** **

**ZÁRÓDOLGOZAT**

Készítették:

Horváth Leticia

Novák Gábor

Miskolc

2022.

**Miskolci SZC Kandó Kálmán Informatikai Technikum**

Az 5-0613-12-03 számú Szoftverfejlesztő és -tesztelő technikus szakképesítés záródolgozat

**LMS- Könyvtárnyilvántartó rendszer**

Készítették:

Horváth Leticia

Novák Gábor

Miskolc

2022.

Tartalom

[Bevezetés 4](#_Toc101788308)

[Fejlesztői eszközök 5](#_Toc101788309)

[XAMPP 5](#_Toc101788310)

[Visual Studio 2022 5](#_Toc101788311)

[.NET 6 és C#10 5](#_Toc101788312)

[Fluent WPF 6](#_Toc101788313)

[Material Design 6](#_Toc101788314)

[Git és Github 6](#_Toc101788315)

[Jira 7](#_Toc101788316)

[React 7](#_Toc101788317)

[Tesztelés 7](#_Toc101788318)

[Felhasználói dokumentáció 9](#_Toc101788319)

[Asztali alkalmazás 9](#_Toc101788320)

[Bejelentkezés 10](#_Toc101788321)

[Főoldal 10](#_Toc101788322)

[Navigációs panel 11](#_Toc101788323)

[Könyvek 11](#_Toc101788324)

[Kölcsönzés 12](#_Toc101788325)

[Tagok 14](#_Toc101788326)

[Kilépés 15](#_Toc101788327)

[Webes alkalmazás 16](#_Toc101788328)

[Főoldal 17](#_Toc101788329)

[Bejelentkezés 17](#_Toc101788330)

[Könyvek oldal 18](#_Toc101788331)

[Keresés 18](#_Toc101788332)

[Profil 19](#_Toc101788333)

[Kijelentkezés 19](#_Toc101788334)

[Továbbfejlesztési tervek 20](#_Toc101788335)

[Források 22](#_Toc101788336)

# Bevezetés

Záródolgozatunk célja egy olyan nyilvántartó alkalmazás fejlesztése volt, ami iskolai vagy kisebb városi könyvtáraknak nyújt segítséget adataik tárolásában és kezelésében. Az ilyen intézmények számára elengedhetetlen az adatok permanens tárolása, rendszerezése és karbantartása, valamint a visszakereshetőség biztosítása. Ezzel szemben még napjainkban is rengeteg olyan kisvárosi, illetve iskolai könyvtár létezik, ahol a készletet Excel táblázatokban tárolják, sőt előfordul olyan is, hogy a kölcsönzéseket még kézzel írt olvasójegyekkel menedzselik. Erre a problémára szerettünk volna megoldást kínálni egy mindenki által könnyen kezelhető, különösebb hozzáértést nem igénylő szoftver elkészítésével és egy hozzátartozó weboldallal, ahol a könyvtári tagok nyomon tudják követni a kölcsönzéseiket, illetve böngészni a készletet.

A felhasználói felület két részből áll. Az egyik, egy C# nyelven írt asztali kliens, amely a könyvtár dolgozói számára lehetővé teszi az adminisztrációs teendők elvégzését, például az új tagok és könyvek felvételét, adataik módosítását, vagy a kölcsönzéssel kapcsolatos műveleteket.

A másik rész pedig egy webes felület, amely elkészítéséhez Reactot használtunk. Ezen az oldalon a könyvtár tagjai tudnak böngészni a bent lévő könyvek között, valamint bejelentkezés megtekinthetik kölcsönzéseiket.

# Fejlesztői eszközök

## XAMPP

Az adatbázisunkhoz a MySQL-t választottuk, amely egy SQL-en alapuló, relációs adatbáziskezelő-szoftver. Ez szinte a legelterjedtebb adatbáziskezelő, az egyszerű használata, gyorsasága és megbízhatósági miatt. A MySQL bárki számára ingyenesen elérhető, a legegyszerűbb beszerzési módja az XAMPP nevű szoftvercsomag, melyben megtalálható a MySQL, egy Apache webszerver, és egy phpMyAdmin nevű adatbáziskezelő. Könnyen telepíthető a csomag, és minden szükséges beállítást elvégez magától, nekünk pedig már csak használni kell az általa feltelepített alkalmazásokat. A csomagban ezen kívül megtalálható még a PHP és a Perl programozási nyelv, egy Mercury nevű email szerver és egy FileZilla nevű FTP szerver. Mi az egyszerűsége, és a már meglévő tapasztalataink és tudásunk alapján választottuk ezt. Könnyedén létre tudtuk benne hozni az általunk használni kívánt táblákat, típusokat tudtunk adni a táblázatok mezőinek, és egyszerűen tudtunk kapcsolatokat definiálni a táblák között. (például szerzők - könyv\_szerzők - könyvek) Ebben az adatbáziskezelőben nagyon egyszerűen tudtunk dolgozni, hiszen nem igényel külön kódolást, a táblákat, kapcsolatokat, vizuális módon létre tudjuk benne hozni. A legjobb, hogy az általunk vizuálisan elvégzett dolgok MySQL kód formában is megtekinthetőek.

## Visual Studio 2022

Mivel egy modern programot szerettünk volna elkészíteni, így az általunk használt programoknak/technológiáknak is igyekeztünk a legújabb verzióját használni. A fejlesztés elején még csak a Visual Studio 2019 állt rendelkezésünkre, viszont a fejlesztés közben megjelent az új, 2022-es verzió, ezért átálltunk arra. Az átállás zökkenőmentesen ment, semmiféle nehézség nem adódott belőle, hiszen maga a felhasználói felület szinte semmiben nem változott.

A 2022-es verzióban sokat fejlődött a kód javaslat rendszer, nagyban megkönnyítve ezzel a munkánkat, illetve maga az egész alkalmazás gyorsabb lett és kevesebb erőforrást használ.

## .NET 6 és C#10

A Visual Studio új verziójával jött az új .NET és C# verzió is, mi pedig azonnal tudtuk, hogy ezt szeretnénk használni. Az új feature-ök használatához a szerver komponensünk némi átíráson esett át, viszont egy sokkal szebb, rövidebb és átláthatóbb kódbázist kaptunk. A C#10 számos újítást hozott, mint például a "Fájl hatókörű névtér deklaráció" vagy a "Globális Using Utasítás", melyek segítségével rövidebbé tehettük a kódunkat. Jobb teljesítményt is hozott a .NET 6, így a szerverünk és az asztali alkalmazásunk is kevesebb erőforrást használ. Külön plusz számunkra, hogy a .NET 6 egy úgynevezett Long Term Support verzió, azaz a Microsoft 2024 végéig támogatást, vagy esetleges javításokat ad ki hozzá.

## Fluent WPF

A Fluent WPF csomag (nuget package) az alkalmazás ablakának kölcsönöz modern kinézetet. Lehetővé teszi, hogy áttetsző ablakokat hozzunk létre. A segítségével a Windows sötét és világos témája közti váltást is meg tudtuk oldani, annak érdekében, hogy mind a két beállításon jól mutasson az alkalmazásunk.

