



Object-oriented Analysis and Design, Fall 2015  
Department of Computer Engineering  
Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang

# Pin up

## Co-ordination

1. นางสาวกุลยา	วิสุทธิ์เทพ	56010100	sec1
2. นางสาวชนกานต์	ชอบรส	56010242	sec1
3. นางสาวชวลี	เจริญพรพรมาม	56010280	sec1
4. นายชัชวาลล์	ภิกพสุขาวดี	56010285	sec1
5. นายธนต์	พีระไพศาลทรัพย์	56010564	sec1
6. นายนนทปวิธ	สามพวงทอง	56010631	sec1
7. นางสาวนริศรา	ทองเผือก	56010654	sec1
8. นายวิวิธชัย	เรืองชาติสมบัติ	56011148	sec1

# ANALYSIS

## Use Case

### 1. Pin

เมื่อผู้ใช้สนใจ **Topic** และต้องการจะจัดเก็บ **Topic** ไว้ในตารางกิจกรรมส่วนตัวสามารถทำได้โดยการกด **Pin** ในหน้า **Topic** นั้นๆ

### 2. Delete Pin

เมื่อผู้ใช้ต้องการลบ **Topic** ที่ถูก **Pin** ไว้ในตารางกิจกรรมส่วนตัว ผู้ใช้ต้องทำการกดวันที่ที่มี **Topic** นั้นอยู่ จากนั้นแอปพลิเคชันจะแสดง **Topic** ทั้งหมดที่ถูก **Pin** ไว้ที่เกิดขึ้นในวันนั้น เลือก **Topic** ที่ต้องการ แล้วกดลบระบบจะทำการนำ **Topic** นั้นออกจากตารางกิจกรรมส่วนตัวของผู้ใช้

### 3. Post

ผู้ใช้สามารถสร้าง **Topic** ได้ โดย **Topic** จะประกอบด้วย ชื่อเรื่อง วัน-เวลา และ สถานที่จัดกิจกรรม รายละเอียดกิจกรรม รูปภาพประกอบ

### 4. Delete Post

ผู้ใช้สามารถทำการลบ **Topic** ที่ตนเองสร้างขึ้นได้

### 5. Search

เมื่อผู้ใช้ต้องการค้นหา **Topic** จะสามารถค้นหาได้ 2 แบบ คือ ค้นหาตาม **Keyword** , ค้นหาตามประเภท ผู้ใช้สามารถเลือกรูปแบบการจัดเรียง ได้แก่ จัดเรียงตามตัวอักษร, จัดเรียงตามวันที่เกิดกิจกรรม, จัดเรียงตามระดับคะแนน, จัดเรียงตามวันที่ **post** กิจกรรม จากนั้นระบบจะทำการค้นหาและแสดง **Topic** ตามโหมดที่เลือก

### 6. Comment

ผู้ใช้สามารถแสดงความคิดเห็นใน **Topic** ต่างๆได้

### 7. Delete Comment

ผู้ใช้สามารถลบความคิดเห็นที่ได้แสดงไว้ใน **Topic** ต่างๆได้ โดยจะสามารถลบได้เฉพาะที่เป็นของตนเองเท่านั้น

## 8. Rating

เมื่อผู้ใช้งานต้องการให้คะแนน **Topic** ผู้ใช้กดเลือก **Topic** แล้วกดเลือกระดับคะแนนที่ต้องการ จากนั้นกดตกลง คะแนนที่ผู้ใช้ให้จะนำไปประมวลผลระดับความน่าสนใจของ **Topic** ต่อไป

