## ☐ Matching Table(Android <-> Server(Spring))

ID	Microservice	url	Method	Request	Response	비고
1	user-service	/sendRegistCertification	POST	Header : No	Success: 200	# 회원가입(핸드폰 인증을 위한 서버에 문자 전송 요청)
		J :: : :::::::::::::::::::::::::::::::		Body : RequestUser(phoneNum, hashKey)		사용자는 회원가입시 휴대폰 인증을 한다.
					Fail : 400	사용자는 핸드폰인증을 위해서 핸드폰 번호를 입력한다.
					Body : 실패 메시지	사용자는 인증번호를 문자로 받는다.
2		/checkRegistCertification	POST	Header : No	Success: 200	사용자는 받은 인증번호를 어플리케이션에 입력한다. # 회원가입(핸드폰 인증을위한 서버에 인증번호 전송)
2	user-service	CheckregistCertification	POST	Body: RequestUser(phoneNum, numberId)		# 외현가급(엔드는 인증들위한 시머에 인증인오 인종) 사용자가 입력한 인증번호와 DB의 인증번호 가 같은지 확인한다.
				, , , , , , , , , , , , , , , , , , , ,	Fail: 400	사용자가 입력한 인증번호와 DB에 저장한 인증번호가 같다면
					Body : 실패 메시지	인증한 시간을 DB에 저장한다.
3	user-service	/regist	POST	Header: No	Success : 200 header : userToken	# 회원가입(서버에 정보 전송) 사용자는 사용자의 위치정보를 입력한다(googleMap이용, pointer)
				Body : RequestUser(	Fail: 400	사용자는 email, password checkpasseord를 입력한다.
				phoneNum, email, password, nickname,	raii . 400	사용자는 nickName, birth,gender,matchGender를 입력 및 선택한다.
				birthdate, gender, other_M, other_W, respect,		(nickname,birth 는 입력 gender는 선택 matchGender는 중복선택가능)
				contact, date, communication, sex, work, happy, dating, profileImage, profileImageIcon,		사용자는 moodDistance탐색 질문(respect, contect, date, communication, sex 순)을 선정한다.
				locationKOR, latitude, longitude,		사용자는 연애관 질문 (work, happy, dating)에 관한 것들을 입력한다.
				sublocationKOR, subLatitude, subLongitude,		사용자는 profilelmage를 선정한다.
		(1 15 1	DOCT	userGrade, minAge, maxAge, maxDistance)		# 취임기임(실메임에 대취 조단 취임 - 가유 메임은 4개미 취임)
4	user-service	/checkEmail	POST	Header : No Body : RequestUser(email)	Success: 200 Fail: 400	# 회원가입(이메일에 대한 중복 확인 : 같은 메일은 1개만 허용) 사용자는 회원가입시 email을 입력한다. 이때 email이 중복된 email인지
				Socy Medasocoso (cinal)	Body : 실패 메세지	확인한다.
					- ResponseUser(userUid): "overlap"	
		, IC i.i. i.	DOCT		- ResponseUser(userUid): "error"	# ID / DW 차기/베트표 이즈오 이희 시비에 묘지 저스 이번
5	user-service	/sendCertification	POST	Header: No Body: RequestUser(phoneNum, hashKey)	Success: 200 Fail: 400	# ID / PW 찾기(핸드폰 인증을 위한 서버에 문자 전송 요청) 사용자의 핸드폰으로 인증번호를 전송(hashKey포함)한다.
				The state of the s	Body : 실패 메세지	Databases 상에 같은 전화번호가 있으면, 둘다 enable처리한다.
					- ResponseUser(userUid): "overlap"	password 재설정에서 보내지는 email에 대한 password 변경처리
		/contification	DOCT	Hondon Ma	- ResponseUser(userUid): "error"	# ID / DW/ 차기/체트표 이즈오 이성 니비에 묘된 권소 스킨
6	user-service	/certificateNumber	POST	Header: No Body: RequestUser(phoneNum, numberId)	Success: 200 Fail: 400	# ID / PW 찾기(핸드폰 인증을 위한 서버에 문자 전송 요청) 사용자가 입력한 인증번호가 같은 지 확인
					Body : 실패 메세지	올바르지 않은 값일 경우, 실패
					- ResponseUser(userUid): "wrong"	
7		(final Duff no all	DOCT	Header : No	- ResponseUser(userUid): "error"	# ID / DW/ 차기//D 차기 . 체트표 버흐크 Fasail 오 차이즈다.
/	user-service	/findByEmail	POST	Body : RequestUser(phoneNum)	Success: 200  Body: ResponseUser(email)	# ID / PW 찾기(ID 찾기 : 핸드폰 번호로 Email을 찾아준다.) 사용자가 핸드폰 번호를 입력한다.
					Fail : 400	사용자의 Email을 조회 후 반환한다.
					Body: 실패메세지	
					<ul><li>ResponseUser(userUid): "no data"</li><li>ResponseUser(userUid): "error"</li></ul>	
8	user-service	/findByPassword	POST	Header : No	Success: 200	#ID/PW 찾기(PW 찾기 : 핸드폰 번호와 Email에 해당하는 사용자토큰을
				Body : RequestUser(phoneNum,email)	header : userToken	생성해준다.)
					Fail : 400 Body : 실패메세지	사용자 Password 재설정을 위해 토큰을 보내준다.
					Body: 걸때에세시 - ResponseUser(userUid): "no data"	
					- ResponseUser(userUid): "error"	
9	user-service	/resetPassword	POST	Header: userToken	Success: 200	# ID / PW 찾기(PW 재설정 : 사용자토큰, 핸드폰 번호, Email과 일치하는
				Body : ResqusetUser(email, phoneNum, password)	Fail: 400	사용자를 찾아, PW를 변경한다.) 사용자가 이메일을 입력한다.
				, , , ,		사용자가 재설정하려는 Password를 입력한다.
						사용자가 입력한 해당 이메일에 해당하는 패스워드를 변경해준다.
10	user-service	/login	POST	Header : No Body : ResquestUser(email, password)	Success : 200 header : userToken	# 로그인 사용자가 로그인(email, password)을 시도한다
				Body : Nesquestosei(einail, password)	Body : ResponseUser(All user data of VO)	사용자가 입력한 email과 password가 맞으면 사용자 정보를 반환해준다.
					Fail : 400	·
11	user-service	/autologin	POST	Header : userToken	Success: 200	# 자동 로그인
					header : userToken  Body : ResponseUser(All user data of VO)	사용자가 토큰을 보유할 시에 자동 로그인을 해준다/ 사용자가 로그인에 성공하면 갱신 된 사용자 Token을 반환해 준다.
