

**Diseño y explotación de la base de datos MoviesDB**

Luis Aguilar, laaguilar21@utpl.edu.ec **(estudiante líder)**

Steven Vega, srvega4@utpl.edu.ec

Marco Herrera, mshm1296@hotmail.com

15 de enero de 2025

**Índice de contenidos**

[1. Diseño y explotación de la base de datos MoviesDB 3](#_Toc187847308)

[1.1. Obtención de datos 3](#_Toc187847309)

[1.1.1. Análisis descriptivo del dataset 3](#_Toc187847310)

[1.1.2. Descripción de datos del dataset 3](#_Toc187847311)

[1.1.3. Descripción de los requisitos de datos para MoviesDB 3](#_Toc187847312)

[1.2. Modelado Conceptual 4](#_Toc187847313)

[1.2.1. Proceso de modelado conceptual 4](#_Toc187847314)

[1.2.2. Resumen del modelo conceptual 4](#_Toc187847315)

[1.3. Modelado Lógico 4](#_Toc187847316)

[1.3.1. Proceso de modelado lógico 4](#_Toc187847317)

[1.3.2. Diccionario de Datos 4](#_Toc187847318)

[1.4. Modelado Físico 5](#_Toc187847319)

[1.4.1. Creación de la base de datos 5](#_Toc187847320)

[1.4.2. Inserción de datos 5](#_Toc187847321)

[1.5. Validación y Explotación de la base de datos 5](#_Toc187847322)

[1.5.1. Consultas 5](#_Toc187847323)

[1.5.2. Inserción de datos 5](#_Toc187847324)

[2. Referencias Bibliográficas 5](#_Toc187847325)

# Diseño y explotación de la base de datos MoviesDB

El proyecto consiste en diseñar y desarrollar una base de datos llamada \*\*MoviesDB\*\* a partir de un dataset en formato CSV con información sobre películas. Las etapas incluyen la extracción y análisis de datos, el diseño del modelo conceptual para identificar entidades y relaciones, la creación del modelo lógico con tablas y relaciones normalizadas, el desarrollo del modelo físico mediante la implementación de la base de datos en SQL, y finalmente la validación y explotación de la base con consultas e inserciones para garantizar su funcionalidad y utilidad.

## Obtención de datos

### Análisis descriptivo del dataset

|  |  |
| --- | --- |
| **Nombre del Dataset** | Movies pi\_movies\_complete |
| **Objetivo del Dataset** | Gestionar y analizar información sobre películas, actores, equipos técnicos, calificaciones y palabras clave para estudios de popularidad, rentabilidad y otros análisis relacionados con la industria cinematográfica. |
| **Cuántas películas están incluidas** | Por determinar (se debe contar el total de registros únicos en la tabla Pelicula). |
| **Cuáles son los géneros más comunes y menos presentados** | Los géneros más comunes y menos presentados pueden analizarse explorando el campo genres en la tabla Pelicula y calculando la frecuencia de cada género. |
| **Cuántas películas se producen por año** | Se puede obtener al agrupar los datos por año usando el campo release\_date. Esto permite identificar tendencias anuales en la producción cinematográfica. |
| **Se define si hay más de un director de películas incluido** | Sí, se puede verificar revisando los datos en la tabla Crew donde el campo Job es "Director". |
| **Cuáles otros datos o metadatos se incluyen en el Dataset** | Información sobre presupuesto, recaudación, duración, idiomas, países de producción, palabras clave, calificaciones de usuarios, popularidad y colecciones asociadas. |
| **Qué información podría analizar con este Dataset (al menos 3 ideas)** | 1. Relación entre presupuesto y recaudación.  2. Comparación de géneros más populares por región o año.  3. Análisis de tendencias en duración de películas a lo largo de los años. |
| **Con qué datos de películas que encuentran en sitios como - Netflix, IMBD, TMBD- se relaciona el dataset** | Se relaciona con métricas como popularity, vote\_average, genres, cast, crew, y otros metadatos comunes en plataformas como IMDb y TMDB. |

