



최종설계 보고서

리더 | 2020158007 | 김태양

팀원 | 2016200047 | 최용현

팀원 | 2020158004 | 김승진

팀원 | 2020158035 | 조한웅

목차

1.	프로젝트 개요 -----
1.1	프로젝트 주제 및 설명
1.2	주제 선정 이유
2.	요구사항 분석 -----
2.1	요구사항 분석 및 수집(객체 및 속성)
2.2	요구사항 분석 및 수집(관계)
3.	개념적 설계 -----
3.1	1 차 개념적 설계
3.2	최종 개념적 설계
4.	논리적 설계 -----
4.1	논리적 설계 1 단계
4.2	논리적 설계 2 단계
4.3	논리적 설계 4 단계
4.4	논리적 설계 5 단계
4.5	논리적 설계 6 단계
4.6	최종 논리 스키마
5.	물리적 설계 -----
5.1	테이블 기술서
5.2	테이블 관계 다이어그램

6. 트랜잭션 설계 -----

6.1 주요 기능

6.2 화면 흐름도

6.3 주요기능 명세

6.4 뷰 및 인덱스

6.5 트랜잭션 SQL 문

7. 별첨-----

7.1 소감

1. 프로젝트 개요

1.1 프로젝트 주제 및 설명

NETFLIX는 TV 프로그램, 영화, 애니메이션, 다큐멘터리 등 다양한 콘텐츠를 인터넷 연결이 가능한 디바이스에서 시청할 수 있는 스트리밍 서비스이다. 총 4,200만여 장의 영상물을 보유하고 있으며 2018년 3분기 기준으로 전세계에서 1억 3,700만 명의 회원들에게 서비스 중으로 온라인 동영상 시장 점유율 1위를 기록하고 있다.

1.2 주제 선정 이유

직접 대면하지 않는 시대가 오면서 NETFLIX는 매우 많은 사람이 이용하고 있다. NETFLIX에는 다양한 영상들이 존재하고, 원하는 시간에 원하는 영상을 볼 수 있는 환경에 제공되며, 내가 원하는 작품들을 쉽게 검색할 수 있다. 이처럼 NETFLIX는 우리 주변에서 쉽게 접할 수 있는 동영상 스트리밍 서비스이고, 많은 영상들이 존재하고, 많은 사람들이 이용하는 만큼 방대한 데이터를 가지고 있다.

2. 요구사항 분석

2.1 요구사항 분석 및 수집(개체와 속성)

1. NETFLIX는 세 개의 구독 상품이 존재한다.
2. 구독상품은 구독 상품번호(고유함), 최대 동시 접속 가능 수, 최대 화질, 가격, 이름을 구성요소로 갖는다. 구독상품마다 차이점이 존재한다.
3. 회원은 각 회원들의 회원 번호(고유함), 회원 이름, 이메일, 비밀번호, 전화번호, 생년월일, 성별을 가진다. 회원들은 구독을 하지 않으면 NETFLIX에 접속을 할 수 없다. 회원은 최대 5개의 프로필을 만들 수 있다.
4. 프로필은 프로필 번호(고유함), 프로필 이름, 연령제한, 접속현황을 가진다. 회원은 하나의 프로필에 들어가 작품을 시청하게 된다.
5. 콘텐츠는 콘텐츠 번호(고유함), 제목, 소개, 연령등급, 평가, NETFLIX 오리지널을 가진다. NETFLIX에는 다양한 콘텐츠와 그에 대한 데이터들이 있다.

6. 영상은 영상 번호(고유함), 줄거리, 재생시간을 가진다. 하나의 영상은 콘텐츠의 시즌, 회차의 데이터를 포함한다.
7. 각 언어들은 언어 번호(고유함)과 언어 정보를 가진다. 언어는 콘텐츠에 다양한 종류의 자막과 더빙을 제공한다.
8. 내용경고는 경고 번호(고유함), 내용 정보를 가진다. 위험한 내용을 가지는 콘텐츠에 내용정보를 태그해 사용자에게 위험한 요소들을 알려준다.
9. 장르는 장르 번호(고유함), 장르명을 가진다. 각 콘텐츠에는 다양한 장르들이 존재하고 장르를 통해 작품을 검색할 수 있다.
10. 제작사는 제작사 번호(고유함), 제작사명을 가진다. 각 제작사는 콘텐츠를 제작한다.
11. 사람은 사람 번호(고유함)와 이름을 가진다. 사람들은 콘텐츠를 제작하는데 참여한다.
12. 각 콘텐츠에 참여하는 사람들은 역할들을 가지고 있다. 역할은 역할 번호(고유함)와 역할명을 가지고 있다.

2.2 요구사항 분석 및 수집(관계)

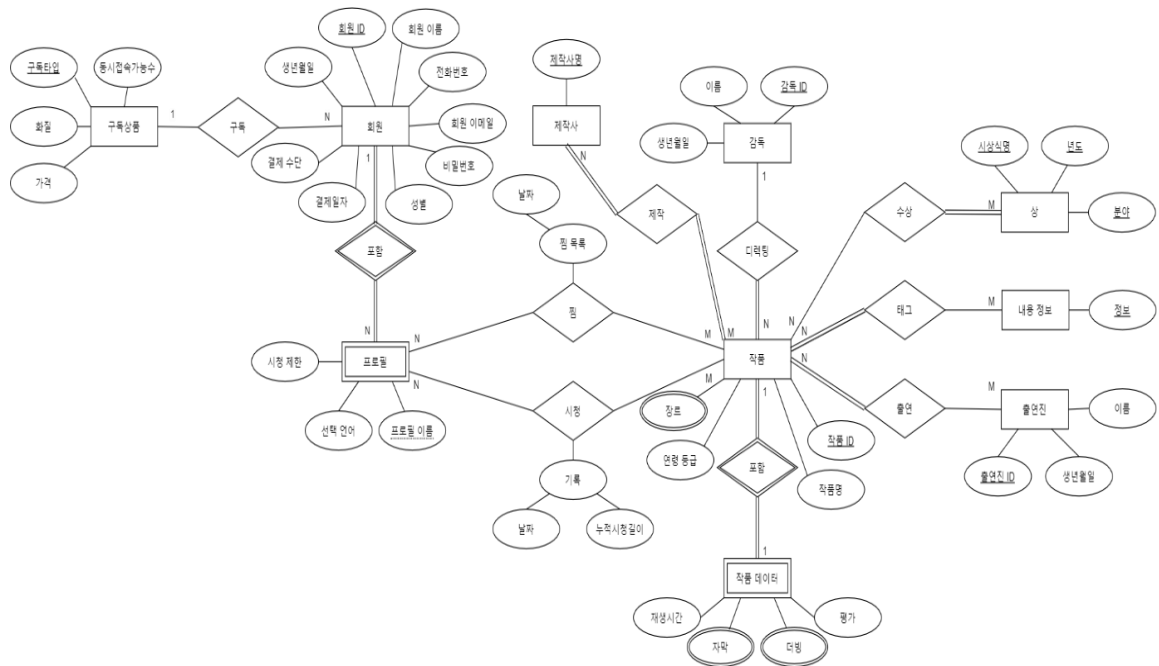
1. 각 회원들은 프로필을 가지고 있다. 모든 회원은 5개까지의 프로필을 포함할 수 있고, 그 프로필들은 1명의 회원이 포함한다.
2. 각 콘텐츠들은 영상을 가지고 있다. 콘텐츠는 여러 개의 영상을 포함하고, 한 영상은 하나의 콘텐츠에 포함된다. 영상은 콘텐츠의 회차와 시즌을 가지고 있다.
3. 프로필은 영상들을 시청한다. 프로필은 여러 개의 영상을 시청할 수 있다. 각 영상마다 여러 개의 프로필이 시청할 수 있다. 프로필이 영상을 시청하면 보기 시작한 시각과 종료한 시각, 영상을 중단한 위치가 저장된다.
4. 콘텐츠는 여러 자막을 지원받는다. 한 콘텐츠에는 여러 자막이 존재할 수 있고, 각 언어의 자막은 여러 콘텐츠에 지원할 수 있다.
5. 콘텐츠는 여러 더빙을 지원받는다. 콘텐츠에 음성이 없을 수 없고, 한 언어의 더빙은 여러 콘텐츠를 지원할 수 있다.
6. 콘텐츠에는 내용경고가 존재한다. 한 콘텐츠에 여러 개의 내용경고를 가지고 있고, 한 내용경고는 여러 개의 콘텐츠에 태그될 수 있다.
7. 콘텐츠는 여러 개의 장르를 가진다. 콘텐츠마다 여러 장르들을 가질 수 있고, 여러 콘텐츠에 한 장르가 태그될 수 있다.

8. 제작사는 콘텐츠를 제작한다. 제작사는 여러 개의 콘텐츠를 제작할 수 있고, 한 콘텐츠를 여러 제작사가 제작할 수 있다.
9. 사람은 콘텐츠에 역할을 가지고 참여한다. 콘텐츠에는 여러 명의 사람들이 참여하고, 참여한 사람들마다 역할이 존재한다. 사람은 여러 콘텐츠에 여러 역할로 참여할 수 있다.
10. 회원은 구독상품을 구독한다. 각 회원마다 하나의 구독상품만 구독할 수 있으며, 각 구독상품은 여러 회원이 구독할 수 있다.
11. 프로파일은 콘텐츠를 찜 할 수 있다. 한 프로필이 여러 콘텐츠를 찜 하고, 찜 한 콘텐츠들은 목록에 저장된다. 한 영상은 여러 프로필이 찜 할 수 있다.

3. 개념적 설계

3.1 1차 개념적 설계

ER모델 설계는 명세된 요구사항과 그에 따라 도출된 엔티티, 애트리뷰트를 바탕으로 진행되었다. 설계된 ER모델은 이후에도 다음 설계 과정을 거치며 발견한 수정이 필요한 사항들을 지속적으로 수정하는 과정을 거쳤다. 수정 전의 ER모델과 수정이 필요한 사항들은 다음과 같다.



