

**Application**

---

**Busking Service**

**Hello.**

**We are**

**981 PROJECT.**

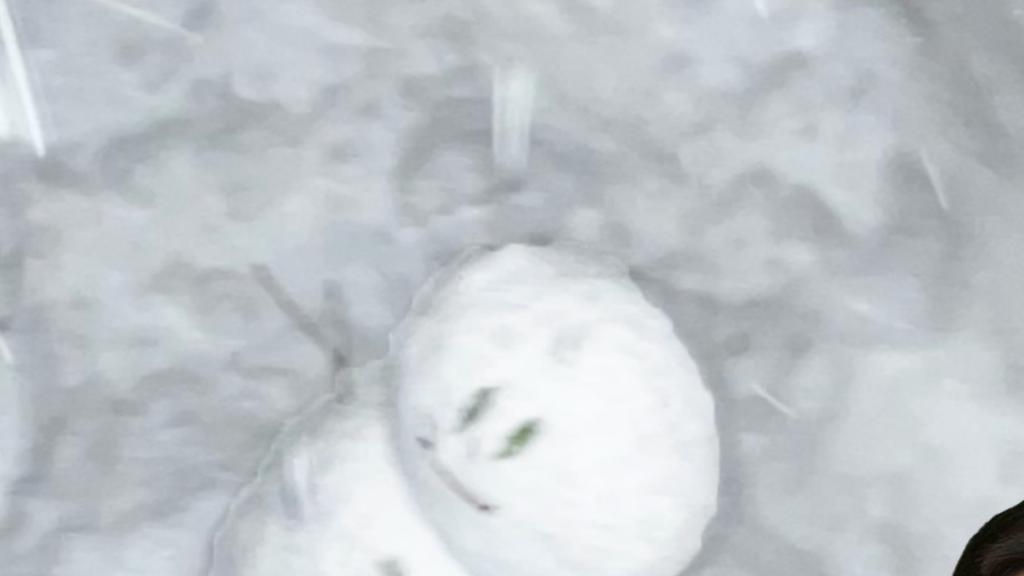




**Developer 김근성(2001)**



**Designer 김민지(1999)**



**Developer 장홍준(1999)**



**PM 박준수(1998)**



# Contents.

- 1. About 'DunDun'**
- 2. Architecture Design**

D u a D u a



Duca  
Duca



**About 'DunDun'**

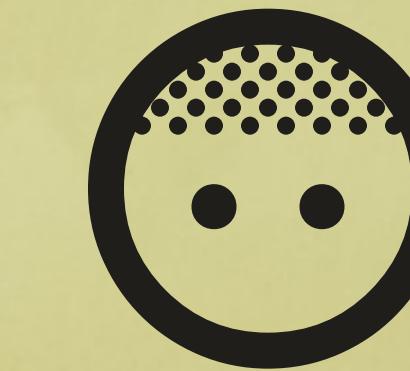
## **User Scenario**



**HONGJUN**



**MINJI**



**GEUNSEONG**



Permanent Marker



Sharpie® FINE POINT Permanent Marker AP

**About 'DunDun'**

## User Scenario



**HONGJUN**

색소폰에 흥미가 있는 개발자



**MINJI**



**GEUNSEONG**



Permanent Marker

밴드 활동 ing  
자신을 표현할 수 있는  
채널의 부재가 아쉬울 때



Sharpie FINE POINT Permanent Marker AP

**About 'DunDun'**

## User Scenario



**HONGJUN**



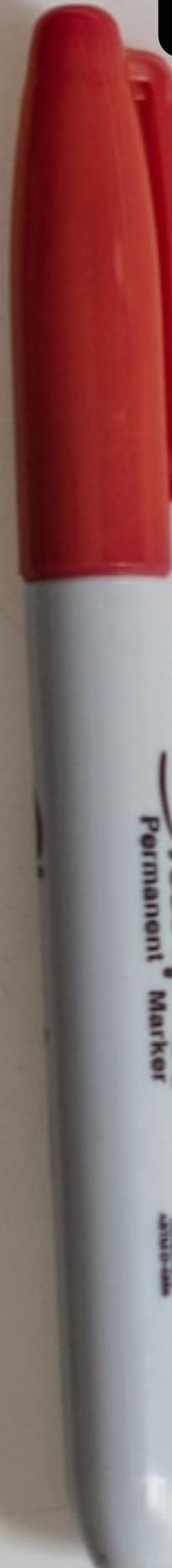
**MINJI**

인디음악을 즐겨듣는 디자이너



**GEUNSEONG**

인디밴드 계정 팔로잉  
매번 공연을 가고 싶지만  
일정과 장소에 대한 정보 놓침



**About 'DunDun'**

## User Scenario



**HONGJUN**



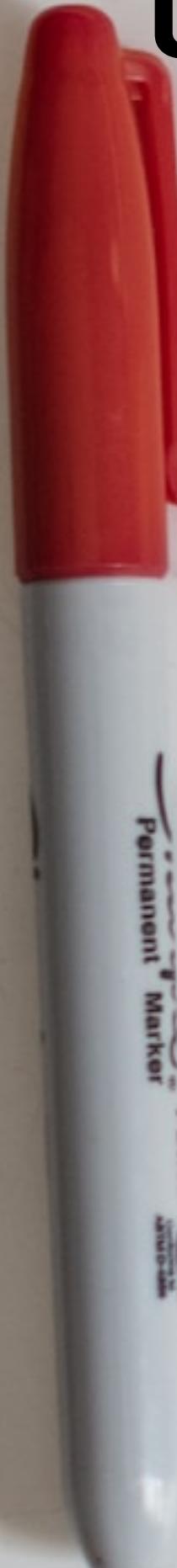
**MINJI**



**GEUNSEONG**

아이돌을 더 좋아하는.!!

코로나 때 라이브 무대에  
관심이 생김  
이제는 공연을 보고 싶지만  
어디서 언제 하는지 모름



## **About 'DunDun'**

## **Pain point!**

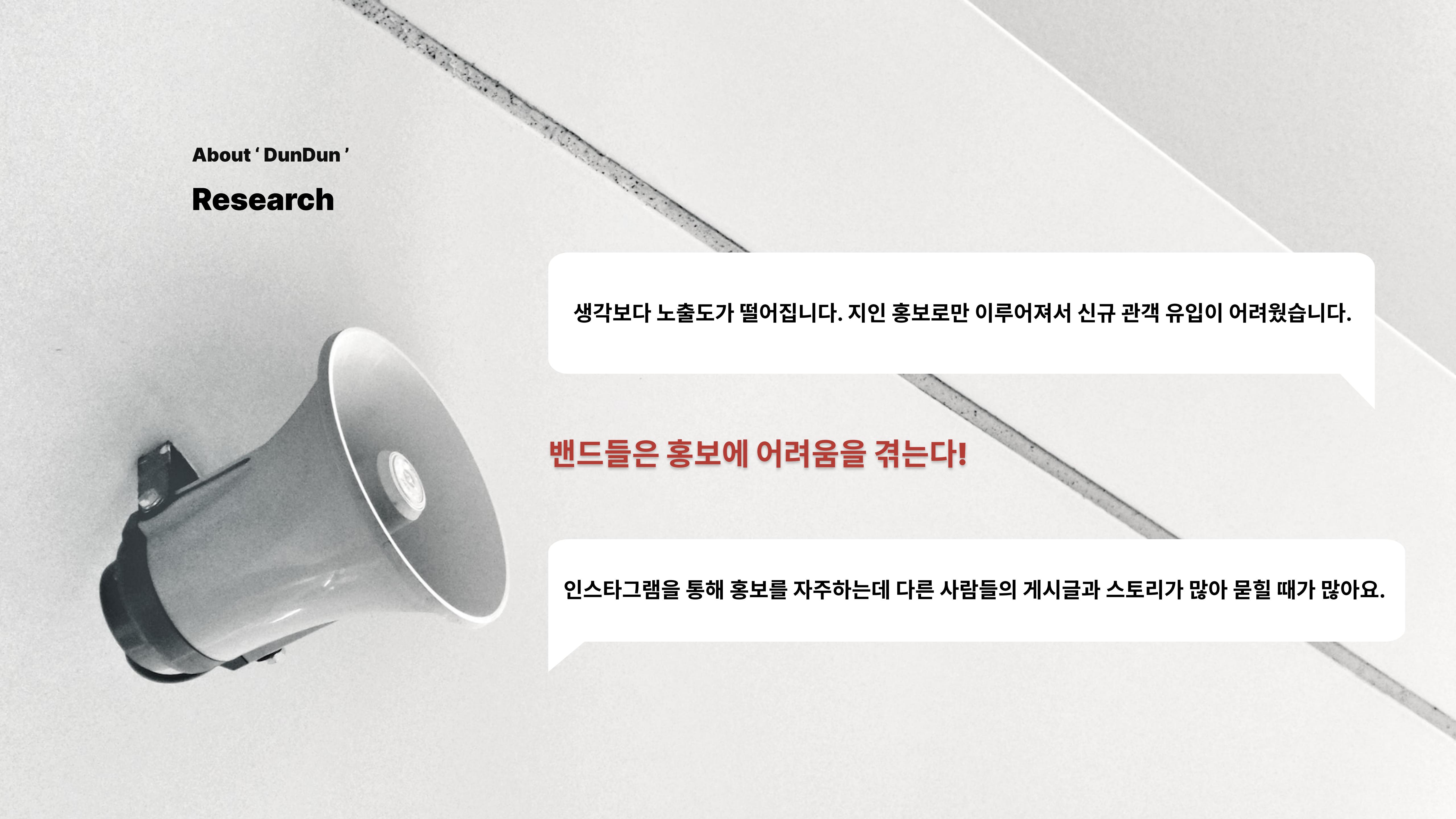
자신들을 홍보하고 싶지만  
그럴 수 있는 채널의 부재를 느낀  
아티스트들이 존재할 것이다.

## About 'DunDun'

### Needs point!

1. 인디 문화를 사랑하고,  
자신이 좋아하는 밴드를  
꾸준하게 팔로우하고 싶은 팬

2. 인디 음악에 흥미를 느끼게 되었지만  
어디서부터 시작해야 할지 모르겠는 리스너



## About 'DunDun' Research

생각보다 노출도가 떨어집니다. 지인 홍보로만 이루어져서 신규 관객 유입이 어려웠습니다.

**밴드들은 홍보에 어려움을 겪는다!**

인스타그램을 통해 홍보를 자주하는데 다른 사람들의 게시글과 스토리가 많아 묻힐 때가 많아요.

**About 'DunDun'**

**Research**

유저들은 모두 **인스타그램**을 통해 버스킹과 공연을 접한다.

→ 그러나 일정과 장소를 정확하게 알지 못하는 점에서 문제점을 느낀다.