### Descripción de datos del dataset

| **Columna** | **Tipo de Dato** | **Descripción** |
| --- | --- | --- |
| Adult | boolean | Identifica si la pelicula es para adultos o no. |
| belong\_to\_collection | JSON | Identifica a que grupo de peliculas pertenece una pelicula |
| budget | Double | Presupuesto de la pelicula |
| genres | JSON | Género o categorías asociadas. |
| homepage | String | Ruta de la pelicula a una pagina web con informacion sobre ella |
| id | String | Identificador de la pelicula |
| imdb\_id | String | Identificador de la base de datos de la pelicula |
| original\_languague | String | Idioma original de la pelicula |
| original\_title | String | Titulo original de la pelicula |
| overview | String | Descripcion de la pelicula |
| popularity | String | Popularidad de la pelicula |
| poster\_Path | String | Ruta del poster de la coleccion |
| production\_companies | JSON | Compañias productoras de la pelicula |
| production\_countries | JSON | Pais o paises en el que se produce la pelicula |
| release\_date | String | Fecha de lanzamiento de la pelicula |
| revenue | String | Ganancias de la pelicula |
| runtime | String | Duracion de la pelicula |
| spoken\_languagues | JSON | Idiomas al que esta doblada la pelicula |
| status | String | Estado de la pelicula |
| tagline | String | Representacion leve de la pelicula |
| title | String | Ruta del póster de la colección. |
| video | boolean | Si contiene o no un video |
| vote\_average | Double | Promedio de votos |
| vote\_count | Int | Conteo de votos |
| keywords | JSON | Las palabras clave de la pelicula |
| cast | JSON | Los actores que intervienen dentro de la pelicula con su respectivo orden, identificadores y personaje al que interpretan |
| crew | JSON | El equipo tecnico detras de la pelicula que ayuda en su produccion y desarrollo |
| ratings | JSON | La validacion y aceptacion por parte de los usuarios de la pelicula |

### Descripción de los requisitos de datos para MoviesDB

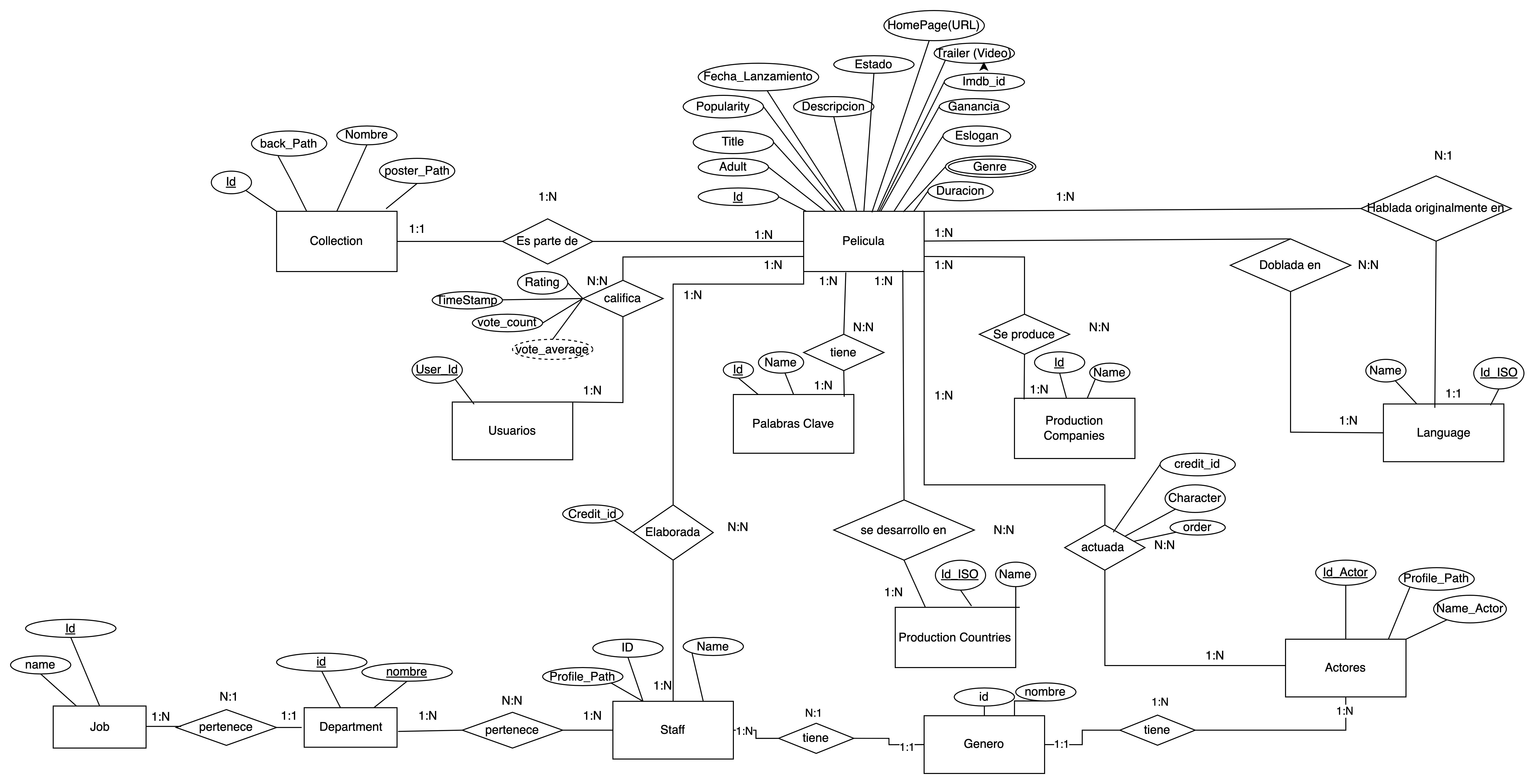
Para crear una base de datos a partir del dataset, es esencial identificar tablas principales: películas, géneros, elenco, equipo técnico, producción, palabras clave y calificaciones. Las relaciones entre estas tablas deben ser normalizadas, destacando relaciones muchos a muchos, como entre películas y géneros, o películas y elenco. Es necesario extraer datos anidados en formato JSON para tablas intermedias, como movie\_genres o movie\_cast, asegurando integridad con claves primarias y foráneas en todas las tablas relacionadas.

