

**WatchLux Projekt Dokumentáció**

Készítette: Makra Martin, Szilágyi István, Berényi Gergő

**1. A szoftver célja**

A WatchLux projekt egy olyan óra webáruház, amelyhez egy háttérrendszer is tartozik, ahol az adminok tudják kezelni a termékeket, a rendeléseket és a felhasználókat. A célunk az volt, hogy egy átlátható, könnyen kezelhető, modern webshopot hozzunk létre, ami jól működik mind a vásárlók, mind az üzemeltetők számára.

**A rendszer három fő részből áll:**

**- Webes felület (frontend):** itt böngészhetők a termékek, lehet szűrni, keresni, regisztrálni, belépni, kosárba rakni és rendelni.

**- Szerveroldal (backend):** ez kezeli az adatokat, API-kon keresztül kommunikál a frontenddel, és gondoskodik arról, hogy minden működjön.

**- Asztali admin alkalmazás:** itt tudják az adminok szerkeszteni az adatokat, emellett a titkári rangokat beosztani a korlátozott hozzáféréshez és új órákat felvenni, megrendeléseket kezelni.

A backend különféle szűrést és kulcsszavas keresést is tud, és rugalmasan kezeli a különböző terméktulajdonságokat (mint például márka, típus, vízállóság stb.). A rendszer minden része MySQL adatbázist használ.

A cél tehát nem csak egy sima webshop létrehozása volt, hanem egy jól működő, komplex rendszeré, amit a valóságban is könnyen lehetne használni.

**2. A rendszer főbb komponenseinek technikai leírása**

A WatchLux projekt három fő részből épül fel: egy webes vásárlói felületből (frontend), egy szerveroldali háttérrendszerből (backend), valamint egy asztali adminisztrációs alkalmazásból. Ezek a komponensek egymással együttműködve biztosítják a rendszer teljes funkcionalitását.

**2.1 Frontend – Webes felület (React)**

Ez a rész az, amit a látogatók, vásárlók használnak. A React alapú frontend biztosítja a gyors, interaktív és modern felhasználói élményt. A felület mobilbarát (reszponzív), vagyis telefonon és asztali gépen is jól használható.

**A főbb funkciók:**

- Terméklista megjelenítése: képekkel, árakkal, részletes adatokkal.

- Szűrés és keresés: különböző szempontok (pl. márka, ár, nem, meghajtás) alapján.

- Felhasználói fiók: regisztráció, bejelentkezés, profiladatok kezelése.

- Kosár és rendelés: termékek hozzáadása, mennyiség módosítása, megrendelés véglegesítése.

A felület Axios segítségével kommunikál a backenddel (REST API-n keresztül), az adatokat JSON formátumban kapja meg.

**2.2 Backend – Szerveroldal (Node.js + MySQL)**

A szerveroldal felelős a kapcsolati logika és az adatbázis-kezelés lebonyolításáért. A backend Node.js környezetben készült, az adatbázis pedig MySQL alapú.

**Fő jellemzők:**

- REST API: minden frontend kéréshez külön végpont tartozik (GET, POST, stb.).

- Dinamikus szűrés: a backend képes tetszőleges kombinációjú szűrőfeltételek alapján lekérni a termékeket.

- Kulcsszavas keresés: több oszlopban történő keresés egyetlen mezőből (pl. megnevezés, típus, márka, stb.).

- Adatbiztonság: jelszavak SHA-256 hash-eléssel kerülnek eltárolásra.

- Ár intervallum lekérdezés: automatikusan számolt minimum és maximum ár alapján.

A backend biztonságosan kapcsolódik az adatbázishoz a mysql2/promise csomag segítségével, és minden lekérdezés paraméterezett, hogy megelőzze az SQL injection támadásokat.

**2.3 Asztali admin alkalmazás – C# WPF**

A rendszer harmadik komponense egy Windows-os asztali alkalmazás, amelyet az adminisztrátorok és titkárok használnak. A program C# nyelven, WPF- ben készült.

**Fő funkciók:**

- Termékek kezelése: új órák felvétele, meglévők módosítása vagy törlése ugyan ezt a táblák kezelésével.

- Felhasználók és vásárlók kezelése: adatok megtekintése, módosítása, új felhasználók felvétele.

- Rendelések megtekintése: megrendelések listázása, részletek megtekintése.

- Szűrés és keresés: termékek és vásárlók között is lehet keresni és szűrni különböző feltételek alapján.

A felület átlátható, logikusan felépített, minden mezőhöz megfelelő validáció tartozik. Az alkalmazás közvetlenül a MySQL adatbázissal kommunikál.

**3. Működés műszaki feltételei**

A rendszer működéséhez a backendet és a frontendet külön kell elindítani Visual Studio Code-ban. Mindkettő Node.js alapú, ezért az indulás előtt telepíteni kell a szükséges csomagokat.