Sun

Mon

Tue

Wed

Thu

Fri

About 'DunDun'

Research

1

2

3

4

5

7

8

9

10

11

12

14

15

16

17

18

19

21

22

23

24

25

26

공연의 위치를 알려주는 지도 기능과

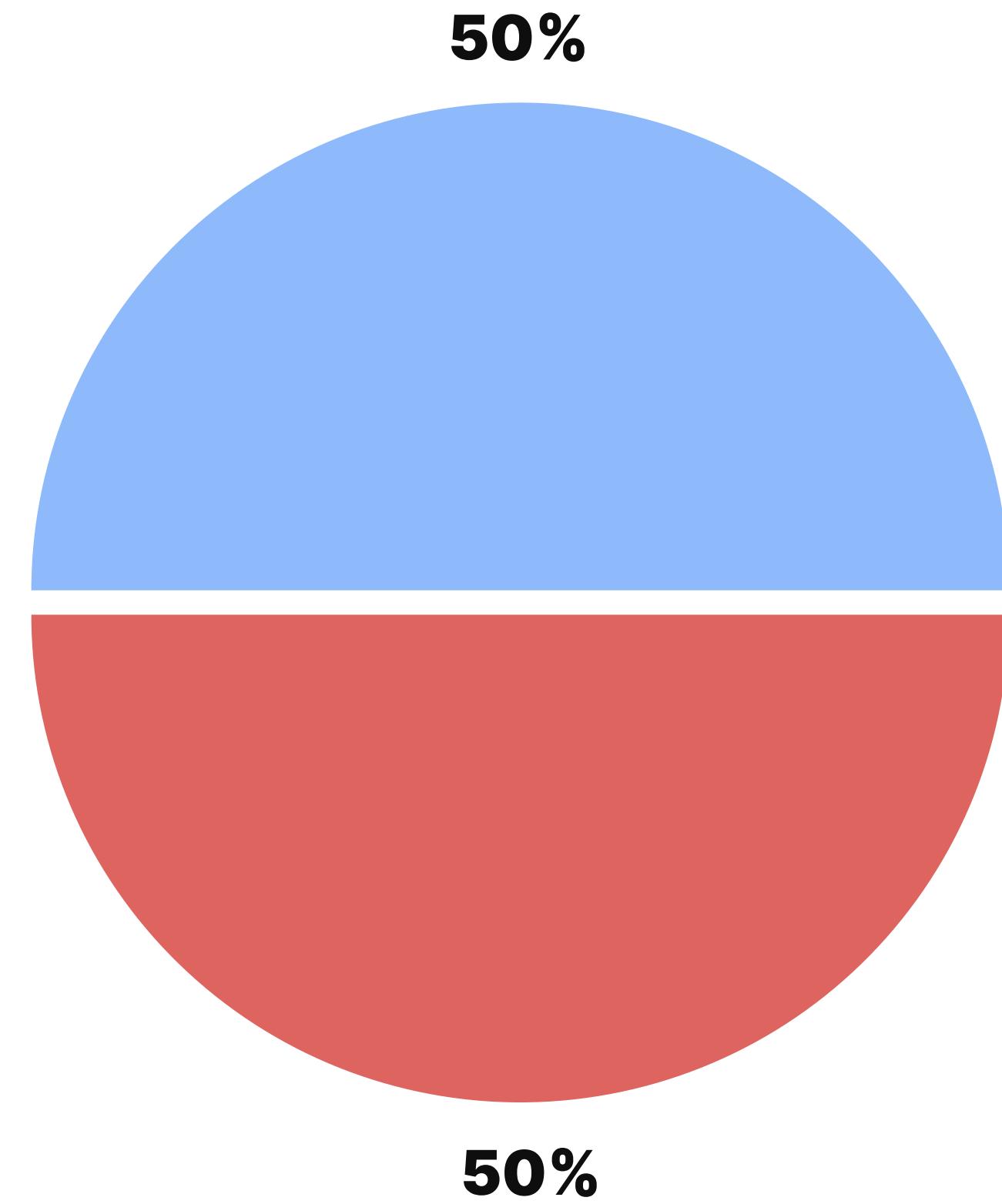
공연의 날짜를 알려주는 캘린더 기능을 가장 필요로 한다.

## About 'DunDun'

## Research

2.1 평소에 버스킹 문화, 혹은 LIVE 공연 문화에 대해 관심을 갖고 계신가요?  
(버스킹, 인디밴드, 지인의 밴드 라이브 공연 등등)

- 관심이 없다.
- 관심이 있다.



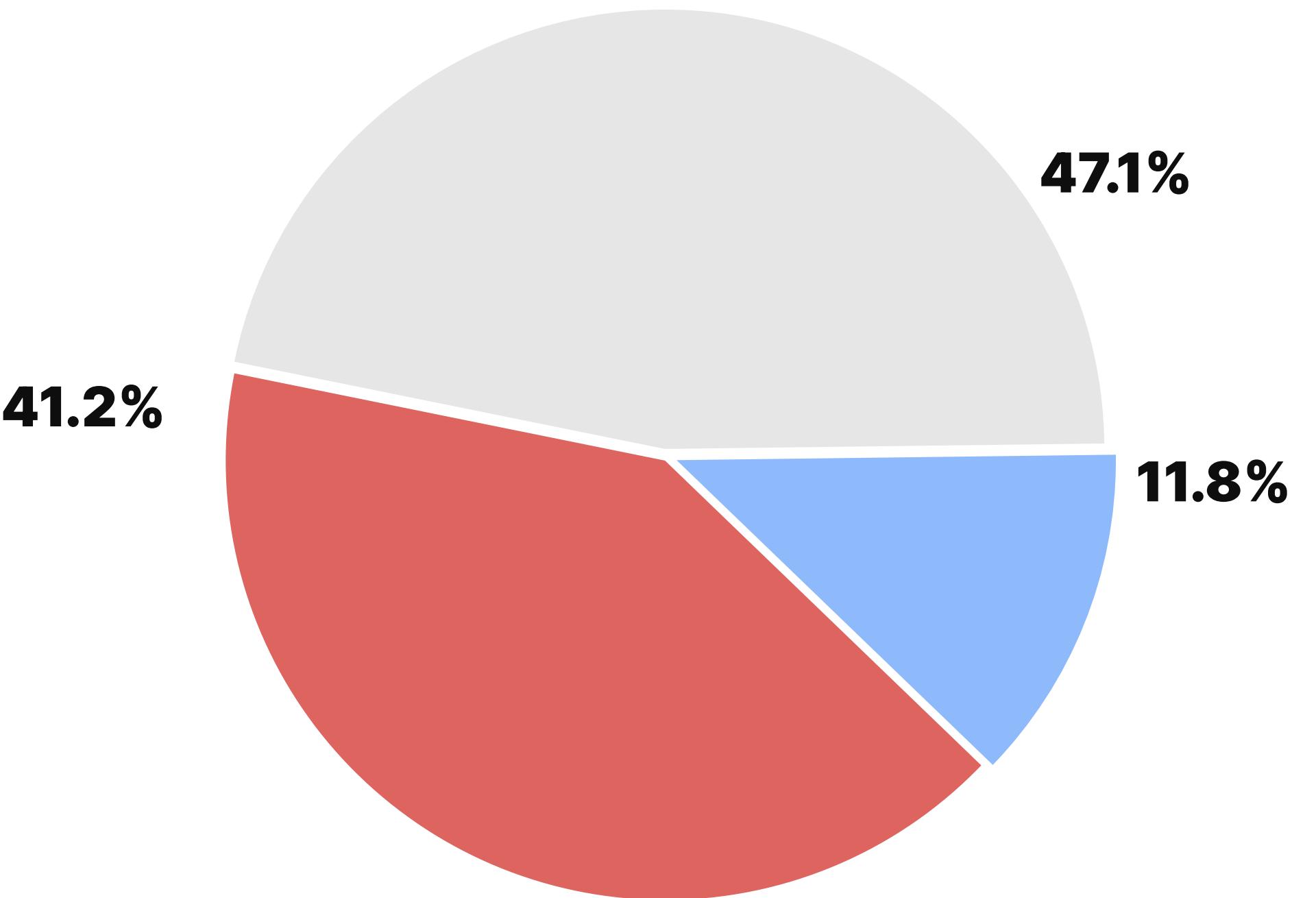
생각보다 많은 사람들이 버스킹, 공연 문화에 관심을 갖고 있다.

## About 'DunDun'

## Research

2.1.1 (관심이 있다)라고 답변하신 경우,  
COVID-19이 그 관심에 영향을 끼쳤다고 생각하십니까?

- 이전부터 관심이 있었다.
- COVID-19 이후 관심이 줄어들었다.
- COVID-19 이후 관심이 커졌다.



코로나는 공연문화에 부정적인 영향을 끼쳤다.