■ 수정이 필요한 사항

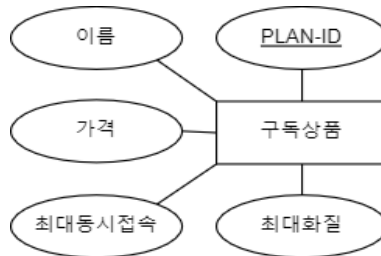
- 엔티티들이 기본 키로 이름을 가진다.
 - 문자는 물리 설계 시 char 타입을 가지기 때문에 기본키가 가지는 인덱스를 통한 튜플 검색의 활용에 부적합하다. 기본키를 인덱스 검색에 적합한 숫자형 키로 대체하여 해결한다.
- 회원은 결제수단과 결제일자를 애트리뷰트로 갖는다.
 - 결제수단과 결제일자는 구독상품과의 구독관계에서 생성되는 것이므로 구독 관계가 가지는 애트리뷰트이다.
- 프로필은 선택언어 애트리뷰트를 갖는다.
 - 실제 NETFLIX는 프로필 별로 언어를 선택할 수 없으므로 선택언어 애트리뷰트를 제거한다.
- NETFLIX는 구독 상품에 따라 동시 접속이 가능한 프로필의 수가 제한된다.
 - 동시접속인원 수 파악을 위해 프로필에 필요한 애트리뷰트를 추가한다.
- 1개의 작품은 1개의 엔티티로 표현된다.
 - 작품마다 가지는 여러 에피소드를 하나의 엔티티로 표현할 수 없으므로 수정이 필요하다.
- 프로필과 작품은 시청 관계를 가진다.
 - 프로필은 작품이 아닌 작품이 포함하는 영상과 시청 관계를 가진다.
- 작품에 종속된 작품 데이터 엔티티가 있다.
 - 작품이 갖는 데이터들을 따로 엔티티로 분리해야 할 이유가 없다.

- 작품 데이터는 다치 애트리뷰트인 자막, 더빙을 가진다.
 - 작품마다 지원하는 더빙과 자막에 대한 정보는 동일한 도메인을 가진다. 따라서 엔티티 타입을 새로 생성해 작품 엔티티와 자막과 더빙 관계를 설정한다.
- 작품 데이터는 재생시간 애트리뷰트를 가진다.
 - 재생시간 데이터는 작품이 아니라 작품이 가지는 영상들이 가지므로 수정이 필요하다.
- 출연진과 감독은 사람이라는 카테고리에 포함된다.
 - 사람 엔티티 타입과 역할 엔티티 타입을 만들어 사람마다 참여하는 콘텐츠에서의 역할을 가지게 한다.
- 장르는 다치 애트리뷰트이다.
 - 장르 엔티티를 새로 만들고 작품과의 관계를 설정한다.
- 트랜잭션 시 편의를 위해 결제일자 애트리뷰트를 만료일자 애트리뷰트로 수정한다.
- 찜 관계가 낱자 애트리뷰트를 직접 가지도록 수정한다.
- 기존의 시청 관계가 가지던 애트리뷰트들을 없애고, 시청기록을 바탕으로 한 콘텐츠 추천 기능의 구현을 위해 시작시각, 종료시각, 중단시간 애트리뷰트를 추가한다.
- NETFLIX에는 작품이 수상한 상에 대한 정보가 명시되어 있지 않으므로 상 엔티티 타입을 삭제한다.
- 평가 애트리뷰트를 작품과 연결한다.

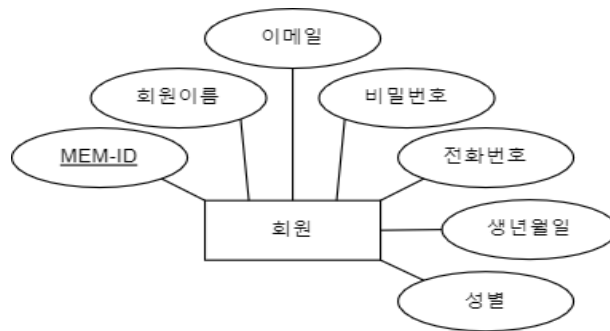
3.2 최종 개념적 설계

수정된 최종 ER모델에 대한 분석은 다음과 같다.

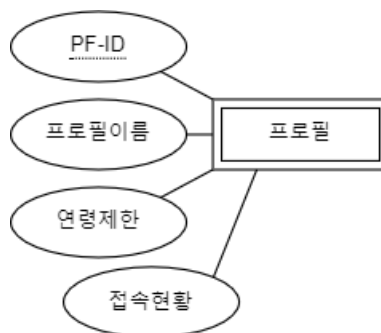
3.2.1 엔티티 타입 및 애트리뷰트 식별



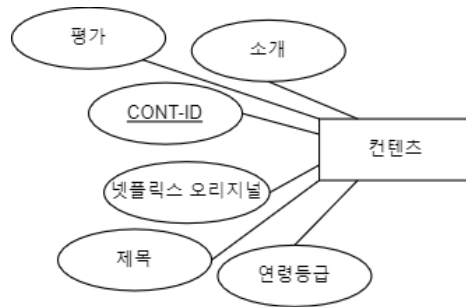
구독상품 엔티티 타입은 PLAN_ID(구독 상품 ID), 상품 이름, 가격, 최대 동시 접속 가능 수, 최대 화질 등 5개의 애트리뷰트를 가지며, PLAN_ID를 기본키로 가진다.



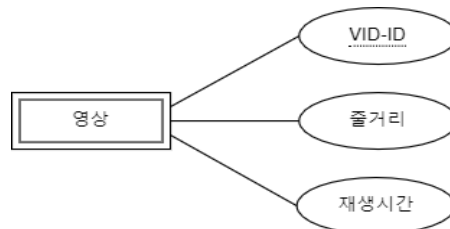
회원 엔티티 타입은 MEM_ID(회원 ID), 회원 이름, 이메일, 비밀번호, 전화번호, 생년월일, 생일 등 7개의 애트리뷰트를 가지며, MEM_ID를 이용해서 고유하게 식별한다.



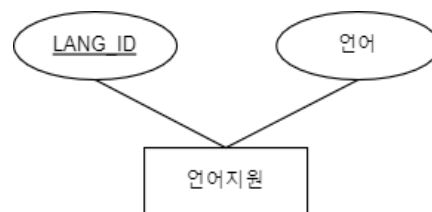
프로필 엔티티 타입은 약한 엔티티 타입으로서, PF_ID(프로필 ID), 프로필 이름, 연령제한, 접속현황 등 4개의 애트리뷰트를 가지며, PF_ID는 부분 키이므로 점선의 밑줄을 그어 표시하였다.



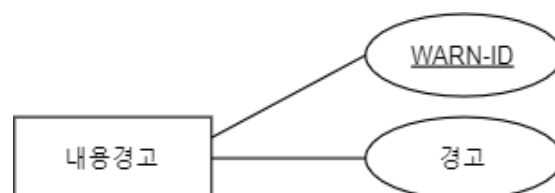
컨텐츠 엔티티 타입은 CONT_ID(콘텐츠 ID), 제목, 소개, 연령등급, 평가, NETFLIX 오리지널 등 6개의 애트리뷰트를 가진다. 이때 NETFLIX 오리지널 애트리뷰트는 NETFLIX가 직접 제작한 작품들을 의미하며, 이 중 CONT_ID를 통해 이들을 고유하게 식별한다.



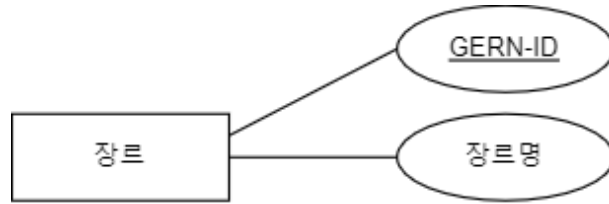
영상 엔티티 타입은 약한 엔티티 타입으로, VID_ID(비디오 ID), 줄거리, 재생시간을 애트리뷰트로 가지며, 이때 재생시간은 영상의 총 재생시간을 말한다. 이중에서 VID_ID는 부분 키이므로 점선의 밑줄을 그어 표시한다.



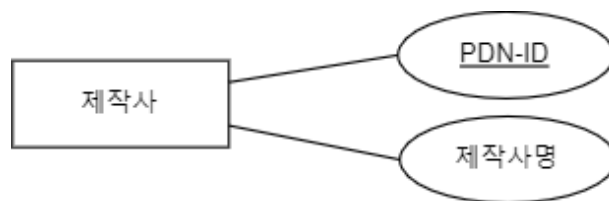
언어지원 엔티티 타입은 LANG_ID(언어 ID), 언어 등 두 개의 애트리뷰트를 가지고, LANG_ID를 이용해서 고유하게 식별한다.



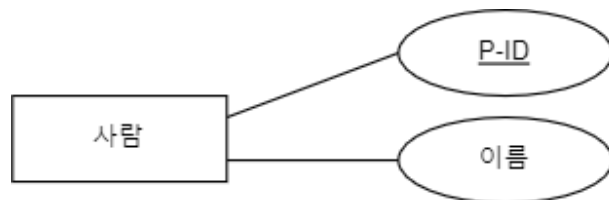
내용 경고 엔티티 타입은 WARN_ID(경고 ID), 경고 등 두 개의 애트리뷰트를 가지며, WARN_ID를 통해 고유하게 식별된다.



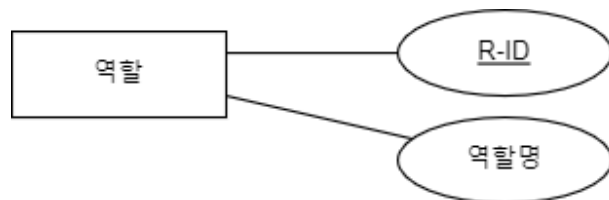
장르 엔티티 타입은 GERN_ID(장르 ID), 장르명 등 두 개의 애트리뷰트를 가지며, 장르 엔티티는 GERN_ID를 통해 고유하게 식별된다.



제작사 엔티티 타입은 PDN_ID(제작사 ID), 제작사명 등 두 개의 애트리뷰트를 가지고, PDN_ID를 통해 고유하게 식별된다.

