

API 이름	controller	권한(header)	구현 여부	http method	endpoint	request body	response body	status code message	ApiOperation (swagger 설명)	담당자	비고
예시	TestController	로그인한 사용자		POST	/test/ok /test/ok<data> /test/fail /test/error		{ "status": "code" "message": "성공" "data": { "아이디" } }	200: 테스트 성공 400: 테스트 실패	api 테스트	김가영	특이사항 데이터 상세 설명 자유롭게 작성
사용자 로그인	UserController	x	O	POST	api/user/login	{ "userId": "String, 아이디", "password": "String, 비밀번호" }	{ "status": 200, "message": "로그인 성공", "data": { "refreshToken": "string", "accessToken": "string", "tokenType": "Bearer", "nickname": "string, 닉네임" } }	200 : 로그인 성공 401 : 실패 400 : 신고 및 수 누적으로 계정이 정지되었습니다.	로그인	김가영	
사용자 회원가입	UserController	x	O	POST	api/user/signup	{ "userId": "String, 아이디", "password": "String, 비밀번호", "email": "String, 이메일", "name": "String, 이름", "nickname": "String, 닉네임", "birthYear": int, 출생연도 }	{ status: 200 message: data: { } }	200: 회원가입 완료 401: 실패	회원가입	김가영	
아이디 중복 체크	UserController	x	O	POST	api/user/userId	{ "userId": "String, 아이디" }	{ status: 200, message: , data: { } }	200: OK 409: 이미 등록된 아이디입니다	아이디 중복 체크	김가영	
이메일 중복 체크	UserController	x	O	POST	api/user/email	{ "email": "String, 이메일" }	{ status: , message: , data: { } }	200: 이메일로 인증 코드 전송 성공 409: 이미 등록된 이메일입니다 500: 인증 코드 전송 실패 (인증 코드 메일이 전송되지 않았을 때)	이메일 중복 체크	김가영	
이메일 인증 코드 체크	UserController	x	O	POST	api/user/verify>email	{ "email": "String, 이메일", "code": "String, 인증코드" }	{ status: , message: , data: { } }	200: 이메일 인증 완료 400: 인증코드를 확인해주세요	이메일 인증코드 체크	김가영	
닉네임 중복 체크	UserController	x	O	POST	api/user/nickname	{ "nickname": "String, 이메일" }	{ status: , message: , data: { } }	200: OK 409: 이미 등록된 닉네임입니다	닉네임 중복 체크	김가영	
사용자 정보 조회	UserController	Authorization: Bearer {access token}	O	GET	api/user/profile	없음	{ status: 200 message:"회원 정보 조회 성공" data: { userId : String, email: String, profileImg:"s3에 저장된 이미지 경로" name: String, nickname: String, year: int, } }	200: "회원 정보 조회 성공"	회원정보 조회	김가영	
access token 재발급	AuthController	Authorization: Bearer {refresh token}	O	GET	api/auth/reissue	없음	{ status: 200 message:"access token 재발급 성공" data: { "refreshToken": "string 기존 refresh token", "accessToken": "string 새로운 access token", "tokenType": "Bearer" } }	401: "JWT token expired" refresh token이 만료됨을 의미함 200: "access token 재발급 완료"	access token 재발급	김가영	access token이 만료되면 JWT token expired 메시지를 클라이언트로 전송 access token이 만료되었다는 메시지를 받은 클라이언트는 해당 refresh token을 담아서 access token 재발급 요청을 보낸다. refresh token이 만료되지 않았다면 서버는 기존 refresh token, 새로운 access token을 클라이언트로 전송한다. refresh token이 만료되었다면 서버는 JWT token expired 메시지를 클라이언트로 전송한다. 두 도메인 모두 만료되면 사용자는 다시 로그인을 해야 한다.
