

## 데이터 수집 및 저장 데이터 베이스 설계서

### □ 개요

- 산출물 단계 : 데이터 수집 및 저장
- 평가 산출물 : 데이터베이스 설계서
- 제출 일자 : 2025. 08. 22
- 깃허브 경로 : [SKN14-Final-1Team](#)

개요	<ul style="list-style-type: none"><li>• 소개</li><li>• 시스템 개요</li></ul>
구조	<ul style="list-style-type: none"><li>• 시스템 아키텍처</li><li>• 요구사항 매트릭스</li></ul>
설계	<ul style="list-style-type: none"><li>• 데이터 설계</li><li>• 설계 근거</li><li>• ERD</li></ul>

<p>소개</p>	<ul style="list-style-type: none"> <li>- <b>RDB목적</b> <ul style="list-style-type: none"> <li>- 본 데이터베이스는 사용자 가입/승인 흐름, 문서/모드별 챗 세션과 메시지, 즐겨찾기(카드), 그리고 사용자별 <b>API</b> 키 관리까지 포함한 <b>AI</b> 기반 문서 검색 시스템의 핵심 데이터를 저장·관리하기 위해 설계되었습니다.</li> </ul> </li> <li>- <b>RDB범위</b> <ul style="list-style-type: none"> <li>- 사용자 계정과 상태 관리, 승인 로그 이력 관리, 세션·메시지 보관, 즐겨찾기 카드/카드&amp;메시지 매핑, 선택적 <b>API</b> 키 발급 및 사용 권한 범위 관리 등을 포함합니다.</li> </ul> </li> <li>- <b>Vector DB목적</b> <ul style="list-style-type: none"> <li>- Google API 및 사내 내부 문서에 대한 <b>RAG</b>(검색 증강 생성) 기술을 적용하여, 사용자의 질문에 가장 정확하고 연관성 높은 정보를 제공하기 위해 설계되었습니다.</li> </ul> </li> <li>- <b>Vector DB 범위</b> <ul style="list-style-type: none"> <li>- GoogleAPI의 범위는 OAuth 2.0, YouTube Data API, People API, Maps Platform, Drive API, Sheets API, Gmail API, Calendar API . Firestore REST API ,Firebase API, Firebase Authentication, BigQuery API 총 11개로 구성되어 있습니다.</li> <li>- 사내 내부 문서는 <b>OpenAI</b> API를 활용하여 자체 생성한 합성 데이터셋(<b>Synthetic Dataset</b>)을 사용합니다. 이는 사용자의 직급 및 부서에 따른 문서 접근 권한 제어 기능을 테스트하기 위한 목적으로 구성되어 있습니다.</li> </ul> </li> </ul>
<p>시스템 개요</p>	<ul style="list-style-type: none"> <li>- <b>시스템 역할</b> <ul style="list-style-type: none"> <li>- 가입 사용자의 승인 상태(pending/approved/rejected) 관리</li> <li>- 채팅 모드별 ChatSession 생성 및 ChatMessage 기록</li> <li>- 메시지 선택을 통한 카드(Card) 생성 및 즐겨찾기 관리</li> <li>- 사용자별 <b>API Key</b> 생성·보관</li> <li>- <b>API</b> 전문 어시스턴트: <b>OpenAI</b>의 <b>GPT-4o</b> 모델을 기반으로, 구글 <b>API</b> 문서에 대한 질의응답, 코드 예시 제공, 이미지/음성을 통한 질문 기능을 지원합니다.</li> </ul> </li> </ul>