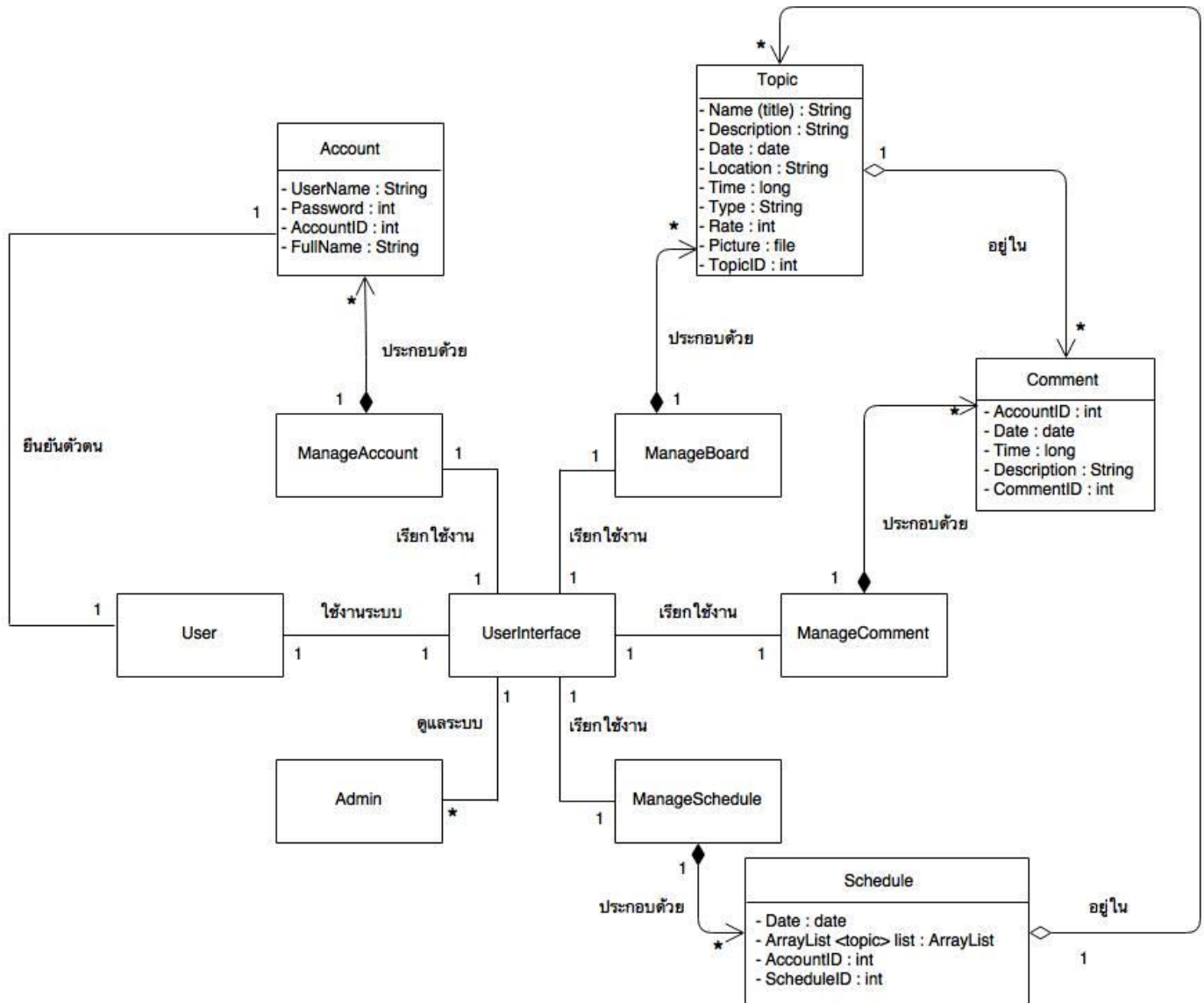
## 9. Create Account

เมื่อผู้ใช้งานเริ่มใช้งานแอปพลิเคชันครั้งแรกผู้ใช้งานต้องกรอกข้อมูลที่จำเป็นแล้วกดตกลงเพื่อสร้างบัญชีสำหรับการใช้งานในแอปพลิเคชัน **Pin Up**

## 10. Delete Account

เมื่อ **Admin** ตรวจพบบัญชีผู้ใช้งานที่ไม่เหมาะสม **Admin** สามารถเลือกลบบัญชีผู้ใช้งานจากการจัดการบัญชีผู้ใช้งานของ **Admin** ได้

## Domain Model



## Abstraction

- 1. User** คือ ผู้ใช้งานโปรแกรม มีความต้องการที่จะใช้งานฟังก์ชันต่างๆในโปรแกรม เช่น ต้องการการแจ้งเตือนจาก Topic ที่ปักหมุดไว้
- 2. Account** คือ class interface เพื่อเป็นการยืนยันผู้ใช้งาน โดยผู้ใช้งานต้องสมัครaccount เพื่อยืนยันการเข้าระบบ (log in) ซึ่งประกอบด้วย ชื่อบัญชีผู้ใช้, รหัสผ่าน, IDบัญชีผู้ใช้, ชื่อ-นามสกุลผู้ใช้
- 3. ManageAccount** คือ การจัดการข้อมูลของ account ทุกๆ account จะมีฟังก์ชันการจัดการกับ account ต่างๆ เช่น createAccount() เป็นต้น มีความสัมพันธ์กับ class account แบบ composition เนื่องจากในManageAccount จะประกอบด้วย account หลายๆaccount

4. Topic เป็น class interface เพื่อเป็นรูปแบบสำหรับการสร้าง topic ขึ้น โดยจะประกอบด้วย ชื่อ Topic, รายละเอียด, วันที่จัดกิจกรรม, สถานที่ที่จัดกิจกรรม, เวลาในการจัดงาน, ชนิดของ Topic, คะแนนความนิยม, รูปภาพ

5. ManageBoard คือการจัดการข้อมูลของ Topic ซึ่งจะรวบรวม Topic ทั้งหมด มาอยู่ในกระดาน ซึ่งจะมีความสัมพันธ์ทั้งสองคลาสเป็น composition โดย ManageBoard จะมีส่วนประกอบเป็น Topic หลายๆ Topic รวมกัน และมีฟังก์ชันการจัดการ Topic เช่น SearchByType() เป็นต้น

6. Comment การแสดงความคิดเห็นซึ่งจะอยู่ใน Topic ต่างๆ หากเกิดการแสดงความคิดเห็นเกิดขึ้น Comment มีความสัมพันธ์กับ Topic แบบ Aggregation คือ comment เป็นส่วนประกอบของ Topic ใน Topic จะมีการเรียงลำดับของ comment ตามวันที่ที่เกิดการแสดงความคิดเห็น ซึ่ง class Comment จะประกอบด้วย IDบัญชีผู้ใช้, วันที่ comment, เวลาที่ comment, ความคิดเห็น

7. ManageComment คือการจัดการข้อมูลของ Comment ที่จะเชื่อมกับแต่ละ Topic มีฟังก์ชันที่ใช้ในการจัดการ Comment ต่างๆ เช่น addComment() deleteComment() เป็นต้น class ManageComment มีความสัมพันธ์กับ class Comment แบบ composition คือ class Comment เป็นส่วนประกอบของ class ManageComment

8. Schedule เป็นปฏิทินจนสำหรับการปักหมุดกิจกรรม ในแต่ละวัน ประกอบด้วย วันที่ถูกปักหมุด, รายการของ Topic ที่จะปักหมุด, ID บัญชีผู้ใช้

