Motorkölcsönző weboldal

Egy olyan szolgáltatást hoztunk létre, ami támogatja egy motorkölcsönző működését. A szolgáltatás révén az ügyfelek online módon válogathatnak és foglalhatnak le az elérhető motorok közül. A motor kölcsönző pedig a szoftver segítségével kezeli és tárolja a motorok és foglalások adatait.

Ez a szolgáltatás az ügyfelek részére egy weboldalon keresztül érheti el, valamint a fontosabb funkciók (élő foglalások megtekintése, törlése) mobilra optimalizált változatában érhetőek el. A motorkölcsönző a motorok adatainak kezelésére szolgáló felületeket egy kliens alkalmazásban érheti el.

A weboldalon rengeteg motorból tudnak válogatni az ügyfelek, ami folyamatosan cserélünk és bővítünk.

A motorkölcsönző szolgáltatást úgy képzeltük el, hogy minden kerületben van egy-egy telephelyünk, ahol vannak a mi motorjaink. A telephelyről lehet elvinni a motorokat és valamelyik másik vagy ugyanarra telephelyünkre visszavinni.

Minden telephelyen vannak recepciósok, akiknek a feladata, hogy kiadják a motorokat és a kért felszereléseket az adott méretben (protektoros ruha, cipő, sisak) az ügyfeleknek.

Jelenleg csak Budapesten vannak a telephelyeink, de jövőben szeretnénk majd terjeszkedni, illetve egy kerületen belüli telephely bővítésére.

# Motorokról:

Az ügyfél tud szűrni márkára, elhelyezésre, korra, férőhelyre (egy v. két férőhelyes), sebességváltó típusra (automata, manuális) és meghajtás típusára (elektromos, benzin), illetve kölcsönzési intervallumra is.

A felhasználó legalább 80%-os tank töltöttséggel kapja meg a motorját és minimum annyival kell leadnia. Máskülönben, leadásnál plusz költséget számolunk fel.

Ha a motor sérült állapotban kerül leadásra akkor ezt a recepciós feljegyzi és levonja a kaucióból.

# Felhasználói adatok:

Weboldalon a Regisztráció gombra kattintva tud az ügyfél regisztrálni, ha eddig nem regisztrált. A regisztráció során elkérjük a bérlő teljes nevét, az email címét, az általa megadott jelszavát, a telefonszámát, a jogosítvány azonosító kódját, miket vezethet ezzel a jogosítvánnyal (A1 vagy A2 vagy B) és egy-egy képet a Jogosítvány mindkét oldalát oldaláról.

A regisztráció után kap az ügyfél egy ellenőrző e-mailt és addig nem tud kölcsönözni motort ameddig az ellenőrző e-mailnél rá nem nyomot a megerősítésre. Plusz védelem, hogy a recepciósnak jóvá kell hagynia az ügyfelek által megadott adatokat. Ameddig nem erősíti meg a recepciós, hogy minden rendben, addig az ügyfél nem tud bérelni.

Bejelentkezéskor az email-t meg a felhasználó által megadott jelszót kéri el az oldal, ha egyezik az adatbázisban lévő email jelszó párossal a megadott email jelszó páros akkor a bal felső sarokban ott lesz az ügyfél neve, így most már tud motort bérelni.

Bejelentkezés nélkül nem tud bérelni a felhasználó motort mindenképp regisztrálnia kell. Ha a bérlés gombra kattint, az webes látogató és nincs bejelentkezve akkor a weboldal megkéri, hogy jelentkezzen be vagy regisztráljon vagy, hogy várjon egy kis türelemmel amíg meg nem erősítik az adatait.

# Felhasználói felület:

Ahogy megnyitja az oldalt lát egy nagy képet és hogy milyen weboldal ez, alatta ott van, hogy miért minket a válasszanak, kölcsönzés menete, és hogy miben emelkedünk ki a több rivális céggel szemben.

A weboldal navigációs sávjában van egy Főoldal gomb, Motorok gomb, Helyszínek gomb és egy Rólunk gomb. A navigációs sávon helyezkedik el még a User képe a jobb sarokban és ha arra rá kattintasz akkor megjelenik a Bejelentkezés és a Regisztrációs gomb. Ha már bejelentkeztél akkor az előzők helyett egy Profil valamit egy Kijelentkezés gombot látsz, és még a navigációs sávon vannak a mi social mediás eléréseinek is.

Főoldal gomb: Ha rákattint akkor a főoldalra visz át.

Motorok gomb: Ha rákattint akkor át visz egy oldalra, ahol az összes motort megnézheti képpel együtt. A Bérlő tud szűrni a fentiekben említett paraméterekre szerint „Motorokról” alcímben. Az egyes motorokról a márkáját, típusát, teljesítményét (Le, kW), hogy milyen a sebbeség váltó, és hogy jelenleg hol tartózkodik a motor, ha nincs ott helyszín akkor egy bérlőnél van.

Helyszínek gomb: Ha rákattint akkor át visz egy olyan oldalra, ahol meg lehet nézni a google térkép segítségével, hogy hol vannak a telephelyeink.

Rólunk gomb: Ha rákattint akkor át visz egy olyan oldalra, ahol a mi kamu őstörténetünkről lehet olvasni.

Bejelentkezés gomb: Ha rákattint akkor át visz egy olyan oldalra, ahol be lehet jelentkezni. A fenti részben „Felhasználói adatok” alcímben megemlítjük, hogy miket kérünk el a felhasználótól bejelentkezéskor.

Regisztrációs gomb: Ha rákattint akkor át visz egy olyan oldalra, ahol lehet regisztrálni. A fenti részben „Felhasználói adatok” alcímben megemlítjük, hogy miket kérünk el a felhasználótól regisztráláskor.

Profil gomb: Ha rákattint akkor át visz egy olyan oldalra, ahol látod a rendeléseidet itt is lehet törölni és módosítani a bérlést. Itt látod még az adataidai, amiket módosíthat.

Kijelentkezés gomb: Ha rákattint akkor át visz egy olyan oldalra, ahol ki lehet jelentkezni.

A footerben megtalálhatóak az elérhetőségünk (helyszín, email, telefonszám) valamint itt is vissza tudsz menni a főoldalra a “Kezdőlap” linkre kattintva linkre kattintva. Az adatvédelmet és a felhasználói feltételeket itt lehet megnézni, valamint a nyitvatartásunkat.

# Adminisztrátori alkalmazás

Az adminisztrátori alkalmazás szerveroldali komponense egy központi számítógépen fut, amelyhez az adminok a saját gépeikről csatlakoznak egy kliens alkalmazás segítségével. Három munkakört különböztetünk meg az adminisztrátori alkalmazásban: Szuperadmin, Szerelő és Recepciós. Minden egyes munkakör más oldalakat lát, vagyis például egy Recepciós nem tud felvenni új alkalmazottat.

A Szuperadminok foglalkoznak az Adminisztrátori alkalmazás felhasználóival. Ők tudnak felvenni új embereket az alkalmazásba, tudják módosítani a szerepköreiket, a nevüket, az email címüket és a jelszavakat, valamint deaktiválni tudják az alkalmazottakat, és vissza is tudják állítani az felhasználójukat. Ezen kívül meg tudják nézni a logokat, és ezáltal láthatják, hogy az egyes alkalmazottakat miket csináltak a felületen.

A Szerelők a motorokkal foglalkoznak. Hozzá tudnak adni a flottához új motorokat úgy, hogy beviszik az adatokat a motorról, mint például a rendszámát, márkáját. Ki tudják venni a flottából a motorokat, és módosíthatják is az adataikat, mint például a teljesítmény vagy akár a bérlési ár. A Szerelők, a részletek oldalon meg tudják nézni a motor hibáit és adatait, és itt tudják berakni a szervizbe. Ha szervizben van a motor, akkor az ügyfelek nem tudják kibérelni.

A Recepciósok az ügyfelek motor bérlésével és adategyeztetésével foglalkoznak.

**Szuperadmin felület:**

Amikor a Szuperadmin belép a felületére, az első dolog, amit látni fog, az összes adminisztrátor listája, akik már regisztráltak. Innen több lehetőség közül választhat:

* **Új admin hozzáadása:** Itt új adminisztrátorokat adhat hozzá. Ehhez meg kell adnia az admin teljes nevét, egy egyedi email címet és egy megfelelő jelszót (kis- és nagybetűk, számok, speciális karakterek). Emellett ki kell választania a pozícióját (Szuperadmin, Szerelő vagy Recepciós). Ha minden adatot megadott, akkor a "Hozzáadás" gombra kattintva a rendszer ellenőrzi, hogy minden rendben van-e. Ha igen, hozzáadja az új adminisztrátort.
* **Műveleti napló áttekintése:** Itt megtekintheti, hogy az egyes adminisztrátorok milyen műveleteket végeztek el a rendszerben.
* **Deaktivált adminok:** Itt megtekintheti a deaktivált adminokat, és ha szükséges, visszaállíthatja őket.
* **Kijelentkezés:** Ha befejezte a munkát, itt tud kijelentkezni a Szuperadmin.

A Szuperadmin bármelyik adminisztrátort deaktiválhatja, és az adminok adatait is módosíthatja, például a nevüket, email címüket, jelszavukat és pozíciójukat.

**Szerelő felület:**

A Szerelő felülete egy kicsit másképp működik. Amikor belép, az első dolog, amit látni fog, az összes elérhető motor, amit bérlésre kínálnak. Itt több opció közül választhat:

* **Új motor hozzáadása:** Itt meg kell adnia néhány adatot a motorról, például a márkáját, bérlési árát és lóerejét. A rendszer ellenőrzi, hogy minden adatot helyesen adott-e meg, és ha igen, hozzáadja a motort a flottához.
* **Szervizben:** Itt láthatja azokat a motorokat, amelyek szervizbe kerültek. Ha egy motor javítása befejeződött, akkor a szerelőnek ki kell törölnie a javított problémákat, mielőtt visszaállítja a motort.
* **Kijelentkezés:** Ha végezett, itt tud kijelentkezni a felületről.

A szerelők módosíthatják a motor adatait és törölhetik őket a rendszerből. Minden motor adatainál található egy részletes gomb, amellyel megtekintheti a motor problémáit, és itt tudja szervizbe küldeni.

**Recepciós felület:**

A Recepciós felületén két fő gomb található:

* **Ügyfél adatok ellenőrzése és jóváhagyása:** Itt tudja ellenőrizni és jóváhagyni az ügyfél adatait.
* **Kölcsönzés:** itt tudja kezelni a kölcsönzés adatait, például rögzíteni a motor átadását vagy visszavételét.

**Ügyfél adatok ellenőrzése és jóváhagyása:**

A Recepciós ezen az oldalon látja azokat az ügyfeleket, akiket még nem hagytak jóvá. A jóváhagyáshoz megtudja nézni a Recepciós az ügyfél adatait és a jogosítványát és ha mindent jónak lát akkor jóváhagyhatja és bérelhet az ügyfél.