## Material Design

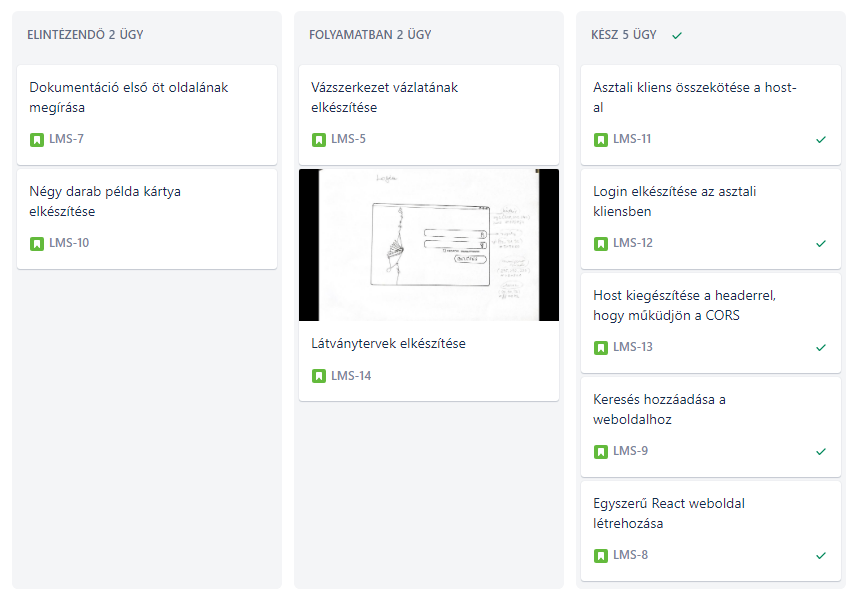
A Material Design egy formanyelv, melyet a Google fejlesztett ki. A Google állítása szerint ezt a formanyelvet a papír és tinta alapján gondolták el. Modern és letisztult kinézetet kölcsönöz az alkalmazásunknak és nem vonja el a figyelmet a munkáról. A Material Design elemeit főként az űrlapoknál, gomboknál, valamint a naptárnál, dátumválasztóknál használtuk. Az előre megírt elemek nagyban megkönnyítették a munkánkat, mert így nem kellett teljesen új kinézetet programoznunk egy-egy komponensnek.

## Git és Github

A verziókezeléshez a Git rendszert használjuk, amely nagyban megkönnyítette a közös munkát. Egyszerűen tudtunk vele dolgozni egy adott projektrészen, nyomon követhettük, hogy éppen mit csinált a másik, és könnyen visszakereshettünk dolgokat az előző verziókból.

A Git repository-nk a Githubon került tárolásra. Ez volt a legkézenfekvőbb megoldás számunkra, mivel teljesen ingyen hozhattunk létre privát repository-kat, így csak mi láthattuk a kódunkat.

## Jira

A Jira egy úgynevezett ügy- és projektkövető rendszer, ahol egyszerűen követhetjük a projektünk haladását. A Jira felületén tudunk Sprinteket létrehozni, határidőt állítani nekik, valamint feladatokat rögzíteni a Sprinthez. Ezekután mozgatni tudjuk a feladatokat, így változtatva az állapotukat. Az állapot lehet “Elintézendő”, “Folyamatban” és “Kész”. A Sprintek és Feladatok segítségével mi is egyszerűen tudtuk követni a projekt haladását és rendszerezni a jövőbeli teendőinket.

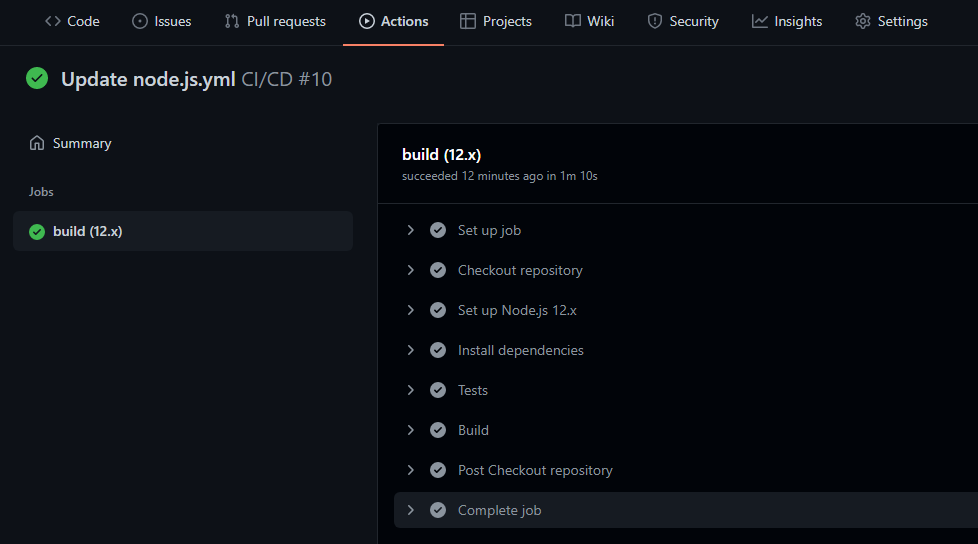
1.ábra – Sprint a Jira felületén

## React

A weboldalunkhoz a React nevű Javsascript keretrendszert választottuk. Ez az egyik legelterjedtebb, és mivel különben is szerettük volna megtanulni, ez volt a kézenfekvő választás a weboldalunk elkészítéséhez. A React filozófiája, hogy egy adott oldal több komponensből áll, melyeket akár újra fel is lehet használni egy másik oldalon. A komponensek tudnak egymásnak adatokat is átadni, így lehetséges a kommunikáció közöttük, illetve frissíthetünk is egy komponenst, az oldal újratöltése nélkül.

## Tesztelés

Teszteléshez a Github felületét használtuk, külön feltöltve ide a projektünket, és megírva a pipeline-t, amely levezérli a tesztelés folyamatait. Azért ezt választottuk, mert ennek a használata sokkal egyszerűbb volt számunkra, mint a Gitlabos felület. A Github CI (Continuous Integration, azaz folyamatos integráció)/CD (Continuous Delivery, azaz folyamatos szállítás) rendszere sokkal fejlettebb a Gitlab-énál. Nagyon fontos egy szoftver fejlesztési ciklusa során a tesztelés, ezt mi sem szerettük volna kihagyni.



2. ábra – Tesztek futtatása a Github felületén

A tesztsorozat több részből áll:

