# 감성 AI 다이어리 '새김' - API Specification

## 1. 문서 개요

### 문서 목적

본 문서는 '새김' 서비스의 RESTful API 명세를 정의합니다. 모든 API 엔드포인트, 요청/응답 형식, 에러 처리, 보안 정책을 포함한 완전한 API 가이드를 제공합니다.

### 문서 정보

* **작성일**: 2025년 8월 9일
* **버전**: 1.0
* **작성자**: 새김꾼들
* **관련 문서**: PRD.md, TRD.md, ERD.md, PERSONA.md, FLOW\_CHART.md

### API 버전

* **현재 버전**: v1
* **Base URL**: https://saegim-api.aicc-project.com
* **API Path**: /api

## 2. API 표준

### 2.1 요청 표준

#### HTTP 메서드

| **메서드** | **용도** | **예시** |
| --- | --- | --- |
| GET | 리소스 조회 | 다이어리 목록 조회 |
| POST | 리소스 생성 | 새 다이어리 작성 |
| PUT | 리소스 전체 수정 | 다이어리 수정 |
| PATCH | 리소스 부분 수정 | 다이어리 상태 변경 |
| DELETE | 리소스 삭제 | 다이어리 삭제 |

#### 요청 헤더

Content-Type: application/json

Accept: application/json

Authorization: Bearer {access\_token}

X-Request-ID: {uuid}

Accept-Language: ko-KR

#### 요청 본문 인코딩

* **문자 인코딩**: UTF-8
* **날짜 형식**: ISO 8601 (예: 2025-08-09T12:00:00Z)
* **ID 형식**: UUID v4

### 2.2 응답 표준

#### 표준 응답 구조

interface BaseResponse<T = any> {

success: boolean; // 요청 성공 여부

data: T | null; // 응답 데이터

message?: string; // 사용자 메시지 (선택적)

timestamp: string; // ISO 8601 형식 (예: "2025-08-09T12:00:00Z")

request\_id: string; // 요청 추적 ID (UUID)

}

**성공 응답 예시:**

{

"success": true,

"data": {

// 실제 응답 데이터

},

"message": null,

"timestamp": "2025-08-09T12:00:00Z",

"request\_id": "550e8400-e29b-41d4-a716-446655440000"

}

#### 에러 응답 구조

interface ErrorDetail {

code: string; // 에러 코드 (예: "AUTH\_TOKEN\_EXPIRED")

field?: string; // 관련 필드명 (폼 검증 오류 시)

message: string; // 에러 상세 메시지

}

interface ErrorResponse {

success: false; // 항상 false

data: null; // 항상 null

message: string; // 에러 요약 메시지

errors?: ErrorDetail[]; // 에러 상세 목록 (선택적)

timestamp: string; // ISO 8601 형식

request\_id: string; // 요청 추적 ID (UUID)

}

**에러 응답 예시:**

{

"success": false,

"data": null,

"message": "요청 처리 중 오류가 발생했습니다.",

"errors": [

{

"code": "AUTH\_TOKEN\_EXPIRED",

"field": null,

"message": "액세스 토큰이 만료되었습니다."

}

],

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "550e8400-e29b-41d4-a716-446655440000"

}

#### 페이지네이션 응답 구조

interface PaginationInfo {

page: number; // 현재 페이지 (1부터 시작)

page\_size: number; // 페이지 크기

total\_items: number; // 전체 항목 수

total\_pages: number; // 전체 페이지 수

has\_next: boolean; // 다음 페이지 존재 여부

has\_previous: boolean; // 이전 페이지 존재 여부

}

interface PaginatedResponse<T> {

success: true; // 항상 true

data: T[]; // 데이터 배열

pagination: PaginationInfo; // 페이지네이션 정보

message?: string; // 사용자 메시지 (선택적)

timestamp: string; // ISO 8601 형식

request\_id: string; // 요청 추적 ID (UUID)

}

**페이지네이션 응답 예시:**

{

"success": true,

"data": [...],

"pagination": {

"page": 1,

"page\_size": 20,

"total\_items": 100,

"total\_pages": 5,

"has\_next": true,

"has\_previous": false

},

"message": null,

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "550e8400-e29b-41d4-a716-446655440000"

}

### 2.3 HTTP 상태 코드

| **코드** | **의미** | **사용 케이스** |
| --- | --- | --- |
| 200 | OK | 성공적인 GET, PUT 요청 |
| 201 | Created | 성공적인 POST 요청 |
| 204 | No Content | 성공적인 DELETE 요청 |
| 400 | Bad Request | 잘못된 요청 형식 |
| 401 | Unauthorized | 인증 실패 |
| 403 | Forbidden | 권한 없음 |
| 404 | Not Found | 리소스 없음 |
| 422 | Unprocessable Entity | 검증 실패 |
| 429 | Too Many Requests | Rate Limit 초과 |
| 500 | Internal Server Error | 서버 오류 |
| 503 | Service Unavailable | 서비스 일시 중단 |

## 3. 감정 분석 시스템

### 3.1 감정 처리 로직

'새김' 서비스의 감정 분석은 **AI 자동 분석 + 사용자 선택적 수정** 방식으로 동작합니다.

