

# Adatbázis alapú rendszerek

2023-2024/2

Csütörtök 16:00

IB152L-6

Könyvesbolt

Készítette:

Szabó Tímea

Veszeli Karina

Nagy Péter

## Munka felosztása

- ✧ **Szabó Tímea:** Dokumentum szerkesztése, Feladat szöveges leírása, Követelménykatalógus, Adatfolyam diagramok, Egyedmodell, EKT-diagram, Relációs adatelemzés, Szerep-Funkció mátrix, Képernyőtervek, Adatbázist létrehozó script, Egységes frontend dizájn, Adatbáziskapcsolat üres felhasználói felülettel rendelkező alkalmazáshoz, Bejelentkezési űrlap elkészítése, Könyv kezelés oldal, Áruház kezelés oldal, Fiók kezelés oldal, Könyvek oldal, Főoldal, Kosár oldal, Könyv adatlap oldal, Készletinformációs oldal, Dokumentáció aktualizálása
- ✧ **Veszeli Karina:** Követelménykatalógus, Adatfolyam diagramok, Egyedmodell, Táblák leírása, Egyed-Esemény mátrix, Adatbázist létrehozó script, Bejelentkezési űrlap elkészítése, Könyv kezelés oldal, Áruház kezelés oldal, Kosár oldal, Keresés címre vagy szerzőre, Kedvenc szerző és műfaj, Készlet kimerüléséről figyelmeztetés küldése
- ✧ **Nagy Péter:** Feladat szöveges leírása, Követelménykatalógus, Adatfolyam diagramok, Egyedmodell, Funkciók megadása, Adatbázist létrehozó script, Regisztrációs űrlap elkészítése, Bejelentkezési űrlap elkészítése, Profil oldal, Fiók kezelés oldal, Sikerlista oldal, Újdonságok oldal, Főoldal, Készletinformációs oldal, Számla készítés

## Értékelési mód

- ✧ **1. Mérföldkő:** Csapat
- ✧ **2. Mérföldkő:** Csapat
- ✧ **3. Mérföldkő:** Csapat
- ✧ **4. Mérföldkő:** Egyéni
- ✧ **Végső leadás:** Egyéni

## Feladat szöveges leírása

A Streeler könyvesbolt weboldal célja, hogy egy könnyen kezelhető platformot kínáljon a könyvek szerelmeseinek, ahol mindenki megtalálhatja a számára legmegfelelőbb olvasnivalót és élvezheti az online vásárlás nyújtotta kényelmet. Emellett a Streeler üzleteiben egy kávézó is található, ahol a vendégek kényelmesen elfogyaszthatják kedvenc kávéjukat és egy kényelmes környezetben olvashatnak.

A látogatók regisztrálhatnak és böngészhetik a könyv kínálatot. A regisztrált felhasználók bejelentkezhetnek, így lehetőségük van vásárlásra. Kosárba helyezhetik a megvásárolni kívánt könyveket és törölhetnek elemeket belőle még a fizetés előtt. Választhatják a házhozszállítást vagy a személyes átvételt az üzletben. Az adminisztrátorok könyveket, műfajokat, áruházakat és azok készleteit kezelhetik.

## Követelménykatalógus

- ✧ **Regisztráció, bejelentkezés és kijelentkezés:**
  - A látogatók regisztrálhatnak és böngészhetik a könyv kínálatot, azonban nem rendelhetnek
  - A regisztrált felhasználók bejelentkezhetnek a rendszerbe, így lehetőségük van a vásárlásra
  - A bejelentkezett felhasználók kijelentkezhetnek
  - A bejelentkezett felhasználó a profil oldalon megtekintheti a kedvenc szerzőjét és műfaját az általa eddig megvásárolt könyvek alapján

#### ✧ **Könyvek kezelése:**

- A rendszer lehetővé teszi a könyvek hozzáadását, szerkesztését és törlését az arra jogosultaknak
- A könyvek adatai között szerepelnek: név, kiadás éve, kiadó, szerző, oldalszám, kötés, méret, ár, műfaj

#### ✧ **Könyvek keresése és szűrése:**

- Keresés címre, szerzőre
- Szűrés műfajokra
- A találatok számát kigyűjti a rendszer

#### ✧ **Könyvadatlapok megtekintése:**

- Az adott könyv oldalán megjelenítésre kerülnek a könyv adatai és kosárba helyezhetjük azt
- Megtekinthetjük a könyv elérhetőségét az áruházakban
- A szerzőre és kiadóra kattintva egy google keresést indíthatunk el

#### ✧ **Legújabb könyvek megjelenítése:**

- A kezdőoldalon megjelenik a 5 legutóbb felvitt könyv
- Az újdonságok oldalon részletesebb listát kapunk

#### ✧ **Legfelkapottabb könyvek megjelenítése:**

- A kezdőoldalon megjelenik a 3 legtöbbet eladott könyv
- Az sikerlista oldalon a teljes listát láthatjuk

#### ✧ **Műfajok kezelése:**

- A rendszer lehetővé teszi műfajok és alműfajok hozzáadását, szerkesztését és törlését az adminisztrátoroknak
- Könyvek sorolhatók műfajokba

#### ✧ **Áruházak kezelése:**

- Az országban több üzlet tartozhat a könyvesbolt-hálózathoz
- A rendszer lehetővé teszi az áruházak hozzáadását, szerkesztését és törlését az adminisztrátoroknak

