

# KICKZO PMP

김수경(iOS)

김주원(FE) 신인호(FE)

우다현(BE) 유성욱(BE)

# 목차

## 1. 팀 소개

## 2. 팀 목표

## 3. 개인 목표

- 현재의 모습
- 왜 문제일까
- 두 달 후의 모습
- 실천 방안

## 4. 팀 목표 정리

- 판탐개가 되기위한 3가지 스킬
- 그라운드 룰

## 5. 프로젝트

- 선정이유
- 소개
- 목표
- 목표 달성을 위한 성과지표
- 일정

## 1. 팀 소개

# KICKZO



김수경  
iOS



김주원  
FE



신인호  
FE



우다현  
BE



유성욱  
BE

## 2. 팀 목표

# 개인 목표

-  ( 김수경 / iOS ) - 같이 일하고 싶은 개발자
-  ( 김주원 / FE ) - 배움을 즐기는 개발자
-  ( 신인호 / FE ) - 누구를 만나든 가치를 만드는 사람
-  ( 우다현 / BE ) - 신뢰를 얻는 개발자
-  ( 유성욱 / BE ) - 즐거움을 좇는 개발자

## 2. 팀 목표

**판교가 탐내는 개발자가 되자!**

## 2. 팀 목표

**판교가 탐내는 개발자가 되려면?**

- 기술적으로 / 소프트 스킬적으로 같이 일하고 싶은 사람이 되자!
- 정량적 데이터를 쌓아 나를 알리자!



### 3. 개인 목표 ( 김수경 / iOS )

#### 현재의 모습

##### 경험

- iOS 외의 BE와 FE 개발 **경험이 전무함**

##### 태도

- 라이브러리 사용을 지양하지만, **기술적 성장과 생산성 균형** 고민

##### 기술

- 얕고 넓은 **iOS 기술 스택**(UIKit, SwiftUI, RxSwift, Combine)
- 네트워크 추상화와 DIP를 적용해 **테스트 가능한 코드**를 작성했으나, 테스트 커버리지 부족

#### 왜 문제일까

##### 경험

- 아키텍처 설계 등의 **기술 회의에** 적극적으로 참여하지 못 함
  - BE의 서버 - 저장소 관계, SW(Redis, kafka)의 차이
  - FE의 메모리, 상태 관리 측면

##### 태도

- **바닐라 코드**로의 개발은 기술적 이해를 높일 수 있지만, 생산성과 효율성 저해
- **라이브러리의 남용**은 기술적 고도화를 달성할 수 없음

##### 기술

- 다양한 iOS 기술 스택을 사용해 보았기 때문에 도입은 쉽지만, 깊은 기술적 고도화를 많이 시켜보지 못해, 기술적 부채를 느낀
- 네트워크 커버리지가 낮으면 crash가 발생하여 프로젝트의 완성도가 낮음



### 3. 개인 목표 ( 김수경 / iOS ) - 같이 일하고 싶은 개발자

#### 두 달 후의 모습

##### 경험

- 타도메인 영역의 기술 용어를 알고, 협업 시 대화에 적극적으로 참여할 수 있음

##### 태도

- 시간과 자원분배를 고려한 라이브러리 사용 및 선택에 대해 확장된 시야를 가짐

##### 기술

- 최신 기술 동향에 뒤처지지 않는 iOS 개발자
- crash가 발생하지 않는 앱을 개발할 수 있는 개발자

#### 실천 방안

기술

소프트 스킬

정량적 데이터

##### 경험

- Flask, Node.js 를 사용해서 실제 서버 구현 및 프로젝트 연동
- 상태 관리, 메모리 관리에 대해 작동 원리를 학습하여 FE 개발자와 토론을 통한 FE(Web, App) 개발 토론

##### 태도

- 라이브러리를 사용하거나 또는 참고하되, 그 동작 원리를 이해하고 필요한 부분을 직접 구현하여 최적화 할 수 있는 능력을 함양

##### 기술

- Swift 5.5 기술인 **Swift Concurrency**와 Swift 5.9의 Macro, SwiftTesting, SwiftData의 **신기술 도입 → 성능 최적화**
- UI테스트 단위테스트(네트워크) **테스트 가능한 코드**를 작성



### 3. 개인 목표 ( 김주원 / FE )

#### 현재의 모습

##### 경험

- 새로운 기술에 대해 그것이 얼마나 걸릴지 파악하기 어려움
- 프로젝트 전반에 대한 관리가 어려움

##### 태도

- 잘 알지 못한 기술이라 생각되는 것이 많아 적극적으로 의견을 내기 어려움
- 내가 BE / iOS도 알아야 한다고 생각하지 못함

##### 기술

- CRUD에 관련된 주제와 경험에서 벗어나기 힘듦
- 내가 짠 코드나 방식이 얼마나 효율적인지 파악하지 못하고 의견에만 집중함

#### 왜 문제일까

##### 경험

- 앞으로 주니어 개발자가 되어 새로운 업무를 담당하게 되는데, 그 업무가 스스로 얼마나 걸릴지 미리 파악하기 어려움
- 무엇이 더 효율적인 방식의 코드인지 겪어보지 못해 기준이 없음