#### 감정 처리 플로우

1. **AI 자동 분석**: 다이어리 글과 업로드된 사진을 AI가 분석하여 감정 자동 추출
2. **신뢰도 저장**: AI 분석 결과와 함께 신뢰도 (0.0-1.0) 저장
3. **사용자 수정 옵션**: 사용자가 원할 경우 AI 분석 결과를 수동으로 수정 가능

#### 감정 데이터 구조

{

"ai\_emotion": "happy", // AI가 분석한 감정 (자동)

"ai\_emotion\_confidence": 0.92, // AI 분석 신뢰도 (자동)

"user\_emotion": null // 사용자 수정 감정 (선택사항)

}

#### 감정 타입

* **happy** (😊): 행복, 기쁨, 즐거움
* **sad** (😢): 슬픔, 우울, 아쉬움
* **angry** (😡): 화남, 짜증, 분노
* **peaceful** (😌): 평온, 차분, 안정
* **worried** (🫨): 걱정, 불안, 불안

#### API 동작 방식

* **다이어리 작성**: user\_emotion 필드는 선택사항. 미입력 시 AI 분석 결과만 저장
* **다이어리 수정**: user\_emotion 필드로 AI 분석 감정 수정 가능
* **다이어리 조회**: ai\_emotion과 user\_emotion 모두 반환하여 클라이언트에서 최종 감정 결정

## 4. 에러 코드 체계

### 4.1 에러 코드 분류

| **범위** | **카테고리** | **설명** |
| --- | --- | --- |
| 1000-1999 | AUTH | 인증/인가 관련 |
| 2000-2999 | DIARY | 다이어리 관련 |
| 3000-3999 | AI | AI 서비스 관련 |
| 4000-4999 | IMAGE | 이미지 처리 관련 |
| 5000-5999 | VALIDATION | 입력 검증 관련 |
| 9000-9999 | SYSTEM | 시스템 관련 |

### 4.2 주요 에러 코드

#### 인증 에러 (1000-1999)

| **코드** | **에러 ID** | **설명** |
| --- | --- | --- |
| AUTH\_SOCIAL\_PROVIDER\_ERROR | 1001 | 소셜 제공자 오류 |
| AUTH\_SOCIAL\_TOKEN\_INVALID | 1002 | 유효하지 않은 OAuth 토큰 |
| AUTH\_SOCIAL\_USER\_DENIED | 1003 | 사용자가 권한 거부 |
| AUTH\_SOCIAL\_EMAIL\_REQUIRED | 1004 | 이메일 정보 필수 |
| AUTH\_SOCIAL\_PROVIDER\_UNSUPPORTED | 1005 | 지원하지 않는 제공자 |
| AUTH\_TOKEN\_EXPIRED | 1006 | 액세스 토큰 만료 |
| AUTH\_TOKEN\_INVALID | 1007 | 유효하지 않은 토큰 |
| AUTH\_PERMISSION\_DENIED | 1008 | 권한 없음 |
| AUTH\_REFRESH\_TOKEN\_EXPIRED | 1009 | 리프레시 토큰 만료 |

#### 다이어리 에러 (2000-2999)

| **코드** | **에러 ID** | **설명** |
| --- | --- | --- |
| DIARY\_NOT\_FOUND | 2001 | 다이어리를 찾을 수 없음 |
| DIARY\_ACCESS\_DENIED | 2002 | 다이어리 접근 권한 없음 |
| DIARY\_UPDATE\_DENIED | 2003 | 다이어리 수정 권한 없음 |
| DIARY\_DELETE\_DENIED | 2004 | 다이어리 삭제 권한 없음 |
| DIARY\_LIMIT\_EXCEEDED | 2005 | 일일 다이어리 작성 한도 초과 |

#### AI 서비스 에러 (3000-3999)

| **코드** | **에러 ID** | **설명** |
| --- | --- | --- |
| AI\_SERVICE\_UNAVAILABLE | 3001 | AI 서비스 일시적 오류 |
| AI\_TOKEN\_LIMIT\_EXCEEDED | 3002 | AI 토큰 한도 초과 |
| AI\_INVALID\_STYLE | 3003 | 지원하지 않는 스타일 옵션 |
| AI\_GENERATION\_FAILED | 3004 | AI 텍스트 생성 실패 |
| AI\_RATE\_LIMIT\_EXCEEDED | 3005 | AI API 호출 한도 초과 |
| AI\_REGENERATION\_FAILED | 3006 | AI 재생성 실패 |
| AI\_GENERATION\_NOT\_FOUND | 3007 | 재생성 대상을 찾을 수 없음 |

#### 이미지 에러 (4000-4999)

| **코드** | **에러 ID** | **설명** |
| --- | --- | --- |
| IMG\_SIZE\_EXCEEDED | 4001 | 파일 크기 초과 (15MB) |
| IMG\_INVALID\_FORMAT | 4002 | 지원하지 않는 이미지 형식 |
| IMG\_UPLOAD\_FAILED | 4003 | 이미지 업로드 실패 |
| IMG\_NOT\_FOUND | 4004 | 이미지를 찾을 수 없음 |
| IMG\_PROCESSING\_FAILED | 4005 | 이미지 처리 실패 |
| IMG\_COUNT\_EXCEEDED | 4006 | 이미지 갯수 초과 (최대 10개) |

#### 검증 에러 (5000-5999)

| **코드** | **에러 ID** | **설명** |
| --- | --- | --- |
| VALIDATION\_FAILED | 5001 | 입력 데이터 검증 실패 |
| VALIDATION\_EMAIL\_INVALID | 5002 | 유효하지 않은 이메일 형식 |
| VALIDATION\_PASSWORD\_WEAK | 5003 | 비밀번호 복잡도 미달 |
| VALIDATION\_REQUIRED\_FIELD | 5004 | 필수 필드 누락 |

#### 시스템 에러 (9000-9999)

| **코드** | **에러 ID** | **설명** |
| --- | --- | --- |
| SYSTEM\_ERROR | 9001 | 서버 내부 오류 |
| SYSTEM\_DB\_ERROR | 9002 | 데이터베이스 연결 실패 |
| SYSTEM\_STORAGE\_ERROR | 9003 | 스토리지 서비스 오류 |
| SYSTEM\_RATE\_LIMIT | 9004 | API 호출 한도 초과 |
| SYSTEM\_MAINTENANCE | 9005 | 시스템 점검 중 |

### 3.3 에러 응답 형식

모든 에러 응답은 다음과 같은 표준화된 형식을 따릅니다:

{

"success": false,

"data": null,

"message": "오류 요약 메시지",

"errors": [

{

"code": "에러 코드 (예: AUTH\_SOCIAL\_PROVIDER\_ERROR)",

"error\_id": 에러 ID (예: 1001),

"field": "관련 필드명 (선택적)",

"message": "상세 에러 메시지"

}

],

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_abc123"

}

## 4. 인증 및 보안

### 4.1 인증 방식

* **인증 타입**: JWT Bearer Token
* **토큰 위치**: Authorization 헤더
* **토큰 형식**: Bearer {access\_token}

### 4.2 토큰 정책

| **토큰 타입** | **만료 시간** | **용도** |
| --- | --- | --- |
| Access Token | 1시간 | API 호출 인증 |
| Refresh Token | 7일 | Access Token 갱신 |

### 4.3 캐싱 정책

#### 4.3.1 캐시 전략 개요

**캐싱 계층 구조:**

| **캐시 레벨** | **기술** | **대상 데이터** | **TTL** | **목적** |
| --- | --- | --- | --- | --- |
| **브라우저** | HTTP Cache | 정적 리소스 | 24시간 | CDN 트래픽 감소 |
| **CDN** | CloudFront | 이미지, 정적 파일 | 24시간 | 글로벌 접근 최적화 |
| **애플리케이션** | Redis | API 응답, 세션 | 가변 | 응답 시간 단축 |
| **데이터베이스** | 쿼리 캐시 | 자주 조회되는 데이터 | 자동 | DB 부하 감소 |

#### 4.3.2 API 응답별 캐싱 정책

**AI 관련 API:**

* **AI 글귀 생성**: 1시간 TTL, 사용자별 개인화 캐시
* **AI 키워드 추천**: 30분 TTL, 입력 해시 기반 캐시
* **감정 리포트**: 24시간 TTL, 월간 데이터 캐시

**다이어리 API:**

* **다이어리 목록**: 10분 TTL, 사용자별 캐시
* **다이어리 상세**: 30분 TTL, 읽기 전용 캐시
* **통계 데이터**: 1시간 TTL, 집계 결과 캐시

**이미지 API:**

* **이미지 메타데이터**: 24시간 TTL, 변경 빈도 낮음
* **썸네일**: 7일 TTL, 영구 캐시에 가까움
* **합성 이미지**: 1시간 TTL, 임시 결과물

#### 4.3.3 캐시 관련 HTTP 헤더

**클라이언트 캐시 제어:**

Cache-Control: max-age=3600, private # 1시간 개인 캐시

Cache-Control: max-age=86400, public # 24시간 공개 캐시

Cache-Control: no-cache, private # 캐시 사용 안함

ETag: "abc123def456" # 리소스 버전 식별

Last-Modified: Wed, 08 Jan 2025 12:00:00 GMT

**캐시 상태 정보:**

X-Cache-Status: HIT | MISS | STALE | UPDATING

X-Cache-Key: ai:gen:hash\_abc123 # Redis 캐시 키

X-Cache-TTL: 3600 # 남은 캐시 시간(초)

X-Cache-Age: 120 # 캐시 생성 후 경과 시간(초)

#### 4.3.4 캐시 무효화 정책

**자동 무효화:**

* 사용자 데이터 변경 시 관련 캐시 자동 삭제
* AI 재생성 요청 시 이전 결과 캐시 무효화
* 감정 통계 업데이트 시 월간 캐시 갱신

**수동 무효화 (관리자):**

DELETE /api/cache/invalidate

Content-Type: application/json

Authorization: Bearer {admin\_token}

{

"cache\_keys": ["ai:gen:\*", "stats:emotion:\*"],

"user\_id": "550e8400-e29b-41d4-a716-446655440000"

}

### 4.4 Rate Limiting

| **API 타입** | **제한** | **단위** | **설명** |
| --- | --- | --- | --- |
| 일반 API | 1,000 | requests/minute | 일반 읽기/쓰기 작업 |
| AI API | 30 | requests/minute | AI 텍스트 생성 요청 (전환경 공통) |
| 이미지 업로드 | 60 | uploads/hour | 이미지 업로드 (최대 15MB/파일) |
| 버스트 허용 | 2,000 | requests/minute | 순간 최대 허용 (10초간) |

**Rate Limiting 응답 헤더:**

X-RateLimit-Limit: 1000

X-RateLimit-Remaining: 999

X-RateLimit-Reset: 1641628800

**초과 시 응답 (429 Too Many Requests):**

{

"success": false,

"data": null,

"message": "API 호출 한도를 초과했습니다.",

"errors": [

{

"code": "SYSTEM\_RATE\_LIMIT",

"error\_id": 9004,

"message": "API rate limit exceeded. Please retry after 60 seconds.",

"retry\_after": 60

}

],

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_abc123"

}

### 4.4 보안 헤더

X-Content-Type-Options: nosniff

X-Frame-Options: DENY

X-XSS-Protection: 1; mode=block

Strict-Transport-Security: max-age=31536000; includeSubDomains; preload

Content-Security-Policy: default-src 'self'

### 4.5 암호화 정책

#### 4.5.1 데이터 전송 암호화

* **프로토콜**: TLS 1.3 이상 필수
* **암호화 스위트**: TLS\_AES\_256\_GCM\_SHA384, TLS\_CHACHA20\_POLY1305\_SHA256
* **인증서**: Let's Encrypt 또는 상용 CA 인증서
* **HSTS Preload**: 브라우저 HSTS preload 리스트 등록

#### 4.5.2 데이터 저장 암호화

* **비밀번호**: bcrypt (cost factor: 12)
* **민감 데이터**: AES-256-GCM
* **암호화 대상 필드**:
  + diaries.content - 일기 내용
  + diaries.ai\_generated\_text - AI 생성 텍스트
  + users.email - 이메일 (검색용 SHA-256 해시 별도 저장)
* **이미지 파일**: MinIO SSE-S3 암호화