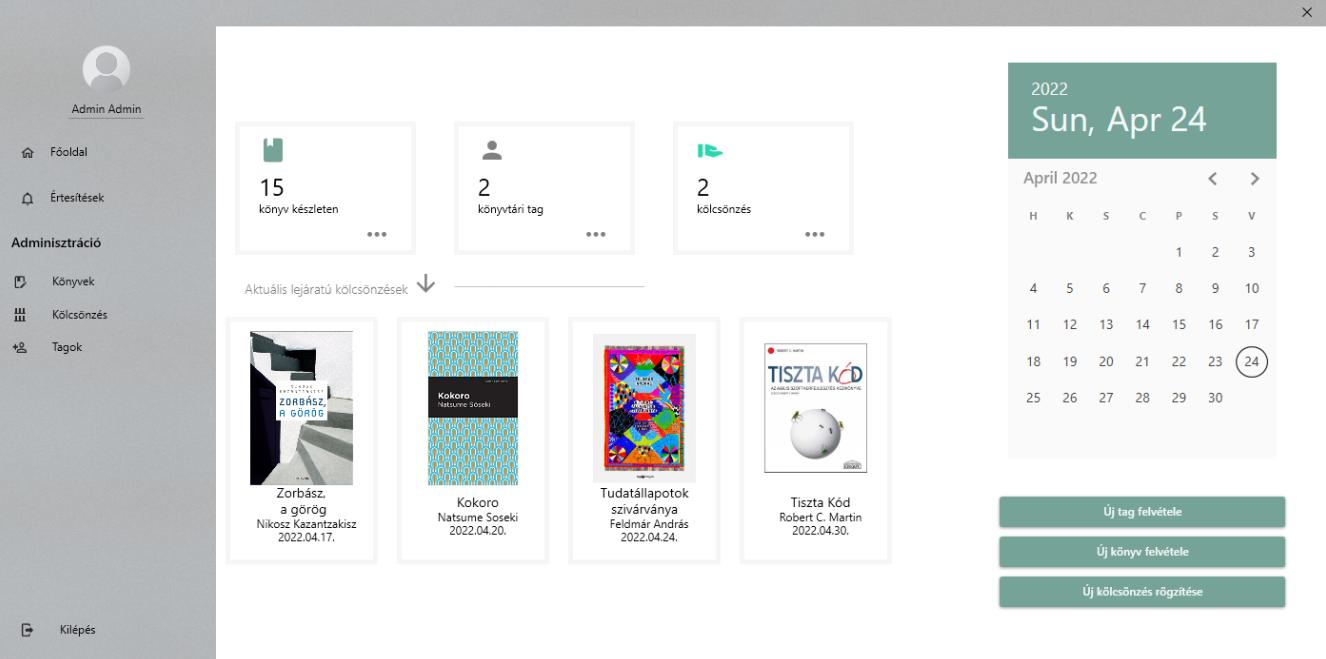
* Build Teszt - A build fordítja át a kódunkat a számítógép által érthető utasításokra, a tesztünk pedig ennek a megvalósulását ellenőrzi.
* Unit Teszt – A Unit Teszt az alkalmazás egyes összetevőinek tesztelését foglalja magában.

A fent felsorolt teszteket a Reactos webkliensen futtattuk le. A pipeline-t pedig a kimeneti képekkel együtt feltöltöttük a Teszt mappába.

# Felhasználói dokumentáció

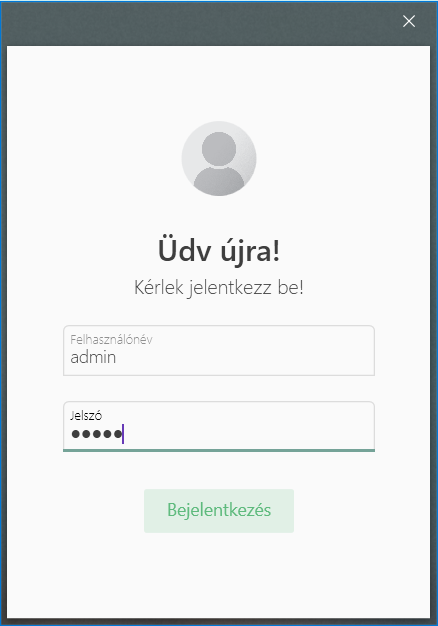
## Asztali alkalmazás

Ahogy a bevezetésben is szó esett róla, az asztali alkalmazásunk egy C# nyelven WPF-ben megírt nyilvántartó program. Ez a szoftver nem direkt módon kommunikál az adatbázissal, hanem egy külön, általunk írt API-n keresztül, így megfelelve a Restful irányelvnek. Törekedtünk egy bárki által könnyen megtanulható, átlátható alkalmazás létrehozására.

Szerettük volna, ha az alkalmazást használó könyvtári dolgozó gyorsan és egyszerűen tudja elvégezni a könyvtári feladatokat. Amit csak tudtunk, igyekeztünk automatizálni, mint például az olvasójegy szám, könyvtári kód generálást, ezzel is gyorsítva a felhasználó munkáját.

3. ábra – Az asztali alkalmazás főoldala

### Bejelentkezés



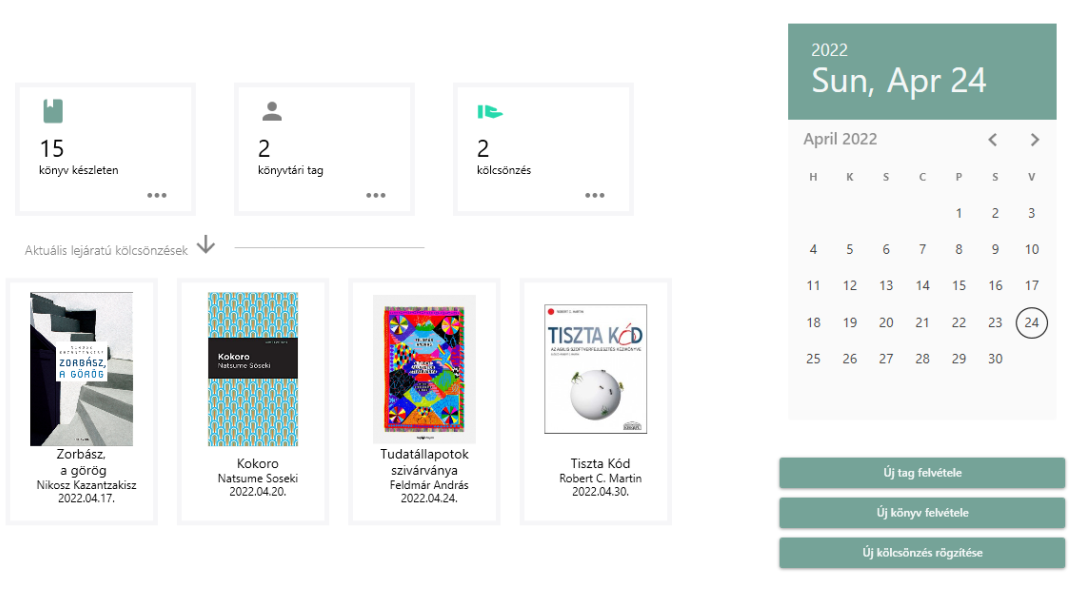
Az alkalmazást megnyitva egy bejelentkezési képernyő fogadja a felhasználót, ahol a helyes felhasználónév-jelszó kombinációval lehetősége van bejelentkezni.

A bejelentkezési folyamat igénybe vesz pár másodpercet a jelszó titkosításából adódóan.

Jelenleg egy példafelhasználóval lehet bejelentkezni, melynek a felhasználóneve „a”, a jelszava pedig szintén „a”. Ezt azért választottuk, mert a fejlesztés során rengeteg időt megtakarított nekünk a rövid felhasználónév-jelszó páros. Ha helyesen adtuk meg az adatokat, akkor bezárul a bejelentkezési ablak és láthatóvá válik a főoldal, ahol egy „Sikeresen bejelentkeztél” üzenet fogad minket. Amennyiben hibás adatokat adott meg, a rendszer egy hibaüzenetet küld a felhasználónak.

