

## ⑥ 개발 관련 참고사항

개발에 필요한 외부 api/라이브러리 등을 정리해 놓은 자료입니다.

### BE에 요구되는 사항

- 이메일 발송: SMTP 사용 (Sendgrid 사용하지 않음)
  - 프로젝트 관련 계정 생성 후 연동 예정 (####@wecruitcorp.com)
  - 계정의 '수신확인' 기능으로 전송 및 수신 내역 확인 가능
  - 실패 시 계정 메일로 실패 내역 전송 받을 수 있음
    - no\_reply@worksmobile.com에서 메일이 옴
  - 12월 18일 이전: 개발팀 내에서만 테스트하므로, 임시로 다른 계정 사용 (BE 내 협의 필요)
  - 12월 18일~: PM과 함께 테스트하므로, pm용 계정 연동 후 테스트
- 발송 제한 규칙
  - API 1회 요청: 500건 이내
  - 1일 요청: 네이버 웍스 내부 규칙이라 밝힐 수 없다고 함
    - 해당 요청량을 넘어서면 스팸되었다고 통보오며, 네이버 웍스 측에 스팸처리 풀어달라고 요청해야 함
  - 사용량 많아지면 SMTP 타 계정으로 이동 고려해야 함
- SMS 발송: 네이버 클라우드 내 SMS 발송 기능 사용 (국내만 발송)

관리번호	평가대상여부	이름	전화번호	이메일	조직식별자	상사직원	동료직원	부회장직원	...
20230001		김지영	010-1234-5678	qndjnf@korea.com	마케팅				
20230002		이민준	010-1234-5679	sddsfsv@korea.com	마케팅				
20230003		박서연	010-1234-5680	wiiquwioi@korea.com	마케팅				
20230004		최준호	010-1234-5681	sjkfahsjak@korea.com	마케팅				
20230005	Y	정은지	010-1234-5682	fjkgjdfkzh@korea.com	마케팅	20210001	20210004	20210006	
20230006		강민서	010-1234-5683	sdaafjshk@korea.com	마케팅				
20230007		윤현우	010-1234-5684	ajfhjkjhk@korea.com	마케팅				
20230008		장서윤	010-1234-5685	asjkqwj@korea.com	마케팅				
20230009	Y	한승민	010-1234-5686	skajmans@korea.com	마케팅	20210004	20230002	20230004	
20230010		송지원	010-1234-5687	jkshjkh@korea.com	마케팅				
20210001		황재현	010-1234-5688	qiouao@korea.com	법무				
20210002		김소민	010-1234-5689	akshfjkh@korea.com	법무				
20210003	Y	이예준	010-1234-5690	dfgndmm@korea.com	법무	20210001	20210006	20230007	
20210004		박지민	010-1234-5691	qwoeioui@korea.com	법무				
20210005		김현우	010-1234-5692	oweshjkh@korea.com	법무				
20210006		정유진	010-1234-5693	qkqjkjlk@korea.com	법무				

사용자가 업로드하는 엑셀파일 예시

- 엑셀 파일: 읽기, 생성 툴 필요

• 일기

- 프로젝트, 질문지 내용을 엑셀로 받아서 BE에서 유효성 체크 진행
- 유효성 체크 후 성공, 실패에 따라 결과값 메시지를 다르게 추산하여 전송
- 엑셀 파일 생성
  - 프로젝트 종료 후, 결과값을 엑셀로 다운로드 받을 수 있는 기능이 있음
  - 2024년 1월 출시 버전에는 pdf 보고서 생성 기능 없음

- **스케줄러 기능**

- 2024년 1월 출시 버전에는 기획된 내용 없으나, 향후 추가 가능

Next

계정 및 인증 API



Last modified 1mo ago

강민

# ① 계정 및 인증 API

Page description (optional)

## 보안 적용 사항

### 1. 인증(이메일/SMS)

1 +

- 1일 500회 제한, 분당 100회 제한 (10/12 업데이트)
- 로그인 시에도 사용되어 횟수 늘림(추후 횟수 늘어날 수 있음)

### 2. 세션 타임아웃 (токن 만료 시간 제어)

- access 토큰 만료 시간: 1분
- refresh 토큰 만료 시간: 1시간

### 3. 계정 동시 접속 제한

(11/23 업데이트)

- 로그인한 identify 정보로만 확인하여 접속 제한  
=> 예) 이메일로 로그인: 해당 이메일로 된 토큰 정보만 삭제  
      휴대폰으로 로그인: 해당 휴대폰으로 된 토큰 정보만 삭제
- 

(11/23 업데이트)

### 4. 마스터키

- 일반 사용자 계정 외에, 마스터키로 로그인할 수 있는 기능 추가
- 1차 개발 때 개발 불가능할 경우, 1차 개발 후 우선순위 상으로 개발 진행
  - 엑셀 파일 업로드 등 이슈 핸들링 위함

## (12/14 업데이트) 인사담당자 회원 가입하기(비인가)

POST /sign-up

💡 Feature

이메일 인증

이메일 중복 확인

## 휴대폰 인증

### 휴대폰 중복 확인

이메일과 휴대폰 OR 조건으로 중복 확인  
(=둘중 하나라도 다른 인사담당자 계정 정보와 중복되면 안됨)

### 관리자에게 가입 안내 알림 전송

#### 관리자 승인 이후 이용 가능

- admin 페이지 개발 전: DB에서 수동으로 승인 처리 예정
- 추가 데이터: 사용자 상태, 승인/승인 거절 시간, 탈퇴/삭제 시간

상태: # userStatus

#### 가입 정보별 최대 길이 (10/12 업데이트)

- 현 체커 기준 최대 데이터 길이 참고함 (회사이름: 34자, 이름: 34자, 이메일: 58자)

- 이메일: 80자
- 휴대폰: 20자
- 이름: 40자
- 직급: 40자
- 회사명: 40자
- 회사 연락처: 80자 (이메일 최대글자수를 따름)

#### 휴대폰(전화번호) 포맷 (10/12 업데이트)

- 9-11자리의 숫자  
-> API 혹은 엑셀 내의 값 모두 국가번호 없이 010-1234-1234로 - 포함되나, DB에 저장될 때에는 숫자로만 저장됨

#### profile 파일값에 들어갈 수 있는 확장자 (10/12 업데이트)

- tiff, jfif, bmp, gif, svg, png, jpeg, svgz, jpg, webp, ico, xbm, dib, pjp, apng, tif, pjpeg, avif
- 구글 프로필 사진 확장자 제한과 동일하게 함
- 대문자도 있을 수 있으니, 대소문자 구분하지 말 것

#### (11/10 업데이트)

프로필 사진 입력 시 기본이미지와 MBTI 이미지는 FE에 저장된 파일을, 사용자 입력 이미지는 사용자가 업로드한 파일을 multipart로 보냄

#### (11/2 업데이트)

request에서 profile을 제외한 데이터는 body value에 JSON 형태로 배치

```
profile: 'ESTJ',
body: {
  email: 'abc@naver.com',
  emailCode: '284733',
  emailSignature: '238f93ujfhds',
  ...
  position: '대리',
},
```

- (12/13 업데이트) 거절된 인사담당자 가입 불가: 이미 사용중인 이메일/휴대폰이라고 알림

- (12/14 업데이트)  
이미지가 아닌 파일을 프로필로 업로드하는 경우, 처리 추가  
status: 400  
message: 파일 형식이 맞지 않습니다.

#### Headers

Name	Type	Description
Content-type*	String	application/json or multipart/form-data

#### Request Body

Name	Type	Description
body: company: scale	String	규모 (Select)
body: company: contact*	String	회사 연락처 (최대 80자) (전화번호 혹은 이메일)
body: company: name*	String	회사명 (최대 40자)
body: name*	String	이름 (최대 40자)
body: emailSignature*	String	이메일 인증서명
body: emailCode*	String	이메일인증번호 (6자)
body: email*	String	이메일 (최대 80자)
body: agreeMarketing*	Boolean	마케팅 정보 동의
profile*	File	프로필 사진
body: position	String	직급 (최대 40자)
body: phone*	String	휴대폰 (최대 20자)
body: phoneSignature*	String	휴대폰 인증서명
body: phoneCode*	String	휴대폰 인증번호 (6자)

200: OK 성공 :: 400: Bad Re... 400: Bad Re... 409: Con... 400: Bad Re...

```
{  
    message: "성공했습니다."  
}
```

## (12/14 업데이트) 인사담당자 회원정보 수정하기

PUT /me

👉 Feature

이메일 변경은 불가능해서 제외됨

휴대폰 정보 수정 시

- 휴대폰 인증
- 휴대폰 중복 확인

(11/10 업데이트)

프로필 사진 입력 시 기본이미지와 MBTI 이미지는 FE에 저장된 파일을, 사용자 입력 이미지는 사용자가 업로드한 파일을 multipart로 보냄

(11/10 업데이트)

수정 시 profile, phone 값 있으면 업데이트, 없으면 이전 값 유지

(11/2 업데이트)

request에서 profile을 제외한 데이터는 body value에 JSON 형태로 배치

```
profile: 'ESTJ',
body: {
  name: '홍길동',
  company: {
    name: '(주)워크루트',
    contact: '031-738-4483',
    scale: 'S',
  }
  ...
  position: '대리',
},
```

- (12/14 업데이트)

이미지가 아닌 파일을 프로필로 업로드하는 경우, 처리 추가

status: 400

message: 파일 형식이 맞지 않습니다.

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}
Content-Type*	String	application/json or multipart/form-data

#### Request Body

Name	Type	Description
body: agreeMarketing*	Boolean	마케팅 정보 동의
body: company: scale	String	규모 (Select)
body: position	String	직급
body: company: contact*	String	회사 연락처 (전화번호 혹은 이메일)

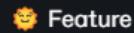
body: company: name*	String	회사명
body: name*	String	이름
body: phone	String	휴대폰
body: code	String	인증번호
body: signature	String	인증서명
profile	File	프로필 사진

200: OK 성공 :: 400: Ba... 400: Ba... 409: ... 401: Un... 400: Ba...

```
{
  message: "성공했습니다."
}
```

## (10/27 업데이트) 인사담당자 회원정보 불러오기

GET /me



인사담당자, 리뷰어 모두 속하는 경우 인사담당자 회원정보만 출력

데이터 없는 속성의 경우 null로 응답하도록 통일

(11/13 업데이트)

profile은 유형 관계 없이 BE에서 Base64 인코딩 형태로 받아서 렌더링

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 401: Unauthorized 액세스 토큰이 유효하지 ...

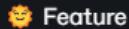
```

{
  name: "홍길동",
  email: "example@example.com",
  phone: "01012341234",
  company: {
    name: "네이버",
    contact: "01012341234",
    scale: { key: "S", value: "50인 이상 100인 미만" }, // 10/27 업데이트
  },
  profile: "Base64", // 11/10 업데이트
  position: "대리", // 10/11 업데이트
  agreeMarketing: true,
}

```

## 인사담당자 회원 탈퇴하기

**DELETE** /me



### 삭제 시 데이터 처리방법

- 본인/회사정보/프로젝트/진단지 정보 모두 삭제
- 단, 본인의 프로젝트에 다른 관리자가 초대되어 있는 경우, 프로젝트 삭제 하지 않음  
→ 이때 진단지는 삭제 가능한지 검토 필요 (진단지 정보 복사하여 저장하는 등 방지 대책 마련)

탈퇴 안내 알림 전송

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

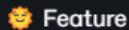
200: OK 성공

401: Unauthorized 액세스 토큰이 유효하지 ...

```
{  
    message: "성공했습니다."  
}
```

## (12/13 업데이트) 로그인(비인가)

**POST** /sign-in



### 사용자 정보 조회

- 인사담당자와 리뷰어 테이블 내 정보 모두 확인

### 토큰 발급

- 사용자 id값 저장 필요 (인사담당자의 경우, 프로젝트 생성 시 본인 여부 확인 때 필요)

- 토큰에 사용자 타입 정보 저장 > # userType

- 토큰 만료 시간 설정은 상단 <보안 적용 사항> 참고

- 로그인 방식에 따라 사용자 정보 다르게 저장

- 예: 이메일로 로그인 => 인사담당자와 리뷰어 중, 각각 해당 이메일로 된 계정정보 엮어서 토큰에 저장

- 예: 휴대폰 번호로 로그인 => 인사담당자와 리뷰어 중, 각각 해당 휴대폰번호로 된 계정정보 엮어서 토큰에 저장

- 토큰에 로그인한 사용자의 연락처 정보 저장 제안 (10/12 업데이트)

- > 리뷰어는 전체 정보를 토큰에 저장하는 대신 이메일 혹은 휴대폰번호 값만 저장

-> 리뷰어의 경우, 프로젝트마다 본인 id가 다를 수 있음  
-> 토큰에 전체 사용자 id를 저장하고 있기 보다 검색이 가능한 연락처를 갖고 있다가 BE에서 확인하는  
것이 좋을 것으로 판단됨

- (12/13 업데이트) 승인 거절했을 때는 승인이 통과되지 않았음을 알려주는 의미에서 미승인 경고문구가 뜨  
게 해 놓음. 아이디, 코드가 틀렸다고 할 경우에는 승인이 된 것으로 오해할 수 있기 때문임.

- (12/13 업데이트) 계정과 코드 일치 여부를 먼저 확인한 후, 그 다음에 승인된 인사담당자인지 아닌지 확인  
함

#### Request Body

Name	Type	Description
identify*	String	이메일 or 휴대폰
code*	String	인증번호
signature*	String	인증서명

200: OK 성공 :: 400: Bad R... 400: Bad R... 400: Bad R... 403: Forbid...

```
{  
    refreshToken: "H+CNGDDrUQRyI52wMtq9bwuIRTSX7B",  
    accessToken: "BSJENGDDASVEyI52wMtq9bwuIRTSX7B"  
}
```

## 로그아웃

POST /sign-out

💡 Feature

DB에 동시 접속 제한 위해 저장한 리프레시 토큰 만료 처리

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 401: Unauthorized 액세스 토큰이 유효하지 ...

```
{  
    message: "성공했습니다."  
}
```

## 인증번호 발송 요청(비인가)

**POST** <https://hunters.wecruitpro.com/auth-request>

🌟 Feature

인증코드 안내 알림 전송

- **identify**에 이메일로 기재된 경우 이메일로, 휴대폰인 경우 SMS 알림만 전송
- 메일, SMS 인증코드양식: 숫자 6자리 (10/12 업데이트)

**Request Body**

Name	Type	Description
identify*	String	이메일 or 휴대폰

200: OK 성공 :: 502: Bad Gateway 외부 메일/SMS ... :: 400: Bad Request 인증 요청 한...

```
{  
    // 이메일인 경우  
    identify: "example@example.com",  
    signature: "H+CNGDDxUQRyI52wMtq9bwuIRTSX7B"  
  
    // 휴대폰인 경우  
    identify: "01012345678",  
    signature: "H+CNGDDxUQRyI52wMtq9bwuIRTSX7B"  
}
```

JSON ↻

## 인증번호 검증(비인가)

**POST** <https://hunters.wecruitpro.com/auth-check>

🌟 Feature

**Request Body**

Name	Type	Description
code*	String	인증코드
signature*	String	인증서명
identify*	String	이메일 or 휴대폰

200: OK 성공 :: 400: Bad Request 인증 코드가 잘못된 경우

```
{  
    message: "성공했습니다."  
}
```

## 토큰 재발급

POST <https://hunters.wecruitpro.com/token>

🌟 Feature

access 토큰 만료 1분

refresh 토큰 만료 1시간

### Headers

Name	Type	Description
Authorization*	String	Bearer \${RefreshToken}

200: OK 성공 ::

401: Unauthorized 리프레시 토큰이 유효하...

```
{  
    refreshToken: "H+CNGDDrxUQRyI52wMtq9bwuIRTSX7B",  
    accessToken: "BSJENGDDASVEyI52wMtq9bwuIRTSX7B"  
}
```

## (1/8 업데이트) 약관 , 개인정보처리방침 다운로드 (비인가)

GET <https://hunters.wecruitpro.com/export/file>

### FE

- 1) responseType: 'blob'
- 2) 파일에 따라 type 값 전송
  - "TERMS" : 약관
  - "PRIVACY" : 개인정보 처리방침
  - "MARKETING": 마케팅 정보 수신
  - "CANDIDATE\_PRIVACY": 대상자 개인정보 처리방침 (\* 11/30 업데이트)
  - "REFEREE\_PRIVACY": 레퍼리 개인정보 처리방침 (\* 11/30 업데이트)
  - "PROJECT\_FORM": 프로젝트 엑셀 양식 (\* 11/30 업데이트)
  - "QUESTIONS\_FORM": 질문지 엑셀 양식 (\* 11/30 업데이트)
  - "CANDIDATE\_THIRD\_PARTY" : 대상자 제 3자 동의 (\* 1/8 업데이트)
  - "REFEREE\_THIRD\_PARTY" : 레퍼리 제 3자 동의 (\* 1/8 업데이트)

### BE

- 비회원도 접근 가능
- 서버 데이터 폴더 내부에 파일명\_언어명.pdf로 지정 예정  
약관: terms\_of\_use\_ko.pdf,  
개인정보 처리방침: privacy\_policy\_ko.pdf  
마케팅 정보 수신동의: marketing\_policy\_ko.pdf  
대상자 개인정보 처리방침: candidate\_policy\_ko.pdf  
레퍼리 개인정보 처리방침: referee\_policy\_ko.pdf  
프로젝트 엑셀 양식: project\_form\_ko.xlsx (\* 11/6 업데이트)  
질문지 엑셀 양식: questions\_form\_ko.xlsx (\* 11/6 업데이트)  
대상자 제 3자 동의: candidate\_third\_party\_ko.pdf (\* 1/8 업데이트)

- (12/14 업데이트) content-type에 파일 타입 명시.