#### ✧ **Könyvek elérhetősége áruházakban:**

- A rendszer megjeleníti, hogy egy adott könyv melyik áruházakban kapható

#### ✧ **Készlet nyilvántartása:**

- A rendszer nyilvántartja a készleteket boltonként

#### ✧ **Készlet kimerülésének figyelése:**

- A rendszer figyelmeztetést küld, ha egy termék készlete kimerül

#### ✧ **Kosár:**

- A felhasználó kosárba helyezheti a megvásárolni kívánt könyveket

#### ✧ **Könyv statisztikáinak aktualizálása:**

- A rendszer növeli a könyv eladott példány számát, ha a vásárló megveszi azt

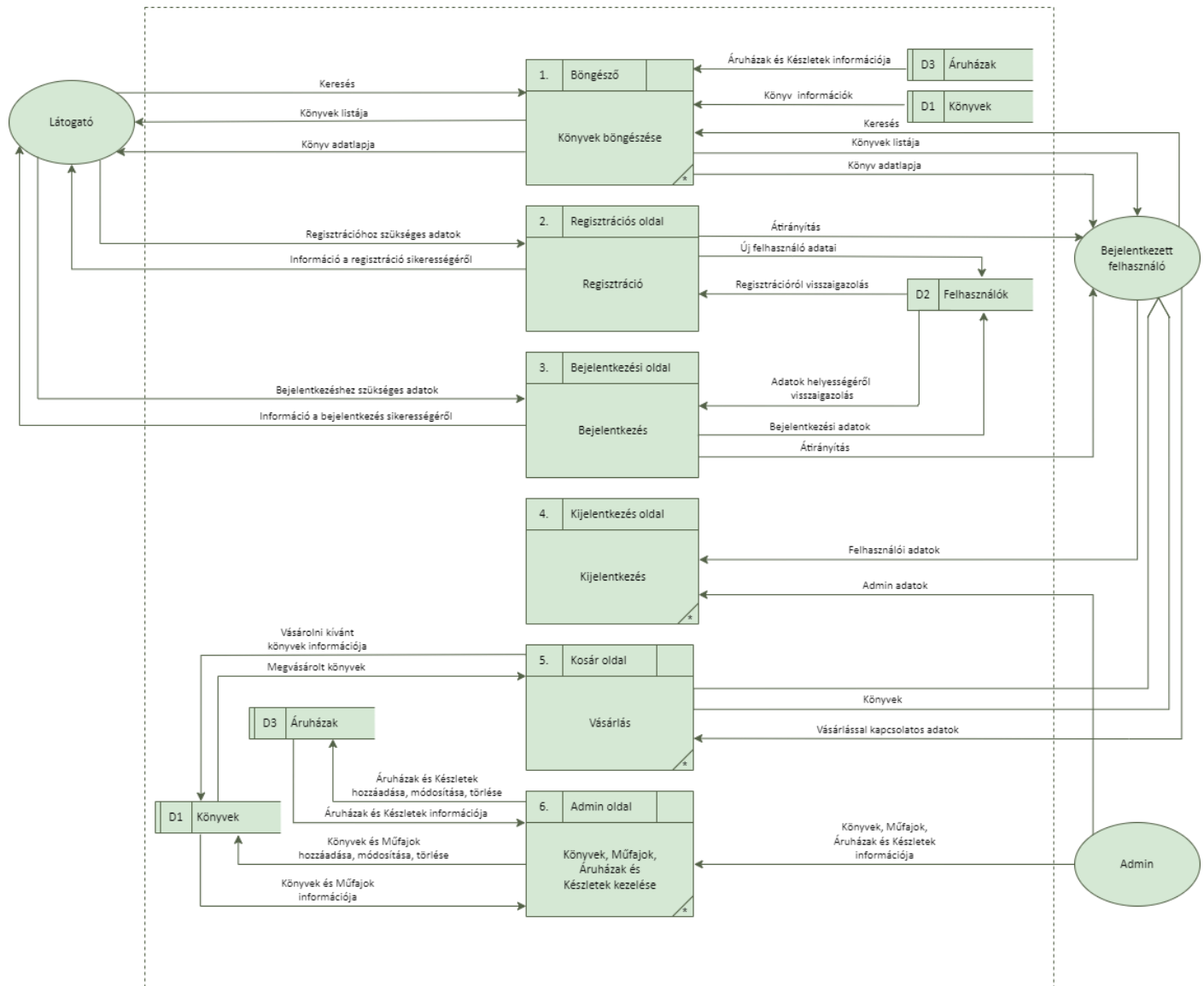
#### ✧ **Vásárlás:**

- A felhasználó megvásárolt könyveit tárolja a rendszer

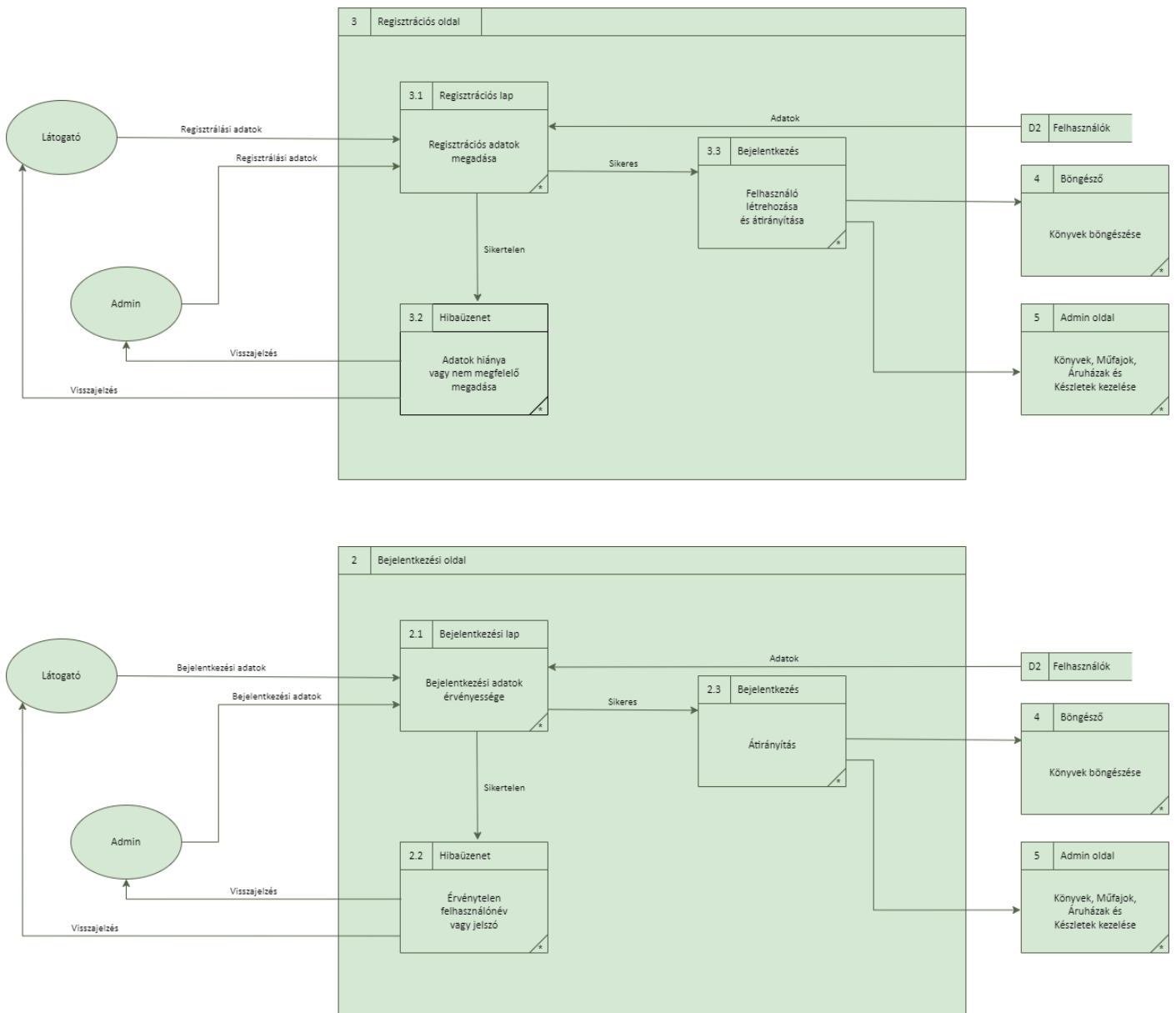
# Adatfolyam diagram (DFD)