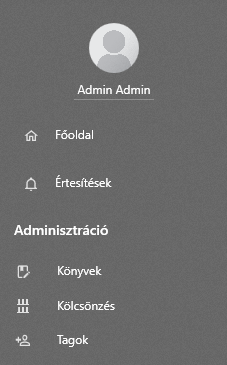
4.ábra – Bejelentkezési felület

### Főoldal

A főoldal szerkezetének kialakításakor üzenőfalszerű hatásra törekedtünk. Szerettünk volna egy olyan felületet, ami összegyűjti a leggyakrabban használt műveleteket egy képernyőre, ezzel gyorsabbá téve elérésüket. A másik célunk pedig minimális mennyiségű statisztika megjelenítése volt, hátha ez segítséget nyújt a könyvtár számára.

5.ábra – Üzenőfal az asztali alkalmazás főoldalán

### Navigációs panel

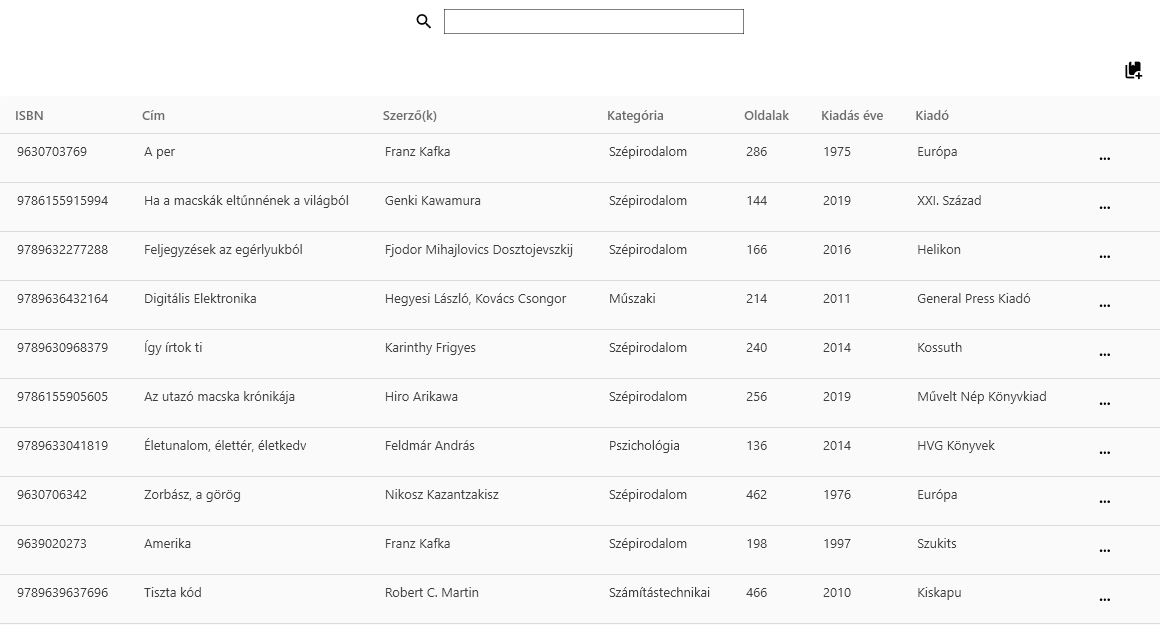
A baloldali transzparens sávban található a program menüje. Mivel ez egy könyvtárnyilvántartó alkalmazás, így a legjellemzőbb funkciói a listázás és az adatfelvitel. A főmenü egyelőre négy menüpontból áll:

6.ábra - Navigációs panel

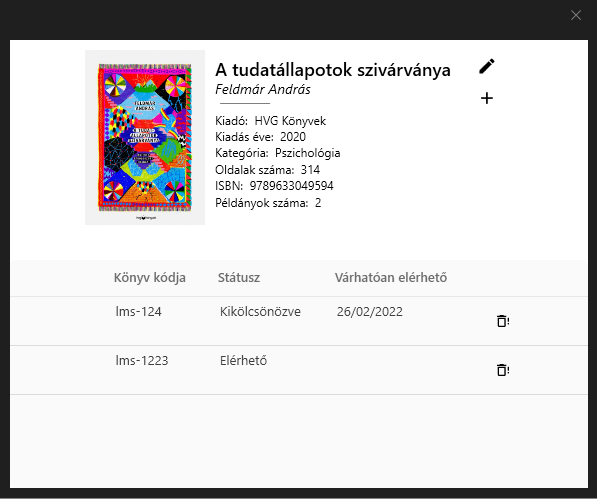
* Főoldal
* Könyvek
* Tagok
* Kölcsönzések

A menüpontok mindegyikében lehetőség van, az adatok listázásán kívül, a hozzájuk kapcsolódó adminisztrációs feladatok elvégzésére is.

### Könyvek

A könyvek menüpontot megnyitva a könyvtárban lévő könyvek listáját tekintheti meg a felhasználó. A lista fejlécei kattinthatóak, kattintás után megjelenő nyilakkal pedig az adott oszlop tartalma növekvő és csökkenő sorrendbe rendezhető (szöveges mezőknél ez az ABC sorrendet jelenti.). Efelett a lista felett található egy kereső, melynek segítségével szűrni tudjuk a listát. Címre és a könyv szerzőjére egyaránt lehet keresni. Található még egy plusz gomb a lista felett, mellyel új könyvet tudunk felvinni a rendszerbe. Új könyv felvitelekor automatikusan létrejön egy könyvtári azonosítóval ellátott példány az adott könyvből.   
A lista elemek a sor végén elhelyezett részletek gombbal egyenként is megnézhetőek.

