

JIKJI팀 아키텍처

김정현 박종하 이한솔 김영현

주제에 따른 기능

전체 구조

기술 스택

업무 분담

DB 구조

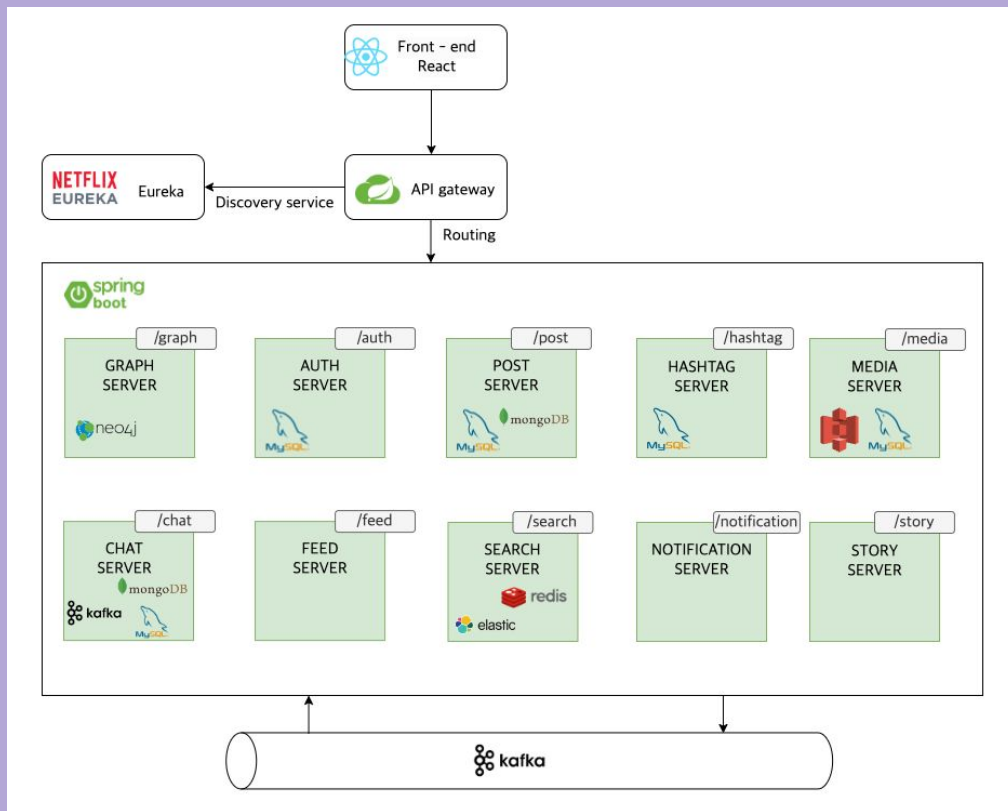


Instagram

기능

- 게시물 생성, 조회, 수정, 삭제
- 게시물 댓글
- 스토리 생성, 조회, 수정, 삭제
- 댓글 생성, 조회, 삭제
- 해시태그 생성, 삭제
- 사용자 ID, 이름, 해시태그 검색
- 팔로우 기능
- 채팅 기능
- 알림 기능

전체 구조



기술 스택

Language



Framework



* react 17

Database



DevOps



kubernetes



kafka



Amazon S3

Auth
김영현

Media
박종하

Follower
김정현

Post/Hashtag
이한솔

Comment/Feed
박종하

Notification
김정현, 김영현

Search
이한솔

Chat
이한솔, 김영현

Story
김정현, 박종하

Posts

Posts Entity

표

Aa 이름	속성	참고
id	PK	
user_pk	not null FK	게시글을 올린 사용자의 pk
description	nullable	소개글 최대 150자
comment_disable	nullable	default=false
likes_disable	nullable	default=false
created_date	nullable	"yyyy-MM-dd"
likes		default = 0, 좋아요 수
location_y	nullable	
location_x	nullable	
custom_accessiblity_caption	nullable	접근성>대체 텍스트

Post-images Entity

하나의 포스트당 최대 10장까지 사진 첨부 가능

표

Aa 이름	태그	참고
id	PK	
post_pk	not null FK	
image_url	not null	gcp or aws storage 업로드된 링크