**Kölcsönzés:**

A Recepciósok a Kölcsönzések oldalon csak a jövőbeli, illetve a még le nem zárult bérléseket látja (a már lezárult bérlések nem jelennek meg neki). Keresni tud a bérlés azonosítója vagy az ügyfél neve alapján. Ezeket a bérléseket kezelheti, és akár törölni is tud bérléseket, például, ha az ügyfél nem jött el a lefoglalt motorért.

Ha a bérlést kezeli, három gombot talál: az egyik a "Vissza", amely visszaváltja őt az előző nézethez, míg a másik két gomb a "RÉSZLET" és az "ÁTADVA" gombok.

Ha a "RÉSZLET" gombra kattint, akkor meg tudja nézni az ügyfél és a motor adatait, és ellenőrizheti, hogy az ügyfél rendelkezik-e a motor vezetéséhez szükséges jogosítvánnyal. Az ügyfél a helyszínen fizeti ki a bérlés díját a Recepciósnak, a motor csak ezt követően kerül átadásra.

Az "ÁTADVA" gombot pedig akkor használja, amikor a motort ténylegesen átadja az ügyfélnek. Ekkor a motor helyszín értéke nullára változik, jelezve, hogy az ügyfélnél van. Ezt a gombot A Recepciós csak egyszer tudja megnyomni, így nem adhatja át többször a motort.

Ha az ügyfél visszahozza a motort, a Recepciós egy checklist alapján ellenőrzi annak állapotát. Ha bármilyen hibát talál, azt feljegyzi, és a kárt levonja a kaucióból. A hibák a motor visszavételével kapcsolatos adatok között kerülnek tárolásra.

|  |  |
| --- | --- |
| motorcycles table |  |
| brand | A motor márkája (szöveg, maximum 20 karakter) |
| type | A motor típusa (szöveg, maximum 100 karakter) |
| licencePlate | A motor rendszáma (szöveg, maximum 7 karakter, egyedi) |
| year | A motor évjárata (szám, 0 – 65.535) |
| gearbox | A motor sebeségváltó típusa (szöveg, maximum 25 karakter) |
| fuel | A motor üzemanyag típusa (szöveg, maximum 1 karakter) |
| powerLe | A motor lóereje (törtszám) |
| powerkW | A motor kilowattja (törtszám) |
| engineSize | A motor motorjának mérete (törtszám) |
| drivingLicence | A motort milyen jogosítvánnyal vezethető (szöveg, maximum 4 karakter) |
| places | A motor férőhelye (szám, 0 - 255) |
| price | A motor bérlési összege (szám, 0 - 4294967295) |
| deposit | A motor kaució összege (szám, 0 - 4294967295) |
| trafficDate | A motor forgalmijának lejárati ideje (dátum) |
| location | A motor aktuális helyszíne (szöveg, lehet null értéke) |
| image | A motor képének a neve (szöveg) |
| isInService | A motor szervizben van-e (logikai érték) |
| problamComment | A motor problémái (szöveg, lehet null értéke) |

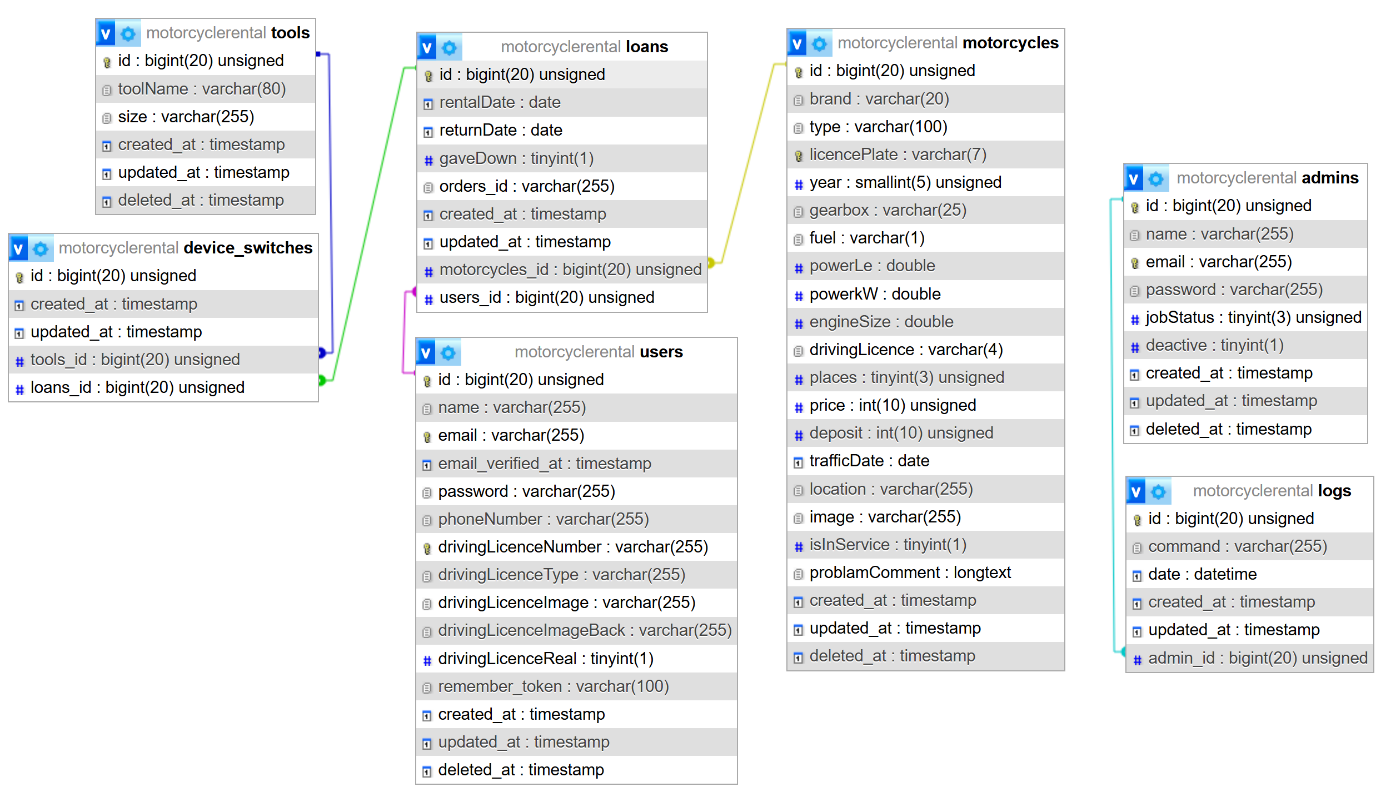
|  |  |
| --- | --- |
| loans table |  |
| rentalDate | A bérlő bérlésének kezdeti dátuma (dátum) |
| returnDate | A bérlő bérlésének visszahozási dátuma (dátum) |
| gaveDown | A motort leadták-e (logikai érték) |
| problemDescription | A Userrel vagy a motorral adott problémák (szöveg) |
| orders\_id | Minden egyes rendelésnek egyedi azonosítója (szöveg) |
| motorcycle\_id | Az adott kölcsönzéshez tartozó motorcycles azonosító (szám, egyedi) |
| users\_id | Az adott kölcsönzéshez tartozó users azonosító (szám, egyedi) |

|  |  |
| --- | --- |
| tools table |  |
| toolName | Az eszköz/felszerelés neve (szöveg, maximum 80 karakter) |
| size | Az eszköz/felszerelés mérete (szöveg, maximum 255 karakter) |

|  |  |
| --- | --- |
| users table |  |
| name | A felhasználó neve (szöveg, maximum 255 karakter) |
| email | A felhasználó email címe (szöveg, maximum 255 karakter, egyedi) |
| password | A felhasználó jelszava (szöveg, maximum 255 karakter) |
| phoneNumber | A felhasználó telefonszáma (szöveg, maximum 255 karakter) |
| drivingLicenceNumber | A felhasználó jogosítvány egyedi azonosítója (szöveg, maximum 255 karakter, egyedi) |
| drivingLicenceType | A felhasználó jogosítvány típusa (szöveg, maximum 255 karakter) |
| drivingLicenceImage | A felhasználó jogosítvány képének a neve (az előlapi oldala) (szöveg, maximum 255 karakter) |
| drivingLicenceImageBack | A felhasználó jogosítvány képének a neve (az hátoldali oldala) (szöveg, maximum 255 karakter) |
| drivingLicenceReal | A felhasználó jogosítványa valódi és érvényes-e (logikai érték) |

|  |  |
| --- | --- |
| device\_switches table |  |
| tools\_id | Az adott eszközhöz tartozó eszköz/felszerelés azonosító (szám, egyedi) |
| loans\_id | Az adott eszközhöz tartozó kölcsönzési azonosító (szám, egyedi) |

|  |  |
| --- | --- |
| admin table |  |
| name | Az admin neve (szöveg, maximum 255 karakter) |
| email | Az admin email címe (szöveg, maximum 255 karakter, egyedi) |
| password | Az admin jelszava (szöveg, maximum 255 karakter) |
| jobStatus | Az admin munkaköri státusza (szám, maximum 255 karakter) |
| deactive | Az admin deaktív-e (logikai érték) |



|  |  |
| --- | --- |
| logs table |  |
| command | A parancs leírása (szöveg, maximum 255 karakter) |
| date | A parancs végrehajtási dátuma (dátum) |
| admin\_id | Az adott loghoz tartozó admin azonosító (szám, egyedi) |

# Felhasználói weboldal

## Kölcsönzés működése

A felhasználók egy általunk megosztott szerveren futó weboldalra tudnak fellépni, amelyen a motorokat tudják kibérelni. Ezeket a motorokat kívánt ELÉRHETŐ napokon tudják kibérelni, személy szerint kívánt eszközökkel, melyeket rendelés előtt egy nappal tudnak törölni a vagy hozzáadni a felhasználó. Minden eszköz típusból legfeljebb 2 db-ot lehet kölcsönözni. Az eszközökért egyedi kauciót számolunk fel a típustól függően.

Kedvezményekkel is kínáljuk a felhasználókat. Ha 3 vagy annál több napra kölcsönöz a személy, akkor -20% kedvezményt vonunk le a napi árból, 7 vagy annál több napnál meg már -30% a járó kedvezmény.

## Regisztráció

A felhasználó amíg nem regisztrált be addig nem tud bármilyen jelentős aktivitást végezni az oldalon(bérlést). Ezért el kell végeznie a regisztrációt, ha szeretne bérelni az oldalunkon.

Meg kell adnia a teljes nevét, email címét, telefonszámát, jogosítvány azonosítóját, jogosítvány típusát és olvasható kép formátumban a jogosítványának elejét és hátulját. Ezen felül egy egyedi jelszóval is kell rendelkeznie, amit saját maga ad meg.

A jelszónak legalább 8 karakter hosszúnak kell lennie, tartalmaznia kell kis és nagybetűt, és legalább egy számot.

### Megerősítés

Regisztrálást követően küldünk egy a felhasználó által megadott elérhetőségi email címre, amellyel megerősíti, hogy tényleg övé az email cím vagy hogy létezik-e.

