

# SZAKDOLGOZAT

BKSZC WEISS MANFRÉD TECHNIKUM, SZAKKÉPZŐ  
ISKOLA ÉS KOLLÉGIUM

greencarsandparts.hu

Témavezető:

Kovács László Bálint

Készítette:

Szítás Péter

szoftverfejlesztő

BUDAPEST

2024

## Tartalom

<b>Elektromos jármű hirdetések weboldal dokumentáció .....</b>	<b>5</b>
<b>1. Bevezetés .....</b>	<b>5</b>
1.1. A téma ismertetése .....	5
1.2. Témaválasztás indoklása .....	5
1.3. Szakmai célkitűzések .....	5
<b>2. Fejlesztői dokumentáció .....</b>	<b>6</b>
2.1. Fejlesztői környezet és eszközök .....	6
Fejlesztői Hardware / hardver (részlet).....	6
2.2. Adatmodell és adatbázis tervezése .....	6
Brands tábla .....	6
Users tábla.....	7
2.3. Alkalmazott technológiák.....	8
2.4. Implementációs folyamat.....	8
2.5. Fontosabb algoritmusok .....	8
Session kezelés.....	8
Adatbázis kapcsolódás .....	9
Adatlekérdezés és megjelenítés .....	9
Dinamikus tartalom megjelenítése.....	9
Képek kezelése.....	10
JavaScript a képgaléria kezelésére .....	10
felhasználói jogosultság ellenőrzése .....	10
hirdetés feladása.....	10
Képfeltöltés kezelése .....	11
Sikeres hirdetésfeladás kezelése .....	11
Hiba kezelése .....	11
Felhasználó bejelentkezésének ellenőrzése .....	11
Űrlap hirdetés feladásához .....	12
Fontos input mezők.....	12
Képek feltöltése .....	12
Márkák dinamikus listája .....	13
2.6. Tesztelési stratégia.....	19
<b>3. Felhasználói dokumentáció.....</b>	<b>20</b>
3.1. A rendszer áttekintése .....	20
Főbb funkciók és felépítés .....	20
3.2. Rendszerkövetelmények .....	20

Hardverkövetelmények .....	20
Szoftverkövetelmények.....	21
Adatbázis követelmények .....	21
Funkcionális követelmények .....	21
Nem Funkcionális Követelmények .....	21
Dokumentáció és Támogatás .....	21
<b>3.3 Telepítés és indítás.....</b>	<b>21</b>
<b>3.4 A rendszer használata .....</b>	<b>22</b>
Regisztráció és Bejelentkezés .....	22
Hirdetések Kezelése .....	23
Keresés .....	23
Profil Szerkesztése .....	23
Kijelentkezés.....	24
<b>3.5 Gyakori problémák és megoldásaik.....</b>	<b>24</b>
1. Bejelentkezési hibák .....	24
2. Képfeltöltési problémák .....	24
3. Hirdetések frissítési/szerkesztési hibái.....	24
4. Lassú oldalbetöltési sebesség.....	25
<b>3.6 Jövőbeli Fejlesztések .....</b>	<b>25</b>
<b>4. Összefoglalás .....</b>	<b>26</b>
<b>5. Forráskód és dokumentáció .....</b>	<b>27</b>
<b>6. Irodalomjegyzék és források .....</b>	<b>28</b>

# 1. Bevezetés

## 1.1. A téma ismertetése

Ebben a projektben egy olyan weboldalt hoztam létre, amely összeköti az elektromos járművek eladóit a potenciális vevőkkel. A cél egy olyan közösségi piactér létrehozása volt, ahol a felhasználók könnyedén böngészhetik a hirdetéseket, kapcsolatba léphetnek az eladókkal, és információkat cserélhetnek.

## 1.2. Témaválasztás indoklása

Az elektromos járművek piaca gyorsan bővül, de még mindig hiányoznak a szükséges online piacterek, amelyek kifejezetten erre a célközönségre szabottak. E hiány felismerésével döntöttem úgy, hogy létrehozok egy dedikált platformot, amely elősegíti az elektromos járművek adás-vételét.

