

API 문서

1. 유저 관련

- [회원가입](#)
- [로그인](#)
- [로그인 정보 가져오기\(로그인시\)](#)
- [이메일 확인](#)
- [비밀번호 변경\(token이 있을 경우\)](#)
- [비밀번호 변경\(로그인 안했을 시\)](#)
- [휴대폰 인증\(sms 발송, 이메일 찾기, 번호변경\)](#)

- [휴대폰 인증 확인\(코드 확인\)](#)
- [회원탈퇴](#)
- [디바이스 토큰 등록\(push 관련\)](#)

2. 원 관련

- [원 생성](#)
- [원 조회\(검색\)](#)
- [원 수정](#)
- [원 삭제](#)
- [기관검색으로 생성된 원 체크](#)

- [반 생성](#)
- [반 조회](#)
- [반 조회\(member_id 없이\)](#)
- [반 수정](#)
- [반 삭제](#)

- [초대장 생성](#)
- [초대장 수락](#)

- [요청장 생성](#)
- [요청장 수락](#)
- [초대장 조회, 요청장 조회\(기관\)](#)

- [원아 생성](#)(삭제 예정)
- [원아 조회](#)

- [휴무일 생성](#)
- [휴무일 조회](#)
- [휴무일 삭제](#)

- [스쿨멤버 조회](#)
- [스쿨멤버 상세조회](#)
- [스쿨멤버 정보 변경](#)

3. 공지 관련

- [공지 생성](#)
- [공지 리스트 요청](#)
- [공지 상세 조회](#)
- [공지관련 멤버조회](#)
- [공지 수정](#)
- [공지 삭제](#)

- [좋아요 생성 및 삭제](#)
- [해당 공지 좋아요 리스트](#)

- [댓글 생성](#)
- [대댓글 생성](#)
- [해당 공지 댓글 리스트](#)
- [댓글 수정](#)
- [댓글 삭제](#)

- [복마크 생성 및 삭제](#)
- [복마크 조회](#)(공지, 앨범, 이미지)
- [성문정보 조회](#)
- [성문삭제](#)

- [공유 가능 멤버리스트 조회](#)
- [공유하기](#)

4. 파일 관련

- [파일 업로드\(deprecated\)](#)
- [파일 업로드\(업래키 없이 등록\)](#)
- [파일 삭제](#)
- [파일 수정](#)

5. 유틸 관련

- [시군구 데이터](#)
- [기관 검색\(어린이집, 유치원\)](#)

6. 공지 관련

- [쿼리 조회](#)

7. 스쿨 멤버 관련

- [프로필 이미지 변경](#)
- [배경 이미지 변경](#)

8. 쿼리 컨텐츠 관련

- [알림장 생성](#)
- [알림장 조회](#)
- [알림장 수정](#)
- [알림장 확인](#)
- [알림장 상세조회](#)
- [알림장 삭제](#)

- [등하원 알림장 생성](#)
- [등하원 알림장 조회](#)
- [등하원 관련 알림장 상세조회](#)
- [등하원 알림장 확인](#)
- [등하원 알림장 삭제](#)

- [투약의뢰서 생성](#)
- [투약의뢰서 확인](#)
- [투약의뢰서 조회](#)
- [투약의뢰서 상세 조회](#)
- [투약의뢰서 삭제](#)

9. 앨범 관련

- [앨범 생성](#)
- [앨범 수정](#)
- [앨범 삭제](#)
- [앨범 조회](#)
- [앨범 상세조회](#)

- [좋아요](#)
- [댓글](#)
- [복마크](#)
- [이미지만 복마크](#)

10. 채팅 관련

- [채팅 가능 멤버리스트](#)

[요청사항 페이지](#)

[배포일자별 업데이트](#)

[웹뷰 브릿지](#)

[현재 서버에서 샌드버드 처리](#)

9:41
회원가입
다음

이메일 주소
중복체크

비밀번호
8~20자의 영문, 숫자 조합
올바르지 않은 비밀번호 형식입니다.

비밀번호 재입력
비밀번호를 한번 더 입력해주세요.
비밀번호가 일치합니다.
비밀번호가 일치하지 않습니다.

☒ 전체 동의
☒ 이용약관 동의
☒ 개인정보 수집 및 이용 동의
☒ 이벤트 및 광고성 정보 수신 (선택)

- API 명
- 회원가입
- Method
- POST
- URL
/api/auth/signup
- Body(Request)
{
 ~ type(string): local | kakao | google | apple | naver
 ~ user_info : {
 ~ email(string), required = false(local 일 경우만 필수)
 ~ password(string), required = false(local 일 경우만 필수)
 ~ phone(string), required = true,
 ~ name(string), required = true
 }
 ~ access_token(string), required=false(소셜 가입만 필수),
 firebase의 경우 idToken
}

- Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- token : 토큰
- data : {
 "user": {
 "id": (uuid),
 "email": (string),
 "phone": (string),
 "activated":(boolean),
 "name": (string),
 "provider": (string),
 "createdAt": (string),
 "updatedAt": (string)
 },
 "school_member": [{
 "member_id": (uuid),
 "member_nickname": (string),
 "nickname": (string),
 "member_description": (string),
 "class_name": (string),
 "class_start_date": (string),
 "class_end_date":(string),
 "school_name": (string),
 "member_image": (string),
 "member_image_id": (string),
 "member_background_image": (string),
 "background_image_id":(string),
 "member_type": (string),
 "member_grade": (int),
 "user_id": (uuid),
 "school_id": (uuid),
 "class_id": (uuid),
 "school_role_id": (uuid),
 "member_phone": (string),
 "kid_name": (string),
 "kid_id": (uuid)



1. API 명
- 로그인
2. Method
- POST
3. URL
/api/auth/signin
4. Body(Request)
{
 ~ type(string): local | kakao | google | apple | naver
 ~ user_info : {
 ~ email(string), required = false(local 일 경우만 필수)
 ~ password(string), required = false(local 일 경우만 필수)
 }
 ~ access_token(string), required=false(소셜 로그인만 필수),
 firebase의 경우 idToken
}

5. Return Type
- JSON

```
6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- token : 토큰
- data : {
  "user": {
    "id": (uuid),
    "email": (string),
    "phone": (string),
    "activated":(boolean),
    "name": (string),
    "provider": (string),
    "createdAt": (string),
    "updatedAt": (string)
  },
  "school_member": [{
    "member_id": (uuid),
    "member_nickname": (string),
    "nickname": (string),
    "member_description": (string),
    "class_name": (string),
    "class_start_date": (string),
    "class_end_date":(string),
    "school_name": (string),
    "member_image": (string),
    "member_image_id": (string),
    "member_background_image": (string),
    "background_image_id":(string),
    "member_type": (string),
    "member_grade": (int),
    "user_id": (uuid),
    "school_id": (uuid),
    "class_id": (uuid),
    "school_role_id": (uuid),
    "member_phone": (string),
    "kid_name": (string),
    "kid_id": (uuid)
  }
}
```

1. API 명
- 로그인 정보 가져오기(로그인 시)
2. Method
- GET
3. URL
/api/auth/get-info
4. Header
Authorization : Bearer {token}

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- token : 토큰
- data : {
"user": {
"id": (uuid),
"email": (string),
"phone": (string),
"activated":(boolean),
"name": (string),
"provider": (string),
"createdAt": (string),
"updatedAt": (string)
},
"school_member": [{
"member_id": (uuid),
"member_nickname": (string),
"nickname": (string),
"member_description": (string),
"class_name": (string),
"class_start_date": (string),
"class_end_date":(string),
"school_name": (string),
"member_image": (string),
"member_image_id": (string),
"member_background_image": (string),
"background_image_id":(string),
"member_type": (string),
"member_grade": (int),
"user_id": (uuid),
"school_id": (uuid),
"class_id": (uuid),
"school_role_id": (uuid),
"member_phone": (string),
"kid_name": (string),
"kid_id": (uuid)
},
}

1. API 명
- 이메일 중복 확인
2. Method
- POST
3. URL
/api/auth/check-email
4. Body(Request)
{
 ~ user_info : {
 ~ email(string)
 }
}

5. Return Type
- JSON

6. Protocol(Response)
- **message** : 해당하는 API 결과 메시지 내용
- **resultCode** : 기본(200), 에러(400 이상)

lullaforkids@gmail.com

사용 가능한 이메일 주소입니다.

resultCode 200일 경우 사용가능한 이메일
resultCode 200이 아닐 경우 중복 이메일

□□□□

이미 사용중인 이메일 주소입니다.

9:41

<

비밀번호 재설정

완료

새로운 비밀번호를 설정해 주세요.

새 비밀번호(8~20자의 영문, 숫자 조합)

새 비밀번호 재입력(8~20자의 영문, 숫자 조합)

1. API 명
- 비밀번호 재설정(로그인시)
2. Method
- POST
3. URL
/api/auth/set-password
4. Header
Authorization : Bearer {token}
5. Body(Request)
{
 ~ user_info : {
 ~ password(string)
 }
}

5. Return Type
- JSON
6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

9:41



이메일/비밀번호 찾기



이메일 찾기

비밀번호 찾기

김유진

010-1111-2222

인증번호 재요청

인증번호 4자리를 입력하세요.

10:00

인증하기

1.

API 명

- 휴대폰 인증(코드 생성 및 sms발송)

2.

Method

- POST

3.

URL

/api/auth/generate-code

4.

Body(Request)

{

~ verify : {

~ phone(string) ex) 010-1234-1234, 01012341234 둘다 사용가능

~ type(int) 0: 회원가입 시 인증, 1: 이메일 찾기 시 인증, 2: 비밀번호 재설정 시 인증, 3 휴대폰 번호

변경

}

}

5. Return Type

- JSON

6. Protocol(Response)

- message : 해당하는 API 결과 메시지 내용

- resultCode : 기본(200), 에러(400 이상)

9:41



이메일/비밀번호 찾기

다음

이메일 찾기

비밀번호 찾기

김유진

010-1111-2222

인증번호 재요청

인증번호 4자리를 입력하세요.

10:00

인증하기

1.

API 명

- 휴대폰 인증(코드 확인)

2.

Method

- POST

3.

URL

/api/auth/match-code

4.

Body(Request)

```
{
  ~ verify : {
    ~ phone(string)
    ~ code : (string)
    ~ type(int) 0: 회원가입 시 인증, 1: 이메일 찾기 시 인증, 2: 비밀번호 재설정 시 인증,
                 3: 휴대폰 번호 변경(인증이 되면 바로 해당하는 번호로 변경 되어짐)
  }
}
```

3의 경우 로그인이
되어있어야함

1.

Header

Authorization : Bearer {token}

5. Return Type

- JSON

6. Protocol(Response)

- message : 해당하는 API 결과 메시지 내용

- resultCode : 기본(200), 에러(400 이상)

```
- data : {
  email : email
}
```

type 이 1이었던 경우만 data 반환

type 이 10이 아닌 경우 data 반환 없음

9:41



이메일/비밀번호 찾기

김유진 회원님의 이메일 계정은
lulla2021@google.com 입니다.

9:41

<

비밀번호 재설정

완료

새로운 비밀번호를 설정해 주세요.

새 비밀번호(8~20자의 영문, 숫자 조합)

새 비밀번호 재입력(8~20자의 영문, 숫자 조합)

1. API 명
- 비밀번호 재설정(로그인 안했을 시)
2. Method
- POST
3. URL
/api/auth/reset-password
4. Body(Request)

```
{  ~ user_info : {    ~ phone : (string)    ~ password : (string)  }
```

5. Return Type
- JSON

6. Protocol(Response)
- **message** : 해당하는 API 결과 메시지 내용
- **resultCode** : 기본(200), 에러(400 이상)

인증 type을 2로 받아야지 활성화가 됩니다.

1. API 명
- 회원 탈퇴(로그인 했을 경우)
2. Method
- POST
3. URL
/api/auth/signout
4. Header
Authorization : Bearer {token}

5. Return Type
- JSON

6. Protocol(Response)
- **message** : 해당하는 API 결과 메시지 내용
- **resultCode** : 기본(200), 예러(400 이상)

1. API 명
- 디바이스 토큰 등록
2. Method
- POST
3. URL
/api/auth/register-token
4. Header
Authorization : Bearer {token}
5. body
user_info : {
 ~token : (string)
 ~type (int) 0 안드로이드, 1 애플
}

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 예러(400 이상)

9:41

<

원 정보 직접 입력

등록

원 정보

시/도 선택

지역 선택

상세 주소

원 이름

원장 명

원 전화번호

- API 명
- 원 생성
- Method
- POST
- URL
/api/school
- Header
Authorization : Bearer {token}
- Body(Request)
~ school : {
 ~ name(string) required = true
 ~ tel(string) required = true
 ~ address(string) required = true
 ~ district_one_id(string) required = true
 ~ district_two_id(string)
 ~ admin_name(string) required = true
 ~ institutions_id(string)
}

institutions_id 의 경우 기관 검색으로 얻을 수 있음

```
5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- data : {
  "school": {
    "id": (uuid),
    "name": (string),
    "tel": (string),
    "address": (string),
    "description": (string),
    "admin_name": (string),
    "district_one_name": (string),
    "district_two_name": (string),
  },
  "school_member": {
    "member_id": (uuid),
    "member_nickname": (string),
    "member_description": (string),
    "class_name": (string),
    "school_name": (string),
    "member_image": (string),
    "member_type": (string),
    "member_grade": (int),
    "user_id": (uuid),
    "school_id": (uuid),
    "class_id": (uuid),
    "school_role_id": (uuid),
    "member_phone": (string),
    "kid_id": (string),
    "kid_name": (string)
  }
}
```

9:41



이미 등록된 원입니다.
원 정보를 다시 확인해주세요.

확인

1. API 명
- 기관검색으로 생성된 원 체크
2. Method
- POST
3. URL
/api/school/check
4. Header
Authorization : Bearer {token}
5. Body(Request)
~ school : {
 ~ institutions_id(string)
}

5. Return Type
- JSON

6. Protocol(Response)
- **message** : 해당하는 API 결과 메시지 내용
- **resultCode** : 기본(200), 에러(400 이상)

9:41

원

저장

시/도

지역

서울특별시

광진구

상세 주소

농동로 112, 건국대학교

원 이름

건국대학교 부속 유치원

원장 명

남선화

원 전화번호

02-3333-4444

원 정보 삭제

원에 소속된 반과 구성원이 있는 경우 삭제 할 수 없습니다.

