

กลุ่มที่ 6 เจ้าหนูสูงกนิภูฐานกับคนแคระห์ง 9

LIFE@KMITL

- ค้นหาวิชาเลือก
- รีวิuwิชาเรียน
- สรุปวิชาเรียน
- รีวิuhnอพัก

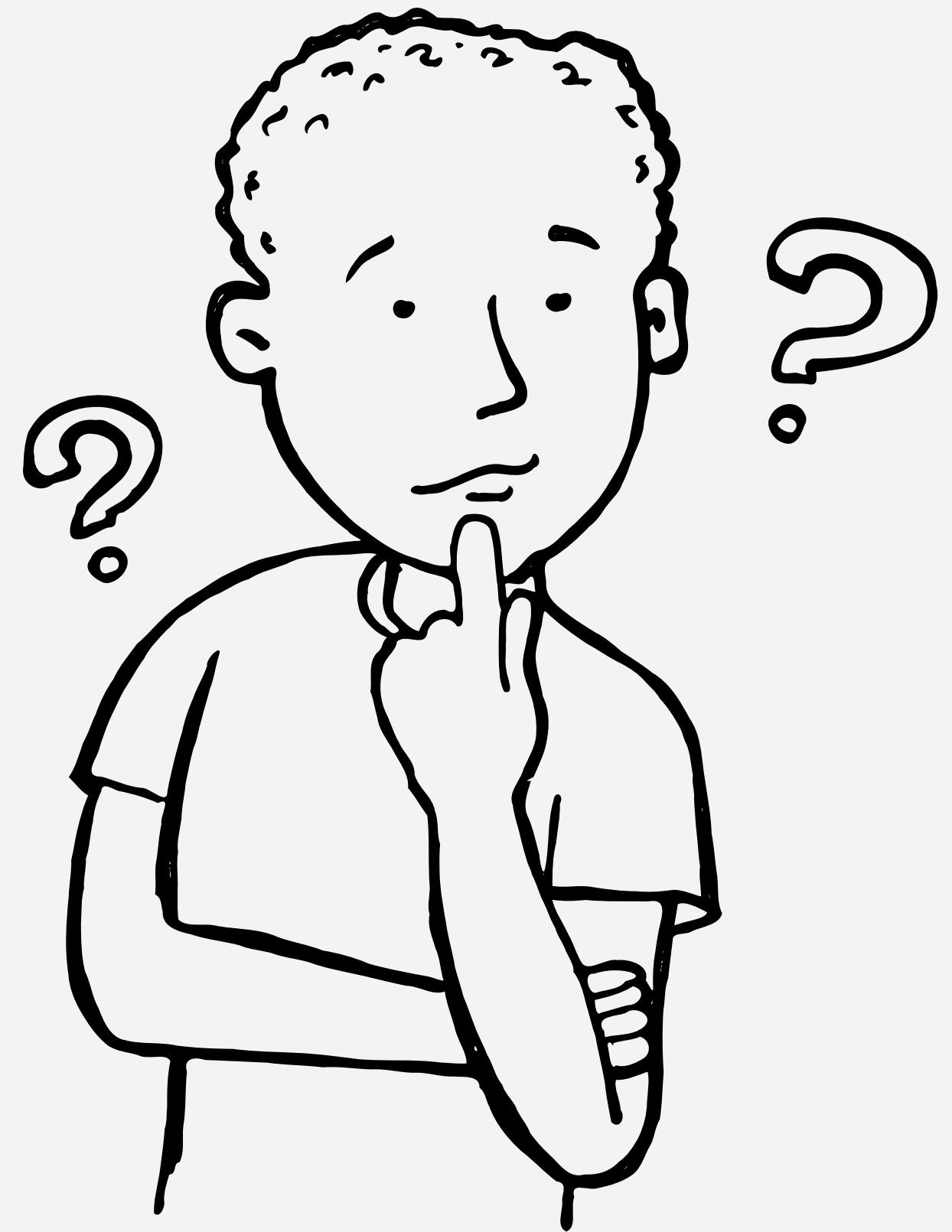


ສມາຜິກ

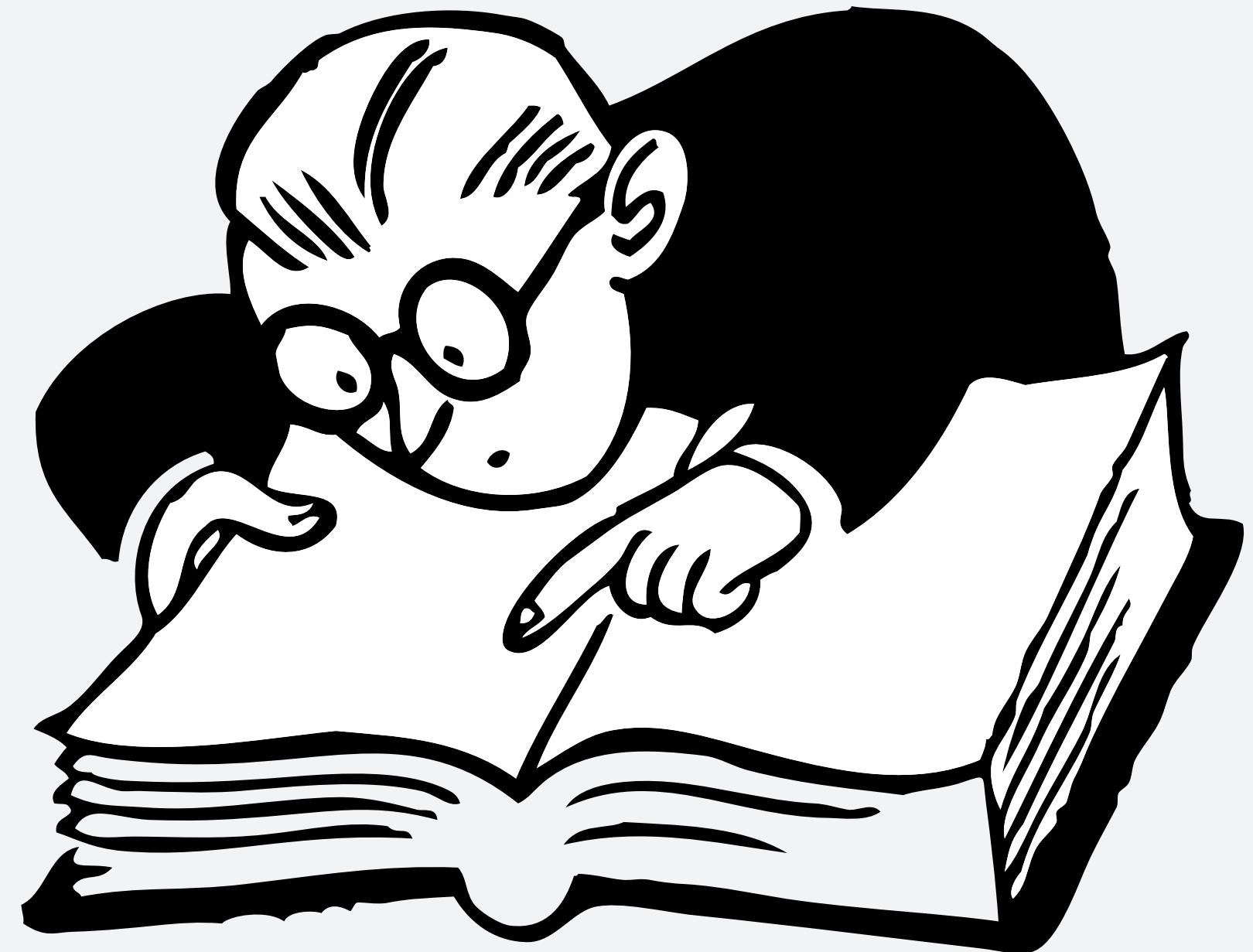
Points for discussion

61010014	ກນິ້ວຂ້າ	ຄສີປົກລົງຈັນກົດ
63010035	ກຖະໜາງ	ແສວງຕີຣິຜລ
63010212	ໜວກຄະ	ໃບດຳຮຈສັກດີ
63010235	ຈິນາຮີປ	ມືສວນນິລ
63010255	ຮູນກຮ	ວິທກກຸລ
63010283	ໝກຳກຮ	ມູລພິນິລ
63010320	ຄົ້ນຮູນນັ້ນກ	ເຮືອງພານີ່ຈ
63010326	ຄົ້ນຮູພງຫົ່ງ	ນາດສາມັດດີ
63010956	ສາບຖື້ພໍ່ໄມ່ນໍ້າ ນິຍມກອງ	
63011013	ສຸເມຣ	ລວນສໍາຮາບູ