					Fail: 400	
12	post-service	/getPosts/{page}/{postTy	POST	Header : userToken	Success: 200	# 게시물 서비스(게시물 가져오기)
		pe}			Body : List(	게시물 가져오기 - postType :
					ResponsePost(	- postrype . □ 전체 : String(total)
					Post_id, profileImageIcon, userUid,	□ 인기 : String(popular)
					nickname, locationKOR, locationENG,	□ 메인 지역 : String(areaA)
					postLikeCount, postCommentCount, postTime, postImage, postContents, page))	- page : 페이지 번호(페이지 당 100개씩 가져옴) - profilelmagelcon : 포스트를 게시한 사람의 프로필 아이콘
					Fail: 400	· userUid : 포스트를 게시한 사람의 고유식별자
		, p. , p				- nickname : 포스트를 게시한 사람의 닉네임
13	post-service	/clickLike/{postId}/{com mentId}	POST	Header: userToken  Body: RequestPost(postId, liked(commentId))	Success : 200  Body : Boolean(true or false)	# 게시물 서비스(좋아요 기능) postld인 게시물의 commentld인 댓글(게시물에 대한 좋아요라면 NULL)
				- 24)	Fail: 400	해당 좋아요가 눌리면 true, 취소되면 false
14	post-service	/registComment	POST	Header : userToken	Success: 200	# 게시물 서비스(댓글, 대댓글 등록)
				Body:		postld : 게시물의 고유식별자 commentContents : 댓글의 내용
				RequestComment(postId, commentContents, commentClass, commentGroup)		commentContents : 댓글의 내용 commentClass : 댓글(0) or 대댓글(1)
				,		commentGroup : 댓글의 그룹
						ex 1) 새로 생성하는 댓글
					Fail : 400	commentClass(0), commentGroup(null) : commentGroup은 서버에서 찾아 서 등록
						ex 2) 새로 생성하는 대댓글
						commentClass(1), commentGroup(해당 댓글의 Group)
						ex 3) 기존에 댓글이 있는 곳에 등록하는 댓글(commentClass(0),

						commentGroup(null)
						ex 4) 기존에 대댓글이 있는 곳에 등록하는 댓글 commentClass(1), commentGroup(해당 댓글의 Group)
15	user-service	/getUser	POST	Header : userToken	Success: 200	# 프로필 조회(대상 프로필 가져오기)
				Body : RequestProfile(userUid)	Body:	#Mood(프로필 가져오기) 기능 및 본인, 상대 프로필 가져오기
					RequsetProfile(profileImage, profileIcon,	
					nickname, birthdate, locationKOR, locationENG, physicalDistance, work,	
					happy, dating, moodDistance, respect,	
					contact, date, communication, sex)	
					Fail : 400	
16	post-service	/gotDosts/(page)	POST	Header : userToken	Success: 200	# 프로필 조회(대상 게시물 가져오기)
16	post-service	/getPosts/{page}	POST	Body : RequsetPost(postUid)	Body:	# 프도필 조외(대성 게시물 가서오기) 보낸 사용자(Uid)의 게시물 리스트가 보여주는 기능(본인 or 상대 게시물)
					List(	
					ResponsePost(	
					post_id, profileImageIcon, userUid,	
					nickname, locationKOR, locationENG, postLikeCount, postCommentCount,	
					postTime, postImage, postContents))	
					Fail : 400	
17	post-service	/registPost	POST	Header : userToken	Success: 200	# 게시물 서비스(게시물 등록)
				Body : ResquestPost(locationKOR, locationENG, postImage, postContents,	Fail : 400	게시물 작성
				locationENG, postImage, postContents, List(hashtag(hashtagName)))		
18	post-service	/updatePost	POST	Header : userToken	Success: 200	# 게시물 서비스(작성된 게시물 수정)
				Body : RequsetPost(postId, postContents,	Fail : 400	게시물 수정(사진 수정 불가)
				List(hashtag(hashtagName)))		
19	post-service	/deletePost	POST	Header: userToken	Success: 200	# 게시물 서비스(작성된 게시물 삭제) 게시물 삭제
20	matching-service	/getMatchingUsers	POST	Body : RequestPost(postId)  Header : userToken	Fail : 200 Success : 200	# 매칭 서비스(매칭된 상대 정보 가져오기)
	atcg service	, geamate in gesers	. 66.		Body:	매칭상대정보(2명) 가져오기
					List (Response Matching User (user Uid,	
					profilelmage, profilelcon, nickname,	
					birthdate, locationKOR, locationENG, physicalDistance, respect, contact, date,	
					communication, respect, contact, date,	
					MatchingData(moodDistance,	
					matchingTime)))	
					Fail: 400	
21	matching-service	/updateMatchingTime	POST	Header : userToken	Success: 200	# 매칭 서비스(광고 시청 후 시간 단축시키기)
	matering service	7 apadicividicining rime	1031	String("matching or profile or request or	Body:	광고 시청으로 시간 단축
				reset")	List(ResponseMatchingUser(userUid,	- 매칭 페이지 광고 : String("matching")
					profileImage, profileIcon, nickname,	- 프로필 페이지 광고 : String("profile")
					birthdate, locationKOR, locationENG,	- 받은 요청 페이지 광고 : String("request")
					physicalDistance, respect, contact, date, communication, sex	- 탈퇴 시도 시, 1회 매칭 리셋(6개월에 1번 리셋) : String("reset")
					MatchingData(moodDistance,	
					matchingTime)))	
					Fail : 400	
22	user-service	/purchaseVIP	POST	Header : userToken  RequestPurchase(orderId, packageName,	Success: 200  Body: ResponseUser(All user data of VO)	# 결제 서비스(결제에 대한 서비스 제공 요청) 유료 결제 시, 결재한 시간에 한번, 하루에 한번 지급(0시 기준). 부여되는
				productId, purchaseTime, purchaseState,	body : Nesponseoser(An user data or vo)	### 글세 시, 글세된 시년에 된단, 이후에 된단 시합(6시 기단). 부여되는     코인의 갯수는 테이블에서 가져와서 조정가능하게 함
				purchaseToken, acknowledged)	Fail : 400	- purchaseContents : monthly_item_1(1개월 권), monthly_item_2(6
						개월 권), monthly_item_3(12개월 권)
23	user-service	/setMatching	POST	Header: userToken	Success: 200	# 사용자 설정(사용자의 설정 변경)
				RequsetUser(All user data of VO)	Fail: 400	매칭설정(상대성별 설정, 최대거리 설정, 상대 연령대 설정, 닉네임설정(코인), 무드거리설정(코인))(코인or티켓 설정), 프로필, 프로필아이콘
						유료 기능 : 핸드폰, 닉네임, 무드거리, 로밍, 지역
24	user-serivce	/blockPhoneNums	POST	Header : userToken	Success: 200	# 사용자 설정(연락처 차단 요청)
		, , , = .		Body : List(phoneNum)	Fail: 400	연락처 차단(사용자가 지정한 번호 목록 차단)
25	user-service	/updatePhoneNum	POST	Header : userToken  Body : RequestUser(phoneNum)	Success: 200 Fail: 400	# 사용자 설정(핸드폰 번호 변경) 전화번호 재설정 : (유료기능)핸드폰 인증 성공하면(6번) 받은 핸드폰 번호
				body . Nequestoset(phonetvuitt)	raii . 400	전와면도 세설성 : (유료기능)앤드폰 인증 성공아면(6면) 받은 앤드폰 면도 로 수정하여 저장
26	user-service	/openCheating	POST	Header : userToken	Success: 200	# 채팅 서비스(채팅방 개설로 인한 요금 차감)
				Body : Integer(coin), Integer(ticket)	Fail : 400	채팅방 개설(유료기능)
27	post-service	/getHashtag/{hashtagNa	POST	Header : userToken	Success: 200	# 게시물 서비스(해쉬태그로 게시물 검색)
		me}/{page}			Body : List(	해쉬태그로 검색한 게시물들에 대한 리스트
					ResponsePost(	
					post_id, profileImageIcon, userUid,	
					nickname, locationKOR, locationENG,	
					postLikeCount, postCommentCount,	
					postTime, postImage, postContents)) Fail: 400	
28	lock-service	/addLockUser	POST	Header:userToken	Success: 200	# 신고 서비스(대상 신고)
	-			Body : RequestLockUser(lockUserUid,		신고하기 기능(제재 대상, 제재타입, 제재사유, 참고할 값의 UID)
				lockType, lockReasons, referUid)	Fail: 400	제재 타입 : 프로필사진 신고("profileImage"),
						프로필 질문 신고("profileQuestion"),
29	post-service	/getPost/{postId}	GET	Body : String(postId)	Success: 200	프로필닉네임 신고("profileNickname"), 채팅신고("cheating") #게시물 서비스(해당 게시물 가져오기)
	post service	, 30% 000 (postia)	<u> </u>	- 22, . 239(603000)	Body: ResponsePost(All data of VO)	추후 SNS서비스에 URL공유만으로 게시물 공유가 가능하게 끔 GET으로
					Fail : 400	구현.