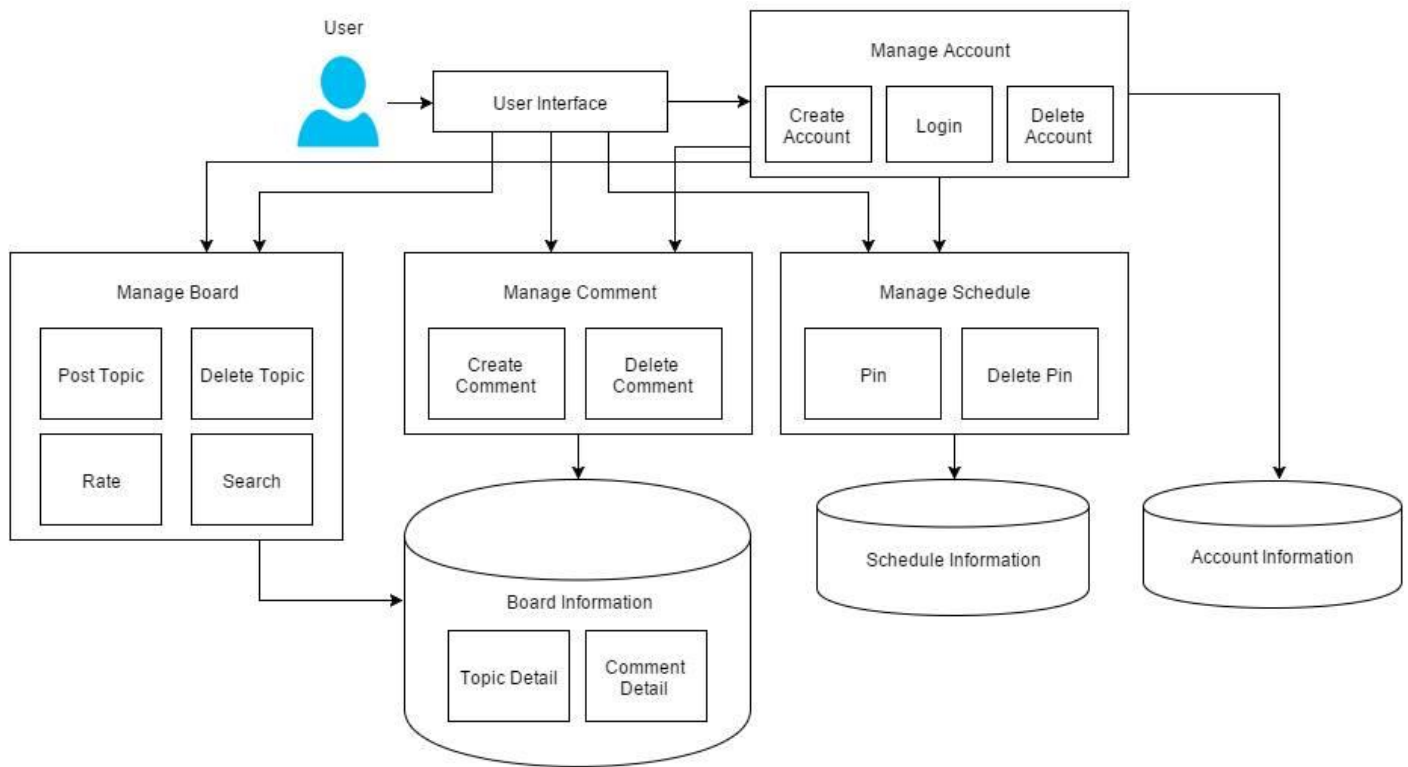
9. ManageSchedule คือการจัดการข้อมูลของ Schedule มีความสัมพันธ์กับ Schedule แบบ composition คือ schedule เป็นส่วนประกอบของ ManageSchedule มีฟังก์ชันในการจัดการเกี่ยวกับข้อมูลต่างๆ เช่น Pin() เป็นต้น

10. Admin คือบุคคลที่ดูแลระบบ คอยดูแลความเรียบร้อยและความปลอดภัยโดยทั่วไปของระบบทั้งหมด สิทธิที่ Admin จะทำได้เช่น ลบ Topic ที่เข้าข่ายอนาจาร หรือผิดกฎหมาย เป็นต้น

11. UserInterface รวมฟังก์ชันการใช้งานระบบต่างๆ ให้ user เลือกใช้ เช่น createAccount(), post(), pin() เป็นต้น

# DESIGN

## Application Architecture



### อธิบาย Architecture

1. **Manage Account** ทำหน้าที่เกี่ยวกับการจัดการบัญชีผู้ใช้ มีการติดต่อกับ **Account Information** (ฐานเก็บข้อมูล Account)
2. **Account Information** เก็บข้อมูลของบัญชีผู้ใช้งานทุกคนประกอบด้วย **Username, Password, ชื่อ-นามสกุล, Account ID** มีการติดต่อกับ **Manage Account**
3. **Manage Schedule** ทำหน้าที่เกี่ยวกับการจัดการตารางกิจกรรมส่วนตัวของผู้ใช้มีการติดต่อ **Schedule Information (เก็บข้อมูล Schedule) ,Manage Account ,User Interface**
4. **Schedule Information** เก็บข้อมูลตารางกิจกรรมส่วนตัวของแต่ละ **Account** มีการติดต่อกับ **manage schedule**
5. **Manage Comment** ทำหน้าที่จัดการ **Comment – Create ,Delete** มีการติดต่อกับ **Board Information, User Interface และ Manage Account**

6. **Board Information** ทำหน้าที่เก็บข้อมูลTopicและCommentของแต่ละTopic มีการติดต่อกับ Manage Comment และ Manage Board
7. **Manage Board** ทำหน้าที่จัดการเกี่ยวกับ Board – Post Topic, Delete Topic, Search และRate มีการติดต่อกับ Board Information, Manage Account และ User Interface
8. **User Interface** เป็นส่วนที่ติดต่อกับผู้ใช้งานฟังก์ชันในการใช้งานต่างๆให้เลือกใช้

## Subsystems / Components

### 1. ระบบจัดการผู้ใช้งาน (Manage Account)

- ส่วนสมัครสมาชิก(create Account) : จะจัดเก็บข้อมูลต่างๆของกาสมัครสมาชิกของผู้ใช้งาน ประกอบด้วย **username, password, ชื่อ-นามสกุล และ Account ID** เข้าสู่ฐานข้อมูล
- ส่วนการเข้าใช้ระบบ(Log-in) : เปรียบเทียบบัญชีผู้ใช้งานกับข้อมูลของผู้ใช้ภายในฐานข้อมูล เพื่อให้ผู้ใช้เข้าถึงข้อมูลของตนเองได้อย่างถูกต้อง
- ส่วนการลบบัญชีผู้ใช้(delete Account) : ลบข้อมูลบัญชีผู้ใช้งานจากฐานข้อมูล โดย **admin** จะตรวจสอบบัญชีผู้ใช้งานที่ไม่เหมาะสมแล้วทำการลบบัญชีผู้ใช้งานออกจากฐานข้อมูล หรือผู้ใช้งานสามารถลบบัญชีผู้ใช้งานของตนเองได้

