Szakdolgozat

Készítette: Szendefi Krisztofer Dániel, Kun-Székely Bence

# Tartalomjegyzék

[1. Tartalomjegyzék 2](#_Toc170110496)

[2. Bevezetés és témaválasztás 6](#_Toc980104717)

[3. Tervezés, drótvázak 6](#_Toc1248070001)

[Tervezett funkciók 6](#_Toc811833574)

[Nyilvános menü 6](#_Toc1871501365)

[További nyilvános felület funkciók 6](#_Toc1687528075)

[Ügyviteli funkciók: 7](#_Toc1717579219)

[Drótvázak 8](#_Toc353714395)

[4. Felhasználói dokumentáció 9](#_Toc1542728504)

[A program célja, lényegesebb funkciói 9](#_Toc1018019729)

[A program elérése 10](#_Toc222644331)

[A program fő funkcióinak leírása 10](#_Toc165793725)

[Menüpontok az oldalon 10](#_Toc1352533918)

[Kezdőoldal 11](#_Toc1803442031)

[Kategóriák 11](#_Toc1814197104)

[Keresés 12](#_Toc1469570598)

[Felhasználó (csak bejelentkezet felhasználóknál) 12](#_Toc712707045)

[Kosár 13](#_Toc2138799842)

[Termékek adatainak megtekintése valamint válogatás 14](#_Toc328277011)

[Bejelentkezés és regisztráció 15](#_Toc1694209612)

[Bejelentkezés: 15](#_Toc1501235376)

[Admin felületek és elérésük 16](#_Toc1610500462)

[Modellek kezelése: 16](#_Toc54432769)

[Kategóriák kezelése: 19](#_Toc1010511109)

[Felhasználókezelés: 20](#_Toc1658529786)

[Rendelések: 20](#_Toc1477760579)

[Beszerzés: 22](#_Toc1087709987)

[Hibajelzések magyarázata 22](#_Toc625503652)

[Bejelentkezés és regisztrációs hibák: 22](#_Toc1522397592)

[Kosár hiba: 23](#_Toc596266365)

[Beszerzés hiba (admin felület): 23](#_Toc724367130)

[Felhasználó módosítás figyelmeztetés: 25](#_Toc958160167)

[Információkérés 25](#_Toc150278687)

[5. Fejlesztői dokumentáció 25](#_Toc1873538264)

[Fejlesztői környezet 25](#_Toc870854481)

[Adatbázis 26](#_Toc1532549208)

[Adatbázis-tervezés: 26](#_Toc1850480149)

[Egyed-kapcsolat diagramm: 28](#_Toc1603198519)

[Lekérdezések: 28](#_Toc1096727262)

[Triggerek: 28](#_Toc1542526704)

[Táblák szerkezete a táblaszintű megszorításokkal: 29](#_Toc1579977033)

[Backend 30](#_Toc757384397)

[Modellek és controllerek 31](#_Toc1497135841)

[API végpontok 32](#_Toc2103863708)

[Frontend 33](#_Toc197917974)

[Reszponzivitás 33](#_Toc639112461)

[Tesztelés 35](#_Toc348179734)

[Bejelentkezés és regisztráció 35](#_Toc696831471)

[Admin felület 35](#_Toc1867485409)

[Útvonal tesztek 36](#_Toc2105969328)

[6. Fejlesztési lehetőségek 40](#_Toc1484776673)

[Fejleszési lehetőségek: 40](#_Toc532362904)

[7. Összefoglalás, köszönetnyilvánítás 40](#_Toc966531112)

[8. Irodalomjegyzék 40](#_Toc828235653)

# Bevezetés és témaválasztás

GitHub repository: <https://github.com/szakdolgozat2024/Szakdolgozat.git>

A projekt célja egy általánosan használható webáruház létrehozása, amit egyszerűmódosítani és újrafelhsaználni bármilyen más boltnak. A bemutatás, átláthatóság érdekében az alkalmazást egy bútor, háztartási boltként építettük ki, ahol a vendégek bútorokat és háztartási elemeket böngészhetnek, kereshetnek és vásárolhatnak meg. A projekt célja egy lehetséges felhasználás minél több szükséges funkciójának biztosítása, valamint ezen funkciók használatának könnyűvé és átláthatóvá tétele. Ez többekközött tartalmazza a felhasználók kezelését, termékek közötti böngészés lehetőségét, valamint vásárolt termékek áttákinthetőségét. Emellett továbbá lehetőség van az olyan adminisztrációs feladatok ellátására mint pédául a kínálat szerkesztése, készlet valamint jogosulságok kezelése.

A applikációba való belépés vendégként is lehetséges de a teljes funkcionalitás eléréséhez regisztrálni kell. Az adminisztrációs funkciók eléréséhez admin jogosultságú felhasználóként kell belépni. Az oldal tesztelésének érdekében beépítettünk két felhasználót a két felhasználót vásárló illetve admin jogosultsággal.

# Tervezés, drótvázak

## Tervezett funkciók

### Nyilvános menü

* Kezdőlap: Navigáljon a kezdőlapra.
* Kategóriák: Bútorkategóriák listájának megtekintése, böngészés kategóriák alapján.
* Keresés: Adott bútorok keresése.
* Bevásárlókosár: A bevásárlókosárban lévő tételek megtekintése és szerkesztése.
* Felhasználói fiók:
* Bejelentkezés/Bejelentkezés: Meglévő felhasználók bejelentkezhetnek, vagy új felhasználók fiókokat hozhatnak létre.
* Felhasználói profil: Frissítse a személyes adatokat.
* Kijelentkezés: Kijelentkezés a felhasználói fiókból.

### További nyilvános felület funkciók

* Aktuális feltételek szerint termékek megjelenítése.
* Termékrészletek megtekintése: Részletes információk megjelenítése a kiválasztott termékről.
* Egyes elemek külön megjelenítése
* Kosárba helyezés: A vásárlók termékeket helyezhetnek a kosarukba.
* Termékiformációk megtekintése: A kiválasztott modellek termékei és paraméterei megtekintése.
* Kosár
* Kosár szerkesztése: A vásárlók módosíthatják bevásárlókosaruk tartalmát.
* Fizetés: Tovább a fizetéshez és a megrendelés visszaigazolásához.
* Felhasználói adatok módosítása.

### Ügyviteli funkciók:

Beszerzés:

* Termékek beszerzése

Termékek kezelése:

* Új termék hozzáadása: Az adminisztrátorok új termékeket adhatnak hozzá.
* Termék szerkesztése: Az adminisztrátorok frissíthetik a termék részleteit (pl. név, leírás, ár).
* Termék eltávolítása: A adminok eltávolíthatnak termékeket.

Modellek kezelése:

* Új modell hozzáadása: Az adminisztrátorok új bútorokat adhatnak a katalógushoz.
* Modell szerkesztése: Az adminisztrátorok frissíthetik a modell részleteit (pl. név, leírás, ár).
* Modell eltávolítása: A adminok eltávolíthatnak modelleket a katalógusból.

Kategóriák kezelése:

* Új kategória hozzáadása: Az adminisztrátorok új kategóriákat adhatnak hozzá.
* Kategória módoítása: Az adminisztrátorok módosíthatják a már meglévő kategóriák nevét.
* Kategória törlése: Az adminisztrátorok törölhetik a kategóriákat.

Felhasználókezelés:

* Felhasználók megtekintése: Az adminisztrátorok megtekinthetik az összes regisztrált felhasználó listáját.
* Felhasználói engedélyek módosítása: Az adminisztrátorok előléptethetik a normál felhasználókat rendszergazdai státuszba.

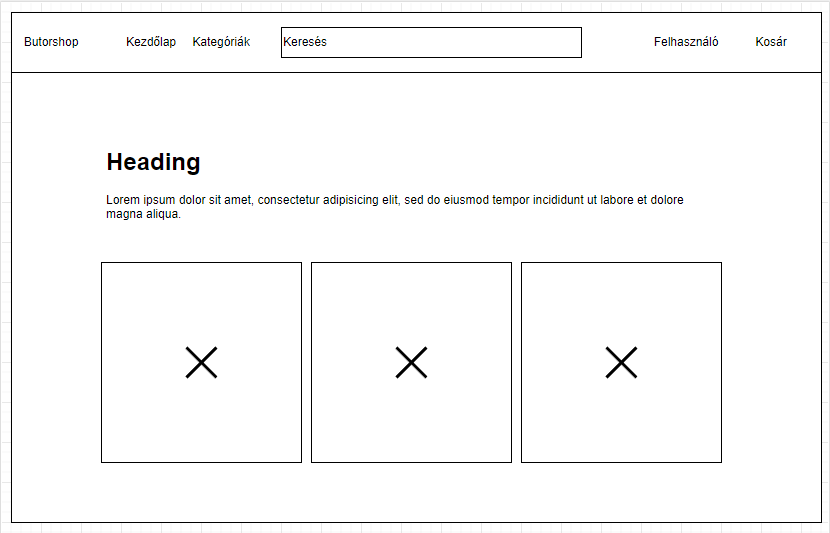
Rendelések:

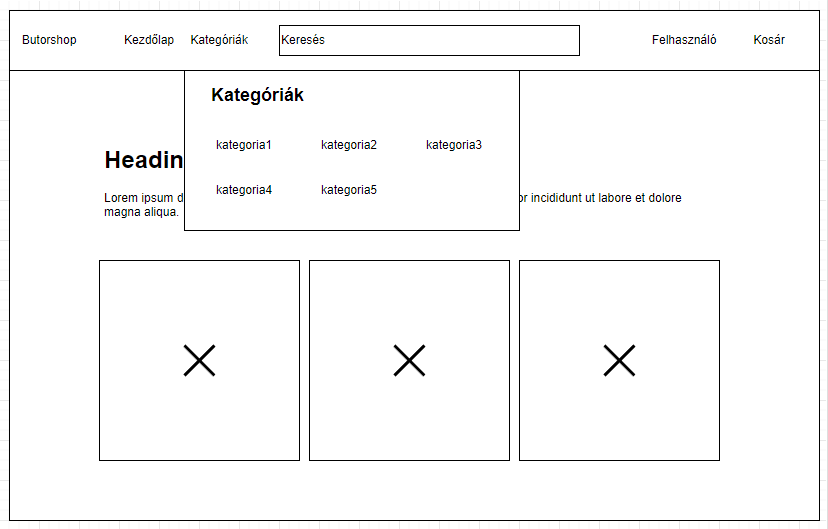
* Rendelések felvitele: a rendeléseket rögzítheti a vásárló.
* Rendelések törlése: a téves törléseket törölheti a vásárló.
* Rendelések állapota: a vásárló megtekintheti a rendelései állapotát.

Drótvázak

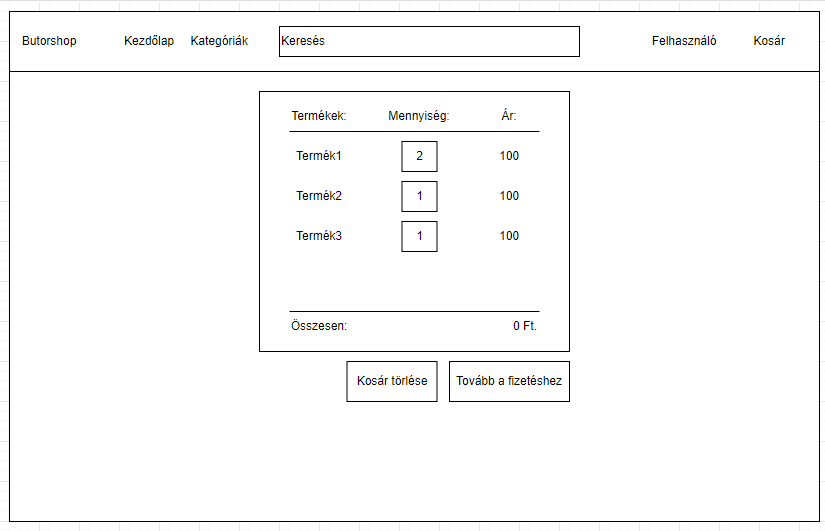
Az oldal kIalakítása során igyekeztünk a drótvázakhoz vizuálisan közeli végeredményt létrehozni, azonban fejlesztés során néhány helyen módosítottunk az átláthatóság, logikus, könnyű kezelhetőség érdekében.