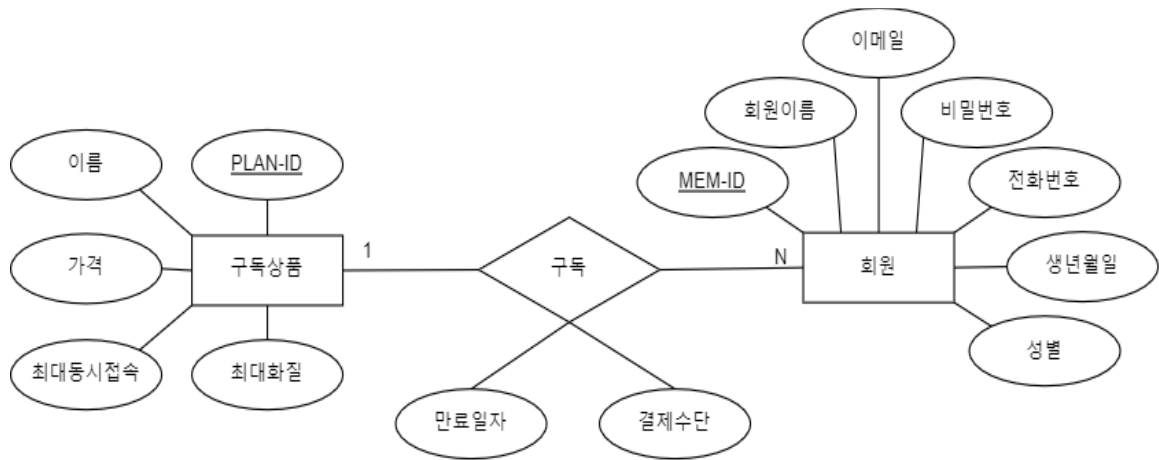


사람 엔티티 타입은 P_ID(사람 ID), 이름 등 두 개의 애트리뷰트를 가지고, P_ID를 통해 고유하게 식별된다.

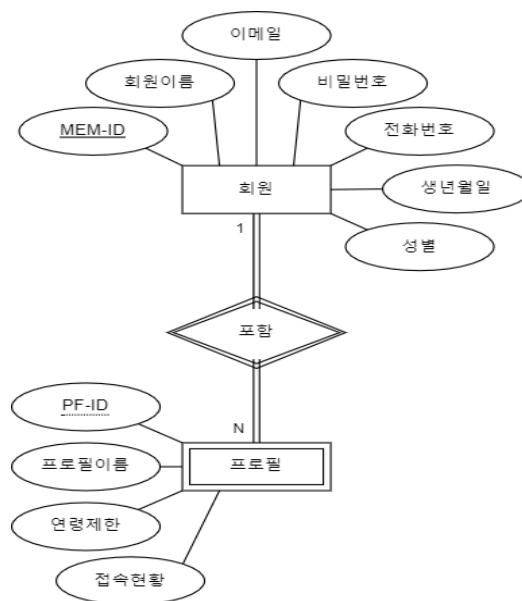


역할 엔티티 타입은 R_ID(역할 ID), 역할명 등 두 개의 애트리뷰트를 가지며, R_ID를 통해 고유하게 식별된다.

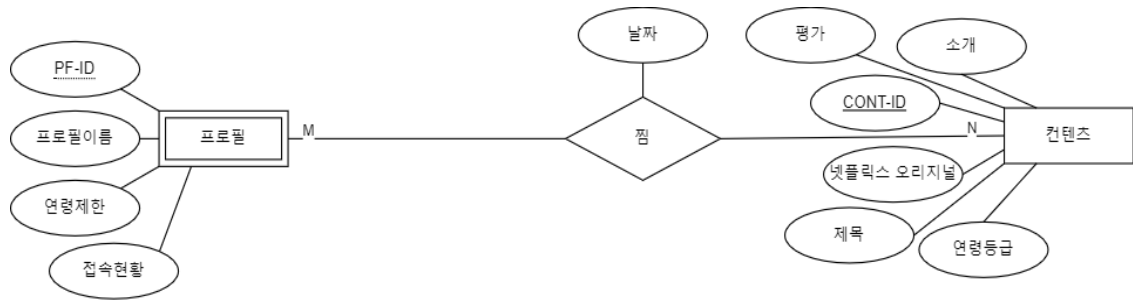
3.2.2 관계 및 애트리뷰트 식별



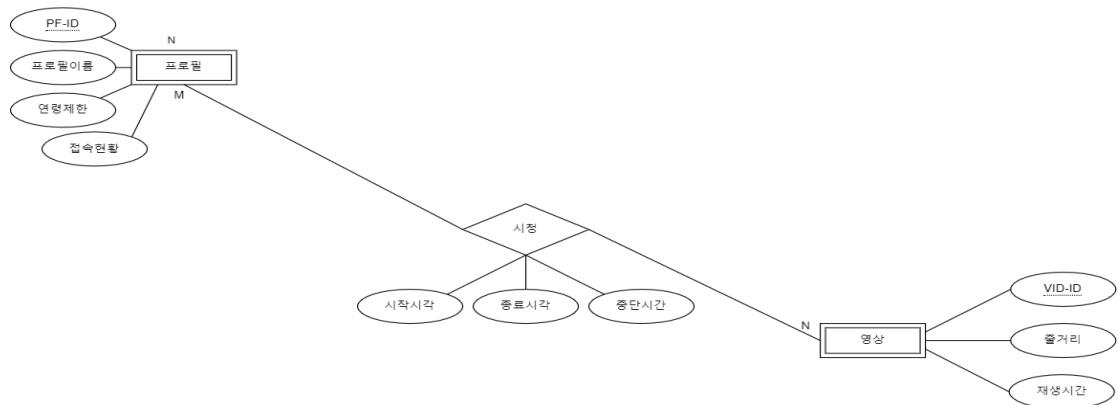
구독 상품 엔티티와 회원 엔티티는 구독 관계 타입으로 연결된다. 회원 한 명은 한 개의 구독상품을 구독할 수 있고, 구독상품 한 개는 여러 명의 회원이 구독할 수 있으므로 구독 상품 엔티티와 회원 엔티티는 1 : N 관계이다. 또한, 한 구독 상품은 회원을 아무도 가지지 않을 수 있고, 회원의 경우 만료일자가 지난 후 다시 결제를 안 할 경우 구독 상품을 가지지 않을 수 있으므로 구독 상품 엔티티와 회원 엔티티는 구독 관계타입에 부분 참여한다.



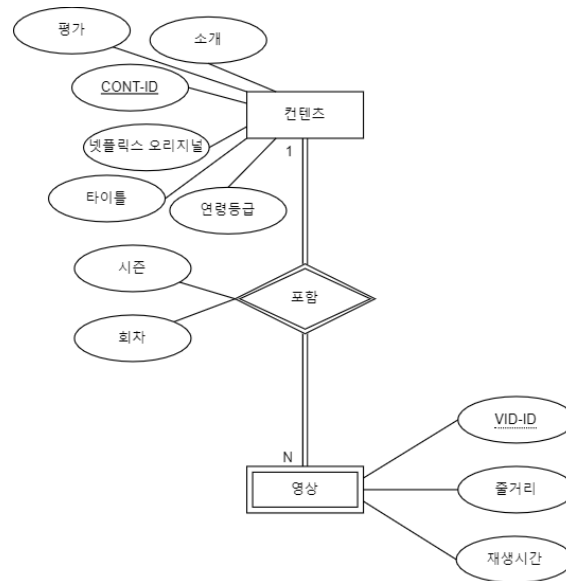
회원 엔티티와 프로필 엔티티는 포함 관계 타입으로 연결되어 있다. 회원 한 명은 여러 개(최대 5개)의 프로필을 가질 수 있고, 프로필 하나는 한 명의 회원만 가진다. 이 때문에 회원 엔티티와 프로필 엔티티는 포함관계에 대해 1 : N 관계를 가지고, 회원은 최소 한 개의 프로필을 가져야 하고, 프로필도 한 명의 회원을 가져야 하므로 회원 엔티티와 프로필 엔티티는 포함 관계에 대해 전체 참여이다.



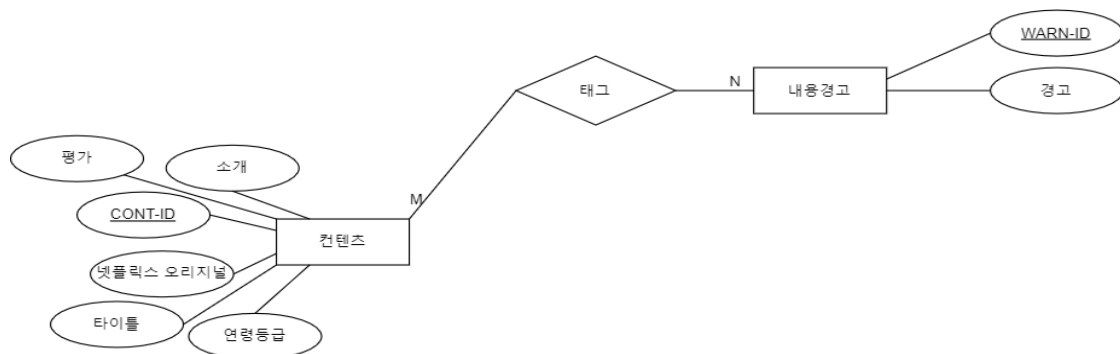
프로필 엔티티와 컨텐츠 엔티티는 찜 관계 타입으로 연결되어 있다. 프로필 한 개는 여러 컨텐츠를 찜 할 수 있고, 하나의 컨텐츠는 여러 개의 프로필에 찜 당할 수 있다. 그러므로 프로필 엔티티와 컨텐츠 엔티티는 M : N 관계이다. 그리고 이 관계는 날짜 애트리뷰트가 필요하다. 이 관계에서 프로필은 컨텐츠를 아예 찜을 하지 않을 수 있고, 컨텐츠 중에는 프로필에 찜 당하지 않은 작품이 존재할 수 있다. 이 때문에 프로필 엔티티와 컨텐츠 엔티티는 찜 관계에 대해 부분 참여한다.



