# Backend programozás – Friends

Az alábbi feladatban a Jóbarátok (Friends) sorozat adatait kezelő alkalmazás backend szerverét kell elkészítenie.

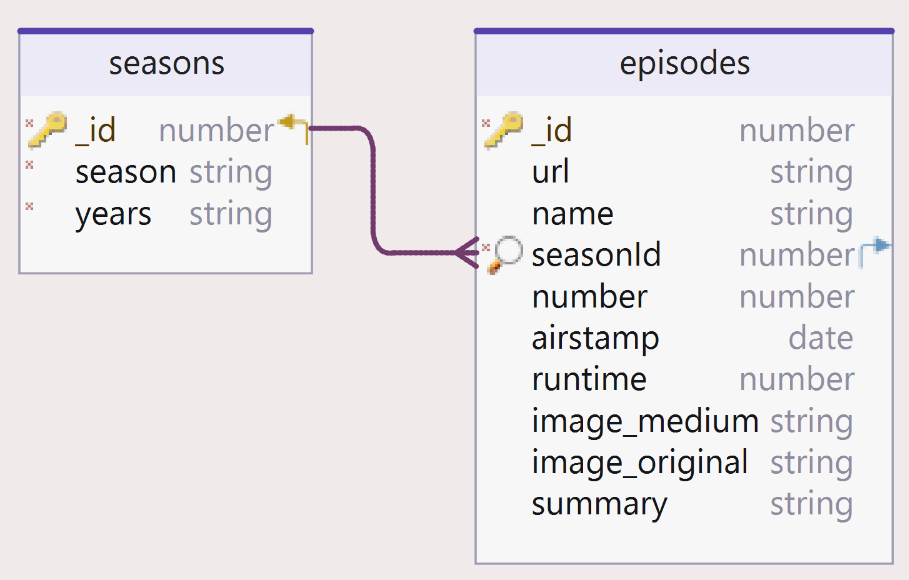
1. Hozzon létre backend szerver projektet az Ön által választott programnyelven, illetve fejlesztési környezetben! A projektmappát „Vezetéknév\_Keresztnév\_backend” formában nevezze el!
2. Készítsen adatbázist friends néven!
3. A lenti ábrák a feladathoz tartozó adatbázis tábláit (kollekcióit), azok mezőit, a mezők típusait mutatják a választott technológiától függően. Ezek alapján:
   1. hozzon létre modelleket a megfelelő adattípusokkal és beállításokkal!
   2. az adatbázisban hozza létre a két táblát!

Relációs modell:

A képen szöveg, képernyőkép, szám, szoftver látható

Automatikusan generált leírás

NoSQL adatbázis modell:



1. Alapértelmezett értékek:

**episodes tábla:**

image\_medium-> „<https://bgs.jedlik.eu/no_image.png>”

image\_original-> „<https://bgs.jedlik.eu/no_image.png>”

seasonId -> legyen indexelve

A létrehozott végpontok teszteléséhez – a választott technológiától függően – a *episodes* táblához tartozó *episodes.[csv/json]* és a *seasons* táblához a *seasons.[csv/json]* fájlokban található minta adatokat használhatja.

**Figyelj rá, hogy az 1. feladatrész hasonló nevű fájljai itt nem használhatóak, más szerkezettel és tartalommal bírnak.**

A további feladatok megoldása során ügyeljen arra, hogy a megoldása az elvárt végpontokon kívül egyéb végpontokat ne tartalmazzon! A végpontok funkcionalitását megvalósító kontroller(ek)ben a szükséges metódusokon kívül egyéb (felesleges) kódokat ne hozzon létre!

1. Kérdezze le egy adott évadhoz tartozó epizódok adatait! A válasz tartalma a lenti mintáktól eltérhet a választott technológiának megfelelően, de a mintákban szereplő összes adatot tartalmaznia kell.Ha az évad megnevezése nem létezik az adatbázisban, akkor „*Hibás évad!*” üzenettel és **404 NOT FOUND** státuszkóddal térjen vissza!

|  |  |  |  |
| --- | --- | --- | --- |
| Metódus | URL | Body | Válasz |
| *GET* | /api/search-by-season/{season} *pl.: /api/search-by-*season*/*fourth | üres | JSON array/szöveg |

Lehetséges válasz üzenet (**200 OK** státuszkóddal):

[

{

"id": 719,

"url": "http://www.tvmaze.com/episodes/40719/friends-4x01-the-one-with-the-jellyfish",

"name": "The One With the Jellyfish",

"seasonId": 4,

"number": 1,

"airstamp": "1997-09-25T20:00:00-04:00",

"runtime": 30,

"image\_medium": "http://static.tvmaze.com/uploads/images/medium\_landscape/45/114628.jpg",

"image\_original": "http://static.tvmaze.com/uploads/images/original\_untouched/45/114628.jpg",

"summary": "At the beach house, Ross and Rachel appear on the threshold of a new \r\nera; Phoebe gets to know her birth mother (Teri Garr); Monica is stung \r\nby a jellyfish."

