

API 명세서

▼ 유저

구분	주기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)	비고
유저(user, oauth)	일반 회원가입 ▼	POST	/api/users/register	RequestPart "regist": { "type": "user", "password": "String", "name": "String", "personalld": "String", "birthDate": "Date", "email": "String", "introduction": "String" }, "profileImage": "form-data"	201 User registered successfully	
	이메일 중복 확인 🗸	POST	/api/users/duplicate/email	RequestBody "input" : String	200 ok 409 error	
	personalld 중복 확인 🗸	POST	/api/users/duplicate/nickname	RequestBody "input" : String	200 ok 409 error	
	소셜 회원가입	POST	/api/users/register	RequestPart "regist": { "type": "social", "name": "String", "personalld": "String", "birthDate": "Date", "introduction": "String", "socialDomain": "String", "socialld": "String" }, "profileImage": "form-data"	201 User registered successfully	
	카카오 소셜 ID	POST	/api/oauth/kakao	RequestParam "code": "String"	200 ok { "socialDomain": "KAKAO", "socialId": "String" }	
	구글 소셜 ID	POST	/api/oauth/google	RequestParam "code" : "String"	200 ok { "socialDomain": "GOOGLE", "socialId": "String" }	
	네이버 소셜 ID	POST	/api/oauth/naver	RequestParam "code" : "String"	200 ok { "socialDomain": "NAVER", "socialId": "String" }	
	소셜 회원가입 여부 확인 🗸	GET	/api/oauth/check	RequestBody "socialId":String	200 ok 401 error	200 - 로그인 페이지 401 - 회원 가입 필요
	일반 로그인 🗸	POST	/api/users/login	RequestBody { "email": "String", "password": "String" }	200 ok "personalld" : String "accesstoken" : String	
	소셜 로그인	POST	/api/oauth/login	RequestBody { "socialDomain": "String", "socialId": "String" }	200 ok "personalld" : String "accesstoken" : String	
	비밀번호 변경(이메일 코드 전송) 🗸	POST	/api/users/email/send	RequestBody { "email": "String", "personalId": "String" }		
	비밀번호 변경(이메일 코드 검증) 🗸	POST	/api/users/email/verify	RequestBody { "email": "String", "code": "String" }		
	비밀번호 변경 🗸	PATCH	/api/users/password	RequestBody { "email": "String", "password": "String" }		
	로그아웃 🗸	POST	/api/users/logout		200 ok	AuthenticationPrincipal 사용자 정보로 조회
	자신의 프로필 조회 ✓	GET	/api/users/get		200 ok { "type": "String", "id": "String", "name": "String", "personalld": "String", "profileImageUr!": "String", "introduction": "String", "birthDate": "Date" "followerCount": Long, "followeeCount":Long "isFollow":boolean }	

구분	주기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)	비고
	프로필 조회 ✓	GET	/api/users/{personalId}		200 ok { "type": "String", "id": "String", "name": "String", "personalld": "String", "profileImageUrl": "String", "introduction": "String", "birthDate": "Date" "followerCount": Long, "followeeCount":Long "isFollow":boolean }	이미지 없으면 null isFollow : 내 프로필은 false 반환
	유저 수정 🗸	PUT	/api/users/update	RequestPart { "name":String, "introduction":String, "interests":List <string> } "profileImage": "form-data"</string>	200 ok	AuthenticationPrincipal 사용자 정보로 조회
	사용자 검색	GET	/api/users/search	RequestBody { "word": String "size": int (default=10) "cursorId": String }	"cursorld": "String, "responses": List{ "Personalld": String, "name": String, "ProfileImageUrl": String}	cursorld : personalld 이미지 없으면 null
	JWT 재발급 🗸	POST	/api/auth/reissue		만료되면 401	