	<p>- 사내 문서 전문 챗봇: <b>Qwen3-8B</b> 소형 언어 모델(sLLM)을 기반으로, 사내 문서에 대한 질의응답을 수행하며, 사용자의 선택에 따라 '공손한 말투'와 '친구 말투'로 소통합니다.</p> <p><b>- 주요 기능</b></p> <ul style="list-style-type: none"> <li>- 사용자 인증 및 상태 변경 이력 보관(ApprovalLog)</li> <li>- 세션 단위의 대화 흐름 저장(ChatSession, ChatMessage)</li> <li>- 즐겨찾기 카드 관리(Card, CardMessage)</li> <li>- API 키 관리(ApiKey)</li> </ul>																
시스템 아키텍처	<p><b>- 데이터베이스 구조</b></p> <p>- 관계형 데이터베이스(RDBMS)로 설계되며, 핵심 테이블은 다음과 같습니다.</p> <ul style="list-style-type: none"> <li>- <b>user</b> : 사용자 기본 정보/조직 정보/상태</li> <li>- <b>approval_log</b> : 가입 승인/거절 이력</li> <li>- <b>chat_session</b> : 세션 메타(제목/모드/소유자)</li> <li>- <b>chat_message</b> : 세션 내 메시지(역할/본문/시각)</li> <li>- <b>card</b> : 즐겨찾기 카드(사용자 소유, 선택적으로 세션 연결)</li> <li>- <b>card_message</b> : 카드-메시지 매핑 및 순서</li> <li>- <b>api_key</b> : 사용자별 API Key와 권한 범위</li> </ul> <p><b>- 적용한 데이터베이스</b></p> <p>-사내 내부 문서의 경우 <b>chromaDB</b>에 벡터화 하여 저장함</p>																
요구사항 매트릭스	<p>요구사항 및 관련 테이블</p> <table> <tr> <th>요구사항</th><th>관련 테이블</th></tr> <tr> <td>사용자 계정 및 상태 관리</td><td>user</td></tr> <tr> <td>가입 승인/반려 이력 저장</td><td>approval_log</td></tr> <tr> <td>모드별 챗 세션 관리</td><td>chat_session</td></tr> <tr> <td>세션 내 메시지 저장</td><td>chat_message</td></tr> <tr> <td>즐거찾기 카드 생성/조회</td><td>card</td></tr> <tr> <td>카드에 포함된 메시지와 순서</td><td>card_message</td></tr> <tr> <td>사용자별 API 키 관리</td><td>api_key</td></tr> </table>	요구사항	관련 테이블	사용자 계정 및 상태 관리	user	가입 승인/반려 이력 저장	approval_log	모드별 챗 세션 관리	chat_session	세션 내 메시지 저장	chat_message	즐거찾기 카드 생성/조회	card	카드에 포함된 메시지와 순서	card_message	사용자별 API 키 관리	api_key
요구사항	관련 테이블																
사용자 계정 및 상태 관리	user																
가입 승인/반려 이력 저장	approval_log																
모드별 챗 세션 관리	chat_session																
세션 내 메시지 저장	chat_message																
즐거찾기 카드 생성/조회	card																
카드에 포함된 메시지와 순서	card_message																
사용자별 API 키 관리	api_key																

데이터 설계	- 테이블 설명		
	1) user : 사용자 계정 정보		
	컬럼명	데이터 타입	설명
	id	VARCHAR(50)	사용자 (PK), 로그인 ID
	email	VARCHAR(255)	이메일
	password	VARCHAR(255)	해시 비밀번호
	name	VARCHAR(100)	이름
	phone	VARCHAR(30)	전화번호
	gender	VARCHAR(20)	성별
	birthday	DATE	생년월일
	rank	VARCHAR(50)	직급
	department	VARCHAR(100)	부서
	status	ENUM('pending','approved','rejected')	가입 상태
	last_login	DATETIME	최근 로그인 시각
	created_at	DATETIME	생성 시각
	updated_at	DATETIME	수정 시각
	2) approval_log : 계정 승인/거절 기록		
	컬럼명	데이터 타입	설명
	id	INT	로그 PK
	user_id	INT	대상 사용자
	action	VARCHAR(100) ENUM('pending','approved','rejected')	처리 종류
	reason	VARCHAR(500)	사유
	created_at	DATETIME	기록 시각

### 3) chat\_session : 채팅 세션

컬럼명	데이터 타입	설명	제약
id	INT	세션 PK	PK, AUTO INCREMENT
user_id	INT	세션 소유자	FK → user.id, NOT NULL, ON DELETE CASCADE
title	VARCHAR(200)	세션 제목	NOT NULL
mode	VARCHAR(100) ENUM('google_api', 'internal')	세션 모드	NOT NULL
created_at	DATETIME	생성 시각	NOT NULL

