

**Debreceni Szakképzési Centrum**  
**Baross Gábor Középiskolája és Kollégiuma**  
**Debrecen, 4030**  
**Budai Ézsaiás u. 8/A.**

**A SZAKKÉPESÍTÉS AZONOSÍTÓ SZÁMA:54 213 05**  
**SZAKKÉPESÍTÉS MEGNEVEZÉSE: SZOFTVERFEJLESZTŐ**

**Záródolgozat címe:**

**PrimeChem**

**Záródolgozat készítője:**

**Tóth Péter**

**Konzulens: Kovács István Zoltán**

**Konzultációs időpontok:**


**Debrecen, 2023.**

# Tartalom

1. Bevezetés .....	2
1.1 Szakdolgozat témája:.....	2
1.2 Miért ezt választottam? .....	2
1.3 Elképzelt tervek: .....	2
2. A munkához használt programok .....	3
3. Felhasználói dokumentáció.....	4
3.1 Rendszerkövetelmények.....	4
3.1.1 Minimális rendszerkövetelmények.....	4
3.1.2 Ajánlott rendszerkövetelmények .....	6
3.2 A program megnyitása.....	8
3.3 A program használata .....	11
3.3.1 Menüpontok (Felhasználói felület, bejelentkezés nélkül).....	11
3.3.2 Regisztráció és bejelentkezés .....	17
3.3.3 Bejelentkezett felhasználó (Változások).....	20
3.3.4 Admin felület és lehetőségei .....	21
4. Fejlesztői dokumentáció .....	26
4.1 Adatbázis táblák .....	26
4.1.1 Felhasználók.....	26
4.1.2 Termékek.....	27
4.1.3 Lomtar.....	28
5. Fontosabb kód részletek.....	29
5.1 Regisztráció adatainak felvitele az adatbázisba és hibák kezelése. ....	29
5.2 Termékek kártyáinak megjelenítése. ....	32
6. Tesztelés .....	35

7. Fejlesztési lehetőségek.....	36
8. Összegzés.....	37
9. Irodalomjegyzék.....	38
10. Ábrajegyzék .....	39

# **1. Bevezetés**

## **1.1 Szakdolgozat témája:**

A 2023-as záróvizsgám beadandójához témának egy PrimeChem nevezetű cégre esett a választásom. A cég autókozmetikai vegyszerekkel, ehhez kellő gépekkel, ezek alkatrészeivel és egyéb autókozmetikai segédeszközökkel foglalkozik.

## **1.2 Miért ezt választottam?**

Azért esett a választásom erre a projektre mert a nagybátyám ennek a cégnek a tulajdonosa és érdekel maga a cég és ami mögötte van. Emellett nincs neki webshopja vagy katalógusa, ami szerintem hatalmas lendületet adhatna a cég fejlődésének.

A célom ezzel a projekttel az, hogy egy jól használható és működő weboldalt hozzak létre jelenlegi tudásom szerint. E cél eléréséhez megpróbálok megtenni minden tőlem telhetőt.

## **1.3 Elképzelt tervek:**

Szakdolgozatom elsődleges célja volt, hogy létrehozzak egy weboldalt, amin a felhasználó látja a választott cég termékeit, majd regisztrálást és bejelentkezést követően lehetősége van arra, hogy a rendeléshez szükséges információkat megnézhesse. Továbbá egy adatbázis létrehozása, amelyben a regisztrált felhasználók adatait, a termékek adatait és egy lomtárat tartalmazza.

## **2. A munkához használt programok**

### Windows 11

A Windows 11 operációs rendszert használtam szakdolgozatom elkészítésénél, mert ez a jelenleg legfrissebb Windows operációs rendszer és kompatibilis a legtöbb általam használt programmal.

### Visual Studio Code

A szakdolgozatom a Visual Studio Code nevű program segítségével készítettem. Ez a program az egyik legkönnyebben kezelhető és átlátható program a HTML, CSS és JavaScript típusú nyelvek szerkesztésére.

### MySQL

A MySQL SQL alapú relációs adatbázis-kezelő szerveret használtam a weboldalon található adatok tárolásához.

### Mozilla Firefox

A program teszteléséhez és böngészéshez a Mozilla Firefox böngészőt használtam. Ennek a böngészőnek a segítségével teszteltem a weboldalam és kerestem információkat az interneten weboldalam fejlesztéséhez.

### Windows intéző

A windows saját fájlkezelő programját használtam szakdolgozatom fájljainak rendezéséhez. Ez a program bárkinek elérhető, aki windows operációs rendszert használ.

### XAMPP

Ez a program egy virtuális szervert hoz létre a gépen, amely lehetőséget ad arra, hogy egy MySQL adatbázist üzemeltessenek.

## **3. Felhasználói dokumentáció**

### **3.1 Rendszerkövetelmények**

#### **3.1.1 Minimális rendszerkövetelmények**

Windows operációs rendszer (32-bit és 64-bit)

- CPU: Intel Core i3-3240 2x3,4GHz vagy újabb
- RAM: 512MB (32-bites rendszernél) / 2GB (64-bites rendszernél)
- HDD: 200 MB
- GPU: Integrált Videókártya 64MB RAM
- Monitor: 800x600-as felbontás vagy nagyobb
- OP rendszer: Microsoft Windows 7 vagy újabb
- Internet

Linux operációs rendszer

- CPU: Intel Core i3-3240 2x3,4GHz vagy újabb
- RAM: 512MB (32-bites rendszernél) / 2GB (64-bites rendszernél)
- HDD: 200 MB
- GPU: Integrált Videókártya 64MB RAM
- Monitor: 800x600-as felbontás vagy nagyobb
- OP rendszer: Ubuntu, Debian, Fedora, stb...
- Internet

### Macintosh operációs rendszer

- CPU: Macintosh számítógép egy Intel x86 processzorral vagy jobb
- RAM: 512MB
- HDD: 200MB
- GPU: Integrált Videókártya 64MB RAM
- Monitor: 800x600-as felbontás vagy nagyobb
- OP rendszer: macOS 10.9 – 10.13
- Internet

### Android operációs rendszer

- CPU: 800MHz Single-Core ARM1176JZF-S vagy újabb
- RAM: 256MB SDRAM vagy több
- Belső tárhely: 512MB (Min. 180MB felhasználó által is elérhető) vagy több
- Képernyő méret: 3,2” inch vagy nagyobb
- Felbontás: 320 x 480 pixel vagy nagyobb
- OP rendszer: Android 1.5 vagy újabb
- Internet

### 3.1.2 Ajánlott rendszerkövetelmények

Windows operációs rendszer (32-bit és 64-bit)

- CPU: AMD Ryzen 5 2600 Six-Core vagy jobb
- RAM: Kingston 16GB DDR4 vagy jobb és több
- HDD: 1TB vagy több
- GPU: AMD RX 560 vagy jobb
- Monitor: 1920x1080-as felbontás vagy nagyobb
- OP rendszer: Microsoft Windows 10 vagy újabb
- Internet

Linux operációs rendszer

- CPU: AMD Ryzen 5 2600 Six-Core vagy jobb
- RAM: Kingston 16GB DDR4 vagy jobb és több
- HDD: 1TB vagy több
- GPU: AMD RX 560 vagy jobb
- Monitor: 1920x1080-as felbontás vagy nagyobb
- OP rendszer: Ubuntu, Debian, Fedora, stb...
- Internet

Macintosh operációs rendszer

- CPU: Intel Xeon E5 25MB vagy jobb
- RAM: 16GB vagy több
- HDD: 1TB vagy több
- GPU: Radeon Pro W6800X Duo vagy jobb
- Monitor: 1920x1080-as felbontás vagy nagyobb
- OP rendszer: macOS 10.9 – 10.13
- Internet