## 1.3. Szakmai célkitűzések

Célom egy intuitív, felhasználóbarát weboldal létrehozása volt, amely előtérbe helyezi az elektromos járművekkel kapcsolatos hirdetéseket. Fontosnak tartottam, hogy a platform kényelmes és biztonságos legyen a felhasználók számára, valamint hogy hozzájáruljon az elektromos járművek piacának növekedéséhez.

## 2. Fejlesztői dokumentáció

### 2.1. Fejlesztői környezet és eszközök

A fejlesztéshez a következő eszközöket és technológiákat használtam:



- **Visual Studio Code** mint fejlesztői környezet a kód szerkesztéséhez.
- **PHP** a szerveroldali logika implementálásához.
- **MySQL** az adatok tárolásához.

Fejlesztői Hardware / hardver (részlet)

**Processzor:** AMD Ryzen 5 3550H  
**Memória mérete:** 8,00 GB  
**Rendszer típusa:** 64 bites Windows 10 operációs rendszer, x64-alapú processzor  
**Videókártya neve:** NVIDIA GeForce GTX 1650

### 2.2. Adatmodell és adatbázis tervezése

Az adatmodell kialakításához az alábbi entitásokat azonosítottam és modelleztem: Adatbázis Tervezés és Implementáció

Advertisements tábla

- **Leírás:** A **Advertisements** tábla tartalmazza a hirdetésekkel kapcsolatos alapinformációkat. Ez az alapja minden hirdetésnek az oldalon.
- **Mezők:**
  - **id:** Az egyedi azonosító minden hirdetés számára. Típus: INT.
  - **title:** A hirdetés címe. Típus: VARCHAR(255).
  - **description:** A hirdetett termék vagy szolgáltatás részletes leírása. Típus: TEXT.
  - **type:** A hirdetés típusa, amely jelen esetben 'car' vagy 'battery' lehet. Típus: ENUM('car','battery').
  - **user\_id:** A hirdetés feladójának felhasználói azonosítója, kapcsolódik a **Users** táblához. Típus: INT.
  - **brand:** A hirdetett termék márkaneve. Típus: VARCHAR(255).

id	: int(11)
title	: varchar(255)
description	: text
type	: enum('car','battery')
user_id	: int(11)
brand	: varchar(255)

Brands tábla

- **Leírás:** A **Brands** tábla tartalmazza azokat a márkaneveket, amelyek az **Advertisements** táblában hirdetett járművekhez kapcsolódnak.

- **Mezők:**

- **id:** Az egyedi azonosító minden márka számára. Típus: INT.
- **name:** A márkanév. Típus: VARCHAR(255).

elektromos_jarmuvek brands	
id	int(11)
name	varchar(255)

#### Users tábla

- **Leírás:** A **Users** tábla a weboldal felhasználóinak adatait tartalmazza, beleértve a regisztráció során megadott információkat.

- **Mezők:**

- **id:** Felhasználói egyedi azonosító. Típus: INT.
- **fullname:** A felhasználó teljes neve. Típus: VARCHAR(255).
- **username:** A felhasználó által választott felhasználónév. Típus: VARCHAR(50).
- **email:** A felhasználó email címe. Típus: VARCHAR(100).
- **password:** A felhasználó jelszava, titkosítva tárolva. Típus: VARCHAR(255).

elektromos_jarmuvek users	
id	int(11)
fullname	varchar(255)
username	varchar(50)
email	varchar(100)
password	varchar(255)

- **Logs tábla:**

- **id:** Egyedi azonosító (int)
- **timestamp:** Naplózási időpont (timestamp)
- **user\_id:** Tevékenységet végző felhasználó azonosítója (int)
- **action:** Végrehajtott művelet leírása (varchar)

elektromos_jarmuvek logs	
id	int(11)
timestamp	timestamp
user_id	int(11)
action	varchar(255)

- **Images tábla:**

- **id:** Egyedi azonosító (int)
- **ad\_id:** Hozzá tartozó hirdetés azonosítója (int)
- **file\_path:** Képfájl elérési útvonala (varchar)

elektromos_jarmuvek images	
id	int(11)
ad_id	int(11)
file_path	varchar(255)