예:

pdf → Content-Type="application/pdf"

excel → Content-Type="application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"

- (12/15 업데이트) 한글 파일명 사용할 수 있도록 헤더 인코딩 추가

### 🔗 Broken link

#### Query Parameters

Name	Type	Description
type*	String	파일 ("TERMS": 약관) ("PRIVACY": 개인정보 처리방침)

200: OK 성공

404: Not Found 해당 파일이 ...

400: Bad Request 위에 기재된 type...

Access-Control-Expose-Headers: Content-Disposition  
Content-Type: {파일 타입} // 12/14 업데이트  
Content-Disposition: attachment; filename=(파일명).pdf // 12/14 업데이트

```
{  
    data: (파일 데이터)  
}
```

JavaScript



Previous

개발 관련 참고사항

Next

다면평가 프로젝트



감민

Last modified 2mo ago

# ① 프로젝트 API

Page description (optional)

## (24/01/03 업데이트) 인사담당자용: 프로젝트 생성하기

`POST /project`

Feature

FE

- 프로젝트를 생성한 인사담당자 id는 최초 생성 시에만 배열에서 제거 불가 (UI 내 제어)
- 무작위매칭 선택지, 평가자 추가 선택지 => FE 생성

-----  
 (10/27 업데이트)

엑셀 샘플 값:

<https://we recruit corp.atlassian.net/browse/WCRT-362> >

파일명: 엑셀 정보 업로드 예시 > 시트명: 인력정보 요구 항목 참고

인사담당자가 아닌 경우 403 에러 리턴

엑셀 파일은 파일, 파일명 별도 저장 필요

- 확장자 제한: `xlsx, xls`
- 생성 후, 사용자가 엑셀 파일다운로드 가능

관리자 기능

### 1. 프로젝트:관리자 = 多:多 관계

- 관리자는 모두 같은 수준임: 프로젝트를 생성한 사람을 별도로 관리하거나 처리하지 않음
- 처음 생성한 사용자가 탈퇴 시에도 프로젝트에 연동된 다른 사용자가 해당 프로젝트 운용 가능

### 2. 생성 시에 employers 값은 처음 프로젝트를 생성한 인사담당자 id가 필수로 저장됨

-- FE가 토큰에서 사용자 id값 가져와서 전송 예정 (\* 혹시 다른 방법이 더 나을 경우 알려주세요)

기입 정보별 최대 길이 (10/19 업데이트)

- 현 체커 기준 최대 데이터 길이 참고함 (질문지 이름: 90자, 수험번호: 30자)

- 프로젝트명: 100자

<엑셀 파일 내 값>

- 관리번호: 40자
- 이름: 40자
- 휴대폰: 20자
- 이메일: 80자
- 조직식별자: 40자
- 직급식별자: 자연수 (100이하만 사용 예상) (10/27 업데이트)

(10/23 업데이트) 엑셀 유효성 검사 때 파일을 전송하기 때문에 생성 시에 중복으로 전송할 필요 없음, 파일  
고유면은 대신 전달해주시면 유효성 검사 때 보내주시면 됩니당

(10/25 업데이트)

randomSeniorNum, randomPeerNum, randomJuniorNum 값 중 하나라도 있으면 후보자와 같은 조직식별자 내에서 직급식별자로 직급을 구분하여 명수를 채워줌, 관리자가 지정한 레퍼리는 제외  
명수를 채울 수 있는 직급식별자에 해당하는 사람이 부족할 때는 채우지 않는다.

(12/28 업데이트)

프로젝트 생성 시 생성한 사람은 자동으로 연동  
employers 값은 추가 인사담당자가 없을 경우 빈 배열  
추가 인사담당자가 존재하는 경우 인 담 id 값 담긴 배열

(24/01/03 업데이트)

대상자 중에 아래의 케이스에 1명이라도 발생할 경우, 에러 출력

1. 랜덤 설정되어 있지 않음 (randomSeniorNum, randomPeerNum, randomJuniorNum 값 없음)
2. 합의 설정 없음 (isAddReferee: false)
3. 지정 레퍼리 없음: 상사, 동료, 부하로 된 레퍼리 모두 없음  
=> 400 에러, "레퍼리가 없는 대상자가 있습니다."

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}
Content-Type*	String	application/json

#### Request Body

Name	Type	Description
title*	String	프로젝트명
employers*	Number[]	관리자 id 배열
isAddReferee*	Boolean	레퍼리 추가선택 여부
excel*	String	파일 고유명
questionsId*	Number	질문지 id
randomJuniorNum	Number	랜덤 부하 수
randomPeerNum	Number	랜덤 동료 수
randomSeniorNum	Number	랜덤 상사 수

200: OK 성공	401: Un...	403: ...	400: Ba...	400: Ba...	400: Ba...
------------	------------	----------	------------	------------	------------

```
{
  message: "성공했습니다."
}
```

## 업로드 개요 응답(생성 전)

POST /project/excel

(10/27 업데이트)

엑셀 샘플 값:

<https://we recruit corp.atlassian.net/browse/WCRT-362> >

파일명: 엑셀 정보 업로드 예시 > 시트명: 인력정보 요구 항목 참고

### 🌟 Feature

인사담당자가 아닌 경우 403 에러 리턴

엑셀 파일 업로드 시 유효성 검사 (BE)

- 엑셀 파일 항목 중 필수값

1. 관리번호
2. 휴대전화번호 혹은 이메일(둘 중 하나 값은 필수로 입력)
3. 조직식별자

\* 평가대상여부 공란 가능하여 필수값 제외함 : 10/18 업데이트

- 관리번호 리스트는 여러개 입력이 가능하며, **띄어쓰기**를 구분자로 사용

(예: 상사직원란에 12 123 입력 => 상사직원 관리번호 12, 상사직원 관리번호 123)

=> 구분자 관련된 에러 체킹은 따로 없음

- 에러 메시지 리스트

(#에 이슈 있는 엑셀칸 번호 함께 표기: 예: A4, C3)

\* 실제 구현된 것에 맞춰서 수정됨 (24/01/03)

1. 관리번호 빙칸 불가: "숫자만 입력해 주세요(#)"

2. 관리번호 누락: "숫자만 입력해 주세요(#)" "값을 입력해 주세요(#)"

3. 관리번호 중복: "중복된 관리번호가 있습니다 (#, #, #)"

4. 평가대상여부 값이 O나 빙칸이아닌 경우: "평가대상여부에 'O' 또는 빙칸으로 입력해 주세요 (#)"

5. 이메일 혹은 휴대폰번호 둘다 입력 안된 경우: "휴대전화번호 혹은 이메일을 입력해 주세요(#)"

6. 이메일이 양식에 맞지 않는 경우: 이메일 양식에 맞게 입력해 주세요(#)

7. 휴대폰번호 양식에 맞지 않는 경우: 휴대전화번호 양식에 맞게 입력해 주세요(#)

8. 조직식별자 누락: "조직식별자를 입력해 주세요(#)"

9. 관리번호 항목에 없는 번호가 기재됨

- CASE1. sub-project id가 있는 경우(일부 수정인 경우)

- 상사, 동료, 부하직원 란에 기재된 관리번호 값이

- DB에 저장된 프로젝트 내 관리번호에서 찾을 수 없는 경우

(예: 상사직원란에 12 123 입력, DB 내 관리번호는 12, 13밖에 없음)

"해당되는 관리번호가 없습니다 (#, #, #)"

- CASE2. sub-project id가 없는 경우(전체 수정인 경우)

상사, 동료, 부하직원 란에 기재된 관리번호 값이

엑셀 내의 관리번호에서 찾을 수 없는 경우

(예: 상사직원란에 12 123 입력, 엑셀 내 관리번호는 12, 13밖에 없음)

"해당되는 관리번호가 없습니다(#)"

10. 상사, 동료, 부하의 관리번호란에 띄어쓰기 외 구분자를 사용할 경우

"해당되는 관리번호가 없습니다(#)" "관리번호와 띄어쓰기로만 정렬해주세요(#)"

11. 상사, 동료, 부하의 관리번호란에 대상자 자신을 적은 경우

"대상자를 제외하고 입력해 주세요(#)"

12. 평가대상여부 값이 모두 빈칸인 경우: "평가대상자를 한 명 이상 선택해 주세요"

**sub-project id**가 있는 경우

대상자의 관리번호는 업데이트하지 않음: DB 내에 아마 저장된 값을 따라감

유효성 검사 후 메시지 리턴

- 통과 성공시: 엑셀 업로드 개요 표현을 위한 데이터 응답
- 통과 실패시: 유효성 검사에서 나온 모든 에러 모아서 한번에 출력  
=> BE에서 여러 메시지 배열로 리턴 가능한지 확인 필요

(10/23 업데이트) 엑셀 유효성 검사 때 파일을 전송하기 때문에 생성 시에 중복으로 전송할 필요 없음, 파일 고유명을 대신 전달해서 유효성 검사 때 보낸 엑셀과 동일한지 확인

(10/25 업데이트)

직급식별자 예시)

직급식별자 2일 경우 1은 상사, 2는 동료, 3,4는 부하

직급식별자 5일 경우 1,2,3,4는 상사, 5는 동료, 6,7,10은 부하

직급식별자 값이 한 개 이상 채워졌는데, 전부 채워지지 않은 경우

"해당되는 직급식별자가 없습니다(#)"

직급식별자가 자연수가 아닌 경우

"숫자만 입력해 주세요(#))"

- (12/14 업데이트)

엑셀이 아닌 파일을 업로드하는 경우, 처리 추가

status: 400

message: 파일 형식이 맞지 않습니다.

## Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}
Content-Type*	String	multipart/form-data

## Request Body

Name	Type	Description
excel*	Binary	엑셀 파일

200: OK 성공

::

401: Unauth...

400: Bad Re...

403: Forbi...

400: Bad Re...

```
{
    "excel": "(파일 고유명)",
    "candidate": [
        {
            "identifier": "(관리번호)",
            "referees": 5
        },
        {
            "identifier": "(관리번호)",
            "referees": 3
        }
    ],
    "group": [
        {
            "identifier": "(관리번호)",
            "reviewers": 5
        },
        {
            "identifier": "(관리번호)",
            "reviewers": 3
        }
    ]
}
```

## (10/30 업데이트) 인사담당자용: 가입한 인사담당자인지 확인하기

**POST** /check-employer

(10/30 업데이트)

FE: API 요청 전, 프로젝트의 관리자로 이미 추가된 사람은 아닌지 확인 필요

-----

### 🌟 Feature

인사담당자가 아닌 경우 403 에러 리턴

email 값이 서비스에 회원가입한 사용자인지 체크 후 해당되는 사용자 정보 응답

- 상태 ACTIVE인 인사담당자 -> 성공 메시지 리턴
- 상태 WAITING, REJECT인 인사담당자 -> 실패 메시지 리턴

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

### Request Body

Name	Type	Description
email*	String	이메일

200: OK 성공 :: 401: Unauthorized ... :: 400: Bad Request ... :: 403: Forbidden ... :: 403: Forbidden ...

```
{  
    "id": 3,  
    "name": "김민수",  
    "email": "example@example.com",  
    "profile": "(base64로 인코딩된 주소)"  
}
```

## (10/25 업데이트) 인사담당자용: 프로젝트 수정을 위한 데이터 가져오기

GET /project/{id}

해당 프로젝트에 본인이 관리자로 들어 있는지 확인 후, 아닐 경우 403 에러 리턴

데이터가 없는 경우 안 보내는 것이 아닌 null로 보내도록 통일

status: # projectStatus

### Path Parameters

Name	Type	Description
*	String	프로젝트 id

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 404: Not Found id... :: 401: Unauthorized ... :: 403: Forbidden 본...

```
{  
    "id": 213,  
    "status": "WAITING", // 시작 대기인 경우  
    "status": "PROCESSING", // 진행중인 경우  
    "status": "CLOSED", // 종료된 경우  
    "title": "2023년 하반기 다면평가",  
    "questionsId": 25,  
    "excel": {  
        "candidate": [  
            {  
                "identifier": "(관리번호)",  
                "referees": 5  
            },  
            {  
                "identifier": "(관리번호)",  
                "referees": 3  
            }  
        ]  
    }  
}
```

```
        },
        "group": [
            {
                "identifier": "(관리번호)",
                "reviewers": 5
            },
            {
                "identifier": "(관리번호)",
                "reviewers": 3
            }
        ],
        "randomSeniorNum": 1, // 랜덤 상사 수 (10/25 업데이트)
        "randomPeerNum": 2, // 랜덤 동료 수 (10/25 업데이트)
        "randomJuniorNum": 1, // 랜덤 부하 수 (10/25 업데이트)
        "isAddReferee": true,
        "employers": [
            {
                "id": 3,
                "name": "김민수",
                "email": "example@example.com",
                "profile": "(base64로 인코딩된 데이터)"
            },
            {
                "id": 13,
                "name": "김이재",
                "email": "example@example.com",
                "profile": "(base64로 인코딩된 데이터)"
            }
        ]
    }
}
```

## (24/01/03 업데이트) 인사담당자용: 프로젝트 엑셀 유효성 검사 및 업로드 개요 응답(수정 전)

**PUT** /project/{id}/excel

(10/27 업데이트)

엑셀 샘플 값:

<https://we recruit corp.atlassian.net/browse/WCRT-362> >

파일명: 엑셀 정보 업로드 예시 > 시트명: 인력정보 요구 항목 참고

### ☀ Feature

인사담당자가 아닌 경우 403 에러 리턴

엑셀 파일 업로드 시 유효성 검사(BE)

- 엑셀 파일 항목 중 필수값

1. 관리번호
2. 휴대전화번호 혹은 이메일(둘 중 하나 값은 필수로 입력)
3. 조직식별자

\* 평가대상여부 공란 가능하여 필수값 제외함 : 10/18 업데이트

- 관리번호 리스트는 여러개 입력이 가능하며, 띄어쓰기 를 구분자로 사용

(예: 상사직원란에 12 123 입력 => 상사직원 관리번호 12, 상사직원 관리번호 123)

=> 구분자 관련된 에러 체킹은 따로 없음

- 에러 메시지 리스트

(#에 이슈 있는 엑셀칸 번호 함께 표기: 예: A4, C3)

\* 실제 구현된 것에 맞춰서 수정됨 (24/01/03)

