

Film Management Systém – Databázový projekt

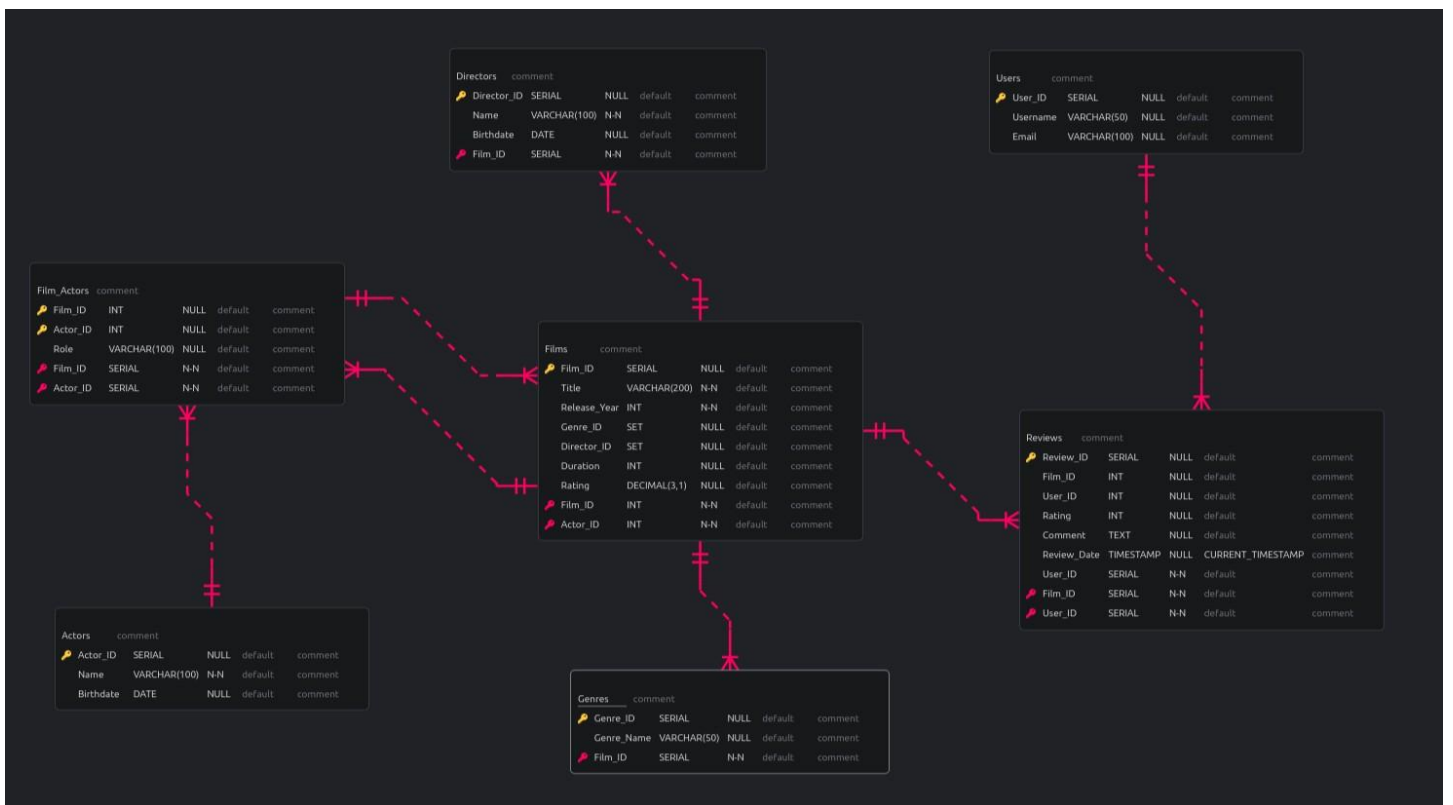
Projekt vytvořil Matyáš Dvořák

Tento dokument obsahuje kompletní dokumentaci databázového projektu "Film Management Systém". Cílem projektu bylo navrhnout, implementovat a naplnit relační databázi pro správu filmu, jejich režiséru, herců a uživatelských recenzí.

Součástí projektu je:

- fyzicky návrh databáze (SQL struktura tabulek)
- naplnění daty (minimálně 20 záznam/tabulku v průměru)
- návrh ERD (Entity-Relationship Diagram)
- ukázka práce s daty (SQL dotazy)

ERD – Entity Relationship Diagram



Popis tabulek a relaci

TABULKY V DATABAZI

1. Genres

- Sloupce:

- Genre_ID (SERIAL, PK): unikátní identifikátor žánru

- Genre_Name (VARCHAR): název žánru

- Využití: Tabulka obsahuje základní výčet žánru (např. Sci-Fi, Thriller).

2. Directors

- Sloupce:

- Director_ID (SERIAL, PK): unikátní ID režiséra

- Name (VARCHAR): cele jméno režiséra

- Birthdate (DATE): datum narození

- Využití: Obsahuje základní údaje o režisérech, např. Christopher Nolan.

