

CINECODE

Movie Recommendation Algorithm Project

Team 3

Minjoo Kim

Yeonji Choi

INDEX

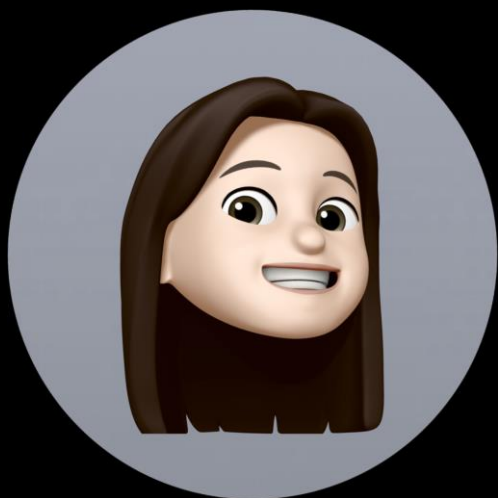
01 팀원 소개 및 역할 분담

02 기획

03 구현 기술

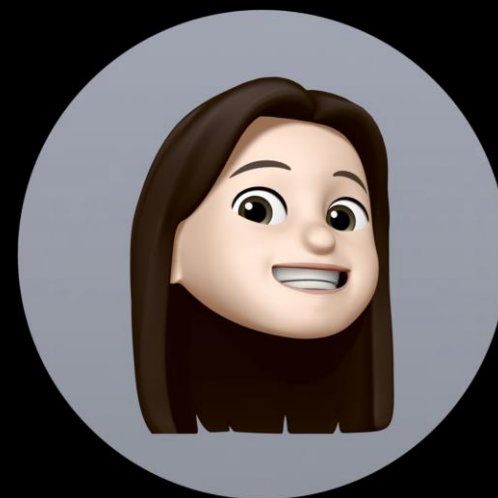
04 시연

05 개선 사항 및 QnA



팀장 김민주

(DB) 초기 영화, 장르 데이터 로드
(FE&BE) Navbar, 영화 상세 정보 및 상태 관리,
리뷰 CRUD



팀원 최연지

(FE&BE) 로그인, 회원가입, 회원 정보 수정,
검색 및 감정 인식 기반 영화 추천,
Navbar 시계 설정 기능

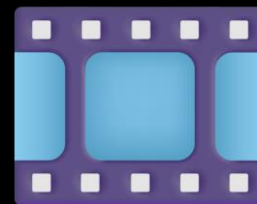
기획 의도



음성 인식을 통한
감정 분석 영화 추천 서비스



영화 리뷰 작성 및 공유



개인화된 영화 목록 생성

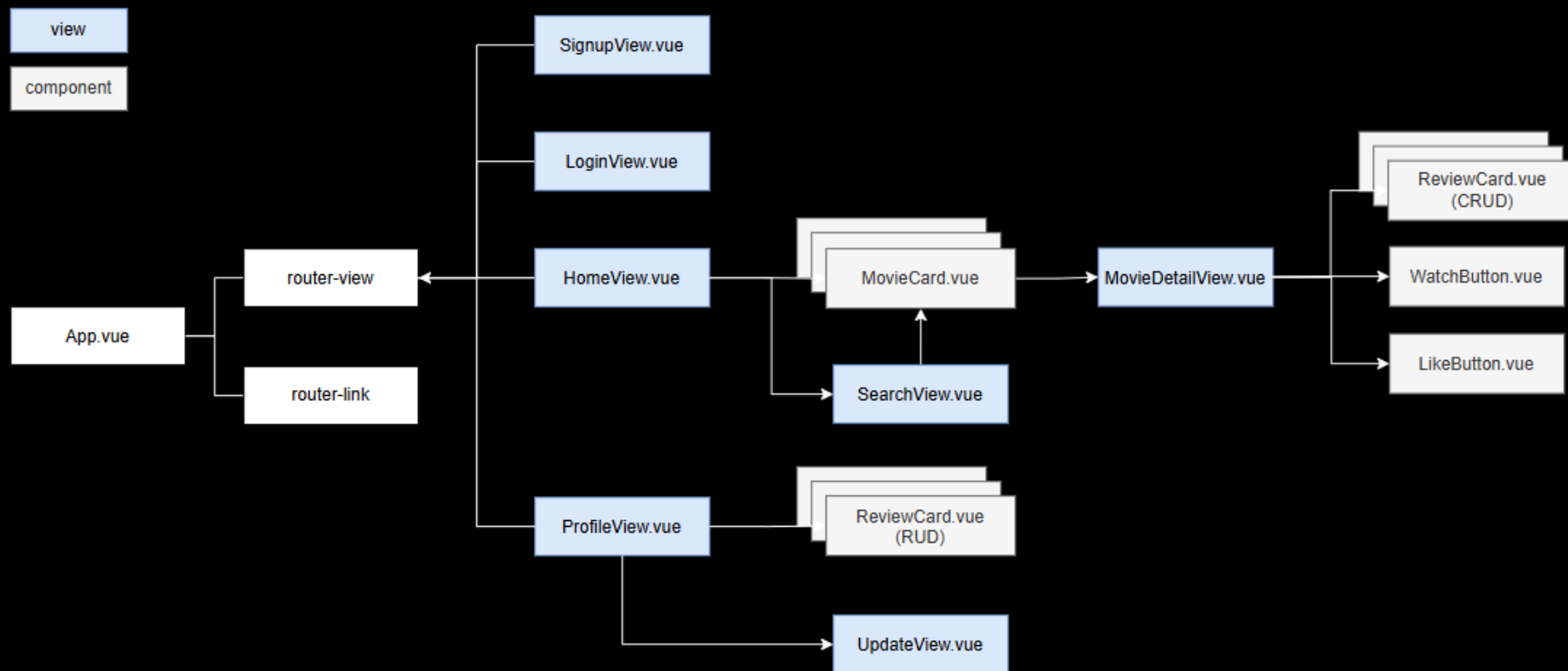
개발 일정

[illegible]

ERD



컴포넌트 구조도



API (1)

TMDB API Key

```
back > .env
1  VITE_TMDB_API_KEY='[REDACTED]'
2

back > init1.py > ...
1  import requests
2  import json
3  import os
4  import environ
5
6  # 환경변수 설정
7  env = environ.Env(DEBUG=(bool, True))
8  BASE_DIR = os.path.dirname(os.path.abspath(__file__))
9  environ.Env.read_env(env_file=os.path.join(BASE_DIR, '.env'))
10 TMDB_API_KEY = env('TMDB_ACCESS_TOKEN')
11
12 # TMDB API 헤더 설정
13 headers = {
14     'Authorization': f'Bearer {TMDB_API_KEY}',
15     'accept': 'application/json'
16 }
17
18 # 영화 데이터 가져오기
19 def get_movie_datas():
20     total_data = []
```


API (2)