### 2. ระบบจัดการตารางกิจกรรมส่วนตัว (Manage Schedule)

- ส่วนปักหมุดกิจกรรม (pin) : จัดเก็บ **topic** ที่ผู้ใช้สนใจลงในตารางกิจกรรมส่วนตัว(Schedule Information) ประกอบด้วย **วันที่, Topic ID และ Account ID**
- ส่วนลบปักหมุดกิจกรรม(delete Pin) : ทำการลบ **topic** ที่ผู้ใช้ต้องการยกเลิกจากตารางกิจกรรมส่วนตัว(Schedule Information)

### 3.ระบบจัดการการแสดงความคิดเห็น(Manage Comment)

- ส่วนสร้างความคิดเห็น(create Comment) : ทำการจัดเก็บความคิดเห็นที่ผู้ใช้แสดงความคิดเห็นต่อ**topic**นั้นๆ ประกอบด้วย **comment, Comment ID, Topic ID และ Account ID**
- ส่วนลบความคิดเห็น (delete Comment) : ทำการลบความคิดเห็นที่ผู้ใช้ต้องการจะลบ โดยต้องเป็นการแสดงความคิดเห็นของตนเองเท่านั้น

### 4.ระบบการจัดการ Board กิจกรรม (Manage Board)

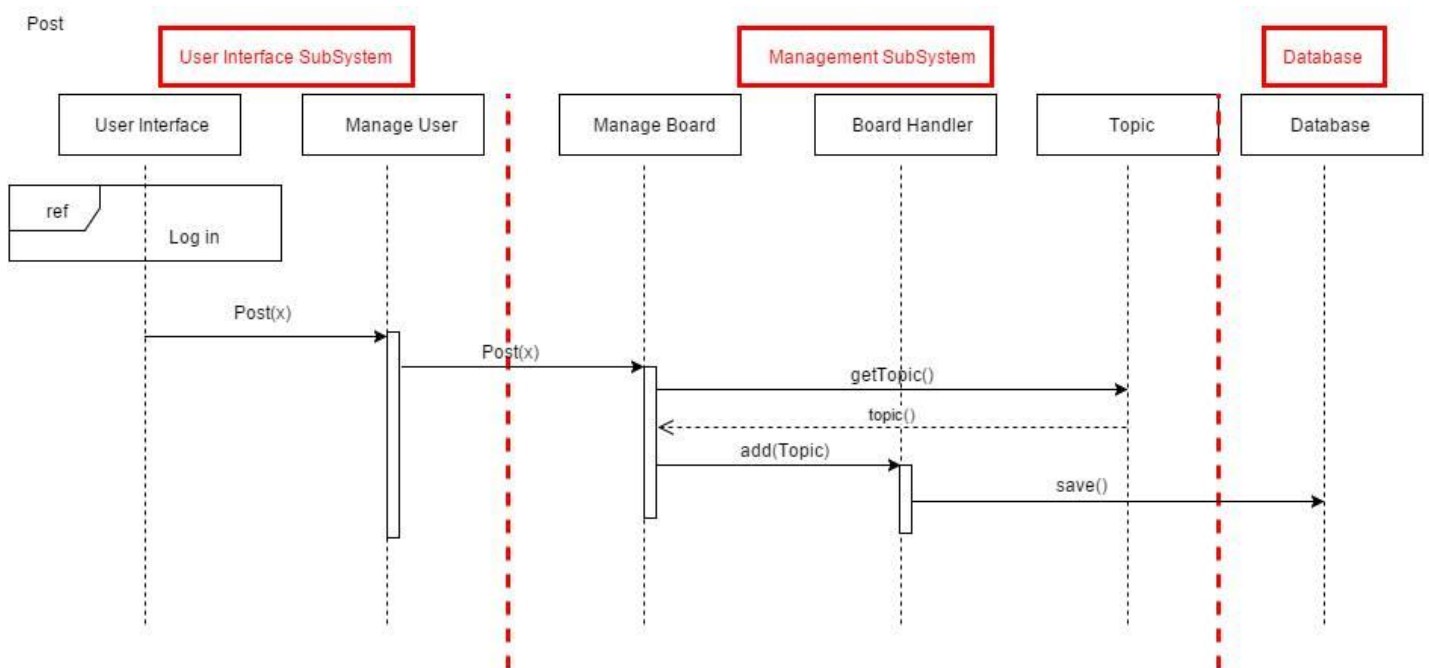
- ส่วนการสร้างกิจกรรม (Post Topic) : ทำการจัดเก็บข้อมูลรายละเอียดต่างๆเกี่ยวกับกิจกรรมที่ผู้ใช้สร้างขึ้นไว้ใน **board** ประกอบด้วย **รายละเอียดกิจกรรม, Account ID, Topic ID**
- ส่วนการลบกิจกรรม (delete Topic) : ทำการลบกิจกรรมที่ผู้ใช้แจ้งต่อ**Admin**ว่าต้องการจะลบออกจากฐานข้อมูล



