

-영화리뷰 공유사이트를 만들게 된 계기(주제 선정이유)

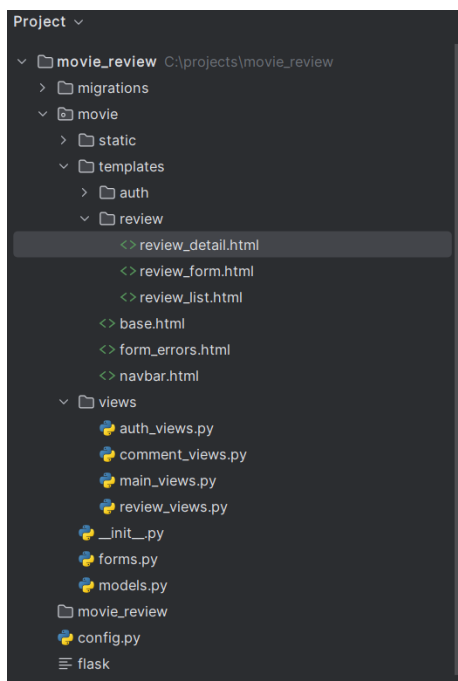
요즘 ott는 2030시대는 10명 중 9명은 시청한다고 기사를 보았다.

기사 내용과 같이 주변 친구들 대부분은 ott를 시청한다. 그리고 ott에는 수많은 영화가 있다. 하지만 이렇게 수많은 영화 중에 어떤영화가 재밌는지 정확히 알고 싶을 때가 있다. 인터넷에는 최신상영작이라면 수많은 바이럴 광고 영상들로 정확히 그 영화의 평점이 어떨지 알 수 없을 때가 있다. 그래서 가까운 사람끼리 영화 리뷰를 공유하는 웹사이트를 만들어보고자 했다.

-개발 결과

우선 Flask 애플리케이션을 app.py 파일로 설정하고, 가상 환경을 C:\wenvs에 생성하여 개발을 시작했다. FLASK_APP 환경 변수를 자동으로 인식하도록 설정하여, 별도의 환경 변수 설정 없이 Flask 서버를 실행할 수 있게 되었다. 또한, myproject.cmd 파일을 통해 가상 환경 활성화 및 서버 실행을 간편하게 만들었다.

그리고 최종적으로 완성된플라스크 프로젝트 구조는 아래와 같다.



모델구성은 다음과 같이 설정하였다.

[리뷰 모델 속성]

- id: 리뷰 데이터의 고유 번호
- title: 영화 제목
- content: 리뷰 내용
- create_date: 리뷰 작성일시
- rating: 평점

[댓글 모델 속성]

- id: 댓글 데이터의 고유 번호
- review_id: 리뷰 데이터의 고유 번호 (어떤 리뷰에 달린 댓글인지 알아야 함)
- content: 댓글 내용
- create_date: 댓글 작성일시

이름	타입	스키마
▼ 테이블 (3)		
> alembic_version		CREATE TABLE alembic
> comment		CREATE TABLE comme
> review		CREATE TABLE review
인덱스 (0)		
뷰 (0)		
트리거 (0)		

속성명	설명
id	질문 데이터의 고유 번호
subject	질문 제목
content	질문 내용
create_date	질문 작성일시

그리고 답변 모델은 다음과 같은 속성이 필요하다.

[답변 모델 속성]

속성명	설명
id	답변 데이터의 고유 번호
question_id	질문 데이터의 고유 번호(어떤 질문에 달린 답변인지 알아야 하므로 질문 데이터의 고유 번호가 필요하다)
content	답변 내용
create_date	답변 작성일시

위는 기본 파이보 게시판의 모델이고 각각의 모델을 참고해 movie, comment 모델을 만들었다. user 모델은 기존의 파이보 게시판과 동일하게 만들었다.

최종으로 완성된 영화리뷰공유게시판 구현내용은 다음과 같다.

Pybo

ksy (로그아웃)



번호	제목	작성일시
1	파일럿	2024-12-22 18:21:41.496807
2	조명가게	2024-12-19 06:00:24.822038
3	소방관	2024-12-19 03:20:32.573214

리뷰하고 싶은 영화 작성하기

Pybo



소방관

낫뻐!

2024-12-19 03:20:32.573214

수정

삭제

1개의 댓글이 있습니다.

나는 별로였다

2024-12-19 03:20:46.401773

위와 같이 입력 수정 삭제 로그인 회원가입이 모두 가능한 영화리뷰공유 게시판을 만들었다. 영화명을 직접 입력하고 다른친구들과 댓글을 통해 서로 영화에 대한 평가를 할 수 있게 되었고 이를 참고해 영화시청 여부를 정할 수 있을 것이다.