},

{

"id": 720,

"url": "http://www.tvmaze.com/episodes/40720/friends-4x02-the-one-with-the-cat",

"name": "The One With the Cat",

"seasonId": 4,

…

]

Lehetséges válasz üzenet (**404 NOT FOUND** státuszkóddal):

{

"message": " Hibás évad!"

}

1. Új évaddal bővül a sorozat. Hozzon létre végpontot az új évad rögzítésére az alábbi beállításokkal! Sikeres rögzítés esetén **201 CREATED** státuszkóddal és az újonnan beszúrt rekord azonosítóját tartalmazó JSON üzenettel térjen vissza! Bármely kötelező mező hiánya esetén **400 BAD REQUEST** hibakóddal térjen vissza és a hibaüzenetben legyen megtalálható a „Hiányos adatok!” szöveg!

Kötelező mezők: season, years

|  |  |  |  |
| --- | --- | --- | --- |
| Metódus | URL | Body | Válasz |
| *POST* | /api/add-season | JSON | JSON/szöveg |

Példa a kérés body-ban elküldött JSON tartalmára:

{

"season":"eleven",

"years": "2023-2024"

}

Válasz üzenet (**201 CREATED** státuszkóddal):

{

"id": 11

}

Lehetséges válasz üzenet (**400 BAD REQUEST** státuszkóddal):

{

"message": "Hiányos adatok!"

}

1. Hozzon létre végpontot az adatbázisban évad törlésére az alábbi beállításokkal! Az {id} helyén a törölni kívánt évad azonosítója szerepeljen! Sikeres törlés esetén **204 NO CONTENT** státuszkóddal térjen vissza! Amennyiben a megadott id nem létezik **404 NOT FOUND** státuszkóddal és „Az évad nem létezik!” üzenettel térjen vissza! Amennyiben a megadott évad létezik, de egy vagy több rekord (epizód) hivatkozik rá az „N” oldalon, akkor ne törölje az évadot! Ebben az esetben térjen vissza a **403 FORBIDDEN** státuskóddal és „Az évad nem törölhető!” üzenettel!

|  |  |  |  |
| --- | --- | --- | --- |
| Metódus | URL | Body | Válasz |
| *DELETE* | /api/delete-season/{id} pl.: /api/delete-season/11 | üres | üres/szöveg |

Lehetséges válasz üzenet (**404 NOT FOUND** státuszkóddal):

{

"message": "Az évad nem létezik!"

}"

Lehetséges válasz üzenet (**403 FORBIDDEN** státuszkóddal):

{

"message": "Az évad nem törölhető!"

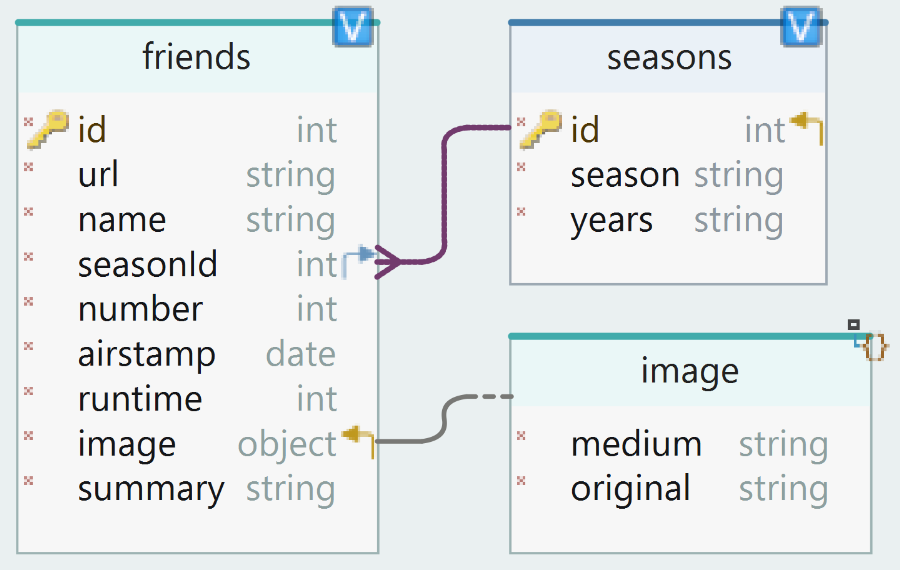
}"