Főoldal:

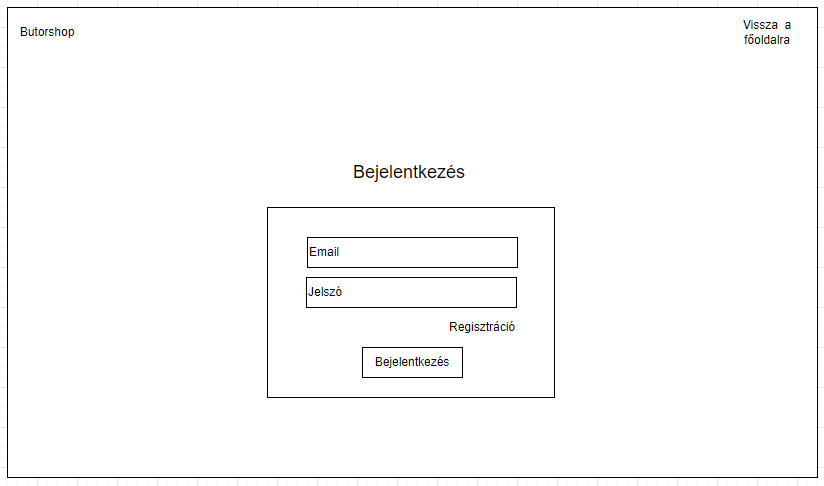


Kategóriák:

Kosár:



Bejelentkezés:



# Felhasználói dokumentáció

## A program célja, lényegesebb funkciói

A program egy online webáruház ami egy áruház legtöbb lehetséges igényét igyekszik kielégíteni. A webalkalmazás használata során a vendég (profillal nem rendelkező) státuszú felhasználók regisztrálhatnak fiókokat, amelyek segítségével hozzáadhatnak tételeket a bevásárlókosarukhoz, valamint menthetik adataikat a későbbi egyszerűbb kezelés érdekében. A vásárlók emellett tudnak kategóriák szerint böngészni a bútorok között valmaint kersni a termékek kínálatában, hogy könnyen megtalálhassák a kívánt bútorelemeket.

A webalkalmazás ezenkívűl tartalmaz egy admin felületet is amivel az áruház dolgozói intézhetik az áruházzal kapcsolatos ügyeket.

Termékek kezelése: Az adminisztrátorok nyomon követhetik a termékek állapotát, frissíthetik azok adatait és eltávolíthatják őket a katalógusból szükség esetén. Ezen keresztül lehetőség van a beszerzés jegyzésére és a készlet növelésére.

Új termék hozzáadása: Az adminisztrátorok új bútorelemeket adhatnak hozzá a katalógushoz, megadva a megfelelő információkat, mint például a név, leírás és ár.

Termék szerkesztése: Az adminisztrátorok frissíthetik a meglévő termékek részleteit, beleértve a nevet, leírást és árat is, hogy azok mindig naprakészek legyenek.

Termék eltávolítása: A rendszergazdák jogosultságot kapnak a termékkatalógusból való eltávolításra, amennyiben szükséges. Ez lehetővé teszi a katalógus karbantartását és frissítését a felesleges vagy elavult termékek eltávolításával.

## A program elérése

Az alkalmazás webes környezetben érhető el egy böngészőprogram segítségével. Használata a legtöbb modern böngészőprogramot futtatni képes digitális eszközről lehetséges. Az admin felület több táblázatot valamint hosszabb szövegeket tartalmaz, így annak használata kissebb szélességű eszközökről (pl.: okostelefon) nem javasolt.

## A program fő funkcióinak leírása

### Menüpontok az oldalon

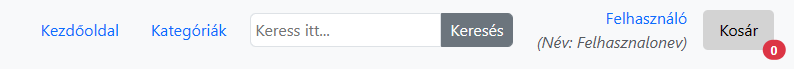
Az oldalon található menüpontok az alábbi képen láthatóak:

Vendég verzió:



* Kezdőoldal: Visszanavigálhat a kezdőoldalra.
* Kategóriák: Kategóriák szerint böngészhet a termékek között.
* Keresés: Kereshet a termékek kínálatában.
* Bejelentkezés: Bejelentkezhet, valamint regisztrálhat az oldalra.

Vásárló verzió:



* Felhasználó: Megtekintheti rendeléseit valamint módosíthatja adatait a felhasználó.
* Kosár: Megtekintheti, módosíthatja, törölheti vagy véglegesítheti kosara tartalmát a felhasználó.

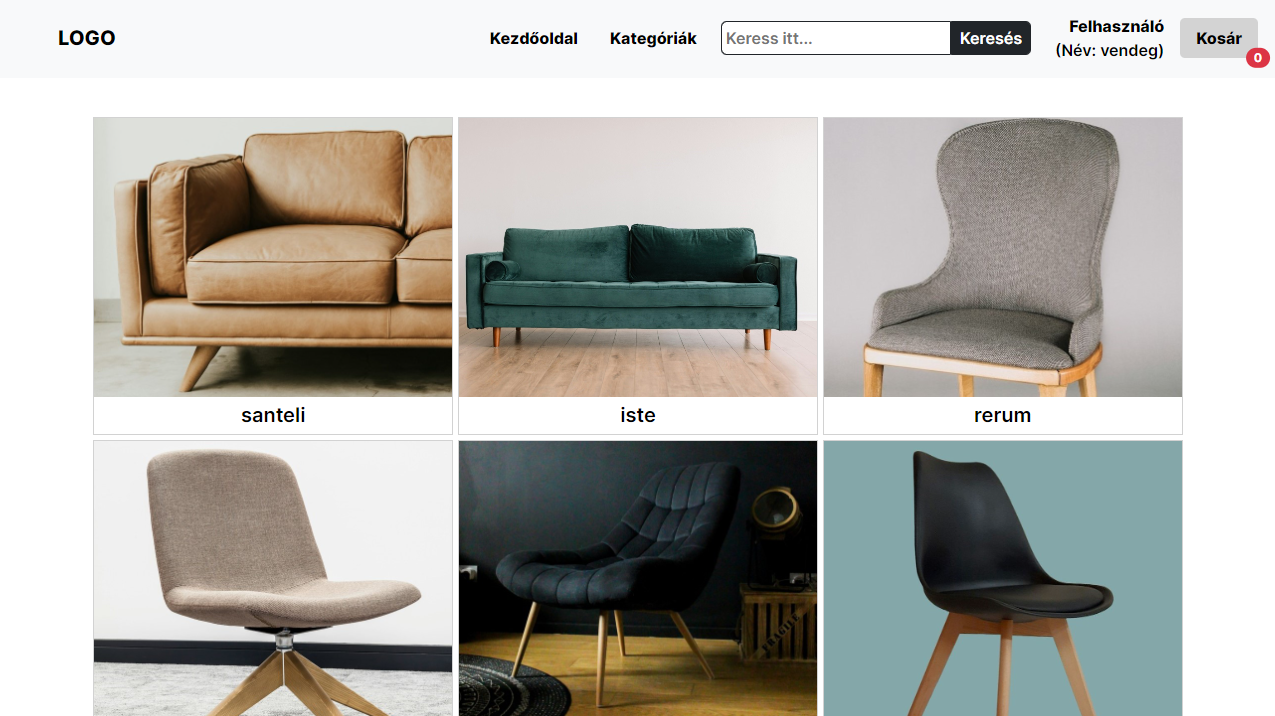
Admin verzió:



* Admin: Az admin jogokkal rendelkező felhasználók itt módosíthatják a termékek illetve modellek adatait, hozhatnak létre újakat Emellett itt intézhetik a felhsználók, rendelések és beszerzés kezelését is.

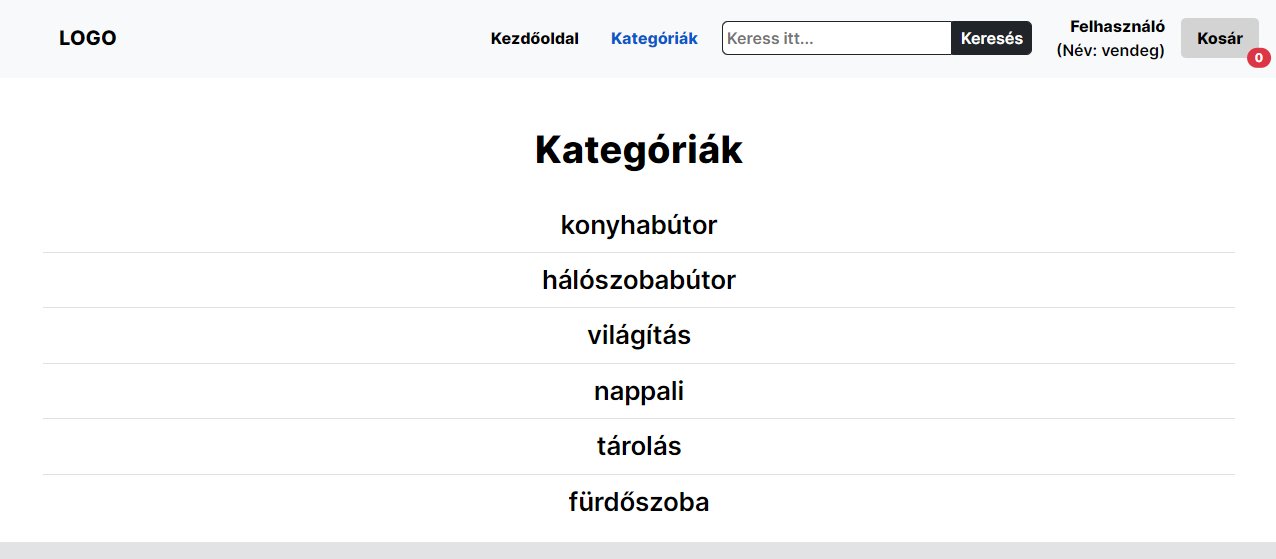
### Kezdőoldal

A felhasználók elsősorban itt tudnak böngészni a termékek között. Egy adott termékre kattintva megjelennek annak adatai, variációi.



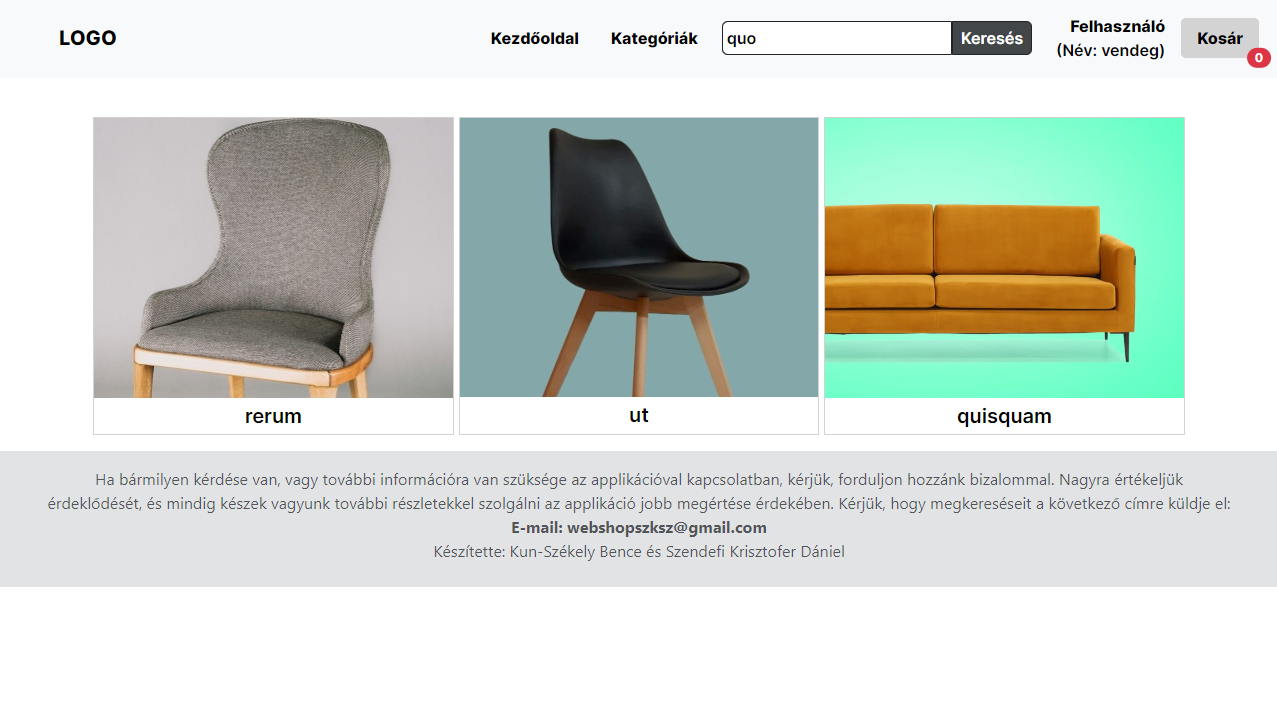
### Kategóriák

Ezen az oldalon a felhasználók kategóriákra bontva böngészhetnek a termékek között.