1. 관리번호 빈칸 불가: "숫자만 입력해 주세요(#)"
2. 관리번호 누락: "숫자만 입력해 주세요(#)" "값을 입력해 주세요(#)"
3. 관리번호 중복: "중복된 관리번호가 있습니다 (#)"
4. 평가대상여부 값이 O나 빈칸이아닌 경우: "평가대상여부에 'O' 또는 빈칸으로 입력해 주세요 (#)"
5. 이메일 혹은 휴대폰번호 둘다 입력 안된 경우: "휴대전화번호 혹은 이메일을 입력해 주세요(#)"
6. 이메일이 양식에 맞지 않는 경우: 이메일 양식에 맞게 입력해 주세요 (#)
7. 휴대폰번호 양식에 맞지 않는 경우: 휴대전화번호 양식에 맞게 입력해 주세요(#)
8. 조직식별자 누락: "조직식별자를 입력해 주세요(#)"
9. 관리 번호 항목에 없는 번호가 기재됨

- CASE1. **sub-project id**가 있는 경우(일부 수정인 경우)

- 상사, 동료, 부하직원 란에 기재된 관리번호 값이  
- DB에 저장된 프로젝트 내 관리번호에서 찾을 수 없는 경우  
(예: 상사직원란에 12 123 입력, DB 내 관리번호는 12, 13밖에 없음)  
- "해당되는 관리번호가 없습니다(#, #, #)"

- CASE2. **sub-project id**가 없는 경우(전체 수정인 경우)

상사, 동료, 부하직원 란에 기재된 관리번호 값이  
엑셀 내의 관리번호에서 찾을 수 없는 경우  
(예: 상사직원란에 12 123 입력, 엑셀 내 관리번호는 12, 13밖에 없음)  
"해당되는 관리번호가 없습니다(#)"

2024/01/03 추가분

10. 상사, 동료, 부하의 관리번호란에 띄어쓰기 외 구분자를 사용할 경우

"해당되는 관리번호가 없습니다(#)" "관리번호와 띄어쓰기로만 정렬해주세요(#)"

11. 상사, 동료, 부하의 관리번호란에 대상자 자신을 적은 경우

"대상을 제외하고 입력해 주세요(#)"

12. 평가대상여부 값이 모두 빈칸인 경우: "평가대상자를 한 명 이상 선택해 주세요"

**sub-project id**가 있는 경우

대상자의 관리번호는 업데이트하지 않음: DB 내에 아마 저장된 값을 따라감

유효성 검사 후 메시지 리턴

- 통과 성공시: 엑셀 업로드 개요 표현을 위한 데이터 응답
- 통과 실패시: 유효성 검사에서 나온 모든 에러 모아서 한번에 출력  
=> BE에서 에러 메시지 배열로 리턴 가능한지 확인 필요

(10/23 업데이트) 엑셀 유효성 검사 때 파일을 전송하기 때문에 생성 시에 중복으로 전송할 필요 없음, 파일 고유명을 대신 전달해서 유효성 검사 때 보낸 엑셀과 동일한지 확인

(10/25 업데이트)

직급식별자 예시)

직급식별자 2일 경우 1은 상사, 2는 동료, 3,4는 부하  
직급식별자 5일 경우 1,2,3,4는 상사, 5는 동료, 6,7,10은 부하

직급식별자 값이 한 개 이상 채워졌는데, 전부 채워지지 않은 경우  
"해당되는 직급식별자가 없습니다(#)"

직급식별자가 자연수가 아닌 경우  
"숫자만 입력해 주세요(#)"

- (12/14 업데이트)  
엑셀이 아닌 파일을 업로드하는 경우, 처리 추가  
status: 400  
message: 파일 형식이 맞지 않습니다.

#### Path Parameters

Name	Type	Description
*	String	프로젝트 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}
Content-Type*	String	multipart/form-data

#### Request Body

Name	Type	Description
excel*	Binary	엑셀 파일

200: OK 성공 :: 401: Unauthorized :: 400: Bad Request :: 403: Forbidden :: 404: Not Found :: 400: Bad Request

```
{  
    "excel": "(파일 고유명)",  
    "candidate": [  
        {  
            "identifier": "(관리번호)",  
            "referees": 5  
        },  
        {  
            "identifier": "(관리번호)",  
            "referees": 3  
        }  
    ],  
    "group": [  
        {  
            "identifier": "(관리번호)",  
            "reviewers": 5  
        },  
        {  
            "identifier": "(관리번호)",  
            "reviewers": 3  
        }  
    ]  
}
```

```

    },
    {
        "identifier": "(관리번호)",
        "reviewers": 3
    }
]
}

```

## (10/31 업데이트) 인사담당자용: 프로젝트 수정하기 (projectStatus 가 WAITING인 상태)

**PUT** /project/{id}

(10/27 업데이트)

엑셀 샘플 값:

<https://wecruitcorp.atlassian.net/browse/WCRT-362> >

파일명: 엑셀 정보 업로드 예시 > 시트명: 인력정보 요구 항목 참고

### 🌟 Feature

해당 프로젝트에 본인이 관리자로 들어 있는지 확인 후, 아닐 경우 403 에러 리턴

프로젝트 시작 전에만 전체 수정 가능하며, 후보자별 하위 프로젝트가 하나라도 시작 시 전체 수정 불가  
 - project status 가 'WAITING' 이 아닌데 이 api로 수정요청할 경우, 403 에러 리턴

엑셀 파일은 파일, 파일명 별도 저장 필요

- 생성 후, 사용자가 엑셀 파일다운로드 가능
- excel 전송값이 없는 경우, 해당 값 없데이트

하지 않음

employers 값은 한 명 이상의 인사담당자가 연동되어야 함

- 수정 시 올린 값이 기존에 저장된 값과 같은 경우: 유지
- 수정 시 올린 값이 기존에 저장된 값에서 없어진 경우: 해당 정보 row 삭제
- 수정 시 올린 값이 기존에 저장된 값에서 추가된 경우: 해당 정보 추가

(10/23 업데이트) 엑셀 유효성 검사 때 파일을 전송하기 때문에 생성 시에 중복으로 전송할 필요 없음, 파일 고유명을 대신 전달해서 유효성 검사 때 보낸 엑셀과 동일한지 확인

(10/25 업데이트)

randomSeniorNum, randomPeerNum, randomJuniorNum 값 중 하나라도 있으면 후보자와 같은 조직식별자 내에서 직급식별자로 직급을 구분하여 명수를 채워줌, 관리자가 지정한 레퍼리는 제외 명수를 채울 수 있는 직급식별자에 해당하는 사람이 부족할 때는 채우지 않는다.

(24/01/03 업데이트)

대상자 중에 아래의 케이스에 1명이라도 발생할 경우, 에러 출력

1. 랜덤 설정되어 있지 않음 (randomSeniorNum, randomPeerNum, randomJuniorNum 값 없음)
2. 할의 설정 없음 (isAddReferee: false)
3. 지정 레퍼리 없음: 상사, 동료, 부하로 된 레퍼리 모두 없음  
 => 400 에러, "레퍼리가 없는 대상자가 있습니다."

### Path Parameters

Name	Type	Description
<b>id*</b>	String	프로젝트 id

## Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}
Content-Type*	String	multipart/form-data

## Request Body

Name	Type	Description
title*	String	프로젝트명
employers*	Number[]	관리자 id 배열
isAddReferee*	Boolean	레퍼리 추가선택 여부(합의)
questionsId*	Number	질문지 id
randomSeniorNum	Number	랜덤 상사 수
randomPeerNum	Number	랜덤 동료 수
randomJuniorNum	Number	랜덤 부하 수
excel	String	파일 고유명

200: OK 성공	⋮	4...	4...	4...	4...	4...	4...	4...	4...
{ message: "성공했습니다." }									

## 인사담당자용: 프로젝트 수정하기 (projectStatus가 WAITING이 아닌 상태)

`PATCH /project/{id}`

### 🌟 Feature

해당 프로젝트에 본인이 관리자로 들어 있는지 확인 후, 아닐 경우 403 에러 리턴

프로젝트 시작 후에는 관리자만 수정 가능하여 API 분리

employers 값은 한 명 이상의 인사담당자가 연동되어야 함

- 수정시 올린 값이 기존에 저장된 값과 같은 경우: 유지
- 수정시 올린 값이 기존에 저장된 값에서 없어진 경우: 해당 정보 row 삭제
- 수정시 올린 값이 기존에 저장된 값에서 추가된 경우: 해당 정보 추가

## Path Parameters

Name	Type	Description
id	String	프로젝트 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

#### Request Body

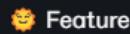
Name	Type	Description
employers*	Number[]	관리자 id 배열

200: OK 성공 :: 401: Unauthorized :: 400: Bad Request :: 404: Not Found :: 403: Forbidden

```
{
  message: "성공했습니다."
}
```

## (12/15 업데이트) 인사담당자용: 업로드한 엑셀 다운로드

GET /project/{id}/excel



프로젝트 수정 시 업로드한 엑셀을 다운로드하는 기능

- (12/14 업데이트) content-type에 파일 타입 명시.

예:

pdf → Content-Type="application/pdf"

excel → Content-Type="application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"

- (12/15 업데이트) 한글 파일명 사용할 수 있도록 헤더 인코딩 추가

#### Path Parameters

Name	Type	Description
id*	String	프로젝트 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 404: Not Found id... :: 403: Forbidden 본... :: 401: Unauthorized ...

Access-Control-Expose-Headers: Content-Disposition  
Content-Type: {파일 타입} // 12/14 업데이트  
Content-Disposition: attachment; filename=(파일명).xlsx // 12/14 업데이트

```
{  
    data: (파일 데이터)  
}
```

## (10/25 업데이트) 인사담당자용: 프로젝트 상태 변경하기(시작/종료)

PUT /project/{id}/status

### 🌟 Feature

프로젝트 전체 시작과 전체 종료를 할 수 있음

api status: # projectStatusRequest

프로젝트 status: # projectStatus

하위 프로젝트의 status: # subProjectStatus

- 프로젝트 시작인 경우 api status: START

(10/25 업데이트)

프로젝트의 status를 PROCESSING으로 변경

하위 프로젝트의 status를 ACCEPTING으로 변경

대상자의 status를 ACCEPTING으로 변경

평가자 status: WAITING >> WAITING

프로젝트 내 모든 대상자들에게만 이메일/SMS 알림 즉시 전송

=> 이메일, SMS 값을 확인한 후, 기재된 곳으로 발송

예: 이메일만 기재된 경우 이메일 발송

휴대폰번호만 기재된 경우 SMS 발송

이메일과 휴대폰번호 모두 기재된 경우 이메일, SMS 모두 발송

=> 이메일, SMS 발송 실패해도 api 성공처리함 (프로젝트별로 수동 재전송 가능 있음)

- 프로젝트 종료인 경우 api status: END

프로젝트의 status를 CLOSED으로 변경

하위 프로젝트 상태 변경, 알림 전송 없음

### Path Parameters

Name	Type	Description
id*	String	프로젝트 id

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

#### Request Body

Name	Type	Description
status*	String	status 요청값 전송

200: OK 성공	⋮	404: N...	401: Un...	400: B...	403: Fo...	400: B...
<pre>{     message: "성공했습니다." }</pre>						

## (2/21 업데이트) 프로젝트 리스트 보여주기 (인사담당자)

GET /projects

👉 Filter&Sort

# projectListOrder

# projectListStatus

👉 Enum

# projectStatus

👉 Initial Query

keyword: ""

order: "NEW"

status: "PROCESSING"

page: "0"

👉 Feature (요청)

keyword

- 빈 값일 경우, 전체 리스트를 보여줌 (기본값)

- keyword 값 검색 범위: 프로젝트명

=> keyword를 일부라도 포함하는 경우 검색 결과로 리턴

status

- 프로젝트의 상태

- 빈 값 없음

- 값이 "CLOSED"인 경우에는 종료된 프로젝트 리스트를 보여줌

page값은 0부터 시작하며 기본 0임

한 페이지에 보이는 리스트 개수: 10개로 고정

### 🌟 Feature (응답)

응답 데이터가 없는 경우 안 보내는 것이 아닌 null로 보내도록 통일

#### (11/13 업데이트)

해당 사용자와 관계된 프로젝트 정보를 모두 보여줌 (projects, pagination 값)

- 기본 조건: 로그인한 정보 (이메일 값 혹은 휴대폰값) 기준, 본인이 프로젝트의 관리자(인사담당자)로서 참여하는 프로젝트 정보를 보여줌
- 기본 조건 + query 에 모두 and 조건으로 맞아야 함

#### percent

- 값 산출 기준: (전체 필수 문항 응답 완료된 대상자 + 평가자 수) / (전체 필수 문항 응답 완료된 대상자 + 평가자 수)
- 프로젝트 종료할 때까지 대상자, 레퍼리 모두 응답 수정 가능
- 응답은 필수값 모두 입력 안해도 제출 가능 (임시저장처럼 사용)

#### closedAt

- "프로젝트" status가 CLOSED인 경우, 프로젝트 종료일시 추가

#### totalElements (processing, closed 포함) - 2/21 업데이트

- "쿼리와 관계없이" 해당 사용자가 관리자인 프로젝트 정보를 카운팅해서 노출함
- processing: 프로젝트 status가 CLOSED가 아닌 것을 카운팅 (WAITING, PROCESSING)
- closed: 프로젝트 status가 CLOSED인 것을 카운팅

### Query Parameters

Name	Type	Description
page*	String	페이지 수
order*	String	정렬
keyword*	String	검색
status*	String	프로젝트 상태

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 400: Bad Request... :: 401: Unauthorized 액션 :: 403: Forbidden 인사

{

"projects": [ // query의 모든 값과 일치되는 본인의 프로젝트

,

```
i
    "id": 253, // 프로젝트 id
    "status": "WAITING" // 프로젝트 상태 (projectStatus)
    "title": "2023년 하반기 디자인 평가", // 프로젝트명
    "createdAt": "2023-10-05T14:48:00.000Z",
    "questionsName": "2023년 하반기 질문지", // 질문지명
    "percent": 50,
    "closedAt": "2023-10-10T14:48:00.000Z" // 종료된 경우 프로젝트 종료일시 추가
},
],
"pagination": { // query의 모든 값과 일치되는 본인의 프로젝트
    "totalElements": 5, // 11/13 업데이트
    "totalPages": 25,
    "currentPage": 1
},
"totalElements": { // (2/21 업데이트) query와 관계 없이 꼭고 status만 고려한 본인이 관리하는 프로젝트 수
    "processing": 10, // 진행중인 프로젝트 수 (status WAITING, PROCESSING 인 케이스)
    "closed": 0 // 종료된 프로젝트 수 (status CLOSED 인 케이스)
}
}
```

## (11/13 업데이트) 프로젝트 리스트 보여주기 (리뷰어)

**GET** /projects/reviewer

👉 Filter&Sort

# projectListOrder

# projectListStatus

👉 Enum

# candidateStatus

# refereeStatus

👉 Initial Query

keyword: ""

order: "NEW"

status: "PROCESSING"

page: "0"

👉 Feature (요청)

keyword

- 빈 값일 경우, 전체 리스트를 보여줌 (기본값)

- keyword 값 검색 범위: 프로젝트명

=> keyword를 일부라도 포함하는 경우 검색 결과로 리턴

status

- 프로젝트의 상태
- 번 값 없음
- 값이 "CLOSED"인 경우에는 종료된 프로젝트 리스트를 보여줌

page값은 0부터 시작하며 기본 0임

한 페이지에 보이는 리스트 개수: 10개로 고정

### 💡 Feature (응답)

응답 데이터가 없는 경우 안 보내는 것이 아닌 null로 보내도록 통일

### (2/19 업데이트)

해당 사용자와 관계된 프로젝트 정보 중를 status가 WAITING이 아닌 것만 모두 보여줌 (projects, pagination 값)

- 기본 조건: 로그인한 정보 (이메일 값 혹은 휴대폰값) 기준, 본인이 프로젝트의 리뷰어(대상자 혹은 레퍼리)로서 참여하는 프로젝트 정보를 보여줌
- 기본 조건 + query 에 모두 and 조건으로 맞아야 함

### closedAt

- "프로젝트" status가 CLOSED인 경우, 프로젝트 종료일시 추가

### candidateName

- 평가자의 경우, 본인이 평가하는 대상자의 정보가 필요함
- 리뷰어 이름은 선택값이기 때문에, 값이 저장된 상황에 따라 다르게 전송함
  - 이름 값이 있으면 이름,
  - 이름 값이 없으면 관리번호를 응답값으로 보냄

### Query Parameters

Name	Type	Description
page*	String	페이지 수
order*	String	정렬
keyword*	String	검색
status*	String	프로젝트 상태

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 400: Bad Request 쿼리가 잘못... :: 401: Unauthorized 액세스 토큰이 유...