##### 태도

- 팀 내에서 적극적인 의견을 내지 못해 팀 활동에 제대로 기여하지 못하고 다른 사람들의 의견에 따르게 됨
- 사고의 확장이 어려움

##### 기술

- CRUD를 벗어난 심화된 기술적 고민에 대해 경험도 적고 고민이 부족함
- 코드에 대한 테스트를 해보지 못해 구현능력만 나아지고 코드가 개선되는지를 확인하지 못함



### 3. 개인 목표 ( 김주원 / FE ) - 배움을 즐기는 개발자

## 두 달 후의 모습

### 경험, 태도

- 내가 했던 코드가 아니라 프로젝트 전반에 대해 이해함
  - FE에서 내가 코딩하지 않은 부분까지 코드를 이해함
  - FE에서 예상치 못한 문제에 대비할 수 있음
  - UI 테스트 및 iOS, 백엔드에게도 컴포넌트를 이해시킴
  - 트러블 슈팅 및 고민해결 과정을 복습하고 공유함
  - 위 과정들을 정량적인 데이터로 쌓음

### 기술

- 주제 위주가 아니라 기술적 도전 중심의 프로젝트
  - CRUD뿐만 아니라 기술적 도전을 통해 깊게 고민함
  - 새로운 기술 스택 사용하여 나의 기술 스택을 넓힘
  - 새로운 기술을 다뤄보며 다양하게 고민함

### 실천 방안

기술

소프트 스킬

정량적 데이터

### 경험

- PR이 올라오면 **코드 리뷰**를 진행
- 단위테스트**(커버리지 80% → 기능O, UI 컴포넌트 X), **E2E테스트**(시나리오)
- Storybook+TSDoc**을 도입하여 **공유** 및 **시각적 회귀 테스트**
- 최소 주1회(맡은 기능마다 1개 이상) **블로그 포스팅**
- 주 1회 **회고 및 문서화**

### 기술

- 채팅**(**이미지 캐싱**, **Lazy Loading**, **가상화기법**→노드800개 이하) 데이터 관리 고도화
- 테스트** 도입, **Zustand** 사용
- WebRTC**를 사용한 **음성채팅**, **소켓**을 이용한 고도화된 **채팅**



### 3. 개인 목표 (신인호 / FE)

#### 현재의 모습

##### 경험

- 백엔드 기술 스택과 아키텍처 설계에 대한 경험이 부족
- 기술적 난도가 높은 프로젝트에서는 방향성을 설정하는 데 어려움
- 프로젝트 관리 경험은 있으나, 데이터 기반의 팀 운영 경험 부족

##### 태도

- UI/UX에 대한 관심은 높지만, BE 기술에 대한 관심은 부족
- 기술적 문제 해결 시 최선의 기술적 선택보다 프로젝트 완성도를 우선시해 현실적 선택을 하는 경향이 있음

##### 기술

- 최근 iOS 개발에 참여하면서 FE 기술에 다시 적응할 시간이 필요
- 테스트 코드 작성 경험이 부족해 코드 품질 관리에서 한계를 느낀다

#### 왜 문제일까

##### 경험

- 원활한 소통을 위해 기술적 이해도는 필수이며, 명확한 방향성이 부족하면 팀원들에게 혼선과 불필요한 시행착오를 초래할 수 있음
- 유연한 팀 운영으로 인해 프로젝트 일정이 지연됨

##### 태도

- 기술적 자신감이 부족하면 서비스 고도화를 주저하게 되고, 이를 기획으로만 해결할 경우 개발자로서의 성장이 정체될 수 있음

##### 기술

- 기술 트렌드와 도구의 변화를 따라가지 못해 더 나은 방향을 제시하기 어렵고, 적응 시간이 소요되어 생산성 저하로 이어질 수 있음
- 대규모 사용자 환경에서 성능 문제가 발생할 가능성이 있음



### 3. 개인 목표 (신인호 / FE) - 누구를 만나든 가치를 만드는 사람

#### 두 달 후의 모습

##### 경험

- 웹 서비스 전반에서 원활한 기술적 소통 가능
- 난도 높은 서비스 기획에도 적극적인 아키텍처 설계
- 데이터 기반의 체계적인 프로젝트 관리 역량 보유

##### 태도

- 특정 기술에 얹매이지 않고 문제 해결에 집중하는 개발자로 성장
- 문제를 개발 관점에서 분석하고 해결할 수 있는 통찰력 확보

##### 기술

- 프론트엔드 최신 기술 스택을 활용해 보고, 비교군과의 장단점을 이해
- 코드 품질을 높이고, 대규모 트래픽을 처리할 수 있는 역량 보유

#### 실천 방안

기술      소프트 스킬      정량적 데이터

##### 경험

- 인증과 채팅 도메인 **아키텍처 설계** 및 **FE, BE 구현**
- Docker Compose를 활용해 **MSA 개발 환경 구축**
- MSA 관련 서적과 강의 **학습** 및 **공유**하고 프로젝트에 **적용**
- **데일리 스크럼, 컨벤션 문서화** 및 **PR 리뷰**를 통해 프로젝트 관리