- API 명
- 원 조회
- Method
- GET
- URL
/api/school
- Header
Authorization : Bearer {token}
- Query(Request)
- member_id, required =true
- search 검색하고자 하는 내용

검색의 경우 데이터(search가 존재시)

↓
datas : [{
 "id": (uuid),
 "name": (string),
 "tel": (string),
 "address": (string),
 "description": (string),
 "member_nickname": (string),
 "admin_name": (string),
 "district_one_name": (string),
 "district_one_id": (string),
 "district_two_name": (string),
 "district_two_id": (string),
},...],

5. Return Type

- JSON

6. Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 예외(400 이상)

```
- data : {
  "id": (uuid),
  "name": (string),
  "tel": (string),
  "address": (string),
  "description": (string),
  "member_nickname": (string),
  "admin_name": (string),
  "district_one_name": (string),
  "district_one_id": (string),
  "district_two_name": (string),
  "district_two_id": (string),
  "class_data": [{
    "name": (string),
    "id": (uuid),
    "start_date": (string),
    "end_date": (string),
    "description": (string)
  }, ...]
},
```

9:41

원

저장

시/도

지역

서울특별시

광진구

상세 주소

농동로 112, 건국대학교

원 이름

건국대학교 부속 유치원

원장 명

남선화

원 전화번호

02-3333-4444

원 정보 삭제

원에 소속된 반과 구성원이 있는 경우 삭제 할 수 없습니다.

- API 명
- 원 수정
- Method
- POST
- URL
/api/school/update
- Header
Authorization : Bearer {token}
- Body(Request)

```

school : {
  id : (어린이집 id),
  name : (어린이집 이름),
  tel : (전화번호),
  admin_name : (원장 이름),
  address : (주소),
  district_one_id : (시/도 id),
  district_two_id : (지역 id)
}

```

5. Return Type

- JSON

6. Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```

- data : {
  "id": (uuid),
  "name": (string),
  "tel": (string),
  "address": (string),
  "description": (string),
  "member_nickname": (string),
  "admin_name": (string),
  "district_one_name": (string),
  "district_one_id": (string),
  "district_two_name": (string),
  "district_two_id": (string),
  "class_data": [{
    "name": (string),
    "id": (uuid),
    "start_date": (string),
    "end_date": (string),
    "description": (string)
  }, ...]
},

```


9:41

원

저장

시/도

지역

서울특별시

광진구

상세 주소

능동로 112, 건국대학교

원 이름

건국대학교 부속 유치원

원장 명

남선화

원 전화번호

02-3333-4444

원 정보 삭제

원에 소속된 반과 구성원이 있는 경우 삭제 할 수 없습니다.

1. API 명
- 원 삭제
2. Method
- POST
3. URL
/api/school/delete
4. Header
Authorization : Bearer {token}
5. Body(Request)
member_id - 삭제하고자 하는 원의 member_id

5. Return Type
- JSON
6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)



The image shows a mobile application interface for creating a class. At the top, the status bar displays the time 9:41 and signal icons. Below the status bar is a header with a green back arrow and the title '반' (Class). The main content area is titled '반 정보' (Class Information). It contains two input fields: the first is a dropdown menu labeled '시작일 - 종료일' (Start Date - End Date) with a downward arrow, and the second is a text input field labeled '반 이름' (Class Name). Below these fields is a grey button labeled '반 추가하기' (Add Class).

1. API 명
- 반 생성
2. Method
- POST
3. URL
/api/class
4. Header
Authorization : Bearer {token}
5. Body(Request)

~ class_name(string)
~ class_start_date(date)
~ class_end_date(date)
~ member_id(string)

5. Return Type
- JSON

6. Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- data : {

```
"class_id": (uuid),  
"class_name": (string),  
"class_description": (string),  
"class_start_date": (string),  
"class_end_date": (string),  
"school_id": (uuid),  
"teachers": [],  
"guardians": [],  
"teacher_invitation": [],  
"teacher_demand": [],  
"guardian_invitation": [],  
"guardian_demand": []
```

```
},
```

9:41

반

반 정보

시작일 - 종료일

반 이름

반 추가하기

바나나반

2020.05.04 - 2020.05.17

체리반

2020.05.04 - 2020.05.17

망고반

2020.05.04 - 2020.05.17

딸기반

2020.05.04 - 2020.05.17

- API 명
- 반 전체 조회
- Method
- GET
- URL
/api/class
- Header
Authorization : Bearer {token}
- Query(Request)
- member_id(string)

5. Return Type
- JSON

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- datas : [{
"class_id": (uuid),

"class_name": (string),

"class_description": (string),

"class_start_date": (string),

"class_end_date": (string),

"teachers": [{

[api/member 리스폰스 참고](#) - kid 없음

}],

"teachers_demand" : [{

"user_id": (uuid),

"username": (string),

"demand_id": (uuid)

}],

"teachers_invitation" : [{

"member_id" : (uuid),

"member_nickname" : (string),

"user_id" : (uuid),

"username" : (string),

"invitation_id" : (uuid)

}],

"guardians": [{

[api/member 리스폰스 참고](#)

},

},],

"guardians_demand" : [{

"user_id": (uuid),

"user_name": (string),

"demand_birth": (string),

"demand_name": (string),

"demand_gender": (int),

9:41

<

선생님 정보

조회

선생님 정보

랄라유치원

반 선택

1. API 명
- 반 조회(member_id 없이)
2. Method
- GET
3. URL
/api/class/info
4. Header
Authorization : Bearer {token}
5. Query(Request)
- id(string) 해당하는 원의 id

5. Return Type
- JSON

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- datas : [
 {
 "name": (string) ex. "테스트 0 반",
 "id": (uuid) ex. "b2804cc0-ca71-11eb-a280-1d94c8c40ebb",
 "start_date": (string) ex. "2020-03-01",
 "end_date": (string) ex. "2020-12-01"
 },
]

9:41

<

반 정보 수정

저장

시작일 - 종료일

2020.05.04 - 2020.05.17

▼

반 이름

복숭아반

반 정보 삭제

반에 소속된 구성원이 있는 경우 삭제 할 수 없습니다.

1. API 명
- 반 수정
2. Method
- POST
3. URL
/api/class/update
4. Header
Authorization : Bearer {token}
5. Body(Request)

```
{
  ~ class_id
  ~ class_name(string)
  ~ class_start_date(date)
  ~ class_end_date(date)
  ~ member_id(string)
}
```

5. Return Type
- JSON

6. Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- data : {
 "class_id": (uuid),
 "class_name": (string),
 "class_description":(string),
 "class_start_date": (string),
 "class_end_date": (string),
 "school_id": (uuid),
 "teachers": [],
 "guardians": [],
 "teacher_invitation": [],
 "teacher_demand": [],
 "guardian_invitation": [],
 "guardian_demand": []
},

9:41

<

반 정보 수정

저장

시작일 - 종료일

2020.05.04 - 2020.05.17

▼

반 이름

복숭아반

반 정보 삭제

반에 소속된 구성원이 있는 경우 삭제 할 수 없습니다.

1. API 명
- 반 삭제
2. Method
- POST
3. URL
/api/class/delete
4. Header
Authorization : Bearer {token}
5. Body(Request)
{
 ~ class_id
 ~ member_id
}

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)



1. API 명
- 초대장 생성

2. Method
- POST

3. URL
/api/invitation

4. Header
Authorization : Bearer {token}

5. Body(Request)

```

~ invitation : {
  ~ relation(string) 원아와 보호자의 관계
  ~ phone(string)
  ~ kid_name
  ~ kid_id (가족의 경우 매핑시 필요)
  ~ role_name(역할)
  ~ type(int), (2: 관리자 ,3 : 담임 ,4 :보조담임 ,5 학부모 ,6 :
가족)
  ~ class_id(string) : required=false (관리자의 경우 필요 X)
},
member_id(string)

```

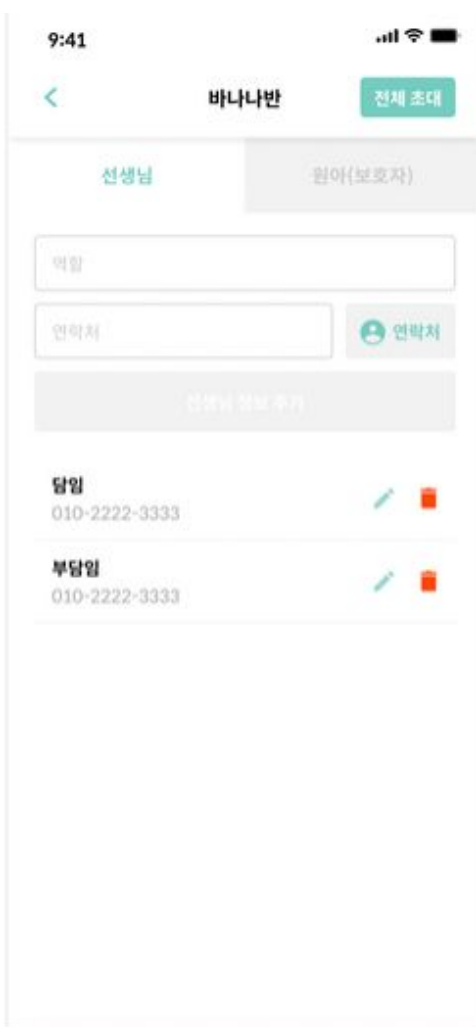
5. Return Type
- JSON

```

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- data : {
  "id": (uuid),
  "confirmed": (boolean),
  "type": (number),
  "school_name": (string),
  "school_description": (string),
  "school_address": (string),
  "school_tel": (string),
  "class_name": (string),
  "class_description": (string),
  "kid_name": (string)
},

```

기획 변경으로 인해서 변경 예정



1. API 명
- 초대장 요청장 조회(기관)
2. Method
- GET
3. URL
/api/class
4. Header
Authorization : Bearer {token}
5. Query(Request)
- member_id, required = false
- id, (해당하는 반의 id)

5. Return Type
- JSON

```
6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- data : {
  "teacher_invitation": [{
    "user_id": (uuid),
    "username": (string),
    "invitation_id": (uuid),
    "invitation_type": (number)
  }],
  "teacher_demand": [{
    "user_id": (uuid),
    "username": (string),
    "demand_id": (uuid),
    "demand_type": (number)
  }],
  "guardian_invitation": [{
    "user_id": (uuid),
    "username": (string),
    "user_phone": (string),
    "invitation_id": (uuid),
    "invitation_type": (number)
  }],
  "guardian_demand": [{
    "user_id": (uuid),
    "user_name": (string),
    "demand_birth": (string),
    "demand_name": (string),
    "demand_gender": (number),
    "demand_id": (uuid),
    "demand_relation": (string),
    "demand_type": (number)
  }],
}
```

기획 변경으로 인해서 변경 예정



1. API 명
- 초대장 조회
2. Method
- GET
3. URL
/api/invitation
4. Header
Authorization : Bearer {token}
5. Query(Request)
- member_id, required = false
(member_id 존재 시 해당하는 학급에 대한 초대장 조회)
(member_id 미 존재 혹은 보호자 등급일 경우
해당하는 유저에 대한 초대장 조회)

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- datas : [{
 ~ id(uuid)
 ~ type(string)
 ~ description(string)
 ~ confirmed(boolean)
 ~ school_name(string)
 ~ school_description(string)
 ~ school_address(string)
 ~ school_tel(string)
 ~ class_name(string)
 ~ class_description(string)
 ~ kid_name(string)
 ...],

기획 변경으로 인해서 변경 예정

이미지 없음

1. API 명
- 원아 생성
2. Method
- POST
3. URL
/api/kid
4. Header
Authorization : Bearer {token}
5. Body(Request)
{
 ~ kid : {
 ~ name(string)
 ~ age(int)
 ~ class_id(string)
 ~ note(string)
 }
 member_id(string)
}

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- data : {
 ~ id(uuid)
 ~ name(string)
 ~ age(int)
 ~ class_id(uuid)
 ~ note(string)
}

1. API 명
- 원아 조회
2. Method
- GET
3. URL
/api/kid
4. Header
Authorization : Bearer {token}
5. Query(Request)
- member_id(string), required=true

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- datas : [{
"kid_id": (uuid),
"kid_name": (string),
"kid_age": (int),
"kid_note": (string),
"created_at": (string),
"updated_at": (string),
"member_id": (uuid),
"member_nickname": (string),
"class_name": (string),
"class_start_date": (string),
"class_end_date": (string),
"school_name": (string),
}, ...]

이미지 없음

9:41

< 휴무일 완료

휴무일과 사유를 등록해주세요.

(휴무일을 설정하시면 보호자가 휴무일에 등하원 관련 알림을 보낼때 경고메시지를 보내고, 버스운행이 있는 경우 기사님에게 운행알람을 전송하지 않습니다.)

기간 설정 시작 연/월/일 종료 연/월/일

휴무기간 설정을 완료해주세요.

사유 (예: 여름방학, 임시공휴일 등)

사유를 입력해주세요.

+ 휴무일 추가

1. API 명
- 휴무일 생성
2. Method
- POST
3. URL
/api/holiday
4. Header
Authorization : Bearer {token}
5. Body(Request)