구글 회원 가입, 로그인	AuthController	x	O	POST	api/auth/google	{ "token": "구글에서 발급한 credentialResponse.credential" }	{ "status": 200, "message": "로그인 성공", "data": { "refreshToken": "string", "accessToken": "string", "tokenType": "Bearer", "nickname": "로그인한 사용자 닉네임" } }	200 : 로그인 성공 404 : 회원가입이 필요합니다	구글 소셜 로그인	김가영	

API 이름	controller	권한(header)	구현 여부	http method	endpoint	request body	response body	status code message	ApiOperation (swagger 설명)	담당자	비고
닉네임 수정	UserController	Authorization: Bearer (access token)	O	PATCH	api/user/nickname	{ "newValue": "String, 새 닉네임" }	{ status: 200 message: "닉네임 수정 완료" data: { "nickname": "새 닉네임" } }	200 : 성공 409: 이미 등록된 닉네임입니다	회원 정보 수정 회원 정보(닉네임) 수정	김가영	
비밀번호 수정	UserController	Authorization: access token	O	PATCH	api/user/pw	{ "oldValue": "String, 기존 비밀번호", "newValue": "String, 새 비밀번호" }	{ status: 200 message: "비밀번호 수정 완료" data: { } }	200: "비밀번호 수정 완료"	회원 정보 수정 회원 정보(비밀번호) 수정	김가영	
프로필 사진 등록	UserController	"Authorization": Bearer (access token) "Content-Type": "multipart/form-data"	O	POST	api/user/img	없음	{ "status": 200, "message": "프로필 사진 수정 완료", "data": { "profileImg": "s3에 저장된 이미지 경로" } }	200: "프로필 사진 수정 완료"	회원 프로필 사진 수정	김가영	
아이디 찾기	UserController	x	O	POST	api/user/find-id	{ "email": "String, 이메일", "name": "String, 이름" }	{ status: 200 message: "아이디 조회 성공" data: { userId: "사용자 아이디" } }	404: "유자가 존재하지 않습니다"	아이디 찾기	김가영	
비밀번호 찾기	UserController	x	O	POST	api/user/find-pw	{ "userId": "String, 아이디", "email": "String, 이메일" }	{ status: 200 message: "임시 비밀번호 발급 성공" data: { } }	404: "유자가 존재하지 않습니다" 500: "메일을 전송하지 못했습니다"	비밀번호 찾기	김가영	
영상 목록 조회	UserController	Authorization: Bearer (access token)	O	GET	api/user/videos	없음	{ status: 200 message: "사용자 영상 목록 조회 성공" data: [{"videoId": int, 영상번호, "title": string, 스파치 제목, "recordTime": string, yymmdd, 영상 녹화 날짜, "playTime": int, 재생시간(분)], } }	200: 사용자 영상 목록 조회 성공 404: 유저가 존재하지 않습니다.	영상 목록 조회	김가영	
스터디 룸 목록 조회	MeetingController	Authorization: Bearer (access token)	O	POST	api/meeting/list	{ int categoryId; String word; }	{ status: 200 message: "스터디룸 목록 조회 성공" data: { { String sessionId; String managerId; int categoryId; String title; int currentPeople; int maxPeople; Boolean isPrivate; String password; }, } } }	200 : "방 정보 조회 성공" 404 : "검색 조건 오류"	검색 조건에 맞는 방 목록 조회 검색 조건: 방 제목	김영대	
스터디 룸 상세 조회	MeetingController	Authorization: Bearer (access token)	O	POST	api/meeting/detail	{ String sessionId; }	{ status: 200 message: "조회 성공" data: { String sessionId; String managerId; int categoryId; String title; int currentPeople; int maxPeople; Boolean isPrivate; String password; } }	200 : 조회 성공 401 : 해당 방을 조회할 수 없습니다.	roomCode 방의 상세 정보를 조회한다.	김영대	
