

Index	Method	URI	description
1	POST	/api/auth	로그인
2	POST	/api/user	회원가입
3	GET	/api/user/check	ID, Email 중복 확인
4	POST	/api/user/modifiedChar	캐릭터 값 수정
5	POST	/api/user/musicListName	리스트 받기
6	POST	/api/user/musicList	리스트의 노래 id 받기
7	POST	/api/user/makeList	리스트 만들기
8	POST	/api/user/deleteList	리스트 삭제
9	POST	/api/user/addSong	리스트에 노래 추가
10	POST	/api/user/deleteSong	리스트에 노래 삭제
11	POST	/media	음원 파일 업로드
12	POST	/media/delete	음원 파일 삭제
13	GET	/media	노래 스트리밍
14	POST	/api/music	노래 DB 등록

15	GET	/api/music/title	노래 제목 검색
16	GET	/api/music/artist	노래 가수 검색
17	GET	/api/music/category	노래 장르 검색
18	GET	/api/music/recent	최근 노래 리스트 가져오기
19	GET	/api/music/personalGenre	유저의 선호 장르 노래 추천
20	POST	/api/music/popular	인기 노래 리스트 가져오기
21	POST	/api/music/play	노래 재생 횟수 추가
22	GET	/api/follow	팔로우 리스트 가져오기
23	GET	/api/follow/follower	팔로워 리스트 가져오기
24	POST	/api/follow	팔로우 추가
25	POST	/api/follow/delete	팔로우 취소
26	POST	/api/auth/email	인증 메일 전송
27	GET	/api/user/info	유저 정보 받기
28	GET	/api/music/uploadList	업로드 리스트 받기
29	POST	/api/musiczone/createzone	음원존 생성

30	GET	/api/musiczone/joinZone	음원존 가져오기
31	GET	/rtc/channel/role/uid/userId	Agora 토큰 가져오기
32	GET	/rtc/channel/delete	Agora 토큰 서버에서 채널 삭제하기

1. 로그인

input : id, password (string)

output : token, user-data

⇒ id 와 password 가 DB 와 일치하면 token 과 유저의 데이터를 준다.

2. 회원가입

input : id, email, password, nickname, character, preferredGenres

output : X

⇒ userDB 에 데이터 추가한다.

3. ID, Email 중복 확인

input : id or email (string)

output : idExist, emailExist

⇒ 중복된 데이터가 존재하는 경우 true, 없는 경우 false 로 리턴한다.

4. 캐릭터 값 수정

input : value (int)

output : user 의 캐릭터 값

⇒ 입력받은 value 로 유저의 charactor 값을 변경해준다.

5. 리스트 받기

input : X

output : listName (리스트들의 이름 배열)

⇒ list 들의 이름이 담긴 배열을 준다.

6. 리스트의 노래 id 받기

input : listName (String)

output : music (노래가 담긴 리스트)

⇒ listName 에 해당하는 리스트에 담긴 노래들을 준다.

7. 리스트 만들기

input : listName (String)

output : listName (리스트들의 이름 배열)

⇒ listName 을 입력 받으면 중복 확인 후 listName 을 추가한 리스트를 반환한다.

8. 리스트 삭제

input : listName (String)

output : listName (리스트들의 이름 배열)

⇒ listName 을 입력 받으면 DB 의 listName 에서 해당 리스트를 삭제한다.

9. 리스트에 노래 추가

input : listName (String), musicList (Array)

output : music (Array)

⇒ listName 을 입력 받고 musicList 를 받으면 노래를 추가하고 해당 리스트를 준다.

10. 리스트에 노래 삭제

input : listName (String), musicId (String)

output : music (Array)

⇒ listName 을 입력 받고 musicId 를 받으면 노래를 삭제하고 해당 리스트를 준다.

11. 음원 파일 업로드

input : 음악 파일, 이미지 파일 (mp3, wav, ogg, jpg, png)

output : locate, imageLocate

⇒ 음악 파일과 이미지 파일을 저장하고 저장한 위치를 돌려준다.