Post-usertags Entity

표

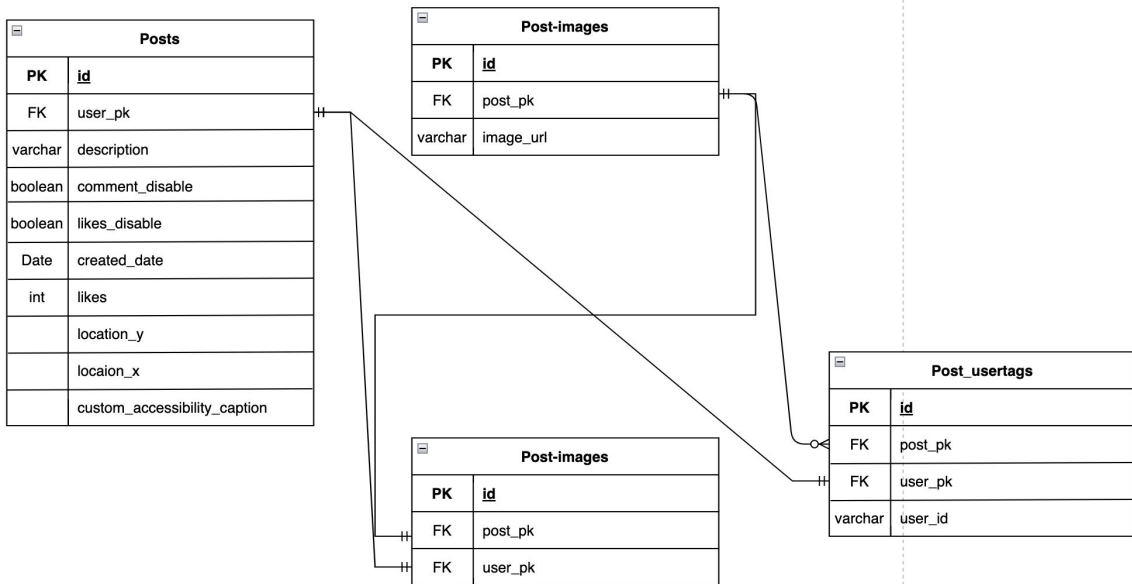
Aa 이름	태그	
id	PK	
post_pk	not null FK	
user_pk	not null FK	
user_id	not null	

Post-bookmarks Entity

표

Aa 이름	태그	
id	PK	
post_pk	not null FK	
user_pk	not null FK	

Posts



Comments

Comments Entity

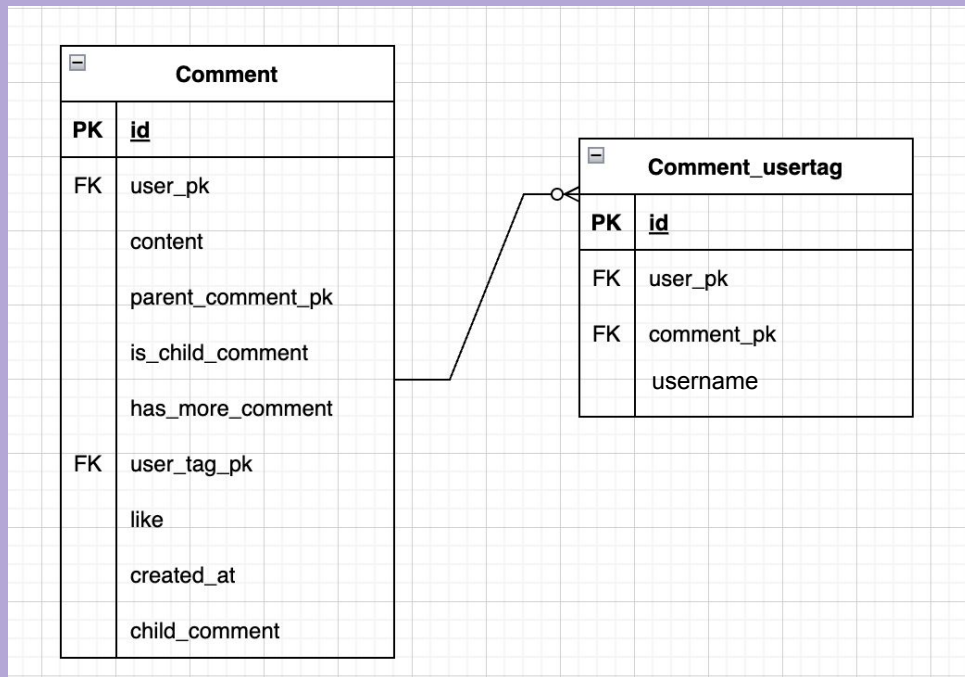
Aa 이름	≡ 태그	≡ 참고	≡ 제한
id	PK		
user_pk	not null FK	댓글 단 유저의 id	
content	not null	댓글 내용	length ≤ 200
parent_comment_pk	nullable	대댓글을 달은 댓글의 pk	
is_child_comment	not null	true: 대댓글, false: 일반 댓글	
has_more_comment	not null	default=false, 대댓글이 달린 여부	
user_tags_pk	nullable FK	@유저 태그 목록	
is_ranked	not null	default = false, 상위에 보여질 댓글	본인 게시물이거나 좋아요가 댓글 평균이상이라면 true
like	not null	default = 0, 좋아요 수	
created_date	not null		