Además, la base debe garantizar consistencia mediante validación de formatos (como fechas o identificadores únicos) y la creación de índices en campos clave para optimizar consultas. Tablas como ratings requieren relaciones uno a muchos con películas, mientras que producción y países involucrarán múltiples relaciones. Este diseño permitirá manejar datos complejos como palabras clave y miembros del equipo técnico de manera estructurada, facilitando consultas robustas y escalabilidad en la gestión de información cinematográfica.

## Modelado Conceptual

### Proceso de modelado conceptual

Este modelo conceptual se creó analizando cuidadosamente los datos disponibles, identificando las entidades principales como "Películas", "Géneros", "Palabras Clave", "Actores" y "Staff", que representan los conceptos clave del sistema. Cada una de estas entidades contiene información específica, como el título de la película, su popularidad, o los nombres de actores y miembros del equipo técnico. Para representar las conexiones entre estas entidades, se identificaron relaciones importantes, como "una película tiene varios géneros" o "una película puede ser actuada por muchos actores". Además, se procesaron datos más complejos, como las listas de géneros o palabras clave, separándolos en tablas propias para organizar mejor la información. Esto asegura que los datos estén bien estructurados, sin duplicaciones y listos para ser utilizados.



### Resumen del modelo conceptual

|  |  |
| --- | --- |
| **Cantidad** | |
| Entidades | 11 |
| Relaciones | 10 |

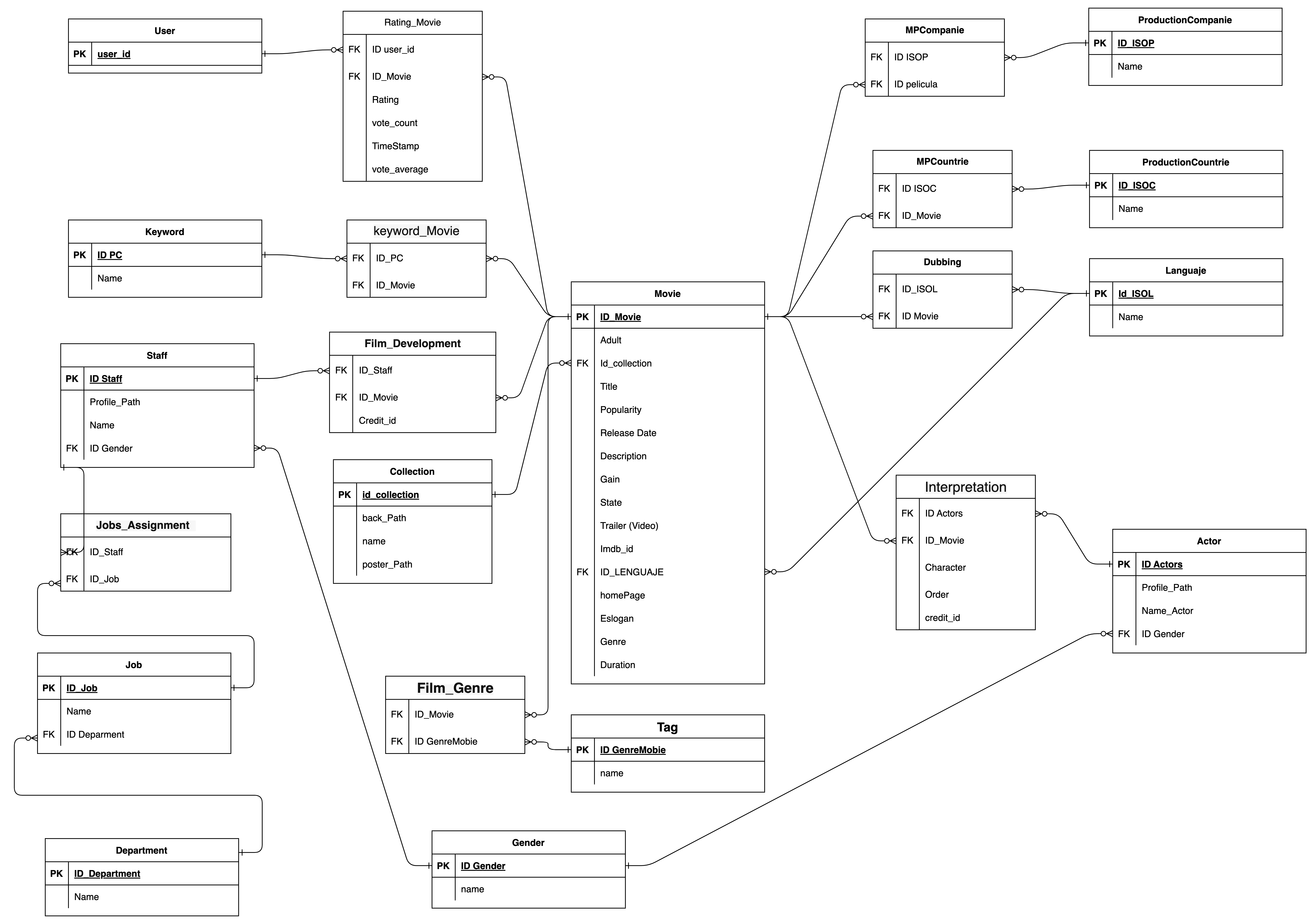
|  |  |
| --- | --- |
| **Atributos de entidades** | |
| **Entidad** | **Atributos** |
| Película | Id, Title, Descripción, Fecha\_Lanzamiento, Popularity, Eslógan, Duración, Estado, Adult, Imdb\_id, HomePage, Ganancia, trailer, Genre |