- **Profiles tábla:**

- **id:** Egyedi azonosító (int)
- **user\_id:** Felhasználó azonosítója (int)
- **fullname:** Teljes név (varchar)
- **phone:** Telefonszám (varchar)
- **address:** Cím (varchar)

elektromos_jarmuvek profiles	
id	: int(11)
user_id	: int(11)
fullname	: varchar(100)
phone	: varchar(20)
address	: varchar(255)

- **Felhasználók:** Információkat tartalmaz a weboldal felhasználóiról.
- **Hirdetések:** A felhasználók által feladott hirdetések adatait tartalmazza.
- **Márkák:** Az elektromos járművek márkáinak listáját kezeli.

## 2.3 Alkalmazott technológiák

A weboldal fejlesztése során számos technológiát vettem be, beleértve a HTML-t, CSS-t a stílusokhoz, JavaScriptet és jQuery-t az interaktivitáshoz.

## 2.4 Implementációs folyamat

A projektet agilis fejlesztési módszertanok alapján hajtottam végre, rövid iterációkban dolgozva, amely lehetővé tette a gyors visszajelzés integrálását és a rugalmas fejlesztést.

## 2.5 Fontosabb algoritmusok

Az egyik kulcsfontosságú algoritmus a hirdetések dinamikus szűrése volt, amely lehetővé teszi a felhasználók számára, hogy gyorsan megtalálják a keresett járműveket.

### Session kezelés

```
if session_status session_start
```

Ez a kódrészlet ellenőrzi, hogy a PHP munkamenet elindult-e már. Ha nem, akkor elindítja azt. A munkamenetek kezelése elengedhetetlen a webalkalmazásokban, mivel lehetővé teszik állapotinformációk (pl. felhasználói bejelentkezési adatok) tárolását kérések között.



## Adatbázis kapcsolódás

```
$conn=$(new mysqli $servername $username $password $dbname)
if [ $? -ne 0 ]
then
    die "Connection failed: " $conn
fi
```

Ez a rész hozza létre a kapcsolatot az adatbázissal. Ha nem sikerül a kapcsolódás, a program megszakítja a futást, megakadályozva ezzel, hogy érzékeny vagy rosszul formázott adatokat próbáljon tovább feldolgozni.

## Adatlekérdezés és megjelenítés

```
$query="SELECT advertisements.id, advertisements.title,
advertisements.description, advertisements.type, users.fullname,
GROUP_CONCAT(images.file_path) AS image_paths FROM advertisements JOIN users ON
advertisements.user_id = users.id LEFT JOIN images ON advertisements.id =
images.ad_id WHERE advertisements.user_id = '$user_id' GROUP BY
advertisements.id LIMIT 10"
$result=$(conn query $query)
```

Ez a lekérdezés összekapcsolja az **advertisements**, **users**, és **images** táblákat, hogy egyetlen lekérdezésben gyűjtse össze az adatokat. A **GROUP\_CONCAT** függvény használata különösen érdekes, mivel lehetővé teszi több kép elérési útjának egyetlen mezőbe való összefűzését, amit később fel lehet használni a képek megjelenítésére.

## Dinamikus tartalom megjelenítése

```
echo "<header style='background-color: #333; ...'"
```

A hirdetés részleteit, beleértve a címet, leírást, típust, feladót, és az elérhető képeket, dinamikusan jeleníti meg. A stílusok inline megadása helyett ajánlott lenne CSS osztályokat használni a jobb karbantarthatóság és stíluskezelés érdekében.

## Képek kezelése

```
if $row 'image_paths' echo "<button onclick='openGallery("
json_encode $image_paths ")" ' ...'
```

Ez a rész kezeli a hirdetéshez kapcsolódó képek dinamikus megjelenítését. A **GROUP\_CONCAT** segítségével összegyűjtött képutakat felhasználva lehetőség nyílik arra, hogy a felhasználó megtekinthesse az összes hirdetési képet egy új ablakban.