#### 4.5.3 키 관리

* **키 저장**:
  + 개발: 환경 변수 (.env 파일)
  + 운영: 서버 환경 변수 또는 Docker secrets
* **키 로테이션**: 90일 주기
* **키 백업**: 암호화된 백업 3개 이상 유지

## 5. API 엔드포인트

### 5.1 인증 (Authentication)

#### 소셜 로그인 시작

GET /api/auth/social/{provider}/login

**경로 매개변수:**

* provider: google | kakao | naver

**성공 응답 (302 Redirect):**

OAuth 인증 서버로 리다이렉트

Location: https://accounts.google.com/o/oauth2/v2/auth?client\_id=xxx&redirect\_uri=xxx&response\_type=code&scope=openid%20email%20profile&state=xxx

**에러 응답 (400 Bad Request):**

{

"success": false,

"data": null,

"message": "지원하지 않는 소셜 로그인 제공자입니다.",

"errors": [

{

"code": "AUTH\_SOCIAL\_PROVIDER\_UNSUPPORTED",

"error\_id": 1005,

"field": "provider",

"message": "google, kakao, naver만 지원됩니다."

}

],

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_abc123"

}

#### 소셜 로그인 콜백

GET /api/auth/social/{provider}/callback?code={code}&state={state}

**경로 매개변수:**

* provider: google | kakao | naver

**쿼리 매개변수:**

* code: OAuth 인증 코드
* state: CSRF 방지용 상태 값

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"access\_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",

"refresh\_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",

"token\_type": "bearer",

"expires\_in": 3600,

"user": {

"user\_id": "550e8400-e29b-41d4-a716-446655440000",

"email": "encrypted\_email",

"nickname": "새김이",

"provider": "google"

}

},

"message": "로그인에 성공했습니다.",

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_def456"

}

**에러 응답 (400 Bad Request):**

{

"success": false,

"data": null,

"message": "소셜 로그인에 실패했습니다.",

"errors": [

{

"code": "AUTH\_SOCIAL\_TOKEN\_INVALID",

"error\_id": 1002,

"field": "code",

"message": "유효하지 않은 OAuth 인증 코드입니다."

}

],

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_def456"

}

#### 소셜 계정 연결 해제

DELETE /api/auth/social/{provider}/disconnect

**요청 헤더:**

Authorization: Bearer {access\_token}

**경로 매개변수:**

* provider: google | kakao | naver

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"provider": "google",

"disconnected\_at": "2025-01-08T12:00:00Z"

},

"message": "소셜 계정 연결이 해제되었습니다.",

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_ghi789"

}

#### 토큰 갱신

POST /api/auth/refresh

**요청 헤더:**

Authorization: Bearer {refresh\_token}

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"access\_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",

"token\_type": "bearer",

"expires\_in": 3600

},

"message": null,

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_ghi789"

}

#### 로그아웃

POST /api/auth/logout

**요청 헤더:**

Authorization: Bearer {access\_token}

**성공 응답 (200 OK):**

{

"success": true,

"data": null,

"message": "로그아웃되었습니다.",

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_jkl012"

}

### 5.3 다이어리 (Diary)

#### 다이어리 작성

POST /api/diary

**요청 헤더:**

Authorization: Bearer {access\_token}

Content-Type: application/json

**요청 본문:**

{

"title": "제주도 여행 첫날",

"content": "오늘은 제주도에 도착한 첫날이다. 날씨가 너무 좋아서...",

"user\_emotion": "happy",

"images": [

"550e8400-e29b-41d4-a716-446655440001",

"550e8400-e29b-41d4-a716-446655440002"

],

"ai\_generated\_text": "푸른 바다가 속삭이는 제주의 첫날...",

"is\_public": false

}

**검증 규칙:**

* title: 최대 255자 (선택)
* content: 텍스트 (선택)
* user\_emotion: enum (happy, sad, angry, peaceful, worried) - 선택사항, AI 분석 감정을 사용자가 수정하고 싶을 때 입력
* images: UUID 배열, 최대 10개

**성공 응답 (201 Created):**

{

"success": true,

"data": {

"diary\_id": "550e8400-e29b-41d4-a716-446655440003",

"title": "제주도 여행 첫날",

"created\_at": "2025-01-08T12:00:00Z"

},

"message": "다이어리가 성공적으로 작성되었습니다.",

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_mno345"

}

#### 다이어리 목록 조회

GET /api/diary?page=1&page\_size=20&month=2025-01&emotion=happy

**요청 헤더:**

Authorization: Bearer {access\_token}

**쿼리 파라미터:** | 파라미터 | 타입 | 필수 | 설명 | |----------|------|------|------| | page | integer | N | 페이지 번호 (기본: 1) | | page\_size | integer | N | 페이지 크기 (기본: 20, 최대: 100) | | month | string | N | 조회 월 (YYYY-MM 형식) | | user\_emotion | string | N | 사용자가 수정한 감정으로 필터링 | | ai\_emotion | string | N | AI가 분석한 감정으로 필터링 |

**성공 응답 (200 OK):**

{

"success": true,

"data": [

{

"diary\_id": "550e8400-e29b-41d4-a716-446655440003",

"title": "제주도 여행 첫날",

"user\_emotion": "happy",

"ai\_emotion": "happy",

"ai\_emotion\_confidence": 0.95,

"thumbnail": "https://cdn.saegim.app/thumbnails/abc123.jpg",

"created\_at": "2025-01-08T12:00:00Z"

},

{

"diary\_id": "550e8400-e29b-41d4-a716-446655440004",

"title": "카페에서의 오후",

"user\_emotion": "peaceful",

"ai\_emotion": "peaceful",

"ai\_emotion\_confidence": 0.88,

"thumbnail": null,

"created\_at": "2025-01-07T15:30:00Z"