Comment_usertag ...

Aa 이름	≡ 태그	≡ description
comment_tag_id	PK	usertag 의 id
user_pk	FK	user 의 pk
comment_pk	FK	comment의 pk

1. 새로 만들기

Comments



Media

Media Entity

Aa 이름	:≡ 속성	:≡ type	≡ Description
id	PK	LONG	미디어의 ID
created_at		TIMESTAMP	미디어 create 날짜
url	NOT NULL	String	미디어의 URL
type	NOT NULL	String	IMAGE, VIDEO 타입
media_name	NOT NULL	String	미디어 이름
user_name	NOT NULL	String	upload 한 유저의 이름

Feed

Feed Entity ...

Aa 이름

☰ 태그

☰ description

<u>id</u>	PK	feed의 id
<u>userId</u>	FK	유저의 id
<u>postId</u>	FK	post id
<u>createdAt</u>	TIMESTAMP	생성된 날짜
<u>username</u>	NOT NULL	유저 이름

Stories

Aa 이름	::: 속성	≡ 참고
id	PK	
user_pk	not null FK	스토리를 올린 사용자의 pk
image_url	nullable	gcp or aws storage 업로드된 링크
created_date	nullable	"yyyy-MM-dd"
regrammed_user_pk	nullable FK	공유한 스토리일 경우 원본 사용자의 pk

Story-likes Entity

표

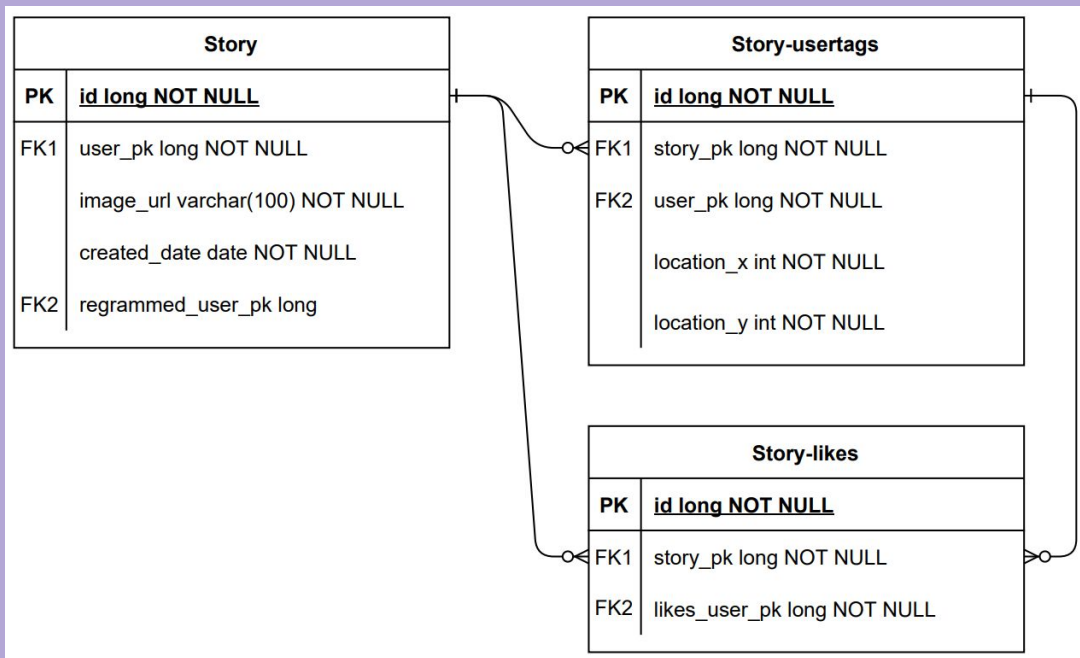
Aa 이름	::: 태그	≡ 참고
id	PK	
story_pk	not null FK	좋아요가 달린 스토리의 pk
likes_user_pk	not null FK	좋아요를 누른 유저의 pk

