

데이터베이스 설계 과제

영화 예매 시스템

시스템 개요

- 영화관의 영화 예매 및 취소 과정을 효율적으로 관리하기 위한 시스템을 개발한다.
- 일반 이용자(영화 관람객)는 영화 검색, 상영 일정/좌석 조회, 예매/예매 이력 조회, 리뷰 작성/조회 기능을 사용한다.
- 관리자는 영화 및 상영 일정, 상영관, 좌석 정보를 관리하며 회원 정보, 예매 현황, 리뷰 현황을 조회한다.

기능 요구사항

- 영화 검색 기능
- 영화 예매 기능
- 예매 취소 기능
- 상영 일정 조회 기능
- 상영관 좌석 조회 기능
- 영화 리뷰 작성 기능
- 영화 리뷰 조회 기능
- 회원 관리 기능(가입, 로그인, 로그아웃, 정보 수정, 탈퇴 등)
- 관리자 기능(상영정보, 상영관, 회원정보, 예매 현황 관리 등)

비기능 요구사항

- 신뢰성: 예매, 취소 등의 정보는 시스템 장애 발생 후에도 보존되어야 한다.
- 보안/권한: 예매, 취소, 내역 조회 등의 정보는 로그인 된 회원만 이용할 수 있어야 하며 관리자 기능은 권한이 부여된 계정으로만 접근할 수 있어야 한다.
- 성능: 버튼 클릭 시 화면이 5초 이내로 표시되어야 한다.

요구사항 정의

1. 영화 예매 시스템의 회원으로 가입하려면 이름, 아이디, 비밀번호, 전화번호를 입력해야 한다.
2. 회원은 아이디로 식별되며, 등급과 적립금이 부여될 수 있다.
3. 회원 정보는 일반과 관리자를 구분할 수 있는 권한이 포함된다.
4. 영화는 영화번호로 식별되며, 영화번호, 제목, 상영시간, 장르, 관람등급 등의 정보를 유지해야 한다.
5. 상영 정보는 상영번호로 식별되며, 상영번호, 상영일시, 상영관, 영화정보 등의 정보를 유지해야 한다.
6. 회원은 여러 회차의 영화를 예매할 수 있고, 하나의 좌석은 동일한 상영 정보에 한해 한 회원만 예매할 수 있다.

요구사항 정의

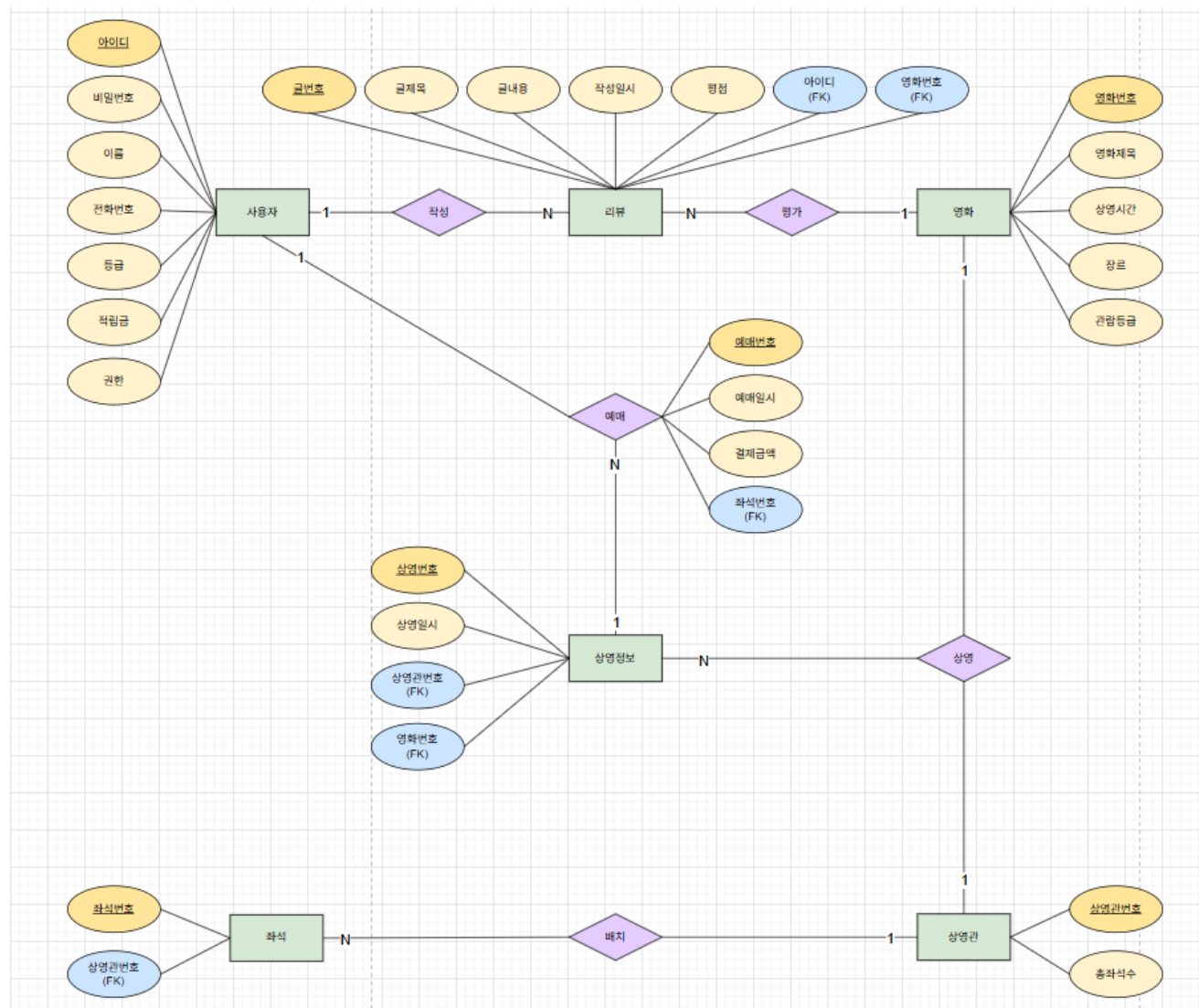
7. 회원이 영화를 예매하면 예매번호, 예매일시, 결제금액, 좌석번호, 상영정보 등의 정보를 저장해야 한다.
8. 회원은 영화 제목, 장르 등의 정보를 통해 영화를 검색할 수 있다.
9. 회원은 인증 후에만 영화를 예매·취소할 수 있다.
10. 상영관은 상영관 번호로 식별되며, 상영관 번호와 총 좌석 수를 유지해야 한다. 하나의 상영관은 여러 좌석을 가진다.
11. 좌석은 좌석 번호와 상영관 번호로 구분되며, 각 좌석은 하나의 상영관에만 속한다.
12. 시스템은 상영 시작 일정 시간 전까지 예매 취소를 허용하며, 취소 시 예매 및 결제 정보를 갱신해야 한다.

