

IMDB feladat dokumentációja

Adatbázisok gyakorlat 2023/24/1

Borus Benedek István

2023. november 26.

1. Specifikáció

Egy olyan alkalmazás, amely filmeket, sorozatokat és színészeket tart nyilván. Az alkalmazásba lehet regisztrálni. A belépett felhasználó értékelést adhat a filmekhez és sorozatokhoz. Továbbá a belépett felhasználók képesek új filmek és sorozatok felvételére is.

Tárolt adatok (nem feltétlen jelentenek önálló táblákat):

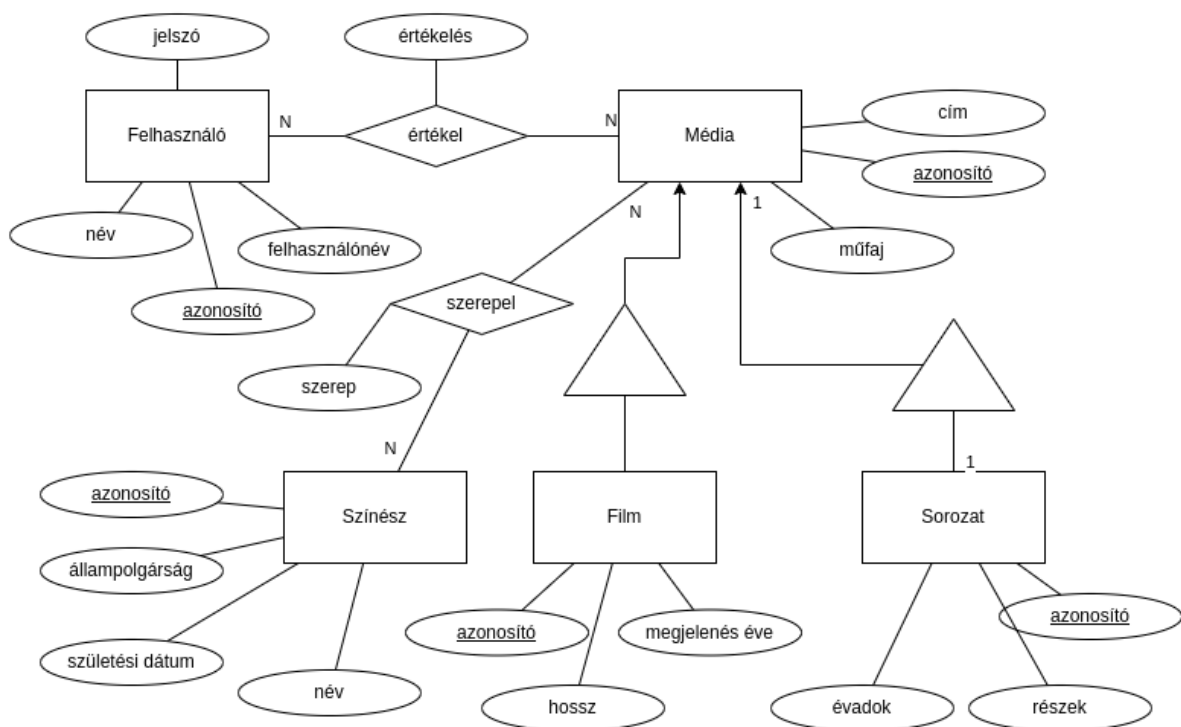
- Felhasználó: felhasználó név, jelszó, név
- Filmek: cím, játékidő, műfaj, megjelenés éve, értékelés
- Színészek: születési dátum, név, állampolgárság
- Színészek: születési dátum, név, állampolgárság

1.1. Reláció az adatok között

Egy színész több filmben/sorozatban is szerepelhet. Egy színész egy filmben/sorozatban több szerepet is játszhat.

2. Kivitelezés

2.1. E-K diagram



1. ábra. E-K diagram

2.2. Relációséma leképezés

Felhasználó(azonosító, név, jelszó, felhasználónév)

Média(azonosító, cím, *műfaj* azonosító, *film* azonosító, *sorozat* azonosító)

Film(azonosító, hossz, megjelenés éve)

Sorozat(azonosító, részek, évadok)

Színész(azonosító, *ország* azonosító, születési dátum, név)

Szerep(azonosító, *színész* azonosító, *média* azonosító, szerep neve)

Értékelés(*felhasználó* azonosító, *média* azonosító, értékelés)

Ország(azonosító, ország neve)

Műfaj(azonosító, műfaj neve)

A leképzett relációséma már 3. normálformában van, mert a Szerepek kivételével mindegyik egyednek egyéni azonosítója van. A Szerepek esetében minden tulajdonság az elsődleges kulcstól függ, így nem sérti a 3NF feltételeit.

