

삼성 청년 SW 아카데미 8기 A403

[마루] 가장 높은 자리, 으뜸자리

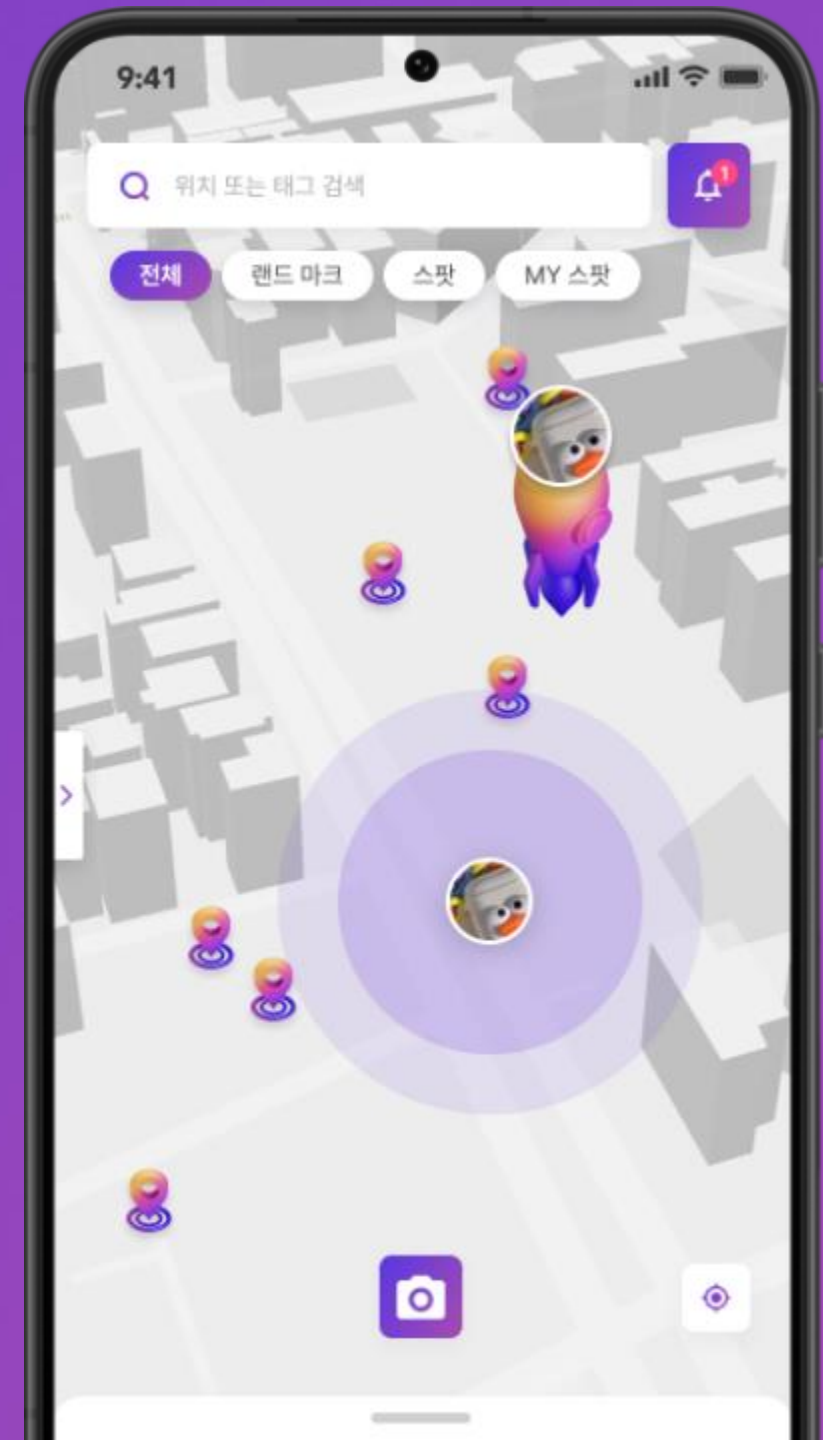
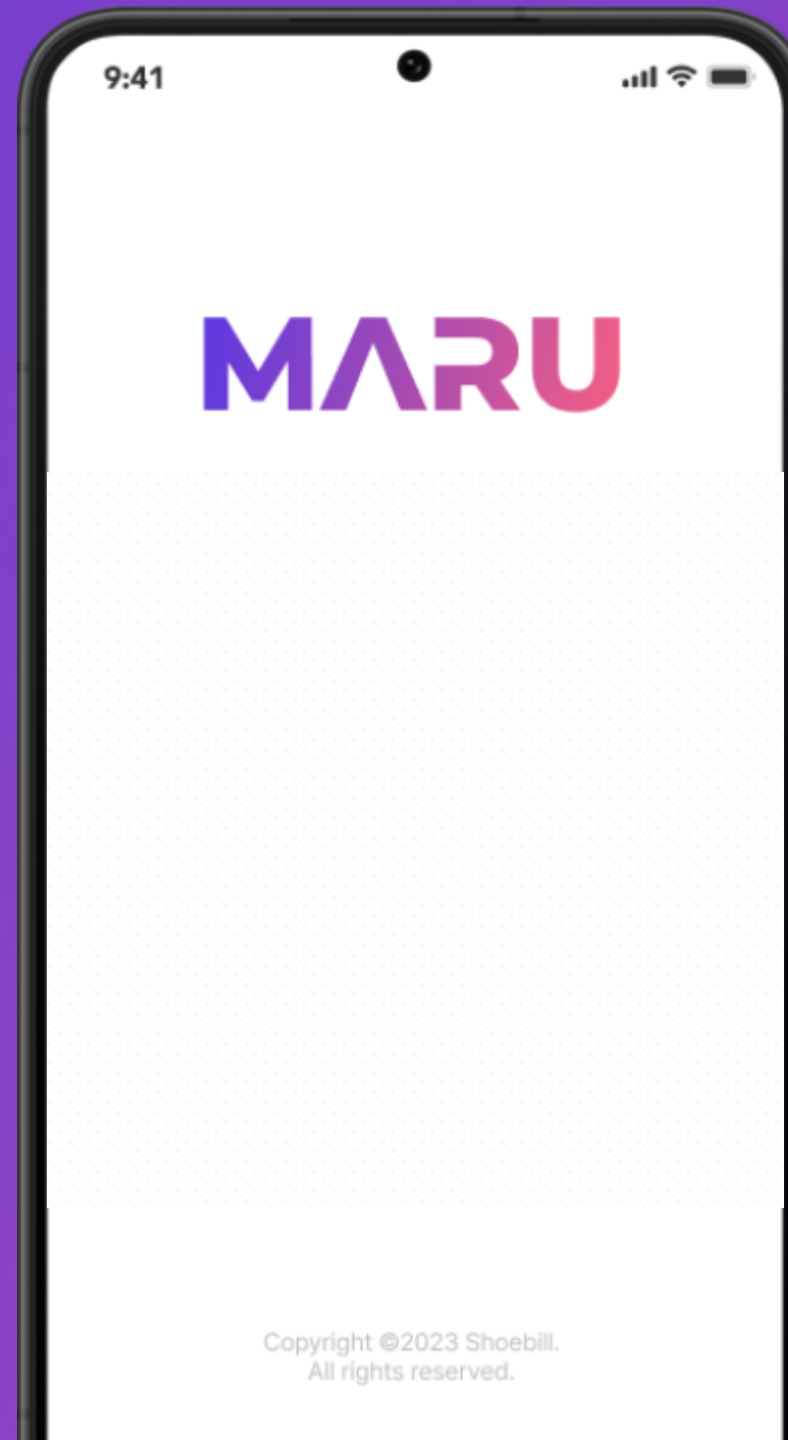
# MARU

