

REVIEWVERSE

เว็บไซต์สำหรับจัดการและรีวิวสื่อบันเทิง



BOOK

ສມາຜິກໃນກລ່າມ

- ▶ ບາງສາວຄັນຈຸວດີ ທາລີ້ຫາຕີ 663380013-4
- ▶ ບາຍຄຣັນຍພງສ໌ ສາບຸສົມບົຍ 663380237-2
- ▶ ບາຍປີຍະວັຕນໍ ສວັສກຸລ 663380218-6
- ▶ ບາຍກົດຕິທັຍ ວຣນພຖກເງ 663380200-5
- ▶ ບາຍນິຕິຮຣ ຄຊຣຕັນ 663380215-2



ที่มาและความสำคัญ

ในยุคดิจิทัล ผู้คนมักพึ่งพาบริวเวอร์จากผู้ใช้งานจริงก่อนตัดสินใจเลือกซื้อหนังสือหรือสื่อบันเทิง แต่ระบบบริวเวอร์ปัจจุบันกระฉับกระเฉย ทำให้ค้นหาและเปลี่ยนความคิดเห็นได้ยาก

จึงได้พัฒนา ReviewVerse เว็บไซต์สำหรับรีวิวหนังสือและสื่อบันเทิง ผู้ใช้สามารถให้คะแนน
แสดงความคิดเห็น และค้นหาสื่ออย่างสะดวก พร้อมระบบจัดหมวดหมู่และบริหารจัดการผู้ใช้

สร้างพื้นที่อ่อนไลน์สำหรับแบ่งปันความคิดเห็น และเปลี่ยนมุมมอง พร้อมช่วยผู้ใช้ตัดสินใจเลือก
สื่อด้วยอย่างมีประสิทธิภาพ