ปัญหาทั่วไป



nauvihachaleokdaiyak



ไม่ด้อยมรภ.วิชาเรียน



ไม่มีศรubaเน็ต

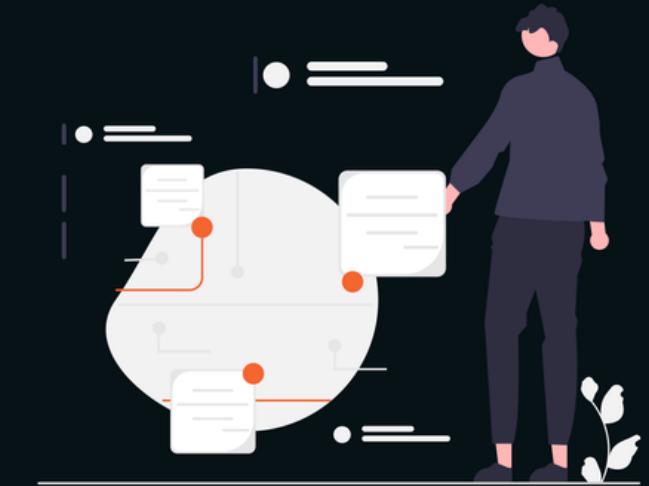


ຫາຫອພັກໄມ້ໄດ້



แก้ปัญหา





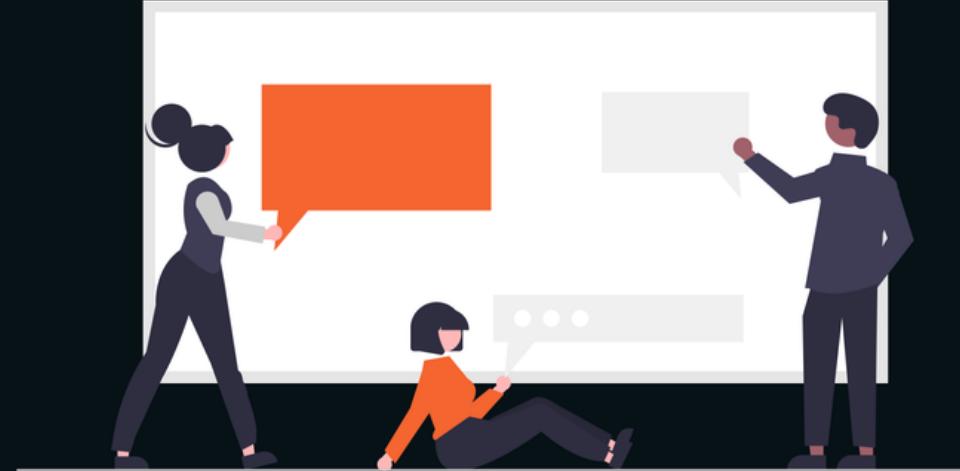
ค้นหาวิชาเลือก



IFE@
KMITL



สรุบวิชาเรียน



รีวิววิชาเรียน

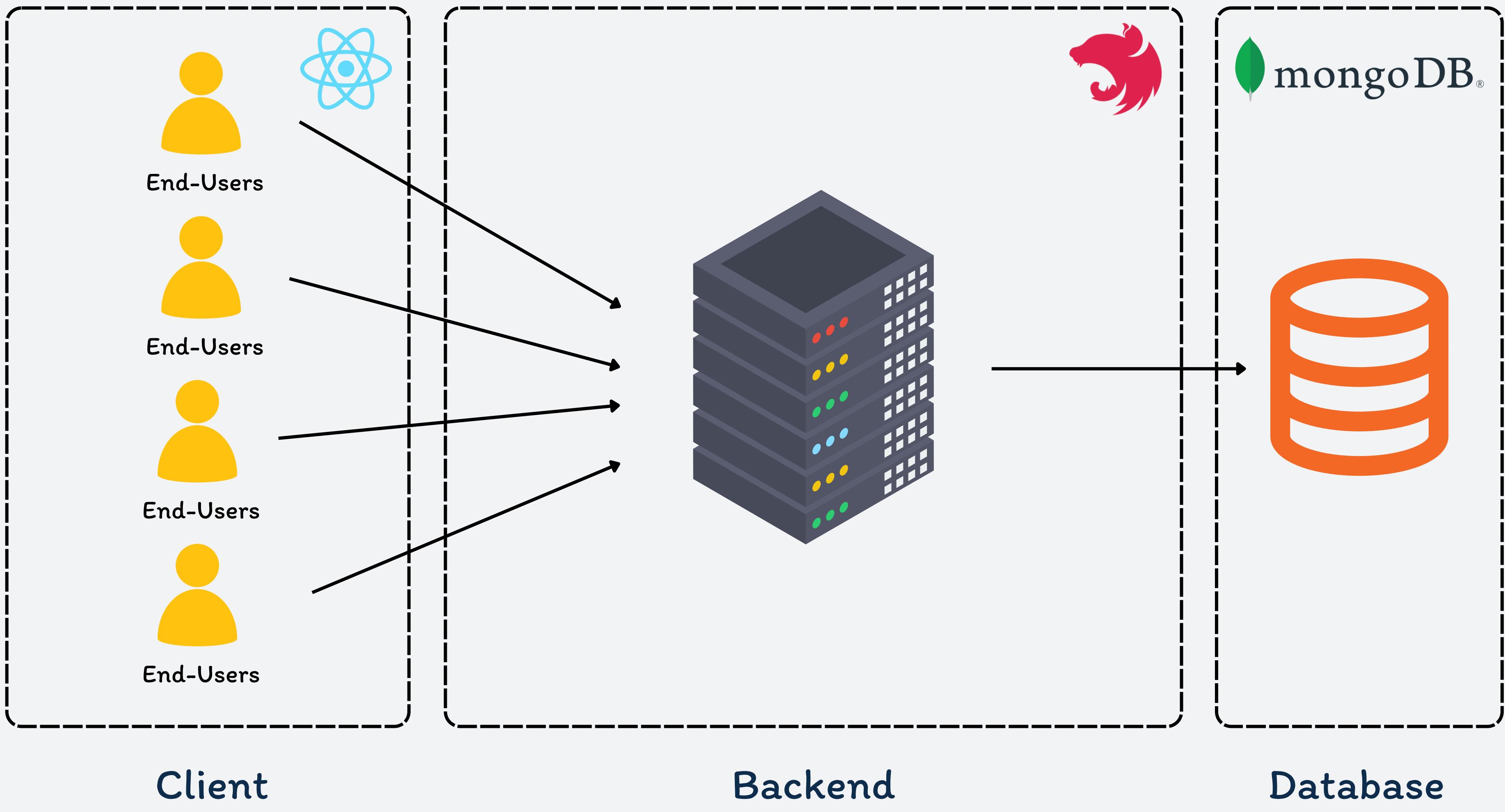


