Adatbázis: drinkisland

Táblák:

* Alapanyagok: Az alapanyagok nevét és mennyiségének mértékegységét tárolja.
* Italok: A koktélok nevét tárolja.
* Posts: A posztok írójának id-jét, a poszt címét és a szövegét tárolja.
* Receptek: Az italokhoz a recepteket tárolja.
* Tipus: Az italok típusát tárolja.
* Users: A felhasználók adatait tárolja.

alapanyagok:

* id integer
* name varchar(50)
* unit varchar(50)

italok:

* id integer
* name varchar(50)

posts:

* id integer
* user\_id integer
* title varchar(200)
* text varchar(2000)

receptek:

* id integer
* ital\_id integer
* alapanyag\_id integer
* amount integer

tipus:

* id integer
* ital\_id integer
* teli tinyint
* nyari tinyint
* edes tinyint
* savanyu tinyint
* keseru tinyint
* alkoholos tinyint
* alkoholmentes tinyint

users:

* id integer
* username varchar(255)
* email varchar(255)
* password varchar(255)
* remember\_token varchar(100)
* created\_at timestamp
* updated\_at timestamp
* login\_attempts integer
* banned\_time timestamp
* is\_admin tinyint

Kapcsolatok:

* A posts tábla user\_id mezője a users tábla id mezőjére hivatkozik.
* A receptek tábla ital\_id mezője az italok tábla id mezőjére hivatkozik.
* A receptek tábla alapanyag\_id mezője az alapanyagok tábla id mezőjére hivatkozik.
* A tipus tábla ital\_id mezője az italok tábla id mezőjére hivatkozik.

Kontrollerek:

* Auth/AuthController
* AlapanyagController
* ItalController
* KoktelController
* PostController
* ProfileController
* ReceptController
* TipusController

Eljárások:

AuthController:

register():

A ”/register” végponton érhető el, felhasználók regisztálását segítő eljárás. A felhasználó adatait a users tábába menti el. Paraméternek egy Request-et vár.

Request adatok:

* username varchar(255)
* email varchar(255)
* password varchar(255)
* password\_confirmation varchar(255)

Hogyha nem létezik ilyen felhasználónevű user, az e-mail cím megfelelő formátumú és a jelszók megegyeznek, akkor felveszi a felhasználót a users táblába.

login()

A ”/login” végponton érhető el, felhasználók bejelentkezését segítő eljárás. Paraméternek egy Request-et vár.

Request adatok:

* username varchar(255)
* password varchar(255)

Hogyha létezik ilyen felhasználónevű user, és a jelszó megfelelő, akkor belép a felhasználóval, és generál egy tokent a felhasználónak, amivel azonosíthatja magát, és visszaadja ezt.

logout()

A ”/ logout” végponton érhető el, hogyha a felhasználó be van jelentkezve. A felhasználók kijelentkezését segítő eljárás. Paraméternek egy Request-et vár, amiben a bejelentkezéskor kapott tokent várja.

Futtatásakor, ha a felhasználó megfelelő tokent ad vissza akkor kijelentkezik a felhasználó és törli ezt a tokent.

ProfileController:

getUserProfileData()

A /getUserProfileData végponton érhető el akkor, hogyha egy felhasználó be van jelentkezve. A felhasználó felhasználónevét adja vissza.

getUsers()

A /getUsers végponton érhető el, hogyha egy admin van bejelentkezve.

Ha meghívjuk, akkor visszatér az összes felhasználó összes adatával a users táblából.

setNewPassword()

A /setNewPassword végpontról érhető el, ha a felhasználó be van jelentkezve.

Request adatok:

* password varchar(255)
* password\_confirmation varchar(255)

Ha meghívjuk, és átadjuk a bejelentkezéskor kapott tokenünket, akkor megváltoztatja a felhasználó jelszavát.

setAdmin()

A /setAdmin végponton érhető el, ha adminként van a felhasználó bejelentkezve, és bemenetként egy Requestet vár.

Request adatok:

* username varchar(255)

Ha a username nevű felhasználó nem admin, akkor admin jogot ad neki, ha pedig admin, akkor elveszi azt.

deleteAccount()

A /deleteAccount végponton érthető el, ha a felhasználó be van jelentkezve.

Ha meghívja a felhasználó, és megfelelő tokent ad át, akkor törli a felhasználót és a felhasználóhoz tartozó posztokat.

AlapanyagController:

addAlapanyag()

A /addAlapanyag végponton érhető el, ha admin van bejelentkezve, és egy Requestet vár.

Request adatok:

* name varchar(50)
* unit varchar(50)

Ha olyan alapanyagnevet kapunk, ami még nem létezik, akkor az alapanyagok táblához hozzáad egy újabb alapanyagot és a mértékegységét.

getAlapanyagByName()

A /getAlapanyagByName végponton érjük el, és egy Requestet vár.