## ✧ Fizikai:

### ■ A DFD 1. szintje

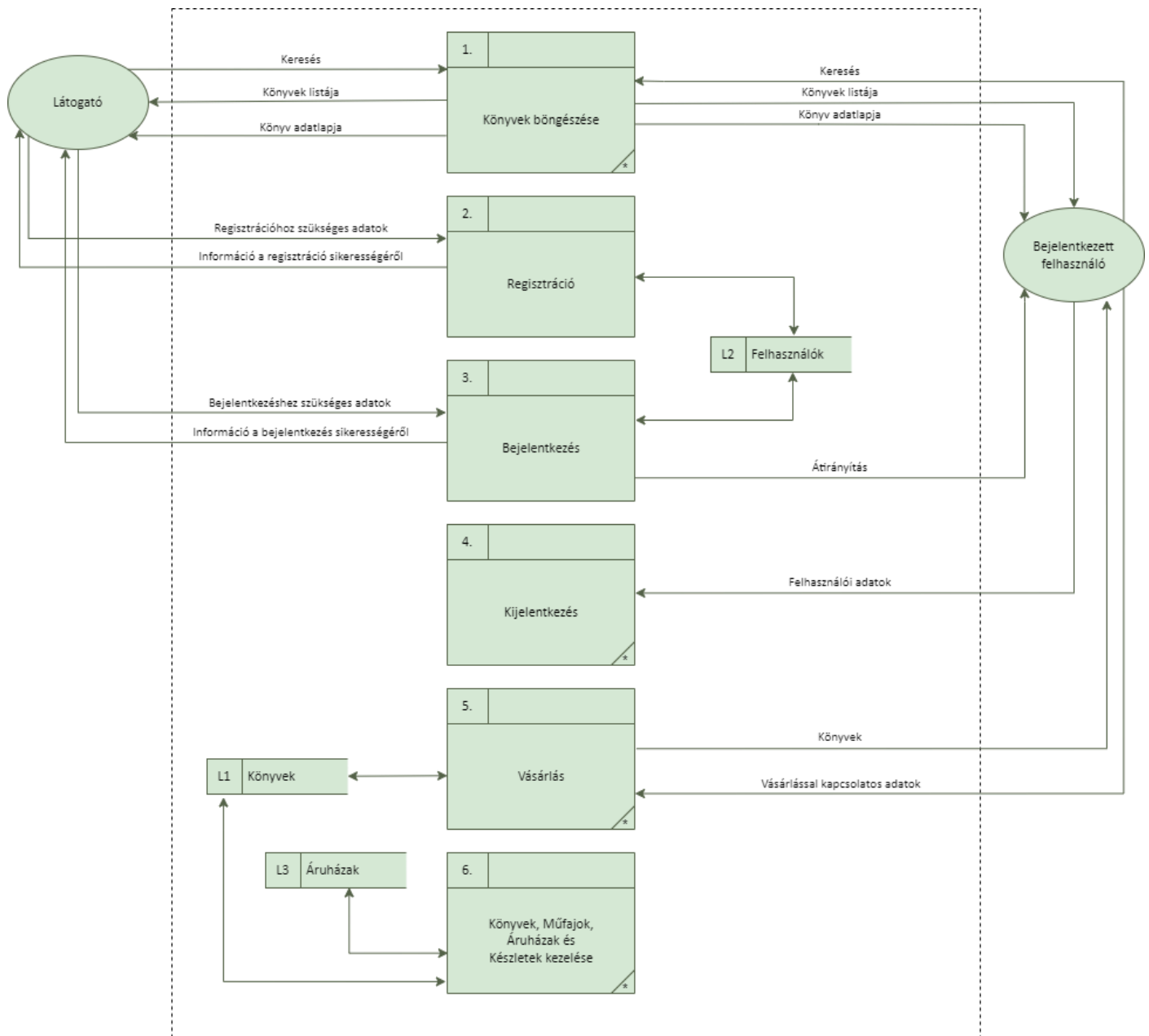


■ A DFD 2. szintje (Regisztráció és Bejelentkezés)