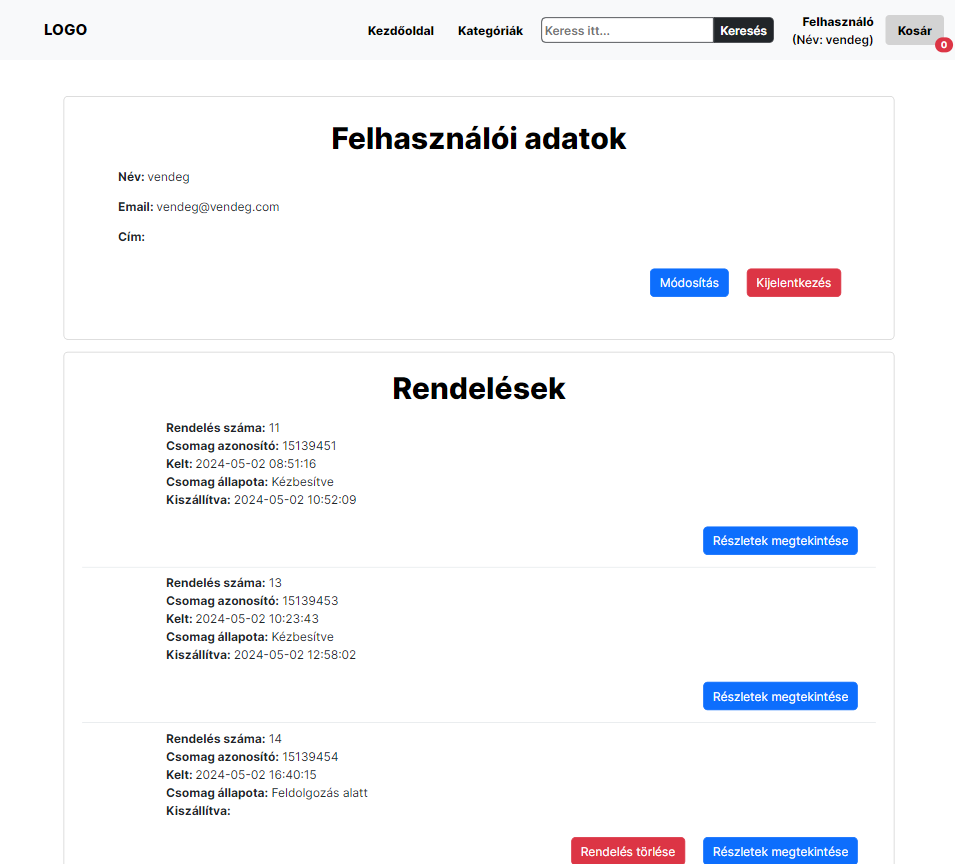
### Keresés

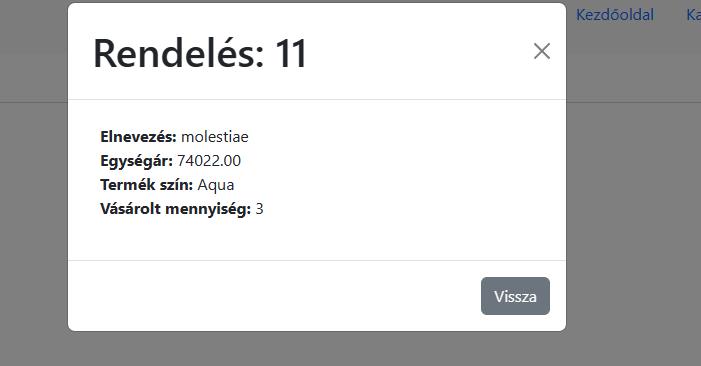
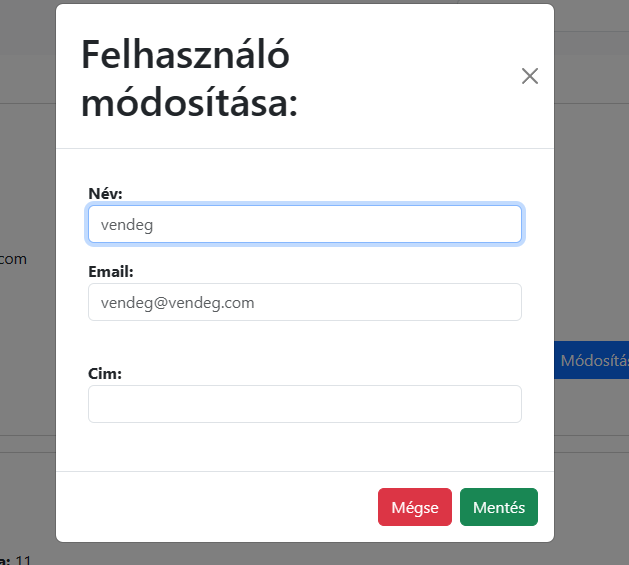
Itt a felhasználók a keresőmezőbe írva majd a keresés gombra kattintva kereshetnek a készleten lévő termékek között.



### Felhasználó (csak bejelentkezet felhasználóknál)

* Ezen az olalon a felhasználók módosíthatják valamint kiegészíthetik adataikat a módosítás gombra kattintva.
* A kijelentkezés gombra kattintva a felhasználók elhagyhatják profiljukat vendégként visszatérve az oldalra.
* A rendelések részletek megtekintése gombjára kattintva megtekinthetőek annak adatai
* A rendelés törlése gombra kattintva a rendelés törlődik a rendszerből

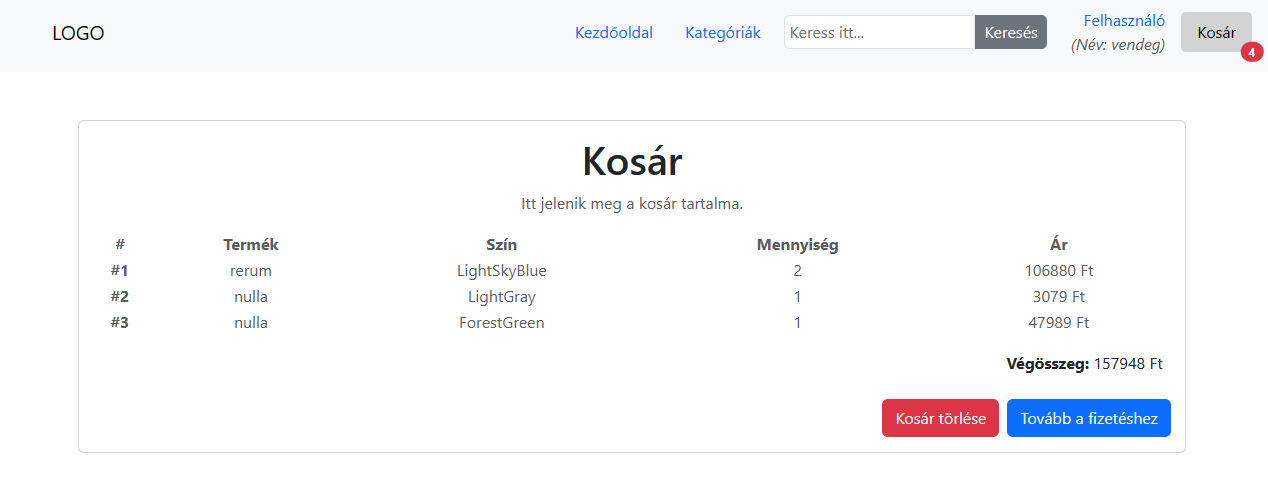


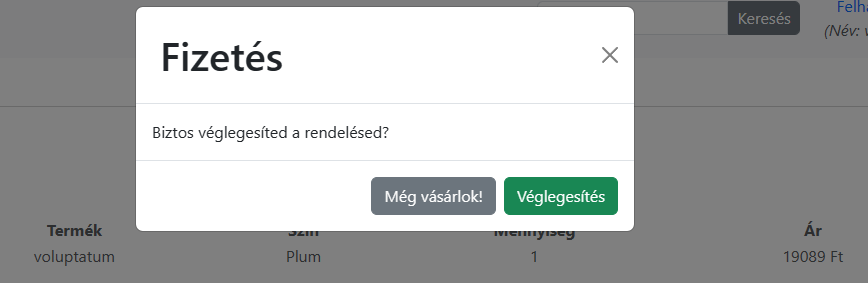


### Kosár

Itt tekinthetőek meg a kosárba helyezett termékek.

* A tovább a fizetésre gombra kattintva véglegesíthetőek a rendelések. Ekkor felugrik egy ablak ami megkérdezi hogy biztosan véglegesíteni szeretnénk e a rendelést. A véglegesítés gombra kattintva ekkor mentésre kerül a rendelés (ezzel a felhasználó -> rendelések pontjában megtekinthető lessz a leadott rendelés).
* A kosár törlése gombra kattintva a kosár tartalma törlődik.

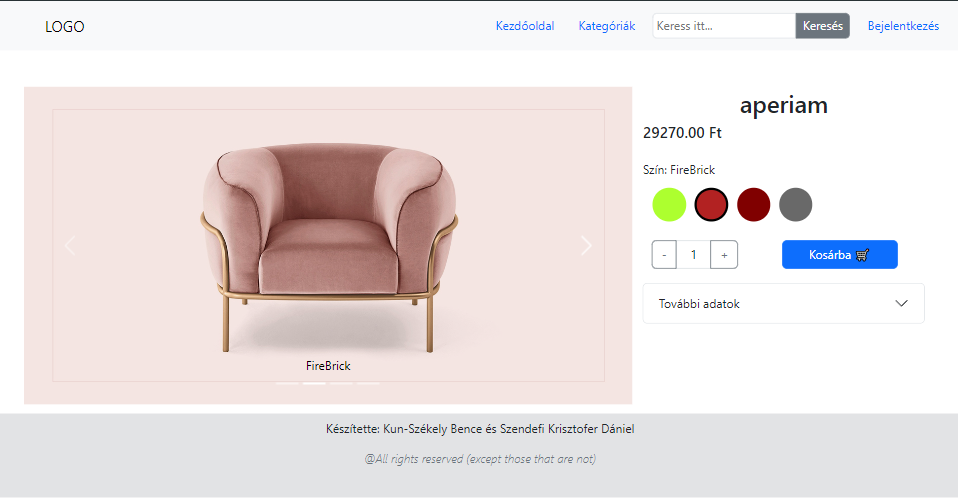




### Termékek adatainak megtekintése valamint válogatás

Egy modellre kattintva a főoldalon, vagy a kategóriák oldalon, a felhasználó tovább lép az adott modell megjelenítőjére. Itt a következő lehetőségei vannak:

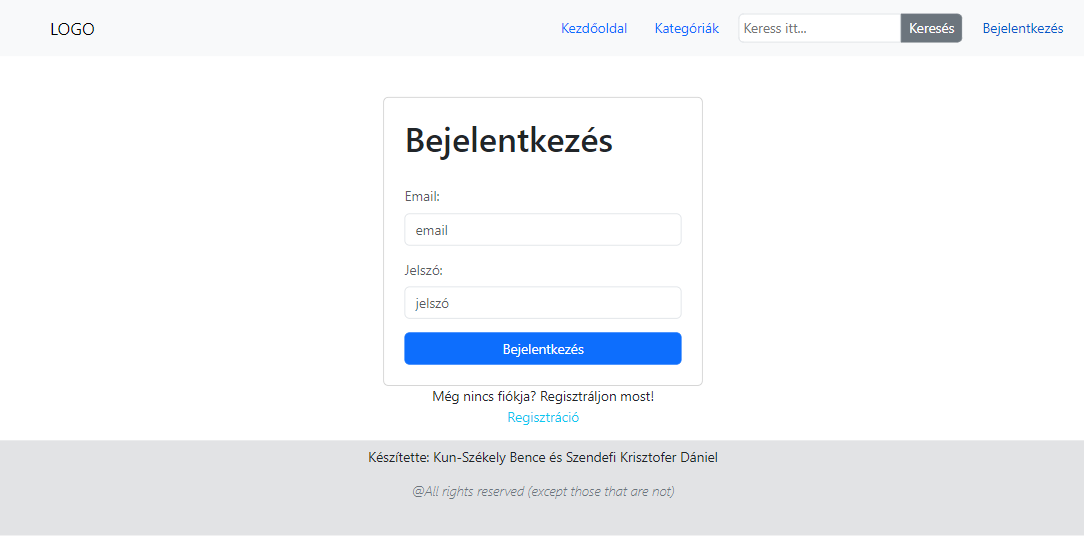
* Megtekintheti a különböző termékek képeit.
* A színkörökre kattintva megtekintheti a kiválasztott termék árát, részletes leírását, és az részletes adatait.
* Növelheti, vagy csökkentheti a kívánt mennyiséget.
* Ha a kiválasztott terméket meg szeretné vásárolni, ez a “Kosárba” gomb megnyomásával teheti.
* Fontos, hogy ezt csak azután teheti, ha előzetesen regisztrált és bejelentkezett, különben a rendszer nem engedi a kosárba helyezést.