Request adatok:

* name varchar(50)

Ha létező nevet adunk meg, akkor visszaadja az ahhoz az névhez tartozó alapanyagot.

getAlapanyagById()

A /getAlapanyagById végponton érjük el, és egy Requestet vár.

Request adatok:

* id integer

Ha létező id-t adunk meg, akkor visszaadja az ahhoz az id-hez tartozó alapanyagot.

getAlapanyagok()

A /getAlapanyagok végponton érjük el.

Visszaadja az összes alapanyagot.

modifyAlapanyag()

A /modifyAlapanyag végponton érjük el, és egy Requestet vár, ahol id és egy name attribútumokat várja, ahol a name az új név, az unit pedig az új mértékegység.

Request adatok:

* id integer
* name varchar(50)
* unit varchar(50)

Ha létező id-t adunk meg, és nincs még olyan név, amire meg szeretnénk változtatni, akkor megváltoztatja az alapanyagok táblában az adatokat.

deleteAlapanyag()

A /deleteAlapanyag végponton érjük el, ha adminként vagyunk bejelentkezve, és egy Requestet vár.

Request adatok:

* id integer

Ha létezik ilyen id-jű alapanyag, és egyik receptben sincs összetevőként bent, akkor törli az alapanyagot. Azt, hogy benne van-e egy receptben a Receptkontroller getReceptByAlapanyagId() metódussal lehet lekérdezni, ami egy id-t kér Request-ként.

ItalController

addItal()

A /addItal végponton lehet elérni, ha admin van bejelentkezve a felhasználó és egy Requestet vár.

Request adatok:

* name varchar(50)

Ha olyan italnevet adunk, ami még nem létezik, akkor az italok táblához hozzáad egy újabb italt.

getItalByName()

A /getItalByName végponton érjük el, és egy Requestet vár.

Request adatok:

* name varchar(50)

Ha létező nevet adunk meg, akkor visszaadja az ahhoz az névhez tartozó italt.

getItalById()

A /getItalById végponton érjük el, és egy Requestet vár.

Request adatok:

* id integer

Ha létező id-t adunk meg, akkor visszaadja az ahhoz az id-hez tartozó italt.

getItalok()

A /getItalok végponton érjük el.

Visszaadja az összes italt.

modifyItal ()

A /modifyItal végponton érjük el, ha adminként vagyunk bejelentkezve, és egy Requestet vár, ahol id és egy name attribútumokat várja, ahol a name az új név.

Request adatok:

* id integer
* name varchar(50)

Ha létező id-t adunk meg, és nincs még olyan név, amire meg szeretnénk változtatni, akkor megváltoztatja az italok táblában az adatokat.

deleteItal ()

A /deleteItal végponton érjük el, ha adminként vagyunk bejelentkezve, és egy Requestet vár.

Request adatok:

* id integer

Ha létezik ilyen id-jű ital, akkor törli az italt. A receptek táblából kitörli az összes olyan receptet, ami ehhez az ital id-hez tartozik, és a tipusok táblából pedig a hozzá tartozó típust a TipusController deleteTipusByItalId() metódusával.

KoktelController

getKoktelok()

A /getKoktelok végponton érhető el, és a koktélok receptjeit adja vissza.

Egy JSON file-okat tartalmazó tömböt ad vissza, a következő példán látható módon:

[

{

"Ital": "Limonádé",

"Recept": {

"citromlé": "2 cl",

"szódavíz": "20 cl",

"Jégkocka": "6 db",

"mentalevél": "1 db",

"lime": "2 szelet",

"cukorszirup": "2 cl"

}

},

{

"Ital": "APEROL SPRITZ",

"Recept": {

"narancskarika": "1 szelet",

"szóda": "4 cl",

"prosecco": "12 cl",

"Jégkocka": "1 db"

}

}

]

searchKoktelokByName()

A /searchKoktelokByName végponton érjük el, és egy Requestet vár.

Request adatok:

* text varchar

Az előzőhöz hasonló JSON fájlokat tartalmazó tömböt ad vissza, csak azokat az italokat tartalmazva, amiknek a nevében benne van a Requestben kapott text paraméter.

getReceptByName()

A /getReceptByName végponton lehet elérni, és egy Requestre vár.

Request adatok:

* name varchar(50)

Egy JSON fájlt ad vissza, amiben annak a koktélnak a receptje van, amelyiknek a nevét a name attribútumban kapta.

getTeli()

A /getTeli végponton érhető el, és visszaadja az összes olyan koktél receptjét, melynek a típusa téli. Ezt az típusok táblából lehet lekérdezni.

Egy JSON fájlokat tartalmazó tömböt ad vissza.

getNyari()

A /getNyari végponton érhető el, és visszaadja az összes olyan koktél receptjét, melynek a típusa nyári. Ezt az típusok táblából lehet lekérdezni.

Egy JSON fájlokat tartalmazó tömböt ad vissza.

getEdes()

A /getEdes végponton érhető el, és visszaadja az összes olyan koktél receptjét, melynek a típusa édes. Ezt az típusok táblából lehet lekérdezni.