```
{
    - holiday : {
      ~ name(string)
      ~ description(string)
      ~ start_date(date)
      ~ end_date(date)
    }
    - member_id(string), required=true
  }
```

5. Return Type

- JSON

6. Protocol(Response)

- message : 해당하는 API 결과 메시지 내용

- resultCode : 기본(200), 에러(400 이상)

- data : {
 ~ id(uuid)
 ~ name(string)
 ~ description(string)
 ~ start_date(string)
 ~ end_date(string)
 }

9:41

< 휴무일 완료

휴무일과 사유를 등록해주세요.

(휴무일을 설정하시면 보호자가 휴무일에 등하원 관련 알림을 보낼때 경고메시지를 보내고, 버스운행이 있는 경우 기사님에게 운행일람을 전송하지 않습니다.)

기간 설정 시작 연/월/일 종료 연/월/일

휴무기간 설정을 완료해주세요.

사유 (예: 여름방학, 임시공휴일 등)

사유를 입력해주세요.

+ 휴무일 추가

1. API 명
- 휴무일 조회
2. Method
- GET
3. URL
/api/holiday
4. Header
Authorization : Bearer {token}
5. Query(Request)
- member_id(string), required=true

5. Return Type

- JSON

6. Protocol(Response)

- message : 해당하는 API 결과 메시지 내용

- resultCode : 기본(200), 에러(400 이상)

- datas : [{
~ id(uuid)
~ name(string)
~ description(string)
~ start_date(string)
~ end_date(string)
}, ...]

9:41

<

휴무일 설정

휴무일 정보

시작일 - 종료일

▼

휴무일 이름

휴무일 정보 추가

여름방학

2020.05.04 - 2020.05.17

1. API 명
- 휴무일 삭제
2. Method
- POST
3. URL
/api/holiday/delete
4. Header
Authorization : Bearer {token}
5. Body(Request)
 - member_id(string), required=true
 - holiday {
 - ~ id

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)



1. API 명
- 스쿨 멤버 조회
2. Method
- GET
3. URL
/api/member
4. Header
Authorization : Bearer {token}
5. Query(Request)
- member_id(string), required=false
(member_id 가 있을 경우 해당하는 원에 대한 정보를 보여줌)
- type(string), required=false
(teacher, guardian)

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- datas : [{
"member_id": (uuid),
"member_nickname": (string),
"member_description": (string),
"class_name": (string),
"class_start_date": (string),
"class_end_date": (string),
"school_name": (string),
"member_image": (string),
"member_image_id": (uuid)
"member_background_image": (string)
"background_image_id": (uuid),
"member_type": (string),
"member_grade": (int),
"school_id": (uuid),
"class_id": (uuid),
"school_role_id": (uuid),
"member_phone": (string),
"kid_name": (string),
"kid_id": (uuid),
"email": (string)
"phone": (string)
"is_admin": (boolean),

}, ...]



- API 명
- 스쿨 멤버 상세 조회
- Method
- GET
- URL
/api/member?id=
- Header
Authorization : Bearer {token}
- Query(Request)
- id (string), required=true (해당하는 멤버의 id)
- member_id(string), required=true
(member_id 가 있을 경우 해당하는 원에 대한 정보를 보여줌)

- Return Type
- JSON

6. Protocol(Response)
 - message : 해당하는 API 결과 메시지 내용
 - resultCode : 기본(200), 에러(400 이상)
 - data : {
 "member_id": (uuid),
 "member_nickname": (string),
 "member_description": (string),
 "class_name": (string),
 "class_start_date": (string),
 "class_end_date": (string),
 "school_name": (string),
 "member_image": (string),
 "member_image_id": (uuid)
 "member_background_image": (string)
 "background_image_id": (uuid),
 "member_type": (string),
 "member_grade": (int),
 "school_id": (uuid),
 "class_id": (uuid),
 "school_role_id": (uuid),
 "member_phone": (string),
 "kid_name": (string),
 "kid_id": (uuid),
 "email": (string)
 "phone": (string)
 "is_admin": (boolean),
 "chat_active": (boolean),
 }

1. API 명
- 스쿨 멤버 정보 변경
2. Method
- POST
3. URL
/api/member/update
4. Header
Authorization : Bearer {token}
5. Body(JSON)
{
- member_id(본인의 member_id)uuid
- nickname(변경하고자 하는 닉네임) string
- member_description(변경하고자 하는 설명)string
- image_id (업로드를 미리한 후 나오는 해당 파일 ID)
- background_image_id (업로드를 미리한 후 나오는 해당 파일 ID)
}

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- datas : {
"member_id": (uuid),
"member_nickname": (string),
"nickname": (string),
"member_description": (string),
"class_name": (string),
"class_start_date": (string),
"class_end_date": (string),
"school_name": (string),
"member_image": (string),
"member_image_id": (uuid)
"member_background_image": (string)
"background_image_id": (uuid),
"member_type": (string),
"member_grade": (int),
"school_id": (uuid),
"class_id": (uuid),
"school_role_id": (uuid),
"member_phone": (string),
"kid_name": (string),
"kid_id": (uuid),
"email": (string)
"phone": (string)
"is_admin": (boolean),
}



1. API 명
- 프로필 이미지 변경
2. Method
- POST
3. URL
/api/member/image
4. Header
Authorization : Bearer {token}
5. body(Form-data)
- member_id(string), required=true
- file : 변경할 이미지 등록

6. Return Type
- JSON

7. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

이미지는 따로 존재하지는 않음



1. API 명
- 배경사진 이미지 변경
2. Method
- POST
3. URL
/api/member/background-image
4. Header
Authorization : Bearer {token}
5. body(Form-data)
- member_id(string), required=true
- file : 변경할 이미지 등록

6. Return Type
- JSON

7. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

이미지는 따로 존재하지는 않음



1. API 명
- 공지 생성
2. Method
- POST
3. URL
/api/post
4. Body(Request)
- {
 ~ post : {
 ~title(string)
 ~contents(string)
 ~important(boolean)
 ~post_type(int) ex) 0:공지, 1:앨범
 ~is_published(boolean)
 ~allowed_class([:클래스 ID])
 ~allowed_member([:멤버 ID])
 ~file_list([:파일 ID])
 ~survey_id(string)
 }
 ~ member_id(멤버 아이디)
}

원장일 경우 글의 대상선택이 전체일 경우에는
allowed_class와 **allowed_member**값을 보내지 않으면됨
데이터 반환 형식은 상세공지 API와 동일

- 5.Return Type
- JSON

6.Protocol(Response)

```
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 예러(400 이상)
- data : {
  "id": (uuid),
  "title": (string),
  "contents": (string),
  "is_modified": (boolean),
  "published_time": (date),
  "important": (boolean),
  "author_id": (uuid),
  "created_at": (string),
  "like_count": (int),
  "comment_count": (int),
  "is_like": (boolean),
  "is_bookmark": (boolean),
  "author_nickname": (string),
  "allowed_member": [],
  "allowed_class": [],
  "class_name": (string),
  "school_name": (string),
  "author_type": (string),
  "author_grade": (int),
  "images": [ {
    "id": (uuid),
    "address": (string),
    "size": (int),
    "thumbnail_address": (string),
    "type": (string),
    "name": (string),
    "width": (int),
    "height": (int),
    "index": (int),
    "length": (int)
  } ],
  "files": [ 이미지와 동일 width,height,index,length,
    thumbnail_address 만 빠짐 ],
  "author_image": (string)
}
```

선생님 김영민

테스트 0 반

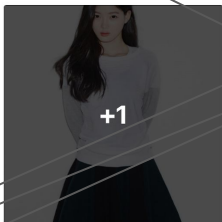
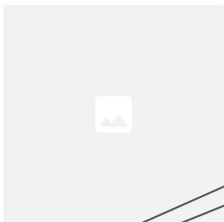
부모님 참관 수업 안내

[구름반 김민준(민준맘) 외 2명]

공지 내용 테스트 중 0

설문지 (제출 7일 전)

결과 상세보기



선생님 김영민

테스트 0 반

1. API 명
- 공지 리스트 요청
2. Method
- GET
3. URL
/api/post
4. Query

- member_id : 해당 프로필 ID, required=true
- page, int(ex page=1)
- temp, boolean : true일 경우 임시저장한 공지글을 가져옴
- class.: 반의 ID (string) (ex
1. class=uuid,uuid
2. class=[uuid,uuid])
- search : string 검색할 내용
- category : string (author, content)
ex)
1. category=[author,content]
2. category=author,content

5. Return Type
- JSON

- 6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```

- datas : [
  "id": (uuid),
  "title": (string),
  "contents": (string),
  "published_time":(string) ,
  "important": (boolean),
  "author_id": (uuid),
  "like_count": (int),
  "comment_count": (int) ,
  "is_like":(boolean),
  "is_bookmark": (boolean),
  "author_nickname": (string),
  "author_image": (string),
  "allowed_member": [
    {"nickname": (string),
    "id": (uuid),
    "member_image":(string) ,
    "kid_name": (string),
    "class_name":(string)
    },...],
  "allowed_class": [
    {"name": (string),
    "id": (uuid) ,
    "allowed_member": (위의 allowed_member 와 동일) }, .....],
  "class_name": (string),
  "school_name":(string),
  "author_type":(string),
  "author_grade":(int),
  "images": [{
    "id": (uuid),
    "address": (string_,
    "size": (int),
    "thumbnail_address": (string),
  
```

"survey_id": "(uuid)",
 "survey_end_date": "(date)",
 "complete_survey": (boolean) 설문조사를 완료했는지 여부
 "survey_end": (boolean) 설문조사의 기한이 지났는지 여부
 "question_count": (number) 설문문의 질문 수,

● 2020.10.30 금요일

④ 부모님 청관 수업 안내

9

■ 5

1

1

1. API 명
- 공지 상세 조회
2. Method
- GET
3. URL
/api/post?id=&member_id=
4. Query
- member_id : 해당 프로필 ID, required=true
- id : 해당 공지 id , required = true

```
comment": {  
    "id": (string),  
    "content": (string),  
    "created_at": (string),  
    "updated_at": (string),  
    "member_id": (uuid),  
    "member_nickname": (string),  
    "member_image": (string),  
    "member_grade": (int),  
    "member_type": (string),  
    "school_role_id": (uuid),  
    "kid_id": (uuid),  
    "kid_name": (string),  
    "class_id": (uuid),  
    "class_name": (string),  
    "tocomment": ({위의 comment와 동일}), ....}
```

```
- "like": {  
  "id": (uuid),  
  "created_at": (string),  
  "member_id": (uuid),  
  "member_nickname": (string),  
  "member_image": (string),  
  "member_grade": (int),  
  "member_type": (string),  
  "school_role_id": (uuid),  
  "kid_id": (uuid),
```

- JSON

- message : 해당하는 API 결과 메시지 내용

```
- data : {
```

```

"post": {
  "id": (uuid),
  "title": (string),
  "contents": (string),
  "published_time": (string),
  "important": (boolean),
  "author_id": (uuid),
  "like_count": (int),
  "comment_count": (int),
  "is_like": (boolean),
  "is_bookmark": (boolean),
  "author_nickname": (string),
  "author_image": (string) ,
  "allowed_member": [{
    "nickname": (string),
    "id": (uuid),
    "member_image": (string),
    "kid_name": (string),
    "class_name": (string)
  }],
  "allowed_class": [{
    "name": (string),
    "id": (uuid),
    "allowed_member": (위의 allowed_member 와 동일)
  }],

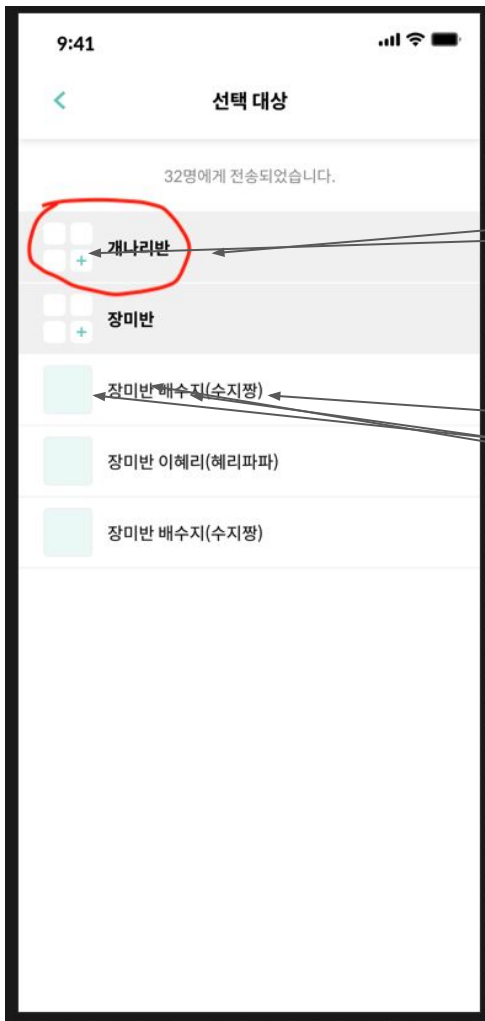
```

```

}, ...].

"class_name": (string),
"school_name": (string),
"author_type": (string),
"author_grade":(int),
"created_at":(string)-임시저장일 때 사용,
"images": [{
    "id": (uuid),
    "address": (string

```



1. API 명
- 공지 관련 멤버 조회

2. Method
- GET

3. URL
/api/post/member?member_id={member ID}&id={post ID}

4. Query
- member_id : 해당 프로필 ID, required=true
- id : 해당 공지 ID, required = true

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- data : {

```
"allowed_class": [{  
  "name": (string),  
  "id": (uuid),  
  "allowed_member": [{  
    "nickname": (string),  
    "id": (uuid),  
    "member_image": (string),  
    "kid_name": (string),  
    "class_name": (string)  
  }, ...],  
},  
"allowed_member": [{  
  "nickname": (string),  
  "id": (uuid),  
  "member_image": (string),  
  "kid_name": (string),  
  "class_name": (string)  
}, ...]]
```



1. API 명
- 공지 수정
2. Method
- POST
3. URL
/api/post
4. Body(Request)
- {
 ~ post : {
 ~ id
 ~ title(string)
 ~ contents(string)
 ~ important(string)
 ~ is_published(boolean)
 ~ allowed_class([:클래스 ID])
 ~ allowed_member([:멤버 ID])
 ~ file_list([:파일 ID])
 ~ deleted_list([:파일 ID])
 ~ survey_id(설문 ID)
 }
 ~ member_id(멤버 아이디)
}

5.Return Type

- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용

- resultCode : 기본(200), 에러(400 이상)

```
- data : {
  "id": (uuid),
  "title": (string),
  "contents": (string),
  "is_modified": (boolean),
  "published_time": (string),
  "important": (boolean),
  "author_id": (uuid),
  "created_at": (string),
  "like_count": (int),
  "comment_count": (int),
  "is_like": (boolean),
  "is_bookmark": (boolean),
  "author_nickname": (string),
  "allowed_member": [],
  "allowed_class": [],
  "class_name": (string),
  "school_name": (string),
  "author_type": (string),
  "author_grade": (int),
  "images": [{
    "id": (uuid),
    "address": (string),
    "size": (int),
    "thumbnail_address": (string),
    "type": (string),
    "name": (string),
    "width": (int),
    "height": (int),
    "index": (int),
    "length": (int)
  }],
  "files": [이미지와 동일 width,height,index,length,
    thumbnail_address 만 빠짐],
  "author_image": (string)
}
```




