

REST API



Http Status Code

- 200(성공)
- 201(생성)
- 202(수용)
- 400(잘못된 요청)
- 403(금지된 요청 e.g. 로그인 안 함)
- 404(페이지 없음)
- 500(서버 에러)



Custom Status Code

- 200(성공)
- 201(생성)
- 202(수용)
- 400(잘못된 요청)
- 403(금지된 요청 e.g. 로그인 안 함)
- 404(페이지 없음)
- 500(서버 에러)



- no
- id
- password
- nickname
- email
- report_count
- deal_count
- grade
- authority
- is_locked
- is_deleted

순서	URL	Page	Method	Table	Service	Request	Response	Response JSON
1	api/user	/admin	GET	User	findUserList(회원 목록 가져오기)	@PageBean(페이 지, 검색 키워드, 검 색어)	List <user> 200: 회원 리스트 조회 성공 400: 페이지 가 명시되지 않은 경우, 검색 키워드 가 잘못된 경우 403: 관리자 아님 500: 서버 연결 오류</user>	[{ "no": Integer, "id": String, "nickname": String, "email": String, "reportCount": Integer, "dealCount": Integer, "grade": float }]
2	/api/user/{userId}	/mypage,	GET	User	findUser(회원 정보 가져 오기)	@userId header : accessToken	User 200 : 회원 정보 조회 성공 403 : 로그인 안함 or 만료 404: 대상 없음 500 : 서버 연결 오류	admin이 아니면 진한 글씨만 조회 가능함 { "no": Integer, "id": String, "nickname": String, "name": String, "phone": String, "gender": boolean "email": String, "reportCount": Integer, "grade": float "dealCount" : number, "isLocked": boolean }
3	/api/user	/signup	POST	User	addUser(회원가입)	@UserRequestDto	201 : 회원 생성 완료 400 : 회원 가입에 필요한 정 보 누락 or 아이디 중복 500 : 서버 연결 오류	
4	/api/user/{id}	/mypage/modify	PUT	User	modifyUser(회원 정보 수정)	@id	200 : 회원 정보 수정 완료 403 : 로그인 만료 or 안 함, 다른 사용자가 회원 정보를 수정 하려는 경우 404: 수정하려는 대상 없음 500 : 서버 연결 오류	

순서	URL	Page	Method	Table	Service	Request	Response	Response JSON
5	/api/user	90	DELETE	user	removeUser(회원 탈퇴)	Х	200 : 본인 회원	
6	/api/user/{id}		DELETE	User	removeUser(타 회원 탈 퇴 시키기)	@id	탈퇴 성공 200 : 회원 탈퇴 성공 403 : 로그인 만료 or 안함, 다 른 사용자가 회원 탈퇴하려는 경우 404 : 존재하지 않 은 회원에 대한 탈 퇴 요청 500 : 서 버 연결 오류	
7	/api/user/login	/login	POST	User	login(로그인)	@id, @password	201 : 로그인 성공 으로 인한 토큰 생 성 완료 400 : 로 그인 실패 500 : 서버 연결 오류	{ accessToken: String refreshToken: String }
8	/api/user/logout	х	GET	User	logout(로그아웃)	header: accessToken	200 : 로그아웃 성 공 403 : 로그인하 지 않은 회원에 대 한 로그아웃 요청 500 : 서버 연결 오류	
9	/api/user/email		POST	User	authEmail(이메일 인증)	{ 'email': String }	200 : 이메일 전송 성공 400: 이메일 전송 실패	
10	/api/user/authCheck		POST	User	authCheck(이메일 인증 코드 체크)	{ 'email': String, 'authkey': String }	{201, boolean}	
11	/api/user/idCheck/{id}		GET	User	icheck(아이디 중복 검 사)	@PathVariable String id	{200, boolean}	
12	/api/user/password		POST	User	modifyPassword(비밀 번호 변경)		200 : 비밀번호 변 경 성공	
13	/api/user/item/like/{itemNo}		POST	User, Item, UserItemLike	addUserItemLike(아이 템 찜 등록)	@PathVariable String itemNo		
14	/api/user/item/like	/mypage/like	GET	User, Item,UserItemLike	findUserItemLikeList(아 이템 찜 목록 전체 조회)			
15	/api/user/item/like/{itemNo}		DELETE	User, Item,UserItemLike	removeLikeItem(아이템 찜 취소)	@PathVariable String itemNo		
16	/api/user/deal/sell	/mypage/sell	GET	User, Item	findSellDealHistory (로 그인한 유저의 판매 내역 조회)	header: accessToken	예외 처리는 현재 수정 진행중	{ status: int(물건 상황) price: int (물건 가격) title: String (물건 이름) thumbnaillink: String (썸네일 사진 링 골) name: String (판매자 이름) } { no : int title : string, content : string, updatedAt : timestamp, price : number, status : number, image : string, likeCount : number, categoryList : List <string>, view : number }</string>
17	/api/user/deal/buy	/mypage/buy	GET	User, Item	findBuyDealHistory (로 그인한 유저의 구매 내역 조회)	header: accessToken		{ status: int(물건 상황) price: int (물건 가격) title: String (물건 이름) thumbnaillink: String (썸네일 사진 링크) name: String (판매자 이름) } { no: int title: string, content: string, updatedAt: timestamp, price: number, status: number, image: string, likeCount: number, categoryList: List <string>, view: number }</string>
18	/api/user/nicknameCheck/{nickname}		GET	User	nicknameCheck	@PathVariable String nickname		
19	/api/user/deal/sell/{username}		GET		FindItemListResponse	@PathVariable Stirng nickname		{ no : int title : string, content : string, updatedAt : timestamp, price : number, status : number, image : string, likeCount : number, categoryList : List <string>, view : number }</string>
20	/api/user/deal/buy/{username}		GET		FindItemListResponse	@PathVariable Stirng nickname		{ no : int title : string, content : string, updatedAt : timestamp, price : number, status : number, image : string, likeCount : number, categoryList : List <string>, view : number }</string>



