**BZSH Külkereskedelmi Technikum leltár program**

**dokumentáció**

**Fejezetek:**

1. Bevezetés
2. Fejlesztői dokumentáció
3. Felhasználói dokumentáció
4. Tesztelés
5. Összefoglalás
6. **Bevezetés:**

A következő dokumentáció kívánja bemutatni a BZSH Külkereskedelmi Technikum Leltár programját, annak specifikációit, elkészültét.

A program back-end szoftverét Express.js keretrendszer szolgáltatja, MySQL adatbázis környezetben.

Az alkalmazás front-end szoftver része Angular keretrendszerben íródott.

Az alkalmazás célja, hogy egy felhasználóbarát felületet biztosítson, amely megfelel az iskola alapvető igényeinek, és támogatja a hatékony leltározást.

1. **Fejlesztői dokumentáció:**

**BACK-END**

-Fejlesztéshez használt eszközök technológiák

-Adatbázis felépítés

-Mappa struktúra

-Végpont kifejtés

-Request body

-Funkciók és szervizek:

-Tesztelés

-Fejlesztési lehetőségek

**Fejlesztéshez használt eszközök, technológiák:**

Visual Studio Code

Postman, JMETER, Swagger

DBeaver - 25.0.1

MySQL adatbázis környezet – MySQL 8.0

Express.js 4.21.2

**Adatbázis felépítés:**

**Táblák:**

**item\_name:**

A tábla célja, hogy egy listát biztosítson az iskolában fellelhető és leltárazandó tárgyakból.

Oszlopai:

ID: Azonosító a tárgyaknevekhez. – integer

ITEM: A tárgy neve. – string

**item:**

Felhasználva az item\_name táblát egyedi azonosítást biztosít minden tárgy számára így különbséget téve pl. laptop és laptop közt.

Oszlopai:

ID: Azonosító a tárgyakhoz. – integer

ITEMNAMEID: Idegenkulcs ami hivatkozza a tárgynevet. – integer

**storage\_place:**

A tábla célja, hogy egy listát biztosítson az iskolában fellelhető tárolókból.

Oszlopai:

ID: Azonosító a tárolókhoz. – integer

STORAGE: A tároló neve. – string

**storage\_conn:**

A tábla célja, hogy összeköttetést biztosítson tárgyak és tárolók között, ehhez felhasználva az item és storage\_place táblák adatait.

Oszlopai:

ID: Azonosító – integer

ITEMID: Idegenkulcs, ami hivatkozza a tárgyat. – integer

STORAGEPLACEID: Idegenkulcs, ami hivatkozza a tárolót. – integer

**user:**

Eltárolja a felhasználókat és tulajdonságait.

Oszlopai:

ID: Azonosító a felhasználókhoz. – integer

USEREMAIL: Tartalmazza a felhasználóhoz tartozó emailt. - string

USERPASSWORD: Tartalmazza a felhasználóhoz tartozó jelszót, egy hash formájában. - string

ISADMIN: Igaz/Hamis értékpár jelzi, hogy a felhasználó admin vagy felhasználói szerepkörrel rendelkezik-e. -boolean

ISFIRSTLOGIN: Igaz/Hamis értékpár, a regisztrációs műveletek egyediségéből adódóan ezen érték alapján a felhasználó jelszócseréjét kötelezi. -boolean

ISDISABLED: Igaz/Hamis értékpár jelzi, hogy a felhasználó állapota aktív vagy tiltott. – boolean

CREATEDAT: Időbélyeg, a létrehozás idejét rögzíti. – timestamp

UPDATEDAT: Időbélyeg, a felhasználóval kapcsolatos változások idejét rögzíti. – timestamp

CREATEDBY: Idegenkulcs, ami hivatkozza létrehozó személyt. – integer

**logs:**

A logs tábla rögzíti minden tárgyakkal kapcsolatos művelet egyenként történő rögzítését.

Oszlopai:

ID: Azonosító – integer

ITEMID: Idegenkulcs, ami hivatkozza a tárgyat. – integer

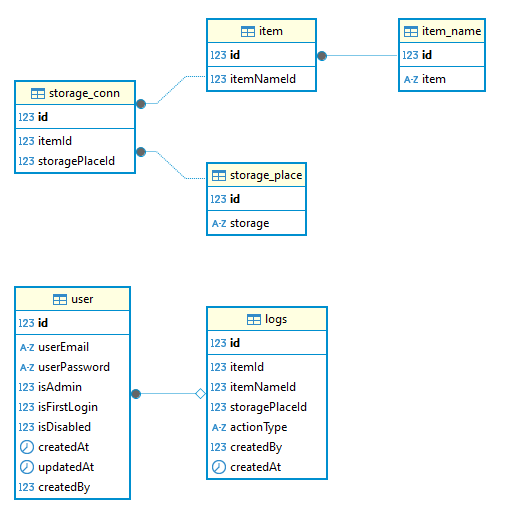
ITEMNAMEID: Idegenkulcs ami hivatkozza a tárgynevet. – integer

STORAGEPLACEID: Idegenkulcs, ami hivatkozza a tárolót. – integer

ACTIONTYPE: Enum lista ('ADD', 'UPDATE', 'DELETE'), a művelet milyenségét jelzi. – enum