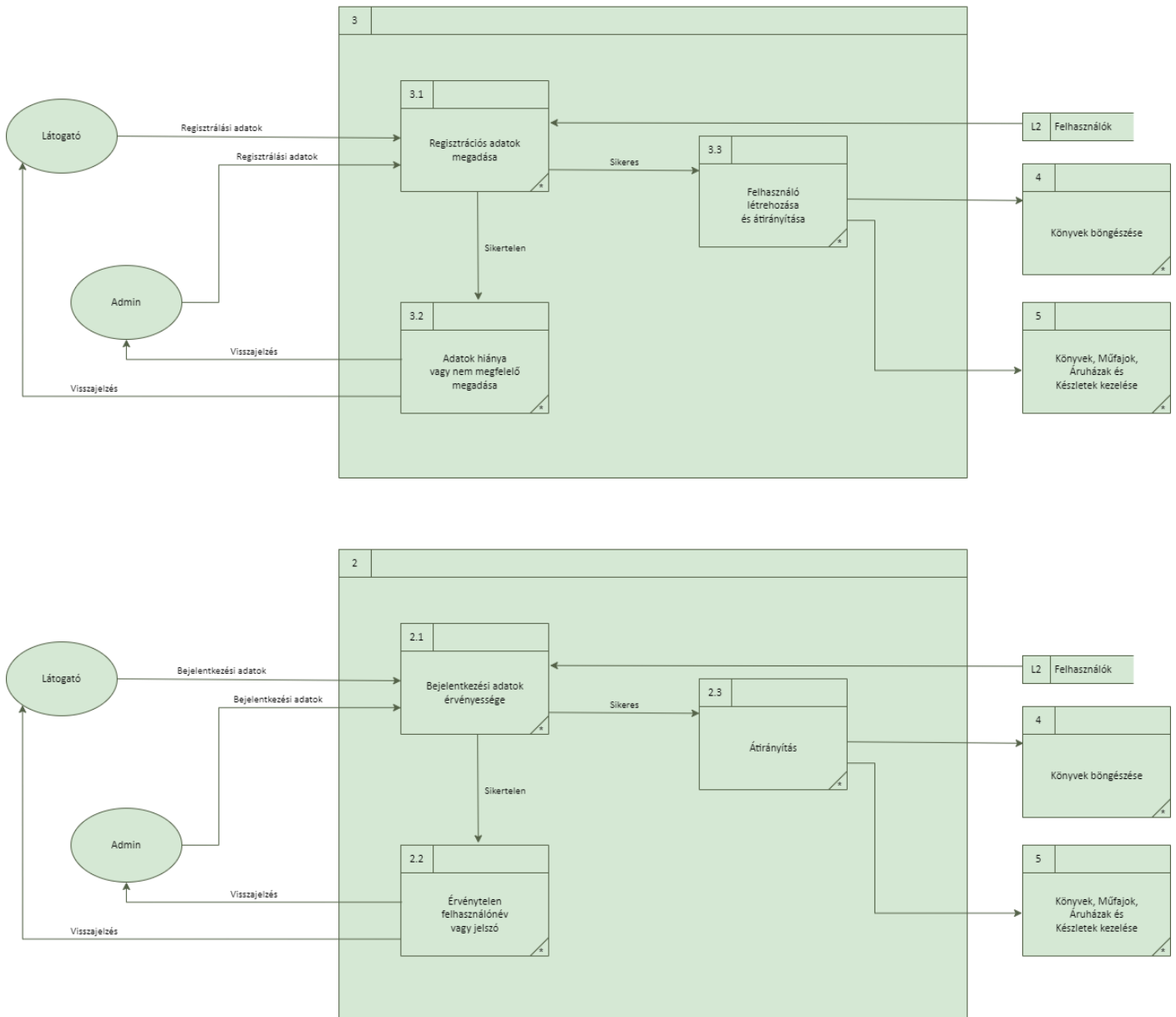


## ✧ Logikai:

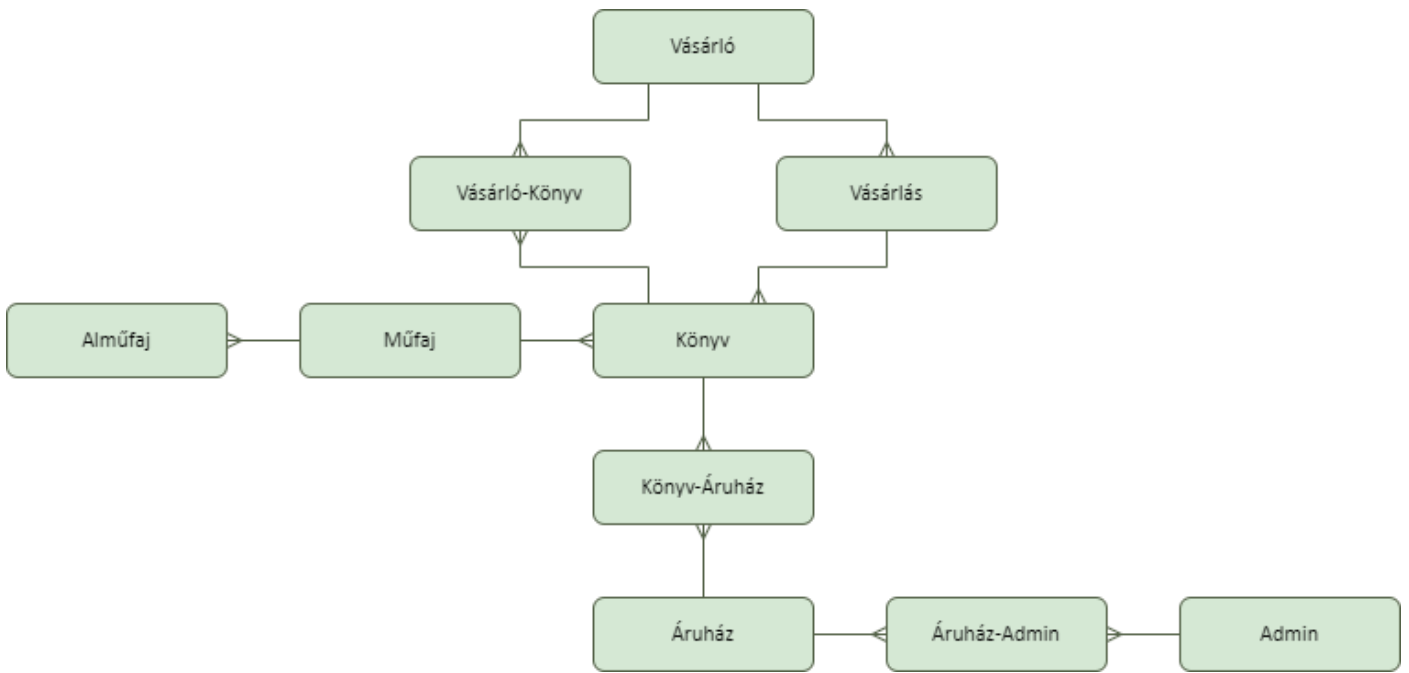
### ▪ A DFD 1. szintje



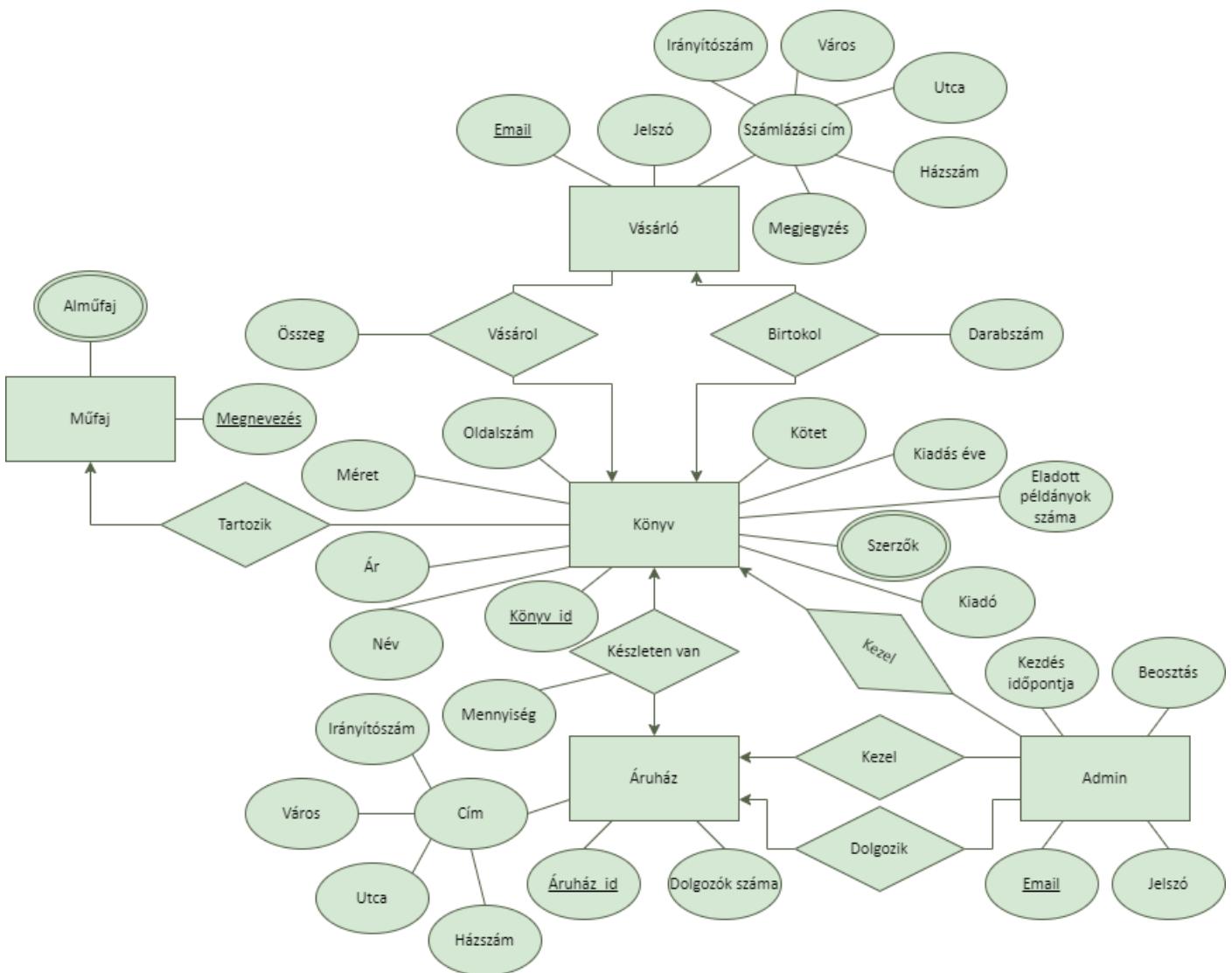
■ A DFD 2. szintje (Regisztráció és Bejelentkezés)



## Egyedmodell



## EKT-diagram





## Relációs adatelemzés

### ✧ Felülről lefelé történő elemzés:

- 1NF: Minden érték egyedi, nem szerepel többértékű mező.

Vásárló(Email, Jelszó, Irányítószám, Város, Utca, Megjegyzés, *Könyv\_id*)

Admin(Email, Jelszó, Kezdés\_időpontja, Beosztás, *Áruház\_id*)

Könyv(Könyv\_id, Név, Kiadás éve, Kiadó, Oldalszám, Méret, Kötet, Ár,

Eladott\_példányok\_száma, *Műfaj\_megnevezés*, *Áruház\_id*)

KönyvSzerző(Könyv\_id, Szerző)

Műfaj(Műfaj\_megnevezés)

Alműfaj(*Műfaj\_megnevezés*, Alműfaj)

Áruház(Áruház\_id, Irányítószám, Város, Utca, Házsám, Dolgozók\_száma, *Könyv\_id*)

- 2NF: Az összes attribútum közvetlenül függ az elsődleges kulcsoktól, nincs olyan másodlagos attribútum, amely részlegesen függ másodlagos attribútumtól.

Vásárló(Email, Jelszó, Irányítószám, Város, Utca, Megjegyzés)

VásárlóKönyv(Email, *Könyv\_id*)

Admin(Email, Jelszó, Kezdés\_időpontja, Beosztás)

AdminÁruház(Email, *Áruház\_id*)

Könyv(Könyv\_id, Név, Kiadás éve, Kiadó, Oldalszám, Méret, Kötet, Ár,

Eladott\_példányok\_száma)

KönyvSzerző(Könyv\_id, Szerző)

KönyvMűfaj(Könyv\_id, *Műfaj\_megnevezés*)

Műfaj(Műfaj\_megnevezés)

Alműfaj(*Műfaj\_megnevezés*, Alműfaj)

Áruház(Áruház\_id, Irányítószám, Város, Utca, Házsám, Dolgozók\_száma)

ÁruházKönyv(Áruház\_id, *Könyv\_id*)

- 3NF: Az összes attribútum közvetlenül függ az elsődleges kulcsoktól, nincs olyan másodlagos attribútum, amely tranzitívan függ másodlagos attribútumtól.

Vásárló(Email, Jelszó, Irányítószám, Város, Utca, Megjegyzés)

VásárlóKönyv(Email, *Könyv\_id*)

Admin(Email, Jelszó, Kezdés\_időpontja, Beosztás)

AdminÁruház(Email, *Áruház\_id*)

Könyv(Könyv\_id, Név, Kiadás éve, Kiadó, Oldalszám, Méret, Kötet, Ár,

Eladott\_példányok\_száma)

KönyvSzerző(Könyv\_id, Szerző)

KönyvMűfaj(Könyv\_id, *Műfaj\_megnevezés*)

Műfaj(Műfaj\_megnevezés)

Alműfaj(*Műfaj\_megnevezés*, Alműfaj)

Áruház(Áruház\_id, Irányítószám, Város, Utca, Házsám, Dolgozók\_száma)

ÁruházKönyv(Áruház\_id, *Könyv\_id*)

## ✧ Alulról felfelé történő elemzés:

Vásárlás(Email, Jelszó, Irányítószám, Város, Utca, Megjegyzés, Könyv\_id, Szerzők, Név, Kiadás éve, Kiadó, Oldalszám, Méret, Kötet, Ár, Eladott\_példányok\_száma, Műfaj\_megnevezés, Alműfaj, Áruház\_id)

Dolgozik(Email, Jelszó, Kezdés\_időpontja, Beosztás, Áruház\_id, Irányítószám, Város, Utca, Házszám, Dolgozók\_száma, Könyv\_id)

A Szerzők és az Alműfaj itt többértékű attribútum.

- 1NF: Minden érték egyedi, nem szerepel többértékű mező.