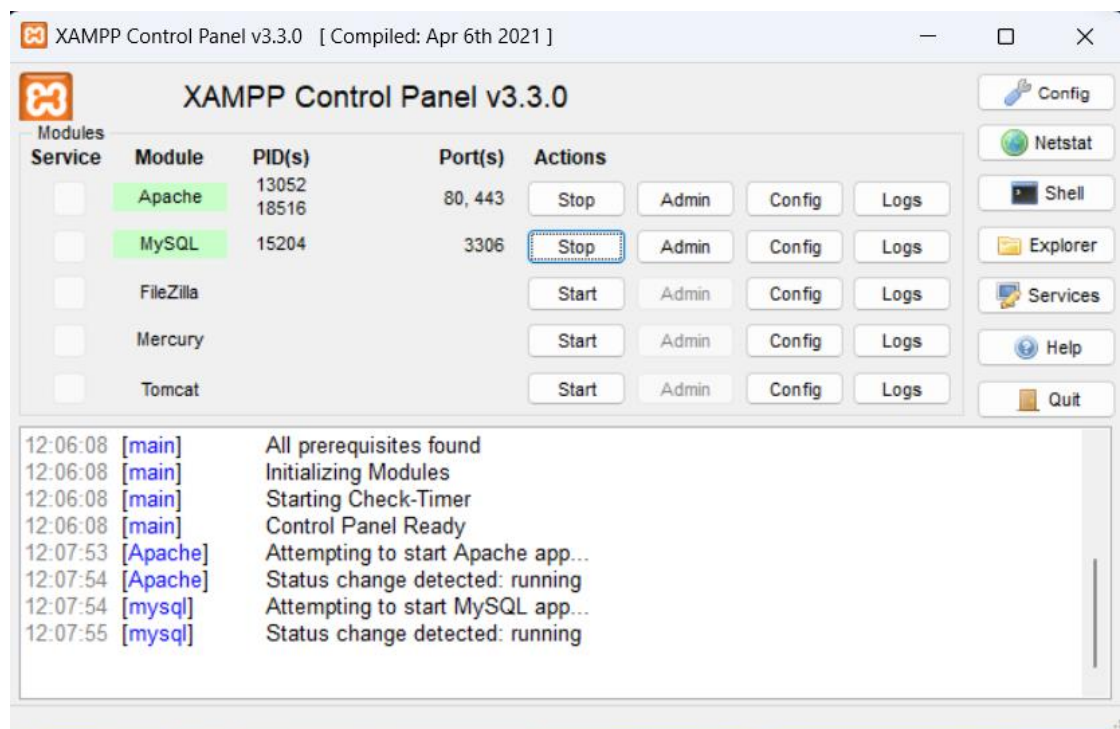


## Android operációs rendszer

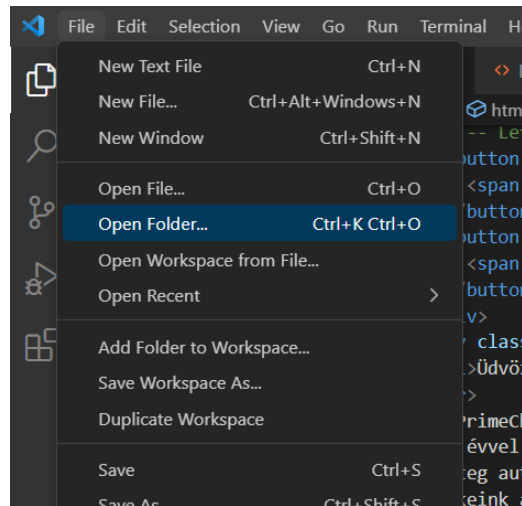
- CPU: Okta-mag (2x2,05 GHz Cortex-A76 és 6x2,0 GHz Cortex-A55) vagy jobb
- RAM: 6GB vagy több
- Belső tárhely: 64GB vagy több
- Képernyő méret: 6,53” inch vagy nagyobb
- Felbontás: 1080x2340 pixel vagy nagyobb
- OP rendszer: Android 9.0 vagy újabb
- Internet

## 3.2 A program megnyitása

Ahhoz, hogy a programot használni tudjuk, mindenképp be kell kapcsolnunk az eszközt, amin használni szeretnénk. Ha asztali számítógépet vagy laptopot használ nyomja meg a bekapcsoló gombot és várjunk amíg a rendszer üzemképes állapotba nem kerül. Amennyiben úgy véli felfedezni, hogy a rendszer teljes mértékben betöltött, el kell indítanunk az adatbázist, amelyet a XAMPP segítségével tudunk megejteni. A XAMPP elindítása után az ablakon az Apache és a MySQL modult is el kell indítani a start gomb megnyomásával.

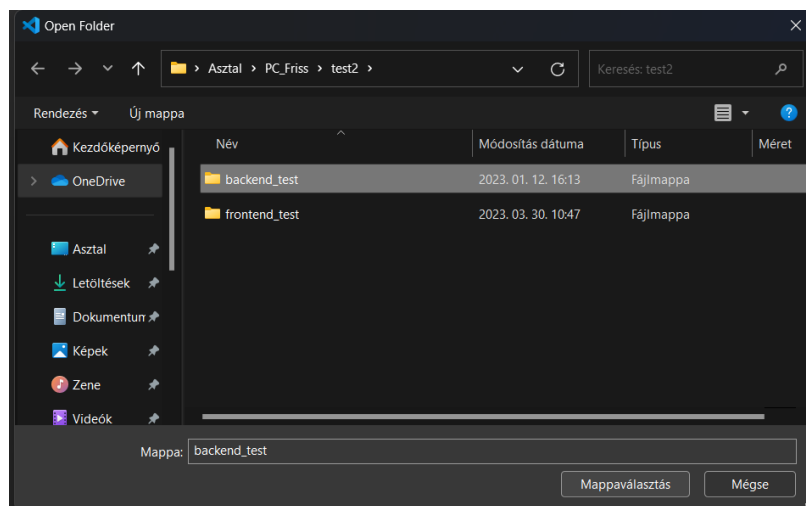


1. ábra Program indítása Xampp



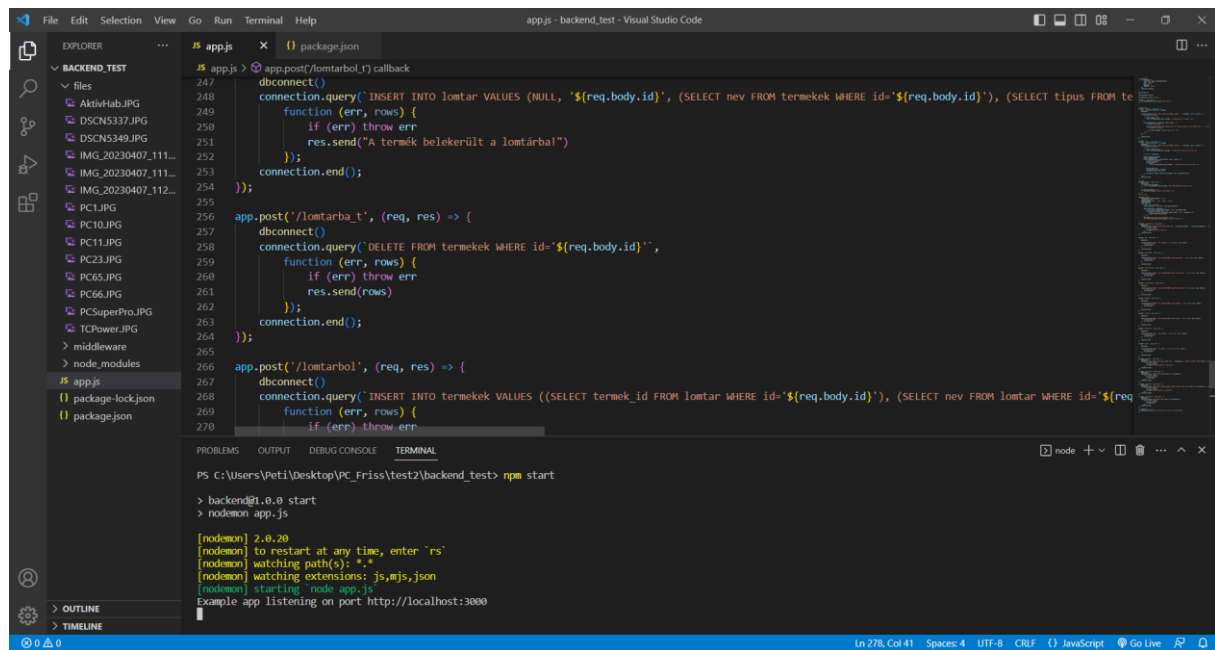
2. ábra Program indítása VSCode - mappa tallózása

Ez után a Visual Studio alkalmazást kell igénybe vennünk. A program megnyitása után meg kell nyitnunk programunk „backend” nevű mappáját.



3. ábra Program indítása VSCode

A mappa megnyitása után bal oldalt ki kell választanunk az App.js nevű fájlt majd az alul felugró terminál ablakba be kell írni az „npm start” kódsort amely hatására el fog indulni a program backendje.



