



데이터베이스설계 기말 프로젝트

제출일	2024년 12월 01일	학과	컴퓨터공학전공
과목	데이터베이스설계_01	학번	2020112736
담당교수	이강만 교수님	이름	안성현

1. 개요

1.1 과제 개요

본 프로젝트는 영화 리뷰 플랫폼의 데이터베이스 설계 및 구현을 목표로 하며, 사용자가 영화에 대한 리뷰를 작성, 수정, 삭제하고, 다른 사용자의 리뷰와 평점을 조회할 수 있는 기능을 제공한다.

1.2 데이터베이스 설계의 필요성

본 프로젝트는 여러 사용자가 동시에 리뷰를 작성하고 조회할 수 있는 시스템을 구현한다. 데이터 중복으로 인한 비효율성, 데이터의 비일관성과 같은 문제를 해결하기 위해 데이터의 일관성과 효율적인 처리를 보장할 수 있는 데이터베이스 설계가 필요하다. 이를 위해, 관계형 데이터베이스 MySQL을 사용해 데이터를 체계적으로 저장하고, 무결성 제약 조건으로 데이터의 일관성을 보장하는 방안을 마련한다.

1.3 과제 목표 및 해결하고자 하는 문제

효율적인 데이터베이스 설계를 통해 영화 리뷰 플랫폼을 구현하는 본 프로젝트는 사용자가 영화별 리뷰를 효율적으로 조회할 수 있는 시스템을 구축하여 영화에 대한 의견을 나눌 수 있는 공간을 제공한다. 이를 통해, 영화에 대한 사용자들의 다양한 시각을 공유하고 리뷰와 별점을 통해 다른 사용자들이 영화를 선택하는데 도움을 주는 것을 목표로 한다.

2. 명세서 작성 및 데이터베이스 설계

2.1 요구 사항 분석

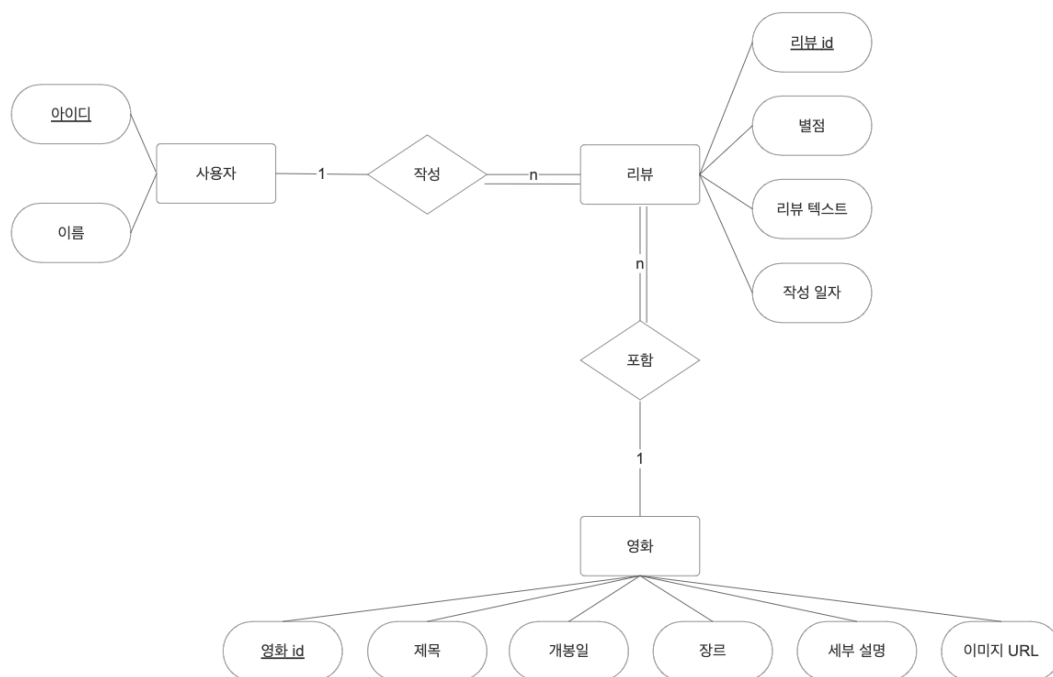
영화 리뷰 플랫폼의 데이터베이스 설계를 위해 시스템의 주요 기능과 요구사항을 명확하게 정의한다.

