และ Member ซึ่งเป็นเพียงสมาชิกกลุ่ม  
โดยทั้ง Leader และ Member มีสิทธิ์ในการจัดการข้อมูล User ของตนเอง

### Grouplist

รายชื่อของกลุ่มทั้งหมดที่มีในระบบ

### Group

กลุ่มที่รวมผู้ใช้งานเข้าด้วยกัน อาจเป็น ผู้ที่มีความสนใจเช่นเดียวกัน อยู่ในสถานที่, สถาบัน,  
บริษัท เดียวกัน โดยกลุ่มจะมีข้อมูลเบื้องต้นของสมาชิกในกลุ่ม และมีการแข่งขันกันระหว่างกลุ่ม  
ด้วยคะแนนก้าวเดิน ที่คำนวณจากสมาชิกทั้งหมดในกลุ่มนั้นๆ โดยมีการแสดงผลการแข่งขันทาง  
Leaderboard

### Leaderboard

บอร์ดแสดงคะแนนของแต่ละกลุ่ม โดยจะมีการจัดอันดับ(Ranking) สำหรับแต่ละเดือน และ  
แต่ละสัปดาห์ ซึ่งเมื่อครบรอบเวลา(เดือน/สัปดาห์) คะแนนประจำกลุ่มจะถูกรีเซ็ตกลับเป็น 0  
และจะเริ่มจัดอันดับใหม่อีกครั้ง

### Scavenger hunt

กิจกรรม Scavenger hunt ซึ่งเป็นกิจกรรมพิเศษ เพื่อดึงความสนใจจากผู้ใช้งาน  
ให้เข้าร่วมกิจกรรม และกระตุ้นการใช้งานของผู้ใช้งานรายใหม่  
กิจกรรมนี้ จะเป็นกิจกรรมพิเศษ ที่ถูกจัดโดย Admin โดยเมื่อ ผู้ใช้งานสามารถทำตามเงื่อนไข  
ของกิจกรรมสำเร็จ ผู้ใช้งานจะได้รับแอดัมพิเศษ สำหรับแลกรับรางวัลได้

### Event Menu

รายการกิจกรรมทั้งหมดใน Scavenger hunt ที่มีให้ผู้เล่นในแต่ละทีมได้เข้าร่วม

### Event

กิจกรรมที่ผู้เล่นสามารถทำได้ในกิจกรรม Scavenger hunt ซึ่งมี Admin จะเป็นผู้สร้าง/จัด  
มีรายละเอียดเกี่ยวกับเงื่อนไข สถานที่ และแอดัมรางวัล  
เมื่อผู้ใช้งานสามารถทำตามเงื่อนไขได้สำเร็จ

### Reward Menu

รายการรางวัลทั้งหมดที่สามารถแลกเปลี่ยนได้

### Reward

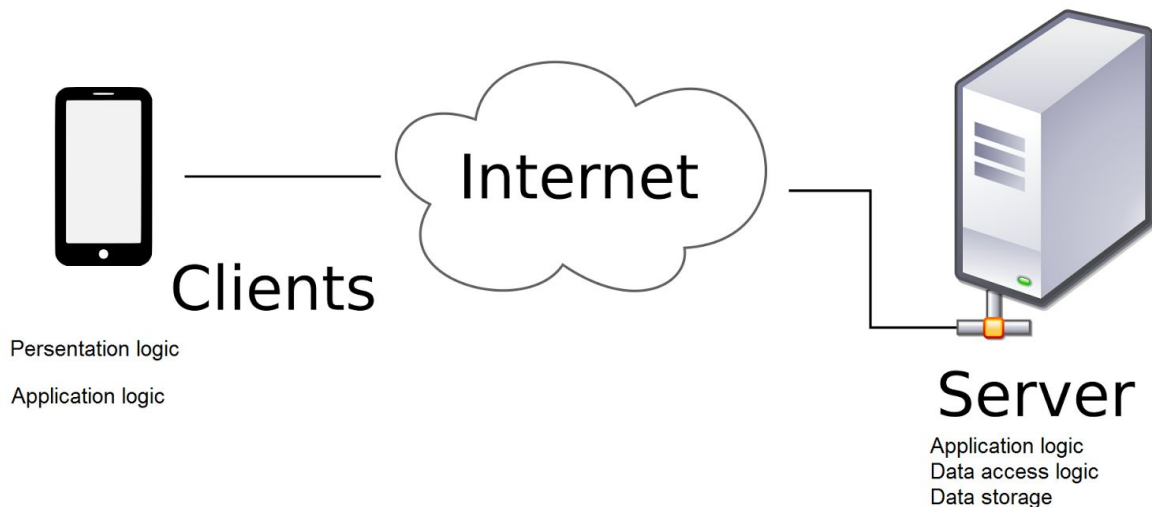
รางวัลที่ผู้ใช้งานสามารถนำแอดัมพิเศษที่มีมาแลกได้ มีการเปลี่ยนแปลง เพิ่ม-ลด โดย Admin  
เป็นผู้จัดการ แต่ละรางวัลจะมีรายละเอียดเกี่ยวกับรางวัล และจำนวนแอดัมที่ต้องใช้แลก