CREATEDBY: Idegenkulcs, ami hivatkozza létrehozó személyt. – integer

CREATEDAT: Időbélyeg, a létrehozás idejét rögzíti. – timestamp



**Mappa struktúra**

BZSHKulkerStorage

│

└──BACKEND

│

├──associations

│ └──associations.js

│

├──models

│ ├──ItemModel.js

│ ├──ItemNameModel.js

│ ├──LogModel.js

│ ├──StorageConnModel.js

│ ├──StoragePlaceModel.js

│ └──UserModel.js

│

├──router

│ └──IeltarRouter.js

│

├──services

│ ├──auth

│ │ └──User.service.js

│ │

│ ├──log

│ │ ├──ExcelLog.service.js

│ │ └──Logs.service.js

│ ├──storage

│ │ ├──Item.service.js

│ │ ├──ItemName.service.js

│ │ ├──Storage.service.js

│ │ └──StorageConn.service.js

│ ├──Storage.service.js

│ └──StorageConn.service.js

│

├──.gitignore

├──config.js

└──index.js

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Végpont | Metódus | Leírás | Elérés |
| 1. | /storagePlace | GET | Tárolók listázása. | felhasználó |
| 2. | /storagePlace | POST | Tároló felvétele. | admin |
| 3. | /storagePlace/{id} | DELETE | Tároló törlése azonosító alapján. | admin |
| 4. | /itemName | GET | Tárgynevek listázása. | felhasználó |
| 5. | /itemName | POST | Tárgynevek felvétele. | admin |
| 6. | /itemName/{id} | DELETE | Tárgynév törlése azonosító alapján. | admin |
| 7. | /item | GET | Tárgyak listázása. | admin |
| 8. | /item | POST | Tárgyak felvétele. | felhasználó |
| 9. | /item | PUT | Tárgyak áthelyezése. | admin |
| 10. | /item | DELETE | Tárgyak törlése. | admin |
| 11. | /users | GET | Felhasználók listázása. | admin |
| 12. | /disable | PATCH | Felhasználók letiltása | admin |
| 13. | /enable | PATCH | Felhasználók feloldása. | admin |
| 14. | /demote | PATCH | Admin jog elvétele. | admin |
| 15. | /promote | PACTH | Admin jog kiosztása. | admin |
| 16. | /register | POST | Felhasználó rögzítése. | admin |
| 17. | /login | POST | Bejelentkezés. | felhasználó |
| 18. | /firstLogin | PUT | Bejelentkezés és jelszóváltás. | felhasználó |
| 19. | /passwordChange | PUT | Jelszóváltás. | felhasználó |
| 20. | /log | POST | Műveleti napló letöltés. | admin |

**Végpont kifejtés:**

1. A GET http metódusú /storagePlace végpont célja a rögzített tárolóhelyek listázása a felhasználó számára, hogy azokból választva megelőzzük a tárgyrögzítés során előforduló hibákat.
2. A POST http metódusú /storagePlace végpont célja a tárolóhelyek rögzítése.
3. A DELETE http metódusú /storagePlace/{id} végpont célja a tárolóhelyek törlése esetleges hibás rögzítés vagy tárolóhely megszűnés esetén.
4. A GET http metódusú /itemName végpont célja a rögzített tárgynevek listázása a felhasználó számára, hogy azokból választva megelőzzük a tárgyrögzítés során előforduló hibákat.
5. A POST http metódusú /itemName végpont célja a tárgynevek rögzítése.
6. A DELETE http metódusú /itemName{id} végpont célja a tárgynevek törlése esetleges hibás rögzítés vagy tárgynevek megszűnése esetén.
7. A GET http metódusú /item végpont célja a rögzített tárgyak listázása.
8. A POST http metódusú /item végpont célja új tárgyak rögzítése.
9. A PUT http metódusú /item végpont célja a rögzített tárgyak termek közti áthelyezését végzi.
10. A DELETE http metódusú /item végpont célja a már rögzített tárgyakból való törlés selejtezett tárgyak esetén.
11. A /users végpont a felhasználókat listázza.
12. A /disable végpont a felhasználók letiltását kezeli.
13. A /enable végpont a letiltott felhasználók feloldását kezeli.
14. A /demote végpont az admin felhasználók jogának elvételét kezeli.
15. A /promote végpont az felhasználók adminná promótálását kezeli.
16. A /register végpont az admin felhasználó számára nyújt lehetőséget vele egyenrangú admint vagy kevesebb jogkörrel rendelkező felhasználót felvennie. A végpont e-mail alapján küld egy random jelszót az új felhasználónak.
17. A /login végpont végzi a bejelentkeztetést.
18. A /firstLogin végpont az új felhasználók számára és elfelejtett jelszó esetén esetén kötelező jelszóváltást alkalmaz.
19. A /passwordChange végpont kezeli a jelszóváltást a felhasználóknak, e-mail alapján.
20. A /log végpont egy Excel alapú listázást végez, ami letölthetővé válik. Az Excelben rögzülnek a tárgyakhoz tartozó tulajdonságok, műveletek és a műveletet elvégző felhasználó.

**Request bodyk:**

|  |  |  |
| --- | --- | --- |
| 1. | /storagePlace | nincs |
| 2. | /storagePlace | **{**  **"storage": "Main Warehouse"**  **}** |
| 3. | /storagePlace/{id} | nincs |
| 4. | /itemName | nincs |
| 5. | /itemName | **{**  **"item": "Sample Item Name"**  **}** |
| 6. | /itemName/{id} | nincs |
| 7. | /item | nincs |
| 8. | /item | **{**  **"itemNameId": 1,**  **"storagePlaceId": 2,**  **"quantity": 3**  **}** |
| 9. | /item | **{**  **"itemIdList": [**  **1, 2, 3**  **],**  **"newStoragePlaceId": 4**  **}** |
| 10. | /item | **{**  **"itemIdList": [**  **1, 2, 3**  **]**  **}** |
| 11. | /users | nincs |
| 12. | /disable | **{**  **"userEmail": "string"**  **}** |
| 13. | /enable | **{**  **"userEmail": "string"**  **}** |
| 14. | /demote | **{**  **"userEmail": "string"**  **}** |
| 15. | /promote | **{**  **"userEmail": "string",**  **}** |
| 16. | /register | **{**  **"userEmail": "string",**  **"isAdmin": false**  **}** |
| 17. | /login | **{**  **"userEmail": "string",**  **"userPassword": "string"**  **}** |
| 18. | /firstLogin | **{**  **"userPassword": "string"**  **}** |
| 19. | /passwordChange | **{**  **"userEmail": "string"**  **}** |
| 20. | /log | **{**  "itemNameId"**: 0,**  "storagePlaceId"**: 0,**  "createdBy"**: 0,**  "fromDate"**: "string",**  "toDate"**: "string"**  **}** |