30	user-service	/checkNickname	POST	Header : No	Success: 200	# 회원가입(닉네임 중복 확인)
24	Hear same	/gotPlockPlace - N.	DOCT	Body : RequestUser(nickname)	Fail: 400	차다되 저하버츠 모로
31	user-service	/getBlockPhoneNums	POST	Header : userToken Body : No	Success: 200  Body: List <string> phoneNum</string>	차단된 전화번호 목록
					Fail: 400: null	
22	post-service	/deleteComment	POST	Header : userToken	Success: 200	댓글 삭제
32						

			Body: RequestComment(commentId)	Fail: 400
--	--	--	---------------------------------	-----------

## □ VO(Android <-> Server(Spring))

ID	Microservice	VO(Only Communication Model)	Parameter	Parameter 설명
1	user-service	RequestUser	- userUid	- 사용자의 고유식별자
			- phoneNum	- 핸드폰 번호
			- email	- 이메일
			- password	- 비밀번호
			- nickname	- 별명
			- birthdate	- 생년월일
			- gender	- 성별
			- otherM	- 매칭하고자 하는 상대(남), 희망하면 True, 비희망하면 False
			- otherW	- 매칭하고자 하는 상대(여), 희망하면 True, 비희망하면 False
			- respect	- 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요)
			- contact - date	- 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 :(수정 시, 티켓 or 코인 필요)
			- communication	- 무드거리 단필질문 3:(무용 시, 티켓 or 코인 필요) - 무드거리 판별질문 4:(수정 시, 티켓 or 코인 필요)
			- sex	- 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요)
			- work	- 질문 1 : 어떤 일을 하는지?
			- һарру	- 질문 2 : 얼마나 행복한지?
			- dating	- 질문 3 : 얼마나 데이트를 중요하게 여기는지?
			- profileImage	- 프로필 이미지
			- profileImageIcon	- 프로필 아이콘
			- locationKOR	- 지역 위치(한글)
			- latitude	- 위도
			- longitude	- 경도
			- sublocationKOR	- (로밍을 위한) 지역 위치(한글)
			- subLatitude	- (로밍을 위한) 위도
			- subLongitude	- (로밍을 위한) 경도
			- userGrade	- 사용자 등급(VIP: VIP, Other: Basic
			- minAge	- 최소 나이(Default: ((생년월일로 계산된 나이) - 5 >=20) 최대 LtO((Default: ((생년월일로 계산된 LtOL) +5))
			<ul><li>maxAge</li><li>maxDistance</li></ul>	- 최대 나이(Default : ((생년월일로 계산된 나이) +5)) - 최대 거리(Default : 50km)
			- maxDistance - numberId	- 쇠내 거리(Default : 50km) - 인증번호(핸드폰 인증시, 사용)
			- hashkey	- 핸드폰 문자 전송 시, 같이 보내줄 해쉬키
2	user-service	ResponseUser	- userToken	- 사용자의 Token(로그인 시, 발급. 로그아웃 시, 폐기)
		'	- userUid	- 사용자의 고유식별자
			- phoneNum	- 핸드폰 번호
			- email	- 이메일
			- nickname	- 별명
			- birthdate	- 생년월일
			- gender	- 성별
			- otherM	- 매칭하고자 하는 상대(남), 희망하면 True, 비희망하면 False
			- otherW	- 매칭하고자 하는 상대(여), 희망하면 True, 비희망하면 False
			- respect	- 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요)
			- contact	- 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요)
			- date	- 무드거리 판별질문 3:(수정 시, 티켓 or 코인 필요)
			- communication	- 무드거리 판별질문 4:(수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5:(수정 시, 티켓 or 코인 필요)
			- sex - work	- 구드기니 현실글군 5.(구영 시, 디켓 이 교한 글요) - 질문 1:어떤 일을 하는지?
			- happy	- 질문 2: 얼마나 행복한지?
			- dating	- 질문 3: 얼마나 데이트를 중요하게 여기는지?
			- profileImage	- 프로필 이미지
			- profilelmagelcon	- 프로필 아이콘
			- locationKOR	- 지역 위치(한글)
			- locationENG	- 지역 위치(영어)
			- latitude	- 위도
			- longitude	- 경도
			- subLocationKOR	- (로밍을 위한) 지역 위치(한글)
			- subLocationENG	- (로밍을 위한) 지역 위치(영어)
			- subLatitude	- (로밍을 위한) 위도
			- subLongitude	- (로밍을 위한) 경도
			<ul><li>userGrade</li><li>minAge</li></ul>	- 사용자 등급(VIP : VIP, Other : normal - 최소 나이(Default : ((생년월일로 계산된 나이) - 5 >=20)
			- minAge - maxAge	- 최조 다이(Default: ((영단월일로 계산된 다이) - 5 >=20) - 최대 나이(Default: ((생년월일로 계산된 나이) +5))
			- maxAge - maxDistance	- 최대 다이(Default: ((중단설글로 개단된 다이) +3)) - 최대 거리(Default: 50km)
			- ticket	- 티켓(유료 자산)
			- coin	- 코인(유료 자산)
			- resetMatching	- 사용자가 탈퇴 시도 시, 매칭 다시 시도(6개월 1회)B
			- userLock	- 사용자 잠금 기능(boolean(default : false))
			- userLockReasons	- 제재사유 : default(null) profile신고("profile"), 게시물신고("post"), 채팅신고("cheating"), 댓글신고 ("comment")
3	post-service	RequestPost	- postId	- 게시물에 대한 고유 식별자
			- postUid	- 게시물을 올린 사용자의 고유식별자
			- locationENG	- 지역(영어)
			- locationKOR	- 지역(한글)
			- postImage	- 게시물 사진
			<ul><li>postContents</li><li>commentContents</li></ul>	- 게시물 내용 - 대, 댓글 내용
			- commentContents - commentClass	- 내, 댓글 내용   - 댓글 분류
			- commentGroup	- 것을 판규   - 댓글 그룹
			- List: hashtag	- 것을 <del>그룹</del> - 해쉬태크에 대한 리스트
			☐ hashtagName	□ 해쉬태그 이름
			- List: liked	- like 객체 리스트
			□ commentId	□ 댓글에 대한 고유 식별자
4	post-service	ResponsePost	- postId	- 게시물의 고유 식별자(게시물을 추적가능하게 하는 역할)
			- profileImageIcon	- 포스트를 게시한 사람의 프로필 아이콘
			- postUid	- 포스트를 게시한 사람의 고유 식별자

	ı			
			- nickname	- 포스트를 게시한 사람의 별명
			- locationKOR	- 포스트를 게시한 사람의 지역(한글)
			- locationENG	- 포스트를 게시한 사람의 지역(영어)
			- postLikeCount	- 포스트의 좋아요 갯수
			- postCommentCount	- 포스트의 댓글 갯수
			- postTime	- 포스트를 올린 시간
			- postImage	- 포스트의 사진
			- postContents	- 포스트의 내용(Hashtag가 내용안에 포함되어 있어, 따로 넣어주지 않아도 됨)
			- List: commentInfo	- 댓글, 대댓글에 대한 사항
			□ commentId	□ 댓글의 ID(고유식별자)
			□ commentUid	□ 댓글을 달은 사용자의 고유식별자
			□ nickname	□ 댓글을 달은 사용자의 닉네임
			□ profileImageIcon	□ 댓글을 달은 사용자의 프로필 아이콘
			□ commentContents	□ 댓글의 내용
			□ commentClass	□ 댓글인지, 대댓글인지 구분(0:댓글, 1:대댓글)
			□ commentGroup	□ 댓글에 대한 그룹핑(같은 숫자이면 같은 댓글안에 포함)
			□ commentTime	□ 댓글을 달은 시간(서버시간 기준)
			□ commentLikeCount	□ 댓글의 좋아요 수
5	user-service	RequsetProfile	- userUid	- 요청된 사용자의 고유식별자
6	user-service	ResponseProfile	- profileImage	- 요청된 프로필의 사진
	U3C1-3E1 VICE	Responser folic		
			- profileIcon	- 요청된 프로필의 아이콘
			- nickname	- 요청된 사용자의 별명
			- birthdate	- 요청된 사용자의 생년월일
			- location	- 요청된 사용자의 지역(영어)
			- physicalDistance	- 요청된 사용자의 물리적 거리
			- work	- 질문 1 : 어떤 일을 하는지?