서울에 세워보는 나만의 랜드마크

뱀새 쫓는 황새팀

이영차 배창민 이다운

이에진 이호준 황준현



# Index

1. UCC
2. 기획 배경
3. 주요 기능
4. 문제 해결
5. 산출물
6. 팀원 소개



매일 같이 오가는  
**집. 회사.**  
지겹지 않으신가요??



# 1. 기획 배경

문제 인식

MARU



그렇다고 또,  
마음 먹고 훌쩍 떠나기가  
쉽지는 않아요.

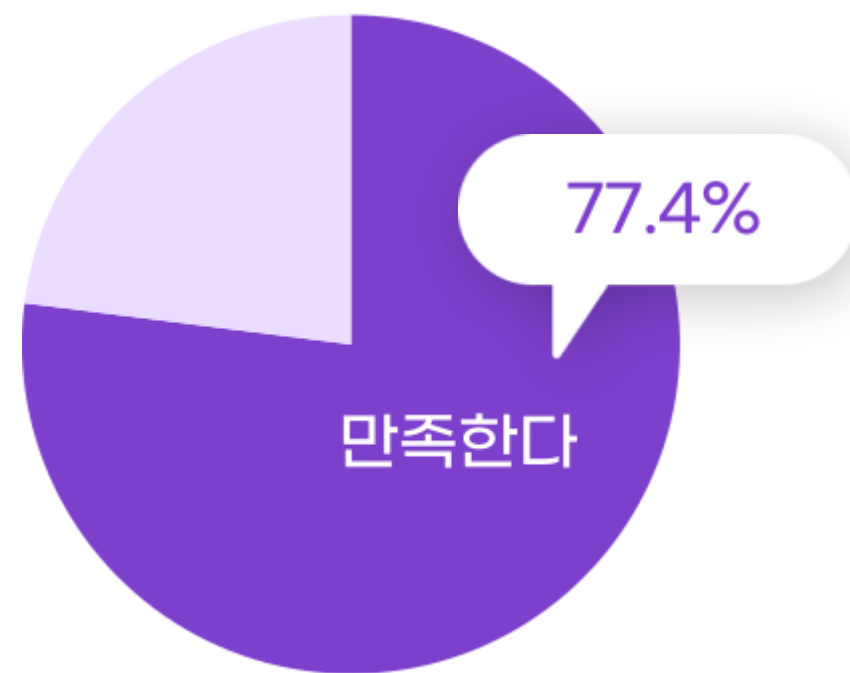


# 1. 기획 배경

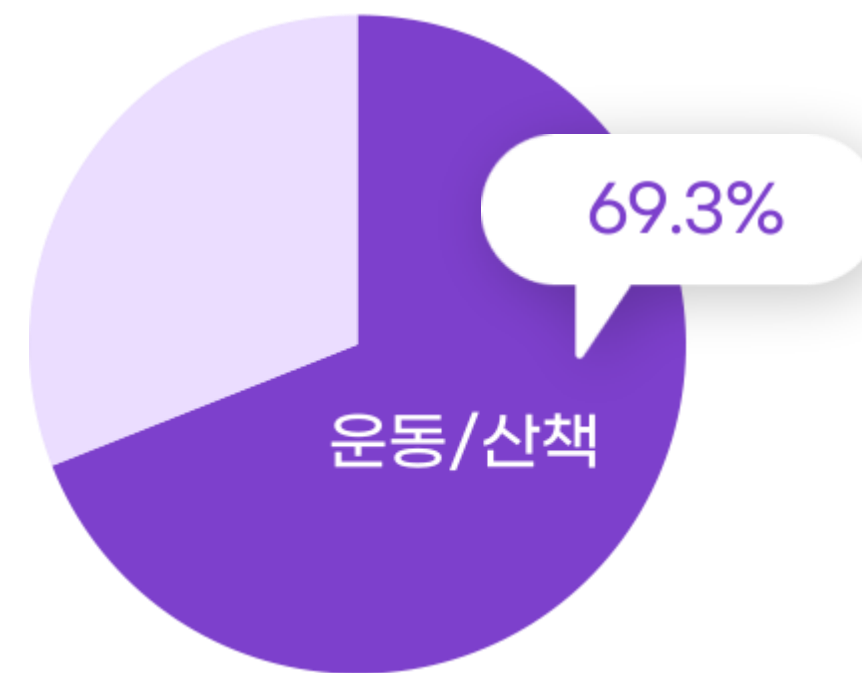
시장 분석

MARU

| 동네 만족도 조사 결과



| 동네에서 주로 하는 활동 조사 결과



“

**거주지에서의 활동을 선호하는 사람들 多**

”



