

# 관계형 데이터베이스 설계

## 1. 목표

- 데이터를 생성, 조회, 수정, 삭제할 수 있는 Web application 제작
- Django web framework를 통한 데이터 조작
- Authentication에 대한 이해
- Database 1:N, M:N 관계의 이해와 데이터 관계 설정

## 2. 준비사항

### A. 언어

- i. Python 3.9+
- ii. Django 3.2+

### B. 도구

- i. Visual Studio Code
- ii. Chrome Browser

## 3. 요구사항

커뮤니티 서비스의 상세 기능 개발을 위한 단계로, 모델간의 관계 설정 후 데이터의 생성, 조회, 수정, 삭제할 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/
  settings.py
  urls.py

templates/
  base.html

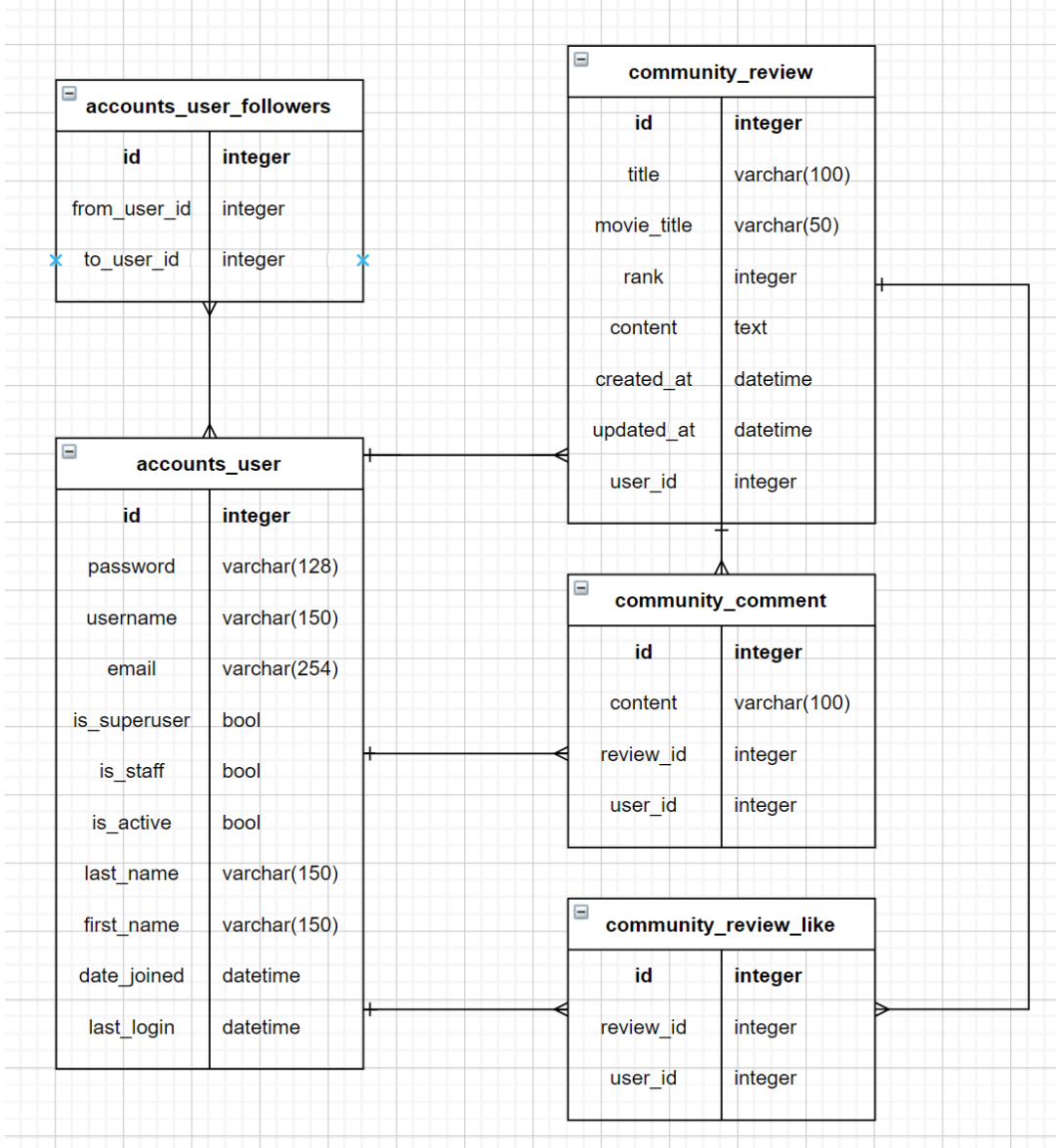
accounts/
  migrations/
  templates/
    accounts/
      *.html
  forms.py
  models.py
  urls.py
  views.py
  ...

