

# Prooktatás PHP Vizsgamunka

## Farkas Viktória

# Dokumentáció

### 1. Terv és az ötlet

A terv egy olyan adatbázis létrehozása, amit admin felület segítségével lehet módosítani, adatokat törölni és felvinni.

Több ötlet is volt ennek a megvalósítására, az egyik a dolgozók munkaruha kiadásának és használatának adminisztrációja de végül egy élményvezetést nyújtó szolgáltatás honlapját és admin felületét terveztem meg. Az ötlet onnan jött, hogy a páromat megleptük születésnapjára egy élményvezetéssel ahol több autóval is lehetett menni. Ezt egyszerűsítve egy pályára szűkítettem, méghozzá a Hungaroringre akik - hogy ne álljon üresen a pálya, amikor nem F-1 és egyéb versenyek mennek - kiadnák alvállalkozóknak a területet, hogy bizonyos dátumokon, mint egy autós találkozó lehetne menni az autókkal pár kört. Ők lettek a szuper kreatív nevű cég, az Élményvezetés.hu Kft.

A honlapjukon megtekinthető dátum szerint, melyik autó lesz elérhető, vagy szűrhetnek a látogatók autókra kategória szerint (mint olasz autók, német, amerika, retro stb..). Ezen kívül megtekinthetik az adott autó leírását.

Ezeket az admin egy erre kialakított admin felületen módosíthatja.

A fent leírtakon kívül, egy kezdetleges, de piacképes alap weboldal létrehozását is célként tűztem ki, hogy vizuálisan is visszadj a elképzelésemet.

### 2. Cél

A vizsgamunka célja(nem sorrendben):

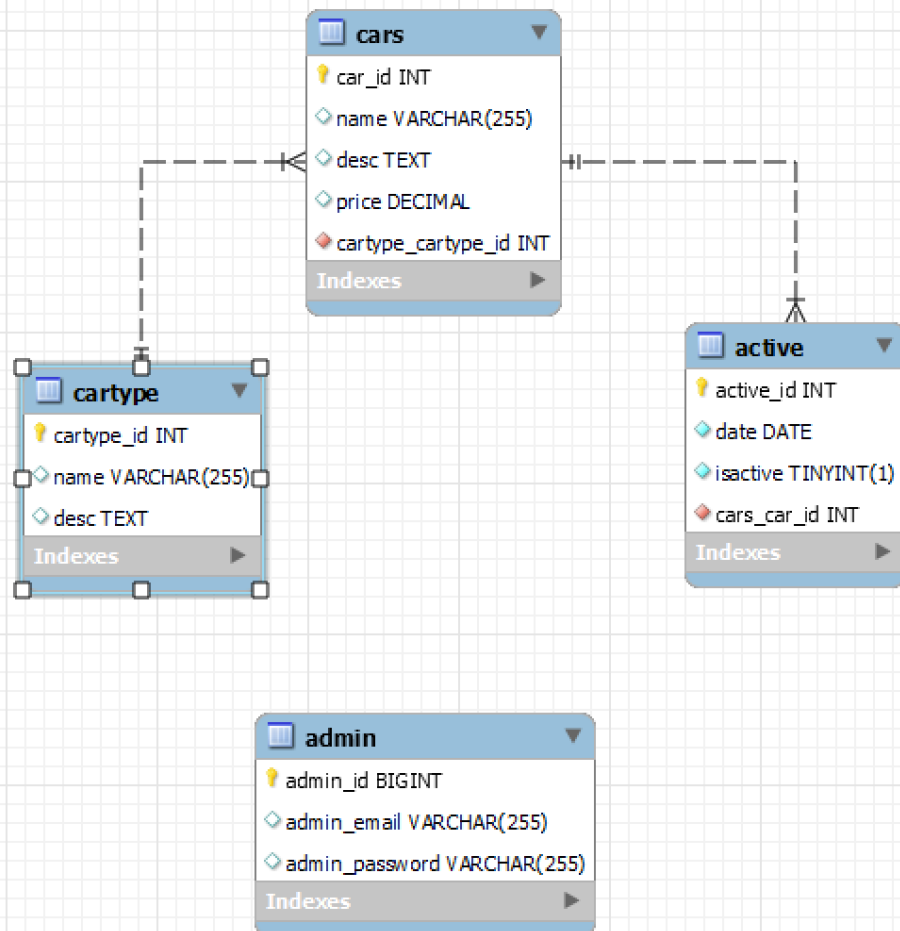
- MySQL adatbázis létrehozása, a szükséges adatok lekérdezése, megjelenítése,
- adatbázis összekötése PHP-val,
- admin belépő és jelszó létrehozása,
- admin felület létrehozása,
- adatbázis adatainak törlése, módosítása, új adat felvételének lehetősége az admin felületen keresztül,
- a látogatói oldalon is látszódnak ezek a változások,
- mellékes, de front-end oldalról is passzoló oldal létrehozása

