

# pong-game API document

## REST API

### auth ( Passport, Jwt ) [ sangmipa ]

CRUD	HTTP	URI	설명
로그인	GET	/auth/login	
회원가입	POST	/auth/register	
2차 인증	GET POST	/auth/2fa	- 2차 인증메일 - 2차 인증 로그인
로그아웃	DELETE	/auth/logout	
42인증	GET	/auth/ft	

### user [ younhwan ]

CRUD	HTTP	URI	설명
유저 정보	GET	/user/{id}	- 유저 정보 가져오기
유저 검색	POST	/user/search	- 유저 닉네임으로 검색하기
개인 정보 (sangmipa)	GET PATCH	/user/me	- 개인 정보 가져오기 - 개인 정보 수정하기
개인 이미지 (sangmipa)	GET	/user/image	- 개인 이미지 가져오기
유저 이미지 (sangmipa)	GET	/user/image/{id}	- 유저 이미지 가져오기

### friend [ younhwan ]

CRUD	HTTP	URI	설명
친구	GET POST	/friend	- 친구 목록 가져오기 - 친구 추가하기

CRUD	HTTP	URI	설명
	DELETE	/friend/{id}	- 친구 삭제하기

## block [ younhwan ]

CRUD	HTTP	URI	설명
차단 기능	GET POST	/block	- 차단 목록 조회하기 - 유저 차단하기

## game [ ilhna ]

CRUD	HTTP	URI	설명
게임 기록	GET POST	/game/history	- 게임 히스토리 가져오기 - 게임 히스토리 등록하기

## channel [ ilhna ]

CRUD	HTTP	URI	설명
채널	GET POST	/channel	- 채널 목록 가져오기 - 채널 생성하기

## mute [ ilhna ]

CRUD	HTTP	URI	설명
무음	GET POST	/mute?channelId={channelId} /mute	- 무음 목록 가져오기 - 무음 참여자 등록하기
삭제	DELETE	/mute/{mute_id}	

## ban [ ilhna ]

CRUD	HTTP	URI	설명
금지	GET POST	/ban?channelId={channelId} /ban	- 금지 목록 가져오기 - 금지 참여자 등록하기

## participant [ ilhna ]

CRUD	HTTP	URI	설명
참여자	GET POST	/participant?channelId={channelId} /participant	- 채널 참여자 목록 가져오기 - 채널 참여자 등록
참여자 정보 수정	PATCH	/participant/{participant_id}	- 채널 참여자 정보 수정하기
참여자 추방	POST	/participant/{participant_id}/kick	- 채널에서 추방 하기

## chatroom [ younhwan ]

CRUD	HTTP	URI	설명
채팅방	GET POST	/chatroom	- 채팅방 목록 가져오기 - 채팅방 생성하기

## chat [ younhwan ]

CRUD	HTTP	URI	설명
채팅	GET POST	/chat?destId={destId} /chat	- DM 채팅 기록 가져오기 - DM 저장하기

# WebSocket Client API

## channel

namespace	type	event	DTO	설명
/socket/channel	emit	create	title: string, visibility: \$Enums.ChannelVisibility, password: string   null _____ res: boolean, channelId: string,	- 채널 생성, 채널 에 자동 참여 - 생성 유저 OWNER 권한 부 여 _____ - 성공하면 res = true - create된 채널 id 수신
/socket/channel	emit	edit	channelId: string, title: string, visibility: \$Enums.ChannelVisibility, password: string   null _____ res: bool, message: string,	- 채널 정보 수정 (Owner만 가능) - 클라이언트에서 수정된 정보만이 아닌, 모든 정보를 전송 _____ - 성공 = true, 실 패 = false - message에 실 패 이유
/socket/channel	emit	role	channelId: string, userId: string, toUserId: string, role: \$Enum.ParticipantRole, _____ res: boolean	- OWNER(userId) 가 toUserId에 Admin권한 부여 - 채널 소유자의 권한(관리자 X)
	on	role	toUserId: string, role: \$Enum.ParticipantRole,	toUserId: role이 바뀐 userId role: 새로운 role
/socket/channel	emit	join	channelId: string, password: string   null _____ res: bool,	- 채널 입장 - PROTECTED 일 경우 서버에서 비번 검증 - 서버에서 결과를

namespace	type	event	DTO	설명
				on join으로 전달 ----- - 입장 실패 시 { res: false, message: error }
/socket/channel	on	join	channelId: string, userId: string, nickname: string, profileImageURI: string	- 채널에 들어온 유저
/socket/channel	emit	info	PartialType	- 서버에 채널 정 보 요청 ----- - 채널에 대한 정 보
/socket/channel	emit	message	channelId: string, userId: string, message: string ----- res: bool	- 채널에 메시지 보내기 - 뮤트된 사람의 메시지는 서버에 서 메시지 내용을 바꿔서 클라이언 트에 전송
/socket/channel	on	message	userId: string message: string	- 채널에서 메시지 수신
/socket/channel	emit	game	channelId: string, userId: string, toNickname: string,	- CLI 게임 신청 - [ ] Room 이름 을 channelId + userId + toUserId + c(닉 네임 순서는 알파 벳 순 정렬로 만든 다)
/socket/channel	on	game	channelId: string, userId: string, nickname: string, toUserId: string,	- 게임 신청 수신
/socket/channel	emit	leave	----- res: true	- 채널 나가기

namespace	type	event	DTO	설명
/socket/channel	on	leave	channelId: string, userId: string, nickname: string,	- 채널 나간 유저
/socket/channel	emit	mute	channelId: string, userId: string, toUserId: string, nickname: string,	- 유저 뮤트 - 채널 관리자의 권한 (소유자 X) userId: 뮤트하는 유저(관리자) toUserId: 뮤트 될 유저 nickname: 뮤트 될 유저 닉네임
/socket/channel	on	mute	channelId: string, userId: string, nickname: string,	- 뮤트된 유저
/socket/channel	emit	kick	channelId: string, userId: string, toUserId: string nickname: string,	- 유저 킥 - 채널 관리자의 권한 (소유자 X) userId: 킥하는 유저(관리자) toUserId: 킥 당할 유저
/socket/channel	on	kick	channelId: string, userId: string, nickname: string,	- 킥 당한 유저
/socket/channel	emit	ban	channelId: string, userId: string, toUserId: string nickname: string,	- 유저 밴 - 채널 관리자의 권한 (소유자 X) userId: 밴하는 유저(관리자) toUserId: 밴 당할 유저
/socket/channel	on	ban	channelId: string, userId: string, nickname: string,	- 밴 당한 유저
/socket/channel	on	error	message: string,	- 서버에서 발생한 에러 메시지

## ChatRoom

namespace	type	event	DTO	설명
/socket/chatroom	emit	join	destId: string,	- 유저 chatroom 입장 - id 유무 확인
	emit	message	destId: string, message: string	- 상대방에 메시지 보내기 - 뮤트된 사람의 메시지는 서버에서 메시지 내용을 바꿔서 클라이언트에 전송
	on	message	destId: string, message: string nickname: string	- 상대방에서 메시지 수신