1. API 명
- 공지 삭제
2. Method
- POST
3. URL
/api/post/delete
4. Body(Request)
- {
 ~ post : {
 ~ id (type : list 형태로 [id, id, id])
 }
 ~ member_id (멤버 아이디)
}

5. Return Type
- JSON

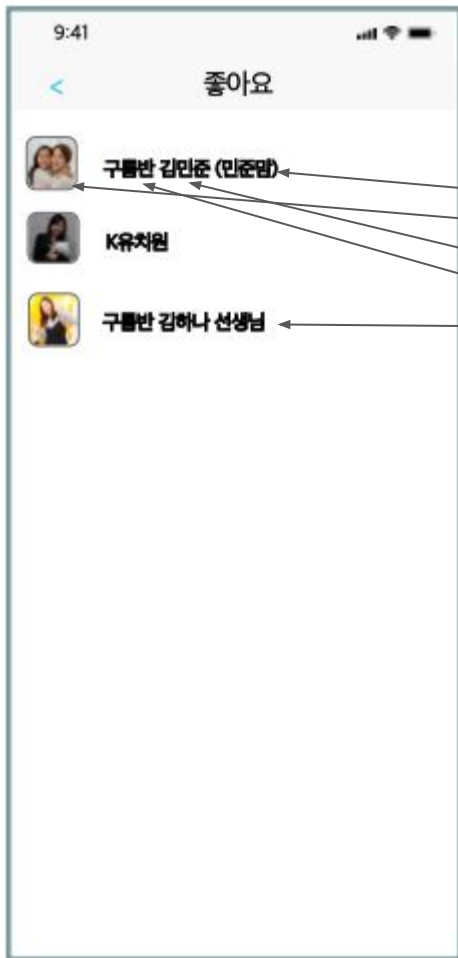
6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)



1. API 명
- 좋아요 생성 및 삭제
2. Method
- POST
3. URL
/api/post/like
4. Body(Request)
- {
 like : {
 post_id : uuid
 },
 ~ member_id(string)
}
5. Query
post : 해당 공지 ID

6.Return Type
- JSON

7.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)



1. API 명
- 해당 공지 좋아요 리스트 반환

2. Method
- GET

3. URL
/api/post/like?post

4. Query
member_id, required=true

5. Query
post : 해당하는 포스트의 ID, required = true

6.Return Type
- JSON

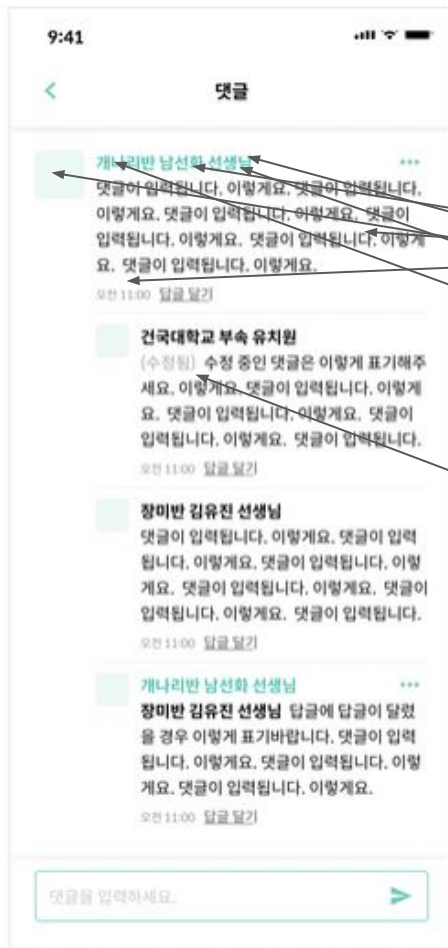
7.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- data :[{
"id": (uuid),
"created_at":(string) ,
"member_id": (uuid),
"member_nickname": (string),
"member_image": (string),
"member_grade": (int),
"member_type": (string),
"school_role_id": (uuid),
"kid_id": (uuid),
"kid_name": (string),
"class_id": (uuid),
"class_name": (string)
}, ...]



1. API 명
- 댓글 생성
2. Method
- POST
3. URL
/api/post/comment
4. Body(Request)
- {
 ~ comment : {
 ~ post_id
 ~ content(string)
 }
 ~ member_id(string)
}
5. Query
post : 해당 공지 ID

6.Return Type
- JSON

7.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 에러(400 이상)
- data : {
 ~ id(uuid)
 ~ content(uuid)
 ~ member_id(uuid)
 ~ post_id(uuid)
}



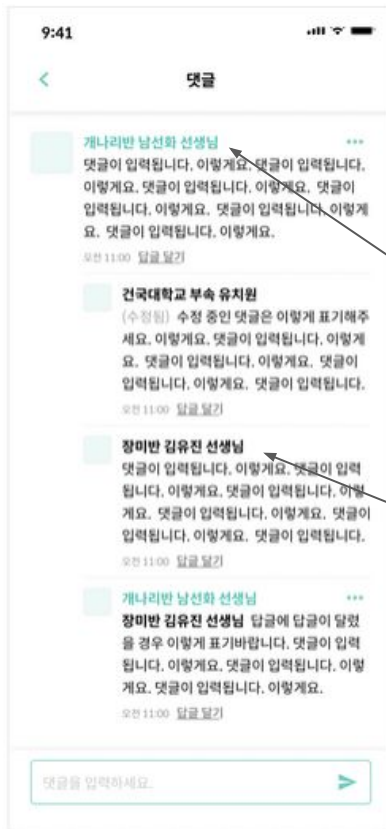
1. API 명
- 해당 공지글 댓글 리스트 반환
2. Method
- GET
3. URL
/api/post/comment?post=&member_id=
4. Query
member_id, required=true
id, required=true

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 예러(400 이상)
- datas : [{
 "idx": (uuid),
 "content": (string),
 "created_at": (string),
 "updated_at": (string),
 "member_id": (uuid),
 "member_nickname": (string) ,
 "member_image": (string),
 "member_grade":(int),
 "member_type": (string),
 "school_role_id": (uuid),
 "kid_id": (uuid),
 "kid_name": (string),
 "class_id": (uuid),
 "class_name": (string),
 "target_member_id": (uuid),
 "target_member_nickname": (string),
 "target_class_name": (string),
 "target_kid_name": (string),
 "target_member_type": (string),
 "target_member_grade": (int),
 "is_modified" : (boolean),
 "is_deleted":(boolean),
 "tocomment" : (array 위의 comment와 동일),
 }, ...]



장미반 김유진 선생님



1. API 명
- 대댓글 생성

2. Method
- POST

3. URL
/api/comment

4. Body
{
- comment : {
 ~ content
 ~ comment_id :
 ~ target_member_id :
}
- member_id
}

5. Return Type
- JSON

6. Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 에러(400 이상)
- data : {
 ~ id(uuid)
 ~ content(uuid)
 ~ member_id(uuid)
 ~ target_member_id(uuid)
}

대댓글의 답글의 경우

comment_id : 최상단의 댓글 id

target_member_id : 답글달고 싶은 댓글의 멤버 아이디



1. API 명
- 댓글 수정
2. Method
- POST
3. URL
/api/comment/update
4. Body
{
 - comment : {
 ~ content
 ~ id
 }
 - member_id
}

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- data : {
 ~ id(uuid)
 ~ content(uuid)
 ~ member_id(uuid)
}



1. API 명
- 댓글 삭제
2. Method
- POST
3. URL
/api/comment/delete
4. Body
{
 comment : {
 id:uuid
 },
 - member_id
}

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

대댓글 삭제도 이와 같음



1. API 명
- 북마크 생성 및 삭제

2. Method
- POST

3. URL
/api/post/bookmark?post=

4. Body(Request)
- {
 ~bookmark : {
 ~post_id
 }
 ~ member_id(string)
}

5. Query
post : 해당 공지 ID

6.Return Type
- JSON

7.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)



1. API 명
- 북마크 조회
2. Method
- GET
3. URL
/api/bookmark?type=
4. Query(Request)
- {
 ~ member_id(string)
 ~ type(int) // 0은 전체, 1은 공지, 2는 앨범
}

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- datas: [{
 "title": (string),
 "contents": (string),
 "published_time": (string) ,
 "important":(boolean),
 "author_nickname": (string),
 "author_id": (uuid),
 "author_grade": (int),
 "author_type": (string),
 "school_name": (string),
 "class_name": (string),
 "author_images":(string),
 "id": (uuid),
 "type": (int),
 "post_id": (uuid),
 "album_id": (uuid),
 "image_id": (uuid),
 "file_type": (string),
 "album_image": (string),
}, ...]

임시 저장 불러오기

임시 저장

하늘반 외

제목을 입력하세요. (공백 포함 최대 24자)

중요 공지 활성화

설문

2개의 질문이 있습니다.

마감: 2020.12.25 오후 12시(제출 마감 24시간 전)

설문조사

카메라

和



파일

1. API 명
- 설문 삭제
2. Method
- POST
3. URL
/api/survey/delete
4. Query(Request)
- survey : {
 ~ id(string) : 설문 id
}

5.Return Type

- JSON

6. Protocol(Response)

- message : 해당하는 API 결과 메시지 내용

- resultCode : 기본(200), 에러(400 이상)

9:41

<

대상

완료

전체

미응답자

>

반 선택

>

개별 선택

>

1. API 명
- 공유 가능한 멤버 리스트
2. Method
- get
3. URL
/api/post/share-list
4. Query(Request)
- id(string) 해당하는 공지 id
- member_id(string) 본인 member_id

5.Return Type
- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```
"data":
{
  "non_response": [
    {
      "id": (uuid) ex ."b29dbfd0-ca71-11eb-a280-1d94c8c40ebb",
      "member_nickname": (string) ex ."테스트 0 반 재민(보호자)",
      "class_name": (string) ex ."테스트 0 반",
      "kid_name": (string) ex ."재민",
      "member_image": (string) ex . null
    }, ....],
  "class_list": [
    {
      "class_id": (uuid) ex ."b2804cc0-ca71-11eb-a280-1d94c8c40ebb",
      "class_name": (string) ex ."테스트 0 반"
    }...],
  "individual": [
    {
      "id": (uuid) ex ."b29dbfd0-ca71-11eb-a280-1d94c8c40ebb",
      "member_nickname":(string) ex . "테스트 0 반 재민(보호자)",
      "member_image": (string) ex . null
    },.... ]
  }
```



1. API 명
- 공지 공유
2. Method
- post
3. URL
/api/post/share
4. Body(Request)
 - id(string) 해당하는 공지 id
 - member_list ([member_id]) 공유할 멤버 id list
 - class_list ([class_id]) 공유할 class id list
 - is_all (boolean) 전체 공유일 경우 true
 - message(string) 공유와 같이 보낼 메시지

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

테스트 미완

내용을 입력하세요.
모든 게시물은 임시저장이 가능합니다.



설문조사



카메라



앨범



파일

제한 조건
한 파일당 50mb

1. API 명
- 파일 업로드

2. Method
- POST

3. URL
/api/post/file-upload

4. body(형식 : multipart/form-data) - 단일 업로드
~ file(key) : file data(이미지, 비디오, 파일 등등)
~ post_id(key) : post id
~ member_id(key) : member id
~ index : (int) - 이미지와 동영상만 해당
~ width : (int) - 이미지만 해당
~ height : (int) - 이미지만 해당
~ duration : (int) - 동영상만 해당

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지
내용
- resultCode : 기본(200), 에러(400 이상)

파일 하나당 API 하나씩 태워서 보내야함

ex) 파일이 10개일 경우 API도 10번 호출

그렇게 하지않으면 안에 있는 form-data에 대한 값 매칭이
어려움

내용을 입력하세요.
모든 게시물은 임시저장이 가능합니다.



설문조사



카메라



앨범



파일

1. API 명
- 파일 업로드(외래키 없이 등록)
2. Method
- POST
3. URL
/api/file
4. body(형식 : multipart/form-data) - 단일 업로드
~ file(key) : file data(이미지, 비디오, 파일 등등)
~ member_id(key) : member id
~ index : (int) - 이미지와 동영상만 해당
~ width : (int) - 이미지만 해당
~ height : (int) - 이미지만 해당
~ duration : (int) - 동영상만 해당

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- data : {
 "id" : (uuid),
 "name" : (string),
 "type" : (string) (ex. image,application,video)
 "address" : (string),
 "size" : (int),
 "author_id" : (uuid),
 "height" : (int),
 "width" : (int),
 "index" : (int) (입력 값이 없을 시 -1)
}

파일 하나당 API 하나씩 태워서 보내야함

ex) 파일이 10개일 경우 API도 10번 호출

그렇게 하지않으면 안에 있는 form-data에 대한 값 매칭이
어려움

내용을 입력하세요.

모든 게시물은 임시저장이 가능합니다.



설문조사



카메라



앨범



파일

1. API 명
- 파일 삭제
2. Method
- POST
3. URL
/api/file/delete
4. body(형식 : application/json)
- {
 file : {
 - id : [:파일 id] or string(파일 ID)
 }
}

5.Return Type

- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용

- resultCode : 기본(200), 에러(400 이상)

내용을 입력하세요.
모든 게시물은 임시저장이 가능합니다.



설문조사



카메라



앨범



파일

1. API 명
- 파일 수정

2. Method
- POST

3. URL
/api/file/update

4. body(형식 : application/json)
- {
 file : [{
 - id : string(파일 id),
 - index : int()
 }]
}

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

파일 수정시에 값 대부분이 파일에 종속적인
값이라서
수정할 수 있는 값은 **index** 정도 밖에 없음

9:41

<

원 정보 직접 입력

등록

원 정보

시/도 선택

지역 선택

상세 주소

원 이름

원장 명

원 전화번호

1. API 명
- 시군구 데이터

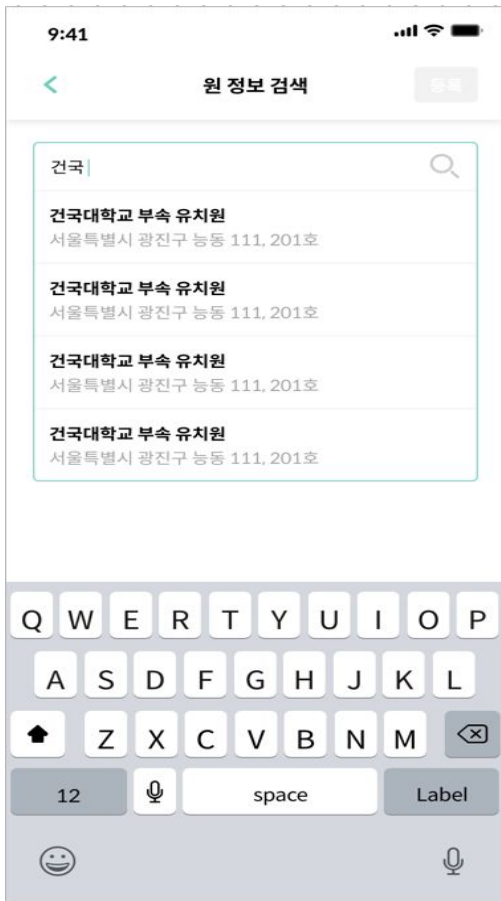
2. Method
- GET

3. URL
/api/district?id={}

4. Query
id = 해당 지역의 id(두번째 지역 호출시 필요, required=false)

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- datas :[{
 ~id : uuid
 ~name : string
}]



1. API 명
- 기관 검색(어린이집, 유치원)
2. Method
- GET
3. URL
/api/institutions?search={}
4. Query
search = 검색하고자 하는 내용(required=true)

5.Return Type

- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)
- datas : [{
 ~ id : uuid
 ~ name : string,
 ~ admin : string,
 ~ tel : string,
 ~ district_one_id : uuid,
 ~ district_two_id : uuid,
 ~ district_one_name : string,
 ~ district_two_name : string,
 ~ address : string
}]

search 에 빈칸이 들어갈 경우
빈데이터 반환

변경 예정



1. API 명
- 쿼리 조회(어린이집, 유치원)
2. Method
- GET
3. URL
/api/quick?member_id={member_id}