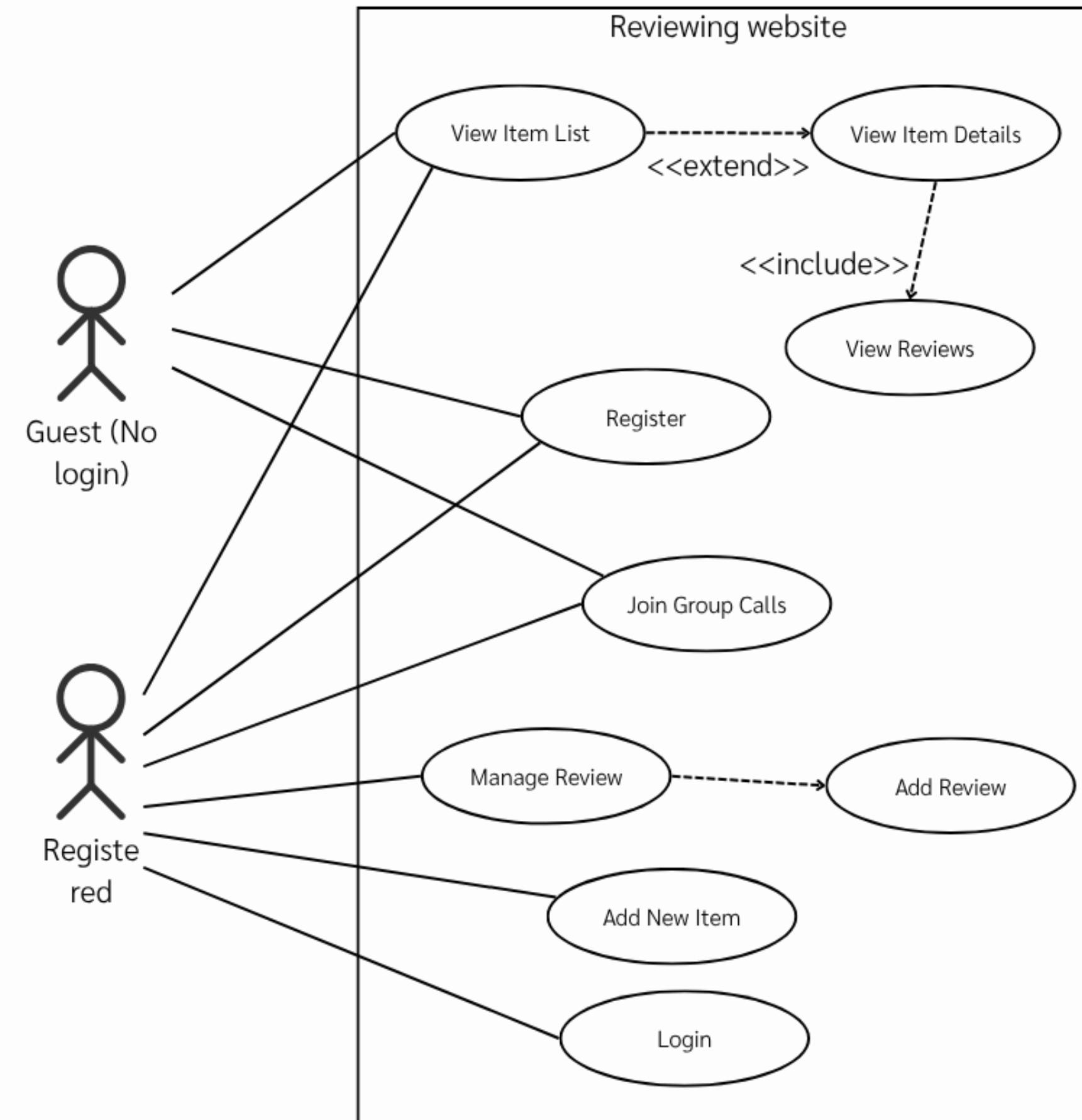
ວັດຖຸປະສົງຄໍ

ພັນນາແພລຕິໂວຣົມເວີບໄຊຕົກ໌ເປັນຄູນຍົວມືອງຂ້ອມູລ ແລະ ຮົວວໍາສໍາຮັບສ້ວບັນເຖິງ 4 ປະເກດ

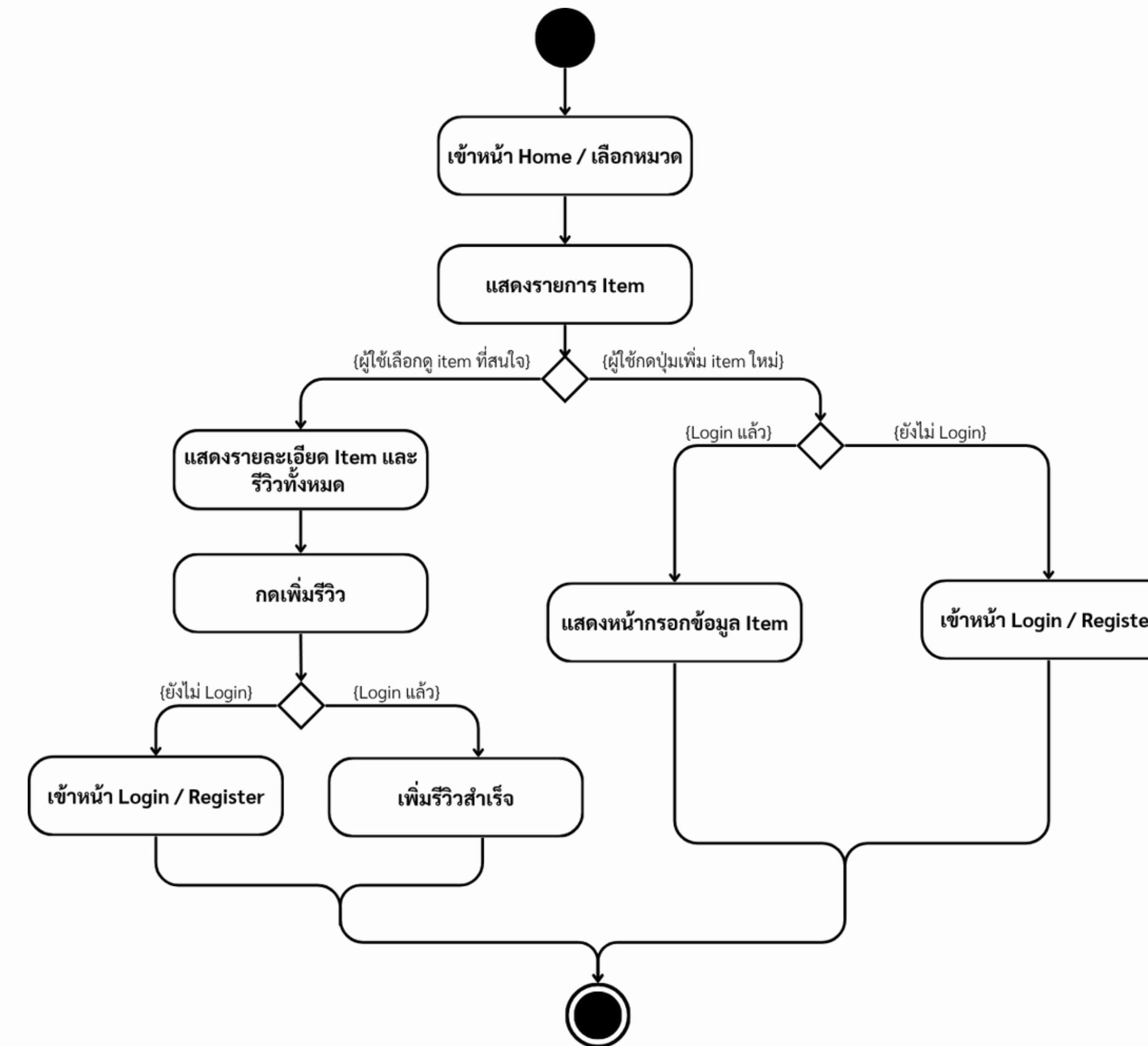
- ຜັນສື່ວ
- ກາພຍນຕົກ
- ກາຮຕູນ (ອົນເມະ/ມັງຈະ)
- ເກມ