Item

- no
- title
- created_at
- updated_at
- content
- price
- status
- image
- buyer(최종적으로 살 사람)
- seller

순서	URL	Page	Method	Table	Service	Request	Response	Response JSON
1	/api/item? category="category"&keyword="keyword"&page="page"	/item	GET	item	findItemList(아이템 목록 가져 오기)	category, keyword : string page : integer	List <item> 성공 (200), 서버 에러 (500)</item>	{ content: [{ "no": Integer "title": String, "update_at": TimeStamp, "content": String, "price": Integer, "status: Integer, "image": String, "likeCount": Integer, "categoryList": [String,] }], "totalPages": integer }
2	/api/item/{itemNo}	/item/:no	GET	item, User	findItem(아이템 상세 페이지)	@itemNo	Item 성공(200), 대상 없음(404), 서버 에러(500)	{ "no": Integer "title": String, "created_at": TimeStamp, "updated_at": TimeStamp, "content": String, "price": Integer, "status": Integer, "imageList": [String,], "likeCount": Integer, "categoryList": [String,], "seller": { "nickname": String, "grade": float, }, "dateList": [{ "date": TimeStamp, "status":integer }] }
3	/api/item	/item/register	POST	item	additem(판매할 아이템 등록)	{ "title" : String, "content" : String, "price" : Integer, "imageList" : [String,], "dateList": [TimeStamp,], "categoryList" : [Integer,] }	생성(201), 잘못된 요청(400, 필수값 이 채워지지 않거 나. 이상한 값 입 력) 로그인 하지 않음(403), 서버 에러(500)	
4	/api/item/{itemNo}	/item/:no/modify	PUT	item	modifyItem(아이템 상세 페이 지 수정)	@itemNo { "title" : String, "content" : String, "price" : Integer, "imageList" : [String,], "dateList": [TimeStamp,], "categoryList" : [Integer,] }	업데이트 성공 (200), 잘못된 요 청(400), 로그인 하지 않음, 또는 자기 게시물이 아 남(403), 대상 없 음(404), 서버 에 러(500)	
5	/api/item/{itemNo}	x	DELETE	item	removeltem(등록한 아이템 삭 제)	@itemNo	삭제 성공(200), 거래중인상품 삭 제불가능(400), 권 한 없음(403, 로그 인 하지 않거나, 자기 게시글이 아 님), 대상 없음 (404), 서버 에러 (500)	
6	/api /item/{itemNo}/deal		PUT	item	modifyStatusDeal(아이템 상 태 거래전에서 중으로 변경 0 → 1 / 거래중)	@itemNo { buyer: string // nickname }	상태변경성공 (200), 이미 거래 중이거나 완료된 상품(400), 자기 상품이 아님(403), 대상없음(404), 서 버 에러(500) 상태변경성공	
7	/api /item/{itemNo}/confirmation		PUT	item	modifyStatusConfirmation(아 이템 상태 거래중에서 완료로 변경 $1 \rightarrow 2$ I 거래확정)	@itemNo	(200), 거래중인 상품아님(400), 자 기 상품이 아님 (403), 대상없음 (404), 서버 에러 (500)	
8	/api /item/{itemNo}/cancel		PUT	item	modifyStatusCancel(아이템 상태 거래중에서 전으로 변경 1→0 / 거래취소)	@itemNo	상태변경성공 (200), 거래중인 상품이 아님(400), 자기 상품이 아님 (403), 대상없음 (404), 서버 에러 (500)	



