



# 기능 명세 & 더미 데이터

🕒 생성일	@2023년 3월 3일 오후 4:58
📌 프로젝트 진행 단계	기획 문서



기능 명세서 & Request Data & Response Data  
CTRL + ALT + T 하면 전체 토글 열기/닫기 가능

## ✓ 기능 명세서 브레인 스토밍

## ✓ 기능 명세서

### 유저

이메일 인증 전송 [POST]  
이메일 인증 확인 [POST]  
이메일 중복 확인 [GET]  
회원 가입 - 정보 입력 [POST]  
로그인 [POST]  
소셜 로그인 (일단 구글) [POST]  
엑세스토큰 재발급 [POST]  
로그아웃 [DELETE] → 뭘냐?  
닉네임 중복 확인 [GET]  
임시 비밀번호 전송 [POST]  
현재 비밀번호 확인 [POST]  
새로운 비밀번호 수정 [PUT]  
사용자 프로필 조회 [GET]  
사용자 프로필 수정 [PUT]  
사용자 프로필 이미지 수정 [PUT]  
사용자 키우는 식물 생성 [POST]  
사용자 키우는 식물 닉네임 수정 [PUT]  
사용자 키우는 식물 리스트 조회 [GET]  
사용자 키우는 식물 삭제 [DELETE]  
회원 탈퇴 [DELETE]

### 피드

피드 추천 조회  
피드 팔로우 조회

### 관찰 일지

관찰 일지 생성 [POST]  
관찰 일지 수정 [PUT]  
관찰 일지 리스트 조회 - [GET]  
관찰 일지 삭제 [DELETE]  
유저가 북마크한 관찰 일지 목록 조회 [GET]  
관찰 일지 북마크 [POST]  
관찰 일지 북마크 취소 [DELETE]

### 일지

변경 사항  
일지 생성 [POST]  
일지 리스트 조회 (타래 식 조회) [GET]  
일지 상세 조회 [GET]

[일지 수정](#) [PUT]  
[일지 삭제](#) [DELETE]  
[일지 좋아요 생성](#) [POST]  
[일지 좋아요 삭제](#) [DELETE]  
[일지 댓글 생성](#) [POST]  
[일지 댓글 리스트 조회](#) [GET]  
[일지 댓글 삭제](#) [DELETE]  
[일지 댓글 수정](#) [PUT]

#### 게시글

[게시글 생성](#) [POST]  
[게시글 리스트 조회](#) [GET]  
[게시글 상세 조회](#) [GET]  
[게시글 수정](#) [PUT]  
[게시글 삭제](#) [DELETE]  
[게시글 댓글 생성](#) [POST]  
[게시글 댓글 리스트 조회](#) [GET]  
[게시글 댓글 삭제](#) [DELETE]  
[게시글 댓글 수정](#) [PUT]  
[게시글 좋아요 생성](#) [POST]  
[게시글 좋아요 삭제](#) [DELETE]

#### 검색

[식물 이름 검색](#) [GET] , plantcontroller :  
[사용자 검색](#) [GET] , usercontroller  
[태그 검색](#) [GET] , feedcontroller  
[인기 식물 조회](#) [GET] , plantcontroller  
[인기 관찰일지 조회](#) [GET], diarycontroller  
[나와 같은 식물을 키우는 사람들 조회](#) [GET] , usercontroller  
[식물 도감 리스트 조회](#) [GET] plantcontroller  
[식물 도감 - 이름 검색](#) [GET] plantcontroller  
[식물 도감 상세 조회](#) [GET] plantcontroller  
[식물 이름 검색 - 내키식 추가시](#) [GET] plantcontroller

#### 스케줄

[월간 스케줄 조회](#) [GET]  
[주간 스케줄 조회](#) [GET]  
[일정 생성](#) [POST]  
[일정 수정](#) [PUT]  
[일정 완료](#) [POST]  
[일정 완료 취소](#) [PUT]  
[일정 삭제](#) [DELETE]

#### 알림

[알림 리스트 조회](#) [GET]  
[알림 삭제](#) [DELETE]

#### 팔로잉

[팔로잉 생성](#) [POST]  
[팔로잉 리스트 조회](#) [GET]  
[팔로잉 삭제 \(언팔로우\)](#) [DELETE]  
[팔로워 리스트 조회](#) [GET]  
[팔로워 삭제](#) [DELETE]

#### 공통 코드

[공통 코드 목록 조회](#) [GET]

- pk ⇒ id

## 유저

### 위로가기

- 로그아웃은 프론트에서 토큰 만료

#### ▼ 이메일 인증 전송 [POST]

##### Request Data

- 이메일 (email) - String

```
{
  "email": "test@ssafy.com"
}
```

##### Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS",
  "data": true,
}
```

#### ▼ 이메일 인증 확인 [POST]

##### Request Data

- 인증번호 (authCode) - String

```
{
  "authCode": "123456"
}
```

##### Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
  "data": true,
}
```

#### ▼ 이메일 중복 확인 [GET]

##### Request Data

- 이메일 (email) - String ⇒ PathVariable

```
{
  "email": "test@ssafy.com"
}
```

##### Response Data

- 상태 코드 (result) - String
- 중복 확인 결과 (data) - Boolean

```
{
  "result": "SUCCESS",
```

```
"data": true, // 이메일 사용 가능하면(중복 아니면) true, 중복이면 false
}
```

#### ▼ 회원 가입 - 정보 입력 [POST]

##### Request Data

- 이메일 (email) - String
- 닉네임 (nickname) - String
- 비밀번호 (password) - String

```
{
  "email": "test@ssafy.com"
  "nickname": "ssafy",
  "password": "test1234*",
}
```

##### Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
  "data": true,
}
```

#### ▼ 로그인 [POST]

##### Request Data

- 이메일 (email) - String
- 비밀번호 (password) - String

```
{
  "email": "test@ssafy.com"
  "password": "test1234!"
}
```

##### Response Data

- 상태 코드 (result) - String
- 유저 닉네임 (nickname)
- 유저 프로필 사진 (profileImagePath)

```
{
  "result": "SUCCESS",
  "data": {
    "nickname": "test",
    "profileImagePath": "/image/default.png",
  }
}
```

- header에 포함시킴

```
ResponseHeader
"authorization" : (bearer + accessToken)
"X-Refresh-Token" : (refreshToken)
```

## ▼ 소셜 로그인 (일단 구글) [POST]

oauth 로그인시 깃헙, 구글 같은 경우는 여기서 refreshToken 줍니다.  
따라서 db 에 해당 refreshToken 저장하는 api 필요함

### Request Data

- header에 accessToken과 refreshToken을 담아서 보냄

```
Request Header

"Authorization" : "bearer " + accessToken
"X-Refresh-Token": (refreshToken) String
```

### Response Data

- 상태 코드 (result) - String
- 유저 닉네임 (nickname)
- 유저 프로필 사진 (profileImagePath)

```
{
  "result": "SUCCESS",
  "data": {
    "nickname" : "test",
    "profileImagePath" : "/image/default.png",
  }
}
```

## ▼ header에 accessToken과 refreshToken을 담아서 반환