{

```
"projects": [ // query의 모든 값과 일치되는 본인의 프로젝트
{
    // 대상자평가인 경우 (target CANDIDATE)
    "id": 10, // 하위 프로젝트 id
```

```

    "title": "2023년 하반기 다면평가", // 프로젝트명
    "createdAt": "2023-10-05T14:48:00.000Z",
    "questionsName": "2023년 하반기 질문지", // 질문지명
    "status": "WAITING", // 시작 대기중인 경우 (candidateStatus)
    "status": "ACCEPTING", // 대상자 동의 요청 중인 경우 (candidateStatus)
    "status": "AGREEING", // 합의중인 경우 (candidateStatus)
    "status": "PROCESSING", // 진행중인 경우 (candidateStatus)
    "status": "COMPLETE", // 완료된 경우 (candidateStatus)
    "closedAt": "2023-10-10T14:48:00.000Z" // 종료된 경우 프로젝트 종료일시 추가
},
{
    // 레퍼리평가인 경우 (target REFEREE)
    "id": 10, // 하위 프로젝트 id
    "title": "2023년 하반기 다면평가", // 프로젝트명
    "createdAt": "2023-10-05T14:48:00.000Z",
    "questionsName": "2023년 하반기 질문지", // 질문지명
    "candidateName": "홍길동", // 대상자의 이름 혹은 관리번호 (대상자 이름이 있는 경우)
    "candidateName": "124", // 대상자의 이름 혹은 관리번호 (대상자 이름이 없는 경우)
    "status": "WAITING", // 시작 대기중인 경우 (refereeStatus)
    "status": "PROCESSING", // 진행중인 경우 (refereeStatus)
    "status": "COMPLETE", // 완료된 경우 (refereeStatus)
    "closedAt": "2023-10-10T14:48:00.000Z" // 종료된 경우 프로젝트 종료일시 추가
}
],
{
    "pagination": { // query의 모든 값과 일치되는 본인의 프로젝트
        "totalElements": 50, // 11/13 업데이트
        "totalPages": 25,
        "currentPage": 1
    }
}

```

## (01/03 업데이트) 하위 프로젝트 리스트 보여주기

**GET** /project/{id}/sub-projects

👉 Filter&Sort

# subProjectListOrder

# subProjectListStatus

👉 Enum

# subProjectStatus

👉 Initial Query

keyword: ""

order: "NEW"

status: "ALL"

page: "0"

👉 Feature (요청)

keyword

- 빈 값일 경우, 전체 리스트를 보여줌 (기본값)
- keyword 값 검색 범위: 후보자 이름, 레퍼리 이름, 후보자 관리번호, 레퍼리 관리번호 (12/28 추가)
  - => keyword를 일부라도 포함하는 경우 검색 결과로 리턴

#### status

- 하위 프로젝트의 상태
- 빈 값 없음

page값은 0부터 시작하며 기본 0임

한 페이지에 보이는 리스트 개수: 10개로 고정

#### 🌟 Feature (응답)

응답데이터가 없는 경우 안 보내는 것이 아닌 null로 보내도록 통일

프로젝트 내 하위 프로젝트 정보를 모두 보여줌 (subProjects, pagination 값)

- 기본 조건: 로그인한 정보 (이메일 값 혹은 휴대폰값) 기준, 본인이 프로젝트의 관리자(인사담당자)로서 참여하는 프로젝트의 하위 프로젝트 정보를 보여줌
  - => project id 기준으로, 본인이 관리자(인사담당자)가 아닌 경우 예러 리턴

(11/13 업데이트)

- 기본 조건 + query 에 모두 and 조건으로 맞아야 함
  - status 값에 따른 조건
    - all: 하위 프로젝트 상태 무관한 전체 하위프로젝트 수 카운트
    - complete: 하위 프로젝트 내 대상자, 레퍼리 모두 응답 완료한 하위프로젝트 수 카운트
    - incomplete: 하위 프로젝트 내 대상자, 레퍼리 모두 응답 완료되지 않은 하위프로젝트 수 카운트

(11/13 업데이트)

title

- 프로젝트 제목

percent

- 값 산출 기준: (전체 필수 문항 응답 완료된 대상자 + 평가자 수) / (전체 필수 문항 응답 완료된 대상자 + 평가자 수)

updatedAt

- 하위 프로젝트의 모든 대상자, 레퍼리가 평가지에 응답을 한 날짜 중 최신 날짜를 띄워줌
- 질문지 응답에 대해 업데이트 날짜만 보여주므로, 다음의 경우에는 null로 리턴됨
  - 예) 하위 프로젝트 상태가 PROCESSING가 아닌 경우 => null
  - 하위 프로젝트 상태가 PROCESSING이지만, 응답을 저장한 사람이 없는 경우 => null

(11/15 업데이트)

- 후보자 정보에 profile 값 추가
  - profile은 유형 관계 없이 BE에서 Base64 인코딩 형태로 받아서 렌더링

(01/03 업데이트) referees 값

프로젝트의 설정에 따라 다르게 표시됨

case 1-1. 프로젝트 isAddReferee 값이 false, 하위 프로젝트 상태가 WAITING, ACCEPTING, CONFIRMING 인 경우

출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리

case 1-2. 프로젝트 isAddReferee 값이 false, 하위 프로젝트 상태가 PROCESSING인 경우

출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리, 검토 완료 후 직급 부여된 레퍼리

case 2-1. 프로젝트 isAddReferee 값이 true, 하위 프로젝트 상태가 WAITING, ACCEPTING, AGREEING인 경우

출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리

case 2-2. 프로젝트 isAddReferee 값이 true, 하위 프로젝트 상태가 CONFIRMING인 경우

출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리, 합의 완료 후 직급 부여된 레퍼리

case 2-3. 프로젝트 isAddReferee 값이 true, 하위 프로젝트 상태가 PROCESSING인 경우

출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리, 합의 완료 후 직급 부여된 레퍼리, 검토 완료 후 직급 부여된 레퍼리

#### Path Parameters

Name	Type	Description
*	String	프로젝트 id

#### Query Parameters

Name	Type	Description
keyword*	String	검색
order*	String	정렬
status*	String	상태
page*	String	페이지 수

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

401: Unauthorized 액세스 토큰이 유효하지 ... :: 2.. 400: Bad R... 403: Forbidden...

```
{  
    // 액세스 토큰 없거나 시간이 만료되거나, JWT 토큰 형식이 아닌 경우  
    message: "인증이 유효하지 않아 토큰을 재발급합니다."  
}
```

## (1/31 업데이트) 인사담당자용: 프로젝트 보고서 다운로드

GET /project/{id}/report

🌟 Feature

프로젝트 종료 시 하위 프로젝트별로 대상자, 레퍼리들의 점수를 1개의 엑셀파일로 제작 후, 하위프로젝트별 엑셀 파일들을 묶어서 하나의 압축파일로 다운로드

#### <파일명>

- 프로젝트 압축 파일명: 프로젝트명.zip
- 하위 프로젝트 개별 엑셀 파일명: 후보자명 혹은 관리번호.xlsx  
=> 후보자명 있을 경우 후보자명, 없을 경우 관리번호로 이름 지정

#### <개별 엑셀 파일 값>

- 소제목, 상세설명 제외하고 객관식/주관식 질문인 경우만 추출
- 주관식 답변의 경우 첫번째 원점수 칸에 기입

제1회	제2회	제3회	제4회	제5회	제6회	제7회	제8회	제9회	제10회	제11회	제12회	제13회	제14회	제15회	제16회	제17회	제18회	제19회	제20회
제1회	제2회	제3회	제4회	제5회	제6회	제7회	제8회	제9회	제10회	제11회	제12회	제13회	제14회	제15회	제16회	제17회	제18회	제19회	제20회

#### 컬럼 값

- 관리번호: 질문의 관리번호 (A1&2 줄 합침)
- 범주1~3: 질문의 범주 (B1&2, C1&2, D1&2 각각 줄 합침)
- 질의문: 질문 내용 전체 (F1&2 줄 합침)
- 이름(관계): 대상자, 레퍼리의 이름 및 관계 표시 (F1부터 가로컬럼 2개씩 합침)  
=> 이름 없을 경우 관리번호 (관계) 표시
- 원점수: 사용자가 기입한 응답 그대로 표시
- 조정값: 객관식인 경우 조정방법 고려해서 조정값 표시  
=> (24/01/11 업데이트) 상세 내용은 자라 이슈 내 프로젝트 결과값 엑셀 파일 참고  
<https://wecruitcorp.atlassian.net/jira/software/projects/WCRT/boards/1?selectedIssue=WCRT-362>
- (12/14 업데이트) content-type에 파일 타입 명시.  
예:

pdf → Content-Type="application/pdf"

excel → Content-Type="application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"

- (12/15 업데이트) 한글 파일명 사용할 수 있도록 헤더 인코딩 추가

- (1/31 업데이트)

특이점수 제외법 구현사항: 점수 외에 특별하게 정렬하지 않음

상위 점수는 먼저 생성된 것부터 제외함.

하위 점수는 나중에 생성되는 것을 제외함.

예: 5, 5, 4, 3, 3이며 상위/하위 20% 삭제 시 => X, 5, 4, 3, X

#### Path Parameters

Name	Type	Description
id*	String	프로젝트 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 404: Not Found id... :: 401: Unauthorized ... :: 403: Forbidden 본...

Access-Control-Expose-Headers: Content-Disposition  
Content-Type: application/json // 12/14 업데이트

Content-Type: application/json // 12/14 업데이트  
Content-Disposition: attachment; filename=(파일명).zip // 12/14 업데이트

```
{  
    data: (파일 데이터)  
}
```

## (24/01/04 업데이트) 인사담당자용: 프로젝트 리마인드

POST /project/{id}/remind

### 🌟 FE

\* 리마인드 버튼 비활성화 (2024/01/04 업데이트)

- 1) 프로젝트 상태: 종료됨
- 2) 프로젝트 상태: 진행중 & percent: 100

-----

### 💡 Feature

+ :: 해당 요청 시점에 미완료된(모든 필수 응답 완료 안됨) 대상자, 레퍼리에게 리마인드 알림 발송



=> 이메일, SMS 값을 확인한 후, 기재된 곳으로 발송

예: 이메일만 기재된 경우 이메일 발송

휴대폰번호만 기재된 경우 SMS 발송

이메일과 휴대폰번호 모두 기재된 경우 이메일, SMS 모두 발송

=> 이메일, SMS 발송 실패해도 api 성공처리함

발송 제한 시점 추가 (2024/01/04 업데이트)

1. 하위 프로젝트 내 리뷰어 상태: 응답 완료 (COMPLETE)
2. 하위 프로젝트 내 리뷰어 상태: 대기중 (WAITING)
3. 하위 프로젝트 상태: CONFIRMING => 대상자, 레퍼리 모두에게 발송하지 않음

### Path Parameters

Name	Type	Description
id*	String	프로젝트 id

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 404: Not Found id... :: 401: Unauthorized ... :: 403: Forbidden 본...

```
{  
    message: "성공했습니다."  
}
```

←

Previous

다면평가 프로젝트

Next

하위 프로젝트 API

→

강민

Last modified 7h ago

# ① 하위 프로젝트 API

Page description (optional)

## (11/20 업데이트) 인사담당자, 대상자용: 하위 프로젝트 상세 보여주기

`GET /sub-project/{id}`

 Feature

접근

- 인사담당자: 해당 프로젝트에 관리자인 경우에만 항상 접근 가능  
=> 관리자가 아닌 인사담당자는 403 에러 리턴

- 대상자: 해당 프로젝트의 대상자 & `isAddReferee` 값이 `true`인 경우 & 하위 프로젝트의 `status`가 `AGREEING`인 경우 접근 가능  
=> 세 조건을 만족시키지 못할 경우 403 에러 리턴

- 레퍼리이거나, 프로젝트와 관련 없는 사용자인 경우 => 403 에러 리턴

하위 프로젝트가 진행중인 경우 각 후보자와 레퍼리의 `updatedAt` 값은 응답이 변경될 때마다 갱신됨

프로젝트 `isAddReferee` 값이 `false`인 경우

하위 프로젝트 `status`가 `WAITING`, `PROCESSING`일 때,  
액셀에서 지정된 레퍼리와 랜덤으로 할당한 레퍼리만 표시

프로젝트 `isAddReferee` 값이 `true`인 경우

하위 프로젝트 `status`가 `WAITING` 일 때,  
=> 액셀에서 지정된 레퍼리와 랜덤으로 할당한 레퍼리만 표시

하위 프로젝트 `status`가 `AGREEING`일 때,

=> 액셀에서 지정된 레퍼리와 랜덤으로 할당한 레퍼리, 같은 조직 식별자 레퍼리를 표시

하위 프로젝트 `status`가 `PROCESSING`일 때,

=> 액셀에서 지정된 레퍼리와 랜덤으로 할당한 레퍼리, 합의된 레퍼리를 표시

(10/26 업데이트) `referees` 값

- 프로젝트의 설정에 따라 다르게 표시됨
- case 1. 프로젝트 `isAddReferee` 값이 `false`인 경우

- 하위 프로젝트 상태가 `WAITING`, `ACCEPTING` 경우

- 출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리

- 하위 프로젝트 상태가 `CONFIRMING`인 경우

- 출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리, 같은 조직 식별자 레퍼리

- 하위 프로젝트 상태가 `PROCESSING`인 경우

- 출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리, (인담)추가 선택된 레퍼리

- case 2. 프로젝트 isAddReferee 값이 true인 경우
- 하위 프로젝트 상태가 WAITING, ACCEPTING 경우
- 출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리
  
- 하위 프로젝트 상태가 AGREEING, CONFIRMING인 경우
- 출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리, 같은 조직 식별자 레퍼리
  
- 하위 프로젝트 상태가 PROCESSING인 경우
- 출력되는 레퍼리 종류: 엑셀 지정 레퍼리, 랜덤 레퍼리, (인감, 대상자)추가 선택된 레퍼리

### Select

# relationship

### Enum

# subProjectStatus

# candidateStatus

# refereeStatus

=> 대상자, 레퍼리의 완료 기준: 본인의 필수 응답값 모두 응답 완료

### updatedAt

- 해당 리뷰어가 평가지에 응답을 한 날짜 중 최신 날짜를 띠워줌
- 질문지 응답에 대해 업데이트 날짜만 보여주므로, 다음의 경우에는 전송값이 없거나 null로 리턴됨  
예) 하위 프로젝트 상태가 PROCESSING가 아닌 경우 => updatedAt을 전송하지 않음  
하위 프로젝트 상태가 PROCESSING이지만, 응답을 하지 않은 경우 => null

(10/27 업데이트)

기본이미지 및 MBTI 이미지는 프론트에서 보관 후 응답 값(# profile)에 맞게 렌더링  
사용자 입력 이미지는 백에서 Base64 인코딩 형태로 받아서 렌더링

(11/20 업데이트) 하위 프로젝트 상세에서 진행도 표현을 위한 status: 'PROCESSING'일 때 percent 값 추가

- 값 산출 기준: (전체 필수 문항 응답 완료된 대상자 + 평가자 수) / (전체 필수 문항 응답 완료된 대상자 + 평가자 수)

relationship 값 케이스 구분

- 지정 레퍼리인 경우

직급식별자	상사직원	동료직원	부하직원
2			
2			

위 사진에 해당하는 관계 값

[PUT] /sub-project/{id}/referee-relationship

위 API를 통해 업데이트될 수 있음

- 랜덤 레퍼리인 경우

직급식별자 값(relationshipIdentifier)과 동일한 값

[PUT] /sub-project/{id}/referee-relationship

위 API를 통해 업데이트될 수 있음

- 추가 레퍼리인 경우

[PUT] /sub-project/{id}/referee-relationship

위 API를 통해 지정된 값

**Path Parameters**

Name	Type	Description
id*	String	하위 프로젝트 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK (10/26 업데이트) 성공 :: 404: Not Found :: 401: Unauthorized :: 403: Forbidden