1. 사용자는 각 영화에 대한 별점과 리뷰를 작성할 수 있어야 한다.
2. 서비스를 이용하기 위해서는 회원가입 및 로그인을 하여야 한다.
3. 사용자가 되기 위해 카카오 로그인을 통해 카카오 아이디와 이름을 제공한다.
4. 사용자는 카카오 아이디로 식별한다.
5. 사용자는 메인 화면에서 영화 목록을 열람할 수 있어야 한다.
6. 영화는 영화id, 영화제목, 개봉일, 장르 정보, 세부 설명, 이미지 url을 유지한다.
7. 영화는 영화id로 식별한다.
8. 사용자는 각 영화에 대한 모든 리뷰와 평균 별점을 확인할 수 있어야 한다.
9. 리뷰는 리뷰id, 사용자id, 영화id, 별점, 리뷰 텍스트, 작성 일자를 유지한다.
10. 리뷰는 리뷰id로 식별한다.
11. 사용자는 여러 영화에 대한 리뷰를 작성할 수 있어야 한다.
12. 하나의 영화에 대한 리뷰를 여러 사용자가 작성할 수 있어야 한다.
13. 사용자는 자신이 쓴 리뷰에 대한 수정, 삭제를 할 수 있어야 한다.

2.2 개념적 설계

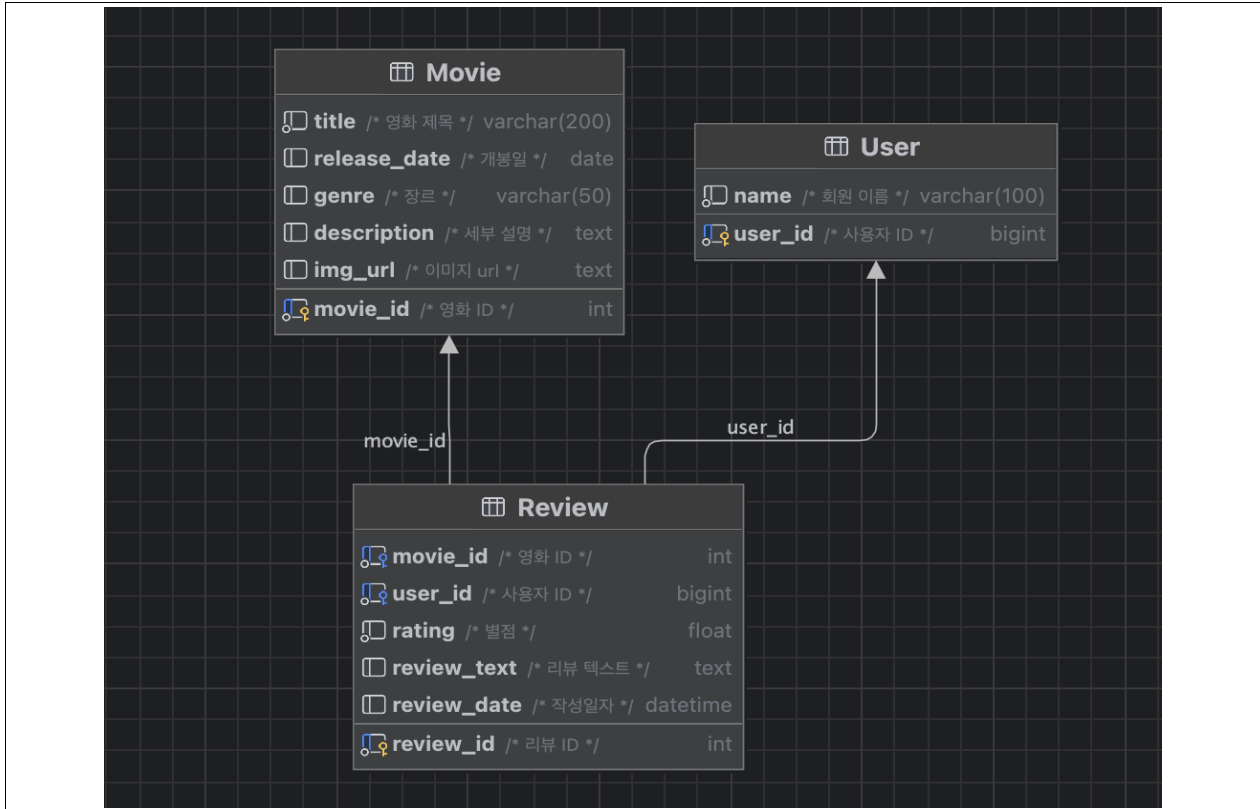
요구사항 분석의 결과물을 개념적 데이터 모델을 이용해 개념적 구조로 나타낸다. 중요한 개체를 추출하고, 개체 간의 관계를 결정하여 E-R 다이어그램으로 표현한다.

개체	속성
사용자	<u>아이디</u> , 이름
영화	<u>영화 id</u> , 제목, 개봉일, 장르, 세부 설명, 이미지 URL
리뷰	<u>리뷰 id</u> , 별점, 리뷰 텍스트, 작성 일자



2.3 논리적 설계

개념적 설계의 결과물인 E-R 다이어그램을 데이터베이스의 릴레이션 스키마로 변환한다. 변환 규칙을 고려하여 설계를 진행한다.