### Admin

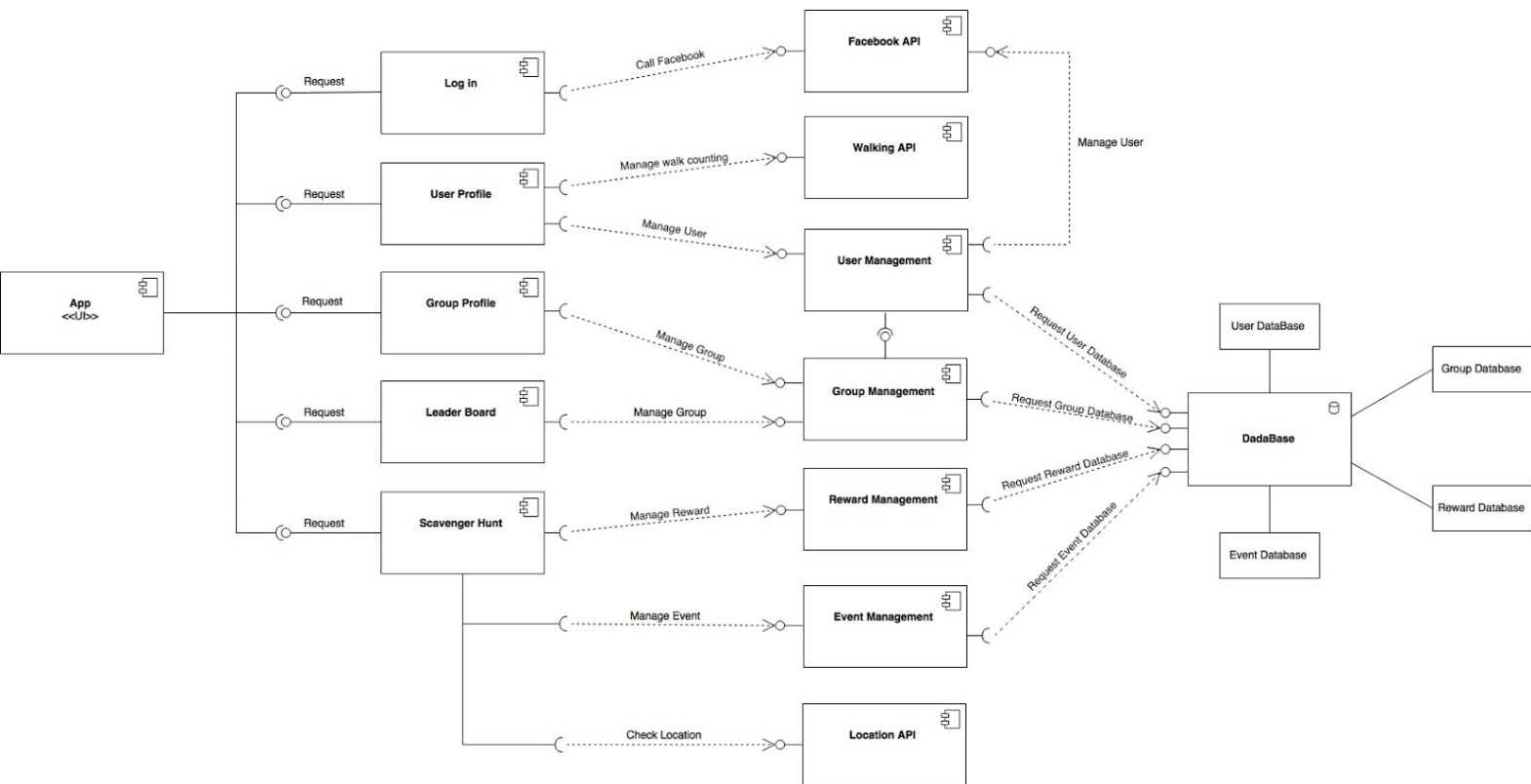
ผู้ที่มีสิทธิ์ในการแก้ไขเปลี่ยนแปลงและจัดการเกี่ยวกับกิจกรรมและของรางวัลในระบบ