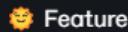
```
{
    // 하위 프로젝트가 시작 대기중인 경우
    "status": "WAITING", // 하위 프로젝트 status
    "percent": null, // 11/20 업데이트
    "isAddReferee": false, // 프로젝트의 isAddReferee 값
    "isAddReferee": true,
    "candidate": {
        "id": 30,
        "name": "이강민", // 이름 없으면 관리번호 노출
        "group": "개발", // 조직식별자
        "email": "example@example.com",
        "phone": "01012341234",
        "profile": null,
        "updatedAt": null,
        "status": "WAITING"
    },
    "referees": [
        {
            "id": 20,
            "name": "김병지", // 이름 없으면 관리번호 노출
            "relationship": { "key": "SENIOR", "value": "상사" }, // 10/31 업데이트: 레퍼리
            "relationship": null, // 10/31 업데이트: 레퍼리 관계가 지정되지 않은 경우
            "relationshipIdentifier": { "key": "SENIOR", "value": "상사" } // 직급식별자가
            "relationshipIdentifier": null, // 직급식별자가 존재하지 않는 경우
            "email": "example@example.com",
            "phone": "01012341234",
            "profile": null,
            "updatedAt": null,
            "status": "WAITING"
            "isSelected": true, // 엑셀에서 지정되거나 무작위 지정된 레퍼리인 경우
            "isSelected": false // 프로젝트 생성 시 지정되지 않은 레퍼리인 경우
        }
    ],
    // 하위 프로젝트가 대상자 등의 요청인 경우
    "status": "ACCEPTING", // 하위 프로젝트 status
    "percent": null, // 11/20 업데이트
    "isAddReferee": false, // 프로젝트의 isAddReferee 값
    "isAddReferee": true,
    "candidate": {
        "id": 30,
        "name": "이강민", // 이름 없으면 관리번호 노출
        "group": "개발", // 조직식별자
        "email": "example@example.com",
        "phone": "01012341234",
        "profile": null,
        "updatedAt": null,
        "status": "ACCEPTING"
    }
}
```

```
"phone": "01012341234",
"profile": null,
"updatedAt": null,
"status": "ACCEPTING"
},
"referees": [
{
"id": 20,
"name": "김병지", // 이름 없으면 관리번호 노출
"relationship": { "key": "SENIOR", "value": "상사" }, // 10/31 업데이트: 레퍼리
"relationship": null, // 10/31 업데이트: 레퍼리 관계가 지정되지 않은 경우
"relationshipIdentifier": { "key": "SENIOR", "value": "상사" } // 직급식별자가
"relationshipIdentifier": null, // 직급식별자가 존재하지 않는 경우
"email": "example@example.com",
"phone": "01012341234",
"profile": null,
"updatedAt": null,
"status": "WAITING"
"isSelected": true, // 엑셀에서 지정되거나 무작위 지정된 레퍼리인 경우
"isSelected": false // 프로젝트 생성 시 지정되지 않은 레퍼리인 경우
},
],
// 하위 프로젝트가 할의중인 경우
"status": "AGREEING", // 하위 프로젝트 status
"percent": null, // 11/20 업데이트
"isAddReferee": false, // 프로젝트의 isAddReferee 값
"isAddReferee": true,
"candidate": {
"id": 30,
"name": "이강인", // 이름 없으면 관리번호 노출
"group": "개발", // 조직식별자
"email": "example@example.com",
"phone": "01012341234",
"profile": "DEFAULT or ISFP or (Base64)", // 10/27 업데이트
"updatedAt": null,
"status": "AGREEING"
},
"referees": [
{
"id": 20,
"name": "김병지", // 이름 없으면 관리번호 노출
"relationship": { "key": "SENIOR", "value": "상사" }, // 레퍼리 관계가 지정된 경우
"relationship": null, // 레퍼리 관계가 지정되지 않은 경우
"relationshipIdentifier": { "key": "SENIOR", "value": "상사" } // 직급식별자가
"relationshipIdentifier": null, // 직급식별자가 존재하지 않는 경우
"email": "example@example.com",
"phone": "01012341234"
"profile": null,
"updatedAt": null,
"status": "WAITING"
"isSelected": true, // 엑셀에서 지정되거나 무작위 지정된 레퍼리인 경우
"isSelected": false // 프로젝트 생성 시 지정되지 않은 레퍼리인 경우
},
],
// 하위 프로젝트가 검토중인 경우
"status": "CONFIRMING", // 하위 프로젝트 status
"percent": null, // 11/20 업데이트
"isAddReferee": false, // 프로젝트의 isAddReferee 값
"isAddReferee": true,
"candidate": {
"id": 30,
"name": "이강인", // 이름 없으면 관리번호 노출
"group": "개발", // 조직식별자
"email": "example@example.com",
"phone": "01012341234"
"profile": null,
"updatedAt": null,
"status": "WAITING"
"isSelected": true, // 엑셀에서 지정되거나 무작위 지정된 레퍼리인 경우
"isSelected": false // 프로젝트 생성 시 지정되지 않은 레퍼리인 경우
}
]
```

```
"name": "이감인", // 이름 없으면 관리번호 노출
"group": "개발", // 조직식별자
"email": "example@example.com",
"phone": "01012341234",
"profile": "DEFAULT or ISFP or (Base64)", // 10/27 업데이트
"updatedAt": null,
"status": "CONFIRMING"
},
"referees": [
{
"id": 20,
"name": "김병지", // 이름 없으면 관리번호 노출
"relationship": { "key": "SENIOR", "value": "상사" }, // 레퍼리 관계가 지정된 경우
"relationship": null, // 레퍼리 관계가 지정되지 않은 경우
"relationshipIdentifier": { "key": "SENIOR", "value": "상사" } // 직급식별자가
"relationshipIdentifier": null, // 직급식별자가 존재하지 않는 경우
"email": "example@example.com",
"phone": "01012341234",
"profile": null,
"updatedAt": null,
"status": "WAITING",
"isSelected": true, // 엑셀에서 지정되거나 무작위 지정된 레퍼리인 경우
"isSelected": false // 프로젝트 생성 시 지정되지 않은 레퍼리인 경우
},
],
// 하위 프로젝트가 진행중인 경우
"status": "PROCESSING", // 하위 프로젝트 status
"percent": 50, // 11/20 업데이트
"isAddReferee": false, // 프로젝트의 isAddReferee 값
"isAddReferee": true,
"candidate": {
"id": 30,
"name": "이감인", // 이름 없으면 관리번호 노출
"group": "개발", // 조직식별자
"updatedAt": "2023-10-05T14:48:00.000Z", // 대상자의 최근 응답 업데이트 날짜
"email": "example@example.com",
"phone": "01012341234",
"profile": "DEFAULT or ISFP or (Base64)", // 10/27 업데이트
"status": "PROCESSING", // 진행중인 경우
"status": "COMPLETE" // 완료된 경우
},
"referees": [
{
"id": 20,
"name": "김병지", // 이름 없으면 관리번호 노출
"relationship": { "key": "SENIOR", "value": "상사" },
"relationshipIdentifier": { "key": "SENIOR", "value": "상사" } // 직급식별자가
"relationshipIdentifier": null, // 직급식별자가 존재하지 않는 경우
"email": "example@example.com",
"updatedAt": "2023-10-05T14:48:00.000Z", // 레퍼리의 최근 응답 업데이트 날짜
"phone": "01012341234",
"profile": "DEFAULT or ISFP or (Base64)", // 10/27 업데이트
"status": "PROCESSING", // 진행중인 경우
"status": "COMPLETE" // 완료된 경우
"isSelected": true, // 엑셀에서 지정되거나 무작위 지정된 레퍼리인 경우
"isSelected": false // 프로젝트 생성 시 지정되지 않은 레퍼리인 경우
}
]
}
```

## (10/31 업데이트) 인사담당자용: 하위 프로젝트 상태 변경하기(시작/합의 완료/검토 완료)

PUT /sub-project/{id}/status



- 요청 status 값이 START일 때

(10/26 업데이트)

프로젝트 status: WAITING >> PROCESSING

하위 프로젝트 status: WAITING >> ACCEPTING

대상자 status: WAITING >> ACCEPTING

평가자 status: WAITING >> WAITING

프로젝트 내 모든 대상자들에게만 이메일/SMS 알림 즉시 전송

=> 이메일, SMS 값을 확인한 후, 기재된 곳으로 발송

예: 이메일만 기재된 경우 이메일 발송

휴대폰번호만 기재된 경우 SMS 발송

이메일과 휴대폰번호 모두 기재된 경우 이메일, SMS 모두 발송

=> 이메일, SMS 발송 실패해도 api 성공처리함 (프로젝트별로 수동 재전송 기능 있음)

- 요청 status 값이 AGREE일 때

(10/26 업데이트)

하위 프로젝트 status: AGREEING >> CONFIRMING

대상자 status: AGREEING >> CONFIRMING

평가자 status: WAITING >> WAITING

- 하위프로젝트 내 대상자에게 합의 완료되었다고 이메일/SMS 알림 즉시 전송

=> 이메일, SMS 값을 확인한 후, 기재된 곳으로 발송

=> 이메일, SMS 발송 실패해도 api 성공처리함 (프로젝트별로 수동 재전송 기능 있음)

- 요청 status 값이 CONFIRM일 때

(10/26 업데이트)

하위 프로젝트 status: CONFIRMING >> PROCESSING

대상자 status: CONFIRMING >> PROCESSING

평가자 status: WAITING >> PROCESSING

- 하위프로젝트 내 대상자, 레퍼리에게 평가를 요청한다고 이메일/SMS 알림 즉시 전송

=> 이메일, SMS 값을 확인한 후, 기재된 곳으로 발송

=> 이메일, SMS 발송 실패해도 api 성공처리함 (프로젝트별로 수동 재전송 기능 있음)

### Path Parameters

Name	Type	Description
id*	String	하위 프로젝트 id

### Headers

Name	Type	Description

<b>Authorization*</b>	String	Bearer \${AccessToken}
<b>Request Body</b>		
<b>Name</b>	<b>Type</b>	<b>Description</b>
status*	String	(10/31 업데이트) status 요청값 전송
200: OK 성공 :: 401... 404... 400... 400... 403... 400...		
<pre>{   message: "성공했습니다." }</pre>		

## (12/22 업데이트) 인사담당자용, 대상자용: 레퍼리 직급 부여하기

**PUT** /sub-project/{id}/referee-relationship

👉 Select

# relationship

(12/22 업데이트)

=> 대상자: 인사담당자가 엑셀에서 지정하거나, 무작위매칭한 레퍼리 정보는 변경 불가  
이 경우 400 에러 리턴 (클라이언트 요청이 잘못된 경우)  
=> 인사담당자: 모든 정보 레퍼리 직급에 대해 수정 가능

(10/26 업데이트)

사용자별로 직급을 부여할 권한이 없는 단계에서는 변경 불가  
=> 대상자: 하위프로젝트가 AGREEING일 때만 수정 가능  
=> 인사담당자: 하위 프로젝트가 CONFIRMING일 때만 수정 가능  
=> 그 외의 경우 400 에러 리턴 (클라이언트 요청이 잘못된 경우)

(12/22 업데이트)

인사담당자이면서 리뷰어인 경우, 토큰만으로 대상자인지, 인사담당자인지 확인 불가  
=> body에 role 값 추가, 대상자로서 보내는 것이면 CANDIDATE, 인사담당자로서 보내는 경우 EMPLOYER 값 전송

토큰 사용자 == 해당 하위 프로젝트 대상자가 아니면 403 에러 리턴

요청 버튼 클릭 시 자동으로 하위 프로젝트 내 메시지 생성

- 메시지 내용 => PM 작성분 참고:
- 프로젝트 관리자 메일로 전송

### Path Parameters

Name	Type	Description
id*	String	하위 프로젝트 id

## Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

## Request Body

Name	Type	Description
relationships*	Object[]	관계 지정 객체 배열
role*	String	전송 주체 (CANDIDATE 혹은 EMPLOYER)

200: OK 성공	:::	4...	4...	2...	4...	4...	4...	4...	4...
{ message: "성공했습니다." }									

## (12/22 업데이트) 인사담당자, 대상자용: 하위 프로젝트 메세지 생성하기

POST /sub-project/{id}/message

FE

(12/22 업데이트) 합의 기능 사용할 때만 인사담당자, 대상자 사용 가능하도록 화면 제어  
- project 설정에서 isAddReferee: true 인 경우

-----

🌟 Enum

# target

해당 프로젝트의 관리자나 하위 프로젝트의 대상자인 경우만 메시지 생성 가능  
=> 아닐 경우 403 에러 리턴

메시지 작성주체(target), 작성 시간, 메시지 내용 DB에 저장

메시지의 내용을 메일로 발송

- 메일 내용은 PM이 전달 예정

CASE1. target이 EMPLOYER 일 경우

=> 해당 하위프로젝트 대상자의 이메일, SMS 값을 확인한 후, 기재된 곳으로 발송

CASE2. target이 CANDIDATE 일 경우

=> 해당 프로젝트 관리자의 이메일로 발송

기입 정보별 최대 길이 (10/19 업데이트)

- 현 헌터스 기준 최대 데이터 길이 참고함 (Q&A 메시지: 3982자)

- 메시지: 4000자

#### Path Parameters

Name	Type	Description
id*	String	하위 프로젝트 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

#### Request Body

Name	Type	Description
target*	String	메시지 작성 주체
message*	String	메세지

200: OK 성공 :: 401: Unauthorized :: 404: Not Found :: 400: Bad Request :: 403: Forbidden

```
{  
    message: "성공했습니다."  
}
```

## (12/22 업데이트) 하위 프로젝트 메세지 리스트 보여주기

GET /sub-project/{id}/messages

FE

(12/22 업데이트) 합의 기능 사용할 때만 인사담당자, 대상자 사용 가능하도록 화면 제어

- project 설정에서 isAddReferee: true 인 경우

-----

#### 💡 Feature

가장 최근에 작성된 글이 배열 맨 앞에 위치되도록 정렬

#### Path Parameters

Name	Type	Description
id*	String	하위 프로젝트 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

+ :: 200: OK 성공 :: 401: Unauthorized ... :: 404: Not Found id... :: 403: Forbidden 프...

```
{  
  "messages": [  
    // 인사담당자인 경우  
    {  
      "id": 3,  
      "createdAt": "2023-10-05T14:48:00.000Z",  
      "target": "EMPLOYER",  
      "name": null,  
      "content": "댓글 내용입니다."  
    },  
    // 대상자인 경우  
    {  
      "id": 4,  
      "createdAt": "2023-10-05T14:48:00.000Z",  
      "target": "CANDIDATE",  
      "name": "(이름 or 관리번호)",  
      "content": "댓글 내용입니다."  
    },  
  ]  
}
```



Previous  
프로젝트 API

Next  
리뷰어 API



강민

Last modified 2mo ago

# 😊 리뷰어 API

Page description (optional)

## (11/14 업데이트) 리뷰어 정보 불러오기

`GET /sub-project/{id}/reviewer`

(11/14 업데이트)

FE

레퍼리인 경우, profile 값 유무로 동의 여부 확인

### 🌟 Feature

매 하위 프로젝트마다 리뷰어 정보가 다르게 저장됨

하위 프로젝트 정보 & 토큰정보로 리뷰어 본인 맞는지 확인

=> 아닐 경우 403 에러 리턴

엑셀에서 이름은 선택값이기 때문에 없을 시 관리번호를 표시

(11/13 업데이트)

profile은 유형 관계 없이 BE에서 Base64 인코딩 형태로 받아서 렌더링

candidate의 name, profile 추가

- 하위 프로젝트에서, 본인이 평가자일 경우, 후보자 이름 (이름 값이 없을 경우 관리번호), 프로필 사진 값 추가해서 리턴

(11/14 업데이트)

status 추가

=> 후보자(대상자)인 경우 : # candidateStatus 값 출력

=> 레퍼리인 경우: # refereeStatus 값 출력

### Path Parameters

Name	Type	Description
<code>id*</code>	String	하위 프로젝트 id

### Headers

Name	Type	Description
<code>Authorization*</code>	String	Bearer \${AccessToken}

```
{
    "name": "홍길동", // 이름 혹은 관리번호
    "email": "example@example.com",
    "phone": "010-1234-5678",
    "profile": "Base64", // 11/10 업데이트
    "status": "WAITING", // 11/14 업데이트 / 시작 대기 (후보자인 경우: candidateStatus)
    "status": "ACCEPTING", // 11/14 업데이트 / 대상자 동의 요청중 (후보자인 경우: candidateStatus)
    "status": "AGREEING", // 11/14 업데이트 / 합의중 (후보자인 경우: candidateStatus)
    "status": "CONFIRMING", // 11/14 업데이트 / 검토 대기중 (후보자인 경우: candidateStatus)
    "status": "PROCESSING", // 11/14 업데이트 / 진행중 (후보자인 경우: candidateStatus)
    "status": "COMPLETE", // 11/14 업데이트 / 완료 (후보자인 경우: candidateStatus)
    "status": "WAITING", // 11/14 업데이트 / 시작 대기 (레퍼리인 경우: refereeStatus)
    "status": "PROCESSING", // 11/14 업데이트 / 시작 대기 (레퍼리인 경우: refereeStatus)
    "status": "COMPLETE", // 11/14 업데이트 / 완료 (레퍼리인 경우: refereeStatus)
    "candidate": { // 11/13 업데이트, 토큰 보유자가 평가자일 경우에만 값 노출
        "name": "홍길동", // 후보자(대상자) 이름 혹은 관리번호
        "profile": "Base64", // 후보자(대상자) 프로필 사진 값
    }
}
```

## (12/14 업데이트) 리뷰어 동의 및 프로필 사진 등록하기

**POST** /sub-project/{id}/reviewer

FE: 프로필 기본 사진은 FE 서버에 저장된 것 사용

FE: 프로필 사진 최대 사이즈 10MB로 제한

매 하위 프로젝트마다 리뷰어 정보가 다르게 저장됨

하위 프로젝트 정보 & 토큰정보로 리뷰어 본인 맞는지 확인

=> 아닐 경우 403 에러 리턴

프로필 사진 업로드

유형: # profiletype

기본 유형의 경우, 프로필 사진 파일값이 없어 null로 보내지기 때문에 profile 선택값으로 지정

profile 파일값에 들어갈 수 있는 확장자

- tiff, jfif, bmp, gif, svg, png, jpeg, svgz, jpg, webp, ico, xbm, dib, pjp, apng, tif, pjpeg, avif
- 구글 프로필 사진 확장자 제한과 동일하게 함
- 대문자도 있을 수 있으니, 대소문자 구분하지 말 것

(10/26 업데이트)

- 후보자(대상자) 동의 후 상태 변경

프로젝트의 isAddReferee: true 일 때

하위 프로젝트 status: ACCEPTING => AGREEING

대상자 status: ACCEPTING => AGREEING

평가자 status: WAITING => WAITING

프로젝트의 isAddReferee: false 일 때

하위 프로젝트 status: ACCEPTING => CONFIRMING

아워 프로젝트 status: ACCEPTING >> CONFIRMING

대상자 status: ACCEPTING >> CONFIRMING

평가자 status: WAITING >> WAITING

(11/14 업데이트)

- 레퍼리는 동의 후 상태 변경 없음

(11/10 업데이트)

프로필 사진 입력 시 기본이미지와 MBTI 이미지는 FE에 저장된 파일을, 사용자 입력 이미지는 사용자가 업로드한 파일을 multipart로 보냄

- (12/14 업데이트)

이미지가 아닌 파일을 프로필로 업로드하는 경우, 처리 추가

status: 400

message: 파일 형식이 맞지 않습니다.

#### Path Parameters

Name	Type	Description
id*	String	하위 프로젝트 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}
Content-Type*	String	multipart/form-data

#### Request Body

Name	Type	Description
profile*	File	프로필 사진

200: OK 성공 :: 404: Not Found :: 401: Unauthorized :: 403: Forbidden :: 400: Bad Request

```
{  
    message: "성공했습니다."  
}
```

## (12/29 업데이트) 리뷰어 진단지 상세 보여주기

GET /sub-project/{id}/questions

Enum

# relationship

## Feature

토큰정보로 리뷰어 본인 맞는지 확인

=> 아닐 경우 403 에러 리턴

(10/26 업데이트)

리뷰어의 status가 PROCESSING 혹은 COMPLETE일 때 접근 가능

=> 대상자는 # candidateStatus, 레퍼리는 # refereeStatus 확인

=> 아닐 경우 403 에러 리턴

하위 프로젝트 정보 & 토큰의 본인 정보 조합하여 결과값 리턴

=> 결과값: 진단지 내용 및 저장된 응답값

=> 진단지 내용 중 본인의 relationship에 해당하는 문항만 리턴

<질문지 공통 변수 값>

- id: 질문지 id
- order: 순서
- description: 질의문

CASE1: 질문지인 경우

- CASE1-1: 객관식인 경우, 아래 변수 추가됨

- category1: 범주1
- isRequired: 필수/선택여부
- isLongAnswer(주관식/객관식 여부)
- relationship (응답 대상)
- sentences: 객관식 문항 (문항 수 제한 없음)

12/19 업데이트: 문항 id와 description key 만 사용

- score: 질의 대상자가 응답한 내용 (객관식)

- CASE1-2: 주관식인 경우, 아래 변수 추가됨

- category1: 범주1
- isRequired: 필수/선택여부
- isLongAnswer(주관식/객관식 여부)
- relationship (응답 대상)
- answer: 질의 대상자가 응답한 내용 (주관식)

CASE2: 소제목인 경우

- 질문지 공통 변수만 사용, 나머지는 모두 값 없음

- category1: 무조건 "소제목"으로 표기되어야 함

CASE3: 상세설명인 경우

- 질문지 공통 변수만 사용, 나머지는 모두 값 없음

- category1: 무조건 "상세설명"으로 표기되어야 함

(10/31 업데이트)

하위 프로젝트 상세보기와 API주소 변경 겹쳐서 수정

(11/29 업데이트) score는 해당하는 점수 값이 오는 것이 아닌 해당하는 점수의 id 값을 받는 응답으로 수정  
(중복 방지를 위해) 진단지 응답 시에도 해당 score의 id 값을 보내는 요청으로 수정

## Path Parameters

Name	Type	Description
------	------	-------------

<b>id*</b>	String	하위 프로젝트 id
------------	--------	------------

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 404: Not Found :: 401: Unauthorized :: 403: Forbidden :: 403: Forbidden

```
{
  "questions": [
    { // 객관식 혹은 주관식 질문인 경우
      "id": 20, // 개별 진단지 문항 id, 질문 응답 시 전송
      "category1": "관계역량", // 범주1
      "category2": "리더십", // 범주2
      "category3": "방향제시능력", // 범주3
      "order": 1, // 문항 순서
      "description": "공동의 목표 달성을 위해 구성원들에게 나아가야 할 방향을 제시한다.", // 설명
      "isRequired": true, // 필수
      "isRequired": false, // 선택
      "isLongAnswer": true, // 주관식
      "isLongAnswer": false, // 객관식
      "relationship": "SENIOR", // 질의 대상: 상사
      "relationship": "PEER", // 질의 대상: 동료
      "relationship": "JUNIOR", // 질의 대상: 부하
      "relationship": "SELF", // 질의 대상: 본인
      "relationship": "ALL", // 질의 대상: 전체 (본인, 상사, 동료, 부하)
      "sentences": null, // 문항 및 배점, 주관식인 경우 null
      "sentences": [ // 문항 및 배점, 객관식인 경우
        {
          // 문항1
          "id": 32 // 문항 id 값
          "description": "전혀 그렇지 않다",
        },
        {
          // 문항2
          "id": 15 // 문항 id 값
          "description": "그렇지 않다",
        },
        {
          // 문항3
          "id": 24 // 문항 id 값
          "description": "보통이다",
        },
        {
          // 문항4
          "id": 34 // 문항 id 값
          "description": "그렇다",
        },
        {
          // 문항5
          "id": 53 // 문항 id 값
          "description": "자주 그렇다",
        }
      ],
    }
  ]
}
```

```

    "score": 53, // 본인의 저장된 응답값: 객관식인 경우, 점수에 해당하는 id 값
    "answer": "열정적으로 일하는 모습을 높이 평가합니다." // 본인의 저장된 응답값: 주관식
},
{
    // 소제목인 경우
    "id": 21,
    "category1": "소제목",
    "order": 2,
    "description": "리더십 (Leadership) 관련 질문"
},
{
    // 상세설명인 경우
    "id": 22,
    "category1": "상세설명",
    "order": 3,
    "description": "리더십은 구성원이 자발적으로 조직의 활동에 참여하여 이를 달성하도록 유
}
]
}

```

## (11/29 업데이트) 리뷰어 진단지 응답하기

`POST /sub-project/{id}/answers`

🌟 Feature

FE: 필수 응답값 미제출된 것 알려주고, 제출할지 확인하는 알림창 띄울 것

-----  
토큰정보로 리뷰어 본인 맞는지 확인  
=> 아닐 경우 403 에러 리턴

프로젝트가 종료되었는지 확인  
=> 종료된 경우 403 에러 리턴

(10/26 업데이트)  
리뷰어의 status가 PROCESSING 혹은 COMPLETE일 때 접근 가능  
=> 대상자는 # candidateStatus, 레퍼리는 # refereeStatus 확인  
=> 아닐 경우 403 에러 리턴

하위 프로젝트 정보 & 토큰의 본인 정보 조합하여 응답값 저장  
=> 진단지 내용 중 본인의 relationship에 해당하는 응답값만 저장 필요

응답값 중간 저장 가능  
객관식일 때는 score, 주관식일 때는 answer key로 전송

리뷰어가 응답을 저장했을 때만, 본인의 최신 응답 날짜를 저장해야 함  
=> /project/{id}/sub-projects, /sub-project/{id} 의 updateAt 산출을 위해 필요

기입 정보별 최대 길이 (10/19 업데이트)  
- 현 체커 기준 최대 데이터 길이 참고함 (서술형 응답: 2480자)

- answer: 4000자

(11/29 업데이트) score는 해당하는 점수 값이 오는 것이 아닌 해당하는 점수의 id 값을 받는 응답으로 수정  
(중복 방지를 위해) 진단지 응답 시에도 해당 score의 id 값을 보내는 요청으로 수정

#### Path Parameters

Name	Type	Description
id*	String	하위 프로젝트 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

#### Request Body

Name	Type	Description
answers	Object[]	답변

200: OK 성공 :: 204: ... | 404: ... | 401: ... | 4... | 403: ... | 403: ...

```
{  
    message: "성공했습니다."  
}
```



Previous  
하위 프로젝트 API

Next  
진단지 API



강민

Last modified 2mo ago

# ☺ 진단지 API

Page description (optional)

## (10/31 업데이트) 진단지 생성하기

**POST** /questions

(10/27 업데이트)

엑셀 샘플 값:

<https://wecruitcorp.atlassian.net/browse/WCRT-362>

파일명: 엑셀 정보 업로드 예시 > 시트명: 진단지 구성 항목\_ver3 대표님 컨펌 완료 참고

### 💡 Feature

엑셀 파일은 파일, 파일명 별도 저장 필요

- 확장자 제한: xlsx, xls
- 생성 후, 사용자가 엑셀 파일다운로드 가능

프로젝트에 종속된 형태가 아닌 인사담당자에 종속된 형태

=> 인사담당자 탈퇴로직 기재사항 고려 필요: 인사담당자 탈퇴 후, 프로젝트를 다른 관리자에게 공유하면, 진단지는 어떻게 공유되어야 하는가?

기입 정보별 최대 길이

- 현 체커 기준 최대 데이터 길이 참고함 (질문지 이름: 90자)

- 진단지명: 100자

<엑셀 파일 내 값>

- 관리번호: 40자
- 범주1~3: 80자
- 질의문, 문항 1~n: 길이 제한 없음 (longtext)

(10/23 업데이트) 엑셀 유효성 검사 때 파일을 전송하기 때문에 생성 시에 중복으로 전송할 필요 없음, 파일 고유명을 대신 전달해서 유효성 검사 때 보낸 엑셀과 동일한지 확인

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}
Content-Type*	String	multipart/form-data

### Request Body

Name	Type	Description
title*	String	진단지명

title	String	문서상	
excel*	String	파일 고유명	
200: OK 성공 ::	401: Unauthorized 액션	400: Bad Request 클라이언트 오류	
<pre>{   message: "성공했습니다." }</pre>			

## (12/14 업데이트) 진단지 엑셀 유효성 검사

`POST /questions/excel`

(10/27 업데이트)

엑셀 샘플 값:

<https://we recruit corp.atlassian.net/browse/WCRT-362> >

파일명: 엑셀 정보 업로드 예시 > 시트명: 진단지 구성 항목\_ver3 대표님 컨펌 완료 참고

💡 Feature 엑셀 파일 업로드 시 유효성 검사 (BE)

Case 1. 범주1 값이 "소제목" 또는 "상세설명"인 경우

💡 엑셀 파일 항목 중 필수값

관리번호

순서

질의문

💡 에러 메시지 리스트 (#에 이슈 있는 엑셀칸 번호 함께 표기: 예: A4, C3)

12/21 업데이트: 실제 구현된 내용대로 수정함

관리번호 띄어쓰기 불가: "숫자만 입력해 주세요(#)"

관리번호 누락: "값을 입력해 주세요(#)"

관리번호 중복: "중복된 관리번호가 있습니다(#)"

순서 띄어쓰기 불가: "순서에 있는 띄어쓰기를 지워 주세요(#)"

순서 누락: "순서를 입력해 주세요(#)"

순서 중복: "중복된 순서가 있습니다(#)"

질의문 누락: "질의문을 입력해 주세요 (#)"

관리번호 띄어쓰기 불가: "관리번호에 있는 띄어쓰기를 지워 주세요 (#, #, #)"

관리번호 누락: "관리번호를 입력해주세요 (#, #, #)"

관리번호 중복: "중복된 관리번호가 있습니다 (#, #, #)"

순서 띄어쓰기 불가: "순서에 있는 띄어쓰기를 지워 주세요 (#, #, #)"

순서 누락: "순서를 입력해 주세요 (#, #, #)"

순서 중복: "중복된 순서가 있습니다 (#, #, #)"

질의문 누락: "질의문을 입력해 주세요 (#, #, #)"

Case 2. 범주1 값이 "소제목" 또는 "상세설명"이 아닌 경우

💡 엑셀 파일 항목 중 필수값

관리번호

순서

질의문

답변형식

필수/선택  
질의대상

💡 에러 메시지 리스트 (#에 이슈 있는 엑셀칸 번호 함께 표기: 예: A4, C3)  
조정방법 미기재 가능

12/21 업데이트: 실제 구현된 내용대로 수정함

관리번호 띄어쓰기 불가: "숫자만 입력해 주세요(#)"  
관리번호 누락: "값을 입력해 주세요(#)"  
관리번호 중복: "중복된 관리번호가 있습니다(#)"  
순서 띄어쓰기 불가: "순서에 있는 띄어쓰기를 지워 주세요(#)"  
순서 누락: "순서를 입력해 주세요(#)"  
순서 중복: "중복된 순서가 있습니다(#)"  
질의문 누락: "질의문을 입력해 주세요(#)"  
역배점 값이 "0"나 빈 칸이 아닌 경우: "역배점 값을 '0' 또는 빈칸으로 입력해 주세요(#)"  
답변형식 값이 "객관식", "주관식"이 아닌 경우: "답변형식 값을 '객관식' 또는 '주관식'으로 입력해 주세 (#)"