| Collection | Id, Nombre, back\_Path, poster\_Path |
| Palabras Clave | Id, Name |
| Production Companies | Id, Name |
| Production Countries | Id\_ISO, Name |
| Language | Id\_ISO, Name |
| Actores | Id\_Actor, Name\_Actor, Profile\_Path |
| Staff | ID, Name, Profile\_Path |
| Usuarios | User\_Id |
| Department | id, nombre |
| Job | Id, name |

|  |  |
| --- | --- |
| **Atributos de relación** | |
| Relación | Atributos |
| Es parte de | Id, Nombre |
| Tiene | Id, Name |
| Se produce | Id, Name |
| Se desarrolló en | Id\_ISO, Name |
| Doblada en | Id\_ISO, Name |
| Hablada originalmente en | Id\_ISO, Name |
| Actuada | Credit\_id, Character, order, Id\_Actor |
| Elaborada | Credit\_id |
| Califica | TimeStamp, vote\_count, vote\_average, Rating |
| Pertenece | Id, name |

## Modelado Lógico

### Proceso de modelado lógico

Cada entidad del modelo conceptual (como "Película", "Género", "Actor") se convirtió en una tabla en el modelo lógico. Los atributos identificados en el modelo conceptual (como Title, Popularity o Name) ahora son columnas en las tablas correspondientes. Las relaciones identificadas en el modelo conceptual, como "una película tiene muchos géneros" o "una película es actuada por varios actores", fueron representadas mediante tablas intermedias o "join tables". Por ejemplo, las relaciones de muchos a muchos entre películas y géneros resultaron en la tabla Film\_Genre, mientras que la relación entre películas y palabras clave generó la tabla keyword\_Movie. Las entidades y tablas intermedias se enriquecieron con llaves primarias (por ejemplo, ID en las tablas principales) y llaves foráneas (como movie\_id en Film\_Genre) para garantizar la integridad referencial entre las tablas.