Az Ország és Műfaj azért került külön sémába, mert így külső kulcsos megszorításokkal ellenőrizhetjük, hogy milyen értékek kerülhetnek a *műfaj*, illetve *ország* mezőkbe.

2.3. Táblatervek

A megvalósítás során angol neveket használtam a táblákban. Azonosítónak UUIDv4 típusú globálisan egyedi azonosítókat használtam. Ugyan ez a program nem biztonság kritikus, jó szokás globálisan egyedi azonosítókat használni egymás után következő egész számok helyett.

Az `actor_media` táblán azért szükséges az egyedi azonosító, hogy egy színész több szerepet is betölthessen egy filmben.

2.3.1. media (Média)

Megnevezés	Típus	Megjegyzés
<u>id</u>	VARCHAR(36)	azonosító
title	VARCHAR(100)	cím
genre	VARCHAR(36)	műfaj azonosító
movie	VARCHAR(36)	film azonosító
series	VARCHAR(36)	sorozat azonosító

2.3.2. series (Sorozat)

Megnevezés	Típus	Megjegyzés
<u>id</u>	VARCHAR(36)	azonosító
seasons	INTEGER	évadok
episodes	INTEGER	részek

2.3.3. movies (Film)

Megnevezés	Típus	Megjegyzés
<u>id</u>	VARCHAR(36)	azonosító
length	INTEGER	hossz percben
released	INTEGER	megjelenés éve

2.3.4. genres (Műfaj)

Megnevezés	Típus	Megjegyzés
<u>id</u>	VARCHAR(36)	azonosító
name	TEXT	név

2.3.5. actors (Színész)

Megnevezés	Típus	Megjegyzés
<u>id</u>	VARCHAR(36)	azonosító
name	TEXT	teljes név
birthday	DATE	születési dátum
country	VARCHAR(2)	állampolgárság országcódja

2.3.6. countries (Ország)

Megnevezés	Típus	Megjegyzés
<u>id</u>	VARCHAR(2)	két betűs országcód
name	TEXT	név

2.3.7. users (Felhasználó)

Megnevezés	Típus	Megjegyzés
<u>id</u>	VARCHAR(36)	azonosító
name	TEXT	teljes név
username	VARCHAR(100)	felhasználónév
password	VARCHAR(72)	jelszó bcrypt hash-e

2.3.8. ratings (Értékelés)

Megnevezés	Típus	Megjegyzés
<i>media</i>	VARCHAR(36)	média azonosító
<i>user</i>	VARCHAR(36)	felhasználó azonosító
rating	INTEGER	értékelés 1-100-ig

2.3.9. actor_media (Szerep)

Megnevezés	Típus	Megjegyzés
<u>id</u>	INTEGER	azonosító
<i>actor</i>	VARCHAR(36)	színész azonosító
<i>media</i>	VARCHAR(36)	média azonosító
role	TEXT	betöltött szerep

2.4. Megvalósított funkciók

2.4.1. Főoldal

Megvalósítás helye: `index.php`

A főoldalon egy keresés mező van, amivel értelemszerűen lehet a sorozatok és filmek között keresni. Ha a mezőt üresen hagyjuk, akkor az összes tárolt adatot megkapjuk.

A főoldalon találhatóak továbbá az "érdekesség" jellegű lekérdezések is.

Legmagasabban értékelt filmek 2019 után

A specifikációban sorozatok szerepeltek, de sorozatokról nem tároljuk a megjelenés évét, így filmekkel valósítottam meg. A háttérben futó lekérdezés:

index.php:12

```
SELECT
    m.id as id, m.title as title, g.name as genre,
    m2.length as length, m2.released as released,
    AVG(r.rating) as avg_rating, COUNT(r.rating) as ratings
FROM `media` m
INNER JOIN `movies` m2 ON m.movie = m2.id
LEFT JOIN `ratings` r ON m.id = r.media
LEFT JOIN `genres` g ON m.genre = g.id
WHERE m2.released > 2019
GROUP BY m.id
ORDER BY avg_rating DESC
LIMIT 5;
```

Kiadott filmek száma évekre lebontva

Ennél a lekérdezésnél egy olyan trükköt alkalmaztam, hogy legeneráltam a számok listáját 2000 és a jelenlegi év között azáltal, hogy egy segédváltozóval a media tábla minden sorához egy – az előzőnél eggyel nagyobb – számot rendeltem. Utána ezt illesztettem a kiadott filmek számához, évenkénti csoportosítással.