스터디 룸 생성	MeetingController	Authorization: Bearer (access token)	O	POST	api/meeting/create	{ String managerId; int categoryId; String title; int maxPeople; Boolean isPrivate; String password; }	{ status: 200 message: "스터디룸 생성 성공" data: { String sessionId; String managerId; String categoryId; String title; int currentPeople; int maxPeople; Boolean isPrivate; String password; }, String token; } }	200 : "방을 성공적으로 생성하였습니다" 404 : "잘못된 정보 입력"	방을 생성한다.	김영대	

API 이름	controller	권한(header)	구현 여부	http method	endpoint	request body	response body	status code message	ApiOperation (swagger 설명)	담당자	비고
스터디 룸 종료 / 나가기	MeetingController	Authorization: Bearer (access token)	O	POST	api/meeting/leave	{ String userId; String sessionId; String token; }	{ status: 200 message: "성공" }	200 : "스터디 룸 종료" 403 : "스터디 룸 종료 권한이 없습니다." 404 : "이미 종료된 스터디 룸"	발정 : 방을 종료한다. 참가자 : 방을 나간다.	김영대	
스터디 룸 참가	MeetingController	Authorization: Bearer (access token)	O	POST	api/meeting/join	{ String sessionId; String userId; String password; }	{ status: 200 message: "스터디룸 참가" data: { String sessionId; String managerId; int categoryId; String title; int currentPeople; int maxPeople; Boolean isPrivate; String password; }, String token; }	200 : "입장 성공" 400 : "자리 없음" 400: "입력 내용을 다시 확인해주세요" (비밀번호 오류) 404 : "방이 존재하지 않습니다"	스터디 룸에 참가한다.	김영대	
대본 카테고리 조회	ScriptController	Authorization: Bearer (access token)	O	GET	api/script/list	없음	{ "status": 200, "message": "모든 대본 카테고리 조회 성공", "data": [{ "categoryId": 1, "categoryName": "뉴스" }, { "categoryId": 2, "categoryName": "발표" }] }	400: 조회 실패		김가영	
대본 목록 조회	ScriptController	Authorization: Bearer (access token)	O	GET	api/script/list/{categoryId} categoryId: int	없음	{ "status": 200, "message": "대본 리스트 조회 성공", "data": { "categoryId": 1, "categoryName": "뉴스", "scripts": [{ "scriptId": 1, "scriptTitle": "대본 제목1" }, { "scriptId": 2, "scriptTitle": "대본 제목2" }, { "scriptId": 3, "scriptTitle": "대본 제목3" }] } }	200 : 대본 리스트 조회 성공 404: "not found"	대본 목록 조회 선택한 대본 카테고리에 해당하는 대본 조회	김가영	
대본 상세 조회	ScriptController	Authorization: Bearer (access token)	O	GET	api/script/{scriptId}	없음	{ status: 200 message: "조회 성공" data: { "category": "string, 대본 카테고리", "title": "string, 대본 제목", "content": "string, 대본 내용", "originalVideo": "string, 원본 영상 url" } }	200 : 대본 상세 조회 성공 404 : "not found"	대본 상세 조회 scriptId번 대본 상세 정보 조회	김가영	
대본 등록	관리자용	Authorization: Bearer (access token)	O	POST	api/script/regist	{ speechId: "int, 발표 번호", categoryId: "int, 카테고리 번호", title: "String, 대본 제목", videoUrl: "String, 발표 원본 영상", content: "String, 대본 내용" }	{ status: 200 message: "등록 성공" }	200 : 등록 성공 400 : 중복된 제목	새로운 대본 등록	김가영	
대본 수정	관리자용	Authorization: Bearer (access token)	O	PUT	api/script/{scriptId}	{ categoryId: "int, 카테고리 번호", title: "String, 대본 제목", videoUrl: "String, 발표 원본 영상", content: "String, 대본 내용" }	{ status: 200 message: "수정 성공" }	200 : 수정 성공 400 : 수정 실패 (유효성 검사 등)	대본 수정	김가영	