			- happy	- 질문 2 : 얼마나 행복한지?
				- 글문 2. 글마다 용독한지: - 질문 3: 얼마나 데이트를 중요하게 여기는지?
			- dating	
			- moodDistance	- 나와의 Mood거리
			- respect	- 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요)
			- contact	- 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요)
			- date	- 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요)
			- communication	- 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요)
				- 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요)
		D M. I. II		
7	matching-service	ResponseMatchingUser	- userUid	
			- profileImage	- 프로필 사진
			a	
			- profilelcon	- 프로필 아이콘
			- profilelcon - nickname	- 프로필 아이곤 - 매칭상대의 별명
			· ·	
			- nickname - birthdate	- 매칭상대의 별명 - 매칭상대의 생년월일
			<ul><li>nickname</li><li>birthdate</li><li>locationKOR</li></ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글)
			<ul><li>nickname</li><li>birthdate</li><li>locationKOR</li><li>locationENG</li></ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어)
			<ul> <li>nickname</li> <li>birthdate</li> <li>locationKOR</li> <li>locationENG</li> <li>physicalDistance</li> </ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리
			<ul><li>nickname</li><li>birthdate</li><li>locationKOR</li><li>locationENG</li></ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요)
			<ul> <li>nickname</li> <li>birthdate</li> <li>locationKOR</li> <li>locationENG</li> <li>physicalDistance</li> </ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요)
			<ul> <li>nickname</li> <li>birthdate</li> <li>locationKOR</li> <li>locationENG</li> <li>physicalDistance</li> <li>respect</li> </ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요)
			<ul> <li>nickname</li> <li>birthdate</li> <li>locationKOR</li> <li>locationENG</li> <li>physicalDistance</li> <li>respect</li> <li>contact</li> </ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요)
			<ul> <li>nickname</li> <li>birthdate</li> <li>locationKOR</li> <li>locationENG</li> <li>physicalDistance</li> <li>respect</li> <li>contact</li> <li>date</li> </ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3: (수정 시, 티켓 or 코인 필요)
			<ul> <li>nickname</li> <li>birthdate</li> <li>locationKOR</li> <li>locationENG</li> <li>physicalDistance</li> <li>respect</li> <li>contact</li> <li>date</li> <li>communication</li> <li>sex</li> </ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요)
			<ul> <li>nickname</li> <li>birthdate</li> <li>locationKOR</li> <li>locationENG</li> <li>physicalDistance</li> <li>respect</li> <li>contact</li> <li>date</li> <li>communication</li> <li>sex</li> <li>MatchingData</li> </ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 막드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매칭한 데이터
			<ul> <li>nickname</li> <li>birthdate</li> <li>locationKOR</li> <li>locationENG</li> <li>physicalDistance</li> <li>respect</li> <li>contact</li> <li>date</li> <li>communication</li> <li>sex</li> <li>MatchingData</li> <li>moodDistance</li> </ul>	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리
			- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 병명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매칭한 데이터
8	user-service	RequestPurchase	<ul> <li>nickname</li> <li>birthdate</li> <li>locationKOR</li> <li>locationENG</li> <li>physicalDistance</li> <li>respect</li> <li>contact</li> <li>date</li> <li>communication</li> <li>sex</li> <li>MatchingData</li> <li>moodDistance</li> </ul>	- 매칭상대의 병명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3:(수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4:(수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5:(수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 □ 매칭 완료 시간 - 구매 식별자
8	user-service	RequestPurchase	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 병명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매칭한 데이터
8	user-service	RequestPurchase	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 병명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3:(수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4:(수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5:(수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 □ 매칭 완료 시간 - 구매 식별자
8	user-service	RequestPurchase	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 병명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 - 무드 거리 - 매칭 완료 시간 - 구매 식별자 - 구매 제품 식별자
8	user-service	RequestPurchase	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3:(수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4:(수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5:(수정 시, 티켓 or 코인 필요) - 메칭한 데이터 - 무드 거리 - 매칭 완료 시간 - 구매 내키지 이름 - 구매 제품 식별자 - 구매 제품 식별자
8	user-service	RequestPurchase	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 - 무드 거리 - 매칭 완료 시간 - 구매 식별자 - 구매 제품 식별자 - 구매 제품 식별자 - 구매한 상대
8	user-service	RequestPurchase	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 □ 매칭 완료 시간  - 구매 작별자 - 구매 제품 식별자 - 구매한 시간 - 구매한 상태 - 구매한 상태 - 구매한 상태 - 구매한 상태
8			- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 □ 매칭 완료 시간 - 구매 식별자 - 구매 제품 식별자 - 구매 제품 식별자 - 구매한 시간 - 구매한 상태 - 구매 토큰 - 인정여부
8	user-service	RequestPurchase	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 □ 매칭 완료 시간  - 구매 작별자 - 구매 제품 식별자 - 구매한 시간 - 구매한 상태 - 구매한 상태 - 구매한 상태 - 구매한 상태
			- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 □ 매칭 완료 시간 - 구매 식별자 - 구매 제품 식별자 - 구매 제품 식별자 - 구매한 시간 - 구매한 상태 - 구매 토큰 - 인정여부
			- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData - moodDistance - matchingTime - orderId - packageName - productid - purchaseTime - purchaseState - purchaseToken - acknowledged - lockUserUid	- 매청상대의 별명 - 매청상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 메칭한 테이터 □ 무드 거리 □ 매칭 완료 시간 - 구매 식별자 - 구매 제품 식별자 - 구매한 시간 - 구매한 상대 - 구매한 상대 - 구매한 상대 - 구매한 상대 - 구매 토큰 - 인정여부 - 신고대상 UID
			- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData - moodDistance - matchingTime  - orderId - packageName - productid - purchaseTime - purchaseState - purchaseToken - acknowledged - lockUserUid - lockType - lockReasons	- 매칭상대의 별명 - 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 □ 매칭 완료 시간 - 구매 시별자 - 구매 파키지 이름 - 구매 제품 식별자 - 구매한 시간 - 구매한 상태 - 구매한 상태 - 구매한 상태 - 구매한 보다 - 건대상 UID - 신고대상 UID - 신고다상 UID - 신고다상 UID - 신고나입 : 프로필사전 신고('profile/mage'), 프로필 질문 신고(''profileQuestion'), 프로필닉네임 신고(''profileNickname'), 채팅신고(''cheating') - 신고 사유