Se aplicó un proceso de normalización para descomponer datos complejos y eliminar redundancias. Por ejemplo, en lugar de almacenar directamente los países o compañías de producción en la tabla de películas, se crearon tablas separadas como ProductionCountrie y ProductionCompanie vinculadas a la tabla principal de películas mediante relaciones.Algunas relaciones más específicas, como el personal (Staff) y sus asignaciones de trabajo (Jobs\_Assignment), se manejaron mediante la creación de tablas adicionales que modelan detalles específicos, como departamentos y roles en las películas.

### Diccionario de Datos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: User** | | | | |
| **Atributo** | **Tipo de Dato** | **Dominio** | **Descripción** | **Restricciones** |
| user\_id | INT | >0 | Identificador del usuario | Único, no nulo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: Rating\_Movie** | | | | |
| **Atributo** | **Tipo de Dato** | **Dominio** | **Descripción** | **Restricciones** |
| ID\_user\_id | INT | >0 | Referencia al usuario que realiza la calificación | No nulo |
| ID\_Movie | INT | >0 | Referencia a la película calificada | No nulo |
| Rating | DOUBLE | 0-10 | Calificación asignada por el usuario | No negativo |
| vote\_count | INT | >0 | Número de votos registrados | No negativo |
| TimeStamp | DATE | - | Fecha y hora de la calificación | No nulo |
| vote\_average | DOUBLE | 0-10 | Promedio de calificaciones | No negativo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: Keyword** | | | | |
| **Atributo** | **Tipo de Dato** | **Dominio** | **Descripción** | **Restricciones** |
| ID\_PC | INT | >0 | Identificador de palabra clave | Único, no nulo |
| Name | STRING | 10-50 | Nombre de la palabra clave | No nulo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: keyword\_Movie** | | | | |
| **Atributo** | **Tipo de Dato** | **Dominio** | **Descripción** | **Restricciones** |
| ID\_PC | INT | >0 | Referencia a la palabra clave asociada | No nulo |
| ID\_Movie | INT | >0 | Referencia a la película asociada | No nulo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: Staff** | | | | |
| **Atributo** | **Tipo de Dato** | **Dominio** | **Descripción** | **Restricciones** |
| ID\_Staff | INT | >0 | Identificador del miembro del staff | Único, no nulo |
| Profile\_Path | STRING | - | Ruta al perfil del miembro del staff | Opcional |
| Name | STRING | - | Nombre del miembro del staff | No nulo |
| ID\_Gender | INT | >0 | Referencia al género del miembro | No nulo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: Jobs\_Assignment** | | | | |
| Atributo | Tipo de Dato | Dominio | Descripción | Restricciones |
| ID\_Staff | INT | >0 | Referencia al miembro del staff asignado | No nulo |
| ID\_Job | INT | >0 | Referencia al trabajo asignado | No nulo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: Job** | | | | |
| **Atributo** | **Tipo de Dato** | **Dominio** | **Descripción** | **Restricciones** |
| ID\_Job | INT | >0 | Identificador del trabajo | Único, no nulo |
| Name | STRING | - | Nombre del trabajo | No nulo |
| ID\_Department | INT | >0 | Referencia al departamento del trabajo | No nulo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: Department** | | | | |
| **Atributo** | **Tipo de Dato** | **Dominio** | **Descripción** | **Restricciones** |
| ID\_Department | INT | >0 | Identificador del departamento | Único, no nulo |
| Name | STRING | - | Nombre del departamento | No nulo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: Film\_Development** | | | | |
| **Atributo** | **Tipo de Dato** | **Dominio** | **Descripción** | **Restricciones** |
| ID\_Staff | INT | >0 | Referencia al miembro del staff involucrado | No nulo |
| ID\_Movie | INT | >0 | Referencia a la película en desarrollo | No nulo |
| Credit\_id | STRING | - | Identificador del crédito | Único, no nulo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: Collection** | | | | |
| **Atributo** | **Tipo de Dato** | **Dominio** | **Descripción** | **Restricciones** |
| ID\_collection | INT | >0 | Identificador de la colección | Único, no nulo |
| back\_Path | STRING | - | Ruta de la imagen de fondo | Opcional |
| name | STRING | - | Nombre de la colección | No nulo |
| poster\_Path | STRING | - | Ruta de la imagen del póster | Opcional |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabla: Movie** | | | | |
| **Atributo** | **Tipo de Dato** | **Dominio** | **Descripción** | **Restricciones** |
| ID\_Movie | INT | >0 | Identificador de la película | Único, no nulo |
| Adult | BOOLEAN | true/false | Indicador si la película es para adultos | No nulo |
| Title | STRING | - | Título de la película | No nulo |
| Popularity | NUMERIC | >0 | Popularidad de la película | No negativo |
| Release Date | DATE | - | Fecha de estreno | Opcional |
| Description | STRING | - | Descripción de la película | Opcional |
| Gain | NUMERIC | - | Ingresos generados | Opcional |
| State | STRING | - | Estado de producción | Opcional |
| Trailer | STRING | - | Enlace al tráiler | Opcional |
| imdb\_id | STRING | - | ID de IMDb | Único, no nulo |
| homePage | STRING | - | Página principal | Opcional |
| Duration | NUMERIC | >0 | Duración en minutos | No negativo |