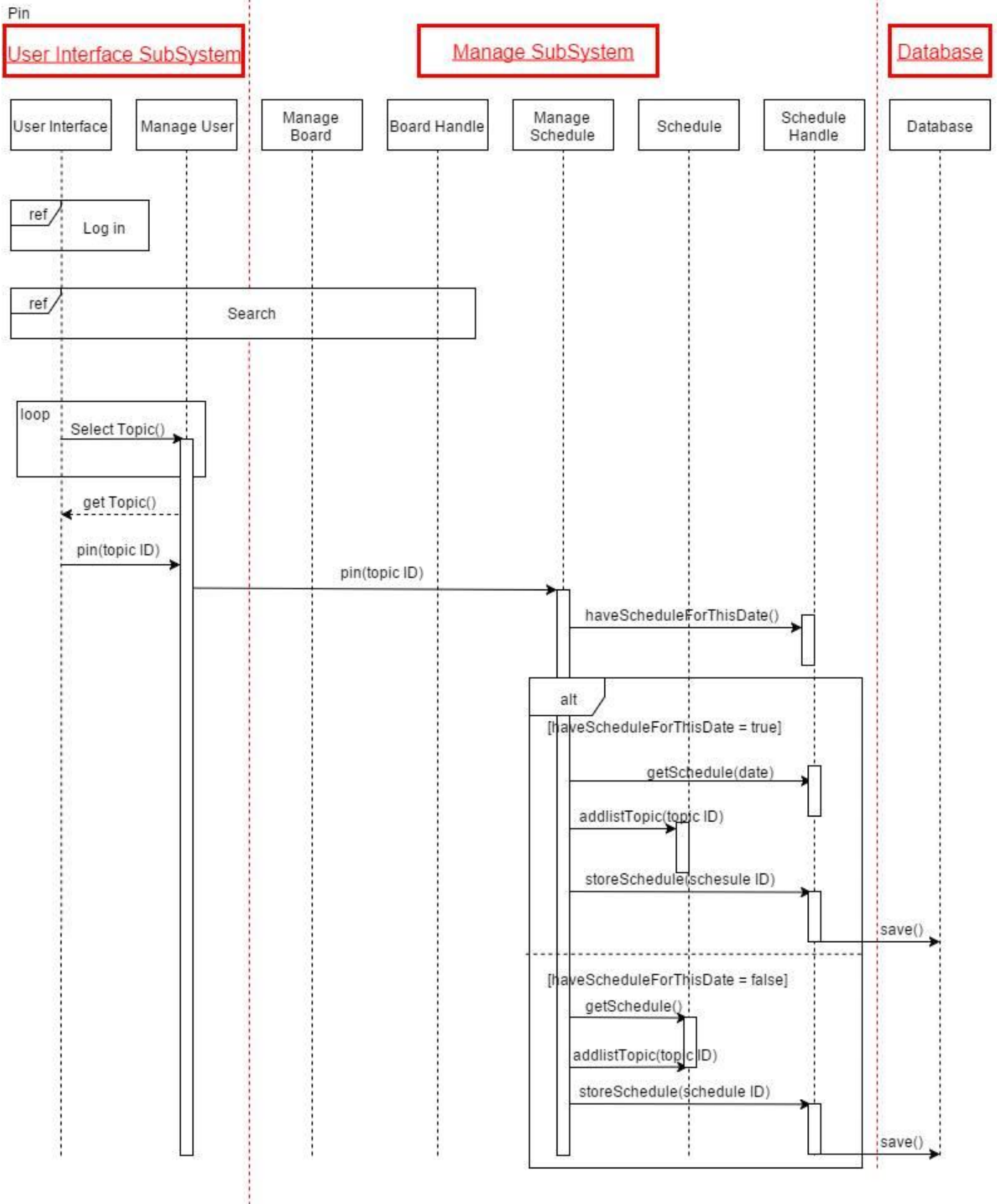
-ส่วนการให้คะแนนกิจกรรม (Rate) : ทำการจัดเก็บคะแนนที่ผู้ใช้ท่านอื่นที่ไม่ใช่เจ้าของกิจกรรมดังกล่าวให้ต่อกิจกรรมนั้นๆ

-ส่วนการค้นหากิจกรรม (Search) : ทำการค้นหา topic ตามที่ผู้ใช้ได้เลือก(เลือกได้สองวิธี คือ หาตามKeyword และหาตามType) และนำมาจัดเรียงตามที่ใช้กำหนด(จัดเรียงได้สี่วิธี คือ จัดเรียงตามตัวอักษร,จัดเรียงตามวันที่เกิดกิจกรรม, จัดเรียงตามระดับคะแนน, จัดเรียงตามวันที่ post กิจกรรม)