```
ResponseHeader
"authorization" : (bearer + accessToken)
"X-Refresh-Token" : (refreshToken)
```

## ▼ 액세스토큰 재발급 [POST]

### Request Data

```
Request Header

"Authorization" : "bearer " + accessToken
"X-Refresh-Token": (refreshToken) String
```

### Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS",
  "data" : null,
}
```

- 헤더에는 아래처럼 담김

```
ResponseHeader
"authorization" : (bearer + accessToken)
"X-Refresh-Token" : (refreshToken)
```

#### ▼ 로그인아웃 [DELETE] → 넣나?

- 저장소에서 refresh Token 삭제하는 용도로 저장

##### | Request Data

x

- header에 정보 담아서 보내줌

```
RequestHeader
"authorization": (accessToken) String
"X-Refresh-Token": (refreshToken) String
```

##### | Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS",
  "data" : null,
}
```

#### ▼ 닉네임 중복 확인 [GET]

##### | Request Data

- 닉네임 (nickname) - String

```
{
  "nickname": "test"
}
```

##### | Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS",
  "data": null,
}
```

#### ▼ 임시 비밀번호 전송 [POST]

##### | Request Data

- 이메일 (email) - String

```
{
  "email": "test@ssafy.com"
}
```

## Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
}
```

### ▼ 현재 비밀번호 확인 [POST]

## Request Data

- 수정 비밀번호 (password) - String

```
{
  "password": "test5678!"
}
```

## Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
}
```

### ▼ 새로운 비밀번호 수정 [PUT]

## Request Data

- 수정 비밀번호 (newPassword) - String

```
{
  "newPassword": "test5678!"
}
```

## Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
}
```

### ▼ 사용자 프로필 조회 [GET]

## Request Data

- 사용자 닉네임 (nickname) - String

```
{
  "nickname": "test" // pathVariable
}
```

## Response Data

- 사용자 닉네임 (nickname) - String
- 사용자 프로필 이미지 경로 (profileImagePath) - String
- 사용자 한 줄 소개 (introduction) - String
- 사용자 팔로잉 수 (followingCount) - Integer
- 사용자 팔로워 수 (followerCount) - Integer
- 내가 팔로우하고 있는지 여부 (isFollowed) - Boolean

```
{
  "result": "SUCCESS",
  "data": {
    "nickname": "test",
    "profileImagePath": "/public/images/profile.png",
    "introduction": "안녕하세요 test 입니다",
    "followingCount": 111,
    "followerCount": 222,
    "isFollowed": true
  }
}
```

### ▼ 사용자 프로필 수정 [PUT]

## Request Data

- 변경되는 사용자 닉네임 (nickname) - String
- 한줄 소개 (introduction) - String

```
{
  "nickname": "test",
  "introduction": "introduction소개소개",
}
```

## Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
}
```

### ▼ 사용자 프로필 이미지 수정 [PUT]

## Request Data

- 사용자 프로필 이미지 formData (MultipartFile)




```
{
  "profileImage": formData
}
```

## Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
}
```

### ▼ 사용자 키우는 식물 생성 [POST]

- flow : 식물 추가를 할 때 → 식물 이름 검색 → 내 식물 선택(plantId 획득) → plantId 식물 추가
  - 이미지 분석을 포기..? (임시 
    - ⇒ AI는 병충해 분석 + 일러스트
  - 아니면 식물 도감을 dataset에 맞게 정제..?

## Request Data

- 식물 id (plantId) - Long
- 내키식 별명 (plantNickname) - String

```
{
  "plantId": 1,
  "plantNickname": "myPlant"
}
```

## Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
}
```

### ▼ 사용자 키우는 식물 닉네임 수정 [PUT]

## Request Data

- 내가 키우는 식물 id (userPlantId) - Long
- 내키식 별명 (plantNickname) - String

```
{
  "userPlantId": 1,
  "plantNickname": "myPlant"
}
```

## Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
}
```

### ▼ 사용자 키우는 식물 리스트 조회 [GET]

#### Request Data

- 사용자 닉네임 (nickname) - String

Pathvariable

```
{
  "nickname": "test"
}
```

#### Response Data

- 사용자가 키우는 식물 리스트 array
  - 내키식 id (userPlantId) - Long
  - 식물id (plantId) - Long
  - 식물 별명 (plantNickname) - String
  - 식물 이름 (plantName) - String
  - 식물 이미지 파일 경로 (plantImagePath) - String

```
{
  "result": "SUCCESS",
  "data": [
    {
      "userPlantId": 1,
      "plantId": 1,
      "plantNickname": "myPlant",
      "plantName": "대파",
      "plantImagePath": "/public/images/plant.png"
    },
    {
      "userPlantId": 2,
      "plantId": 2,
      "plantNickname": "myPlant",
      "plantName": "토마토",
      "plantImagePath": "/public/images/plant.png"
    }
  ]
}
```

### ▼ 사용자 키우는 식물 삭제 [DELETE]

#### Request Data

- 사용자가 키우는 식물id (uesrPlantId) - Long

```
{
  "userPlantId": 1
}
```

#### Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
}
```

## ▼ 회원 탈퇴 [DELETE]

### | Request Data

x

### | Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS"
}
```

## 피드

## ▼ 피드 추천 조회

### | Request Data

- page : Integer
- size: Integer

```
page, size - RequestParam

https://chicorita.com/.../{nickname}/post?page=10&size=10
```

### | Response Data

- 피드 List 정보
  - 유저 정보
    - 닉네임 (nickname) : String
    - 프로필 이미지 경로 (profileImagePath) : String
    - 한 줄 소개 (introduction) : String
    - 팔로잉 카운트 (followingCount) : Integer
    - 팔로워 카운트 (followerCount) : Integer
  - 피드 유형 ((공통 코드)feedCode) : String
  - 관찰 날짜 (FEED\_POST인 경우 null) (observationDate) : LocalDate
  - 피드 Id (feedId): Long
  - 내용 간략 (15자) (content): String
  - 이미지 경로 (imagePath) : String
  - 좋아요 수 (likeCount): Long
  - 좋아요 체크 (isLiked) : Boolean

- 댓글 수 (commentCount): Long
- 등록 날짜 (createdAt): LocalDateTime