}

],

"pagination": {

"page": 1,

"page\_size": 20,

"total\_items": 42,

"total\_pages": 3,

"has\_next": true,

"has\_previous": false

},

"message": null,

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_pqr678"

}

#### 다이어리 상세 조회

GET /api/diary/{diary\_id}

**요청 헤더:**

Authorization: Bearer {access\_token}

**경로 파라미터:**

* diary\_id: 다이어리 UUID

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"diary\_id": "550e8400-e29b-41d4-a716-446655440003",

"title": "제주도 여행 첫날",

"content": "오늘은 제주도에 도착한 첫날이다. 날씨가 너무 좋아서...",

"user\_emotion": "happy",

"ai\_emotion": "happy",

"ai\_emotion\_confidence": 0.95,

"images": [

{

"image\_id": "550e8400-e29b-41d4-a716-446655440001",

"url": "https://cdn.saegim.app/images/abc123.jpg",

"thumbnail\_url": "https://cdn.saegim.app/thumbnails/abc123.jpg"

}

],

"ai\_generated\_text": "푸른 바다가 속삭이는 제주의 첫날...",

"is\_public": false,

"created\_at": "2025-01-08T12:00:00Z",

"updated\_at": "2025-01-08T12:00:00Z"

},

"message": null,

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_stu901"

}

#### 다이어리 수정

PUT /api/diary/{diary\_id}

**요청 헤더:**

Authorization: Bearer {access\_token}

Content-Type: application/json

**요청 본문:**

{

"title": "제주도 여행 첫날 (수정)",

"content": "오늘은 제주도에 도착한 첫날이다. 날씨가 정말 화창해서...",

"user\_emotion": "happy",

}

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"diary\_id": "550e8400-e29b-41d4-a716-446655440003",

"updated\_at": "2025-01-08T13:00:00Z"

},

"message": "다이어리가 수정되었습니다.",

"timestamp": "2025-01-08T13:00:00Z",

"request\_id": "req\_vwx234"

}

#### 다이어리 삭제

DELETE /api/diary/{diary\_id}

**요청 헤더:**

Authorization: Bearer {access\_token}

**성공 응답 (200 OK):**

{

"success": true,

"data": null,

"message": "다이어리가 삭제되었습니다.",

"timestamp": "2025-01-08T13:00:00Z",

"request\_id": "req\_yza567"

}

### 5.3 AI 서비스 (AI)

#### AI 글귀 생성

POST /api/ai/generate

**요청 헤더:**

Authorization: Bearer {access\_token}

Content-Type: application/json

**요청 본문:**

{

"sentence": "오늘 하루가 선물 같았다",

// 감정은 AI가 글과 사진을 분석하여 자동 추출 (사용자 수정 가능)

"style": {

"tone": "romantic",

"length": "medium",

"type": "poem"

}

}

**스타일 옵션:**

* tone: romantic, healing, calm, humorous
* length: short (4-8줄), medium (8-16줄), long (16줄 이상)
* type: poem, prose, quote

**성공 응답 (200 OK):**

**응답 헤더:**

Cache-Control: max-age=3600, private

X-Cache-Status: MISS

X-Cache-Key: ai:gen:hash\_abc123

X-Cache-TTL: 3600

X-Rate-Limit-Remaining: 29

X-Rate-Limit-Reset: 1641646800

**응답 헤더:**

Cache-Control: max-age=3600, private

X-Cache-Status: MISS

X-Cache-Key: ai:gen:hash\_abc123

X-Cache-TTL: 3600

X-Rate-Limit-Remaining: 29

X-Rate-Limit-Reset: 1641646800

{

"success": true,

"data": {

"generated\_text": "노을이 물든 바다 위로\n평화가 내려앉는 저녁\n오늘 하루가 선물처럼\n마음에 고이 담겨진다\n\n파도 소리는 자장가가 되고\n붉은 하늘은 편지가 되어\n내일을 기다리는 설렘으로\n오늘을 부드럽게 마무리한다",

"style\_applied": {

"tone": "romantic",

"length": "medium",

"type": "poem"

},

"tokens\_used": 150,

"generation\_id": "gen\_abc123",

"cache\_info": {

"cached": false,

"cache\_key": "ai:gen:hash\_abc123",

"ttl\_seconds": 3600,

"expires\_at": "2025-01-08T13:00:00Z"

},

"cache\_info": {

"cached": false,

"cache\_key": "ai:gen:hash\_abc123",

"ttl\_seconds": 3600,

"expires\_at": "2025-01-08T13:00:00Z"

}

},

"message": null,

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_bcd890"

}

#### AI 글귀 재생성

POST /api/ai/regenerate

**요청 헤더:**

Authorization: Bearer {access\_token}

Content-Type: application/json

**요청 본문:**

{

"generation\_id": "gen\_abc123",

"sentence": "오늘 하루가 선물 같았다",

"style": {

"tone": "healing",

"length": "short",

"type": "poem"

},

"exclude\_previous": true

}

**요청 필드:** | 필드명 | 타입 | 필수 | 설명 | |--------|------|------|------| | generation\_id | string | N | 이전 생성 ID (캐시 무효화용) | | sentence | string | N | 입력 문장 | | style | object | N | 스타일 설정 | | exclude\_previous | boolean | N | 이전 결과 제외 여부 (기본: false) |

**성공 응답 (200 OK):**

**응답 헤더:**

Cache-Control: no-cache, private

X-Cache-Status: REGENERATED

X-Cache-Invalidated: ai:gen:hash\_abc123

X-Rate-Limit-Remaining: 28

X-Rate-Limit-Reset: 1641646800

**응답 헤더:**

Cache-Control: no-cache, private

X-Cache-Status: REGENERATED

X-Cache-Invalidated: ai:gen:hash\_abc123

X-Rate-Limit-Remaining: 28

X-Rate-Limit-Reset: 1641646800

{

"success": true,

"data": {

"generated\_text": "고요한 바다가 건네는\n평화로운 인사말\n노을빛 물결 속에서\n하루의 마침표를 찍는다",

"style\_applied": {

"tone": "healing",

"length": "short",

"type": "poem"