## JavaScript a képgaléria kezelésére

```
function openGallery
```

Egy JavaScript függvény, amely lehetővé teszi a képek egy új böngészőablakban való megjelenítését, javítva ezzel a felhasználói élményt.

## felhasználói jogosultság ellenőrzése

```
if isset $_SESSION 'user_id' header "Location: login.php" exit
```

Ez a kód ellenőrzi, hogy a felhasználó be van-e jelentkezve. Ha nem, átirányítja a **login.php** oldalra. Ez biztosítja, hogy csak bejelentkezett felhasználók adhassanak fel hirdetéseket.

## hirdetés feladása

```
$insert_ad_query "INSERT INTO advertisements (title, description, type,
brand, user_id) VALUES ('$title', '$description', '$type', '$brand',
'$user_id')"
```

Ez a parancs végzi a hirdetés adatbázisba való felvitelét. Fontos megjegyezni, hogy bár ez a kód részlet itt nem foglalkozik a felhasználói bevitel sanitizálásával, a valós alkalmazásokban elengedhetetlen lenne a biztonsági rések elkerülése érdekében.

### Képfeltöltés kezelése

```
foreach $_FILES["images"] as $key
do
    if $_FILES["images"][$key]["error"] < 1
    then
        $tmp_name=$_FILES["images"][$key]["tmp_name"]
        $file_name=$_FILES["images"][$key]["name"]
        $target_file=$_uploads_dir.$file_name
        if move_uploaded_file($tmp_name,$target_file)
        then
            $insert_image_query="INSERT INTO images (ad_id, file_path) VALUES ('$ad_id',
            '$target_file')"$conn=query($insert_image_query)
        else
            echo "Hiba a képfeltöltés során: "
            $_FILES["images"][$key]
```

Ez a rész kezeli a hirdetéshez csatolt képek feltöltését. Minden kép feltöltését követően a script hozzáadja az adatbázishoz, kapcsolva a képet az adott hirdetéshez.

### Sikeres hirdetésfeladás kezelése

```
echo "Sikeres hirdetésfeladás!"
header("Location: index.php")
exit
```

Sikeres hirdetésfelvétel után a felhasználót visszairányítja a főoldalra, ezzel jelezve a sikeres műveletet.

### Hiba kezelése

```
echo "Hiba a hirdetésfeladás során: "$conn
```

Amennyiben hiba történik a hirdetés feladása során, a script egy hibaüzenetet jelenít meg a felhasználó számára.

### Felhasználó bejelentkezésének ellenőrzése

```
if isset($_SESSION["user_id"])
then
    header("Location: login.php")
    exit
```

Ez a kódrészlet ellenőrzi, hogy a felhasználó be van-e jelentkezve (vagyis létezik-e `user_id` a munkamenetben). Ha nem, átirányítja a felhasználót a bejelentkezési oldalra. Ez biztosítja, hogy csak bejelentkezett felhasználók férjenek hozzá az oldalhoz.

## Űrlap hirdetés feladásához

```
action "create_ad_process.php" method "post" enctype "multipart/form-data"
```

Az űrlap a hirdetésfeladási adatok beküldéséhez, beleértve a képek feltöltését is (**enctype="multipart/form-data"**).

## Fontos input mezők

- **Cím és leírás mezők:**

```
type "text" name "title" required name "description" rows "4" required
```

Ezek a mezők tartalmazzák a hirdetés címét és leírását. Mindkettő kitöltése kötelező (**required**).

- **Típus és márka választó:**

```
name "type" required name "brand" required
```

Lehetővé teszik a felhasználó számára, hogy kiválassza a hirdetés típusát és a jármű márkáját. Ezek a mezők szintén kötelezően kitöltendőek.

## Képek feltöltése

```
type "file" name "images[]" accept "image/*" multiple required
```

Ez a mező lehetővé teszi több kép egyidejű feltöltését a hirdetéshez. A **multiple** attribútum több fájl kiválasztását teszi lehetővé, míg az **accept="image/\*"** korlátozza a feltölthető fájlokat képtípusokra.