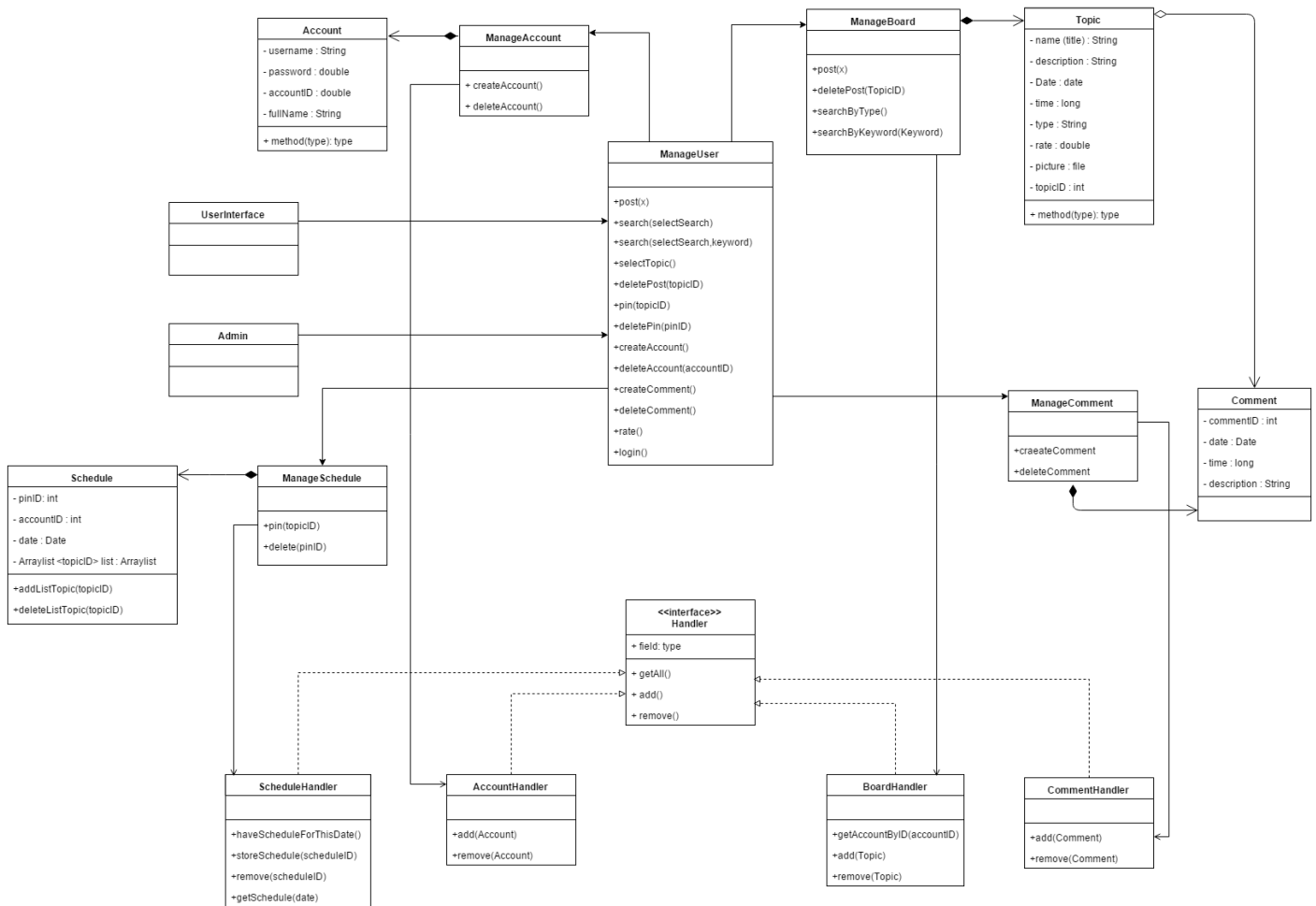
## Sequence Diagrams



x: Attribute ทั้งหมดของ post



# Design Diagrams

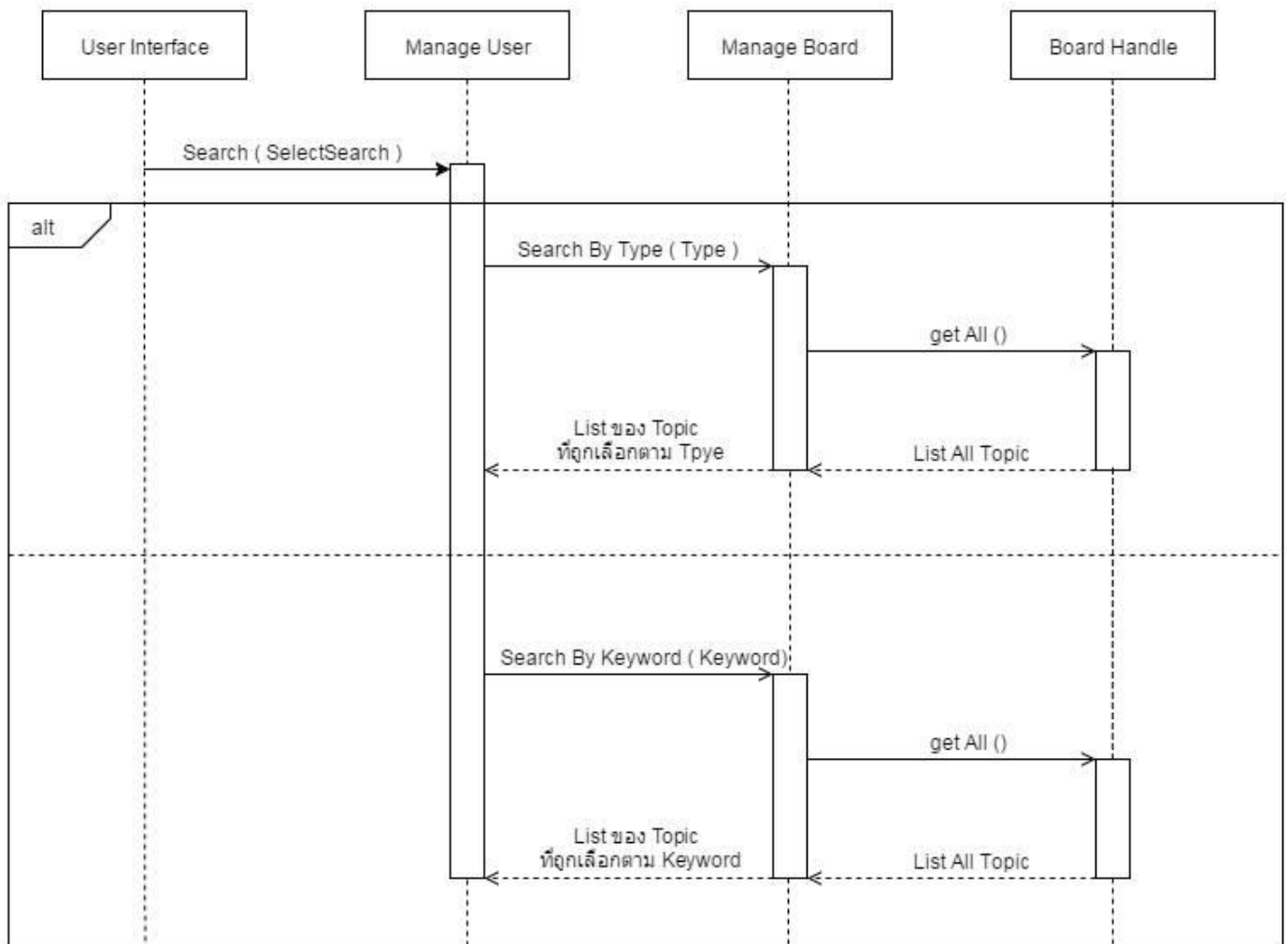


1. **UserInterface** เป็นส่วนติดต่อกับผู้ใช้งาน
2. **ManageUser** เป็นส่วนจัดการเกี่ยวกับฟังก์ชันทั้งหมดของระบบ เชื่อมต่อกับ **ManageAccount**, **ManageBoard**, **ManageSchedule**
3. **ManageAccount** มีฟังก์ชันที่ใช้ในการจัดการเกี่ยวกับบัญชีผู้ใช้ สามารถสร้าง และ ลบ บัญชีผู้ใช้ได้
4. **ManageBoard** มีฟังก์ชันที่ใช้ในการจัดการ **Topic** ทั้งหมด สามารถ **search** หา **Topic** ได้หลายวิธี เช่น search by name, search by type, search by rate, search by time และสามารถ **Post Topic** และลบ **Topic** ได้
5. **ManageSchudele** มีฟังก์ชันจัดการกับการปักหมุดทั้งหมด สามารถเพิ่มการปักหมุดกิจกรรม และลบการปักหมุดกิจกรรมได้ ในตารางกิจกรรมส่วนตัว
6. **ManageComment** มีฟังก์ชันจัดการเกี่ยวกับ **comment** ทั้งหมด สามารถเพิ่มความคิดเห็น และลบความคิดเห็นได้
7. **Account** เป็นคลาสที่ทำหน้าที่เก็บข้อมูลของบัญชีผู้ใช้ เช่น user name เป็นต้น

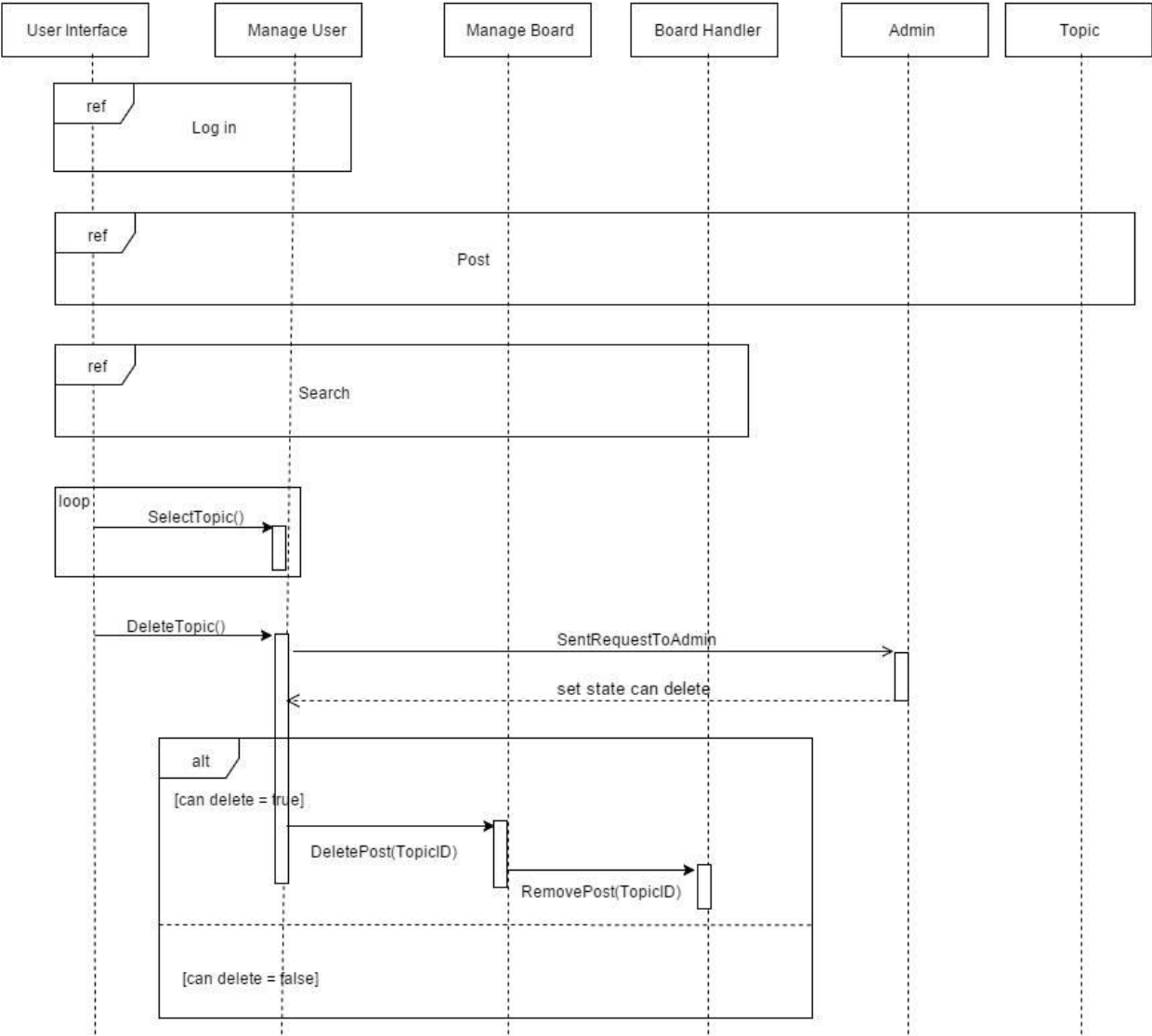
8. **Topic** เป็นคลาสที่ทำหน้าที่เก็บรายละเอียดข้อมูลของกิจกรรม
