mOVIE NETWORKING: FINAL PROJECT

Sara Osama Kilany - 20135012

Table of Contents

[Project Idea and Requirements 2](#_Toc111986837)

[Entity Relation Diagram (ERD) 3](#_Toc111986838)

[Relational model 4](#_Toc111986839)

[Database DDL 5](#_Toc111986840)

[Database DML Queries 5](#_Toc111986841)

[QUERY1 5](#_Toc111986842)

[QUERY2 5](#_Toc111986843)

[QUERY3 5](#_Toc111986844)

[QUERY4 6](#_Toc111986845)

# Project Idea and Requirements

MovieNetworking needs to start a platform that gathers all movie fans along with movies and their details.

The platform will have records for 2 main items - movies and series- that contain: the name, title, storyline mpaa rating, and popularity of the movie on the platform.

The page for movies and series can contain videos and photos uploaded to the platform.

Each item has one or more genres.

An item can be produced by one or more production companies.

Any item has one or more actors, writers, and directors.

Movies should save the duration, release date in cinemas and end date of movie in theatres.

Series save data for number of seasons, starting date for the series, end date if it's finished or none if this still airing.

Each series consists of at least one season, each season has its number (e.g. season 1, 2 or 4), overview of the season, the start date and end date of that season.

Each season includes one or more episodes, each episode has a brief overview, the airing date and the episode number in the at particular season.

Videos need to keep the following data: the file type, link to the video, date it was uploaded, title, and size.

Images are similar to videos. Both can be used in item's pages or in users' posts.

There are pages that contain data for people, there are 4 types of personas: actor, writer, director and user.

Each person is abstracted on the db with the name, gender, date of birth. They can have images and videos in their pages and profiles.

Each user adds email, phone and country to their profiles and can follow any number of users, comment and like their posts and make posts of his own.

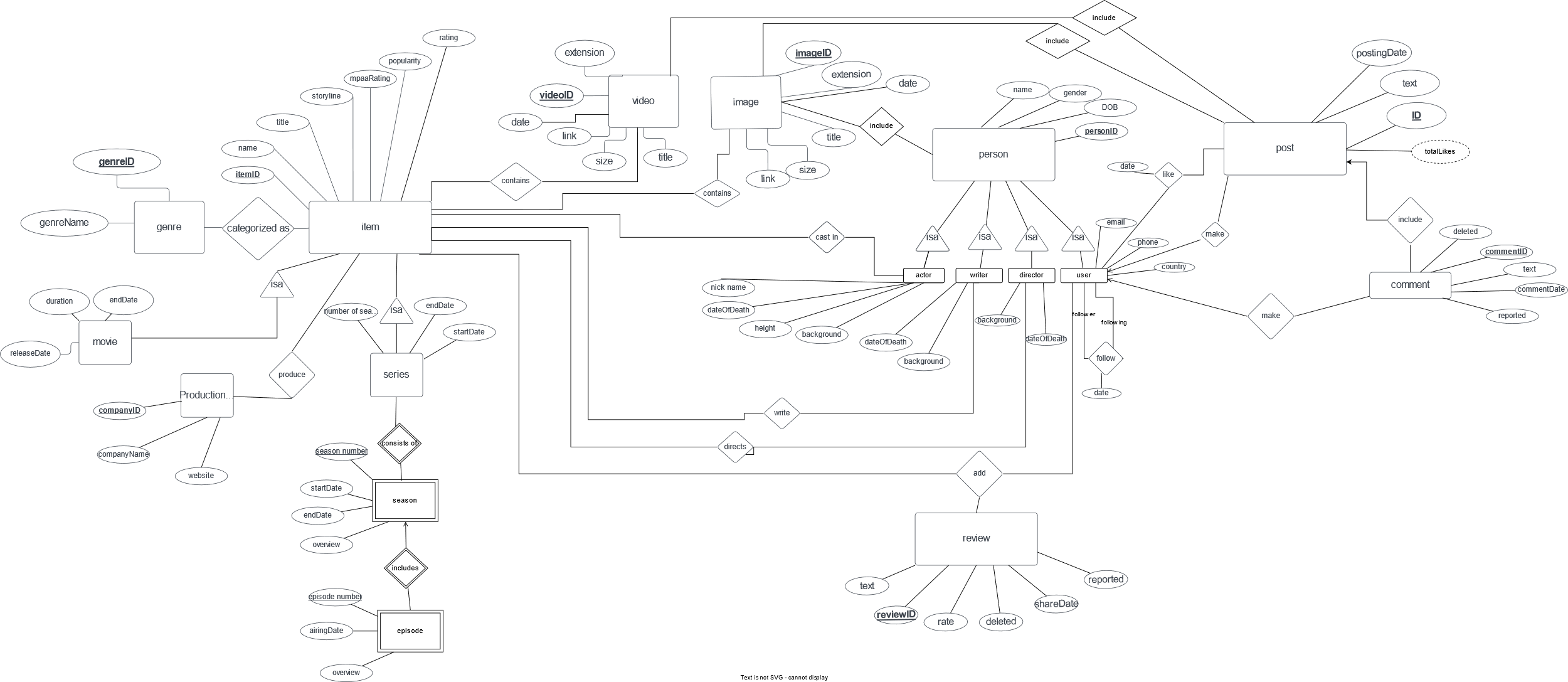
Any user can be followed by any number of users. And the date of following should be kept in records.

