



# API 명세서

## API 명세서

Aa 이름	구분	우선 순위	메소드	주소	request	response(json)	담당자
<u>OpenVidu</u> <u>녹화 종료</u>	실시간 모임	0순위	POST	/api/video/recording/stop	headers {  accessToken(String) } request { recording: string, userId: string, questionId: Integer }  recording : sessionId	200 OK	sooin
<u>OpenVidu</u> <u>세션 닫기</u> (세션 전체 종료)	실시간 모임	0순위	DELETE	/api/video/close-session	headers {  accessToken(String) } request { sessionName: string }	200 OK	sooin
<u>OpenVidu</u> <u>세션 참여</u> <u>자 명단 반</u> <u>환</u>	실시간 모임	0순위	GET	/api/video/sessions/{sessionId}/participants	headers {  accessToken(String) }	200 OK { "participants": [ "user1", "user2", "user3" ] }	sooin
<u>OpenVidu</u> <u>세션 연결</u>	실시간 모임	0순위	POST	/api/video/sessions/{sessionId}/connections	headers {  accessToken(String) }	200 OK { token: String // 세션 정보 }	sooin
<u>그룹 가입</u>	그룹 관리	0순위	POST	/api/groups/{groupId}/members	accessToken(String)	200 OK  성공한 경우 { message : "success" }  해당 시간에 다른 모임이 있는 경우 { message : "GroupJoinSameTime" } 인원이 다 찬 경우 { message : "GroupFull" }	우담 박
<u>OpenVidu</u> <u>세션 나가</u> <u>기 (사용자</u> <u>종료)</u>	실시간 모임	0순위	POST	/api/video/remove-user	headers {  accessToken(String) } request { sessionName: string, token : string }  세션 네임 = 그룹 아이디	200 OK	sooin







Aa 이름	구분	우선 순위	메소드	주소	request	response(json)	담당자
<u>그룹 생성</u>	그룹 관리	0순위	POST	/api/groups	headers { <b>accessToken(String)</b> } requestbody{ title : String, diseaseId : Integer, isPrivate : Boolean, privatePassword : String, isHost : Boolean, startDate : Date: YYYY-MM-DD 'T'HH:MM:SS'Z', period : Integer , meetingTime : Integer, dayOfWeek : Integer, minMember : Integer, maxMember : Integer, description: String }	201 Created  생성자가 해당 시간에 예정된 모임이 없는 경우 { message : "success" }  생성자가 해당 시간에 예정된 모임이 있는 경우 { message : "fail" }  현재 시간 보다 이전 시간에 그룹을 생성하려는 경우 { message : "timeOver" }	우담 박
<u>OpenVidu 녹화 시작</u>	실시간 모임	0순위	POST	/api/video/recording/start	headers { <b>accessToken(String)</b> } requestbody{ session: string, outputMode: string,  hasAudio: boolean,  hasVideo: boolean, }	200 OK	sooin
<u>OpenVidu 세션 생성</u>	실시간 모임	0순위	POST	/api/video/sessions	headers { <b>accessToken(String)</b> } requestbody{ customSessionId : Integer  <b>userId: string</b> }  }	200 OK  { data: { sessionId: Integer isHost: boolean } }  404 (Not Found)	sooin
<u>질문 보내기</u>	미팅관리 실시간 모임	1순위	GET	/api/meetings/{meetingId}/questions		200 OK  { data: [ { questionId : Integer, detail : String } ] }	아현 전
<u>답변 순서 전달</u>	미팅관리 실시간 모임	1순위	GET	/api/users/{groupId}/list		{ data: [ {  <b>no: integer</b> userId : String, userName : String } ] }	아현 전
<u>그룹 탈퇴</u>	그룹 관리	1순위	DELETE	/api/groups/{groupId}/members	headers { <b>accessToken(String)</b> }	200 OK	우담 박
<u>그룹 목록 조회 (마이페이지, 마이페이지)</u>	그룹 관리	1순위	POST	/api/groups/{groupId}/list	headers { <b>accessToken(String)</b> }	200 OK  { content: [ }	우담 박