### 3. Adatbázis

**Adatbázis neve: vezetes**

Táblák:

- active - dátum, amikor megrendezésre kerül egy találkozó, isactive: igen vagy nem, tehát be lehet állítani hogy az adott dátumon az autó ott lesz-e vagy sem, cars\_car\_id foreign keyel van összekötve hogy autókat lehessen hozzárendelni ehhez a táblához.
- admin - email és hashelt jelszó
- cars - autók "neve", leírása, ára és a cartype-al való foreign key kapcsolja össze a kategóriát
- cartype - kategória neve és leírása(üres)

**4. Létrehozott fájlok**

- [nav.html](#)
- modify.php
- modify-date.php
- nav.php
- add-car.php
- add-car-2.php
- add-cars.php
- admin.php
- add-date.php
- admin-car.php
- admin-date.php
- cars.php
- date.php
- delete.php
- delete-date.php
- footer.php
- login.php
- main-date.php
- main-cars.php
- main-admin.php
- main.php
- [top.js](#)
- [admin-load-page.js](#)
- [form-style.css](#)
- [grid-2.css](#)
- [main-style.css](#)
- [main-admin-style.css](#)
- [nav-style.css](#)
- [reset.css](#)
- [adatbázis: vezetes](#)

## 5. Oldal bemutatása, mi hol található

<http://localhost/phpvizsga/main.php> - **Ez a főoldal.**

Itt a látogató a ha legörget a *“tekintse meg autóinkat”* résznél kiválaszthatja, hogy kategóriák vagy dátum alapján szeretné megnézni a kínálatot. Ugyanezt el lehet érni a menüből az **Autók** legördülő listából.

**Menü részei: Főoldal - Autók - Admin**

Az oldal lentebbi része főleg front-end, itt látszanak a vezetésen készült képek, egy bemutatkozó szöveg és a pályáról egy carousel, majd a lábléc.

JS-el létrehoztam egy gombot, amivel az oldal tetejére vissza lehet ugrani.

**A két fő kategória:**

- **Kategória szerint:** <http://localhost/phpvizsga/main-cars.php>

Kategóriák szerint vannak az autók kategorizálva:

```
"SELECT cars.*, cartype.name AS category_name FROM cars
      JOIN      cartype      ON      cars.cartype_cartype_id =
cartype.cartype_id
      ORDER BY cartype.name";
```

- **Dátum szerint:** <http://localhost/phpvizsga/main-date.php>

Dátum szerint vannak az autók kategorizálva:

```
$sql = "SELECT cars.name AS car_name, cars.desc AS
car_desc, cars.price AS car_price, active.date AS
active_date, active.isactive AS isactive
      FROM active
      JOIN cars ON active.cars_car_id = cars.car_id
      JOIN cartype ON cars.cartype_cartype_id =
cartype.cartype_id
      ORDER BY active_date, cars.name";
```

A menüből érhető el az **Admin** felület is. A bejelentkezéshez az alábbi felhasználónevet és jelszót hoztam létre:

```
/*      // TEST 2 - ADD USER - működik
echo $_ADM->save("phpoktatas@vizsga.hu", "teszt123")
? "OK" : $_ADM->error; */

/*      // TEST 2 - GET USER - működik
print_r($_ADM->get(1)); */

// TEST - VERIFY USER LOGIN
/*      echo $_ADM->verify("phpoktatas@vizsga.hu", "teszt123")
? "OK" : $_ADM->error;
print_r($_SESSION); */
```

**Felhasználónév:** [phpoktatas@vizsga.hu](mailto:phpoktatas@vizsga.hu)

**jelszó:** teszt123

Innen vissza is lehet lépni a főoldalra.