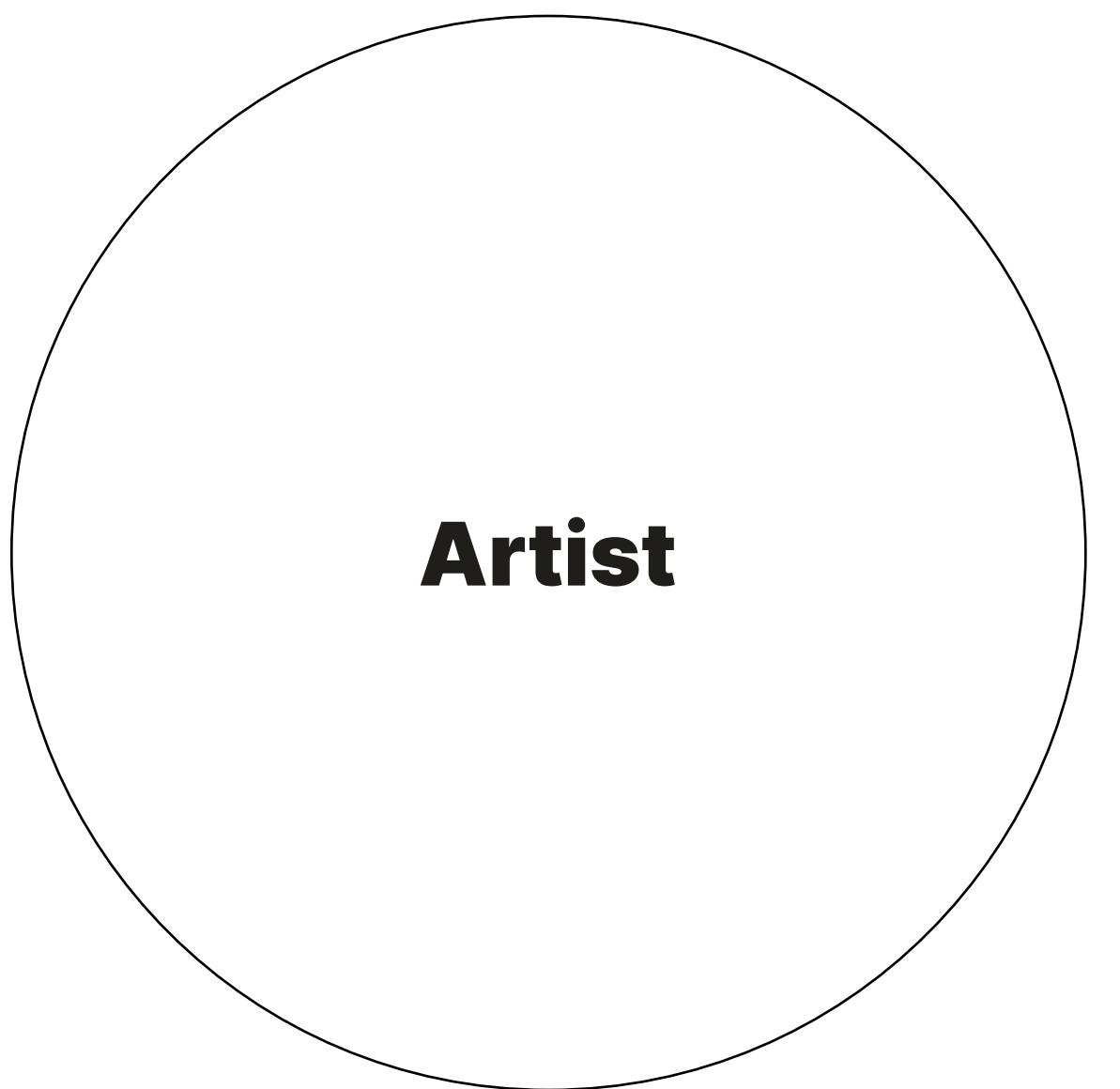
**About 'DunDun'**  
**PM's Tip!**



**Target Audience**

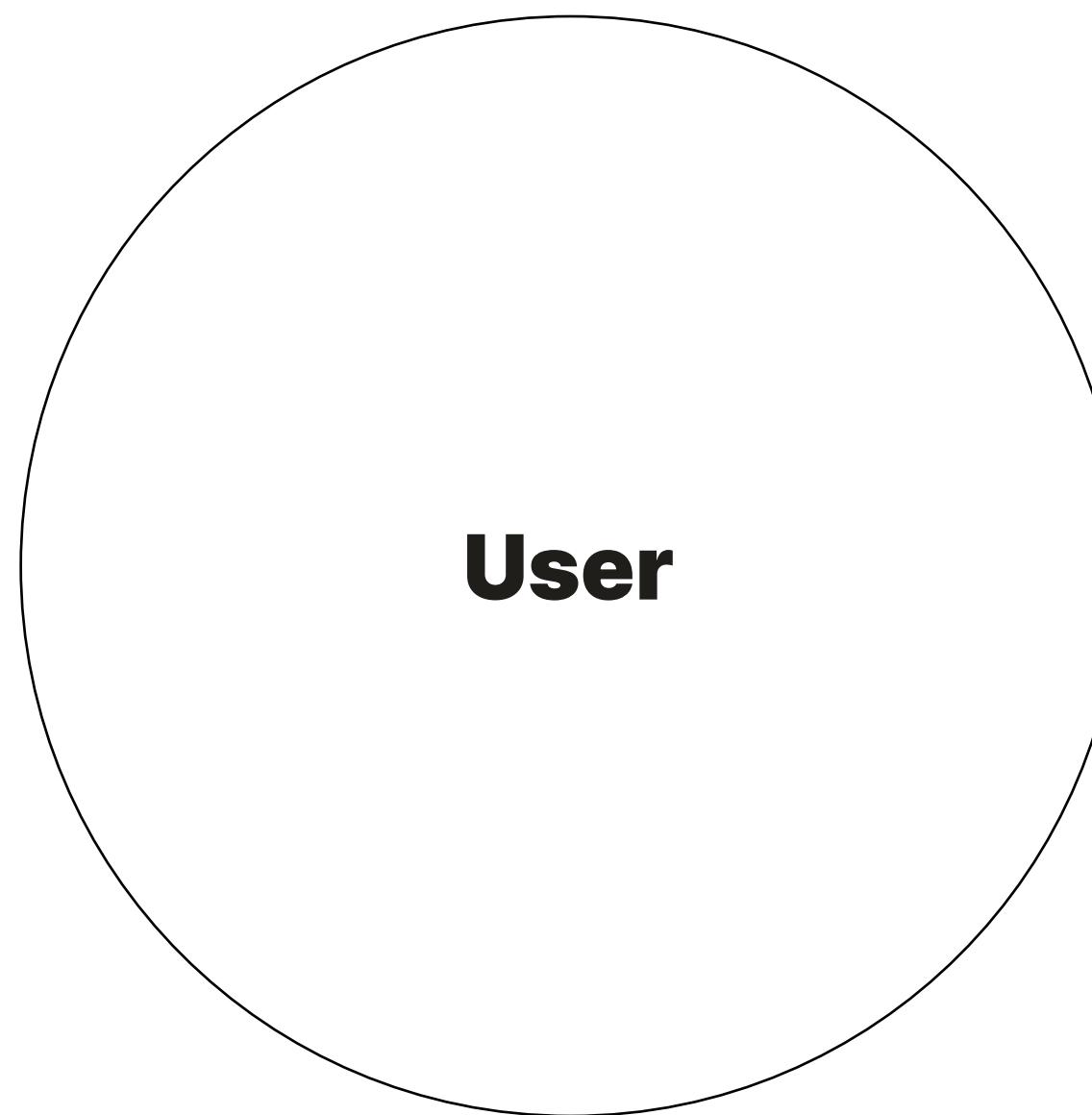
**About 'DunDun'**

## **Conclusion**



**Artist**

**Target**

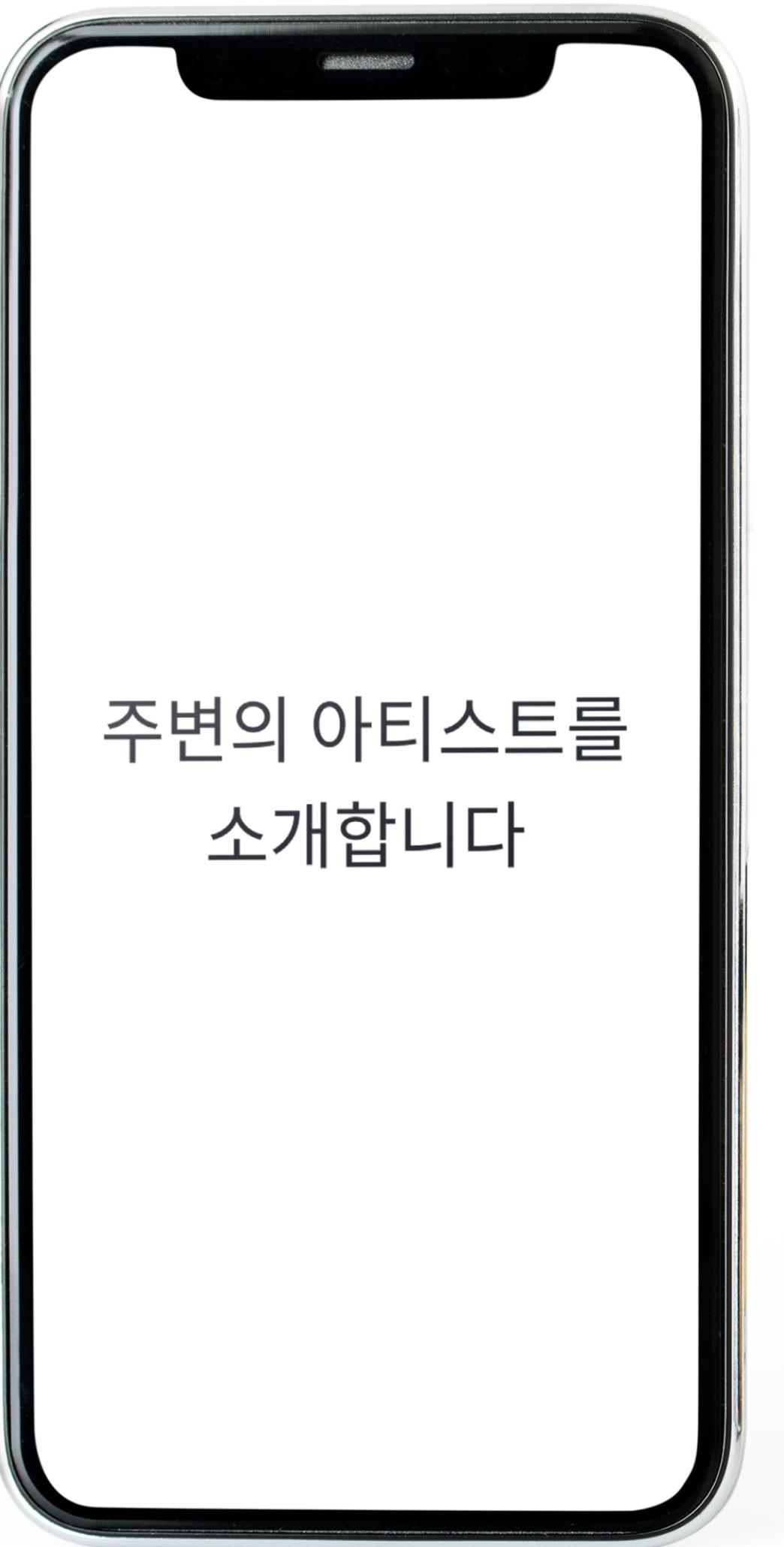


**User**

**About 'DunDun'**

## **Wire Frame**

### **On-Boarding**



주변의 아티스트를  
소개합니다



다음으로 넘어가기



관심있는 아티스트를  
팔로우 해보세요

## About 'DunDun'

# Wire Frame - 공지

**Dun Dun**

OASISU  
2 minutes ago

<공지합니다>

안녕하세요! 밴드 버스커입니다.  
오늘은 오후 6시에 홍대에서 시작하려고 합니다~!  
홍대 버스킹은 처음인데요, 근처 계신분들은 놀러오세요~!😊

BUTTON    BUTTON      

근성은 혼란스럽다.  
(GEUNSEONG'S ROCK)  
4 minutes ago

<공지합니다>

Home    Search    Calendar    Maps    Settings



**OASISU**

안녕하세요? 그룹 사운드 오아시수입니다!  
저희는 브리티쉬 POP을 연주합니다.

Follow

OASISU  
2 minutes ago

<공지합니다>

안녕하세요! 밴드 버스커입니다.  
오늘은 오후 6시에 홍대에서 시작하려고 합니다~!  
홍대 버스킹은 처음인데요, 근처 계신분들은 놀러오세요~!😊

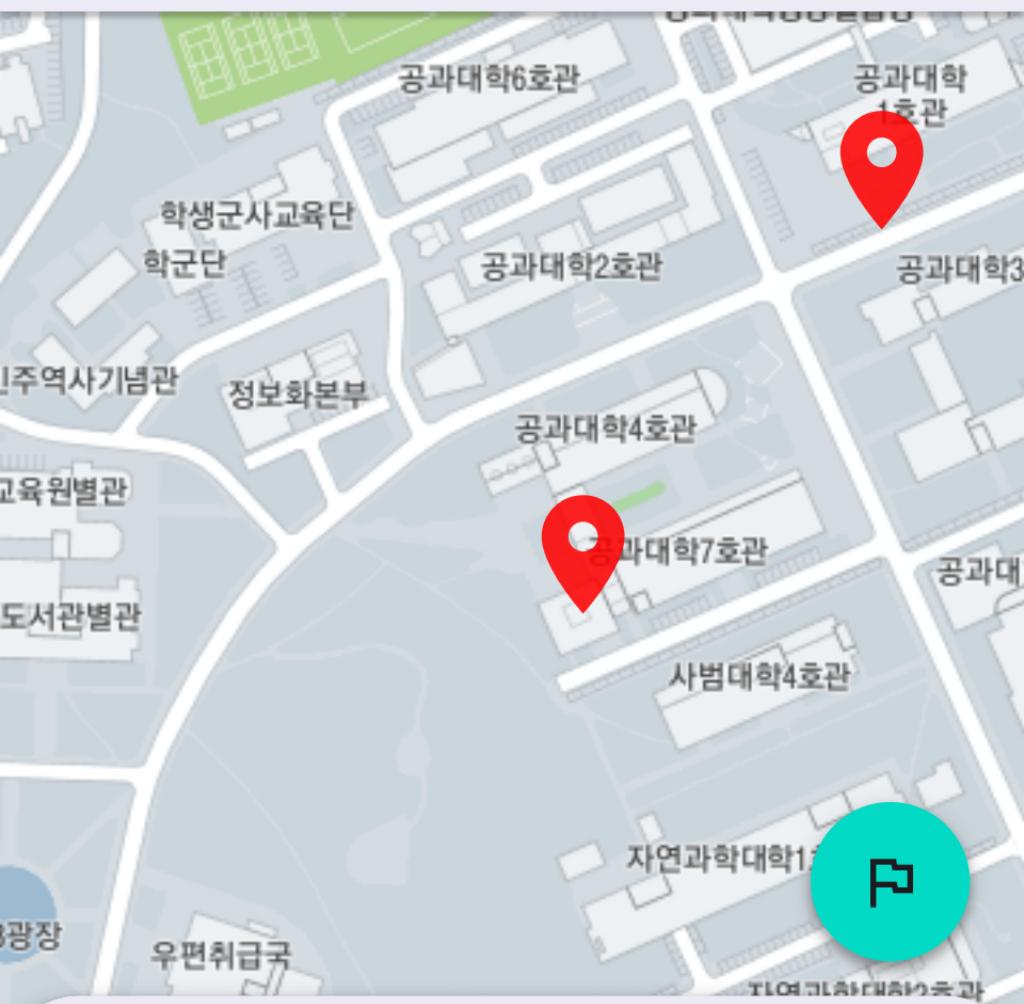
Home    Search    Calendar    Maps    Settings