```
json
{
  "result": "SUCCESS"
  "data": [
    {
      "user": {
        "nickname": "test",
        "profileImagePath": "/img/default.png",
        "introduction": "ssss",
        "followingCount": 0,
        "followerCount": 1,
        "isFollowed": false,
      },
      "feedCode": "FEED_POST",
      "observationDate": null,
      "feedId": 1,
      "content": '내 아기방토 입으로 호로...', // 잘린 글자
      "imagePath": "/img/default.png",
      "likeCount": 0,
      "isLiked": true,
      "commentCount": 2,
      "createdAt": "2023-03-03-...."
    },
    {
      "user": {
        "nickname": "test",
        "profileImagePath": "/img/default.png",
        "introduction": "ssss",
        "followingCount": 0,
        "followerCount": 1,
        "isFollowed": false,
      },
      "feedCode": "FEED_DIARY",
      "observationDate": "2023-03-04",
      "feedId": 1,
      "content": '내 아기방토 입으로 호로...', // 잘린 글자
      "imagePath": "/img/default.png",
      "likeCount": 0,
      "isLiked": true,
      "commentCount": 2,
      "createdAt": "2023-03-03-...."
    }
  ]
}
```

## ▼ 피드 팔로우 조회

### Request Data

- page : Integer
- size: Integer

page, size - RequestParam

<https://chicorita.com/.../{nickname}/post?page=10&size=10>

### Response Data

- 피드 List 정보
  - 유저 정보
    - 닉네임 (nickname) : String
    - 프로필 이미지 경로 (profileImagePath) : String

- 한 줄 소개 (introduction) : String
- 팔로잉 카운트 (followingCount) : Integer
- 팔로워 카운트 (followerCount) : Integer
- 피드 유형 ((공통 코드)feedCode) : String
- 관찰 날짜 (FEED\_POST인 경우 null) (opservatinoDate) : LocalDate
- 피드 Id (feedId): Long
- 내용 간략 (15자) (content): String
- 이미지 경로 (imagePath) : String
- 좋아요 수 (likeCount): Long
- 좋아요 체크 (isLiked) : Boolean
- 댓글 수 (commentCount): Long
- 등록 날짜 (createdAt): LocalDateTime

```
json
{
  "result": "SUCCESS"
  "data": [
    {
      "user": {
        "nickname": "test",
        "profileImagePath": "/img/default.png",
        "introduction": "ssss",
        "followingCount": 0,
        "followerCount": 1,
        "isFollowed": false,
      },
      "feedCode": "FEED_POST",
      "opsevationDate": null,
      "feedId":1,
      "content": '내 아기방토 입으로 호로...', // 잘린 글자
      "imagePath": "/img/default.png",
      "likeCount":0,
      "isLiked": true,
      "commentCount" : 2,
      "createdAt":2023-03-03-....
    },
    {
      "user": {
        "nickname": "test",
        "profileImagePath": "/img/default.png",
        "introduction": "ssss",
        "followingCount": 0,
        "followerCount": 1,
        "isFollowed": false,
      },
      "feedCode": "FEED_DIARY",
      "opsevationDate": "2023-03-04",
      "feedId":1,
      "content": '내 아기방토 입으로 호로...', // 잘린 글자
      "imagePath": "/img/default.png",
      "likeCount":0,
      "isLiked": true,
      "commentCount" : 2,
      "createdAt":2023-03-03-....
    },
  ]
}
```

## 관찰 일지

### 위로가기

- 필드 차이

- startDate : 사용자가 직접 선택하는 관찰 일지 시작 날짜
- createdAt : 관찰일지 데이터가 생성된 시간 (DB자동생성)

#### ▼ 관찰 일지 생성 [POST]

##### RequestData

- userPlantId (내키식 id) - Long
- image (메인 이미지) - Multipart/Type
- startDate (날짜) - LocalDate
- title (내용) - String

```
{
  "userPlantId" : 1,
  "image": multipart file,
  "startDate": "2023-03-03",
  "title": "제목",
}
```

##### ResponseData

- 상태코드

```
{
  "result": "SUCCESS",
  "data": null
}
```

#### ▼ 관찰 일지 수정 [PUT]

##### RequestData

- diarySetId (관찰일지 id) - Long
- userPlantId (내키식 id) - Long
- image (메인 이미지) - Multipart/Type
- startDate (날짜) - LocalDate
- title (내용) - String

```
{
  "diarySetId": 1,
  "userPlantId" : 1,
  "image": multipart file,
  "startDate": "2023-03-03",
  "title": "제목",
}
```

##### ResponseData

- 상태코드

```
{
  "result": "SUCCESS"
  "data": null
}
```

## ▼ 관찰 일지 리스트 조회 - [GET]

### RequestData

- nickname (유저 닉네임) - String
- page (페이지) - Integer
- size (사이즈) - Integer

```
{
  "nickname" : "rst1212", // pathVariable
  "page": "1", // requestParam (Query)
  "size": "1" // requestParam (Query)
}
```

### ResponseData

- List<관찰일지>
  - diarySetId (관찰일지 id) - Long
  - imagePath (메인 이미지) - String
  - bookmarkCount (북마크 개수) - Integer
  - isBookmarked (북마크 여부) - Boolean
  - diaryCount (일지 페이지 수) - Integer
  - title (제목) - String

```
{
  "result": "SUCCESS",
  "data": [
    {
      "diarySetId": 1,
      "imagePath": "image1/jpg",
      "bookmarkCount": 32,
      "isBookmarked": false,
      "diaryCount": 10,
      "title": "제목",
    },
    {
      "diarySetId": 1,
      "imagePath": "image1/jpg",
      "bookmarkCount": 32,
      "isBookmarked": false,
      "diaryCount": 10,
      "title": "제목",
    },
    {
      "diarySetId": 1,
      "imagePath": "image1/jpg",
      "bookmarkCount": 32,
      "isBookmarked": false,
      "diaryCount": 10,
      "title": "제목",
    }
  ]
}
```

## ▼ 관찰 일지 삭제 [DELETE]

### RequestData

- diarySetId (관찰일지 id) - Long

```
{
  "diarySetId" : 1, // Path
}
```

## ResponseData

- 상태코드

```
{
  "result": "SUCCESS"
  "data": null
}
```

## ▼ 유저가 북마크한 관찰 일지 목록 조회 [GET]

## RequestData

- nickname (유저 닉네임) - String
- 페이지 - Integer
- 사이즈 - Integer

```
{
  "nickname" : "test", // path
  "page": "1", // query
  "size": "1" // query
}
```

## ResponseData

- 리스트<관찰일지>
  - 관찰일지 id - Long
  - 메인 이미지 - String
  - 북마크 개수 - Integer
  - 북마크 여부 - Boolean
  - 일지 페이지 수 - Integer

```
{
  "result": "SUCCESS",
  "data": [
    {
      "diarySetId": 1,
      "imagePath": "image1/jpg",
      "bookmarkCount": 32,
      "isBookmarked": false,
      "diaryCount": 10,
      "title": "제목",
    },
    {
      "diarySetId": 1,
      "imagePath": "image1/jpg",
      "bookmarkCount": 32,
      "isBookmarked": false,
      "diaryCount": 10,
      "title": "제목",
    },
    {
      "diarySetId": 1,
      "imagePath": "image1/jpg",
      "bookmarkCount": 32,
    }
  ]
}
```



```

        "isBookmarked": false,
        "diaryCount": 10,
        "title": "제목",
    }
}
}

```

#### ▼ 관찰 일지 북마크 [POST]

##### RequestData

- diarySetId (관찰 일지 아이디) - Long

```

{
  "diarySetId": 1,
}

```

##### ResponseData

- 상태코드

```

{
  "result": "SUCCESS",
  "data": null
}

```

#### ▼ 관찰 일지 북마크 취소 [DELETE]

##### RequestData

- diarySetId (관찰 일지 아이디) - Long