요구사항 정의

13. 회원은 가입 후 자신의 회원 정보를 수정할 수 있으며, 과거 예매 이력을 조회할 수 있다.
14. 회원은 관람한 영화에 대해 리뷰를 작성할 수 있다.
15. 리뷰에는 리뷰 번호, 리뷰 제목, 리뷰 내용, 작성 일시, 평점 정보와 함께 작성자 아이디와 영화번호가 저장되어야 한다.
16. 한 영화에는 여러 리뷰가 작성될 수 있으며, 각 리뷰는 하나의 영화와 하나의 회원에 의해 작성된다.
17. 관리자는 영화 정보와 상영 일정(상영정보), 상영관 및 좌석 정보를 등록·수정·삭제할 수 있다.
18. 관리자는 회원 정보와 예매 현황(상영관 좌석 배정, 예매율 등) 및 리뷰 현황을 조회할 수 있다.
19. 관리자 기능은 관리자 권한을 가진 인증된 사용자만 접근할 수 있다.

개념적 설계

E-R 다이어그램



논리적 설계

릴레이션 스키마							
회원	<u>아이디</u>	비밀번호	이름	전화번호	등급	적립금	권한
영화	<u>영화번호</u>	영화제목	상영시간	장르	관람등급		
상영관	<u>상영관번호</u>	총좌석수					
좌석	<u>좌석번호</u>	상영관번호					
상영정보	<u>상영번호</u>	상영일시	상영관번호	영화번호			
예매	<u>예매번호</u>	예매일시	결제금액	좌석번호	아이디	상영번호	
리뷰	<u>글번호</u>	글제목	글내용	작성일시	평점	아이디	영화번호

논리적 설계

테이블 명세서						
테이블 이름			회원			
속성 이름	데이터 타입	Null 허용여부	기본값	기본키	외래키	제약조건
아이디	Varchar2(20)	N		PK		
비밀번호	Varchar2(20)	N				
이름	Varchar2(20)	N				
전화번호	Varchar2(20)	N				
등급	Varchar2(10)	N	basic			basic, gold, vip만 허용
적립금	Int	N	0			0 이상
권한	Varchar2(10)	N	user			user, admin만 허용

논리적 설계

테이블 명세서						
테이블 이름			영화			
속성 이름	데이터 타입	Null 허용여부	기본값	기본키	외래키	제약조건
영화번호	Varchar2(20)	N		PK		
영화제목	Varchar2(50)	N				
상영시간	Int	N				0 이상
장르	Varchar2(20)	N				
관람등급	Varchar2(10)	N	15세			전체, 12세, 15세, 19세만 허용