#### 4. ábra Program indítása VSCode npm starttal

Ezek után pedig a programunk „frontend” nevű mappájában ki kell keresnünk az index.html nevű fájlt, amely a programunk kezdőlapjára fog dobni. Ehhez viszont szükségünk van egy böngészőre, amelyen keresztül meg tudjuk nyitni weblapunkat. A következő listában felsorolt böngészők alkalmasak erre:

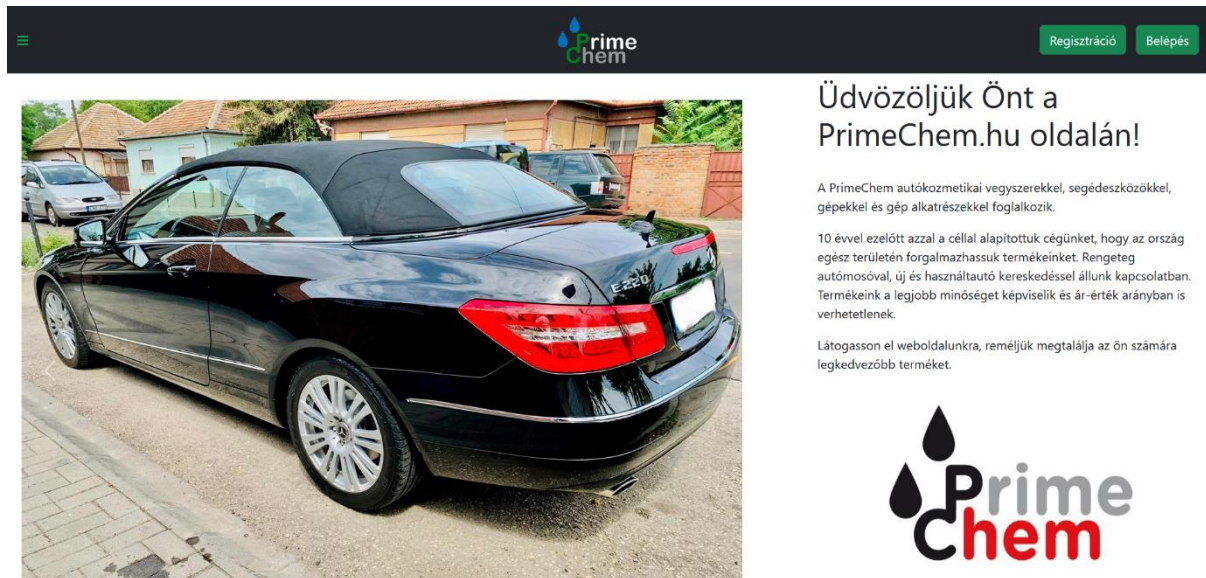
- Google Chrome
- Mozilla Firefox
- Opera vagy Opera GX
- Safari
- Microsoft Edge

Ha bármelyiket is felismerte a listában szereplő böngészők közül, és van is a számítógépén olyan, akkor minden készen áll arra, hogy elindítsuk az index.html nevű fájlunkat.

### 3.3 A program használata

#### 3.3.1 Menüpontok (Felhasználói felület, bejelentkezés nélkül)

A weboldal betöltését követően ez a kép fogadja a felhasználót:



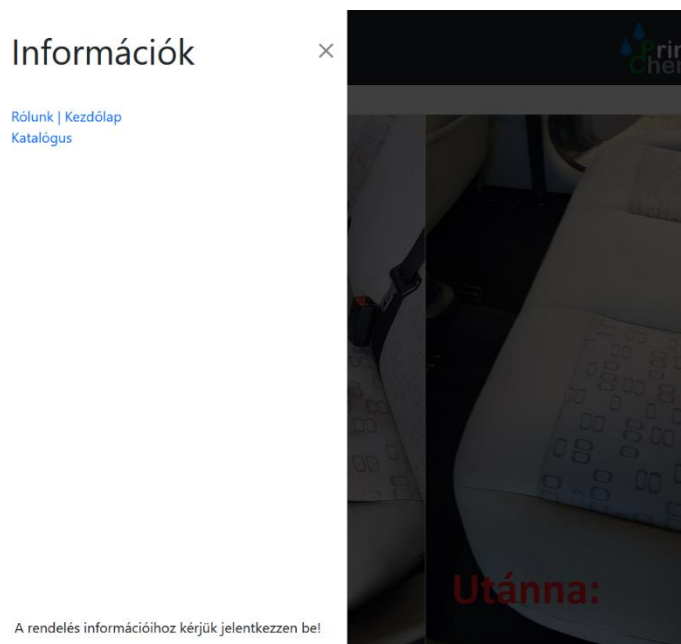
5. ábra Kezdőlap

A felület tetején találunk egy navigációs részt, amelyen megtalálható:

- 3 vonallal jelölt oldalról előjövő menü
- PrimeChem logó, ami a kezdőlapra irányít vissza
- Regisztrációs gomb
- Belépés gomb

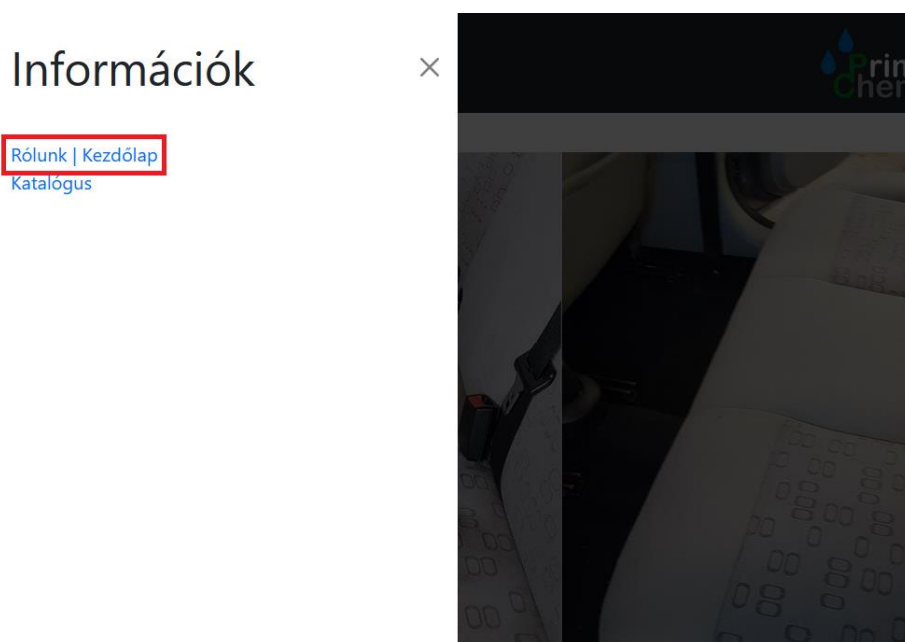
A fenti képen található a főoldal, amelynek bal felső sarkában található egy 3 vonallal jelölt oldalról behúzódó menü. (A könnyebb megtalálás érdekében piros négyzettel kereteztem be azokat a pontokat, amikről szó lesz.)

A gombot megnyomva ez a kép tárul elénk:



6. ábra Oldalról becsúszó fül megnyitása 1.

A „Rólunk | Kezdőlap” gombra kattintva a fentebb mutatott főoldal fog megjelenni.



7. ábra Rólunk Kezdőlap gomb

A „katalógus” gombra kattintva pedig megjelenik egy újabb lenyíló menü, amelyben megtalálhatóak a termék kategóriák.

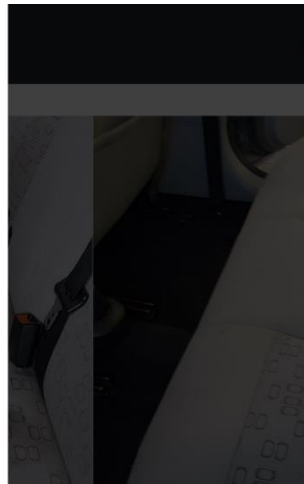
## Információk

[Rólunk](#) | [Kezdőlap](#)

[Katalógus](#)

- Tisztítószer
- Samponok
- Viaszok
- Felnitisztítók
- Ablaktisztítók
- Ápolószer

×



8. ábra Katalógus gomb lenyitva

Bármelyik kategóriára kattintva egy új oldal fog megjelenni, amelyen megtalálható az összes adatbázisba feltöltött termék abban a kategóriában, amelyre rákattintottunk.

A megnyíló oldalak:



[Regisztráció](#)[Belépés](#)

### Tisztítószer



PC-01

Leírás: Általános tisztítószer.



PC-10



Leírás: Aktív előmosó



PC-11


Leírás: Motormosó, nyári előmosó

9. ábra Megnyíló oldal - Tisztítószer





[Regisztráció](#)
[Belépés](#)

### Samponok





**PC-31**  
 Leírás: Kézi sampon  
 Ár: 690Ft/liter vagy kilogramm  
 Kiszerezés: 1-20l




**PC Super Pro**  
 Leírás: Porsampon

10. ábra Megnyíló oldal - Samponok





[Regisztráció](#)
[Belépés](#)

### Viaszok





**PC-65**  
 Leírás: Gurulógyőngyv




**PC-66**  
 Leírás: Vizes viasz

11. ábra Megnyíló oldal - Viaszok

[Regisztráció](#)
[Belépés](#)

### Felntítisztítók



**PC-23**  
 Leírás: Fém tisztító (savas)

12. ábra Megnyíló oldal - Felntítisztítók



## Ablaktisztítók



**PC-55**  
 Leírás: Üveg tisztító  
 Ár: 990Ft/liter  
 Kiszerezés: 1-25l

13. ábra Megnyíló oldal – Ablaktisztítók

## Ápolószerek



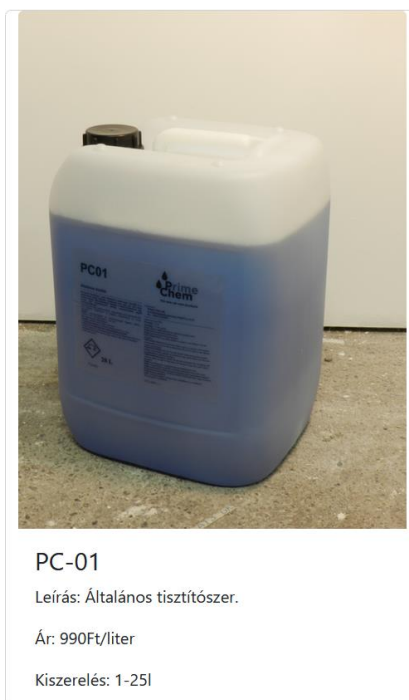
**PC-41**  
 Leírás: Belső műanyagápoló matt  
 Ár: 1690Ft/liter  
 Kiszerezés: 1-20l