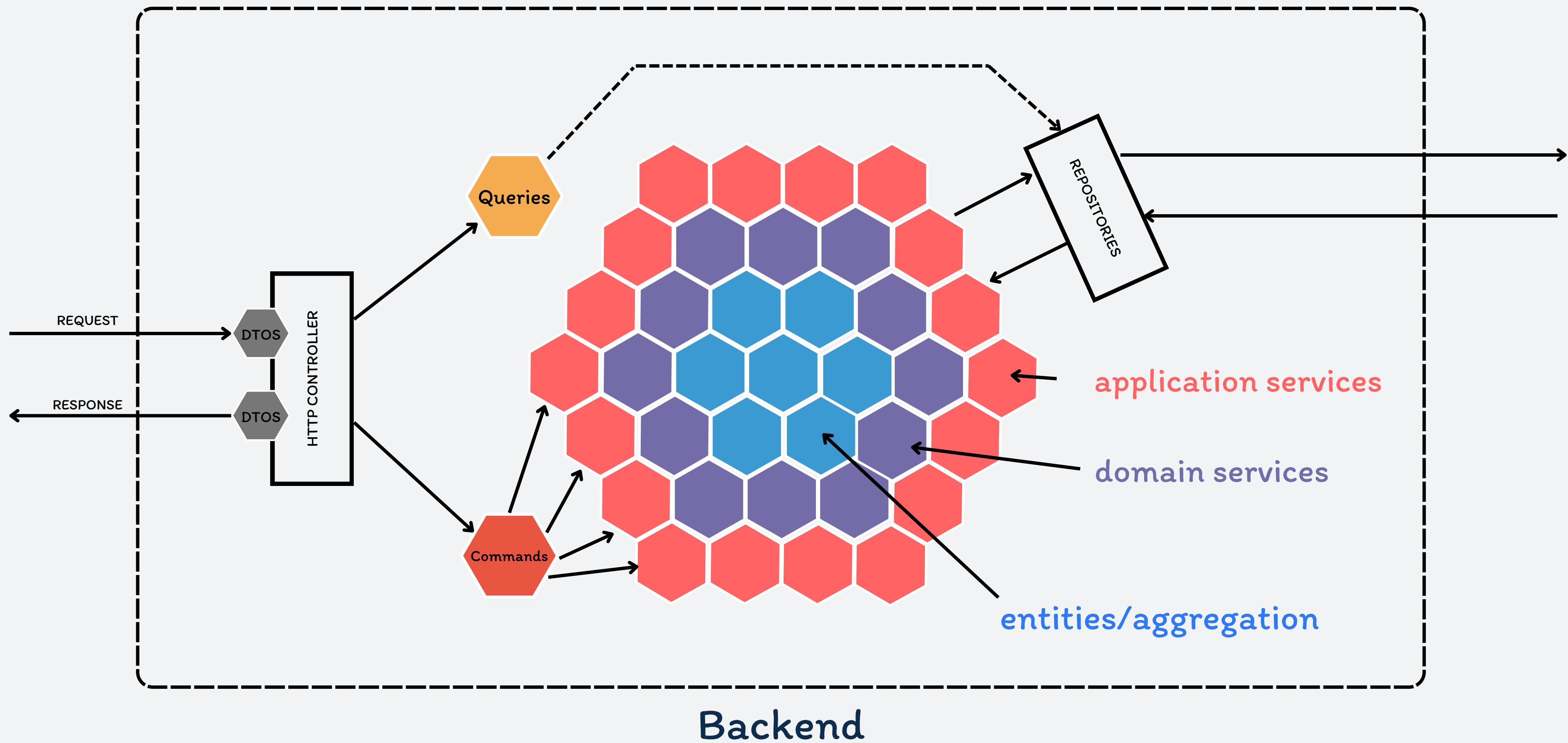
รีวิวหอพัก



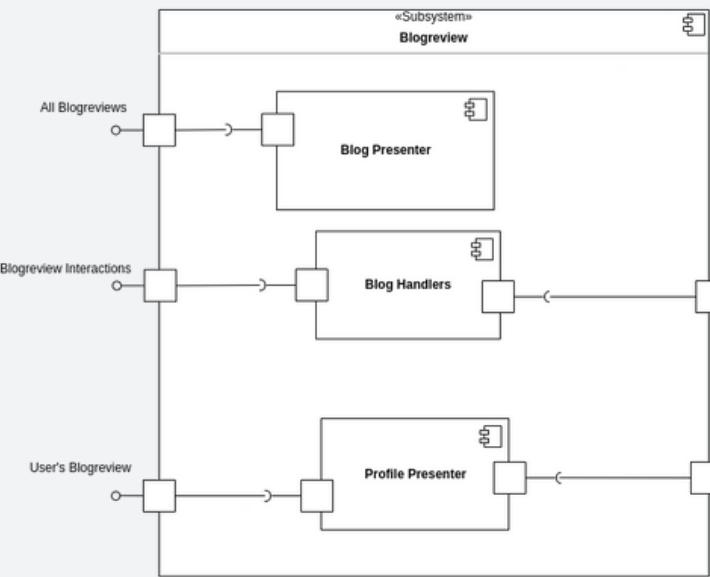


Software Architecture

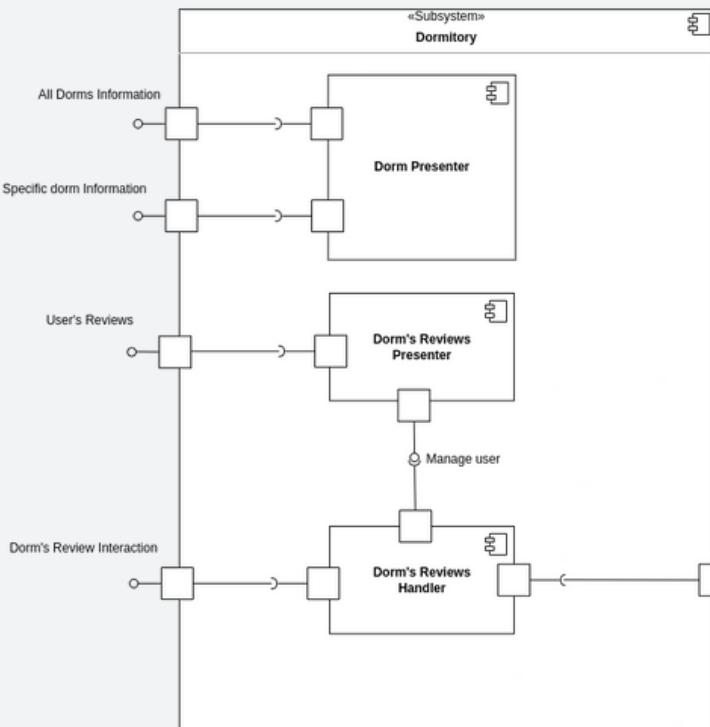




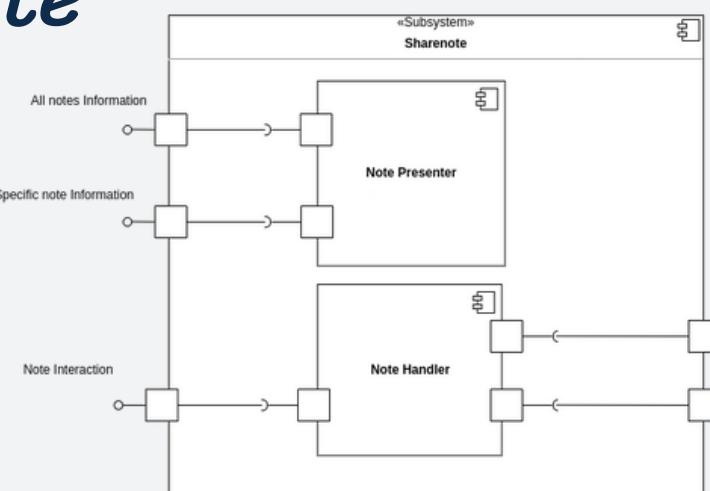
Blogreview



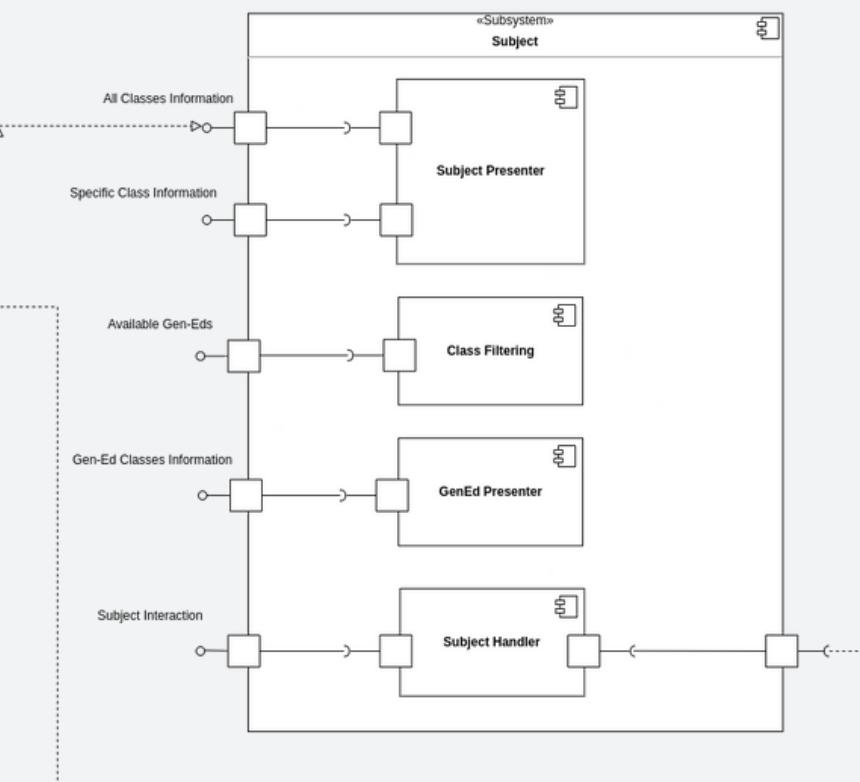
Dorm



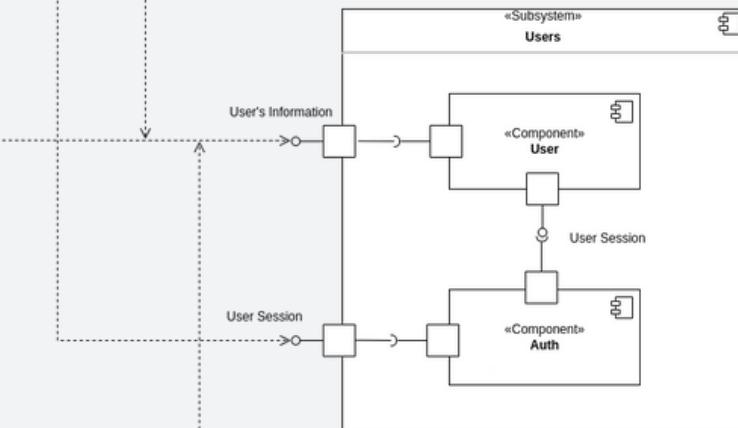
Sharenote



Subject

