Nyomtatókölcsönző - Printer-Bookings

Székesfehérvári SZC Széchenyi István Műszaki Technikum

Záródolgozat

2024

Thék Olivér, Török Zoltán

# **Bevezetés**

Ez a webalkalmazás egy képzeletbeli nyomtatókölcsönző cég adminisztrációs felülete. A programot kifejezetten nyomtatók, fénymásolók bérbeadásával foglalkozó cégek tudják használni. Ha az ügyfél, aki lehet egy személy vagy akár egy nagyvállalat, gazdasági és kényelmi szempontok miatt a nyomtató, fénymásoló, szkenner hosszútávú bérlése mellett dönt, és a mi cégünket választja, az alkalmazással az adminisztrátor a printerek és ügyfelek közötti kapcsolatot tudja kezelni, létrehozhat szerződéseket és munkalapokat. A két fős fejlesztőcsapat egyik tagja egy ilyen vállalkozás részvevője és az általuk használt nyilvántartó program csak asztali alkalmazás formájában létezett. A távolról, szabadon választott eszközről történő munkavégzés és a program nyilvános kiterjeszthetősége miatt felmerült az igény egy webes alkalmazás létrehozására.

**Fejlesztői dokumentáció**

A webalkalmazás elkészítésének ötlete egy valós munkakörnyezetből ered. Egyelőre ez egy nyilvántartó, szerződések és munkalapok létrehozására alkalmas program, amely a későbbiekben könnyen fejleszthető és kiegészíthető egy olyan alkalmazásra, amely bárkinek segítségére lehet, aki nyomtatót szeretne bérelni.

Az alkalmazás használatához csak egy internetre kapcsolódó és webböngészővel rendelkező eszközre van szükség. Ezzel az adminisztrátornak lehetőséget ad mobil eszközökön vagy asztali gépen szabadon választott helyekről történő munkavégzésre.

A fejlesztői környezet

A fejlesztői környezet kiválasztásánál a legmodernebb kompakt technikák használata volt a célunk, ezért választottuk a Next.js-t amelyben a backend és a frontend is megvalósítható.

A rendszerfejlesztés során az alábbi technológiákat és szolgáltatásokat használtuk fel, hogy az hatékonyan és megbízhatóan működjön.

A **Next.js** egy keretrendszer React.js alapú webalkalmazásokhoz, amely nagyon egyszerűvé teszi a fejlesztési folyamatot. A Next.js segítségével könnyen tudunk dinamikus és reszponzív webalkalmazásokat készíteni és optimalizálni a teljesítményt. A fejlesztést segíti a fájlrendszerhez hasonló navigáció, amely átláthatóbbá teszi a programkódot. A legfontosabb dolog azonban az, hogy alapvetően szerverkomponenseket hozunk létre, és csak a kód első sorában megjelölt komponensek lesznek kliens oldalon renderelt komponensek. Ez a felépítés biztonságos, gyors, mert a cache és minden érzékeny adat a szerveren tárolódik. A Printer-Bookings forráskódjában a kliens oldali komponensek az app/ui könyvtárban vannak. Az összes többi komponens szerverkomponens.

A **backend** az app/lib és az app/api könyvtárakban van, a többi fájl a frontend. A **frontend** pedig két részre osztható, az egyik az app/ui útvonalon található kliens oldalon renderelt, és a többi könyvtárban található szerver oldalon futó programfájlokra.

A **Prisma** egy ORM, azaz objektum relációs modell, amellyel a fejlesztők könnyen és hatékonyan tudnak relációs adatbázisokkal dolgozni az általuk választott objektumorientált környezetben. A Prisma használatával egyszerűen tudunk adatbázisokat modellezni és lekérdezéseket futtatni a rendszerben az SQL nyelv használata nélkül.

A **Clerk** egy ingyenesen is használható átfogó felhasználókezelő platform, amellyel a felhasználó azonosítását és az alkalmazáshoz való hozzáférési jogait felügyeljük. A főoldalon kívül minden oldal védett, tehát a mi esetünkben csak az adminisztrációs joggal felruházott felhasználók láthatják, módosíthatják vagy hozhatják létre az adatokat.

A regisztráció (sign up) után a Printer-Bookings alkalmazás az app/api könyvtárban található webhook segítségével a Clerk által regisztrált felhasználó adatait az adatbázisban is létrehozza. Ezután az új szerződések, munkalapok létrehozását, módosítását az aktív felhasználóhoz köti.

A Vercel egy felhőszolgáltatás, amely lehetővé teszi a webalkalmazások gyors és egyszerű fejlesztését és közzétételét. A Vercel használatával könnyen tudjuk publikálni a webalkalmásainkat az interneten és közben automatizáljuk a fejlesztési és közzétételi folyamatokat.

Hardverigény

A fejlesztői hardverigénye egy Node.js futtatására alkalmas konfiguráció, 1.6 GHz-es vagy gyorsabb processzor, és 1GB RAM, a minimum.

Szoftverigény

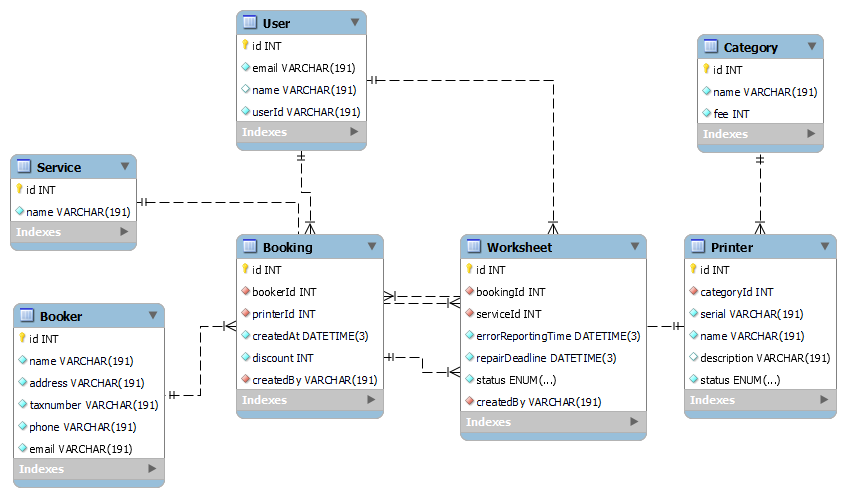
Linux (Debian): Ubuntu Desktop 20.04, Debian 10, Linux (Red Hat): Red Hat Enterprise Linux 8, Fedora 36, MacOS (utolsó három verzió) vagy Windows 10, 11 ajánlott.

