

CareCode API 仕様

Table of Contents

| | |
|------------------------|---|
| 1. 概要 | 2 |
| 1.1. 目的 | 2 |
| 2. 環境 | 2 |
| 2.1. JWT 認証 | 2 |
| 3. API 仕様 | 2 |
| 3.1. ユーザー管理 | 2 |
| 3.1.1. ユーザー登録 | 2 |
| 3.1.2. ユーザー検索 | 2 |
| 3.2. 施設管理 | 3 |
| 3.2.1. 施設登録 | 3 |
| 3.2.2. 施設検索 | 3 |
| 3.3. 投稿管理 | 3 |
| 3.3.1. 投稿登録 | 3 |
| 3.3.2. 投稿検索 | 3 |
| 3.4. 通知管理 | 3 |
| 3.4.1. 通知登録 | 3 |
| 3.4.2. 通知検索 | 4 |
| 3.5. 設定管理 | 4 |
| 3.5.1. 設定登録 | 4 |
| 3.5.2. 設定検索 | 4 |
| 3.6. その他 | 4 |
| 3.6.1. その他1 | 4 |
| 3.6.2. その他2 | 4 |
| 4. 開発環境 | 5 |
| 5. 運用環境 | 5 |
| 6. データベース | 5 |
| 6.1. ユーザー (User) | 5 |
| 6.2. 施設 (CareFacility) | 5 |
| 6.3. 投稿 (Post) | 6 |
| 7. その他 | 6 |
| 7.1. その他1 | 6 |
| 7.2. その他2 | 6 |

1. 개요

CareCode API는 RESTful API입니다.

1.1. API 정보

- **Base URL:** <http://localhost:8080>
- **API 버전:** v1
- **인증:** JWT Bearer Token

2. 인증

2.1. JWT 인증

API는 Authorization 헤더에 JWT 토큰을 사용합니다:

```
Authorization: Bearer <your-jwt-token>
```

3. API 엔드포인트

3.1. 인증 관련

3.1.1. 로그인

```
POST /auth/login
Content-Type: application/json

{
  "email": "user@example.com",
  "password": "password"
}
```

3.1.2. 회원가입

```
POST /auth/register
Content-Type: application/json

{
  "email": "user@example.com",
  "password": "password",
  "name": "이름"
}
```

```
}
```

3.2. □□□□

3.2.1. □□ □□ □□

```
GET /api/facilities?page=0&size=10
Authorization: Bearer <token>
```

3.2.2. □□ □□ □□

```
GET /api/facilities/{id}
Authorization: Bearer <token>
```

3.3. □□□□

3.3.1. □□□□ □□

```
GET /api/community/posts?page=0&size=10
Authorization: Bearer <token>
```

3.3.2. □□□□ □□

```
POST /api/community/posts
Authorization: Bearer <token>
Content-Type: application/json
```

```
{
  "title": "□□",
  "content": "□□",
  "category": "GENERAL"
}
```

3.4. □□□□

3.4.1. □□□□ □□

```
GET /api/health/records?childId={childId}
Authorization: Bearer <token>
```

3.4.2. วัคซีน วัคซีน

```
POST /api/health/records
Authorization: Bearer <token>
Content-Type: multipart/form-data
```

```
{
  "childId": 1,
  "recordType": "VACCINATION",
  "title": "วัคซีน",
  "description": "วัคซีน วัคซีน",
  "recordDate": "2024-01-15"
}
```

3.5. วัคซีน วัคซีน

3.5.1. วัคซีน วัคซีน

```
GET /api/policies?category=EDUCATION&page=0&size=10
Authorization: Bearer <token>
```

3.5.2. วัคซีน วัคซีน

```
GET /api/policies/{id}
Authorization: Bearer <token>
```

3.6. วัคซีน วัคซีน

3.6.1. วัคซีน วัคซีน

```
POST /api/chatbot/sessions
Authorization: Bearer <token>
Content-Type: application/json
```

```
{
  "message": "วัคซีน วัคซีน"
}
```

3.6.2. วัคซีน วัคซีน

```
POST /api/chatbot/sessions/{sessionId}/messages
Authorization: Bearer <token>
Content-Type: application/json
```

```
{
  "message": "API 호출 실패"
}
```

4. HTTP 상태 코드

| 200 | 성공 | | 201 | 생성 | | 400 | 잘못된 요청 | | 401 | 인증 실패 | | 403 | 권한 없음 | | 404 | 자원 없음 | | 500 | 서버 오류 |

5. 예외 처리

```
{
  "timestamp": "2024-01-15T10:30:00",
  "status": 400,
  "error": "Bad Request",
  "message": "잘못된 요청",
  "path": "/api/facilities"
}
```

6. 데이터 모델

6.1. 사용자 (User)

```
{
  "id": 1,
  "email": "user@example.com",
  "name": "홍길동",
  "role": "USER",
  "createdAt": "2024-01-15T10:30:00",
  "updatedAt": "2024-01-15T10:30:00"
}
```

6.2. 시설 (CareFacility)

```
{
  "id": 1,
  "name": "서울 병의원",
  "address": "서울 강남",
  "phone": "02-1234-5678",
  "capacity": 50,
  "currentCount": 30,
}
```

```
"rating": 4.5,
"facilityType": "KINDERGARTEN"
}
```

6.3. 게시물 (Post)

```
{
  "id": 1,
  "title": "제목",
  "content": "내용",
  "author": {
    "id": 1,
    "name": "작성자"
  },
  "category": "GENERAL",
  "createdAt": "2024-01-15T10:30:00",
  "viewCount": 10,
  "likeCount": 5
}
```

7. API 테스트

7.1. 환경 설정

1. 프로젝트 실행: `bash ./gradlew bootRun`
2. Swagger UI 실행: `http://localhost:8080/swagger-ui.html`
3. API 문서 생성: `bash ./gradlew generateApiDocs`

7.2. 테스트

API 테스트 Postman이나 curl을 사용할 수 있습니다:

```
# 로그인
curl -X POST http://localhost:8080/auth/login \
  -H "Content-Type: application/json" \
  -d '{"email":"user@example.com","password":"password"}'

# API 테스트
curl -X GET http://localhost:8080/api/facilities \
  -H "Authorization: Bearer <your-token>"
```