# 1. 기획 배경

타겟팅

MARU

맨날 집, 회사만 오가는  
동네 산책을 즐기는



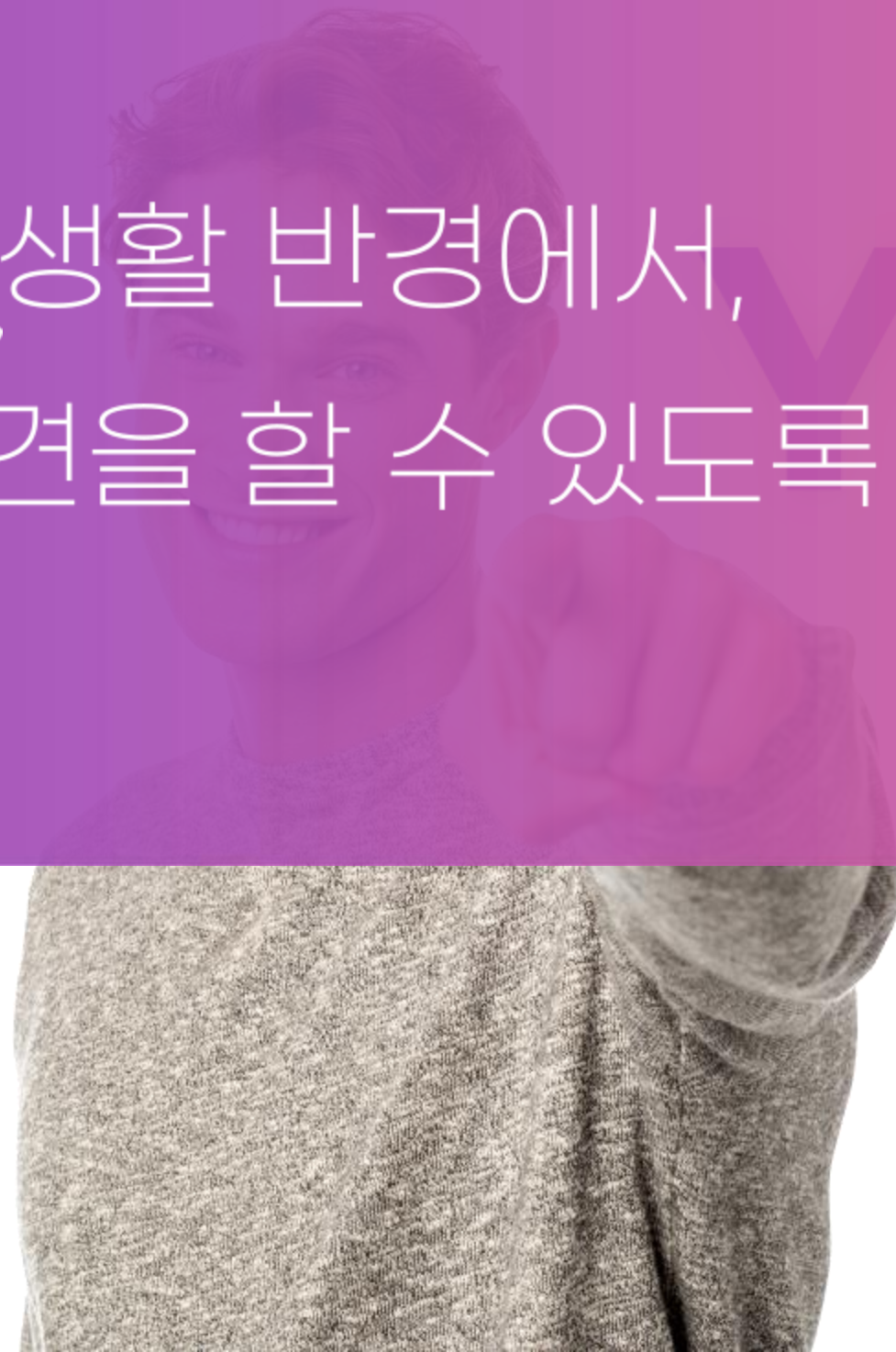
YOU

# 1. 기획 배경

타겟팅

MARU

익숙한 생활 반경에서,  
새로운 발견을 할 수 있도록





## 상호작용의 매개체



사람과 지역의 상호작용의 극대화를 위해 **게이미피케이션** 도입