Egy JSON fájlokat tartalmazó tömböt ad vissza.

getSavanyu()

A /getSavanyu végponton érhető el, és visszaadja az összes olyan koktél receptjét, melynek a típusa savanyú. Ezt az típusok táblából lehet lekérdezni.

Egy JSON fájlokat tartalmazó tömböt ad vissza.

getKeseru()

A /getKeseru végponton érhető el, és visszaadja az összes olyan koktél receptjét, melynek a típusa keserű. Ezt az típusok táblából lehet lekérdezni.

Egy JSON fájlokat tartalmazó tömböt ad vissza.

getAlkoholos()

A /getAlkoholos végponton érhető el, és visszaadja az összes olyan koktél receptjét, melynek a típusa alkoholos. Ezt az típusok táblából lehet lekérdezni.

Egy JSON fájlokat tartalmazó tömböt ad vissza.

getAlkoholmentes()

A /getAlkoholmentes végponton érhető el, és visszaadja az összes olyan koktél receptjét, melynek a típusa alkoholmentesi. Ezt az típusok táblából lehet lekérdezni.

Egy JSON fájlokat tartalmazó tömböt ad vissza.

PostController

addPost()

A /addPost végponton lehet elérni, ha be van jelentkezve a felhasználó és egy Requestet vár.

Request adatok:

* user\_id integer
* title varchar(200)
* text varchar(2000)

A posts táblához hozzáadja az adatokat.

getPosts()

A /getPosts végponton érjük el.

Visszaadja az összes posztot.

getPostsByUserId()

A / getPostsByUserId végponton érjük el, és egy Requestet vár.

Request adatok:

* user\_id integer

Ha létező user\_id-t adunk meg, akkor visszaadja az ahhoz az id-hez tartozó posztokat.

modifyPost()

A /modifyPost végponton érjük el, ha be vagyunk jelentkezve, és egy Requestet vár.

Request adatok:

* user\_id integer
* title varchar(200)
* text varchar(2000)

Ekkor megváltoztatja a posts tábában a poszt címét és szövegét.

deletePost ()

A /deletePost végponton érjük el, ha be vagyunk jelentkezve, és egy Requestet vár.

Request adatok:

* id integer

Ha létezik ilyen id-jű poszt, akkor törli a posztot.

Ha adminként vagyunk bejelentkezve, akkor bármelyik posztot tudjuk törölni, különben csak a sajátunkat.

ReceptController

addRecept()

A /addRecept végponton lehet elérni, ha adminként van bejelentkezve a felhasználó és egy Requestet vár.

Request adatok:

* ital\_id integer
* alapanyag\_id integer
* amount integer

A receptek táblához hozzáadja az adatokat.

getReceptek()

A /getReceptek végponton érjük el.

Visszaadja az összes receptet.

getReceptByAlapanyagId()

A /getReceptByAlapanyagId végponton érjük el, és egy Requestet vár.

Request adatok:

* id integer

Visszaadja az első receptet, ahol a megadott id az alapanyag id.

modifyRecept()

A /modifyRecept végponton érjük el, ha adminként vagyunk bejelentkezve, és egy Requestet vár.

Request adatok:

* ital\_id integer
* alapanyag\_id integer
* amount integer

Ekkor megváltoztatja a receptek tábában a recept adatait.

deleteRecept()

A /deleteRecept végponton érjük el, ha adminként vagyunk bejelentkezve, és egy Requestet vár.

Request adatok:

* id integer

Ha létezik ilyen id-jű recept, akkor törli a receptet.

TipusController

addTipus()

A /addTipus végponton lehet elérni, ha adminként van bejelentkezve a felhasználó és egy Requestet vár.

Request adatok:

* ital\_id integer
* teli tinyint
* nyari tinyint
* edes tinyint
* savanyu tinyint
* keseru tinyint
* alkoholos tinyint
* alkoholmentes tinyint

A tipusok táblához hozzáadja az adatokat, ha még nem létezik ilyen ital id-hez rekord.

getTipusByItalId()

A /getTipusByItalId végponton érjük el, és egy Requestet vár.

Request adatok:

* ital\_id integer

Visszaadja az első típust, ahol a megadott ital\_id az ital\_id.

getTipusok()

A /getTipusok végponton érjük el.

Visszaadja az összes ital típusát.

modifyTipus()

A /modifyTipus végponton érjük el, ha adminként vagyunk bejelentkezve, és egy Requestet vár.

Request adatok:

* ital\_id integer
* teli tinyint
* nyari tinyint
* edes tinyint
* savanyu tinyint
* keseru tinyint
* alkoholos tinyint
* alkoholmentes tinyint

Ekkor megváltoztatja a tipus tábában a típus adatait.

deleteTipus()

A /deleteTipus végponton érjük el, ha adminként vagyunk bejelentkezve, és egy Requestet vár.

Request adatok:

* id integer

Törli a típust, ha annak az ital\_id-je az id.