## Bejelentkezés és regisztráció

### Bejelentkezés:

A menüben a bejelentkezés gombra kattintva átirányítódik az oldal a bejelentkezési felületre. Itt meg kell adnia az előzetesen regisztrált felhasználói fiókjának email címét és jelszavát. (Ha még ezt nem tette meg az oldal alján található “Regisztráció” gomb megnyomásával kezdheti ezt meg.) Sikeres bejelentkezés után a program átirányít a kezdőoldalra.

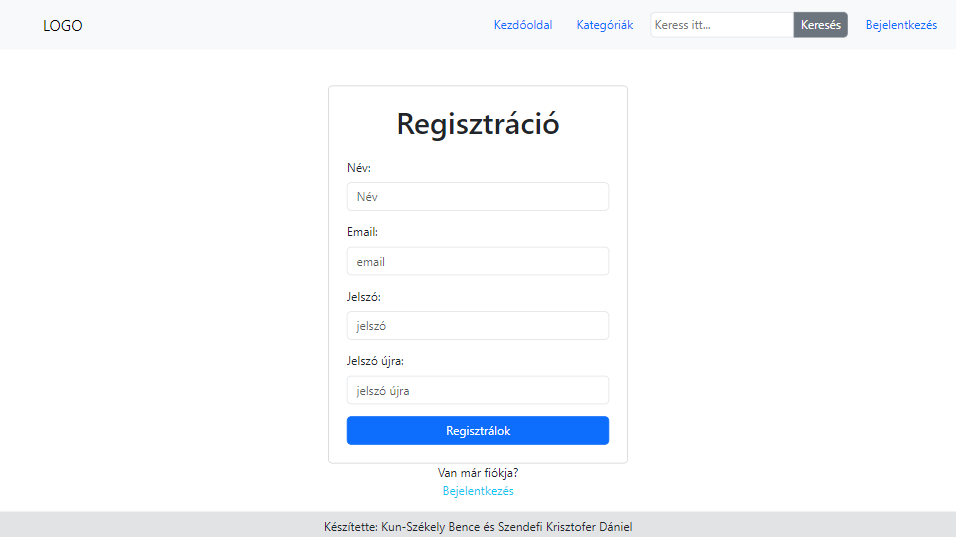
Regisztráció:

A regisztrációs felületet a bejelentkezés oldal alján található “Regisztráció” gombban kezdheti meg.

A regisztráció menete:

1. Adja meg a felületen használni kívánt nevét.
2. Adja meg az email címét. Fontos, hogy egy email címmel csak egyszer lehet regisztálni.
3. Adjon meg egy egyedi és erős jelszót. Fontos, hogy a két jelszó egyezzen
4. Ezután kattintson a “Regisztráció” gombra.

Sikeres regisztráció után a program automatikusan bejelentkezik és átirányít a kezdőoldalra.



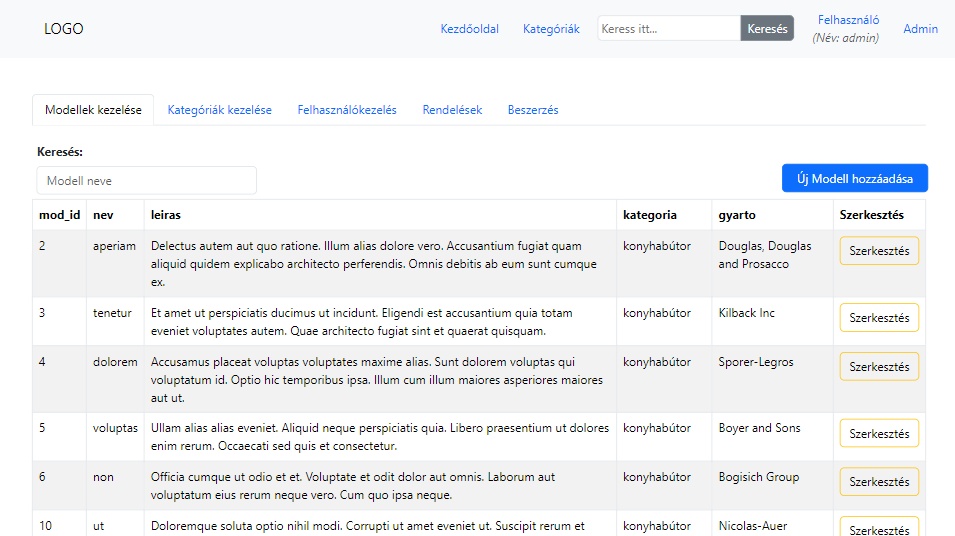
## Admin felületek és elérésük

Az admin hozzáférési szintű felhasználók a bejelentkezés után a menüben az admin gombra kattintva elérik az alábbi adminisztrációs funkciókat:

### Modellek kezelése:

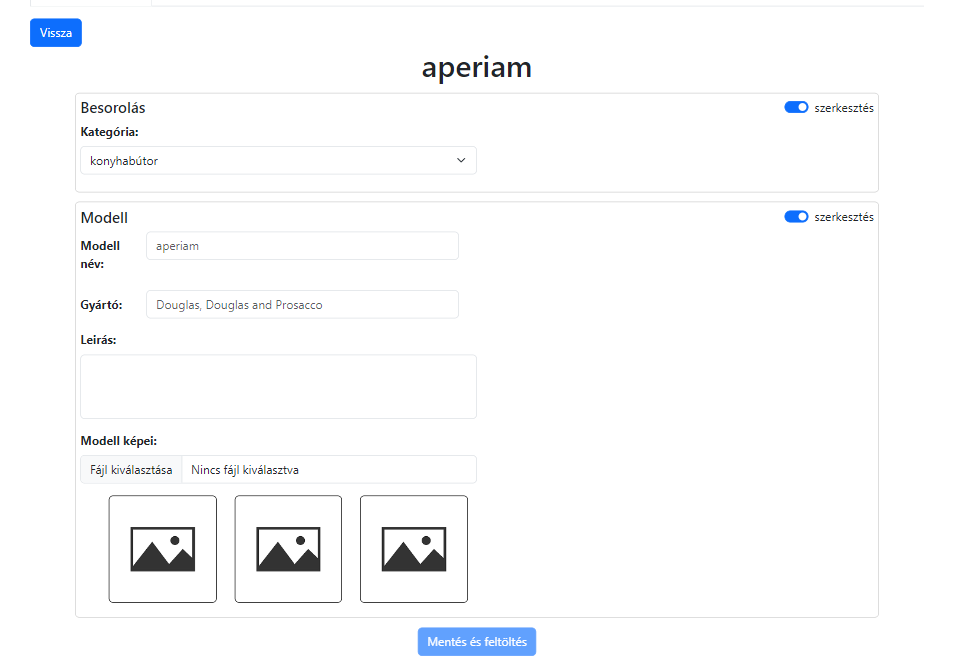
A “modell kezelése“ gombra kattintva a megjelenő felületen megjelenik a modellek listája és a következő funkciók:

* Keresés: A mezőbe írva leszűkítheti a modellek listáját a modellek nevei alapján.
* Szerkesztés: Módosíthat a meglévő modell adatain, vagy új terméket hozhat létre hozzá.
* Új Modell hozzáadása: Ha a keresett modell még nem létezik újat lehet létrehozni.

A táblázat modelljei mellett található “Szerkesztés” gombra kattintva módosíthatja az adott modellt.

A módosítás menete:

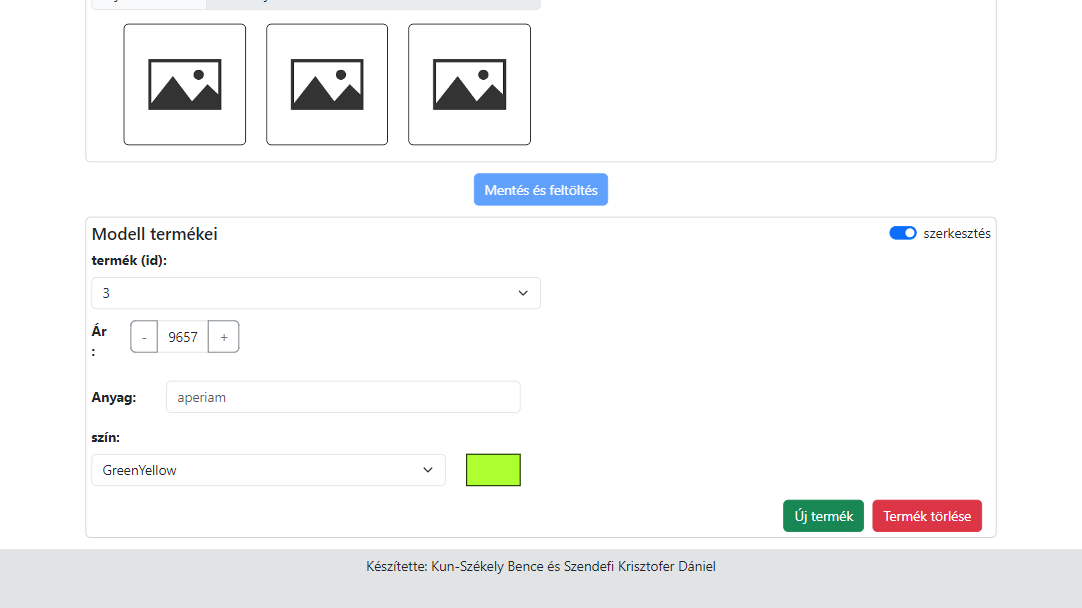
1. Válassza ki az új kategóriát.
2. Adja meg az új nevet, gyártót és leírást



Szükség esetén itt tud a modellhez új termékeket adni, ennek menete a következő:

1. Görgessen az oldal aljára, ahol megtalálja a “Modell termékei” felületet.
2. Válassza ki a termék árát, anyagát és színét.
3. Kattintson az “Új termék” gombra.

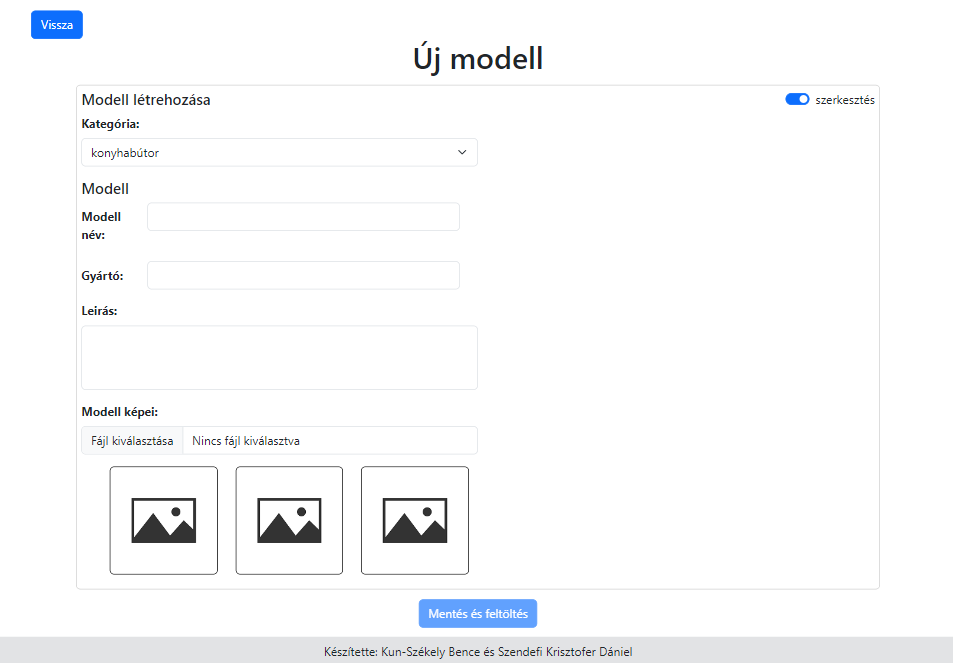
Sikeres feltöltés esetén a rendszer visszajelez.