index.php:29

```
SELECT y.value as `year`, COUNT(m.id) as `count` FROM `media` m
INNER JOIN `movies` m2 ON m.movie = m2.id
RIGHT JOIN
    (SELECT (@val := @val + 1) - 1+2000 AS value
     FROM media, (SELECT @val := 0) AS tt) y
    ON y.value = m2.released
WHERE y.value BETWEEN 2000 AND year(curdate())
GROUP BY y.value
ORDER BY y.value DESC;
```

Legtöbb filmben szereplő színész sorozatai

Egy al-lekérdezéssel megállapítom a legmagasabban értékelt színészt, amit utána szűrőként használok az érintett sorozatok lekérdezésében.

index.php:61

```
SELECT
    m.id as id, m.title as title, g.name as genre,
    s.episodes as episodes, s.seasons as seasons,
    am.role as role,
    AVG(r.rating) as avg_rating, COUNT(r.rating) as ratings
```

```

FROM actor_media am
LEFT JOIN `media` m ON m.id = am.media
INNER JOIN `series` s ON s.id = m.series
LEFT JOIN `ratings` r ON m.id = r.media
LEFT JOIN `genres` g ON m.genre = g.id
WHERE am.actor = (
    SELECT a.id FROM `actors` a
    LEFT JOIN actor_media am ON a.id = am.actor
    LEFT JOIN `media` m ON am.media = m.id
    WHERE m.movie IS NOT NULL
    GROUP BY a.id
    ORDER BY COUNT(am.media) DESC
    LIMIT 1
)
GROUP BY am.id
ORDER BY avg_rating DESC;

```

Színészek legjobb filmjei

Ehhez a lekérdezéshez egy segéd nézetet hoztam létre, hogy áttekinthetőbb legyen a kód. A nézet kilistázza az összes színészt az összes filmmel és a filmek összesített értékeiséekkel. Utána az alábbi lekérdezés ezt tovább szűri azáltal, hogy összeilleszti a nézetet önmagával és kiválasztja azokat az eseteket, amikor nincs jobb film (tehát megtaláltuk a legjobb filmet).

index.php:106

```

SELECT d1.* FROM
movies_of_actors as d1
LEFT JOIN movies_of_actors d2
ON d1.a_id = d2.a_id AND d1.avg_rating < d2.avg_rating
WHERE d2.avg_rating IS NULL
ORDER BY `d1`.`actor` ASC;

```

movies_of_actors VIEW:

```

SELECT
    `m`.`id` AS `id`, `m`.`title` AS `title`,
    `g`.`name` AS `genre`, `m2`.`length` AS `length`,
    `m2`.`released` AS `released`, `am`.`role` AS `role`,
    `a`.`name` AS `actor`, AVG(`r`.`rating`) AS `avg_rating`,
    COUNT(`r`.`rating`) AS `ratings`, `a`.`id` AS `a_id`
FROM `actor_media` `am`
LEFT JOIN `actors` `a` ON `a`.`id` = `am`.`actor`
LEFT JOIN `media` `m` ON `m`.`id` = `am`.`media`
INNER JOIN `movies` `m2` ON `m2`.`id` = `m`.`movie`
LEFT JOIN `ratings` `r` ON `m`.`id` = `r`.`media`
LEFT JOIN `genres` `g` ON `m`.`genre` = `g`.`id`
GROUP BY `a`.`id`, `m`.`id`
ORDER BY `a`.`name` ASC;

```

2.4.2. Egyéb funkciók

- Felhasználó tud regisztrálni és bejelentkezni
- Filmek és sorozatok külön-külön és egyben is kereshetőek és listázhatóak
- Filmek részletei megtekinthetőek egy részletező oldalon
- Filmek értékelhetőek, az értékelések összegezve és részletezve is megtekinthetőek
- Film és sorozat hozzáadható, szerkeszthető és törölhető

Szereplők módosításakor a mentés gomb használata után megjelenik alul egy új mező, ahova további színészeket vehetünk fel. Ha egy színészt eltávolítanánk egy filmről vagy sorozatról, akkor a neve helyett válasszuk az üres mezőt.

- Színészek listázhatóak
- Színész hozzáadható, szerkeszthető és törölhető

2.5. Futtatás

A programhoz tartozik egy `docker-compose.yml` fájl, amivel gyorsan elindíthatók a szükséges szoftverek. A `localhost:8081` címen fut a phpMyAdmin, ahol a csatolt `imdb.sql` fájlt beimportálhatjuk. Ebben a fájlban szerepelnek a táblák, nézet és adatok felviteléhez szükséges SQL parancsok. Importálás után a `localhost:80` címen érjük el a webalkalmazást. Alapértelmezetten létezik egy felhasználó `admin` felhasználónévvel és jelszóval.