```

{
  "diarySetId": 1,
}

```

##### ResponseData

- 상태코드

```

{
  "result": "SUCCESS",
  "data": null
}

```

## 일지

### 🔥 위로가기

#### 변경 사항

- (기존) DiarySet에만 댓글 달기 ⇒ (변경) 댓글은 각 Diary마다 달기

#### ▼ 일지 생성 [POST]

## Request Data

- 관찰 일지 id (diarySetId) - Long ⇒ **pathVariable**
- 설명글 (content) - String
- 태그 목록들 (tags) - List<String>
- 관찰 일시 (opsevationDate) - LocalDate
- 이미지 (image) - multipart/form-data

```
{
  "diarySetId": 1, // pathVariable
  "content": "testtest",
  "tags": ["test1", "test2"],
  "opsevationDate": "2023-03-04",
  "image": multipart File
}
```

## Response Data

- 상태 코드

```
{
  "result": "SUCCESS",
  "data": null
}
```

### ▼ 일지 리스트 조회 (타래 식 조회) [GET]

- 댓글은 상세 조회에서!

## Request Data

- 관찰 일지 id - Long ⇒ **pathVariable**

```
{
  "diarySetId": 1 // pathVariable
}
```

## Response Data

- 작성자 정보(user)
  - 닉네임 (nickname) - String
  - 유저 프로필 이미지 (profileImagePath) - String
- 제목 (title) - String
- 관찰 시작 일자 (startDate) - LocalDate
- 일지 리스트 (diaryList) - List<Object>
  - 일지 id (diaryId) - Long
  - 내용 (content) - String
  - 태그 리스트 (tags) - List<String>
  - 관찰 일자 (opsevationDate) - LocalDate

- 일지 생성 시각 (createdAt) - LocalDateTime
- 일지 이미지 (imagePath) - String
- 댓글 개수 (commentCount) - Integer

```
{
  "result": "SUCCESS",
  "data": {
    "user": {
      "nickname": "test",
      "profileImagePath": "http://placekitten.com/200/300",
    },
    "title": "제목", // 관찰 일지 제목
    "startDate": "2023-03-03", // diarySet 관찰 시작 시간
    "diaryList": [
      {
        "diaryId": 1,
        "content": "testtest",
        "tags": ["test1", "test2"],
        "opsevationDate": "2023-03-04",
        "createdAt": "2023-03-03T11:00:11", // diary 생성 시간
        "imagePath": "http://placekitten.com/200/300",
        "commentCount": 10,
      },
      {
        "diaryId": 2,
        "content": "testtest",
        "tags": ["test1", "test2"],
        "opsevationDate": "2023-03-04",
        "createdAt": "2023-03-03T11:00:11", // diary 생성 시간
        "imagePath": "http://placekitten.com/200/300",
        "commentCount": 10,
      }
    ]
  }
}
```

## ▼ 일지 상세 조회 [GET]

### Request Data

- 일지 id - Long ⇒ **pathVariable**

```
{
  "diaryId": 1, // pathVariable
}
```

### Response Data

- 관찰 일지 id (diarySetId) - Long
- 관찰 일자 (opsevationDate) - LocalDate
- 글 내용 (content) - String
- 태그 목록들 (tags) - List<String>
- 내 프로필 이미지 (imagePath) - String
- 일지 생성 일시 (createdAt) - LocalDateTime
- 좋아요 개수 (likeCount) - Integer
- ▼ ~~댓글 리스트 (commentList) - List<Object>~~
  - 댓글 id (commentId) - Long
  - 유저 정보 (user) - Object
    - 닉네임 (nickname) - String

- 유저 프로필 이미지 (profileImagePath) - String
- 댓글 내용 (content) - String
- 유저 언급 (mentionNickname) - String
- 댓글 생성 일시 (createdAt) - LocalDateTime

```
{
  "result": "SUCCESS",
  "data": {
    "diarySetId" : 1,
    "opservationDate": "2023-03-04",
    "content": "testtest",
    "tags": ["test1", "test2"],
    "imagePath": "/image/default.png",
    "likeCount": 1,
    "craetedAt": "2023-03-04T11:00:08",
  }
}
```

## ▼ 일지 수정 [PUT]

### Request Data

- 관찰 일지 id (diaryId) - Long ⇒ **pathVariable**
- 설명글 (content) - String
- 태그 목록들 (tags) - List<String>
- 이미지 (image) - Multipart File

```
{
  "diaryId": 1, // pathVariable
  "content": "testtest",
  "tags": ["test1", "test2"],
  "image": multipartFile
}
```

### Response Data

- 상태 코드

```
{
  "result": "SUCCESS",
  "data": null
}
```

## ▼ 일지 삭제 [DELETE]

- 일지 id (diaryId) - Long ⇒ **pathVariable**

```
{
  "diaryId": 1 // pathVariable
}
```

### Response Data

- 성공 표시

```
{
  "result": "SUCCESS",
}
```

```
"data": null
}
```

#### ▼ **일지 좋아요 생성 [POST]**

##### | Request Data

- 피드 id (feedId) - Long ⇒ **pathVariable**

```
{
  "feedId": 1, // pathVariable
}
```

##### | Response Data

- 성공 표시

```
{
  "result": "SUCCESS",
  "data": null
}
```

#### ▼ **일지 좋아요 삭제 [DELETE]**

##### | Request Data

- 피드 id (feedId) - Long ⇒ **pathVariable**

```
{
  "feedId": 1, // pathVariable
}
```

##### | Response Data

- 성공 표시

```
{
  "result": "SUCCESS",
  "data": null
}
```

#### ▼ **일지 댓글 생성 [POST]**

##### | Request Data

- 피드 Id (feedId) : Long ⇒ **pathVariable**
- 댓글 내용 (content) : String
- 멘션 유저 닉네임 (mentionNickname) : String (nullable)

```
postId - pathvariable
https://chicorita/feed/1/comment

json
{
  "feedId": 1, // pathVariable
  "content": "입 짱 크시네요",
}
```

```
}
  "mentionNickname": "송편빈대떡"
}
```

## Response Data

- 상태 코드

```
{
  "result": "SUCCESS",
  "data": null
}
```

## ▼ 일지 댓글 리스트 조회 [GET]

### Request Data

- 일지 Id (leafId) : Long ⇒ **pathVariable**
- page (page) : Integer
- size (size) : Integer

```
{
  "feedId": 1, // pathVariable
  "page": 1, // query
  "size": 1, // query
}
```

### Response Data

- 댓글 정보 List<Object>
  - 댓글 id (commentId) : Long
  - 유저 정보 (user) : Object
    - 사용자 닉네임 (nickname) : String
    - 프로필 이미지 (profileImagePath) : String
  - 댓글 내용 (content) : String
  - 멘션 유저 닉네임 (mentionNickname) : String
  - 등록 날짜 (createdAt) : LocalDateTime

```
{
  "result": "SUCCESS",
  "data": [
    {
      "commentId" : 20,
      "user": {
        "nickname" : "빈빔빈빔",
        "profileImagePath" : "/s3/profile/fasdjfiojwf빈img.jpg",
      },
      "content" : "이거봐라 얘 또 먹는다",
      "mentionNickname" : "염쓰",
      "createdAt" : "2023-03-03-...."
    },
    {
      "commentId" : 20,
      "user": {
        "nickname" : "빈빔빈빔",
        "profileImagePath" : "/s3/profile/fasdjfiojwf빈img.jpg",
      },
    },
  ],
}
```