Each post is made by one user, commented and liked by any number of users. The post tracks posting date, text, any images or videos added to it, and total likes on the post made by users.

Each comment can be deleted and/or reported. The comment text and date of publishing is always shown to users on the platform.

A review is added by the user related to a movie or series item, the review should maintain text, user's rate defaulted to null, the share date of the review, status of the review if it's deleted and/or reported.

# Entity Relation Diagram (ERD)



# Relational model

Item (itemID, name, title, mppaRating, popularity, rating)

Movie(itemID, duration, releaseDate, endDate)

Series(ItemID, number of seasons, starteDate, endDate)

Season(ItemID, season number, startDate, endDate, overview)

Episode(itemID, seasonNumber, episodeNumber, airingDate, overview)

ProductionCompany(CompanyID, companyName, webisteLink)

Genre(GenreID, genreName)

Item-genre(genreID,ItemID)

Video(videoID, uploadDate,link,size,title,fileExtension)

photo(photoID, uploadDate,link,size,title,fileExtension)

Item-image(imageID,itemID)

Item-video(videoID, itemID)

Actor(personID, nickname, dateOfDeath, height, background)

Actor-castin-item(itemId,actorID)

Writer(personID, dateOfDeath, background)

Writer-item(writerID, itemID)

Director(personID, dateOfDeath, background)

Direct-item(itemID,directorID)

User(personID, email, phone, country)

User-follow(followerID, followingID, followDate)

Post(postID, text, postingDate, userID)

User-like-post(likeDate, userID, postID)

Comment(postID, commentID, text, userID, reported, deleted)

Review(reviewID, rate,deleted,reported,shareDate,text)

Review-Item-user(userID,reviewID,ItemID)

# Database DDL

Attached the file for creation relations and insertion of data in file ' SQLDDLcreate.sql'

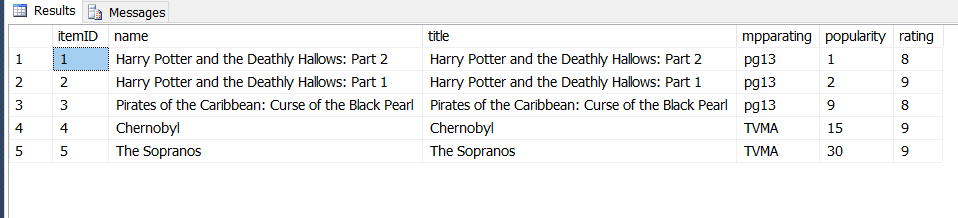
# Database DML Queries

Attached the file for creation relations and insertion of data in file ' SQLDMLquery.sql'

## QUERY1

All movies in DB

select \* from item



## QUERY2

All actors' names in Harry Potter and the Deathly Hollows: Part 1

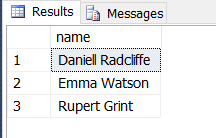
select name

from person

where personID in (select actorID

from actor\_castin\_item

where itemID = ( select itemID from item where name='Harry Potter and the Deathly Hallows: Part 2'))

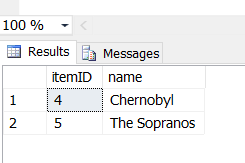


## QUERY3

All id, names of TV series in the DB

select i.itemID, name from item i, series s

where i.itemID = s.itemID



## QUERY4

Names of the movies reviewed by Victor Baldwin

select i.name

from item i, movie m

where (i.itemID= m.itemID) and m.itemID in (select it.itemID from review\_item\_user it

where it.userID=

(select person.personID from users,person

where users.personID=person.personID and person.name='Victor Baldwin')

);

