**User Curating API Documentation: v2 API**

1. /mapp/v2/user/curate/list?sort=”sort”&amount=”amount”&last\_id=”last\_id”
   1. GET
   2. 자신이 쓴 큐레이팅 리스트를 조회할 때 사용하는 API이다.
   3. 요청 인자: 쿼리 파라미터로 sort, amount, last\_id를 받는다. 각각이 하는 역할은 아래와 같다:  
      sort: 1이면 오름차순 정렬, -1이면 내림차순(최신순) 정렬을 시켜준다.  
      amount: 한 로딩에 불러와 줄 오브젝트의 개수.  
      last\_id: 중요! 이게 스크롤 로딩의 핵심이다. 오름차순 정렬을 했든, 내림차순 정렬을 했든 현재 사용자가 보고 있는 리스트의 가장 마지막 Curate의 \_id 속성을 입력해야 한다. MongoDB에게 “이거 다음부터 조회 해 주세요~”라는 역할로 쓰이기 때문이다.  
      혹시나 해서 말하는데 쌍따옴표 안들어갑니다.
   4. 반환 인자: Curate 스키마의 List.
   5. 기타 사항: 스크롤 로딩할 때 사용되는 API임. 그러면 문제가, 처음 로드할 땐 어떻게 해야 하는데? 라고 물을 수 있는데, last\_id 인자 자체를 아예 안 주면 된다. 그러면 처음 로드하는 것으로 인식하고 오름/내림차순 정렬에 따른 첫 리스트를 반환 해 준다.
   6. 오류 코드
      1. 500: 서버 내부 알고리즘 오류
2. /mapp/v2/user/curate/nearbyCurating?fastWeight=”시간 빠른 정도 가중치”&distWeight=”거리 가까운 순 가중치”&starWeight=”평점 높은 순 가중치”&radius=”km 단위 반경”
   1. GET
   2. 자신의 주변에 존재하는 의사들을 추천받을 때 사용하는 API이다.
   3. 요청 인자: 쿼리 파라미터로 fastWeight, distWeight, starWeight, radius를 받는다.  
      fastWeight, distWeight, starWeight는 합쳐서 1이 나오는, 각각 0과 1 사이의 실수 값을 입력받는다. 이름 그대로 시간 빠른 순, 거리순, 평점 높은 순에 따른 가중치이다.  
      radius는 km 단위로 반경을 입력받는다. 이건 내생각에 그냥 5km로 둬도 상관 없을 듯? 그런데 유저한테 선택권을 당근마켓 지역 고르기 처럼 주는 UX가 떠오른다면 그렇게 만들어도 됨.
   4. 반환 인자: Doctor에 Psychiatry가 JOIN되고, 또 Schedule이 JOIN된 특수한 형태의 스키마를 제공한다.  
      예시를 제공하겠다:  
      텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

      AI가 생성한 콘텐츠는 부정확할 수 있습니다.Postman으로 API 테스트 할 시에 얻은 결과물이다.   
      여기서 starScore, timeScore, distanceScore은 백엔드에서 점수 계산할 때 사용하는 것들이고(최고 점수가 각각 1이 되게끔 정규화)  
      중요한건 leastTime이다. 이는 이 의사와 약속을 맺을 수 있는 가장 가까운 시각을 나타내는 것인데, 2100-01-01은 소위 말하는 “정보 없음”으로 처리하고, ‘약속 시각은 DM으로 직접 문의’ 뭐 이렇게 처리하면 된다. 그렇다고 정보 없음: 약속 시각은 DM으로 직접 문의 이런 식으로 쓰라는 말이 아니라 다른 앱들 UX처럼 기존의 가장 가까운 시각이 있던 자리에 저런 글을 써 놓아야지. UX 디자인 할 때 항상 배민, 당근, 이런 모범 사례를 참고해서 프로그래밍해라. 나도 폰에 지금 오만가지 앱 다 깔려있고 UX 벤치마킹용이다. 정말 부탁이니까 UX 디자인 조금 신경써서 해 주길 바란다. 백엔드에서 제공하는걸 그대로 뿌리지 말고 유저가 느끼기에 직관적이게끔 가공을 잘 해 달라는 소리다.
   5. 기타 사항: 심혈을 기울여 만든 API다. 리스트로 결과물이 한 번에 제공되더라도, 게임 매칭이나 소개팅처럼 한 번에 한 명씩 보여주고 뒤로 넘기는 방식으로 만들도록 하고,   
      라인, 삼각형이(가) 표시된 사진