```

        "content" : "이거봐라 애 또 먹는다",
        "mentionNickname" : "염쓰",
        "createdAt" : "2023-03-03-...."
    }
}
}

```

## ▼ 일지 댓글 삭제 [DELETE]

### Request Data

- 댓글 Id (commentId) : Long ⇒ **pathVariable**
- 피드 Id (feedId) : Long ⇒ **pathVariable**

```

{
  "feedId": 1, // pathVariable
  "commentId": 1, // pathVariable
}

```

### Response Data

```

{
  "result": "SUCCESS",
  "data": null
}

```

## ▼ 일지 댓글 수정 [PUT]

### Request Data

- 댓글 Id (commentId) : Long ⇒ **pathVariable**
- 피드 Id (feedId) : Long ⇒ **pathVariable**
- 댓글 내용 (content) : String
- 멘션 유저 닉네임 (mentionNickname) : String (nullable)

```

{
  "commentId": 1, // pathVariable
  "feedId": 1, // pathVariable

  "content": "임 평수 잘 빠지셨어요",
  "mentionNickname": "송아지빈수레"
}

```

### Response Data

```

{
  "result": "SUCCESS",
  "data": null
}

```

## 게시글

## ▼ 게시물 생성 [POST]

### Request Data

- 내용(content) : String
- 사진 파일 (image): MultipartFile (nullable)
- 태그 List (tags) ( nullable)
  - 태그 내용 : String

```
form-data
{
  "content": '내 아기방토 입으로 호로록'
  "image": multipartFile // '내아기방토.jpg'
  "tags": ['방울토마토', '방토']
}
```

### Response Data

- 상태 코드

```
{
  "result": "SUCCESS"
  "data":null
}
```

## ▼ 게시물 리스트 조회 [GET]

- My 프로필에서 보이는 목록 조회

### Request Data

- 유저 닉네임(nickname) : String
- page : Integer
- size: Integer

```
nickname - pathvariable
page, size - RequestParam

{
  "nickname" : "test", // pathvariable
}
https://chicorita.com/.../{nickname}/post?page=10&size=10
```

### Response Data

- 포스트 List 정보
  - 포스트 Id (postId): Long
  - 포스트 내용 간략 (15자) (content): String
  - 댓글 수 (commentCount): Long



- 좋아요 수 (likeCount): Long
- 등록 날짜 (createdAt): LocalDateTime

```
json
{
  "result": "SUCCESS"
  "data": [
    {
      "postId":1,
      "content": '내 아기방토 입으로 호로...', // 잘린 글자
      "commentCount" : 2,
      "likeCount":0,
      "createdAt":2023-03-03-....
    },
    {
      "postId":2,
      "content": '내 아기방토 입으로 호로록 하...',
      "commentCount" : 5,
      "likeCount":9,
      "createdAt":2023-03-03-....
    },
    {
      "postId":3,
      "content": '아기방토 내 입에 저장',
      "commentCount" : 22,
      "likeCount":17,
      "createdAt":2023-03-03-....
    }
  ]
}
```

## ▼ 게시물 상세 조회 [GET]

### Request Data

- 포스트 Id (postId): Long

```
leafId- pathvariable
https://chicorita.com/post/1
```

### Response Data

- 포스트 Id : Long
- 포스트 내용 : String
- 사진 파일 : String
- 댓글 수 : Long
- 댓글 list
- 좋아요 수 : Long
- 등록 날짜 : LocalDateTime
- 태그 List
  - 태그 내용 : String

```
json
{
  "result": "SUCCESS"
  "data":{
    "postId":1,
```

```

"content": '내 아기방토 입으로 호로록',
"imagePath": 's3/post/sljkjfsidof내아기방토.jpg',
"tags": ['방울토마토', '방토'],
"createdAt": "2023-03-04T11:00:08", // 게시물 생성 날짜
"likeCount": 0,
"commentCount": 2,
}
}

```

## ▼ 게시물 수정 [PUT]

### Request Data

- 포스트 Id : Long
- 내용 : String
- 사진 파일 : MultipartFile (nullable)
- 태그 List ( nullable)
  - 태그 내용 : String

```

postId- pathvariable

https://chicorita.com/post/1

form-data 안에 이미지 + 데이터 같이 보내기
{
  "postId" : 1, // pathVariable
  "content": '내 아기방토 입으로 호롱구이',
  "image": multipart File, //'방토방토.jpg'
  "tags": ['방울토마토', '방토', '툼메이러']
}

```

### Response Data

- 상태 코드

```

{
  "result": "SUCCESS"
  "data": null
}

```

## ▼ 게시물 삭제 [DELETE]

### Request Data

- 포스트 Id : Long

```

leafId- pathvariable

https://chicorita.com/post/1
{
  "postId" : 1,
}

```

### Response Data

```

{
  "result": "SUCCESS"
}

```

```
"data":null
}
```

#### ▼ 게시글 댓글 생성 [POST]

##### Request Data

- 포스트 Id : Long
- 댓글 내용 : String
- 멘션 유저 닉네임 : String (nullable)

```
leafId- pathvariable
https://chicorita/post/1

json
{
  "postId": 1, // pathVariable
  "content": "입 장 크시네요",
  "mentionNickname": "송편빈대떡"
}
```

##### Response Data

- 상태 코드

```
{
  "result": "SUCCESS"
  "data":null
}
```

#### ▼ 게시글 댓글 리스트 조회 [GET]

##### Request Data

- 포스트 Id : Long
- page : Integer
- size: Integer

```
postId- pathvariable
page,size - RequestParam

https://chicorita.com/post/2?page=2&size=10
```

##### Response Data

- 댓글 정보 List
  - 댓글 id : Long
  - 사용자 닉네임 : String
  - 프로필 이미지 : String
  - 댓글 내용 : String
  - 멘션 유저 닉네임 : String
  - 등록 날짜 : LocalDateTime

```
{
  "result": "SUCCESS"
  "data": [
    {
      "commentId": 20,
      "user": {
        "nickname": "빈빔빈빔",
        "profileImage": "/s3/profile/fasdjfiojwf빈img.jpg",
      },
      "content": "이거봐라 애 또 먹는다",
      "mentionNickname": "염쓰",
      "createdAt": "2023-03-03-...."
    },
    {
      "commentId": 20,
      "user": {
        "nickname": "빈빔빈빔",
        "profileImage": "/s3/profile/fasdjfiojwf빈img.jpg",
      },
      "content": "이거봐라 애 또 먹는다",
      "mentionNickname": "염쓰",
      "createdAt": "2023-03-03-...."
    }
  ]
}
```

## ▼ 게시물 댓글 삭제 [DELETE]

### Request Data

- 댓글 Id : Long
- 게시물 Id : Long

postId, commentId - pathvariable  
<https://chicorita.com/post/1/comment/123>

### Response Data

