**[smooth] 2차 코드 리뷰 - backend**

updated at: 2022.02.04.

< Chat-server >

1. 채팅 로직에 대해서

1. Controller : client가 보낸 채팅 CRUD 를 messageMapping으로 받아온다.
2. MessageSender : 채팅을 해당 카프카 topic으로 보낸다.
3. **MessageListener(**[링크](https://github.com/stove-smooth/sgs-smooth/blob/develop/src/backend/chat/src/main/java/com/example/chatserver/config/message/MessageListener.java)**) : 해당 채팅에 대한 로직 처리(redis로 유저정보 캐싱 등)**
4. 채팅을 클라이언트로 전송

kafka topic들

1. chat-server-topic : 다이렉트 채팅을 위한 토픽
2. channel-server-topic : 커뮤니티 채팅을 위한 토픽
3. etc-direct-topic : 다이렉트 안의 채팅 기능들을 위한 토픽(삭제,수정,타이핑 등)
4. etc-community-topic: 커뮤니티 안의 채팅 기능들을 위한 토픽(삭제,수정,타이핑 등)
5. file-topic: 파일 업로드를 위한 토픽

채팅 서버 3대를 사용하고 있기 때문에 kafka로 동기화처리를 하고있습니다.

커뮤니티 채팅 , 다이렉트(개인,개인) 채팅으로 두 가지 채팅 로직이 있습니다.

각각의 채팅은 다른 토픽을 구독하고 있습니다.

채팅 로직 3번 [링크](https://github.com/stove-smooth/sgs-smooth/blob/develop/src/backend/chat/src/main/java/com/example/chatserver/config/message/MessageListener.java) 에서

메세지를 맵핑하고, redis로 정보를 가져오는 등, 대부분 작업이 이루어지고 있습니다.

모든 채팅 로직을 리스너 안에 작성한 부분에 대해서 코드리뷰를 받고싶습니다.

< Community-server>

초대장 만들기

# 상황

* 디스코드 내에서 다른 사용자를 커뮤니티에 초대하기 위해서 ‘초대 코드’를 생성해서 제공하고 있습니다.
* 하나의 커뮤니티 내에 각 사용자별로 고유한 초대장을 생성할 수 있고, 생성한 경우 이미 존재하는 초대장의 코드를, 없는 경우 새로 생성해서 코드를 제공하고 있습니다.
* 코드는 아래와 같이 작성했습니다.

# 코드



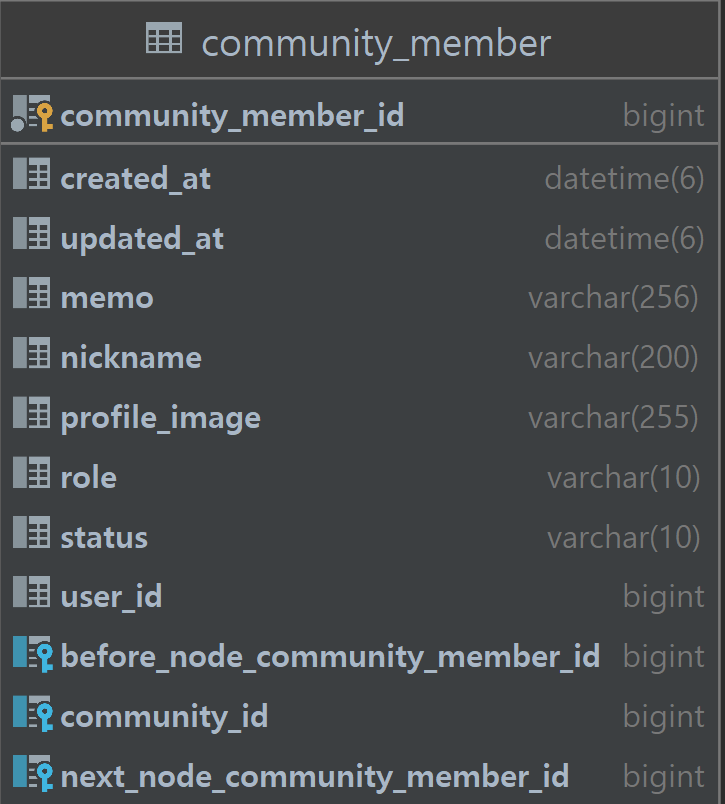
* 고유한 초대코드를 만들기 위해서 커뮤니티와 사요자를 기준으로 DB에 값을 저장하고 해당 값을 인코딩(url shortener와 유사한 로직)하여 전달하고 있는데 이와 관련해서 효율적인 방법이 있는지 리뷰받고 싶습니다.
* 초대 코드를 랜덤으로 값을 생성하는 경우 일치하는지 계속해서 확인할 필요가 있고, uuid를 생성하게 되면 값이 너무 길어져서 해당 방법은 사용하지 못 했습니다.

채널 순서 바꾸기

# 상황

* 디스코드에서 커뮤니티의 순서나 채널의 순서를 자유롭게 수정하는 기능을 제공하고 있습니다.
* 순서를 저장하기 위해서 ‘순서 컬럼’을 이용해서 값을 저장하고자 했지만 그 경우 순서를 변경할 때 모든 로우의 값을 수정해야하기 때문에 해당 경우를 배제했고,
* 순서를 최대한으로 보장하기 위해 ‘이전 노드’와 ‘이후 노드’의 아이디 값을 저장하여 활용하고 있습니다.
* 순서를 변경해야 하는 상황에서 가장 효율적으로 수정하고 관리하기 위해 좋은 방법이 어떤 것일지 고민하고 있습니다.

community\_member 테이블





CommnityMember.java



* 현재는 순서를 변경할 때 이전 노드의 값과 이후 노드의 값을 변경하여 저장하고 있습니다.