논리적 설계

테이블 명세서						
테이블 이름			상영관			
속성 이름	데이터 타입	Null 허용여부	기본값	기본키	외래키	제약조건
상영관번호	Varchar2(20)	N		PK		
총좌석수	Int	N				0 이상

논리적 설계

테이블 명세서						
테이블 이름			좌석			
속성 이름	데이터 타입	Null 허용여부	기본값	기본키	외래키	제약조건
좌석번호	Varchar2(10)	N		PK		
상영관번호	Varchar2(20)	N			FK	

논리적 설계

테이블 명세서						
테이블 이름			상영정보			
속성 이름	데이터 타입	Null 허용여부	기본값	기본키	외래키	제약조건
상영번호	Varchar2(20)	N		PK		
상영일시	Date	N				
상영관번호	Varchar2(20)	N			FK	
영화번호	Varchar2(20)	N			FK	

논리적 설계

테이블 명세서						
테이블 이름			예매			
속성 이름	데이터 타입	Null 허용여부	기본값	기본키	외래키	제약조건
예매번호	Varchar2(20)	N		PK		
예매일시	Date	N				
결제금액	Int	N				0 이상
좌석번호	Varchar2(10)	N			FK	
아이디	Varchar2(20)	N			FK	
상영번호	Varchar2(20)	N			FK	

논리적 설계

테이블 명세서						
테이블 이름			리뷰			
속성 이름	데이터 타입	Null 허용여부	기본값	기본키	외래키	제약조건
글번호	Varchar2(20)	N		PK		
글제목	Varchar2(20)	N				
글내용	Varchar2(200)	N				
작성일시	Date	N				
평점	Int	N				1 이상 5 이하
아이디	Varchar2(20)	N			FK	
영화번호	Varchar2(20)	N			FK	

SQL 구현

```
CREATE TABLE 회원 (  
  아이디      VARCHAR2(20) PRIMARY KEY,  
  비밀번호    VARCHAR2(20) NOT NULL,  
  이름        VARCHAR2(20) NOT NULL,  
  전화번호    VARCHAR2(20) NOT NULL,  
  등급        VARCHAR2(10) DEFAULT 'basic' NOT NULL  
              CHECK (등급 IN ('basic', 'gold', 'vip')),  
  적립금      INT          DEFAULT 0 NOT NULL  
              CHECK (적립금 >= 0),  
  권한        VARCHAR2(10) DEFAULT 'user' NOT NULL  
              CHECK (권한 IN ('user', 'admin'))  
);
```

```
INSERT INTO 회원 VALUES ('hhs01', '1234', '현혜수', '010-1111-2222', 'basic', 0, 'user');  
INSERT INTO 회원 VALUES ('kgy01', '5678', '김재영', '010-3333-4444', 'basic', 500, 'user');
```

	아이디	비밀번호	이름	전화번호	등급	적립금	권한
1	hhs01	1234	현혜수	010-1111-2222	basic	0	user
2	kgy01	5678	김재영	010-3333-4444	basic	500	user

SQL 구현

```
CREATE TABLE 영화 (  
    영화번호    VARCHAR2(20) PRIMARY KEY,  
    영화제목    VARCHAR2(30) NOT NULL,  
    상영시간    INT          NOT NULL  
                CHECK (상영시간 >= 0),  
    장르        VARCHAR2(20) NOT NULL,  
    관람등급    VARCHAR2(10) DEFAULT '15세' NOT NULL  
                CHECK (관람등급 IN ('전체', '12세', '15세', '19세'))  
);
```

```
INSERT INTO 영화 VALUES ('a001', '주토피아', 110, 'animation', '전체');  
INSERT INTO 영화 VALUES ('a002', '주토피아2', 108, 'animation', '전체');  
INSERT INTO 영화 VALUES ('a003', '나우유씨미3', 112, 'action', '12세');
```

	⇄ 영화번호	⇄ 영화제목	⇄ 상영시간	⇄ 장르	⇄ 관람등급
1	a001	주토피아	110	animation	전체
2	a002	주토피아2	108	animation	전체
3	a003	나우유씨미3	112	action	12세

SQL 구현

```
CREATE TABLE 상영관 (  
    상영관번호 VARCHAR2(20) PRIMARY KEY,  
    총좌석수    INT          NOT NULL  
                CHECK (총좌석수 >= 0)  
);
```

```
INSERT INTO 상영관 VALUES ('T01', 100);  
INSERT INTO 상영관 VALUES ('T02', 130);
```

