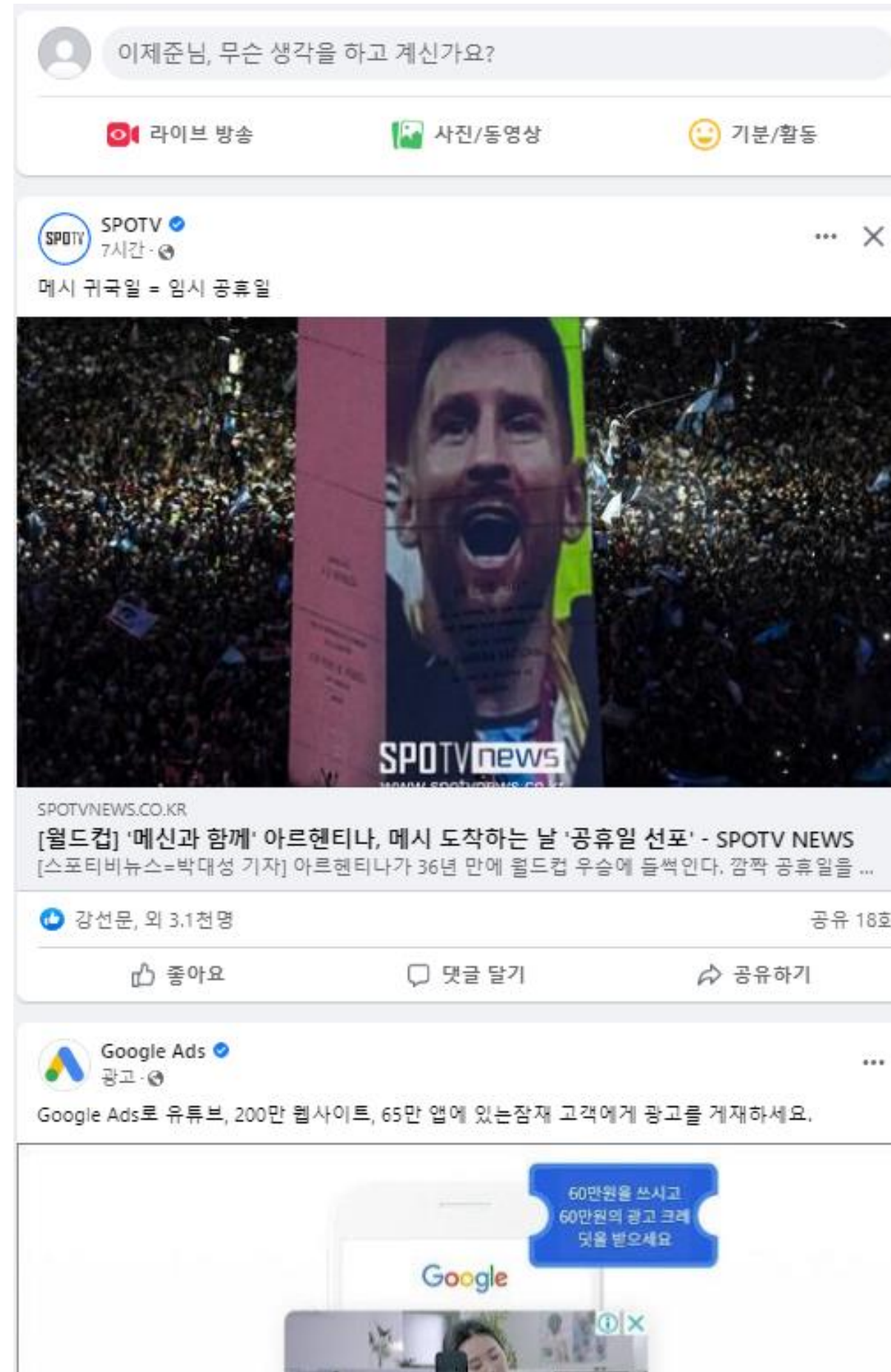




## Backend 요구사항

# 기능 요구사항



# 기능 요구사항

## □ 기능 요구 사항

### 유저

유저는 회원가입, 로그인을 통해 가입할 수 있습니다.

(필요한 값: **email, nickname**)

- (도전)회원가입과 로그인 시 **JWT**와 **Session** 중 하나를 선택할 수 있습니다.

### 글

유저는 게시글을 올릴 수 있습니다.

(필요한 값: 게시글 제목, 내용)

- 유저는 게시글을 수정할 수 있습니다.

- 유저는 게시글을 삭제할 수 있습니다.

# 기능 요구 사항

## □ 기능 요구 사항

### 좋아요

모든 유저는 모든 게시글에 좋아요를 남길 수 있습니다.

모든 유저는 게시글에 남긴 좋아요를 취소할 수 있습니다.

### 댓글

모든 유저는 게시글에 댓글을 남길 수 있습니다.

모든 유저는 게시글에 남긴 댓글을 삭제할 수 있습니다.

# 기능 요구사항

## □ 기능 요구 사항

### 글 조회

글 조회는 최신순, 좋아요 많은 순으로 정렬할 수 있습니다.  
메인 화면에서는 모든 글을 최신순으로 보여줄 수 있습니다.  
(단, 페이징은 하지 않으셔도 됩니다.)

### 마이페이지 조회

유저의 정보들을 볼 수 있습니다.  
내가 쓴 글을 볼 수 있습니다.  
닉네임과 이메일을 수정할 수 있습니다.  
- 중복되면 안됩니다.

# 기능 요구사항

## □ 기능 요구 사항

indent(인덴트, 들여쓰기) depth를 3이 넘지 않도록 구현한다. 2까지만 허용한다.  
예를 들어 while문 안에 if문이 있으면 들여쓰기는 2이다.

힌트: indent(인덴트, 들여쓰기) depth를 줄이는 좋은 방법은 함수(또는 메서드)를 분리하면 된다.

3항 연산자를 쓰지 않는다.

함수(또는 메서드)의 길이가 15라인을 넘어가지 않도록 구현한다.

함수(또는 메서드)가 한 가지 일만 잘하도록 구현한다.

else 예약어를 쓰지 않는다.

힌트: if 조건절에서 값을 return 하는 방식으로 구현하면 else를 사용하지 않아도 된다. else를 쓰지 말라고 하니 switch/case로 구현하는 경우가 있는데 switch/case도 허용하지 않는다.

# 기능 요구사항

## □ 기능 요구 사항

구조는 service, domain, controller로 주축을 잡는다.

- 계층간 데이터 전달은 DTO를 활용한다.
- JPA 사용 시 Optional을 사용하여 엔티티를 return한다.
- Domain 패키징은 엔티티(도메인)별로 한다.

모든 에러는 IllegalArgumentException으로 처리한다.

필요한 dependency는 implementation을 한다.

브랜치는 main으로 간 후 pull 후 새로운 브랜치를 만들어서 작업한다.



중요한 것은 본인 스스로  
찾아보고 생각하고 구현하는 것.



# 배 포