

08 PJT

알고리즘을 적용한 서버 구성

INDEX

- 알고리즘을 적용한 서버 구성
 - 목표
 - 준비사항
 - 제공사항
 - 요구사항
 - 제출

목표

프로젝트 목표

- 데이터를 생성, 조회, 수정, 삭제할 수 있는 Web Application 제작
- AJAX 통신과 JSON 구조에 대한 이해
- Database many to one relationship(N:1)에 대한 이해
- Database many to many relationship(M:N)에 대한 이해
- 영화 추천 알고리즘 설계

준비사항

| 개발도구

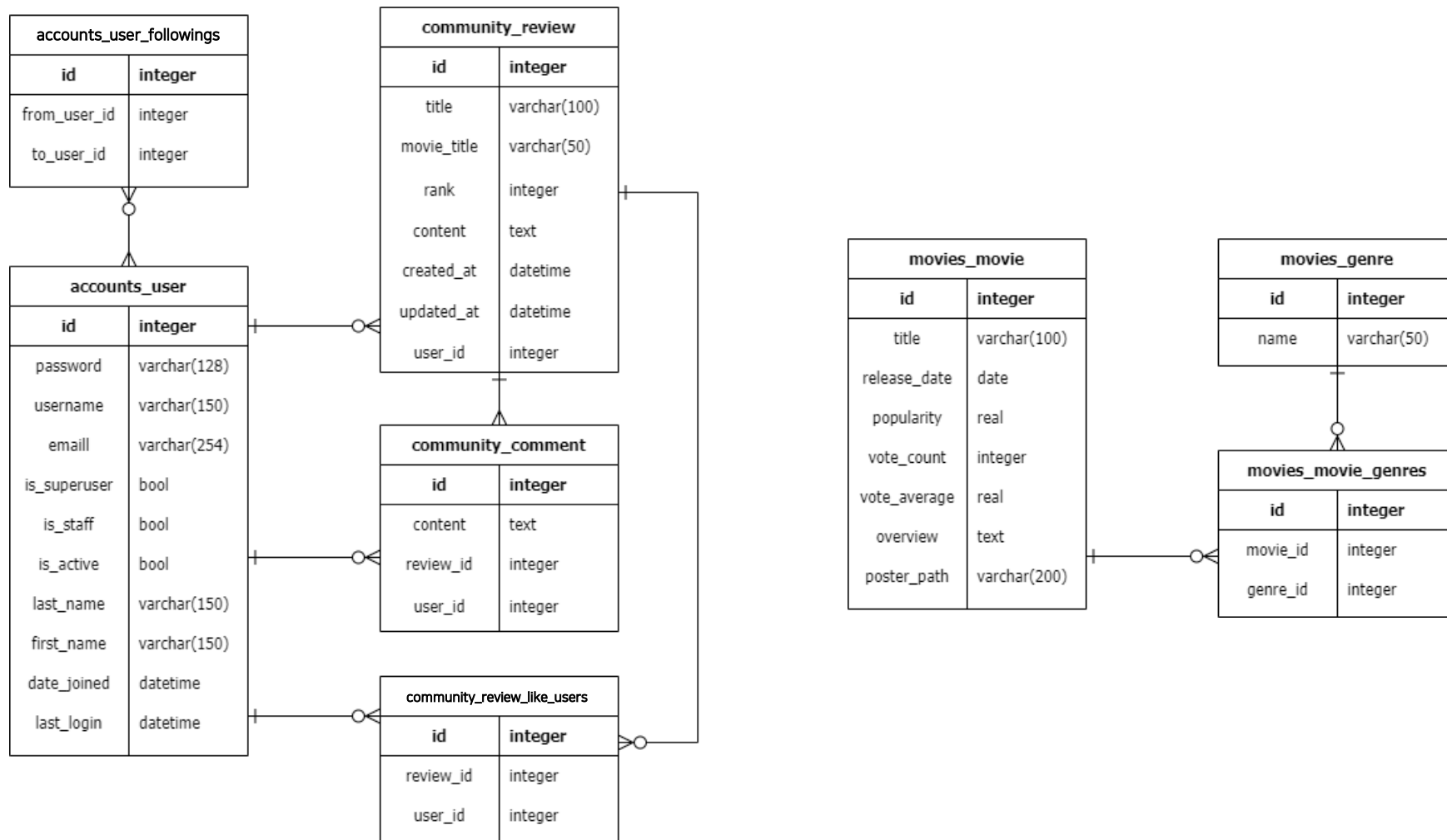
- Visual Studio Code
- Google Chrome
- Django 3.2+