      AI가 생성한 콘텐츠는 부정확할 수 있습니다.  
      이걸 Apple의 카메라 색조 조정 UX인 휴대 전화, 스크린샷, 휴대용 통신 장치, 모바일 기기이(가) 표시된 사진

      AI가 생성한 콘텐츠는 부정확할 수 있습니다.  
      스크린샷, 예술이(가) 표시된 사진

      AI가 생성한 콘텐츠는 부정확할 수 있습니다.  
      이거를 따라서 만들면 너무 좋을 것 같다…
3. /mapp/v2/user/curate/todayCurate
   1. POST
   2. 새로워진 큐레이팅 방식에 대응하는 큐레이팅 API이다.
   3. 요청 인자: 없음(토큰으로 본인을 인증)
   4. 반환 인자: 정상 응답(200)시 Curate 스키마 한 개
   5. 특이 사항: 이전과 달리 하루에 여러 번 수행해도 거절되지 않는다. 단, 맨 처음에 수행했던 AI 기반 고민 해석본을 하루동안 유지한다. <- 이 점을 말로 잘 풀어서 이용자에게 고지 할 것!  
      또한 Curate 스키마 던져주는걸로 [의사가 보는 내 큐레이팅 정보 미리보기] 같은 느낌의 버튼을 하나 만들어서 미리보기 할 수 있도록 해 주면 좋을 것 같다.  
      또한, 여기서 얻은 Curate 스키마의 \_id 속성을 잘 가지고 있도록 하자.
   6. 오류 코드
      1. 500: 서버 내부 알고리즘 오류
4. /mapp/v2/user/curate/showTo/:did
   1. PATCH
   2. 새로워진 큐레이팅 방식에 대응하는 큐레이팅 API이다.
   3. 요청 인자: did 자리에 Doctor의 \_id 속성을 넣어주면 된다.
   4. 반환 인자: 정상 응답(200)시 madePublicOnlyToGivenDoctor context를 반환, 실질적인 반환값 없음
   5. 특이 사항: 반환값이 없다. 따라서 어떻게 작동해야 하는가?  
      이 의사랑 대화 요청을 시도하면, 이 API를 호출함과 동시에 프론트에서 해당 의사에게의 DM으로 바로 전환, /mapp/v2/doctor/curate/view/:cid API(v2 APIs > curating > doctor\_curating\_documentation 참고)를 활용한 링크를 특수한 형태로 전달한다. 마치 카카오톡에서 서로 바로가기를 보내는 것 처럼. 이 방식은 프론트에서 많이 고민 해 봐야 할 것 같다.  
      그러면 의사가 봤을 때 그 DM을 누르면 /mapp/v2/doctor/curate/view/:cid API로 바로 접근할 수 있도록 해야 할 것이다.  
      난이도가 꽤 어려워 보이는 Challenging한 도전과제 같은데 잘 해보길 응원한다.
   6. 오류 코드:
      1. 403: 오늘 아직 시도한 큐레이팅이 없는데 이 API를 호출함
      2. 405: 그런 의사가 없음
      3. 500: 서버 내부 알고리즘 오류
5. /mapp/v2/user/curate/makePublic/:cid
   1. PATCH
   2. 새로워진 큐레이팅 방식에 대응하는 큐레이팅 API이다.
   3. 요청 인자: cid 자리에 Curate의 \_id 속성을 넣어주면 된다.
   4. 반환 인자: 정상 응답(200)시 madePublicToAll Context를 반환, 실질적인 반환값 없음.
   5. 특이 사항: 본인의 큐레이팅을 관리하는 화면에서 공개적으로 큐레이팅을 받고 싶을 때 토글 버튼같은걸로 공개할 수 있도록 만든다.
   6. 오류 코드:
      1. 401: 본인의 큐레이팅이 아님
      2. 203: 이미 공개 속성을 가진 큐레이팅 게시글임
      3. 500: 서버 내부 알고리즘 오류
6. /mapp/v2/user/curate/makePrivate/:cid
   1. PATCH
   2. 새로워진 큐레이팅 방식에 대응하는 큐레이팅 API이다.
   3. 요청 인자: cid 자리에 Curate의 \_id 속성을 넣어주면 된다.
   4. 반환 인자: 정상 응답(200)시 madePrivate Context를 반환, 실질적인 반환값 없음.
   5. 특이 사항: 본인의 큐레이팅을 관리하는 화면에서 공개 속성을 취소하고 싶을 때 호출한다.
   6. 오류 코드:
      1. 401: 본인의 큐레이팅이 아님
      2. 203: 이미 비공개 속성을 가진 큐레이팅 게시글임
      3. 500: 서버 내부 알고리즘 오류