

Könyvtári nyilvántartó

Alapkövetelmények

- Git repo
 - Rendszeres commit minden csapattagtól (egyéni értékelés)
- Egy solution használata
 - Legalább .NET 6.0 használata
 - Legalább három külön projekt megléte a feladat követelményei alapján
 - ASP.NET Core Web API
 - Blazor WebAssembly App (2x)
- Kódolási konvenciók alkalmazása:
 - StyleCop.Analyzers nuget package használata az összes projektben
<https://www.nuget.org/packages/StyleCop.Analyzers/>
- A solution buildelhető

Bármely alapkövetelmény hiánya esetén a feladat értékelhetetlennek minősül! (Elégtelen érdemjegy)

Feladatléírás

A feladat egy könyvtárban működő Webes alkalmazás implementálása.

A feladat három részből áll. Egy a könyvtáros pultján működő felhasználói felület ahol láthatja a könyvtárban nyilvántartott könyvek listáját, könyvtári tagokat tud regisztrálni és tudja kezelni az olvasók adatait, kölcsönzéseit. Egy másik felület, ami a könyvtárban lévő ügyfelek számára fenntartott gépeken működik, az olvasó lekérheti a kölcsönözhető és kikölcsönzött könyvek listáját. Az alkalmazás backend-jét egy SQL adatbázis alkotja, ebben tárolja az adatokat amit egy Web API applikáció endpoint-jain keresztül módosíthatunk és érhetünk el. Az API-n keresztül kell történnjen minden adattal kapcsolatos művelet és lekérdezés.

- Könyvek adatai
 - Cím
 - Szerző
 - Kiadó
 - Leltári szám
 - Kiadás éve
- Könyvtár tagok adatai
 - Név
 - Nem lehet: üres, whitespace, különleges karakterek szűrése pl !?_~;#
 - Lakcím
 - Olvasószám
 - Születési dátum
- Kölcsönzések adatai
 - Olvasószám (Tag)
 - Leltári szám (Könyv)
 - Kölcsönzés ideje
 - Visszahozás határideje

- Csak későbbi dátum lehet mint a kölcsönzés ideje (DateTime típus használata)

Szerver - ASP.NET Core WEB API alkalmazás

Az adatbázisban tárolt adatok az API-n keresztül érhetők el. A Blazor kliensek az API végpontjait hívva érik el illetve módosítják az adatokat. Kölcsönzött könyvek és tagok kezelése itt történik, az API tárolja le az adatokat adatbázisban.

- Adatbázis Entity Framework Core használatával (MSSQL LocalDB)
- CRUD műveletek implementálása a feladatnak megfelelően
- Endpointokon beérkező adatok validálása backend oldalon
- UNIT tesztek írása minden validációs logikára

Könyvtáros kliens - Blazor WebAssembly alkalmazás

A könyvtáros pultján működik, látja a könyvtárban lévő könyvek listáját. A felületen képes új tagokat regisztrálni, kezelni az adataikat és könyveket kiadni, visszavenni tőlük.

Követelmények:

- A tagok táblázatos formában jelennek meg
 - Új tagot tud hozzáadni
 - Meg tudja nézni a tagok adatait
 - Látja a kikölcsönzött könyveket
 - Legyen kiemelve, ha egy könyv nem lett visszaadva időben
- A könyvek táblázatos formában jelennek meg
 - Adott állapot lekérdezése könyvenként
 - Bent van vagy ki van kölcsönözve
 - Kinél van kikölcsönözve
 - Mikorra kell visszavinnie
- Az érkező ügyfélhez ki tud adni vagy visszavenni könyvet
 - Csak bent lévő könyveket tud kiadni
 - A könyv állapota megváltozik kölcsönzöttre
 - Letárolódnak az kölcsönzés adatai
- UNIT tesztek írása a bevitt adatok validációjára

Ügyfél kliens - Blazor WebAssembly alkalmazás

Az ügyfelek ezen a felületen név megadásával tájékoztatást kaphatnak a kölcsönzéseiről, megtekinthetik a könyvtárban nyilvántartott könyvek listáját és azok kölcsönzési állapotát.

- A kölcsönözhető könyvek táblázatos formában jelennek meg
 - Adott állapot lekérdezése könyvenként
 - Bent van vagy ki van kölcsönözve
 - Kinél van kikölcsönözve
 - Mikorra kell visszavinnie
- Az általa kikölcsönzött könyvek adatai
- Határidők

- Időrendi sorrendben dátum és idő szerint rendezve (Az adatok hozzáadási sorrendje nem minősül dátum szerinti rendezésnek)