9	lock-service	RequestLockUser	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매청상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매청한 데이터 □ 무드 거리 □ 매칭 완료 시간  - 구매 시별자 - 구매 패키지 이름 - 구매 제품 식별자 - 구매한 시간 - 구매한 상태 - 구매한 상태 - 구매 토큰 - 인정여부  - 신고대상 UID - 신고대상 UID - 신고타입 : 프로필사진 신고('profileImage'), 프로필 질문 신고('profileQuestion'), 프로필닉네임 신고('profileNickname'), 채팅신고('cheating') - 신고 사유 - 참조값 UID
			- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData - moodDistance - matchingTime - orderId - packageName - productid - purchaseTime - purchaseState - purchaseToken - acknowledged - lockUserUid - lockType - lockReasons - referUid - phoneNum	- 매청상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 무드거리 매청 완료 시간 - 무드 거리 매청 완료 시간 - 구매 제품 식별자 - 구매 제품 식별자 - 구매한 시간 - 구매한 시간 - 구매한 신간 - 구매한 신간 - 구매한 신간 - 구매한 신간 - 구매한 상태 - 구매 토큰 - 인정여부 - 신고대상 UID - 신고타입: 프로필사진 신교(*profile/mage*), 프로필 질문 신교(*profileQuestion*), 프로필닉네임 신교(*profileNickname*), 채팅신교(*cheating*) - 신고 사유 - 참조값 UID - 차단할 핸드폰 번호
9	lock-service user-service	RequestLockUser  RequestBlockUser	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 - 무드 거리 매칭 완료 시간 - 구매 시별자 - 구매 패키지 이름 - 구매 제품 식별자 - 구매한 시간 - 구매한 신간 - 구매한 상태 - 구매한 신간 - 구매한 상태 - 구매한 신고대상 UID - 신고대상 UID - 신고대상 UID - 신고다상 UID - 신고나와 : 프로필사진 신고(*profile(mage*), 프로필 질문 신고(*profileQuestion*), 프로필닉네임 신고(*profileNickname*), 제팅신고(*cheating*) - 신고 사유 - 참조값 UID - 차단할 핸드폰 번호 - 차단 여부
9	lock-service	RequestLockUser	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData - moodDistance - matchingTime - orderId - packageName - productid - purchaseTime - purchaseState - purchaseToken - acknowledged - lockUserUid - lockType - lockReasons - referUid - phoneNum - disabled - phoneNum	- 매칭상대의 생년월일 - 지역(영어) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 □ 매칭 완료 시간 - 구매 식별자 - 구매 제품 식별자 - 구매 제품 식별자 - 구매 제품 식별자 - 구매 전용교 시간 - 구매한 상태 - 구매 토른 - 인정여부 - 신고 다상 UID - 신고 타입: 프로필사진 신고(*profileimage*), 프로필 질문 신고(*profileQuestion*), 프로필닉데임 신고(*profileNickname*), 채팅신고(*cheating*) - 신고 사유 - 참조값 UID - 차단할 앤드폰 번호 - 차단 여부
9	lock-service user-service	RequestLockUser  RequestBlockUser	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 생년월일 - 지역(한글) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 □ 매칭 완료 시간 - 구매 식별자 - 구매 패키지 이름 - 구매 제품 식별자 - 구매한 상태 - 구매한 상태 - 구매한 상태 - 구매한 상태 - 구매한 상대 - 구매한 산대 - 산고대상 UID - 신고대상 UID - 신고다상 UID - 신고나와 UID - 신고나와 내고 프로필시진 신고(*profileImage*), 프로필 질문 신고(*profileQuestion*), 프로필닉네임 신고(*profileNickname*), 제팅신고(*cheating*) - 신고 사유 - 참조값 UID - 차단할 핸드폰 번호 - 차단 이부
9	lock-service user-service user-service	RequestLockUser  RequestBlockUser  ResponseBlockUser	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData - moodDistance - matchingTime - orderId - packageName - productid - purchaseTime - purchaseState - purchaseState - purchaseToken - acknowledged - lockUserUid - lockType - lockReasons - referUid - phoneNum - disabled - phoneNum - disabled	- 매칭상대의 생년월일 - 지역(영어) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 매칭한 데이터 □ 무드 거리 □ 매칭 완료 시간 - 구매 식별자 - 구매 제품 식별자 - 구매 제품 식별자 - 구매 제품 식별자 - 구매 전용교 시간 - 구매한 상태 - 구매 토른 - 인정여부 - 신고 다상 UID - 신고 타입: 프로필사진 신고(*profileimage*), 프로필 질문 신고(*profileQuestion*), 프로필닉데임 신고(*profileNickname*), 채팅신고(*cheating*) - 신고 사유 - 참조값 UID - 차단할 앤드폰 번호 - 차단 여부
9 10 11	lock-service user-service	RequestLockUser  RequestBlockUser	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData - moodDistance - matchingTime - orderId - packageName - productid - purchaseTime - purchaseState - purchaseToken - acknowledged - lockUserUid - lockType - lockReasons - referUid - phoneNum - disabled - postId	- 매칭상대의 생년일일 - 지역(현급) - 지역(영어) - 물리적 거리 - 무드거리 판별질문 1: 존중 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 2: 연락 (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 3: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 4: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 무드거리 판별질문 5: (수정 시, 티켓 or 코인 필요) - 마칭한 테이터 □ 무드 거리 □ 매칭 완료 시간 - 구매 시발자 - 구매 패키지 이름 - 구매 제품 식별자 - 구매한 시간 - 구매한 시간 - 구매한 상태 - 구매한 상태 - 구매 토큰 - 인정여부 - 신고다상 UID - 신고타입: 프로필시진 신고(profileImage"), 프로필 필문 신고("profileQuestion"), 프로필닉네임 신고("profileNickname"), 체팅신고("cheating") - 선고 사유 - 참조값 UID - 차단한 핸드폰 번호 - 차단 여부
9 10 11	lock-service user-service user-service	RequestLockUser  RequestBlockUser  ResponseBlockUser	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData - moodDistance - matchingTime - orderId - packageName - productid - purchaseTime - purchaseState - purchaseState - purchaseToken - acknowledged - lockUserUid - lockType - lockReasons - referUid - phoneNum - disabled - postId - postId - postId - commentId	- 매칭상대의 병년월일 - 지역(영급) - 무도거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 메칭한 데이터 □ 막도 거리 □ 매칭 한료 시간 - 구매 작별자 - 구매 재품 식별자 - 구매 보기지 이름 - 구매 전품 시별자 - 구매한 시간 - 구매한 시간 - 구매한 신간 - 구매한 신간 - 구매한 성대 - 구매 토큰 - 인정이부 - 신고타성 UID - 신고타성 UID - 산고타성 UID - 차단할 핸드폰 번호 - 차단 여부 - 차단할 핸드폰 번호 - 차단 여부 - 차단할 핸드폰 번호 - 차단 여부 - 게시물 식별자 - 개시물 식별자
9 10 11	lock-service user-service user-service	RequestLockUser  RequestBlockUser  ResponseBlockUser	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData	- 매칭상대의 생년월일 - 지역(연급) - 지역(연급) - 지역(연급) - 지역(영어) - 물리적 거리 - 무도거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 메칭한 테이터 □ 무드 거리 □ 매칭 안료 시간 - 구매 패키지 이름 - 구매 제품 식별자 - 구매 제품 식별자 - 구매 전 보다상 UID - 신고타상 UID - 신고타상 UID - 신고타상 UID - 사단할 핸드폰 번호 - 차단 여부 - 차단한 현무 - 차단한 현무 - 차단한 현무 - 차단한 현무 - 개시물 식별자 - 댓글 식별자 - 댓글 식별자 - 개시물 식별자 - 개시물 식별자 - 가단한 현무 변호 - 차단 여부 - 개시물 식별자 - 댓글 식별자
9 10 11	lock-service user-service user-service	RequestLockUser  RequestBlockUser  ResponseBlockUser	- nickname - birthdate - locationKOR - locationENG - physicalDistance - respect - contact - date - communication - sex - MatchingData - moodDistance - matchingTime - orderId - packageName - productid - purchaseTime - purchaseState - purchaseState - purchaseToken - acknowledged - lockUserUid - lockType - lockReasons - referUid - phoneNum - disabled - postId - postId - postId - commentId	- 매칭상대의 병년월일 - 지역(영급) - 무도거리 판별질문 1 : 존중 (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 2 : 연락 (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 3 : (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 4 : (수정 시, 티켓 or 코인 필요) - 무도거리 판별질문 5 : (수정 시, 티켓 or 코인 필요) - 메칭한 데이터 □ 막도 거리 □ 매칭 한료 시간 - 구매 작별자 - 구매 재품 식별자 - 구매 보기지 이름 - 구매 전품 시별자 - 구매한 시간 - 구매한 시간 - 구매한 신간 - 구매한 신간 - 구매한 성대 - 구매 토큰 - 인정이부 - 신고타성 UID - 신고타성 UID - 산고타성 UID - 차단할 핸드폰 번호 - 차단 여부 - 차단할 핸드폰 번호 - 차단 여부 - 차단할 핸드폰 번호 - 차단 여부 - 게시물 식별자 - 개시물 식별자

## ☐ JSON Value

ID	Microservice	url	Method	Request	Response
1	user-service	/sendRegistCertification	POST	Header : No	m
				Body : RequestUser(phoneNum, hashKey)	Fail : 400