```
"announcement": {  
  "id": (uuid),  
  "date": (string),  
  "condition": (int),  
  "meal": (int),  
  "temperature": (int),  
  "sleep": (string),  
  "defecation": (int),  
  "member_id": (uuid),  
  "description": (string),  
  "is_confirmed": (boolean)  
},  
"attendance": {  
  "id": (uuid),  
  "date": (string),  
  "status": (int),  
  "description": (string),  
  "member_id": (uuid),  
  "is_confirmed": (boolean)  
},  
"handover": {  
  "id": (uuid),  
  "date": (string),  
  "name": (string),  
  "relation": (string)
```

5.Return Type

- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용

- resultCode : 기본(200), 에러(400 이상)

-

```
"data": [{  
  "member_id": (uuid),  
  "member_nickname": (string),  
  "member_description": (string),  
  "member_phone": (string),  
  "member_image": (string),  
  "member_grade": (int),  
  "user_id": (uuid),  
  "kid_id": (uuid),  
  "class_name": (string),  
  "school_name": (string),  
  "drug": {  
    "id": (uuid),  
    "symptom": (string),  
    "schedule": (string),  
    "storage_method": (string),  
    "dosage": (string),  
    "medical_type": (string),  
    "times": (string),  
    "note": (string),  
    "member_id": (uuid),  
    "file_id": (uuid),  
    "is_confirmed": ,  
    "date": (string)
```

```
  },  
  "date": (string)  
},  
}
```

9:41

알림장

전송

개나리반 김민정(민정아 쌤)

건국대학교 부속 유치원

임시 저장 불러오기

임시 저장

2020.05.04.수요일

알림장 내용이 입력됩니다. 김민정 학생의 내용이 이렇게 입력됩니다.

사진 & 동영상 (최대 50개의 사진, 1개의 동영상 첨부 가능)

3:25

오늘의 기록

기본

좋음

보통

나쁨

체온

정상

미열

고열

식사

조금

정량

많이

배변

안함

설사

보통

딱딱함

수면

8:00

9:00

10:00

11:00

12:00

Total Sleeping : 01시간 30분

- API 명
- 알림장 생성
- Method
- POST
- URL
/api/announce
- body
{
 ~ member_id(본인 member_id)
 ~ to_member_id(받는 사람의 member_id)
 ~ date(type : dateOnly, ex. '2021-06-15')
 ~ sleep(type : int)
 ~ condition(type : int, 0 좋음, 1 보통, 2 나쁨)
 ~ meal(type : int, 0 조금, 1 정량, 2 많이)
 ~ temperature(type : int, 0 정상, 1 미열, 2 고열)
 ~ defecation(type : int, 0 안함, 1 설사, 2 보통, 3 딱딱함)
 ~ is_reserved(type : boolean, 예약의 경우 true)
 ~ is_published(type : boolean, 임시저장의 경우 false)
 ~ is_record(type : boolean, 오늘의 기록 활성화시 true)
 ~ file_list(type : array<id>, file의 id 배열)
 ~ content(type:string, 전하고자 하는 내용)
}

5.Return Type - JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```

"data": {
  "id": (uuid)
  "date": (string) ex.("2021-06-16"),
  "content": (string),
  "meal": (int),
  "temperature": (int),
  "is_record": (boolean),
  "sleep": (int),
  "defecation": (int),
  "is_confirmed": (boolean),
  "is_reserved": (boolean),
  "is_published": (boolean),
  "member_nickname": (string),
  "member_id": (uuid),
  "to_member_nickname": (string),
  "to_member_id": (uuid),
  "school_id": (uuid),
  "class_id": (uuid),
  "school_name": (string),
  "class_name": (string),
}
```

to_member_id 처리
보호자의 경우 해당하는 교사
교사의 경우 해당하는 보호자

9:41

알림장

전송

개나리반 김민정(민정아 쌤)

건국대학교 부속 유치원

임시 저장 불러오기

임시 저장

2020.05.04.수요일

알림장 내용이 입력됩니다. 김민정 학생의 내용이 어떻게 입력됩니다.

사진 & 동영상 (최대 50개의 사진, 1개의 동영상 첨부 가능)

3:25

오늘의 기록

기분

좋음

보통

나쁨

체온

정상

미열

고열

식사

조금

정량

많이

배변

안함

설사

보통

똥딱함

수면

8:00

9:00

10:00

11:00

12:00

Total Sleeping : 01시간 30분

- API 명
- 알림장 수정
- Method
- POST
- URL
/api/announce/update
- body
{
 ~ member_id(본인 member_id)
 ~ to_member_id(교사의 경우만 필요, 보호자의 member_id)
 ~ id (type : uuid, 해당하는 알림장 id)
 ~ date(type : dateOnly, ex. '2021-06-15')
 ~ sleep(type : int)
 ~ condition(type : int, 0 좋음, 1 보통, 2 나쁨)
 ~ meal(type : int, 0 조금, 1 정량, 2 많이)
 ~ temperature(type : int, 0 정상, 1미열, 2고열)
 ~ defecation(type : int, 0 안함, 1설사, 2 보통, 3똥딱함 ,
 ~ is_reserved(type : boolean , 예약의 경우 true)
 ~ is_published(type : boolean, 임시저장의 경우 false)
 ~ is_record(type : boolean, 오늘의 기록 활성화시 true)
 ~ file_list(type : array<id>, file의 id 배열)
 ~ deleted_list(type : array<id>, file의 id 배열)
 ~ content(type:string, 전하고자 하는 내용)
 }

5.Return Type - JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본 (200), 에러 (400 이상)

```

"data": {
  "id": (uuid)
  "date": (string) ex. ("2021-06-16"),
  "content": (string),
  "meal": (int),
  "temperature": (int),
  "is_record": (boolean),
  "sleep": (int),
  "defecation": (int),
  "is_confirmed": (boolean),
  "is_reserved": (boolean),
  "is_published": (boolean),
  "member_nickname": (string),
  "member_id": (uuid),
  "to_member_nickname": (string),
  "to_member_id": (uuid),
  "school_id": (uuid),
  "class_id": (uuid),
  "school_name": (string),
  "class_name": (string),
}
```

수정시에 기존데이터를 가져올 경우
수면시간의 경우 어떤 시간에 잤는지를 확인이
불가하지만
기획서에는 어떤 시간에 잤는지는 표시하지
않기는함

9:41

<

개나리반 남선화 선생님

>

알림장

투약의뢰서

결석

인계자변경

<

2021 4월

>

총 4개의 알림장이 있습니다.

삭제

2021.05.22

예약

가정 > 원

알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장...

더보기

2021.04.01

원 > 가정

알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장...

더보기

2021.05.22

가정 > 원

알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장...

더보기

2021.04.01

원 > 가정

알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장...

더보기

- API 명
- 알림장 조회
- Method
- GET
- URL
/api/announce
- query
{
~ temp(type : boolean, 임시자장을 불러올 경우 true, required= false),
~ member_id(type : uuid, 본인의 member_id, required =true),
~ to_member_id(type : uuid, 교사의 경우만 필요, 보호자의 member_id, required=false)
~ date(type: string ex . 2020-01 ,없을 경우 오늘 날짜 기준의 달로 뽑아서 리턴)
}

sender 가 0이면 가정 > 원
1이면 원 > 가정

5.Return Type - JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본 (200), 에러 (400 이상)

```

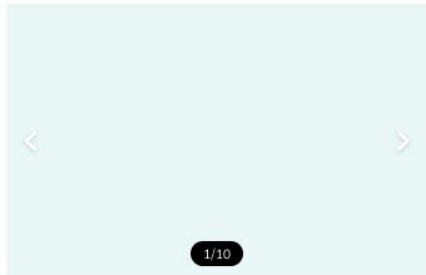
"datas": [{
  "id": (uuid) ex . "cb52ad50-e511-11eb-9422-9571ef677d70",
  "date": (string) ex . "2021-07-15",
  "content": (string) ex . "건강합니다.",
  "sender": (int) ex . 0,
  "meal": (int) ex . 1,
  "temperature": (int) ex . 1,
  "is_record": (boolean) ex . true,
  "sleep": (int) ex . 1,
  "defecation": (int) ex . 1,
  "is_confirmed": (boolean) ex . false,
  "class_name": (string) ex . "테스트 0 반",
  "member_id": (uuid) ex . "b29dbfd0-ca71-11eb-a280-1d94c8c40ebb",
  "to_member_id": (uuid) ex . "11a4b5a0-deeb-11eb-8e61-21e895c60e6e",
  "class_id": (uuid) ex . "b2804cc0-ca71-11eb-a280-1d94c8c40ebb",
  "school_name": (string) ex . "테스트 어린이집",
  "school_id": (uuid) ex . "b27b91d0-ca71-11eb-a280-1d94c8c40ebb",
  "chat_id": (uuid) ex . "cbb5dbf0-e511-11eb-9422-9571ef677d70",
  "is_published": (boolean) ex . true,
  "sender": (int) ex . 0,
  "images": [
    {
      "id":(uuid) ex . "c148da70-e500-11eb-889c-51b8b542fb99",
      "address": (string) ex .
        "https://s3lulla.s3.ap-northeast-2.amazonaws.com/lulla_1626307696706__58_.jpg",
      "size": (int) ex . 248338,
    }
  ]
}

```




알림장

2020.05.02 수요일



민정이는 오늘 하루 잘 보냈습니다. 오늘은 성훈이와 로봇을 가지고 놀았어요. 민정이는 인형과 소꿉놀이보다는 블록과 로봇에 더 관심이 많습니다. 민정이는 오늘 하루 잘 보냈습니다. 오늘은 성훈이와 로봇을 가지고 놀았어요. 민정이는 인형과 소꿉놀이보다는 블록과 로봇에 더 관심이 많습니다. 민정이는 오늘 하루 잘 보냈습니다. 오늘은 성훈이와 로봇을 가지고 놀았어요. 민정이는 인형과 소꿉놀이보다는 블록과 로봇에 더 관심이 많습니다.



나쁨



미열



많이



설사



1시간 30분

확인

1. API 명
- 알림장 상세조회
2. Method
- GET
3. URL
/api/announce
4. query
{
~ member_id(type : uuid, 본인의 member_id, required =true),
~ id(type : uuid, 해당하는 알림장의 id)
}

5.Return Type

- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

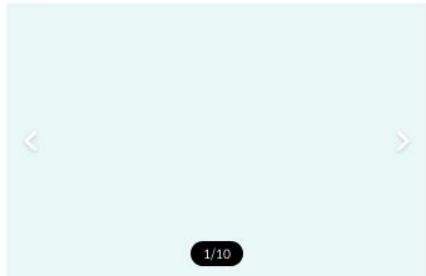
```
"data": {
  "id": (uuid) ex . "cb52ad50-e511-11eb-9422-9571ef677d70",
  "date": (string) ex . "2021-07-15",
  "content": (string) ex . "건강합니다.",
  "sender": (int) ex . 0,
  "meal": (int) ex . 1,
  "temperature": (int) ex . 1,
  "is_record": (boolean) ex . true,
  "sleep": (int) ex . 1,
  "defecation": (int) ex . 1,
  "is_confirmed": (boolean) ex . false,
  "is_reserved": (boolean) ex . false,
  "is_published": (boolean) ex . true,
  "member_nickname": (string) ex . "테스트 0 반 재민(보호자)",
  "member_id": (uuid) ex . "b29dbfd0-ca71-11eb-a280-1d94c8c40ebb",
  "to_member_nickname": (string) ex . "테스트 0 반 test user ",
  "to_member_id": (uuid) ex . "11a4b5a0-deeb-11eb-8e61-21e895c60e6e",
  "school_id": (uuid) ex . "b27b91d0-ca71-11eb-a280-1d94c8c40ebb",
  "class_id": (uuid) ex . "b2804cc0-ca71-11eb-a280-1d94c8c40ebb",
  "sender": (int) ex . 0,
  "school_name": (string) ex . "테스트 어린이집",
  "class_name": (string) ex . "테스트 0 반",
  "images": [
    {
      "id": (uuid) ex . "c148da70-e500-11eb-889c-51b8b542fb99",
      "address": (string) ex .
```

9:41



알림장

2020.05.02 수요일



민정이는 오늘 하루 잘 보냈습니다. 오늘은 성훈이와 로봇을 가지고 놀았어요. 민정이는 인형과 소꿉놀이보다는 블록과 로봇에 더 관심이 많습니다. 민정이는 오늘 하루 잘 보냈습니다. 오늘은 성훈이와 로봇을 가지고 놀았어요. 민정이는 인형과 소꿉놀이보다는 블록과 로봇에 더 관심이 많습니다. 민정이는 오늘 하루 잘 보냈습니다. 오늘은 성훈이와 로봇을 가지고 놀았어요. 민정이는 인형과 소꿉놀이보다는 블록과 로봇에 더 관심이 많습니다.