Ezen felül az adminisztrátornak meg kell erősítenie, hogy valós, és aktív a jogosítvány, amelyet feltöltött a felhasználó.

Amíg a fenti 2 feltétel nem valósul meg addig a felhasználó képtelen bármilyen aktivitást végezni(bérlést).

## Bejelentkezés

A bejelentkezéskor a felhasználó az email címével és a jelszavával tud belépni a profiljába, ha a felhasználó elfelejtette a jelszavát, akkor az **“Elfelejtett jelszó”** mezőre kattintva az adott email címre küldünk egy emailt, amelyre kattintva be tudja állítani az új jelszavát.

Bejelentkezés után a felhasználó bármilyen aktivitást végezhet, ha az összes regisztrációs feltételeket teljesítette. Tud új motort bérelni eszközt hozzáadni vagy törölni a bérléshez és törölni a rendelést, ha megadott időintervallumon kívül esik a kívánt művelet.

### Profil

Bejelentkezés után a felhasználó megtudja tekinteni és módosítani a profilját. A megnevezését, elérhetőségeit, és a jogosítványról készült képeit, ha esetleg újat csináltat a felhasználó. Az utóbbi esetben újabb ellenőrzésen megy keresztül a jogosítványa.

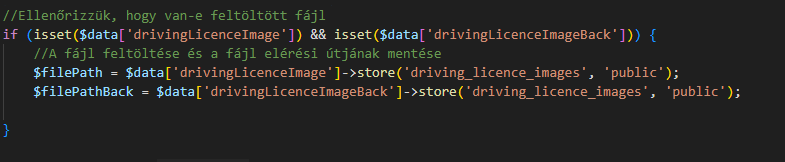
Ezen felül itt tudja a rendelési előzményeit megtekinteni, azokat törölni vagy módosítani.

## A működés

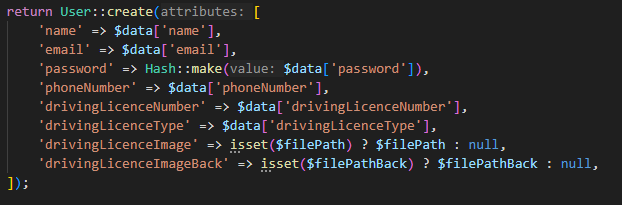
### Regisztráció

Az oldalra lépve a felhasználónak regisztrálnia kell foglalás érdekében. Amelyet a jobb oldalt a lenyíló menüben a regisztráció gombra nyomva tud megtenni.

Az adatok megadása során a regisztrációs gombra nyomva feltöltjük az adatbázisba a felhasználó által megadott adatokat, ezzel regisztrálva őt.

1. **Fájlok ellenőrzése és feltöltése:**
   * Megvizsgáljuk, hogy a drivingLicenceImage és drivingLicenceImageBack mezők tartalmaznak-e feltöltött fájlokat.
   * Ha igen, a fájlokat a store metódussal a public/driving\_licence\_images mappába mentjük, és elmentjük az elérési útjukat.

**2. Új felhasználó létrehozása:**

* + A User::create metódus segítségével létrehozunk egy új felhasználót az adatbázisban, ahol:
    - A name, email, phoneNumber, drivingLicenceNumber, és drivingLicenceType mezők a kapott adatokból töltődnek.
    - A password biztonságosan, hash-elve kerül mentésre a Hash::make használatával.
    - ****A feltöltött fájlok elérési útját (drivingLicenceImage, drivingLicenceImageBack)

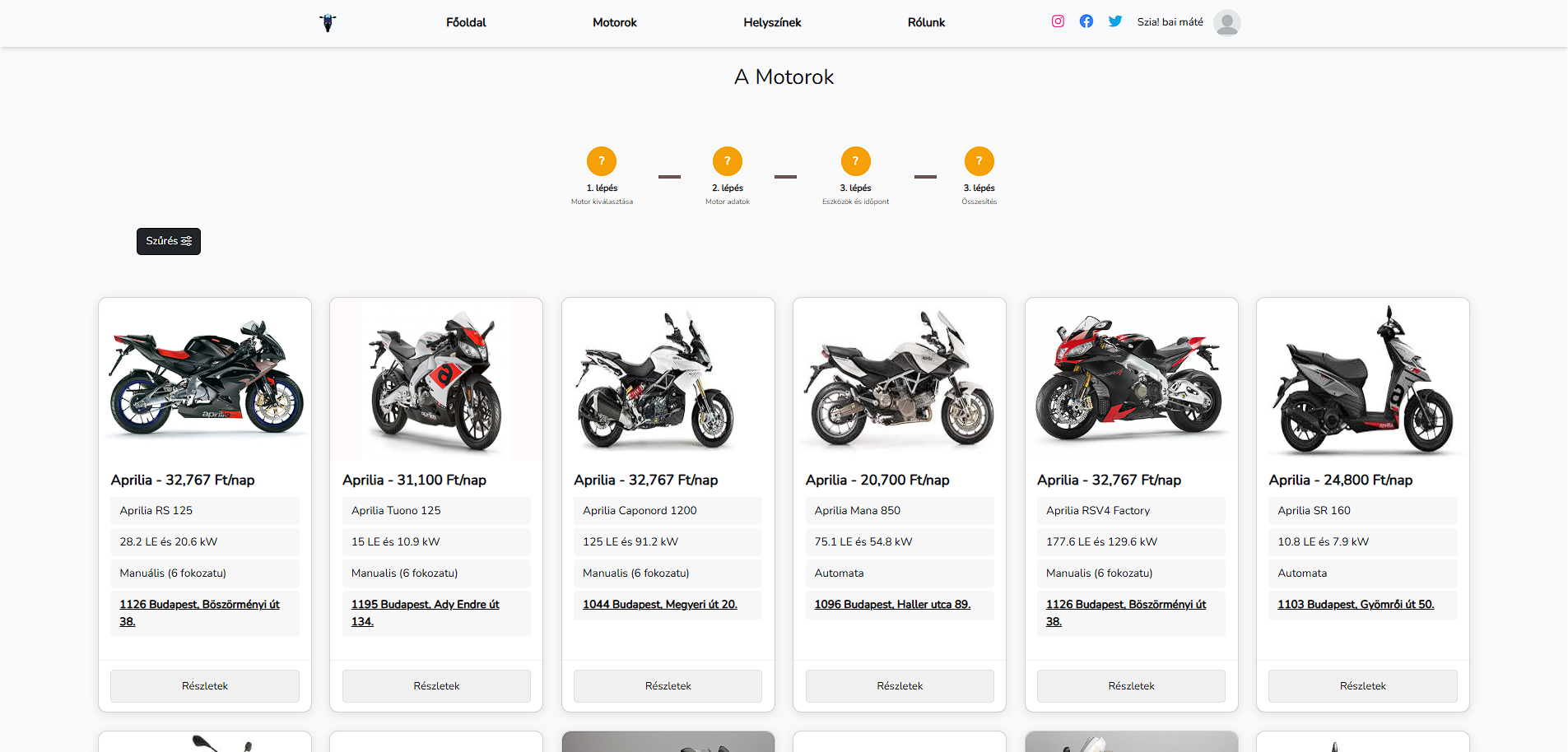
|  |  |  |  |
| --- | --- | --- | --- |
| name | "John Doe" | Yes | User's full name. |
| email | "[john@example.com](mailto:john@example.com)" | Yes | Unique email address. |
| password | "secret123" | Yes | Hashed via Hash::make(). |
| phoneNumber | "123456789" | Yes | Contact number. |
| drivingLicenceNumber | "AB123456" | Yes | Driver's license ID. |
| drivingLicenceType | "B" | Yes | License category (e.g., B, C). |
| drivingLicenceImage | UploadedFile (front side) | Yes | Stored in public/driving\_licence\_images. |
| drivingLicenceImageBack | UploadedFile (back side) | Yes | Stored in public/driving\_licence\_images. |

Ezt követően, korábban említve az email címet meg kell erősítenie a felhasználónak. Ehhez egy beépített laraveles metódust használtunk, amely elküldi az adott művelethez szükséges email-t.

### Bérlés

Fent megemlítve, a navbar-on(a fenti menüben) lehet navigálni az oldalak között.

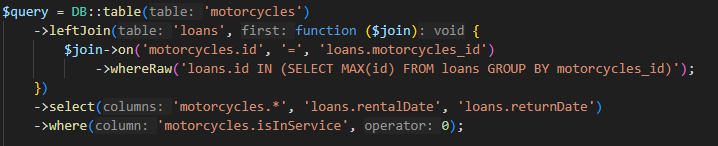
A ***motorok*** fülre kattintva ezt az oldal látjuk:



Itt felsorolja az összes motort, ami létezik a rendszerünkben. Amelynek az elérési útvonala így van implementálva:



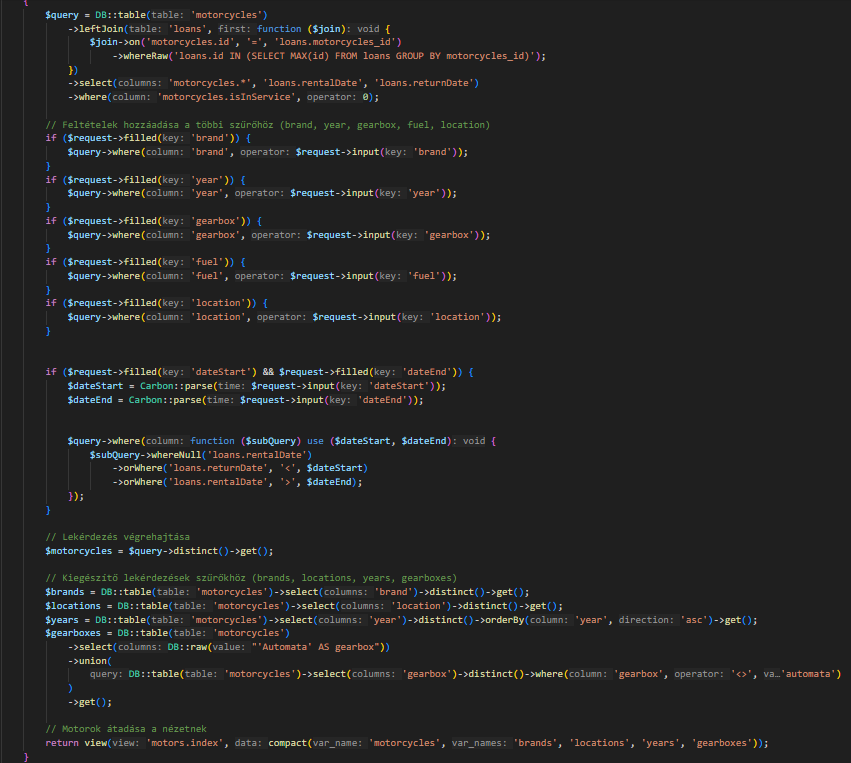
A motorok adatai lekérése az adatbázisból:



Sorrendbe helyezi a megjelenített motorokat és csak azokat jeleníti meg amelyek nincsenek szervízi munkálatok alatt.