**Funkciók és szervizek:** BZSH Külkereskedelmi Technikum leltár funkciók - excel

**Tesztelés:** BZSH Külkereskedelmi Technikum leltár teszt – excel

**Fejlesztési lehetőségek:**

Felhasználói jogkörök átdolgozása. Kibővíteni a jogokat SuperAdmin Admin Felhasználó. Jobban elkülönített jogokkal jobban kialakítható lenne a felhasználók közötti hierarchia és ezáltal elkülöníthető lenne több funkció.

Felhasználókon végzett műveletek naplózása.

**FRONT-END:**

-Fejlesztéshez használt eszközök, technológiák:

-Mappa struktúra

-Komponensek

-Szervizek

-Útvonalfigyelők

-Továbbfejlesztési lehetőségek

**Fejlesztéshez használt eszközök, technológiák:**

Visual Studio Code

Frontend keretrendszer - Angular 18.2.12

Stílus formázás

CSS keretrendszer - Bootstrap 5.3.2

Angular specifikus bootstrap komponens - @ng-bootstrap/ng-bootstrap 17.0.1

Táblázat exportálás

XLSX Könyvtár

**Mappa struktúra**

src/  
├── app/  
│ ├── components/   
│ │ ├── dashboard/   
│ │ ├── login/   
│ │ ├── navbar/   
│ │ ├── password-change/   
│ │ └── user-control/   
│ │  
│ ├── guards/   
│ │ ├── admin.guard.ts  
│ │ ├── auth.guard.ts  
│ │ └── first-login.guard.ts  
│ │  
│ ├── services/   
│ │ ├── auth/   
│ │ ├── item/   
│ │ ├── log/   
│ │ ├── storage/   
│ │ └── user/   
│ │  
│ ├── app-routing.module.ts   
│ ├── app.component.ts   
│ ├── app.component.html   
│ ├── app.component.css   
│ ├── app.component.spec.ts   
│ ├── app.module.ts   
│ └── environment.ts   
│  
├── assets/   
│ ├── logo.png  
│ └── logo\_no\_text.png  
│  
├── index.html   
├── main.ts   
└── styles.css

**Komponensek**

**App (src\app\app.component.\*)**

Alkalmazás gyökér komponense, melynek feladata a navigációs sáv (app-navbar) és az aloldalak megjelenítése (router-outlet). Emelett szabályozza, hogy a navigációs sáv mikor jelenhet meg.

A konstruktorban iratkozik fel a Router eseményeire és figyeli a NavigationEnd-et. Ha az aktuális útvonal /login vagy /passwordchange a navigációs sáv nem fog megjelenni.

**Osztályváltozók:**

**showNavbar: boolean = false;** - Szabályozza a navigációs sáv megjelenítését

**Dashboard (src\app\components\dashboard)**

Szerpe a leltárba felvitt termékek, termék típusok, tárolók megjelenítése, rendezése, CRUD műveletek megvalósítása jogosultsági szinttől függően.

**Osztályváltozók:**

**items: any[] = [];** - Termékek eltárolva a loadItems metódusban generált termékkóddal

**filteredItems: any[] = [];** - Szűrt, rendezett adatok

**storagePlaces: any[] = [];** - Tárolók

**itemNames: any[] = [];** - Terméktípusok

**users: any[] = [];** - Felhasználók

**selectedItemIds: number[] = [];** - Kiválasztott termékek listája

**selectAllChecked: boolean = false;** - Összes megjelenített elem kivan-e választva

**selectedItemNames: number[] = [];** - Szűréskor kiválaszott terméktípusok listája

**selectedStoragePlaces: number[] = [];** - Szűréskor kiválasztott tárolók listája

**newStoragePlaceId: any;** - Tároló azonosítója ahova a termékek áthelyezésre kerülnek

**newStoragePlace: any;** - Új tároló létrehozásához használt változó

**newItemName: any;** - Új terméktípus létrehozásához használt változó

**currentSortColumn: string | null = "id";** - Kiválaszott rendezési fejléc

**isAscending: boolean = true;** - Növekvő/Csökkenő sorrendű rendezés

**fileName: string = '';** - Címke letöltéshez fájl név

**filterText: string = '';** - Keresőmező értéke

**alertMessage: string | null = null;** - Figyelmeztető üzenet szövege

**isError: boolean = false;** - Üzenet típusa Hiba/Siker

**timeoutId: any = null; - Üzenet automatikus eltűnés idejének újra indítására használt változó többszöri metódus híváskor**

**deleteType: 'itemName' | 'storagePlace' = 'itemName';** - Terméktípus/Tároló törlés típusa

**deleteTypeId: number | null = null;** - Törlendő adat azonosítója

**newItem: any = {**

**itemNameId: null,**

**storagePlaceId: null,**

**quantity: 0,**

**};** - Új termék adatai

**logFilters = {**

**itemNameId: undefined,**

**storagePlaceId: undefined,**

**userId: undefined,**

**createdBy: undefined,**

**fromDate: undefined,**

**toDate: undefined**

**};** - Műveleti napló szűrő adatai

**currentPage: number = 1;** - Jelenleg megjelenített oldal száma

**itemsPerPage: number = 10;** - Hány termék jelenjen meg oldalanként

**itemsPerPageOptions: number[] = [5, 10, 25, 50, 100, 250];** - Termék/Oldal opciók

**Konstruktor:**

authService – Admin jogosultság ellenőrzése

itemService - Termékek lekérdezése, létrehozása, módosítása, törlése

itemNameService - Terméktípusok kezelése

storageService - Tárolók kezelése

logService - Műveleti napló lekérése és exportálása

userService - Felhasználók kezelése (csak admin esetén) – műveleti napló szűréshez