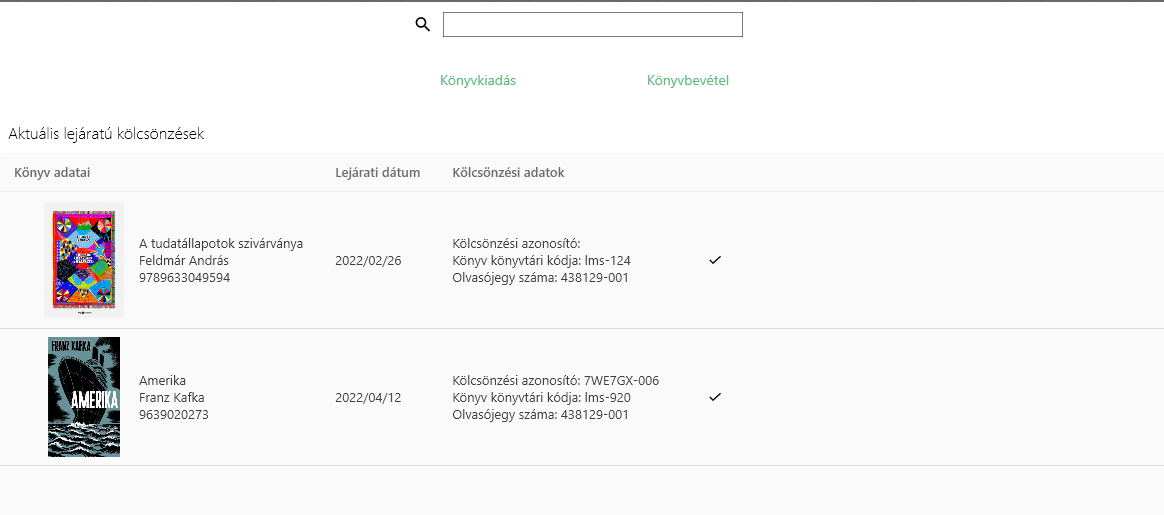
7.ábra – Könyvek listája

Kattintás után felugrik egy ablak, ami a könyvhöz tartozó adatokat jeleníti meg, a hozzátartozó borító képével együtt. Ezen az oldalon van lehetősége a felhasználónak megtekinteni az adott könyvből raktáron lévő példányokat, illetve új példányt rögzíteni a plusz gomb segítségével. Minden példányhoz tartozik egy könyvtári azonosító, amelyet a rendszer automatikusan generál. Például „lms-124”, ahol az lms a könyvtár nevének kezdőbetűiből összeálló mozaik szó. A kötőjel után következő számsor pedig a könyv és a példány adatbázisbeli azonosítójából tevődik össze. A példányok mellett lévő „kuka” ikonnal lehetőség van a példányok törlésére is. Óvintézkedésként csak a példányok törlésére van lehetőség, így elkerülve azt, hogy a könyvet véletlenül törölje a felhasználó a teljes adatbázisból. Ha törölni szeretné a könyvet, akkor törölnie kell az összes meglévő példányt belőle. Az utolsó példány törlésekor maga a könyv is törlésre kerül. A könyv címe mellett lévő ceruza ikonnal lehetőség van a könyv adatainak szerkesztésére. Látható még ezen az oldalon a példányok státusza, amely lehet elérhető, vagy kikölcsönözve. Amennyiben ki van kölcsönözve, láthatóvá válik a várható visszaérkezési dátuma is.

8. ábra - Egy könyv részleteinek megtekintése

### Kölcsönzés

A sorban következő menüpont a kölcsönzés. A felhasználó itt tudja menedzselni a könyvek kiadását és bevételezését. Mint a könyveknél, a lap tetején itt is található egy kereső, mellyel az aktuális lejáratú kölcsönzések között tud szűrni, a könyv könyvtári kódja alapján. A fejlécre kattintva itt sorba rendezhetőek az oszlopok, mint a könyveknél. A könyvek a lejárati dátumuk alapján vannak rendezve, így a legközelebbi lejárati dátummal rendelkező könyv kerül a lista tetejére. Annak érdekében, hogy megkönnyítsük a kölcsönzéssel járó adminisztrációt, a listán lévő könyvek mellett elhelyeztünk egy gombot (pipa ikon), ami a felhasználót átirányítja egy ablakra, ahol a kölcsönzés adataival kitöltött űrlap található.

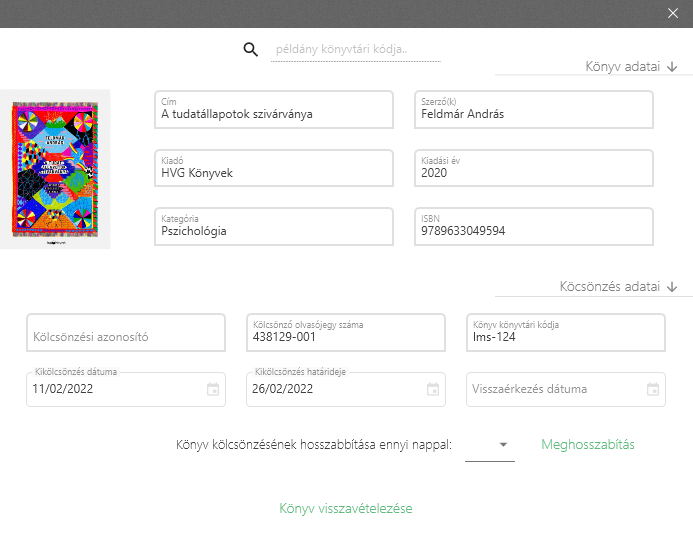
Itt a felhasználónak nincs már más dolga, mint megadni a visszaérkezése dátumát a követekező formátumban: „éééé-hh-nn”, vagy a dátum ikonra kattintva, egy lenyíló naptáron kiválasztani azt. A „könyv visszavitelezése” gombra kattintva a könyv visszakerül a készletbe, és a kölcsönzés befejeződik.

9.ábra – Kölcsönzések kilistázása

Kölcsönzés hosszabbítását szintén ezen az ablakon lehet elvégezni. Ezt kétféleképpen teheti meg a felhasználó. Az egyik a „könyv kölcsönzésének meghosszabbítása ennyi nappal” szöveg mellett található legördülő menü, ahol előre meghatározott értékek (1,2,3,5,7 10) közül választhat. A másik opció pedig az e felett található, „Kikölcsönzés határideje” címkével ellátott mező, mely ugyanúgy működik, mint a fentebb említett „Visszaérkezés dátuma”, tehát kézzel beírva, vagy a naptárból kiválasztva. Az érték kiválasztása után a művelet a „Meghosszabbítás” gombra való kattintással fejezhető be.

A könyv bevételezése művelet a listából való kikeresésen kívül, elvégezhető másként is. Ehhez a kölcsönzések listája felett elhelyezkedő „Könyvbevétel” gombra kell kattintania a felhasználónak. Ezután felugrik egy üres űrlap, ami automatikusan kitöltődik, ha a felhasználó az űrlap felett található kereső mezőbe beírja a könyv könyvtári kódját. Azért választottuk a könyvtári kód alapján való keresést, mert ez fizikailag is megtalálható a könyvön, így annak a leolvasása tűnt leggyorsabb és legegyszerűbb módnak. Miután a könyvtári kódot beütöttük, az entert lenyomva az űrlap mezői automatikusan kitöltésre kerülnek. Ezek után, ahogy a gyors gombos megoldásnál is, annyi a teendőnk, hogy a visszaérkezés dátumát beállítsuk, és a „Könyv visszavételezése” gombbal véglegesítjük a kölcsönzést.

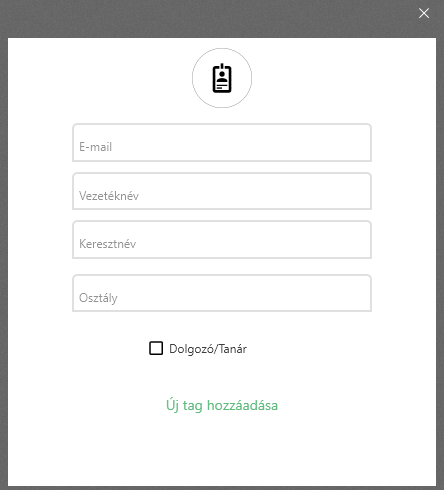
Ha a felhasználó a könyvet kiadni szeretné, azt ugyanezen az ablakon teheti meg, a „Könyvkiadás” gomb segítségével. Ez szinte ugyanúgy működik, mint a könyv bevételezése. Egy könyvtári kód segítségével a program kitölti az űrlap majdnem minden mezőjét, a kölcsönző olvasójegy számán és a kölcsönzés határidején kívül. Itt értelemszerűen nem kell még megadnunk a visszaérkezési dátumát, a kölcsönzés azonosítót pedig a program magának generálja. A „Könyv kiadása” gombra kattintva elindul a kölcsönzés.