관리번호 띄어쓰기 불가: "관리번호에 있는 띄어쓰기를 지워 주세요 (#, #, #)"  
관리번호 누락: "관리번호를 입력해 주세요 (#, #, #)"  
관리번호 중복: "중복된 관리번호가 있습니다 (#, #, #)"  
순서 띄어쓰기 불가: "순서에 있는 띄어쓰기를 지워 주세요 (#, #, #)"  
순서 누락: "순서를 입력해 주세요 (#, #, #)"  
순서 중복: "중복된 순서가 있습니다 (#, #, #)"  
순서 숫자 형식이 아닌 경우: "순서에 숫자만 입력해 주세요 (#, #, #)"  
질의문 누락: "질의문을 입력해 주세요 (#, #, #)"  
역배점 값이 "0"나 빈 칸이 아닌 경우: "역배점 값을 '0' 또는 빈칸으로 입력해 주세요 (#, #)"  
답변형식 값이 "객관식", "주관식"이 아닌 경우: "답변형식 값을 '객관식' 또는 '주관식'으로 입력해 주세요 (#, #)"  
답변형식 누락: "답변형식을 입력해 주세요 (#, #, #)"  
답변형식이 "객관식"일 때, 문항 데이터가 2개 이상 채워지지 않은 경우: "문항 데이터를 2개 이상 입력해 주세요 (#, #, #)"  
필수/선택 값이 "필수", "선택"이 아닌 경우: "필수/선택 값을 '필수' 또는 '선택'으로 입력해 주세요 (#, #)"  
필수/선택 누락: "필수/선택을 입력해 주세요 (#, #, #)"  
질의대상 값이 "본인", "상사", "동료", "ALL"이 아닌 경우: "질의대상 값을 본인/상사/동료/ALL 중 하나의 값으로 입력해 주세요 (#, #)"  
질의대상 누락: "질의대상을 입력해 주세요 (#, #, #)"  
가중치 형식(n:n:n:n)대로 기입되지 않은 경우: "가중치를 1:1:1:1과 같이 숫자 4개의 비율로 입력해 주세요 (#, #, #)"  
조정방법 형식대로 기입되지 않은 경우: "조정방법을 M/평균값+표준편차값/#숫자~#숫자/#숫자~/~#숫자 중 하나의 형식으로 입력해 주세요 (#, #, #)"

### 💡 유의사항

답변형식이 "객관식"이고 질의대상이 "ALL"인 경우에만 가중치 값을 확인하여 적용

답변형식이 "객관식"인 경우에만 조정방법 값을 확인하여 적용

가중치

- 본인:상사:동료:부하 비율
- 객관식 & 질의대상이 "ALL" => 가중치 미입력시 기본값(0:1:1:1)으로 설정
- 가중치에는 0 설정 가능
  - => 가중치, 조정값에서 제외하고 싶을 경우 0으로 설정됨

조정방법 계산식은 CTO님이 작성한 다면평가 프로세스 파워포인트 확인

- 조정할 수 없는 경우, 미시행
  - => 예: 조정 방식이 기재되어 있지 않음 (엑셀에서 빈칸), 해당 문항 모두 조정하지 않음
  - => 예: 가중치에서 0:1:1:1로 설정되면, 대상자는 점수 조정 대상에서 제외됨
  - => 예: 평균+표준편차 (M+S) 설정인 경우, A라는 평가자가 피평가자 1명만 있는 경우,  
△ 평가자의 유통값은 조정하지 않으 (조정시에는 보통 0이거나 계산 불가)

### 🌟 유효성 검사 후 메시지 리턴

통과 성공시: 성공 메세지 응답

통과 실패시: 유효성 검사에서 나온 모든 에러 모아서 한번에 출력

BE에서 에러 메시지 배열로 리턴 가능한지 확인 필요

(10/23 업데이트) 엑셀 유효성 검사 때 파일을 전송하기 때문에 생성 시에 중복으로 전송할 필요 없음, 파일 고유명을 대신 전달해서 유효성 검사 때 보낸 엑셀과 동일한지 확인

- (12/14 업데이트)

엑셀이 아닌 파일을 업로드하는 경우, 처리 추가

status: 400

message: 파일 형식이 맞지 않습니다.

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}
Content-Type*	String	multipart/form-data

### Request Body

Name	Type	Description
excel*	Binary	엑셀 파일 (관리번호, 역배점, 범주1, 범주2, 범주3, 순서, 질의문, 답변형식, 필수/선택, 질의대상, 가중치, 조정방법, 문항1, 문항2..., 문항n)

200: OK 성공    ::    401: Unauthorized    400: Bad Request    403: Forbidden    400: Bad Request

```
{
  excel: "(파일 고유명)"
}
```

## (12/15 업데이트) 업로드한 엑셀 다운로드

GET /questions/{id}/excel

### 🌟 Feature

진단지 수정 시 기존에 업로드한 엑셀을 다운로드하는 기능

- (12/14 업데이트) content-type에 파일 타입 명시.

예:

pdf → Content-Type="application/pdf"  
excel → Content-Type="application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"

- (12/15 업데이트) 한글 파일명 사용할 수 있도록 헤더 인코딩 추가

#### Path Parameters

Name	Type	Description
id*	String	진단지 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 404: Not Found id에 ... :: 401: Unauthorized 액... :: 403: Forbidden ...

Access-Control-Expose-Headers: Content-Disposition  
Content-Type: {파일 타입} // 12/14 업데이트  
Content-Disposition: attachment; filename=(파일명).xlsx // 12/14 업데이트

{  
  data: (파일 데이터)  
}

## 진단지 삭제하기

**DELETE** /questions/{id}

진단지 삭제 시, 기존의 프로젝트와 연결되어 있다면,  
해당 프로젝트 운영에 지장이 없도록 설계 필요

#### Path Parameters

Name	Type	Description
id*	String	진단지 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 404: Not Found id에 ... :: 401: Unauthorized 액... :: 403: Forbidden ...

```
{  
    message: "성공했습니다."  
}
```

## (11/13 업데이트) 진단지 리스트 보여주기

GET /questions-list

💡 Filter&Sort

# questionListOrder

# questionListPrice

💡 Initial Query

keyword: ""

order: "NEW"

price: "ALL"

page: "0"

💡 Feature (요청)

keyword

- 빈 값일 경우, 전체 리스트를 보여줌 (기본값)

- keyword 값 검색 범위: 진단지명

=> keyword를 일부라도 포함하는 경우 검색 결과로 리턴

price

- 진단지의 유료화여부

=> 인사담당자 본인이 만든 진단지는 무료로 고려함

- 빈 값 없음

page값은 0부터 시작하며 기본 0임

한 페이지에 보이는 리스트 개수: 10개로 고정

💡 Feature (응답)

응답 데이터가 없는 경우 안 보내는 것이 아닌 null로 보내도록 통일

(11/13 업데이트)

해당 사용자와 관계된 질문지 정보를 모두 보여줌 (questionsList, pagination 값)

- 기본 조건: 로그인한 정보 (이메일 값 혹은 휴대폰값) 기준, 본인이 생성한 질문지 정보를 보여줌

- 기본 조건 + query 에 모두 and 조건으로 맞아야 함 (단, 삭제된 진단지 제외)

price 값에 따른 조건

- ALL: 인사담당자가 생성한 진단지와 위크루트에서 제공하는 진단지 전체 갯수,

- PAY: 위크루트에서 제공하는 진단지 중 유료 진단지 갯수,

- FREE: 인사담당자가 생성한 진단지와 위크루트에서 제공하는 무료 진단지 갯수를 의미

### # answerType

=> 소제목, 상세설명 제외한 질문들의 유형

questionsLength 값은 총 문항 수 (소제목, 상세설명 미포함)

-----  
(11/8 업데이트)

### # price

=> 진단지 유료/무료 특성

## Query Parameters

Name	Type	Description
keyword*	String	검색
order*	String	정렬
price*	String	가격
page*	String	페이지 수

## Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 400: Bad Request :: 401: Unauthorized 액세스 토큰 오류 :: 403: Forbidden 인증 실패

```
{  
    "questionsList": [  
        {  
            "id": 20, // 질문지 id  
            "title": "2023년 하반기 질문지",  
            "createdAt": "2023-10-05T14:48:00.000Z", // 질문지 생성일자  
            "answerType": "ALL", // 진단지에 객관식과 주관식이 모두 섞인 경우  
            "answerType": "SHORT", // 진단지에 객관식만 있는 경우  
            "answerType": "LONG", // 진단지에 주관식만 있는 경우  
            "questionsLength": 30, // 소제목, 상세설명 제외한 객관식, 주관식 총 문항수  
            "price": "PAY", // (11/8 업데이트) 진단지 무료 / 유료 여부  
        },  
        ],  
        "pagination": { // query의 모든 값과 일치되는 진단지  
            "totalElements": 50, // 11/13 업데이트  
            "totalPages": 25,  
            "currentPage": 1  
        }  
    }  
}
```

## (12/04 업데이트) 진단지 상세 보여주기

GET /questions/{id}

## 💡 Enum

### # relationship

## 💡 Feature (응답)

(12/04 업데이트) 엑셀에서 기재한 순서의 오름차순으로 리턴 (예: 순서 1-> 100)

토큰정보로 진단지를 생성한 인담이 맞는지 확인

=> 아닐 경우 403 에러 리턴

<질문지 공통 변수 값>

- id: 질문지 id
- order: 순서
- description: 질의문

CASE1: 질문지인 경우

- CASE1-1: 객관식인 경우, 아래 변수 추가됨
  - category1~3: 범주
  - isRequired: 필수/선택여부
  - isLongAnswer: 주관식/객관식 여부
  - relationship: 응답 대상
  - weight: 가중치
    - => relationship 값이 ALL인 경우 필요, 기재된 값 없으면 기본값 (0:1:1:1) 사용
  - adjustment: 조정방법
  - sentences: 객관식 문항 및 배점 (문항 수 제한 없음)
    - \*\*\* 역배점에 해당하면, score가 역산하여 노출됨
- CASE1-2: 주관식인 경우, 아래 변수 추가됨
  - category1~3: 범주
  - isRequired: 필수/선택여부
  - isLongAnswer(주관식/객관식 여부)
  - relationship (응답 대상)

CASE2: 소제목인 경우

- 질문지 공통 변수만 사용, 나머지는 모두 값 없음
- category1: 무조건 "소제목"으로 표기되어야 함

CASE3: 상세설명인 경우

- 질문지 공통 변수만 사용, 나머지는 모두 값 없음
- category1: 무조건 "상세설명"으로 표기되어야 함

## 💡 리뷰어 진단지 상세보기와 다른점

리뷰어 관계 값에 따라 응답이 변경되지 않음

리뷰어의 응답한 값인 answer, score 값이 없음

isReversePoints, weight, adjustment 값 받음

(11/29 업데이트) score는 해당하는 점수 값이 오는 것이 아닌 해당하는 점수의 id 값을 받는 응답으로 수정  
(중복 방지를 위해) 진단지 응답 시에도 해당 score의 id 값을 보내는 요청으로 수정

## Path Parameters

Name	Type	Description
id*	String	진단지 id

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 401: Unauthorized ... 404: Not Found 리... 403: Forbidden 진단...

```
{
  "title": "2023 하반기 질문지", // 10/31 업데이트: 진단지 제목
  "questions": [
    { // 객관식 혹은 주관식 질문인 경우
      "id": 20, // 개별 진단지 문항 id, 질문 응답 시 전송
      "category1": "관계역량", // 범주1
      "category2": "리더십", // 범주2
      "category3": "방향제시능력", // 범주3
      "isReversePoints": true, // 역배점인 경우
      "isReversePoints": false, // 역배점이 아닌 경우
      "order": 1, // 문항 순서
      "description": "공동의 목표 달성을 위해 구성원들에게 나아가야 할 방향을 제시한다.", // 설명
      "isLongAnswer": true, // 주관식
      "isLongAnswer": false, // 객관식
      "isRequired": true, // 필수
      "isRequired": false, // 선택
      "relationship": "SENIOR", // 질의 대상: 상사
      "relationship": "PEER", // 질의 대상: 동료
      "relationship": "JUNIOR", // 질의 대상: 부하
      "relationship": "SELF", // 질의 대상: 본인
      "relationship": "ALL", // 질의 대상: 전체 (본인, 상사, 동료, 부하)
      "weight": "1:2:1:1", // 가중치
      "adjustment": null, // 조정하지 않는 경우
      "adjustment": "M", // 평균일치법인 경우
      "adjustment": "70+20", // 평균 표준편차 일치법인 경우, 평균 + 표준편차값 기재
      "adjustment": "#10~#10", // 특이점수 제외법: 하위, 상위 n% 배제
      "adjustment": "#10~", // 특이점수 제외법: 하위 n% 배제
      "adjustment": "~#10", // 특이점수 제외법: 상위 n% 배제
      "sentences": null, // 문항 및 배점, 주관식인 경우 null
      "sentences": [ // 문항 및 배점, 객관식인 경우
        {
          // 문항1
          "description": "전혀 그렇지 않다",
          "id": 24 // 점수에 해당하는 id 값
        },
        {
          // 문항2
          "description": "그렇지 않다",
          "id": 25 // 점수에 해당하는 id 값
        },
        {
          // 문항3
          "description": "보통이다",
          "id": 26 // 점수에 해당하는 id 값
        }
      ]
    }
  ]
}
```

```

    {
        // 문항4
        "description": "그렇다",
        "id": 27 // 점수에 해당하는 id 값
    }
    {
        // 문항5
        "description": "자주 그렇다",
        "id": 28 // 점수에 해당하는 id 값
    }
]
},
{
    // 소제목인 경우
    "id": 21,
    "category1": "소제목",
    "order": 2,
    "description": "리더십 (Leadership) 관련 질문"
},
{
    // 상세설명인 경우
    "id": 22,
    "category1": "상세설명",
    "order": 3,
    "description": "리더십은 구성원이 자발적으로 조직의 활동에 참여하여 이를 달성하도록 유
}
]
}

```

## (10/31 업데이트) 프로젝트 생성 시 필요한 진단지 데이터 불러오기

`GET /questions-list-select`

🌟 Feature

токен 사용자가 작성한 진단지 데이터 전체 불러오기

- 삭제된 진단지 제외하고 로드함
- 진단지 삭제 시 상태만 바꿔서 보유할 수 있어 상태 확인 필요

### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

200: OK 성공 :: 401: Unauthorized 액세스 토큰이 ... :: 403: Forbidden 진단지를 생성한 ...

```

{
    "questions": [
    {
        "id": 1,
        "title": "2023년 하반기 질문지", // 10/31 업데이트
    },
    {
        "id": 2,
    }
]
}

```

```
"title": "2024년 하반기 질문지", // 10/31 업데이트
```

```
    }  
]  
}
```



Previous  
리뷰어 API

Next  
**Select**



감민

Last modified 7h ago



## ☺ Select

Page description (optional)

### (10/27 업데이트) 선택지 리스트 불러오기

`GET /select`

↳ **Select**와 ↳ **Filter&Sort**에 작성된 데이터 참조

type 값은 위의 작성된 데이터의 제목

#### Query Parameters

Name	Type	Description
type*	String	선택지명

200: OK 선택지 리스트 출력

400: Bad Request 해당되는 선택지가 없을 ...

```
{
    // Select, Filter&Sort에 작성된 제목명
    "relationship": [
        { "key": "SENIOR", "value": "상사" }, // Select, Filter&Sort에 작성된 key, value
        { "key": "PEER", "value": "동료" },
        { "key": "JUNIOR", "value": "후배" }
    ]
}
```

## scale

인사담당자의 회사 규모

Key	Value
XS	50인 미만
S	50인 이상 100인 미만
M	100인 이상 300인 미만
L	300인 이상 500인 미만
XL	500인 이상

## relationship

Key	Value
SENIOR	상사
PEER	동료
JUNIOR	후배



Previous  
진단지 API

Next  
**Filter&Sort**



Last modified 2mo ago



## 😊 Filter&Sort

⋮

Page description (optional)