Vásárlás1(Email1, Jelszó, Irányítószám, Város, Utca, Megjegyzés, Könyv\_id, Név, Kiadás éve, Kiadó, Oldalszám, Méret, Kötet, Ár, Eladott\_példányok\_száma, Műfaj\_megnevezés, Áruház\_id)

KönyvSzerző(Könyv\_id, Szerző)

MűfajAlműfaj(Műfaj\_megnevezés, Alműfaj)

Dolgozik1(Email2, Jelszó, Kezdés\_időpontja, Beosztás, Áruház\_id, Irányítószám, Város, Utca, Házszám, Dolgozók\_száma, Könyv\_id)

- 2NF: Az összes attribútum közvetlenül függ az elsődleges kulcsoktól, nincs olyan másodlagos attribútum, amely részlegesen függ másodlagos attribútumtól.

Ez teljesül.

- 3NF: Az összes attribútum közvetlenül függ az elsődleges kulcsoktól, nincs olyan másodlagos attribútum, amely tranzitívan függ másodlagos attribútumtól.

{ Email1 } → { Könyv\_id } → { Kiadás éve, Kiadó, Oldalszám, Méret, Kötet, Ár, Eladott\_példányok\_száma, Műfaj\_megnevezés, Alműfaj }

{ Könyv\_id } → { Műfaj\_megnevezés } → { Alműfaj }

Vásárlás2(Email, Jelszó, Irányítószám, Város, Utca, Megjegyzés)

VásárlóKönyv(Email, Könyv\_id)

Dolgozik2(Email, Jelszó, Kezdés\_időpontja, Beosztás)

AdminÁruház(Email, Áruház\_id)

Könyv(Könyv\_id, Név, Kiadás éve, Kiadó, Oldalszám, Méret, Kötet, Ár, Eladott\_példányok\_száma)

KönyvSzerző(Könyv\_id, Szerző)

KönyvMűfaj(Könyv\_id, Műfaj\_megnevezés)

Műfaj(Műfaj\_megnevezés)

MűfajAlműfaj(Műfaj\_megnevezés, Alműfaj)

Áruház(Áruház\_id, Irányítószám, Város, Utca, Házszám, Dolgozók\_száma)

ÁruházKönyv(Áruház\_id, Könyv\_id)

## Táblák leírása

- ✧ **VÁSÁRLÓ:** A Vásárló tábla tárolja a vásárlók adatait, mint például az e-mail címet, amely a tábla elsődleges kulcsa is, illetve a jelszót, az irányítószámot, a várost, az utcát és egy esetleges megjegyzést a címhez.

Név	Típus	Leírás
email	varchar2	A Vásárló tábla kulcsa, a vásárló e-mail címe (egyedi azonosító).
jelszó	varchar2	A vásárló jelszava.
irányítószám	number	Vásárló lakcímének irányítószáma.
város	number	Vásárló lakcímének városa.
utca	number	Vásárló lakcímének utca neve.
megjegyzés	varchar2	Egyéb megjegyzések a címmel kapcsolatban.

- ✧ **VÁSÁRLÓKÖNYV:** A VásárlóKönyv tábla összekapcsolja a vásárlókat a megvásárolt könyvekkel, tárolva az vásárló e-mail címét és a könyv azonosítóját. Ez a két külső kulcs azonosítja. Valamint tárolja az adott vásárló mely könyvekből hány darabot vett már meg.

Név	Típus	Leírás
email	varchar2	A VásárlásKönyv tábla kulcsa.
könyv_id	number	A könyv tábla kulcsa, itt külső kulcs.
darabszám	number	A vásárló birtokolt könyveinek darabszáma.

- ✧ **ADMIN:** Az Admin tábla rögzíti a dolgozók adatait, beleértve az e-mail címet, amely a tábla elsődleges kulcsa is, illetve a jelszót, a kezdési időpontot és a beosztást.

Név	Típus	Leírás
email	varchar2	Dolgozó e-mail címe (egyedi azonosító).
jelszó	varchar2	Dolgozó jelszava.
kezdés_időpontja	date	A dolgozó kezdési időpontja.
beosztás	varchar2	Dolgozó beosztása.

- ✧ **ADMINÁRUHÁZ:** Az AdminÁruház tábla az adminisztrátorok és az áruházak közötti kapcsolatot rögzíti, az adminisztrátorok e-mail címe és az áruházak azonosítója alapján. Ez a két attribútum külső kulcsa a táblának.

Név	Típus	Leírás
email	varchar2	Dolgozó e-mail címe (külső kulcs az Admin táblából).
áruház_id	number	Áruház azonosítója (külső kulcs az Áruház táblából).

- ✧ **KÖNYV:** A Könyv tábla tárolja a könyvek adatait, mint például a nevét, kiadás évét, a kiadót, az oldalszámot, a méretet, a kötet számát, az árat és az eladott példányok számát. Elsődleges kulcsa a könyv\_id.

Név	Típus	Leírás
könyv_id	number	A Könyv tábla kulcsa.
név	varchar2	A könyv neve/címe.
kiadás_éve	number	A könyv kiadásának éve.
kiadó	varchar2	A kiadó neve.
oldalszám	number	A könyv oldalszáma.
méret	varchar2	A könyv mérete.
kötet	number	A könyv kötete (ha van).
ár	number	A könyv ára.
eladott_példányok_száma	number	Az eladott példányok száma.

- ✧ **KÖNYVSZERZŐ:** A KönyvSzerző tábla összekapcsolja a könyveket a szerzőkkel, tárolva a könyv azonosítóját, amely a tábla kulcsa is, és a szerző nevét.

Név	Típus	Leírás
könyv_id	number	Könyv azonosítója (külső kulcs a Könyv táblából).
szerző	varchar2	A könyv szerzője.

- ✧ **KÖNYVMŰFAJ:** A KönyvMűfaj tábla kapcsolja össze a könyveket a műfajokkal, tárolva a könyv azonosítóját és a műfaj megnevezését, ez a két külső kulcs azonosítja a táblát.

Név	Típus	Leírás
könyv_id	number	A Könyv tábla kulcsa, itt külső kulcs.
műfaj_megnevezés	varchar2	A Műfaj tábla kulcsa, itt külső kulcs.

- ✧ **MŰFAJ:** A Műfaj tábla tárolja a könyvek műfajainak megnevezéseit. Egy attribútuma van, amely egyben a tábla kulcsa is.

Név	Típus	Leírás
műfaj_megnevezés	varchar2	A Műfaj tábla kulcsa (egyedi azonosító).

- ✧ **ALMŰFAJ:** Az Alműfaj tábla kapcsolja össze a műfajokat az alműfajokkal. Az alműfaj\_megnevezés attribútum azonosítja, amely elsődleges kulcsa, a műfaj\_megnevezés külső kulcs.

Név	Típus	Leírás
műfaj_megnevezés	varchar2	A műfaj megnevezése (külső kulcs a Műfaj táblából).
alműfaj_megnevezés	varchar2	Az alműfaj megnevezése (egyedi azonosító).

- ✧ **ÁRUHÁZ:** Az Áruház tábla tárolja az áruházak adatait, mint például az áruház azonosítóját, amely a tábla elsődleges kulcsa is, illetve az irányítószámot, a várost, az utcát, a házszámot és a dolgozók számát.

Név	Típus	Leírás
áruház_id	number	Áruház azonosítója (egyedi azonosító).
irányítószám	number	Az áruház címének irányítószáma.
város	varchar2	Az áruház címének városa.
utca	varchar2	Az áruház címének utca neve.
házszám	number	Az áruház címének házszáma.
dolgozók_száma	number	Az áruház dolgozóinak száma.

- ✧ **ÁRUHÁZKÖNYV:** Az ÁruházKönyv tábla összekapcsolja az áruházakat a készleten lévő könyvekkel, tárolva az áruház azonosítóját és a könyv azonosítóját, melyek külső kulcsok. Valamint tárolja mely áruházakban hány darab érhető el az adott könyvből, ha ez a szám alacsony, úgy értesítést párosít hozzá, mely lehet "Nincs készleten" vagy "Ez az utolsó könyv".

Név	Típus	Leírás
áruház_id	number	A Áruház tábla kulcsa, itt külső kulcs.
könyv_id	number	A Könyv tábla kulcsa, itt külső kulcs.
készlet	number	Az könyv darabszáma egy adott áruházban.
értesítés	varchar2	A könyv készletéről értesítés.