### Márkák dinamikus listája

```
$servername "localhost" $username "root" $password "" $dbname  
"elektromos_jarmuvek" $conn new mysqli $servername $username $password  
$dbname if $conn die "Connection failed: " $conn $brand_query  
"SELECT name FROM brands ORDER BY name" $result $conn query $brand_query  
if $result 0 while $row $result fetch_assoc echo "<option  
value='{ $row['name'] }'>{ $row['name'] }</option>" $conn close
```

Ez a PHP kód a **brands** táblából kérdezi le az összes márkát, és minden márkát egy-egy **<option>** elemként jelenít meg a márka kiválasztóban. Ez dinamikusan tölti fel az elérhető márkák listáját, biztosítva, hogy a felhasználók mindig a legfrissebb adatok közül választhatnak.

### Kulcsfontosságú kódrészletek:

#### 1. Bejelentkezés ellenőrzése:

```
if isset $_SESSION 'user_id' header "Location: login.php" exit
```

Ez a kódrészlet ellenőrzi, hogy a felhasználó be van-e jelentkezve. Ha nincs, átirányítja a **login.php** oldalra.

#### 2. Kijelentkezés kezelése:

```
if isset $_POST 'logout' session_unset session_destroy  
header "Location: login.php" exit
```

Ez lehetővé teszi a felhasználók számára, hogy kijelentkezzenek. A munkamenet törlése után a felhasználót visszaállítja a bejelentkezési oldalra.

#### 3. Dinamikus tartalom beillesztése:

```
include 'advertisements.php'
```

A kezdőoldal tartalmát dinamikusan bővíti a **advertisements.php** által generált hirdetések listájával, így a felhasználók azonnal láthatják a legfrissebb hirdetéseket.

#### 4. Navigációs menü:

```
href "index.php" href "create_ad.php" href "search.php"
```

Ez a kódrészlet a felhasználói interakció központi eleme, lehetővé téve a felhasználók számára, hogy könnyen navigáljanak a weboldalon belül.

#### 1. Felhasználói hitelesítés:

```
$check_user_query "SELECT * FROM users WHERE username='$username'"  
$result $conn query $check_user_query
```

Itt a script lekérdezi az adatbázist, hogy megtalálja-e a megadott felhasználónevet, és ellenőrzi a jelszót a `password_verify` segítségével.

#### 2. Sikeres bejelentkezés kezelése:

```
session_start $_SESSION 'user_id' $row 'id' $_SESSION 'username'  
$row 'username' header "Location: index.php"
```

Sikeres hitelesítés esetén a felhasználó munkamenet-változókat kap, és átirányításra kerül a kezdőoldalra.

#### 3. Hibás bejelentkezési kísérletek naplózása:

```
logAction $conn $row 'id' "Failed login attempt"
```

Ha a jelszó ellenőrzése sikertelen, a függvény naplózza a hibás bejelentkezési kísérletet.

#### 1. Munkamenet indítása és bejelentkezési állapot ellenőrzése:

```
session_start if isset $_SESSION 'user_id' header "Location:  
index.php" exit
```

Ez a rész megakadályozza, hogy a már bejelentkezett felhasználók újra a bejelentkezési oldalra jussanak.

## 2. Bejelentkezési űrlap:

```
action "login.php" method "post" for "username" type "text"
name "username" required for "password" type "password" name "password"
required type "submit"
```

A felhasználói felület, ahol a felhasználók megadhatják bejelentkezési adataikat.

## 3. Stíluslap linkelése:

```
rel "stylesheet" href "style.css"
```

A stíluslap biztosítja az oldal vizuális megjelenését és a felhasználói felület kialakítását.

## 4. Navigációs menü és lábléc:

```
href "index.php" href "register.php"
```

Ezek a navigációs linkek lehetővé teszik a felhasználók számára, hogy könnyen navigáljanak a regisztrációs oldalra vagy vissza a főoldalra.

## 1. Profiladatok lekérdezése és mentése:

```
$query "SELECT * FROM profiles WHERE user_id = $user_id"
```

Ezzel a lekérdezéssel kinyerhetők a felhasználó profiladatok az adatbázisból.