- no
- category

순서	URL	Page	Method	Table	Service	Request	Response	Response JSON
1	api/category	/item, /item/write	Get	category	findCategories → 전체 카테고리를 가져 오는 service	-	category(전체) 카테고리 전체 조회 성공 (200) 서버 에러(500)	[{ no : integer, category : String } 전체 카테고리!]

Video

- no
- item_no
- link
- start_time
- end_time
- delete_time
- (만약 유예기간을 주면 관계 릴레이션을 새로 만들어야 한다.)

순서	URL	Page	Method	Table	Service	Request	Response	Response JSON
1	/api/video/{no}	/mypage/buy/video/:no(item) /mypage/sell/video/:no(item)	GET	video, item	findVideo (특정 판매 아이템의 비디오 를 조회)	@itemNo(상품번 호)	Video, 성공(200) 내가 거래하지 않 은/로그인 안 한 상 태로 비디오 조회 시도 (403) 비디오 가 존재하지 않는 아이템 (404) 서버 에러 (500)	{ "no" : integer, "itemTitle": String, "seller" : String, "link" : String, "startTime" : TimeStamp, "endTime" : TimeStamp }
2	/api/video/{no}	х	DELETE	video, item	removeVideo (특정 아이템의 비디오 를 제거)	@no	비디오 제거 성공 (200) 거래 당사자 가 아닌 사람이 비 디오 제거 시도 (403) 아이템에 비 디오가 존재하지 않음 (404) 서버 에러 (500)	
3	/api/video/{no}	х	POST	video, item	addVideo (특정 아이템의 비디오를 추가)	@VideoDto	비디오 추가 성공 (201) 확장자 다른 파일 첨부(400) 거 래 당사자가 아닌 사람이 비디오 추 가 시도 (403) 아 이템이 없는데 추 가 시도(404) 서버 에러 (500)	
4	/api/video/extension/{no}		PUT	video, item	extendVideoDeadLine	@no(비디오 PK)		연장된 비디오 날짜

Message

- no
- room_no
- sender
- content • created_at

순서	URL	Page	Method	Table	Service	Request	Request_json	Response	Response_json
1	api/room/{roomNo}/message	/chat	GET	Message	findMessage(@room_no를 전 송 받아서 대화 정 보를 불러올 때, 사용할 서비스) 해 당 방이 없을 경우 생성해서 새로운 방 보여주기	@room_no		List <message> 메세지 가져오기 성공(200) 내 메세 지가 아닌 것을 가 져오려고 함(403) 서버 에러 (500)</message>	[{"sender": String, "content": String "isRead": Boolean "createdAt": TimeStamp }, { "sender": String, "content": String "isRead": Boolean "createdAt": TimeStamp },]
2	api/room/{roomNo}/message	채팅하는 컴포넌 트	POST	Message	addMessage(메 세지를 전송할 때 사용할 서비스)	@Message JSON header: { accessToken }	{ header: { accessToken }, body: {"content": string, } }	채팅 전송 성공 (200) 메세지에 보 낼 수 없는 것이 들어 있음(400) 로 그인 하지 않음 (403) 서버 에러 (500)	

Conference(화상) (수정 완료)

- no
- is_active
- item_no
- confirmed_date

순서	URL	Page	Method	Table	Service	Request	Request_json	Response	Response_json
1	/api/conference/{no}	/conference	GET	Conference (User와 Item과 조인해서 정보 가 져옴)	findConference(@no)	@no		(만들어진 화상 채팅 의 no, 200(화상 조 회 성공)) 403(내 화 상이 아닌 다른 것을 가지고 오려함) 404(대상 없음) 500(서버 에러)	{ "no": int, "isActive": boolean, "itemNo": int, "comfirmedDate": TimeStamp }
2	x	x	POST	Conference	addConference(@is_active, @item_no)	Conference	{ "isActive": booelan (false), "itemNo": int, "comfirmedDate": TimeStamp }	status 201(화상 생 성 성공) 400(잘못된 요청) 500(서버 에 러)	클라이언트에서 부르는 메소드 가 아닌데 response가 필요한 가? 빼는거 고려
3	api/conference/{no}	х	DELETE	Conference	removeConference(@no)	@no		status 200(삭제 성 공) 403(내 화상이 아닌 것을 삭제 하려 함) 404(대상 없음) 500(서버 에러)	마찬가지