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

manage.py
...
```

## B. Model

데이터베이스에서 모델의 ERD(Entity-Relationship Diagram)는 아래와 같습니다.



## C. Form

Review 모델과 Comment 모델의 데이터 검증, 저장, 에러메시지, HTML을 모두 관리하기 위해 ModelForm을 사용합니다.

User 모델의 데이터 검증, 저장, 에러메시지, HTML을 관리하기 위해 커스텀 ModelForm을 사용합니다.

User 모델의 인증(로그인, 로그아웃)은 django에서 제공하는 Form을 사용합니다.

## D. URL

URL은 app\_name과 name을 설정하여 적용합니다.

### i. accountsapp

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 or POST	logout/	인증된 사용자 인증 해제(로그아웃)
GET	<username>/	사용자 상세 조회 페이지
POST	<username>/follow/	팔로우 추가/팔로우 취소

### ii. communityapp

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/create/	댓글 생성
POST	<review_pk>/like/	좋아요 기능

리뷰 삭제, 리뷰 수정, 댓글 삭제 등의 구현은 선택사항입니다.

## E. 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
  - 사용자의 인증 여부와 관계없이 전체 리뷰 목록 조회 페이지로 이동할 수 있는 링크를 표시합니다.
  - 사용자가 인증되어 있지 않다면, 로그인 페이지와 회원가입 페이지로 이동할 수 있는 링크를 표시합니다.
  - 사용자가 인증되어 있다면, 새로운 리뷰 작성페이지로 이동할 수 있

는 링크, 로그아웃 할 수 있는 링크, 프로필 확인페이지로 이동할 수 있는 링크를 표시합니다.

## ii. 리뷰 좋아요 기능

1. 단일 리뷰 상세 조회 페이지에 **좋아요** 버튼과 좋아요 개수를 표시합니다.
2. 이미 좋아요 버튼을 누른 경우 취소 버튼을 표시합니다.
3. 인증된 사용자만 리뷰에 좋아요 할 수 있습니다.

## iii. 유저 팔로우 기능

1. 사용자 프로필 페이지에 팔로워 수와 팔로잉 수를 표시합니다.
2. 프로필 페이지에는 해당 프로필 페이지의 사용자를 팔로우 할 수 있는 버튼을 표시합니다.
3. 인증(로그인)된 사용자만 다른 사용자를 팔로우 할 수 있으며, 사용자는 자기 자신을 팔로우 할 수 없습니다.
4. 이미 팔로우 중인 경우에는 팔로우 취소 버튼을 표시합니다.

## iv. 추가적인 Styling

1. Bootstrap을 활용하여 자유롭게 스타일링 합니다.

## 4. 결과

- A. 제출 기한은 **21.10.22(금) 18:00**까지입니다. 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 수정할 수 있습니다.
- D. <https://lab.ssafy.com>에 pjt07 프로젝트를 생성하고 해당 프로젝트에 업로드하여 제출합니다.
- E. 각 반 담당 교수님을 Maintainer로 설정해야 합니다.