제공사항

프로젝트 정보

- 제공된 스켈레톤 코드를 기반으로 프로젝트를 구성합니다.
- Django 프로젝트의 이름은 **mypjt** 입니다.
- 앱 이름은 **movies, community, accounts** 입니다.

ERD (Entity-Relationship Diagram)



Fixture

- fixture 파일은 제공된 스켈레톤 코드의 movies 앱 내에 위치해 있습니다.
- fixture data의 load는 반드시 모델 정의 및 migrate 이후에 직접 진행합니다.

```
$ python manage.py loaddata movies.json  
Installed 119 object(s) from 1 fixture(s)
```

URL (1/3)

- movies 앱은 다음 URL 요청에 맞는 역할을 가집니다.

URL 패턴	역할
/movies/	전체 영화 목록 페이지 조회
/movies/<movie_pk>/	단일 영화 상세 페이지 조회
/movies/recommended/	추천 영화 페이지 조회

URL (2/3)

- community 앱은 다음 URL 요청에 맞는 역할을 가집니다.

URL 패턴	역할
/community/	전체 리뷰 목록 페이지 조회
/community/create/	새로운 리뷰 생성 페이지 조회 & 단일 리뷰 데이터 저장
/community/<review_pk>/	단일 리뷰 상세 페이지 조회
/community/<review_pk>/ comments/create/	단일 댓글 데이터 저장
/community/<review_pk>/like/	단일 리뷰 좋아요 기능

URL (3/3)

- accounts 앱은 다음 URL 요청에 맞는 역할을 가집니다.

URL 패턴	역할
/accounts/signup/	회원 생성 페이지 조회 & 단일 회원 데이터 생성 (회원가입)
/accounts/login/	로그인 페이지 조회 & 세션 데이터 생성 및 저장 (로그인)
/accounts/logout/	세션 데이터 삭제 (로그아웃)
/accounts/profile/<username>/	사용자 상세 조회 페이지 (프로필 조회)
/accounts/<user_pk>/follow/	사용자 팔로우 기능

View (1/3)

- movies 앱은 다음 역할을 가지는 view 함수를 가집니다.

View 함수명	역할	허용 HTTP Method
index	전체 영화 목록 페이지 조회	GET
detail	단일 영화 상세 페이지 조회	GET
recommended	추천 영화 페이지 조회	GET

View (2/3)

- community 앱은 다음 역할을 가지는 view 함수를 가집니다.

View 함수명	역할	허용 HTTP Method
index	전체 리뷰 목록 페이지 조회	GET
create	새로운 리뷰 생성 페이지 조회 & 단일 리뷰 데이터 저장	GET & POST
detail	단일 리뷰 상세 페이지 조회	GET
create_comment	단일 댓글 데이터 저장	POST
like	단일 리뷰 좋아요 기능	POST

View (3/3)

- accounts 앱은 다음 역할을 가지는 view 함수를 가집니다.

View 함수명	역할	허용 HTTP Method
signup	회원 생성 페이지 조회 & 단일 회원 데이터 생성 (회원가입)	GET & POST
login	로그인 페이지 조회 & 세션 데이터 생성 및 저장 (로그인)	GET & POST
logout	세션 데이터 삭제 (로그아웃)	GET & POST
profile	사용자 상세 조회 페이지 (프로필 조회)	GET
follow	사용자 팔로우 기능	POST

요구사항

| 공통 요구사항

- .gitignore 파일을 추가하여 불필요한 파일 및 폴더는 제출하지 않도록 합니다.
- 명시된 요구사항 이외에는 자유롭게 작성해도 무관합니다.
- 요구사항 완성 후에는 Bootstrap을 이용하여 자유롭게 스타일링 합니다.