Story-usertags Entity

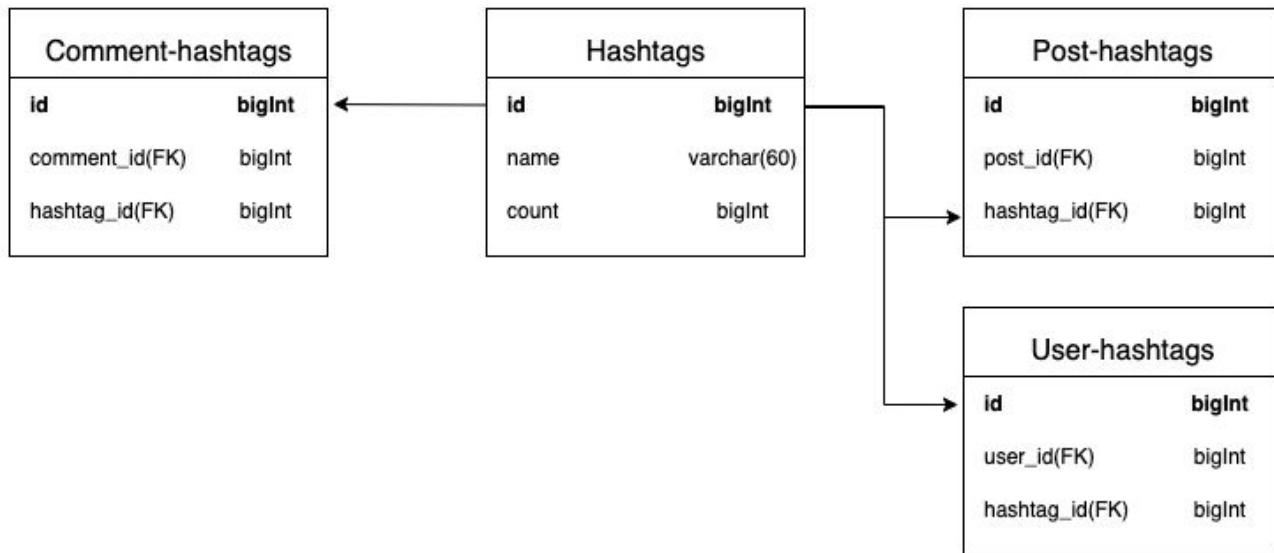
표

Aa 이름	::: 태그	≡ 참고
id	PK	
story_pk	not null FK	태그가 달린 스토리의 pk
user_pk	not null FK	스토리가 태그한 유저의 pk
location_x	not null	스토리에서 태그 위치의 x축 값
location_y	not null	스토리에서 태그 위치의 y축 값

Stories



Hashtags



Auth

User Entity

Aa 이름	태그	텍스트
id	PK	
email	not null	이메일 형식만 가능, 중복안됨
password	not null	특수문자 포함 8~20자만 가능
nickname	not null	다른 닉네임과 중복 안됨
phone_number	not null	전화번호 형식만 가능
email_auth	not null boolean	default=false, 이메일 인증
created_date	not null	생성일
user_profile_img	not null	default=회색 바탕

User_roles Entity ...