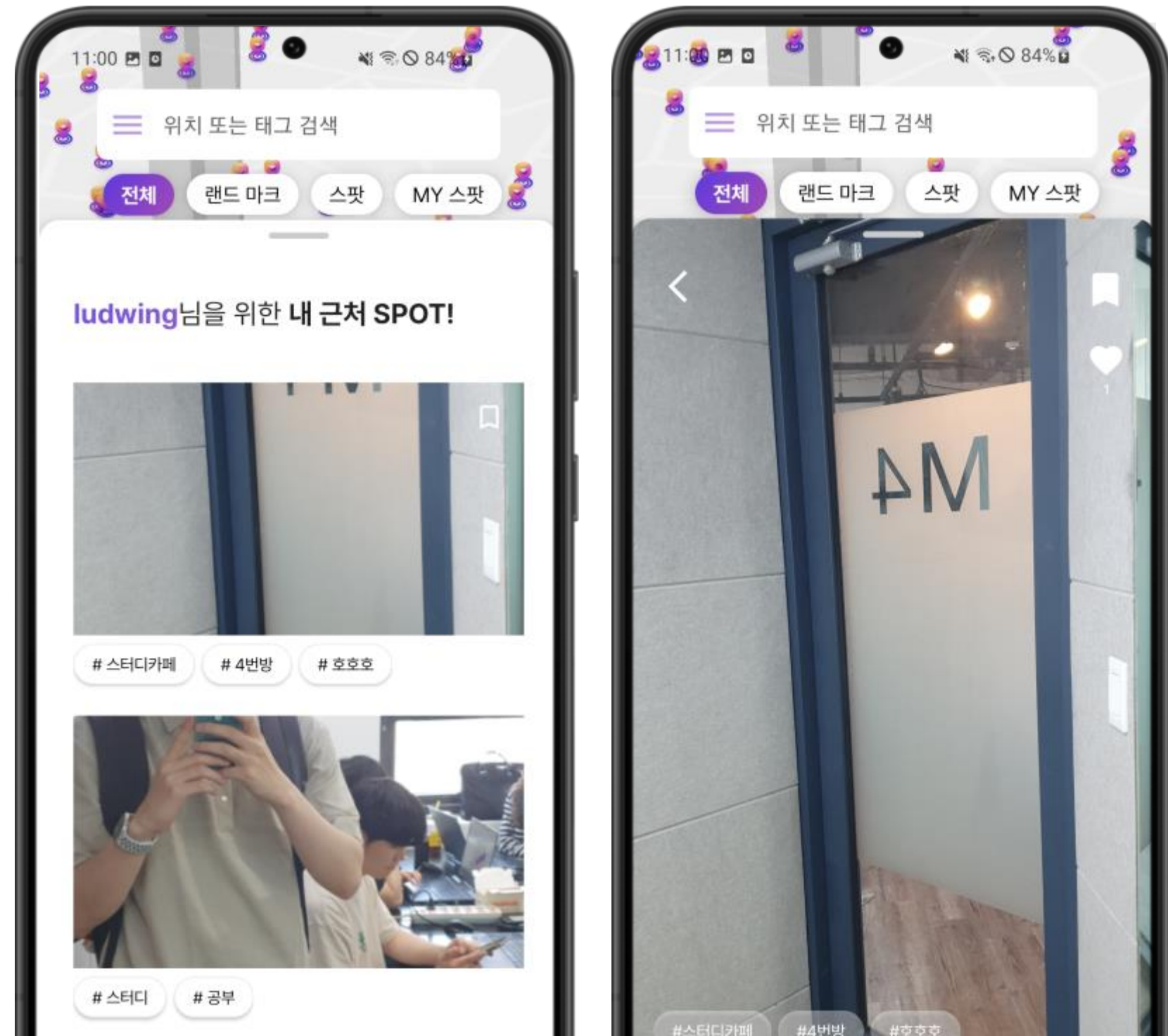
**사람과 사람의 상호 작용**으로 확장

## 3. 주요 기능 장소 공유

MARU

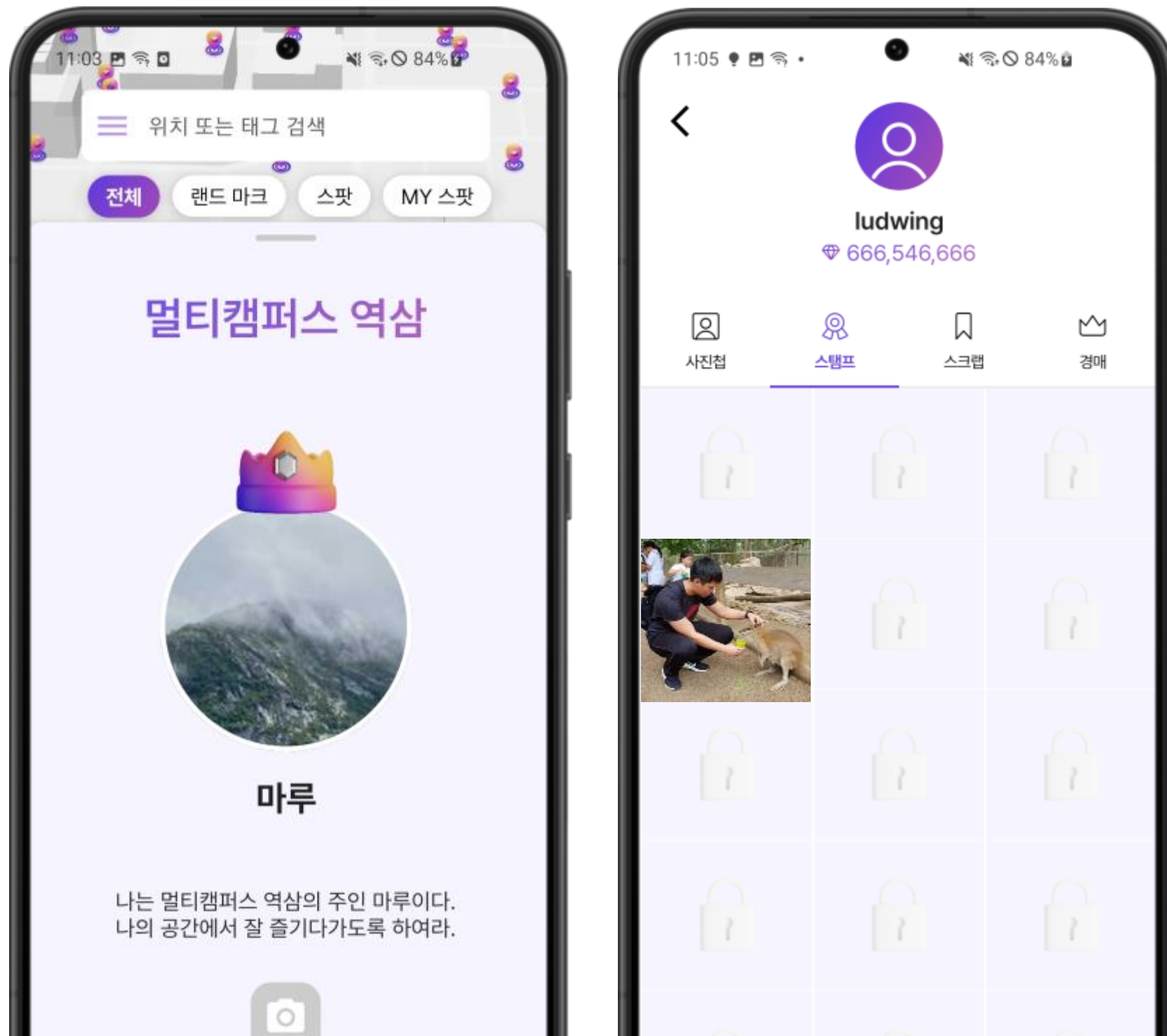
### | 장소 및 사진 공유

- 자신의 특별한 장소를 쉽게 공유
