Final Project Submission

INST 327 - Database Design and Modeling

Henry Stought, Adam Santilli, Fernando Fisk, Michael Hough, Reed Fodge

Section: 0203

12/14/2020

Team 5

Introduction

The database that we created for this project was centralized around movies in the United States and the United Kingdom after the year 2000. Our database stores information from a plethora of movies during this timeframe including its title, genre, content rating, the year it was released, the name of the director, the lead actor/actress and certain financial aspects such as the budget and the gross income of the movie. A problem that many film executives come across in their field of work is creating something that will capture their audience's attention, keep them engaged and will give them the greatest return on their film investment, this database works to expedite that process. This database was created for movie studios and film executives to have a place where all the important information regarding movies and their releases are stored. Using this information, they can observe the trends and relationships between certain film components (actor director combinations and gross income, etc.) to achieve the highest commercial success when creating their next feature film.

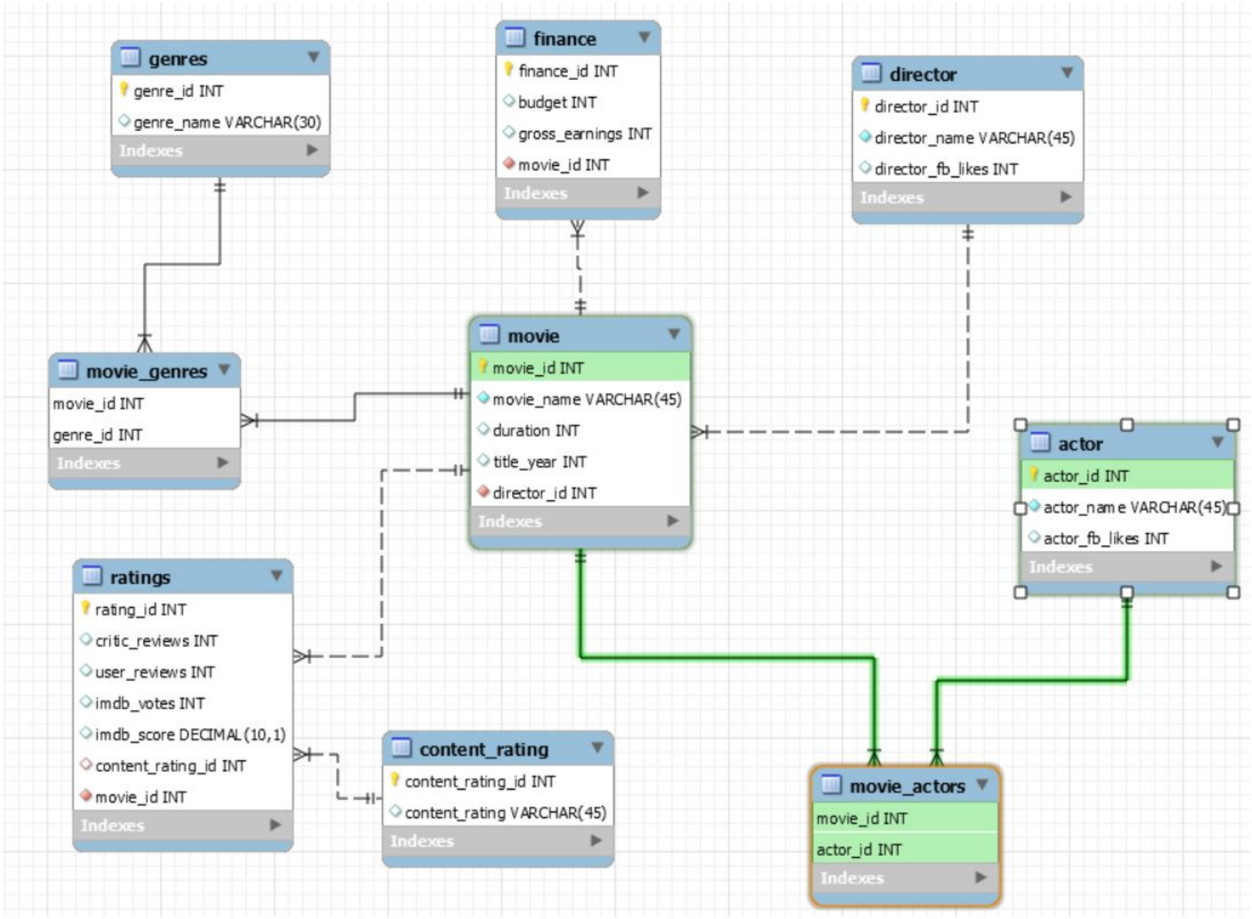
Database Description

Logical design: When deciding how to start designing our database and arranging the film components in the most effective way possible to achieve the goal of our database, we came up with nine tables in total. We originally created four tables: the finance table, director table, ratings table, and the movie table all of which contained information specific to the movies we included in our database. Once these tables were created, we proceeded to create the actors table where we listed some of the lead and supporting actors associated with the movies. After creating those tables we decided to include two tables containing some of the most popular movie genres and all of the possible content ratings that each movie could have in order for users to be able to take these important film components into account when observing trends in our database. The last two tables that we created were the movie_actors and movie_genres table; in the movie_actors table we connected the actor table and the movie table by showing which actors were involved in which movies, and in the movie_genres table we connected the genres table and the movie table by showing which genres were associated with which movies.