## About 'DunDun'

## Wire Frame - 지도



# Dun Dun



## OASISU

2023년 5월 14일 07:00 PM

전남대학교 공과대학 7호관 603호

+ more

## 근성은 혼란스럽다

2023년 6월 5일 04:00 PM

전남대학교 농생대학 3호관 309호

+ more

## 밴드이름을고민해보자

2023년 6월 19일 06:00 PM

전남대학교 인공지능대학

+ more

## About 'DunDun'

# Wire Frame - 캘린더



August 2023 ▾

S	M	T	W	T	F	S
			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	31	

### 혁오 정기 공연

📍 홍대 상상마당

📅 08.17(목) 16시 30분

About 'DunDun'

## NEXT STEP

유사 서비스 분석

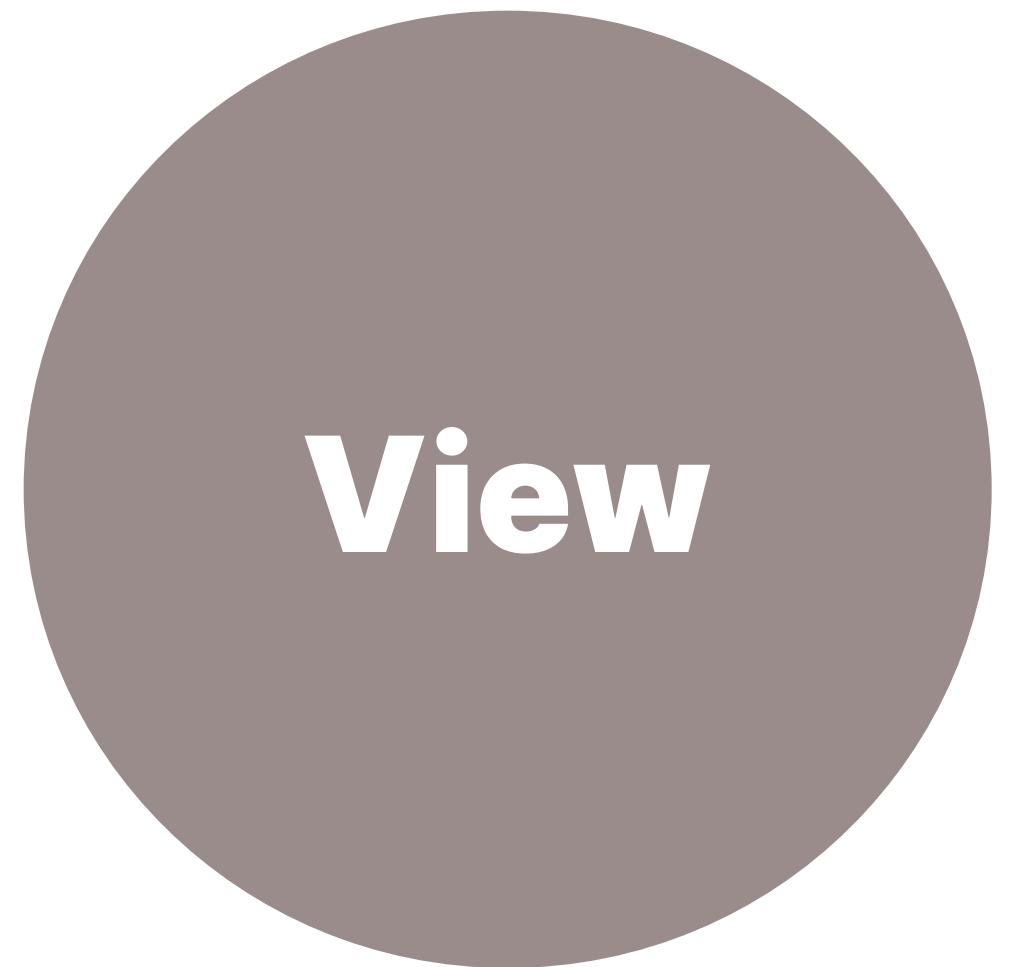
**Business Model**

**User Flow**

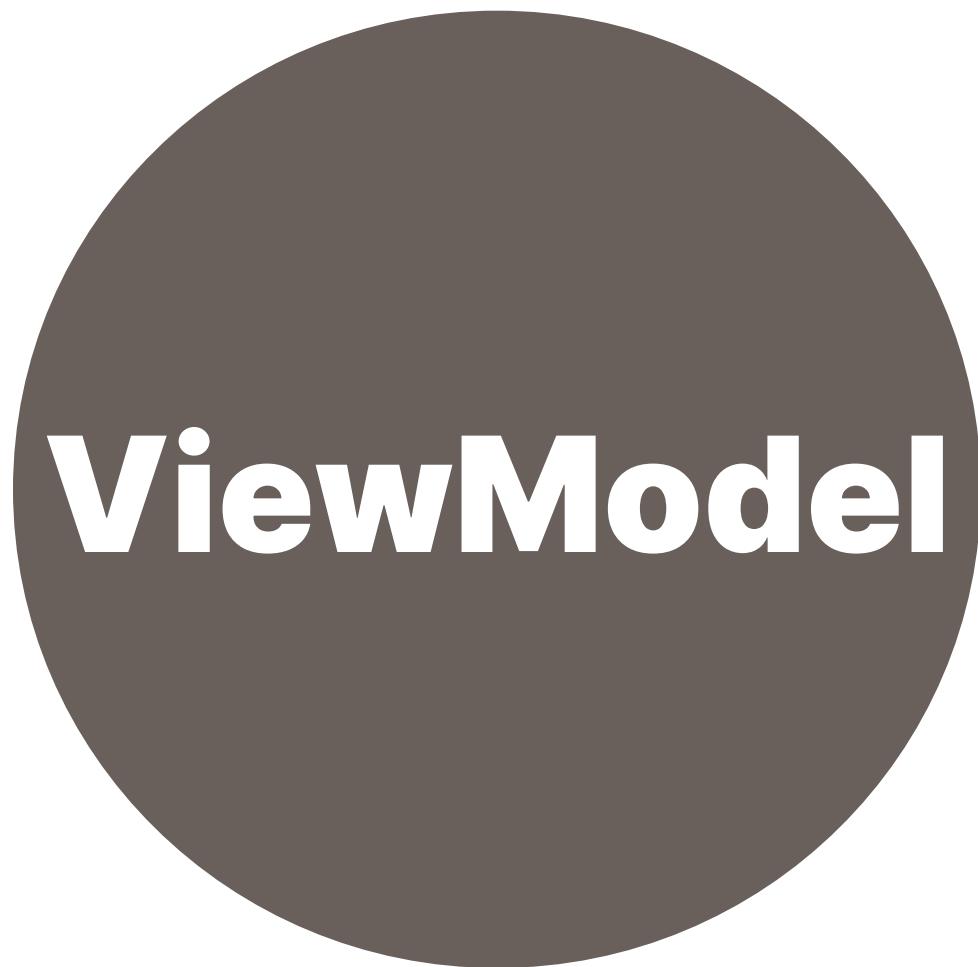