**Backend elindítása**

1. Visual Studio Code-ban nyisd meg a backend mappát.

2. Jobb kattintás az üres részre, majd válaszd: Open in Integrated Terminal.

3. A terminálba írd be:

elsőnek: npm install

majd az előzőt követően: npm start

4. Ez elindítja a backend szervert, ami az adatokat az adatbázisból fogja kiszolgálni.

**Frontend elindítása**

1. Nyisd meg a frontend mappát is Visual Studio Code-ban.

2. Ismét jobb kattintás → Open in Integrated Terminal.

3. A terminálba írd be:

elsőnek: npm install

majd az előzőt követően: npm start

4. Ez elindítja a webes felületet, amit a böngészőben lehet használni (http://localhost:3000 címen).

**Adatbázis beállítása**

A rendszerhez szükséges egy 'ora' nevű MySQL adatbázis. Ezt a .sql fájlt importálni kell például XAMPP segítségével.

Felhasználónév: root

Jelszó: root

Ez a beállítás azért ilyen egyszerű, hogy a vizsgáztatók gyorsan ki tudják próbálni a rendszert. Természetesen ez bármikor megváltoztatható a dbconfig.js vagy. env fájlban és akár létrehozható külön felhasználó is.

**Admin panel (WPF alkalmazás)**

Az admin felület egy különálló Windows alkalmazás, amit egy installer segítségével lehet elindítani. A program használatához is szükséges, hogy az adatbázis elérhető legyen, tehát az 'ora' adatbázisnak futnia kell.

**Fontos:** a rendszer minden része úgy lett kialakítva, hogy a vizsgáztatók számára könnyen tesztelhető legyen. A beállítások (pl. adatbázis felhasználónév, jelszó) egyszerűek, de biztonságosabb értékekre is lecserélhetők a fájlokban.

A működéshez elég egy alap Node.js és MySQL környezet, Visual Studio Code és Visual Studio használata ajánlott.

A rendszer modularitása lehetővé teszi, hogy később könnyen bővíthető vagy módosítható legyen.

**4. A rendszer használatának rövid bemutatása**

A WatchLux webáruház használata egyszerű és gyors. A látogatók a kezdőlapon a termékek fülre átkattintva azonnal böngészhetnek az órák között. Lehetőség van különböző szűrési feltételek beállítására, például márka, ár vagy típus szerint. A termékeket listában látják, képpel és alapinformációkkal. Egy termékre kattintva megjelenik a részletes oldal, ahol hozzá lehet adni a kosárhoz és a kedvencekhez is.

A kosárban szerkeszthető a darabszám azaz mennyiség és törölni is lehet a kosárból a terméket.

A vásárláshoz nem szükséges a regisztráció vagy bejelentkezés hiszen vendég vásárlóként is lehet vásárolni. A felhasználó a saját fiókjában módosíthatja adatait, és megnézheti korábbi rendeléseit is. A rendelés leadása után visszaigazolást kap.

Az admin panel egy különálló alkalmazás, amelyben az adminisztrátor vagy a office kolléga hozzáadhat új órákat, módosíthatja a meglévőeket, kezelheti a vásárlók adatait és nyomon követheti a megrendeléseket emellett fontos kiemelni hogy csak az admin jogosultságú alkalmazott/ tulaj tudja kezelni a felhasználókat. Az adminisztrációs felület a bejelentkezésnél leellenőrzi a jogosultságot az adatbázisban a felhasznalok táblában ez admin vagy office jogosultság lehet ennek megfelelően tölti be a rendszer a szerkeszthető ablakokat. Az admin felület egyszerűen kezelhető, minden funkció pár kattintással elérhető. A bejelentkezések és letöltések menete és használata:

**4.1 adatbázis használata:**

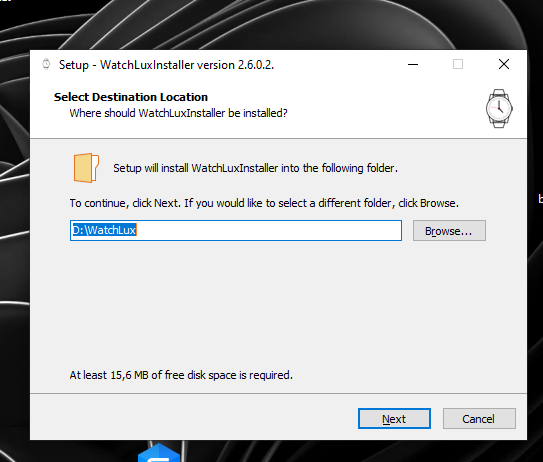
1: ora adatbázis restore, a restore-nál csak simán végig kell kattintani next-el és a végén finish gomb.

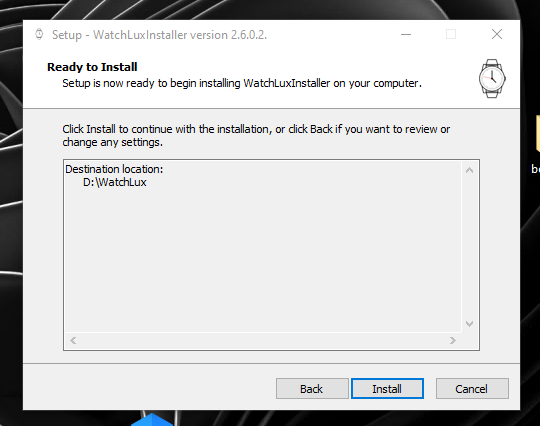
Innentől kezdve az adatbázishoz nem kell nyúlnunk.