## Modelado Físico

### Creación de la base de datos

**Explicación breve del proceso**

El modelado físico de la base de datos se llevó a cabo utilizando el programa MySQL Workbench, una herramienta especializada que facilita el diseño, creación y administración de bases de datos relacionales. Este proceso inició con la revisión y análisis del modelo lógico final previamente elaborado, el cual sirvió como base para estructurar de manera eficiente el modelo físico. A partir de este análisis, se redactó progresivamente el script SQL necesario para la creación de la base de datos, asegurando que todas las especificaciones del diseño lógico y el diccionario de datos se reflejaran de manera precisa.

Durante la implementación, se definieron cuidadosamente los atributos de cada tabla, asegurando la correcta asignación de tipos de datos y restricciones necesarias para garantizar la integridad y consistencia de la información. Además, se establecieron las claves primarias como identificadores únicos de cada registro y las claves foráneas para representar las relaciones entre las tablas, cumpliendo con los principios de normalización. Este enfoque permitió construir una base de datos final optimizada y preparada para soportar las operaciones requeridas por el sistema.

**Sentencias DDL de la creación de la base de datos**

CREATE DATABASE integrador2;

USE integrador2;

CREATE TABLE Gender (

ID\_Gender INT PRIMARY KEY,

Name VARCHAR(50) NOT NULL

);

CREATE TABLE Department (

ID\_Department INT PRIMARY KEY,

Name VARCHAR(100) NOT NULL

);

CREATE TABLE Job (

ID\_Job INT PRIMARY KEY,

Name VARCHAR(100) NOT NULL,

ID\_Department INT,

FOREIGN KEY (ID\_Department) REFERENCES Department(ID\_Department)

);

CREATE TABLE Staff (

ID\_Staff INT PRIMARY KEY,

Profile\_Path VARCHAR(255),

Name VARCHAR(100) NOT NULL,

ID\_Gender INT,

FOREIGN KEY (ID\_Gender) REFERENCES Gender(ID\_Gender)

);

CREATE TABLE Jobs\_Assignment (

ID\_Staff INT,

ID\_Job INT,

PRIMARY KEY (ID\_Staff, ID\_Job),

FOREIGN KEY (ID\_Staff) REFERENCES Staff(ID\_Staff),

FOREIGN KEY (ID\_Job) REFERENCES Job(ID\_Job)

);

CREATE TABLE Collection (

ID\_Collection INT PRIMARY KEY,

Back\_Path VARCHAR(255),

Name VARCHAR(100),

Poster\_Path VARCHAR(255)

);

CREATE TABLE Language (

ID\_ISOL VARCHAR(10) PRIMARY KEY,

Name VARCHAR(100) NOT NULL

);

CREATE TABLE Movie (

ID\_Movie INT PRIMARY KEY,

ID\_Collection INT,

Title VARCHAR(255) NOT NULL,

Popularity DECIMAL(10, 2),

Release\_Date DATE,

Description TEXT,

Gain DECIMAL(15, 2),

State VARCHAR(50),

Trailer VARCHAR(255),

IMDB\_ID VARCHAR(50),

ID\_Language VARCHAR(10),

Homepage VARCHAR(255),

Slogan VARCHAR(255),

Genre VARCHAR(100),

Duration INT,

Adult BOOLEAN,

FOREIGN KEY (ID\_Collection) REFERENCES Collection(ID\_Collection),

FOREIGN KEY (ID\_Language) REFERENCES Language(ID\_ISOL)

);