**Wire frame**

**Architecture Design**

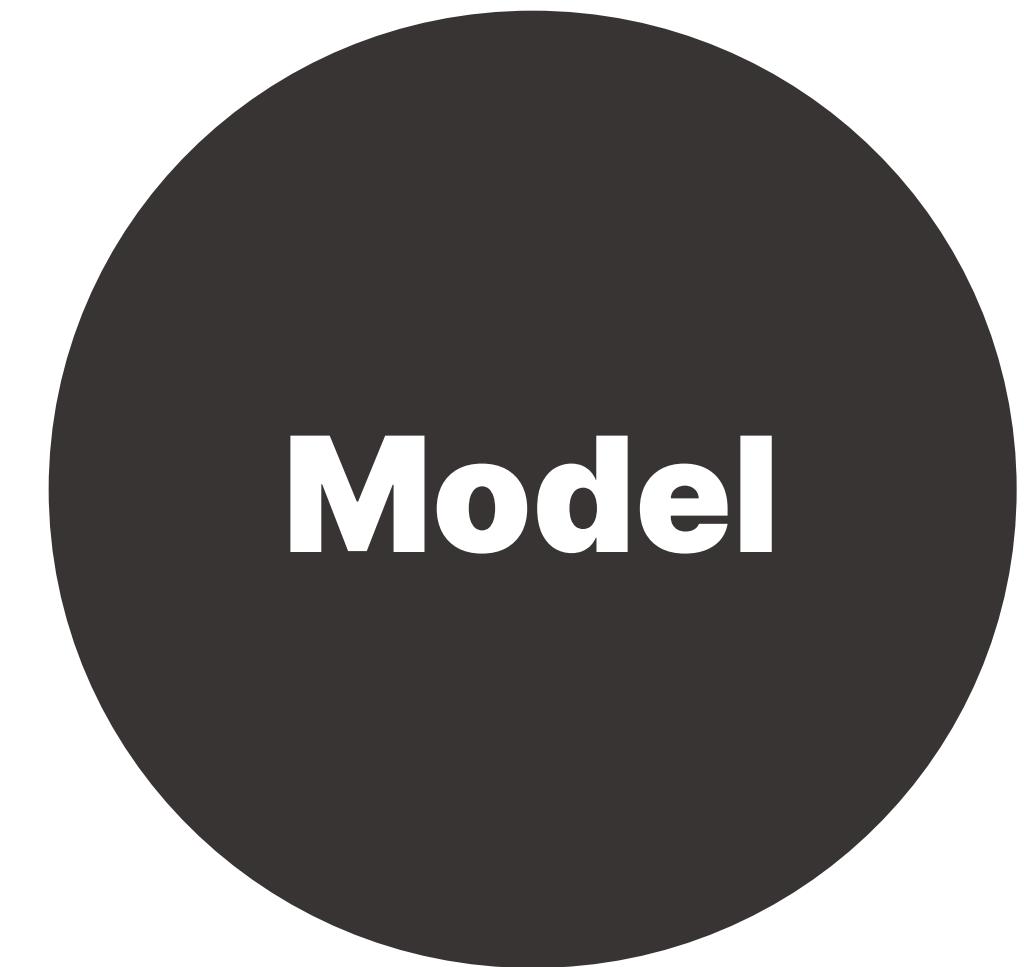
## **Architecture**



**View**



**ViewModel**

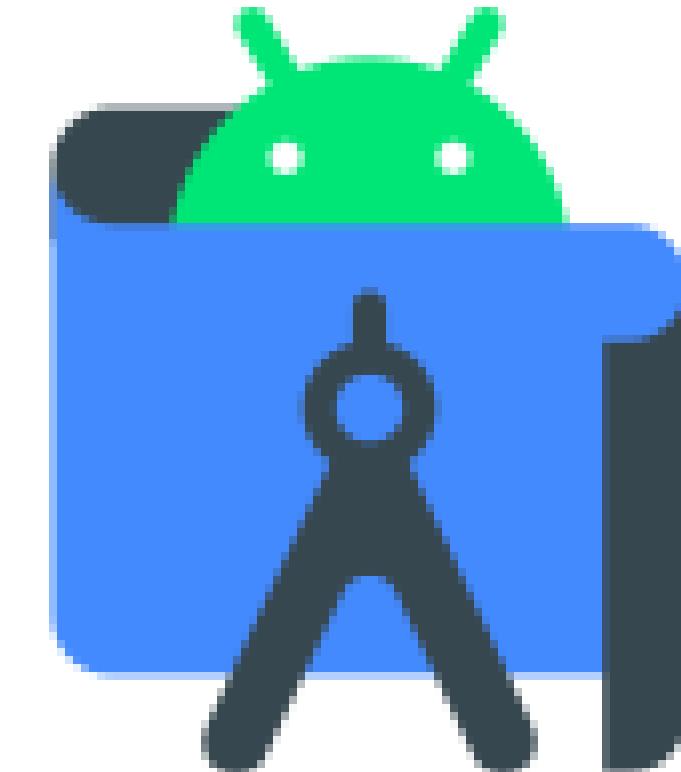


**Model**

**Architecture Design**

**Stack**

 **Kotlin**



**Firebase**

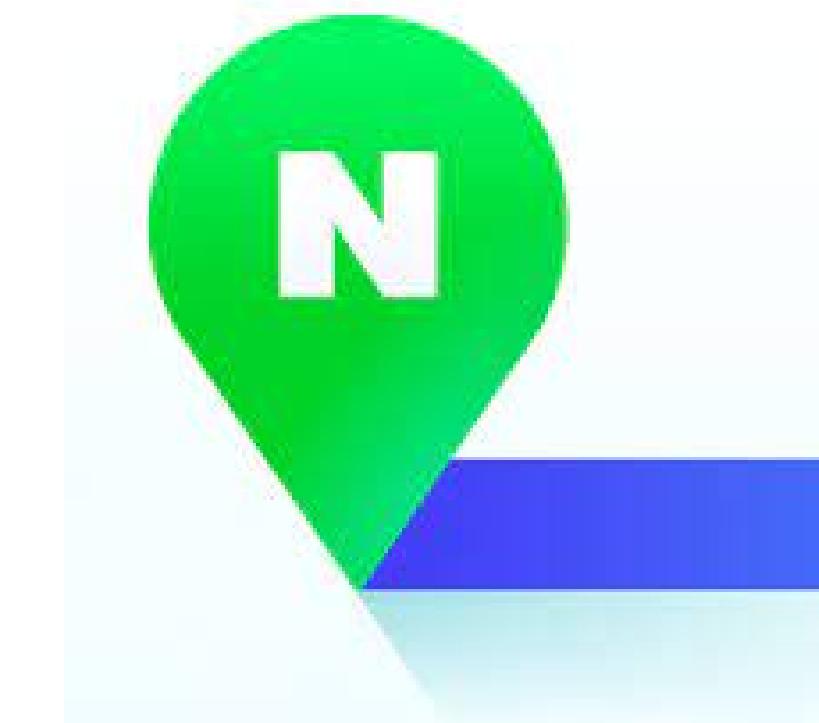
**Architecture Design**

**Liberry**

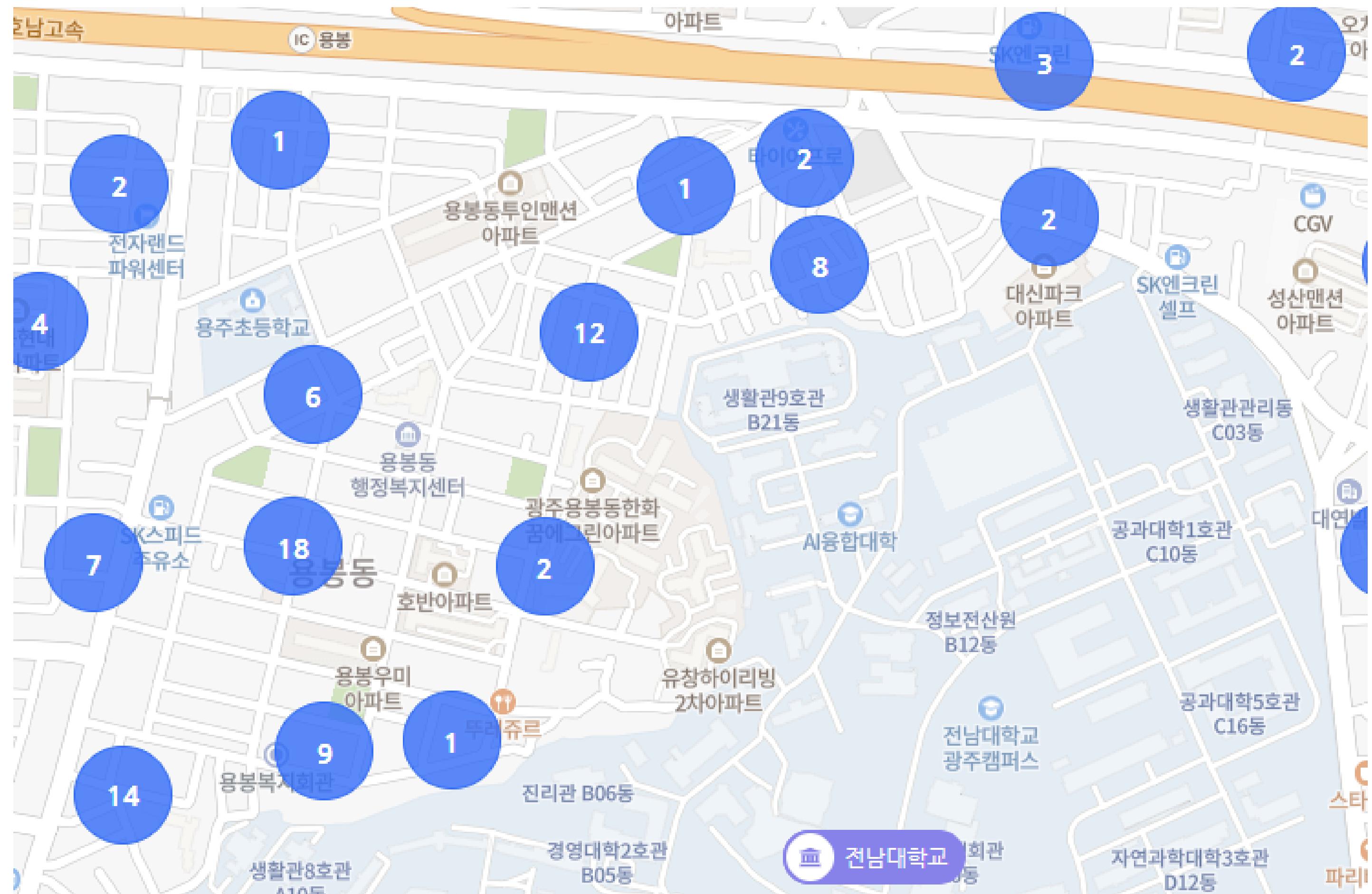
 **Kotlin**

라이브러리 아직  
안적음

 **Firebase**

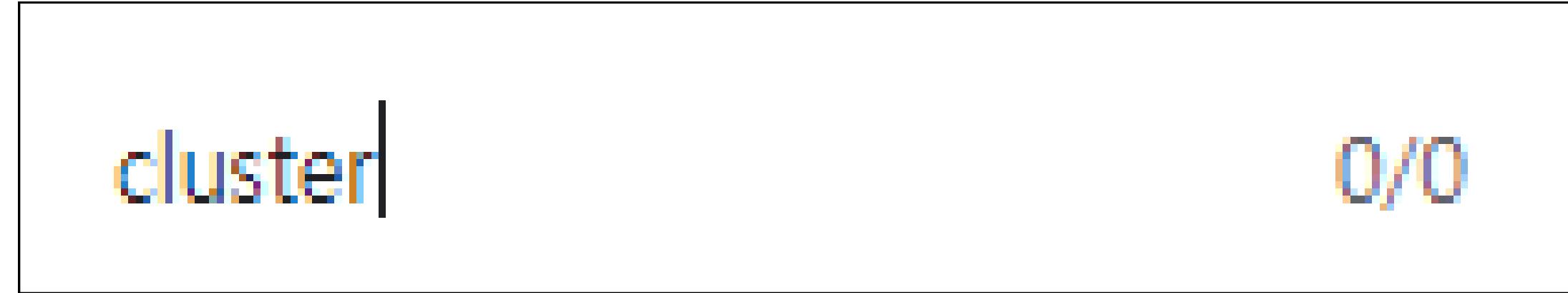


# **kakao map 마커 클러스터링 기술 조사**



**Architecture Design**

## **Architecture**



**android 카카오 지도 api는 클러스터링을 지원하지 않는다.**

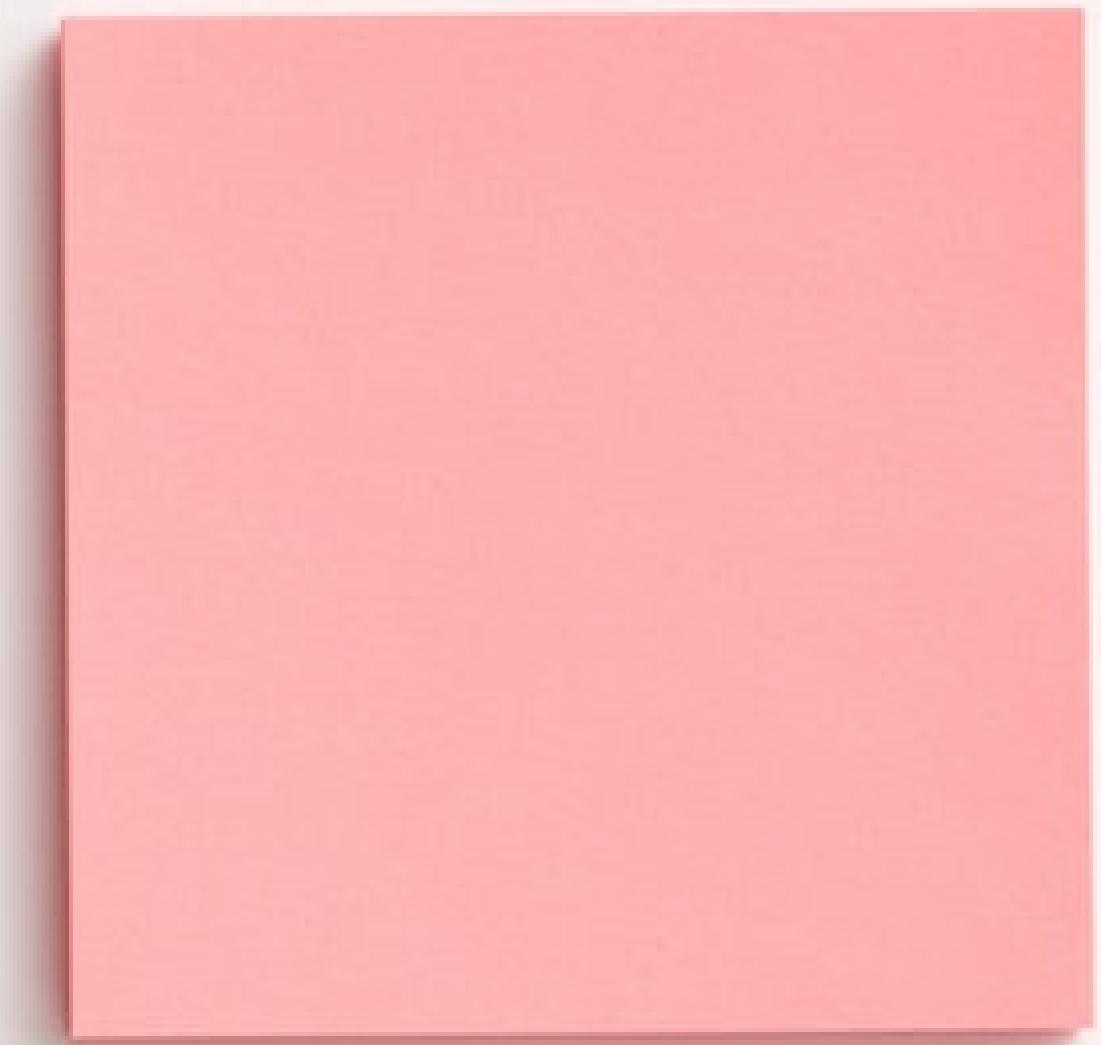