},

"tokens\_used": 120,

"generation\_id": "gen\_def456",

"is\_regenerated": true,

"cache\_info": {

"cached": false,

"previous\_cache\_invalidated": "ai:gen:hash\_abc123",

"new\_cache\_key": "ai:gen:hash\_def456",

"ttl\_seconds": 3600,

"expires\_at": "2025-01-08T13:05:00Z"

},

"cache\_info": {

"cached": false,

"previous\_cache\_invalidated": "ai:gen:hash\_abc123",

"new\_cache\_key": "ai:gen:hash\_def456",

"ttl\_seconds": 3600,

"expires\_at": "2025-01-08T13:05:00Z"

}

},

"message": "새로운 글귀가 생성되었습니다.",

"timestamp": "2025-01-08T12:05:00Z",

"request\_id": "req\_regen789"

}

#### 월간 감정 리포트

GET /api/ai/emotion-report?year=2025&month=1

**요청 헤더:**

Authorization: Bearer {access\_token}

**쿼리 파라미터:** | 파라미터 | 타입 | 필수 | 설명 | |----------|------|------|------| | year | integer | Y | 년도 | | month | integer | Y | 월 (1-12) |

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"period": "2025-01",

"total\_entries": 25,

"emotion\_distribution": {

"happy": 12,

"peaceful": 7,

"sad": 3,

"worried": 2,

"angry": 1

},

"emotion\_percentage": {

"happy": 48.0,

"peaceful": 28.0,

"sad": 12.0,

"worried": 8.0,

"angry": 4.0

},

"keyword\_cloud": [

{ "word": "여행", "count": 5, "weight": 0.9 },

{ "word": "가족", "count": 4, "weight": 0.8 },

{ "word": "일상", "count": 4, "weight": 0.8 },

{ "word": "행복", "count": 3, "weight": 0.7 },

{ "word": "휴식", "count": 3, "weight": 0.7 }

],

"ai\_summary": "이번 달은 전반적으로 긍정적인 감정이 우세했습니다. 특히 여행과 가족과의 시간이 행복감의 주요 원천이었으며, 일상 속에서도 작은 기쁨을 찾는 모습이 보입니다. (AI가 글과 사진을 분석한 감정 기반)",

"trend\_analysis": "월초 대비 월말로 갈수록 평온한 감정이 증가하는 추세를 보였습니다. (AI 분석 감정 기반)",

"recommendations": [

"긍정적인 감정 상태를 유지하기 위해 규칙적인 여가 활동을 계속하세요.",

"스트레스가 있었던 날들을 되돌아보고 대처 방법을 기록해보세요."

]

},

"message": null,

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_hij456"

}

### 5.5 알림 (Notification)

#### 알림 목록 조회

GET /api/notifications

**요청 헤더:**

Authorization: Bearer {access\_token}

Content-Type: application/json

**쌍리 매개변수:**

* page: 페이지 번호 (기본값: 1)
* limit: 페이지당 아이템 수 (기본값: 20, 최대: 100)
* type: 알림 유형 필터 (diary\_reminder | report\_ready | ai\_complete)
* is\_read: 읽음 상태 필터 (true | false)

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"notifications": [

{

"id": "550e8400-e29b-41d4-a716-446655440001",

"user\_id": "550e8400-e29b-41d4-a716-446655440000",

"type": "diary\_reminder",

"title": "다이어리 작성 알림",

"message": "오늘의 감정을 새김에 기록해보세요.",

"data": {

"action": "write\_diary",

"suggested\_emotion": "peaceful"

},

"is\_read": false,

"scheduled\_at": "2025-08-09T21:00:00Z",

"created\_at": "2025-08-09T21:00:00Z"

},

{

"id": "550e8400-e29b-41d4-a716-446655440002",

"user\_id": "550e8400-e29b-41d4-a716-446655440000",

"type": "report\_ready",

"title": "월간 감정 리포트 준비 완료",

"message": "8월 감정 리포트가 준비되었습니다. 확인해보세요!",

"data": {

"action": "view\_report",

"report\_id": "report\_2025\_08",

"month": "2025-08"

},

"is\_read": true,

"scheduled\_at": "2025-08-31T09:00:00Z",

"created\_at": "2025-08-31T09:00:00Z"

}

],

"pagination": {

"current\_page": 1,

"total\_pages": 3,

"total\_items": 45,

"has\_next": true

}

},

"message": null,

"timestamp": "2025-08-09T12:00:00Z",

"request\_id": "req\_notif123"

}

#### 알림 읽음 처리

PATCH /api/notifications/{notification\_id}/read

**경로 매개변수:**

* notification\_id: 알림 ID (UUID)

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"id": "550e8400-e29b-41d4-a716-446655440001",

"is\_read": true,

"read\_at": "2025-08-09T12:30:00Z"

},

"message": "알림을 읽음 처리했습니다.",

"timestamp": "2025-08-09T12:30:00Z",

"request\_id": "req\_read123"

}

#### 알림 설정 조회

GET /api/notifications/settings

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"diary\_reminder": {

"enabled": true,

"time": "21:00",

"days": ["mon", "tue", "wed", "thu", "fri", "sat", "sun"]

},

"report\_notification": {

"enabled": true,

"monthly\_report": true

},

"ai\_processing": {

"enabled": true,

"completion\_alerts": true

},

"browser\_push": {

"enabled": true,

"permission\_granted": true

}

},

"message": null,

"timestamp": "2025-08-09T12:00:00Z",

"request\_id": "req\_settings123"

}

#### 알림 설정 업데이트

PUT /api/notifications/settings

**요청 본문:**

{

"diary\_reminder": {

"enabled": true,

"time": "20:30",

"days": ["mon", "wed", "fri"]