9. **Comment** เป็นคลาสที่ทำหน้าที่เก็บความคิดเห็น
10. **Schedule** เป็นคลาสที่ทำหน้าที่เก็บการปักหมุดกิจกรรมต่างๆ(ปักหมุด **Topic**) ลงในตารางกิจกรรมส่วนตัว
11. **Admin** ผู้ดูแลและจัดการระบบทั้งหมด
12. **Handler** เป็นคลาส interface มีmethod พื้นฐานเป็น `getAll()` , `add()`, `remove()`
13. **AccountHandler** เป็นคลาสที่ติดต่อกับ database ของข้อมูล **Account** ทั้งหมด สามารถดึงข้อมูล **Account** จาก database , เพิ่ม **Account** เข้าไป save ไว้ที่ database และการลบ **Account** ด้วย
14. **BoardHandler** เป็นคลาสที่ติดต่อกับ database ของข้อมูล **Topic** ทั้งหมด สามารถดึงข้อมูล **Topic** จาก database , เพิ่ม **Topic** เข้าไป save ไว้ที่ database และการลบ **Topic** ด้วย
15. **CommentHandler** เป็นคลาสที่ติดต่อกับ database ของข้อมูล **Comment** ทั้งหมด สามารถดึงข้อมูล **Comment** จาก database , เพิ่ม **Comment** เข้าไป save ไว้ที่ database และการลบ **Comment** ด้วย
16. **ScheduleHandler** เป็นคลาสที่ติดต่อกับ database ของข้อมูล **Schedule** ทั้งหมด สามารถดึงข้อมูล **Schedule** จาก database , เพิ่ม **Schedule** เข้าไป save ไว้ที่ database และการลบ **Schedule** ด้วย