**Architecture Design**

**Architecture**

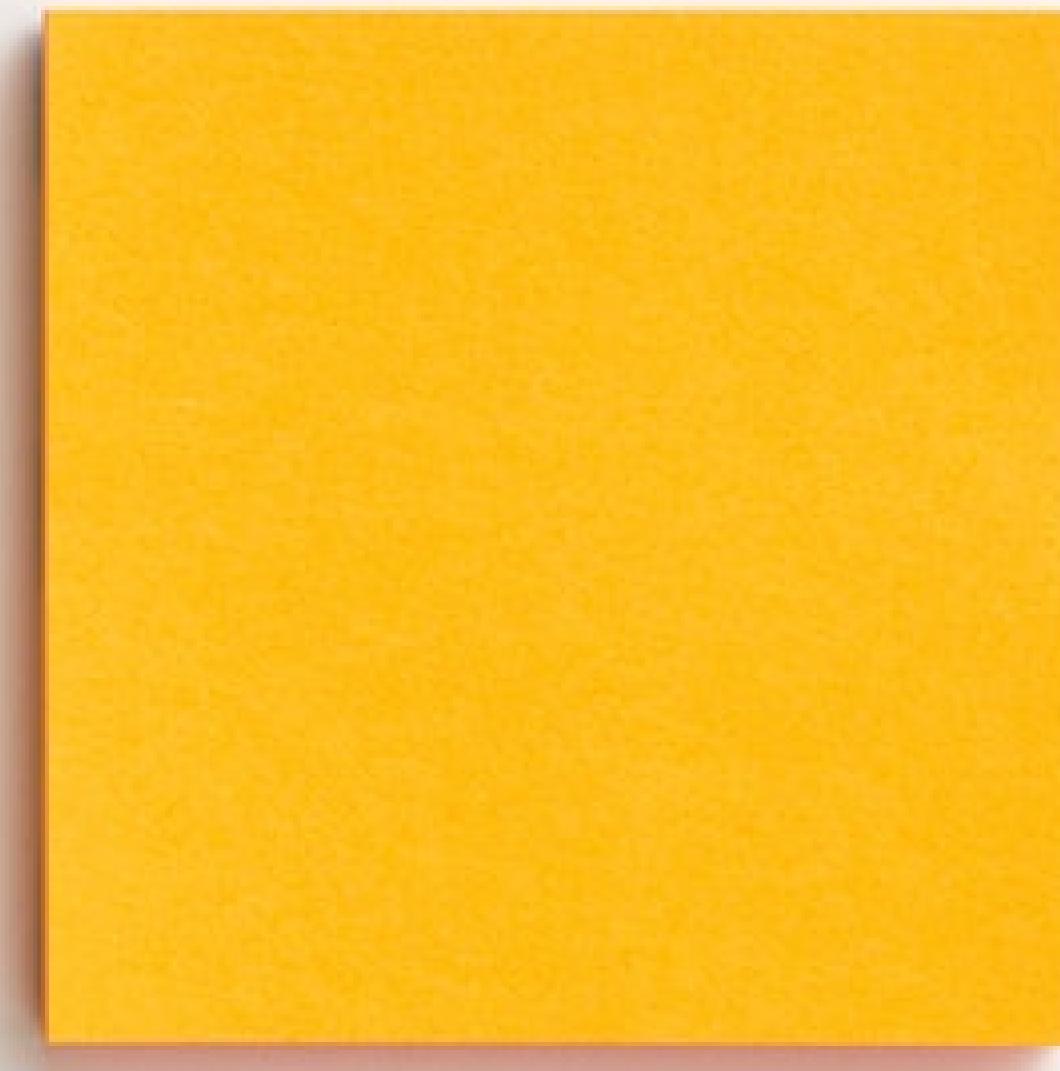
**일반 마커로 표현해보자**



**android 카카오 지도 api는 클러스터링을 지원하지 않는다.**

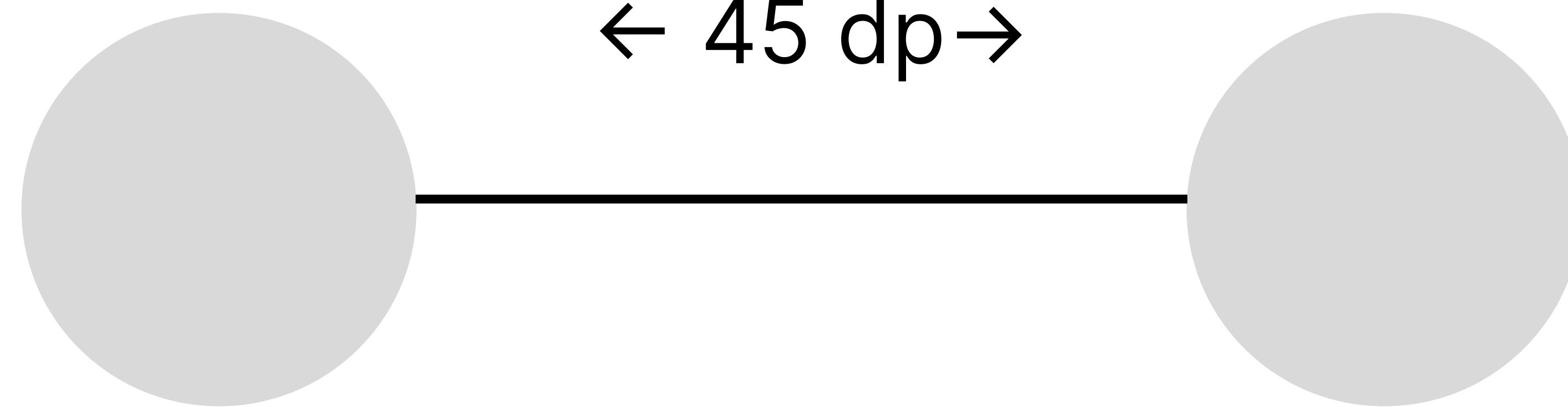


**Architecture Design  
Restrictions for Markers**



**Architecture Design**

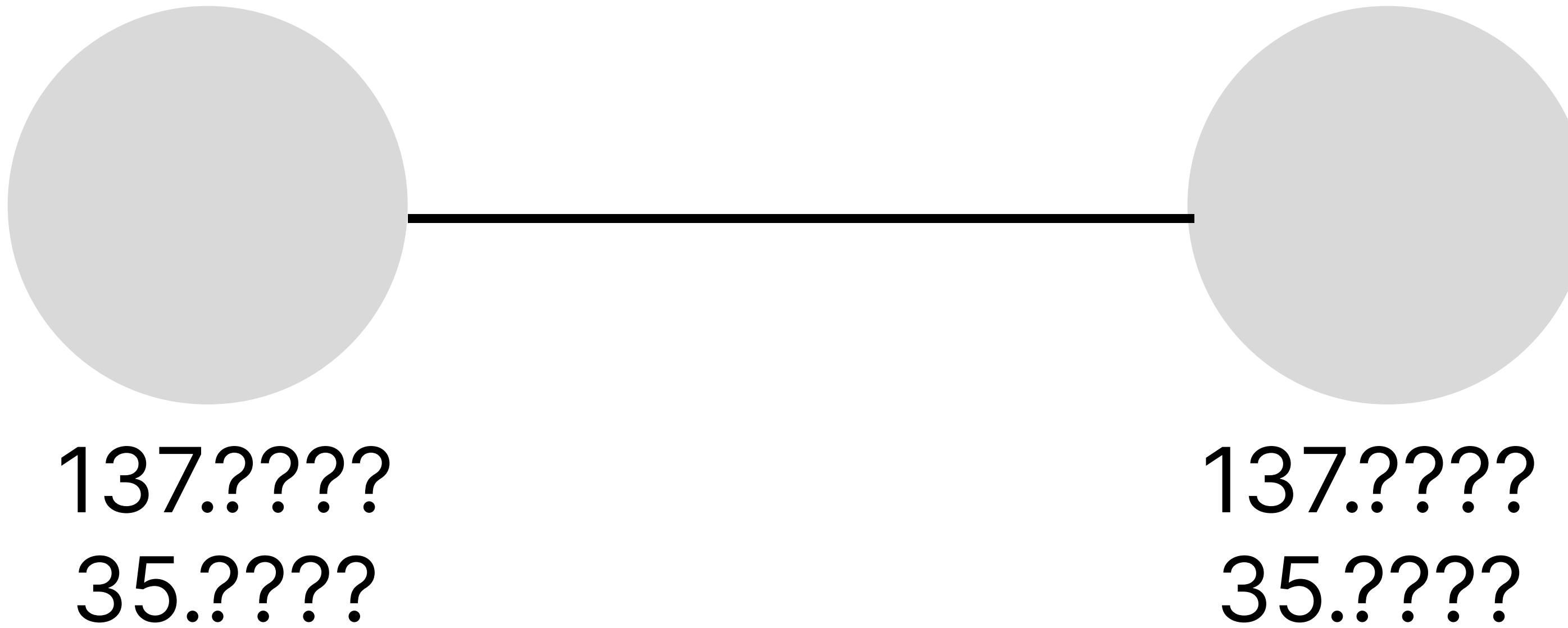
## **Architecture**



**Architecture Design**

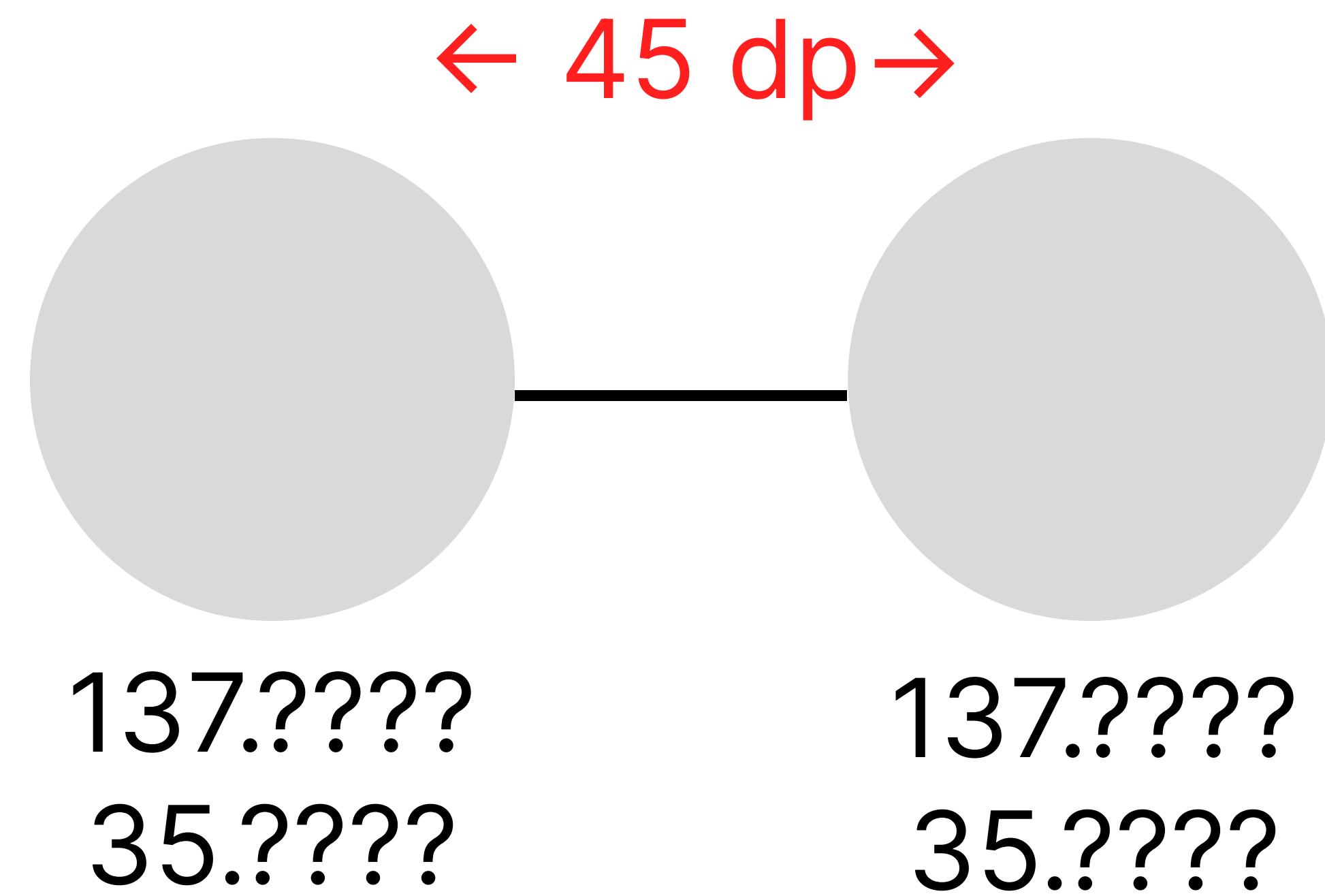
## **Architecture**

← 45 dp →



**Architecture Design**

## **Architecture**



## Architecture Design

### 1. DP와 Pixel의 비율 구하기

```
density = resources.displayMetrics.density
```

## Architecture Design

1. DP와 Pixel의 비율 구하기