Ha az oldalon a bal oldali szűrés gombra kattint a felhasználó akkor tud szűrni, hogy milyen motort szeretne látni. Ez a korábban részletezve is volt, hogy milyen feltételekkel tud.

### A szűrés működése:



1. **Brand szűrés:** Ellenőrizzük, hogy a lekérdezés tartalmazza-e a márkát (brand). Ha igen, a lekérdezést szűrjük a megadott márkára.
2. **Évjárat szűrés:** Ha az évjárat (year) meg van adva, csak az adott dátumban elérhető motorokat kérdezzük le.
3. **Sebességváltó szűrés:** Ha a sebességváltó típusa (gearbox) meg van adva, a találatokat az alapján szűkítjük.
4. **Üzemanyag szűrés:** Ha az üzemanyag típusa (fuel) szerepel a lekérdezésben, annak megfelelően szűrjük az adatokat.
5. **Helyszín szűrés:** Ellenőrizzük, hogy a helyszín (location) mező ki van-e töltve, és csak az adott helyszínhez tartozó adatokat adjuk vissza.
6. **Időszak szűrés:** Ha kezdő és záró dátum (dateStart és dateEnd) is meg van adva, ellenőrizzük, hogy az adott időszak alatt a jármű elérhető-e. Ez kizárja azokat, amelyek már foglalva vannak a megadott időszakra.
7. **Egyedi eredmények lekérdezése**: A lekérdezés végén az eredményt szűrjük, hogy csak az egyedi (ismétlődés nélküli) találatokat kapjuk meg.
8. **Szűrő adatok előkészítése:** Külön lekérdezésekkel begyűjtjük a szűrőkhöz szükséges egyedi adatokat, például a márkák, helyszínek, évjáratok, vagy sebességváltó típusok listáját.
9. **Adatok átadása a nézetnek**: Az összes összegyűjtött adatot (motorok és szűrők) elküldjük a nézetnek (view), hogy a felhasználó számára megjeleníthetők legyenek.

|  |  |  |
| --- | --- | --- |
| Component | SQL Equivalent | Description |
| Base Query | SELECT motorcycles.\*, loans.rentalDate, loans.returnDate | Fetches motorcycles not marked as isInService. |
| Left Join with Loans | LEFT JOIN loans ON motorcycles.id = loans.motorcycles\_id AND loans.id IN (...) | Joins with the latest loan record for each motorcycle using a subquery. |
| Date Range Filter | WHERE (loans.rentalDate IS NULL OR loans.returnDate < ? OR loans.rentalDate > ?) | Excludes motorcycles rented during the specified dateStart-dateEnd range. |
| Distinct Results | SELECT DISTINCT motorcycles.\* | Ensures no duplicate motorcycle records. |

A részletek gombra kattintva az oldal átdob minket egy másik oldalra, ahol az adott motornak az adatait tekinthetjük meg részletesen.

Ennek az elérési útvonala:



Innentől csak a motor id-jával dolgozunk, és annak adatait jelenítjük meg.



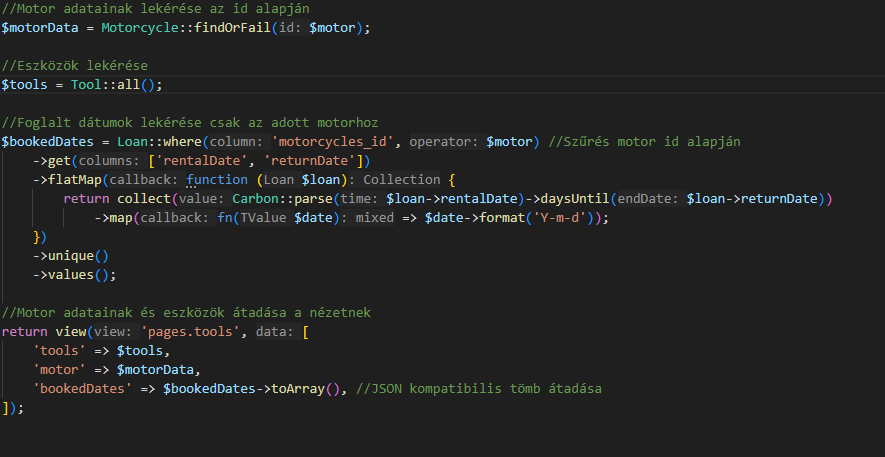
Amint a felhasználó meggyőződött arról, hogy ezt a motort szeretné, rányom a bérlés gombra, ami ezt az elérési útvonalat tartalmazza:



Itt a bérléshez adja hozzá a felhasználó milyen eszközöket szeretne a motorja mellé, és a kiválasztott dátum intervallumot, hogy mely időszakban szeretné a motort bérelni.

Itt a rendszer az alábbiakban működik:

1. Motor adatainak lekérése: A megadott motor azonosító (id) alapján lekérjük a motor adatait az adatbázisból. Ha nem található ilyen azonosító, hibát dob.
2. Eszközök listázása: Lekérjük az összes elérhető eszközt (tools) az adatbázisból. Ez általában a motorhoz kapcsolódó kiegészítők vagy felszerelések listája.
3. Foglalt dátumok lekérése: Az adott motorhoz tartozó kölcsönzési időszakokat (rentalDate és returnDate) gyűjtjük össze, majd:
   * **Időszakok felosztása:** Az egyes kölcsönzések kezdő- és záró dátumai között lévő összes napot meghatározzuk.
   * **Formázás:** Az időpontokat Y-m-d formátumra alakítjuk, hogy egyszerű JSON-ként továbbadhatók legyenek.
   * **Duplikációk eltávolítása:** Eltávolítjuk az ismétlődő dátumokat, hogy minden foglalt nap csak egyszer szerepeljen.
   * **Értékek kigyűjtése:** A rendezett, ismétlődés nélküli foglalt dátumok listáját alakítjuk ki.
4. Adatok átadása a nézetnek: A nézetnek (view) továbbítjuk az összegyűjtött adatokat:
   * Az összes eszközt (tools),
   * Az adott motor adatait (motor),

A foglalt dátumokat JSON-kompatibilis formában (bookedDates).

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Type | Example Value | Use Case |
| $tools | Collection | [{"id":1,"name":"Helmet"},...] | Display available tools for selection. |
| $motor | Motorcycle | {"id":5,"brand":"Yamaha",...} | Show motorcycle details on the page. |
| $bookedDates | Array | ["2025-05-01","2025-05-02",...] | Disable booked dates in a calendar UI. |

Így a következő oldalon, amelyet a tovább gomb megnyomásával érünk el, tudjuk alkalmazni a megadott adatokat. Eszközök, azok méretei és a kibérlési intervallum.

Amelynek az elérési útvonala:

Ezen az oldalon összesítjük a bérléshez tartozó adatokat.

1. A motor adatait
2. felhasználó adatait
3. A kapcsolt eszközök adatait

Mindemellett bekalkulálva a bérlésnek a végösszegét levonva a kedvezményt, ha van.

Az árba beletartozik az eszközök utáni kaució és az eszközök napi bérlési ára is.

Innen a rendelés megerősítése gombra nyomva a rendelés rögzítésre kerül.

Ennek az elérési útvonala:



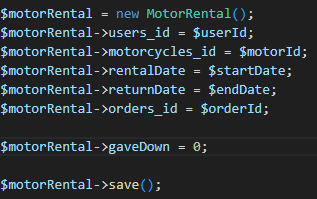
A rendelés mentésének több táblába való művelete:

1. Felhasználói azonosító lekérése**:** Az aktuális bejelentkezett felhasználó azonosítóját (userId) lekérjük az Auth::id() függvénnyel.
2. A képen szöveg, képernyőkép, Betűtípus látható

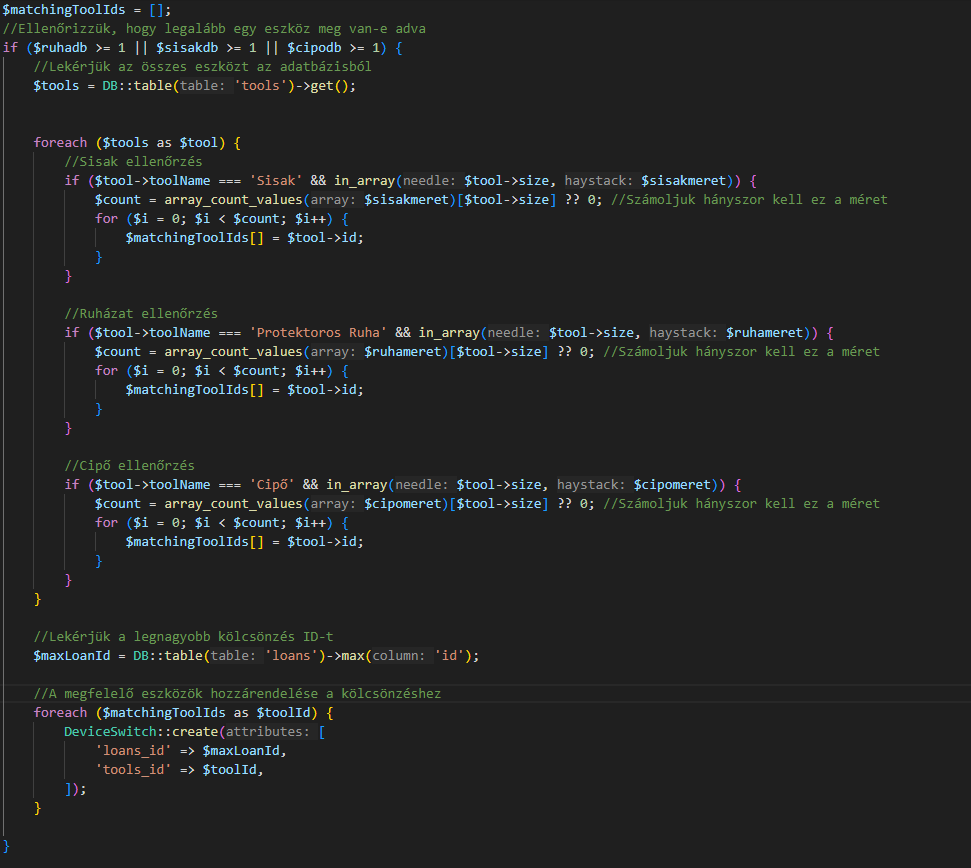
   Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.**Session adatok lekérése:** A motorbérléshez szükséges adatokat, mint a bérlés kezdő- és végdátuma (startDate, endDate), illetve a különböző eszközök darabszámát és méretét (sisakdb, ruhadb, cipodb, sisakmeret, ruhameret, cipomeret), a session-ből töltjük be.
3. **Rendelési azonosító generálása:** Egyedi rendelési azonosítót (orderId) hozunk létre, amely az ORD- előtagból és egy véletlenszerű, 8 karakter hosszú nagybetűs karakterláncból áll.