```
{
  "result": "SUCCESS"
  "data": null
}
```

## ▼ 게시물 댓글 수정 [PUT]

### Request Data

- 댓글 Id : Long
- 게시물 Id : Long
- 댓글 내용 : String
- 멘션 유저 닉네임 : String (nullable)

postId, commentId - pathvariable  
<https://chicorita.com/post/1/comment/123>  
 json  
 {  
 "content": "입 평소 잘 빠지셨어요",  
 }

```
"metionNickname": "송아지빈수레"
}
```

## Response Data

```
{
  "result": "SUCCESS"
  "data": null
}
```

### ▼ 게시물 좋아요 생성 [POST]

## Request Data

- leaf id (leafId) - Long ⇒ **pathVariable**

```
{
  "postId": 1, // pathVariable
}
```

## Response Data

- 성공 표시

```
{
  "result": "SUCCESS"
  "data": null
}
```

### ▼ 게시물 좋아요 삭제 [DELETE]

## Request Data

- leaf Id (leafId) - Long ⇒ **pathVariable**

```
{
  "postId": 1, // pathVariable
}
```

## Response Data

- 성공 표시

```
{
  "result": "SUCCESS"
  "data": null
}
```

## 검색

### 🔍 위로가기

### ▼ 식물 이름 검색 [GET] , plantcontroller :

## Request Data

- 검색 타입 (searchType) - (공통코드) String / “식물 이름 검색”
- 검색어 (search) - String
- 페이지 번호 (page) - Integer
- 페이지 게시글 수 (size) - Integer

```
{
  "searchType": "001001", // (임시) 식물 이름 검색
  "search": "토마토",
  "page": 0,
  "size": 10,
}
```

## Response Data

- 식물 정보 List
  - 식물 정보 id (plantId) - Long
  - 식물 이미지 경로 (imagePath) - String
  - 식물 이름 (plantName) - String

```
{
  "result": "SUCCESS",
  "data" : [
    {
      "plantId": 1,
      "imagePath": "/img/default.jpg",
      "plantName": "방울토마토",
    },
    {
      "plantId": 2,
      "imagePath": "/img/default.jpg",
      "plantName": "찰토마토",
    }
  ]
}
```

### ▼ 사용자 검색 [GET], usercontroller

## Request Data

- 검색 타입 (saerchType) - (공통코드) String / “사용자 검색”
- 검색어 (search) - String
- 페이지 번호 (page) - Integer
- 페이지 게시글 수 (size) - Integer

```
{
  "searchType": "001003", // (임시) 사용자 검색
  "search": "테스트유저",
  "page": 0,
  "size": 10,
}
```

## Response Data

- 유저 정보 리스트 List
  - 유저 닉네임 (nickname) - String
  - 유저 프로필 사진 경로 (profileImagePath) - String

```
{
  "result": "SUCCESS",
  "data": [
    {
      "nickname": "테스트유저1",
      "profileImagePath": "/img/default.jpg",
    },
    {
      "nickname": "테스트유저2",
      "profileImagePath": "/img/default.jpg",
    }
  ]
}
```

## ▼ 태그 검색 [GET], feedcontroller

### Request Data

- 검색 타입 (searchType) - String
- 검색어 (search) - String
- 페이지 번호 (page) - Integer
- 페이지 게시물 수 (size) - Integer

```
{
  "searchType": "001004", // (임시) 태그 검색
  "search": "토마토태그",
  "page": 0,
  "size": 10,
}
```

## Response Data

포스트 & 일지는 같은 부모 클래스를 상속할 예정

- 리프 + 일지 List
  - 리프 or 일지 id (feedId) - Long
  - 타입 (feedCode) - String / “리프” or “일지”
  - 이미지 경로 (imagePath) - String / nullable
  - 내용 (content) - String

```
{
  "result": "SUCCESS",
  "data" : [
    {
      "feedId" : 1,
      "feedCode": "002001", // (임시) 포스트
      "imagePath": "/img/default.jpg",
      "content": "알라리알라포스트"
    },
  ],
}
```

```
{
  "feedId" : 2,
  "feedCode": "002002", // (임시) 일지
  "imagePath": "/img/default.jpg",
  "content": "알라리알라일지"
}
]
```

## ▼ 인기 식물 조회 [GET], plantcontroller

### Request Data

x

### Response Data

가장 많이 키우는 상위 top N개 식물 조회

- N개 식물 정보 List
  - 식물 정보 id / Long
  - 식물 이름 / String
  - 식물 이미지 경로 / String

```
{
  "result": "SUCCESS",
  "data" : [
    {
      "plantId": 1,
      "plantName": "토마토",
      "imagePath": "/img/default.jpg",
    },
    {
      "plantId": 2,
      "plantName": "토마토",
      "imagePath": "/img/default.jpg",
    },
    {
      "plantId": 3,
      "plantName": "토마토",
      "imagePath": "/img/default.jpg",
    }
  ]
}
```

## ▼ 인기 관찰일지 조회 [GET], diarycontroller

### Request Data

x

### Response Data

관찰 일지에 속한 일지들의 좋아요 수 총 합이 가장 많은 것

- N개 관찰 일지 List
  - 관찰 일지 id / Long
  - 관찰 시작 날짜 : LocalDate
  - 관찰 일지 중 첫번째 일지 사진 경로 / String



```
{
  "result": "SUCCESS",
  "data" : [
    {
      "diarySetId": 1,
      "startDate": "2023-03-08",
      "imagePath": "/img/default.jpg",
      "title": "제목",
    },
    {
      "diarySetId": 2,
      "startDate": "2023-03-08",
      "imagePath": "/img/default.jpg",
      "title": "제목",
    },
    {
      "diarySetId": 3,
      "startDate": "2023-03-08",
      "imagePath": "/img/default.jpg",
      "title": "제목",
    },
  ],
}
```

#### ▼ 나와 같은 식물을 키우는 사람들 조회 [GET] , usercontroller

클릭 했을 때 유저 프로필로 이동

##### | Request Data

x

##### | Response Data

- N개 유저 정보 조회 List
  - 유저 닉네임 / String
  - 식물 이름 / String
  - 식물 이미지 경로 / String

```
{
  "result": "SUCCESS",
  "data" : [
    {
      "nickname": "테스트유저",
      "plantName": "토마토",
      "imagePath": "/img/default.jpg"
    }
  ]
}
```

#### ▼ 식물 도감 리스트 조회 [GET] plantcontroller

##### | Request Data

- 인덱스 검색 (search) - String
- 페이지 번호 (page) - Integer
- 페이지 게시글 수 (size) - Integer

```
{
  "index": "1", // requestparam
  "page": 0, // requestparam
  "size": 10, // requestparam
}
```

## Response Data

- 식물 정보 List
  - 식물 정보 id (plantId) - Long
  - 식물 이름 (plantName) - String

```
{
  "result": "SUCCESS",
  "data" : [
    {
      "plantId": 1,
      "plantName": "가지",
    }
  ]
}
```

## ▼ 식물 도감 - 이름 검색 [GET] plantcontroller

### Request Data

- 검색어 (search) - String
- 페이지 번호 (page) - Integer
- 페이지 게시글 수 (size) - Integer