2. 45dp를 pixel로 바꾸기

```
density * 55.toDouble()
```

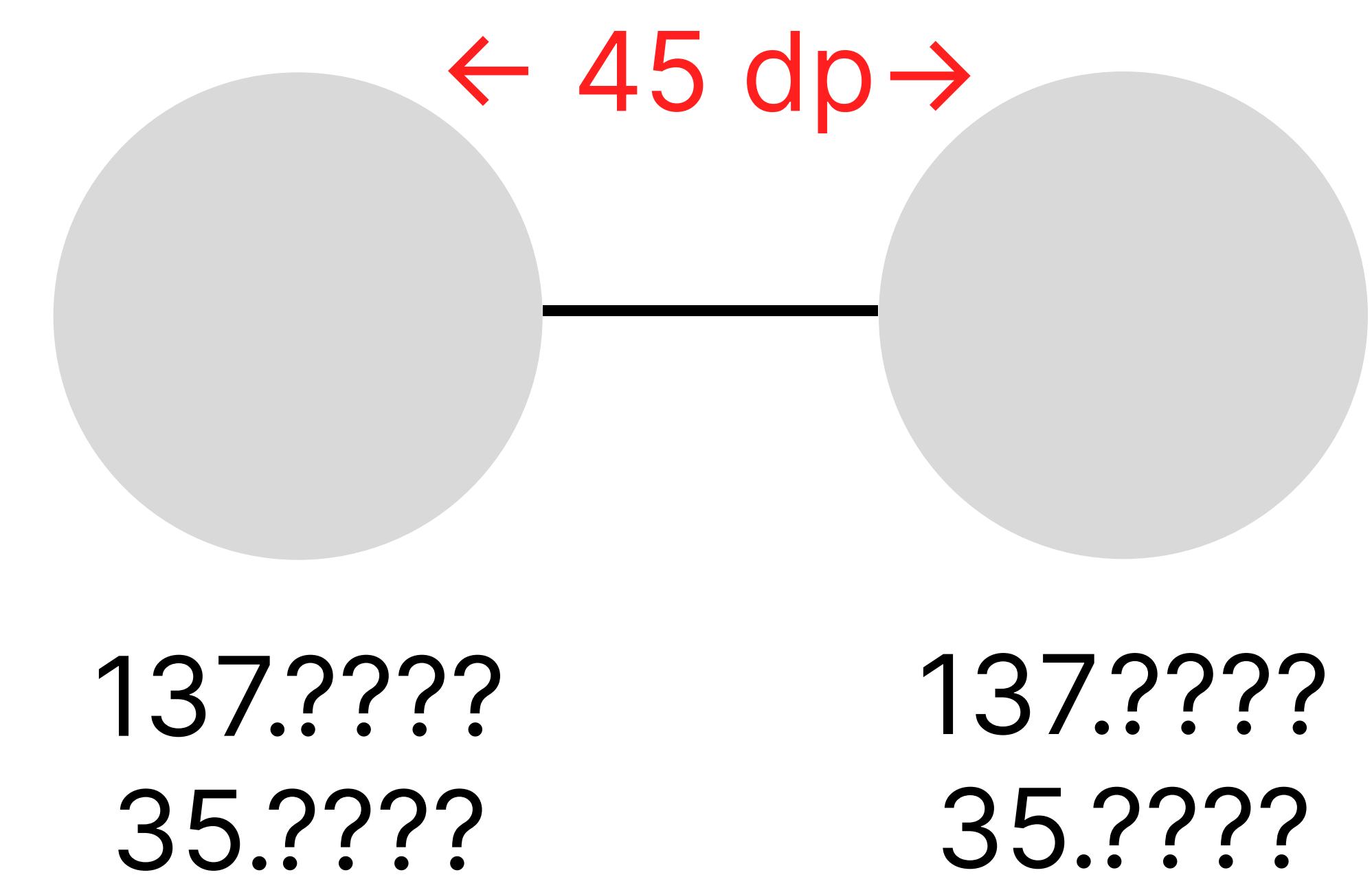
## Architecture Design

1. DP와 Pixel의 비율 구하기
2. 45dp를 pixel로 바꾸기
3. (0,0)의 위경도를 구하고 (0,N)의 위경도 구하기)

```
val temp2 = mapPointWithScreenLocation(  
    xPixel: 0.0,  
    yPixel: 0.0  
).mapPointGeoCoord.Latitude
```

## Architecture Design

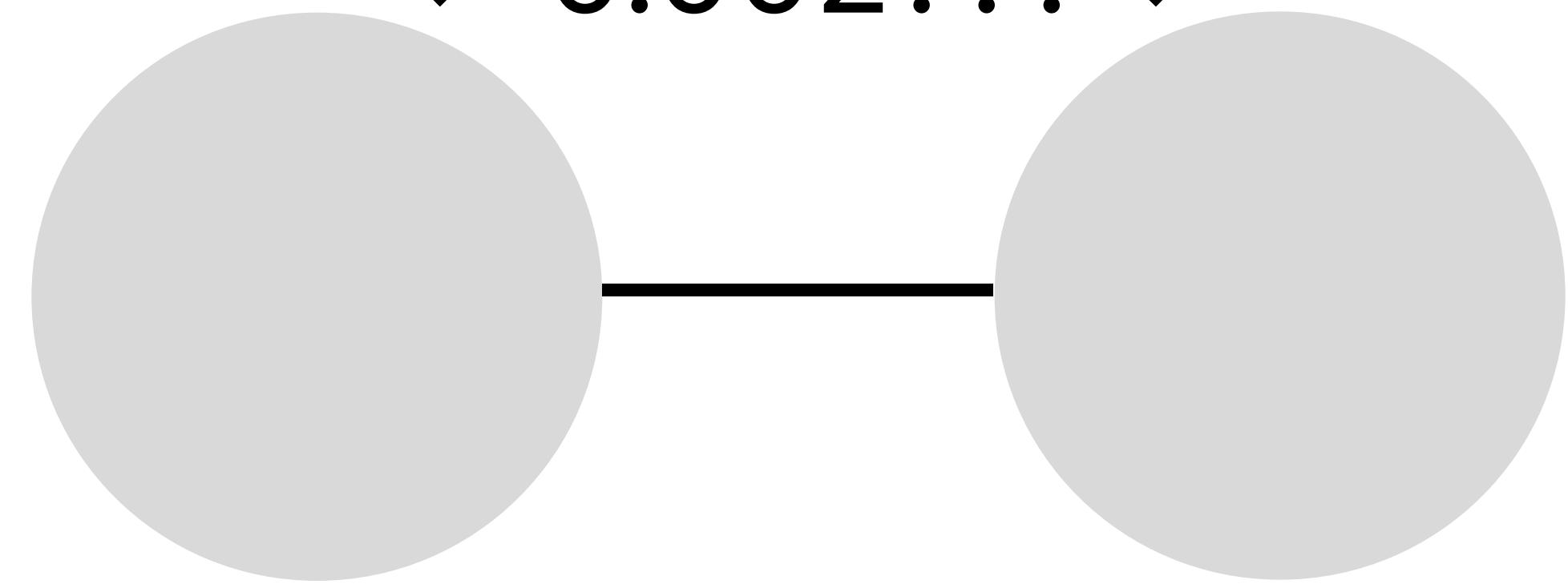
1. DP와 Pixel의 비율 구하기
2. 45dp를 pixel로 바꾸기
3. (0,0)의 위경도를 구하고 (0,N)의 위경도 구하기)
4. 두 값의 차이 = 현재 확대 level에서 45dp만큼의 위경도 값차이



**Architecture Design**

**Architecture**

← 0.002???

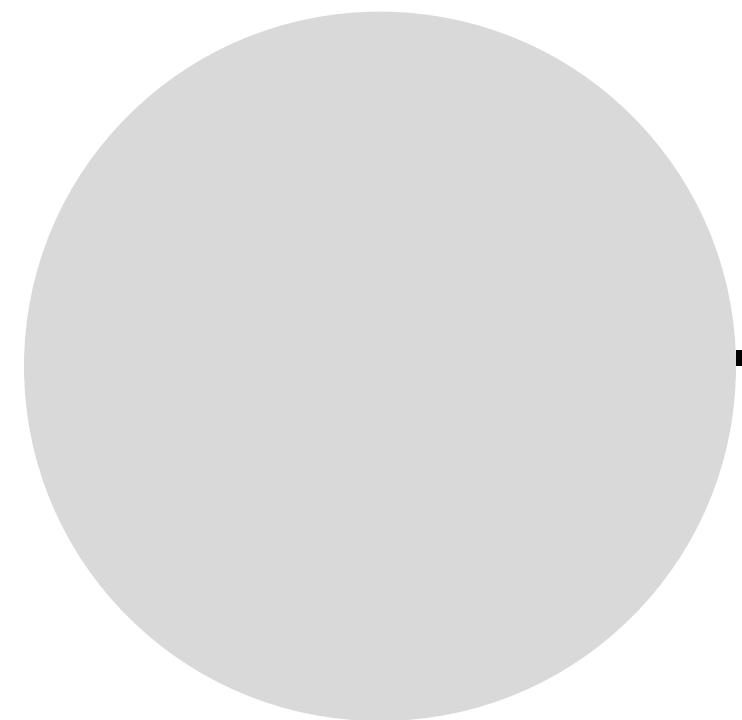


137????  
35????