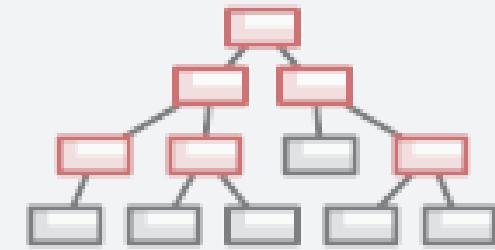
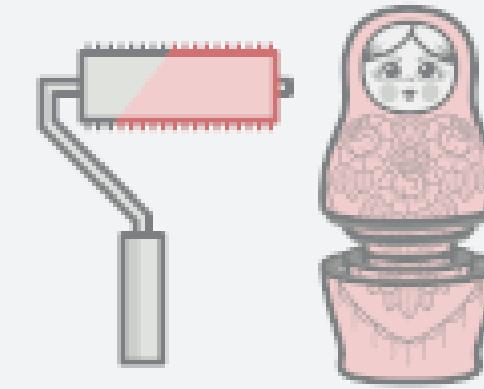
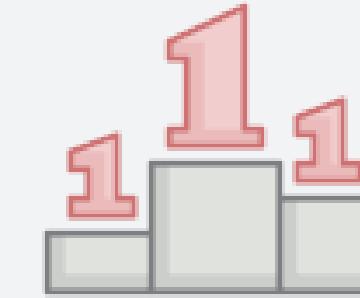
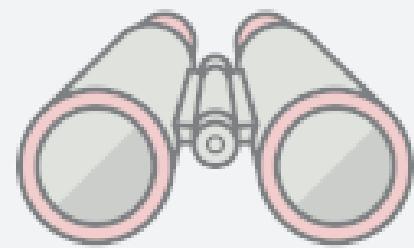
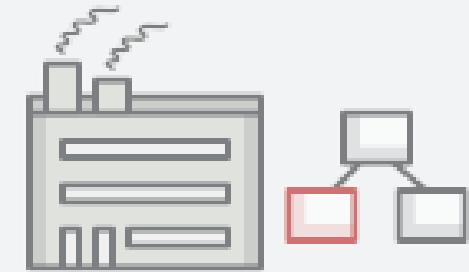
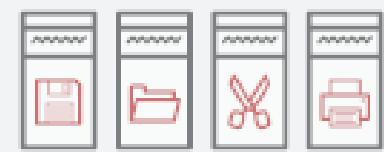
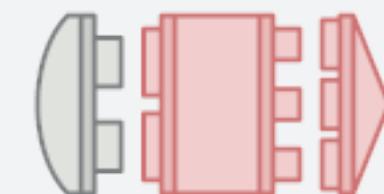
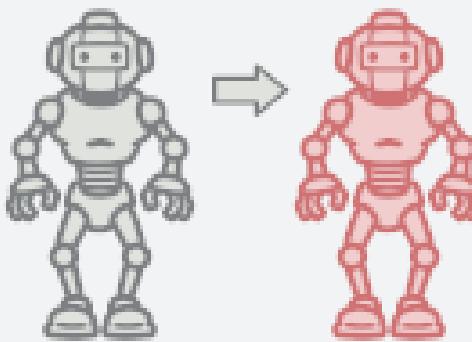