1. **Motorbérlés mentése:** Létrehozunk egy új MotorRental példányt, amely tartalmazza a bérléshez kapcsolódó adatokat, például:
   * Felhasználói azonosító (users\_id),
   * Motor azonosító (motorcycles\_id),
   * Bérlés kezdő- és végdátuma,
   * Egyedi rendelési azonosító. Továbbá, az gaveDown mezőt alapértelmezetten 0-ra állítjuk, jelezve, hogy az eszközöket még nem adták le.



1. **Eszközök ellenőrzése:** Ha bármelyik eszközből van megadva darabszám (sisak, ruha, cipő), akkor az adatbázisból lekérjük az összes elérhető eszközt.
2. **Eszközök szűrése és hozzárendelése:**
   * Az eszközöket (tools) típus és méret alapján ellenőrizzük.
   * Megszámoljuk, hogy az adott típusból és méretből mennyi szükséges.
   * Az eszköz azonosítókat (toolId) hozzáadjuk a megfelelő listához.
3. **Kölcsönzés azonosítójának lekérése:** Lekérjük a kölcsönzések közül a legnagyobb azonosítót (maxLoanId), hogy az új eszközhozzárendelések ehhez a kölcsönzéshez kapcsolódjanak.



1. **Eszközök mentése:** A szűrés után az összes megfelelő eszközt a kölcsönzéshez rendeljük a DeviceSwitch modellel, ami az eszközök és a kölcsönzések közötti kapcsolatot kezeli.

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter/Data | Example Value | Source | Description |
| motorId | 5 | URL parameter | ID of the rented motorcycle. |
| userId | 12 | Auth | Authenticated user's ID. |
| startDate/endDate | "2025-05-01"/"2025-05-05" | Session | Rental period dates. |
| helmetCount/gearCount/shoeCount | 2/1/1 | Session | Number of selected tools per type (helmet, protective gear, shoes). |
| helmetSizes/gearSizes/shoeSizes | ["M", "L"] | Session | Selected sizes for each tool type. |

1. **Adatok átadása a nézetnek:** Az összegyűjtött adatokat továbbítjuk a végső nézet (final\_page) számára, beleértve:
   * A rendelési azonosítót,
   * A motor azonosítót,
   * A bérlés kezdő- és végdátumát,
   * A kiválasztott eszközök azonosítóit,
   * És az eszközökhöz kapcsolódó darabszámot és méreteket.

Innentől már csak átvennie kell a felhasználónak a motort az adott időpontban(napon).

Ezután a felhasználó a profil menüben megtudja nézni a saját adatait és rendeléseit, amelyeket tud módosítani, utóbbit még törölni is.A képen szöveg, szoftver, Weblap, Webhely látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

Baloldalt az adatok módosítása gombra nyomva egy felugró ablak jelenik meg ahol a felhasználó a meglévő adatait tudja módosítani.

Onnan a mentés gombra nyomva egy functiont hívunk meg melynek alábbi az elérési útvonala:

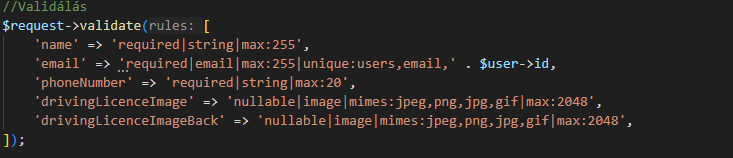


Ebben a funkcionben az adatbázisban a felhasználó adatait frissítjük.

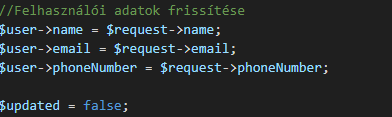
1. **Aktuális felhasználó lekérése:** Az Auth::user() segítségével lekérjük az aktuálisan bejelentkezett felhasználó adatait.



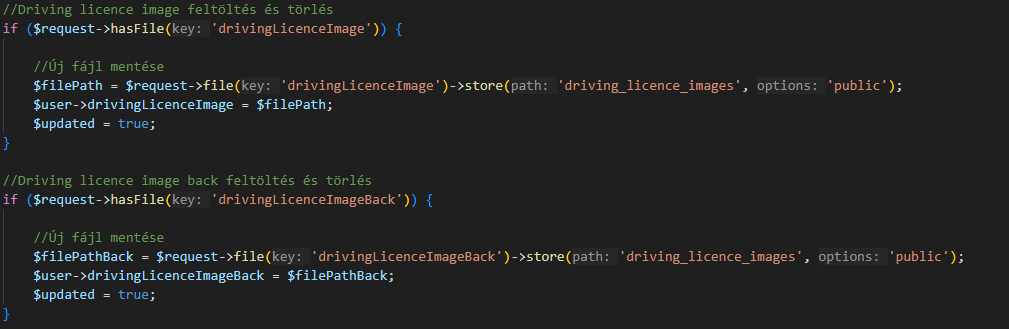
1. **Adatok validálása:**
   * A felhasználói adatokat a validate metódussal ellenőrizzük:
     + name: Kötelező, szöveg típusú, maximum 255 karakter.
     + email: Kötelező, érvényes e-mail cím, maximum 255 karakter, és egyedi az users táblán belül, kivéve a jelenlegi felhasználó e-mail címét.
     + phoneNumber: Kötelező, szöveg típusú, maximum 20 karakter.
     + drivingLicenceImage, drivingLicenceImageBack: Nem kötelező, de ha van, akkor kép típusúnak kell lennie a megadott fájltípusokkal és maximum 2048 KB mérettel.



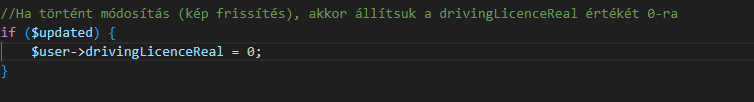
1. **Felhasználói adatok frissítése:**
   * Az Auth által lekért felhasználó adatait (name, email, phoneNumber) frissítjük a kérésből ($request) érkező adatokkal.



1. **Képek kezelése:**
   * Ellenőrizzük, hogy a kérés tartalmazza-e az adott fájlt (hasFile).
   * Ha igen, akkor a fájlt az store metódussal feltöltjük a public tárhelyre, a driving\_licence\_images mappába.
   * Az új fájl elérési útját eltároljuk a felhasználó megfelelő attribútumában (drivingLicenceImage vagy drivingLicenceImageBack).

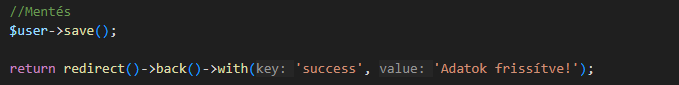
Az updated változót igazra állítjuk, jelezve, hogy történt változtatás.

1. **Jogosítvány státusz frissítése:**

Ha történt bármilyen frissítés (például új kép feltöltése), akkor a drivingLicenceReal értékét 0-ra állítjuk, jelezve, hogy az új jogosítvány képeket még nem ellenőrizték.

1. **Adatok mentése:**

A felhasználó adatait az adatbázisba mentjük a save metódussal.

**Visszairányítás:** Sikeres frissítés esetén a felhasználót visszairányítjuk az előző oldalra, és egy success üzenetet küldünk, amely tájékoztatja, hogy az adatok sikeresen frissültek.

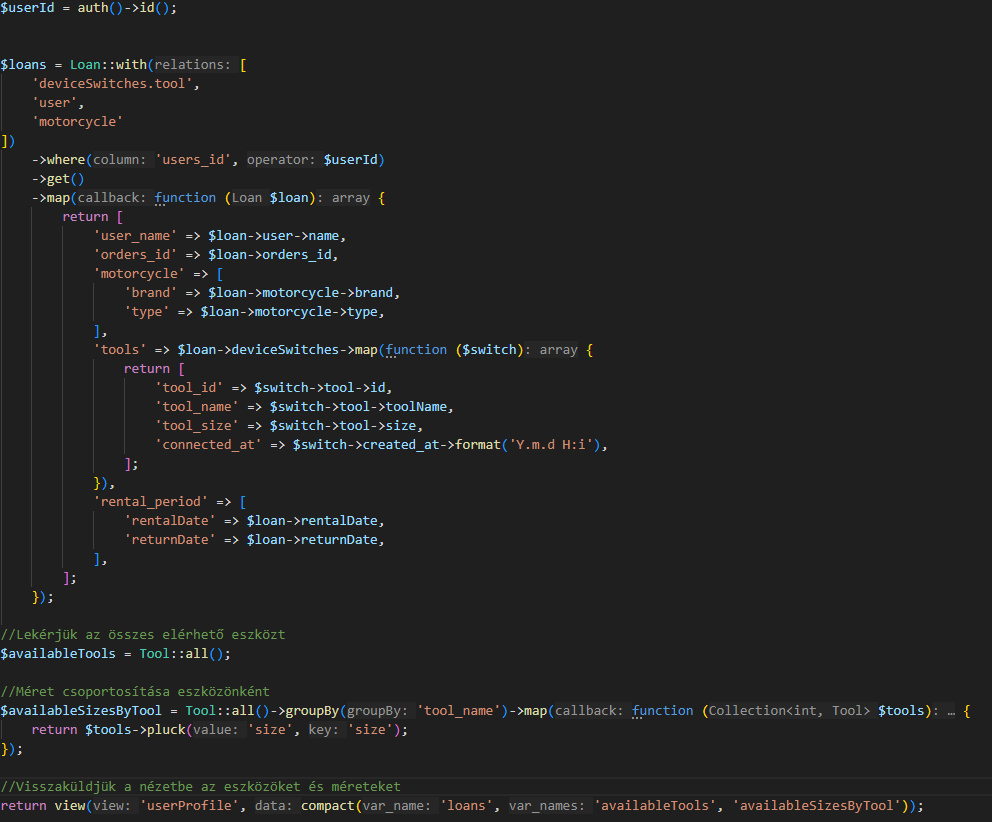
|  |  |
| --- | --- |
| Component | Action |
| Basic Info | Updates name, email, and phoneNumber from request. |
| Image Uploads | - Stores new images to public/driving\_licence\_images. - Updates DB paths. |
| Verification Reset | Sets drivingLicenceReal=0 if license images are updated. |

## **Rendelési előzmények**

A felhasználó megtudja tekinteni a korábbi rendeléseit.

Ehhez meghívtunk egy másik functiont melynek elérési útvonala:

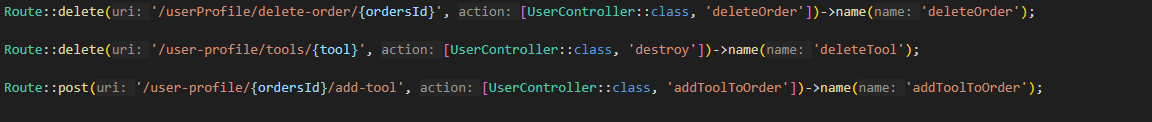


Itt lekérjük azokat az adatokat a loans táblából, ahol a felhasználó id-ja megegyezik az users\_id-val. Hozzáadva azokat az eszközöket (ha vannak) amiknek a rendelési id-ja megegyezik a felhasználó foglalásának id-jával.