**PC-47**  
 Leírás: Gumi és külső műanyag  
 Ár: 2790Ft/liter  
 Kiszerezés: 1-20l

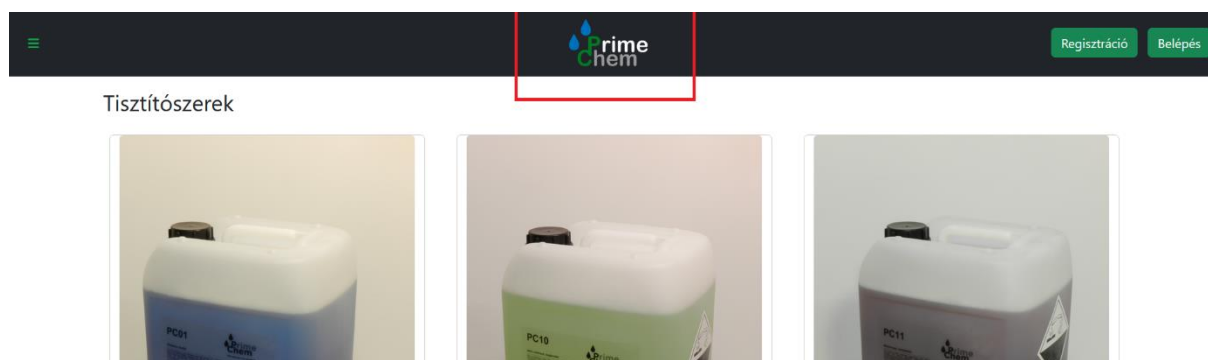
14. ábra Megnyíló oldal - Ápolószerek

A megnyíló oldalakon kártyákban vannak elhelyezve a termékek képei, nevei, rövid leírásai, árai és kiszerezései. Ezeket az adatokat mind az adatbázisból nyeri ki a program.



15. ábra Termék kártya

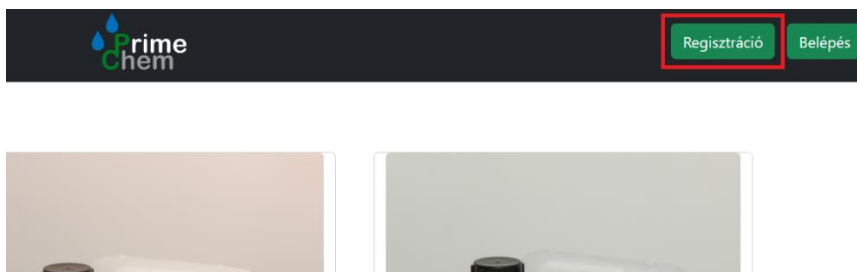
Minden oldal tetején megtalálható egy PrimeChem logó, ami egy gombként funkcionál. Bármelyik oldalon rákattintunk mindig a főoldalra dob vissza.



16. ábra Primechem logo gomb

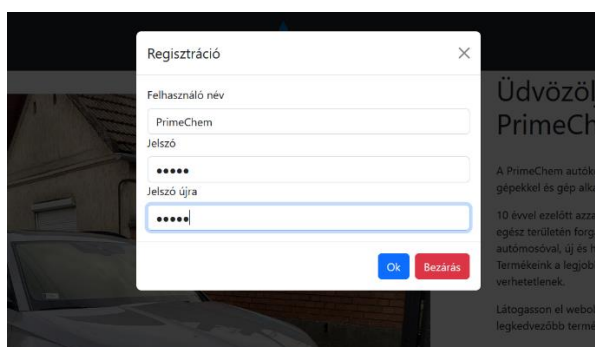
### 3.3.2 Regisztráció és bejelentkezés

Mindegyik oldal jobb felső sarkában van egy „Regisztráció” és egy „Belépés” gomb, ezáltal a felhasználó bárhol is tud regisztrálni az oldalra és bárhol is tud lépni a már meglévő fiókjába.



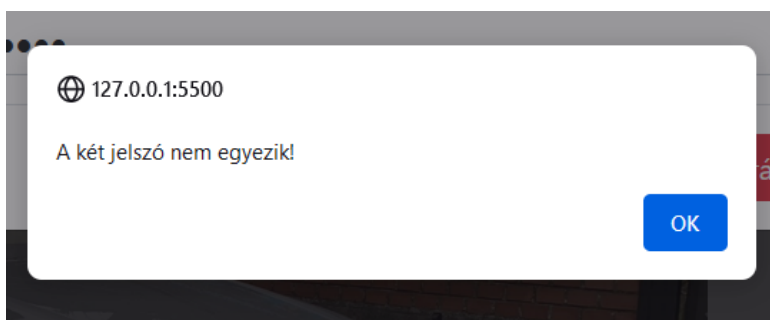
17. ábra regisztráció és belépés gomb

A regisztráció gombra kattintva egy modális ablak fog megjelenni, amelyen 3 darab mező helyezkedik el. Az elsőbe a jövőbeli felhasználónevet kell beírni, a másik kettőbe pedig a kitalált jelszót, ezekbe a mezőbe beírt karakterek pontokként jelennek meg a titkosítás jegyében.



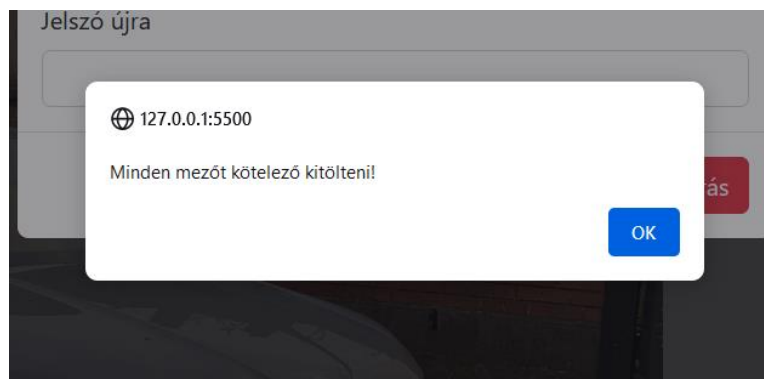
18. ábra a regisztráció ablaka

Ha a két jelszó nem egyezik meg, akkor egy kisebb felugró ablakon a következő hibaüzenet jelenik meg:



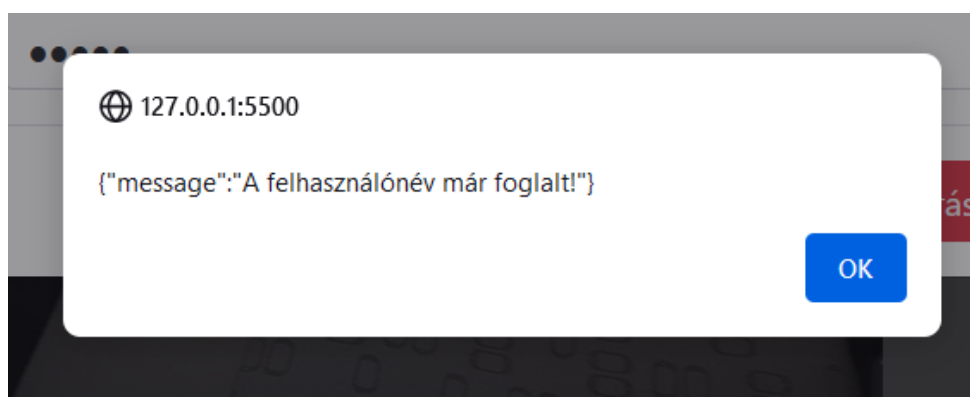
19. ábra Hibaüzenet 1

Ha valamelyik mező üresen marad akkor ez az üzenet jelenik meg:



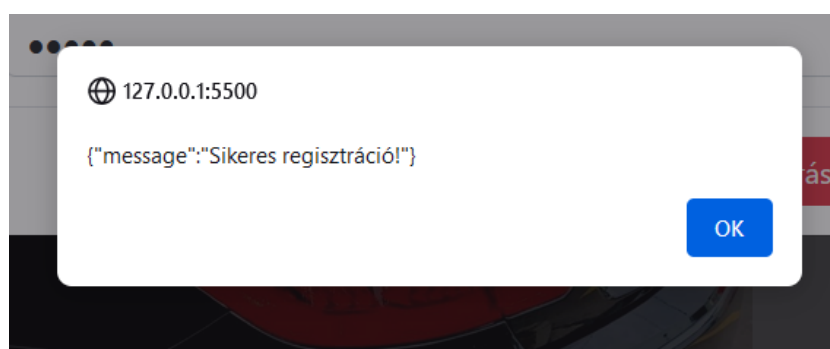
20. ábra Hibaiüzenet 2

Ha a felhasználó olyan felhasználó nevet próbál beírni, amit már tartalmaz az adatbázis, tehát foglalt, akkor a következő üzenet jelenik meg:



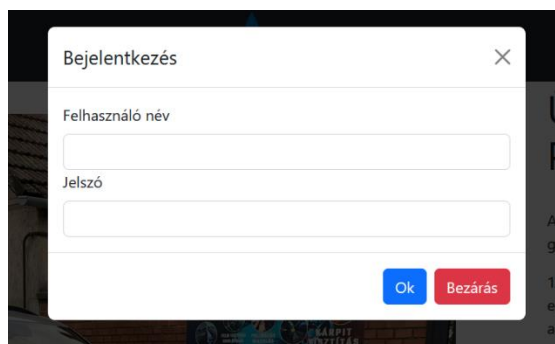
21. ábra Hibaiüzenet 3

Ha a két jelszó megegyezik, a sikeres regisztrációt követően ez az üzenet jelenik meg:



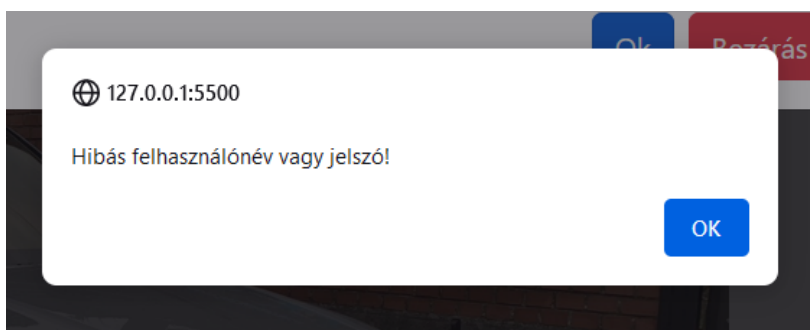
22. ábra Sikeres regisztráció üzenete

Sikeres regisztrációt követően a felhasználó minden probléma nélkül használhatja a regisztráció gombtól jobbra lévő belépés gombot. A gombra kattintást követően ugyan úgy, mint a regisztrációnál, itt is egy modális ablak jelenik meg, viszont itt csak kettő darab bemeneti mező található meg (felhasználónév és jelszó).

A modális ablak a "Bejelentkezés" címmel rendelkezik, és egy zárási gombbal (X) a jobb felső sarokban. Két bemeneti mező található: "Felhasználó név" és "Jelszó". Az ablak alján két gomb van: "Ok" (kék) és "Bezárás" (piros).

23. ábra Bejelentkezés modális ablak

Ugyan úgy, mint a regisztrációnál, itt is történhetnek problémák a bejelentkezés során, például az üresen maradt mező és a rosszul beírt jelszó vagy felhasználónév.



24. ábra Bejelentkezés hiba 1.

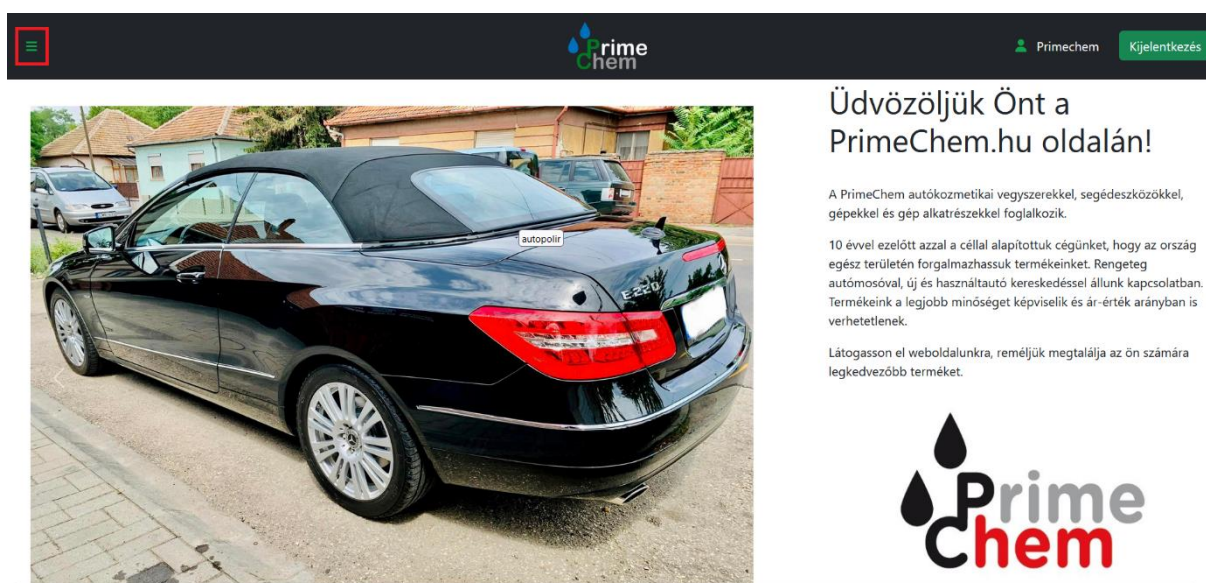
### 3.3.3 Bejelentkezett felhasználó (Változások)



25. ábra Bejelentkezett felhasználónév és kijelentkezés

Sikeres bejelentkezést követően úgy tűnhet, hogy csak újra a főoldalra dobott vissza, pedig van 1-2 szembetűnő változás. Amit elsőre észrevehet az ember az az oldal jobb felső sarkában található „Kijelentkezés” gomb, és a tőle balra lévő design elem, ami írja a bejelentkezett felhasználó felhasználónevét és egy kis zöld ikont.

Ezen változások mellett a bal felső sarokban található oldalról behúzódó menüben is történtek változások. A menüt megnyitva egy újabb kattintható linkkel bővült a lehetőségek listája. Ezt „Rendelés információk” néven találjuk meg.



26. ábra Rendelés oldal megnyitása 1.

## Információk



[Rólunk | Kezdőlap](#)

[Katalógus](#)

[Rendelés Információk](#)



27. ábra Rendelés oldal megnyitása 2.

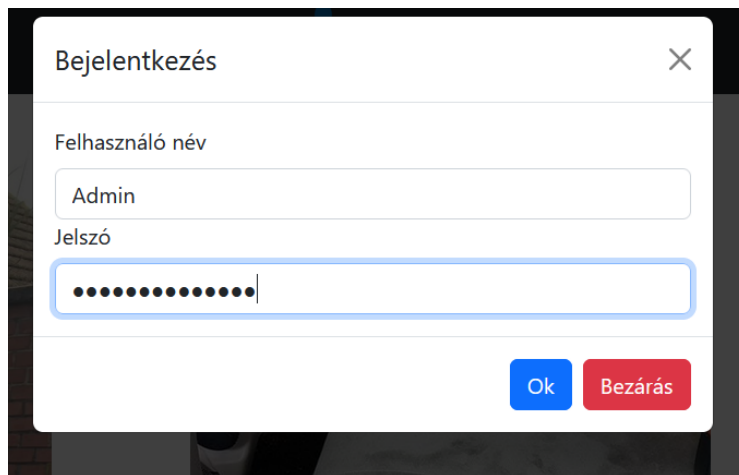
A linkre kattintva egy új oldal jeleni meg a képernyőn, amely tartalmazza a rendeléshez szükséges információkat. Az oldalon lehetőség van email címen keresztül rendelni, ha az email címre kattintunk egy kiválasztott levelező alkalmazással már írhatjuk is az üzenetünket.

Primechem [Kijelentkezés](#)

28. ábra Rendelés oldal

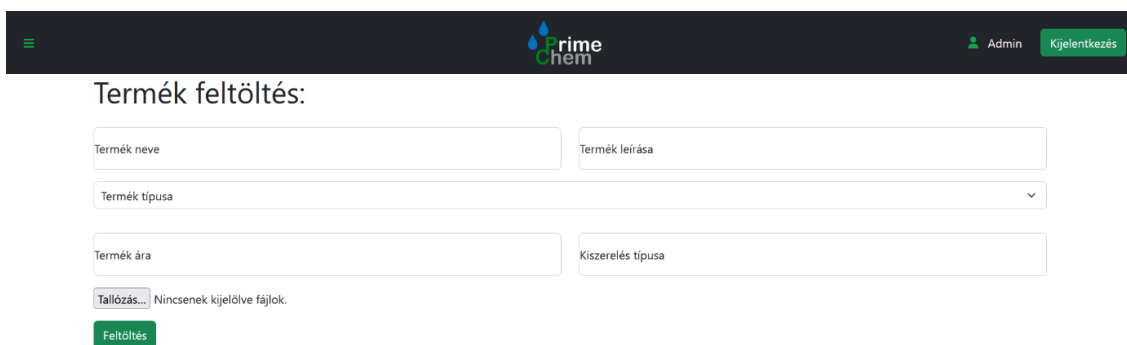
### 3.3.4 Admin felület és lehetőségei

Az programnak van egy háttér része, amely csak az oldal üzemeltetőjének elérhető. Erre az oldalra is úgy kell bejelentkezni mint egy sima felhasználónak viszont ide csak egy megadott felhasználónévvel és jelszóval lehetséges a belépés.