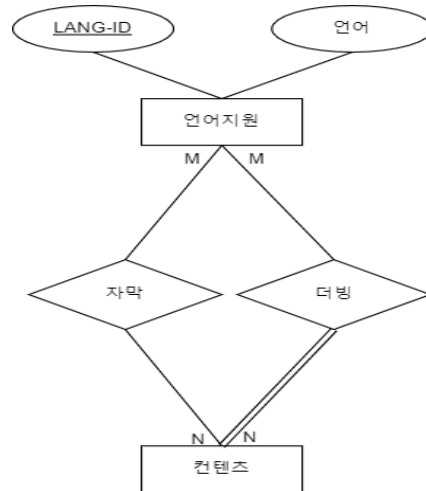
프로필 엔티티와 영상 엔티티는 시청 관계로 연결되어 있다. 프로필 한 개는 여러 개의 영상을 시청 할 수 있고, 한 개의 영상은 여러 개의 프로필이 시청 할 수 있다. 그래서 프로필 엔티티와 영상 엔티티는 M : N 관계를 형성한다. 이 때 프로필은 영상을 아예 보지 않을 수 있고, 프로필들이 아예 보지 않은 영상이 존재할 수 있다. 그렇기 때문에 프로필과 영상은 시청 관계에 대해 부분 참여하고, 시청 관계는 시작시각, 종료시각, 중단시간을 애트리뷰트로 갖는다.



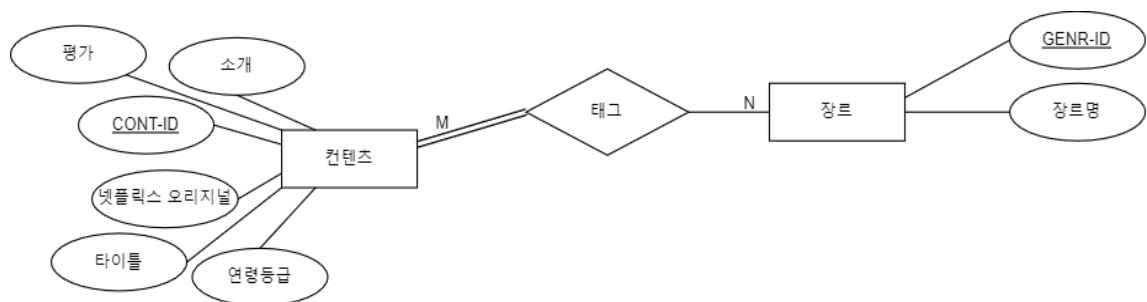
컨텐츠 엔티티와 영상 엔티티는 포함 관계로 연결된다. 하나의 컨텐츠는 여러 개의 영상을 무조건 포함해야 하고, 하나의 영상은 하나의 컨텐츠를 무조건 포함해야 한다. 이 때문에 컨텐츠 엔티티와 영상 엔티티는 1 : N 관계이고, 각각 포함 관계에 대해 전체참여이다. 또 포함 관계는 시즌 애트리뷰트와 회차 애트리뷰트를 가진다.



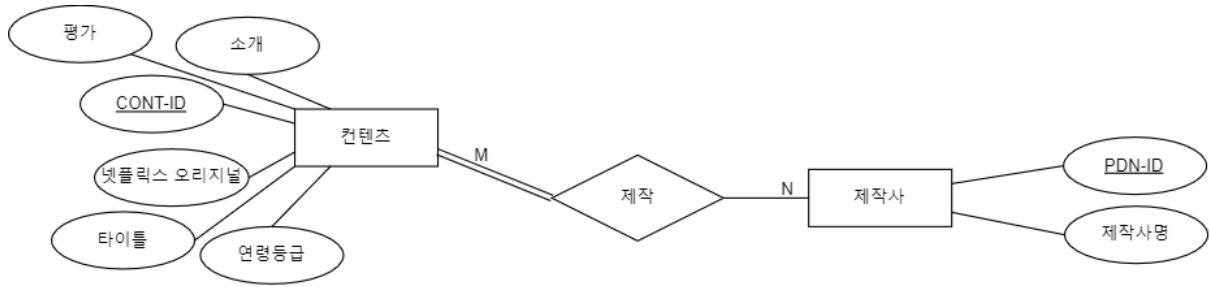
컨텐츠 엔티티와 내용경고 엔티티는 태그 관계로 연결되어 있다. 한 개의 컨텐츠는 여러 개의 내용 경고를 태그할 수 있고, 한 개의 내용경고는 여러 개의 컨텐츠에 태그될 수 있다. 그러므로 컨텐츠 엔티티와 내용경고 엔티티는 태그 관계에 대해 M : N 관계를 형성한다. 그리고 하나의 컨텐츠는 내용경고를 태그하지 않을 수 있고, 한 개의 내용경고는 컨텐츠에 태그되지 않을 수 있다. 따라서 컨텐츠 엔티티와 내용경고 엔티티는 태그 관계에 대해 부분 참여한다.



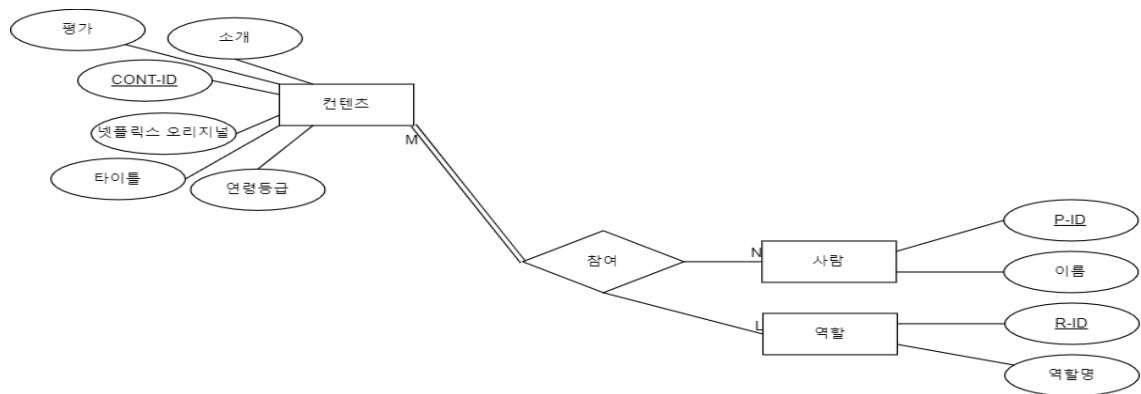
언어 지원 엔티티와 컨텐츠 엔티티는 두 개의 관계 타입으로 연결된다. 먼저 자막 관계 타입에서, 한 개의 언어는 여러 개의 컨텐츠에 자막이 될 수 있고, 한 개의 컨텐츠는 여러 개의 언어를 자막으로 가질 수 있다. 그래서 언어 지원 엔티티와 컨텐츠 엔티티의 자막 관계에서는 M : N 관계이다. 그리고 하나의 컨텐츠에 자막이 없을 수 있고, 언어 중에는 컨텐츠 자막으로 지원하지 않는 언어가 있을 수 있다. 그러므로 언어지원 엔티티와 컨텐츠 엔티티는 자막 관계에서 부분 참여이다. 더빙 관계에서는, 한 개의 언어는 여러 개의 컨텐츠에 더빙으로 지원할 수 있다. 또 한 개의 컨텐츠는 여러 더빙이 있을 수 있다. 그러므로 언어 지원 엔티티와 컨텐츠 엔티티는 M : N 관계이고, 언어들 중에는 컨텐츠에 더빙으로 지원하지 않는 언어가 존재할 수 있지만, 한 컨텐츠는 무조건 한 개 이상의 더빙이 존재한다. 그러므로 언어 지원 엔티티는 더빙 관계에 대해 부분 참여이고, 컨텐츠 엔티티는 더빙 관계에 대해 전체 참여이다.



컨텐츠 엔티티와 장르 엔티티는 태그 관계로 연결되어 있다. 하나의 컨텐츠는 여러 개의 장르를 태그할 수 있고, 한 개의 장르는 여러 개의 컨텐츠가 태그할 수 있다. 따라서 컨텐츠 엔티티와 장르 엔티티는 태그 관계에 대해 M : N 관계이다. 또한 한 개의 컨텐츠는 한 개 이상의 장르를 무조건 태그해야 하고, 컨텐츠가 태그하지 않은 장르가 존재할 수 있다. 따라서 컨텐츠 엔티티는 태그 관계에 대해 전체 참여하고, 장르 엔티티는 태그 관계에 대해 부분 참여한다.

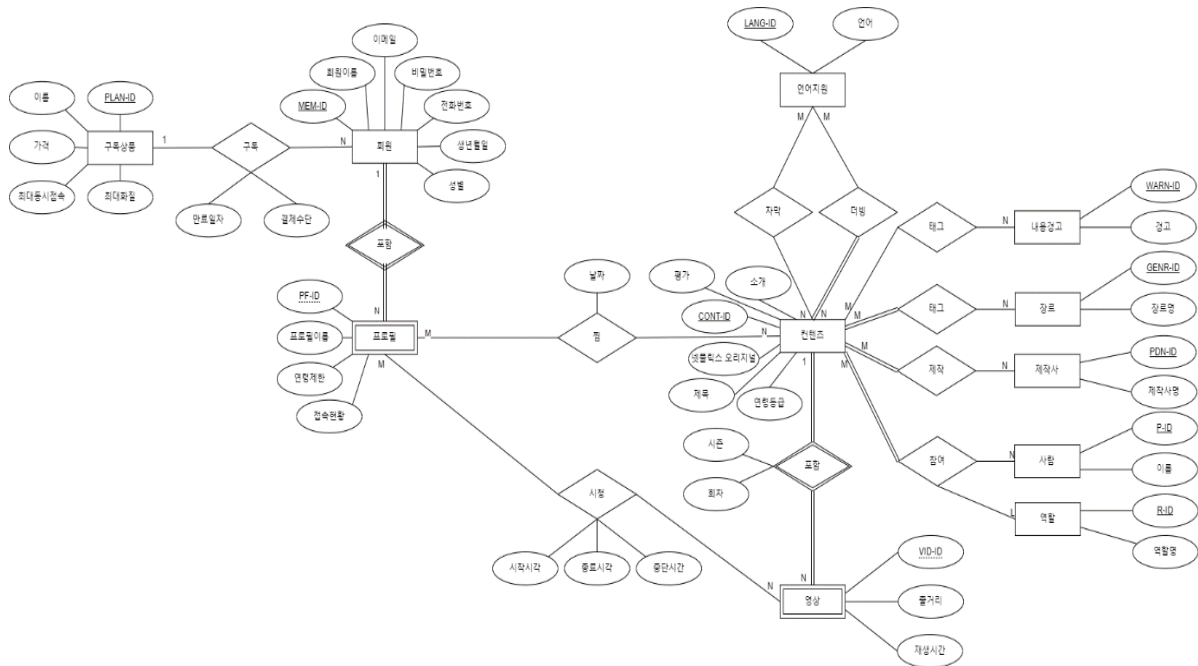


컨텐츠 엔티티와 제작사 엔티티는 제작 관계로 연결되어 있다. 한 개의 컨텐츠는 여러 개의 제작사가 제작할 수 있고, 한 개의 제작사는 여러 개의 컨텐츠를 제작할 수 있다. 그러므로 컨텐츠 엔티티와 제작사 엔티티는 제작 관계에 대해 M : N 카디널리티를 갖는다. 컨텐츠 엔티티는 무조건 한 개 이상의 제작사가 제작해야 하고, 제작사 중에는 컨텐츠를 제작하지 않은 제작사가 존재할 수 있다. 때문에 컨텐츠는 제작 관계에 대해 전체 참여하고, 제작사는 제작 관계에 대해 부분 참여한다.