**4.2 Adminisztrációs panel:**

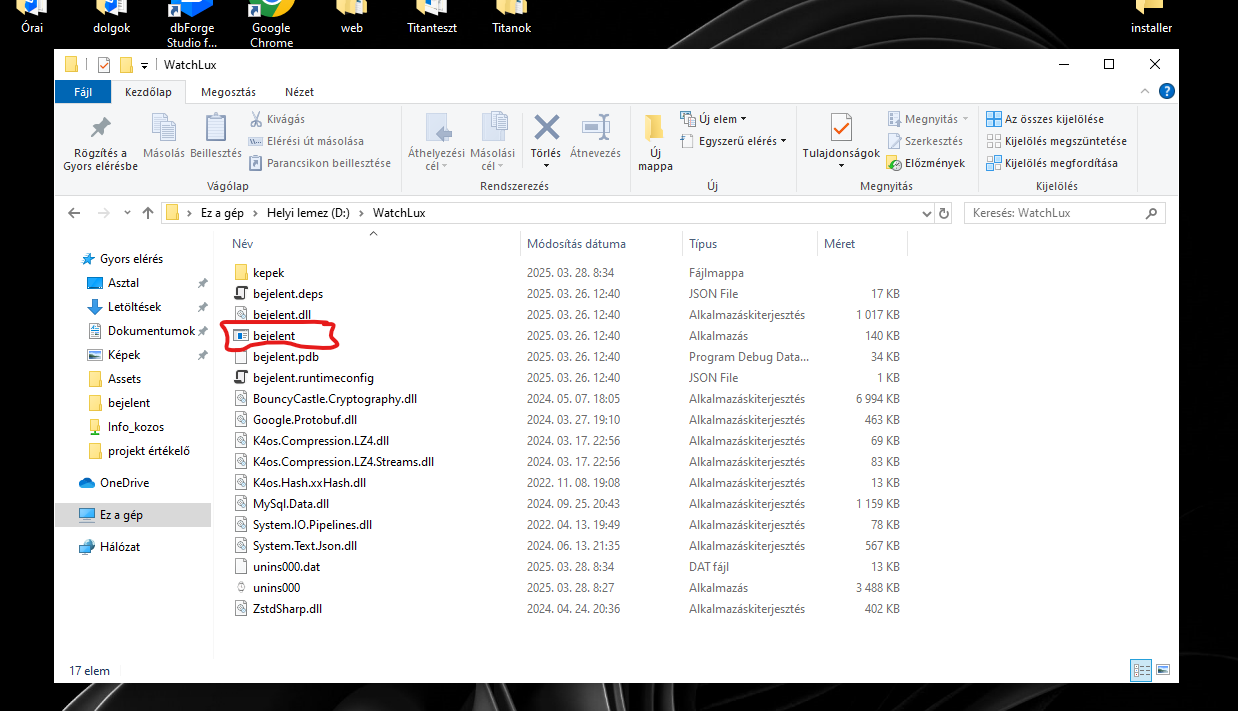
1: A github-on fent található WatchLuxInstallert kell letölteni és a telepítést elindítani a next gombal. A gomb megnyomása után lehetőségünk van kiválasztani a mentés helyét ezután az install gomb megnyomásával telepítjük a programot sikeres telepítés után ha a tálcán lévő keresőbe beírjuk a WatchLux szót ki fogja adni a programot dupla kattintással indul is az adminisztrációs panel. Ha a kereső sávban nem jelenik meg a panel akkor ahova a mentés sikerült ott tudjuk elindítani a exe-t.

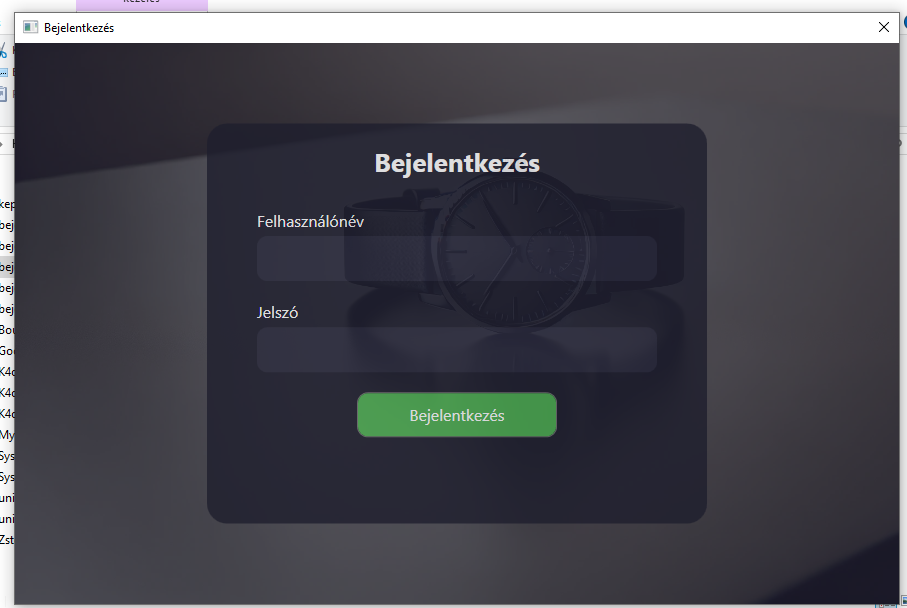
1:



2:

3:



 Sikeres elindítás után ezt a panelt kell látnunk:

A bejelentkezésnél 2 jogosultság van admin és office rang. Az admin tudja csak kizárólagosan szerkeszteni a felhasználókat új titkár felvétele/törlése/módosítása ha office jogosultságú ember lép be akkor ő csak a órához és számlához való táblákat tudja szerkeszteni.