## Application Architecture

Walk-Ranger Application นั้น ถูกออกแบบให้เป็น application ที่ทำการบันทึกการเดิน การก้าว เพื่อเก็บเป็นสถิติของแต่ละผู้ใช้ โดยนำค่าที่นับไปเฉลี่ยกับคนอื่นๆในทีม แล้วนำเอาค่าเฉลี่ยไปจัดอันดับ แข่งขันกับกลุ่มอื่นๆ อีกทั้งมีกิจกรรม Scavenger hunt ที่เป็นกิจกรรมไปสถานที่ต่างๆ ได้รับแต้ม มาแลกของรางวัล เป็นส่วนเสริมเพื่อให้ตัว Application มีความน่าสนใจมากขึ้น จากการเป็นระบบ ที่มีการทำงานหลายอย่างบนระบบ การทำงานจึงควรแบ่งกันทำ ทั้งฝั่งของ Client และ Server เพื่อป้องกันการเกิดปัญหาของระบบและข้อมูล ดังนั้นจึงเลือกสถาปัตยกรรมแบบ Client-Server Architecture โดยแบ่งการทำงานในส่วนของการแปลง code นำมาแสดงผลบน andriod device (Presentation logic) และการประมวลผลต่างๆ ที่เกี่ยวกับนับก้าว (Application logic) จะอยู่ทางฝั่งของ Client ส่วนการประมวลผลในการจัดอันดับค่าเฉลี่ยของกลุ่ม (Application logic), การบันทึกเรียกใช้ข้อมูลลง data base (Data access logic) และเป็นที่เก็บข้อมูลจริงๆ (Data Storage) จะอยู่ทางฝั่งของ Server



สถาปัตยกรรมแบบ Client-Server



component diagram

## รายละเอียดของ Architecture

ระบบนี้ ประกอบด้วย 2 ส่วน คือส่วนของ Client และ Server

Client จะเป็นส่วนของ Android Application ประกอบด้วย

- Application UI

เป็นส่วนที่ทำหน้าที่เป็น Input หรือ Output เพื่อติดต่อ สื่อสาร กับผู้ใช้  
ส่วนนี้ จะติดต่อกับ Component ที่สำคัญอื่นๆ ที่อยู่ในฝั่ง Client ด้วยกัน

- Log in

เป็น Component เกี่ยวกับการเข้าสู่ระบบ โดยมีการเชื่อมต่อกับ Facebook API  
ในการดึงข้อมูลจาก Facebook

- User Profile

เป็น Component เกี่ยวกับข้อมูลของ User มีการติดต่อกับ User List ในฝั่งของ  
Server และติดต่อกับ Walking API เพื่อบันทึกจำนวนก้าวของ User

- Group Profile

เป็น Component เกี่ยวกับข้อมูลของกลุ่มต่างๆ ติดต่อกับ Group List ในฝั่ง Server  
เพื่อปรับปรุง แก้ไข เพิ่ม ลบ หรือดึงข้อมูล

- Leader Board

เป็น Component เกี่ยวกับบอร์ดที่ใช้แสดงอันดับและคะแนนของแต่ละกลุ่ม  
ติดต่อกับ Group List ในฝั่ง Server เพื่อดึงข้อมูล และจัดอันดับ เพื่อนำมาแสดงบนบอร์ด