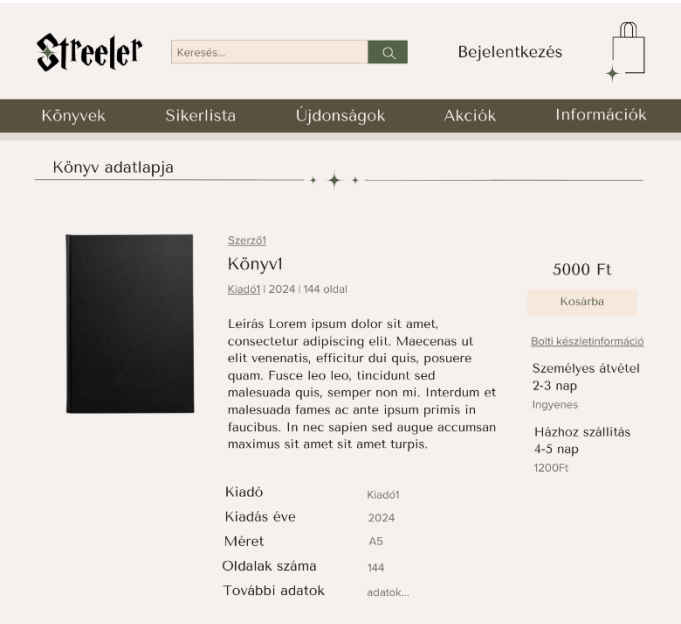
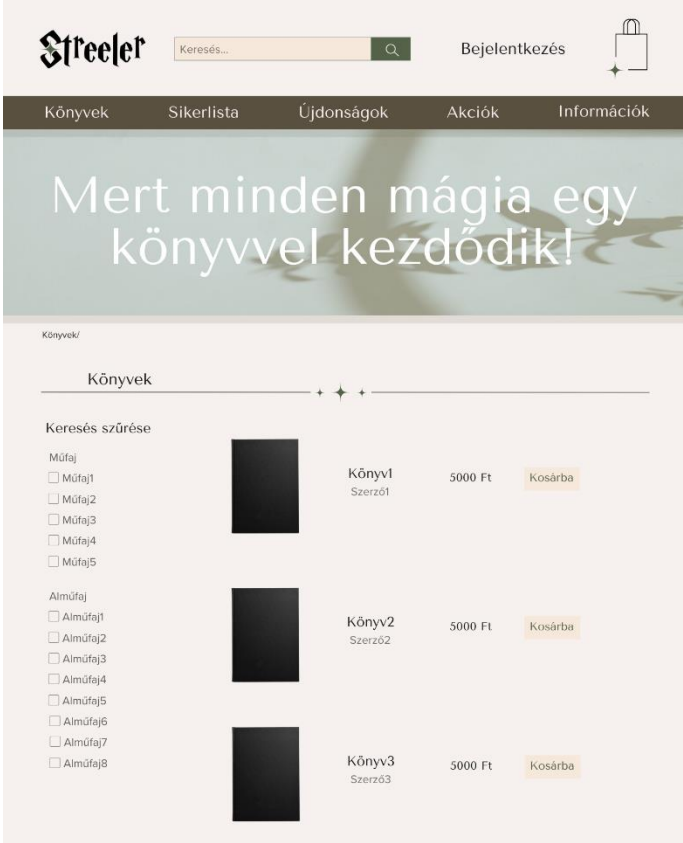
## Funkciók megadása


Funkció-meghatározás					SSADM-2
Projekt/rendszer:	Elemző:	Dátum:	Változat:	Állapot:	Oldal:
Könyvesbolt	Nagy P.	2024.03.06.	V1	munka	2
Funkciónév:			Funkció azonosító:		
Regisztráció			2.1		
Típus:					
Online, felhasználói					
Felhasználói szerepek:					
Jogosult: Látogató					
Funkció leírás:					
A regisztrációs funkció lehetővé teszi a felhasználók számára, hogy új fiókot hozzanak létre a könyvesboltban.					
Regisztrációs folyamat:					
<div>✧ A felhasználó megadja a regisztrációs űrlapon a következő adatokat:</div> <div><div>▪ Név</div><div>▪ Felhasználónév</div><div>▪ E-mail cím</div><div>▪ Jelszó</div><div>▪ Jelszó megerősítése</div></div> <div>✧ A rendszer ellenőrzi a beírt adatok érvényességét.</div> <div>✧ A rendszer elmenti a felhasználó adatait az adatbázisban.</div> <div>✧ A rendszer e-mailt küld a felhasználónak a regisztráció sikerességéről.</div>					
Hibakezelés:					
Ha a regisztrációs űrlap mezői üresen vannak hagyva, akkor erről üzenetablak jelenik meg, hogy az adott mező kitöltése kötelező.					
Ha érvénytelen e-mail címet ad meg a felhasználó, akkor erről üzenetablak jelenik meg, hogy érvénytelen e-mail címet adott meg.					
Ha a jelszó és a jelszó megerősítése mező nem egyezik, akkor erről üzenetablak jelenik meg, hogy a jelszavaknak egyezniük kell.					
AFD-eljárások: 2.1					
Események:			Esemény gyakorisága, Tömegszerűség:		
Minden alkalommal, amikor egy új felhasználó regisztrál a rendszerben.			Naponta átlagosan 4-6		
Követelménykatalógusra hivatkozás:					
1. követelmény					
Kapcsolódó funkciók:					
2.1 Regisztráció					
4.1 Kijelentkezés					
Lekérdezés gyakorisága:					
Felhasználói igényektől függ.					
Szolgáltatási szint követelményei:					
Leírás:		Célérték:		Tartomány:	
Regisztráció ideje		7 mp		2-15 mp	


Funkció-meghatározás					SSADM-3
Projekt/rendszer:	Elemző:	Dátum:	Változat:	Állapot:	Oldal:
Könyvesbolt	Nagy P.	2024.03.06.	V1	munka	1
Funkciónév:			Funkció azonosító:		
Bejelentkezés			3.1		
Típus:					
Online, felhasználói					
Felhasználói szerepek:					
Jogosult: Felhasználó					
Funkció leírás:					
A bejelentkezési funkció lehetővé teszi a regisztrált felhasználók számára, hogy belépjenek a fiókjukba és hozzáférjenek a könyvesbolt szolgáltatásaihoz.					
Bejelentkezési folyamat:					
✧ A felhasználó megadja a bejelentkezési nevét és jelszavát a bejelentkezési űrlapon.					
✧ A rendszer ellenőrzi a beírt adatokat az adatbázisban tárolt adatokkal.					
✧ Sikeres bejelentkezés esetén a rendszer átirányítja a felhasználót a fiókjába.					
✧ Sikertelen bejelentkezés esetén a rendszer hibaüzenetet jelenít meg.					
Hibakezelés:					
Ha a bejelentkezési név vagy jelszó mező üresen van hagyva, akkor erről üzenetablak jelenik meg, hogy a bejelentkezési név és a jelszó megadása kötelező.					
Ha bejelentkezési név vagy jelszó hibás, akkor erről üzenetablak jelenik meg, hogy hibás adat/adatok lettek megadva.					
AFD-eljárások: 3.1					
Események:			Esemény gyakorisága, Tömegszerűség:		
Minden alkalommal, amikor egy felhasználó bejelentkezik a rendszerbe.			Naponta átlagosan 16-24		
Követelménykatalógusra hivatkozás:					
1. követelmény					
Kapcsolódó funkciók:					
2.1 Regisztráció					
4.1 Kijelentkezés					
Lekérdezés gyakorisága:					
Felhasználói igényektől függ.					
Szolgáltatási szint követelményei:					
Leírás:		Célérték:		Tartomány:	
Bejelentkezés ideje		7 mp		2-15 mp	