},

"report\_notification": {

"enabled": true,

"monthly\_report": true

},

"ai\_processing": {

"enabled": false,

"completion\_alerts": false

},

"browser\_push": {

"enabled": true

}

}

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"updated\_settings": {

"diary\_reminder": {

"enabled": true,

"time": "20:30",

"days": ["mon", "wed", "fri"]

},

"report\_notification": {

"enabled": true,

"monthly\_report": true

},

"ai\_processing": {

"enabled": false,

"completion\_alerts": false

},

"browser\_push": {

"enabled": true,

"permission\_granted": true

}

}

},

"message": "알림 설정이 업데이트되었습니다.",

"timestamp": "2025-08-09T12:15:00Z",

"request\_id": "req\_update123"

}

### 5.6 이미지 (Image)

#### 이미지 업로드

POST /api/image/upload

**요청 헤더:**

Authorization: Bearer {access\_token}

Content-Type: multipart/form-data

**요청 본문:**

* files: 이미지 파일 배열 (binary[])
* 최대 파일 갯수: 10개
* 파일당 최대 크기: 15MB
* 총 업로드 크기: 120MB
* 지원 형식: JPEG, PNG, WebP

**성공 응답 (201 Created):**

{

"success": true,

"data": {

"image\_id": "550e8400-e29b-41d4-a716-446655440005",

"url": "https://cdn.saegim.app/images/def456.jpg",

"thumbnail\_url": "https://cdn.saegim.app/thumbnails/def456.jpg",

"size": 2048576,

"mime\_type": "image/jpeg",

"width": 3000,

"height": 2000,

"exif\_removed": true

},

"message": "이미지가 성공적으로 업로드되었습니다.",

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_klm789"

}

#### 이미지 합성

POST /api/image/synthesis

**요청 헤더:**

Authorization: Bearer {access\_token}

Content-Type: application/json

**요청 본문:**

{

"image\_id": "550e8400-e29b-41d4-a716-446655440005",

"text": "노을이 물든 바다 위로\n평화가 내려앉는 저녁",

"template\_id": "template\_romantic\_01",

"options": {

"font": "NanumMyeongjo",

"font\_size": 24,

"color": "#FFFFFF",

"position": "bottom",

"opacity": 0.9,

"shadow": true

}

}

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"synthesized\_image\_url": "https://cdn.saegim.app/synthesized/ghi789.jpg",

"download\_url": "https://cdn.saegim.app/download/ghi789.jpg",

"expires\_at": "2025-01-08T13:00:00Z",

"synthesis\_id": "syn\_nop012"

},

"message": null,

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_nop012"

}

#### 이미지 삭제

DELETE /api/image/{image\_id}

**요청 헤더:**

Authorization: Bearer {access\_token}

**성공 응답 (200 OK):**

{

"success": true,

"data": null,

"message": "이미지가 삭제되었습니다.",

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_qrs345"

}

### 5.5 통계 (Statistics)

#### 감정 통계 조회

GET /api/stats/emotions?start\_date=2025-01-01&end\_date=2025-01-31

**요청 헤더:**

Authorization: Bearer {access\_token}

**쿼리 파라미터:** | 파라미터 | 타입 | 필수 | 설명 | |----------|------|------|------| | start\_date | string | Y | 시작 날짜 (YYYY-MM-DD) | | end\_date | string | Y | 종료 날짜 (YYYY-MM-DD) |

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"period": {

"start": "2025-01-01",

"end": "2025-01-31"

},

"daily\_emotions": [

{

"date": "2025-01-01",

"emotions": {

"happy": 2,

"peaceful": 1

}

},

{

"date": "2025-01-02",

"emotions": {

"worried": 1

}

}

],

"total\_summary": {

"happy": 12,

"peaceful": 7,

"sad": 3,

"worried": 2,

"angry": 1

},

"most\_frequent\_emotion": "happy",

"emotion\_streak": {

"current": {

"user\_emotion": "happy",

"days": 3

},

"longest": {

"user\_emotion": "happy",

"days": 5,

"period": {

"start": "2025-01-10",

"end": "2025-01-14"

}

}

}

},

"message": null,

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_tuv678"

}

#### 사용 통계 조회

GET /api/stats/usage?month=2025-01

**요청 헤더:**

Authorization: Bearer {access\_token}

**성공 응답 (200 OK):**

{

"success": true,

"data": {

"period": "2025-01",

"diary\_stats": {

"total\_created": 25,

"total\_words": 12500,

"average\_words\_per\_diary": 500,

"with\_images": 18,

"with\_ai\_text": 20

},

"ai\_usage": {

"text\_generations": 35,

"keyword\_suggestions": 15,

"total\_tokens\_used": 5250,

"average\_tokens\_per\_request": 150

},

"image\_stats": {

"total\_uploaded": 45,

"total\_size\_mb": 125.5,

"synthesized\_images": 12

},

"activity\_heatmap": {

"most\_active\_hour": 21,

"most\_active\_day": "Sunday",

"activity\_by\_hour": {

"0": 0,

"1": 0,

"2": 0,

"3": 0,

"4": 0,

"5": 0,

"6": 1,

"7": 2,

"8": 3,

"9": 2,

"10": 1,

"11": 1,

"12": 2,

"13": 1,

"14": 1,

"15": 2,

"16": 1,

"17": 2,

"18": 3,

"19": 4,

"20": 5,

"21": 8,

"22": 6,

"23": 3

}

}

},

"message": null,

"timestamp": "2025-01-08T12:00:00Z",

"request\_id": "req\_wxy901"

}

## 6. 데이터 모델

### 6.1 감정 타입 (EmotionType)

enum EmotionType {

HAPPY = 'happy', // 😊 행복

SAD = 'sad', // 😢 슬픔

ANGRY = 'angry', // 😡 화남

PEACEFUL = 'peaceful', // 😌 평온

WORRIED = 'worried', // 🫨 불안

}

**UI/UX 디자인 가이드**

감정 타입의 시각적 표현 및 디자인 가이드라인은 별도 문서에서 관리됩니다. 자세한 내용은 [DESIGN\_SYSTEM.md](http://./DESIGN_SYSTEM.md#3-%EA%B0%90%EC%A0%95-%ED%83%80%EC%9E%85-uiux-%EC%8B%9C%EC%8A%A4%ED%85%9C)를 참조하세요.