A fejlesztés Windows-on Visual Studio Code-dal és a Github verziókövető használatával történt, ezért párhuzamosan is tudtunk dolgozni a projekten.

Az adatbázis szerkezete

A terv hat weboldalt és egy kezdőlapot tartalmazott: ügyfelek (Booker), printerek, printer kategóriák (Category), szerződések (Booking), munkalapok (Worksheet), bejelentett jellemző hibák (Service) és statisztikák, amely egyértelműen meghatározta az adatbázis szerkezetét is.

Az adatbázis 7 táblából áll.



A **User** tábla tartalmazza a hozzáféréssel rendelkező felhasználók adatait, és 4 mezővel rendelkezik.

* Id: a felhasználó azonosítója, amely egész szám, elsődleges kulcs és automatikusan egyesével növekszik.
* Email: a felhasználó emailcíme, amellyel regisztrált, típusa varchar.
* Name: a felhasználó neve, amit nem kötelező megadni, típusa varchar.
* UserId: a felhasználó Clerk által is tárolt azonosítója, típusa varchar.

A **Booker** tábla tartalmazza az ügyfelek adatait, 6 mezővel rendelkezik.

* Id: az ügyfél azonosítója, típusa egész szám, elsődleges kulcs és automatikusan egyesével növekszik.
* Name: az ügyfél neve, típusa varchar.
* Address: az ügyfél címe, típusa varchar.
* Taxnumber: az ügyfél adószáma, típusa varchar.
* Phone: az ügyfél telefonszáma, típusa varchar.
* Email: az ügyfél emailcíme, típusa varchar.

A **Category** tábla tartalmazza a printer kategóriák adatait, 3 mezővel rendelkezik.

* Id: a kategória azonosítója, típusa egész szám, elsődleges kulcs és automatikusan egyesével növekszik.
* Name: a kategória neve, típusa varchar.
* Fee: a kategóriához tartozó printerek bérleti díja, típusa egész szám.

A **Printer** tábla tartalmazza a printerek adatait, 6 mezővel rendelkezik.

* Id: a printer azonosítója, típusa egész szám, elsődleges kulcs és automatikusan egyesével növekszik.
* CategoryId: a printer kategóriájának azonosítója, idegen kulcs (Category: id).
* Serial: a printer cikkszáma, típusa varchar.
* Name: a printer neve, típusa varchar.
* Description: a printer rövid leírása, típusa varchar.
* Status: a printer állapota “szabad” vagy “foglalt”, típusa enum.

A **Booking** tábla tartalmazza a szerződések részleteit, azaz az ügyfél- és az általa bérelt printerek adatait, 6 mezővel rendelkezik.

* Id: a szerződés azonosítója, típusa egész szám, elsődleges kulcs és automatikusan egyesével növekszik.
* BookerId: az ügyfél azonosítója, idegen kulcs (Booker: id).
* PrinterId: a bérelt printer azonosítója, idegen kulcs (Printer: id).
* CreatedAt: a bérleti szerződés létrejöttének időpontja, típusa DateTime.
* Discount: az adható kedvezmény százalékban, típusa egész szám.
* CreatedBy: a felhasználó azonosítója, aki a szerződést létrehozta, idegen kulcs (User: UserId).

A **Service** tábla tartalmazza az ügyfelek által bejelentett jellemző és ismétlődő printerhibákat, amelyek a hiba regisztrálásakor a munkalapokra kerülnek. 2 mezővel rendelkezik.

* Id: a hiba azonosítója, elsődleges kulcs, típusa egész és értéke automatikusan növekszik.
* Name: a hiba rövid leírása, típusa varchar.

A **Worksheet** tábla tartalmazza a munkalapokat, amelyekben a szerződés részletei és a hiba leírása szerepel, 7 mezővel rendelkezik.

* Id: a munkalap azonosítója, típusa egész szám, elsődleges kulcs és értéke automatikusan növekszik.
* BookingId: a szerződés azonosítója, idegen kulcs (Booking: id).
* ServiceId: a hiba azonosítója, idegen kulcs (Service: id).
* ErrorReportingTime: a hiba bejelentésének időpontja, típusa DateTime.
* RepairDeadline: a hiba kijavításának határideje, típusa DateTime.
* Status: a hibajavítás állapota, értéke lehet “folyamatban” vagy “befejezett”, típusa enum.
* CreatedBy: a felhasználó azonosítója, aki a munkalapot létrehozta, idegen kulcs (User: UserId).

**A Printer-Bookings teljes használatához jogosultság szükséges, a belépéshez szükséges felhasználói adatokat emailben küldtük el.**

**Az alkalmazás a** [**https://printer-bookings.vercel.app**](https://printer-bookings.vercel.app) **címen érhető el.**

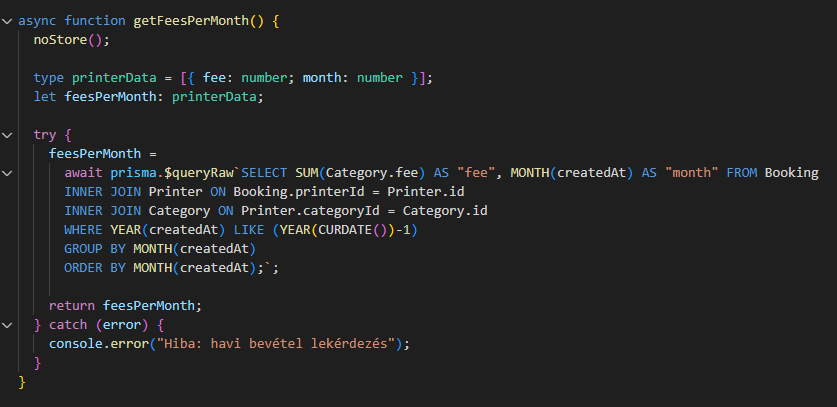
A beléptetés a Clerk-kel történik. A regisztrációt kikapcsoltuk, és csak az általunk megadott email címekkel lehet belépni a felületre.