- Scavenger hunt

เป็น Component เกี่ยวกับกิจกรรมพิเศษ “Scavenger Hunt” ซึ่งจะมีรายละเอียดกิจกรรมย่อยๆให้ผู้ใช้ที่สนใจได้เข้าร่วม มีระบบเข้าร่วมกิจกรรม และระบบแลกเปลี่ยนของรางวัลจากแต้มที่ได้รับเมื่อทำกิจกรรมสำเร็จ

มีการเชื่อมต่อกับ Event List และ Reward List ในฝั่ง Server เพื่อจัดการข้อมูล

และเชื่อมต่อกับ Location API เพื่อตรวจสอบตำแหน่งปัจจุบันของผู้ใช้

ในส่วนนี้ จะมีการใช้ API ที่ช่วยอำนวยความสะดวก และเป็นช่องทางการติดต่อกับระบบอื่นๆ

- Facebook API

เป็น API ที่ช่วยในการติดต่อ แลกเปลี่ยนข้อมูล กับระบบ Facebook

- Walking API

เป็น API ที่ใช้ในการนับหรือคำนวณ จำนวนก้าว ที่ผู้ใช้สมาร์ตโฟนนั้นๆได้เดิน

- Location API

เป็น API ที่ใช้ตรวจสอบสถานที่ปัจจุบันของสมาร์ตโฟนโดย Component

ในฝั่งของ Client ทั้งหมดกันผ่านระบบ และความสัมพันธ์ที่เกิดขึ้น ใน Android Application

และจะติดต่อกับ Component ฝั่ง Server ผ่านทาง Web Service

Server เป็นส่วนที่ประกอบด้วย Web service ที่ทำการประมวลผลข้อมูล และ Database ที่ทำหน้าที่เก็บข้อมูล