**Metódusok:**

**Adatok betöltése:**

ngOnInit():

load metódusok meghívása

loadItems csak loadItemNames lefutása után következik be (lásd loadItemNames metódus)

loadUsers csak admin felhasználó esetén fut le

loadItems():

Termékek betöltése, eltárolása

Termék kód generálása és hozzáfűzése a már betöltött adatokhoz

loadStoragePlaces():

Tárolók betöltése, eltárolása

loadItemNames(callback?: () => void):

Terméktípusok betöltése, eltárolása

Szükséges callback használata

a termékkód generálásához a terméktípusoknak már betöltött állapotban kell lennie

ezért a loadItems metódus csak a terméktípusok betöltődése után hívódik meg

ngOnInit(): void {

    this.loadItemNames(() => {this.loadItems();});

  }

loadUsers():

Felhasználók betölése, eltárolása

Csak admin felhasználó esetén hívódik meg – get isAdmin metódussal ellenőrzi

ngOnInit(): void {

if(this.isAdmin){

      this.loadUsers();

    }

}

**Termékekhez tartozó műveletek:**

createItem():

termékek hozzáadása

updateItem(itemIdList: number[], newStoragePlaceId: number):

Kiválasztatott termékek áthelyezése másik tárolóba

deleteItem(itemIdList: number[]):

Kiválasztott elemek törlése

**Tárolóhoz tartozó műveletek:**

createStoragePlace():

Új tároló létrehozása

Ellenőrzi hogy üres/már létező tárolót ne lehessen létrehozni

deleteStoragePlace(id: number):

Tároló törlése ID alapján

**Terméktípushoz tartozó műveletek:**

createItemName():

Új terméktípus létrehozása

Ellenőrzi hogy üres vagy már létező terméktípust ne lehessen létrehozni

deleteItemName(id: number):

Terméktípus törlése ID alapján

**Szűrések:**

applyFilters():

Feladata a szűrők együttes használata

Keresés

Szűrés terméktípusra és tárolóra

Szűrés terméktípusra

deselectItemNames() - Összes kiválasztott elem kikapcsolása

toggleSelectItemName(itemNameId: number) - Elem kiválasztása egyesével

toggleSelectAllItemNames() - Összes elem kiválasztása

Szűrés tárolóra

deselectStoragePlaces() - Összes kiválasztott elem kikapcsolása

toggleSelectStoragePlace(storagePlaceId: number) - Elem kiválasztása egyesével

toggleSelectAllStoragePlaces() - Összes elem kiválasztása

resetFilters()

Keresés és szűrések alaphelyzetbe állítása

**Rendezés:**

orderBy(column: string):

Szűrés táblázatfejléc alapján

A felhasználó által kiválasztott fejléc nevét kapja paraméterként

**Lapozási funkció/termék megjelenítés szabályozása:**

get paginatedItems():

Az oldalon aktuálisan megjelenő termékek beállítása

get totalPages():

Össz oldalszám kiszámítása

updatePagination():

itemsPerPage átállítása esetén visszaállítja a jelenlegi lapot az elsőre

nextPage() és previousPage():

Előre/hátra lépés oldalak között

**Termékek kiválasztása:**

toggleItemSelection(event: Event, itemId: number)

Jelölőnégyzet állapottól függően a négyzethez tartozó termék azonosító hozzáadása/kivétele a selectedItemIds listához

toggleSelectAllItems():

Összes éppen megjelenített jelölőnégyzet átállítása

Azokhoz tartozó termék azonosítók hozzáadása/kivétele a listából

clearSelection():

selectAllChecked mező FALSE értékre állítja

Ez után hívja meg a toggleSelectAllItems() metódust, ami a FALSE érték miatt minden egyéb jelölőnégyzetet FALE-ra állít és üressé teszi a selectItemIds listát

**Termék címkék letöltése:**

downloadTags():

Kijelölt termék/ek címkéjének letöltése

Ha nincs kijelölve termék az összes táblázatban található adatot letölti

**Műveleti napló letöltése** :

downloadLogs():

Műveleti napló letöltése

validateLogFilters():

logFilters ellenőrzése

resetLogFilters():

logFilters alaphelyzetbeállítása

**Admin jogosultság ellenőrzése:**

get isAdmin():

**Terméktípus – tároló visszakeresése azonosítójuk alapján:**

getItemNameById(itemNameId: number):

ID alapján megkeresi az ahhoz tartozó terméktípus nevét

getStoragePlaceById(storagePlaceId: number):

ID alapján megkeresi az ahhoz tartozó tároló nevét

**Terméktípus – tároló törlés megerősítése:**

openTypeDeleteModal(type: 'itemName' | 'storagePlace', id: number):

Törlés típusának eldöntése – Termétípus vagy tároló kerül törlésre

confirmTypeDelete():

Típustól függően törlési metódus meghívása

**Figyelmeztető üzenet kiiratása:**

showMessage(msg: string, isError: boolean, duration: number):

Figyelmeztető üzenet megjelenítése

msg: string - üzenet

isError: boolean – hiba üzenet e

duration: number – kiirás hossza

**HTML:**

Kinézete admin státusztól függően változik

Hozzáadás, szűrés, rendezés, keresés minden felhasználónak elérhető

Admin specifikus elemek:

Áthelyezés, törlés, műveleti napló letöltés, címke letöltés

Terméktípus, tároló listázások, műveletek

Modális ablakok használata:

A kód legvégén találhatóak

Hozzáadás, áthelyezés, törlés megerősítések, letöltések, szűrések

**CSS:**

.custom-select - modális ablakok beviteli mezőihez használt

.scrollable-section – Termékek és Tárolók listázási szekciójánál használt

.scrollable-list – itemName és storagePlace szűrés modális ablakánál használt

**Login (src\app\components\login)**

Feladata a bejelentkezési oldal funkcióinak kezelése mint például email/jelszó alapú hitelesítés, jelszó-visszaállítás, átirányítás.

