

Section: 09

Course: CPS510 - Database Systems 1

Team Number: 17

Assignment Number: A1

TA Name: Reza Barati

Team Members: Anmol Panchal, Deep Patel, Aryan Patel

Application of Interest: Movie Database Management System (DBMS)

Introduction:

For over a century, the film industry has been innovating new methods to deliver digital entertainment to millions of people around the world. From the early days of film reels, to today's digital age, the successful management and distribution of films has been essential in growing this \$77 billion USD industry.

For the smooth operation of such online stores, database management systems (DBMS) are used to ensure that the information stored (time period, movie title, purchase price, etc.) is all up to a professional standard.

A system like this contains several functions to make the management of not only film data (for consumers), but also the data related to user interactions (for producers) easily accessible.

Function	Objective/Description
InsertFilm	Accessed by Owner , to update film data
InsertUser	The Owner POST's a new User into the DBMS
InsertReview	The User POST's a new Review into the DBMS
InsertAward	The Owner POST's a new Award into the Film table with a specific id
GetUser	The Owner sends a GET request to the DBMS to get a specific User . The User may also press a button that will send a GET request to see their own profile
GetReview	The Owner sends a GET request to the DBMS to get a specific Review . The User can also activate an event that will send a GET request to view a Review

To fulfill the objectives of these functions, pieces of information can be represented as **entities**, and later accessed within the system as follows:

- 1.) **Film:** contains title, runtime, release year, director, cast, average user rating
- 2.) **Actor:** first and last name, birthdate, nationality, and filmography
- 3.) **Director:** first and last name, birthdate, nationality, and filmography
- 4.) **Producer:** first and last name, birthdate, nationality, company affiliation, and production history
- 5.) **Studio:** name, location, CEO, films accredited
- 6.) **User:** first and last name, username, contact information (email, password)
- 7.) **Reviews:** customer rating (5-star system), comments, and date of review for associated film title
- 8.) **Awards:** name of award, category of nomination, year of award

Further information about these **entities** are available through the sample tables below, which will be implemented in the source code at later stages of this project.

Film

Title	Runtime	Release Year	Director	Cast	Average User Rating
Oppenheimer	3h	2023	Christopher Nolan	Cillian Murphy, Robert Downey Jr., Matt Damon, etc.	4.3/5
Terminator 2: Judgment Day	2h 17m	1991	James Cameron	Arnold Schwarzenegger, Linda Hamilton, Robert Patrick, etc.	4.3/5
Mission: Impossible - Ghost Protocol	2h 13m	2011	Brad Bird	Tom Cruise, Paula Patton, Simon Pegg, etc.	3.7/5

Actor

First Name	Last Name	Birthdate	Nationality	Filmography
Cillian	Murphy	May 25, 1976	Irish	Oppenheimer, Peaky Blinders, Batman, etc.
Arnold	Schwarzenegger	July 30, 1947	Austrian	Terminator, Predator, Commando, etc.
Tom	Cruise	July 3, 1962	American	Mission Impossible, Top Gun, Rain Man etc.

Director

First Name	Last Name	Birthdate	Nationality	Filmography
Christopher	Nolan	July 30, 1970	British	Inception, Interstellar, Oppenheimer etc.
James	Cameron	August 16, 1954	Canadian	Avatar, Titanic, The Terminator etc.
Brad	Bird	September 24, 1957	American	Mission Impossible, Incredibles, Ratatouille etc.

Producer

First Name	Last Name	Birthdate	Nationality	Company Affiliation	Production History
Christopher	Nolan	July 30, 1970	British	Syncopy Inc.	Oppenheimer, Dunkirk etc.
James	Cameron	August 16, 1954	Canadian	Lightstorm Entertainment	Titanic, True Lies, etc.
Quentin	Tarantino	March 27, 1963	American	A Band Apart	From Dusk Till Dawn, Hostel, etc.

Studio

Name of Studio	Location	CEO	Films Accredited
Universal Studios	Universal City, CA	Mark Woodbury	Fast X, The Super Mario Bros, Jurassic World: Dominion, etc.
Warner Bros.	Burbank, CA	David Zaslav	Tenet, Inception, Shazam, etc.
Paramount Pictures	Los Angeles, CA	Brain Robbins	Top Gun: Maverick, Titanic, Shrek Forever After, Scream, etc.

User (contact information is fictional)

First Name	Last Name	Username	Email	Password
Deep	Patel	deep.patel	dp@film.com	abc123films
Aryan	Patel	aryan.patel	ap@film.com	def456films
Anmol	Panchal	anmol.panchal	ap@film.com	ghi789films

Review (samples of potential customer reviews)

Username	Date of Review	Rating	Description	Movie
Deep	09/09/2023	4/5	Good movie	Oppenheimer
Aryan	09/09/2023	4/5	Excellent movie	Terminator 2: Judgment Day
Anmol	09/09/2023	5/5	Best movie	Interstellar

Award

Name of Award	Award Presenter	Entity Selected	Year Awarded
Best Film	Academy Awards	Birdman	2015
Best Actor	Oscars Ceremony	Joaquin Phoenix	2020
Best Director	Academy Awards	Alfonso Cuaron	2014

These entities end up interacting with each other in various ways, whether that be an end user accessing the service, or components of the DBMS communicating with each other internally.

These **relationships** are outlined below. As an extension, potential entities can be defined separately such as watchlist, account login, to keep things organized and away from large tables.

- 1.) **Movie - Actor:** Each movie is linked with an actor in *one - to - many* relationships. Each movie has a cast of actors. Also the filmography of an actor contains a *one - to - many* relation because one actor can have worked on many movies.
- 2.) **Movie - Director:** Each movie is linked with a director in *one - to - one* relationship. Each movie has one director in the DBMS. The connection between Director - Movie is a *one - to - many* relationship, because directors have a filmography of several movies.
- 3.) **Movie - Producer:** Similar to a director, for most movies this is a *one - to - one* relation while the reverse is a *one - to - many*.
- 4.) **Movie - Studio:** Each movie is created at a specific studio, showing *one - to - one* relation. The reverse is a *one - to - many* because a single studio can create many films.
- 5.) **User - Review:** The review by each user (contains 5-star-rating, comment) is a definitive *one - to - one* relation in the DBMS because the data for each review is linked to a specific user, specific movie, and specific date.
- 6.) **Awards - Movie:** This is a *one - to - one* relation, because an award category can only have one winner for a given year.
- 7.) **User - Watchlist:** Each user can add several films to the watchlist in their account. This represents a *one - to - many* relationship among the data of these entities.

Through the usage of this data, the FilmDBMS aims to centralize data about movie production details, the public interest in films, and the development best practices of implementing the DB with an interactive application.

Although this proposal outlines the base details of a FilmDBMS, at later stages of this project, these can be integrated into a Python based GUI using the tkinter framework.