나쁨



미열



많이



설사



1시간 30분

확인

1. API 명
- 알림장 확인
2. Method
- POST
3. URL
/api/announce/confirm
4. body
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ id(type : uuid, 해당하는 알림장의 id)
}

5.Return Type

- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```
"data":{
  "id": (uuid)
  "date": (string) ex.("2021-06-16"),
  "content": (string),
  "meal": (int),
  "temperature": (int),
  "is_record": (boolean),
  "sleep": (int),
  "defecation": (int),
  "is_confirmed": (boolean),
  "is_reserved": (boolean),
  "is_published": (boolean),
  "member_nickname": (string),
  "member_id": (uuid),
  "to_member_nickname": (string),
  "to_member_id": (uuid),
  "school_id": (uuid),
  "class_id": (uuid),
  "school_name": (string),
  "class_name": (string),
}
```

9:41

<

개나리반 남선화 선생님

발라 어린이집

알림장

투약의뢰서

결석

인계자변경

<

2021 4월

>

총 4개의 알림장이 있습니다.

삭제

2021.05.22

예약

가정 > 원

알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장...

더보기

2021.04.01

원 > 가정

알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장...

더보기

2021.05.22

가정 > 원

알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장...

더보기

2021.04.01

원 > 가정

알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장 내용이 입력됩니다. 알림장...

더보기

- API 명
- 알림장 삭제
- Method
- POST
- URL
/api/announce/delete
- body
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ id(type : array or uuid 해당하는 id 리스트 , 단일 id)
}

5.Return Type

- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

9:41 등하원 알림 전송

개나리반 김민정(민정아 쌤)
건국대학교 부속 유치원

☒ 결석

결석 사유를 입력하세요.

☐ 인계자 변경

9:41 등하원 알림 전송

개나리반 김민정(민정아 쌤)
건국대학교 부속 유치원

☒ 결석

☒ 인계자 변경

인계자 성명

원아와의 관계(ex. 할아버지)

인계자 연락처

1. API 명
- 등하원 알림 생성
2. Method
- POST
3. URL
/api/attend
4. body
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ date(type : onlydate, ex. '2021-06-15')
 ~ type(type : int, 1 결석, 2 인계자변경)
 ~ description(type : string, 결석 사유 내용)
 ~ phone(type:string, 인계자 연락처)
 ~ name(type:string, 인계자 이름)
 ~ relation(type : string, 인계자와 원아의 관계)
 ~ signature_id(type : uuid, 서버에 올려진 서명 파일 id)
}

```
"data": {
  "id": (uuid),
  "date": (string), ex. "2021-06-15",
  "kid_name": (string),
  "description": (string),
  "is_confirmed": (boolean),
  "type": (int),
  "class_name": (string),
  "class_id": (uuid),
  "school_name": (string),
  "school_id": (uuid),
  "kid_id": (uuid),
  "member_id": (uuid)
}
```

결석

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```
"data": {
  "id": (uuid),
  "date": (string), ex. "2021-06-15",
  "kid_name": (string),
  "is_confirmed": (boolean),
  "type": (int),
  "class_name": (string),
  "class_id": (uuid),
  "school_name": (string),
  "school_id": (uuid),
  "kid_id": (uuid),
  "member_id": (uuid),
  "relation": (string),
  "phone": (string),
  "name": (string),
  "signature_id": (uuid),
  "signature_image": (string)
}
```

인계자변경

9:41

신호 아이콘 배터리 아이콘

개나리반 남선화 선생님

할라 어린이집

알림장

투약의뢰서

결석

인계자변경

< 2021 4월 >

총 4 개의 인계자 변경이 있습니다.

[삭제](#)

2021.04.01

인계자 : 김춘삼

2021.04.01

인계자 : 김춘삼

1. API 명
- 등하원 알림장 조회
2. Method
- GET
3. URL
/api/attend
4. query
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ type(type:int, 1은 결석, 2는 인계자변경, 없을시 자동으로 1)
 ~ guardian_id(type : uuid, 보호자의 member_id, 교사만 필수)
 ~ date(string ex '2021-04')
}

```

"datas": [{
  "id": (uuid),
  "date": (string), ex. "2021-06-15",
  "kid_name": (string),
  "description": (string),
  "is_confirmed": (boolean),
  "type": (int),
  "class_name": (string),
  "class_id": (uuid),
  "school_name": (string),
  "school_id": (uuid),
  "kid_id": (uuid),
  "member_id": (uuid)
}]

```

결석

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```

"datas": [{
  "id": (uuid),
  "date": (string), ex. "2021-06-15",
  "kid_name": (string),
  "is_confirmed": (boolean),
  "type": (int),
  "class_name": (string),
  "class_id": (uuid),
  "school_name": (string),
  "school_id": (uuid),
  "kid_id": (uuid),
  "member_id": (uuid),
  "relation": (string),
  "phone": (string),
  "name": (string),
  "signature_id": (uuid),
  "signature_image": (string)
}]

```

인계자변경

9:41

<

인계자 변경

2020.05.02 수요일

원아명

김민정

인계자

김춘삼

원아와 관계

할아버지

인계자 연락처

010-1111-2222

보호자 사인



확인 완료

1. API 명
- 등하원 알림장 상세조회
2. Method
- GET
3. URL
/api/attend
4. query
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ id(type : uuid, 해당하는 알림장 id)
}

```

"data": {
  "id": (uuid),
  "date": (string), ex. "2021-06-15",
  "kid_name": (string),
  "description": (string),
  "is_confirmed": (boolean),
  "type": (int),
  "class_name": (string),
  "class_id": (uuid),
  "school_name": (string),
  "school_id": (uuid),
  "kid_id": (uuid),
  "member_id": (uuid)
}

```

결석

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```

"data": {
  "id": (uuid),
  "date": (string), ex. "2021-06-15",
  "kid_name": (string),
  "is_confirmed": (boolean),
  "type": (int),
  "class_name": (string),
  "class_id": (uuid),
  "school_name": (string),
  "school_id": (uuid),
  "kid_id": (uuid),
  "member_id": (uuid),
  "relation": (string),
  "phone": (string),
  "name": (string),
  "signature_id": (uuid),
  "signature_image": (string)
}

```

인계자변경

9:41

<인계자 변경


2020.05.02 수요일

원아명
김민정

인계자
김춘삼

원아와 관계
할아버지

인계자 연락처
010-1111-2222

보호자 사인


확인 완료

1. API 명
- 등하원 알림장 확인
2. Method
- GET
3. URL
/api/attend/confirm
4. body
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ id(type : uuid, 해당하는 알림장 id)
}

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```
"datas": [{  
  "id": (uuid),  
  "date": (string), ex. "2021-06-15",  
  "kid_name": (string),  
  "description": (string),  
  "is_confirmed": (boolean),  
  "type": (int),  
  "class_name": (string),  
  "class_id": (uuid),  
  "school_name": (string),  
  "school_id": (uuid),  
  "kid_id": (uuid),  
  "member_nickname": (string),  
  "member_id": (uuid)  
}]
```

결석

```
"datas": [{  
  "id": (uuid),  
  "date": (string), ex. "2021-06-15",  
  "kid_name": (string),  
  "is_confirmed": (boolean),  
  "type": (int),  
  "class_name": (string),  
  "class_id": (uuid),  
  "school_name": (string),  
  "school_id": (uuid),  
  "kid_id": (uuid),  
  "member_nickname": (string),  
  "member_id": (uuid),  
  "relation": (string),  
  "phone": (string),  
  "name": (string),  
  "signature_id": (uuid),  
  "signature_image": (string)  
}]
```

인계자변경



1. API 명
- 등하원 알림장 삭제
2. Method
- POST
3. URL
/api/attend/delete
4. body
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ id(type : array or id, 해당하는 알림장 array<id> or id)
}

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

9:41

<
투약 의뢰서
전송

개나리반 김민정(민정이짱)
건국대학교 부속 유치원

2020.05.04 수요일

1

최근 투약 의뢰서

투약 의뢰서 1

감기

식후 30분

실은 보관

물약, 가루약

5ml

1회

민정이 감기와 비타민 투약 의뢰드립니다. 점심약
했었으니 확인 부탁드립니다.

+
투약 의뢰 추가

- API 명
- 투약의뢰서 생성
- Method
- POST
- URL
/api/drug
- body


```
{
  ~ member_id(type : uuid, 본인의 member_id, required =true),
  ~ date(type : dateonly, ex. "2021-06-15")
  ~ signature_id(type : uuid, 서명 파일 id)
  ~ is_reserved(type : boolean, 예약을 원할 경우에는 true)
  ~ is_published(type: boolean, 임시저장의 경우에는 false)
  ~ drug : [{
    ~ symptom(type : string, 증상)
    ~ schedule(type : string, 투약 시간)
    ~ storage_method(type : string, 보관 방법)
    ~ dosage(type : string, 용량)
    ~ medical_type(type : string, 약 종류)
    ~ times( type : string, 횟수)
    ~ content(type : string, 전달사항)
  }]
}
```

5.Return Type

- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```
"data": {
  "id": (uuid) ex . "f18c6600-e51b-11eb-b38f-a1877464f413",
  "date": (string) ex . "2021-07-15",
  "is_reserved": false,
  "is_published": true,
  "drug": [
    {
      "id": (uuid) ex . "f1905da0-e51b-11eb-b38f-a1877464f413",
      "symptom": (string) ex . "감기",
      "schedule": (string) ex . "1qjs",
      "storage_method": (string) ex . "서늘한 곳",
      "dosage": (string) ex . "1ml",
      "is_confirmed": false,
      "medical_type": (string) ex . "물약",
      "name": (string) ex . "재민",
      "signature_id": (uuid) ex . "f1905da0-e51b-11eb-b38f-a1877464f413",
      "signature_image": (string) ex . null,
      "content": (string) ex ."이상없음",
      "times": (string) ex . "1회",
      "performer": (string) ex . ""
    }
  ]
},
```



1. API 명
- 투약의뢰서 조회
2. Method
- GET
3. URL
/api/drug
4. query
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ guardian_id(type : uuid, 보호자의 member_id, 교사만 필수)
 ~ date(type string ex. '2020-01')
}

5.Return Type

- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본 (200), 에러 (400 이상)

```
datas : [ {
  "id": (uuid) ex . "f18c6600-e51b-11eb-b38f-a1877464f413",
  "date": (string) ex . "2021-07-15",
  "is_reserved": false,
  "is_published": true,
  "drug": [
    {
      "id": (uuid) ex . "f1905da0-e51b-11eb-b38f-a1877464f413",
      "symptom": (string) ex . "감기",
      "schedule": (string) ex . "1qjs",
      "storage_method": (string) ex . "서늘한 곳",
      "dosage": (string) ex . "1ml",
      "is_confirmed": false,
      "medical_type": (string) ex . "물약",
      "name": (string) ex . "재민",
      "signature_id": (uuid) ex . "f1905da0-e51b-11eb-b38f-a1877464f413",
      "signature_image": (string) ex . null,
      "content": (string) ex . "이상없음",
      "times": (string) ex . "1회",
      "performer": (string) ex . ""
    }
  ]
}, ...]
```

9:41

<

투약 의뢰서

2020.05.02 수요일

2개의 투약의뢰서가 있습니다.

1

2

원아명

김민정

증상

감기

투약 시간

식후 30분

보관방법

실온

종류

물약, 가루약

용량

5ml


횟수

1회

전달사항

민정이 감기약과 비타민 투약의뢰드립니다. 점심약
경겼으니 확인 부탁드립니다.

보호자 사인



이치은(투약자 입력 필수)

투약

- API 명
- 투약의뢰서 상세조회
- Method
- GET
- URL
/api/drug
- query
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ id(type : uuid, 해당하는 투약의뢰서 id)
}

5.Return Type - JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본 (200), 에러 (400 이상)

```
{
  "id": (uuid) ex . "f18c6600-e51b-11eb-b38f-a1877464f413",
  "date": (string) ex . "2021-07-15",
  "is_reserved": false,
  "is_published": true,
  "drug": [
    {
      "id": (uuid) ex . "f1905da0-e51b-11eb-b38f-a1877464f413",
      "symptom": (string) ex . "감기",
      "schedule": (string) ex . "1qjs",
      "storage_method": (string) ex . "서늘한 곳",
      "dosage": (string) ex . "1ml",
      "is_confirmed": false,
      "medical_type": (string) ex . "물약",
      "name": (string) ex . "재민",
      "signature_id": (uuid) ex . "f1905da0-e51b-11eb-b38f-a1877464f413",
      "signature_image": (string) ex . null,
      "content": (string) ex . "이상없음",
      "times": (string) ex . "1회",
      "performer": (string) ex . ""
    }
  ]
}
```

9:41

<

투약 의뢰서

2020.05.02 수요일

2개의 투약의뢰서가 있습니다.

1

2

원아명

김민정

증상

감기

투약 시간

식후 30분

보관방법

실온

종류

물약, 가루약

용량

5ml


횟수

1회

전달사항

민정이 감기약과 비타민 투약의뢰드립니다. 점심약
경겼으니 확인 부탁드립니다.