Érdekesebb algoritmusok bemutatása:

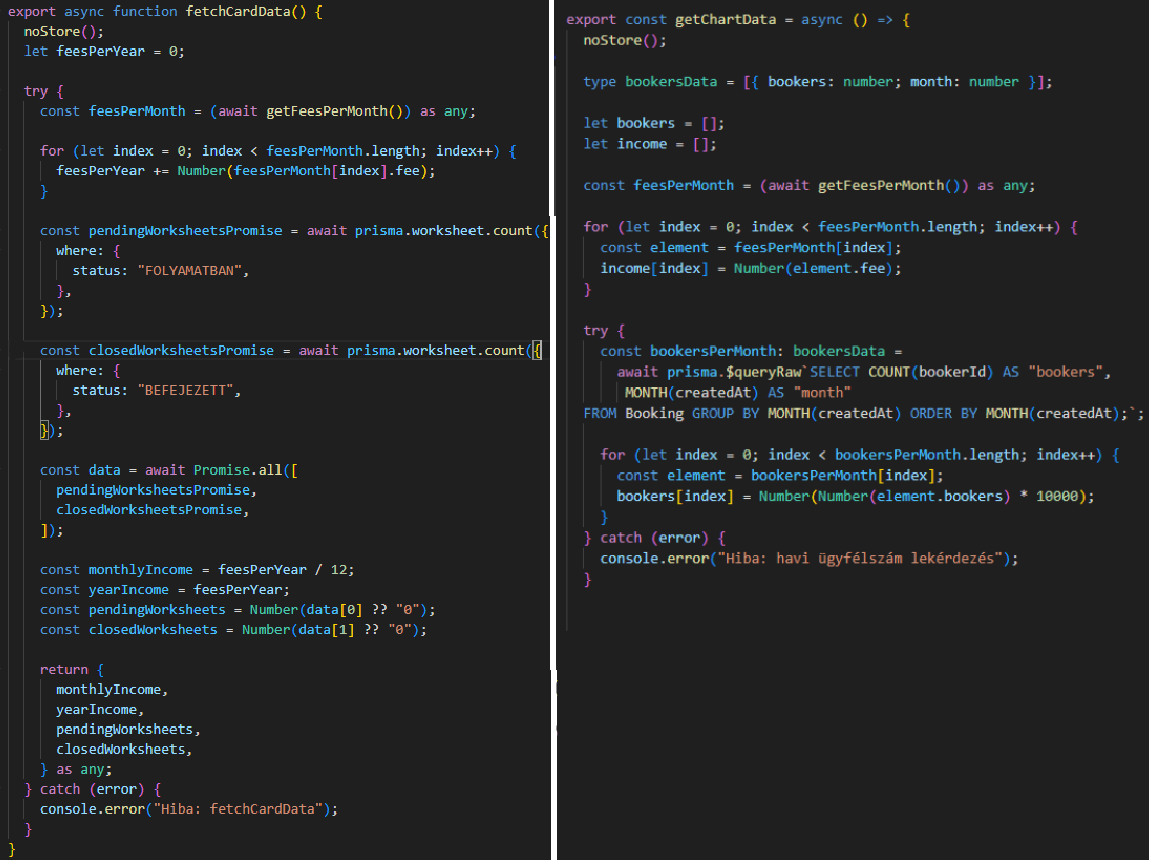
1. Az updateWorksheetStatus függvény felelős az alkalmazás munkalapjai állapotának automatikus frissítéséért azok lejárt határidői alapján. Ez a függvény biztosítja a hatékony és automatizált munkafolyamat-kezelést azáltal, hogy dinamikusan változtatja meg a munkalapok állapotát a lejárt határidők alapján.



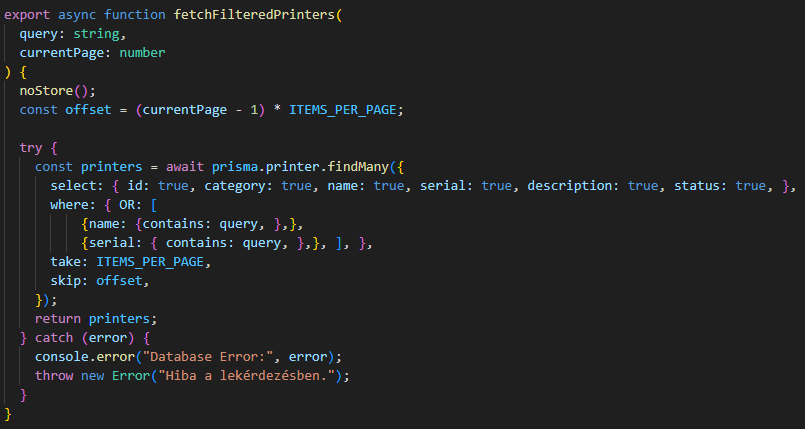
2. getFeesPerMonth: Ez a függvény havi bevételi adatokat számol az előző év során. Az érdekessége abban rejlik, hogy havi alapon összegzi és megjeleníti a bevételt, ami fontos lehet az üzleti elemzésekhez.



3. fetchCardData és getChartData függvények az alkalmazás főoldalán megjelenítendő adatokat gyűjtik és formázzák. Ezek az algoritmusok biztosítják a dinamikusan generált statisztikákat és adatokat a felhasználók számára.