##### 태도

- **기술 용어를 정리**하고 각 기술의 **사용 목적과 장단점을 이해**
- 문제 해결에 필요한 **기술을 조사**하고, **해결 과정을 기록**

##### 기술

- Zustand, TanStack Query, Stomp.js 등 **새로운 기술**을 도입하여 이전에 사용했던 기술들과의 **차이점을 이해**하고, **기술 선택에 대한 안목**을 높임
- **단위 테스트**와 **E2E 테스트**를 통해 안정성을 확보하고, 대규모 트래픽에 대응할 수 있도록 서버와 DB를 **Scale-out 구조로 설계 및 구현**



### 3. 개인 목표 ( 우다현 / BE )

#### 현재의 모습

##### 경험

- FE, iOS와의 소통 경험 부족

##### 태도

- 코드를 구현 후 작동이 되었다는 결과에 집중
  - 리팩토링을 하지 않음
- 새로 공부한 내용에 대해 체계적인 정리, 기록 습관 미흡

##### 기술

- 간단한 CRUD 중심의 BE 개발 경험만 있으며, 데이터 흐름 설계와 성능 최적화에 대한 깊이 있는 경험 부족
- 분산 시스템 아키텍처에 대한 이해 부족

#### 왜 문제일까

##### 경험

- FE, iOS에서 사용하는 기술 스택과 용어를 알면 협업 과정에서 요구사항을 빠르게 반영하며 문제를 효율적으로 해결 할 수 있음

##### 태도

- 리팩토링을 거쳐 확장 가능한 좋은 시스템이 됨
- 기록의 미흡함은 학습한 내용을 실무에 효과적으로 적용하는 능력을 저하시킴

##### 기술

- 협업 기반의 프로젝트를 진행할 때 어떤 기술을 사용하여 구현을 해야할지 등에 대한 소통의 어려움이 존재함
- MSA는 대규모 서비스, 확장 가능한 서비스를 개발할 때 필수적으로 요구되는 기술임



### 3. 개인 목표 ( 우다현 / BE ) - 신뢰를 얻는 개발자

#### 두 달 후의 모습

##### 경험

- 작동하는 코드를 넘어서 성능까지 고려
  - 질의 최적화를 통해 효율적인 데이터 처리
  - 코드 품질 개선을 통해 성능 최적화의 기초를 다짐

##### 태도

- 코드 리뷰 및 팀원들과의 협업을 통해 다양한 관점 공유
- 캠프에서 얻고 배운 것을 데이터로 쌓아 기록하는 습관 만들기

##### 기술

- MSA, Redis, WebSocket 적용을 통해 기술적으로 성장
  - 기술적 고도화를 통해 내 개발 역량 확장
  - 대규모 트래픽을 고려한 서버 및 데이터베이스 설계

#### 실천 방안

기술

소프트 스킬

정량적 데이터

##### 경험

- SQL 쿼리 튜닝을 진행
  - 실행 계획 분석을 통해 비효율적인 쿼리 최적화 혹은 DB 정규화 검토
- SonarQube를 활용하여 복잡한 로직, 중복 코드, 불필요한 메모리 사용 문제 개선

##### 태도

- 코드 리뷰에 참여해 팀원들과 멘토님의 피드백 기록, 적용
- 1주일 마다 배운 내용을 정리 후 팀원과 공유

##### 기술

- WebSocket을 이용한 실시간 데이터 흐름 구현
- MSA 아키텍처 기반의 설계를 적용하여 프로젝트 확장성 경험
- 데이터베이스 쓰기/읽기 성능을 최적화해야 하며, 분산 처리와 캐싱 전략



### 3. 개인 목표 ( 유성욱 / BE )

#### 현재의 모습

##### 경험

- 프로젝트 관리에 대해 깊이 생각해본 경험 없음
- “개발”에 몰입해본 경험이 없음
- 여러 활동(프로젝트, 스터디, 개인공부 등)에서 얻은 것을 기록하지 않음

##### 태도

- 실패를 실패로 끝냄
- 분야가 다른 질문 중 모르는 것은 나와 관계 없다고 생각함

##### 기술

- 백엔드 개발 경험의 부족
- 기술의 선택 이유 >>> 기술 적용 능력

#### 왜 문제일까

##### 경험

- 프로젝트의 시작은 프로젝트의 관리에서 시작된다고 생각함
- 내 꿈은 개발자
- 머릿속에 있는 것은 시간이 지나면 점차 사라짐

##### 태도

- 실패를 통해 얻는 것은 그 무엇보다 값짐
- 팀원의 질문은 곧 프로젝트에 대한 고민, 나도 그 프로젝트의 구성원

##### 기술

- 내가 꿈꾸는 것은 백엔드 개발을 시작으로 데브옵스 업무와 데이터 엔지니어링 업무를 경험하는 것
- 구현 경험이 적어 기술을 적용하는 능력보다 설계에 대해 고민하는 능력이 큼



