

Gestión de Películas y Reseñas

Descripción:

Guarda información de películas, directores, actores, géneros, usuarios y reseñas con puntuaciones.

Esquemas:

- **catalog:** Información de las películas y sus géneros.
- **people:** Información de actores, directores y sus relaciones con películas.
- **users:** Información de usuarios y sus reseñas.

Entidades principales:

- **Movie** (catalog.movies)
- **Genre** (catalog.genres)
- **Director** (people.directors)
- **Actor** (people.actors)
- **User** (users.users)
- **Review** (users.reviews)

Tablas Intermedias:

- **movie_genres** → relación entre películas y géneros.
- **movie_directors** → relación entre películas y directores.
- **movie_actors** → relación entre películas y actores (incluye el campo role).

Relaciones:

Entidad	Esquema	Entidad Relacionada	Tipo de Relación
movies	genres	movie_genres	N:M (Muchos a muchos)
movies	directors	movie_directors	N:M (Muchos a muchos)
Movies	actors	movie_actors	N:M (Muchos a muchos)
Movies	users	reviews	1:N (una película puede tener muchas reseñas)
Users	Movies	reviews	1:N (un usuario puede reseñar muchas películas, pero una sola vez por película gracias al índice único)

Razones y Ventajas

1. Películas (movies) como entidad central

Razón:

- La película es el nodo principal desde el cual se conectan todos los demás elementos: géneros, actores, directores y reseñas.

Ventaja:

- Permite estructurar el sistema con una visión centrada en el contenido, facilitando consultas como “mostrar toda la información de una película”.

2. Relaciones muchos a muchos con tablas intermedias (movie_genres, movie_actors, movie_directors)

Razón:

- Una película puede tener varios géneros, directores y actores, y viceversa.
- Estas relaciones no pueden modelarse directamente con claves foráneas simples, por lo que se usan tablas intermedias con claves primarias compuestas.

Ventajas:

- Mantiene la base de datos normalizada (sin redundancia).
- Permite agregar fácilmente más atributos en el futuro, como character_name en movie_actors.
- Mejora el rendimiento en operaciones de mantenimiento y control de integridad.

3. Esquema people para actores y directores

Razón:

- Los actores y directores son tipos de personas que tienen relaciones específicas con las películas.
- Separarlos permite tener campos adecuados para ambos sin mezclar sus roles.

Ventajas:

- Claridad semántica.
- Permite reutilización y consultas específicas por rol.
- Facilita una futura unificación si se desea unificar “persona” como una entidad genérica.

4. Esquema users y entidad reviews

Razón:

- Los usuarios interactúan con el sistema dejando reseñas y puntuaciones.
- Se usa una tabla reviews conectada a movies y users, con una restricción única para evitar duplicados.

Ventajas:

- Permite registrar todas las opiniones sin ambigüedades.
- El índice compuesto (user_id, movie_id) garantiza integridad lógica (una reseña por película por usuario).
- Flexible para añadir más métricas como “likes” o respuestas a reseñas.

5. División en esquemas (catalog, people, users)

Razón:

- Se agrupan entidades por dominio lógico y responsabilidad.

Ventajas:

- Organización clara del sistema.
- Posibilidad de aplicar roles y permisos específicos por esquema (seguridad).
- Escalabilidad modular: puedes extender un esquema sin afectar los otros.