## 2. Profiladatok frissítése vagy új profil létrehozása:

```
if $result 0 $update_query "UPDATE profiles SET fullname='$fullname',
phone='$phone', address='$address' WHERE user_id=$user_id" else
$insert_query "INSERT INTO profiles (user_id, fullname, phone, address)
VALUES ('$user_id', '$fullname', '$phone', '$address')"
```

Ez a logika dönti el, hogy frissíteni kell-e a meglévő profiladatokat, vagy új profilt kell létrehozni

### 1. Profil feldolgozó script beillesztése:

```
include 'profile_process.php'
```

A **profile\_process.php** beillesztése lehetővé teszi a profiladatok előzetes betöltését és az űrlap adatainak feldolgozását.

### 2. Profil űrlap megjelenítése:

```
action "profile_process.php"      method "post"      for "fullname"
type "text"                      name "fullname"      value "<?php      echo
htmlspecialchars($fullname); ?>"
```

Ez az űrlap rész lehetővé teszi a felhasználó számára, hogy megtekinthesse és szerkeszthesse saját profilinformációit, mint a teljes nevét, telefonszámát és címét.

### 3. Sikeres vagy sikertelen művelet üzenetek kezelése:

```
if isset $message class php echo success success
error php echo message div php endif
```

Ez a kód megjeleníti a felhasználói műveletek eredményét, pl. sikeres profilfrissítést vagy hibát a folyamat során.



## 1. Felhasználónév ellenőrzése és új felhasználó hozzáadása:

```
$check_username_query "SELECT * FROM users WHERE username='$username'"
$result $conn query $check_username_query if $result
0 echo "Ez a felhasználónév már foglalt!" else
$insert_user_query "INSERT INTO users (fullname, username, email,
password) VALUES ('$fullname', '$username', '$email', '$password')"
```

A script ellenőrzi, hogy a választott felhasználónév szabad-e. Ha igen, akkor hozzáadja az új felhasználót az adatbázishoz.

## 2. Sikeres regisztráció üzenete:

```
if $conn query $insert_user_query TRUE echo "Sikeres
regisztráció!" else echo "Hiba a regisztráció során: " $conn
```

Ez a rész visszajelzést ad a felhasználónak a regisztráció sikerességéről vagy egy esetleges hibáról.

### • Alap stílusok:

```
font-family 'Arial' background-color #f4f4f4
```

Az egész oldalra érvényes alapbeállításokat határozza meg, mint a betűtípus és a háttérszín.

### • Fejléc stílusa:

```
background-color #333 color display
```

A weboldal fejlécének hátterét és szövegszínét állítja be, valamint flexbox elrendezést használ.

- **Navigációs menü stílusa:**

```
color hsl 151 86% 36% text-decoration
```

A navigációs linkek színét és aláhúzásának eltávolítását határozza meg.

- **Kijelentkezés gomb stílusa:**

```
.logout-button:hover color #ff6347
```

A kijelentkezés gomb hover (egérrel való fölé helyezésekor aktív) állapotának színét adja meg.

- **Tartalom és űrlapok stílusai:**

```
#main-content text-align display flex-direction
```

A fő tartalom középre igazítását, valamint az űrlapok elrendezését és kinézetét szabályozza.

- **Gombok és linkek interaktív stílusai:**

```
:hover background-color hsla 112 88% 41% 0.814 :hover  
color hsl 193 68% 51%
```

A gombok és linkek interaktív állapotainak (hover) stílusát definiálja, javítva ezzel a felhasználói élményt.

- **Leválasztó és kép konténerek:**

```
.advertisement-container display justify-content  
.image-container border 1px #ddd padding  
5px
```

A hirdetések és képek konténereinek elrendezését és stílusát határozza meg, elősegítve ezzel a vizuális tisztaságot és rendezettséget.