**Bejelentkezés** után az admin felület főoldalán kiírtam, az adott bejelentkező email címét.

Bal oldalon a **navigációs menüből a Főoldalon kívül megtalálható az Autók és Dátum szűrési csoportok.**

JS-el úgy oldottam meg, hogy az *Autók* és *Dátum* ne új ablakban nyíljanak meg, hanem a "fő részben".

Mind a két csoportnál (*Autók és Dátum*) ugyanazokat a funkciókat hajthatjuk végre:

- bel felső sarokból új elemeket adhatunk hozzá az adatbázishoz.
- Jobb oldalon a sorok végén az adott sorra vonatkozóan tudunk módosítani és törölni.

Részletek:

- az *Autók* új hozzáadása pontnál új adatot és kategóriát is hozzá tudunk rendelni, ezek az `add-cars.php`-ban összegyűjtve jelennek meg, de ez két külön fájl, az `add-car.php` és `add-car-2.php`.
- a *Dátum* hozzáadása opciónál az autó fajtájából (checkbox) többet is kiválaszthatunk egyszerre, ezeket mind hozzá fogja adni.
- a szerkesztésnél mindkét esetben úgy jelennek meg az adatok hogy a kiválasztott sorban lévő (autó márkája, leírás, ár stb) megjelenik a módosító mezőben, tehát nem üres mezők szerepelnek, amely segíti az adminisztrátort.
- A törlésen kívül minden módosításról és új adat hozzáadásáról alert-et kapunk akár sikeres akár nem,
- törlésnél nem kérdezi meg, hogy valóban törölni akarunk-e. Ha rákattintunk és sikeres, rögtön törölni fogja, de erről egy új oldalon kapunk üzenetet

A **kijelentkezés** gombbal a login felületre mehetünk vissza.

Ezek az adatok (módosított, törölt, új) mind megjelenik az admin és a fő látogatói felületen is.

## 6. Fájlok összeköttetése, feladatuk, MySQL szűrések

Fájl neve	Feladata, leírása
modify.php	Az <i>Autók</i> menünél módosítható a kiválasztott sort, amely az ahhoz kapcsolódó adatokat felülírja a <i>cars</i> táblábanl.
modify-date.php	Az <i>Dátum</i> menünél módosítható a kiválasztott sort, amely az

	ahhoz kapcsolódó adatokat felülírja a <i>cars</i> , <i>cartype</i> , <i>active</i> táblákban.
nav.php	A <i>main-cars.php</i> és a <i>main-date.php</i> fájlokat tartalmazó Menü sor
add-car.php	SQL lekérdezés, egy új adatsor hozzáadásához, mint autó neve, leírása, ára és kategóriája (a <i>cars</i> táblába)
add-car-2.php	SQL lekérdezés, egy új kategória hozzáadásához (a <i>cartype</i> táblába)
add-cars.php	Az <i>add-car.php</i> és az <i>add-car-2.php</i> fájlokat összesíti
admin.php	Ez a kód egy <i>Admin</i> osztályt definiál. Az admin felületre történő belépéshez felhasználónevet és jelszót generáltunk, mentjük és ellenőrizzük a belépéshez.
add-date.php	SQL lekérdezés, egy új adatsor hozzáadásához, amely lehetővé teszi, hogy kiválasszuk hogy az adott autó mikor elérhető (a <i>active</i> táblába). Több autó fajtát is kiválaszthatunk egyszerre egy dátum-feltétel(elérhető-e?) párosításhoz
admin-car.php	Hasonló I, mint a <i>main-car.php</i> , de eltérő olyan szempontból hogy az adminnak szükséges infók jelennek meg benne
admin-date.php	Hasonló, mint a <i>main-date.php</i> , de eltérő olyan szempontból hogy az adminnak szükséges infók jelennek meg benne
cars.php	SQL lekérdezés, amely táblázatba rendezi különböző feltételek szerint a dátum ( <i>active</i> ), kategória( <i>cartype</i> ) és autó( <i>cars</i> ) kapcsolattal kialakított adatokat. Autó kategóriája szerinti csoportosítás
date.php	SQL lekérdezés, amely táblázatba rendezi különböző feltételek szerint a dátum ( <i>active</i> ), kategória( <i>cartype</i> ) és autó( <i>cars</i> ) kapcsolattal kialakított adatokat. Dátum szerinti csoportosítás
delete.php	Az <i>Autók</i> menüből törli a kiválasztott sort, amely az ahhoz kapcsolódó adatokat törli a <i>cars</i> táblából.
delete-date.php	A <i>Dátum</i> menüből törli a kiválasztott sort, amely az ahhoz kapcsolódó adatokat törli a <i>cars</i> , <i>cartype</i> , <i>active</i> táblából.
login.php	A belépéshez szükséges form és az <i>admin.php</i> és <i>main-admin.php</i> fájlokat tartalmazó fájl. Itt tudunk belépni az admin felületre, az résztvevő fájlok segítségével.
main-date.php	Ez a fájl tartalmazza a <i>nav.php</i> , <i>date.php</i> és a <i>footer.php</i> fájlokat, amelyekkel a látogatói oldalt hoztam létre.