User

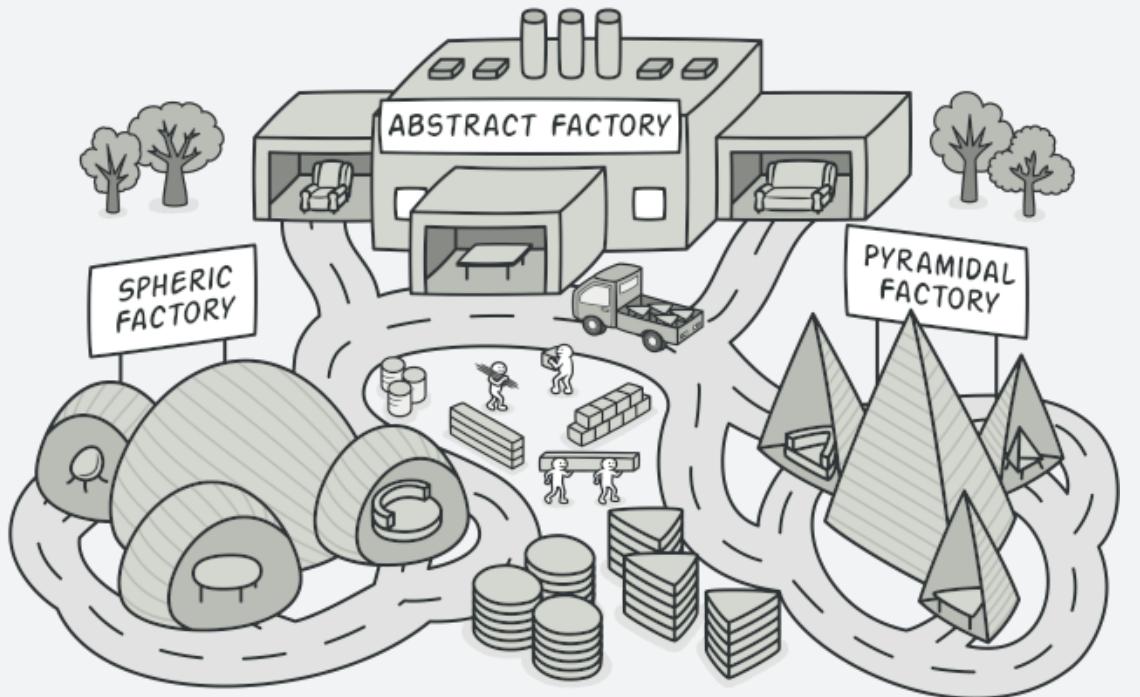


Components Diagram

DESIGN PATTERNS



Abstract Factory

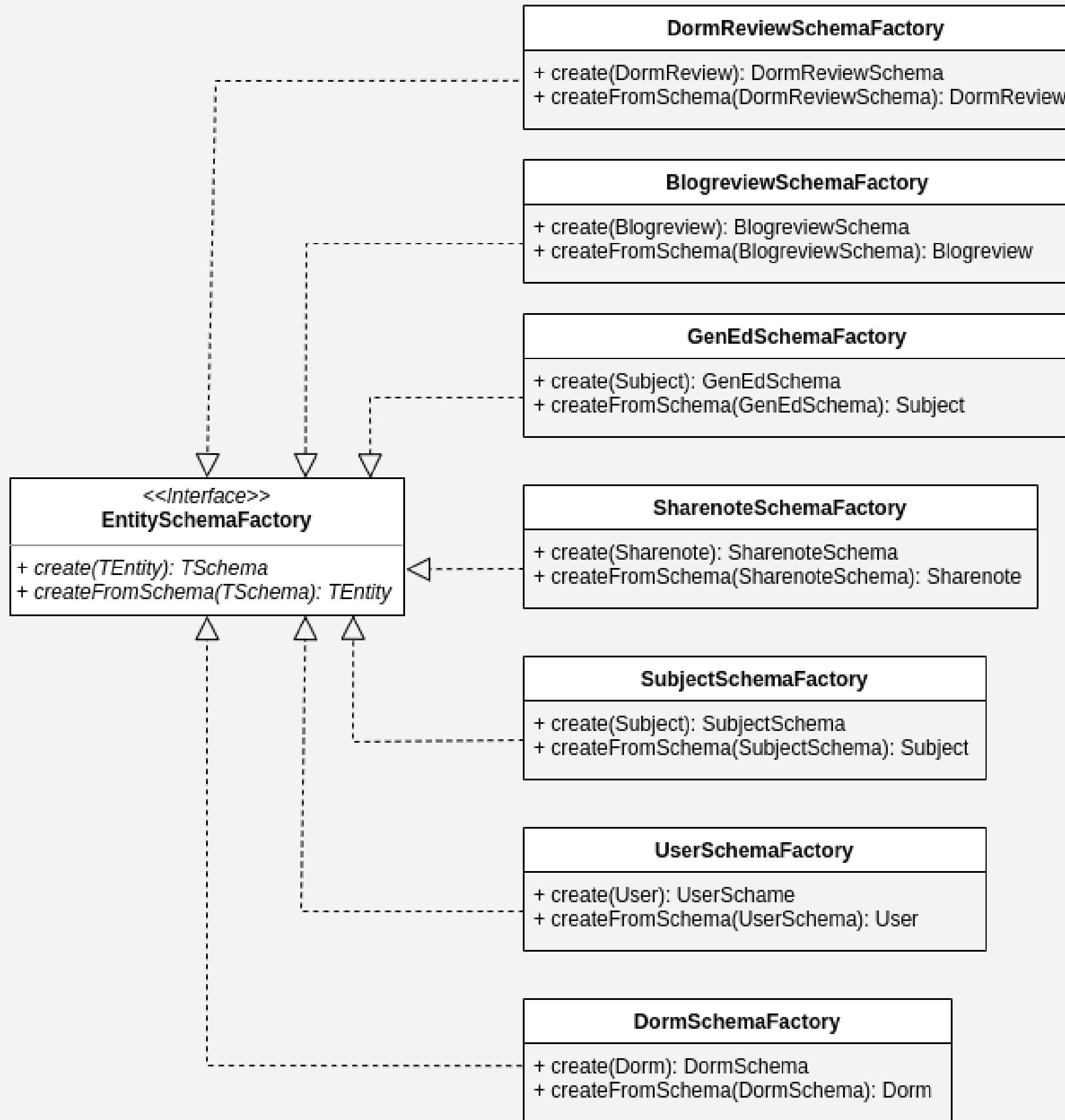


```
export interface EntitySchemaFactory<TSchema extends  
IdentifiableEntitySchema, TEntity extends AggregateRoot> {  
    create(entity: TEntity): TSchem;  