### 3. 개인 목표 ( 유성욱 / BE ) - 즐거움을 좇는 개발자

## 두 달 후의 모습

### 경험

- 프로젝트를 프로젝트 관리의 일부에 포함시킬 수 있음
- 내가 맡은 프로젝트에 집요하게 참여하고 그 속에서 **즐거움**을 찾을 수 있음

### 태도

- 실패를 실패로 끝내지 않고 이를 통해 **성공 발판을 마련할** 수 있음
- **함께 성장함의 기쁨을 깨닫기**

### 기술

- 데브옵스, 데이터 엔지니어링 프로젝트로 빌드업할 수 있을 정도의 기술력  
높은 **백엔드 개발 역량** 보유

## 실천 방안

기술

소프트 스킬

정량적 데이터

### 경험

- 자원(시간, 품질)을 고려한 기획, 명확한 업무 분담, 진척 관리, 결과 분석에 대해 깊이 고민해보고 이를 **문서화**하기
- 정해진 시간 안에 맡은 업무에 대해 책임감을 갖고 임하기

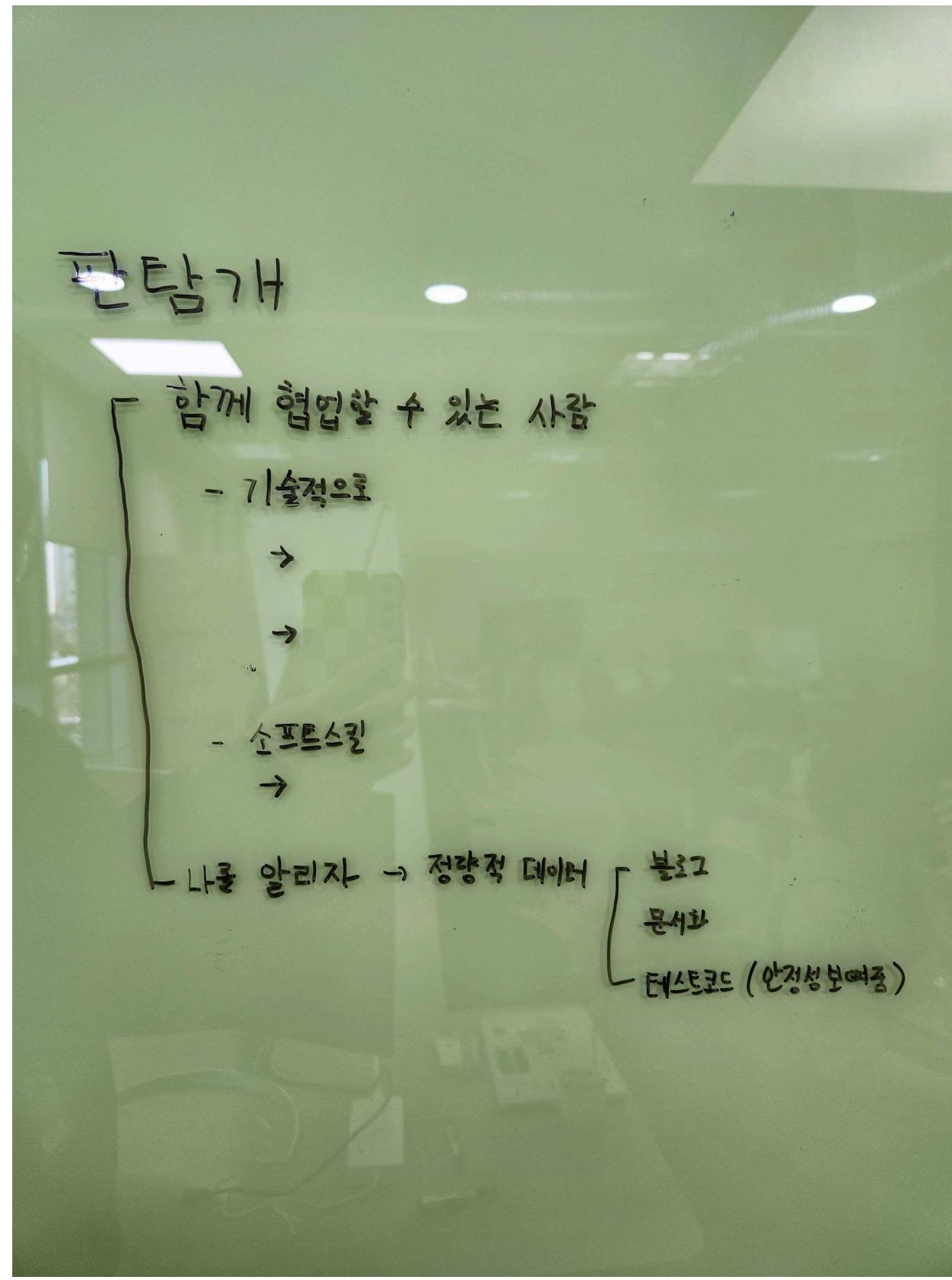