- User List

เป็น Component ที่ทำหน้าที่จัดการและประมวลผลข้อมูลเกี่ยวกับ User

มีการติดต่อกับ Facebook API และ User Profile ในฝั่ง Client

และติดต่อกับ Database เพื่อจัดการข้อมูลของแต่ละ User ที่เก็บใน Database

- Group List

เป็น Component ที่ทำหน้าที่จัดการและประมวลผลข้อมูลเกี่ยวกับ Group ต่างๆ

ติดต่อกับ Group Profile และ Leader Board ในฝั่ง Client

และติดต่อกับ Database เพื่อจัดการข้อมูลของ Group ใน Database

- Reward List

เป็น Component ที่ทำหน้าที่จัดการและประมวลผลข้อมูลเกี่ยวกับ Reward

ติดต่อกับ Scavenger Hunt ในฝั่ง Client

ติดต่อกับ Database เพื่อจัดการข้อมูล Reward ใน Database

- Event List

เป็น Component ที่ทำหน้าที่จัดการและประมวลผลข้อมูลเกี่ยวกับ Event

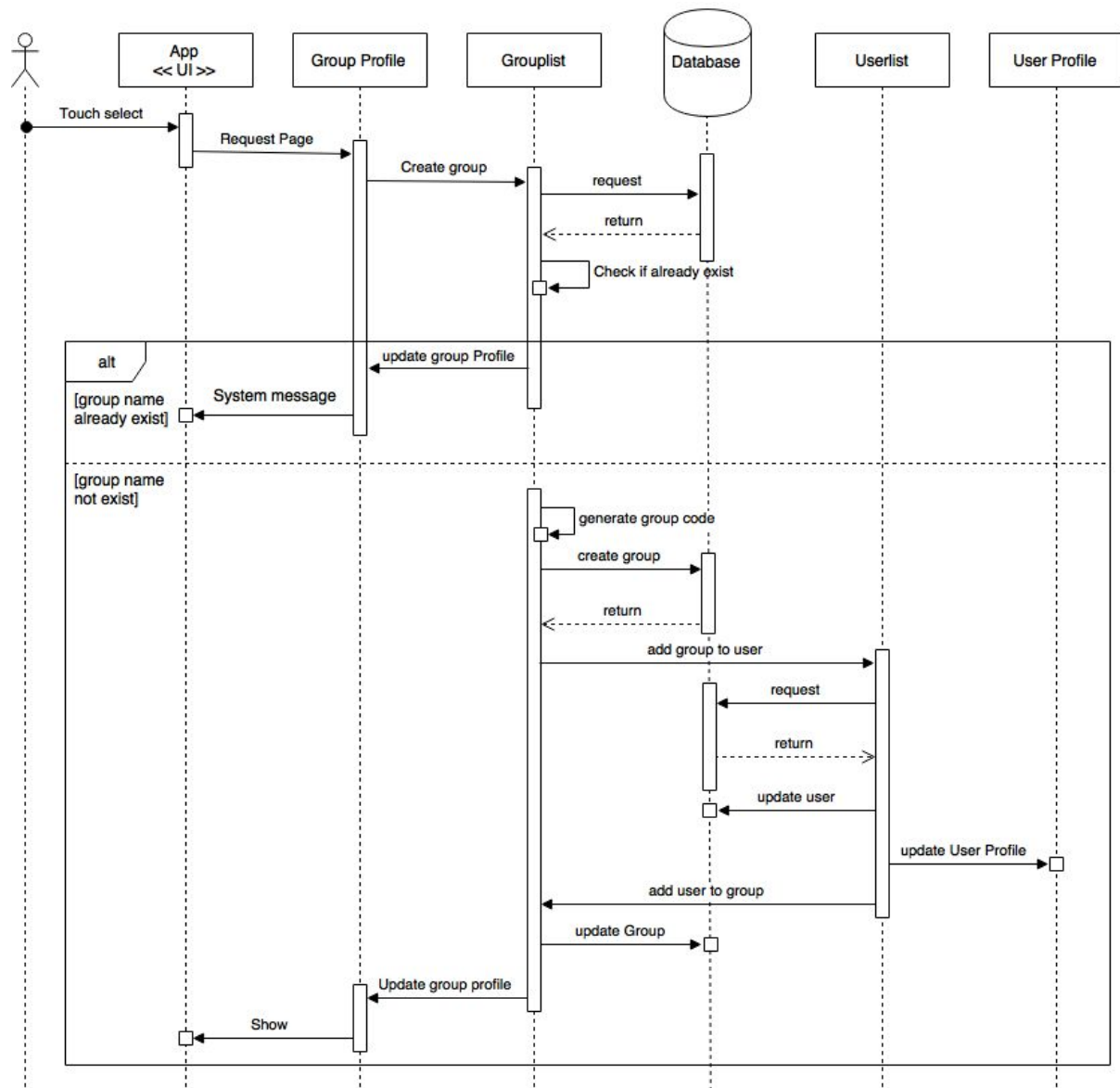
ติดต่อกับ Scavenger Hunt ในฝั่ง Client

ติดต่อกับ Database เพื่อจัดการข้อมูล Event ใน Database

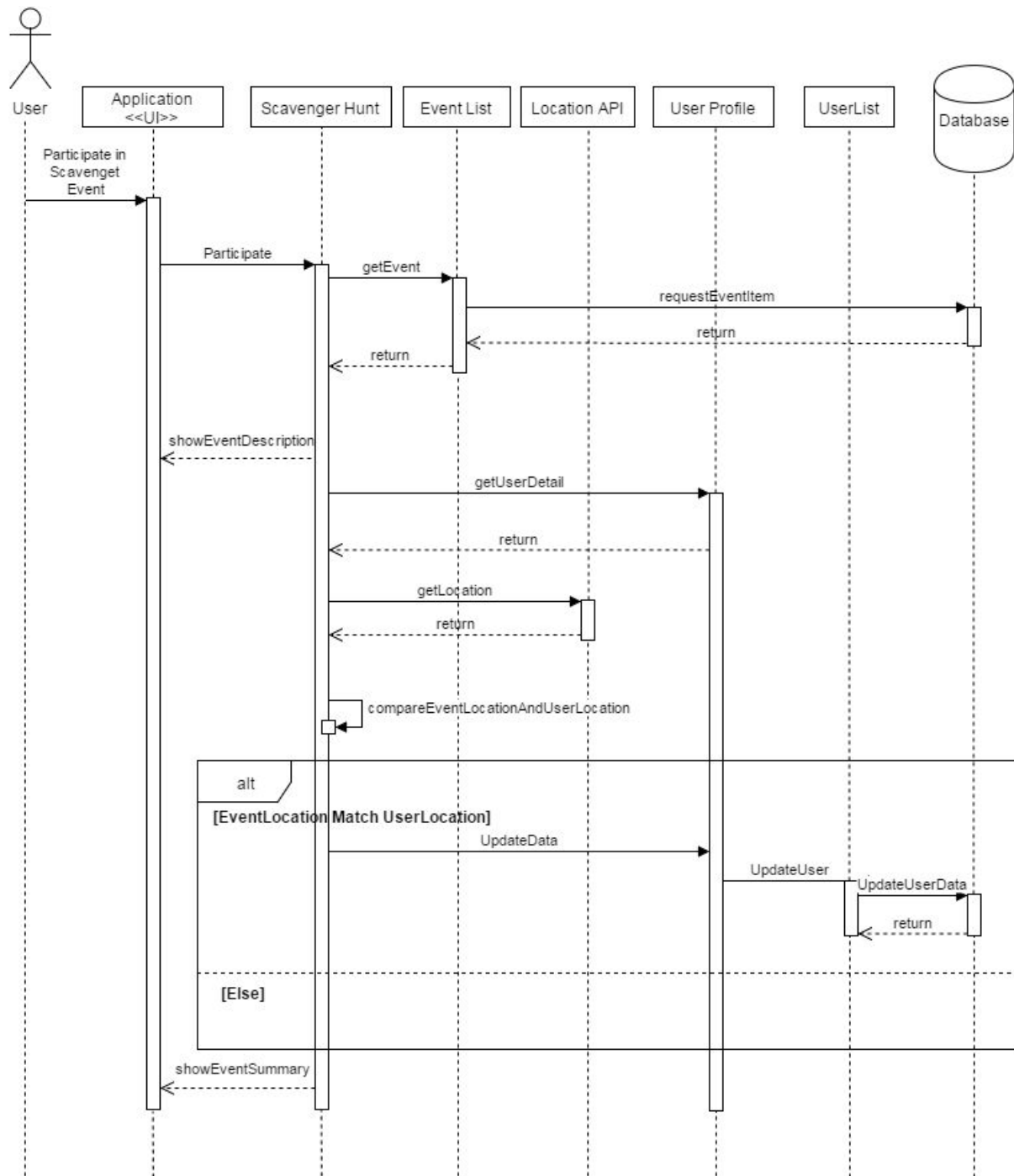