2.4 물리적 설계, 구현

논리적 설계의 결과물인 릴레이션 스키마를 토대로 물리적인 저장 구조를 설계하고 SQL문을 작성하고 데이터베이스를 생성한다.

DB 생성 SQL문

```

-- User, Review 테이블 SQL 문 분량 상 생략

-- Review 테이블 생성 (별점 제약 조건, 외래키 제약 추가)
CREATE TABLE Review (
    (분량 상 생략)
    CONSTRAINT fk_review_movie FOREIGN KEY (movie_id) REFERENCES Movie(movie_id),
    CONSTRAINT fk_review_user FOREIGN KEY (user_id) REFERENCES User(user_id),
    CONSTRAINT chk_rating CHECK (rating >= 0 AND rating <= 5 AND MOD(rating * 10, 5) = 0)
);

-- review_date 생성 Trigger
DELIMITER $$
CREATE TRIGGER set_review_date
BEFORE INSERT ON Review
FOR EACH ROW
BEGIN
    SET NEW.review_date = NOW();
END$$
DELIMITER ;
  
```

```

-- 리뷰 수정 시 review_date 생성 Trigger
DELIMITER $$
CREATE TRIGGER set_review_date_before_update
BEFORE UPDATE ON Review
FOR EACH ROW
BEGIN
    SET NEW.review_date = NOW();
END$$
DELIMITER ;

-- 검색 속도 최적화를 위한 영화 id 인덱스 설정
CREATE INDEX idx_movie_id ON Movie(movie_id);

```

3. 명세서 작성 및 데이터베이스 설계


3.1 프로그램 구현 개요

본 프로젝트는 Spring boot와 JPA를 이용해 리뷰 평점, 리뷰 CRUD 기능을, Thymeleaf 템플릿 엔진을 통해 영화 조회, 리뷰 열람 등의 UI 인터페이스를 표시하는 프론트엔드를 구현하였다. 영화들의 데이터 정보는 TMDB API를 이용해 인기 영화 20개를 랜덤으로 추출하여 데이터베이스에 저장하였다. 추가로, 카카오 API와 REST API를 통해 로그인 기능을 제공한다.