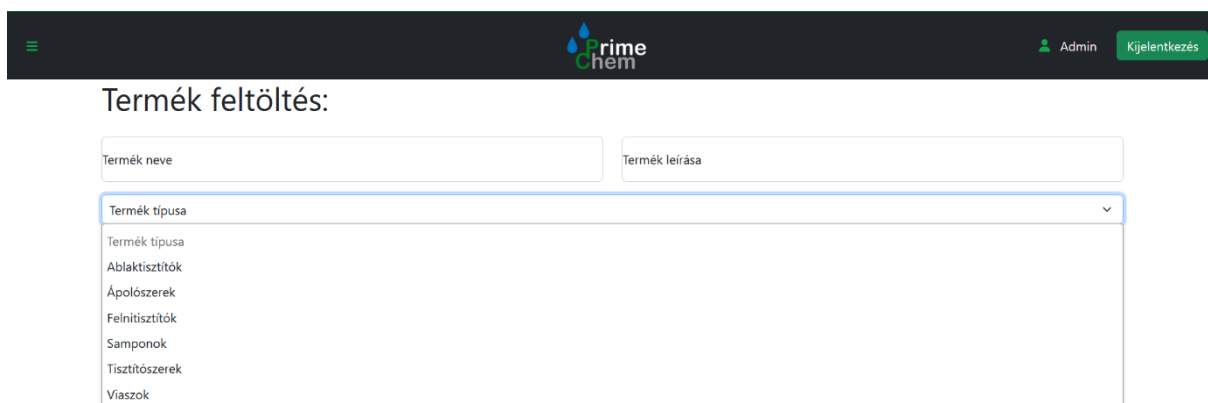
29. ábra Admin bejelentkezés

Az admin fiókba történő sikeres bejelentkezés után a következő oldal tárul elénk:



30. ábra Admin kezdőlap

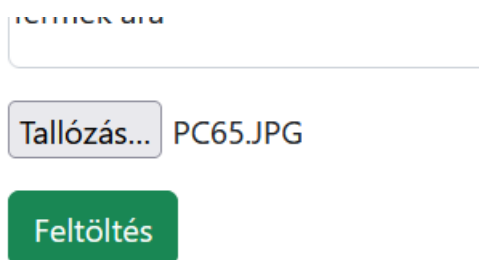
Ezen az oldalon van lehetősége felvinni termékek adatait az adatbázisba. Az első kettő mező sima szöveges beviteli mező, a harmadik egy select menü, ami kattintásra egy lenyíló lista lesz melyből ki tudjuk választani a termékek típusait.



31. ábra Admin termék feltöltés lenyíló menü



A negyedik és az ötödik is sima beviteli mező, ezek után található egy tallózás gomb, amellyel kiválaszthatunk egy kép típusú fájlt, kiválasztás után a kép neve megjelenik a gomb mellett. Mindezek adatbázisba mentéséhez meg kell nyomnunk a „Feltöltés” gombot.



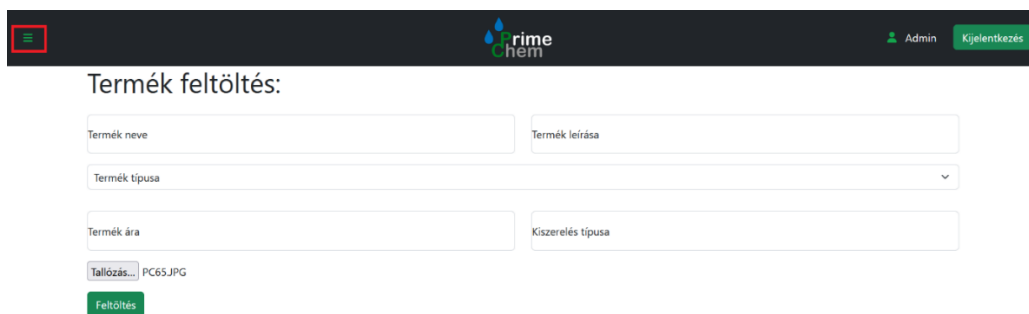
Termék ára

Tallózás... PC65.JPG

Feltöltés

32. ábra A feltöltés gomb

Az oldal bal felső sarkában található egy három vonallal jelzett oldalról behúzódó menü, amelyben található egy „Termék feltöltés” link, ami az admin kezdőoldalra dob vissza, egy „Termékek listája” link és egy „Lomtár” link.



Termék feltöltés:

Termék neve

Termék leírása

Termék típusa

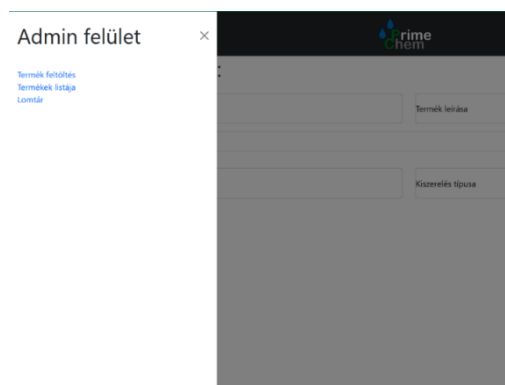
Termék ára

Kiszereles típusa

Tallózás... PC65.JPG

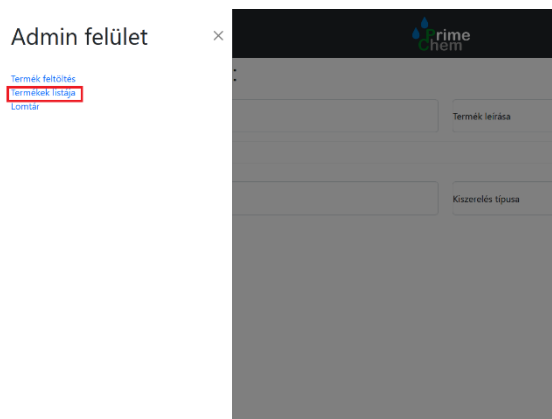
Feltöltés

33. ábra Admin oldalról behúzódó menü megnyitása



34. ábra Admin oldalról behúzódó menü

Ha a „Termékek listája” linkre kattintunk egy új oldalra dobja a program a használgját, melyen megtekinthetjük az oldalon fentlévő termékek listáját táblázat kialakításban.



35. ábra Termékek listája gomb

Termékek listája:					
Termék neve	Termék típusa	Termék leírása	Termék ára	Termék kiszerezése	Lomtárba helyezés
PC-01	Tisztítószer	Általános tisztítószer.	990	1-25l	Termék törlése
PC-10	Tisztítószer	Aktív előmosó	1290	1-25l	Termék törlése
PC-23	Felintisztítók	Fém tisztító (savas)	1490	1-25l	Termék törlése
PC-65	Viaszok	Gurulógyöngy	1990	1-25l	Termék törlése
PC-66	Viaszok	Vizes viasz	2490	1-25l	Termék törlése
PC-11	Tisztítószer	Motormosó, nyári előmosó	1790	1-25l	Termék törlése
PC-31	Samponok	Kézi sampon	690	1-20l	Termék törlése
PC-41	Ápolószer	Belső műanyagápoló matt	1690	1-20l	Termék törlése
PC-47	Ápolószer	Gumi és külső műanyag	2790	1-20l	Termék törlése
PC Super Pro	Samponok	Porsampon	12900	25kg	Termék törlése

36. ábra Termékek listája

Mindegyik sorban megtalálható a termék neve, típusa, leírása, ára és kiszerezése, valamint egy Termék törlése gomb, amellyel kitörölhetjük az adott sorban lévő terméket a megjelenő termékek közül és az átkerül a lomtárba.

Lomtár:					
Termék neve	Termék típusa	Termék leírása	Termék ára	Termék kiszerezése	Lomtárba helyezés
PC-01	Tisztítószer	Általános tisztítószer.	990	1-25l	Termék visszahívása

37. ábra Lomtár

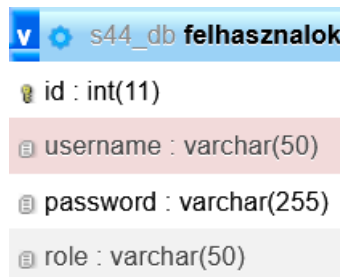
A lomtárban megtalálhatóak a listából törölt termékek adatai és egy „Termék visszahívása” gomb, amellyel a törölt terméket visszaküldhetjük a listába az adatvesztés nélkül.