## 2.6 Tesztelési stratégia

- **Egységtesztelést végeztem:** Különálló funkciók és metódusok tesztelésével biztosítottam, hogy minden alapelem helyesen működjön a várt bemenetekkel. A PHP függvények és adatbázis hívások helyes működését ellenőriztem.
- **Integrációs tesztek hajtottam végre:** Ellenőriztem, hogy az alkalmazás különböző részei, mint az adatbázis lekérdezések és a felhasználói felület, hogyan működnek együtt. Az adatfolyamok és folyamatok közötti kapcsolatokat teszteltem, hogy azok megfelelően integrálódtak-e.
- **Funkcionális tesztek alkalmaztam:** Az egész rendszer szintjén teszteltem, hogy az alkalmazás eleget tesz-e a meghatározott funkcionálitási követelményeknek. Ezt úgy tettem, hogy végigmentem a felhasználói forgatókönyveken, például hirdetések feladásán, módosításán és keresésén.
- **Felhasználói elfogadási tesztek végeztem:** Valós felhasználók segítségével teszteltem, hogy az interfész és a funkciók megfelelnek-e a felhasználói elvárásoknak. Ezáltal biztosítottam, hogy az alkalmazás könnyen kezelhető és megfelelő a célcsoport számára.
- **Biztonsági tesztek folytattam:** Különös figyelmet fordítottam az alkalmazás sebezhetőségeinek felkutatására, mint például az SQL-injekciók és a keresztoldali szkriptelés (XSS) kockázatainak minimalizálására.
- **Teljesítménytesztelést alkalmaztam:** Megvizsgáltam, hogy az alkalmazás hogyan teljesít nagy terhelés alatt. Teszteltem a rendszer válaszidejét és stabilitását, hogy biztosítsam az alkalmazás megbízhatóságát nagy felhasználói forgalom esetén is.

## 3. Felhasználói dokumentáció

### 3.1 A rendszer áttekintése

Az általam használt rendszer egy webalapú hirdetési platform, amely kifejezetten elektromos járművekkel és azok tartozékaival foglalkozik. A célja, hogy összekösse az elektromos járművek iránt érdeklődő felhasználókat a különböző márkák és termékek kínálóival. Az oldal lehetőséget biztosít a felhasználók számára, hogy hirdetéseket adjanak fel, keresgéljenek az elérhető ajánlatok között, és közvetlenül kapcsolatba léphessenek a hirdetőkkel.

#### Főbb funkciók és felépítés

- **Felhasználói profilok:** A felhasználók regisztrálhatnak és saját profilt hozhatnak létre, ahol megadhatják személyes adataikat, és kezelhetik hirdetéseiket.
- **Hirdetések kezelése:** A felhasználók hirdetéseket adhatnak fel, módosíthatják vagy törölhetik azokat. Minden hirdetés tartalmazza a termék címét, leírását, típusát és márkáját, valamint kapcsolódó képeket.
- **Keresés és szűrés:** A látogatók keresőfunkció segítségével kereshetnek a hirdetések között, szűrhetnek márkák és típusok szerint, hogy megtalálják a számukra legmegfelelőbb ajánlatokat.
- **Biztonság és adatkezelés:** A rendszer biztonságos kapcsolatot használ az adatok védelme érdekében, és csak titkosított jelszavakat tárol.
- **Interakció és kapcsolatfelvétel:** A felhasználók közvetlenül kapcsolatba léphetnek a hirdetőkkel a platformon keresztül, megkönnyítve ezzel a kommunikációt és az üzletkötéseket.

Ez a rendszer egy komplex megoldást nyújt a felhasználóknak, lehetővé téve számukra, hogy egyszerűen böngésszenek, hirdessenek és interakcióba lépjenek elektromos járművek piacán. A felhasználóbarát felület és a hatékony adatkezelés elősegíti az információáramlást és a felhasználói élményt.

### 3.2 Rendszerkövetelmények

#### Hardverkövetelmények

- **Szerver:** A rendszernek kompatibilisnek kell lennie Linux és Windows szerver környezetekkel.
- **Adattárolás:** Legalább 1 GB szabad terület az adatbázis fájlok és a feltöltött képek tárolásához.
- **Memória:** Legalább 512 MB RAM a szerveroldali feldolgozáshoz.