대본 삭제	관리자용	Authorization: Bearer (access token)	O	DELETE	api/script/{scriptId}	없음	{ status: 200 message: "삭제 성공" }	200 : 삭제 성공 404 : 삭제 실패 (이미 삭제된 경우 등)	대본 삭제	김가영	
명언 조회	SentenceController	Authorization: Bearer (access token)	O	GET	api/sentence	없음	{ status: 200 message: "조회 성공" data: { "sentence": 명언 } }	200 : 조회 성공 400: 조회 실패	DB에서 명언을 랜덤하게 조회한다.	김가영	

API 이름	controller	권한(header)	구현 여부	http method	endpoint	request body	response body	status code message	ApiOperation (swagger 설명)	담당자	비고
영언 등록	SentenceController	관리자	O	POST	api/sentence	{ "sentence": "String, 영언" }	{ status: 200 message: "등록 성공" }	200 : 등록 성공 400 : 등록 실패	영언을 등록한다	김가영	
영언 삭제	SentenceController	관리자	O	DELETE	api/sentence/{sentenceId}	없음	{ status: 200 message: "삭제 성공" }	200 : 삭제 성공 400 : 삭제 실패	영언을 삭제한다.	김가영	
녹화 시작	SpeechController	Authorization: Bearer {access token}	O	POST	api/speech/start	{ "userId": String, 유저 아이디, "sessionId": "String, 세션 아이디" "title": String, 발표 제목, "personal": boolean, true(혼자연습), false(스터디) "categoryId": int, 카테고리 id, "scriptId": int, 대본 id, 스터디 모드리면 0 }	{ status: 200 message: "성공" data: { "speechId": int, 스피치 PK } }	200 : 성공(녹화 시작) 401 : 실패(세션을 찾을 수 없음) 404 : 현재 녹화가 진행중임	녹화 시작 버튼	김영대, 김가영	
녹화 종료	SpeechController	Authorization: Bearer {access token}	O	POST	api/speech/end	{ "speechId": int, 스피치 PK "decibels": int[], 데시벨 배열 }	{ status: 200 message: "성공" data: { } }	200 : 성공 401 : 실패(발표 정보를 찾을 수 없음)	녹화 종료 데시벨 평가	김영대, 김가영	
10초 등록	SpeechController	Authorization: Bearer {access token} "Content-Type": "multipart/form-data"	O	POST	api/speech/assess?speechId=(int) &isLast=(true 또는 false)	없음	{ status: 200 message: "성공" data: { score : float, 평가 점수 } }	200: 성공 404: 실패(평가 실패, 따로 처리는 X)	score값이 연속으로 3점 이하면 발음에 주의하세요 문구?	김영대, 김가영	
피드백 등록	SpeechController	Authorization: Bearer {access token}	O	POST	api/speech/fb	{ "sessionId": String, 세션 아이디 "content": String, 피드백 내용 }	{ status: 200 message: "성공" data: { } }	200 : 성공 401 : 실패(발표 정보를 찾을 수 없음)	SessionId로 메모리에서 SpeechID 찾기 발표 테이블에서 발표 시작시간 조회해서 현재 시간 - 시작시간 계산 DB에 등록	김영대, 김가영	
개인 코멘트 등록	SpeechController	Authorization: Bearer {access token}	O	POST	api/speech/comment	{ "speechId": String, 스피치 PK "comment": String, 개인 코멘트 }	{ status: 200 message: "성공" data: { } }	200 : 성공 401 : 실패(발표 정보를 찾을 수 없음)		김영대, 김가영	
녹화 영상 업로드	SpeechController	Authorization: Bearer {access token} Content-Type: multipart/form-data	O	POST	api/speech/video	{ "speechId": int, 스피치 PK, "speechTimeInSeconds": int, 영상 시간(밀리초), "video": 영상 } 메시) curl -X "POST" \ "http://localhost:8000/api/speech/video" \ -H 'accept: */*' \ -H 'Authorization: Bearer {access token}' \ -H 'Content-Type: multipart/form-data' \ -F 'speechId=21' \ -F 'speechTimeInSeconds=3422' \ -F 'video=@"영상 이름".webm; type=video/webm'	{ status: 200 message: "성공" data: { } }	200: 영상 업로드 완료 404: 발표 정보를 찾을 수 없음		김가영	