▼ 게시글

구분	주기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)
게시글(board)	홈 피드 조회 🗸	GET	/api/feed	RequestParam "limit":int "cursor":LocalDateTime	{"boardId": Long, "content": String, "personalId": String, "profileImageUr!": String, "likeCount": Long, "commentCount":Long, "keyword": "String", "imageUr!": String, "createdAt": "LocalDateTill "isRecommended": Booles "liked":boolean }
	게시글 작성 ▼	POST	/api/board	RequestPart"board":{"content": "String","x": double,"y": double,"z":int"keyword": "String","pageNumber": int,"interests":List <string>}"boardImage": "form-data"</string>	201 ok{ "boardId": long, "content": "String", "personalId": "String, "likeCount": long, "commentCount": Long, "x": double, "y": double, "z": int, "keyword": "String", "pageNumber": int, "imageUrls": List <string>, "interests": List<string>, "createdAt": "LocalDateTir" "updatedAt": "LocalDateTir"</string></string>
	게시글 상세 🗸	GET	/api/board/{boardId}		201 ok{"boardId": long,"cc "String","profileImageUrI": long,"commentCount":Lon int,"keyword": "String","pa int,"imageUrIs":List <string "localdatetime","updated<="" td=""></string>
	게시글 수정 🗸	PUT	/api/board/{boardId}	RequestPart"board":{"content": "String", "x": double, "y": double, "z":int "keyword": "String", "pageNumber": int, "interests":List <string>} "boardImage": "form-data"</string>	201 ok{"boardId": long,"cc "String","profileImageUrI": long,"commentCount":Lon int,"keyword": "String","pa int,"imageUrIs":List <string "localdatetime","updated<="" td=""></string>
	게시글 삭제 🔽	DELETE	/api/board/{boardId}		200 ok
	게시글 조회 🗸	GET	/api/board/user		200 okList{"boardId": long "String","profileImageUrI": long,"commentCount":Lon int,"keyword": "String","pa int,"imageUrIs":List <string "LocalDateTime","updated</string
	게시글 조회 🗸	GET	/api/board/{personalId}/user		200 okList{"boardId": long "String","profileImageUrI": long,"commentCount":Lon int,"keyword": "String","pa int,"imageUrIs":List <string "localdatetime","updated<="" td=""></string>
	내 프로필 게시글 조회 <mark>▼</mark>	GET	/api/board/profile		{"x": double,"y": double,"z "String","pageNumber": in

다른 프로필 게시 글 조회 <mark>✓</mark>	GET	/api/board/{personalId}/profile		{"x": double,"y": double,"z' "String","pageNumber": int
게시글 프로필 위 치 수정 <mark>✓</mark>	PATCH	/api/board/profile/update	RequestBodyList{"boardId":Long,"x":Double,"y":Double,"z":Double,"pageNumber":Integer}	200 ok
관심사 게시물 + 좋아요 순(검색 페 이지) ▼	GET	/api/board/recommend	RequestBody {size": int (default=10), "cursorLikeCount":Long,"cursorId":Long, "interestId":Long }	200 ok "cursorld": Long, "cursorLikeCount": Long, "InterestId":Long, "response": List{"boardId" "String","profileImageUr!":\$ long,"commentCount":Long "keyword":"String","imageI
게시글 검색 🗹	GET	/api/board/search	RequestBody {"word" : String" size": int (default=10)" cursorId":Long }	200 ok "cursrold":Long, "responses" List{"boardId": long,"conte "String","profileImageUrl":S long,"commentCount":Long "keyword":"String","imagel

