# Fejlesztői dokumentáció

## Adatbázis

## Backend

## Frontend

## LMS Desktop (asztali alkalmazás)

### Osztályok

#### Connection

* Az adatbázis adatai bele vannak írva a kódba, nincsenek titkosítva.
* Saját Select függvény, ami egy lekérdezést fogad és visszaad egy string tömbökből álló listát.
* RunSqlCommand függvény, ami a kapott lekérdezést egyszerűen lefuttatja.

#### HandleFiles

* Egy képfájl feltöltéséért felel a benne lévő Upload függvény.
  + Itt a kép átküldésre kerül a backendnek HttpClient segítségével.
  + Az esetleges hibákat json formátumban küldi vissza a backend.
  + Kivételkezelést is tartalmaz, ahol egyéb hibákat szűrünk ki.

#### HandleQueries

* Ez az osztály felel az egyes lekérdezések, beszúrások és frissítések megvalósítására.
  + A sima lekérdezések csak egy lekérdezést vagy egy fájlnevet várnak el. A fájlok az SqlQueries mappában találhatók.
  + A Delete függvény egy lekérdezés alapján kitörli kijelölt elemeket az adatbázisból.
  + Az összes többi függvény minden egyes Insert és Update műveletet valósít meg, ahol paraméterekben várja a feltöltendő / frissítendő adatokat.

#### Egyéb osztályok

* GeneratePassword -> Egy függvényt tartalmaz, ami a megadott karakterek és hossz alapján generál egy véletlenszerű jelszót.
* HandleGrids -> Függvényeket tartalmaz ComboBox és dataGridView feltöltésére, plusz egy megadott griden belüli keresésre.
* BorderPaint -> Keretet rajzol az adott form köré.
* CloseWindow -> Gombhoz rendelhető ablakbezárás, a megadott formot bezárja.
* DragWindow -> Lehetővé teszi az ablakok mozgatását címke (title) és panelek megadásával.
* HandleKeys -> A megadott gombra meghív egy megadott függvényt (pl.: Az Enter lenyomásakor elmenti a beállításokat, Escape gomb megnyomásakor pedig bezárja az ablakot).
* HandleFonts -> Beállítja a megadott form összes elemére a megadott betűtípust.

### Formok

#### Login

* Ez az első oldal, ami az alkalmazás indításakor megjelenik.
* Az alkalmazás az adatbázis nélkül nem fog elindulni.
* A LoginCheck függvényen belül ellenőrzésre kerülnek a felhasználó adatai. Ha a felhasználónév alapján található ilyen felhasználó, akkor BCrypt Verify segítségével összehasonlítjuk az adatbázisban tárolt titkosított jelszót a felhasználó által megadott jelszó titkosított változatával.
* Ha a felhasználó rossz adatot ad meg, arról tájékoztatja a program.
* A UsernameTextChanged és PasswordTextChanged függvények felelnek a valós idejű hibák kiírására a felhasználónév és jelszóval kapcsolatban.
* A ShowPassword függvény pedig a szem ikont megváltoztatja, és megjeleníti a jelszót.

#### Main

* A Main form felel az összes művelet megvalósítására.
* Minden adat kezelhető külön oldalakon (Dashboard, Books, Borrowings, Reservations, Categories, Authors, Publishers és Users).
* Itt kerülnek meghívásra az egyes SQL lekérdezések.

#### Books

* Az AddBook-ban meg lehet nyitni a ChooseAuthor és ChooseCategory ablakokat, ahol ki lehet választani egy vagy több szerzőt és kategóriát.
* Az adatok validálása után a könyv adatai feltöltésre kerülnek az adatbázisba (InsertBook függvény).
* Opcionálisan képet is lehet feltölteni:
  + A kép kiválasztásakor validálásra kerül a kép, itt ellenőrizve lesz a kép formátuma (png, jpg, jpeg), illetve a kép mérete (Max 5 MB).
  + Az ellenőrzés után át lesz másolva a temp mappába. A fájl neve itt a könyv isbn száma lesz.
  + Ezt követően meghívásra kerül a HandleFiles osztály Upload függvénye.
* Az EditBook szinte teljesen megegyezik az AddBook formmal, azzal a különbséggel, hogy ide betöltődnek a kijelölt könyv adatai (UpdateBook függvény).
* A RemoveBook egész egyszerűen eltávolítja a könyvet az adatbázisból (Delete függvény).

#### Borrowings

* Az AddBorrowingban meg lehet nyitni a könyvek kiválasztására szolgáló ChooseBooks formot, ebből a SelectedISBNs nevű mezőből betöltésre kerülnek az ISBN számok a megfelelő szövegdobozba.
* A validációt követően a dátumok megfeleő formátumba lesznek váltva (InsertBorrowing függvény).
* Az ExtendBorrowing-ban a jelenlegi határidőhöz hozzáadásra kerül a felhasználó által megadott hónapok száma (UpdateBorrowing függvény).
* A ReturnBorrowings-ban a könyv visszahozásának dátuma mindig aznap lesz (Update Borrowing overloaded függvény).

#### Reservations

#### Authors

* Az AddAuthor-ban a validációt követően feltöltésre kerül a szerző (InsertAuthor függvény).
* Az EditAuthor-ban betöltődik a kiválasztott szerző, majd a validációt követően módosításra kerül (UpdateAuthor függvény).
* A RemoveAuthor a paraméterben kapott szerzőt eltávolítja (DeleteFüggvény).