4. Az alkalmazás különböző részein, a Szerződések, Ügyfelek, Printerek és Munkalapok oldalakon az oldal tetején egy keresősáv található, amely a keresett adatok gyors és könnyű hozzáférését teszi lehetővé. Mindegyiknek megvan a saját fetchFiltered függvénye, amely felelős a szűrt adatok lekérdezéséért és visszaadásáért. Ezek a függvények különböző kritériumok alapján szűrik az adatokat, például név, cím vagy egyéb tulajdonság alapján. Kategóriák és Hibák oldalak esetében nincs szükség keresési funkcióra, mivel ezek általában kevesebb adatot tartalmaznak.

Az egyik ilyen függvény:

Minden oldalhoz tartozik egy táblázat is, amely ezeket az adatokat jeleníti meg. A táblázatok segítségével könnyen áttekinthető módon listázhatók a lekért adatok, és lehetőség van a felhasználók számára azok módosítására vagy törlésére is.

Így az alkalmazás különböző részein keresztül egy egységes megközelítésben történik az adatok lekérdezése és megjelenítése, ami segíti az alkalmazás átlátható működését.

Az egyik ilyen táblázat kódja:

**Tesztelői dokumentáció**

Mehet gomb tesztelése

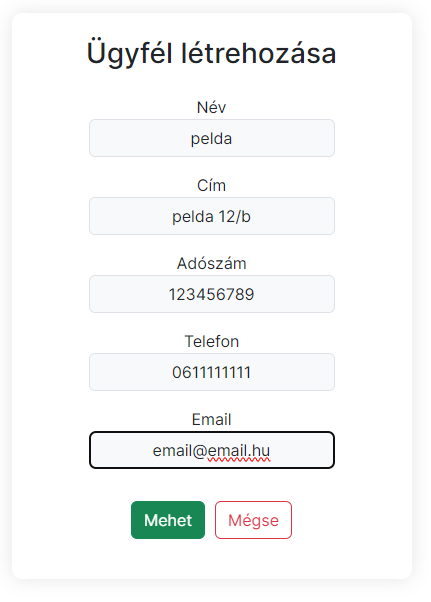
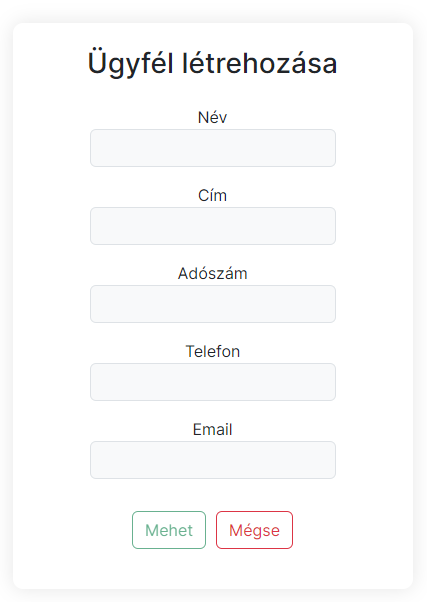
Leírás: Az alkalmazásban ellenőrizzük, hogy a "Mehet" gomb inaktív marad-e, amíg nem töltik fel az összes szükséges input mezőt.

Tesztesetek

Ellenőrizzük, hogy a gomb inaktív marad-e, amikor egyetlen input mező sem lett kitöltve.

Töltsünk ki minden input mezőt kivéve egyet, majd ellenőrizzük, hogy a gomb továbbra is inaktív marad-e.

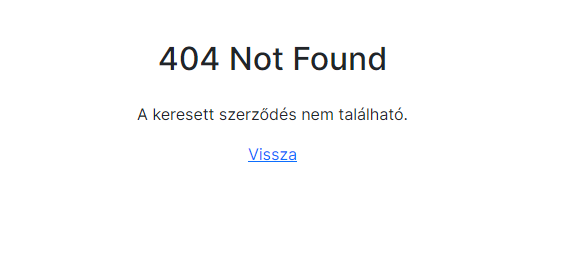
Kitöltjük az összes szükséges input mezőt, és ellenőrizzük, hogy a gomb aktiválódik-e.



"Not Found” tesztelése

Leírás: Tesztesetek készítése olyan esetre, amikor egy rekord szerkesztését próbáljuk megvalósítani, de az adott rekord nem található az adatbázisban.

Tesztesetek

Megpróbálunk egy rekordot szerkeszteni az ID alapján, amely nem található az adatbázisban, majd ellenőrizzük, hogy a program megjeleníti-e a megfelelő hibaüzenetet.

**Fejlesztési lehetőségek**

További fejlesztési lehetőségek rejlenek a rendszerben, amelyekkel még hatékonyabbá és sokoldalúbbá lehet tenni azt. A következő funkciók és fejlesztésekre nem lett volna elég idő:

jelenleg hiányzik a számlázási funkció a rendszerből, ami a hatékony ügyfélkezelést és a pénzügyi adminisztrációt is nehezíti. Ennek megoldása érdekében lehetne bevezetni a számlázási rendszert, amely lehetővé teszi a számlák automatikus generálását és kezelését, valamint a pénzügyi adatok integrálását az ügyfélkapcsolati rendszerbe.

Ezen kívül gondoltunk a nyomtatók számlálójának állásának felvitelére, ami alapján lehetne havi béreket számlázni, és egy raktár készlet modult, hogy lehetővé tegye a raktárban található termékek nyilvántartását és hatékony kezelését.

A számlázási rendszer és a nyomtatók anyagkezelésének bevezetése mellett más fejlesztési lehetőségeket is fontolóra veszünk, például a felhasználói élmény javítását, a teljesítmény optimalizálását és a rendszer bővíthetőségét

Felhasználói élmény javítása: Az alkalmazás felhasználói felületének és felhasználói élményének további fejlesztése és optimalizálása, beleértve a felhasználóbarátabb navigációt és az intuitívabb működést.