```
{
  "search": "토마토", // requestparam
  "page": 0, // requestparam
  "size": 10, // requestparam
}
```

### Response Data

- 식물 정보 List
  - 식물 정보 id (plantId) - Long
  - 식물 이름 (plantName) - String

```
{
  "result": "SUCCESS",
  "data" : [
    {
      "plantId": 1,
      "plantName": "방울토마토",
    },
    {
      "plantId": 2,
      "plantName": "찰토마토",
    }
  ]
}
```

```
]
}
```

#### ▼ 식물 도감 상세 조회 [GET] plantcontroller

##### Request Data

- 식물 정보 id (plantId) - Long

```
{
  "plantId": 1 // pathVariable
}
```

##### Response Data

- 식물 이미지 경로 (imagePath) - String
- 식물 이름 (plantName) - String
- 그 외 식물 정보 / String...String.... / (어떤 정보 제공해줄건지 확실해지면 그걸로)

```
{
  "result": "SUCCESS",
  "data" : {
    "imagePath": "/img/default.jpg",
    "plantName": "찰토마토",
    (대충 여기 식물 정보 추가될 것)
  }
}
```

#### ▼ 식물 이름 검색 - 내키식 추가시 [GET] plantcontroller

##### Request Data

- 검색어 (search) - String

```
{
  "search": "토마토" // pathVariable
}
```

##### Response Data

- List<Object>
  - 식물 id - Long
  - 식물 이름 (plantName) - String

```
{
  "result": "SUCCESS",
  "data" : [
    {
      "plantId": 1,
      "plantName": "토마토"
    },
    {
      "plantId": 2,
      "plantName": "방울토마토"
    }
  ]
}
```

## 스케줄

### 🔗 위로가기

#### ▼ 월간 스케줄 조회 [GET]

##### RequestData

- 조회 년도 (year) - Integer `@RequestParam`
- 조회 월 (month) - Integer `@RequestParam`

```
/schedule/month
{
  "year": 2022,
  "month": 11,
}
```

##### ResponseData

이전month, 현재 month, 다음month 총 3달

- 일간 스케줄 리스트 : `Array<일간 스케줄 조회 ResponseData>`
  - 스케줄 id (scheduleId) - Long
  - 스케줄 날짜 (scheduleDate) - LocalDate
  - 스케줄 카테고리 (물주기, 분갈이..) (scheduleCode)
    - Type [WATER | REPOT | ... ] or string
      - 물주기 : WATER
      - 분갈이하기 : REPOT
      - 가지치기 : PRUNING
      - 영양관리 : NUTRITION
      - 환기하기 : VENTILATION
      - 분무하기 : SPRAY
  - 스케줄 완료 유무 (isCompleted) - Boolean
  - 내키식 정보 (plant)
    - 식물 id (plantId) - Long

- 내키식 id (userPlantId)- Long
- 식물 이름 애칭 (plantNickname) - String

```
{
  "result" : "SUCCESS",
  "data":
  [
    {
      "scheduleId": 12345,
      "scheduleDate": "2019-11-13",
      "scheduleCode": "003001", // "WATER", // 공통 코드
      "isCompleted" : true,
      "plant": {
        "plantId" : 6789,
        "userPlantId" : 6789,
        "plantNickname" : "수선화",
      }
    },
    {
      "scheduleId": 12346,
      "scheduleDate": "2019-11-13",
      "scheduleCode": "REPOT",
      "isCompleted" : true,
      "plant": {
        "plantId" : 6789,
        "userPlantId" : 6789,
        "plantNickname" : "수선화"
      }
    }
  ]
}
```

## ▼ 주간 스케줄 조회 [GET]

### RequestData

- 조회 년도 (year) - Integer @RequestParam
- 조회 월 (month) - Integer @RequestParam
- 조회 일 (day) - Integer @RequestParam

```
/schedule/week
{
  "year": 2022,
  "month": 11,
  "day": 23,
}
```

### ResponseData

- 스케줄 id (scheduleId): Long
- 식물 스케줄 일자 (scheduleDate) - LocalDate
- 식물 스케줄 카테고리 (scheduleCode) - Type
- 식물 스케줄 완료 유무 (isCompleted) - Boolean
- 식물 (plant)
  - 내키식 식물 id (userPlantId) - Long
  - 식물 이름 애칭 (plantNickname) - String
  - 식물 이미지 (plantImagePath) - String
- 식물 스케줄 내용 - String

```
{
  "result" : "SUCCESS",
  "data":
  [
    {
      "scheduleId": 12345,
      "scheduleDate": "2023-03-03",
      "scheduleCode": "003001", //"WATER",
      "isCompleted" : true,
      "plant": {
        "userPlantId" : 6789,
        "plantNickname" : "새싹이",
        "plantImagePath" : "https://img.freepik.com/free-vector/flat-houseplants-illustrated-collection_23-2148934020.jpg",
      }
      "content" : "새싹이 물주기",
    },
    {
      "scheduleId": 12346,
      "scheduleDate": "2023-03-04",
      "scheduleCode": "WATER",
      "isCompleted" : true,
      "plant": {
        "userPlantId" : 6789,
        "plantNickname" : "새싹이2",
        "plantImagePath" : "https://img.freepik.com/free-vector/flat-houseplants-illustrated-collection_23-2148934020.jpg",
      }
      "content" : "새싹이 물주기2",
    }
  ]
}
```

## ▼ 일정 생성 [POST]

Header에 유저 토큰

RequestData @RequestBody

- 스케줄 날짜 (scheduleDate) - LocalDate
- 스케줄 카테고리 (물주기, 분갈이..) (scheduleCode)
  - Type [WATER | REPOT | ... ] or string
- 식물 종류 (userPlantId) - Long → (사용자가 가지고 있는 식물에 대한 id)
- 내용 (content) - String

### ▼ 추가보류

- 주기 (매일, 매주, 매월) - Type [DAILY | WEEKLY | MONTHLY]
- 주기 텀 - List<Number>
  - MONTHLY - [2] // 매월 몇 일 날짜 지정 ?? or 주기를 날짜로 받기 ex) 25일 간격으로..
  - WEEKLY - [1, 3, 5] // 요일을 숫자로..? ex) 월, 수, 금

```
{
  "scheduleDate": "2019-11-13",
  "scheduleCode": "003001", //"WATER",
  "userPlantId" : 6789,
  "content": "~~",
  "cycle": "004001", //"WEEKLY",
  "cyclePeriod" : 3,
}
```

ResponseData

- 생성 여부 - Boolean

```
{
  "result": "SUCCESS",
  "data": null,
}
```

## ▼ 일정 수정 [PUT]

Header에 유저 토큰

### RequestData @RequestBody

- 스케줄 id - long
- 스케줄 날짜 - LocalDate
- 스케줄 카테고리 (물주기, 분갈이..) - Type [WATER | REPOT | ...]
- 식물 종류 (사용자가 키우는 식물) id - Long

#### ▼ 추가보류 (추가가능)

- 주기 (매일, 매주, 매월) - Type [DAILY | WEEKLY | MONTHLY]
- 주기 텀 - List<Number>
  - MONTHLY - [2] // 매월 몇 일 날짜 지정 ?? or 주기를 날짜로 받기 ex) 25일 간격으로..
  - WEEKLY - [1, 3, 5] // 요일을 숫자로..? ex) 월, 수, 금