3.2 프로그램 UI 및 기능 소개

메인 화면


Movie Review
카카오 로그인




위키드



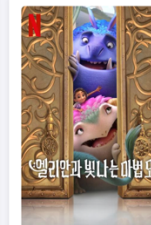
슈퍼배드 4




데드풀과 울버린





글래디에이터 II

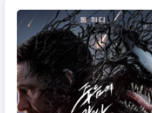



엘리안과 빛나는 마법 모험





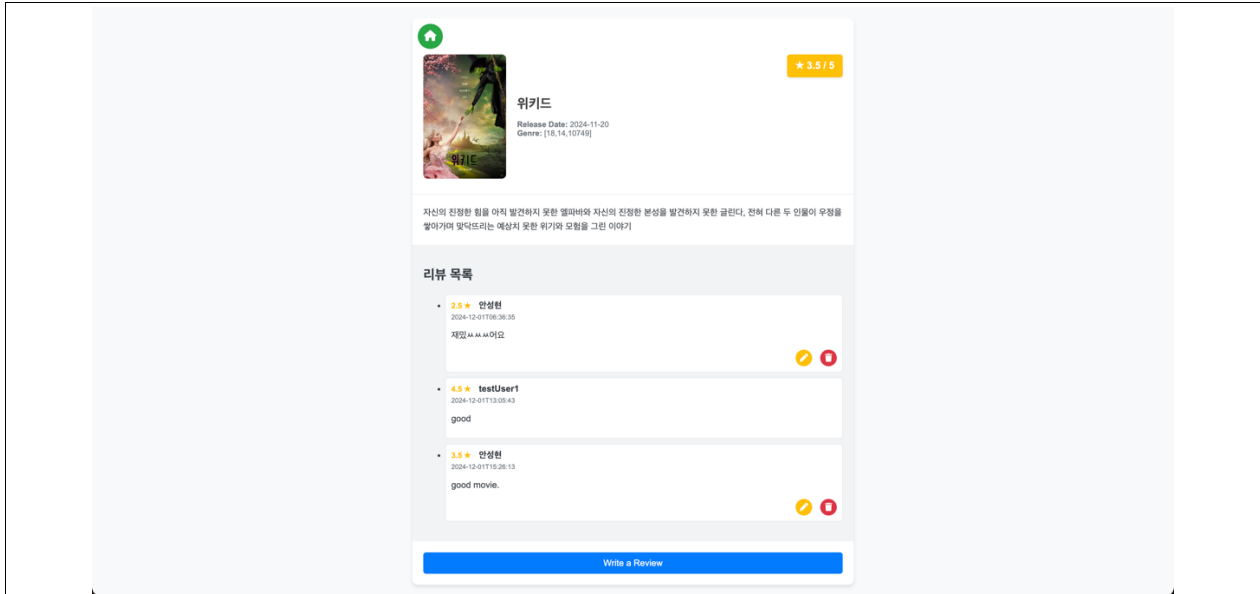




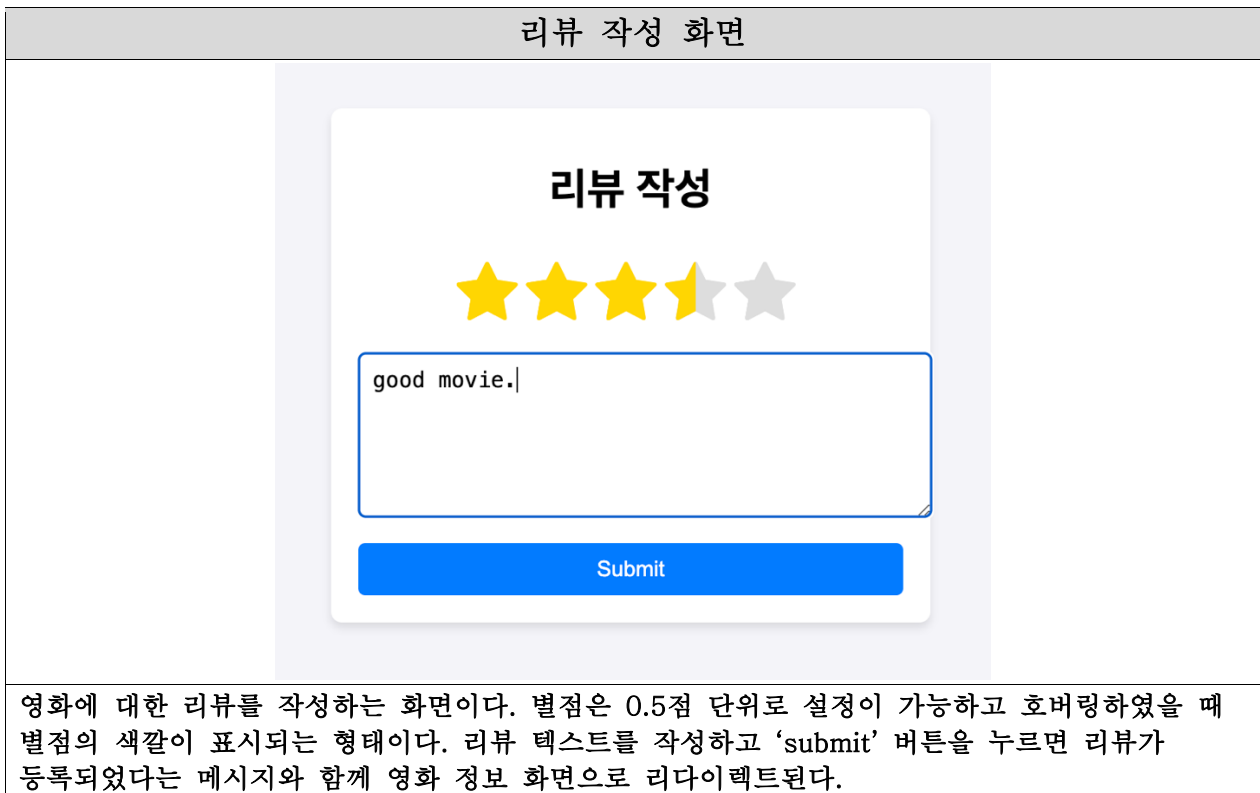


영화 목록을 조회할 수 있는 메인 화면이다. 카카오 api를 이용한 회원가입 및 로그인 기능을 제공한다.

영화 정보 화면



각 영화에 대한 상세 정보 및 리뷰를 확인할 수 있는 화면이다. 우측 상단에는 이 영화에 대한 평균 평점을 SQL avg() 문을 통해 계산하여 표시한다. 본인이 작성한 리뷰에 대한 수정, 삭제 기능도 제공한다.



영화에 대한 리뷰를 작성하는 화면이다. 별점은 0.5점 단위로 설정이 가능하고 호버링하였을 때 별점의 색깔이 표시되는 형태이다. 리뷰 텍스트를 작성하고 'submit' 버튼을 누르면 리뷰가 등록되었다는 메시지와 함께 영화 정보 화면으로 리다이렉트된다.

3.3 관련 링크

배포 URL : <http://211.188.49.69:8082/movies>

깃허브 링크 : <https://github.com/anismynm/DatabaseDesignProject>