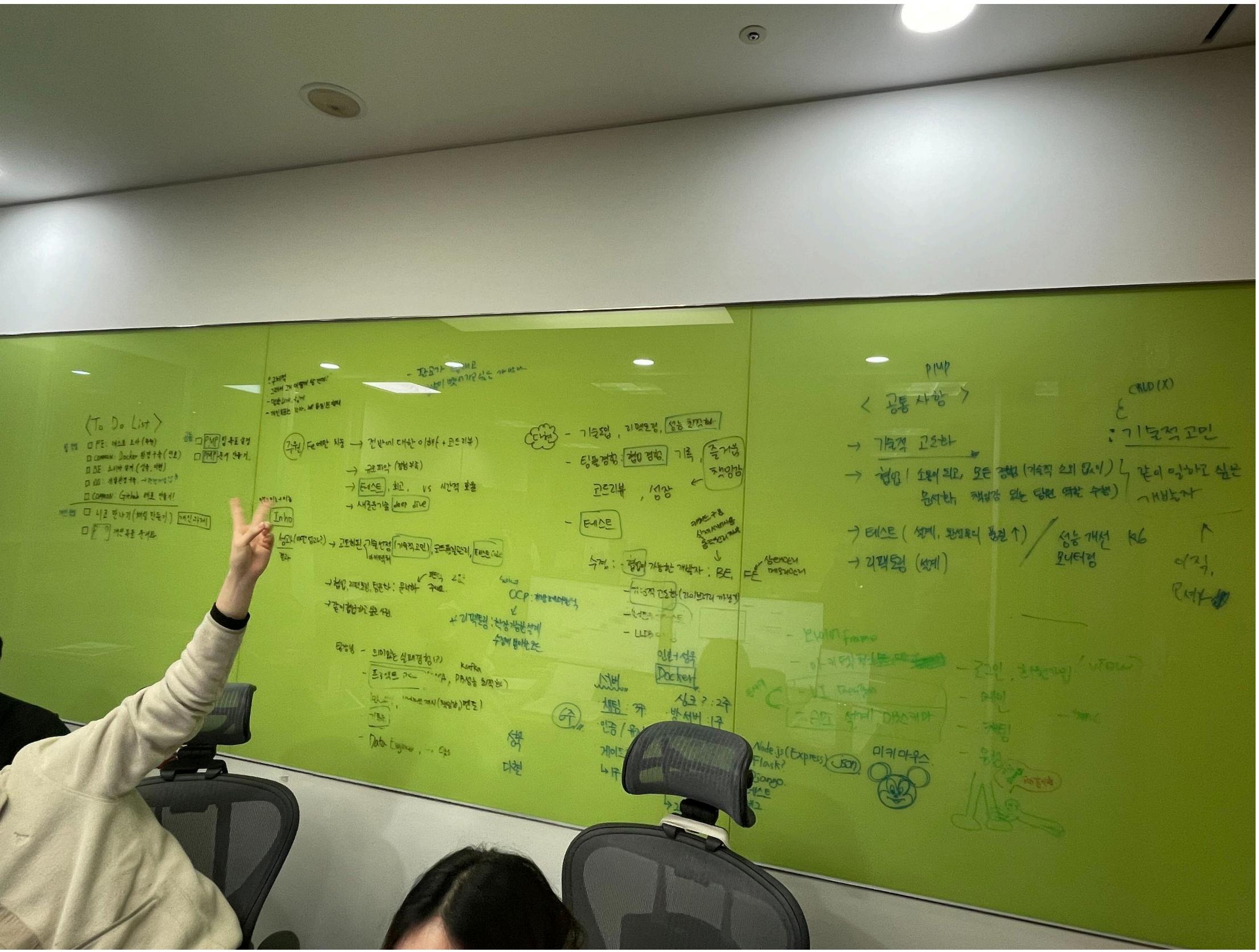
### 태도

- 실패를 통해 깨달은 것을 **정리**하고 **회고**하여 비슷한 문제에 직면했을 때는 같은 실패를 반복하지 않기
- 팀원의 의견에 대해 현실적, 기술적 관점에서 **피드백**하고 팀원의 질문을 매일 기록하여 **함께 고민하는 시간 갖기**

### 기술

- **마이크로서비스**를 효율적으로 설계하고 사용하기
- 데이터베이스 성능 최적화를 고려한 **ERD 설계와 쿼리 튜닝** 경험 보유
- **Kafka의 Producer Consumer**의 동작 원리를 **캠프원에게 설명**
- **Kafka의 Topics와 Partitions**를 설정하고 구현
- 서비스에서 발생하는 **데이터를 만들기** (로그 남기기)

# PMP 문서를 만들면서😊



## 4. 팀 목표 정리 판탐개가 되기위한 3가지 스킬

# 판교가 탐내는 개발자가 되자!

기술적으로 / 소프트 스킬적으로 같이 일하고 싶은 사람이 되자!