보호자 사인



이치은(투약자 입력 필수)

투약

1. API 명
- 투약의뢰서 확인
2. Method
-POST
3. URL
/api/drug/confirm
4. body
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ id(type : uuid, 해당하는 투약의뢰서 안에 있는 drug 개별의 id)
 ~ performer(type : string, 투약자 이름 입력)
}

5.Return Type - JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```
data :{
  "id": (uuid),
  "date": string ex. "2021-06-15",
  "is_reserved": (boolean),
  "is_published": (boolean),
  "signature_id": null,
  "signature_image": null,
  "drug": [
    {
      "id": (uuid),
      "symptom": (string),
      "schedule": (string),
      "storage_method": (string),
      "dosage": (string),
      "is_confirmed": (boolean),
      "medical_type": (string),
      "times": (string),
      "note": (string),
      "performer": "",
    }
  ]
}
```

9:41



투약 의뢰서

2020.05.02 수요일

2개의 투약의뢰서가 있습니다.

1

2

원아명

김민정

증상

감기

투약 시간

식후 30분

보관방법

실온

종류

물약, 가루약

용량

5ml

횟수

1회

전달사항

민정이 감기약과 비타민 투약의뢰드립니다. 점심약
정겼으니 확인 부탁드립니다.

보호자 사인

이치은(투약자 입력 필수)

투약

1. API 명
- 투약의뢰서 삭제
2. Method
-POST
3. URL
/api/drug/delete
4. body
{
 ~ member_id(type : uuid, 본인의 member_id, required =true),
 ~ id(type : array or id, 해당하는 투약의뢰서 id)
}

5.Return Type

- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용

- resultCode : 기본(200), 에러(400 이상)

9:41

<

글쓰기

완료

임시 저장 불러오기

임시 저장

전체

▼

제목을 입력하세요. (공백 포함 최대 24자)

중요 글지 활성화

내용을 입력하세요.

모든 게시물은 임시저장이 가능합니다.

사진 & 동영상

+

- API 명
- 앨범 생성
- Method
- POST
- URL
/api/album
- Body(Request)
- {
 ~ album : {
 ~title(string)
 ~contents(string)
 ~important(boolean)
 ~is_published(boolean)
 ~allowed_class([:클래스 ID])
 ~allowed_member([:멤버 ID])
 ~file_list([:파일 ID])
 }
 }
 ~ member_id(멤버 아이디)
}

원장일 경우 글의 대상선택이 전체일 경우에는
allowed_class와 **allowed_member**값을 보내지 않으면 됨
데이터 반환 형식은 상세앨범 API와 동일

- Return Type
- JSON

6.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 예러(400 이상)
- data : {
 "id": (uuid),
 "title": (string),
 "contents": (string),
 "is_modified": (boolean),
 "published_time": (date),
 "important": (boolean),
 "author_id": (uuid),
 "created_at": (string),
 "like_count": (int),
 "comment_count": (int),
 "is_like": (boolean),
 "is_bookmark": (boolean),
 "author_nickname": (string),
 "allowed_member": [],
 "allowed_class": [],
 "class_name": (string),
 "school_name": (string),
 "author_type": (string),
 "author_grade": (int),
 "images": [{
 "id": (uuid),
 "address": (string),
 "size": (int),
 "thumbnail_address": (string),
 "type": (string),
 "name": (string),
 "width": (int),
 "height": (int),
 "index": (int),
 "length": (int)
 }],
 "author_image": (string)
}

9:41

<

글쓰기

완료

임시 저장 불러오기

임시 저장

전체

▼

제목을 입력하세요. (공백 포함 최대 24자)

중요 글자 활성화

내용을 입력하세요.
모든 게시물은 임시저장이 가능합니다.

사진 & 동영상

+

1. API 명
- 앨범 삭제
2. Method
- POST
3. URL
/api/album/delete
4. Body(Request)
- {
 ~ album : {
 ~ id (uuid)
 }
 ~ member_id(멤버 아이디)
}

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

9:41
< 글쓰기
완료

임시 저장 불러오기

임시 저장

전체

▼

제목을 입력하세요. (공백 포함 최대 24자)

중요 글지 활성화

내용을 입력하세요.

모든 게시물은 임시저장이 가능합니다.

사진 & 동영상

+

- API 명
- 앨범 전 조회
- Method
-GET
- URL
/api/album
- Query(Request)
- member_id

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```

"data": {
  "album": [
    {
      "id": (uuid),
      "title": (string),
      "contents": (string),
      "is_modified": (boolean),
      "published_time": (string) ex. "2021-06-28T12:34:04.330Z",
      "important": (boolean),
      "author_id": (uuid),
      "created_at": (string) ex. "2021-06-28T12:34:04.330Z",
      "like_count": (int),
      "comment_count": (int),
      "is_like": (boolean),
      "is_bookmark": (boolean),
      "author_nickname": (string),
      "allowed_member": [{
        "nickname": (string),
        "id": (uuid),
        "member_image": (string) ,
        "kid_name": (string),
        "class_name": (string) },...],
      "allowed_class": [{
        "name": (string),
        "id": (uuid) ,
        "allowed_member" : (위의 allowed_member 와 동일) }, ...],
      "class_name": (string),
      "school_name": (string),
      "author_type": (string),
      "author_grade": (int),
      "images": [
        {

```

9:41
<
글쓰기
완료

임시 저장 불러오기

임시 저장

전체

▼

제목을 입력하세요. (공백 포함 최대 24자)

중요 글지 활성화

내용을 입력하세요.

모든 게시물은 임시저장이 가능합니다.

사진 & 동영상

+

- API 명
- 앨범 상세조회
- Method
-GET
- URL
/api/album
- Query(Request)
- member_id
- id (album id)

5.Return Type
- JSON

6.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(200), 에러(400 이상)

```

"data": {
  "album": {
    "id": (uuid),
    "title": (string),
    "contents": (string),
    "is_modified": (boolean),
    "published_time": (string) ex. "2021-06-28T12:34:04.330Z",
    "important": (boolean),
    "author_id": (uuid),
    "created_at": (string) ex. "2021-06-28T12:34:04.330Z",
    "like_count": (int),
    "comment_count": (int),
    "is_like": (boolean),
    "is_bookmark": (boolean),
    "author_nickname": (string),
    "allowed_member": [{
      "nickname": (string),
      "id": (uuid),
      "member_image":(string) ,
      "kid_name": (string),
      "class_name":(string) },...],
    "allowed_class": [{
      "name": (string),
      "id": (uuid) ,
      "allowed_member" : (위의 allowed_member 와 동일) }, .....],
    "class_name": (string),
    "school_name": (string),
    "author_type": (string),
    "author_grade": (int),
    "images": [
      {
        "id": (uuid),

```

9:41



댓글

삭제된 댓글입니다.



장미반 김유진 선생님

댓글이 입력됩니다. 이렇게요. 댓글이 입력됩니다. 이렇게요. 댓글이 입력됩니다. 이렇게요. 댓글이 입력됩니다. 이렇게요. 댓글이 입력됩니다. 이렇게요. 댓글이 입력됩니다.

2시간 전 [답글 달기](#)



개나리반 남선화 선생님

...

댓글이 입력됩니다. 이렇게요. 댓글이 입력됩니다. 이렇게요. 댓글이 입력됩니다. 이렇게요. 댓글이 입력됩니다. 이렇게요. 댓글이 입력됩니다. 이렇게요. 댓글이 입력됩니다. 이렇게요.

10시간 전 [답글 달기](#)

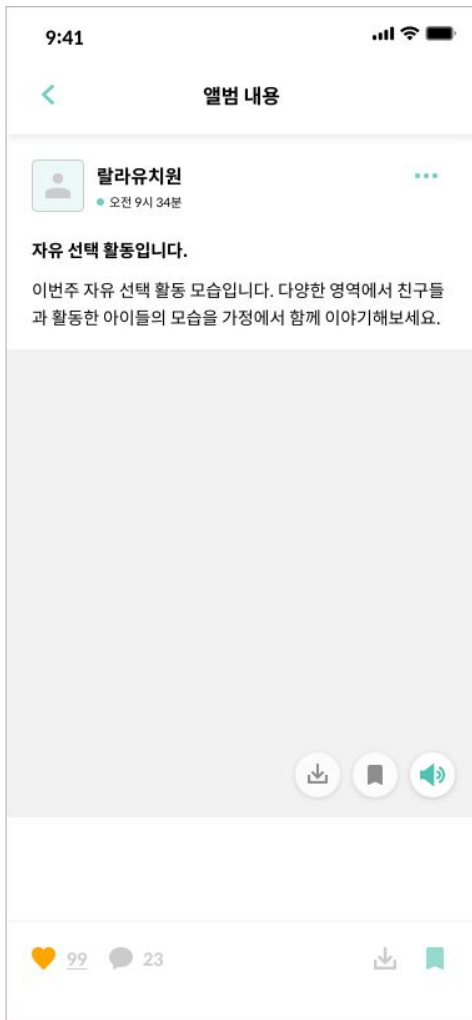
댓글을 입력하세요.



1. API 명
- 댓글 생성
2. Method
- POST
3. URL
/api/album/comment
4. Body(Request)
- {
 ~ comment : {
 ~ album_id
 ~ content(string)
 }
 ~ member_id(string)
}
}
5. Query
post : 해당 공지 ID

6.Return Type
- JSON

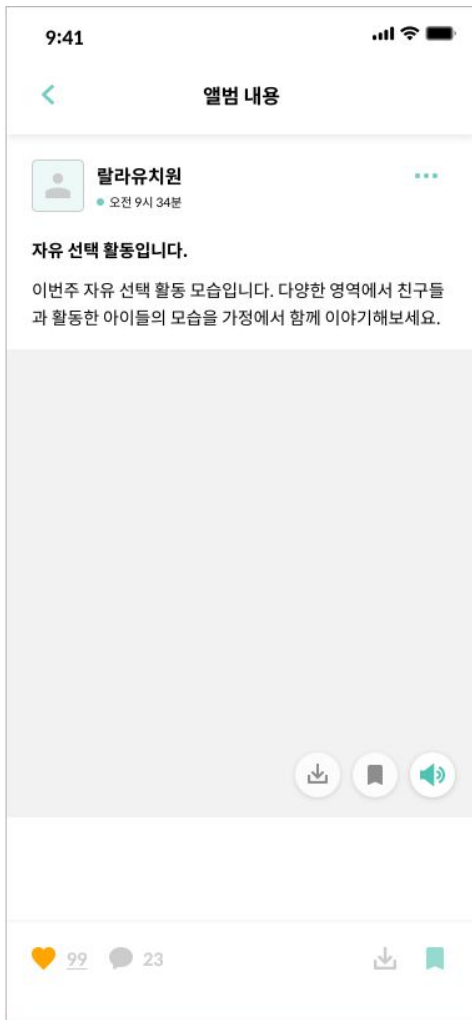
7.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 에러(400 이상)
- data : {
 ~ id(uuid)
 ~ content(uuid)
 ~ member_id(uuid)
 ~ post_id(uuid)
}



1. API 명
- 좋아요 생성
2. Method
- POST
3. URL
/api/album/like
4. Body(Request)
- {
 ~ like : {
 ~ album_id
 }
 ~ member_id(string)
}

6.Return Type
- JSON

7.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 에러(400 이상)



1. API 명
- 앨범 북마크 생성
2. Method
- POST
3. URL
/api/album/bookmark
4. Body(Request)
- {
 ~ bookmark : {
 ~ album_id
 }
 ~ member_id(string)
}

6.Return Type
- JSON

7.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 에러(400 이상)

9:41

<

바나나반

선생님

원아(보호자)

원아명

입학처

입학처

조회

1. API 명
- 초대장 생성
2. Method
- POST
3. URL
/api/invitation
4. Body(Request)
- {
 ~ invitation : {
 ~ phone(string) 핸드폰번호(필수)
 ~ role_name(string) 역할 명(관리자, 교사 필요)
 ~ kid_name(string) 원아 명(보호자 생성시 필요)
 ~ class_id(string) 반 id (교사, 보호자, 가족만 필요)
 ~ kid_id(string) 원아 id (가족 초대시만 필요)
 ~ type(int) (2 : 관리자, 3 : 정교사, 4 : 보조교사, 5 : 보호자, 6 : 가족)
 }
 ~ member_id(string)
}

6.Return Type

- JSON

7.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 에러(400 이상)

```
"data": {  
  "id": (uuid) ex. "5f9ebe30-df87-11eb-a093-2f518c68a555",  
  "confirmed": (boolean) ex.false,  
  "type": (int) ex. 5,  
  "school_id": (uuid) ex. "b27b91d0-ca71-11eb-a280-1d94c8c40ebb",  
  "school_name": (string) ex. "테스트 어린이집",  
  "school_description": (string) ex. "테스트 용으로 생성된 원",  
  "school_address": (string) ex."테스트",  
  "school_tel": (string) ex. "06112341234",  
  "class_id": (string) ex. "b2804cc0-ca71-11eb-a280-1d94c8c40ebb",  
  "class_name": (string) ex. "테스트 0 반",  
  "class_description": (string) ex. "테스트 0 반",  
  "name": (string) ex. "아린",  
  "member_grade": (int) ex .5,  
  "member_type": (string) ex."보호자"  
},
```

9:41

<
원아 정보
등록

원아 정보

할라유치원

하늘반

김민정

2016

4

10

남

여

- API 명
- 초대장 수락
- Method
- POST
- URL
/api/invitation/accept
- Body(Request)
- {
 ~ invitation : {
 ~ id (string) 초대장 id
 ~ gender(int) 0은 남자, 1은 여자(보호자만 필요)
 ~ birth(dateonly) ex `2019-01-02` (보호자만 필요)
 ~ relation(string) 관계 ex) 엄마, 아빠, 할아버지 등...(보호자, 가족 만 필요)
 }
 ~ member_id(string)
}

- Return Type
- JSON

7.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 예러(400 이상)

```

"data": {
  "email": (string) ex. "test@test.co.kr",
  "phone": (string) ex. "01099999999",
  "member_id": (string) ex. "fd8a4b50-df87-11eb-a093-2f518c68a555",
  "nickname": (string) ex. "보호자",
  "member_description": (string) ex. "",
  "class_name": (string) ex. "테스트 0 반",
  "school_name": (string) ex. "테스트 어린이집",
  "member_image": (string) ex.
https://s3lulla.s3.ap-northeast-2.amazonaws.com/lulla\_1624937652713\_\_7448\_\_.png
  "member_image_id": (string) ex . "fd8a4b50-df87-11eb-a093-2f518c68a555",
  "member_background_image": (string) ex
  ."https://s3lulla.s3.ap-northeast-2.amazonaws.com/lulla\_1624937652713\_\_7448\_\_.png"
  "background_image_id": (string) ex . "fd8a4b50-df87-11eb-a093-2f518c68a555",
  "member_type": (string) ex . "보호자",
  "member_grade": (int) ex. 5,
  "user_id": (uuid) ex . "0b33b890-dee8-11eb-b89b-536d925097c4",
  "school_id": (uuid) ex . "b27b91d0-ca71-11eb-a280-1d94c8c40ebb",
  "class_id": (uuid) ex. "b2804cc0-ca71-11eb-a280-1d94c8c40ebb",
  "school_role_id": (uuid) ex. "5f9d85b0-df87-11eb-a093-2f518c68a555",
  "is_admin": (boolean) ex. false,
  "chat_active": (boolean) ex.false,
  "member_nickname": (string) ex. "테스트 0 반 아린(보호자)",
  "kid_id": (uuid) ex. "5f9ebe31-df87-11eb-a093-2f518c68a555",
  "kid_name": (string) ex. "아린"
}