3. Actors

- Sloupce:

- Actor_ID (SERIAL, PK): unikátní ID herce

- Name (VARCHAR): jméno herce

- Birthdate (DATE): datum narození

- využití: Uchovává informace o všech hercích, kteří hráli v uložených filmech.

4. Users

- Sloupce:

- User_ID (SERIAL, PK): ID uživatele

- Username (VARCHAR): přihlašovací jméno

- Email (VARCHAR): emailová adresa

- Využití: Základní informace o uživateli, kteří vkládají recenze.

5. Films

- Sloupce:

- Film_ID (SERIAL, PK): unikátní ID filmu
- Title (VARCHAR): název filmu
- Release_Year (INT): rok vydání
- Genre_ID (FK): odkaz na žánr
- Director_ID (FK): odkaz na režiséra
- Duration (INT): délka filmu v minutách
- Rating (DECIMAL): hodnocení filmu (0.0-10.0)
- Vztahy: N:N s herci, 1:N s recenzemi, N:1 na žánr a režiséra.

6. Film_Actors

- Sloupce:

- Film_ID (FK): odkaz na film
- Actor_ID (FK): odkaz na herce
- Role (VARCHAR): jméno postavy
- Vztah M:N: každý film může mít více hercu, každý herec může hrát ve více filmech.
- Kompozitní primární klíč: (Film_ID, Actor_ID)

7. Reviews

- Sloupce:

- Review_ID (SERIAL, PK): ID recenze
- Film_ID (FK): odkaz na hodnocený film
- User_ID (FK): autor recenze
- Rating (INT): hodnocení (1-10)
- Comment (TEXT): slovní komentář
- Review_Date (TIMESTAMP): datum recenze (výchozí = aktuální)
- Vztah: 1 film může mít více recenzí, každá recenze patří jednomu uživateli.

POZNAMKA K IMPLEMENTACI

- Cizí klíče jsou v případě smazání roseny pomocí CASCADE nebo SET NULL, což umožňuje logické čištění bez porušení integrity.
- `CHECK` constraints zajišťují, že délka a hodnocení jsou v rozumných mezích.

SQL dotazy a jejich využití

SQL DOTAZY A JEJICH VYSVETLENÍ

1. Filmy s žánrem a režisérem

```
-----  
SELECT Films.Title, Genres.Genre_Name, Directors.Name AS Director  
FROM Films  
INNER JOIN Genres ON Films.Genre_ID = Genres.Genre_ID  
INNER JOIN Directors ON Films.Director_ID = Directors.Director_ID;
```

Spojuje tabulku Films s Genres a Directors. Výstupem je přehled všech filmu s jejich žánrem a režisérem.

2. Filmy s herci a jejich roli

```
-----  
SELECT Films.Title, Actors.Name AS Actor, Film_Actors.Role  
FROM Film_Actors  
INNER JOIN Films ON Film_Actors.Film_ID = Films.Film_ID  
INNER JOIN Actors ON Film_Actors.Actor_ID = Actors.Actor_ID;
```

Ukazuje herce přiřazené ke každému filmu pomocí vazební tabulky Film_Actors.

3. Recenze s uživatelským jménem a filmem

```
-----  
SELECT Users.Username, Films.Title, Reviews.Rating, Reviews.Comment  
FROM Reviews  
INNER JOIN Users ON Reviews.User_ID = Users.User_ID  
INNER JOIN Films ON Reviews.Film_ID = Films.Film_ID;
```

Výstupem je seznam recenzí s informací, kdo a k jakému filmu ji napsal.

4. Všechny filmy i bez recenzi

```
SELECT Films.Title, Reviews.Rating, Reviews.Comment
FROM Films
LEFT JOIN Reviews ON Films.Film_ID = Reviews.Film_ID;
```

Ukazuje i filmy, které žádnou recenzi zatím nemají.

5. Filmy Martina Scorseseho dle hodnocení

```
SELECT Films.Title, Films.Release_Year, Films.Rating
FROM Films
INNER JOIN Directors ON Films.Director_ID = Directors.Director_ID
WHERE Directors.Name = 'Martin Scorsese'
ORDER BY Films.Rating DESC;
```

Vyber filmu jednoho režiséra, seřazený od nejlepšího hodnocení.

6. Počet filmu v žánrech s alespoň 5 filmy

```
SELECT Genres.Genre_Name, COUNT(Films.Film_ID) AS Film_Count
FROM Films
INNER JOIN Genres ON Films.Genre_ID = Genres.Genre_ID
GROUP BY Genres.Genre_Name
HAVING COUNT(Films.Film_ID) >= 5
ORDER BY Film_Count DESC;
```

Agregační dotaz, který ukáže jen ty žánry, kde je alespoň 5 filmu.

7. Vytvoření pohledu pro recenze

```
CREATE VIEW user_reviews_summary AS
SELECT
    Users.Username,
    Films.Title AS Film_Title,
    Reviews.Rating,
```

Reviews.Comment,

Reviews.Review_Date

FROM Reviews

INNER JOIN Users ON Reviews.User_ID = Users.User_ID

INNER JOIN Films ON Reviews.Film_ID = Films.Film_ID;

Uloženy pohled (viješ), který lze opakovaně dotazovat např. pres:

```
SELECT * FROM user_reviews_summary;
```