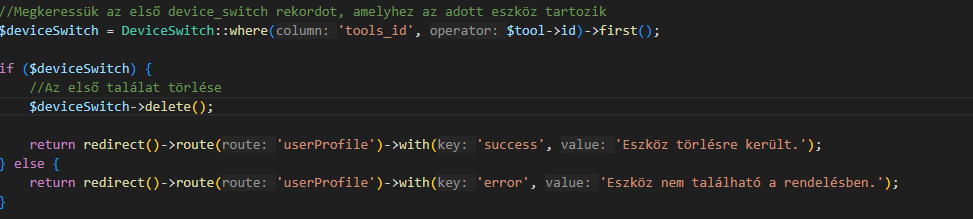
1. **Felhasználó ID lekérése:** Az auth()->id() segítségével lekérjük az aktuális felhasználó azonosítóját.
2. **Kölcsönzések lekérése:**
   * Az Loan modellből lekérjük a felhasználóhoz kapcsolódó összes kölcsönzést, beleértve a kapcsolódó adatokat:
     + deviceSwitches és azok tool kapcsolatai.
     + user és motorcycle adatok.
   * A lekérdezés eredményét átalakítjuk, hogy csak a szükséges adatokat tartalmazza:
     + Felhasználó neve, rendelés azonosítója, motor adatai (márka, típus), eszközök adatai (név, méret, csatlakozás dátuma), és a kölcsönzési időszak.
3. **Elérhető eszközök lekérése:** Lekérjük az összes eszközt a Tool modellből.
4. **Eszközök méreteinek csoportosítása:** Az eszközöket név szerint csoportosítjuk, és a méreteket kilistázzuk.
5. **Nézet visszatérése:** A userProfile nézetnek átadjuk a kölcsönzéseket, az elérhető eszközöket, és azok csoportosított méreteit (loans, availableTools, availableSizesByTool).

|  |  |  |
| --- | --- | --- |
| Field | Source | Format/Example |
| user\_name | $loan->user->name | "John Doe" |
| orders\_id | $loan->orders\_id | "ORD-ABC123" |
| motorcycle | $loan->motorcycle | { brand: "Yamaha", type: "MT-07" } |
| tools | $loan->deviceSwitches | Array of { tool\_name: "Helmet", size: "M" } |
| rental\_period | rentalDate/returnDate | { rentalDate: "2025-05-01", returnDate: "2025-05-05" } |
| created\_at | Loan creation timestamp | "2025-04-30 14:30:00" |

Megjelenítés mellett tudjuk törölni a foglalást, módosítani és törölni az eszközöket. Ezekre külön-külön function-t alkalmazunk, amelyeknek az elérési útvonala:



## Eszköz törlése



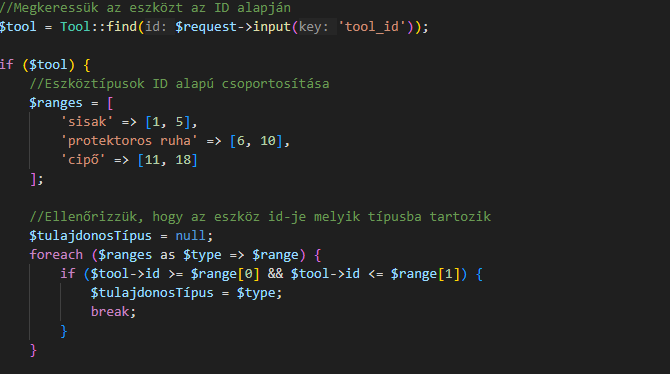
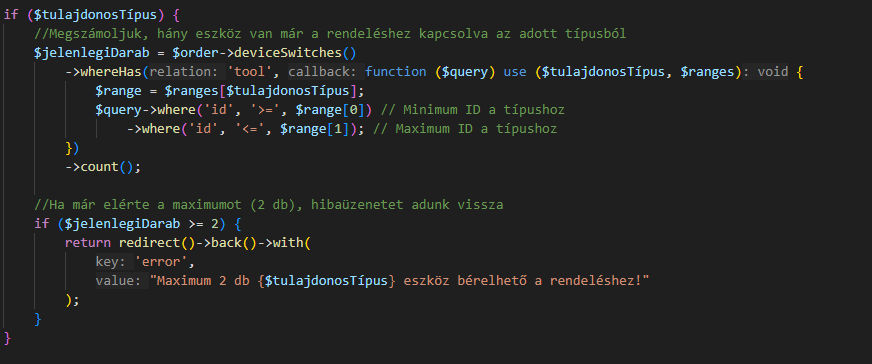
1. **Eszköz keresése:** A DeviceSwitch modellből megkeressük az első rekordot, amely az adott eszközhöz tartozik (tools\_id alapján).
2. **Eszköz törlése:**
   * Ha találunk egy megfelelő rekordot, azt töröljük az adatbázisból.
   * Ezután visszairányítjuk a felhasználót a userProfile oldalra egy sikerüzenettel.
3. **Hiba kezelése:**
   * Ha nem találunk megfelelő rekordot, akkor egy hibauzenettel térünk vissza a userProfile oldalra.

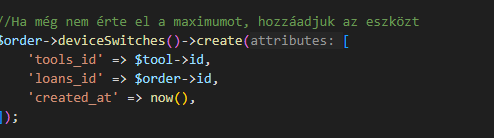
|  |  |
| --- | --- |
| Component | Description |
| $tool | Injected Tool model instance (target tool to remove). |
| DeviceSwitch::where() | Finds the first device\_switches record linked to the tool. |
| delete() | Removes the pivot record (does not delete the tool itself). |

## **Eszköz hozzáadása**

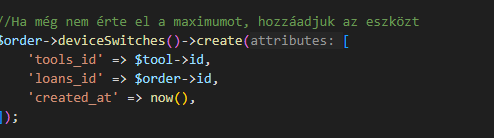
1. **Rendelés keresése:** A Loan modellből lekérjük az első rekordot az orders\_id alapján.



1. **Eszköz ellenőrzése:**
   * Megkeressük az eszközt az adatbázisban az tool\_id alapján.
   * Ha az eszköz létezik, az ID-ja alapján meghatározzuk, hogy melyik eszköz típusba tartozik (sisak, protektoros ruha, cipő).
2. **Eszköz darabszámának ellenőrzése:**
   * Ellenőrizzük, hogy az adott típusú eszközből hány darab kapcsolódik már a rendeléshez.
   * Ha már elérte a megengedett maximumot (2 db), akkor hibát jelzünk.
3. **Eszköz hozzáadása:**
   * Ha még nincs elérve a limit, az új eszközt hozzáadjuk a rendeléshez a deviceSwitches táblában.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter | Type | Example | Source | Description |
| ordersId | Integer | 1 | URL parameter | Unique order identifier. |
| tool\_id | Integer | 3 | Request input | ID of the tool to add. |



## **Rendelés törlése**

1. **Rendelés keresése:** Az orders\_id alapján megkeressük az adott rendelést a Loan modellből.



1. **Kapcsolódó eszközök törlése:**
   * A rendeléshez kapcsolódó összes eszközt (deviceSwitches) töröljük.



1. **Rendelés törlése:**
   * A rendelést (order) is töröljük az adatbázisból.



|  |  |  |
| --- | --- | --- |
| Operation | SQL Equivalent | Effect |
| deviceSwitches()->delete() | DELETE FROM device\_switches WHERE loans\_id=? | Deletes all linked tools. |
| order->delete() | DELETE FROM loans WHERE id=? | Permanently deletes the loan order. |

## Weboldal Tesztelés

## Regisztráció:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| S. No | Action | Inputs | Expected Output | Actual Output | Test Browser | Test Result | Test Comments |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Fill out registration form | Name: John Doe, Email: john.doe@example.com, Password: password123, Phone: 123456789 | Form fields are correctly filled. | Form fields are correctly filled. | Chrome | Pass | Form data accepted. |
| 2 | Submit the registration form | Click on the "Register" button | User registered successfully in the system. | User registered successfully in the system. | Chrome | Pass | Registration completed successfully. |
| 3 | Verify database entry | Check database for user entry | User data exists in the database. | User data exists in the database. | N/A | Pass | User record saved correctly in the database. |
| 4 | Verify file uploads | Check the storage folder for uploaded driving license images. | Files are saved in the storage directory. | Files are saved in the storage directory. | N/A | Pass | Driving license images uploaded successfully. |

*Bérlés hozzáadása:*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| S. No | Action | Inputs | Expected Output | Actual Output | Test Browser | Test Result | Test Comments |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Create a loan record in the database | users\_id: 1, motorcycles\_id: 2, rentalDate: '2025-01-01', returnDate: '2025-01-05', other metadata | Loan record is created successfully in the database. | Loan record is created successfully in the database. | N/A | Pass | Loan record created with the provided details. |
| 2 | Create a device switch record | loans\_id: Generated loan ID, tools\_id: 1 | Device switch record is created successfully. | Device switch record is created successfully. | N/A | Pass | Device switch linked to the created loan record. |
| 3 | Verify loan record | Assert database has a record with id: Loan ID and users\_id: 1 | Loan record exists in the database. | Loan record exists in the database. | N/A | Pass | Loan record verified successfully in the database. |
| 4 | Verify device switch record | Assert database has a record with loans\_id: Loan ID and tools\_id: 1 | Device switch record exists in the database. | Device switch record exists in the database. | N/A | Pass | Device switch record verified successfully. |



### *Felhasználó profil módosítás:*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| S. No | Action | Inputs | Expected Output | Actual Output | Test Browser | Test Result | Test Comments |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Fetch user with ID 2 | User ID: 2 | User is successfully fetched from the database. | User is successfully fetched from the database. | N/A | Pass | User with ID 2 exists in the database. |
| 2 | Simulate authentication | Acting as the user with ID 2 | User is authenticated successfully. | User is authenticated successfully. | N/A | Pass | Authentication simulated successfully. |
| 3 | Mock storage for file uploads | Mock storage for public disk | Storage is mocked successfully. | Storage is mocked successfully. | N/A | Pass | Storage for file uploads prepared. |
| 4 | Prepare request data | name: "Új Név", email: "uj@example.com", phoneNumber: "222222222", file inputs for license images. | Request data is prepared successfully. | Request data is prepared successfully. | N/A | Pass | Request created with updated user data. |
| 5 | Execute update method | Call UserController::update with prepared request. | Controller processes the update request successfully. | Controller processes the update request successfully. | N/A | Pass | User update logic executed successfully. |
| 6 | Verify updated user data | Assert database for updated values: name, email, phoneNumber. | User data in the database matches updated values. | User data in the database matches updated values. | N/A | Pass | Database reflects updated user details. |