▼ 댓글

구분	주기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)
댓글(comment)	댓글 작성 🗸	POST	/api/board/{boardId}/comment	RequsetBody{"comment":"String"}	201 ok{"commentId": Long,"content": "String","pers LocalDateTime,"updatedAt": LocalDateTime}
	댓글 조회 🗸	GET	/api/board/{boardId}/comment		200 ok List{"commentId": Long,"content": "String","persona LocalDateTime,"updatedAt": LocalDateTime}
	댓글 상세 조회 ✓	GET	/api/board/{boardId}/comment/{commentId}		200 ok {"commentId": Long,"content": "String","personalId LocalDateTime,"updatedAt": LocalDateTime}
	댓글 수정	PUT	/api/board/{boardId}/comment/{commentId}	RequsetBody{"comment":"String"}	200 ok{"commentId": Long,"content": "String","pers LocalDateTime,"updatedAt": LocalDateTime}
	댓글 삭제	DELETE	/api/board/{boardId}/comment/{commentId}		
	대댓글 작성 🗸	POST	/api/board/{boardId}/comment/{commentId}/replies	RequestBody{"content":"String}	201 ok{"replyId":Long,"personalld":"String","profileImag
	대댓글 조회 🗸	GET	/api/board/{boardId}/comment/{commentId}/replies		200 ok List{"replyId":Long,"personalId":"String","profileImag
	대댓글 수정	PUT	/api/board/{boardId}/comment/{commentId}/replies/{replyId}		200 ok{"replyId":Long,"personalId":"String","profileImag
	대댓글 삭제	DELETE	/api/board/{boardid}/comment/{commentId}/replies/{replyId}		

▼ 좋아요

				입력 데이터		
구분	주기능	메소드	RestAPI	(Request)	반환 데이터 (Response)	비고
좋아요(like)	게시글 좋아요	POST	/api/boards/{boardId}/likes		201 ok	AuthenticationPrincipal사용자 정보로 조회
	게시글 좋아요 취소 ✓	DELETE	/api/boards/{boardId}/likes			AuthenticationPrincipal사용자 정보로 조회
	게시글 좋아요 조회	GET	/api/boards/{boardId}/likes		200 okList{"personalld" : "String", profileImageUrl : "String"}	
	댓글 좋아요 🗸	POST	/api/comments/{commentId}/likes		201 ok	AuthenticationPrincipal사용자 정보로 조회
	댓글 좋아요 취소 🗸	DELETE	/api/comments/{commentId}/likes			AuthenticationPrincipal사용자 정보로 조회
	댓글 좋아요 조회	GET	/api/comments/{commentId}/likes		200 okList{"personalld" : "String", profileImageUrl : "String"}	
	대댓글 좋아요 🗸	POST	/api/replys/{replyId}/likes			AuthenticationPrincipal사용자 정보로 조회
	대댓글 좋아요 취소	DELETE	/api/replys/{replyId}/likes		200 okList{"personalld" : "String", profileImageUrl : "String"}	AuthenticationPrincipal사용자 정보로 조회
	대댓글 좋아요 조회	GET	/api/replys/{replyId}/likes			

▼ 팔로우

구분	주기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)	비고
팔로우(follow)	팔로우 🗸	POST	/api/follows/follow	RequestBody{"followeePersonalId":String}	201 ok	AuthenticationPrincipal사용자 정보로 조회
	팔로우 취소 🗸	POST	/api/follows/unfollow	RequestBody{"followeePersonalId":String}		AuthenticationPrincipal사용자 정보로 조회
	팔로워 조회 🇸	GET	/api/follows/getfollower	RequestParam { "personalld":"String" }	List{ "personalld": "String", "profileImageUrl":"String", "isFollow": boolean }	isFollow : 로그인한 사용자 기준
	팔로잉 조회 🇸	GET	/api/follows/getfollowee	RequestParam { "personalld":"String" }	List{ "personalld": "String", "profileImageUr!":"String", "isFollow": boolean }	isFollow : 로그인한 사용자 기준

팔로워 검색 🇸	GET	/api/follows/search/follower	RequestParam { "personalId":String, "word":String }	List{ "personalld": "String", "profileImageUrl":"String", "isFollow": boolean }	isFollow : 로그인한 사용자 기준
팔로위 검색 🗸	GET	/api/follows/search/followee	RequestParam { "personalld":String, "word":String }	List{ "personalld": "String", "profileImageUrl":"String", "isFollow": boolean }	isFollow : 로그인한 사용자 기준