컨텐츠 엔티티와 사람 엔티티와 역할 엔티티 사이에는 참여 관계가 존재한다. 한 명의 사람은 여러 개의 역할을 가지고 여러 컨텐츠에 참여하고, 한 컨텐츠는 여러 명의 사람이 여러 개의 역할로 참여한다. 때문에 컨텐츠 엔티티와 사람 엔티티와 역할 엔티티는 L : M : N의 카디널리티를 갖는다. 이때 한 컨텐츠에는 적어도 한 사람이 한 개의 역할로 참여해야 하며, 한 사람은 컨텐츠에 한 역할로 참여하지 않을 수 있다. 때문에 컨텐츠 엔티티는 참여 관계에 대해 전체 참여하고, 사람 엔티티와 역할 엔티티는 참여 관계에 대해 부분 참여한다.

전체 ER 다이어그램은 다음과 같다.



4. 논리적 설계

4.1 논리적 설계 1단계(정규 엔티티타입과 단일 값 애트리뷰트)

해당하는 정규 엔티티 타입은 구독 상품 엔티티, 회원 엔티티, 콘텐츠 엔티티, 내용 경고 엔티티, 장르 엔티티, 제작사 엔티티, 사람 엔티티, 역할 엔티티, 언어지원 엔티티가 존재한다.

- 구독상품(PLAN-ID, 상품명, 가격, 최대동시접속, 최대화질)
- 회원(MEM-ID, 회원이름, 이메일, 비밀번호, 전화번호, 생년월일, 성별)
- 콘텐츠(CONT-ID, 타이틀, 소개, 연령등급, 평가, 넷플릭스 오리지널)
- 내용경고(WARN-ID, 경고내용)
- 장르(GENR-ID, 장르명)
- 제작사(PDN-ID, 제작사명)
- 사람(P-ID, 이름)
- 역할(R-ID, 역할명)
- 언어지원(LANG_ID, 언어)

4.2 논리적 설계 2단계(약한 엔티티 타입과 단일 값 애트리뷰트)

프로필 엔티티와 영상 엔티티가 약한 엔티티 타입이다.

- 프로필(MEM-ID, PF-ID, 프로필이름, 연령제한, 접속현황)
- 영상(CONT-ID, VID-ID, 줄거리, 재생시간, 시즌, 회차)

4.3 논리적 설계 4단계(정규 2진 1 : N 관계 타입)

정규 2진 1 : N 카디널리티를 가지는 관계는 회원과 구독 상품 간의 관계인 구독 관계이다. 1측의 엔티티 타입에 대응되는 것은 구독 상품 엔티티이고, N측의 엔티티 타입에 대응되는 것은 회원 엔티티이므로 구독 상품 엔티티의 기본키를 회원 엔티티의 외래키로 포함시킨다. 이때 회원 엔티티에 구독 상품 릴레이션의 기본 키를 가지는 외래 키 PLAN_ID를 추가한다.

- 회원(MEM-ID, 회원이름, 이메일, 비밀번호, 전화번호, 생년월일, 성별, PLAN-ID, 만료일자, 결제수단)

4.4 논리적 설계 5단계(2진 M : N 관계 타입)

해당되는 관계는 프로필 엔티티와 작품 엔티티의 관계인 찜 관계, 프로필 엔티티와 영상 엔티티의 관계인 시청기록 관계, 콘텐츠 엔티티와 내용경고 엔티티의 관계인 경고태그 관계, 콘텐츠와 장르의 관계인 장르태그 관계, 콘텐츠와 제작사의 관계인 제작 관계, 콘텐츠와 언어 지원의 관계인 자막 관계, 콘텐츠와 언어지원의 관계인 더빙관계가 존재한다. 각각 대응되는 엔티티 타입의 기본 키를 외래 키로 포함시키고, 이 들을 조합해 기본 키를 형성하였다. 관계 애트리뷰트들을 릴레이션에 포함시켰다.

- 찜 목록(MEM-ID, PF-ID, CONT-ID, 날짜)
- 시청기록(MEM-ID, PF-ID, CONT-ID, VID-ID, 시작시각, 종료시각, 중단시간)
- 경고태그(CONT-ID, WARN-ID)
- 장르태그(CONT-ID, GERN-ID)
- 제작(CONT-ID, PDN-ID)
- 자막(CONT-ID, LANG-ID)
- 더빙(CONT-ID, LANG-ID)

4.5 논리적 설계 단계 6(3진 이상의 관계 타입)

해당되는 관계 타입은 콘텐츠 엔티티 타입과 사람 엔티티 타입, 역할 엔티티 타입의 관계인 참여 관계이다. 참여 릴레이션의 기본 키는 각각의 엔티티 타입의 기본 키를 외래 키로 포함시켜 기본 키를 형성하였다.

- 참여(CONT-ID, P-ID, R-ID)

4.6 최종 논리 스키마

위 7개의 단계를 걸쳐 최종적으로 아래와 같은 릴레이션 집합을 형성하였고, 총 19개의 릴레이션을 사상하였다.

- 구독상품(PLAN-ID, 상품명, 가격, 최대동시접속, 최대화질)
- 회원(MEM-ID, 회원이름, 이메일, 비밀번호, 전화번호, 생년월일, 성별, PLAN-ID, 만료일자, 결제수단)
- 콘텐츠(CONT-ID, 타이틀, 소개, 연령등급, 평가, 넷플릭스 오리지널)
- 내용경고(WARN-ID, 경고내용)
- 장르(GERN-ID, 장르명)
- 제작사(PDN-ID, 제작사명)
- 사람(P-ID, 이름)
- 역할(R-ID, 역할명)
- 언어지원(LANG-ID, 언어)
- 프로필(MEM-ID, PF-ID, 프로필이름, 연령제한, 접속현황)
- 영상(CONT-ID, VID-ID, 줄거리, 재생시간, 시즌, 회차)
- 찜 목록(MEM-ID, PF-ID, CONT-ID, 날짜)
- 시청기록(MEM-ID, PF-ID, CONT-ID, VID-ID, 시작시각, 종료시각, 중단시간)
- 경고태그(CONT-ID, WARN-ID)
- 장르태그(CONT-ID, GERN-ID)
- 제작(CONT-ID, PDN-ID)

- 자막(CONT-ID, LANG-ID)
- 더빙(CONT-ID, LANG-ID)
- 참여(CONT-ID, P-ID, R-ID)

5. 물리적 설계

논리적 구조로 표현된 데이터를 물리적 구조로 변경한다.

5.1 테이블 기술서

테이블명	논리명	물리명	내용	타입	기본키	외래키	UNIQUE	NOT NULL	기타
구독상품 / PLAN	구독상품 ID	PLAN_ID	각 구독 상품을 고유하게 식별하기 위한 번호(1)	NUMBER	0		0	0	
	구독상품 이름	PLAN	구독 상품 이름(BASIC)	VARCHAR2(30)			0	0	
	구독 상품 가격	PRICE	각 구독 상품 가격(9500)	NUMBER				0	
	최대동시접속	MAXCONNECT	각 구독 상품마다 동시에 접속할 수 있는 수(1)	NUMBER				0	
	최대 화질	MAXQUALITY	구독 상품 최대 화질(4K+HDR)	VARCHAR2(30)				0	
회원 / MEMBER	회원 ID	MEM_ID	회원들을 고유하게 식별하기 위한 번호(1)	NUMBER	0		0	0	
	회원 이름	NAME	회원 이름(김철수)	VARCHAR2(30)				0	
	회원 이메일	EMAIL	회원 이메일(database1@kpu.ac.kr)	VARCHAR2(30)			0	0	
	비밀번호	PASSWD	비밀번호(teamproject1)	VARCHAR2(30)				0	
	전화번호	PHONENUM	전화번호(010-0123-4567)	VARCHAR2(30)			0		
	생년월일	BIRTH	생년월일(20010122)	DATE					
	성별	SEX	성별(M)	CHAR				0	CHECK(SEX IN ('M', 'F'))
	구독 상품 ID	PLAN_ID	구독 상품 ID(1)	NUMBER		0			
	가입 날짜	DUEDATE	가입날짜(20211121)	DATE					
	결제 방법	PAYMETHOD	결제방법(무통장입금)	VARCHAR2(30)					CHECK(PAYMETHOD IN ('신용카드', '무통장입금', '통신사결제', '간편결제'))
콘텐츠 / CONTENT	콘텐츠 ID	CONT_ID	콘텐츠를 고유하게 식별하기 위한 번호(1)	NUMBER	0		0	0	
	제목	TITLE	작품 제목(마이네임)	VARCHAR2(40)				0	
	소개	INTRO	작품소개(대한민국을 움직이는 권력자들의 하수인 상구)	VARCHAR2(250)					
	연령 등급	CONT_RATING	연령등급(18)	NUMBER				0	CHECK(CONT_RATING IN (0, 7, 12, 15, 18, 19))
	평가	USER_RATING	평가(3.9)	NUMBER					CHECK(USER_RATING BETWEEN 0 AND 5)
	넷플릭스 오리지널	ORIGINAL	넷플릭스에서 만든 작품인지 아닌지 확인(0)	NUMBER				0	CHECK(ORIGINAL IN(0, 1))
내용경고 / WARNING	경고 ID	WARN_ID	내용경고를 고유하게 식별하기 위한 번호(1)	NUMBER	0		0	0	
	내용	CONTENT	내용경고(폭력성)	VARCHAR2(30)				0	