main-cars.php	Ez a fájl tartalmazza a <i>nav.php</i> , <i>cars.php</i> és a <i>footer.php</i> fájlokat, amelyekkel a látogatói oldalt hoztam létre.
main-admin.php	Ez a fájl tartalmazza a <i>admin.php</i> , <i>login.php</i> , <i>admin-cars.php</i> és <i>admin-date.php</i> fájlokat, amelyekkel az admin felület főoldala, a be- és kijelentkezés lehetősége és a menük kiválasztása történik.
main.php	A home page jelenik meg ebben a fájlban. A következő PHP fájlokat tartalmazza: <i>nav.php</i> <i>main-cars.php</i> <i>main-date.php</i> és a <i>footer.php</i>

## 7. Fejlesztés:

Tesztelés közben és logikusan végiggondolva az alábbi lehetőségeket lehet még fejleszteni (nyilván van még más is, de ezek amik szembeötlöttek vagy egyértelműek számomra):

- responzivitás; bár nagy része BS-el van elkészítve, mégis bele kellett nyúlni. Ennek sajnos a responzivitás látta kárát.
- Front-end részről: mobil nézet menüben az aláhúzó csíkokat megregulázni :), a carousel rossz irányba megy
- \$\_get - \$\_postra variálni,
- az adatbázis elérési adatait egy konfigurációs fájlban jobb lenne tárolni, és hivatkozni rá a kódban.
- Az SQL injekció ellen nincs védve, célszerűbb lenne például előre elkészített statementeket vagy adatbázis osztályokat alkalmazni (kivéve admin.php).
- A *kategória* nem jelenik meg a vizuális felületeken, ha új adatot rögzítünk - csak az adatbázisban látszódik.
- ha a beírandó adatok közül valamit kihagyunk nem jelez hibaüzenetet. Tehát hagyhatunk üresen mezőket és ezeket így fogja az adatbázisban is tárolni. Ez nem feltétlenül baj, de célszerű lenne minden mezőt kitölteni.
- nem ellenőrzi hogy az adott adat kombináció szerepel-e már a sorban, vagy az adott pl autóhoz hozzárendelt adatok az új adatokkal nem egyeztethetőek össze.

## 8. Összegzés

Az általam készített oldal egy élményvezetést szolgáltató cégre szabott adminisztrációs felületet biztosít, ahol az adminisztrátorok bejelentkezve adatokat tudnak módosítani, törölni és új adatokat hozzáadni. Az adminisztrátoroknak

lehetőségük van a szükséges adatok rögzítésére (pl. az autó típusa, leírása, bérleti díja, kategóriája, elérhetőségi dátuma).

Az adminisztrációs felület szerepe, hogy megkönnyítse az Élmenyvezetés.hu Kft-nél dolgozók munkáját, lehetővé téve számukra, hogy gyorsan és hatékonyan kezeljék az informatív adatok szerkesztését.