- Database

ติดต่อกับ Web service ต่างๆ ข้างต้น และมีการเก็บข้อมูล User Database, Group Database, Event Database และ Reward Database

## component sequence diagram



component sequence diagram 101 join group



component sequence diagram ของ participate scavengers

## หน้าที่ของ Class ต่างๆ

### ฝั่ง Clients

#### - Walk-Ranger

เป็นคลาสที่ควบคุมการทำงานและการเชื่อมต่อของคลาสทั้งหมด โดยควบคุมการติดต่อสื่อสาร/แลกเปลี่ยนข้อมูล ระหว่างฝั่ง Clients-Server และควบคุมการทำงานและการเชื่อมต่อของคลาสในฝั่ง Client

#### - FacebookData

เป็นคลาสที่ทำหน้าที่จัดการ การติดต่อสื่อสาร กับ Facebook โดยสามารถดึงข้อมูลพื้นฐานของผู้ใช้จากบัญชี Facebook ได้และมีความสามารถในการ แชร่ ข้อมูลไปยังบัญชี Facebook ได้

#### - User

เป็นคลาสที่เก็บข้อมูลส่วนตัวของผู้ใช้แต่ละคน โดยนำข้อมูลพื้นฐานมาจากบัญชี Facebook ผู้ใช้สามารถแก้ไขหรือเปลี่ยนแปลงข้อมูลพื้นฐานของตนเองได้