    createFromSchema(entitySchema: TSchem): TEntity;  
}
```

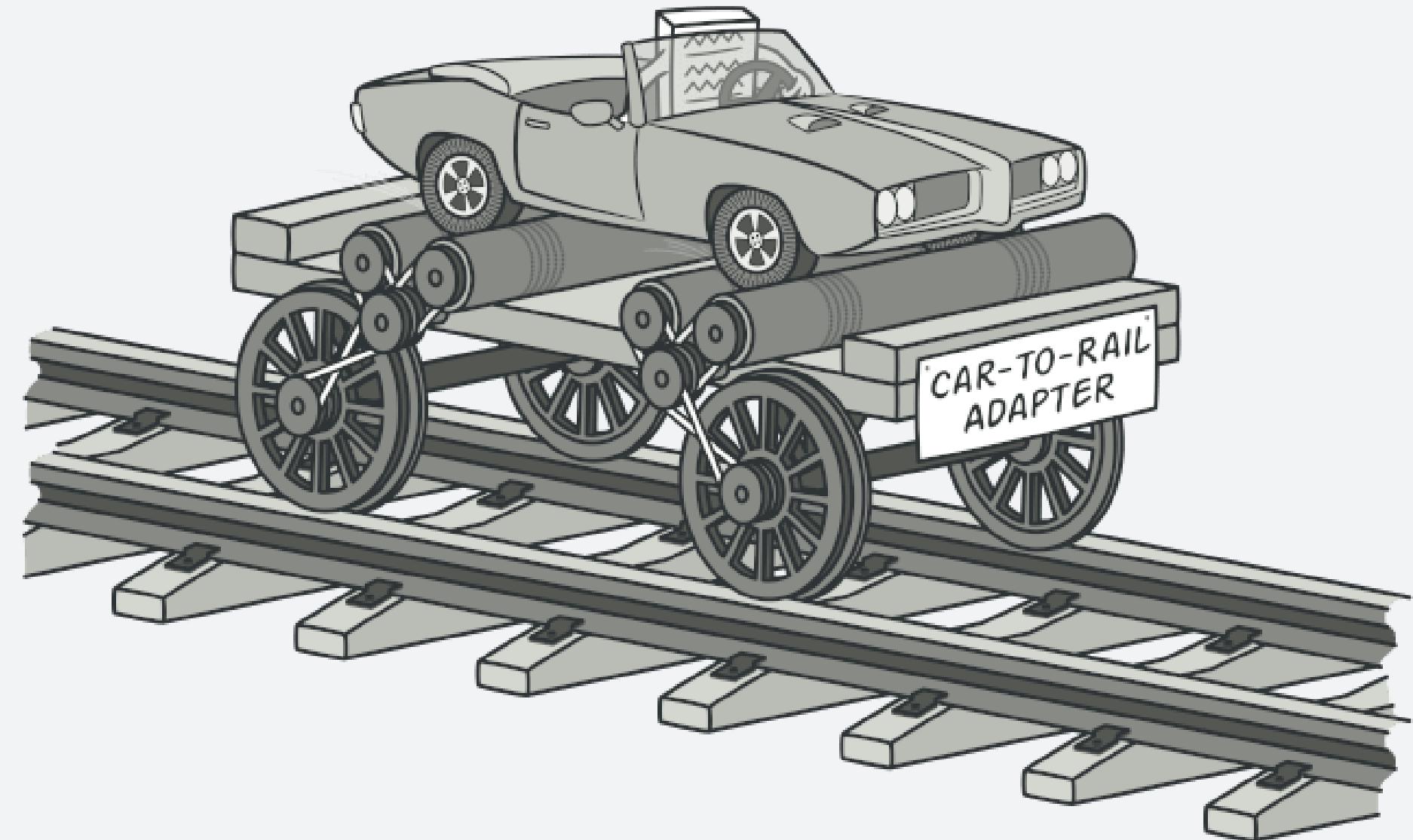
Code Snippets

```
export interface EntitySchemaFactory<TSchema extends IdentifiableEntitySchema, TEntity extends AggregateRoot> {
  create(entity: TEntity): TSchem;
  createFromSchema(entitySchema: TSchema): TEntity;
}
```

