

관계형 데이터베이스 설계

1. 목표

- 데이터를 생성, 조회, 수정, 삭제 할 수 있는 Web Application 제작
- Python Web Framework를 통한 데이터 조작
- Authentication에 대한 이해
- Database 1:N, M:N 관계의 이해와 데이터 관계 설정

2. 준비사항

A. 언어

- i. Python 3.8+
- ii. Django 3.1.X

B. 도구

- i. vsCode
- ii. Chrome Browser

3. 요구사항

커뮤니티 서비스의 상세 기능 개발을 위한 단계로, 모델간의 관계 설정 후 데이터의 생성, 조회, 수정, 삭제 할 수 있는 기능을 완성합니다. 해당 기능은 향후 커뮤니티 서비스의 필수 기능으로 사용됩니다.

A. 프로젝트 구조

pjt07/은 startproject 명령어로 생성되는 project 디렉토리입니다.

community/는 startapp 명령어로 생성되는 application 디렉토리입니다.

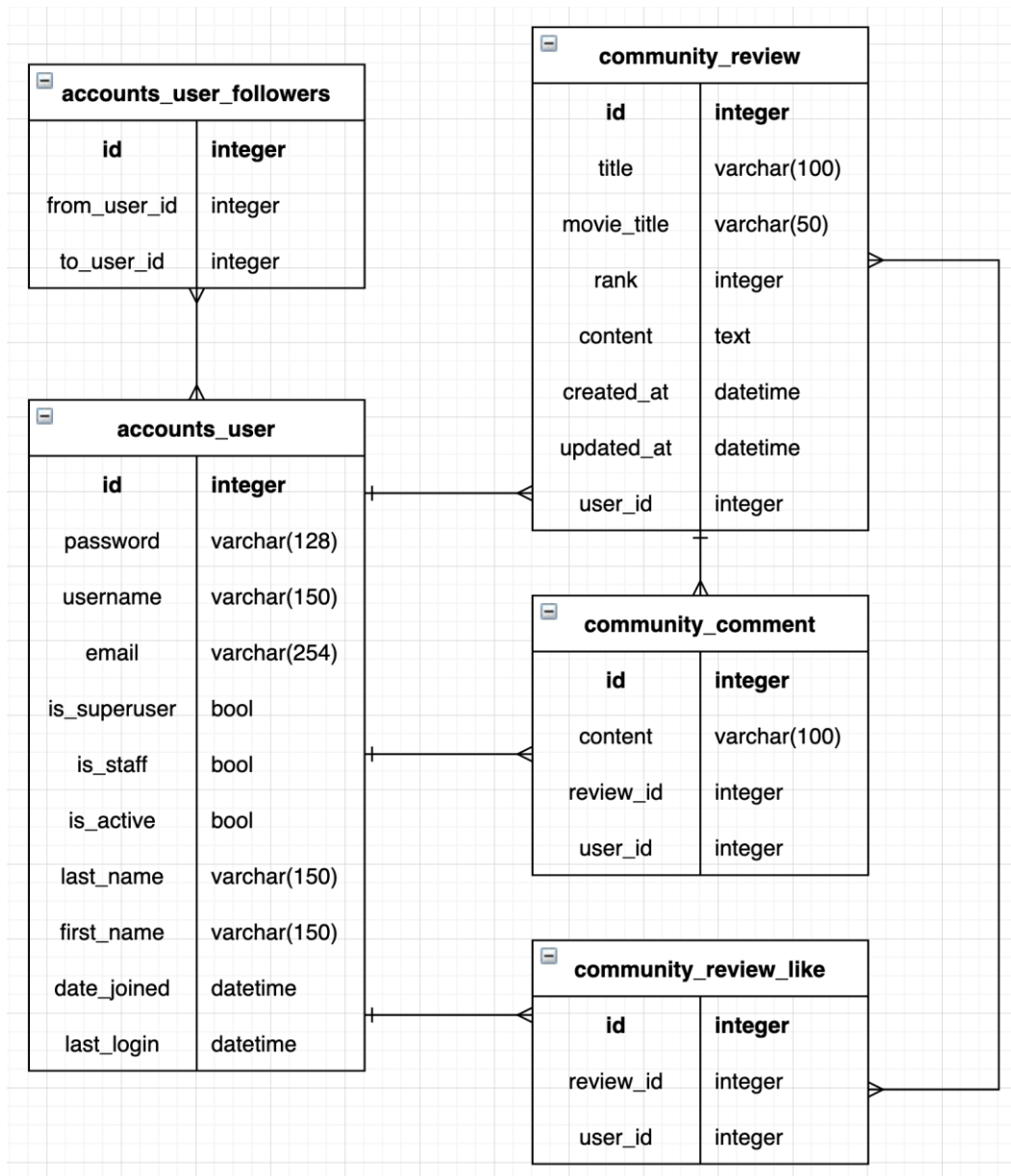
아래의 폴더구조는 주요한 폴더와 파일만 명시되어 있습니다.

```
pjt07/  
  templates/  
    base.html  
  settings.py
```

```
    urls.py
accounts/
    migrations/
    templates/
        accounts/
            *.html
    forms.py
    models.py
    urls.py
    views.py
community/
    migrations/
    templates/
        community/
            *.html
    forms.py
    models.py
    urls.py
    views.py
manage.py
...
```