### *Eszköz hozzáadása a foglaláshoz:*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| S. No | Action | Inputs | Expected Output | Actual Output | Test Browser | Test Result | Test Comments |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Fetch loan order with ID 1 | Loan ID: 1 | Loan order is successfully fetched from the database. | Loan order is successfully fetched from the database. | N/A | Pass | Loan order is available for tool addition. |
| 2 | Validate loan order exists | Assert non-null loan order | Loan order exists in the database. | Loan order exists in the database. | N/A | Pass | Loan order validation successful. |
| 3 | Fetch tools with IDs 1, 6, 11 | Tool IDs: 1 (helmet), 6 (protector suit), 11 (shoes) | Tools are successfully fetched from the database. | Tools are successfully fetched from the database. | N/A | Pass | All required tools exist. |
| 4 | Validate tool existence | Assert non-null tools for IDs 1, 6, 11 | Tools exist in the database. | Tools exist in the database. | N/A | Pass | All tools validated for addition. |
| 5 | Add each tool to the loan order | Post request to addToolToOrder route with tool IDs 1, 6, and 11. | Each tool is linked to the loan order in the database. | Each tool is linked to the loan order in the database. | N/A | Pass | Tools added successfully. |
| 6 | Verify database updates | Assert device\_switches contains records with loans\_id=1 and respective tools\_id. | Records exist in device\_switches table. | Records exist in device\_switches table. | N/A | Pass | Database reflects tool additions. |
| 7 | Verify redirection and success message | Response redirect with session message: "Új eszköz hozzáadva a rendeléshez." | Redirection and success message confirmed. | Redirection and success message confirmed. | N/A | Pass | Proper feedback provided to the user. |



### *Eszköz törlése a foglalásból:*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| S. No | Action | Inputs | Expected Output | Actual Output | Test Browser | Test Result | Test Comments |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Fetch loan order with ID 2 | Loan ID: 2 | Loan order is successfully fetched from the database. | Loan order is successfully fetched from the database. | N/A | Pass | Loan order is available for tool deletion. |
| 2 | Fetch tool with ID 1 | Tool ID: 1 | Tool is successfully fetched from the database. | Tool is successfully fetched from the database. | N/A | Pass | Tool is available for deletion. |
| 3 | Fetch device\_switch linking tool to loan order | Tool ID: 1, Loan ID: 2 | A device\_switch record is found in the database. | A device\_switch record is found in the database. | N/A | Pass | Relationship between tool and loan verified. |
| 4 | Delete the device\_switch record | Perform delete() on the fetched device\_switch record. | The device\_switch record is removed from the database. | The device\_switch record is removed from the database. | N/A | Pass | Tool successfully removed from the loan. |
| 5 | Verify device\_switch record is deleted | Assert absence of record in device\_switches table. | The device\_switch record is no longer present. | The device\_switch record is no longer present. | N/A | Pass | Database reflects the deletion. |



### *Foglalás törlése:*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| S. No | Action | Inputs | Expected Output | Actual Output | Test Browser | Test Result | Test Comments |

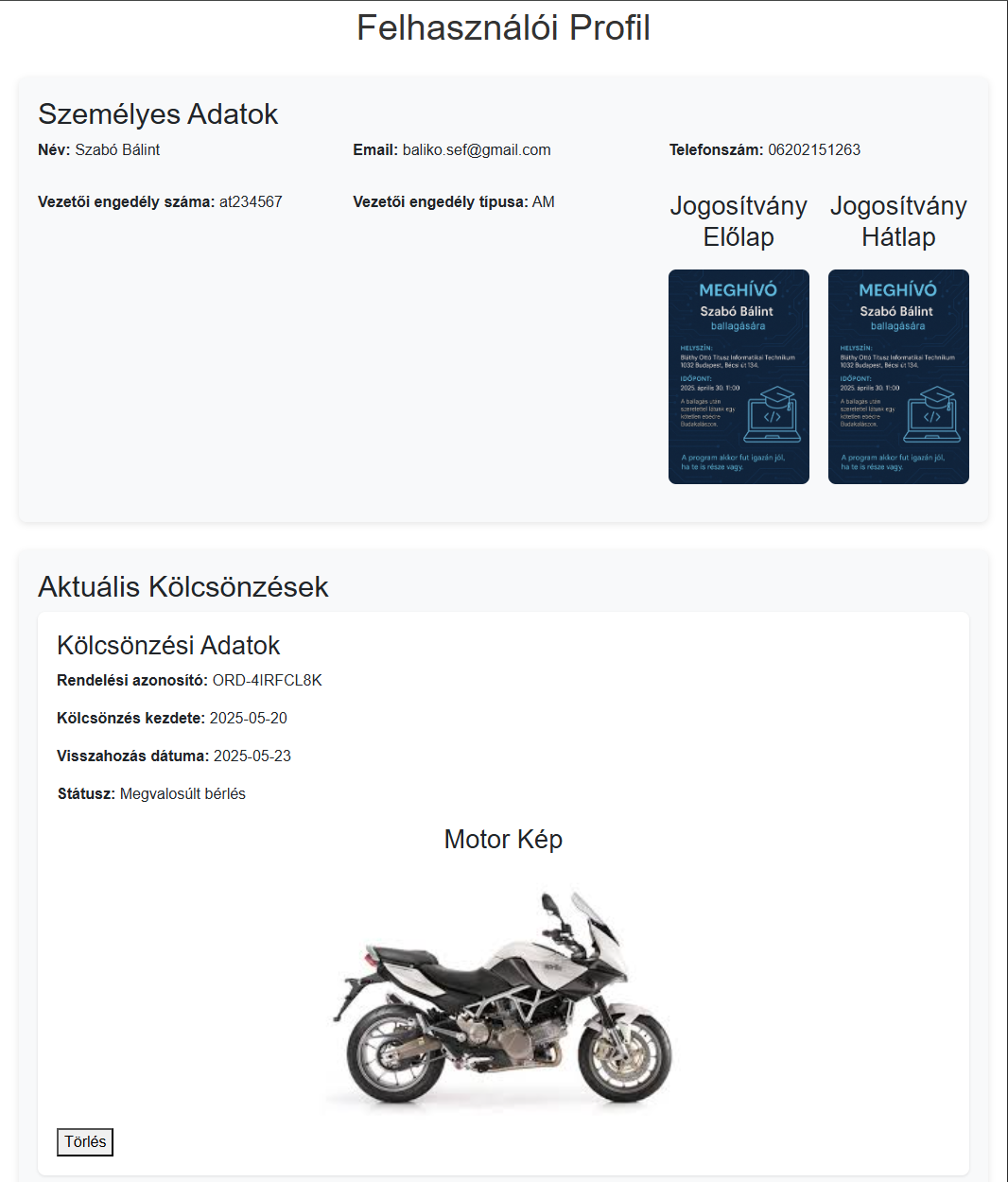
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Fetch loan order with ID 1 | Loan ID: 1 | Loan order is successfully fetched from the database. | Loan order is successfully fetched from the database. | N/A | Pass | Loan order exists for deletion. |
| 2 | Verify associated device\_switch records exist | Loan ID: 1 | Each device\_switch record exists in the database. | Each device\_switch record exists in the database. | N/A | Pass | Relationships verified. |
| 3 | Delete the loan order | Loan ID: 1 | Loan order is removed from the loans table. | Loan order is removed from the loans table. | N/A | Pass | Loan order deletion successful. |
| 4 | Verify associated device\_switch records deleted | Loan ID: 1 | All related device\_switch records are removed. | All related device\_switch records are removed. | N/A | Pass | Database reflects cascading deletions. |
| 5 | Verify redirection and success message | Route: /userProfile | User is redirected with a success message in the session. | User is redirected with a success message in the session. | N/A | Pass | User feedback provided. |



# Mobil application react-ba

A mobilos felületen az ügyfél láthatja az összes rendelését, és a még meg nem történt bérléseit is, amit törölhet, viszont, ha egy napon a rendelés kezdetéig akkor már nem tudja visszamondani a bérlést. A felhasználónak elsőnek be kell jelentkeznie email-jelszó párossal, majd ezután láthatja a saját rendeléseit.

Az oldal elején láthatóak az ügyfél adatai, mint például a telefonszáma vagy az email címe, ha lejjebb görget akkor meg láthatja a béreléseit ami tartalmazza például a rendelési azonosítóját.

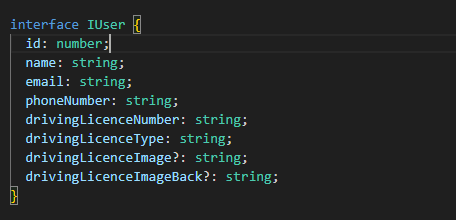


Motorkölcsönzésre nincsen MÉG lehetőség, de megtekinthetőek az elérhető motorok “Foglalj most!” gombra kattintva, mint egy katalógus.

Későbbiekben ezekhez hozzá lehet tenni bérlési funkciókat, de ezek még jelenleg munkálatok alatt folynak.

## Működése:

A laravelben megírt funcionoket kértük le API segítségével. Ezeket kezdtük el alkalmazni a telefonos alkalmazásunkban. Így nem kellett újra az adatbázist feldolgozni a rendszerünkben, hanem “újra használjuk” a korábban használt műveleteket.

Elsősorban feldolgozzuk, hogy milyen adatokat szeretnénk tárolni a fetchelés során, például a felhasználói adatokat:

Itt megadjuk a rendszernek, hogy a felhasználó nevét, email címét és egyéb adatait szeretnénk később alkalmazni.

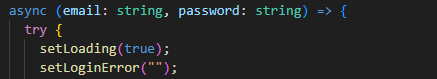
Ezt következően készül el egy “function”, amelyben lekérjük api-n keresztűl a kívánt adatokat.

## Bejelentkezés:

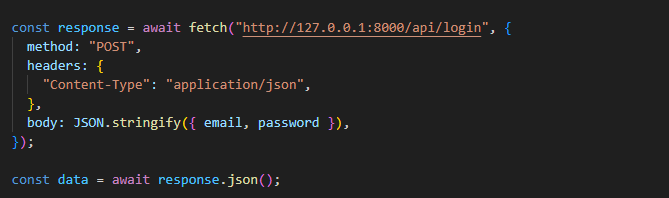
### Paraméterek:

* email (string): A felhasználó email címe.
* password (string): A felhasználó jelszava.

### Működés:

**Kezdő állapotok beállítása**: A loading állapot true-ra állítása és a loginError törlése, ha van.

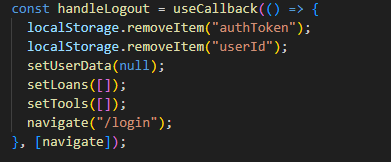
1. **Kérés küldése**: A POST kérést küldi az API-nak (http://127.0.0.1:8000/api/login), amely tartalmazza a felhasználó email címét és jelszavát JSON formátumban.



1. **Válasz feldolgozása**:
   * **Sikeres válasz**: Ha a válasz sikeres, akkor a választ JSON formátumban dolgozza fel, és a token és userId értékeket elmenti a localStorage-ba. Ezt követően lekéri a felhasználói adatokat és átirányítja a felhasználót a dashboard oldalra.

**Sikertelen válasz**: Ha a válasz nem sikeres, a hibaüzenetet jeleníti meg a felhasználónak.

Ezután a felhasználó megtudja tekinteni az adatait, a korábbi rendeléseit, és azokat törölheti, ha még a megadott időintervallumon belül van.