**Osztályváltozók**:

**userEmail: string = ' ';** - Felhasználó által megadott email cím

**userPassword: string = ' ';** Felhasználó által megadott jelszó

**showPassword: boolean = false;** - Jelszó láthatóságának állítása

**alertMessage: string | null = null;** - Figyelmeztető üzenet szövege

**isError: boolean = false;** - Üzenet típusa Hiba/Siker

**timeoutId: any = null;** - Üzenet automatikus eltűnés idejének újra indítására használt változó többszöri metódus híváskor

**Konstruktor:**

userService - Bejelentkezés (loginUser) és jelszó-visszaállítás (passwordChange) végrehajtása

router – Sikeres bejelentkezés utáni átirányítás

**Metódusok:**

ngOnInit()

localStorage kiürítése

loginUser()

Felhasználó bejelentkeztetése

Beírt email és jelszó átadása

Sikeres bejelentkezés esetén navigálás a főoldalra

Sikertelen bejelentkezés kezelése

togglePasswordVisibility()

Jelszó opcionális megjelenítése

passwordChange()

Elfeljetett jelszó esetén email kiküldése

showMessage(msg: string, isError: boolean, duration: number)

Figyelmeztető üzenet megjelenítése

msg: string - üzenet

isError: boolean – hiba üzenet e

duration: number – kiirás hossza

**HTML**:

Bejelentkezés űrlappal kivitelezve

Elfelejtett jelszó modális ablakot alkalmaz

**NavBar (src\app\components\navbar)**

Feladata a felhasználó navigálása oldalak között, valamint a kijelentkeztetés.

**Konstruktor:**

authService – Admin jogosultság ellenőrzése – linkek elrejtése

router – Kijelentkezés (logoutUser()) esetén átirányítás bejelentkezési felületre

**Metódusok:**

get isAdmin()

Ellenőzi, hogy a jelenlegi felhasználó admin-e

Bizonyos linkek csak admin felhasználónak jellennek meg

logoutUser()

Átirányítja a felhasználót a bejelentkezési felületre

**PasswordChange (src\app\components\password-change)**

Felhasználó első bejelenkezésénél köteles megváltozatni jelszavát, amit ezen az oldalon tehet meg.

**Osztályváltozók:**

**newPassword: string = '';** - Új jelszó

**confirmPassword: string = '';** - Új jelszó megerősítése

**showPassword: boolean = false;** - Jelszó láthatóságának állítása

**alertMessage: string | null = null;** - Figyelmeztető üzenet szövege

**isError: boolean = false;** - Üzenet típusa Hiba/Siker

**timeoutId: any = null;** - Üzenet automatikus eltűnés idejének újra indítására használt változó többszöri metódus híváskor

**passwordRequirements = [**

**{ text: 'Min. 8 karakter hosszú', isValid: false },**

**{ text: 'Nagy betű', isValid: false },**

**{ text: 'Kis betű', isValid: false },**

**{ text: 'Szám', isValid: false },**

**{ text: 'Speciális karakter (!@#$%^&\*)', isValid: false }**

**];** - Jelszókövetelmények listája és állapota

**Konstruktor:**

userService – Első bejelentkezéskor jelszó beállítás (firstLogin) végrehajtása

router – Sikeres jelszó változtatás esetén bejelentkezési felületre irányítás

**Metódusok:**

togglePasswordVisibility(): void

Jelszó megjelenítése/titkosítása

changePassword(): void

Feladata a jelszó megváltoztatása

Ellenőrzi a beütött jelszavak egyezőségét

Siker esetén visszairányítás /login oldalra

checkPasswordRequirements()

Ellenőrzi a beütött jelszó formátumát/minőségét a megadott kritériumok alapján

areAllRequirementsValid(): boolean

Ellenőrzi, hogy minden kritériumnak megfelel-e a jelszó

Csak akkor tud a felhasználó a jeleszó változtatás gombra nyomni ha igaz értékkel tér vissza

showMessage(msg: string, isError: boolean, duration: number)

Figyelmeztető üzenet megjelenítése

msg: string - üzenet

isError: boolean – hiba üzenet e

duration: number – kiirás hossza

**HTML:**

Bejelentkezési oldallal megegyező megjelenítés

**UserControl (src\app\components\user-control)**

Csak az adminnak elérhető felület. Feladata a felhasználók megjelenítése, keresése és kezelése, mint például letiltása/engedélyezése, admin jog adása/elvétele.

**Osztályváltozók:**

**newUserEmail: string = ' ';** - Új felhasználó emailcíme regisztációhoz

**newUserAdmin: boolean = false;** - Beállítás, hogy az új felhasználó admin-e

**users: any[] = [];** - Teljes felhasználó lista

**currentUser: string = localStorage.getItem('userEmail') || ' ';** - Az aktuálisan bejelentkezett felhasználó címe

**filteredUsers: any[] = [];** - Szűrt felhasználói lista a kereséshez

**filterText: string = '';** - A keresőmező tartalma

**alertMessage: string | null = null;** - Figyelmeztető üzenet szövege

**isError: boolean = false;** - Üzenet típusa Hiba/Siker

**timeoutId: any = null;** - Üzenet automatikus eltűnés idejének újra indítására használt változó többszöri metódus híváskor

**Konstruktor:**

userSerivce – Felhasználók kezelése

**Metódusok:**

loadUsers(): void

Felhasználók lekérése a szervertől

users és filteredUsers listák feltöltése

Betöltött adatok rendezése

registerUser(): void

Új felhasználó felvétele

disableUser(userEmail: string)