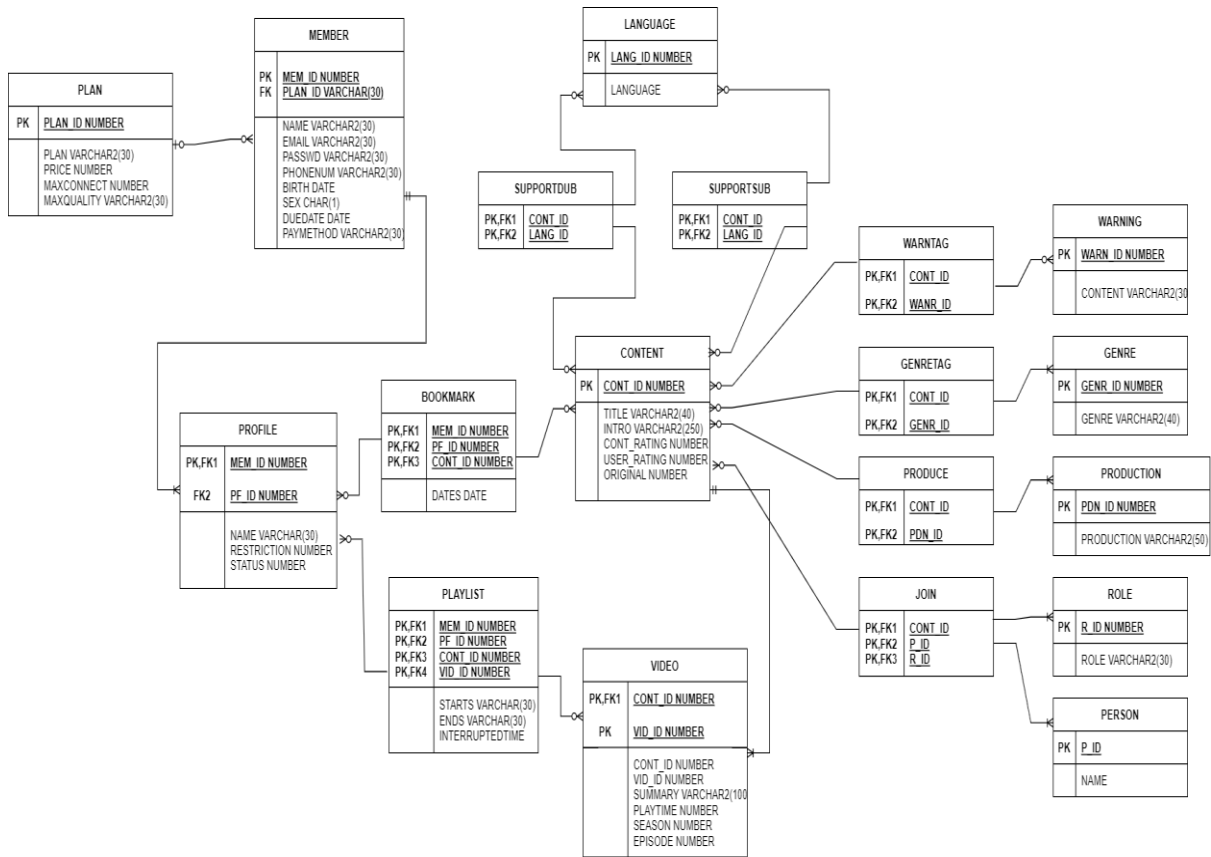
Windows 정품 인증

내용경고 / WARNING	경고 ID	WARN_ID	내용경고들을 고유하게 식별하기 위한 번호(1)	NUMBER	0		0	0	
	내용	CONTENT	내용경고(목록상)	VARCHAR2(30)			0	0	
장르 / GENRE	장르 ID	GENR_ID	장르들을 고유하게 식별하기 위한 번호(1)	NUMBER	0		0	0	
	장르	GENRE	장르(한국 드라마)	VARCHAR2(40)			0	0	
제작사 / PRODUCTION	제작사 ID	PDN_ID	제작사들을 고유하게 식별하기 위한 번호(1)	NUMBER	0		0	0	
	제작사 이름	PRODUCTION	제작사(마블)	VARCHAR2(50)			0	0	
역할 / ROLE	역할 ID	R_ID	역할들을 고유하게 식별하기 위한 번호(1)	NUMBER	0		0	0	
	역할명	ROLE	역할(배우)	VARCHAR2(30)			0	0	
언어 / LANGUAGE	언어 ID	LANG_ID	언어들을 고유하게 식별하기 위한 번호(1)	NUMBER	0		0	0	
	언어	LANGUAGE	언어(한국어)	VARCHAR2(30)			0	0	
프로필 / PROFILE	회원 ID	MEM_ID	회원 ID(1)	NUMBER	0	0	0	0	
	프로필 ID	PF_ID	프로필들을 고유하게 식별하기 위한 번호	NUMBER	0		0	0	CHECK (PF_ID IN (1, 2, 3, 4, 5))
	프로필 이름	NAME	프로필(부영이)	VARCHAR2(30)				0	
	연령제한	RESTRICTION	연령제한(19)	NUMBER					DEFAULT 19 CHECK(RESTRICTION IN(7,12,15,18,19))
	접속현황	STATUS	접속현황(0)	NUMBER					DEFAULT 0 CHECK (STATUS IN (0, 1))
영상 / VIDEO	컨텐츠 ID	CONT_ID	컨텐츠 ID(1)	NUMBER	0	0	0	0	
	영상 ID	VID_ID	영상들을 고유하게 식별하기 위한 번호(1)	NUMBER	0		0	0	
	줄거리	SUMMARY	줄거리(최악의 상황이였다.)	VARCHAR2(100)					
	재생시간	PLAYTIME	재생시간(54)	NUMBER				0	
	시즌	SEASON	시즌(3)	NUMBER					DEFAULT NULL
	회차	EPISODE	회차(7)	NUMBER					DEFAULT NULL

참목록 / BOOKMARK	회원 ID	MEM_ID	회원 ID(3)	NUMBER	0	0	0	0	
	프로필 ID	PF_ID	프로필 ID(1)	NUMBER	0	0	0	0	
	컨텐츠 ID	CONT_ID	컨텐츠 ID(1)	NUMBER	0	0	0	0	
	날짜	DATES	날짜(21-09-20)	DATE				0	
시정기록 / PLAYLIST	회원 ID	MEM_ID	회원 ID(3)	NUMBER	0	0	0	0	
	프로필 ID	PF_ID	프로필 ID(1)	NUMBER	0	0	0	0	
	컨텐츠 ID	CONT_ID	컨텐츠 ID(1)	NUMBER	0	0	0	0	
	영상 ID	VID_ID	영상 ID(1)	NUMBER	0	0	0	0	
	시작 시간	STARTS	시작시간(2021-08-26 16:14)	VARCHAR2(30)				0	
	종료 시간	ENDS	종료시간(2021-08-26 17:20)	VARCHAR2(30)				0	
	중단 시간	INTERRUPTEDTIME	중단시간(54)	NUMBER				0	
경고태그 / WARNING	컨텐츠 ID	CONT_ID	컨텐츠 ID(1)	NUMBER	0	0	0	0	
	경고 ID	WARN_ID	경고 ID(3)	NUMBER	0	0	0	0	
장르태그 / GENRETAG	컨텐츠 ID	CONT_ID	컨텐츠 ID(1)	NUMBER	0	0	0	0	
	장르 ID	GENR_ID	장르 ID(1)	NUMBER	0	0	0	0	
제작 / PRODUCE	컨텐츠 ID	CONT_ID	컨텐츠 ID(1)	NUMBER	0	0	0	0	
	제작사 ID	PDN_ID	제작사 ID(1)	NUMBER	0	0	0	0	
사람 / PERSON	사람 ID	P_ID	사람 ID	NUMBER	0		0	0	
	사람 이름	NAME	사람 이름(한소희)	VARCHAR2(30)				0	

참여 / JOIN	컨텐츠 ID	CONT_ID	컨텐츠 ID(1)	NUMBER	0	0	0	0	
	사람 ID	P_ID	사람 ID(1)	NUMBER	0	0	0	0	
	역할 ID	R_ID	역할 ID(2)	NUMBER	0	0	0	0	
더빙 지원 / SUPPORTDUB	컨텐츠 ID	CONT_ID	컨텐츠 ID(1)	NUMBER	0	0	0	0	
	언어 ID	LANG_ID	언어 ID(1)	NUMBER	0	0	0	0	
자막 지원 / SUPPORTSUB	컨텐츠 ID	CONT_ID	컨텐츠 ID(1)	NUMBER	0	0	0	0	
	언어 ID	LANG_ID	언어 ID(2)	NUMBER	0	0	0	0	

5.2 테이블 관계 다이어그램



6. 트랜잭션 구현

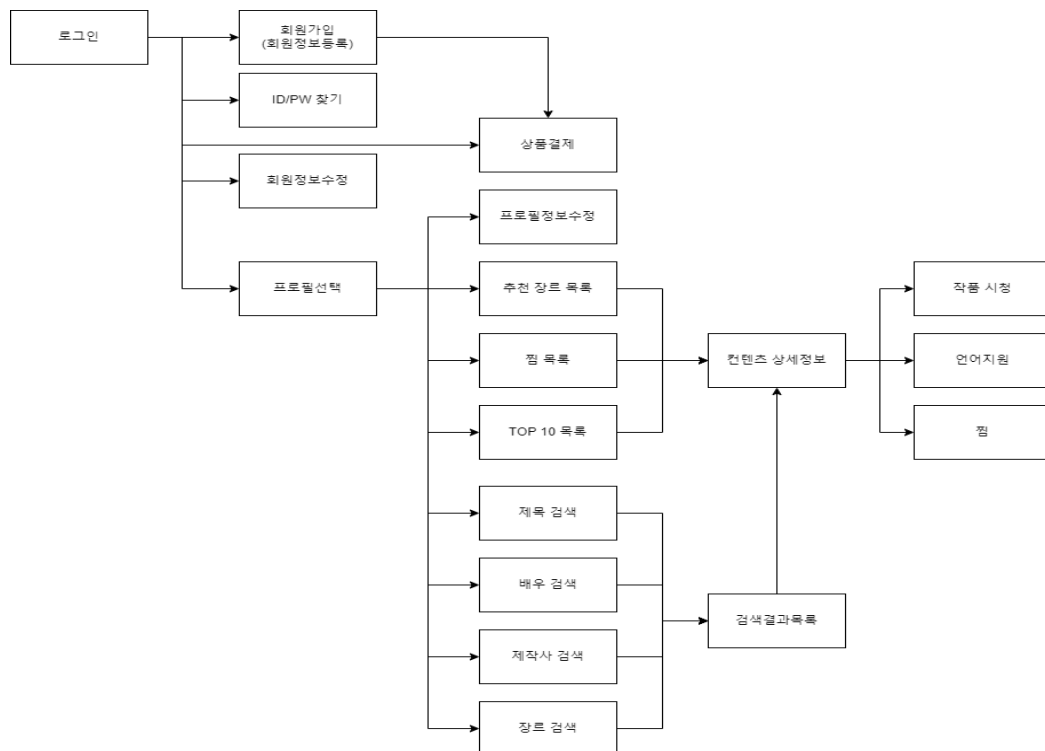
6.1 주요 기능

NETFLIX는 아래와 같은 기능들을 가진다.