A teszteléshez való jelszók és felhasználónevek:

1: Admin bejelentkezés: Felhasználónév: Adminteszt Jelszó: Adminteszt

2: Ofiice bejelentkezés: Felhasználónév: Officeteszt Jelszó: Officeteszt

Sikeres bejelentkezés után 3 tabitem vár minket: Adminisztráció, Óra Tábla, Számlázás

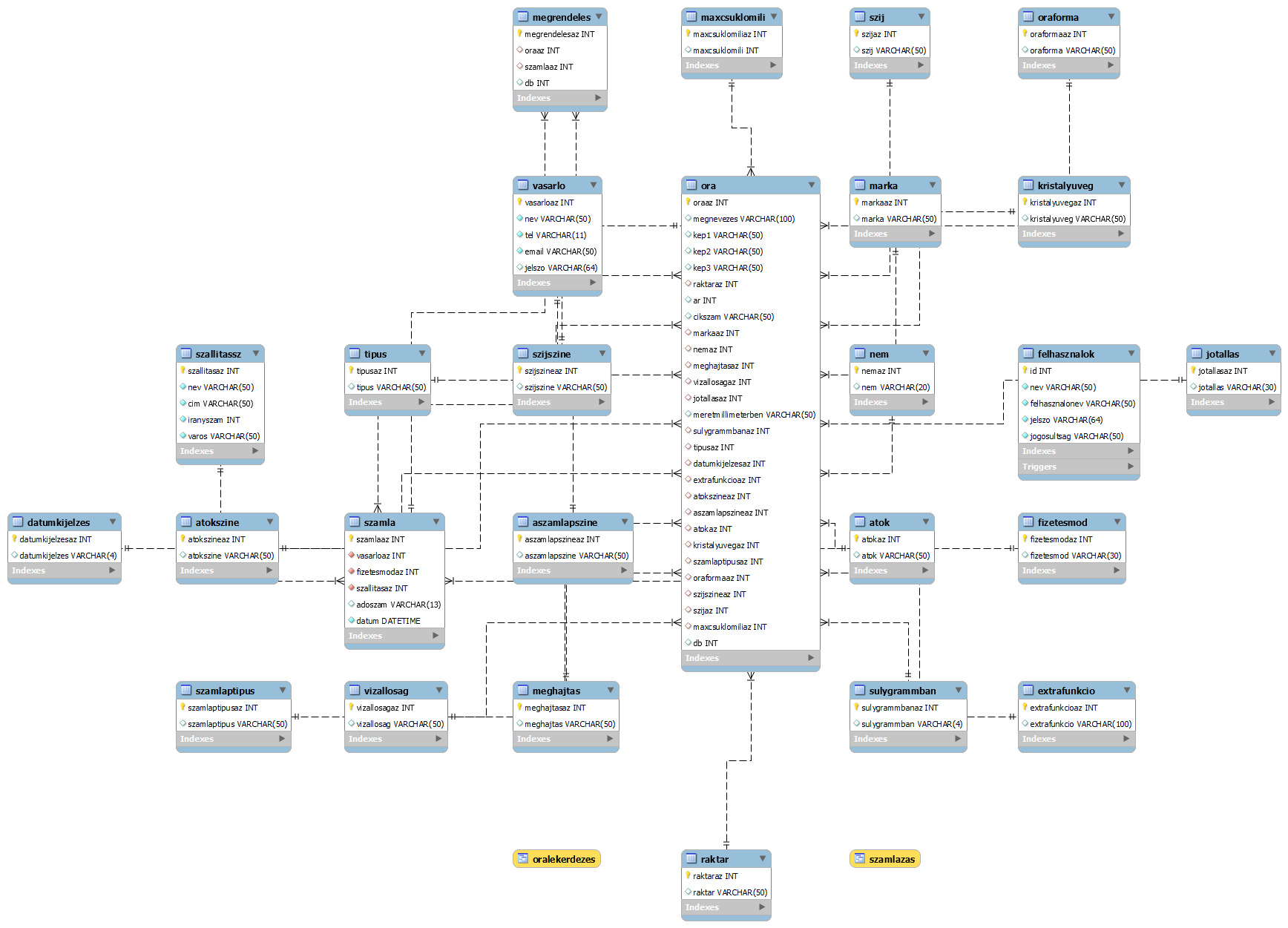
Adminisztráció TabItem-ben tudjuk az adatbázis táblák tartalmait mint pl. egy új szín felvétele vagy jelenlegi módosítása avagy törlése

Óra TabItem-ben tudjuk az adatbázisban az ora táblában a órákat módosítani pl. egy új óra felvétele vagy jelenlegi módosítása avagy törlése.