12. 음원 파일 삭제

input : 음악 파일, 이미지 파일(mp3, wav, ogg, jpg, png)

output : X

⇒ 음악 파일과 이미지 파일을 삭제해준다.

13. 노래 스트리밍

input : pathname (String)

output : musicStream

⇒ pathname 에 존재하는 음악 스트림 데이터를 준다.

14. 노래 DB 등록

input : locate ,imageLocate, title, category, lyrics, info, time (String)

output : music 의 정보, user 의 totalNum 과 musicList

⇒ 음원 정보를 입력받으면 DB 에 등록하고 해당 유저의 upload 리스트에 추가한다.

15. 노래 제목 검색

input : title (string)

output : music (Array)

⇒ title 을 입력받아 해당 title 이 들어있는 제목의 노래들을 리턴한다.

16. 노래 가수(nickname) 검색

input : artist (string)

output : music (Array)

⇒ artist 를 입력받아 해당 artist 단어가 들어있는 가수의 노래들을 리턴한다.

17. 노래 장르(category) 검색

input : category (string)

output : music (Array)

⇒ category 를 입력받아 해당 category 의 노래들을 리턴한다.

18. 최근 노래 리스트 가져오기

input : X

output : recent

⇒ 최근 업로드 된 10 개의 곡 정보를 돌려준다.

19. 유저의 선호 장르 노래 추천

input : X

output : music

⇒ 유저의 선호 장르에서 플레이 수가 많은 순으로 20 곡을 가져온다.

20. 인기 노래 리스트 가져오기

input : X

output : popular (Array)

⇒ 재생 횟수가 많은 순으로 20 곡을 가져온다.

21. 노래 재생 횟수 추가

input : id (음악 id)

output : X

⇒ 해당 노래의 play 횟수를 1 추가해준다.

22. 팔로우 리스트 가져오기

input : X

output : follow

⇒ 내가 팔로우 한 사람들의 id 가 담긴 리스트를 준다.

23. 팔로워 리스트 가져오기

input : X

output : follower

⇒ 나를 팔로우 한 사람들의 id 가 담긴 리스트를 준다.

24. 팔로우 추가

input : userId, userNickname (String)

output : follow (Array)

⇒ follow 에 해당 id 를 추가하고 id 의 follower 리스트에 나를 추가한다.

⇒ 이미 팔로우 되어있는 경우 450 StatusCode 를 돌려준다.

25. 팔로우 취소

input : userId, userNickname (String)

output : follow (Array)

⇒ follow 에 해당 id 를 삭제하고 id 의 follower 리스트에 나를 삭제한다.

26. 인증 메일 전송

input : email, key

output : X

⇒ email 에 key 가 무엇인지 알려주는 메일을 전송한다.

27. 유저 정보 받기

input : userId (String)

output : user (id, nickname, character, followNum, followerNum, preferredGenres, follow)

⇒ 유저의 정보를 가져온다.

28. 업로드 리스트 받기(팔로우용)

input : userId (String)

output : uploadList

⇒ 해당 유저가 업로드 한 음악들의 리스트를 준다.

29. 음원 존 생성

input : timeList, locateList, zoneNumber, titleList

output : msg (String)

⇒ 음원 존을 생성한다. 이미 해당 존에 생성되어있으면 410 status 를 돌려준다.

30. 음원 존 가져오기

input : zoneNumber

output : locate, title, time

⇒ zoneNumber 에 해당하는 음악 리스트와 제목과 현재 노래의 재생시간을 보내준다.

31. 버스킹 토큰 받아오기

input : channel, role, userId

output : Token

⇒ channel 이름, 사용자의 role(Publisher/Subscriber). userId 를 받는다. Agora 토큰 서버에서 해당 채널 이름이 없을 때 role 이 Publisher 라면 채널 디렉터리에 채널 이름을 추가하고 토큰을 보내준다. 해당 채널 이름이 있고 role 이 Subscriber 라면 토큰을 보내준다.

32. 버스킹 채널 지우기

input : channel

output : string

⇒ channel 이름을 받아서 해당 channel 이 디렉터리에 있으면 삭제해준다.