```

9:41

<
원아 정보
등록

원아 정보

할라유치원

구름반

김민정

2016
4
10

남
여

- API 명
- 요청장 생성
- Method
- POST
- URL
/api/demand
- Body(Request)
- {
 ~ demand : {
 ~ name(string) 원아 명(보호자 생성시 필요)
 ~ gender(int) 0은 남자, 1은 여자(보호자만 필요)
 ~ birth(dateonly) ex `2019-01-02` (보호자만 필요)
 ~ relation(string) 관계 ex) 엄마, 아빠, 할아버지 등...(보호자, 가족 만 필요)
 ~ class_id(string) 반 id 필수
 ~ school_id(string) 원 id 필수
 ~ type(int) (3 : 정교사, 4 : 보조교사, 5 : 보호자)
 }
 ~ member_id(string)
}

6.Return Type - JSON

7.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용

- resultCode : 기본(201), 에러(400 이상)

```

"datas": [
  {
    "school_id": (uuid) ex. "b27b91d0-ca71-11eb-a280-1d94c8c40ebb",
    "school_name": (string) ex. "테스트 어린이집",
    "class_id": (uuid) ex. "b2822180-ca71-11eb-a280-1d94c8c40ebb",
    "class_name": (string) ex. "테스트 1 반",
    "phone": (string) ex. "01099999999",
    "name": (string) ex. "비니",
    "birth": (string) ex. "2018-01-01",
    "gender": (int) 1
  }
]

```


9:41   

< 초대 승인 요청 정보

원아 명

김민정

원아 성별

여

원아 생년월일

2016 ▾ 4 ▾ 10 ▾

보호자

김민준

원아와의 관계

이빠

거절

승인

1. API 명
- 요청장 수락
2. Method
- POST
3. URL
/api/demand/accept
4. Body(Request)
- {
 ~ demand : {
 ~ id(string) 요청장 id
 }
 ~ member_id(string)
}

6.Return Type
- JSON

7.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 에러(400 이상)

9:41

<

바나나반

선생님

원아(보호자)

역할

연락처

연락처

초대

담임

010-2222-3333

부담임

010-2222-3333

1. API 명
- 초대장 요청장 조회
2. Method
-GET
3. URL
/api/class
4. Query(Request)
- {
 ~ id(해당 반 id)
 ~ member_id(string)
}

6.Return Type
- JSON

7.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 에러(400 이상)

```
"data": {  
  "class_id": (uuid) ex . "b2804cc0-ca71-11eb-a280-1d94c8c40ebb",  
  "class_name": (string) ex . "테스트 0 반",  
  "class_description": (string) ex . "테스트 0 반",  
  "class_start_date": (string) ex . "2020-03-01",  
  "class_end_date": (string) ex . "2020-12-01",  
  "school_id": (uuid) ex . "b27b91d0-ca71-11eb-a280-1d94c8c40ebb",  
  "teacher_invitation": [  
    {  
      "user_id": (uuid) ex . "0b33b890-dee8-11eb-b89b-536d925097c4",  
      "username": (string) ex . "test user",  
      "phone": (string) ex . "01099999999",  
      "id": (uuid) ex . "45bf5910-df83-11eb-9f14-3f3fb8eb795d",  
      "type": (int) ex .5,  
      "name": (string) ex . "담임"  
    }, ...],  
  "teacher_demand": [ {  
    "user_id": (uuid) ex . "0b33b890-dee8-11eb-b89b-536d925097c4",  
    "username": (string) ex . "test user",  
    "phone": (string) ex . "01099999999",  
    "id": (uuid) ex . "45bf5910-df83-11eb-9f14-3f3fb8eb795d",  
    "type": (int) ex .5,  
    "name": (string) ex . "담임"  
  } ],  
  "guardian_invitation": [  
    {  
      "user_id": (uuid) ex . "0b33b890-dee8-11eb-b89b-536d925097c4",  
      "username": (string) ex . "test user",  
      "phone": (string) ex . "01099999999",  
      "id": (uuid) ex . "45bf5910-df83-11eb-9f14-3f3fb8eb795d",  
      "type": (int) ex .5,  
      "name": (string) ex . "아린"
```



1. API 명
- 채팅 가능 멤버 리스트
2. Method
- GET
3. URL
/api/chat/member-list
4. Query(Request)
- {
 ~ member_id(string)
}

6.Return Type
- JSON

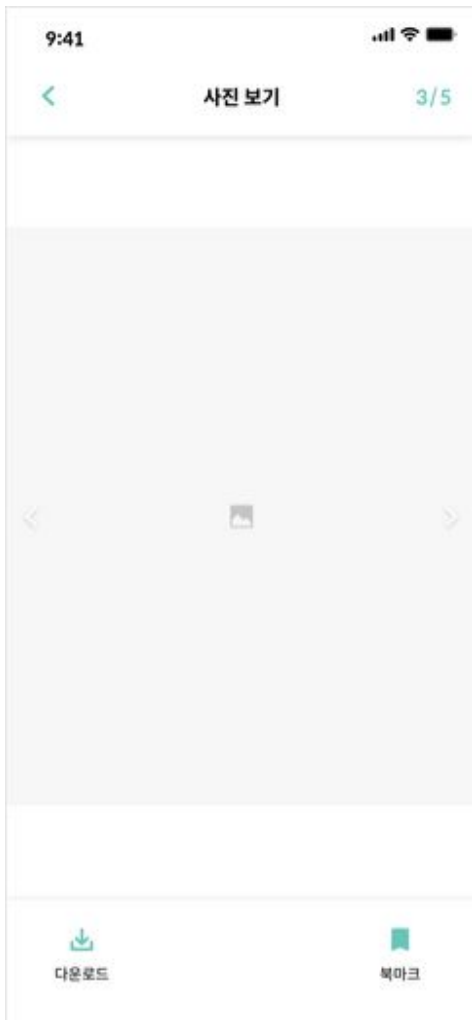
7.Protocol(Response)

- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 에러(400 이상)

```
"data": {
  "id": (uuid) ex . "d39e9410-807b-11eb-bc7c-df4acec7a4d9",
  "name": (string) ex . "테스트 어린이집",
  "admin_list": [
    {
      "id": (uuid) ex . "4f86fea0-deeb-11eb-8e61-21e895c60e6e",
      "member_nickname": (string) ex . "사감 사감",
      "member_description": (string) ex . "",
      "member_image":(string) ex . null,
      "class_name": (string) ex . "테스트 1 반",
      "class_id": (uuid) ex . "d3aac910-807b-11eb-bc7c-df4acec7a4d9",
      "member_grade": (int) ex .5
    },
  ],
  "class_list": [
    {
      "name": (string) ex ."테스트 0 반",
      "id": (uuid) ex . "b2804cc0-ca71-11eb-a280-1d94c8c40ebb",
      "guardian": [
        {
          "id": (uuid) ex ."b29dbfd0-ca71-11eb-a280-1d94c8c40ebb",
          "member_nickname": (string) ex . "테스트 0 반 재민(보호자)",
          "member_description": (string) ex . "",
          "member_image":(string) ex . null,
          "class_name": (string) ex . "테스트 1 반",
          "class_id": (uuid) ex . "d3aac910-807b-11eb-bc7c-df4acec7a4d9",
          "member_grade": (int) ex .5
        },
        ...],
      ]
    }
  ]
}
```

응답데이터에 추가 필요 사항

admin_list와 class_list내에 member 정보에 member_image, member_grade, class_name



1. API 명
- 이미지 북마크
2. Method
- post
3. URL
/api/image/bookmark
4. Body(Request)
- {
 ~ bookmark : {
 ~ image_id 해당하는 이미지의 id
 }
 ~ member_id
}

6.Return Type
- JSON

7.Protocol(Response)
- message : 해당하는 API 결과 메시지 내용
- resultCode : 기본(201), 에러(400 이상)

현재 센드버드 관련 설정

type : MESSG,
custom_type : 'drug', 'announce', 'attend'
data : { id: id(해당하는 콘텐츠의 id)}
message : 도착 메시지, 확인 메시지

user_id : member_id로 사용
nickname : nickname의 경우 서버에서 생성시에는 조합식으로 생성
channel_type : group_channel

현재 콘텐츠 관련 전송 메시지의 순서

1. 서버에서 센드버드에 해당하는 유저가 생성되었는지 확인 요청
2. 없을 시 생성 있을 시 패스
3. 서버에서 센드버드에 해당하는 유저끼리 생성되어있는 그룹 채널이 있는지 확인 요청
4. 없을 시 생성 있을 시 패스
5. 해당하는 custom_type 및 data 와 message로 전송

삭제의 경우(chat 관련해서 데이터를 보관하지만 아직 예외 케이스르 전체 다루지는 못함)

1. db chat 테이블에서 해당하는 콘텐츠의 sendbird_id 및 channel_url을 얻은 후
2. 삭제 요청

요청사항

- post 부분 키 값 제거 및 **datas** 로 변경 - 1월 18일 완료
- **total_count, total_page** 외부로 변경 - 1월 18일 완료
- API 문서 보기편하게 변경(예정)
- **allowed_class** 안에 멤버 정보 필요 - 1월 20일 완료
- 반 선택의 경우 중복 선택이 가능하게 - 1월 20일 완료(class=id,id,id 형식으로 넣으면 됨)
- 키 값 **files** -> **images** 로 변경 - 1월 20일 완료
- **Params** => **Query** 로 변경 - 1월 20일 완료
- **school_role** (grade 1: 원장, 2: 부원장, 3: 담임선생, 4: 보조선생, 5: 보호자)
- GET response 변경 1월 24일

업데이트 목록(1월 18일 이후부터)

- 요청사항(**datas** 로 리스트 삽입, **total_count**, **total_page**) 변경 1월 18일
- 요청사항(**post query** 값으로 **class** 추가, **allowed_class** 안에 해당하는 **member** 반환, **images**로 변경) 1월 20일
- 요청사항(**Params=> Query**로 변경) 1월 20일
- 추가사항(**Survey** 답변에 관한 서비스 로직 생성, 수정 , **Survey** 생성 및 기초 기반 자동 세팅) 1월 20일
- 추가사항(공지 리스트 데이터 형식과 공지 상세 데이터 형식 일치) 1월 20일
- **Response data** 변경(공지, 멤버, 원아, 반, 원, 좋아요, 댓글, 로그인, 회원가입) 1월 24일
- 댓글 관련(댓글을 달았지만 조회에서 걸러지는 것 수정, **flag**로 해당 데이터 변경 확인) 1월 27일
- 공지 관련(임시저장 부분 서비스 생성) 1월 27일
- 시간 관련 에러 수정 및 포맷 동일화 1월 27일
- 댓글 삭제 관련 **API** 수정 1월 28일
- 반에 관한 조회 버그 부분 수정 1월 28일
- 설문조사(생성 및 수정, 조회, 답) 서비스 생성
- **post_type(string->int)** 변경, 2월 8일

웹뷰 브릿지(설문생성, 설문수정)

현재 주소 : <http://lulla-web.s3-website.ap-northeast-2.amazonaws.com/create-survey>

현재 주소 : http://lulla-web.s3-website.ap-northeast-2.amazonaws.com/update-survey?id={survey_id}

1.prev()

-> 웹뷰 창 종료



설문 조사

완료

2.showImage(address)

-> 해당하는 이미지 주소의 이미지를 보여줌

3. showCalendar({start_date: "", end_date: ""})

-> 캘린더 표시 및 해당하는 날짜 선택후 receiveDate(start_date, end_date) 호출

○ 마감기한



4. questionImageSelector()

-> 이미지를 선택하는 창을 띄우고 이미지를 선택 완료한 후 파일을 서버에 등록 후 나온 id 와

address로 setQuestionImg(id, address) 호출

질문

질문을 입력해주세요.



질문

질문을 입력해주세요.



5. questionImageUpdate(id)

-> 이미지를 고른 후 선택 완료시 해당하는 id는 삭제요청 후 다시 이미지 업로드 이후 나온 id, address로 setQuestionImg(id, address) 호출

보기1



보기를 입력해주세요.

6. itemImageSelector(index)

-> 이미지 선택 창을 띄우고 이미지 선택 완료한 후 파일을 서버에 등록 후 나온 id 와 address로 setItemImg(id, address, index) 호출

7. itemImageUpdate(id)

-> 이미지를 선택 후 인자값으로 받은 id는 삭제 호출 후 선택된 이미지 업로드 이후 나온 id 와 address로 setItemImg(id, address, index) 호출

8. sendSurveyId(id)

-> 작성 완료한 설문의 id를 받고 해당하는 웹뷰 종료



설문 조사

완료

웹뷰 브릿지(답변 제출, 답변 수정)

현재 주소 : <http://lulla-web.s3-website.ap-northeast-2.amazonaws.com/submit-answer?id={해당하는 설문 id}> (답변 제출)

현재 주소 : <http://lulla-web.s3-website.ap-northeast-2.amazonaws.com/update-answer?id={해당하는 설문 id}> (답변 수정)

1. getAuthData()

-> 로그인 정보인 token과 현재 프로필인 member_id 를 가지고

receiveAuthData(token, member_id) 호출

2. prev()

-> 웹뷰 창 종료



설문 조사

완료

3. showImage(address)

-> 해당하는 이미지 주소의 이미지를 보여줌

4. showSignature()

-> 설문 완료 후 서명이 필요한 경우 호출 되어짐 서명하는 창을 띄운 후 서명완료 후

해당하는

이미지를 파일로 업로드 후 나온 id로 receiveSignature(id) 호출



제출

5. submitDone({complete_survey : true, survey_id : 해당 설문 id})

-> 설문 완료 후 해당하는 설문 완료 이벤트 전송

웹뷰 브릿지(설문조사 조회)

현재 주소 : <http://lulla-web.s3-website.ap-northeast-2.amazonaws.com/survey?id={해당하는 설문 id}>

1. getAuthData()

-> 로그인 정보인 token과 현재 프로필인 member_id 를 가지고
receiveAuthData(token, member_id) 호출

2. prev()

-> 웹뷰 창 종료



설문 조사

완료

3. showImage(address)

-> 해당하는 이미지 주소의 이미지를 보여줌

센드버드 채널 생성 규약

방생성시 custom_type에 no_remove or remove