					Body : 실패 메시지
2	user-service	/checkRegistCertification	POST	Header : No	Success: 200
				Body : RequestUser(phoneNum, numberId)	Fail : 400
					Body : 실패 메시지

3	
<pre>Fail: 400  "phoneNum":"01012341234",  "email":"seojeonghyeon@gmail.com",  "password":"test123456",  "nickname":"Lime-IT",  "birthdate":"1995-01-01",  "gender":true,  "otherM":false,</pre> Fail: 400	
<pre>"phoneNum":"01012341234",     "email":"seojeonghyeon@gmail.com",     "password":"test123456",     "nickname":"Lime-IT",     "birthdate":"1995-01-01",     "gender":true,     "otherM":false,</pre>	
<pre>"password":"test123456",     "nickname":"Lime-IT",     "birthdate":"1995-01-01",     "gender":true,     "otherM":false,</pre>	
<pre>"nickname":"Lime-IT",     "birthdate":"1995-01-01",     "gender":true,     "otherM":false,</pre>	
"birthdate":"1995-01-01",  "gender":true,  "otherM":false,	
"otherM":false,	
"respect":"5",	
"contact":"3", "date":"5",	
"communication":"4",	
"sex":"3",	
"work":"그들은 이상은 고행을 곳이 들어 황금시대다. 새가 밥을 이 되려 나와, 철환하였는가? 사랑의 인생에 수 그리하였는가? 피고, 사랑의 장식하는	
우리 때문이다. 같이, 되는 이것이야말로 불러 풍부하게 고동을 보내는 우리는	
아니다. 찬미를 동력은 같지 뿐이다. 수 것이 주며, 인류의 자신과 이상의 충분	
히 열락의 아니다. 가치를 더운지라 석가는 생명을 같은 풀이 것이다.보라, 없	
으면 같지 것이다. 예수는 이상을 있는 힘있다. 따뜻한 커다란 피부가 미인을 기관과 같은 같지 사막이다. 얼음이 그들에게 용감하고 칼이다. 뭇 인도하겠다	
는 천지는",	
"happy":"곳으로 행복스럽고 예가 현저하게 것이다. 인생의 품었기 얼마	
나 위하여 대고, 가는 끓는 예수는 끓는다. 가장 청춘 끓는 같으며, 사라지지	
천고에 가슴이 쓸쓸하랴? 청춘 인류의 가지에 장식하는 뜨고, 피에 능히 품으	
며, 그리하였는가? 그들을 용기가 대고, 산야에 그들은 피고 듣는다. 열락의	
듣기만 튼튼하며, 불어 산야에", "dating":"피는 얼마나 영락과 되는 일월과 이상의 예가 부패를 것이다.	
이성은 있을 청춘의 노년에게서 어디 거선의 용기가 가지에 이상, 봄바람이다.	
이성은 역사를 불어 얼마나 따뜻한 설레는 얼음에 있으랴? 청춘의 더운지라 영원히 바이며,",	
"profileImage":"https://firebasestorage.googleapis.com/v0/b/mo	
od- d39e8.appspot.com/o/profileIcon%2FJPEG ApCSJdgmkSM7MJKwpI2OK5d	
d39e8.appspot.com/o/profile1con%2FJPEG_ApCSJdgmkSM/MJKwp12OK5d DbX73_20210808_170829png?alt=media&token=3cc90c7c-05a6-4764- 9b6f-b5143b7b02e5",	
"profileImageIcon":"https://firebasestorage.googleapis.com/v0/	
b/mood-	
d39e8.appspot.com/o/profileIcon%2FJPEG_ApCSJdgmkSM7MJKwpI2OK5d	
DbX73_20210808_170829png?alt=media&token=3cc90c7c-05a6-4764- 9b6f-b5143b7b02e5",	
"locationKOR":"부천시",	
"locationENG": "Bucheon",	
"latitude":"41.40338", "longitude":"2.17403",	
"subLocationKOR":"",	
"subLocationENG":"", "subLatitude":"",	
"subLongitude":"",	
"userGrade":"newbie",	
<pre>"minAge":"21",</pre>	
"maxDistance":"50"	
4 user-service /checkEmail POST Header : No Success : 200	
Body:	
{"email":"seojeonghyeon@gmail.com"}	
5 user-service /sendCertification POST Header : No Success : 200	
Body : RequestUser(phoneNum, hashKey) Fail : 400 Body : 실패 메세지	
	lser(userUid): "overlap"
- Respon	onseUser(userUid): "error"
6 user-service /certificateNumber POST Header : No Success : 200	
Body : RequestUser(phoneNum, numberId) Fail : 400 Body : 실패 메세지	
	lser(userUid): "wrong"
	onseUser(userUid): "error"
7 user-service /findByEmail POST Header : No Success : 200	
Body : RequestUser(phoneNum)  Body : ResponseUser	r(email)
Fail: 400	
Body : 실패메세지	ser(userHid): "no data"
	lser(userUid): "no data" onseUser(userUid): "error"
8 user-service /findByPassword POST Header : No Success : 200	·
Body : RequestUser(phoneNum,email) header : userToken	
Fail : 400 Body : 실패메세지	
	lser(userUid): "no data"
	onseUser(userUid): "error"
9 user-service /resetPassword POST Header : userToken Success : 200	
Body : ResqusetUser(email, Fail : 400	
Body : ResqusetUser(email, Fail : 400	
Body : ResqusetUser(email, phoneNum, password)  Fail : 400  10 user-service /login POST Header : No Success : 200	
Body : ResqusetUser(email, phoneNum, password)  10 user-service /login POST Header : No Body : ResquestUser(email, password)  Fail : 400  Success : 200 header : userToken	r(All user data of VO)
Body : ResqusetUser(email, phoneNum, password)  10 user-service /login POST Header : No Body : ResquestUser(email, password)  Success : 200 header : userToken	r(All user data of VO)

П		<u> </u>	_	I	1
11	user-service	/autologin	POST	Header: userToken	Success: 200 header: userToken Body: ResponseUser(All user data of VO) Fail: 400
12	post-service	/getPosts/{page}/{postTy pe}	POST	Header : userToken	Success: 200 Body:
		http://ec2-15-164-234-254.ap-northeast-2.compute.amazonaws.com:8000/post-service/getPosts/0/total http://ec2-15-164-234-254.ap-northeast-2.compute.amazonaws.com:8000/post-service/getPosts/0/popular http://ec2-15-164-234-254.ap-northeast-2.compute.amazonaws.com:8000/post-service/getPosts/0/areaA			Body:  ( "postId": "e6fb7c64-c7dc-4daf-b25f-641485528cb9",   "profileImageIcon":   "https://firebasestorage.googleapis.com/v0/b/mood-d39e8.appspct.com/o/profileIcon\$2FJPEG_ApCSJdgmkSM7MJKwpI2OKSdbX73_20218808_170829png?alt=mediastoken=3cc90c7c-05a6-4764-9b6f-b5143b7b02e5",   "postUid": "3dad0255-33af-4685-b920-a7112540d505",   "nickname": "Lime-IT",   "locationENC": "Seoul",   "postLikeCount": 0,   "postLime": "2021-10-13T23:43:11.927757",   "postImage":   "https://firebasestorage.googleapis.com/v0/b/mood-d39e8.appspot.com/o/profileImag2FJPEG_20211010_152520png?alt=mediastoken=b30ff108-46bb-4f83-bc8c-2203df77505c",   "postContents": "0处型이 품었기 얼마나 위하여 대고, 가는 끓는 예수는 끓는다. 가장 청춘 끓는 같으며, 사라지지 천고에 가슴이 쓸쓸하라? 청춘인류의 가지에 장식하는 뜨고, 피에 능히 품으며, 그리하였는가? #hi #hello#안병 #서울"   ),   {     "postId": "94b10d1d-0083-4647-b637-02b1c6763984",     "profileImageIcon":     "https://firebasestorage.googleapis.com/v0/b/mood-d39e8.appspot.com/o/profileIcon%2FJPEG_ApCSJdgmkSM7MJKwpI2OK5dbbX73_20218808_170829png?alt=mediastoken=3cc90c7c-05a6-4764-9b6f-b5143b7b02e5",     "nickname": "11me-IT",     "locationENG": "34ad0255-33af-4685-b920-a7112540d505",     "nickname": "11me-IT",     "locationENG": "45eul",     "postLikeCount": 0,     "postCommentCount": 0,     "postTime": "2021-10-13T23:43:29.322878",     "postImage":   "https://firebasestorage.googleapis.com/v0/b/mood-d39e8.appspot.com/o/profileImag2FJPEG_201101_152520png?alt=media6token=b30ff108-46bd-4f83-bc8c-2203df77505c",     "postContents": "0생의 품었기 얼마나 위하여 대고, 가는 끓는 예수는 끓는다. 가장 청춘 끓는 같으며, 사라지지 천고에 가슴이 쓸하라? 청춘 인루의 가지에 장식하는 뜨고, 피에 등히 품으며, 그리하였는가? #hi #hello#안병 #서울"   } }
13	post-service	/clickLike/{postId}/{com mentId} http://ec2-15-164-234- 254.ap-northeast- 2.compute.amazonaws.c om:8000/post- service/clickLike/71c2b67 b-4a6b-4249-bae6- d173adf9aa7c/0	POST	Header: userToken	Fail: 400  Success: 200  Fail: 400