## Interaction Diagram

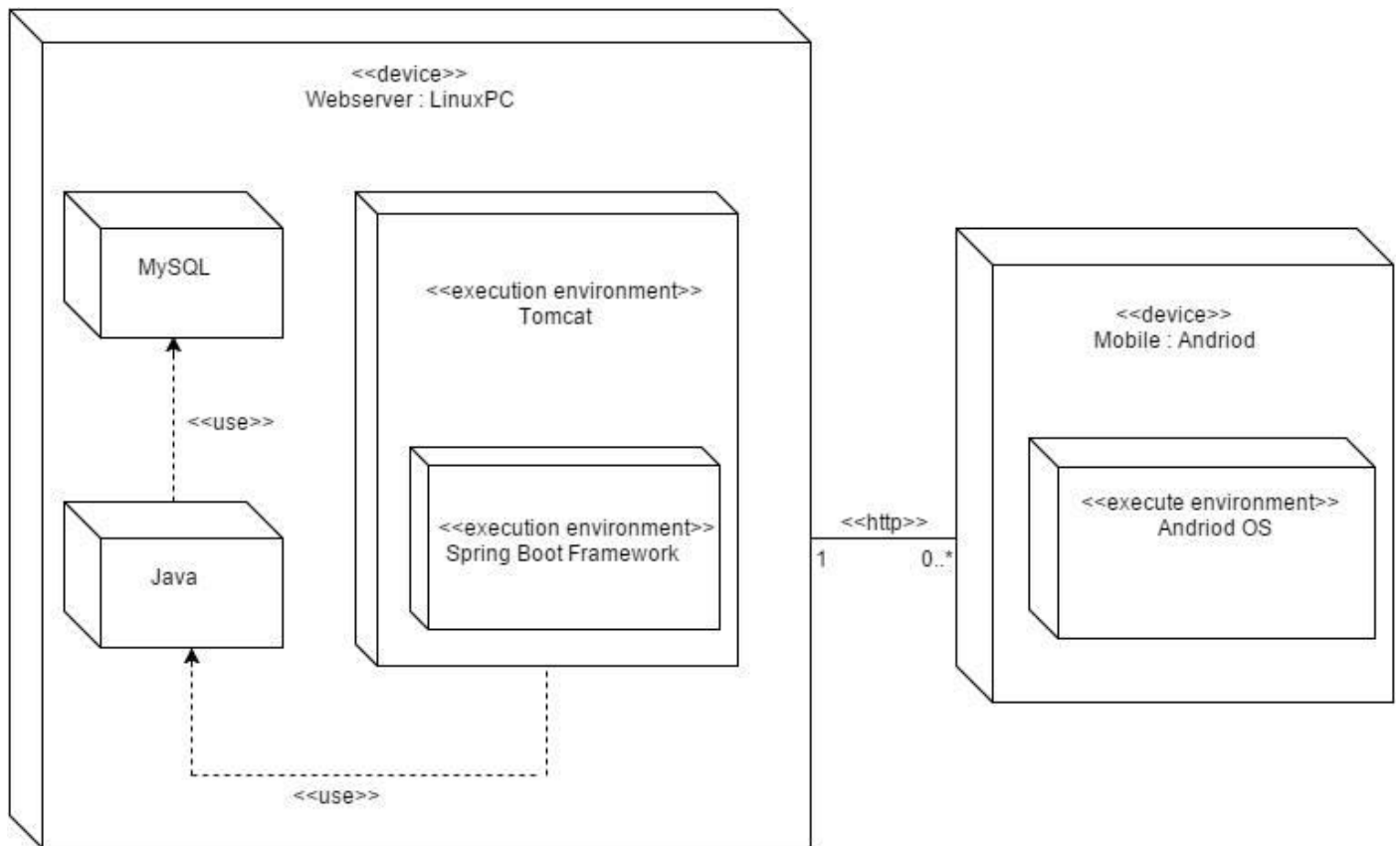
Search



Delete Post



# Deployment Diagram



# IMPLEMENTATION PLANE

	Front and back ends
	Front end
	Back end

Task name		August																															September																																		
		15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	1	2	3	4															
เตรียมการ																																																																			
แบ่งการทำงานในทีม	1 day																																																																		
วางแผนการทำงาน	3 days																																																																		
เลือกหัวข้อ	4 days																																																																		
ศึกษาเครื่องมือและภาษาที่เกี่ยวข้อง	3 weeks																																																																		
ออกแบบ (Design)																																																																			
กำหนด Requirements และ Use Cases	1 week																																																																		
วิเคราะห์ Domain Model	2 days																																																																		
ออกแบบ Architecture	3 days																																																																		
ออกแบบ UI	7 days																																																																		
ออกแบบ Database	2 weeks																																																																		
ออกแบบ logo app.	7 days																																																																		

[illegible]