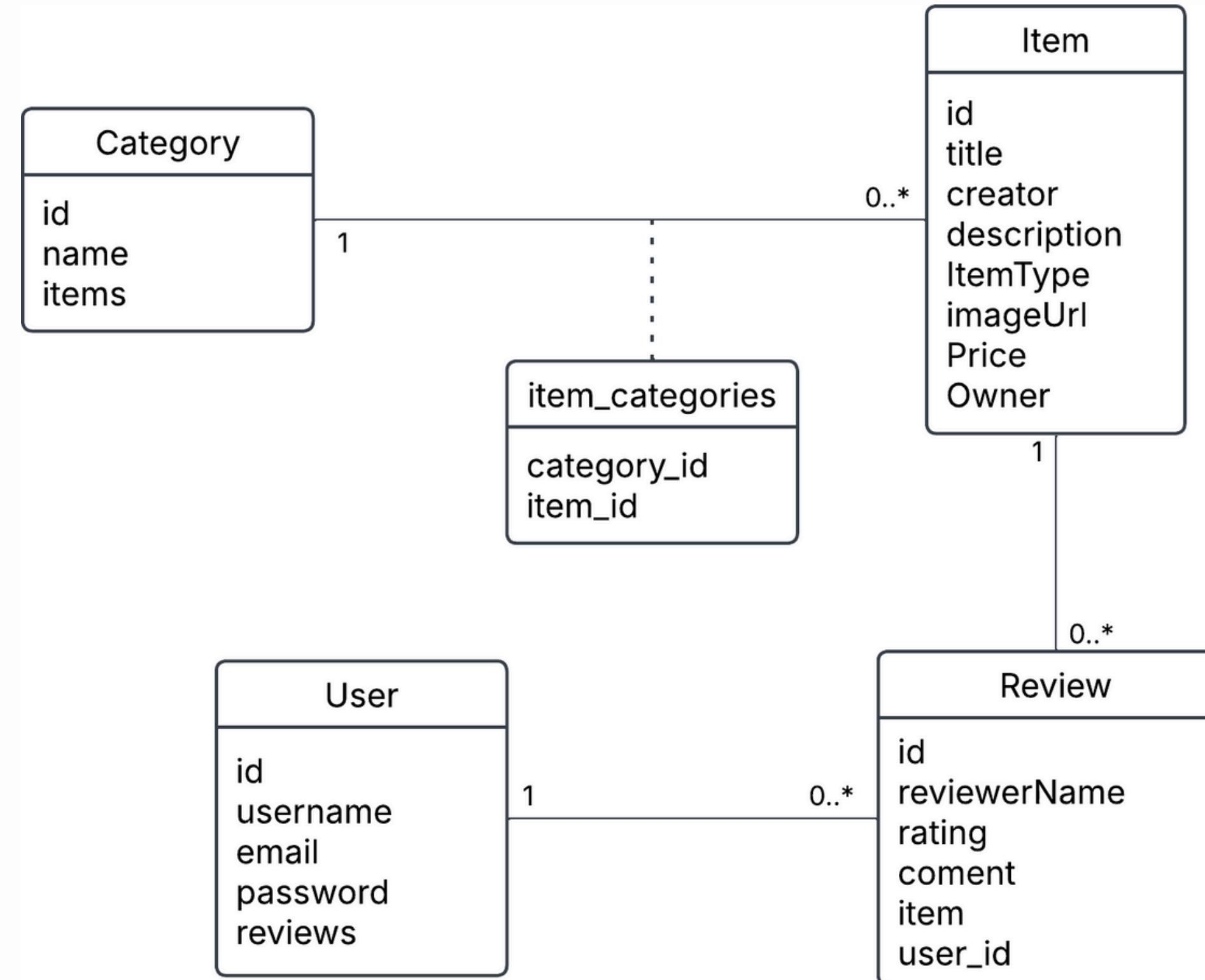
USECASE DIAGRAM



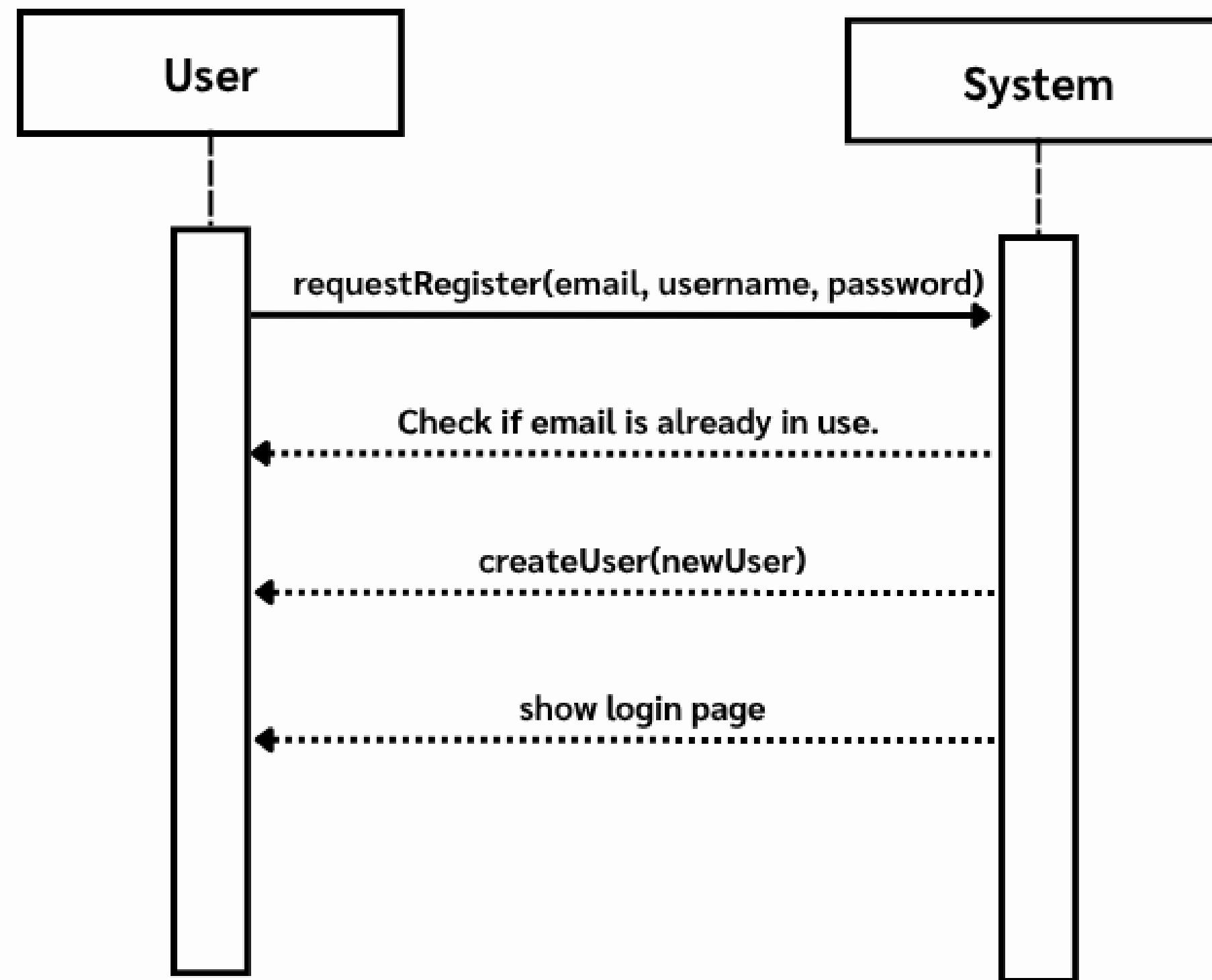
ACTIVITY DIAGRAM



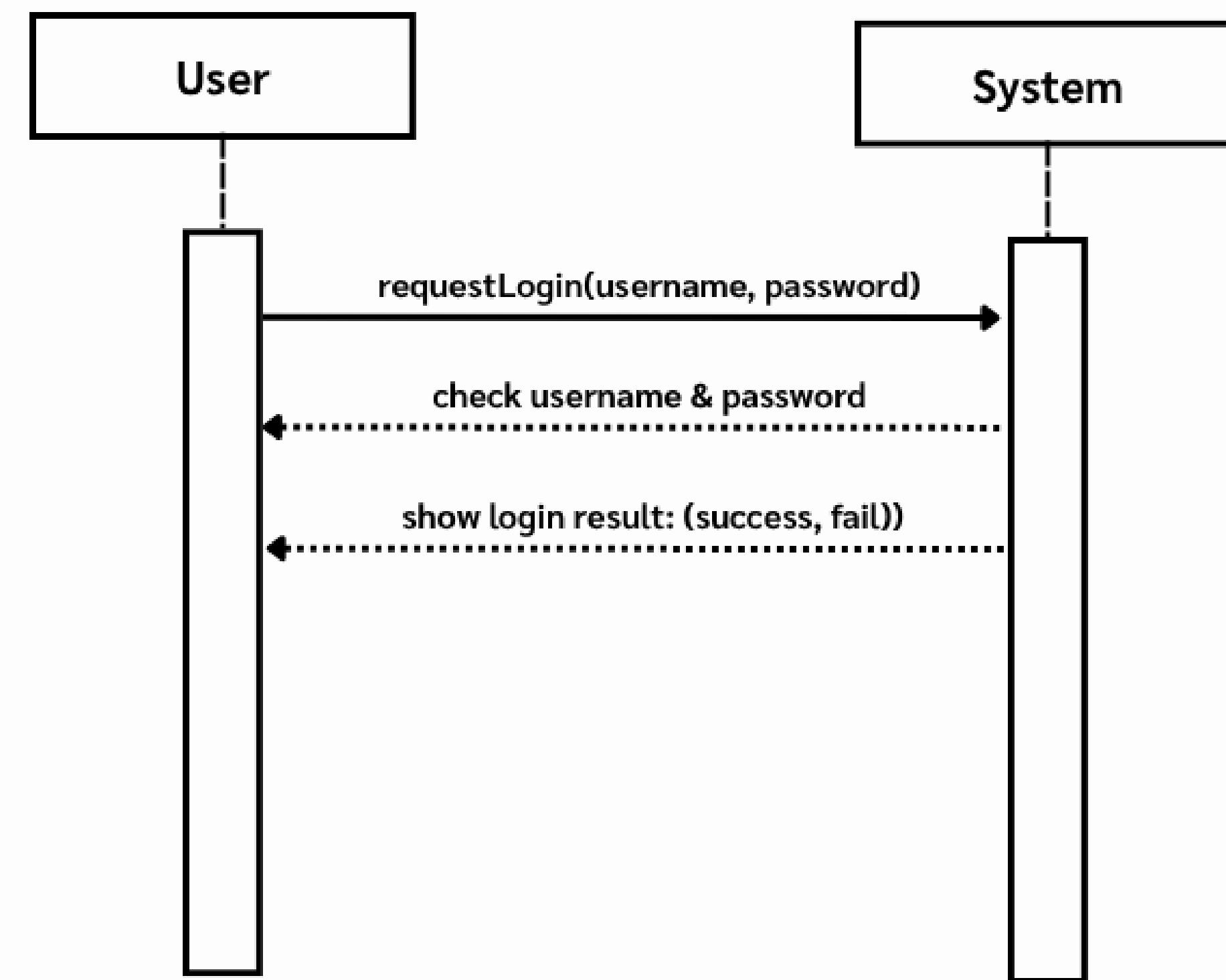
DOMAIN MODEL



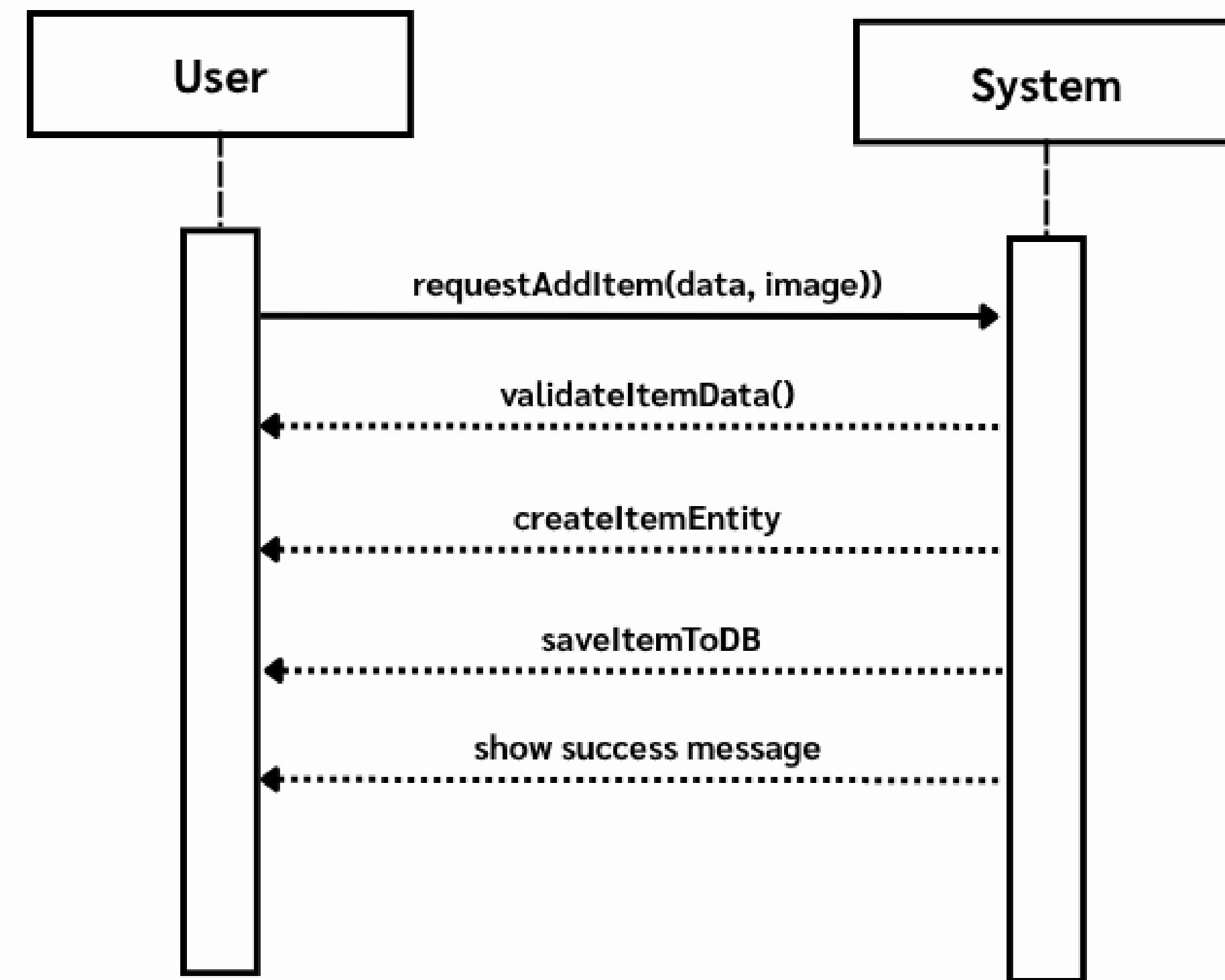
SEQUENCE DIAGRAM : register



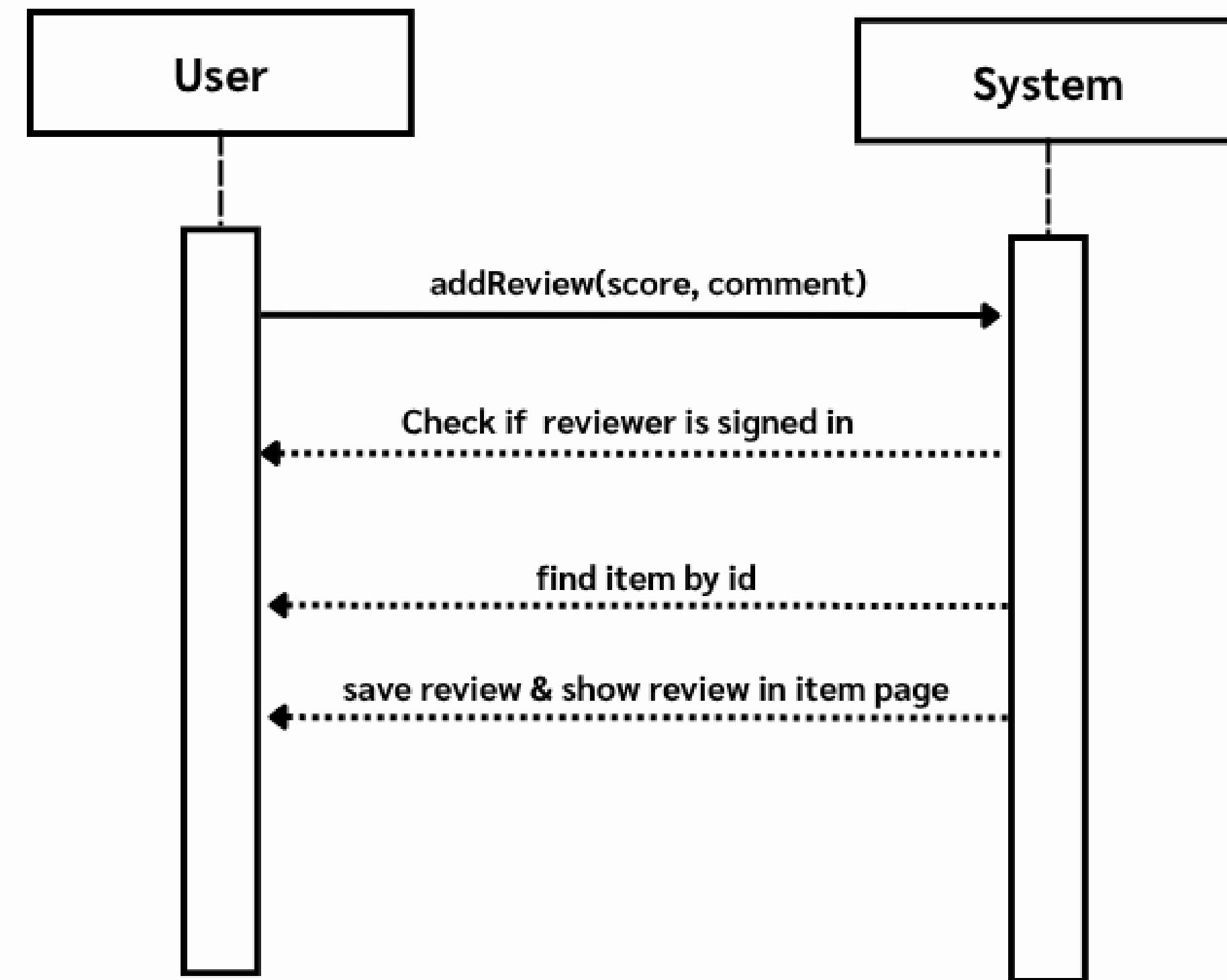
SEQUENCE DIAGRAM : login



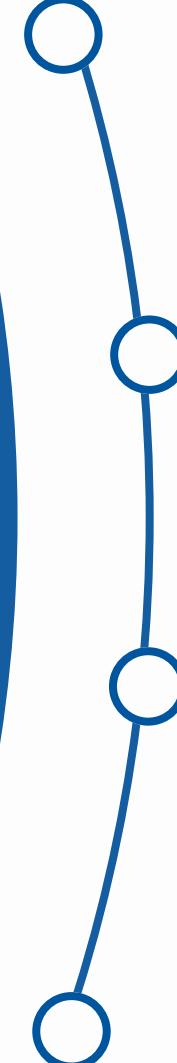