## 4. Fejlesztői dokumentáció

### 4.1 Adatbázis táblák

#### 4.1.1 Felhasználók

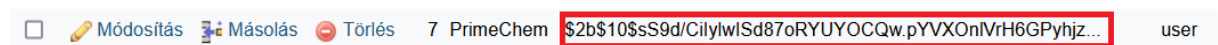
A „felhasznalok” táblát a felhasználók adatainak tárolására használjuk.



s44_db felhasznalok	
id	int(11)
username	varchar(50)
password	varchar(255)
role	varchar(50)

38. ábra Felhasznalok tábla

1. Az első oszlop, az „id” kapta a tábla elsődleges kulcsát. Az oszlop INT típusú és auto increment van hozzá beállítva, tehát nem kell neki értéket beállítani mert automatikusan kap egyet. Ez alapján lehet listázni a felhasználókat.
2. A második a „username”, tehát a felhasználónév, VARCHAR típusú, oszlop tárolja a regisztrációkor megadott felhasználónevet. Bejelentkezés után innen kérjük le a felhasználó nevét, hogy felül, a navigációs részen is írja.
3. A harmadik a „password”, azaz a jelszó oszlop VARCHAR típusú. Itt tároljuk a felhasználókhoz tartozó jelszavakat. Ezek a jelszavak az adatbázisba már titkosítva kerülnek be, így a felhasználón kívül senki sem tudja a jelszót.



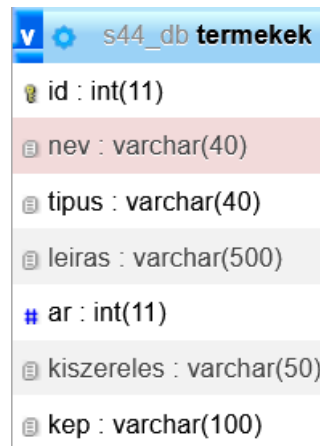
<input type="checkbox"/>	Módosítás	Másolás	Törlés	7 PrimeChem	<span style="border: 2px solid red; padding: 2px;">\$2b\$10\$sS9d/CiylwlSd87oRYUYOCQw.pYVXOnVrH6GPyhjz...</span>	user
--------------------------	-----------	---------	--------	-------------	--	------

39. ábra Titkosított jelszó

4. A negyedik, utolsó VARCHAR típusú oszlop tartalmazza a felhasználó jogosultságát, ezért ez az oszlop a „role” névre hallgat. Két féle jogosultság létezik:
  - user: Ezt kapja a felhasználó alapértelmezetten.
  - admin: ezt kapja az üzemeltető, amellyel az admin oldalra kap belépési lehetőséget. Ezt az adatbázisban kell átírni.

### 4.1.2 Termékek

A „termékek” táblát a termékek adatainak tárolására használjuk.



The screenshot shows the structure of the 'termékek' table in the 's44\_db' database. The table has the following columns:

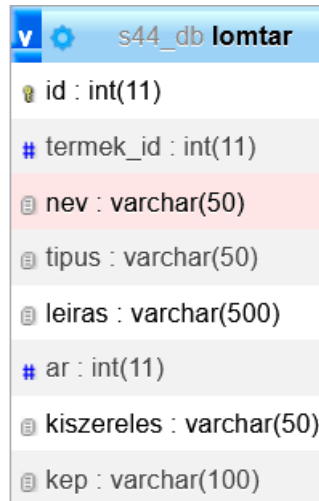
Column Name	Data Type
id	int(11)
nev	varchar(40)
tipus	varchar(40)
leiras	varchar(500)
ar	int(11)
kiszereles	varchar(50)
kep	varchar(100)

40. ábra Termékek tábla

1. Az első oszlop itt is „id” és itt is ez az elsődleges kulcs, úgy, mint a „felhasznalok” táblában. Ez egy INT típusú oszlop, ahol be van állítva az auto increment, tehát ez is automatikusan kap értéket. Ez alapján vannak listázva a termékek az admin felületen, a „termékek listája” oldalon.
2. A második VARCHAR típusú oszlop tartalmazza a termék nevét.
3. A harmadik oszlop is VARCHAR típusú, ez tartalmazza a termék típusát.
4. A negyedik oszlop tartalmazza a termék leírását, ez a szöveges típusú oszlop karakterhatára 500 karakter, mivel egy leírás elég hosszú is lehet.
5. Az ötödik oszlop (INT) a termékek árainak tárolásáért felelős.
6. A hatodik a termékek kiszereléseit tárolja (Pl.:1-25l) VARCHAR típusban.
7. Az utolsó oszlop tartalmazza a termékekhez kiválasztott kép típusú fájl nevét.

### 4.1.3 Lomtar

A „lomtar” táblába kerülnek a „termek” táblából törölt adatok. Felépítése majdnem megegyezik az előbb említett tábla szerkezetével.



The screenshot shows a database table definition for 's44\_db lomtar'. The table has the following columns and data types:

Column Name	Data Type
id	int(11)
termek_id	int(11)
nev	varchar(50)
tipus	varchar(50)
leiras	varchar(500)
ar	int(11)
kiszeres	varchar(50)
kep	varchar(100)

41. ábra Lomtar tábla

1. Az első „id” azonosító oszlop a lomtárba kerüléskor automatikusan kapott értéket tartalmazza.
2. A második „termek\_id” a „termek” táblában kapott azonosítót (id-t) tartalmazza. Ez azért kell, hogy a „termek” táblába való visszaküldéskor a régi azonosítóját kapja meg, így az adat oda-vissza küldésekor semmi sem vesz el.
3. 3-8. A további oszlopok a termékek adatait tartalmazzák.

## 5. Fontosabb kód részletek

### 5.1 Regisztráció adatainak felvitele az adatbázisba és hibák kezelése.

Backend – app.js

```
//Regisztráció
app.post('/register', (req, res) => {
  const { username, password } = req.body;
  dbconnect()

  connection.query('SELECT * FROM felhasznalok WHERE username = ?',
[username], (error, results) => {
    if (error) throw error;

    if (results.length > 0) {
      return res.status(400).json({ message: 'A felhasználónév már
foglalt!' });
    }

    bcrypt.hash(password, saltRounds, (error, hash) => {
      if (error) throw error;

      connection.query('INSERT INTO felhasznalok (id, username,
password, role) VALUES (NULL, ?, ?, "user")', [username, hash], (error) => {
        if (error) throw error;

        res.json({message: "Sikeres regisztráció!"});
      });
    });
  });

  connection.end
});
```

42. ábra Regisztráció backend

```
//a tényleges regisztráció
function registration() {
    if (document.getElementById('regUsername').value != "" &&
document.getElementById('regPassword1').value != "" &&
document.getElementById('regPassword2').value != "") {
        if (document.getElementById('regPassword1').value ==
document.getElementById('regPassword2').value) {
            let datas = {
                username: document.getElementById("regUsername").value,
                password: document.getElementById("regPassword1").value,
            };
            let url = "http://localhost:3000/register";
            let fetchOptions = {
                method: "POST",
                body: JSON.stringify(datas),
                headers: { "Content-type": "application/json; charset=UTF-8" }
            };
            fetch(url, fetchOptions)
                .then(x => x.text())
                .then(y => {
                    alert(y);
                    document.getElementById('regUsername').value = "";
                    document.getElementById('regPassword1').value = "";
                    document.getElementById('regPassword2').value = "";
                });
        }
        else {
            alert("A két jelszó nem egyezik!");
        }
    }
    else {
        alert("Minden mezőt kötelező kitölteni!");
    }
}
```

43. ábra Regisztráció frontend 1.



```

    <!-- Regisztráció Modal -->
    <div class="modal fade" id="registration" data-bs-backdrop="static"
data-bs-keyboard="false" tabindex="-1"
    aria-labelledby="staticBackdropLabel" aria-hidden="true">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title"
id="staticBackdropLabel">Regisztráció</h5>
                <button type="button" class="btn-close" data-bs-
dismiss="modal" aria-label="Close"></button>
            </div>
            <div class="modal-body">
                <label class="form-label">Felhasználó név</label>
                <input class="form-control" type="text" name="regUsername"
id="regUsername" required>
                <label class="form-label">Jelszó</label>
                <input class="form-control" type="password"
name="regPassword1" id="regPassword1" required>
                <label class="form-label">Jelszó újra</label>
                <input class="form-control" type="password"
name="regPassword2" id="regPassword2" required>
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-primary"
onclick="registration()">Ok</button>
                <button type="button" class="btn btn-danger" data-bs-
dismiss="modal">Bezárás</button>
            </div>
        </div>
    </div>
</div>

```

44. ábra Regisztráció frontend 2.

A fent olvasható forráskódokban láthatjuk, hogy az index.html-ben egy modális ablakot hozunk létre és annak body részében találhatóak azok az input mezők, amelyekbe írjuk a regisztrációs adatokat (felhasználónév, jelszó). Mindegyik input mező egyedi azonosítóval (id-val) van ellátva ezáltal hivatkozhatunk rájuk.

A modal lábrészében (footer) található egy „ok” gomb, amely kattintásra meghívja a main.js-ben található „registration” függvényt.