Az ”Új Modell hozzáadása” gombra kattintva megnyílik egy új ív. A szerkesztés kapcsoló felkapcsolásával elkezdheti felvinni az új modell adatait. Az adatfelvitel lépései a következők:

1. Válassza ki a legördülő mezőből a megfelelő kategóriát.
2. Adja meg a modell nevét, gyártóját, és reészletes leírását
3. Adjon hozzá képeket a modellről. (Nem kötelező!)
4. Kattintson a “Mentés és feltöltés” gombra.

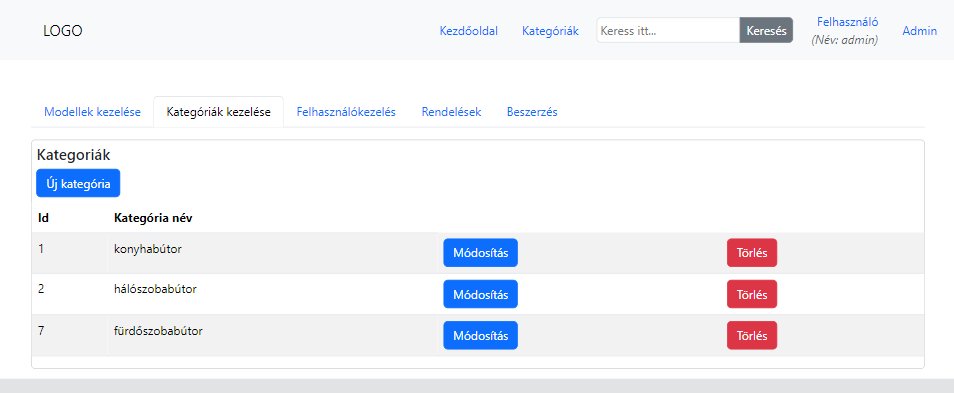
Sikeres feltöltéskor a program visszajelez.

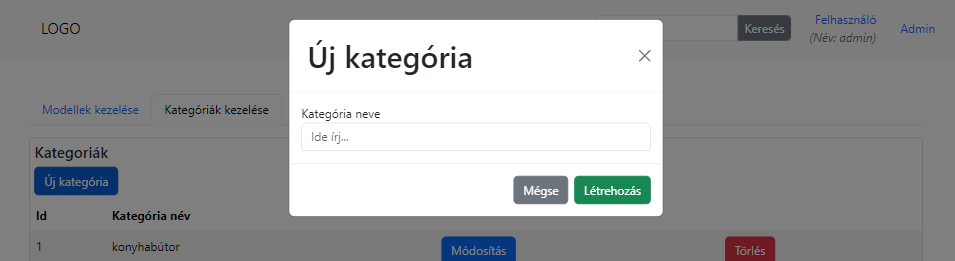


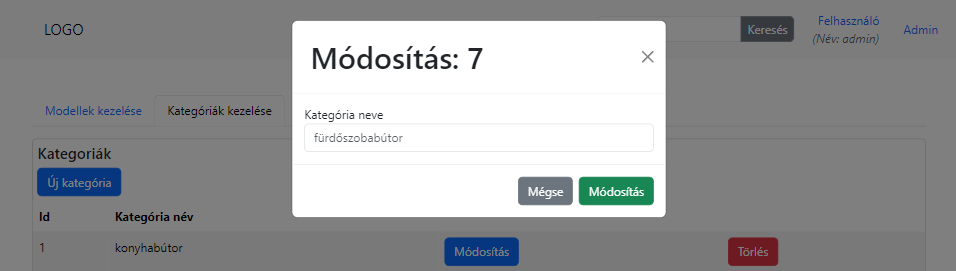
### Kategóriák kezelése:

A “Kategóriák kezelése” fülre kattintva megjelenik a kategóriák listája és a következő funkciók:

* Új kategória: Itt új kategória neveket lehet megadni.
* Módosítás: Adott kategória nevének módosítása.
* Törlés: Kategória törlése.





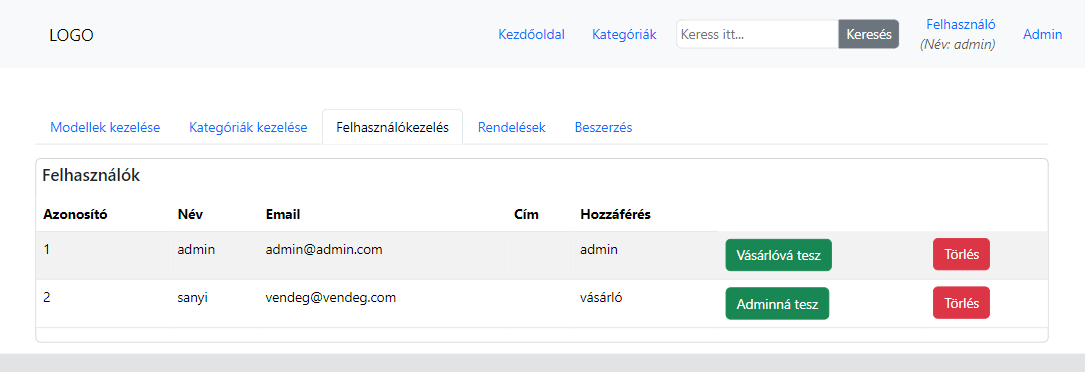


### Felhasználókezelés:

A “Felhasználókezelés” fülre kattintva megjelenik a regisztrált felhasználók listája.

Itt a következő funkciók elérhetőek:

* Admin jogosultságok kiosztása és elvétele
* Regisztrált felhasználók törlése

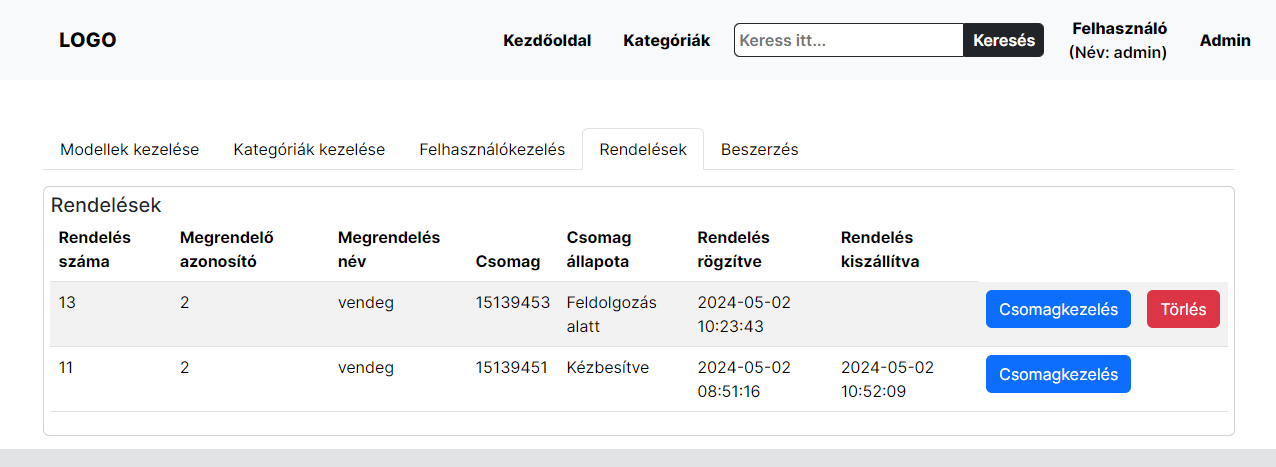


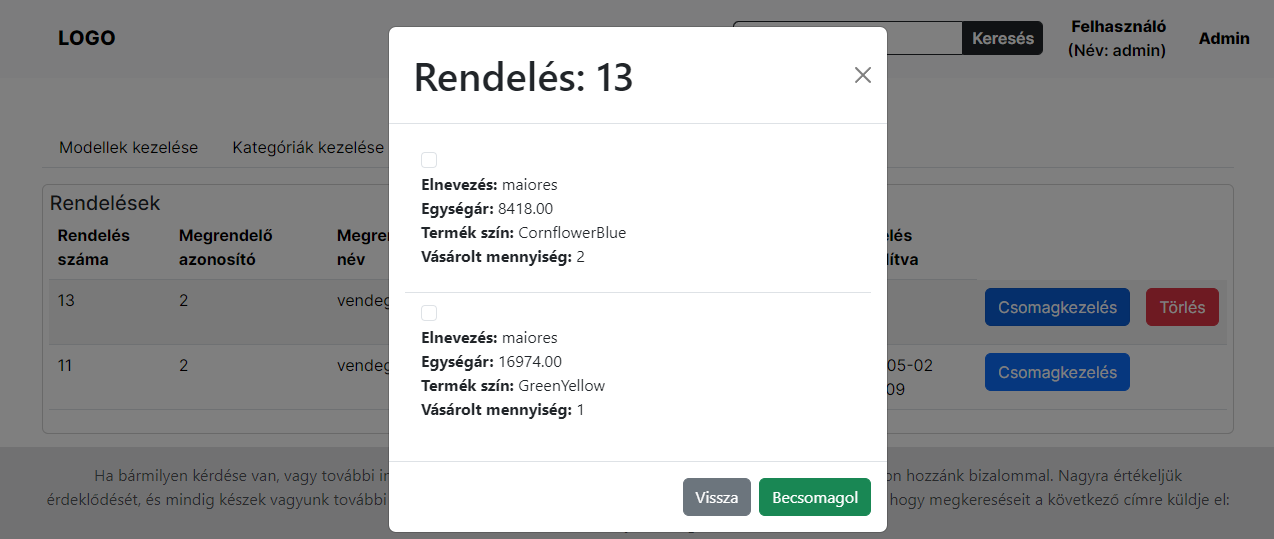
### Rendelések:

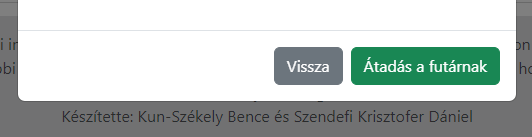
A “Rendelések” fülre kattintva megjelennek a felhasználók rendelései dátumra rendezve.

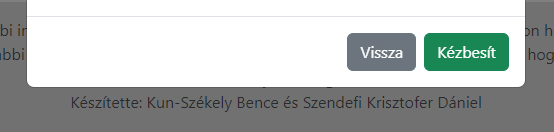
Az elérhető funkiók:

* A “Csomagkezelés” gombra kattintva megnyíló felületen megtekintheti a rendelés termékeit.
* A “Törlés” gombra kattintva törölheti a felvett, de hibásan leadott, vagy át nem vett csomagokat. Fontos, hogy a már kiszállított termékeket törölni nem lehet.

A ”Csomagkezelés” gombra kattintva megtekintheti a csomagba szánt termékek listáját. A termékek csomagolásának segítése érdekében minden termék felett található egy kipipálható gomb. Ezzel egyszerűen lehet követni a már becsomagolt termékeket. Ha minden terméket becsomagolt, kattintson a ”Becsomagol” gombra, és a rendszer automatikusan frissíti a csomag állapotát.

Ha a rendelés be van csomagolva, akkor átadható a futárnak. Ezt a programban jelezni úgy lehet, ha a becsomagolt terméknél rákattint a ”Csomagkezelés” gombra és rákattint az ”Átadás a futárnak” gombra.

Ha a megbízott futár kézbesítette a csomagot, akkor az ugyanitt található ”Kézbesít” gombra kattintva a program megjelöli a rendelést kézbesítettként. Ekkor rögzíti a kézbesítés dátumát és pontos idejét.



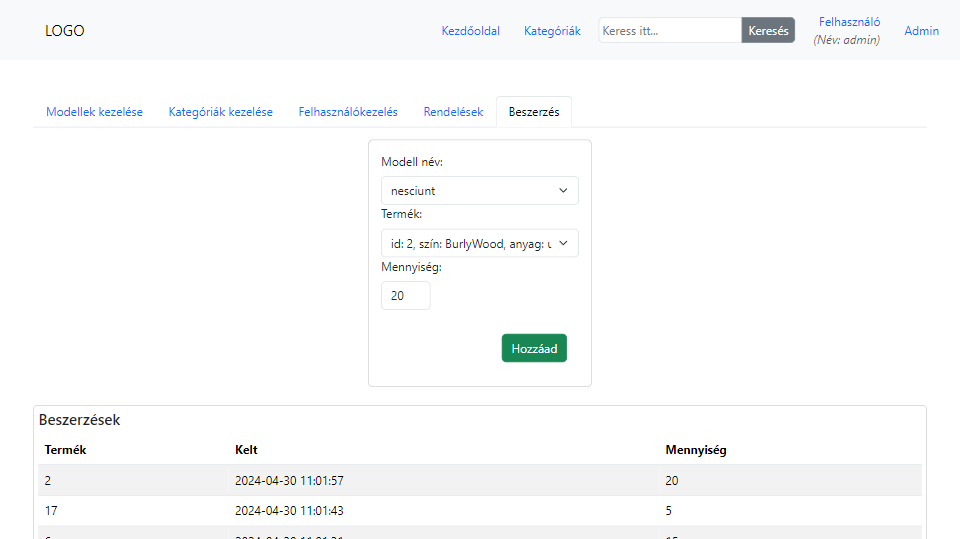
### Beszerzés:

A “Beszerzés” fülre mejgelenik a termék beszerzésének telülete. Itt tudja növelni a termék készletét az adatbázisban, valamint a korábbi beszerzéseket időrendben megtekinteni.