Képernyőtervek





Könyvek
Sikerlista
Újdonságok
Akción
Információk

Kosár tartalma

Könyv1 Szerző1	5000 Ft	-	1	+	Törlés
Könyv2 Szerző2	5000 Ft	-	1	+	Törlés

Végösszeg: 10000 Ft

Szállítás módja

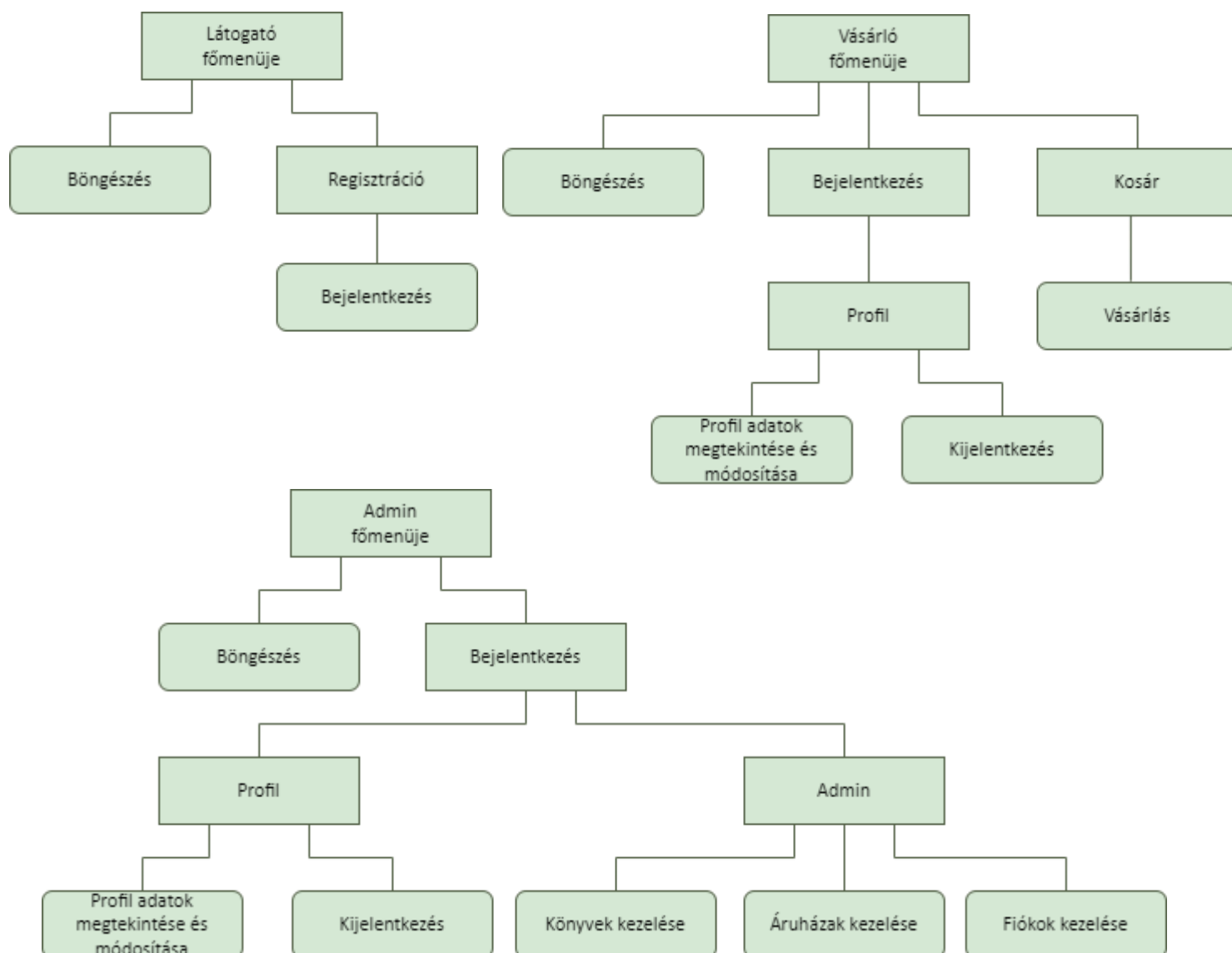
☐ Szállítás házhoz (+1200Ft)
☐ Boltban átvétel (Ingyenes)

Fizetés módja

☐ Bankkártyás fizetés
☐ Készpénzes fizetés

Fizetés

## Menütervek



## Trigger

A rendszer növeli a könyv eladott példány számát és csökkenti a készletet, ha a vásárló megveszi azt

```
CREATE OR REPLACE TRIGGER kosar_keszlet_aktualizalas
  AFTER INSERT OR UPDATE OR DELETE ON VasarloKonyv
  FOR EACH ROW
DECLARE
  max_keszlet NUMBER;
BEGIN
  SELECT MAX(keszlet) INTO max_keszlet
  FROM AruhazKonyv
  WHERE Konyv_id = :NEW.Konyv_id;

  IF INSERTING THEN
    UPDATE Konyv
    SET Eladott_peldanyok_szama = Eladott_peldanyok_szama + :NEW.Darabszam
    WHERE Konyv_id = :NEW.Konyv_id;

    UPDATE AruhazKonyv
    SET keszlet = max_keszlet - :NEW.Darabszam
    WHERE Konyv_id = :NEW.Konyv_id AND keszlet = max_keszlet;

  ELSIF UPDATING THEN
    DECLARE
      darabszam_valtozas NUMBER;
    BEGIN
      darabszam_valtozas := :NEW.Darabszam - :OLD.Darabszam;

      UPDATE Konyv
      SET Eladott_peldanyok_szama = Eladott_peldanyok_szama + darabszam_valtozas
      WHERE Konyv_id = :NEW.Konyv_id;

      UPDATE AruhazKonyv
      SET keszlet = max_keszlet - darabszam_valtozas
      WHERE Konyv_id = :NEW.Konyv_id AND keszlet = max_keszlet;
    END;

  ELSIF DELETING THEN
    UPDATE Konyv
    SET Eladott_peldanyok_szama = Eladott_peldanyok_szama - :OLD.Darabszam
    WHERE Konyv_id = :OLD.Konyv_id;

    UPDATE AruhazKonyv
    SET keszlet = max_keszlet + :OLD.Darabszam
    WHERE Konyv_id = :OLD.Konyv_id and keszlet = max_keszlet;
  END IF;
END;
```

A rendszer figyelmeztet, ha egy adott könyv készlete megfogyott

```
CREATE OR REPLACE TRIGGER KESZLET_KIMERULTSEG_FIGYELES
  BEFORE UPDATE OF Keszlet ON AruhazKonyv
  FOR EACH ROW
BEGIN
  IF :NEW.Keszlet = 0 THEN
    :NEW.Ertesites := 'Nincs készleten';
  ELSIF :NEW.Keszlet = 1 THEN
    :NEW.Ertesites := 'Ez az utolsó könyv';
  ELSE
    :NEW.Ertesites := NULL;
  END IF;
END;
```

## Tárolt eljárás / függvény

### Az 5 legújabb könyvet kiemeli a rendszer

```
create or replace PROCEDURE TOP5KONYV AS
    CURSOR c_konyv IS
SELECT K.KONYV_ID, K.NEV, K.AR, KS.SZERZO
FROM Konyv K
    INNER JOIN KonyvSzerzo KS ON K.Konyv_id = KS.Konyv_id
ORDER BY K.Konyv_id DESC
    FETCH FIRST 5 ROWS ONLY;
BEGIN
FOR r_konyv IN c_konyv LOOP
    DBMS_OUTPUT.PUT_LINE('Könyv ID: ' || r_konyv.KONYV_ID || ', Név: ' || r_konyv.NEV || ', Ár: ' || r_konyv.AR ||
', Szerző: ' || r_konyv.SZERZO);
END LOOP;
END TOP5KONYV;
```

### A Top 3 könyvet kiemeli a rendszer

```
create or replace PROCEDURE TOP3KONYV AS
    CURSOR c_konyv IS
SELECT K.KONYV_ID, K.NEV, K.AR, KS.SZERZO, K.ELADOTT_PELDANYOK_SZAMA
FROM Konyv K
    INNER JOIN KonyvSzerzo KS ON K.Konyv_id = KS.Konyv_id
ORDER BY K.ELADOTT_PELDANYOK_SZAMA DESC
    FETCH FIRST 3 ROWS ONLY;
BEGIN
FOR r_konyv IN c_konyv LOOP
    DBMS_OUTPUT.PUT_LINE('Könyv ID: ' || r_konyv.KONYV_ID || ', Név: ' || r_konyv.NEV || ', Ár: ' || r_konyv.AR ||
', Szerző: ' || r_konyv.SZERZO || ', Eladott példányok száma: ' || r_konyv.ELADOTT_PELDANYOK_SZAMA);
END LOOP;
END TOP3KONYV;
```