	상영관번호	총좌석수
1	T01	100
2	T02	130

SQL 구현

```
CREATE TABLE 좌석 (  
    좌석번호    VARCHAR2(10) PRIMARY KEY,  
    상영관번호  VARCHAR2(20) NOT NULL  
                REFERENCES 상영관(상영관번호)  
);
```

```
INSERT INTO 좌석 VALUES ('T01-A1', 'T01');  
INSERT INTO 좌석 VALUES ('T01-A2', 'T01');  
INSERT INTO 좌석 VALUES ('T01-B1', 'T01');  
INSERT INTO 좌석 VALUES ('T01-B2', 'T01');  
  
INSERT INTO 좌석 VALUES ('T02-A1', 'T02');  
INSERT INTO 좌석 VALUES ('T02-A2', 'T02');  
INSERT INTO 좌석 VALUES ('T02-B1', 'T02');  
INSERT INTO 좌석 VALUES ('T02-B2', 'T02');
```

	좌석번호	상영관번호
1	T01-A1	T01
2	T01-A2	T01
3	T01-B1	T01
4	T01-B2	T01
5	T02-A1	T02
6	T02-A2	T02
7	T02-B1	T02
8	T02-B2	T02

SQL 구현

```
CREATE TABLE 상영정보 (  
    상영번호    VARCHAR2(20) PRIMARY KEY,  
    상영일시    DATE          NOT NULL,  
    상영관번호  VARCHAR2(20) NOT NULL  
                REFERENCES 상영관(상영관번호),  
    영화번호    VARCHAR2(20) NOT NULL  
                REFERENCES 영화(영화번호)  
);
```

```
INSERT INTO 상영정보 VALUES ('S001', TO_DATE('2025-12-01 19:00', 'YYYY-MM-DD HH24:MI'), 'T01', 'a001');  
INSERT INTO 상영정보 VALUES ('S002', TO_DATE('2025-12-02 11:00', 'YYYY-MM-DD HH24:MI'), 'T01', 'a003');  
INSERT INTO 상영정보 VALUES ('S003', TO_DATE('2025-12-02 11:30', 'YYYY-MM-DD HH24:MI'), 'T02', 'a002');
```

	상영번호	상영일시	상영관번호	영화번호
1	S001	2025-12-01 19:00	T01	a001
2	S002	2025-12-02 11:00	T01	a003
3	S003	2025-12-02 11:30	T02	a002

SQL 구현

```
CREATE TABLE 예매 (
  예매번호    VARCHAR2(20) PRIMARY KEY,
  예매일시    DATE          NOT NULL,
  결제금액    INT           NOT NULL
              CHECK (결제금액 >= 0),
  좌석번호    VARCHAR2(10) NOT NULL
              REFERENCES 좌석 (좌석번호),
  아이디      VARCHAR2(20) NOT NULL
              REFERENCES 회원 (아이디),
  상영번호    VARCHAR2(20) NOT NULL
              REFERENCES 상영정보 (상영번호)
);
```

```
INSERT INTO 예매 VALUES ('R001', SYSDATE, 12000, 'T01-A1', 'hhs01', 'S001');
INSERT INTO 예매 VALUES ('R002', SYSDATE, 12000, 'T01-A2', 'k jy01', 'S001');
INSERT INTO 예매 VALUES ('R003', SYSDATE, 12000, 'T02-B2', 'k jy01', 'S003');
```

	예매번호	예매일시	결제금액	좌석번호	아이디	상영번호
1	R001	2025-11-26 22:03	12000	T01-A1	hhs01	S001
2	R002	2025-11-26 22:03	12000	T01-A2	k jy01	S001
3	R003	2025-11-26 22:19	12000	T02-B2	k jy01	S003

SQL 구현

```
CREATE TABLE 리뷰 (  
    글번호    VARCHAR2(20) PRIMARY KEY,  
    글제목    VARCHAR2(20)  NOT NULL,  
    글내용    VARCHAR2(200) NOT NULL,  
    작성일시  DATE          NOT NULL,  
    평점      INT           NOT NULL  
                CHECK (평점 BETWEEN 1 AND 5),  
    아이디     VARCHAR2(20)  NOT NULL  
                REFERENCES 회원 (아이디),  
    영화번호 VARCHAR2(20)  NOT NULL  
                REFERENCES 영화 (영화번호)  
);
```

	글번호	글제목	글내용	작성일시	평점	아이디	영화번호
1	RV001	재밌다	닉이랑 주디 너무 귀엽다	2025-11-26 22:04	5	hhs01	a001
2	RV002	꿀잼	속편 기대돼요	2025-11-26 22:05	5	hhs01	a003

```
INSERT INTO 리뷰 VALUES ('RV001', '재밌다', '닉이랑 주디 너무 귀엽다', SYSDATE, 5, 'hhs01', 'a001');  
INSERT INTO 리뷰 VALUES ('RV002', '꿀잼', '속편 기대돼요', SYSDATE, 5, 'hhs01', 'a003');
```