Teljesítmény optimalizálás: Az alkalmazás teljesítményének optimalizálása és a válaszidők csökkentése érdekében végzett javítások és finomhangolások.

**Felhasználói dokumentáció**

A webalkalmazás ügyfelek, nyomtatók, szerződések, munkalapok, hibák nyilvántartására, felvételére, munkalapok nyomtatására alkalmas. Ügyfelek és nyomtatók egymáshoz rendelésével szerződéseket lehet létrehozni. Ha egy ügyfél nyomtatással vagy a nyomtatóval kapcsolatos hibát jelent a cég felé, munkalapok felvételére is van lehetőség, amelyet kinyomtatva a hibaelhárítás megkezdődhet.

A főoldalon a cég működésével kapcsolatos statisztikák láthatók.

Hardver és szoftver igény

Számítógép vagy mobil eszköz: A programhoz hozzáféréshez szükség van egy számítógépre (laptop vagy asztali gép) vagy mobil eszközre (okostelefon vagy táblagép). A számítógépnek vagy mobil eszköznek képesnek kell lennie webböngésző futtatására.

Operációs rendszer: A program általában böngésző alapú, így az operációs rendszer szempontjából nincs szükség specifikus platformra. A legtöbb modern operációs rendszerrel kompatibilis, beleértve a Windows-t, macOS-t, Linuxot és különböző mobil operációs rendszereket (pl. Android, iOS).

Internetkapcsolat: A program online felületen érhető el, ezért stabil internetkapcsolat szükséges a használathoz. Az eszköz, amelyen futtatja a programot, képes legyen megbízható internetkapcsolat létesítésére.

Internetböngésző: A felhasználóknak internetböngészőre van szükségük a webalkalmazás használatához. Az alkalmazás legtöbb funkcionalitása együttműködik a legelterjedtebb böngészőkkel, mint például a Google Chrome, Mozilla Firefox, Apple Safari vagy a Microsoft Edge.

Operációs rendszer: A felhasználók számítógépén vagy eszközén szükség van egy operációs rendszerre, amely kompatibilis a választott internetböngészővel. Ez lehet Windows, macOS, Linux vagy más operációs rendszer.

Internetkapcsolat: A webalkalmazás használatához internetkapcsolatra van szükség, hogy a felhasználók tudjanak kommunikálni a szerverrel és betölteni az alkalmazás tartalmát.

Fontos megjegyezni, hogy a webalkalmazás futtatása közben a felhasználók nem igényelnek további szoftvereket, csak a fent említett alapvető eszközöket és szolgáltatásokat. Ez lehetővé teszi az alkalmazás könnyű elérhetőségét és használatát széles körben, anélkül hogy bonyolult telepítési folyamatokra lenne szükség.

Összességében a programhoz csak minimális hardveres és szoftveres követelmények szükségesek, amelyeket a legtöbb számítógép vagy mobil eszköz könnyedén teljesíthet. Fontos azonban biztosítani egy stabil internetkapcsolatot a hatékony használat érdekében.

Telepítés és indítás lépéseinek ismertetése

Egy böngésző megnyitása után, lépjen fel a [**printer-bookings.vercel.app**](https://printer-bookings.vercel.app/) -ra.

Az oldal betöltése után bejelentkezés nélkül is látható a főoldal, ahol információk találhatók a cég adatairól.

Ha szeretnénk hozzáférni további funkciókhoz, be kell jelentkezni a weboldalon a programhoz tartozó hitelesítési adatokkal.

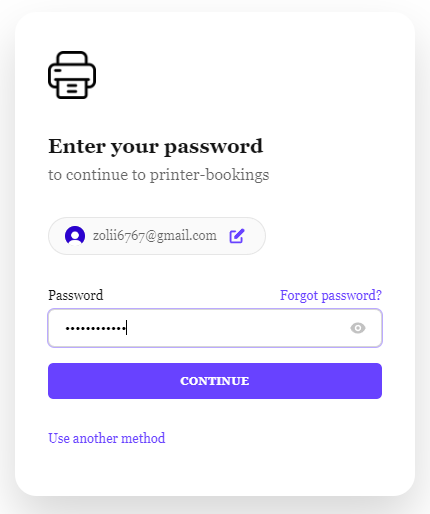
Sikeres bejelentkezés után a program teljes körű funkcionalitása elérhető lesz.

# **A program használata**

A főoldalon a cég működésével kapcsolatos statisztikák láthatók.