SEQUENCE DIAGRAM : addNewItem



SEQUENCE DIAGRAM : addReview



Design Pattern



Architectural Patterns: MVC (Model-View-Controller)

- **Controller:** CallController, AuthController, ItemWebController, CategoryWebController, HomeController
- **Service** (Business Logic / Model Layer): ItemService, CategoryService, ReviewService, UserService
- **View** (UI / Template): Thymeleaf templates ឧុ index.html, book.html

Design Pattern

Dependency Injection (DI) Constructor Injection:

```
private final ItemService itemService;  
public ItemWebController(ItemService itemService, CategoryRepository categoryRepository) {  
    this.itemService = itemService;  
    this.categoryRepository = categoryRepository;  
}
```

Design Pattern

Delegation Pattern: Controller ໄຟລ້າທຳ logic ເອງ →
“delegate” ໃກ້ Service ກໍາ

```
public String itemDetail(@PathVariable Long id, Model model, HttpSession session) {  
    ItemDTO item = itemService.getItemById(id);  
    model.addAttribute("item", item);
```

S.O.L.I.D Principles

S

ItemService - รับผิด
ชอบแค่ business logic
ของ entity นั้น

DTO - เก็บ/ส่งข้อมูล ไม่ทำ
logic อื่น

I

Repository interfaces
(ItemRepository,
CategoryRepository) :
Repository แยกตาม
entity, ไม่บังคับให้
implement method ที่
ไม่จำเป็น

L

Book, Movie, Cartoon,
Game เป็น subclass
ของ **Item**
สามารถส่ง Book, Movie,
Cartoon, Game และ
Item ได้โดยตรง

THANK YOU!