10. ábra - Könyvkiadás

### Tagok

Ez alatt a menüpont alatt a felhasználó a könyvtár tagokkal kapcsolatos adminisztrációt tudja elvégezni. Ahogy a többi menüpontnál is, az adatok itt is listába vannak szedve. A lista felett elhelyezkedő keresővel szűrhetünk olvasójegy és név alapján. A plusz ikonnal tud felvenni új tagot a könyvtárba, ehhez mindössze a nevet, e-mail címet és az osztályt kell megadni, illetve, hogy az illető dolgozó-e vagy diák. Ezt azért csináltuk így, mert a programot elsősorban iskolai könyvtáraknak szántuk. Az adatok kitöltése után, a rendszer generál egy olvasójegy számot, és a tag bekerül a listába, innentől tud kölcsönözni, valamint az e-mail cím olvasójegy párossal be tud lépni a webes felületen, böngészhet a könyvek között, megtekintheti a kölcsönzéseit és hosszabbíthat is.

A könyvtárosnak lehetősége van a könyvtári tag törlésére és az adatainak szerkesztésére is. Ezt úgy tudja megtenni, hogy a lista elem melletti három pontra kattintva megjelenik egy kuka, illetve egy ceruza ikon. A kuka ikonnal az illető törölhető a tagok közül, a ceruzával pedig módosíthatóak az adatai, kivéve az olvasójegy számot.

11. ábra - Új tag felvétele



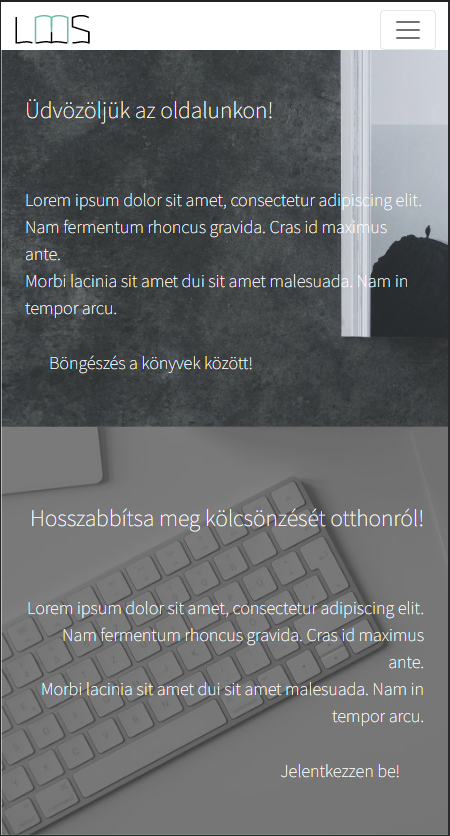
12. ábra - Egy tag adatai

### Kilépés

A kilépés gomb a navigációs menü alján található, a felhasználó ennek segítségével tud kilépni a programból. Ide kattintva minden bezáródik.

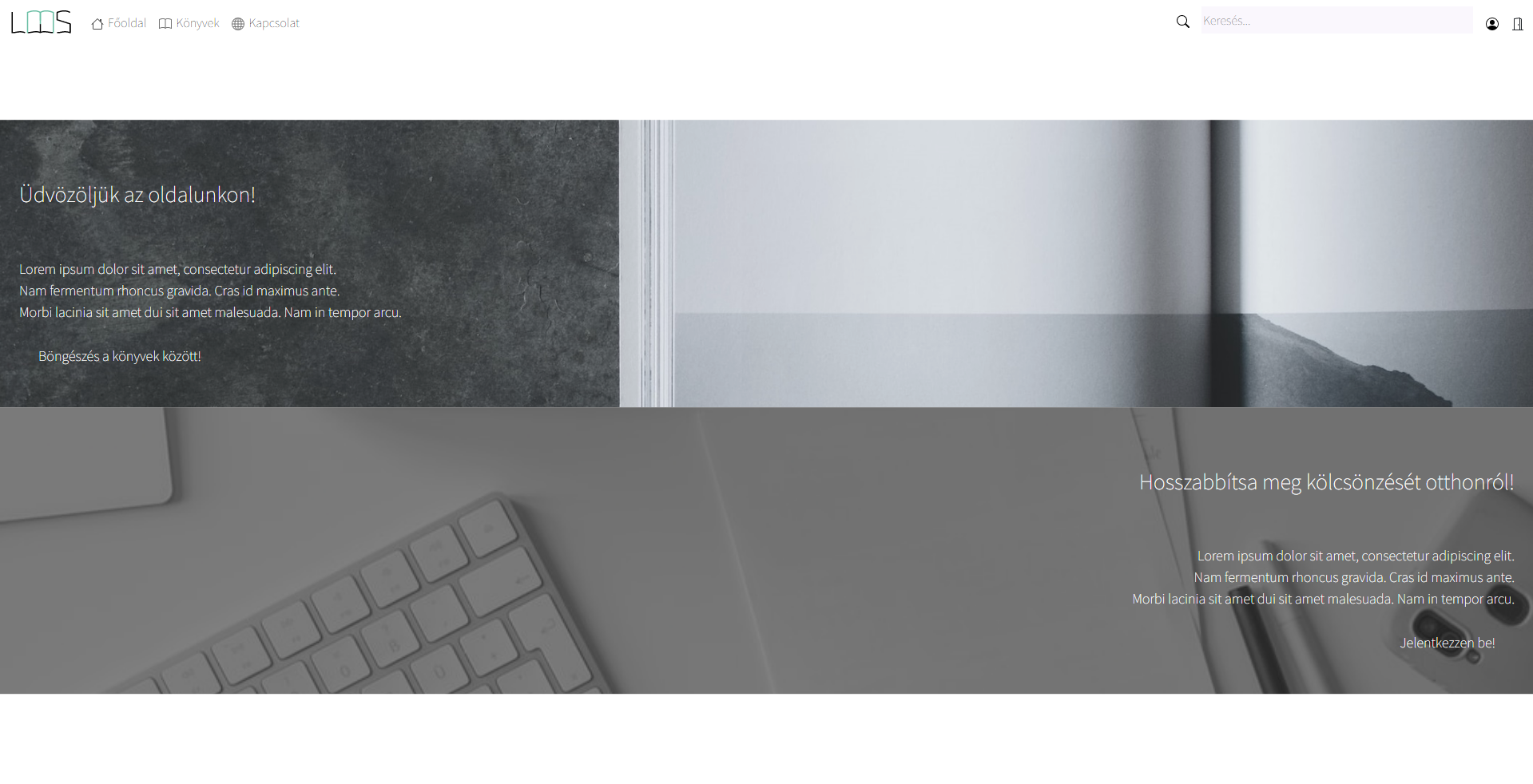
## Webes alkalmazás

A webes alkalmazás elkészítésének egyik célja az volt, hogy a könyvtári tagok az intézményen kívül is tudjanak böngészni a kínálat között, illetve, hogy nyomon tudják követni a saját kölcsönzéseiket. Ugyanakkor szerettük volna megkönnyíteni a könyvtári dolgozó munkáját is, azzal, hogy a könyvek keresését és a hosszabbítások intézését lehetővé tesszük a tagok számára.

A kinézetét tekintve arra törekedtünk, hogy minél egyszerűbb, letisztultabb legyen, annak érdekében, hogy a tagok könnyen el tudjanak igazodni a menüpontok között. Tudjuk, hogy sok látogatónak egyszerűbb az oldalt a mobilján megtekinteni, így arra törekedtünk, hogy az oldal kinézete kisebb képernyőkön is használható legyen.