- 트렌디한 기술의 이해와 적용
- 성능 최적화와 대규모 트래픽 대응
- 테스트와 품질 관리
- 전반적인 영역(FE, BE, iOS)에 대해 소통할 수 있는 기술적 지식 보유
- 데일리 스크럼을 통한 효율적인 시간관리
- 스크럼 데이터 기반의 주간 회고
- PR 템플릿 제작하여 일관화하기
- 회의 하루 전까지 토론할 내용에 대해 공유하기

정량적 데이터를 쌓아 나를 알리자!

- 테스트 코드를 통한 정량적 데이터 추출 및 성능 지표 개선
- 기술 공유 및 회의록 문서화
- 데일리 스크럼, 데일리 회고, 주간 회고 봇 도입을 통해 개인별 스크럼 데이터화

## 4. 팀 목표 정리

판탐개가 되기위한 3가지 스킬

기술

- 트렌디한 기술 습득
- 성능 최적화
- 전반적인 웹 기술 이해
- 테스트

소프트  
스킬

- 효율적인 소통 방법에 대한 이해
- 적극적이고 활발한 공유 문화
- 좋은 질문과 친절한 답변

정량적  
데이터

- 데스트 시 데이터 기록
- 기술 공유 및 회의 시 문서화
- 스크럼, 회고 시 슬랙 봇을 이용해 기록 관리

프  
로  
젝  
트

## 4. 팀 목표 정리 그라운드 룰



기술

- 트렌디한 기술 습득
- 성능 최적화
- 전반적인 웹 기술 이해
- 테스트



소프트  
스킬

- 효율적인 소통 방법에 대한 이해
- 적극적이고 활발한 공유 문화
- 좋은 질문과 친절한 답변



정량적  
데이터

- 데스트 시 데이터 기록
- 기술 공유 및 회의 시 문서화
- 스크럼, 회고 시 슬랙 봇을 이용해 기록 관리



그라운드  
룰

- 코어 타임 지키기
- 데일리 스크럼
- 스프린트 회고

프  
로  
젝  
트

## 4. 팀 목표 정리 그라운드 룰



### 코어 타임 지키기

- 코어 타임 시작 시 데일리 스크럼 진행
- 코어 타임 종료 전 데일리 회고 진행
- 금요일 코어 타임 종료 전 주간 회고

### 데일리 스크럼

- 팀원 컨디션 파악
- 프로젝트 진행 사항 파악
- 공유할 내용이나 건의사항 파악

### 스프린트 회고

- 프로젝트 진행에 어려운 점은 없었는지, 개선할 점은 없는지 관리

## 5. 프로젝트 선정이유

우리가 만들고 싶은 프로젝트는 무엇일까요?

프  
로  
젝  
트

## 5. 프로젝트 선정이유

### 우리가 만들고 싶은 프로젝트는 무엇일까요?

#### 인호:

- 친구들과 함께 음악 영상을 감상하면서 대화할 수 있는 서비스가 없어서 불편했다.
- 캠프 기간동안 대중에게 관심받을 수 있는 가치 있는 서비스를 만들고 싶다.

#### 성욱:

- 규모가 큰 게임 서비스의 일원이 되고 싶다.
- 유의미한 데이터를 만들어낼 수 있는 주제를 개발하고 싶다.

#### 주원:

- 채팅에서 이미지 캐싱, 가상화 기법 등을 적용해 데이터를 관리해 보고 싶다.
- 테스트를 이용해서 안정적으로 관리하고 싶었고, 주제가 적합하다고 생각했다.

#### 다현:

- 실시간 데이터를 다루는 방법을 이해하고 구현해 보고 싶다.
- 단순 클로닝이 아닌 필요한 여러 기능이 모인 새로운 주제를 만들고 싶다.

#### 수경:

- 이전 프로젝트에서 단순한 소켓 통신을 구현했고, 캠프에서 관련 기술을 고도화해 보고 싶다.
- 푸시 알림을 효과적으로 사용할 수 있는 게 채팅 기반 서비스라고 생각했다.