CREATE TABLE Film\_Development (

ID\_Staff INT,

ID\_Movie INT,

Credit\_ID VARCHAR(50),

PRIMARY KEY (ID\_Staff, ID\_Movie),

FOREIGN KEY (ID\_Staff) REFERENCES Staff(ID\_Staff),

FOREIGN KEY (ID\_Movie) REFERENCES Movie(ID\_Movie)

);

CREATE TABLE ProductionCompany (

ID\_ISOP INT PRIMARY KEY,

Name VARCHAR(100) NOT NULL

);

CREATE TABLE ProductionCountry (

ID\_ISOC INT PRIMARY KEY,

Name VARCHAR(100) NOT NULL

);

CREATE TABLE MPCompany (

ID\_ISOP INT,

ID\_Movie INT,

PRIMARY KEY (ID\_ISOP, ID\_Movie),

FOREIGN KEY (ID\_ISOP) REFERENCES ProductionCompany(ID\_ISOP),

FOREIGN KEY (ID\_Movie) REFERENCES Movie(ID\_Movie)

);

CREATE TABLE MPCountry (

ID\_ISOC INT,

ID\_Movie INT,

PRIMARY KEY (ID\_ISOC, ID\_Movie),

FOREIGN KEY (ID\_ISOC) REFERENCES ProductionCountry(ID\_ISOC),

FOREIGN KEY (ID\_Movie) REFERENCES Movie(ID\_Movie)

);

CREATE TABLE Actor (

ID\_Actors INT PRIMARY KEY,

Profile\_Path VARCHAR(255),

Name\_Actor VARCHAR(100),

ID\_Gender INT,

FOREIGN KEY (ID\_Gender) REFERENCES Gender(ID\_Gender)

);

CREATE TABLE Interpretation (

ID\_Actors INT,

ID\_Movie INT,

Character VARCHAR(100),

Order1 INT,

Credit\_ID VARCHAR(50),

PRIMARY KEY (ID\_Actors, ID\_Movie),

FOREIGN KEY (ID\_Actors) REFERENCES Actor(ID\_Actors),

FOREIGN KEY (ID\_Movie) REFERENCES Movie(ID\_Movie)

);

CREATE TABLE Keyword (

ID\_PC INT PRIMARY KEY,

Name VARCHAR(100) NOT NULL

);

CREATE TABLE Keyword\_Movie (

ID\_PC INT,

ID\_Movie INT,

PRIMARY KEY (ID\_PC, ID\_Movie),

FOREIGN KEY (ID\_PC) REFERENCES Keyword(ID\_PC),

FOREIGN KEY (ID\_Movie) REFERENCES Movie(ID\_Movie)

);

CREATE TABLE Tag (

ID\_GenreMovie INT PRIMARY KEY,

Name VARCHAR(100) NOT NULL

);

CREATE TABLE Film\_Genre (

ID\_Movie INT,

ID\_GenreMovie INT,

PRIMARY KEY (ID\_Movie, ID\_GenreMovie),

FOREIGN KEY (ID\_Movie) REFERENCES Movie(ID\_Movie),

FOREIGN KEY (ID\_GenreMovie) REFERENCES Tag(ID\_GenreMovie)

);

CREATE TABLE Rating\_Movie (

ID\_User\_ID INT,

ID\_Movie INT,

Rating DECIMAL(3, 1),

Vote\_Count INT,

TimeStamp DATETIME,

Vote\_Average DECIMAL(3, 1),

PRIMARY KEY (ID\_User\_ID, ID\_Movie),

FOREIGN KEY (ID\_Movie) REFERENCES Movie(ID\_Movie)

);

CREATE TABLE User (

User\_ID INT PRIMARY KEY

);