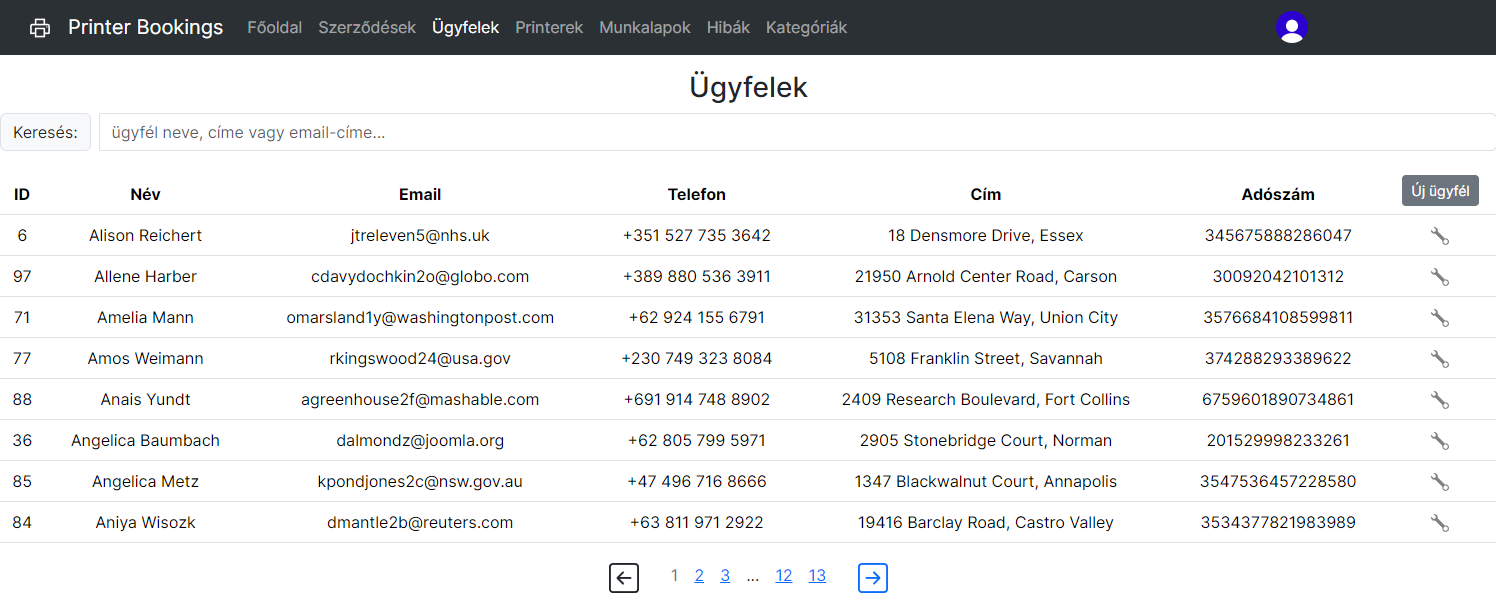
Az éves bevétel, havi átlagbevétel, alattuk egy grafikonon a csatlakozott ügyfelek és a bevételek havi bontásban jelennek meg. Jobb felül a lezárt és a folyamatban lévő hibaelhárítások, alattuk a nyomtatók állapotával kapcsolatos statisztikát tartalmazó kördiagram található.

Az alkalmazás többi funkciója csak bejelentkezés után érhető el.

A bejelentkezés és regisztráció funkcióit a Clerk-en keresztül biztosítjuk. A regisztrációs folyamat során fontos tudni, hogy csak előre engedélyezett e-mail címekkel lehet regisztrálni. Így biztosítjuk a rendszerünk biztonságát és ellenőrzöttségét, valamint megakadályozzuk a jogosulatlan hozzáféréseket és regisztrációkat.

A programunk egy intuitív és könnyen kezelhető platform, amely lehetővé teszi a felhasználók számára a következő tevékenységek végrehajtását.

Ügyfelek kezelése

Új ügyfél hozzáadása: A "Új ügyfél" gombra kattintva a felhasználók egyszerűen hozzáadhatnak új ügyfeleket az adatbázishoz. A megfelelő mezők kitöltése után a "Mehet" gombra kattintva az adatok mentésre kerülnek.

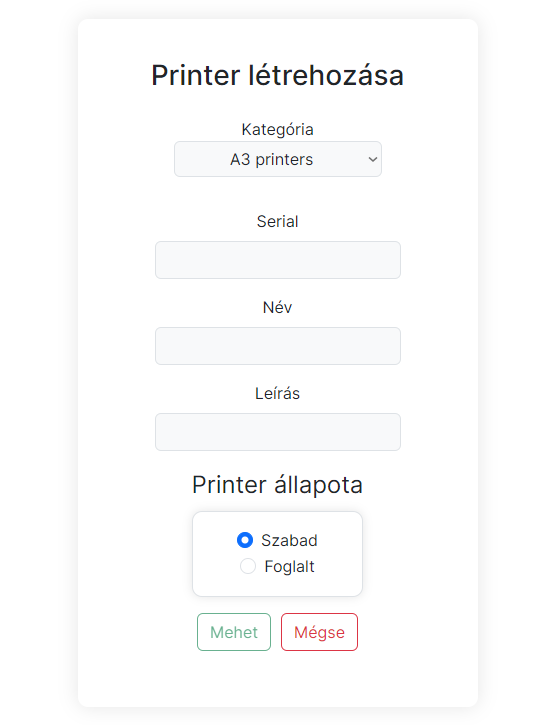
Ügyfelek szerkesztése: A listázott ügyfelek mellett lehetőség van azok szerkesztésére. Könnyen szerkeszthetők az ügyfél adatai a csavarkulcsra kattintva.

Nem került implementálásra a törlés funkció az ügyfelek esetében. A célunk az, hogy az ügyfelek adatai biztonságban maradjanak az adatbázisban, és ne kerüljenek véletlenül vagy szándékosan törlésre.