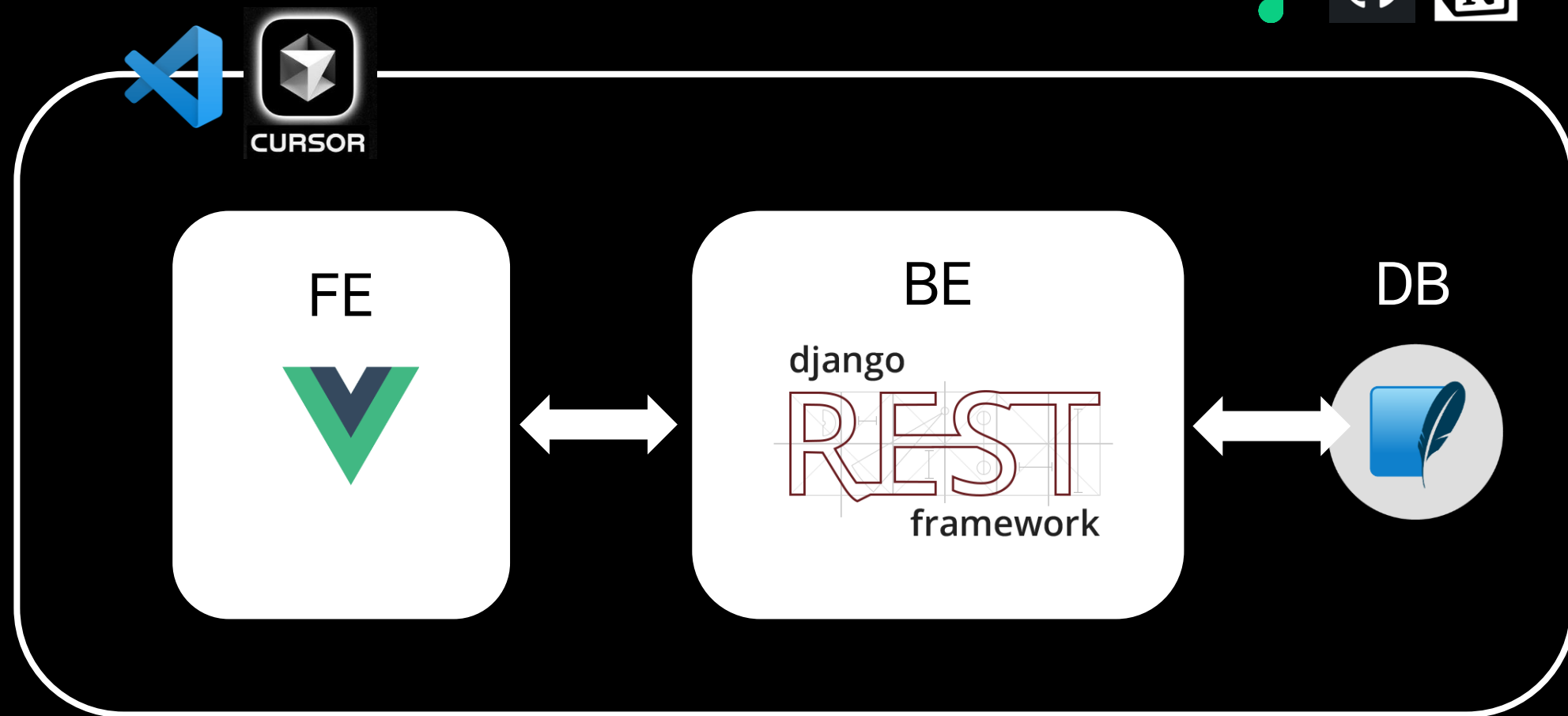
Google Cloud Speech-to-Text, Natural Language API

```
back > cine_code > {} cindcode-93e04ba2fbc3.json > ...
1  {
2    "type": "service_account",
3    "project_id": "cindcode",
4    "private_key_id": [REDACTED],
5    "private_key": [REDACTED],
6    "client_email": [REDACTED],
7    "client_id": [REDACTED],
8    "auth_uri": "https://accounts.google.com/o/oauth2/auth",
9    "token_uri": "https://oauth2.googleapis.com/token",
10   "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
11   "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata/x509/cinecode%40cindcode.iam.gserviceaccount.com"
}

back > cine_code > settings.py > ...
220
221  GOOGLE_APPLICATION_CREDENTIALS = os.path.join(os.path.dirname(os.path.abspath(__file__)), 'cindcode-[REDACTED].json')
222
223  # Google Cloud 클라이언트 설정
224  GOOGLE_CLOUD_CREDENTIALS = service_account.Credentials.from_service_account_file(
225      GOOGLE_APPLICATION_CREDENTIALS
226  )

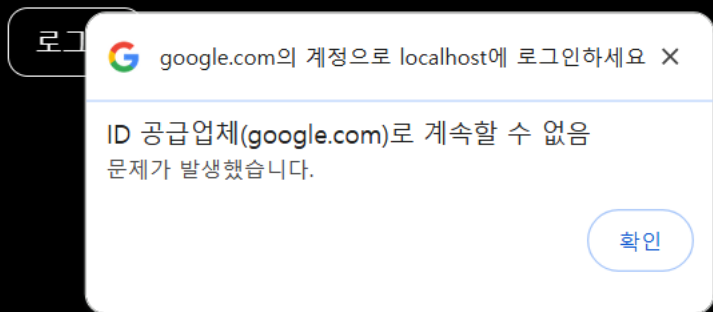
back > movies > views.py > transcribe_file
142  # 음성 파일을 텍스트로 변환하는 함수
143  def transcribe_file(audio_file: str) -> str:
144      os.environ['GOOGLE_APPLICATION_CREDENTIALS'] = settings.GOOGLE_APPLICATION_CREDENTIALS
145
146      client = speech.SpeechClient()
147
148      with open(audio_file, "rb") as f:
149          audio_content = f.read()
```

TOOL



Live Demo

개선 사항



소셜 로그인 구현

```
▶ config: {transitional: {...}, adapter: Array(3), transformRequest: Array(2), transformResponse: Array(2), timeout: 0, xsrfCookieName: 'XSRF-TOKEN', xsrfHeaderName: 'X-XSRF-TOKEN', maxRedirects: 5, maxContentLength: -1, maxBodyLength: -1, validateStatus: function(status) { return status < 300; }, headers: {}},
▼ data:
  ▶ username: ['A user with that username already exists.'],
  ▶ [[Prototype]]: Object
▶ headers: AxiosHeaders {content-length: '58', content-type: 'application/json'}
```

회원 가입 시
ID 중복 검사

QnA

..... The End

감사합니다