CREATE TABLE Dubbing (

ID\_Dubbing INT PRIMARY KEY,

Name VARCHAR(100) NOT NULL,

ID\_Language VARCHAR(10),

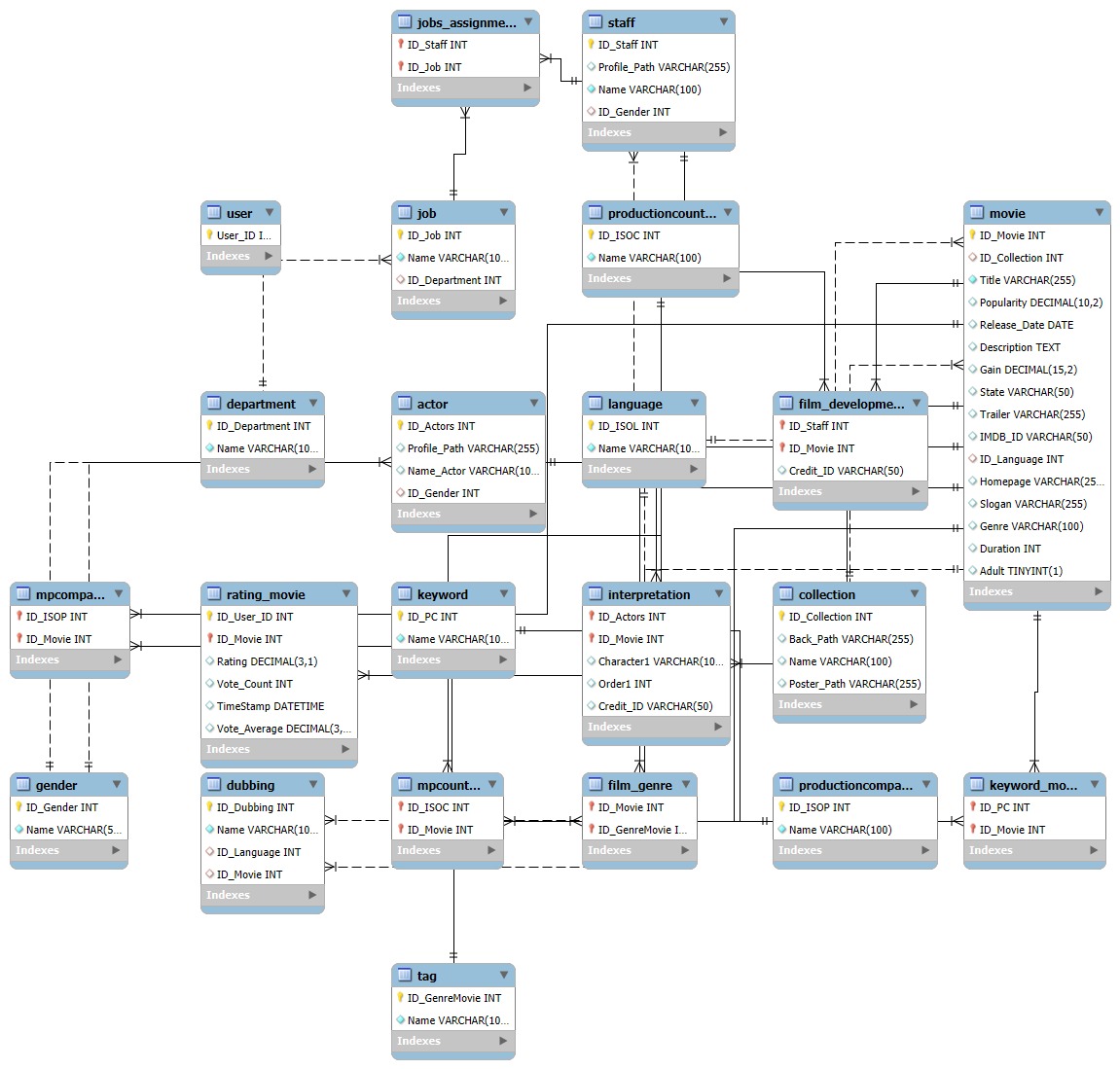
ID\_Movie INT,

FOREIGN KEY (ID\_Language) REFERENCES Language(ID\_ISOL),

FOREIGN KEY (ID\_Movie) REFERENCES Movie(ID\_Movie)

);

**Diagrama SQL exportado desde WorkBench**



### Inserción de datos

Explicación breve del proceso

Adjuntar (en Canvas) un sólo script que contenga tanto la creación y la población de la base de datos.

## Validación y Explotación de la base de datos

Validar con 10 operaciones: 9 consultas y sus resultados y 1 inserción completa de datos de películas.

### Consultas

Deben ser:

* multitabla mínimo combinar tres tablas
* incluir subconsultas
* consultas con grupos de datos (GROUP BY)

Por cada consulta coloque descripción, SQL y resultado.

### Inserción de datos

* Realizar la inserción de datos de una película completa

Finalmente, genere un script de las consultas e inserción y anéxelo a la entrega en canvas.

# Referencias Bibliográficas

Coloque aquí las referencias bibliográficas utilizadas en formato [APA 7ma edición](https://drive.google.com/file/d/1DKSsMRg4eD4-9CctrqEgnK_YrebwYZAA/view?usp=drive_link)