Hibák és Kategóriák kezelése

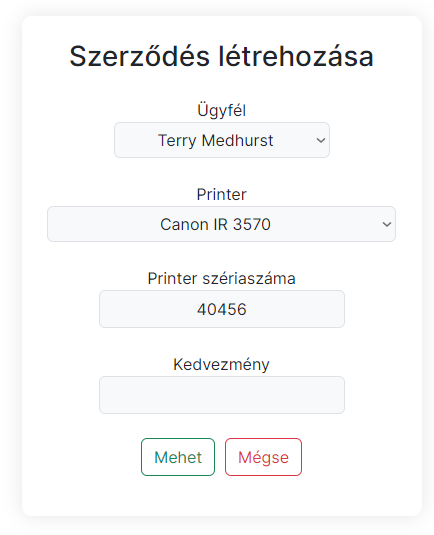
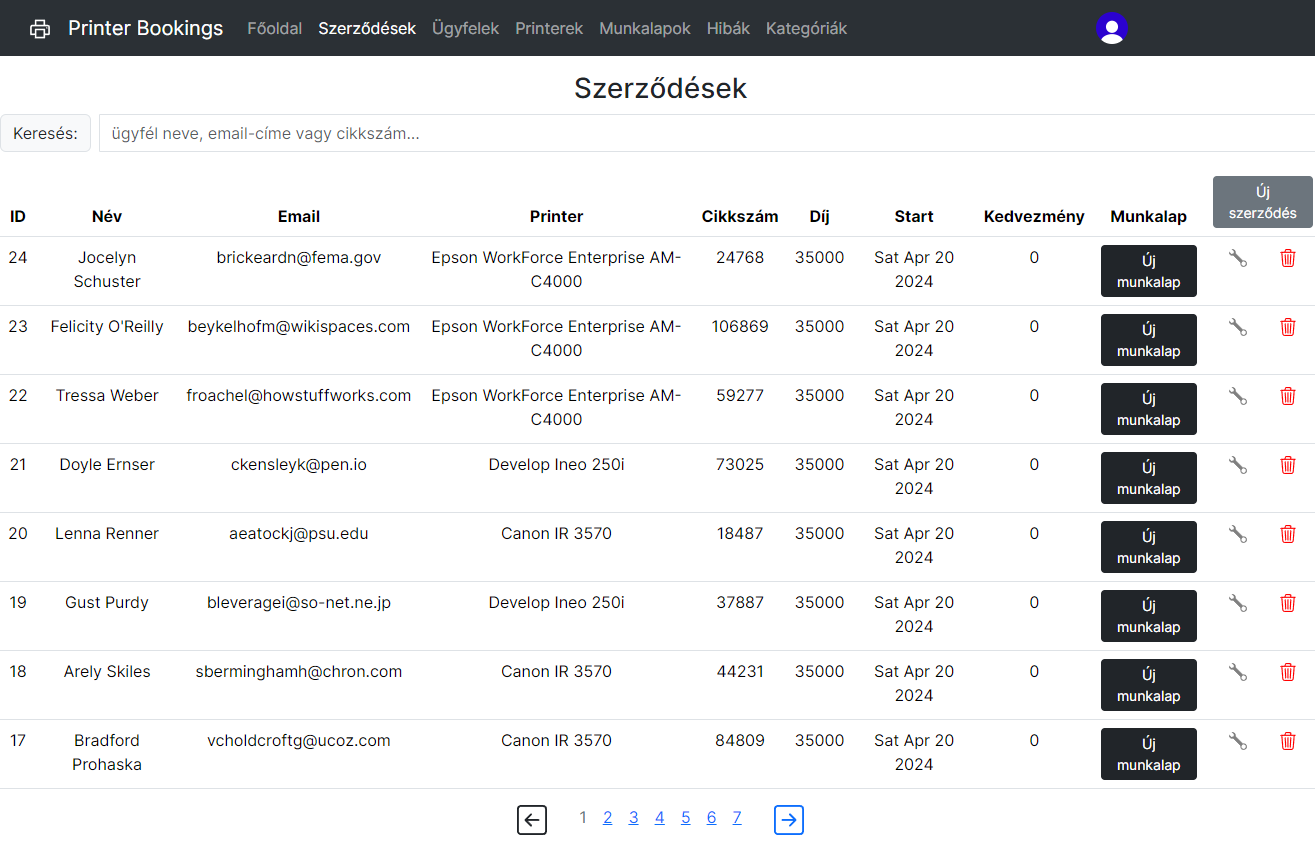
A munkalapon kiválasztható bejelentéseket a ”Hibák” menüpont alatt, a választható eszköz kategóriáját a ”Kategóriák” menü alatt lehet előre felvenni. A kategória létrehozásakor a kategóriának megfelelő havidíj összegét is be kell írni. Ezzel a díjjal kalkulált összeget fizeti majd az ügyfél havonta a nyomtató bérléséért. Ha egy vagy több munkalapon szereplő hibát szeretnénk törölni, akkor az adott munkalapon a hiba mezőben a ”TÖRÖLT HIBA” felirat jelenik meg, de a munkalap nem törlődik.

Printerek kezelése

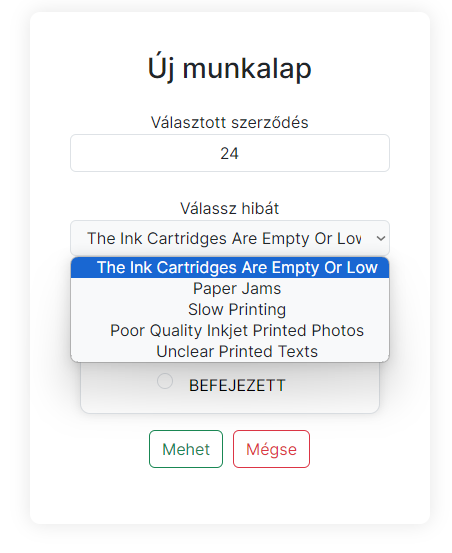
Új printer hozzáadása: az "Új printer" gombra kattintva a felhasználók új nyomtatókat adhatnak hozzá az adatbázishoz. A szükséges információk (kategória, cikkszám, típus, leírás) megadása után a printert azonnal hozzá lehet adni a rendszerhez. A printer állapota alapértelmezett módon ”szabad” lesz, amely akkor változik ”foglalttá”, amikor a nyomtató már egy szerződésen is szerepel. A listázott printerek mellett lehetőség van azok szerkesztésére vagy törlésére is. A törlés esetén, ha a törölni kívánt printer egy vagy több szerződésen szerepel, akkor a printerek listájából a nyomtató törlődik, viszont a szerződésen a printer adatai helyett a ”TÖRÖLT PRINTER” felirat lesz látható és a szerződés nem törlődik. A szerződés szerkesztése gombra kattintva új printert rendelhetünk az ügyfélhez.