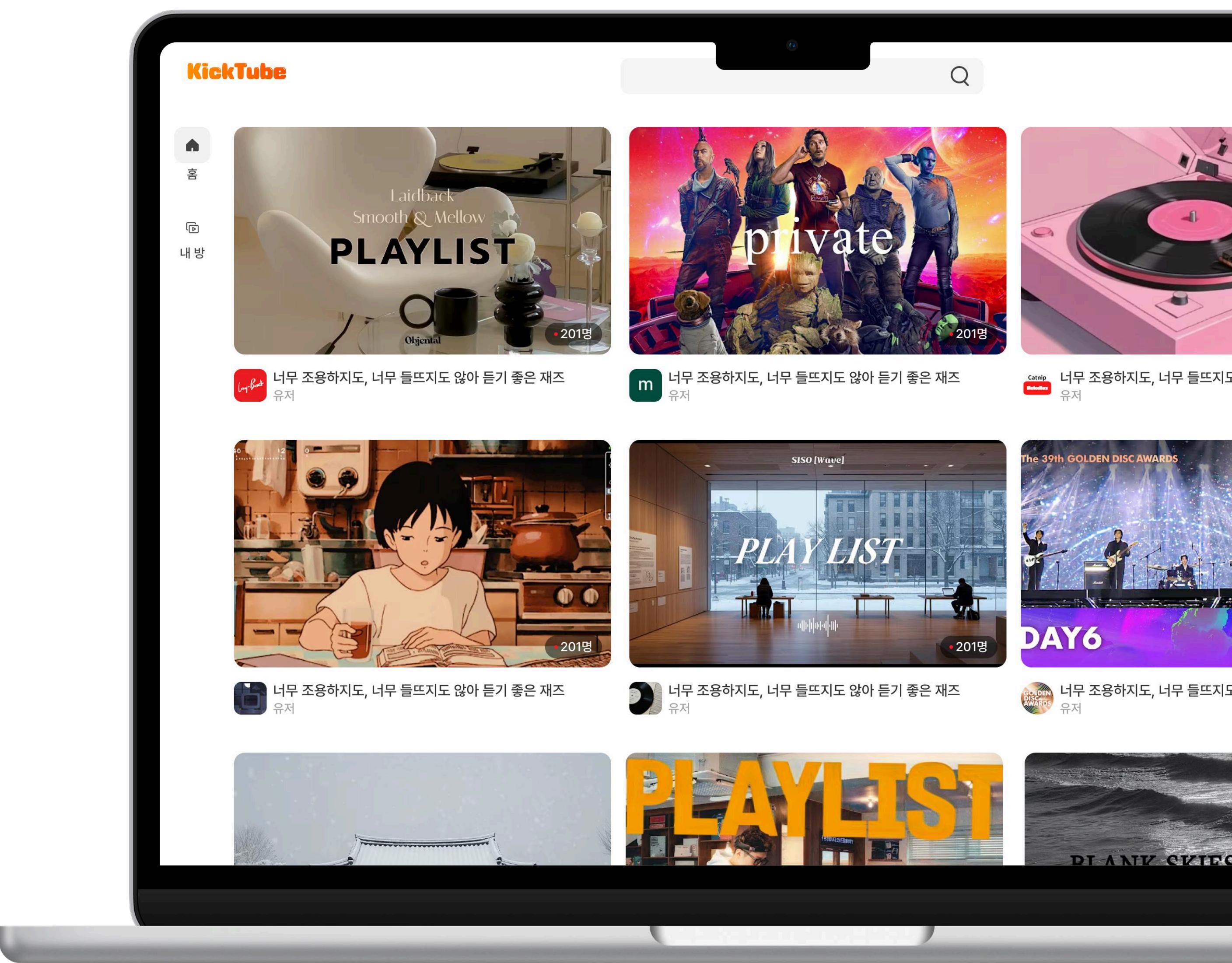
## 5. 프로젝트 선정이유



## 5. 프로젝트 소개

### 영상 동기화 기반 커뮤니티

- 유튜브 영상을 싱크를 맞춰 함께 볼 수 있는 서비스
- 실시간 채팅 / 음성채팅 서비스

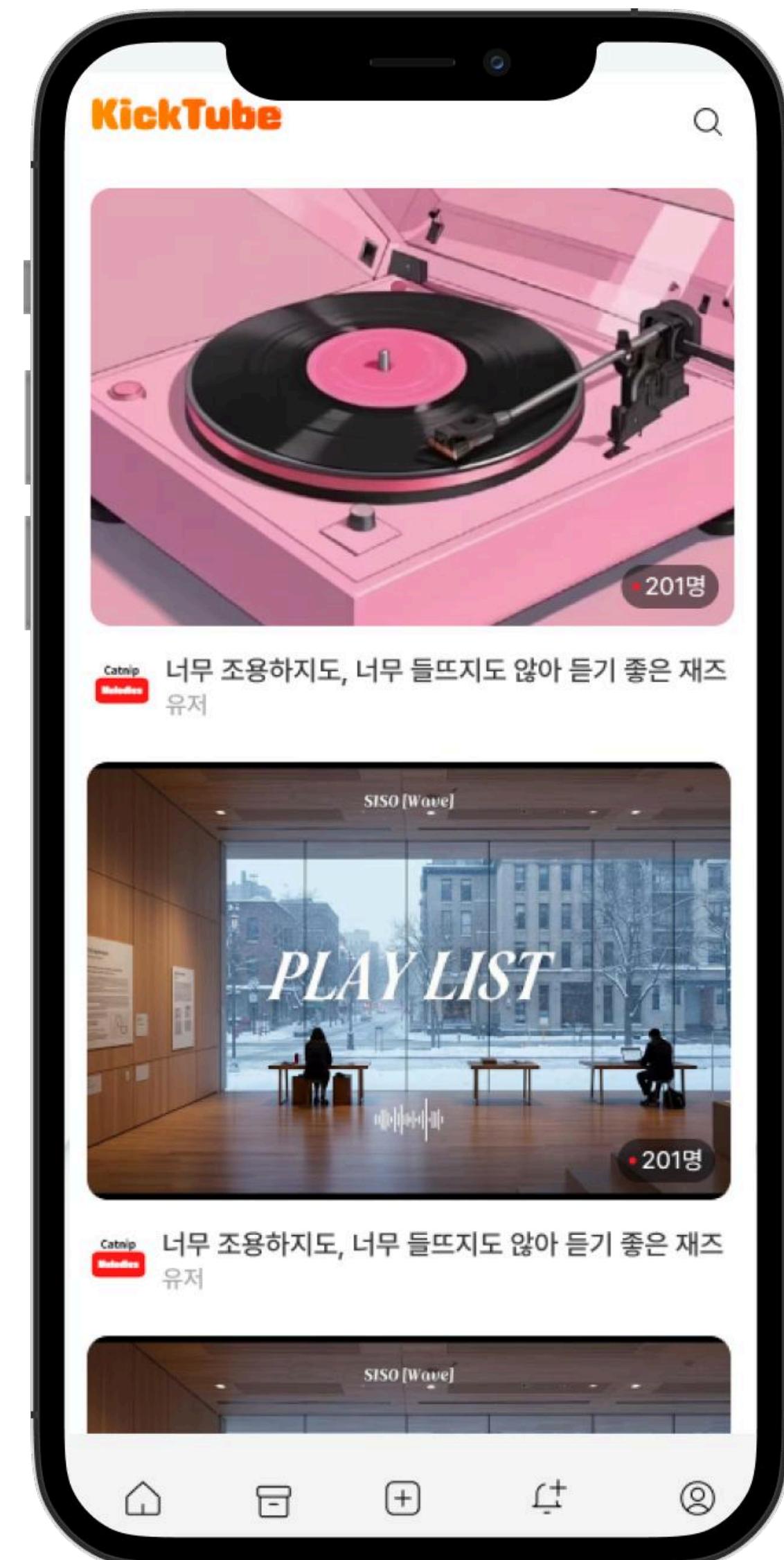


## 5. 프로젝트 목표

기술적으로 완성도 높은 프로젝트

효율적인 협업을 통한 지속적인 개인/팀 성장

정량적 데이터 기반의 프로젝트 관리 경험



## 5. 프로젝트 목표 달성을 위한 평가 지표

### iOS

- 트렌디한 기술의 이해와 적용
  - iOS17 이상부터 사용할 수 있는 SwiftTest, Macro, SwiftData 도입
- 성능 최적화와 대규모 트래픽 대응
  - swift concurrency를 사용한 thread switching 최적화
- 테스트와 품질 관리
  - 단위 테스트, UI 테스트 커버리지 80%

### FE

- 트렌디한 기술의 이해와 적용
  - Tanstack Query, Zustand 등 도입
- 성능 최적화와 대규모 트래픽 대응
  - Lighthouse 성능 지표를 통한 최적화
    - 페이지당 노드 800개 이하
    - 이미지 용량 최적화(약 50KB)