- 로그인
- 회원가입(회원정보등록)
- 상품결제
- 회원정보수정
- ID/PW 찾기
- 프로필선택
- 프로필 정보 수정

- TOP10 추천 목록
- 제목 검색
- 배우 검색
- 제작사 검색
- 장르 검색
- 검색 결과 목록
- 추천 장르 제공
- 찜 목록
- 콘텐츠 상세정보
- 작품 시청
- 언어지원
- 찜
- 찜 목록 토대로 작품 추천

6.2 화면 흐름도



6.3 주요기능 명세

- 로그인

이메일과 패스워드를 확인하여 사용자의 접속을 결정한다.

1. 회원 릴레이션에서 입력된 이메일과 비밀번호와 일치하는 값을 가진 튜플을 검색한다.
2. 튜플이 검색되면 해당 회원으로의 접속을 허용한다.

- 회원가입(회원정보등록)

새로운 회원의 정보를 등록한다.

1. 사용자는 절차에 따라 필요한 정보를 입력한다.
2. 회원 릴레이션에 입력된 정보를 포함한 튜플을 삽입한다.

- 상품결제

-구독상품정보-

모든 구독상품에 대한 정보를 출력한다.

1. 구독상품 릴레이션의 모든 정보를 출력한다.

-결제정보입력-

구독 상품에 따른 결제를 한다.

1. 결제를 완료한 사용자가 선택한 구독상품과 결제수단 데이터를 입력받는다.
2. 회원 릴레이션에 있는 해당 회원의 튜플의 PLAN_ID, 결제수단, 만료일자 애트리뷰트의 값을 변경한다. 만료일자는 '데이터 입력을 받은 시각 + 30(한달)'이다.

- 회원정보수정

가입한 회원의 회원정보를 수정한다.

1. 사용자가 수정을 원하는 회원정보의 입력을 받는다.
2. 회원 릴레이션에 있는 해당 회원에 관한 튜플의 값을 수정한다.

- ID/PW 찾기

사용자의 회원 ID, PW를 찾는 기능이다.

- ID 찾기

1. 사용자가 회원정보로 등록한 전화번호를 입력받는다.
2. 회원 릴레이션에서 같은 값을 가지는 튜플의 ID(=email)를 반환한다.

- PW 찾기

1. 사용자가 회원정보로 등록한 ID(=email)을 입력받는다.
2. 회원 릴레이션에서 같은 값을 가지는 튜플의 비밀번호를 반환한다.

- 프로필선택

- 회원 별 프로필 이름

회원이 가지는 프로필들의 이름을 보인다.

1. 프로필 릴레이션에서 로그인한 회원의 MEM_ID 값을 가진 튜플을 찾아 해당 튜플이 가지는 NAME를 출력한다.

- 프로필 접속 현황 업데이트

회원 프로필의 접속 현황을 수정한다.

1. 프로필 릴레이션에서 회원과 프로필이 일치하는 튜플의 접속현황을 0(접속해제 시) 혹은 1(접속 시)로 수정한다.

- 프로필 정보수정

회원 각각의 프로필 정보를 수정한다.

- 프로필 이름 변경

1. 프로필 릴레이션에서 회원과 프로필이 일치하는 튜플의 이름을 변경한다.

- 프로필 시청제한연령 변경

1. 프로필 릴레이션에서 회원과 프로필이 일치하는 튜플의 시청제한연령을 변경한다.

• TOP 10 추천 목록

전체 사용자들이 최근 30일 이내에 많이 시청한 10개의 작품들을 순위별로 나타낸다. 이때 작품이 포함하는 영상 각각의 총 재생시간의 절반 이상을 시청한 작품만 시청한 작품으로 간주한다.

1. 시청 기록에서 시청시작 시간이 30일 이내인 기록들을 골라낸다.
2. 그 중 영상길이의 절반이상을 시청한 기록을 얻는다.
3. 해당 기록에서 같은 콘텐츠인 영상들의 개수를 세서 순위별로 나열한다.
4. 상위 10개의 작품을 TOP10 작품으로 선정한다.

• 제목 검색

제목으로 작품을 검색한다.

1. 콘텐츠 릴레이션에서 입력한 제목을 값으로 가지는 튜플의 콘텐츠 ID를 얻는다.
2. 해당하는 콘텐츠를 보인다.

• 배우 검색

특정 배우가 출연하는 작품을 검색한다.

1. 사람, 콘텐츠 그리고 둘 사이의 관계를 나타내는 릴레이션을 조인한다.
2. 입력한 배우의 이름을 값으로 가지는 튜플의 콘텐츠 ID와 타이틀을 얻는다.

• 제작사 검색

제작사명으로 작품을 검색한다.

1. 제작사, 콘텐츠 그리고 둘 사이의 관계를 나타내는 릴레이션을 조인한다.
2. 입력한 제작사명을 값으로 가지는 튜플의 콘텐츠 ID와 타이틀을 얻는다.

- 장르 검색

장르로 원하는 작품을 검색한다.

1. 회원이 입력한 장르와 장르 릴레이션에서 동일한 장르를 가지는 튜플들을 찾아낸다.
2. 찾아낸 장르와 동일한 장르를 가지는 콘텐츠 릴레이션의 튜플들의 콘텐츠를 보인다.

- 검색 결과 목록

특정 값으로 검색한 결과를 보인다.

1. 회원이 검색한 검색어와 대응되는 검색 방법을 통해 얻은 튜플들의 콘텐츠 목록을 보인다.

- 추천 장르 목록

사용자의 시청기록을 분석하여 사용자가 즐겨보는 장르의 작품을 목록으로 표시한다.

1. 특정 회원의 특정 프로필의 시청기록을 얻는다.
2. 그 중 최근 30일 내의 기록이면서, 영상시청시간이 영상길이의 절반을 넘는 기록을 얻는다.
3. 해당 기록들에 포함된 콘텐츠의 ID를 통해 30일 이내에 절반 이상 시청한 콘텐츠들이 가지는 장르를 알아낸다.
4. 해당 장르와 같은 콘텐츠를 검색해 목록에 표시한다.

- 찜 목록

사용자가 찜 한 작품들을 목록으로 표시한다.

1. 찜 릴레이션에서 특정 회원의 ID와 특정 프로필의 ID를 가지는 튜플을 검색한다.
2. 해당 튜플들의 콘텐츠 ID 값을 얻어 목록을 표시한다.

- 콘텐츠 상세정보

사용자가 선택한 콘텐츠의 상세정보를 보인다.

1. 입력한 콘텐츠 ID를 가지는 콘텐츠 릴레이션의 튜플을 검색한다.
2. 검색한 튜플이 가지는 해당 작품에 대한 작품정보들을 보인다.

- 작품 시청

선택한 콘텐츠의 영상을 보인다.

1. 해당하는 콘텐츠ID와 영상ID를 가지는 튜플을 검색해 결과를 보인다.

- 언어지원

콘텐츠가 지원하는 더빙, 자막언어를 보인다

1. 콘텐츠 릴레이션과 언어지원 릴레이션 사이의 관계를 나타내는 릴레이션에서 해당하는 콘텐츠ID와 관계된 튜플을 얻는다.
2. 얻은 튜플과 관계된 언어지원의 튜플에서 지원하는 언어에 관한 정보를 얻는다.

- 찜

작품을 찜 하는 기능을 한다.

1. 찜을 하면 찜 릴레이션에 회원 ID, 프로필 ID, 콘텐츠 ID를 키로 가지는 기록을 남긴다.

- 찜 목록 토대로 작품 추천

1. 특정 회원의 특정 프로필이 30일 이내에 찜한 작품들을 찜 릴레이션에서 찾는다.
2. 찾은 작품들의 장르를 장르 릴레이션에서 찾는다.
3. 그 장르들에 포함된 작품들의 작품들을 검색해 보인다.