▼ 보이스 버블

구분 주기	기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)	비고
보이스 버블 내	보이스 업로드 🗸	POST	/api/voice/upload	RequestParam { "file": form-data }	201 ok	AuthenticationPrincipal사용자 정보로 조회
ч	보이스 듣기 🗸	GET	/api/voice/listen		200 ok { "voiceld":Long, "audioUrl":String }	AuthenticationPrincipal사용자 정보로 조회
팔	로워 보이스 듣기 🗹	GET	/api/voice/{voiceld}/listen		200 ok	AuthenticationPrincipal사용자 정보로 조회
내	보이스 삭제 🗹	DELETE	/api/voice/{voiceld}		200 ok	
팔	·로위 보이스 조회 ·	GET	/api/voice/lis		200 ok List { "voiceld":Long," "personalld":String, "profileImageUrl":String, audioUrl":String }	AuthenticationPrincipal사용자 정보로 조회
보여	이스 답장 업로드 🗹	POST	/api/voice/reply/upload	RequestParam { "file":form-data, "voiceId":Long }	201 ok	AuthenticationPrincipal사용자 정보로 조회
보0	이스 답장 조회 ✓	GET	/api/voice/reply		200 ok List { "voiceReplyId":Long, "personalId":String, "profileImageUrl":String, "audioUrl":String }	AuthenticationPrincipal사용자 정보로 조회
卢		GET	/api/voice/reply/listen/{voiceReplyId}		200 ok { "voiceReplyId":Long, "personalId":String,	
	이스 답장 듣기 🗸	(SE)			"profileImageUrl":String, "audioUrl":String }	

▼ 챗봇

구분	주기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)	비고
챗봇	챗봇 답장 생성 🗸	POST	/api/chatbot	RequestBody"message": String	201 ok"message" : String	

▼ 채팅

구분	주기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)	비고
채팅	메세지 저장 🗸	POST	/api/chat/messages	RequestBody{"chatRoomld" : Long"messageContent":String"type": Enum}	{ "id": Long, "userName": String, "personalld": String, "messageContent": String, "createdAt": LocalDateTime" }	웹소켓을 통신할것
	채팅방 메세지 조회	GET	/api/chat/rooms/messages	RequestParam "chatRoomId":Long, "cursor":Long	{ "messages": List { "id": Long, "userName": String, "personalld": String, "messageContent": String, "createdAt": LocalDateTime" } , "nextCursor": Long, "hasNext": boolean }	** 채팅방'** chatParticipants 채 팅방 목록에서 채팅방입장
	채팅방 입장/생성	POST	/api/chat/rooms/messages	RequestParam"otherPersonalId":String	{"chatRoomld": Long, "otherUser": String, "profileImageUrl": String, "lastMessageContent": String, "lastMessageTime": LocalDateTime, "unReadMessageCount": Long}	프로필에서 채팅방 입장

	채팅방 조회 ✓	GET	/api/chat/rooms/list		List{"chatRoomld": Long, "otherUser": String, "profileImageUrl": String, "lastMessageContent": String, "lastMessageTime": LocalDateTime, "unReadMessageCount": Long}	대화나눈적 없으면 목록에 반환 x
	채팅 검색 🗸	GET	/api/chat/rooms/search	RequestParam { "size":int "cursorld" : Long "word":"String }	"cursorld":Long, "responses": List {"chatRoomld": Long, "otherUser": String, "profileImageUrl": String, "lastMessageContent": String, "lastMessageTime": LocalDateTime, "unReadMessageCount": Long}	cursorld : participantsID
	채팅방 기본정보 ✓	GET	/api/chat/rooms/info	RequestParam"chatRoomId":Long	{ "chatRoomId": Long , "personalId": String, "otherPersonalId": String "otherUserName", String, "otherProfileImageUrl", String }	** 채팅방
웹소켓	채팅방연결 🔽		ws://localhost:8080/ws/chat	RequestParam"chatRoomId":Long		** 채팅방
	메세지 발신 🗸		ws://localhost:8080/ws/chat	RequestParam"chatRoomld":Long, RequestBody {"chatRoomld" : Long, "messageContent":String, "type": Enum}		
	메세지 수신 🔽		ws://localhost:8080/ws/chat	RequestParam"chatRoomld":Long	{ "id": Long, "userName": String, "personalld": String, "messageContent": String, "createdAt": LocalDateTime }	