### projectListOrder

Key	Value
NEW	최신순
PROCESSUP	진행률 높은순
PROCESSDOWN	진행률 낮은순

### projectListStatus

Key	Value
PROCESSING	진행중
CLOSED	종료

### subProjectListOrder

Key	Value
NEW	최신순
PROCESSUP	진행률 높은순
PROCESSDOWN	진행률 낮은순

### subProjectListStatus

Key	Value
ALL	전체
COMPLETE	완료
INCOMPLETE	미완료



### questionListOrder

Key	Value
NEW	최신순
TITLE	제목순

## questionListPrice

Key	Value
ALL	전체
PAY	유료
FREE	무료

←

Previous  
Select

Next  
Enum

→

강민

Last modified 7h ago



## Enum

Page description (optional)

:

### userType

Key	Description
EMPLOYER	인사담당자(=프로젝트 관리자)로 회원
REVIEWER	대상자 혹은 레퍼리로 임시 사용자

- 특징

- EMPLOYER 내에서 이메일, 휴대폰번호 모두 고유값임
- REVIEWER 내에서 이메일, 휴대폰번호 모두 고유값임
- 이메일 혹은 휴대폰 번호 정보 기준으로, EMPLOYER면서 REVIEWER일 수 있음
  - 이 경우, 토큰에 EMPLOYER로 저장

### userStatus

Key	Description
WAITING	승인 대기
ACTIVE	사용 가능
REJECT	승인 거절

- EMPLOYER 상태 구분: WAITING (승인 대기), ACTIVE (사용 가능), REJECT (승인 거절)

### projectStatus

Key	Description
WAITING	시작 대기
PROCESSING	진행 중
CLOSED	종료

### subProjectStatus

Key	Description

WAITING	시작 대기
ACCEPTING	대상자 동의 요청중
AGREEING	합의중
CONFIRMING	검토중
PROCESSING	진행중

## candidateStatus

Key	Description
WAITING	시작 대기
ACCEPTING	대상자 동의 요청중
AGREEING	합의중
CONFIRMING	검토 대기중
PROCESSING	진행중
COMPLETE	완료

## refereeStatus

Key	Description
WAITING	시작 대기
PROCESSING	진행중
COMPLETE	완료

## relationship

Key	Description
ALL	전체인 경우
SELF	본인인 경우
SENIOR	상사인 경우
PEER	동료인 경우
JUNIOR	후배인 경우

## price

Key	Value
-----	-------

PAY	유료
FREE	무료

## answerType

Key	Description
ALL	객관식 + 주관식인 경우
LONG	주관식인 경우
SHORT	객관식인 경우

## Heading 1

←	Previous Filter&Sort
---	-------------------------

Next <b>API용 Enum</b>	→
--------------------------	---

강민

Last modified 2mo ago

## 😊 API용 Enum

⋮

DB에 저장되는 값은 아니지만, 요청을 구분하기 위한 키값

### projectStatusRequest

Key	Description
START	프로젝트에 대한 시작 요청
AGREE	프로젝트 합의 단계에서 프로젝트를 승인함
CONFIRM	프로젝트 검토 단계에서 검토를 완료함
END	프로젝트를 종료함



Previous  
Enum

Next  
레포트 API



강민

Last modified 7h ago



## ⑥ 레포트 API

대상자용 레포트 출력을 위한 API입니다.

### (24/06/07 업데이트)인사담당자: 대상자 PDF 보고서 다운로드 요청

`GET /sub-project/{id}/report`

🌟 Feature

프로젝트 종료 후, 개별대상자의 PDF 보고서 다운로드하는 기능

<파일명>

대상자명 혹은 관리번호.pdf

\*\* 대상자명 있을 경우: 대상자명

없을 경우 관리번호

(24/06/07 업데이트)

Query의 simple 값 true일 경우 마스킹 보고서 다운로드 요청

Query의 simple 값 빈 값, false일 경우 일반 보고서 다운로드 요청

#### Path Parameters

Name	Type	Description
id*	String	하위프로젝트 id

#### Query

Name	Type	Description
simple*	boolean	보고서 타입

#### Headers

Name	Type	Description
Authorization*	String	Bearer \${AccessToken}

#### Response

200 :: 404 | 401 | 403

/\* Response header

```
Access-Control-Expose-Headers: Content-Disposition
Content-Type: application/pdf
Content-Disposition: attachment; filename=대상자명 혹은 관리번호.pdf

*/
{
    data: (파일 데이터)
}
```

## BE 서버: PDF 보고서 생성 요청

POST checker360-report.weclruitpro.com /report

### 💡 Feature

- PDF 서버에 개별 레포트 생성 요청
  - 인사담당자 정보는 프로젝트에 가장 먼저 공유된 1명의 정보 노출
    - 대상자 메일 템플릿에 기재되는 인사담당자 정보와 동일한 로직
  - 객관식평가 점수는 소수점 2째자리까지 산출, 셋째자리에서 반올림
    - 차이를 나타내는 값은 0을 제외하고 무조건 기호를 포함한 string 형식으로 출력 (+ 혹은 -)
  - 질문지별로 body 값이 달라짐
    - 위크루트에서 제공하는 질문지를 사용하는 경우
      - competency, workStyle, collaboration, discs 키의 값을 전송
    - 위크루트에서 제공하는 질문지를 사용하지 않는 경우
      - competency, workStyle, collaboration, discs 를 null값으로 전송
- 레포트 서버로의 요청 제한: BE 서버에서만 요청 가능하게 함
  - 다른 곳에서 요청할 경우 에러 리턴 (하단 Response 참고)

### Body

```
// 위크루트에서 제공하는 질문지를 사용하는 경우
{
    "reportedAt": "2024/04/01" // p.1 보고일 = 보고서 생성 요청일
    "employer": {
        "name": "최인사" // p.2 프로젝트 정보 > 담당자 = 프로젝트에 가장 먼저 공유된 인사담당자 1명
        "email": "example@samsung.com" // p.2 프로젝트 정보 > E-Mail = 프로젝트에 가장 먼저 공유된 대상자
        "companyName": "삼성전자" // p.1 고객사
    },
    "project": {
        "name": "2024 상반기 조직 진단" // p.2 프로젝트 정보 > 평가 그룹 = 프로젝트명
        "closedAt": "2024/03/30" // p.2 프로젝트 정보 > 마감기한 = 루프로젝트 종료 버튼 누른 날짜
        "questionsName": "기본 질문지" // p.2 프로젝트 정보 > 평가 질문지 = 질문지명
    },
    "candidate": {
        "name": "홍길동" // p.1 성명, p.2 대상자 = 이름 없을 경우 관리번호 노출
        "email": "candidate@samsung.com" // p.2 대상자 > E-Mail = 이메일 없을 경우 - 로 표시
    },
    "reviewer": {} // p.2 평가 현황, 대상자 + 레퍼리 수, 대상자 수, 상사의 수, 동료의 수, 후배의 수, 등급별 평가 현황, 평가 척도, 평가 척도 등
}
```

```

    "assigns": [10, 1, 3, 2, 4], // p.2 평가 만족 > 시상 = 인사급증시의 금도 필요 후, 쇠장 = 
    "answers": [8, 1, 2, 2, 3] // p.2 평가 현황 > 응답 = 프로젝트 종료 후, 질문지필수 응답 도
},
"competency": { // [워크루트 전용 질문지용] p. 3, 4, 7의 역량으로 점수 표시, 질문지 관리번호
    // 순서: 전체 역량 평균, 열정, 성실성, 전문지식, 기획창의력, 이해판단력, 문제해결능력, 커뮤니
    "candidates": [2.66, 3.00, 3.00, 3.00, 2.00, 2.50, 3.00, 3.00, 3.00 ] // 본인
    "referees": [2.50, 3.10, 3.20, 3.00, 2.50, 2.50, 3.10, 2.83, 2.75 ] // 타인 평균 (타인 평균)
    "refereeDiffs": ["-0.16", "+0.10", "+0.20", "0", "+0.50", "0", "+0.10", "-0.17",
    "seniors": [2.66, 3.00, 3.00, 3.00, 2.00, 2.50, 3.00, 3.00, 3.00 ] // 상사 평균
    "peers": [2.66, 3.00, 3.00, 3.00, 2.00, 2.50, 3.00, 3.00, 3.00 ] // 동료 평균
    "juniors": [2.66, 3.00, 3.00, 3.00, 2.00, 2.50, 3.00, 3.00, 3.00 ] // 부하 평균
    "projectReferees": [2.80, 3.00, 3.00, 2.80, 2.60, 3.00, 2.80, 2.60, 3.00] // 조직
    "projectRefereeDiffs": ["+0.14", "0", "0", "-0.20", "+0.60", "+0.50", "-0.20", "-0.14",
    "projectCandidate": 2.80 // 프로젝트 내 전체 대상자의 전체 역량 평균
    "scatters": [ // scatter 차트용, 대상자 당 전체 역량의 본인 응답 평균, 전체 역량의 본인의
        // 레포트 출력하는 대상자 값의 candidate, referees 값만 제외하고 전달 필요
        // 예: "홍길동" 대상자의 레포트라면, 홍길동 대상자를 제외한 다른 대상자의 candidate, referee
        { "candidate": 3.00, "referees": 2.80 },
        { "candidate": 3.20, "referees": 3.50 },
        { "candidate": 3.30, "referees": 2.90 },
    ]
},
"workStyle": { // [워크루트 전용 질문지용] p.3 의 리더십으로 명수 표시
    // 순서: 대상자 수, 상사 수, 동료 수, 부하 수
    "passions": [1, 3, 3, 3], // 열정조직통합
    "creativities": [0, 3, 3, 3] // 창의형
    "initiatives": [0, 2, 3, 4], // 도전/추진형
    "intellects": [0, 0, 3, 0], // 지식형
    "diligences": [0, 0, 0, 0], // 근면형
    "obediences": [0, 0, 0, 0], // 순종형
},
"collaboration": { // [워크루트 전용 질문지용] p.3 의 협업 역량으로 명수 표시
    // 순서: 대상자 수, 상사 수, 동료 수, 부하 수
    "highers": [1, 3, 3, 3], // 매우 높음
    "highs": [0, 2, 3, 2], // 높음
    "middles": [0, 1, 2, 1] // 보통
    "lows": [0, 0, 1, 0] // 낮음
    "lowers": [0, 1, 0, 2] // 매우 낮음
},
"discs": [ // [워크루트 전용 질문지용] p.3 의 행동 유형으로 명수 표시, 평균은 해당하는 값의 범위
    // 2차원 배열, [외향적:내향적 값 역순 계산] [일:관계 값 역순 계산]
    // 예) [0][0] = [외향적:내향적 6번 문항 (80:20) 선택] [일:관계 6번 문항 (80:20) 선택]
    // 순서: 본인, 상사, 동료, 부하, 평균
    // 예) [0][0] = [0, 1, 0, 0, 0] => 상사 1명만 외향적:내향적 6번 문항 (80:20), 일:관계 6
    [[0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0],
    [[0, 0, 0, 0, 0], [1, 1, 2, 1, 0], [0, 0, 1, 1, 1], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0],
    [[0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0],
    [[0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0],
    [[0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0],
    [[0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0],
    [[0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0], [0, 0, 0, 0, 0],
    ],
"shortAnswers": [ // 전체 객관식 문항별 결과값, 질문지의 순서 오름차순으로 배열 ( 3 → 4 → 5
{
    // 상사, 동료, 부하의 경우 1~5점별로 값 부여한 인원수 표시
    "description": "열정",
    "seniors": [0, 2, 1, 0, 1],
    "peers": [1, 0, 0, 1, 0],
    "juniors": [0, 0, 2, 1, 0],
    "candidate": 3.00, // 본인
    "referee": 2.66, // 타인 평균 (해당 대상자의 레퍼리평균)
    "refereeDiff": "+0.34", // 타인 평균 - 본인
    "projectReferee": 2.00 // 조직 평균 (프로젝트 내 전체레퍼리 평균)
}

```

```

        "projectReferee": 2.00, // 조직 평균 (프로젝트 내 전체레퍼리 평균)
        "projectRefereeDiff": "-1.00" // 조직 평균 - 본인
    },
],
"longAnswers": [ // 전체 주관식 문항별 결과값, 질문지의 순서 오름차순으로 배열 ( 3 → 4 → 5 . .
{
    "description": "(Continue) 대상자가 계속 유지했으면 하는 행동은 무엇인가요? 그리고, .
    "answers": [ // 본인, 레퍼리1, 레퍼리2, 레퍼리 3 순으로 출력 (레퍼리는 등록 먼저한 순
        "A행동을 유지했으면 좋겠습니다.", .
        "B행동을 유지했으면 좋겠습니다.", .
        "C행동을 유지했으면 좋겠습니다.", .
    ]
}
}

// 위크루트에서 제공하는 질문지를 사용하지 않는 경우
{
    "reportedAt": "2024/04/01" // p.1 보고일 = 보고서 생성 요청일
    "employer": {
        "name": "최인사" // p.2 프로젝트 정보 > 담당자 = 프로젝트에 가장 먼저 공유된 인사담당자 1명!
        "email": "example@samsung.com" // p.2 프로젝트 정보 > E-Mail = 프로젝트에 가장 먼저 공유된 이메일
        "companyName": "삼성전자" // p.1 고객사
    },
    "project": {
        "name": "2024 상반기 조직 진단" // p.2 프로젝트 정보 > 평가 그룹 = 프로젝트명
        "closedAt": "2024/03/30" // p.2 프로젝트 정보 > 마감기한 = 루프로젝트 종료 버튼 누른 날짜
        "questionsName": "기본 질문지" // p.2 프로젝트 정보 > 평가 질문지 = 질문지명
    },
    "candidate": {
        "name": "홍길동" // p.1 성명, p.2 대상자 = 이름 없을 경우 관리번호 노출
        "email": "candidate@samsung.com" // p.2 대상자 > E-Mail = 이메일 없을 경우 - 로 표시
    },
    "reviewer": { // p.2 평가 현황, 대상자 + 레퍼리 수, 대상자 수, 상사의 수, 동료의 수, 후배의 수
        "assigns": [10, 1, 3, 2, 4], // p.2 평가 현황 > 지정 = 인사담당자의 검토 완료 후, 최종 평가
        "answers": [8, 1, 2, 2, 3] // p.2 평가 현황 > 응답 = 프로젝트 종료 후, 질문지필수 응답 도록
    },
    "competency": null,
    "workStyle": null,
    "collaboration": null,
    "discs": null,
    "shortAnswers": [ // 전체 객관식 문항별 결과값, 질문지의 순서 오름차순으로 배열 ( 3 → 4 → 5 . .
{
        // 상사, 동료, 부하의 경우 1~5점별로 값 부여한 인원수 표시
        "description": "열점",
        "seniors": [0, 2, 1, 0, 1],
        "peers": [1, 0, 0, 1, 0],
        "juniors": [0, 0, 2, 1, 0],
        "candidate": 3.00, // 본인
        "referee": 2.66, // 타인 평균 (해당 대상자의 레퍼리평균)
        "refereeDiff": "+0.34", // 타인 평균 - 본인
        "projectReferee": 2.00, // 조직 평균 (프로젝트 내 전체레퍼리 평균)
        "projectRefereeDiff": "-1.00" // 조직 평균 - 본인
    },
],
"longAnswers": [ // 전체 주관식 문항별 결과값, 질문지의 순서 오름차순으로 배열 ( 3 → 4 → 5 . .
{
    "description": "(Continue) 대상자가 계속 유지했으면 하는 행동은 무엇인가요? 그리고, .
    "answers": [ // 본인, 레퍼리1, 레퍼리2, 레퍼리 3 순으로 출력 (레퍼리는 등록 먼저한 순
        "A행동을 유지했으면 좋겠습니다.", .
        "B행동을 유지했으면 좋겠습니다.", .
        "C행동을 유지했으면 좋겠습니다.", .
    ]
}
}

```

Response

200 :: 400 403

```
/* Response header

Access-Control-Expose-Headers: Content-Disposition
Content-Type: application/pdf
Content-Disposition: attachment; filename=대상자명 혹은 관리번호.pdf

*/
{
    data: (파일 데이터)
}
```



Previous  
API용 Enum

강민

Last modified 7h ago