Aa 이름	≡ 구분	⦿ 우선 순위	⦿ 메소드	≡ 주소	≡ request	≡ response(json)	👤 담당자
<u>이데이터 용).</u>					<pre> requestbody{  keyword : String } Query Parameter{ page: integer, size : integer, sort : Sting } </pre>	<pre> {   groupId : Integer,    title : String ,   diseaseld : Integer,    <b>diseaseName : String,</b>    isPrivate : Boolean,   privatePassword : String,   inviteCode : String,   isHost : Boolean,   startDate : Date: YYYY-MM-DD HH:MM:SS,   period : Integer,   meetingTime : Integer(0~23),   dayOfWeek : Integer,   minMember : Integer,   maxMember : Integer,   totalMember : Integer,   groupStatus : Integer (0: 예정 1: 진행중, 2 : 완료)  } ] first : Boolean, last: Boolean, totalElements : Integer, totalPages : Integer } </pre>	
<u>그룹 목록 조회</u>	그룹 관리	1순위	POST	/api/groups/list	<pre> headers { <b>accessToken(String)</b> } requestbody{  <b>keyword : String,</b> <b>diseaseld : [1,2,3],</b> <b>isHost : Boolean,</b> <b>startDate : Date: YYYY-MM-DD 'T'HH:MM:SS'Z',</b> <b>period : Integer,</b> <b>startTime : Integer,</b> <b>endTime : Integer,</b> <b>dayOfWeek : [1,2,3]</b> } Query Parameter{ page: integer, size : integer, sort : "startDate,asc" } </pre>	<pre> 200 OK  {   content: [     {       groupId : Integer,        title : String ,       diseaseld : Integer,        <b>diseaseName : String,</b>        isPrivate : Boolean,       privatePassword : String,       inviteCode : String,       isHost : Boolean,       startDate : Date: YYYY-MM-DD HH:MM:SS,       period : Integer,       meetingTime : Integer(0~23),       dayOfWeek : Integer,       minMember : Integer,       maxMember : Integer,       totalMember : Integer,       groupStatus : Integer (0: 예정 1: 진행중, 2 : 완료)      }   ]   first : Boolean,   last: Boolean,   totalElements : Integer,   totalPages : Integer } </pre>	👤 우담 박
<u>질문 끝날 때마다 STT</u>	데이터 분석 및 리뷰	1순위	POST	/api/meetings/{meetingId}/stt	<pre> headers { <b>accessToken(String)</b> } requestbody{ </pre>	<pre> 200 OK  {   transcribedText : text } </pre>	👤 우담 박

Aa 이름	≡ 구분	⦿ 우선 순위	⦿ 메소드	≡ 주소	≡ request	≡ response(json)	👤 담당자
					voiceFile : String }		
<u>신규 관심 그룹(5개)</u>	그룹 관리	2순위	GET	/api/groups/{diseaseId}/new	headers { <b>accessToken(String)</b> }	200 OK { data: [ { groupId : Integer,  title : String , diseaseId : Integer,  <b>diseaseName : String,</b>  isPrivate : Boolean, privatePassword : String, inviteCode : String, isHost : Boolean, startDate : Date: YYYY-MM-DD HH-MM-SS, period : Integer, meetingTime : Integer(11), dayOfWeek : Integer, minMembers : Integer, maxMembers : Integer, totalMembers : Integer, groupStatus : Integer (0: 예정 1: 진행중, 2 : 완료) } ] }	👤 우담 박
<u>리뷰 조회</u>	그룹 관리 리뷰 관리	2순위	GET	/api/groups/{groupId}/reviews	headers { <b>accessToken(String)</b> }	200 OK  { <b>emotionAvg : Integer,</b> data: [ { meetingId : Integer, userId : String, emotionId : Integer, summation : String, cheeringMessage : String, speechAmount : Integer, date : YYYY-MM-DD HH-MM-SS, period : Integer, progressWeeks : Integer, meetingWeek(Integer) } ] }	👤 우담 박
<u>그룹 상세 조회</u>	그룹 관리	2순위	GET	/api/groups/{groupId}	headers { <b>accessToken(String)</b> }	200 OK  { data: { <b>r</b> <b>egistered : Boolean,</b>  groupId : Integer,  title : String, diseaseId : Integer,  <b>diseaseName : String,</b>  description : String,  isPrivate : Boolean,	👤 우담 박

Aa 이름	≡ 구분	⦿ 우선 순위	⦿ 메소드	≡ 주소	≡ request	≡ response(json)	👤 담당자
						privatePassword : String, inviteCode : String, isHost : Boolean, startDate : Date: YYYY-MM-DD 'T'HH-MM:SS', period : Integer, meetingTime : Integer(0~23), dayOfWeek : Integer, minMember : Integer, maxMember : Integer, totalMember : Integer groupStatus : Integer, progressWeeks : Integer  <b>meetingStatus : Integer(0:예정, 1:진행 중: 2:완료),</b> <b>meetingId : Integer</b>  }	
일반한 미팅 조회(메인 페이지용).	미팅관리	2순위	GET	/api/meetings/upcoming	headers { accessToken(String) }	200 OK  { data: { title : String groupId : Integer meetingTime : Integer diseaseId : Integer,  <b>diseaseName : String,</b> <b>meetingDate: YYYY-MM-DD HH-MM-SS,</b> <b>meetingStatus : 0(진행 전), 1(진행 중)</b> } } }	아현 전
그룹 카운팅 (마이페이지용).	마이페이지	2순위	GET	/api/groups/count	headers { accessToken(String) }	200 OK  { <b>preGroup: integer,</b> <b>startingGroup:integer,</b> <b>endGroup,integer</b> } }	아현 전
모임 후 감정 기록 + chat gpt	데이터 분석 및 리뷰	2순위	POST	/api/userreview/{meetingId}/emotions	headers { accessToken(String) } requestbody{ emotionId : Integer }	200 OK	우담 박
내 감정 온도 조회	리뷰 관리 마이페이지 회원관리	2순위	GET	/api/users/temperature	headers { accessToken(String) }	200 OK { temperature : Integer } }	아현 전
토큰 갱신 (엑세스 토큰이 만료일 때).	회원관리	3순위	POST	/api/auth/refresh	requestbody{ userId: String, refreshToken : String }	200 OK: { <b>accessToken: String</b> refreshToken:String } 올바른 토큰이 아닐 때: Invalid refresh token	sooin