1. Hozzon létre Thunder Client vagy Postman kollekciót, melyben az Ön által létrehozott összes útvonalat ellenőrzi! Exportálja a kollekciót a projekt gyökér könyvtárába “Backend\_teszt” néven!
2. A választott technológiának megfelelően adja le:
   1. a teljes projektmappát tömörítve (node\_modules könyvtár nélkül)
   2. az adatbázist létrehozó SQL scriptet vagy állományokat (db.bat és \*.json)
   3. az előző feladatban exportált “Backend\_teszt” nevű kollekciót

# Frontend programozás - Friends

A következő feladatban a népszerű Jóbarátok TV-sorozat (Friends) epizódjait bemutató alkalmazást kell készítenie. A webalkalmazást a kiadott leírás szerint kell elkészítenie, a felhasználói felület kialakítását minta segíti, melyet nem kell követnie a megoldásában.

A backend szerver által kezelt adatbázis tábláit és a táblák kapcsolatát a következő ábra szemlélteti:



A feladat elkészítése során a http://localhost:3000 címen található REST API szerverrel kell dolgoznia. A szerver indításához másolja a források mappából a Backend mappát a „D” megható főkönyvtárába, majd indítsa el az API szervert a D:\Backend\backend.bat állomány futtatásával!

A lokális szerver route-jaival teljesen megegyező online szervert a <https://friends-series.jedlik.cloud> címen érheti el. A feladatok megvalósítására ezt is használhatja.

Ebben a feladatban használandó REST API végpont:

|  |  |  |  |
| --- | --- | --- | --- |
| Metódus | URL | Body | Válasz |
| *GET* | /api/friends?name\_like={**filter**}&\_page={**page**}  &\_limit=15&\_expand=season | üres | JSON array/szöveg |

**Válasz (200-as státuszkóddal):**

A végpont azokkal a Jóbarátok epizódok adataival tér vissza, melyek címében (name mező) megtalálható a megadott {**filter**} karaktersorozat. A végpont „oldalakra tördelve” adja vissza a rekordokat, 15-15 db-ot oldalakként. Az oldal sorszámát 1-től N-ig a {**page**} paraméter határozza meg.

API válasz minta a következő végpont hívásra:

/api/friends?name\_like=**where**&\_page=**2**&\_limit=15&\_expand=season

[

{

"id": 704,

"url": "http://www.tvmaze.com/episodes...sister",

"name": "The One **Where** Chandler Can't Remember Which Sister",

"seasonId": 3,

"number": 11,

"airstamp": "1997-01-09T20:00:00-05:00",

"runtime": 30,

"image": {

"medium": "http://.../medium\_landscape/45/114588.jpg",

"original": "http://.../original\_untouched/45/114588.jpg"

},

"summary": "A drunken Chandler has a ... Rachel for a job.",

"season": {

"id": 3,

"season": "third",

"years": "1996-1997"