#### - Group

เป็นคลาสที่รวม User หลายๆ User อยู่ด้วยกันเพื่อให้เกิดการแข่งขันระหว่างกลุ่ม โดยจะมีข้อมูลต่างๆ ของ Group เช่น ชื่อกลุ่ม, สมาชิกของกลุ่ม, คะแนนรวมทั้งหมดของกลุ่ม, คะแนนรวมประจำสัปดาห์ของกลุ่ม, คะแนนรวมประจำเดือนของกลุ่ม โดยมีความสามารถในการจัดการสมาชิกในกลุ่ม,เปลี่ยนแปลงข้อมูลพื้นฐานของกลุ่ม เป็นต้น

#### - LeaderBoard

เป็นคลาสที่ทำหน้าที่แสดงอันดับและคะแนนของแต่ละกลุ่ม โดยจะมีการจัดอันดับแบบประจำสัปดาห์ และแบบประจำเดือน

#### - ScavengerFacade

เป็นคลาสที่ทำหน้าที่ควบคุมกิจกรรมเกี่ยวกับกิจกรรม Scavenger Hunt ซึ่งมี 2 ส่วน คือส่วนของกิจกรรม และส่วนของของรางวัล

#### - EventMenu

เป็นคลาสที่เก็บรายการกิจกรรมทั้งหมดที่มีเอาไว้

#### - RewardMenu

เป็นคลาสที่เก็บรายการของรางวัลทั้งหมดที่มีเอาไว้

#### - Event

เป็นคลาสที่เก็บข้อมูลเกี่ยวกับกิจกรรมที่สามารถเข้าร่วมได้ โดยมีข้อมูลต่างๆ เช่น พื้นที่ของกิจกรรม, เงื่อนไขของกิจกรรม

#### - Reward

เป็นคลาสที่เก็บข้อมูลเกี่ยวกับของรางวัลที่สามารถแลกเปลี่ยนได้ โดยมีข้อมูลต่างๆ เช่น จำนวนแต้มที่ต้องใช้, คำอธิบายของรางวัล, จำนวนที่เหลืออยู่



## ฝั่ง Server

- DataFacade

เป็นคลาสที่ควบคุมเส้นทางการติดต่อ และการแลกเปลี่ยนข้อมูลต่างๆ ที่รับ-ส่ง ระหว่างฝั่ง Client กับคลาสต่างๆในฝั่ง Server

- UserManagement

เป็นคลาสที่ทำหน้าที่ประมวลผลข้อมูลของบัญชีผู้ใช้ และสามารถ เพิ่ม/ลบ/เปลี่ยนแปลง ข้อมูลบัญชีผู้ใช้ ที่อยู่ใน Database ได้

- User

เป็นคลาสที่ทำหน้าที่เก็บข้อมูลของบัญชีผู้ใช้

- GroupManagement

เป็นคลาสที่ทำหน้าที่ประมวลผลข้อมูลของกลุ่มต่างๆ และสามารถ เพิ่ม/ลบ/เปลี่ยนแปลง ข้อมูลกลุ่ม ที่อยู่ใน Database ได้

- Group

เป็นคลาสที่ทำหน้าที่เก็บข้อมูลของกลุ่ม

- EventManagement

เป็นคลาสที่ทำหน้าที่ประมวลผลข้อมูลของกิจกรรมต่างๆ และสามารถ เพิ่ม/ลบ/เปลี่ยนแปลง ข้อมูลกิจกรรม ที่อยู่ใน Database ได้

- Event

เป็นคลาสที่ทำหน้าที่เก็บข้อมูลของกิจกรรม

- RewardManagement

เป็นคลาสที่ทำหน้าที่ประมวลผลข้อมูลของรางวัลต่างๆ และสามารถ เพิ่ม/ลบ/เปลี่ยนแปลง ข้อมูลของรางวัล ที่อยู่ใน Database ได้

- Reward

เป็นคลาสที่ทำหน้าที่เก็บข้อมูลของรางวัล

- LeaderBoardRanking

เป็นคลาสที่จะนำข้อมูลกลุ่มต่างๆ มาจัดอันดับ ประมวลผล แล้วส่งกลับไปยังฝั่ง Client

**\*\*แก้ component เล็กน้อย(เปลี่ยนชื่อ)/ ทำ class diagram เพิ่ม ฝั่ง server /ตรวจสอบคำอธิบายทั้งหมด**

## Interaction diagram

