

뉴스 추천 서비스 API 인터페이스 설계서

1. 개요

1.1 서비스 개요

- **서비스명:** 뉴스 추천 서비스 (News Recommendation Service)
- **기본 URL:** `/api/news/feed`
- **설명:** 개인화된 뉴스 피드 추천 및 관리 서비스

1.2 주요 기능

- 개인화 뉴스 피드 추천 (첫 페이지)
- 전체 뉴스 최신순 조회 (나머지 페이지)
- 사용자 선호도 기반 카테고리별 뉴스 추천
- 관리자용 특정 사용자 피드 조회

1.3 추천 알고리즘

- **첫 페이지 (page=0):** 개인화 추천
 - 사용자 벡터 기반 상위 3개 카테고리 선정
 - 각 카테고리별 최신 뉴스 7/5/3개씩 할당
 - 총 15개 뉴스 추천
- **나머지 페이지:** 전체 뉴스 최신순
 - `published_at` 기준 내림차순 정렬
 - 카테고리별 랜덤 섞기로 다양성 확보

2. 공통 응답 형식

2.1 ApiResponse 구조

```
{
  "success": boolean,
  "data": T,
  "message": string
}
```

2.2 인증

- **방식:** JWT Bearer Token
- **헤더:** `Authorization: Bearer {token}`
- **토큰 추출:** `@AuthenticationPrincipal String userIdStr`

3. 데이터 모델

3.1 FeedItemDto

```
{
  "newsId": "Long",
  "title": "String",
  "press": "String",
  "link": "String",
  "trusted": "Boolean",
  "publishedAt": "LocalDateTime",
  "createdAt": "LocalDateTime",
  "reporter": "String",
  "viewCount": "Integer",
  "categoryName": "RecommendationCategory",
  "dedupState": "DedupState",
  "imageUrl": "String",
  "oidAid": "String",
  "updatedAt": "LocalDateTime"
}
```

3.2 RecommendationCategory (Enum)

```
POLITICS,      // 정치
ECONOMY,       // 경제
SOCIETY,       // 사회
LIFE,          // 생활
INTERNATIONAL, // 세계
IT_SCIENCE,    // IT/과학
VEHICLE,       // 자동차/교통
TRAVEL_FOOD,   // 여행/음식
ART            // 예술
```

3.3 DedupState (Enum)

```
REPRESENTATIVE, // 대표 기사
RELATED,        // 연관 뉴스
KEPT            // 유지
```

4. API 엔드포인트 상세

4.1 개인화 피드 조회

4.1.1 인증된 사용자 피드 조회

- URL: **GET** /api/news/feed
- 인증: 필수 (JWT Token)
- 설명: 현재 로그인한 사용자의 개인화된 뉴스 피드 조회

요청 파라미터:

파라미터	타입	필수	기본값	설명
page	int	X	0	페이지 번호 (0부터 시작)
size	int	X	21	페이지당 뉴스 개수

요청 예시:

```
GET /api/news/feed?page=0&size=21
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

응답 예시:

```
{
  "success": true,
  "data": [
    {
      "newsId": 12345,
      "title": "AI 기술 발전으로 인한 일자리 변화",
      "press": "테크뉴스",
      "link": "https://example.com/news/12345",
      "trusted": true,
      "publishedAt": "2024-01-15T10:30:00",
      "createdAt": "2024-01-15T10:35:00",
      "reporter": "김기자",
      "viewCount": 1500,
      "categoryName": "IT_SCIENCE",
      "dedupState": "REPRESENTATIVE",
      "imageUrl": "https://example.com/images/12345.jpg",
      "oidAid": "12345_67890",
      "updatedAt": "2024-01-15T10:35:00"
    }
  ],
  "message": null
}
```

응답 코드:

- 200 OK: 성공적으로 피드 조회
- 401 Unauthorized: 인증 실패
- 400 Bad Request: 잘못된 요청 파라미터
- 500 Internal Server Error: 서버 내부 오류

4.1.2 관리자용 특정 사용자 피드 조회

- URL: GET /api/news/feed/{id}
- 인증: 필수 (JWT Token)
- 설명: 특정 사용자 ID의 피드 조회 (개발/테스트/관리 목적)

경로 파라미터:

파라미터	타입	필수	설명
id	Long	O	조회할 사용자 ID

요청 파라미터:

파라미터	타입	필수	기본값	설명
page	int	X	0	페이지 번호 (0부터 시작)
size	int	X	21	페이지당 뉴스 개수

요청 예시:

```
GET /api/news/feed/123?page=0&size=21
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

응답 예시:

```
{
  "success": true,
  "data": [
    {
      "newsId": 12346,
      "title": "경제 정책 변화에 따른 시장 반응",
      "press": "경제일보",
      "link": "https://example.com/news/12346",
      "trusted": true,
      "publishedAt": "2024-01-15T09:15:00",
      "createdAt": "2024-01-15T09:20:00",
      "reporter": "이기자",
      "viewCount": 2300,
      "categoryName": "ECONOMY",
      "dedupState": "REPRESENTATIVE",
      "imageUrl": "https://example.com/images/12346.jpg",
      "oidAid": "12346_67891",
      "updatedAt": "2024-01-15T09:20:00"
    }
  ],
  "message": null
}
```

5. 비즈니스 로직 상세

5.1 개인화 추천 로직 (첫 페이지)

1. 사용자 벡터 업데이트

- `VectorBatchService.upsert(userId)` 호출
- 필요시 사용자 선호도 벡터 재계산

2. 상위 카테고리 선정

- `UserPrefVectorRepository.findTopByUserIdOrderByScoreDesc()`
- 점수 기준 상위 3개 카테고리 조회

3. 카테고리별 뉴스 할당

- 1순위 카테고리: 7개 뉴스
- 2순위 카테고리: 5개 뉴스
- 3순위 카테고리: 3개 뉴스
- 총 15개 뉴스 추천

4. 뉴스 메타 정보 조회

- `RecommendationNewsRepository.findByIdIn()`
- 일괄 조회로 성능 최적화

5.2 전체 뉴스 피드 로직 (나머지 페이지)

1. 최신순 조회

- `published_at` 기준 내림차순 정렬
- 페이징 처리

2. 카테고리 다양성 확보

- 카테고리별로 그룹화
- 랜덤 섞기로 카테고리 순서 다양화
- 번갈아가며 배치

6. 에러 처리

6.1 공통 에러 응답

```
{
  "success": false,
  "data": null,
  "message": "에러 메시지"
}
```

6.2 주요 에러 케이스

- 인증 실패: JWT 토큰이 유효하지 않거나 만료된 경우
- 사용자 없음: 존재하지 않는 사용자 ID로 조회하는 경우
- 잘못된 파라미터: page, size 값이 범위를 벗어나는 경우
- 서버 오류: 데이터베이스 연결 실패 등 내부 오류

7. 성능 고려사항

7.1 최적화 전략

- **인덱스 활용:** `published_at`, `category_name` 인덱스
- **일괄 조회:** `findByIdIn()` 으로 N+1 문제 해결
- **페이징:** 대용량 데이터 처리 시 메모리 효율성
- **캐싱:** 사용자 벡터 캐싱으로 계산 비용 절약

7.2 예상 응답 시간

- **개인화 피드:** 200-500ms
- **전체 뉴스 피드:** 100-300ms
- **네트워크 지연:** 50-100ms

8. 보안 고려사항

8.1 인증 및 권한

- JWT 토큰 기반 인증 필수
- 사용자별 데이터 접근 제어
- 관리자 권한 검증 (특정 사용자 조회 시)

8.2 데이터 보호

- 사용자 개인정보 노출 방지
- API 응답에서 민감한 정보 제거
- 로그에서 개인정보 마스킹

9. 모니터링 및 로깅

9.1 주요 메트릭

- API 응답 시간
- 추천 정확도 (클릭률 등)
- 사용자별 피드 조회 빈도
- 카테고리별 추천 분포

9.2 로깅 레벨

- **INFO:** 정상적인 피드 조회
- **WARN:** 추천 데이터 부족, 벡터 계산 실패
- **ERROR:** 데이터베이스 오류, 서비스 장애