A beszerzés menete a következő:

1. Válassza ki a termékhez tartozó modell nevét.
2. Válassza ki a beszerzett terméket. Ebben segíthet a termék azonosító és a termék színe is, vagy akár anyaga.
3. Válassza ki a megdelelő mennyiséget.
4. Kattinson a “Hozzáad gombra”

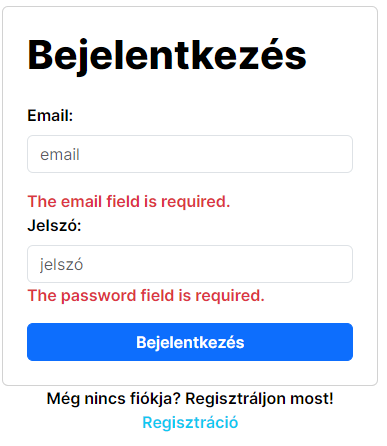
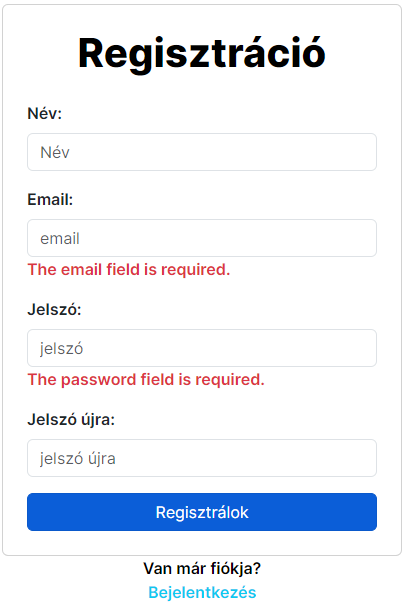
Sikeres rögzítés esetén a lista tetején megjelenik az újonnan beszerzett termék, azonban hiba esetén ezt a felület pirossal jelzi.



## Hibajelzések magyarázata

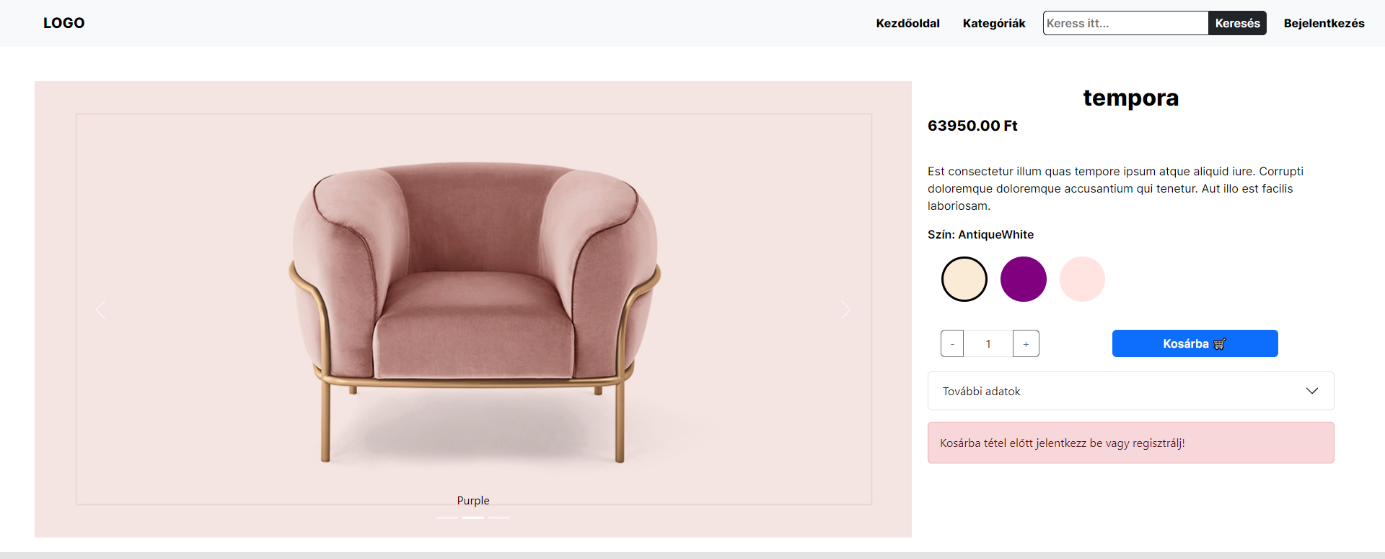
### Bejelentkezés és regisztrációs hibák:

* Az email mezőt nem lehet üresen hagyni.
* A jelszó mezőt nem lehet üresen hagyni.
* A regisztráció során “Jelszó” és “Jelszó újra” mezőknek egyezniük kell.



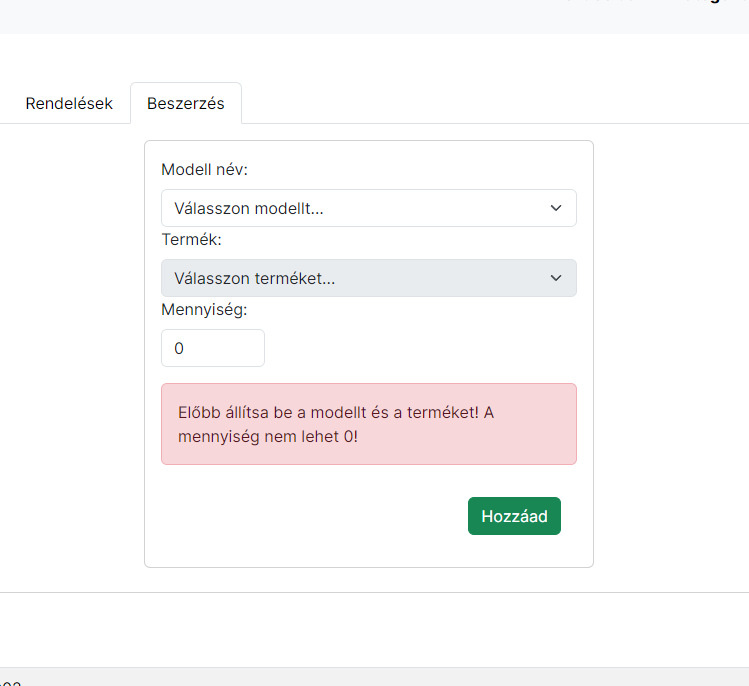
### Kosár hiba:

A kosárba terméket helyezni csupán bejelentkezés, vagy regisztráció után lehet.



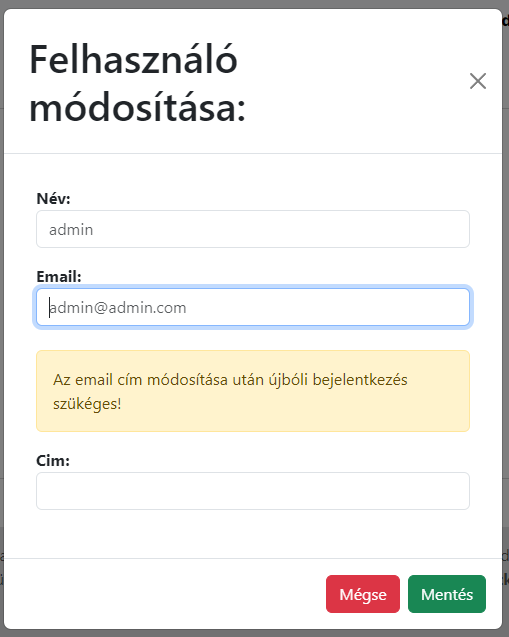
### Beszerzés hiba (admin felület):

A modell nevet, a terméket és a mennyiséget ki kell választani a hozzáadás megnyomása előtt, különben nem engedi a program az adatrögzítést.



### Felhasználó módosítás figyelmeztetés:

Ha a felhasználó az email címét kívánja módosítani, akkor a módosítás után újból be kell jelentkeznie.



## Információkérés

Ha bármilyen kérdése van, vagy további információra van szüksége e dokumentum tartalmával kapcsolatban, kérjük, forduljon hozzánk bizalommal. Nagyra értékeljük érdeklődését, és mindig készek vagyunk további részletekkel szolgálni az applikáció jobb megértése érdekében.

Kérjük, hogy megkereséseit a következő címre küldje el:

E-mail: webshopszksz@gmail.com

# Fejlesztői dokumentáció

## Fejlesztői környezet

Fejlesztői környezetnek a A Visual Studio Code (VS Code) integrált fejlesztői környezetet használtuk. A VS Code átfogó és felhasználóbarát környezetet biztosít, amely támogatja mind a frontend, mind a backend fejlesztést, így ideális az olyan teljes körű projektekhez, mint egy RESTful React alkalmazás. Az alábbi előnyök mind a VS Code mellett szólnak:

* Nagyszámú bővítmény: a bővítmények széles választéka növeli a produktivitást és egyszerűsíti a fejlesztési folyamatot
* Beépített Git-támogatás: A VS Code beépített Git-támogatással rendelkezik, ami leegyszerűsíti a verziókövetést és a kollaborációt.
* Hibakeresési eszközök: Az IDE hatékony hibakeresési eszközöket kínál, amelyek könnyen konfigurálhatóak
* Teljesítmény: A VS Code más IDE-khez képest teljesítményéről és alacsony erőforrásigényéről ismert, ami előnyös lehet, ha összetett alkalmazásokon dolgozunk.
* Integrált terminál: Az IDE-be integrált terminál lehetővé teszi, hogy a kódszerkesztő elhagyása nélkül hajthassunk végre parancsokat.

## Adatbázis

Az adatbázis célja a webáruház adatainak részletes eltárolása. Eltárolja a vásárlók adatait, és rendeléseit. Számon tartja a modellek termék variációit, mennyiségét és részletes adatait, a rendelések tételeit, a futárnak szánt csomagokat, illetve a termékbeszerzések idejét és mennyiségét. A felhasználóknak két hozzáférési rétege van: vásárló, admin.

### Adatbázis-tervezés:

A tervezés során azadatbázis tervet töbször is módosítottuk, a tervezés jobb átláthatóságának érdekében, ezeket áthúzással jelöltük.

**Felhasznalo:** A vásárlók és adminok adatainak eltárolásához szükséges tábla.

Felhasznalo (**azon**, nev, jelszo, hozzaferes, email, cim)

Hozzáférés lehet: 0 - vásárló, 1 – admin.

**Kategória:** A modellek kategorizálása pl.: Konyhabútor, kerti bútor.

Kategoria (**kat\_id**, kategoria\_nev)

**Modell:** A bútorok modelljei, ezek lesznek megjelenítve a felületen. Modellt nem lehet vásárolni, csak a modellből készült termékeket.

Modell (**mod\_id**, nev, leiras, *kategoria,* gyarto, kep)

**Tulajdonsag:** A különböző kategóriáknak megadható tulajdonságai.

Tulajdonsag(azonosito, nev, *kategoria*)

~~--Modell\_tulajdonsag (~~***~~modell,~~* ~~tulajdonsag~~**~~)~~

A modell tulajdonságra nincs szükség, mert a tulajdonság tábla már kozzáköti a kategóriához, és a modell-kategória kapcsolatból eléri a termék is a tulajdonságokat.

**Termek:** A létező modellek különböző variácói. Ezt már meg lehet vásárolni, ezért van ára és készlete is.

Termek (**ter\_id**, ar, *modell,* anyag, szín, keszlet)

**Termek\_tulajdonsag:** A különböző termékek tulajdonságait itt lehet eltárolni. Ide csak a tulajdonság táblában előtőzőleg megadott tulajdonságoknak lehet értéket adni.