UML



Adapter

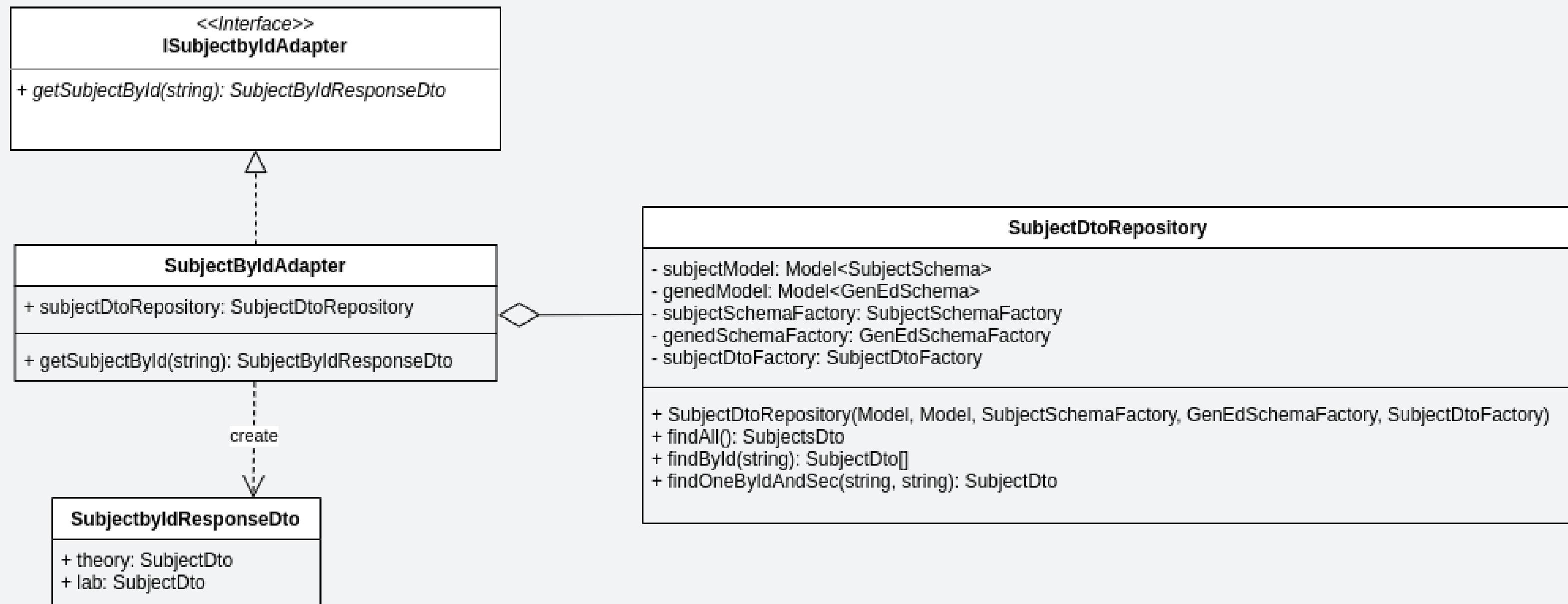


Code Snippets

```
1 export class SubjectByIdQueryHandler implements IQueryHandler {           'rac === 'n') {
2   constructor(private readonly subjectDtoRepository: SubjectDtoRepository) {}           'rac === 'l' && subject.secPair === null) {
3
4   async execute({ id }: SubjectByIdQuery): Promise<SubjectByIdResponseDto[]> {           }
5     const adapter = new SubjectByIdAdapter(this.subjectDtoRepository);           }
6     const res = adapter.getSubjectById(id);           }
7     return res;           }
8   }           }
9 }
```

```
1  export interface ISubjectByIdAdapter {           }
2    getSubjectById(id: string): Promise<SubjectByIdResponseDto[]>;           }
3  }           }
4
5  export class SubjectByIdAdapter implements ISubjectByIdAdapter {           }
6    constructor(private readonly subjectDtoRepository: SubjectDtoRepository) {}           }
7
8    async getSubjectById(id: string): Promise<SubjectByIdResponseDto[]> {           }
9      const foundSubjects = await this.subjectDtoRepository.findById(id);           }
10     const ret: SubjectByIdResponseDto[] = [];           }
11     foundSubjects.forEach((subject) => {           }
12       if (subject.secPair && subject.lectOrPrac === 'n') {           }
13         const pairSubject = foundSubjects.find(({ sec }) => sec === subject.secPair);           }
14         ret.push({           }
15           theory: subject,           }
16           lab: pairSubject,           }
17         );           }
18     );           }
19     return ret;           }
20   }           }
21 }
```