  - 빌드 후 청크 사이즈 500KB 이하 유지
- 테스트와 품질 관리
  - Vitest을 통한 단위 테스트
    - 커버리지 80% 이상
  - Git, 코드 컨벤션을 통한 코드 품질 관리

### BE

- 트렌디한 기술의 이해와 적용
  - Kafka, MSA, Redis 등의 기술 도입
- 성능 최적화와 대규모 트래픽 대응
  - DB Repliaction(쓰기/읽기), Kafka를 통한 TPS, QPS 성능 개선
- 테스트와 품질 관리
  - SonarQube의 code smell 수 제거를 통한 코드 효율 증가와 테스트 커버리지 개선

# 5. 프로젝트 일정 산출

주간회고 기술 공유, 진척 관리  
중간 회고 목표 리마인드, 진척 관리

1월

PMP 작성  
아키텍처 설계  
와이어프레임 설계

UI 디자인  
DB 스키마 설계  
개발 환경 구축  
컨벤션 정리

UI 개발  
API 명세  
정책 문서화

MVP 기능 개발

스프린트 회고

v2 기능 도입  
QA, 테스트 코드 작성

스프린트 회고

성능 테스트  
Scale-out을 위한 구조 정리  
프로젝트 완료 보고서 작성

주간회고

주간회고

주간회고

중간 회고

주간회고

주간회고

주간회고

프로젝트 회고

**판교가 탐내는 키크조**   
**발표를 마치겠습니다.**

**들어주셔서 감사합니다.**