- 앱 지도에서 다른 사람이 공유한 장소 확인 후 방문



### 3. 주요 기능 랜드마크

MARU



#### 랜드마크 방문

- 랜드마크 스탬프 채우기
- 랜드마크 점유로 자신의 사진 알리기

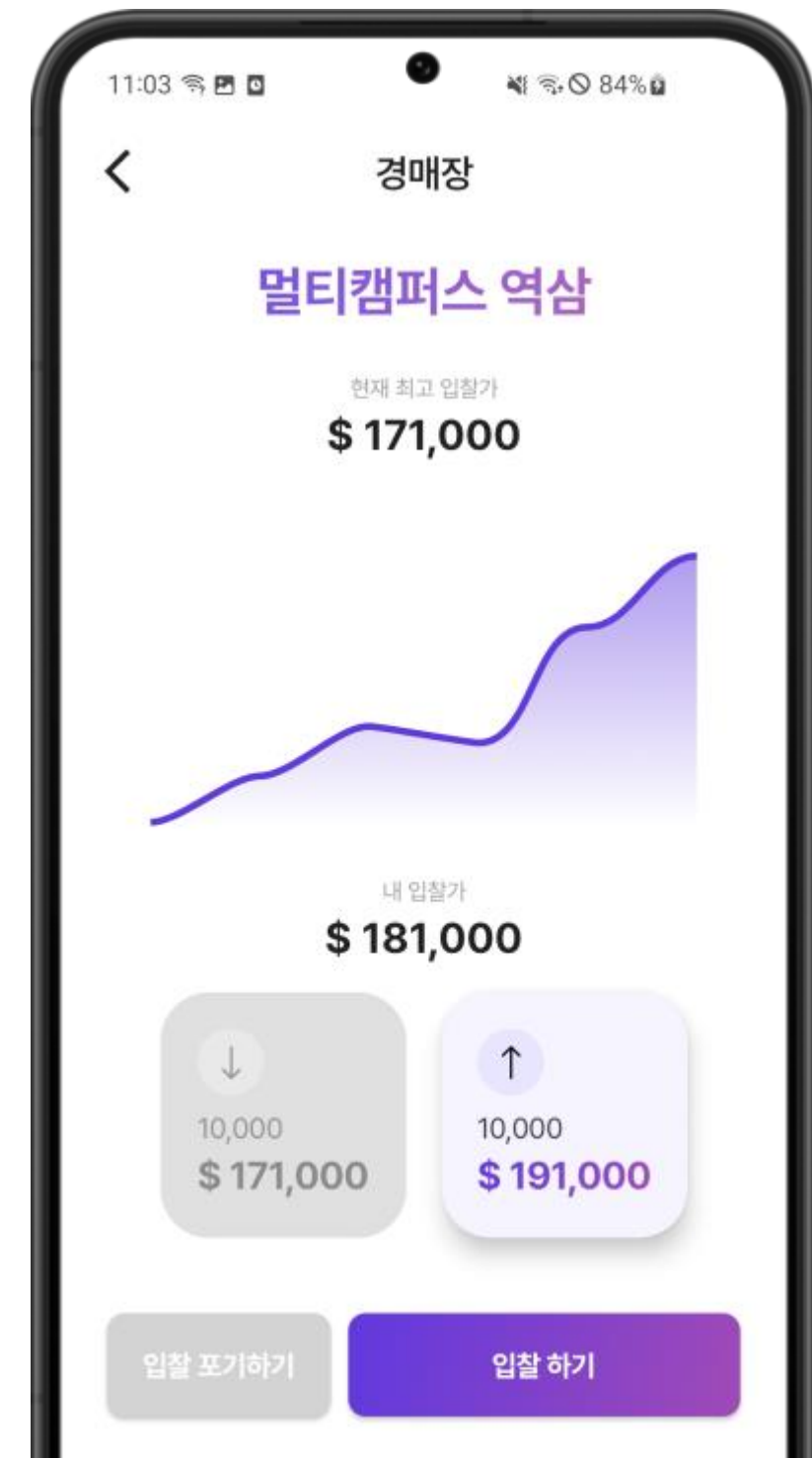
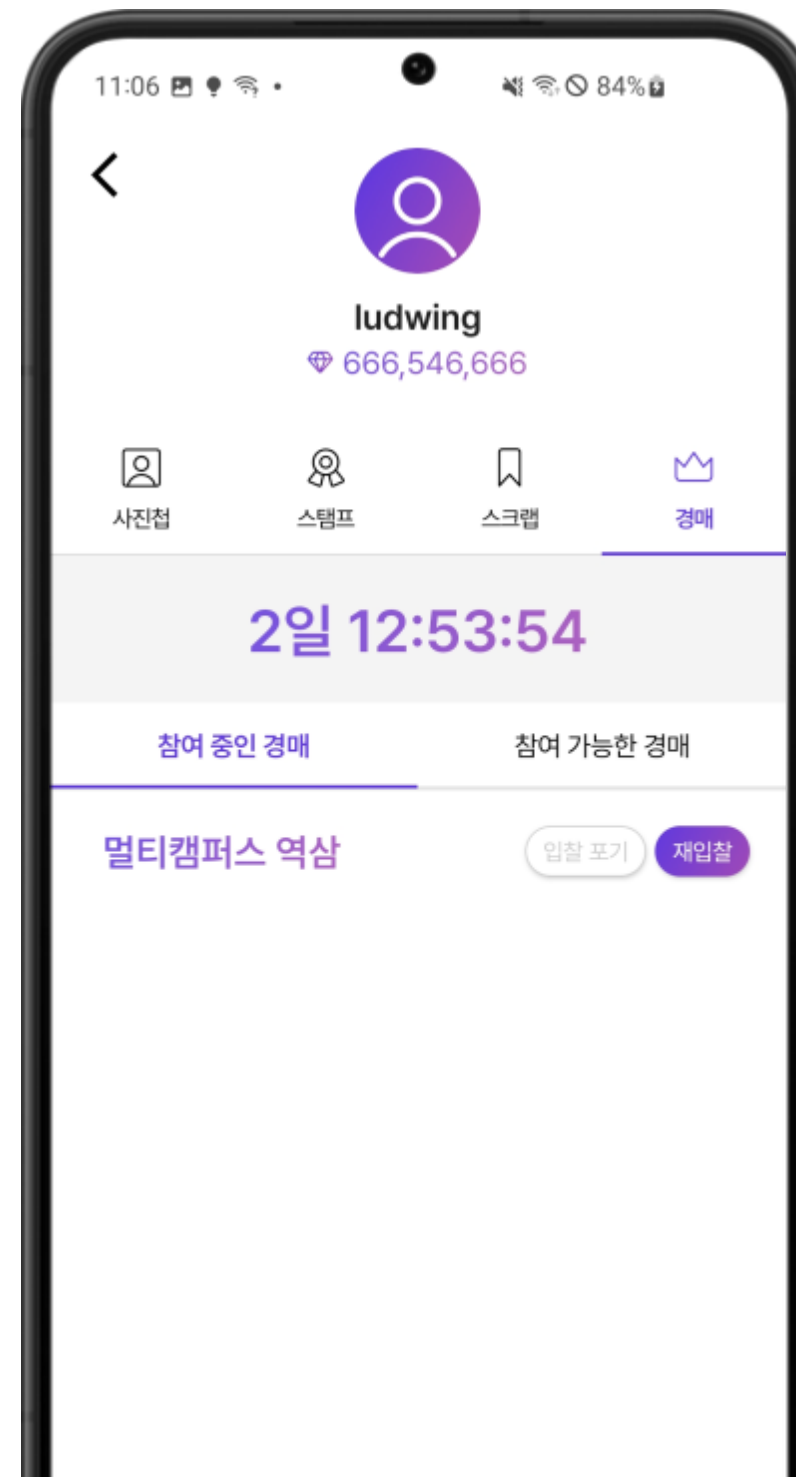
## 3. 주요 기능