Számlázás fülön tudunk módosítani egy létező számlát vagy egy újat hozzáadni vagy törölni.

**4.3 Frontend működése:**

A frontendben a üdvözlő panel fogad minket ezután tudunk a termékek közül választani és szűrni, hozzáadni a kosárhoz és a kedvencekhez. Tudunk bejelentkezni vagy regisztrálni és befejezni a rendelést de tudunk vendégként és bejelentkezett vásárlóként is rendelést leadni.

**5 adatbázis leírása:**

**5.1 Bevezetés**

Az adatbázis egy óra webáruház működését támogatja. A vásárlók regisztrálhatnak, rendeléseket adhatnak le, és az adatbázis eltárolja a számlázási és szállítási adatokat is. Az adatbázis szerkezete normalizált, vagyis jól szervezett, és külön kezeli a termékek adatait, a felhasználókat, a rendeléseket és a jogosultságokat.

**5.2 Táblák áttekintése**

Az adatbázis összesen 26 táblából áll:

- Felhasználók és jogosultságok (pl. felhasznalok tábla)

- Vásárlók és szállítási címek (vasarlo, szallitassz)

- Számlák és megrendelések (szamla, megrendeles)

- Termékek és jellemzőik (ora, marka, szij, meghajtas, stb.)

Emellett két nézet (view) is készült az egyszerűbb lekérdezésekhez:

- szamlazas: a számlák és kapcsolódó adatok egyszerűsített nézete

- oralekerdezes: az órák részletes adatainak megjelenítésére

**5.3 Kapcsolatok és kulcsok**

Minden tábla rendelkezik elsődleges kulccsal (PRIMARY KEY), amely egyedileg azonosítja a rekordokat. Ezek jellemzően automatikusan növekvő (AUTO\_INCREMENT) értékek, például felhasznalok.id vagy szamla.id.

Ezen kívül több tábla használ egyedi mezőket is (UNIQUE), mint például felhasznalonev vagy email, így ugyanaz az érték nem kerülhet be többször.

A táblák között idegen kulcsok (FOREIGN KEY) biztosítják a kapcsolatokat. Például:

- egy szamla egy adott vásárlóhoz tartozik,

- egy megrendeles egy konkrét órához kapcsolódik.

Ezekhez ON DELETE CASCADE vagy SET NULL szabályok tartoznak, hogy törléskor ne maradjanak "árva" adatok.

**5.4 Nézetek (Views)**

A nézetek olyan előre beállított lekérdezések, amik segítik az admin felület vagy a frontend működését. Az adatbázisban két nézet található:

- szamlazas: megkönnyíti a számlák és vásárlói adatok lekérdezését.

- oralekerdezes: részletes információt ad az órákról (pl. márka, típus, ár stb.).

**5.5 Indexek**

Az adatbázisban több index is van a gyorsabb keresés érdekében. Ezek olyan mezőkre vannak létrehozva, amelyeket gyakran használunk szűrésre vagy lekérdezésre (pl. felhasznalonev, email), így a rendszer gyorsabban tud válaszolni a kérésekre.

**5.6 Jogosultságkezelés**

A felhasznalok tábla tartalmazza a bejelentkezéshez szükséges adatokat. Minden felhasználóhoz tartozik egy jogosultsági szint (jogosultsag mező), például admin vagy ügyfélszolgálat.

A jelszavak SHA-256 hash-eléssel kerülnek mentésre az adatbázisba, így nem olvashatók vissza. A jogosultságkezelés nem SQL oldalon, hanem a programkódban (backend és admin) történik.

**6 Teszteredmények:**

Az adminisztrációs felületen minden művelethez csatolva van hibaüzenet ami segítségével megállapítottuk hogy a adatbázishoz hiba nélkül lehet hozzáadni, módosítani és törölni. A manuális teszt eredménye:

|  |  |  |  |
| --- | --- | --- | --- |
| Tesztelt funkció | Bemenet / művelet | Várt eredmény | Teszt eredménye |
| Bejelentkezés | Helyes felhasználónév és jelszó megadása | Belépés sikeres, admin felület megjelenik | Sikeres |
| Bejelentkezés | Hibás felhasználónév és jelszó megadása | Belépés sikertelen, admin felület nem jelenik meg vissza dob a bejelentkezéshez | Sikeres |
| Kijelentkezés | Kijelentkezés gomb megnyomása | Visszairányítás a bejelentkezési felületre | Sikeres |
| Felhasználók hozzáadása | Kitöltött adatmezők mentése | Felhasználók megjelenik az adatbázisban | Sikeres |
| Felhasználók módosítása | Meglévő adat módosítása | Felhasználók adatai frissülnek az adatbázisban | Sikeres |
| Felhasználók törlése | Törlés gomb megnyomása | Felhasználók eltűnik az adatbázisból | Sikeres |
| Tok hozzáadása | Kitöltött adatmezők mentése | Tok megjelenik az adatbázisban | Sikeres |
| Tok módosítása | Meglévő adat módosítása | Tok adatai frissülnek az adatbázisban | Sikeres |
| Tok törlése | Törlés gomb megnyomása | Tok eltűnik az adatbázisból | Sikeres |
| Vásárlók hozzáadása | Kitöltött adatmezők mentése | Vásárlók megjelenik az adatbázisban | Sikeres |
| Vásárlók módosítása | Meglévő adat módosítása | Vásárlók adatai frissülnek az adatbázisban | Sikeres |
| Vásárlók törlése | Törlés gomb megnyomása | Vásárlók eltűnik az adatbázisból | Sikeres |
| Szállítási adatok hozzáadása | Kitöltött adatmezők mentése | Szállítási adatok megjelenik az adatbázisban | Sikeres |
| Szállítási adatok módosítása | Meglévő adat módosítása | Szállítási adatok adatai frissülnek az adatbázisban | Sikeres |
| Szállítási adatok törlése | Törlés gomb megnyomása | Szállítási adatok eltűnik az adatbázisból | Sikeres |
| Fizetésmód hozzáadása | Kitöltött adatmezők mentése | Fizetésmód megjelenik az adatbázisban | Sikeres |
| Fizetésmód módosítása | Meglévő adat módosítása | Fizetésmód adatai frissülnek az adatbázisban | Sikeres |
| Fizetésmód törlése | Törlés gomb megnyomása | Fizetésmód eltűnik az adatbázisból | Sikeres |
| Szállítás hozzáadása | Kitöltött adatmezők mentése | Szállítás megjelenik az adatbázisban | Sikeres |
| Szállítás módosítása | Meglévő adat módosítása | Szállítás adatai frissülnek az adatbázisban | Sikeres |
| Szállítás törlése | Törlés gomb megnyomása | Szállítás eltűnik az adatbázisból | Sikeres |
| Vízállóság hozzáadása | Kitöltött adatmezők mentése | Vízállóság megjelenik az adatbázisban | Sikeres |
| Vízállóság módosítása | Meglévő adat módosítása | Vízállóság adatai frissülnek az adatbázisban | Sikeres |
| Vízállóság törlése | Törlés gomb megnyomása | Vízállóság eltűnik az adatbázisból | Sikeres |
| Típus hozzáadása | Kitöltött adatmezők mentése | Típus megjelenik az adatbázisban | Sikeres |
| Típus módosítása | Meglévő adat módosítása | Típus adatai frissülnek az adatbázisban | Sikeres |
| Típus törlése | Törlés gomb megnyomása | Típus eltűnik az adatbázisból | Sikeres |
| Szíj színe hozzáadása | Kitöltött adatmezők mentése | Szíj színe megjelenik az adatbázisban | Sikeres |
| Szíj színe módosítása | Meglévő adat módosítása | Szíj színe adatai frissülnek az adatbázisban | Sikeres |
| Szíj színe törlése | Törlés gomb megnyomása | Szíj színe eltűnik az adatbázisból | Sikeres |
| Szíj fajtája hozzáadása | Kitöltött adatmezők mentése | Szíj fajtája megjelenik az adatbázisban | Sikeres |
| Szíj fajtája módosítása | Meglévő adat módosítása | Szíj fajtája adatai frissülnek az adatbázisban | Sikeres |
| Szíj fajtája törlése | Törlés gomb megnyomása | Szíj fajtája eltűnik az adatbázisból | Sikeres |
| Súly grammban hozzáadása | Kitöltött adatmezők mentése | Súly grammban megjelenik az adatbázisban | Sikeres |
| Súly grammban módosítása | Meglévő adat módosítása | Súly grammban adatai frissülnek az adatbázisban | Sikeres |
| Súly grammban törlése | Törlés gomb megnyomása | Súly grammban eltűnik az adatbázisból | Sikeres |
| Nem hozzáadása | Kitöltött adatmezők mentése | Nem megjelenik az adatbázisban | Sikeres |
| Nem módosítása | Meglévő adat módosítása | Nem adatai frissülnek az adatbázisban | Sikeres |
| Nem törlése | Törlés gomb megnyomása | Nem eltűnik az adatbázisból | Sikeres |
| Maximális csuklóméret milliméterben hozzáadása | Kitöltött adatmezők mentése | Maximális csuklóméret milliméterben megjelenik az adatbázisban | Sikeres |
| Maximális csuklóméret milliméterben módosítása | Meglévő adat módosítása | Maximális csuklóméret milliméterben adatai frissülnek az adatbázisban | Sikeres |
| Maximális csuklóméret milliméterben törlése | Törlés gomb megnyomása | Maximális csuklóméret milliméterben eltűnik az adatbázisból | Sikeres |
| Raktár hozzáadása | Kitöltött adatmezők mentése | Raktár megjelenik az adatbázisban | Sikeres |
| Raktár módosítása | Meglévő adat módosítása | Raktár adatai frissülnek az adatbázisban | Sikeres |
| Raktár törlése | Törlés gomb megnyomása | Raktár eltűnik az adatbázisból | Sikeres |
| Óraforma hozzáadása | Kitöltött adatmezők mentése | Óraforma megjelenik az adatbázisban | Sikeres |
| Óraforma módosítása | Meglévő adat módosítása | Óraforma adatai frissülnek az adatbázisban | Sikeres |
| Óraforma törlése | Törlés gomb megnyomása | Óraforma eltűnik az adatbázisból | Sikeres |
| Maghajtás hozzáadása | Kitöltött adatmezők mentése | Maghajtás megjelenik az adatbázisban | Sikeres |
| Maghajtás módosítása | Meglévő adat módosítása | Maghajtás adatai frissülnek az adatbázisban | Sikeres |
| Maghajtás törlése | Törlés gomb megnyomása | Maghajtás eltűnik az adatbázisból | Sikeres |
| Márka hozzáadása | Kitöltött adatmezők mentése | Márka megjelenik az adatbázisban | Sikeres |
| Márka módosítása | Meglévő adat módosítása | Márka adatai frissülnek az adatbázisban | Sikeres |
| Márka törlése | Törlés gomb megnyomása | Márka eltűnik az adatbázisból | Sikeres |
| Üveg fajta hozzáadása | Kitöltött adatmezők mentése | Üveg fajta megjelenik az adatbázisban | Sikeres |
| Üveg fajta módosítása | Meglévő adat módosítása | Üveg fajta adatai frissülnek az adatbázisban | Sikeres |
| Üveg fajta törlése | Törlés gomb megnyomása | Üveg fajta eltűnik az adatbázisból | Sikeres |
| Jótállás hozzáadása | Kitöltött adatmezők mentése | Jótállás megjelenik az adatbázisban | Sikeres |
| Jótállás módosítása | Meglévő adat módosítása | Jótállás adatai frissülnek az adatbázisban | Sikeres |
| Jótállás törlése | Törlés gomb megnyomása | Jótállás eltűnik az adatbázisból | Sikeres |
| Extrafunkció hozzáadása | Kitöltött adatmezők mentése | Extrafunkció megjelenik az adatbázisban | Sikeres |
| Extrafunkció módosítása | Meglévő adat módosítása | Extrafunkció adatai frissülnek az adatbázisban | Sikeres |
| Extrafunkció törlése | Törlés gomb megnyomása | Extrafunkció eltűnik az adatbázisból | Sikeres |
| Dátumkijelzés hozzáadása | Kitöltött adatmezők mentése | Dátumkijelzés megjelenik az adatbázisban | Sikeres |
| Dátumkijelzés módosítása | Meglévő adat módosítása | Dátumkijelzés adatai frissülnek az adatbázisban | Sikeres |
| Dátumkijelzés törlése | Törlés gomb megnyomása | Dátumkijelzés eltűnik az adatbázisból | Sikeres |
| A tok színe hozzáadása | Kitöltött adatmezők mentése | A tok színe megjelenik az adatbázisban | Sikeres |
| A tok színe módosítása | Meglévő adat módosítása | A tok színe adatai frissülnek az adatbázisban | Sikeres |
| A tok színe törlése | Törlés gomb megnyomása | A tok színe eltűnik az adatbázisból | Sikeres |
| Számlap színe hozzáadása | Kitöltött adatmezők mentése | Számlap színe megjelenik az adatbázisban | Sikeres |
| Számlap színe módosítása | Meglévő adat módosítása | Számlap színe adatai frissülnek az adatbázisban | Sikeres |
| Számlap színe törlése | Törlés gomb megnyomása | Számlap színe eltűnik az adatbázisból | Sikeres |
| Megrendelés hozzáadása | Kitöltött adatmezők mentése | Megrendelés megjelenik az adatbázisban | Sikeres |
| Megrendelés módosítása | Meglévő adat módosítása | Megrendelés adatai frissülnek az adatbázisban | Sikeres |
| Megrendelés törlése | Törlés gomb megnyomása | Megrendelés eltűnik az adatbázisból | Sikeres |
| A számlap színe hozzáadása | Kitöltött adatmezők mentése | A számlap színe megjelenik az adatbázisban | Sikeres |
| A számlap színe módosítása | Meglévő adat módosítása | A számlap színe adatai frissülnek az adatbázisban | Sikeres |
| A számlap színe törlése | Törlés gomb megnyomása | A számlap színe eltűnik az adatbázisból | Sikeres |
| A számlap típusa hozzáadása | Kitöltött adatmezők mentése | A számlap típusa megjelenik az adatbázisban | Sikeres |
| A számlap típusa módosítása | Meglévő adat módosítása | A számlap típusa adatai frissülnek az adatbázisban | Sikeres |
| A számlap típusa törlése | Törlés gomb megnyomása | A számlap típusa eltűnik az adatbázisból | Sikeres |
| Számla hozzáadása | Kitöltött adatmezők mentése | Számla megjelenik az adatbázisban | Sikeres |
| Számla módosítása | Meglévő adat módosítása | Számla adatai frissülnek az adatbázisban | Sikeres |
| Számla törlése | Törlés gomb megnyomása | Számla eltűnik az adatbázisból | Sikeres |