137????  
35????

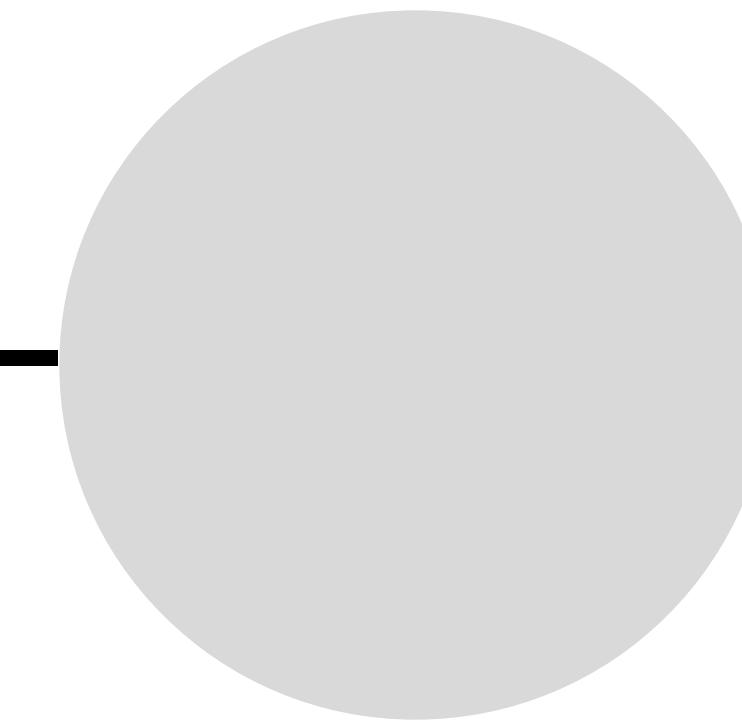
**Architecture Design**

**Architecture**



137????  
35????

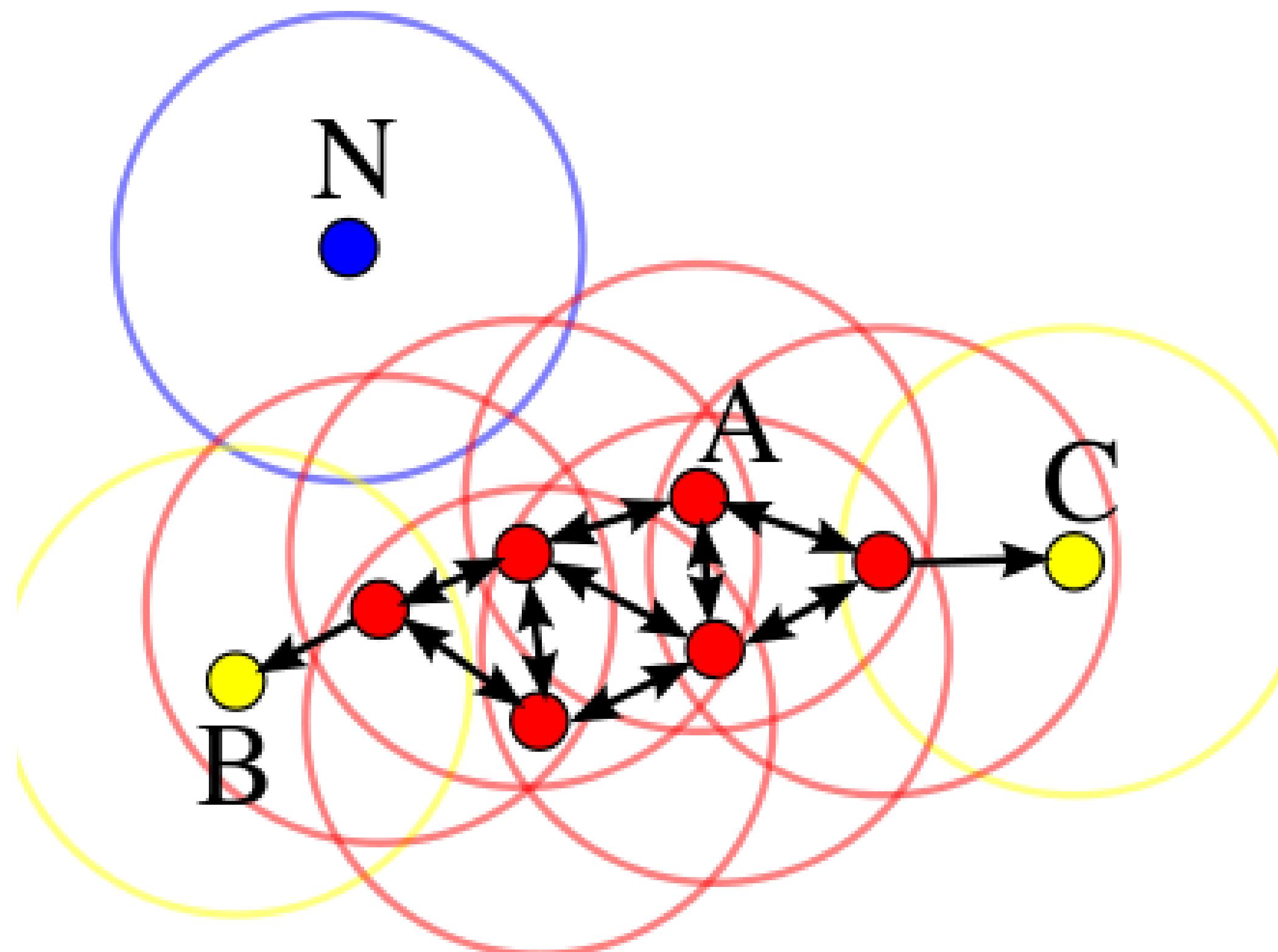
← 0.001???→



137????  
35????

## Architecture Design

# DBSCAN

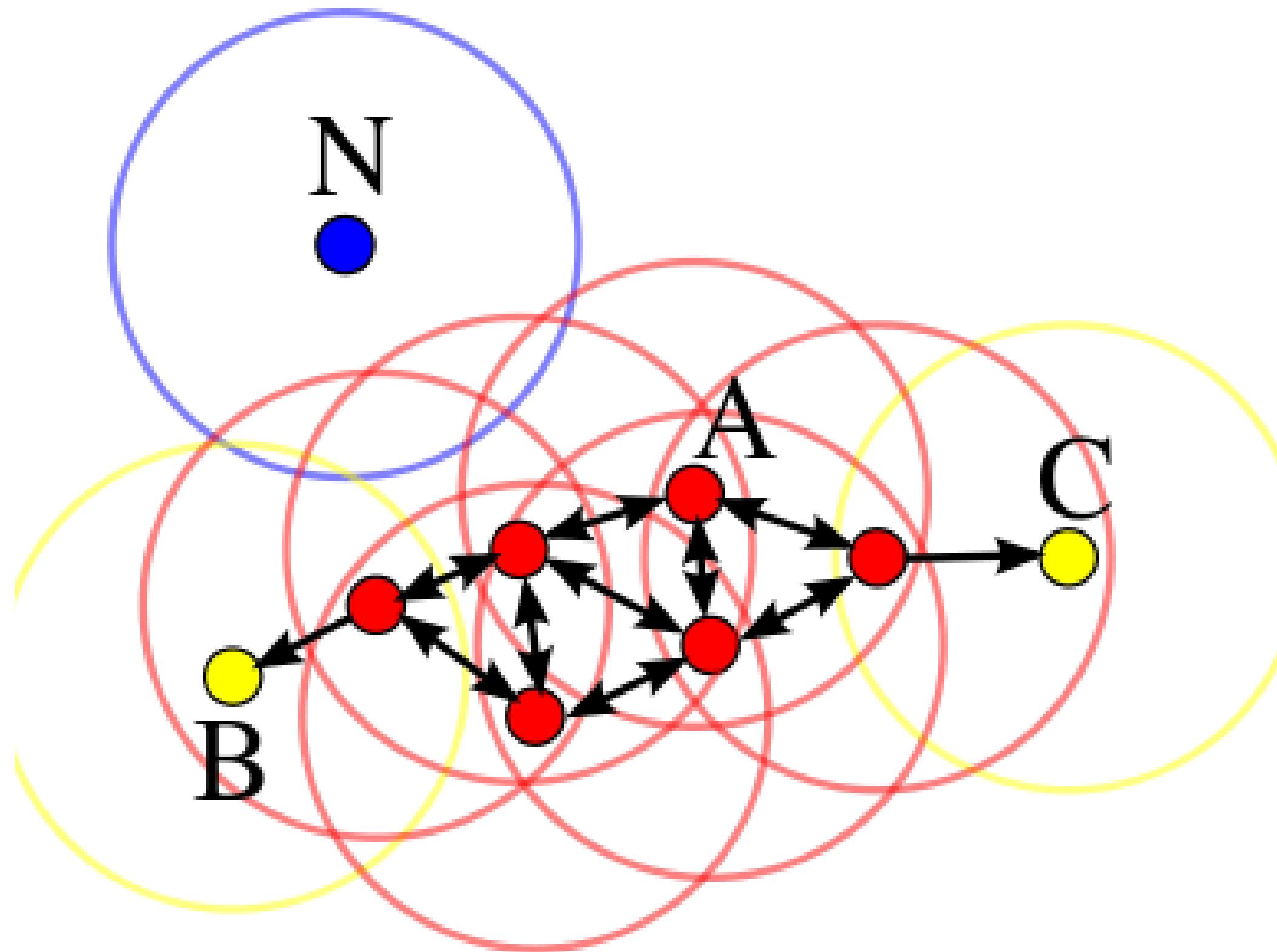


주변에 점이 n개 이상이면 core  
core와 주변점은 한 cluster

core와 core가 만나면  
한 클러스터로 합침

## Architecture Design

# DBSCAN



```
private fun find(x: Int): Int {  
    if (parent[x] == x) {  
        return x;  
    }  
    var y = find(parent[x]);  
    parent[x] = y  
    return y  
}
```

```
private fun check(x: Int, y: Int, d: Double): Boolean {  
    if (x == y) return false  
    val x1: Double = v[x].x  
    val x2: Double = v[y].x  
    val y1: Double = v[x].y  
    val y2: Double = v[y].y  
    return sqrt(x: pow(a: x1 - x2) + pow(a: y1 - y2)) <= d  
}
```

```
private fun merge(x: Int, y: Int) {  
    val a = find(x)  
    val b = find(y)  
    parent[b] = a  
}
```

```
fun clustering(d: Double): List<Pair<Int, Cluster>> {  
    parent.clear()  
    label.clear()  
    for (i in v.indices) {  
        parent.add(i)  
        label.add(0)  
    }  
  
    for (i in v.indices) {  
        var Cnt = 0  
        for (j in v.indices) {  
            if (check(i, j, d)) {  
                Cnt++  
            }  
        }  
        if (Cnt >= 1) {  
            label[i] = 1  
            for (j in v.indices) {  
                if (label[j] == 1) {  
                    if (check(i, j, d)) {  
                        merge(i, j)  
                    }  
                } else {  
                    if (check(i, j, d)) {  
                        label[j] = 2  
                        merge(i, j)  
                    }  
                }  
            }  
        } else {  
            label[i] = 3  
        }  
    }  
    for (i in v.indices) {  
        find(i)  
    }  
}
```