Felhasználó letiltása

enableUser(userEmail: string)

Felhasználó aktiválása

promoteUser(userEmail: string)

Felhasználó adminná tétele

demoteUser(userEmail: string)

Felhasználó admin jogának megvonása

applyFilter(): void

Email cím alapú szűrés

users listát szűri, az eredményeket a filteredUsers listába írja

showMessage(msg: string, isError: boolean, duration: number)

Figyelmeztető üzenet megjelenítése

msg: string - üzenet

isError: boolean – hiba üzenet e

duration: number – kiirás hossza

**HTML:**

Dashboard komponensel megegyező táblázat használat

Regisztációhoz modális ablak használata

**Szervizek**

Az autentikációs fejlécet a **getAuthHeaders()** metódus **(AuthService)** biztosítja. Ez felhasználásra kerül minden szervizben.

A kérések az e**nvironment.ts** fájlban található **apiUrl** címnek a megfelelő végpontjára irányulnak.

**Auth (src\app\services\auth)**

Hitelesítéssel kapcsolatos funkciókat tartalmazza.

**getAuthHeaders(): HttpHeader**

Token lekérése localStorage-ból

HttpHeader-t ad vissza ami az Authorization fejlécet tartalmazza

Ez a fejléc kerül felhasználásra a szerverrel kommunikáló metódusoknál

**isAdmin(): boolean**

Tokenből ellenőrzi hogy a bejelentkezett felhasználó admin-e

**Item (src\app\services\item)**

A szolgáltatás a **termékekkel** kapcsolatos backend kommunikációért felel.

Végpont: /item

**getItems():**

Típus: GET

Cél: az összes termék lekérdezése

**createItem(itemNameId: number, storagePlaceId: number, quantity: number):**

Típus: POST

Cél: új termék/ek rögzítése adott terméktípussal és tárolóval

**updateItem(itemIdList: number[], newStoragePlaceId: number):**

Típus: PUT

Cél: termék/termékek áthelyezése másik tárolóba ID alapján

**deleteItem(itemIdList: number[]):**

Típus: DELETE

Cél: termék/ek törlése ID alapján

**ItemName (src\app\services\item-name)**

Feladata a **terméktípusokkal** kapcsolatos backend kommunikáció

Végpont: /itemName

**getItemNames():**

Típus: GET

Cél: összes terméktípus lekérdezése

**createItemName(item: string):**

Típus: POST

Cél: új terméktípus létrehozása

**deleteItemName(id: number):**

Típus: DELETE

Cél: terméktípus törlése ID alapján

**Log (src\app\services\log)**

Feladata a **műveleti napló letöltéssel** kapcsolatos backend kommunikáció.

Végpont: /log

**downloadLogs(filters: { itemNameId?, storagePlaceId?, createdBy?, fromDate?, toDate? }):**

Típus: POST

Cél: Szűrők átadása a szervernek és a válaszul kapott 'blob' letöltése

**Storage (src\app\services\storage)**

Feladata a **tárolókkal** kapcsolatos backend kommunikáció.

Végpont: /storagePlace

**getStoragePlaces():**

Típus: GET

Cél: összes tároló lekérdezése

**createStoragePlace(storage: string):**

Típus: POST

Cél: új tároló létrehozása

**deleteStoragePlace(id: number):**

Típus: DELETE

Cél: tároló törlése ID alapján

**User (src\app\services\user)**

Feladata **felhasználó és jelszó kezeléssel** kapcsolatos backend kommunikáció.

**GetUsers():**

Típus: GET

Végpont: /users

Cél: összes felhasználó lekérdezése

**disableUser(userEmail: string):**

Típus: PATCH

Végpont: /disable

Cél: Felhasználó letiltása

**enableUser(userEmail: string):**

Típus: PATCH

Végpont: /enable

Cél: Felhasználó újra aktiválása

**promoteUser(userEmail: string):**

Típus: PATCH

Végpont: /promote

Cél: Felhasználó adminná tétele

**demoteUser(userEmail: string):**

Típus: PATCH

Végpont: /demote

Cél: Admin jog elvétele

**loginUser(userEmail: string, userPassword: string):**

Típus: POST

Végpont: /login

Cél: Felhasználó bejelentkeztetése

**registerUser(userEmail: string, isAdmin: boolean):**

Típus: POST

Végpont: /register

Cél: Új felhasználó regisztrálása, admin jog beállítása

**firstLogin(userPassword: string):**

Típus: PUT

Végpont: /firstLogin

Cél: Első bejelentkezés esetén kötelező jelszóváltoztatás

**passwordChange(userEmail: string):**

Típus: PUT

Végpont: /passwordChange

Cél: Új jelszó igénylése a megadott emailhez

**Útvonalfigyelők**

**Admin (src\app\guards\admin.guard.ts)**

Feladata:

Megakadályozni a /users útvonal elérését.

Ellenőrzi a localStorage-ből lekért authToken tartalmát és a lejáratát.

Ha az isAdmin mező igaz belépést engedélyez, egyéb esetben (pl: hiányzó, rossz formátumú token) átirányít a /login oldalra

**Auth (src\app\guards\auth.guard.ts)**

Feldata:

Megakadályozni a /passwordchange, /users és /dashboard útvonalak elérést.

Ellenőrzi a localStorage-ből lekért authToken tartalmát és a lejáratát.

Letiltott felhasználó vagy hiányzó, rossz formátumú token esetén átirányít a /login oldalra.

**FirstLogin (src\app\guards\first-login.guard.ts)**

Feladata:

Kötelezi a felhasználót első belépésekor a jelszóváltoztatásra.

Ellenőrzi a token tartalmát, ha isFirstLogin igaz átirányítja a felhasználót a /passwordchange végpontra.

