

# 개요

---

팀프로젝트에서 내가 개발 했던것에 대한 것을 작성했습니다. 이번 프로젝트에서는 백엔드 부분을 맡았지만 전체적인 디자인과 기본적인 프론트로대도 개발했습니다.

## 자료조사

영화API와, 날씨API는 각각 [TMDB](#), [openweathermap](#) 사이트에서 가져와 사용했다. 웹프레임워크로 Flask를 사용하는데 나와, 팀원들에게 도움이 될 수 있도록 Flask관련 책을 찾아 공유했다 <https://wikidocs.net/book/4542>.

## 개발

MVT모델에서 view부분과, Model부분을 모두 개발했고 template부분의 기본 토대를 개발했다

```
import json
import requests
import dotenv
import os

dotenv_file = dotenv.find_dotenv()
dotenv.load_dotenv(dotenv_file)

API_KEY = os.environ.get('API_KEY')
print(API_KEY)
movies = []
genres = [{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}, {"id": 16, "name": "Animation"}, {"id": 35, "name": "Comedy"}, {"id": 80, "name": "Crime"}, {"id": 99, "name": "Documentary"}, {"id": 18, "name": "Drama"}, {"id": 10751, "name": "Family"}, {"id": 14, "name": "Fantasy"}, {"id": 36, "name": "History"}, {"id": 27, "name": "Horror"}, {"id": 10402, "name": "Music"}, {"id": 9648, "name": "Mystery"}, {"id": 10749, "name": "Romance"}, {"id": 878, "name": "Science Fiction"}, {"id": 10770, "name": "TV Movie"}, {"id": 53, "name": "Thriller"}, {"id": 10752, "name": "War"}, {"id": 37, "name": "Western"}]

for i in range(1, 20):
    url = f"https://api.themoviedb.org/3/discover/movie?include_adult=true&include_video=false&language=ko-KR&page={i}&region=KR&sort_by=popularity.desc"

    headers = {
        "accept": "application/json",
        "Authorization": f"Bearer {API_KEY}"
    }

    response = requests.get(url, headers=headers)

    data = response.json()['results']

    for result in data:
```

```

        movie = {
            "id": result['id'],
            "title": result['title'],
            "release_date": result['release_date'],
            "popularity": result['popularity'],
            "vote_count": result['vote_count'],
            "vote_average": result['vote_average'],
            "overview": result['overview'],
            "poster_path": result['poster_path'],
            "genre_ids": result['genre_ids'],
            "adult": result['adult'],
            "poster_path": result['poster_path']
        }
        movies.append(movie)

json_movies = json.dumps(movies, ensure_ascii=False, indent=4)

with open('movies.json', 'w', encoding="UTF-8") as make_file:
    json.dump(movies, make_file, indent="\t")

```

```

# 영화 데이터 추가
@bp.route('/init_db')
def init_db():
    # genre data추가
    save_genre()

    # 'movies.json' 파일에서 데일 읽기
    with open('data/movies.json', 'r', encoding='UTF-8') as file:
        movies_data = json.load(file)

    # Movie 모델에 데이터 추가
    for movie_data in movies_data:
        save_movie_data(movie_data)

    movie = Movie.query.first()
    movie_name = movie.title
    genres = movie.genres
    movie_genres = [genre.name for genre in genres]

    return render_template('test_movie.html', movie_name=movie_name,
                           movie_genres=movie_genres)

```

처음 TMDB에서 영화데이터를 가져온다. 따라서 이후 `/init_db`에 접속하면 해당 영화데이터들이 db에 저장된다.

forms.py에서 회원가입과, 질문에 대한 폼을 작성하고 해당 form을 템플릿에서 사용을 했다.

다음으로 model에서 필요한 테이블들을 만들었고 view에서 사용자의 위치와, 날씨를 받아오고 해당 날씨에대한 사용자가 선호하는 장르의 영화를 3개 추천하는 View부분을 개발했다.