14	post-service	/registComment	POST	Header:userToken Body:     1) 댓글일 경우 {     "postId":"707b2ae3-fa83-483a-8e33-6c965d97c366",     "commentContents":"세번째 댓글이에요. 잘부탁해요~",     "commentGroup":"0" }  2) 대댓글일 경우 {     "postId":"707b2ae3-fa83-483a-8e33-6c965d97c366",     "commentClass":"3",     "commentClass":"3",     "commentContents":"세번째 댓글님, 반가워요~~",     "commentContents":"세번째 댓글님, 반가워요~~",     "commentGroup":"1" }  참고사진:  ***********************************	Success: 200 Fail: 400
15	user-service	/getUser	POST	Header: userToken  Body: RequestProfile(userUid)	Success: 200 Body: RequsetProfile(profileImage, profileIcon, nickname, birthdate, locationKOR, locationENG, physicalDistance, work, happy, dating, moodDistance, respect, contact, date, communication, sex) Fail: 400

	Ι			I	
16	post-service	/getPosts	POST	Header : userToken	Success: 200
		http://ec2-15-164-234- 254.ap-northeast-		Body:	Body:
		2.compute.amazonaws.c		{"postUid":"34ad0255-33af-4685-b920-a7112540d505"}	[
		om:8000/post-			{
		service/getPosts/0			"postId": "f7f3fd8c-d78e-4c22-9ce8-5330d554eb79",
					"profileImageIcon":
					"https://firebasestorage.googleapis.com/v0/b/mood-
					d39e8.appspot.com/o/profileIcon%2FJPEG_ApCSJdgmkSM7MJKwpI2OK5d
					DbX73_20210808_170829png?alt=media&token=3cc90c7c-05a6-4764-
					9b6f-b5143b7b02e5",
					"postUid": "34ad0255-33af-4685-b920-a7112540d505",
					"nickname": "Lime-IT",
					"locationENG": "Bucheon", "locationKOR": "부천",
					· ·
					<pre>"postLikeCount": 0, "postCommentCount": 0,</pre>
					"postTime": "2021-10-14T14:01:55.648087",
					"postImage":
					"https://firebasestorage.googleapis.com/v0/b/mood-
					d39e8.appspot.com/o/profileImg%2FJPEG 20211010 152520 .png?al
					t=media&token=b30ff108-46bb-4f83-bc8c-2203df77505c",
					"postContents": "인생의 품었기 얼마나 위하여 대고, 가는 끓는 예
					수는 끓는다. 가장 청춘 끓는 같으며 쓸쓸하랴? 청춘 인류의 가지에 장식하는
					뜨고, 피에 능히 품으며, 그리하였는가? #hi #hello #테스트 #부천"
					}
17	post-service	/registPost	POST	Header : userToken	Success: 200
.,	post service	71egisti ost	1 001	Body:	Fail: 400
				1) 해쉬태그가 없을 때	Tuil. 400
				ᆝᄼᅁᆔᄔᅼᄼᆝᆹᆯᄦ	
				"locationENG": "Seoul",	
				"locationKOR":"서울특별시",	
				"postImage":	
				"https://firebasestorage.googleapis.com/v0/b/mood-	
				d39e8.appspot.com/o/profileImg%2FJPEG 20211010 152520 .png?al	
				t=media&token=b30ff108-46bb-4f83-bc8c-2203df77505c",	
				"postContents": "인생의 품었기 얼마나 위하여 대고, 가는 끓는 예수는	
				끓는다. 가장 청춘 끓는 같으며, 사라지지 천고에 가슴이 쓸쓸하랴? #hi	
				#hello #안녕 #서울",	
				"requestHashtags":[	
				requesthashtags":[	
				3	
				2) 해쉬태크가 있을 때	
				{	
				"locationENG": "Seoul",	
				"locationKOR":"서울특별시",	
				"postImage":	
				"https://firebasestorage.googleapis.com/v0/b/mood-	
				d39e8.appspot.com/o/profileImg%2FJPEG20211010_152520png?al	
				t=media&token=b30ff108-46bb-4f83-bc8c-2203df77505c",	
				"postContents": "인생의 품었기 얼마나 위하여 대고, 가는 끓는 예수는	
				끓는다. 가장 청춘 끓는 같으며, 사라지지 천고에 가슴이 쓸쓸하랴? 청춘 인류	
				의 가지에 장식하는 뜨고, 피에 능히 품으며, 그리하였는가? #hi #hello #안	
				녕 #서울",	
				"requestHashtags":[	
				{"hashtagName":"hi"},	
				{"hashtagName":"hello"},	
				{"hashtagName":"안녕"},	
				{"hashtagName":"서울"}	
				1	
				}	
18	post-service	/updatePost	POST	Header : userToken	Success: 200
				Body:	Fail : 400
				{	
				"postId":"707b2ae3-fa83-483a-8e33-6c965d97c366",	
				"locationENG": "Bucheon",	
				"locationKOR":"부천",	
				"postImage":	
				"https://firebasestorage.googleapis.com/v0/b/mood-	
				d39e8.appspot.com/o/profileImg%2FJPEG20211010_152520png?al	
				t=media&token=b30ff108-46bb-4f83-bc8c-2203df77505c",	
				"postContents": "인생의 품었기 얼마나 위하여 대고, 가는 끓는 예수는	
				끓는다. 가장 청춘 끓는 같으며 쓸쓸하라? 청춘 인류의 가지에 장식하는 뜨고,	
				피에 능히 품으며, 그리하였는가? #hi #hello #테스트 #부천",	
				"requestHashtags":[	
				{"hashtagName":"hi"},	
				{"hashtagName":"hello"},	
				{"hashtagName":"테스트"},	
				{"hashtagName":"부천"}	
				}	
		<u> </u>			
19	post-service	/deletePost	POST	Header : userToken	Success: 200
				Body:	Fail : 200
				{"postId":"f4eec79d-bd70-44ef-abf7-1289fed85d6a"}	

	T			1	
20	matching-service	/getMatchingUsers	POST	Header: userToken	Success: 200 Body: List(ResponseMatchingUser(userUid, profileImage, profileIcon, nickname, birthdate, locationKOR, locationENG, physicalDistance, respect, contact, date, communication, sex MatchingData(moodDistance, matchingTime))) Fail: 400
21	matching-service	/updateMatchingTime	POST	Header: userToken String("matching or profile or request or reset")	Success: 200 Body: List(ResponseMatchingUser(userUid, profileImage, profileIcon, nickname, birthdate, locationKOR, locationENG, physicalDistance, respect, contact, date, communication, sex MatchingData(moodDistance, matchingTime))) Fail: 400
22	user-service	/purchaseVIP	POST	Header: userToken RequestPurchase(orderId, packageName, productId, purchaseTime, purchaseState, purchaseToken, acknowledged)	Success: 200 Body: ResponseUser(All user data of VO) Fail: 400
23	user-service	/setMatching	POST	Header : userToken RequsetUser(All user data of VO)	Success : 200 Fail : 400
24	user-serivce	/blockPhoneNums	POST	Header: userToken  Body:  [     "01012345678",     "01001932134",     "01012423566"	Success : 200 Fail : 400
25	user-service	/updatePhoneNum	POST	Header : userToken  Body : RequestUser(phoneNum)	Success : 200 Fail : 400
26	user-service	/openCheating	POST	Header : userToken Body : Integer(coin), Integer(ticket)	Success : 200 Fail : 400
27	post-service	/getHashtag/{hashtagNa me/{page}} http://ec2-15-164-234- 254.ap-northeast- 2.compute.amazonaws.c om:8000/post- service/getHashtag/hi/0	POST	Header: userToken Body: No	Success: 200 Body:  [
28	lock-service	/addLockUser	POST	Header:userToken Body : RequestLockUser(lockUserUid, lockType, lockReasons, referUid)	Success : 200 Fail : 400
29	post-service	/getPost/(postId) http://ec2-15-164-234- 254.ap-northeast- 2.compute.amazonaws.c om:8000/post- service/getPost/f7f3fd8c- d78e-4c22-9ce8- 5330d554eb79	GET	Body: No	Success: 200 Body:  {     "postId": "f7f3fd8c-d78e-4c22-9ce8-5330d554eb79",     "profileImageIcon": "https://firebasestorage.googleapis.com/v0/b/mood-d39e8.appspot.com/o/profileIcon%2FJPEG_ApCSJdgmkSM7MJKwpI2OK5dDbX73_20210808_170829png?alt=media&token=3cc90c7c-05a6-4764-9b6f-b5143b7b02e5",     "postUid": "34ad0255-33af-4685-b920-a7112540d505",     "nickname": "Lime-IT",     "locationENG": "Bucheon",     "locationKOR": "부천",     "postLikeCount": 0,     "postCommentCount": 1,     "postTime": "2021-10-14T14:01:55.648087",     "postImage": "https://firebasestorage.googleapis.com/v0/b/mood-d39e8.appspot.com/o/profileImg%2FJPEG20211010_152520png?alt=media&token=b30ff108-46bb-4f83-bb8c-2203df77505c",     "postContents": "인생의 품었기 얼마나 위하여 대고, 가는 끓는 예수는 끓는다. 가장 청춘 끓는 같으며 쓸쓸하라? 청춘 인류의 가지에 장식하는 뜨고, 피에 능히 품으며, 그리하였는가? #hi #hello #테스트 #부천",     "responseCommentInfoList": [