A main.js-ben a „registration” function-ben először vizsgáljuk, hogy a bemeneti mezők nem üresek-e, ha üresek egy felugró ablakkal és egy hibaüzenettel jelzi nekünk a program. Ezután vizsgáljuk, hogy a két bekért jelszó megegyezik-e, ha nem akkor ezt is egy felugró ablakkal és egy hibaüzenettel jelzi nekünk a program. A vizsgálatokat követően eltároljuk a bekért felhasználónevet és jelszót, majd az adatokat fetcheléssel elküldjük a backendnek és a bemeneti mezők tartalmát töröljük.

A backend app.js nevű fájljában láthatjuk, hogy a dbconnect() paranccsal csatlakozunk az adatbázishoz. Ezután ellenőrizzük, hogy a bekért felhasználónév megtalálható-e az adatbázisban, ha igen akkor egy hibaüzenettel közli ezt velünk a program, ha nem akkor egy SQL parancs segítségével feltölti az adatbázisba a bekért adatokat. Ez egy „sikeres regisztráció” felugró ablakkal jelzi nekünk a program.

## 5.2 Termékek kártyáinak megjelenítése.

Backend – app.js

```
app.get('/ablaktisztitok', (req, res) => {  
  
    dbconnect()  
  
    connection.query(`SELECT * from termekek WHERE tipus="Ablaktisztító"`,  
function (err, rows, fields) {  
    if (err) throw err  
    res.json(rows);  
    })  
  
    connection.end()  
})
```

45. ábra Kártyák backend

Frontend – ablaktisztitok.html

```
<div class="container row" id="ablaktisztitok">  
    <h3>Ablaktisztító</h3>  
</div>
```

46. ábra Kártyák frontend 1.

```
function ablaktisztitok() {
  fetch('http://localhost:3000/ablaktisztitok')
    .then(x => x.json())
    .then(y => megjelenit(y));

  function megjelenit(adatok){
    console.log(adatok);
    let sz=`<h3 class="mt-3">Ablaktisztítók</h3>`
    for(var elem of adatok)
    {
      /*sz += '<div class="col-sm-4">'
      sz += ''
      sz += '</div>'
      */
      sz += `
      <div class="container card mt-3 ml-3 col-lg-4"
style="width:400px">
        
        <div class="card-body">
          <h4 class="card-title">${elem.nev}</h4>
          <p class="card-text">Leírás: ${elem.leiras}</p>
          <p class="card-text">Ár: ${elem.ar}</p>
          <p class="card-text">Kiszereelés: ${elem.kiszereles}</p>
        </div>
      </div>`
    }
    document.getElementById("ablaktisztitok").innerHTML=sz;
  }
}
```

47. ábra Kártyák frontend 2.

A kártyák minden oldalon típusonként jelennek meg, a kód ugyan az, csak a típus változik, ezért most az ablaktisztítók típusú termékek kártyáinak megjelenítését mutatom.

A backend app.js fájljában lekérjük az összes „Ablaktisztítók” típusú termék adatait az adatbázisból.

A frontend script.js fájljában az ablaktisztítók function-on belül lekérjük az adatokat a backendtől, majd egy új „megjelenit” function-ban létrehozunk egy „sz” változót, amelybe a megjelenő kártya HTML kódját írjuk meg. Ezt az „sz” nevű változót egy for ciklusba tesszük, és annyiszor ismételjük, ahány „Ablaktisztítók” típusú termék van az adatbázisban.

Ezek után már csak annyi a dolgunk, hogy az ablaktisztitok.html fájlban egy div-nek megadjuk az „ablaktisztitok” azonosítót, mivel JS fájlunk utolsó kódsorában (`document.getElementById("ablaktisztitok").innerHTML=sz;`) az „ablaktisztitok” azonosítót adtuk meg, így fog megjelenni a kártya.

## 6. Tesztelés

Az oldal felhasználók által való tesztelésére programozásban nem jártas emberek segítségét kértem.

Az oldal megnyitásakor, könnyen zökkenőmentesen betöltött, nem voltak hiányzó képek, az oldal alkalmazkodott az 1920\*1080 pixel méretű képernyőfelbontáshoz és az oldal méretezgetéséhez is. Regisztráció előtt és után is az újonnan megjelenő ablakok mindenhol megjelennek. Probléma volt, hogy az egyik oldal nem akart megjelenni, ehhez a script.js fájlban írtam egy új függvényt és a probléma megoldódott. Találtunk olyan problémát, hogy a termékfeltöltés után a kép nem jelent meg, az adatbázisban „undefined” -ként volt feltöltve. Ezt egy kódsor hozzáadásával orvosoltuk és így már az admin oldalon a termékfeltöltés után egyből látszódik a termék kártyája a megadott oldalon. A termék lista és a lomtár közötti kapcsolat is tökéletesen működik.

Meglátásai szerint a termékek listáját lehetne bővíteni, hogy nagyobb legyen a választék.

## 7. Fejlesztési lehetőségek

- Rendelés rendszer kialakítása
- A termékekre kattintva egy új ablak jelenne meg több információval
- Termékek értékelési lehetősége
- A cég értékelési lehetősége
- Termékek számainak bővítése
- A cégről több információ megadása egy külön oldalon
- Lábléc készítése design szempontjából

## 8. Összegzés

A programba terveztem egy rendelés rendszert, amelyben a választott termék és a felhasználó adatai felkerül az adatbázisba, majd ezeket az admin tudja fogadni és kezelni, viszont ez idő hiányában nem valósult meg. Tervezett dolog volt még egy értékelő rendszer, amely lehetőséget adott volna a termékek értékelésére és a bevitt adatok átlagából következtetni lehetett volna arra, hogy mennyire elégedettek egy adott termékkel, viszont idő hiányában ezt sem sikerült megoldani.

A program írása során rengeteget tanultam a HTML és a Bootstrap leíró nyelvekről és a bennük rejlő potenciálból, amelyeket majd fel tudok használni a jövőm során. Sok dolgot tanultam még a JavaScript nyelvről is, például a függvények használata és a frontend-backend közötti kommunikáció.

## 9. Irodalomjegyzék

- <https://www.w3schools.com/>
- <https://getbootstrap.com/>
- <https://stackoverflow.com/>
- <https://nodejs.org/en>
- <https://expressjs.com/>
- <https://www.npmjs.com/package/nodemon>
- <https://www.npmjs.com/package/express-fileupload>
- <https://developer.mozilla.org>



## 10. Ábrajegyzék

1. ábra Program indítása Xampp .....	8
2. ábra Program indítása VSCode - mappa tallózása .....	9
3. ábra Program indítása VSCode .....	9
4. ábra Program indítása VSCode npm starttal .....	10
5. ábra Kezdőlap.....	11
6. ábra Oldalról becsúszó fül megnyitása 1. ....	12
7. ábra Rólunk Kezdőlap gomb.....	12
8. ábra Katalógus gomb lenyitva.....	13
9. ábra Megnyíló oldal - Tisztítószer .....	13
10. ábra Megnyíló oldal - Samponok .....	14
11. ábra Megnyíló oldal - Viaszok.....	14
12. ábra Megnyíló oldal - Felnitisztítók .....	14
13. ábra Megnyíló oldal – Ablaktisztítók .....	15
14. ábra Megnyíló oldal - Ápolószer .....	15
15. ábra Termék kártya .....	16
16. ábra Primechem logo gomb .....	16
17. ábra regisztráció és belépés gomb .....	17
18. ábra a regisztráció ablaka.....	17
19. ábra Hibaüzenet 1 .....	17
20. ábra Hibaüzenet 2 .....	18
21. ábra Hibaüzenet 3 .....	18
22. ábra Sikeres regisztráció üzenete.....	18
23. ábra Bejelentkezés modális ablak.....	19
24. ábra Bejelentkezés hiba 1.....	19
25. ábra Bejelentkezett felhasználónév és kijelentkezés .....	20
26. ábra Rendelés oldal megnyitása 1. ....	20
27. ábra Rendelés oldal megnyitása 2. ....	21
28. ábra Rendelés oldal.....	21
29. ábra Admin bejelentkezés .....	22
30. ábra Admin kezdőlap .....	22
31. ábra Admin termék feltöltés lenyíló menü.....	22

32. ábra A feltöltés gomb .....	23
33. ábra Admin oldalról behúzódo menü megnyitása .....	23
34. ábra Admin oldalról behúzódo menü.....	23
35. ábra Termékek listája gomb .....	24
36. ábra Termékek listája .....	24
37. ábra Lomtár .....	24
38. ábra Felhasznalok tábla.....	26
39. ábra Titkosított jelszó.....	26
40. ábra Termékek tábla.....	27
41. ábra Lomtar tábla.....	28
42. ábra Regisztráció backend.....	29
43. ábra Regisztráció frontend 1. ....	30
44. ábra Regisztráció frontend 2. ....	31
45. ábra Kártyák backend .....	32
46. ábra Kártyák frontend 1. ....	32
47. ábra Kártyák frontend 2. ....	33

# Eredetiségnyilatkozat

Alulírott **Tóth Péter** nyilatkozom, hogy záró dolgozatom a saját szellemi termékem.

Debrecen, 2023. ápr. 12

**Tóth Péter** .

aláírás