Date(스케줄 날짜) (판매자 전용 정보)

- no
- item_no
- date
- status (0 , 1, 2)

순서	URL	Page	Method	Table	Service	Request	Response	Response JSON
1	/api/date/{item_no}	Calendar 컴포넌 트	GET	Item, Date	findAllItemDateList (판 매 물건의 예약 날짜 목 록 조회)	@item_no	List <date>, 성공함 (200) item_no가 존재하지 않음 (404) 서버 에러 (500)</date>	[{ "no": Integer, "itemNo": Integer, "date": TimeStamp }, { "no": Integer, "itemNo": Integer, "date": TimeStamp }, { "no": int, "itemNo": int, "date": TimeStamp }]
2	api/date/{item_no}	х	POST	Item, Date, user	additemDate(판매 물건 의 예약 날짜 추가)	@DateDto	잘 추가됨 (201) 아이템 미존재 or 유저 미존재 (400) 비로그인 상태에서 요청 or 다른 유저의 user_no로 요청 (403) 서버 에러 (500)	х
3	api/date/{item_no}/{date}	х	DELETE	Item, Date, user	removeItemDate (판매 물건 일정에서 판매자의 특정 날짜 제거)	@item_no, @user_no, @date	잘 제거됨 (200) 다른 유저의 날짜를 제 거하려고 시도함 (403) 해당 유저의 특 정 날짜의 예약 내역이 없어서 지울수 없 음 (404) 서버 에러 (500)	х
4	api/date/{item_no}	х	DELETE	Item, Date, user	removeItemDateList(판 매 물건에서 판매자의 모 든 예약된 날짜 제거)	@item_no, @user_no	잘 제거됨 (200) 다른 유저의 날짜를 제 거하려고 시도함 (403) 유저의 해당 아 이템에 대한 예약 내역이 존재하지 않음 (404) 서버 에러 (500)	x



Reservation(구매자 화상통화 예약) (JSON 수정됨 07/30)

- no
- user_no
- conference_no
- is_seller
- is_attended

순서	URL	Page	Method	Table	Service	Request	Response	Response JSON (수정 전 백업)	Response JSON (수 정 후)
1	/api/reservation/{user_no} → (변경) /api/reservation is	/mypage/schedule	GET	Reservation	findReservation() 사용자별 예약 리스트	@userToken	List <reservation> 200(예약 조회 성공) 403(내 예약이 아닌 다른 것을 가지고 오 려함) 404(대상 없 음) 500(서버 에러)</reservation>	[{ "no": Integer, "userNo": Integer, "conferenceNo": Integer, "isSeller": Boolean, "isAttended": Boolean }, { "no": Integer, "isSeller": Boolean, "isAttended": Boolean, "isAttended": Boolean, "isAttended": Boolean }, { "no": Integer, "userNo": Integer, "userNo": Integer, "isAttended": Boolean, "isSeller": Boolean, "isSeller": Boolean, "isSeller": Boolean, "isAttended": Boolean }]	[{ "title": String; "confirmedDate": TimeStamp; "conferenceNo": int }, { "title": String; "confirmedDate": TimeStamp; "conferenceNo": int }]

순서	URL	Page	Method	Table	Service	Request	Response	Response JSON (수정 전 백업)	Response JSON (수 정 후)
2	/api/reservation/check	구매자가 날짜를 선 택할 때!(예약 확정 할 때)	POST	Reservation, Conference, User, Item, Date	checkBuyerReservation(구매 자가 선택한 예약 날짜 검사)	@userToken { "itemNo" : Integer, "selectDate" : Date }	{ "status" : Integer, "code" : UUID } status - 0: 예약 불가능(구매자가 다른구매 예약해놨을 때) - 1: 예약 가능하지만구매자가 등록한 판매 상품의 날짜 목록이랑 겹침 - 2: 겹치는목록이 없어서 바로 예약 생성 완료 200(성공) 400(잘못된 요청) 500(서버에러)		
3	/api/reservation/confirm	confirm()예약 확정 버튼 누를때네 "확 인">내 판매 예약가 능날짜 지우기 "취 소"> 예약 ㄴㄴ > 다 시 스케줄 버튼 보 여주기	POST	Reservation, Conference, User, Item, Date	addConfirmedReservation(구 매자가 재확인한 예약 생성)	@userToken { "itemNo" : Integer, "selectDate" : Date, "code" : UUID }	201(생성 성공) 400(잘못된 요청) 500(서버 에러)		
4	/api/reservation/date		GET	Reservation, Conference	findReservationDateList() 사 용자가 예약해놓은 날짜들	@userToken	200 (조회 성공)		[{ date:TimeStamp