경매 참여

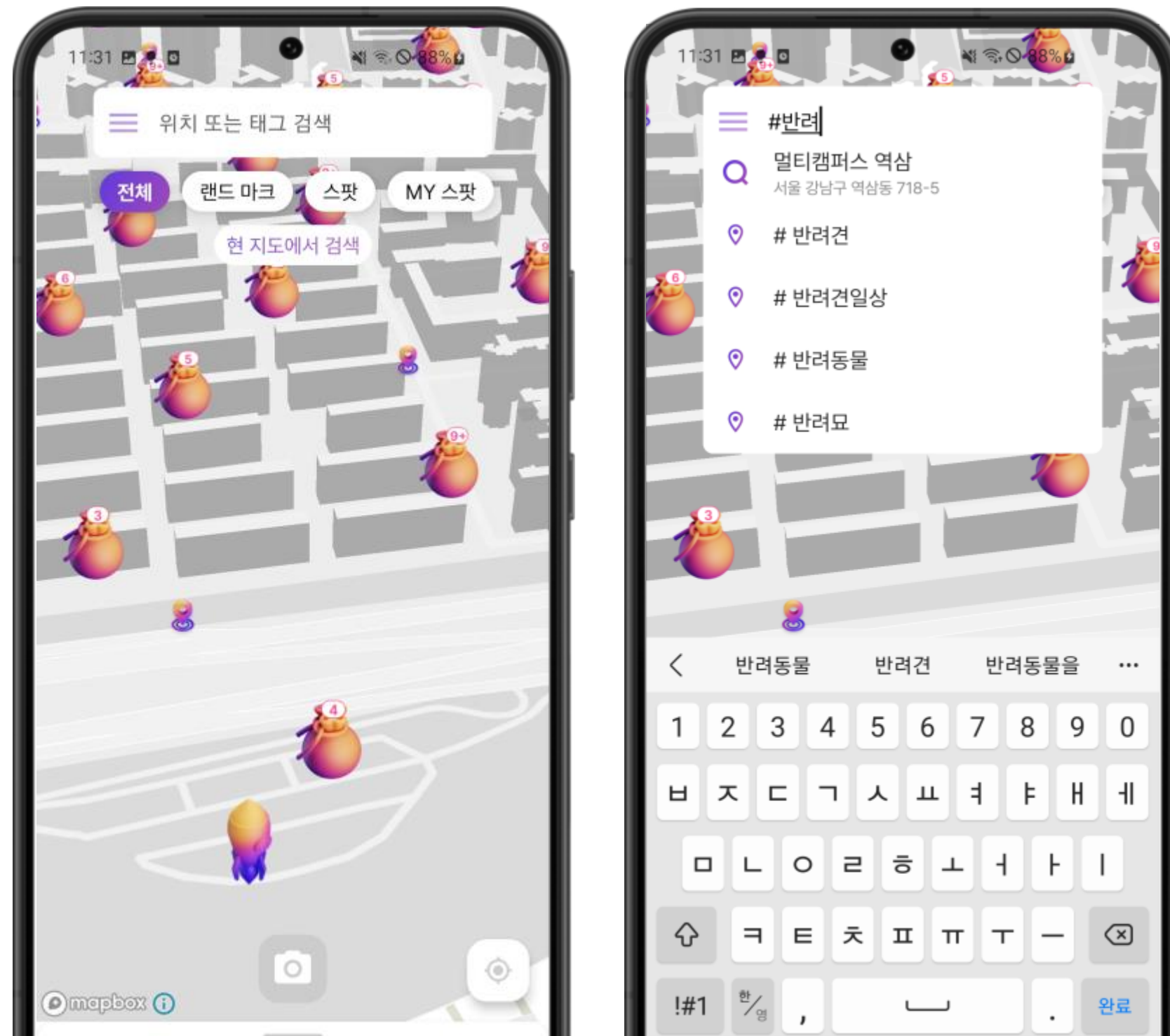
MARU

### | 경매 참여

- 자신의 포인트를 사용하여 경매에 참여
- 경매 낙찰 시 일주일간 랜드마크 주인이 되는 성취감
- 다른 사용자와 상호작용 가능



### 3. 주요 기능 지도 커스텀



#### 지도 커스텀

- 커스텀이 가능한 Map Box 사용
- 클러스터링 기능 구현
- 태그를 사용하여 검색 기능 지원



# 서울에 세워보는 나만의 랜드마크!

랜드마크에 방문해 사진으로 기록을 남기고, 랜드마크 점령 경매에 도전하세요.  
서울을 재미있게 탐험하는 새로운 방법, 지금 바로 마루 서비스에서 만나보세요!