Megakadályozni a /users és /dashboard útvonalak elérést új felhasználók számára jelszóváltoztatás előtt.

**Továbbfejlesztési lehetőségek**

Jelenleg a program termékkódot használ a termékek megkülönböztetésére/beazonosítására. Ez továbbfejleszthető lenne egy QR kód vagy vonalkód alapú rendszerre. Későbbiekben akár mobil eszközzel való beolvasásra, leltározásra. Emellett előnyt jelenthet egy külön raktár választó felület ("összes megjelenítése" opcióval - a program most csak így valósítja meg a megjelenítést) ami választás után csak az adott tárolóhoz tartozó adatokat jeleníti meg.

1. **Felhasználói kézikönyv:**

Felhasználók számára:

A felhasználói jogkörrel bíró felhasználók egy limitált felületet láthatnak, amit egy bejelentkezés előz meg.

Ahogy az alábbi képen látható a bejelentkezéshez a felhasználónak 2 adatra van szüksége. Az e-mail címre, amin keresztül az admin rögzítette a felhasználót és az e-mail címre kiküldött jelszóra.

A screenshot of a login form

AI-generated content may be incorrect.

Ezt követően a felhasználó jelszóváltásra köteles. A jelszóváltoztatás nem lehetséges amíg a jelszó nem felel meg minden az alábbi képeken látható kritériumnak. A teljesült kritériumokat pipával jelzi a program.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

Elfelejtett jelszó esetén a felhasználónak lehetősége van újat kérni. Ezt a bejelentkezés alatt található “Elfelejtett jelszó” feliratra kattintva teheti meg. A feliratra kattintva megjelenik egy felugró ablak, ahol megkell adnia az email címét. A küldés gombra kattintva új jelszót küld a program a megadott email címre. Ez követően első bejelentkezésnél ismét jelszó változtatásra lesz kötelezve a felhasználó.

A screenshot of a login form

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

A sikeres jelszóváltás után az új jelszó használatával ismételt belépést követel meg az oldal, mely, ha sikeres az alábbi képen látható felületre irányít.

A screenshot of a computer

AI-generated content may be incorrect.

A jogosultságok limitálásából adódóan a felhasználó csak megtekinteni, szűrni/rendezni és rögzíteni tudja a leltárazandó tárgyakat.

A hozzáadás gombbal új egyedi tárgyat rögzíthetnek a felhasználók.

A screenshot of a computer

AI-generated content may be incorrect.

A rögzítés során a felhasználó egy listából elérhető termeket és tárgyneveket választhatja, illetve megadhatja, hogy egy teremben hány darab van tárolva egy adat tárgyfajtából.

A screenshot of a computer

AI-generated content may be incorrect.

A Hozzáadás gombbal megerősíti és rögzíti a termékeket, a Mégse gombbal kilép a folyamatból.

A folyamat végén a felületen láthatóak a rögzült tárgyak.

A screenshot of a computer

AI-generated content may be incorrect.

A listázott tárgyak szűrhetőek/rendezhetőek:

A fejlécre kattintás csökkenő/növekvő sorrend között vált. Ezt felfele/lefele mutató nyíl jelzi.

A tárgyak Termék és Tároló alapú szűrését a bekarikázott ikonra kattintva lehet megtenni. Ez egy felugró ablakot hoz elő, ahol kiválaszthatóak a szűrés paraméterei.

A screenshot of a computer

AI-generated content may be incorrect.

Emellett lehetőség van termékek keresésére annak kódja, neve és tárolója alapján. A keresést az alábbi képen jelölt keresőmezőbe írással lehet megkezdeni.

A screenshot of a computer

AI-generated content may be incorrect.

A “Szűrők Törlése” gombbal a program minden szűrőt alaphelyzetbe állít, beleértve a keresőmezőt.

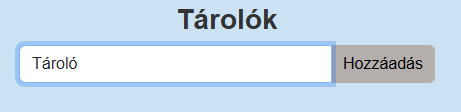
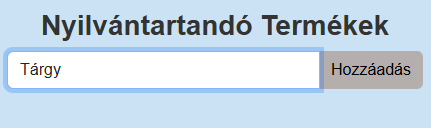
Admin felhasználók számára:

A különleges jogokkal nem rendelkező felhasználók által is elérhető felület az adminok számára sok funkcióval kibővül.

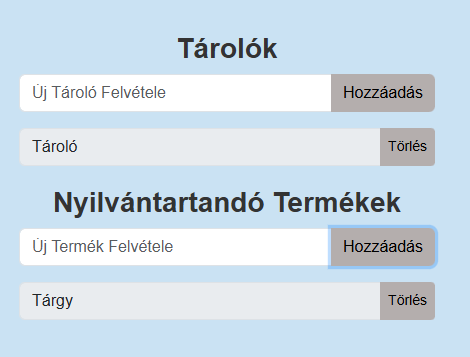
A screenshot of a computer

AI-generated content may be incorrect.

Az admin rögzíthet tárolókat és tárgyakat, ezáltal bővítve a listát, melyből az egyedi tárgy rögzítés során választhat a rögzítő. A listában szereplő tárolók és tárgyak nem törölhetők a listából, amíg bármely egyedileg rögzített tárgy használja valamely tulajdonságot.

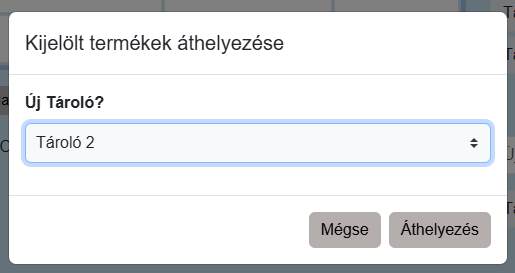
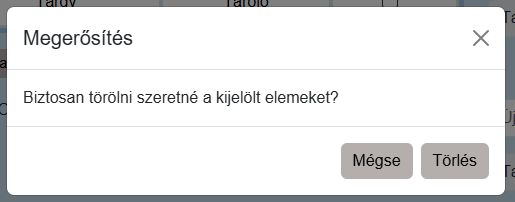
A sikeres rögzítés után:



A kijelölt tárgyak törölhetők és áthelyezhetők más tárolókba. Egyszerre több tárgy is kijelölhető és ugyanazon művelet alkalmazható rajtuk.