6.4 뷰 및 인덱스

- 최근 30일 이내의 모든 시청 기록의 times

```
create view in30days(mem_id, pf_id, cont_id, vid_id, times)

as select mem_id, pf_id, cont_id, vid_id, round((to_date(ends,'yyyy-mm-dd hh24:mi') -
to_date(starts,'yyyy-mm-dd hh24:mi'))*24*60, 2)

from playlist

where add_months((to_date(starts, 'yyyy-mm-dd hh24:mi')),1) > sysdate;
```

- in30days 뷰에서 작품 전체 길이의 50퍼센트 이상 시청한 작품

```
create view in30days50percent

as select mem_id, pf_id, d.cont_id, d.vid_id

from in30days d, video v

where d.cont_id = v.cont_id and d.vid_id = v.vid_id and d.times > v.playtime * 0.5;
```

- in30days에서 작품 전체 길이의 50퍼센트 이상 시청한 작품들

```
create view is50percent

as select distinct v.cont_id, d.vid_id

from video v, in30days d

where v.vid_id = d.vid_id and d.times / v.playtime > 0.5 and d.mem_id = 10 and d.pf_id = 3;
```

- is50per에서의 작품들의 장르

```
create view whichgenre

as select t.genr_id

from content c, genretag t, is50percent p

where c.cont_id = t.cont_id and t.cont_id = p.cont_id;
```

6.5 트랜잭션 SQL문

- 로그인

```
SELECT MEM_ID  
  
FROM MEMBER  
  
WHERE EMAIL = 'database3@kpu.ac.kr' AND PASSWD = 'teamproject3';
```

- 회원가입(회원정보등록)

```
INSERT INTO MEMBER VALUES(11, '김태양', 'database11@kpu.ac.kr', 'teamproject11', '010-  
8953-7437', '20010122', 'M', NULL, ", ");
```

- 상품결제

-구독상품정보

```
SELECT *  
  
FROM PLAN;
```

-결제정보입력

```
UPDATE MEMBER  
  
SET PLAN_ID = 1, DUEDATE = SYSDATE + 30, PAYMETHOD = '신용카드'  
  
WHERE MEM_ID = 11;
```

- 회원정보수정

```
UPDATE MEMBER  
  
SET EMAIL = 'database111@kpu.ac.kr', PASSWD = 'teamproject111', PHONENUM = '010-  
0000-1111', BIRTH = '21010122', SEX = 'F';
```

- ID/PW 찾기

-아이디(이메일) 찾기

```
SELECT EMAIL
```

```
FROM MEMBER
```

```
WHERE PHONENUM = '010-8953-7437';
```

-비밀번호 찾기

```
SELECT PASSWD
```

```
FROM MEMBER
```

```
WHERE EMAIL = 'database3@kpu.ac.kr';
```

- 프로필선택

-회원 별 프로필 이름

```
SELECT NAME
```

```
FROM PROFILE
```

```
WHERE MEM_ID = 1;
```

-프로필 접속 현황 업데이트(접속 시)

```
UPDATE PROFILE
```

```
SET STATUS = 1
```

```
WHERE MEM_ID = 1 AND PF_ID = 2;
```

-프로필 접속 현황 업데이트(접속 해제 시)

```
UPDATE PROFILE
```

```
SET STATUS = 0
```

```
WHERE MEM_ID = 1 AND PF_ID = 1;
```

- 프로필 정보수정

-프로필 이름 변경

```
UPDATE PROFILE
```

```
SET NAME = '야야야야'
```

```
WHERE MEM_ID = '1' AND PF_ID = '1';
```

-프로필 시청제한연령 변경

```
UPDATE PROFILE
```

```
SET RESTRICTION = '15'
```

```
WHERE MEM_ID = '1' AND PF_ID = '1';
```

- TOP 10 추천 목록

- 컨텐츠별 최근 한달 내 50%이상 시청한 횟수

```
select cont_id, count(i.cont_id)
```

```
from in30days50percent i
```

```
group by cont_id
```

```
order by count(i.cont_id) desc;
```

- 제목 검색

```
select cont_id, title
```

```
from content
```

```
where title = '마이네임';
```


- 배우 검색

```
SELECT C.CONT_ID, TITLE  
  
FROM CONTENT C, PERSON P, JOIN J  
  
WHERE C.CONT_ID = J.CONT_ID AND P.P_ID = J.P_ID AND P.NAME = '이정재';
```

- 제작사 검색

```
SELECT C.CONT_ID, TITLE  
  
FROM CONTENT C, PRODUCTION P, PRODUCE R  
  
WHERE C.CONT_ID = R.CONT_ID AND P.PDN_ID = R.PDN_ID AND P.PRODUCTION LIKE '마  
블%';
```

- 장르 검색

```
SELECT DISTINCT TITLE  
  
FROM GENRE G, genretag T, CONTENT C  
  
WHERE G.GENR_ID = T.GENR_ID AND C.CONT_ID = T.CONT_ID AND GENRE LIKE '한국 드라  
마%';
```

- 추천 장르 목록

```
select distinct title  
  
from whichgenre w, genretag t, content c  
  
where t.cont_id = c.cont_id and w.genr_id = t.genr_id;
```

- 찜 목록

```
select title  
  
from bookmark b, content c  
  
where c.cont_id = b.cont_id and mem_id = 3 and pf_id = 1;
```

- 컨텐츠 상세정보

-작품 소개

```
select intro
```

```
from content
```

```
where title = '마이네임';
```

-작품 평가

```
select cont_rating
```

```
from content
```

```
where title = '마이네임';
```

-작품의 시즌

```
select DISTINCT season
```

```
from content c, video v
```

```
where c.cont_id = v.cont_id and c.title = '마이네임';
```

-작품의 에피소드

```
select episode
```

```
from content c, video v
```

```
where c.cont_id = v.cont_id and c.title = '마이네임' and season = 1;
```

-작품에 참여한 배우, 감독들의 이름, 역할

```
select name, role
```

```
from person p, join j, role r, content c
```

```
where r.r_id = j.r_id and p.p_id = j.p_id and j.cont_id = c.cont_id and c.title = '마이네임';
```

-작품의 장르

```
select genre  
  
from genre g, genretag t, content c  
  
where g.genr_id = t.genr_id and c.cont_id = t.cont_id and c.title = '마이네임';
```

-작품의 경고 정보

```
select content  
  
from warning w, warntag t, content c  
  
where w.warn_id = t.warn_id and c.cont_id = t.cont_id and c.title = '마이네임';
```

- 작품 시청

-작품 처음 봤을 때 기록

```
INSERT INTO PLAYLIST VALUES(3, 1, 1, 1, '2021-08-26 16:14', '2021-08-26 17:20', 54);
```

-이전 시청한 작품을 다시 시청했을 때(시청 시작 시각 수정)

```
UPDATE PLAYLIST  
  
SET STARTS = (TO_CHAR(SYSDATE,'yyyy.mm.dd hh24:mi'))  
  
WHERE MEM_ID = '3' AND PF_ID = '1' AND CONT_ID = '1' AND VID_ID = '1';
```

-이전 시청한 작품을 다시 시청했을 때(시청 종료 시각 수정)

```
UPDATE PLAYLIST  
  
SET ENDS = (TO_CHAR(SYSDATE,'yyyy.mm.dd hh24:mi'))  
  
WHERE MEM_ID = '3' AND PF_ID = '1' AND CONT_ID = '1' AND VID_ID = '1';
```

-작품 시청 중단한 지점

```
UPDATE PLAYLIST
```

```
SET INTERRUPTEDTIME = 34
```

```
WHERE MEM_ID = 3 AND PF_ID = 1 AND CONT_ID = 1 AND VID_ID = 1;
```

- 언어지원

-작품 자막

```
SELECT TITLE, LANGUAGE AS SUB
```

```
FROM CONTENT C, LANGUAGE L, SUPPORTSUB P
```

```
WHERE C.CONT_ID = P.CONT_ID AND L.LANG_ID = P.LANG_ID AND C.TITLE = '마이네임';
```

-작품 더빙

```
SELECT TITLE, LANGUAGE AS DUB
```

```
FROM CONTENT C, LANGUAGE L, SUPPORTDUB S
```

```
WHERE C.CONT_ID = S.CONT_ID AND L.LANG_ID = S.LANG_ID AND C.TITLE = '마이네임';
```

- 찜

```
INSERT INTO BOOKMARK VALUES(5, 1, 7, '21-11-05');
```

- 그 외 기능들

-회원의 결제 만료 일자

```
SELECT DUEDATE
```

```
FROM MEMBER
```

```
WHERE MEM_ID = 1;
```

-프로필 생성

```
INSERT INTO PROFILE VALUES(5, 3, '사자', 12, 0);
```

-회원의 시청제한 나이에 맞는 작품

```
select cont_id, title
```

```
from content
```

```
where cont_rating <= (select restriction from profile where mem_id = 4 and pf_id = 2);
```

-작품 다시보기

```
select interruptedtime
```

```
from playlist
```

```
where MEM_ID = 3 AND PF_ID = 1 AND CONT_ID = 1 AND VID_ID = 1;
```

-찜 목록 토대로 추천

```
select distinct c.title
```

```
from content c, genretag t, genre g
```

```
where c.cont_id = t.cont_id and t.genr_id = g.genr_id and g.genr_id in
```

```
(select genr_id
```

```
from bookmark o, genretag e
```

```
where e.cont_id = o.cont_id and o.mem_id = 5 and o.pf_id = 1 and o.dates +30 >  
sysdate);
```

7. 별첨

7.1 소감

김태양-----

처음 해보는 설계라 처음에는 매우 막막했다. 그리고 설계하는 과정에서 해당 주제에 대해 너무 필요없이 깊게 생각하고 있지 않나 하는 생각이 자주 들기도 하였다. 그리고 대학교 와서 처음 팀 내 리더를 맡게 되었는데 역할 분배에 대해 어려움을 느꼈고, 맺고 끊음을 정확하게 하지 못한 것이 아쉬움으로 남았다. 이번 팀프로젝트를 통해 리더가 갖추어야 하는 것들을 좀 더 알게 되었고, 팀프로젝트의 결과물이 제대로 나온 것 같아 뿌듯한 느낌이 든다.

최용현-----

평상시에 아무렇지 않게 넷플릭스를 사용해왔는데, 데이터베이스를 설계해보니 단순해 보이던 것이 사실은 매우 복잡하다는 것을 알게 되었다. 그리고 팀원들과 협업하는 과정에서 역할을 나누고 각각의 결과를 합치는 것도 굉장히 어렵다는 것을 알게 되었다. 그래도 이번 프로젝트를 진행하면서 고생을 한 만큼 데이터베이스 설계 능력과 팀원들과 협업 능력이 좀 더 발전한 것 같아 유익했다.

김승진-----

진행하며 단계별로 팀원들마다 많은 좋은 의견들을 냈고 그것들 중 더 좋은 의견들을 골라 프로젝트에 반영하는 과정이 생각보다 어려웠던 것 같습니다. 결과물을 최종으로 만들어지는데 많은 노력을 한 만큼 끝내고 보니 성취감 큰 것 같습니다.

조한웅-----

데이터베이스 팀프로젝트를 진행하면서 처음이다 보니 어려움이 있었지만 문제를 파악하고 팀원들이 해결방법을 제시하여 프로젝트를 무사히 마칠 수 있었다. 이번 팀프로젝트로 인해 협업능력을 키울 수 있었고 수업시간 때 이해가 어려웠던 부분을 이번 팀프로젝트를 하면서 더욱 더 이해할 수 있게 되었다.