

Rest API 설계

문서 설명

이 문서는 백엔드에서 작성한 REST API의 메서드, url, request body, Output에 대해 정리했다.

React에서 axios로 baseURL 설정을 아래와 같이 하면 된다.

- Method - 요청 HTTP 메서드
 - URI - base url을 제외한 요청 자원 주소
 - Path variable - URI에 {} 안의 값
 - Query string - 쿼리 스트링
 - Body - 요청 시 보낼 payload
 - Response data - 응답 데이터, snake case로 작성되어 있으나 실제 받는 데이터는 camel case로 보내짐!
- **RETURN 형식**
 - status : success, fail
 - data : 데이터 객체
 - message : 설명 메시지

Local: <http://localhost:8081/api>

Ubuntu:

Swagger(dev): <https://kichulove.da-haeng.com/api/swagger-ui.html#>

Header

```
{  
  "Authorization" : string  
}
```

⇒ 토큰 넣을 때 이렇게 해야 해음 **행쇼**

AccountController

Aa Method	≡ URI	≡ 설명	≡ Header	≡ Body	≡ Response data	≡ 비고
--------------	-------	------	----------	--------	-----------------	------

Aa Method	URI	설명	Header	Body	Response data	비고
<u>POST</u>	/users	회원가입 (role ROLE_STUDENT, ROLE_ADMIN)		{ loginId: string, name : string role : string pinCode : long (6자리) }		
<u>POST</u>	/login	로그인 스웨거에 없 습니다		{ loginId: string, password : string }	{ "accessToken": string, "refreshToken": string }	
<u>GET</u>	/users	사용자 정보 조회	access token		{ userId: Long, loginId : string, name: string, role : string, pinCode : Long }	
<u>GET</u>	/stage/admin	마이페이지 (선생님) studentsProgress : 학생별 진척도 (이 름, 버거스테이지 결 과, 버스 스테이지 결 과) 스테이지 결과는 최댓값을 반환→ 3 스테이지까지 성공했 으면 3반환 Avg : 정 수로 반환	access token		{ burgerAvg : long, busAvg : long, ! studentsProgress: [{ name : string, burgerStageResult : long, busStageResult : long }] }	
<u>GET</u>	/stage/student	마이페이지 (학생)	access token		{ name : string, burgerStageResult : long, busStageResult : long }	
<u>PATCH</u>	/stage	해당 stage가 성공 했을 때 maxStageNumber 를 업데이트	access token	{ categoryName : "BURGER", stageNumber : 3 }		
<u>제목 없음</u>						