},

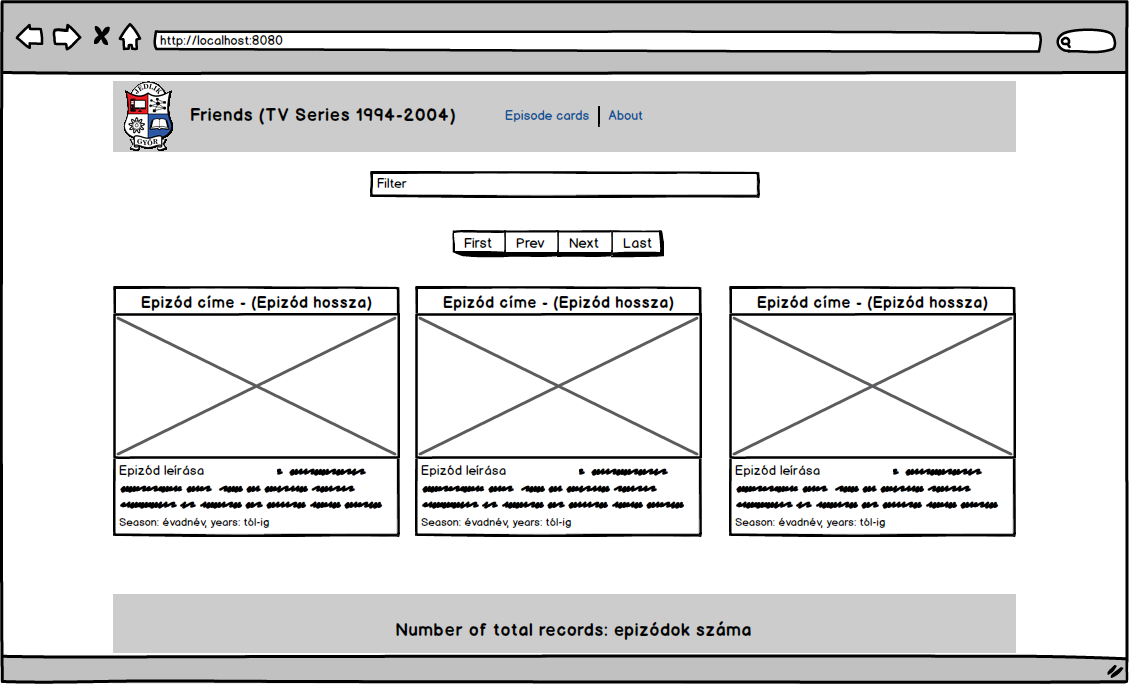
...

]

**Feladatok**

Az Ön által választott frontend keretrendszer (Angular, Vue, stb.) valamelyikével készítse el az alábbi feladatokat! A feladatok elkészítését forrásállományok és minták segítik, melyek a választott technológiának megfelelő alkönyvtárban találhatók a források mappában. A minták egy-egy lehetséges megoldást mutatnak be, pontozásra a kért funkciók megvalósítása kerül.

1. Hozzon létre frontend projektjének Vezetéknév\_Keresztnév\_technológia mappát!
2. Az alábbi minta és a kiadott források segítségével építse fel az alkalmazás menü- és állapotsorát!



A feladat vue.js technológiával történő megvalósítását megnézheti a vue-minta1.jpg és a vue-minta2.jpg állományokban!

1. Oldja meg, hogy az „*Episode cards*” és az „*About*” menüpontok kiválasztásával a megfelelő oldalakat töltse be (routolja) az alkalmazása! Az alkalmazás indulásakor az „*Episode cards*” oldal jelenjen meg automatikusan!
2. Az „*Episode cards*” oldalon a beviteli mező alapértelmezett értéke a „**where**” szó legyen! Ebben a beviteli mezőben adjuk meg az epizód címére (name) vonatkozó szűrési feltételt.
3. Az „*Episode cards*” oldal betöltésekor jelenítse meg a „where” szóra szűrés esetén megjelenő első 15 epizódot. A backend hívásához az alábbi url használható: /api/friends?name\_like=where&\_page=1&\_limit=15&\_expand=season

Az url megtalálható az api\_végpont.txt állományban is.

Megjelenítendő mezők: name, runtime, image.medium, summary, season.season, season.years

1. A beviteli mező értékének megváltozásakor az adatokat újra kérdezze le az API végpontról, majd jelenítse meg az érkező epizódok adatait az oldalon!
2. Jelenítse meg a szűrési feltételnek megfelelő rekordok számát az alkalmazás láblécében!
3. Oldja meg, ha az „*About*” oldalra navigálunk, akkor ne jelenjen meg semmi az állapotsorban!
4. A kiadott mintában szereplő parancsgombok, vagy tetszőlegesen választott lapozó-komponens használatával valósítson meg „lapozás” funkciót! Lehessen az első, az előző, a következő és az utolsó oldalakra lapozni! A megvalósításához használja az url \_page paraméterét! Megoldása kevesebb pontot ér, ha a teljes adathalmazt letöltve csak kliens oldalon valósítja meg a lapozást.
5. Oldja meg, hogy az első és utolsó oldal kiválasztása esetén a megfelelő, lapozást vezérlő elemek (parancsgombok) kerüljenek inaktív állapotba!
6. Forráskódját a választott technológiának megfelelően adja le:
   1. Tömörítés előtt törölje a felesleges állományokat! **Ügyeljen arra, hogy feladatmegoldást tartalmazó mappát/állományt ne töröljön!**
   2. A forráskódját tömörítse be Vezetéknév\_Keresztnév\_technológia.zip néven!