## Öszetett lekérdezések

### aru hazak\_ kezeles.php (279. sor)

```
$query = 'SELECT K.NEV AS Konyv_nev, COALESCE(SUM(AK.Keszlet), 0) AS Osszes_peldany
FROM Konyv K
LEFT JOIN AruhazKonyv AK ON K.Konyv_id = AK.Konyv_id
GROUP BY K.NEV';
$stmt = oci_parse(database(), $query);
oci_execute($stmt);
```

### aru hazak\_ kezeles.php (327. sor)

```
$query = 'SELECT A.Varos AS Varos, COUNT(*) AS Konyvek_szama, LISTAGG(K.Nev, \', \' WITHIN GROUP (ORDER BY K.Nev) AS
Konyvek_nevei
FROM Konyv K
    INNER JOIN AruhazKonyv AK ON K.Konyv_id = AK.Konyv_id
    INNER JOIN Aruhaz A ON AK.Aruhaz_id = A.Aruhaz_id
GROUP BY A.Varos';
$stmt = oci_parse(database(), $query);
oci_execute($stmt);
```

### profil.php (17. sor)

```
$query = "SELECT KS.Szerzo, COUNT(*) AS Vasarolt_konyvek_szama
FROM Vasarlo V
    INNER JOIN VasarloKonyv VK ON V.Vasarlo_email = VK.Vasarlo_email
    INNER JOIN KonyvSzerzo KS ON VK.Konyv_id = KS.Konyv_id
WHERE V.Vasarlo_email = '{$SESSION['username']}'
GROUP BY KS.Szerzo
ORDER BY COUNT(*) DESC
FETCH FIRST 1 ROW ONLY";
$result = oci_parse(database(), $query);
oci_execute($result);
```

### profil.php (30. sor)

```
$query = "SELECT M.Mufaj_megnevezes, COUNT(*) AS Vasarolt_konyvek_szama
FROM Vasarlo V
    INNER JOIN VasarloKonyv VK ON V.Vasarlo_email = VK.Vasarlo_email
    INNER JOIN KonyvMufaj KM ON VK.Konyv_id = KM.Konyv_id
    INNER JOIN Mufaj M ON KM.Mufaj_megnevezes = M.Mufaj_megnevezes
WHERE V.Vasarlo_email = '{$SESSION['username']}'
GROUP BY M.Mufaj_megnevezes
ORDER BY COUNT(*) DESC
FETCH FIRST 1 ROW ONLY";
$result = oci_parse(database(), $query);
oci_execute($result);
```

## oldalmenu.php (13. sor)

```
$query = "SELECT M.MUFAJ_MEGNEVEZES, COUNT(K.Konyv_id) AS konyvek_szama
FROM Mufaj M
LEFT JOIN KonyvMufaj KM ON M.MUFAJ_MEGNEVEZES = KM.Mufaj_megnevezes
LEFT JOIN Konyv K ON KM.Konyv_id = K.Konyv_id
GROUP BY M.MUFAJ_MEGNEVEZES";
$stmt = oci_parse($db, $query);
oci_execute($stmt);
```

## oldalmenu.php (33. sor)

```
$query = 'SELECT M.MUFAJ_MEGNEVEZES || \' \' (\'' || COUNT(K.Konyv_id) || '\')\' AS MEGNEVEZES, K.Konyv_id, K.NEV, K.AR,
KS.SZERZO
FROM Konyv K
INNER JOIN KonyvSzerzo KS ON K.Konyv_id = KS.Konyv_id
LEFT JOIN KonyvMufaj KM ON K.Konyv_id = KM.Konyv_id
LEFT JOIN Mufaj M ON KM.Mufaj_megnevezes = M.Mufaj_megnevezes';

if (isset($_GET['genres']) && !empty($_GET['genres'])) {
    $selectedGenres = implode(',', $_GET['genres']);
    $query .= " WHERE M.MUFAJ_MEGNEVEZES IN ('$selectedGenres')";
}

$query .= ' GROUP BY M.MUFAJ_MEGNEVEZES, K.Konyv_id, K.NEV, K.AR, KS.SZERZO
ORDER BY M.MUFAJ_MEGNEVEZES, K.NEV';

$stmt = oci_parse($db, $query);
oci_execute($stmt);
```

## keszlet.php (41. sor)

```
$query = "SELECT K.KONYV_ID, K.NEV, K.KIADAS_EVE, K.KIADO, K.OLDALSZAM,
KS.SZERZO, A.ARUHAZ_ID, A.IRANYITOSZAM, A.VAROS, A.UTCA, A.HAZSZAM,
A.DOLGOZOK_SZAMA , AK.ARUHAZ_ID, AK.KONYV_ID, AK.KESZLET, COUNT(A.ARUHAZ_ID) OVER () AS ARUHAZAK_SZAMA
FROM Konyv K
INNER JOIN KonyvSzerzo KS ON K.Konyv_id = KS.Konyv_id
LEFT JOIN AruhazKonyv AK ON K.KONYV_ID = AK.KONYV_ID
INNER JOIN Aruhaz A ON A.Aruhaz_id = AK.Aruhaz_id
WHERE K.KONYV_ID = :book_id
GROUP BY K.KONYV_ID, K.NEV, K.KIADAS_EVE, K.KIADO, K.OLDALSZAM, KS.SZERZO, A.ARUHAZ_ID,
A.IRANYITOSZAM, A.VAROS, A.UTCA, A.HAZSZAM, A.DOLGOZOK_SZAMA , AK.ARUHAZ_ID, AK.KONYV_ID, AK.KESZLET";
$stmt = oci_parse($db, $query);
oci_bind_by_name($stmt, ':book_id', $book_id);
oci_execute($stmt);
```

## kereses.php (11. sor)

```
$query = 'SELECT K.Konyv_id, K.NEV, K.AR, KS.SZERZO, COUNT(*) OVER () AS OSSZES_KONYV
FROM Konyv K
INNER JOIN KonyvSzerzo KS ON K.Konyv_id = KS.Konyv_id
LEFT JOIN KonyvMufaj KM ON K.Konyv_id = KM.Konyv_id
LEFT JOIN Mufaj M ON KM.Mufaj_megnevezes = M.Mufaj_megnevezes
WHERE (LOWER(K.NEV) LIKE LOWER(:kereses) OR
LOWER(KS.SZERZO) LIKE LOWER(:kereses) OR
LOWER(TRANSLATE(K.NEV, \'áéíóöúü\', \'aeioouu\')) LIKE LOWER(:kereses) OR
LOWER(TRANSLATE(KS.SZERZO, \'áéíóöúü\', \'aeioouu\')) LIKE LOWER(:kereses))
GROUP BY K.Konyv_id, K.NEV, K.AR, KS.SZERZO';
$stmt = oci_parse($db, $query);
$kereses_param = '%' . $kereses . '%';
oci_bind_by_name($stmt, ':kereses', $kereses_param);
oci_execute($stmt);
```

## sikerlista.php (22. sor)

```
$query = 'SELECT K.KONYV_ID, K.NEV, K.AR, KS.SZERZO, K.ELADOTT_PELDANYOK_SZAMA
FROM Konyv K
INNER JOIN KonyvSzerzo KS ON K.Konyv_id = KS.Konyv_id
ORDER BY K.ELADOTT_PELDANYOK_SZAMA DESC';
$stmt = oci_parse($db, $query);
oci_execute($stmt);
```

## ujdonsagok.php (22. sor)

```
$query = 'SELECT K.KONYV_ID, K.NEV, K.AR, KS.SZERZO
FROM Konyv K
INNER JOIN KonyvSzerzo KS ON K.Konyv_id = KS.Konyv_id
ORDER BY K.Konyv_id DESC FETCH FIRST 12 ROWS ONLY';
$stmt = oci_parse($db, $query);
oci_execute($stmt);
```

## Az alkalmazás telepítése

✧ **A projekt a következő technológiákat/szoftvereket fogja használni a munka során:**

- Oracle19
- Xampp
- PhpStorm (PHP 8.1 verzió)
- SQLDeveloper
- böngésző (Firefox/Chrome)