```
{
  "scheduleId": 12345,
  "scheduleDate": "2019-11-13",
  "scheduleCode": "003001", //"WATER",
  "plant": {
    "userPlantId" : 6789,
  },
  "content": "수정~",
  "cycle": "WEEKLY",
  "cyclePeriod" : [0, 2, 4],
}
```

### ResponseData

- 수정 여부 - Boolean

```
{
  "result": "SUCCESS",
  "data": null,
}
```

## ▼ 일정 완료 [POST]

Header에 유저 토큰

### RequestData

- 스케줄 id (scheduleId) - long @PathVariable

```
{
  "scheduleId": 12345,
}
```

## ResponseData

- 일정 완료 여부 확인 - Boolean

```
{
  "result": "SUCCESS",
  "data": null,
}
```

### ▼ 일정 완료 취소 [PUT]

Header에 유저 토큰

## RequestData

- 스케줄 id - long @PathVariable

```
{
  "scheduleId": 12345,
}
```

## ResponseData

- 일정 완료 취소 여부 확인 - Boolean

```
{
  "result": "SUCCESS",
  "data": null,
}
```

### ▼ 일정 삭제 [DELETE]

Header에 유저 토큰

## RequestData

- 스케줄 id - long @PathVariable

```
{
  "scheduleId": 12345,
}
```

## ResponseData

- 일정 삭제 여부 확인 - Boolean

```
{
  "result": "SUCCESS",
  "data": null,
}
```



## 알림

### 위로가기

#### ▼ 알림 리스트 조회 [GET]

##### RequestData

- Header에 유저 토큰

##### ResponseData

- 알림 id (alertId) - Long
- 알림 내용 (content) - String
- 알림 클릭했을 때 이동할 path (urlPath) - String

##### ▼ L L

- 알림 타입 Type [FOLLOW | COMMENT | LIKE]  
or Type [FOLLOW | COMMENTPOST | COMMENTDIARY | LIKEPOST | LIKEDIARY]
  - (ex 댓글이 달렸습니다.)
- 알림 관련 내용 id - Long
  - 팔로잉 정보 id (로그인한 사용자가 팔로우한 사용자의 id)
  - 댓글 정보 id (댓글이 달린 포스트 및 일지의 id)
  - 좋아요 정보 id (포스트 및 일지의 id)

```
{
  "result" : "SUCCESS",
  "data":
  [
    {
      "alertId": 1,
      "content": "게시글이 생성되었습니다.",
      "urlPath": "/post/1",
    },
    {
      "alertId": 12,
      "content": "게시글이 생성되었습니다.",
      "urlPath": "/post/1",
    },
    {
      "alertId": 13,
      "content": "게시글이 생성되었습니다.",
      "urlPath": "/post/1",
    },
  ],
}
```

#### ▼ 알림 삭제 [DELETE]

Header에 유저 토큰

##### RequestData

- 알림 id (alertId) - Long @PathVariable

```
{
  "alertId": "12345" // pathVariable
}
```

## ResponseData

- 알림 삭제 여부 - Boolean

```
{
  "result": "SUCCESS",
  "data": null,
}
```

## 팔로잉

### 🔥 위로가기

#### ▼ 팔로잉 생성 [POST]

## Request Data

- 팔로잉 하는 사용자 닉네임 (nickname) - String

```
{
  "nickname": "followingTest" // pathVariable
}
```

## Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS",
  "data": null,
}
```

#### ▼ 팔로잉 리스트 조회 [GET]

## Request Data

- 사용자 닉네임 (nickname) - String

```
{
  "nickname": "test" // pathVariable
}
```

## Response Data

- 사용자의 팔로잉 리스트 list
  - 사용자 닉네임 (nickname) - String
  - 사용자 프로필 이미지 (profileImagePath) - String

- 사용자 한줄소개 (introduction) - String
- 내가 팔로우했는지 여부 (isFollowed) - Boolean

```
{
  "result": "SUCCESS",
  "data": [
    {
      "nickname": "test1",
      "profileImagePath": "/public/images/profile.png",
      "introduction": "안녕하세요 test1 입니다",
      "isFollowed": true, // 내가 팔로우한 여부
    },
    {
      "nickname": "test1",
      "profileImagePath": "/public/images/profile.png",
      "introduction": "안녕하세요 test1 입니다",
      "isFollowed": true, // 내가 팔로우한 여부
    },
    {
      "nickname": "test1",
      "profileImagePath": "/public/images/profile.png",
      "introduction": "안녕하세요 test1 입니다",
      "isFollowed": true, // 내가 팔로우한 여부
    },
  ]
}
```

#### ▼ 팔로잉 삭제 (언팔로우) [DELETE]

##### | Request Data

- 언팔로우할 사용자 닉네임 (nickname) - String

```
{
  "nickname": "test2" // pathVariable
}
```

##### | Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS",
  "data": null,
}
```

#### ▼ 팔로워 리스트 조회 [GET]

##### | Request Data

- 사용자 닉네임 (nickname) - String

```
{
  "nickname": "test" // pathVariable
}
```

##### | Response Data

- 사용자의 팔로워 리스트 array
  - 사용자 닉네임 (nickname) - String
  - 사용자 프로필 이미지 (profileImagePath) - String
  - 사용자 한줄소개 (introduction) - String
  - 내가 팔로우했는지 여부 (isFollowed) - Boolean

```
{
  "result": "SUCCESS",
  "data":
  [
    {
      "nickname": "test1",
      "profileImagePath": "/public/images/profile.png",
      "introduction": "안녕하세요 test1 입니다",
      "isFollowed": true, // 내가 팔로우한 여부
    },
    {
      "nickname": "test1",
      "profileImagePath": "/public/images/profile.png",
      "introduction": "안녕하세요 test1 입니다",
      "isFollowed": true, // 내가 팔로우한 여부
    },
    {
      "nickname": "test1",
      "profileImagePath": "/public/images/profile.png",
      "introduction": "안녕하세요 test1 입니다",
      "isFollowed": true, // 내가 팔로우한 여부
    },
  ],
}
```

#### ▼ 팔로워 삭제 [DELETE]

##### | Request Data

- 팔로워 삭제할 사용자 닉네임 (nickname) - String

```
{
  "nickname" : "test",    // PathVariable
}
```

##### | Response Data

- 상태 코드 (result) - String

```
{
  "result": "SUCCESS",
  "data" : null,
}
```

## 공통 코드

### 🔥 위로가기

#### ▼ 공통 코드 목록 조회 [GET]

##### | Request Data

x

## Response Data

- 공통 코드 목록 Map< 공통 그룹 코드, Map<공통 코드, 공통 코드 설명>>

```
{
  "result": "SUCCESS",
  "data" : {
    "FEED" : {
      "FEED_POST" : "포스트",
      "FEED_DIARY" : "일지",
    },
    "SCHEDULE" : {
      "SCHEDULE_WATER" : "물주기",
      "SCHEDULE_PRUNING" : "가지치기",
    }
  }
}
```