*Get it on*



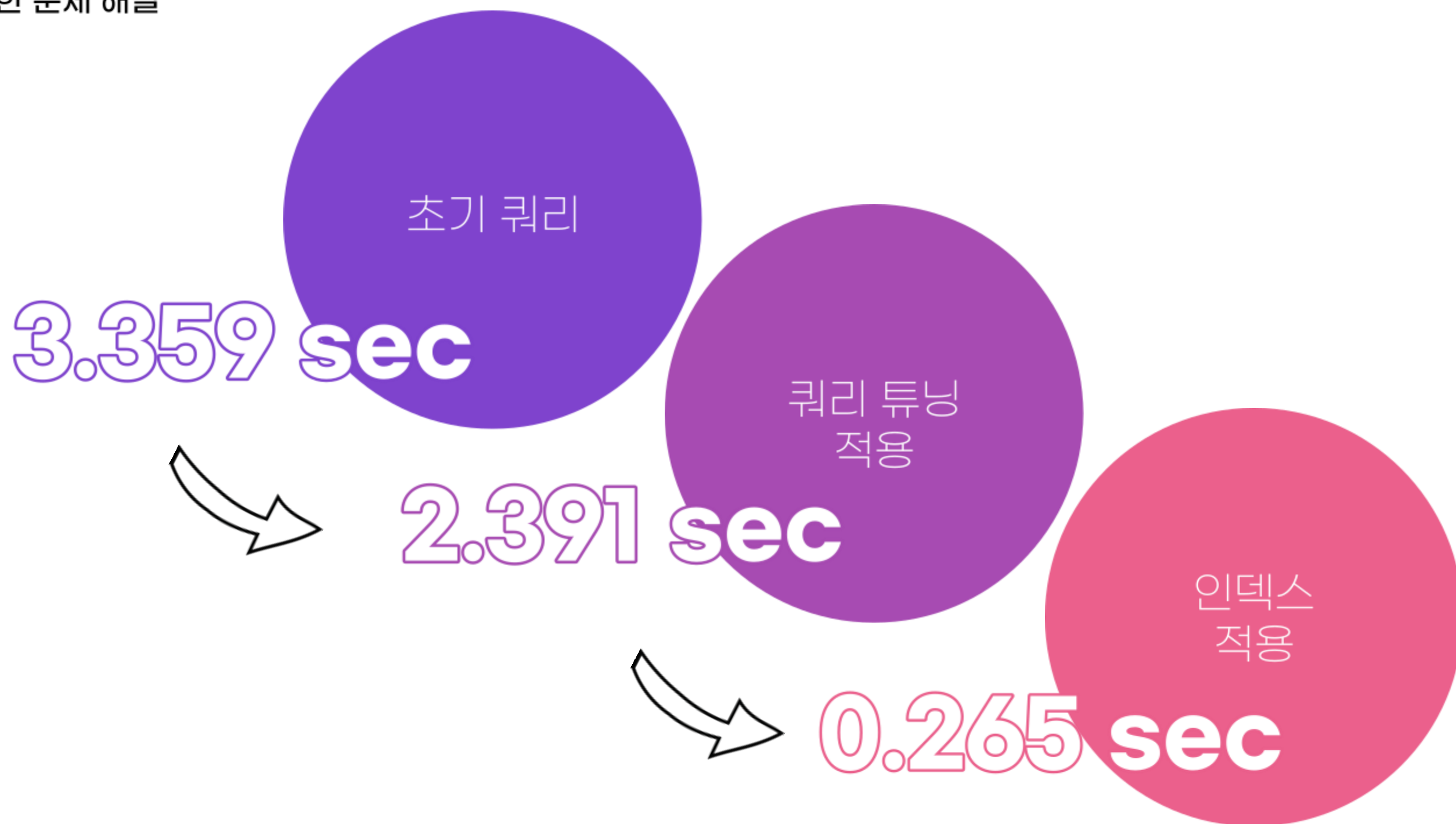
## 4. 문제 해결

지도 내 마커 출력 성능 개선

MARU

| 많은 좌표 로딩으로 인해 서버가 다운되는 문제 발생

- 단계적인 시도를 통한 문제 해결



## 4. 문제 해결

지도 내 마커 출력 성능 개선

MARU

| 많은 좌표 로딩으로 인해 서버가 다운되는 문제 발생

- 단계적인 시도를 통한 문제 해결

**OOM 문제 발생**

0.271 sec

인덱스  
적용



0.265 sec

## 4. 문제 해결

스프링 배치를 이용한 경매

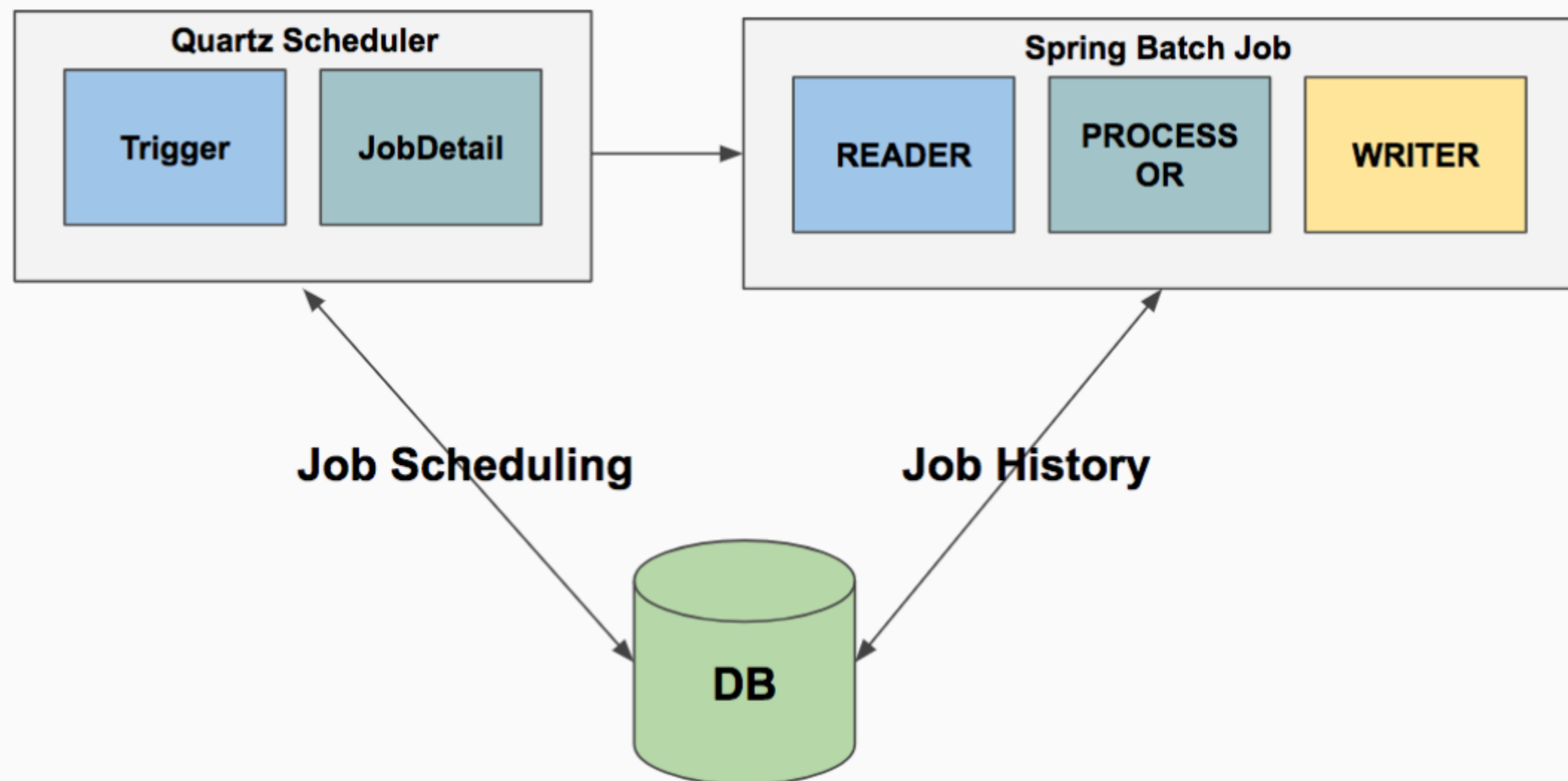
MARU

### | 수많은 입찰 기록들을 한번에 처리해야 하는 문제 발생

- 반복문으로 처리하기에는 서버에 부하 집중
- 트랜잭션 관리, 작업 처리 통계 등 가능한 '**Spring Batch**'를 사용하여 문제 해결
- Named Lock을 사용해서 Spring Batch 작업중인 chunk 단위의 칼럼은 외부에서 접근하지 못하게 처리함