API 이름	controller	권한(header)	구현 여부	http method	endpoint	request body	response body	status code message	ApiOperation (swagger 설명)	담당자	비고
결과 조회	RecordController	Authorization: Bearer (access token)	O	GET	api/record/{speechId}	없음	{ status: 200 message: "성공" data: { "speech": { "speechId": int, "speechTitle": string(발표 제목, 추가 예정), "category": string(다본 또는 발표의 카테고리) "userId": string(발표자 id), "speechMode": string("연습 모드" 또는 "스터디"), "startsAt": string?(발표 시작 시간), "comment": string(개인 의견) }, "video": { (video가 없든 경우 null) "videoPath": string(영상 저장 경로(사용자 컴퓨터)) "playTime": int(영상 재생 시간(분))? }, "score": { "date": null "volume": int(목소리 크기 점수), "speed": int(발화 속도 점수), "clarity": int(명료도 점수), "eye": int(시선 점수)?, "grade": int(종합 등급, volume, speed, clarity의 평균) }, "feedbacks": [(feedback이 없을 경우 빈 배열) { "content": string(피드백 내용), "timeLog": string("분:초", 영상에서의 시간) },], } }	200 : 성공 401 : 발표 정보를 찾을 수 없음	SpeechId에 해당하는 결과 조회 Feedback, Video, Score 값이 조회해야함	김가영	
결과 목록 조회	RecordController	Authorization: Bearer (access token)	O	POST	api/record	없음	{ status: 200 message: "성공" data: { { "speechId": int, "speechTitle": string(발표 제목, 추가 예정) "category": string(다본 또는 발표의 카테고리) "userId": string(발표자), "speechMode": string(혼자 연습/스터디 연습), "startsAt": string?(발표 시작 시간), "comment": string(개인 의견) }, } }	200 : 성공 401 : 실패	가장 최근 스피치 5개 결과를 조회한다.	김가영	
영상 삭제 요청	RecordController	Authorization: Bearer (access token)	O	DELETE	api/record/{speechId}	없음	{ status: 200 message: "성공" data: { "videoPath": String, 비디오 파일 경로 } }	200 : 성공 401 : 실패(영상 정보를 찾을 수 없음)	SpeechId와 연결된 Video Table 값 삭제 200 응답 받으면 front에서 파일 삭제	김가영	
통계 조회	RecordController	Authorization: Bearer (access token)	O	POST	api/record/score	없음	{ status: 200 message: "성공" data: { "totalTime": int(연습모드+스터디모드 연습 시간), "scoreChange": [{ "date": "yyyy-mm-dd hh:mm", "volume": int(목소리 크기 점수), "speed": int(발화 속도 점수), "clarity": int(명료도 점수), "grade": int(종합 등급, volume, speed, clarity의 평균) },], } }		모든 score를 날짜 오름차순으로 조회	김가영	
게시글 조회	BoardController	Authorization: Bearer (access token)	O	GET	api/board	{ pgno: int pageSize: int word: String }	{ status: 200 message: "조회 성공" data: { { int id string title string register_time int like_cnt }, } }	200 : 게시글 조회 성공 400 : 조회 실패	게시판의 게시글을 조회한다. (title로 검색)	박나린	
게시글 등록	BoardController	Authorization: Bearer (access token)	O	POST	api/board/create	{ "userId": String, 작성자 아이디", "title": String, 게시글 제목", "content": String, 게시글 내용", "videoId": int, 영상 번호 id }	{ status: 200 message: "등록 성공" }	200 : 등록 성공 400 : 등록 실패	게시판에 게시글을 등록한다	박나린	

[illegible]