**Frontend és backend teszteredménye a ThunderClient segítségével:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tesztelt végpont | Bemenet / művelet | Várt eredmény | Teszt eredménye |
| GET /products | Nincs bemenet | Visszaadja az összes terméket | Sikeres |
| GET /products/:id | Meglévő azonosító (pl. /products/2) | Visszaadja a kiválasztott terméket | Sikeres |
| POST /filter | {"marka": "Fossil", "minAr": 10000, "maxAr": 50000} | A szűrési feltételeknek megfelelő termékek listája | Sikeres |
| POST /search | {"query": "férfi óra"} | Visszaadja a keresésnek megfelelő találatokat | Sikeres |
| POST /register | {"nev": "Kiss Péter", "tel": "36123456789", "email": "kiss.peter@example.com", "jelszo": "titkos123"} | Felhasználó regisztrálva az adatbázisba | Sikeres |
| POST /check-user | {"email": "kiss.peter@example.com", "tel": "36123456789"} | Megállapítja, hogy a felhasználó létezik-e | Sikeres |
| GET /user-by-email | {"email": "kiss.peter@example.com"} | Visszaadja a felhasználó adatait email alapján | Sikeres |
| GET /profile/:id | /profile/3 | Visszaadja a megadott azonosítójú felhasználó adatait | Sikeres |
| PUT /profile/:id | {"nev": "Kiss Péter", "tel": "36201112222"} | Felhasználói adatok frissítve | Sikeres |
| PUT /profile/password/:id | {"oldPassword": "titkos123", "newPassword": "ujtitok456"} | Jelszó sikeresen megváltoztatva | Sikeres |
| POST /guest | {"name": "Vendég József", "email": "vendeg@pelda.hu", "phone": "36201110000"} | Vendég vásárló elmentve | Sikeres |
| POST /shipping | {"name": "Vendég József", "address": "Teszt utca 1.", "postalCode": "1234", "city": "Budapest"} | Szállítási adatok mentve | Sikeres |
| POST /invoice | {"vasarloaz": 3, "szallitasaz": 5, "fizetesmodaz": 1, "adoszam": "12345678-1-42"} | Új számla készült | Sikeres |
| POST /order-item | {"szamlaaz": 1, "oraaz": 2, "db": 1} | Megrendelési tétel rögzítve | Sikeres |
| GET /brands | Nincs bemenet | Visszaadja az elérhető márkákat | Sikeres |
| GET /genders | Nincs bemenet | Visszaadja az elérhető nemeket | Sikeres |
| GET /meghajtasok | Nincs bemenet | Visszaadja a meghajtás típusokat | Sikeres |
| GET /vizallosagok | Nincs bemenet | Visszaadja a vízállósági szinteket | Sikeres |
| GET /sulyok | Nincs bemenet | Visszaadja az elérhető súlyokat | Sikeres |
| GET /tipusok | Nincs bemenet | Visszaadja az óra típusokat | Sikeres |
| GET /datumkijelzesek | Nincs bemenet | Visszaadja a dátumkijelzési opciókat | Sikeres |
| GET /extrafunkciok | Nincs bemenet | Visszaadja az extra funkciókat | Sikeres |
| GET /atokszinek | Nincs bemenet | Visszaadja a tok színeket | Sikeres |
| GET /aszamlapszinek | Nincs bemenet | Visszaadja a számlap színeket | Sikeres |
| GET /atok | Nincs bemenet | Visszaadja a tok típusokat | Sikeres |
| GET /kristalyuvegek | Nincs bemenet | Visszaadja a kristályüveg típusokat | Sikeres |
| GET /szamlaptipusok | Nincs bemenet | Visszaadja a számlaptípusokat | Sikeres |
| GET /oraformak | Nincs bemenet | Visszaadja az óraformákat | Sikeres |
| GET /szijszinek | Nincs bemenet | Visszaadja a szíjak színeit | Sikeres |
| GET /szijk | Nincs bemenet | Visszaadja a szíjak típusait | Sikeres |
| GET /maxcsuklomili | Nincs bemenet | Visszaadja a maximális csuklóméreteket | Sikeres |
| GET /price-range | Nincs bemenet | Visszaadja a minimális és maximális árakat | Sikeres |