Aa 이름	≡ 구분	⊙ 우선 순위	⊙ 메소드	≡ 주소	≡ request	≡ response(json)	담당자
<u>진행 중 미팅 아이디 반환</u>	미팅관리	3순위	GET	/api/meetings/ongoing/{groupid}	headers { accessToken(String) }	{ "meetingId": 23{meetingId} }	sooin
<u>그룹 가입 시 비밀번호 확인</u>	그룹 관리	3순위	POST	/api/groups/password	headers { accessToken(String) }  { groupid : Integer,  privatePassword : String }	200 OK { true/false }  404 { "message": "해당 groupId가 존재하지 않습니다.", "code": "G1" }	sooin
<u>아이디 찾기</u>	회원관리	3순위	POST	/api/auth/userid	requestbody{ userName : String, phone: String }	200 OK  { data: { userId : String } } 403: 신고 3번 당한 user거나 탈퇴한 유저일 때 { "message": "계정이 정 지되었습니다.", "code": "U5" }	sooin
<u>로그아웃</u>	회원관리	3순위	POST	/api/auth/logout	headers { accessToken(String) }	200 OK	sooin
<u>회원 신고</u>	회원관리	3순위	POST	/api/users/reports	headers { accessToken(String) } requestbody{  reportedUserId : String, reasonId : Integer, reason : String, meetingId: Integer }	200 OK  회원이 처음 신고하고 아 직 3번 미만일때 { message : "신고가 정 상적으로 접수되었습니 다." }  이미 신고한 상태일 때 { message : "You have already reported this user for this meeting" }	sooin
<u>회원 여부 확인</u>	회원관리	3순위	POST	/api/auth/password	requestbody{ userId: String, phone: String }	200 OK  { data: { state : boolean(T/F) } } 403: 신고 3번 당한 user거나 탈퇴한 유저일 때 { "message": "계정이 정 지되었습니다.", "code": "U5" }	sooin
<u>회원 가입</u>	회원관리	3순위	POST	/api/auth/signup	requestbody{ "userId": "String", "userName" : String, "password": String, "phone" : String, "diseaseId" : Integer }	201 Created	sooin

Aa 이름	≡ 구분	⦿ 우선 순위	⦿ 메소드	≡ 주소	≡ request	≡ response(json)	👤 담당자
<u>질병 코드 전달</u>	질병관리	3순위	GET	/api/diseases		200 OK:  { date: [ { diseaseId: Integer, diseaseName : String } ] }	 sooin
<u>회원정보 수정</u>	회원관리	3순위	PATCH	/api/users/profile	headers { accessToken(String) } requestbody{ password : String, phone : String, diseaseId : Integer }	200 OK	 sooin
<u>회원 정보 확인</u>	회원관리	3순위	GET	/api/users/profile	headers { accessToken(String) }	200 OK  { data: { userId : String, userName : String, phone : String, diseaseId : Integer. }  diseaseName : String }	 sooin
<u>로그인</u>	회원관리	3순위	POST	/api/auth/login	requestbody{ "userId" :String, "password" :String }	200 OK: 로그인 성공일 때 {  accessToken: String refreshToken:String userId: string; userName: string; diseaseId: number; diseaseName: string; isAdmin: boolean; }  401: 비밀번호 틀렸을 때 { "message": "비밀번호가 일치하지 않습니다.", "code": "U2" }  404: 아이디 틀렸을 때 { "message": "존재하지 않는 아이디입니다.", "code": "U1" }  403: 신고 3번 당한 user거나 탈퇴한 유저일 때 { "message": "계정이 정지되었습니다.", "code": "U5" }	 sooin
<u>비밀번호 재설정</u>	회원관리	3순위	PUT	/api/auth/password/reset	requestbody{ userId: String, password: String }	200 OK	 sooin
<u>회원탈퇴</u>	회원관리	5순위	PATCH	/api/users/delete	headers {	200 OK	 sooin

Aa 이름	≡ 구분	⌚ 우선 순위	⌚ 메소드	≡ 주소	≡ request	≡ response(json)	👤 담당자
					<code>accessToken(String)</code> }		
<u>신고 유저 복구</u>	회원관리	5순위	PUT	/api/users/reset	headers { <code>accessToken(String)</code> } { "userId" : String, }	200 OK	👤 아현 전
<u>신고 유저 이유 전달</u>	회원관리	5순위	GET	/api/users/{userId}/reasonlist	headers { <code>accessToken(String)</code> }	200 OK [ { "reasonId" : integer, "reasonName": string, "reason" : String } ]	👤 아현 전
<u>신고 유저 목록</u>	회원관리	5순위	GET	/api/users/reportlist	headers { <code>accessToken(String)</code> }	200 OK [ { "userId" : String, "userName" : String } ]	👤 아현 전
<u>아이디 중복체크</u>	회원관리		POST	/api/auth/check-id	{ userId: string }	200 OK { true/false }	👤 sooin