B. Model

데이터베이스에서 모델의 ERD(Entity Relation Diagram)는 아래와 같습니다.



C. URL

URL은 app_name과 name을 설정하여 적용합니다.

i. accounts app

accounts app의 모든 URL 패턴은 accounts/로 시작합니다.

HTTP verb	URL 패턴	설명
GET & POST	signup/	Form 표시 및 신규 사용자 생성(회원가입)
GET & POST	login/	Form 표시 및 기존 사용자 인증(로그인)
GET / POST	logout/	인증된 사용자 인증 해제(로그아웃)
GET	<user_id>/	사용자 상세 조회 페이지
GET / POST	<user_id>/follow/	팔로우 추가 / 팔로우 취소

ii. community app

community app의 모든 URL 패턴은 community/로 시작합니다.

HTTP verb	URL 패턴	설명
GET & POST	create/	Form 표시 및 신규 리뷰 생성
GET		전체 리뷰 목록 조회 페이지
GET	<review_pk>/	단일 리뷰 상세 조회 페이지
POST	<review_pk>/comments/	댓글 생성
POST	<review_pk>/like/	좋아요 기능

D. View & Template

i. 공유 템플릿 생성 및 사용

- 모든 HTML파일은 base.html을 확장(extends)하여 사용합니다.
- base.html은 모든 페이지가 공유하는 상단 네비게이션 바를 표시합니다.
- 네비게이션 바
 - 전체 리뷰 목록 조회 페이지와 새로운 리뷰 작성 페이지로 이동 할 수 있는 링크를 포함합니다.
 - 사용자가 인증되어 있지 않다면, 로그인 페이지와 회원 가입 페이지로 이동 할 수 있는 링크를 포함합니다.
 - 사용자가 인증되어 있다면, 로그아웃 할 수 있는 링크를 표시합니다.

ii. 리뷰 좋아요 기능

- 단일 리뷰 상세 조회 페이지에 좋아요 버튼과 좋아요 개수를 출력합니다.
- 이미 좋아요 버튼을 누른 경우 취소 버튼을 출력합니다.
- 인증된 사용자만 리뷰에 좋아요 할 수 있습니다.

iii. 유저 팔로우 기능

1. 사용자 상세 페이지에 팔로우 버튼과 팔로우수를 출력합니다.
2. 이미 팔로우 버튼을 누른 경우 취소 버튼을 출력합니다.
3. 인증된 사용자만 팔로우 할 수 있습니다.
4. 로그인한 사용자 자신은 팔로우 할 수 없습니다.

iv. 추가적인 Styling

1. Bootstrap을 활용하여 자유롭게 스타일링 합니다.

4. 결과

- A. 제출 기한은 **21.04.02(금) 18:00** 까지 입니다. 제출기한을 엄수하여 주시기 바랍니다.
- B. 반드시 README.md 파일에 각 단계별 상세 구현 과정, 학습 내용 및 어려웠던 부분을 기록하여 제출해야 합니다.
- C. 위에 명시된 사항은 최소 조건이며 추가적인 정보를 수정 할 수 있습니다.
- D. <https://lab.ssafy.com>에 pjt07 프로젝트를 생성하고 해당 프로젝트에 업로드하여 제출합니다.
- E. 각 반 담당 교수님을 Maintainer로 설정해야 합니다.