Spring Batch



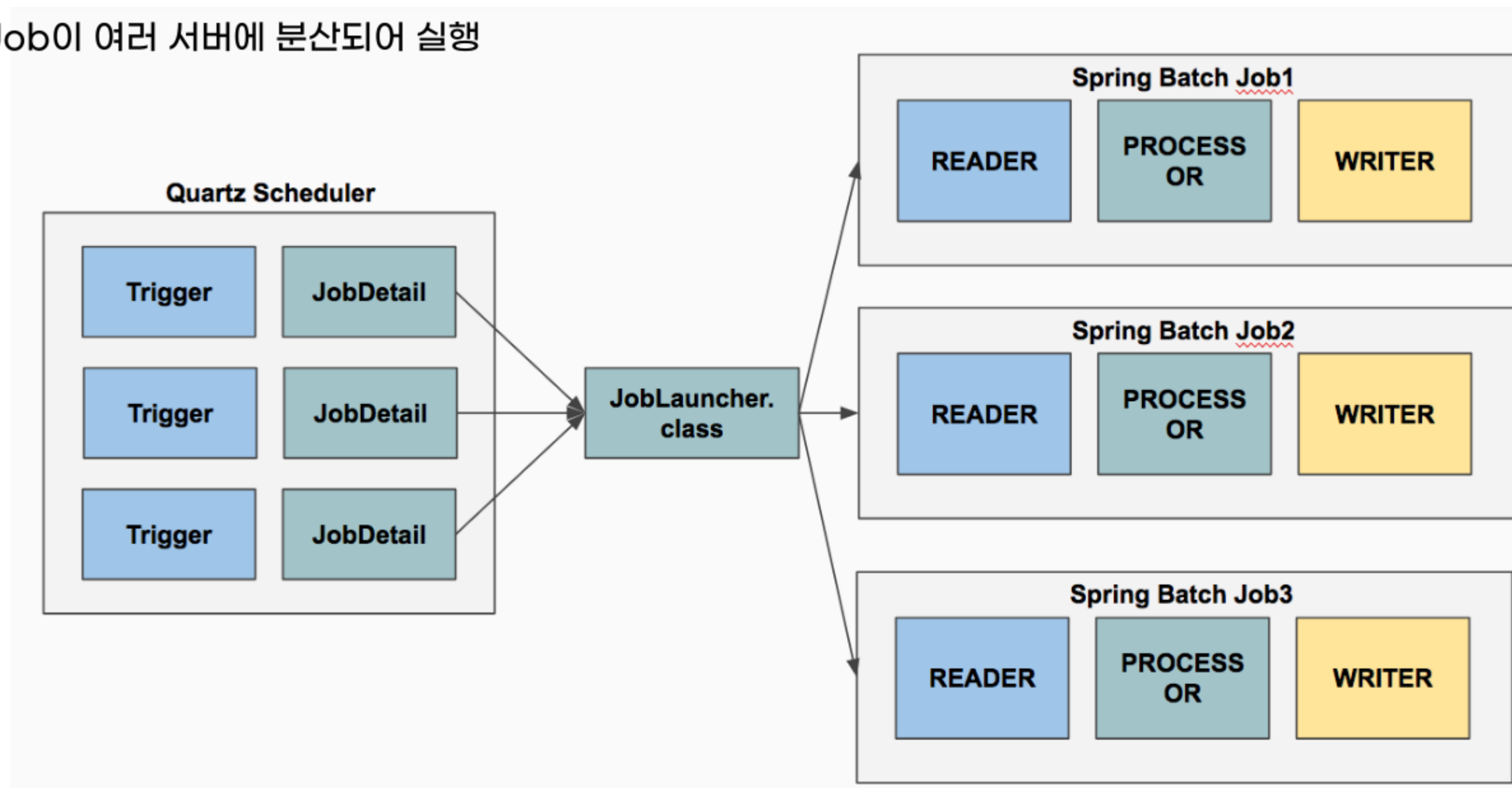
## 4. 문제 해결

스프링 배치를 이용한 경매

MARU

### | 현재 분산 서버를 사용 → 스케줄러 클러스터링이 필요

- Quartz에서 DB 기반의 스케줄러(JdbcJobStore) → 다중 서버 간의 스케줄링이 가능
- 로드 밸런싱 : Spring Batch Job이 여러 서버에 분산되어 실행





## 4. 문제 해결

웹소켓을 활용한 최고 입찰가 갱신

MARU

[

### SSE vs 웹소켓

]

서버에서 클라이언트로 값을 내려줘야 하는 상황 발생

SSE

HTTP 프로토콜로,  
매 서버에서 클라이언트로 푸시할 때마다  
TCP 연결로 인한 오버헤드 발생

⇒ 실시간 데이터 동기화에 어려움

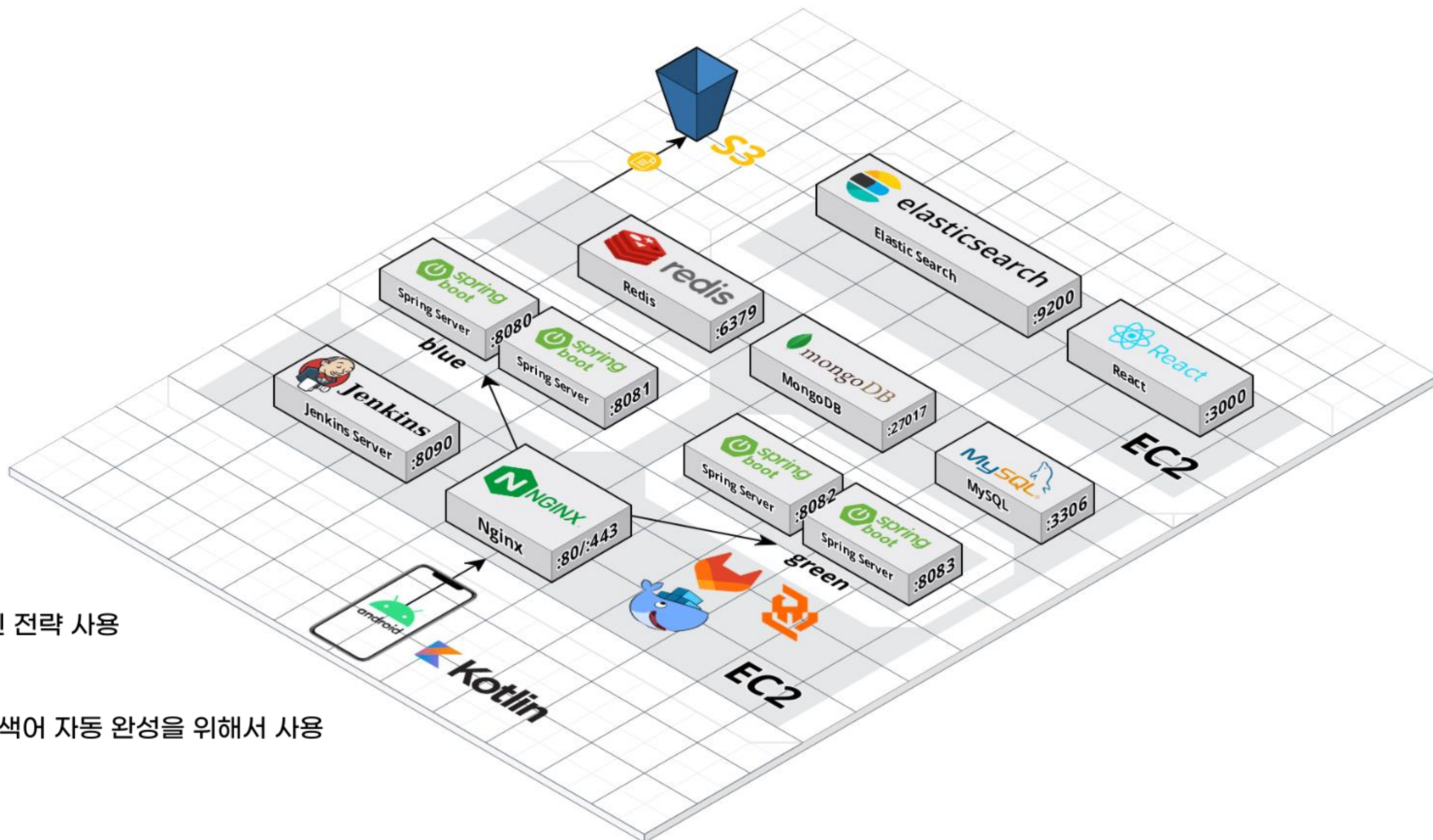


웹소켓

이용 결정

## 5. 산출물

## 시스템 아키텍처



## 시스템 아키텍처

- 무중단 배포 - 블루 그린 전략 사용
  - Round-Robin
- Elastic Search - 검색어 자동 완성을 위해서 사용



## 6. 팀원 소개



- 팀장
- Front End



- Front End
- 디자인



- Front End
- Back End



- Back 팀장
- Infra



- Back End



- Back End

**감사합니다**

Q&A