Termek\_tulajdonsag (***termek*, *tulajdonsag***, ertek)

~~--Keszlet (~~**~~keszlet\_azon~~**~~,~~ *~~termek~~*~~, mennyiseg)~~

A készlet táblára már nincs szükség mivel a készletet a termékeknél kezeljük.

**Beszerzés:** Itt a beszerzett termékek mennyiségét és beszerzési idejét tároljuk. Ha ide felviszünk egy adatot a termék tábla helyes készletét növelni kell.

Beszerzes (***termek*, kelt**, mennyiseg)

**Rendeles:** Amikor a vendég lead egy rendelést, itt kerül bejegyzésre. A csomag csak akkor kerül kitöltésre, ha a csomag létrejött. A kiszállítva dátum csak akkor lesz kitöltve, ha a csomag kézbesítve lett.

Rendeles (**rend\_szam**, *felhasznalo*, kelt)

**Rend\_tetel:** A rendelésben szereplő termékek és mennyiségét itt tároljuk. A csomagolva egy boolean ami jelzi, hogy a rendelés tételenként csomagolva lettek.

Rend\_tetel (***rendeles*, *termek***, mennyiseg, csomagolva)

**Csomag:** A futár cégnek átadott csomag és tárolt állapota. Ha a rend\_tetel-ben minden a csomagban szereplő termék csomagolva sora igaz, akkor lehet csak az állapot csomagolva.

Az állapot lehet: 0 – feldolgozás alatt,1 – csomagolva , 2 - átadva a futárnak, 3 - kézbesítve.

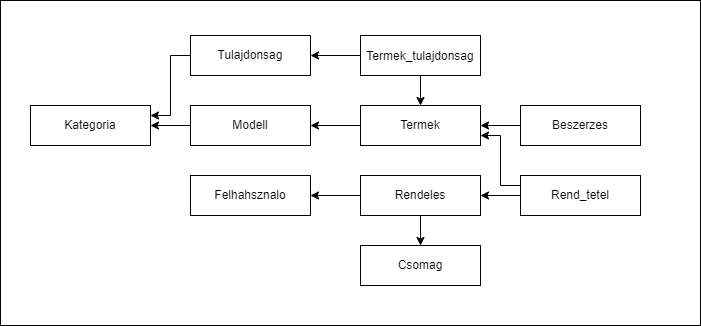
Csomag (**csom\_azon**, allapot)

Rendeles (+ *csomag*, kiszallitva)

A rendelés táblához hozzáadjuk a csomag és a kiszállítva oszlopokat.

A kiszállítva a kézbesítés dátum-ideje.

### Egyed-kapcsolat diagramm:



### Lekérdezések:

* Adott termék tulajdonságai
* Összes kategória összes adatai
* Összes kategória nevei
* Összes modell adatai
* Összes modell adatai és hozzá tartozó termékek, és hozzá tartozó kategória
* Adott modell termékei
* Adott szöveg keresése modellek leírásában
* Adott kategória modelljei
* Adott felhasználó rendelései, a csomaggal összekötve
* Adott rendelés termékei
* Összes rendelés adatai, csomag adataival
* Adott felhasználó adatai
* Összes felhasználó adatai
* Összes beszerzés adatai

### Triggerek:

* Ha a rendelés tétel táblájába felvisznek egy új sort, akkor a rendelés tétel mennyiségét levonjuk a megfelelő termék készlet oszlopából.
* Ha a rendelés tétel táblájából törölnek egy sort, akkor a törölt rendelés tétel mennyiségét hozzáadjuk a megfelelő termék készletéhez.
* Ha a csomag tábla állapot oszlopa 3-asra változik (kiszállítva), akkor a rendelés táblában található kiszállítva oszlop rögzíti a pontos dátumot és időt.

### Táblák szerkezete a táblaszintű megszorításokkal:

**Felhasznalo:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mező | Addattípus | szerep | korlátozás |
| azon | int | generált kulcs |  |
| nev | char(30) | kötelező |  |
| jelszo | char(30) | kötelező | Min. 8 karakter |
| hozzáférés | boolean | alapértelmezett 0 |  |
| email | char(255) | kötelező, egyedi |  |
| cim | char(255) |  |  |

**Kategória:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mező | Addattípus | szerep | korlátozás |
| Kat\_id | int | generált kulcs |  |
| Kat\_nev | char(30) | kötelező |  |

**Modell:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mező | Addattípus | szerep | korlátozás |
| Mod\_id | int | generált kulcs |  |
| nev | char(30) | kötelező |  |
| leiras | char(100) |  |  |
| kategoria | boolean | Külső kulcs (kategória) |  |
| gyarto | char(30) | kötelező |  |
| kep | char(100) |  |  |

**Tulajdonsag:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mező | Addattípus | szerep | korlátozás |
| azonosíto | int | generált kulcs |  |
| nev | char(30) | Kötelező, egyedi |  |
| kategoria | int | Külső kulcs (kategória) |  |

**Termek:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mező | Addattípus | szerep | korlátozás |
| termek\_id | int | generált kulcs |  |
| ar | int | kötelező | Ar >= 0 |
| modell | int | Külső kulcs (modell) |  |
| anyag | char(30) |  |  |
| szin | char(30) |  |  |
| keszlet | int | kötelező | Keszlet >= 0 |

**Termek\_tulajdonsag:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mező | Addattípus | szerep | korlátozás |
| termek | int | Kulcs része, külső kulcs(termék) |  |
| tulajdonsag | int | Kulcs része, külső kulcs (tulajdonság) |  |
| ertek | char(30) | kötelező |  |

**Beszerzés:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mező | Addattípus | szerep | korlátozás |
| termék | int | Kulcs része, külső kulcs(termék) |  |
| kelt | datetime | Kulcs része, kötelező | rendszeridő |
| mennyiség | int | kötelező | Mennyiség >= 0 |

**Rendelés:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mező | Addattípus | szerep | korlátozás |
| rendszam | int | generált kulcs |  |
| felhasznalo | int | Külső kulcs (felhasznalo) | Ar >= 0 |
| csomag | int | Külső kulcs (csomag) |  |
| kelt | datetime | kötelező | rendszeridő |
| kiszállitva | datetime |  |  |

**Rend\_tetel:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mező | Addattípus | szerep | korlátozás |
| rendeles | int | Kulcs része, Külcső kulcs(rendeles) |  |
| termek | int | Külső kulcs (termek) |  |
| mennyiseg | int | kötelező | Mennyiseg >= 0 |
| csomagolva | boolean | Alapételmezett 0 |  |

**Csomag:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mező | Addattípus | szerep | korlátozás |
| csom\_azon | int | generált kulcs |  |
| allapot | int | kötelező | Ar >= 0 |

## Backend

A Backend felépítéséhez Laravelt használtunk. A Laravel egy ingyenes és nyílt forráskódú PHP -alapú webes keretrendszer webalkalmazások készítésére.[1] Model-view-controller, vagy modell-nézet-vezérlő (MVC) programtervezési mintát használ ennek megvalósításához.[2]

* Model (magyarul modell): Az adatokat kezelő, vagyis tulajdonképpen az üzleti logikát megvalósító réteg. Ez felel az adatok tárolásáért, visszaolvasásáért. Itt foglalnak helyet azok a függvények is, amik műveleteket végeznek az adatokon. Része az adatbázis is.
* View (magyarul nézet): A felhasználói felület megjelenítéséért, a felhasználó különféle nyűgjeinek a Vezérlő felé továbbításáért felelős réteg. Itt jelennek meg a felhasználó számára a vezérlőelemek, a felhasználónak szánt adatok megfelelő formában való megjelenítése is itt történik.
* Controller (magyarul vezérlő): Ez a réteg a vezérlőelemek eseményeinek alapján meghívja a modell megfelelő függvényeit, illetve, ha a megjelenítésben érintett adatok változnak, akkor erről értesíti a Nézetet. [3]

### Modellek és controllerek

Modellek:

Ezek a fájlok jelölik a controllerek által kezelhető adatokat. A mi projektünkben az alábbi modellek vannak jelen:

* Beszerzés
* Csomag
* Kategoria
* Modell
* Rend\_tetel
* Rendeles
* Termek\_tulajdonsag
* Termek
* Tulajdonsag
* User

Controllerek:

Ezek a fájlok felelősek az adott HTTP-kérések kezeléséért.

Beszerzés controller: Ez végzi a beszerzések lekérését, illetve az új beszerzések rögzítését.

Csomag controller: Ez felelős a csomagok állapotának megváltoztatásáért.

Modell controller: . Olyan műveletekhez tartalmaz metódusokat, mint a modellek létrehozása, olvasása, frissítése és törlése.

Termék controller: Ez tartalmazza az olyan metódusokat, mint a termékek létrehozása, olvasása, frissítése és törlése.

**Kategória controller:** Ebben található a kategória táblában szereplő adatok lekéréséhez, rögzítéséhez, és módósításához szükséges kódok.

Rendelés controller: Itt található a rendelések lekérése, rögzítése vagy törlése.

Termék tulajdonság controller: Itt lehet lekérni a termékek tulajdonságait.

User controller: Itt lehet a felhasználók adatait, személynként módosítani, törölni, megjeleníteni, vagy az összes felhasználót lekérni.

### API végpontok

Restfull alkalmazások backend részei API végpontok segítségével kommunikálnak a frontend résszel. Míg a frontend kliens feladata a felhsználói interfész megjelenítése a backend feladata az adatok kezelése és azok biztosítása a kliens számára.

API végpontjaink:

Az API végpontjaink a routes/api.php fáljban találhatók.

* GET /termek\_tulajdonsagai/{termek}
* GET /kategoria\_nevek
* GET /osszes\_kategoria
* GET /osszes\_modell
* GET /osszes\_modell\_termekkel
* GET /modell\_termekei/{modell}/{modellNev}
* GET /adott\_termek/{termek}
* GET /modell\_kereses/{kereses}
* GET /adott\_kategoria/{kategoria\_nev}
* GET /bejelentkezett\_user/{email}/{password}
* GET /user\_rendelesei/{azon}
* GET /rendelestermekei/{rend\_szam}
* GET /osszes\_rendeles
* POST /rendelesrogzites
* POST /rendelestetelrogzites
* DELETE /rendelestorles/{rend\_szam}
* POST /uj\_kategoria
* DELETE /kategoria\_torles/{kat\_id}
* PUT /update\_kategoria/{kat\_id}
* PUT /update\_user/{azon}
* PUT /user\_update\_hozzaferes/{azon}
* PUT /update\_csomag/{csom\_azon}
* DELETE /user\_torles/{azon}
* POST /uj\_user
* POST /uj\_user\_azonositoval
* POST /uj\_user\_azonositoval\_hozzaferessel
* GET /osszes\_user
* DELETE /torol\_user/{id}
* POST /uj\_beszerzes
* GET /beszerzesek
* POST /uj\_modell
* PUT /update\_modell/{mod\_id}
* DELETE /modell\_torles/{mod\_id}
* POST /uj\_termek
* PUT /update\_termek/{ter\_id}
* DELETE /termek\_torles/{ter\_id}

## Frontend

Az alkalmazás Frontend oldalához React JS-t használtunk. A React egy deklaratív, effektív, és rugalmas JavaScript könyvtár, felhasználói felületek készítéséhez. Lehetővé teszi komplex felhasználói felületek összeállítását izolált kódrészletekből, amiket “komponenseknek” hívunk. [4] A React lehetővé teszi úgynevezett Single Page Application (SPA) létrehozását, ami azt jelenti, hogy egyetlen oldalon tölti be az adatokat, ezáltal gyorsabb, simább adatelérést biztosít.

Továbbá a stílusokhoz Bootstrap 5-t használtunk.

Projekt szerkezete:

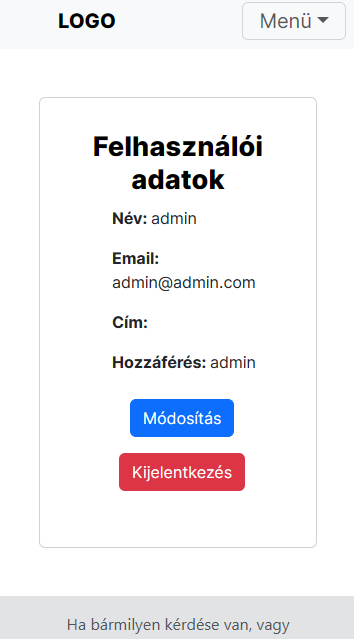
* api: Itt található az Axios és a DataService fájl. Ezek felelnek a Backenddel való kommunikációért.
* components: Ebben a mappában találhatók az oldalakon megjelenített komponensek.
* context: Itt találhatók a React specifikus providerek.
* layouts: Itt találhatóak a layout fileok amik elkülönítik a különböző jogosultsághoz kötött oldalakat, gombokat.
* pages: Ebben a mappában vannak az oldalak tartalmai.