no

• item_no

A 11									
순서	URL	Page	Method	Table	Service	Request	Request Json	Response	Response JSON
1	/api/room	컴포넌트. 메세지 를 포함하는	Post	Room	addRoom(-@item_no-판 매자 아이디는 item_no에 서 조회하고, 구매자 아이 디는 token보고 판별함)	@userToken, @item_no	{ "itemNo": int }	방 생성 성공 (201), item_no가 잘못됨(400), 로그 인 하지 않은 사용 자(403), 서버 에 러(500)	{ roomNo:3 }
2	/api/room/{item_no} /{user}	X -프론트에는 해 당 페이지/컴포넌 트가 없습니다~	Get	Room	findRoom(@item_no @user: buyer 로그인한 유저의 특정 아이템에 관 한 모든 방을 불러온다 아 이템 번호, 구매자 번호에 따라서 방이 별도로 생성 되므로 방을 가져올 때, 둘 을 임력 받아야 한다.)	@item_no, @user_no		Room 올바른 정 보 수신(200), item_no나 user_no가 잘못 됨(400), 로그인 하지 않은 사용자 (403), 서버 에러 (500)	{ "no": int, "itemNo": int }
3	/api/room/{roon_no}	х	Delete	Room	removeRoom(@room_no 현재 로그인 한 유저가 방에서 나감)	@room_no		방 나가기 성공 (200), 로그인 하 지 않음(403), 대 상 없음(404), 서 버 에러(500)	x
4	/api/room title → 이미지소 스 상대방의 이름 상대방 등 급까지 추가	/mypage/chat	Get	Room	findRoom() 유저 토큰보 고 현재 로그인한 유저의 모든 방을 가져올 수 있음 채팅방 없을 시 그냥 빈 칸 보여주기	header: userToken		List <room> 정 보 조회 성공 (200), 로그인 하 지 않음(403), 서 버 에러(500)</room>	[{"opponentNickname": String, "itemImage": String, "lastMessage": String, "opponentGrade": Float, "unreadCount": int, "lastMessageCreatedAt": TimeStamp }, { "opponentNickname": String, "itemImage": String "lastMessage": String "opponentGrade": Float, "unreadCount": int "lastMessageCreatedAt": TimeStamp }]
5	/api/room/item/\${chatNo}		Get	Room	getRoomItem	@roomNo			{ "no" : integer, "title" : String, "price" : integer, "image" : String, "seller" : { "nickname" : String, "grade" : float }, "dateList" : [{ "date" : TimeStamp, "status" : integer },] }

Image

- no
- item_no
- link

순서	URL	Page	Method	Table	Service	Request	Response	Response JSON
1	/api/image/{image_no}	/item/{item_no}	GET	Image	findImage(@image_no)	@userToken, @item_no	이미지 가져오기 성공(200), 이미지 없음(404), 서버 에러(500)	{ "no": int, "itemNo": int "link": String }

순서	URL	Page	Method	Table	Service	Request	Response	Response JSON
2	х	/item/write	POST	Image	addImage(@link)	@link	이미지 저장 성공(201), 이미지 파일 아님(400), 로그인 하지 않음 or 업로드 가능한 사람 아님(403), 서버 에러(500)	x
3	х		PUT	Image	modify(@image_no)	@image_no	이미지 수정 성공(200), 이미지 파일 아님(400), 로그인 하지 않음 or 수정 가능권한 없음(403), 대상 없음(400), 서버 에러(500)	x
4	х		DELETE	Image	removeImage(@image_no)	@image_no	이미지 삭제 성공(200), 삭제 권한 없음(403), 대상 없음(404), 서 버 에러(500)	x

report (수정 완료)

- no
- reporter
- reported
- reason
- report_date
- item_no
- room_no

순서	URL	Page	Method	Table	Service	Request JSON	Response
1	/api/report		POST	Report	addReport(@userToken, @AddReportRequest)	{ reported: String (신고 당하는 사람) reason: String itemNo: int roomNo: int }	{201, 신고번호}(신고 완료) 400 (24시간 이내에 신고해서 추가 신고 불가능) 404 (대상이 없음)