**API 응답 예시:**

* JSON 응답: "user\_emotion": "worried"
* 프론트엔드에서 worried 값을 디자인 시스템의 매핑 테이블에 따라 🫨 불안으로 표시

### 6.2 다이어리 모델 (Diary)

interface Diary {

diary\_id: string; // UUID

user\_id: string; // UUID

title: string;

content: string;

user\_emotion?: EmotionType; // 사용자가 선택한 감정

ai\_emotion?: EmotionType; // AI가 분석한 감정

ai\_emotion\_confidence?: number; // AI 감정 분석 신뢰도 (0.0-1.0)

images: ImageData[];

ai\_generated\_text?: string;

is\_public: boolean;

keywords: string[]; // 키워드

created\_at: string; // ISO 8601 형식

updated\_at: string; // ISO 8601 형식

}

interface EmotionAnalysis {

user\_emotion?: EmotionType; // 사용자가 수정한 감정 (선택사항)

ai\_emotion?: EmotionType; // AI가 글과 사진을 분석하여 추출한 감정

confidence: number; // AI 분석 신뢰도 (0.0-1.0)

is\_match?: boolean; // 사용자 수정과 AI 분석 일치 여부

}

### 6.3 AI 스타일 옵션

interface AIStyleOptions {

tone: 'romantic' | 'healing' | 'calm' | 'humorous';

length: 'short' | 'medium' | 'long';

type: 'poem' | 'prose' | 'quote';

}

### 6.4 이미지 합성 옵션

interface SynthesisOptions {

font: string;

font\_size: number;

color: string;

position: 'top' | 'center' | 'bottom';

opacity: number;

shadow: boolean;

}

## 7. 웹훅 (Webhooks) - 향후 구현

### 7.1 웹훅 이벤트 타입

| **이벤트** | **설명** | **페이로드** |
| --- | --- | --- |
| diary.created | 다이어리 생성 시 | diary 객체 |
| diary.updated | 다이어리 수정 시 | diary 객체 |
| diary.deleted | 다이어리 삭제 시 | diary\_id |
| ai.regenerated | AI 글귀 재생성 완료 | generation 객체 |
| emotion.report\_ready | 월간 리포트 생성 완료 | report 객체 |

### 7.2 웹훅 페이로드 구조

{

"event": "diary.created",

"timestamp": "2025-01-08T12:00:00Z",

"data": {

// 이벤트별 데이터

}

}

## 8. SDK 및 클라이언트 라이브러리

### 8.1 지원 언어

* JavaScript/TypeScript (공식)
* Python (커뮤니티)
* Swift (iOS - 계획)
* Kotlin (Android - 계획)

### 8.2 JavaScript SDK 예시

import { SaeGimClient } from '@saegim/sdk';

const client = new SaeGimClient({

apiKey: 'your-api-key',

baseUrl: 'https://api.saegim.app',

});

// 소셜 로그인 URL 가져오기

const { data: authUrl } = await client.auth.getSocialLoginUrl('google');

window.location.href = authUrl;

// 소셜 로그인 콜백 처리

const { data: tokens } = await client.auth.handleSocialCallback('google', {

code: 'oauth\_code',

state: 'csrf\_state',

});

// 다이어리 작성

const { data: diary } = await client.diary.create({

title: '오늘의 일기',

content: '...',

// 감정은 AI가 글과 사진을 분석하여 자동 추출

});

// AI 글귀 생성

const { data: aiText } = await client.ai.generate({

style: {

tone: 'healing',

length: 'medium',

type: 'poem',

},

});

## 9. 테스트 환경

### 9.1 테스트 서버

* **Base URL**: https://api-test.saegim.app
* **특징**: 프로덕션과 동일한 환경, 테스트 데이터 사용

### 9.2 테스트 계정

{

"email": "test@saegim.app",

"password": "TestPass123!"

}

### 9.3 Rate Limit (테스트 환경)

**기본 정책 (프로덕션과 동일):**

* 일반 API: 1,000 requests/minute
* AI API: 30 requests/minute
* 이미지 업로드: 60 requests/hour

**테스트 전용 확장 정책 (개발/디버깅 시에만 활성화):**

* 개발 환경에서 테스트 목적으로만 AI API 100 requests/minute 허용
* 환경변수 TEST\_RATE\_LIMIT\_EXTENDED=true 설정 시에만 적용
* 프로덕션 환경에서는 절대 활성화 불가

## 10. 변경 로그

### Version 1.0 (2025-08-08)

* 초기 API 명세 작성
* 인증, 다이어리, AI, 이미지, 통계 API 정의

## Appendix

### A. HTTP 헤더 참조

| **헤더** | **설명** | **예시** |
| --- | --- | --- |
| Authorization | JWT 토큰 | Bearer eyJhbGci... |
| Content-Type | 요청 본문 타입 | application/json |
| Accept | 응답 타입 | application/json |
| X-Request-ID | 요청 추적 ID | 550e8400-e29b... |
| Accept-Language | 선호 언어 | ko-KR |

### B. 타임존

* 모든 시간은 UTC 기준
* 클라이언트에서 로컬 시간으로 변환

### C. 문의 및 지원

* **이메일**: [api-support@saegim.app](mailto:api-support@saegim.app)
* **개발자 포털**: <https://developers.saegim.app>
* **API 상태**: <https://status.saegim.app>

*본 문서는 '새김' 프로젝트의 API 명세서로, 지속적으로 업데이트됩니다.*