Ha bejelentkezett a felhasználó a bejelentkezés gomb helyére egy kijelentkezés gomb kerül, amelyre nyomva kilépteti a felhasználót.

## Kijelentkezés:

### Működés:

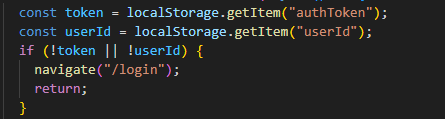
1. **Adatok eltávolítása a localStorage-ból**: A függvény eltávolítja a authToken és userId kulcsokat a localStorage-ból, hogy megszüntesse az autentikációs adatokat.
2. **Felhasználói adatok törlése**: A setUserData(null) hívásával törli a felhasználói adatokat.
3. **Kapcsolódó listák törlése**: A setLoans([]) és setTools([]) hívásokkal törli a hitelekkel és eszközökkel kapcsolatos adatokat.
4. **Navigáció**: A felhasználót átirányítja a /login oldalra a navigate("/login") segítségével.

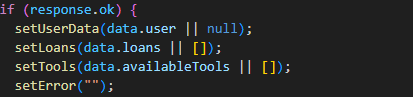
Korábban említve, a felhasználó bejelentkezést követően megtudja tekinteni az adatait, amelyet api-nak a segítségével kérünk le.

A bejelentkezéskor beléptettük a felhasználót és annak a profilnak az id-ját alkalmazzuk a következő lekéréskor.

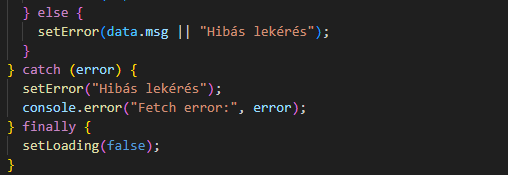
## Profilbetöltés

### Működés**:**

1. **Autentikációs adatok ellenőrzése**: A függvény lekéri a localStorage-ból az authToken és userId adatokat. Ha bármelyik hiányzik, átirányítja a felhasználót a bejelentkezési oldalra, ha a profil fülre kattint.
2. **API kérés küldése**: Ha a token és a felhasználói ID rendelkezésre áll, a függvény egy GET kérést küld az API-nak (http://localhost:8000/api/userProfile/${userId}), az autentikációs tokent az Authorization fejlécben küldve.
3. **Válasz feldolgozása**:

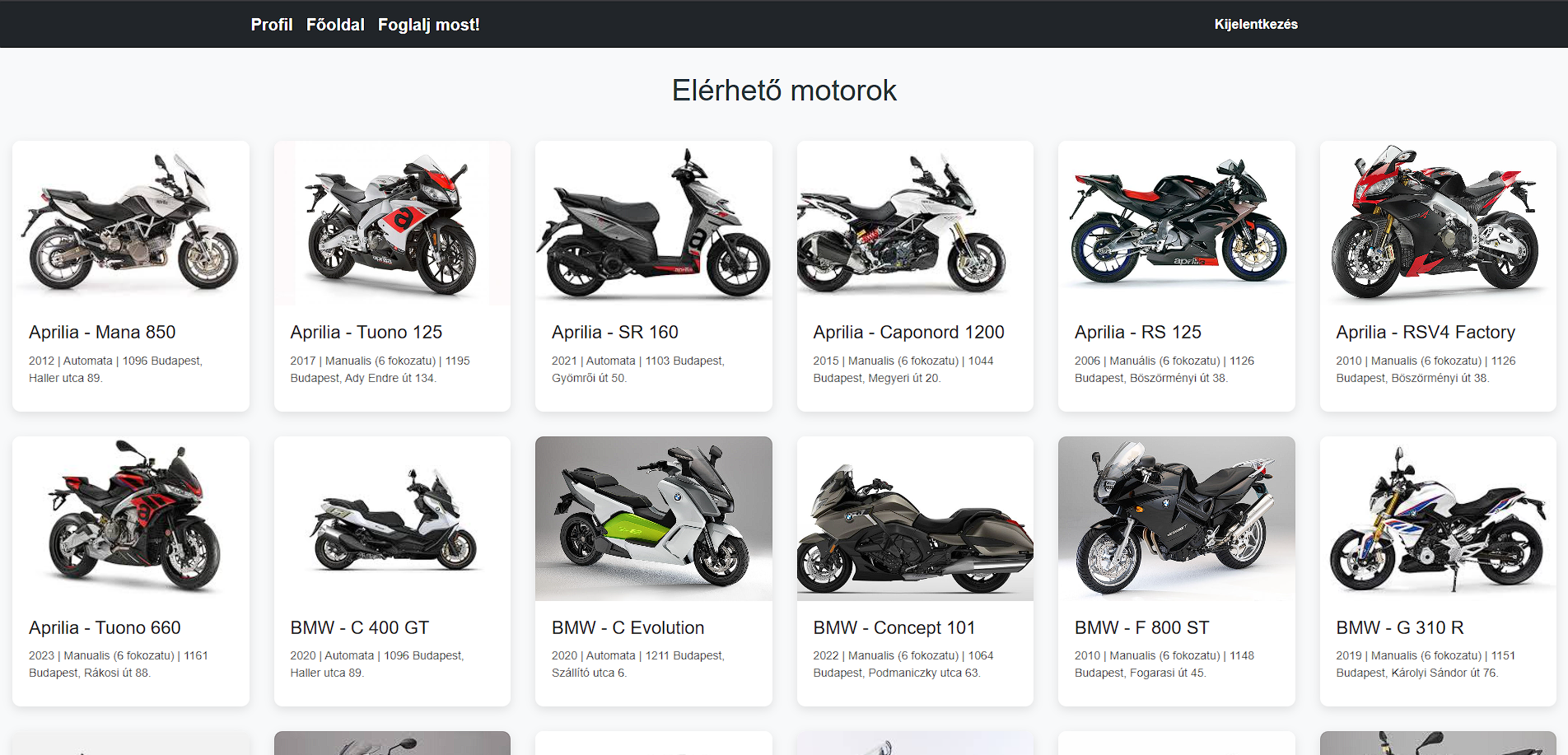
**Sikeres válasz**: Ha a válasz sikeres, a választ JSON formátumban dolgozza fel, és frissíti az alkalmazás állapotát a setUserData, setLoans és setTools hívásokkal. A hibát (error) nullázzák.

* + **Sikertelen válasz**: Ha a válasz nem sikeres, a hibaüzenetet a setError hívással tárolja.

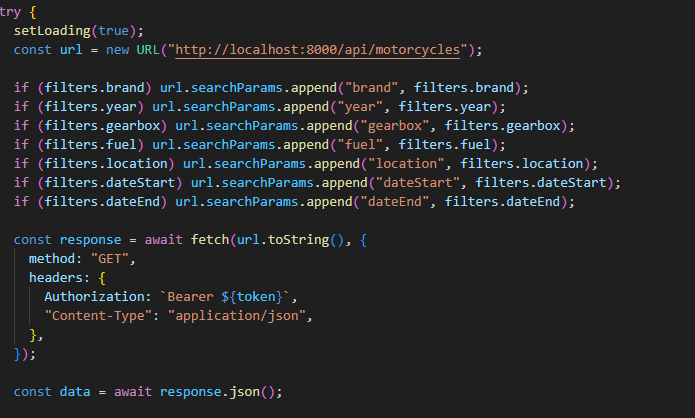
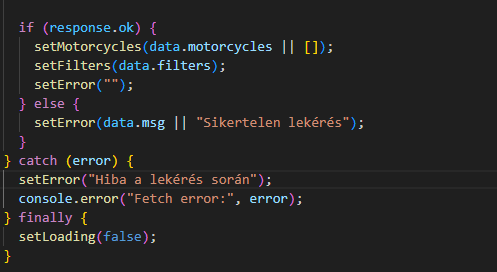


Itt még egyelőre a felhasználó semmi mást nem tud csinálni, csak a rendelés törlését elvégezni. Viszont további lehetőségekre is hagytunk helyet, megírva már a többi beépíthető function API kérését.

Ezen kívül a felhasználó meg tud tekinteni egy katalógust, amelyben a rendszerünkben lévő motorokat tudja megtekinteni.



#### Ennek a lekérése az alábbi ként működik:

1. **URL és szűrők generálása**: A függvény létrehoz egy URL-t a motorok lekéréséhez (http://localhost:8000/api/motorcycles), és hozzáadja a szűrőket a URL keresési paramétereihez, ha azok rendelkezésre állnak.
2. **API kérés küldése**: A függvény egy GET kérést küld az API-nak a generált URL-lel. Az autentikációs tokent a Authorization fejlécben küldi el.
3. **Válasz feldolgozása**:
   * **Sikeres válasz**: Ha a válasz sikeres, a választ JSON formátumban dolgozza fel, és beállítja a motorcycles adatokat, valamint a filters szűrőket. A hibát (error) nullázza.
   * **Sikertelen válasz**: Ha a válasz nem sikeres, a hibaüzenetet a setError hívással tárolja.

Motorkölcsönző weboldal Tartalom:

[Motorokról: 1](#_Toc196760020)

[Felhasználói adatok: 1](#_Toc196760021)

[Felhasználói felület: 2](#_Toc196760022)

[Adminisztrátori alkalmazás 3](#_Toc196760023)

[Felhasználói weboldal 8](#_Toc196760024)

[Kölcsönzés működése 8](#_Toc196760025)

[Regisztráció 8](#_Toc196760026)

[Megerősítés 8](#_Toc196760027)

[Bejelentkezés 8](#_Toc196760028)

[Profil 9](#_Toc196760029)

[A működés 9](#_Toc196760030)

[Regisztráció 9](#_Toc196760031)

[Bérlés 10](#_Toc196760032)

[A szűrés működése: 11](#_Toc196760033)

[Rendelési előzmények 18](#_Toc196760034)

[Eszköz törlése 20](#_Toc196760035)

[Eszköz hozzáadása 20](#_Toc196760036)

[Rendelés törlése 22](#_Toc196760037)

[Weboldal Tesztelés 22](#_Toc196760038)

[Regisztráció: 22](#_Toc196760039)

[Bérlés hozzáadása: 24](#_Toc196760040)

[Felhasználó profil módosítás: 25](#_Toc196760041)

[Eszköz hozzáadása a foglaláshoz: 27](#_Toc196760042)

[Eszköz törlése a foglalásból: 29](#_Toc196760043)

[Foglalás törlése: 30](#_Toc196760044)

[Mobil application react-ba 31](#_Toc196760045)

[Működése: 32](#_Toc196760046)

[Bejelentkezés: 33](#_Toc196760047)

[Paraméterek: 33](#_Toc196760048)

[Működés: 33](#_Toc196760049)

[Kijelentkezés: 35](#_Toc196760050)

[Működés: 35](#_Toc196760051)

[Profilbetöltés 35](#_Toc196760052)

[Működés**:** 35](#_Toc196760053)

[Ennek a lekérése az alábbi ként működik: 37](#_Toc196760054)