| 세부 요구사항

- A. 유저 팔로우 기능
- B. 리뷰 좋아요 기능
- C. Movies 앱 기능
 - 1. 전체 영화 목록 조회
 - 2. 단일 영화 상세 조회
 - 3. 영화 추천 기능
- D. 댓글 좋아요 및 대댓글 기능 (선택 사항)

A. 유저 팔로우 기능

- 프로필 페이지에 팔로워 수와 팔로잉 수를 표시합니다.
- 프로필 페이지에 해당 사용자를 팔로우 할 수 있는 버튼을 표시합니다.
- 인증된 사용자만 다른 사용자를 팔로우 할 수 있으며, 사용자는 자기 자신을 팔로우 할 수 없습니다.
- 팔로우 버튼을 클릭하는 경우, AJAX 통신을 이용하여 서버에서 JSON 데이터를 받아와 상황에 맞게 HTML 화면을 구성합니다.

B. 리뷰 좋아요 기능

- 전체 리뷰 목록 조회 페이지에 **좋아요 버튼**과 **좋아요 개수**를 표시합니다.
- 이미 **좋아요 버튼**을 누른 경우 **좋아요 취소 버튼**을 표시합니다.
- 인증된 사용자만 리뷰에 **좋아요** 할 수 있습니다.
- **좋아요 버튼**을 클릭하는 경우, **AJAX 통신**을 이용하여 서버에서 **JSON 데이터**를 받아와 상황에 맞게 **HTML 화면**을 구성합니다.

C. Movies 앱 기능

1. 전체 영화 목록 조회 (index)

- 사용자의 인증 여부와 관계없이 **전체 영화 목록 조회 페이지**에서 적절한 UI를 활용하여 영화 목록을 제공합니다. (사용자에게 제공할 HTML은 **index.html**)

2. 단일 영화 상세 조회 (detail)

- 사용자의 인증 여부와 관계없이 **단일 영화 상세 조회 페이지**에서 적절한 UI를 활용하여 영화 정보를 제공합니다. (사용자에게 제공할 HTML은 **detail.html**)

C. Movies 앱 기능

3. 영화 추천 기능 (recommended)

- 사용자가 인증되어 있다면, 적절한 알고리즘을 활용하여 10개의 영화를 추천하여 제공합니다. (사용자에게 제공할 HTML은 `recommended.html`)
- 영화를 추천하는 알고리즘은 자유롭게 구상합니다.
- 구현한 알고리즘에 대한 설명은 `README.md`에 상세히 작성합니다.

D. 댓글 좋아요 및 대댓글 기능 (선택 사항)

- 단일 리뷰 상세 조회 페이지에
댓글 좋아요 버튼과 좋아요 개수를 표시합니다.
- 각 댓글에 하위 댓글을 작성할 수 있는
대댓글 기능을 완성합니다.

댓글 목록

2개의 댓글이 있습니다.

• kyle - 저도 재밌게 봤어요! ♥ 4

◦ harry - 2222 ♥ 1

◦ haley - 3333 ♥ 3

◦ alex - 4444 ♥ 0

◦ jun - 5555 ♥ 0

대댓글 작성

• jun - 마지막 탈출 너무 짜릿해요! ♥ 4

◦ kyle - 맞아요 ππ ♥ 2

대댓글 작성

댓글 작성

[\[back\]](#)

25

제출

제출 시 주의사항

- 제출기한은 금일 18시까지입니다. 제출기한을 지켜 주시기 바랍니다.
- 반드시 README.md 파일에 단계별로 구현 과정 중 학습한 내용, 어려웠던 부분, 새로 배운 것들 및 느낀 점을 등을 상세히 기록하여 제출합니다.
 - 단순히 완성된 코드만을 나열하지 않습니다.
- 위에 명시된 요구사항은 최소 조건이며, 추가 개발을 자유롭게 진행할 수 있습니다.
- <https://lab.ssafy.com/>에 프로젝트를 생성하고 제출합니다.
 - 프로젝트 이름은 '프로젝트 번호 + pjt'로 지정합니다. (ex. **01_pjt**)
- 반드시 각 반 담당 교수님을 Maintainer로 설정해야 합니다.