Aa 이름	태그	텍스트
user_email	PK not null	
roles	not null	유저의 역할

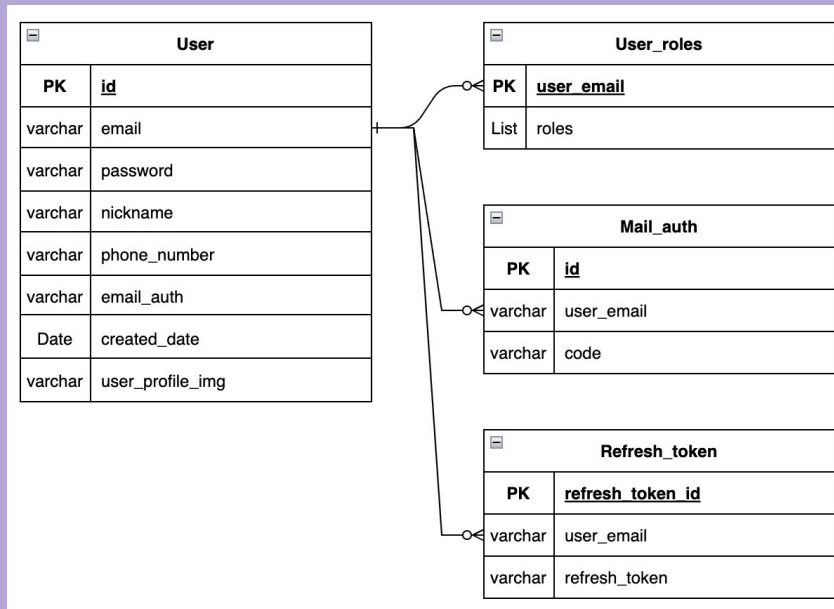
Mail_auth Entity

Aa 이름	태그	텍스트
id	PK	
user_email	not null	
code	not null	이메일 인증 번호

Refresh_token Entity

Aa 이름	태그	텍스트
refresh_token_id	PK not null	
user_email	not null	
refresh_token	not null	refresh_token 정보

Auth



Graph(follower)

Follow Entity

Aa 이름

⋮ 태그

≡ 텍스트

id

PK

index

follower_id

not null

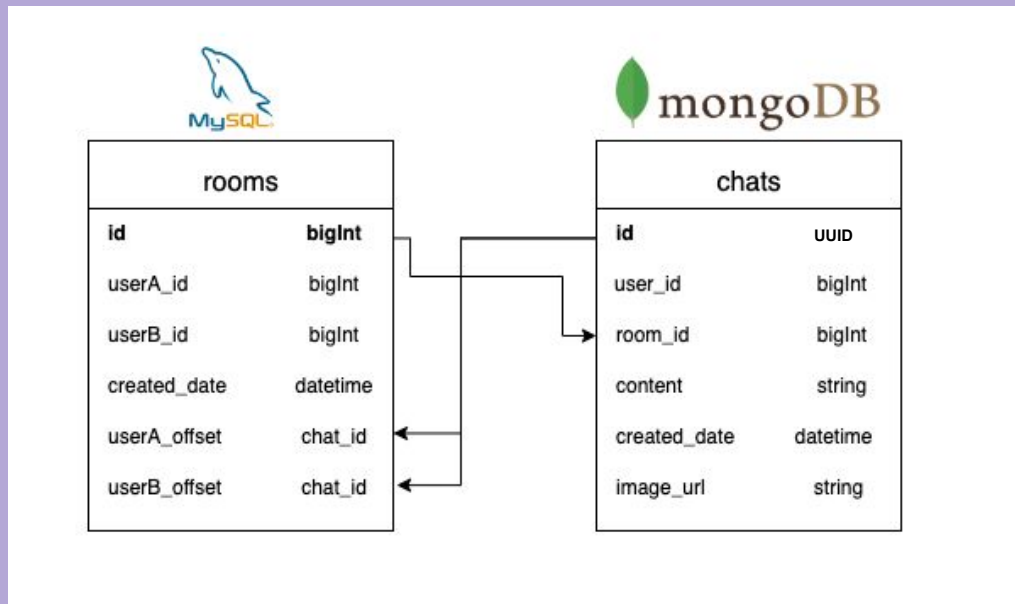
follow하는 유저의 id

followed_id

not null

follow당하는 유저의 id

Chat



Search

표		
Aa 이름	태그	참고
id	KEY	
user_id	string	사용자 아이디
username	string	

표	
Aa 이름	태그
id	KEY
hashtag_name	string