### 4) chat\_message : 채팅 메시지

컬럼명	데이터 타입	설명	제약
id	INT	메시지 PK	PK, AUTO INCREMENT
session_id	INT	소속 세션	FK → chat_session.id, NOT NULL, ON DELETE CASCADE
role	VARCHAR ENUM('user', 'assistant')	발화 주체	NOT NULL
content	TEXT	본문	NOT NULL
created_at	DATETIME	생성 시각	NOT NULL

### 5) card : 즐겨찾기 카드

컬럼명	데이터 타입	설명	제약
id	INT	카드 PK	PK, AUTO INCREMENT
user_id	INT	카드 소유자	FK → user.id, NOT NULL, ON DELETE CASCADE
session_id	INT	관련 세션	FK → chat_session.id, NULL, ON DELETE CASCADE
title	VARCHAR(200)	카드 제목	NOT NULL

created_at	DATETIME	생성 시각	NOT NULL
------------	----------	-------	----------

#### 6) card\_message : 카드-메시지 매핑

컬럼명	데이터 타입	설명	제약
id	INT	매핑 PK	PK, AUTO INCREMENT
card_id	INT	대상 카드	FK → card.id, NOT NULL, ON DELETE CASCADE
message_id	INT	대상 메시지	FK → chat_message.id, NOT NULL, ON DELETE CASCADE
position	INT	카드 내 순서	NOT NULL

#### 7) api\_key : API 키 관리

컬럼명	데이터 타입	설명	제약
id	INT	키 PK	PK, AUTO INCREMENT
user_id	INT	키 소유자	FK → user.id, NOT NULL, ON DELETE CASCADE
name	VARCHAR(100)	키 별칭	NOT NULL
secret_key	VARCHAR(255)	시크릿 키	NOT NULL
created_at	DATETIME	생성 시각	NOT NULL

#### 8) Post : 게시물

컬럼명	데이터 타입	설명	제약
id	integer	게시물의 고유 식별자	Primary Key
title	Varchar(200)	게시물의 제목	Not Null
content	Text	게시물의 내용	Not Null
Author id	integer	작성자의 ID	Foreign Key, Not Null

created_at	Datetime	생성 시각	Not Null
updated_at	Datetime	수정 시각	Not Null
views	integer	조회 수	Not Null, Default: 0
likes	Integer	좋아요 수	Not Null, Default: 0

#### 8-1) Post Likers : 게시물 좋아요

컬럼명	데이터 타입	설명	제약
id	Integer	고유 식별자	Primary Key
Post id	Integer	게시물의 ID	Foreign Key, Not Null
User id	Integer	사용자의 ID	Foreign Key, Not Null

#### 9) Comment : 댓글

컬럼명	데이터 타입	설명	제약
id	Integer	댓글의 고유 식별자	Primary Key
Post id	Integer	게시물의 ID	Foreign Key, Not Null
Author id	Integer	작성자의 ID	Foreign Key, Not Null
content	Text	댓글 내용	Not Null
Created at	Datetime	생성 시각	Not Null

	likes	Integer	좋아요 수	Not Null, [Default: 0]
	9-2) Comment Likers : 댓글 좋아요			
	컬럼명	데이터 타입	설명	제약
	id	Integer	관계의 고유 식별자	Primary Key
	Comment id	Integer	'좋아요'를 받은 댓글 ID	Foreign Key (Comment)
설계 근거	User id	Integer	'좋아요'를 누른 사용자 ID	Foreign Key (User)
	<ul style="list-style-type: none"> <li>- 데이터 무결성 보장 <ul style="list-style-type: none"> <li>- 외래 키(FK)를 활용하여 테이블 간 관계를 명확히 정의함.</li> <li>- 삭제 전파는 기본적으로 <b>CASCADE</b>로 설정하여 고아 레코드 방지</li> </ul> </li> <li>- 보안 <ul style="list-style-type: none"> <li>- 사용자 이메일 등 민감한 데이터는 암호화 또는 추가 보안 조치를 적용할 수 있음.</li> </ul> </li> </ul>			



# ERD