UML

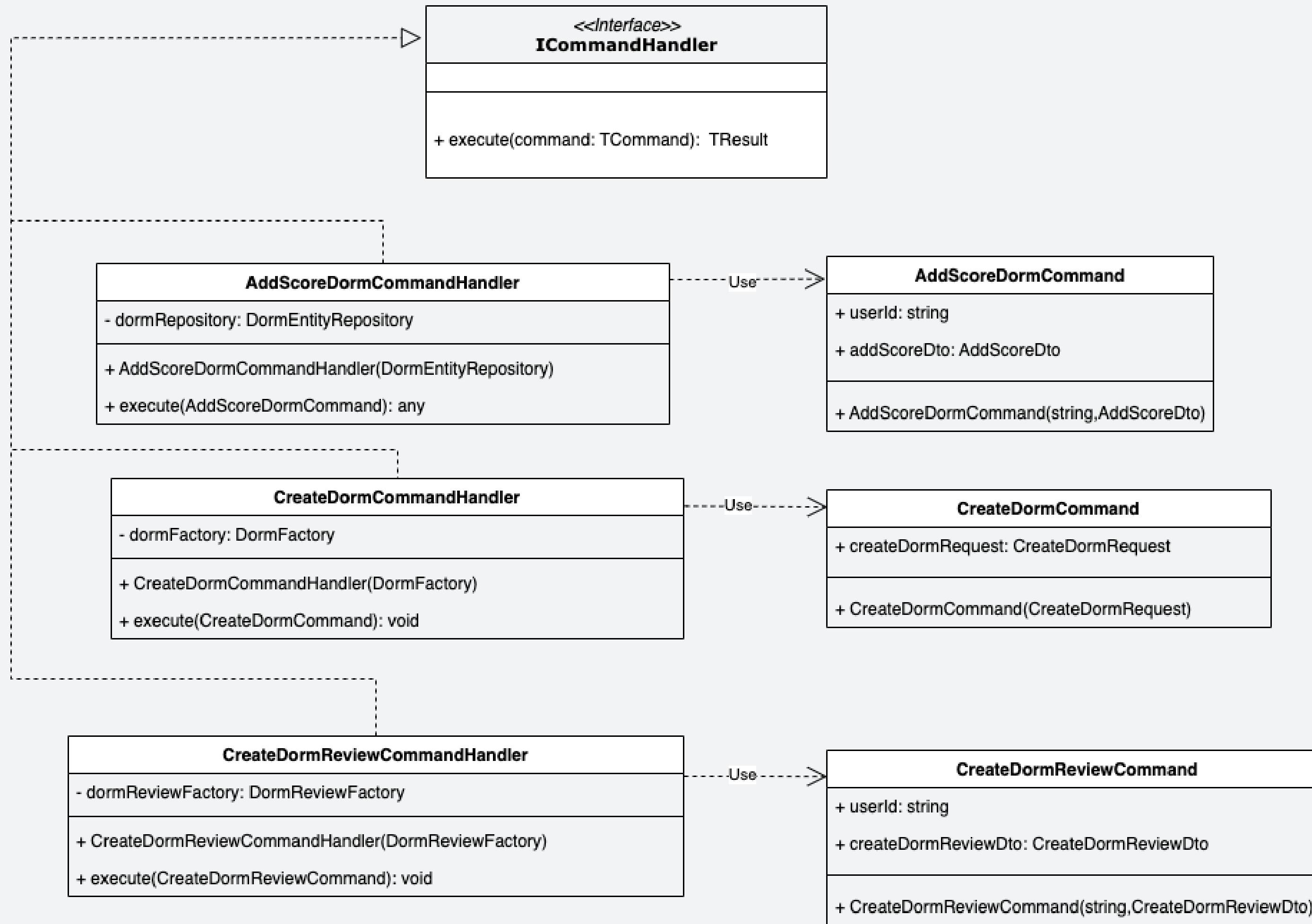


Command



```
● ● ●

1 @Get()
2 async getAllSubjects(): Promise<SubjectsDto[]> {
3   return await this.queryBus.execute<SubjectsQuery, SubjectsDto[]>(new SubjectsQuery());
4 }
5
6 @Get('gened')
7 async getAllGenEds(): Promise<SubjectsDto[]> {
8   return await this.queryBus.execute<AllGenEdQuery, SubjectsDto[]>(new AllGenEdQuery());
9 }
10
11 @Post()
12 async createSubject(@Body() createSubjectRequest: CreateSubjectRequest): Promise<void> {
13   await this.commandBus.execute<CreateSubjectCommand, void>(new CreateSubjectCommand(createSubjectRequest));
14 }
15
16 @Get(':id')
17 async getSubjectById(@Param('id') id: string): Promise<SubjectByIdResponseDto[]> {
18   return this.queryBus.execute<SubjectByIdQuery, SubjectByIdResponseDto[]>(new SubjectByIdQuery(id));
19 }
20
21 @Post('filter')
22 async filterSubject(@Body() filterSubjectRequest: FilterSubjectRequest[]): Promise<SubjectDto[]> {
23   return this.queryBus.execute<FilterSubjectQuery, SubjectDto[]>(new FilterSubjectQuery(filterSubjectRequest));
24 }
```



Security

ความปลอดภัย



SOURCE OF STIMULUS

ผู้ใช้งานที่ไม่มีสิทธิ์



STIMULUS

ต้องการเข้าถึงข้อมูล
ใน database



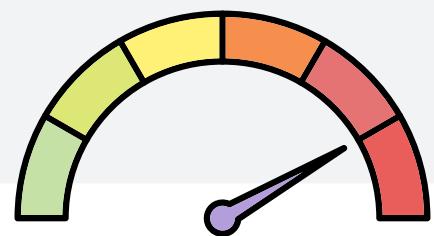
ARTIFACTS & ENVIRONMENT

database ในช่วงการ
ทำงานปกติ



RESPONSE

บอกให้ผู้ใช้งานทราบ
ว่าไม่มีสิทธิ์เข้าถึง



RESPONSE MEASURE

ข้อมูลใน database
คงอยู่ และไม่ถูกเข้า
ถึงโดยผู้ที่ไม่มีสิทธิ์

Modifiability

ความสามารถในการแก้ไข



SOURCE OF STIMULUS

Developer



STIMULUS

ต้องการเพิ่ม / ลบ /
แก้ไข / เปลี่ยนฟังก์ชัน



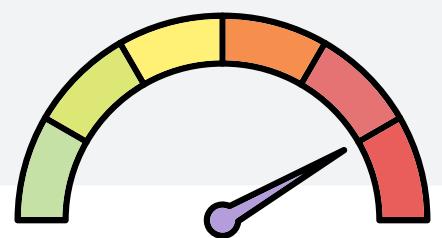
ARTIFACTS & ENVIRONMENT

System & Runtime



RESPONSE

เปลี่ยนแล้วไม่มีผล
กระบวนการ กับ ฟังก์ชันอื่นๆ



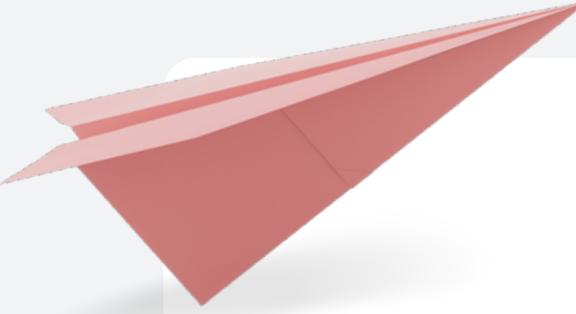
RESPONSE MEASURE

เวลาที่ใช้ในการ
ปรับปรุงแก้ไข
(Auto deploy 2-5
minutes)

Availability

ความพร้อมของ software





Let's Demo~