ws://localhost:8080/ws/chat

▼ 관심사

구분	주기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)	비고
관심사	유저 관심사 추가 🗸	POST	/api/interests	RequestBody { "personalld": String, "interestContents": List <string> }</string>	201 ok	
	유저 관심사 조회 🗸	GET	/api/interests/user		200 okList{"interestId": Long,"content": "String"}	AuthenticationPrincipal사용자 정보로 조회
	유저 관심사 삭제	DELETE	/api/interests			AuthenticationPrincipal사용자 정보로 조회
	전체 관심사 조회 🗸	GET	/api/interests		200 okList{"interestId": Long,"content": "String"}	

▼ 알림

구분	주기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)	비고
알림	사용자 알림 조회 🗸	GET	/api/notification	RequestParam{"size": int (default=10)"cursorId":Long}	"cursorld":Long "responses":List { "notificationId":Long, "receiverPersonalId": "String", "senderPersonalId": "String", "senderProfileImageUrl":"String", "type": Enum, "referenceId": Long, "createdAt": LocalDateTime, "isRead":boolean }	AuthenticationPrincipal사용자 정보로 조회 ENUM: FOLLOW, COMMENT, RECOMMENT, CHAT, VOICEREPLY, BOARDLIKE, COMMENTLIKE
	알림 읽음 처리 🔽	POST	/api/notification/{notificationId}/read		200 ok	
	알림 sse 연결	GET	/api/notification/subscribe			

[referenceId]

FOLLOW → 팔로우 테이블 ID(followId), 이동 필요 시 팔로우한 사람(String senderPersonalId)로 이동

RECOMMENT → 대댓글이 달린 댓글(Long commentId) -> 지금 수정해서 기존 데이터에 들어가 있는 애들은 boardId로 되어있음 테스트 할때는 새로 알림 생성해서 해야함

COMMENTLIKE → 좋아요 달린 댓글(Long commentId)

CHAT → 채팅방 ID(Long chatRoomId)

VOICEREPLY → 답장을 단 Voice(Long voiceId)

BOARDLIKE → 좋아요 달린 게시글(Long boardId)

$\underline{\mathsf{COMMENT}}$ → 댓글이 달린 게시글(Long boardId)

▼ AI

구분	주기능	메소드	RestAPI	입력 데이터(Request)	반환 데이터 (Response)	비고
AI	일반 AI 게시글 생성 🗸	POST	/api/ai/generate/board			백에서 자동화
	특정 게시글 AI 댓글 생성 ✓	GET	/api/ai/generate/comment			백에서 자동화
	랜덤 게시글 AI 댓글 생성 ✓	GET	/api/ai/generate/random/comment			백에서 자동화
	AI 기반 게시글 생성 컨트롤러 ✓	GET	/api/ai/generate/random/board			백에서 자동화
	랜덤 게시글 좋아요 🗸	GET	/api/ai/generate/random/like			백에서 자동화
	특정 게시글 좋아요 🗸	GET	/api/ai/generate/like			백에서 자동화
	AI 게시글 및 댓글에 대댓글 🗸	GET	/api/ai/generate/reply			백에서 자동화
	관심사 기반 AI 팔로우 🗸	GET	/api/ai/generate/follow			백에서 자동화