Áthelyezés Törlés

A screenshot of a computer

AI-generated content may be incorrect.A rögzített tárgyakon elvégzett műveletek a háttérben rögzülnek, melyek egy letölthető excel állományban elérhetők az admin felhasználók számára. A napló letöltéséhez meghatározott szűrők használhatók.

Szűrők:

Tárgy: Kiválasztható, hogy mely konkrét tárgytípuson (pl. Laptop) szeretnénk a műveleteket lekérdezni. Egyszerre a szűrés egy tárgytípus megadását engedi, amennyiben szűrő üres, úgy minden tárgytípus bekerül.

Tároló: Kiválasztható, hogy mely konkrét tárolóban tárolt termékeken (pl. 2-es tároló) szeretnénk a műveleteket lekérdezni. Egyszerre a szűrés egy tároló megadását engedi, amennyiben szűrő üres, úgy minden tárolóban tárolt termék bekerül.

Felhasználó: Egy adott felhasználó által végzett műveleteket kérdez le. Egyszerre a szűrés egy felhasználó megadását engedi, amennyiben szűrő üres, úgy minden felhasználó művelete bekerül.

Kezdő és záró dátum: A tárgyak rögzítésének időpontját a felület elmenti így lehetőség van célzott periódusokban történt műveletek megtekintésére.

A screenshot of a phone

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.Az admin felhasználónak továbbá lehetősége van a kiválasztott termékek címkéjének letöltésére.

A címke letöltés gombra kattintva megjelenik egy felugró ablak ahol megkell adni a fájl tetszőleges nevét. Ez után a letöltés gombra kattintva le is menti a címkéket a program .xlsx formátumba. Ha nincs kijelölt elem a program az összes termék címkéjét letölti. Ezt egy figyelmeztetéssel is jelzi a felhasználó felé.

A screenshot of a computer

AI-generated content may be incorrect.

Felhasználók menü:

A menüben minden admin felhasználó számára elérhető a jog, hogy más felhasználókat regisztráljanak, letiltsanak/feloldjanak, valamint a további jogkezelés (megvonás, jogadás).

A screenshot of a computer

AI-generated content may be incorrect.

Felhasználó felvétel:

A megkezdéshez az alábbi gombot kell megnyomni.

A blue and white icon with a plus sign

AI-generated content may be incorrect.

Megadandó egy létező e-mail cím, amennyiben a rögzíteni kívánt felhasználót admin joggal szeretnénk felvenni a jelölőnégyzetet pipáljuk be.

A screenshot of a chat

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.Rögzítés után:

A felhasználó mellett megjelenő rublikák jelölik a jelenlegi jogkörét és a rajta elvégezhető jogkezelések gombjait.

 - admin jog adása

 - admin jog elvétele

- felhasználó letiltás

Letiltás után:

A screenshot of a computer

AI-generated content may be incorrect.

Felhasználó feloldása a képen bekarikázott gombbal lehetséges.

1. Tesztek

BZSH Külkereskedelmi Technikum leltár teszt - excel

1. Összefoglalás

A projekt célja egy olyan leltárkezelő rendszer elkészítése volt, amelyet a BZSH Külkereskedelmi Technikum tanárai és dolgozói ténylegesen hasznosíthatnak mindennapi munkájuk során. Ennek érdekében a fejlesztés során végig arra törekedtünk, hogy az iskola igényeinek megfelelően készítsük el a munkát.

A legnagyobb kihívást az igények pontos feltérképezése jelentette – például milyen funkciókat rendeljünk hozzá az egyes jogosultsági szintekhez, hogyan legyen kialakítva a felhasználói felület, milyen módon valósuljon meg a regisztráció.

Így fontos tanulságként vontuk le, hogy a jövőben a hatékonyabb fejlesztéshez nélkülözhetetlen a higgadt, átgondolt, megbeszélésekre épülő közös munka.

Az alkalmazás több verzión és módosításon is keresztül ment annak érdekében, hogy a felmerülő elvárásokat a lehető legjobban megvalósítsuk. A kezdeti nehézségek ellenére úgy gondoljuk, sikerült elérnünk a kitűzött célokat: mint például jogosultság alapú hozzáférés, letisztult felhasználói felület, tömeges művelet kezelés, naplózás, szűrési és keresési lehetőségek, valamint email alapú felhasználói regisztráció is beépítésre került.

A jövőben az alkalmazást további funkciókkal szeretnénk bővíteni, hogy még tovább fejleszthessük a felhasználói élményt.

Backend oldalon célunk a felhasználói jogkörök bővítése, például SuperAdmin > Admin > Felhasználó bevezetése, amely lehetővé tenné a funkciók pontosabb szétválasztását és a rendszer biztonságosabb kezelését. Emellett a felhasználókkal végzett műveletek, mint például regisztráció, letiltás vagy jogosultságmódosítás naplózása is hozzájárulna az átláthatósághoz.

Frontend oldalon a jelenlegi termékkód-alapú azonosítás továbbfejleszthető lenne QR-kód vagy vonalkód alkalmazásával, ami lehetőséget nyújtana mobilos eszközökkel történő leltározásra is. Emellett egy külön raktárválasztó felület kialakítása, amely lehetővé tenné, hogy a felhasználó egy adott tárolóra szűkítve tekintse meg az adatokat, ezzel még átláthatóbbá téve a leltározási folyamatot.