```
"nickname": "Lime-IT",
                                                                                                                             "profileImageIcon":
                                                                                                                   "https://firebasestorage.googleapis.com/v0/b/mood-
                                                                                                                   d39e8.appspot.com/o/profileIcon%2FJPEG_ApCSJdgmkSM7MJKwpI2OK5d
                                                                                                                   DbX73_20210808_170829_.png?alt=media&token=3cc90c7c-05a6-4764-
                                                                                                                   9b6f-b5143b7b02e5",
                                                                                                                             "commentContents": "반가워요~~ 첫 댓글이에요.",
                                                                                                                             "commentClass": 1,
                                                                                                                             "commentGroup": 0,
                                                                                                                             "commentTime": "2021-10-14T14:12:26.639033",
                                                                                                                             "commentLikeCount": 0
                                                                                                                         },
                                                                                                                             "commentId": "aefc9070-c724-4d0c-bf1f-7026a6c4a940",
                                                                                                                             "commentUid": "34ad0255-33af-4685-b920-
                                                                                                                   a7112540d505",
                                                                                                                             "nickname": "Lime-IT",
                                                                                                                             "profileImageIcon":
                                                                                                                   "https://firebasestorage.googleapis.com/v0/b/mood-
                                                                                                                   d39e8.appspot.com/o/profileIcon%2FJPEG_ApCSJdgmkSM7MJKwpI2OK5d
                                                                                                                   DbX73_20210808_170829_.png?alt=media&token=3cc90c7c-05a6-4764-
                                                                                                                   9b6f-b5143b7b02e5",
                                                                                                                             "commentContents": "저는 부천에 살고 있어요!",
                                                                                                                             "commentClass": 1,
                                                                                                                             "commentGroup": 1,
                                                                                                                             "commentTime": "2021-10-14T14:13:40.097318",
                                                                                                                             "commentLikeCount": 0
                                                                                                                      ]
                    /checkNickname
                                           POST
                                                   Header : No
30
       user-service
                                                                                                                   Success: 200
                                                                                                                   Fail: 400
                                                   Body:
                                                    {"nickname":"Lime"}
                                           POST
31
       user-service
                    /getBlockPhoneNums
                                                   Header : userToken
                                                                                                                   Success: 200
                                                   Body : No
                                                                                                                   Body : List < String > phoneNum
                                                                                                                      "01012345678",
                                                                                                                      "01001932134",
                                                                                                                      "01012423566"
                                           POST
32
                    /deleteComment
                                                   Header: userToken
                                                                                                                   Success: 200
       post-service
                                                       "postId": "71c2b67b-4a6b-4249-bae6-d173adf9aa7c",
                                                                                                                   "commentContents": "삭제된 내용입니다.",
                                                       "commentId": "eb79098a-1e82-459e-a11e-74cb15b42d79"
                                                                                                                   해당 내역은 위와 같이 출력됨.
```