Physical Design: As stated in the logical design section, our database has nine tables with the main table being the movie table. Almost all of our tables were connected in some way using the movie_id key, in cases where a specific table didn't have a movie_id key we connected that table with a linking table containing the movie_id key by using a key that both tables contained. For example, the content ratings table is not movie specific and therefore does not contain a movie_id key so we connected that table to the movies table through the ratings table using a common key in both tables, the content_rating_id key. The only table that did not need to use the movie_id key to connect to the movie table was the director table due to the fact that it contained the director_id key, one of the keys in the movie table. There are three one to many relationships in our database: ratings, movie_genres, and movie_actors and two many to one relationships: finance and director. In regards to the linking tables: the ratings table has a many to one relationship with the content rating table, the movie_actors table has a many to one relationship with the actor table, and the movie_genres table has a many to one relationship with the genres table.

Sample Data: We retrieved the data in our database from a csv file containing thousands of movies from IMDB's website and we compiled the data ourselves into the database. The source for the data is cited at the bottom of the report. This took us a lot of time and it was an extremely tedious process going back and forth from the file to our database to input the correct information without making mistakes. We decided to only include movies released after the year 2000 that had fairly large budgets in order for our users to be able to draw relevant conclusions from our database and observe the trends of similar types of movies in comparison to one another. Specifically, we included 34 movie entries in the database.

ERD:



Views and Queries:

View Name	Req. A	Req. B	Req. C	Req. D	Req. E
Budget Vs. Gross (View 1)	X				
Avg. Film Budget by Actor (View 2)	X	X	X		

PG-13 Rated Films by Actor (View 3)	X	X		X	X
Avg. IMDB Score by Director (View 4)	X		X		
Ratings Ratio (View 5)	X	X			

We ended up creating 5 queries saved as views for our database which asked and answered specific questions about or database that could be answered using SQL. Our first query creates a view that shows the film's budget and its gross income so that users can compare the two and returns a ratio of budget over income to determine how successful the movies were in comparison to one another. Our second query creates a view that calculates and returns the average film budget for each actor so that users can be able to determine how expensive employing certain actors really is, aside from how much they are being paid. Our third query creates a view that returns the number of PG-13 rated movies that each actor has been a part of for the movies in our database. Our fourth query creates a view that calculates and returns the average IMDB score for each director so that our users can see how successful each director is in comparison to one another by comparing their average IMDB scores. Lastly, our fifth query creates a view that returns a ratio of the number of critic reviews vs the number of user reviews where the ratio is greater than 3.

Changes from Original Design

A large portion of the changes we made from our original design included the pruning of both our database design and dataset. Originally, we had intended to include the Actor IDs of the top three billed actors in the movie, however eventually we decided to remove the second and third billed actors to include just the top billed actor. We did this both as a result of starting to create our ERD as well as instructor feedback which helped us identify and remove redundancies within our database. In addition to removing the second and third billed actors, we also consolidated our social media tables. Originally, we had designed individual social media tables for movies, actors, and directors, however because each of these tables consisted solely of a primary key and single INT value, we decided to reevaluate the structure. After determining the performance gains of this design were negligible, we decided to move the social media rating to their respective master tables. Finally, we reduced our dataset, changing the starting point for movie releases to be 2000 to create a clean, 20-year period to work with.

Lessons Learned

One of the most important lessons we learned from this group project is effective communication. While this can be a hard goal to achieve in normal times, the effects of the COVID-19 pandemic strained communication even further. Being unable to meet in person, we all had to find the most effective ways of expressing our ideas and opinions with one another virtually. This was primarily handled two ways, with a GroupMe group chat and Zoom meetings, which both offered different opportunities for communication. While we at times had difficulty synchronizing our team's goal between one another, we learned how to effectively communicate with each other, so we were all on the same page. In addition, we also ran into some technical issues while using MySQL workbench. For instance, at one point one of our group members was unable to upload our .mwb file to view our ERD diagram, citing an error with unserializing GRT. After some investigation on Google, we managed to solve this issue with updating and restarting MySQL workbench.

Potential Future Work

In its current state, our database is only tracking movies from the year 2000 until now that have had large budgets. If we expanded our timespan to the first feature films in the early 20th century, we would have significantly more data to work with. This increase in data would allow users of the database to more effectively track trends over time, as more data points will always provide a clearer picture for trends. In addition, expanding our data to independent films would allow both smaller filmmakers to determine how to more effectively cater to their audience as well as allowing studios to determine which independent films could prove to be financial successes. We could also expand our data to films outside of the United States, as foreign movie markets can be extremely lucrative to studios. Adding international films would massively expand our database as we would have to add a significant number of new actors and directors on top of the films, as well as having to convert their currencies into USD for the finance tables.

Data Source

Janzen, Shawn. “movie_metadata.csv.” Box, 3 June 2020, umd.app.box.com/s/r3es3mykt4psovrod5mt08fchd9lnsnu/file/673988277404