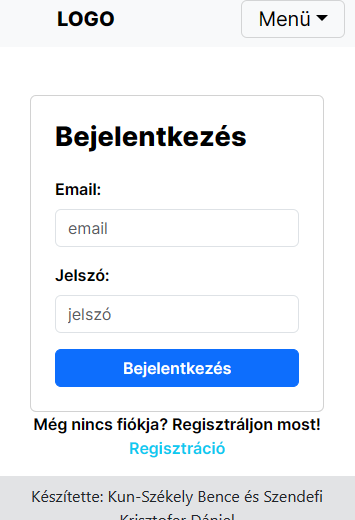
## Reszponzivitás

A **reszponzív weboldal (RWD)** egy olyan megközelítéssel tervezett weboldal, amelynek a célja az, hogy optimális megjelenést biztosítson – könnyű olvashatóság, egyszerű navigáció a lehető legkevesebb átméretezéssel és görgetéssel – a legkülönfélébb eszközökön (az asztali számítógép monitorjától egészen a mobiltelefonokig). [5]

A webaplikációnk egyszerre használ Bootstrap 5 által biztosított töréspontokat valamint css ben egyedileg beállítottakat.

Mobilos nézetek:





## Tesztelés

### Bejelentkezés és regisztráció

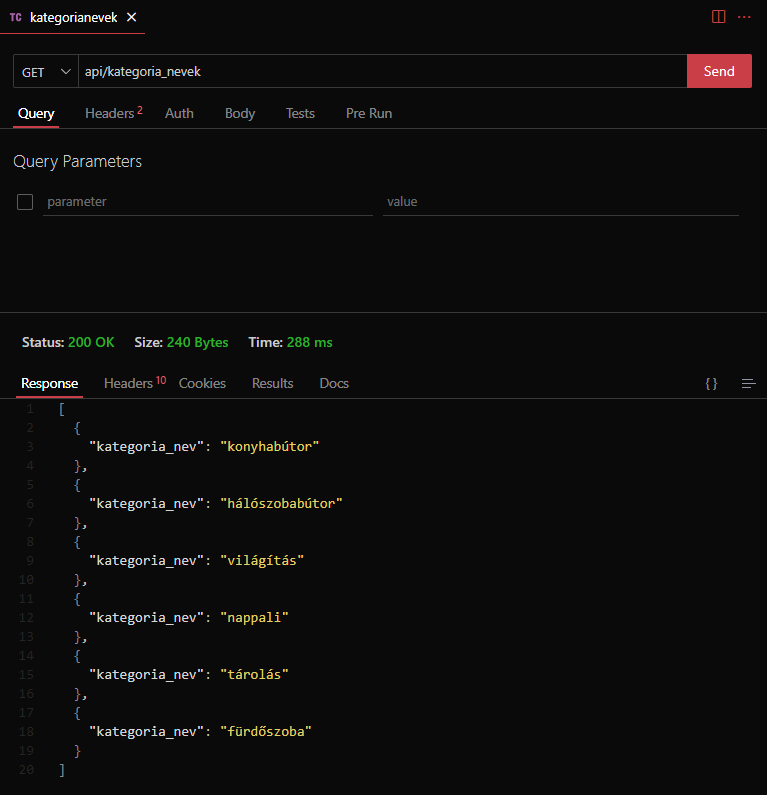
|  |  |  |
| --- | --- | --- |
| Teszteset | Várt eredmény | Kapott eredmény |
| Bejelentkezés megfelelő adatokkal | Bejelentkezik és átirányít a főoldalra | Bejelentkezik és átirányít a főoldalra |
| Bejelentkezés téves adatokkal | A bejelentkezés sikertelen. | Cannot read properties of undefined (reading '0') |
| Regisztráció megfelelően | Létrehoz egy új felhasználót, és bejelentkezik, ezután átirányít a főoldalra | Létrehoz egy új felhasználót, és bejelentkezik, ezután átirányít a főoldalra |
| Regisztráció már létező email címmel | A regisztráció sikertelen, nem kerül fel új felhasználó az adatbázisba. | A regisztráció sikertelen, nem kerül fel új felhasználó az adatbázisba. |
| Kijelentkezés | A bejelentkezett felhasználó kijelentkezik | A bejelentkezett felhasználó kijelentkezik |

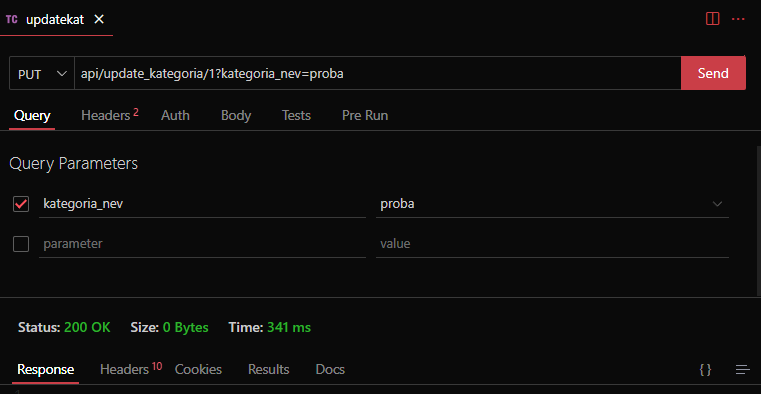
### Admin felület

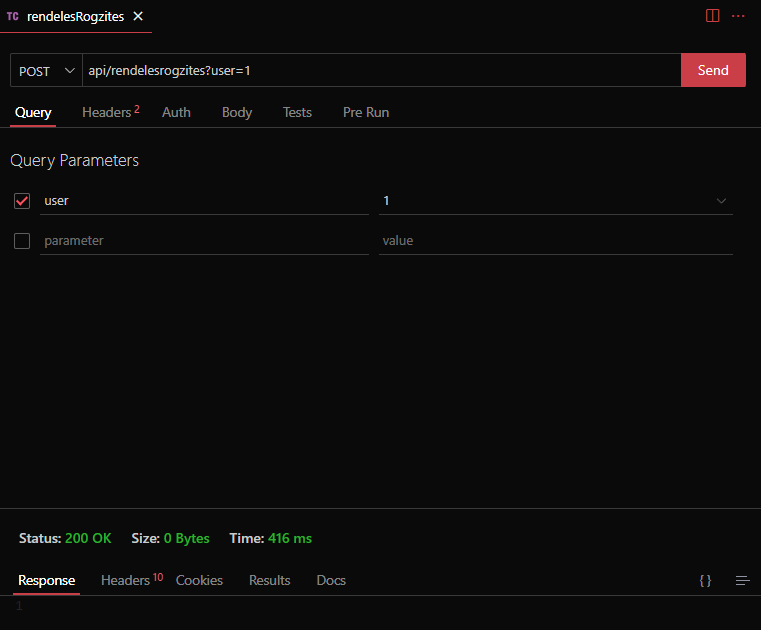
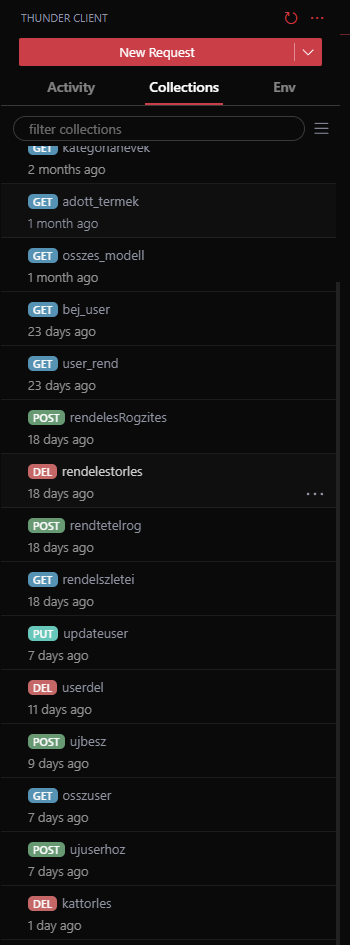
|  |  |  |
| --- | --- | --- |
| Teszteset | Várt eredmény | Kapott eredmény |
| Modell keresés szora | A megfelelő nevű modellek jelennek csak meg | A megfelelő nevű modellek jelennek csak meg |
| Modell keresés üres teli után | Az összes modell megjelenik | Az összes modell megjelenik újra |
| Új modell feltöltése | Új model töltődik fel | Új model töltődik fel az adatbázisba |
| Termék tulajdonságok szerkesztése | A változtatott tulajdonságok megváltoznak | A változtatott tulajdonságok megváltoznak a többi megmarad ahogyan volt. |
| Termék törlése | Termék törlődik | Termék nem törlődött |
| Beszerzés feltölzése minden adat kitöltése nélkül | A progam nem engedi a feltöltést | A progam nem engedi a feltöltést |
| Csomag kezelés | Csomag törlődik | Csomag törlődik |

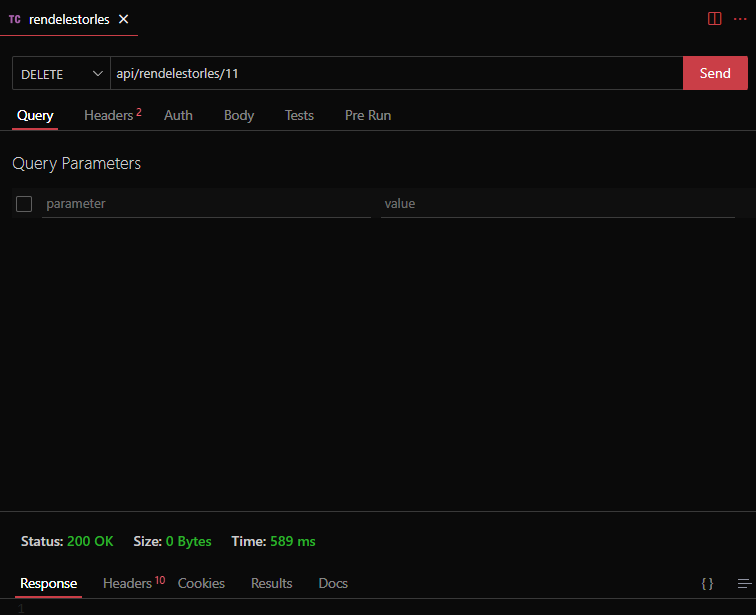
### Útvonal tesztek

Az útvonalakat Thunder Clientben teszteltük. Az ezekről készült néhány képernyőképet alább találja.









# Fejlesztési lehetőségek

Habár a projekt során igyekeztünk a lehetö legtöbb hasznos funkciót beépíteni még maradtak hátra olyan funkciók ami emelnék a projekt színvonalát.

### Fejleszési lehetőségek:

* Fizetés kezelése akár többfajta valutával
* Több szűrési lehetöség
* Árak magjelenítése (vitatható)
* Emailes értesítések
* Termékajánlási algoritmus (akár MI-val)
* Admin oldali statisztikák
* Automatikus számla összeállító

# Összefoglalás, köszönetnyilvánítás

Saját értékelésünk szerint a projekt jól sikerült, rengeteg dolgot tanultunk mind szakmai mind általános szempontból mint például projekt menedzsment, időbeosztás stb. Számunkra mindenképpen értékes eredményként valósult meg ez a projekt és tükrözi az iskolában tanult anyagok jórészét (ha nem is mindent). Mindenképpen szeretnénk tovább folytatni fejlesztését a későbbiekben.

Szeretnnénk ezúton köszönetet mondani családtagjainknak akik támogattak a sok munka és tanulás alatt. Továbbá szeretnénk köszönetet mondani Kupcsikné Fitus Ilonának a rugalmas és kitartó támogatásaiért. Cséfalvay Katalinnak a folyamatos segítségért és támogatásért. Tarczaliné Szabó Ágnes Emílianak a sok segítségért és forrásért. Csolti Péter tanárúrnak a szakszerű és alkalmazható algoritmizálási tudásért.

# Irodalomjegyzék

* <https://en.wikipedia.org/wiki/Laravel> [1]
* <https://hu.wikipedia.org/wiki/Modell-n%C3%A9zet-vez%C3%A9rl%C5%91> [2]
* <https://composite-solutions.eu/blog/az-mvc-minta-laravel> [3]
* <https://hu.legacy.reactjs.org/tutorial/tutorial.html> [4]
* <https://hu.wikipedia.org/wiki/Reszponz%C3%ADv_weboldal> [5]