## ****7. Csapatmunka bemutatása:****

A projektet háromfős csapatban készítettük el, ahol mindenki egy-egy fő területért felelt. A munkát előre egyeztettük és felosztottuk, így mindenki a saját részére tudott koncentrálni.

### ****Munkamegosztás, feladatok, szerepek:****

* **Berényi Gergő** – felelt a backend elkészítéséért. Az ő feladata volt az adatbázis kapcsolatok kezelése, REST API végpontok, és az adatlogika megvalósítása.
* **Makra Martin** – készítette a frontend oldalt. Ő dolgozott a React alapú felhasználói felületen, a rendelés, regisztráció, keresés, szűrés és megjelenítés funkcióin.
* **Szilágyi István** – készítette az adatbázis struktúráját, a nézeteket, valamint az adminisztrációs WPF alkalmazást.

### ****Projektszervezési eszközök, Git:****

A fejlesztést Git segítségével szerveztük. A GitHub-on külön ágakat (branch-eket) használtunk az egyes feladatokra, így mindenki párhuzamosan dolgozhatott a saját részén. A projekt során többször történt összevonás (merge), amit előzetes egyeztetéssel végeztünk.

A projekt során megbeszéléseken egyeztettünk a feladatokról, valamint üzenetváltásban is folyamatosan kapcsolatban voltunk.

### ****Teljesítményarány:****

* **Berényi Gergő (Backend):** 30%
* **Makra Martin (Frontend):** 35%
* **Szilágyi István (Adatbázis + Admin felület):** 35%