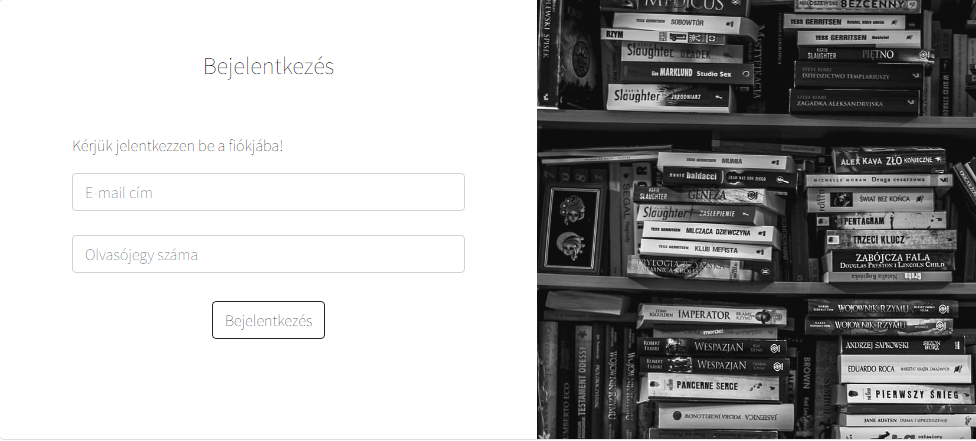
13. ábra - Mobilos nézet

### Főoldal

A főoldalnál egy egyszerű, informatív, letisztult, mégis modern kinézet elkészítése volt a célunk. Az oldal két sávból áll, egy „üdvözlésből”, mely egy kis üdvözlő üzenetet és némi információt tartalmaz, az alján pedig egy gyorsgomb, ami a könyvek oldalra vezet. A másik sáv is informatív jellegű, arról tájékoztatja a látogatókat, hogy bejelentkezés után lehetőségük van a kölcsönzéseik megtekintésére, illetve esetlegesen a meghosszabbításukra.

14. ábra - Webes alkalmazás főoldala

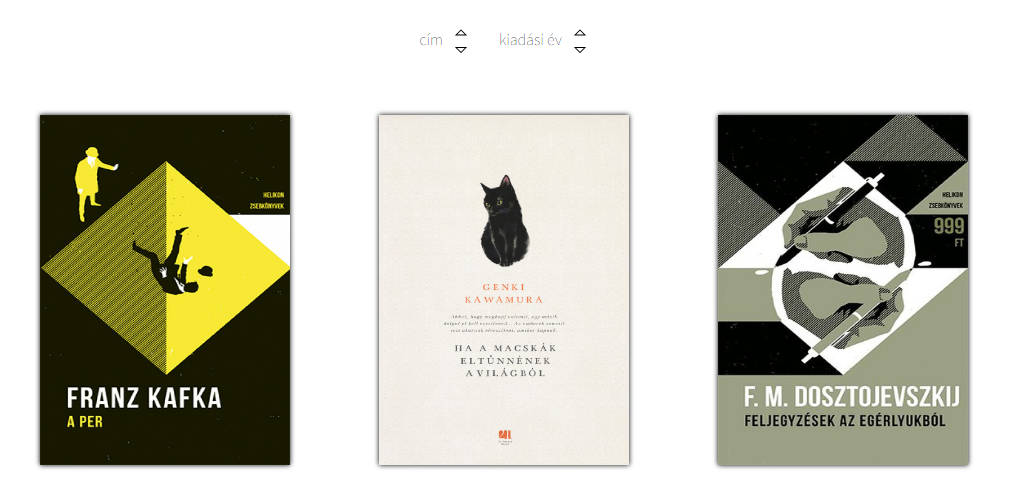
### Bejelentkezés

A weboldalra a könyvtár tagjai, a beiratkozás során megadott e-mail címmel, illetve az akkor kapott olvasójegy számmal tudnak belépni. A könyvtári könyvek közötti keresés lehetséges ugyan felhasználói fiók nélkül, viszont ahhoz, hogy hosszabbítsanak vagy a kölcsönzéseiket nyomon tudják követni, mindenképpen rendelkezni kell olvasójegy számmal és belépni. 

15. ábra bejelentkezési felület a webes alkalmazásban

Sikeres bejelentkezés után az oldal átirányítja a felhasználót a könyvek oldalra, melyen megtekintheti a könyvtárban lévő összes könyvet. Az oldal jobbfelső sarkában megjelenik egy új „ajtó” ikon, mely a kijelentkezést szolgálja.

### Könyvek oldal

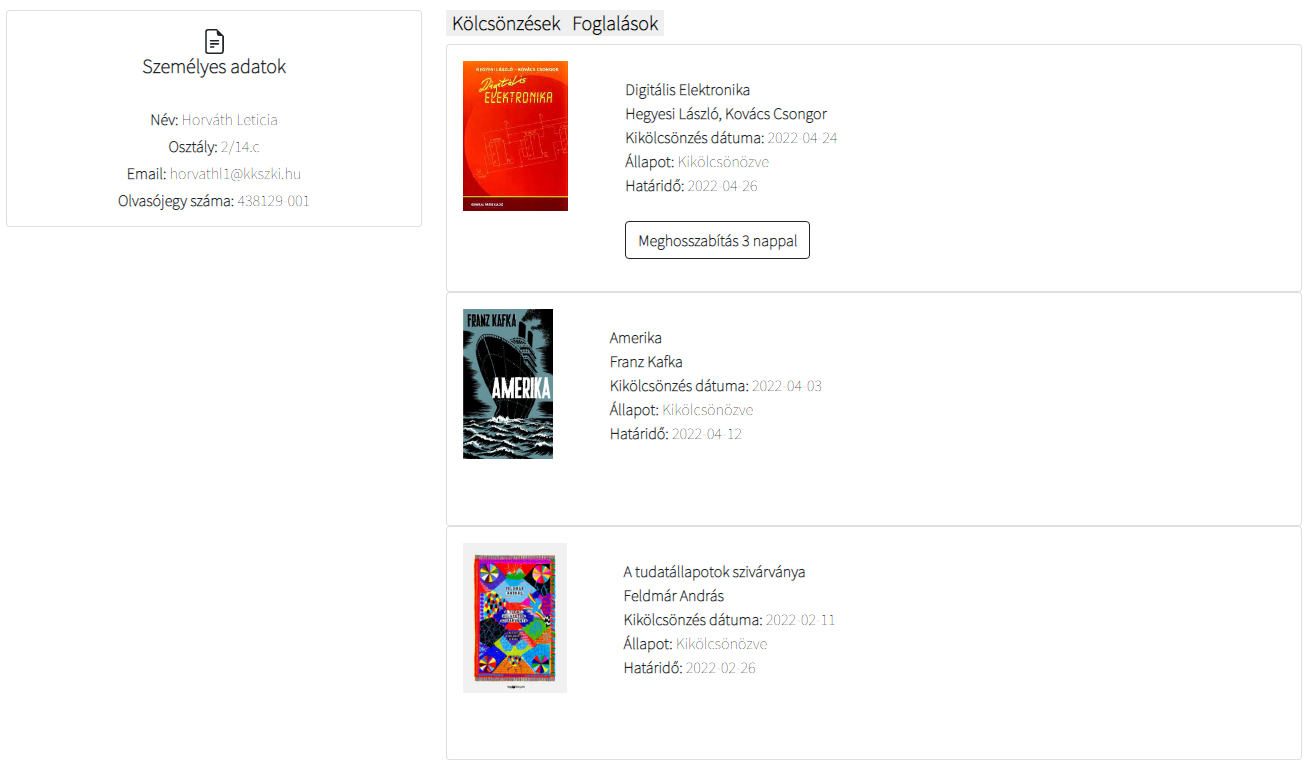
Ezen az oldalon tudják a felhasználók a könyvtárban lévő könyveket böngészni. A könyvek képei egy rácsba rendeződve jelennek meg. Ez a rács dinamikusra lett írva, így mindig annyi elemet helyez egy sorba, amennyi még kényelmesen kifér oda. Ha egy könyv képére visszük az egeret, megjelennek a könyv adatai. A képek kattinthatóak is, kattintás után egy kis ablakban tekinthetőek meg a könyv részletes adatai. A rács felett található két darab címke és a hozzájuk tartozó két-két gomb, melyek a rács rendezését szolgálják. Lehetőség van cím, illetve kiadási év szerint is rendezni.