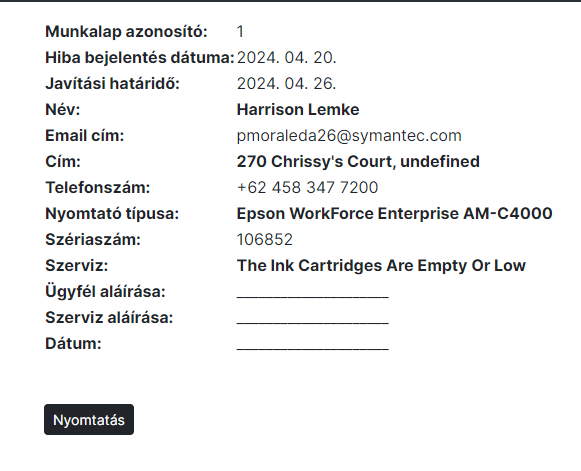
Szerződések kezelése

Új szerződés létrehozása: az "Új szerződés" gombra kattintva a felhasználók szerződéseket hozhatnak létre. Egy legördülő menü segítségével a már felvett ügyfelek közül kell választani és egy még “szabad” printert lehet hozzárendelni. A “Kedvezmény” mezőbe az adott ügyfélnek megítélt kedvezmény százalékban kifejezett értékét lehet beírni. Ez a printer kategóriájának megfelelő havidíjból adott kedvezményt jelenti. Szerződéseket törölni csak nagyon indokolt esetben kell, mert szerződés törlésekor a szerződéshez tartozó összes munkalapot is töröljük.

.

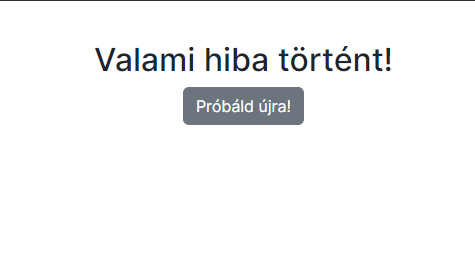
Munkalapok kezelése

Az új munkalap funkcióval a felhasználók könnyen kiválaszthatják a már korábban rögzített hibajelzéseket az adatbázisból. Ezáltal a felhasználóknak lehetőségük van az előzőleg rögzített hibákat újra felhasználni, és hozzájuk új munkalapot létrehozni. Az új munkalap létrehozásának folyamata során a felhasználók a szerződést és hozzá kapcsolt gépet választhatják ki, majd a rendelkezésre álló hibajelzések közül kiválaszthatják azt, amelyik a leginkább megfelel az adott esetben. A munkalapok menü alatt minden munkalapnál található szürke nyomtató ikonra kattintva egy menthető dokumentumot generál. Amit a ”Nyomtatás” gombbal ki is lehet nyomtatni.



# **Hibaüzenetek**

A rendszerünk különböző hibaüzeneteket generálhat a felhasználók számára, amikor problémák merülnek fel a folyamatok során. Az alábbiakban felsoroljuk ezeket a hibaüzeneteket és azt, hogy mikor és hogyan jelennek meg:

"Valami hiba történt!" (Error komponens): Ez az üzenet akkor jelenik meg, amikor a rendszer olyan hibát észlel, amelyet nem tud kezelni vagy feldolgozni. Ebben az esetben a felhasználóknak lehetőségük van az újrapróbálkozásra a "Próbáld újra!" gomb megnyomásával.

"404 Not Found" (NotFound komponens): Ez az üzenet akkor jelenik meg, amikor a felhasználó egy olyan oldalt keres, ami nem létezik a rendszerben vagy nem érhető el. Ebben az esetben a felhasználók visszaléphetnek az előző oldalra a "Vissza" gomb segítségével.

A Clerk szolgáltatás generálhat különböző hibaüzeneteket, amikor problémák merülnek fel a felhasználók által végrehajtott műveletek során. Az alábbiakban felsoroljuk ezeket a hibaüzeneteket és azt, hogy mikor és hogyan jelenhetnek meg:

Sikertelen bejelentkezés: Ha a felhasználó bejelentkezési kísérlete sikertelen volt, a Clerk szolgáltatás különböző hibaüzeneteket generálhat, például "Hibás felhasználónév vagy jelszó" vagy "Nem sikerült bejelentkezni az adatbázishoz". Ha egy felhasználó már bejelentkezett egy ablakban vagy eszközön, és megpróbál bejelentkezni egy másik ablakon vagy eszközön.

Regisztrációs hiba: Amikor a felhasználó regisztrálni próbál, de valamilyen probléma merül fel, például a megadott e-mail cím még nincs engedélyezve a Clerk rendszerében, vagy nem megfelelő formátumú, a Clerk szolgáltatás hibaüzeneteket generálhat a felhasználó számára.

**Elérhetőségek**

Célszerű lehetőséget biztosítani a felhasználók számára, hogy információkat kérjenek vagy kapcsolatba lépjenek velünk. Ehhez kérjük, használják az alábbi elérhetőségi lehetőségeket:

Kapcsolatfelvétel e-mailben: Az alábbi e-mail címeken várjuk az információkérését:

[oliverthek@gmail.com](mailto:oliverthek@gmail.com) vagy [zolii6767@gmail.com](mailto:zolii6767@gmail.com)

# **Záró gondolatok és köszönetnyilvánítás**

Az alkotott projekt elemzésének és végrehajtásának befejezésével a munka vége felé közeledünk. Az elején kitűzött szakmai célok felülvizsgálatakor örömmel állapíthatjuk meg, hogy sok jelentős mérföldkőn sikerült áthaladnunk. Az alkalmazás fejlesztése közben komoly technikai kihívásokba ütköztünk, de ezeket a nehézségeket sikeresen áthidaltuk, mégpedig a jó csapatmunkánknak és eltökéltségünknek köszönhetően.

Ez a projekt sok mindent megtanított nekünk – mind a szakmai fejlődésünkre vonatkozóan, mind pedig a hatékonyabb csapatmunka terén. Talán az adatkezelés volt az egyik legnehezebb dolog, ami megoldást igényelt az alkalmazás architektúrájára vonatkozóan is – de végül stabil és hatékony rendszert hoztunk létre.

A már elkészült mű további életére vonatkozó terveink igen ambiciózusak. Célunk folyamatosan fejleszteni és bővíteni az alkalmazást, ezzel is folyamatosan javítva a felhasználói élményt és szélesebb körben elterjesztve az alkalmazást.

Végül, de nem utolsósorban szeretnénk megköszönni mindenkinek a támogatást, és a segítséget, akik mellettünk álltak ezen az úton.