16. ábra - Könyvek oldal

### Keresés

A navigációs sáv jobb szélén található egy beviteli mező, mely a keresést szolgálja. Lehetőségünk van a könyv címére, illetve a szerzőjére is keresni. Miután beírtuk a keresni kívánt kifejezést, az „enter” gomb lenyomásával lehet keresni. A találatok a „Könyvek” oldalhoz hasonlóan, egy rácsba rendezve jelennek meg.

### Profil

A profil oldalon lehetősége van a tagoknak megtekinteni a saját adataikat, illetve a kölcsönzési előzményeiket. A jelenleg aktív kölcsönzéseket is látni, illetve lehetőség van azokat meghosszabbítani is. A meghosszabbítás csak a lejárati dátum előtt két nappal lehetséges.

17. ábra - Profil oldal

### Kijelentkezés

Természetesen lehetősége van a tagoknak a kijelentkezésre is. Ehhez a jobb felső sarokban található ajtó ikonra kell kattintani. A kijelentkezés azonnal megtörténik, utána pedig átirányítja a felhasználót a könyvek oldalra.

# Továbbfejlesztési tervek

Ahogy épült a projektünk, úgy jutott eszünkbe egyre több továbbfejlesztési terv, amit sajnos idő hiányában nem tudtunk már elkészíteni, viszont összegyűjtöttük őket, hogy a jövőben tudjunk még dolgozni rajtuk. Ezek a következők:

* Kívánságlista – Olyan könyvek listája, melyek még nincsenek meg a könyvtárnak. Ide a weboldalon tudnának felvenni könyveket a felhasználók (egy űrlapot kitöltve), a könyvtár dolgozói pedig értesítést kapnának róla az asztali alkalmazásban. Lehetne egy kívánságlista fül az asztali alkalmazásban is, ahol egy listába szedve látni lehetne őket, az elemek mellett lévő gombbal pedig gyorsan új könyvet lehetne felvenni, így „teljesítve a kívánságot”.
* Foglalás – A könyv adatait tartalmazó modal ablakon lehetőség lenne foglalást leadni, persze csak ha a felhasználó be van jelentkezve, illetve az adott könyvből nincs elérhető példány. A foglalásról szintén értesítést kapnának az asztali alkalmazásban.
* E-mail értesítések – A tagok e-mailben értesítést kapnának az általuk kivett könyv lejárata előtt pár nappal, a lejárat napján, illetve a lejárati dátum után.
* Hírek – Ez egy több részből álló rendszer lenne. Az egyik része a webes kliens főoldalán, vagy egy külön hírek oldalon. Itt, a könyvtárral vagy az iskolával kapcsolatos hírek jelennének meg. Lehetne böngészni a régebbi hírek között is. Az asztali alkalmazásból lehetne új híreket rögzíteni, a már meglévő híreket szerkeszteni, esetleg törölni. Tartozna ehhez egy szövegszerkesztő, amellyel a szöveget lehetne formázni például betűméret, betűtípus, szín, félövér, dőlt, illetve képet beszúrni.
* Részletes Keresés – A keresőt átalakítani, hogy lehessen részletes keresést csinálni. Checkboxokon bejelölni, hogy a szerző, kiadó, kategória, ISBN szám stb. között keressen. A kiadási évnél akár intervallumot is meg lehetne adni.
* Kapcsolat – A weben a kapcsolat fülön lehetne egy űrlap, melynek segítségével üzenetet lehet írni a könyvtár dolgozóinak. Az üzeneten kívül meg kéne adni egy kapcsolattartási e-mail címet, ahova a válasz fog érkezni a könyvtártól, illetve egy nevet, ami alapján címezni tudják a választ.
* Asztali alkalmazás átjárhatósága – A felhasználók oldalon egy felhasználóra kattintva meg lehetne tekinteni, hogy épp milyen könyveket kölcsönzött ki, illetve a kölcsönzési előzményeit is. Az aktív kölcsönzését például innen is lehetne visszavételezni.
* Interaktívabb weboldal – A könyvek oldalon a könyvre kattintva megjelennek annak részletesebb adatai. Ezeket az adatokat kattinthatóvá lehetne tenni. Például a szerzőre kattintva megjelennének az általa írt könyvek, a kategóriára kattintva az oda tartozó könyvek, a kiadóra pedig az összes általuk kiadott könyv.
* Értesítések – Ez az asztali alkalmazásban lenne elérhető, itt lehetne megtekinteni az új foglalásokat, kívánságlistára helyezett könyveket.

# Források

*1. C# programozás lépésről lépésre, Reiter István, Jedlik Oktatási Stúdió KFT.,2018, 9786155012174*

*2. Why XAMPP?,* [*https://xamppguide.com/*](https://xamppguide.com/)*, Letöltés dátuma: 2022-01-22*

*3. What's new in .NET 6* [*https://docs.microsoft.com/en-us/dotnet/core/whats-new/dotnet-6*](https://docs.microsoft.com/en-us/dotnet/core/whats-new/dotnet-6)*, Letöltés dátuma: 2022-02-18*

*4. What's new in Visual Studio 2022* [*https://docs.microsoft.com/en-us/visualstudio/ide/whats-new-visual-studio-2022?view=vs-2022*](https://docs.microsoft.com/en-us/visualstudio/ide/whats-new-visual-studio-2022?view=vs-2022)*, Letöltés dátuma: 2022-02-26*

*5. Introduction - Material Design* [*https://material.io/design*](https://material.io/design)*, Letöltés dátuma: 2022-02-08*

*6. GitLab CI/CD* [*https://docs.gitlab.com/ee/ci/*](https://docs.gitlab.com/ee/ci/)*, Letöltés dátuma: 2022-05-10*

*7. React: A JavaScript library for building user interfaces* [*https://reactjs.org/*](https://reactjs.org/)*, Letöltés dátuma: 2021-12-07*

*8. Mire használható a Jira?* [*https://www.atlassian.com/hu/software/jira/guides/use-cases/what-is-jira-used-for*](https://www.atlassian.com/hu/software/jira/guides/use-cases/what-is-jira-used-for)*, Letöltés dátuma: 2021-11-23*