#### Szoftverkövetelmények

- **Internet kapcsolat**
- **Kliens oldali technológiák:** Modern webböngészők támogatása (Chrome, Firefox, Safari, Edge) legfrissebb vagy előző major verzióiban.

#### Adatbázis követelmények

- A rendszer az **InnoDB** tárolómotorra és az **utf8mb4** karakterkódolásra épül, amely támogatja az emojiakat és különleges karaktereket.
- Táblák: **advertisements**, **brands**, **help\_messages**, **images**, **logs**, **profiles**, **users**.
- Kapcsolatok: Több tábla közötti kapcsolatok (pl. **advertisements** és **users**, **images** és **advertisements**).

#### Funkcionális követelmények

- **Hirdetések kezelése:** Hirdetések létrehozása, módosítása, listázása, és törlése.
- **Felhasználókezelés:** Regisztráció, bejelentkezés, kijelentkezés, profiladatok kezelése.
- **Keresés:** Márkák és típusok alapján szűrt keresési lehetőség.
- **Képfeltöltés:** A hirdetésekhez képek csatolása és megjelenítése.

#### Nem Funkcionális Követelmények

- **Biztonság:** Jelszavak titkosítása, SQL injekció és XSS támadások elleni védelem.
- **Felhasználóbarát Felület:** Reszponzív webdesign, intuitív felhasználói felület.
- **Skálázhatóság:** A rendszernek képesnek kell lennie a növekvő adatmennyiség és felhasználói igények kiszolgálására.
- **Karbanthartóság:** Egyszerű frissítési és karbantartási lehetőségek.

#### Dokumentáció és Támogatás

- Teljes körű dokumentáció a rendszer telepítéséhez, konfigurációjához és használatához.
- Fejlesztői és felhasználói támogatás biztosítása.

Ezek a követelmények biztosítják, hogy a rendszer megfelelően működjön, skálázható és karbantartható legyen, valamint kielégítse a felhasználói igényeket a hirdetések kezelése és megjelenítése terén.

### 3.3 Telepítés és indítás

#### • Böngésző megnyitása:

Elindítottam a számítógémemen egy webböngészőt, például a Google Chrome-ot, Firefoxot vagy Safari-t.

- **Webcím beírása:**

A böngésző címsorába beírtam a következő URL-t: <http://greencarsandparts.hu>.

- **Oldal betöltése:**

Megnyomtam az Enter billentyűt a böngésző címsorában, hogy betöltssem a weboldalt. Ekkor a böngésző kapcsolatot létesített a szerverrel, és lekérte az oldal tartalmát.

- **Navigáció az oldalon:**

Az oldal betöltése után megjelentek a különböző menüpontok és funkciók, amelyek között navigálhatok. Ilyenek például a hirdetések megtekintése, regisztráció, bejelentkezés, vagy új hirdetés feladása.

- **Interakció az oldallal:**

Kipróbáltam az oldal különböző funkcióit: hirdetések böngészése, adatok szűrése, regisztráció és bejelentkezés. Mindezt a felhasználói felületen keresztül, követve az oldalon található utasításokat.

- **Oldal használatának folytatása:**

Miután bejelentkeztem, további funkciók váltak elérhetővé, mint például a saját hirdetések kezelése, profiladatok szerkesztése vagy a hirdetések részletes megtekintése.

- Ezekkel a lépésekkel sikeresen elértem és használtam a [greencarsandparts.hu](http://greencarsandparts.hu) weboldalt a böngészőn keresztül.

## 3.4 A rendszer használata

### Regisztráció és Bejelentkezés

1. **Regisztráció:** Új felhasználók a [register.php](#) oldalon regisztrálhatnak, ahol meg kell adniuk a teljes nevüket, felhasználónevüket, e-mail címüket és jelszavukat. A regisztráció után a felhasználókat átirányítják a bejelentkezési oldalra.

The screenshot shows a web page titled "Regisztráció" (Registration). At the top, there are links for "Főoldal" (Home) and "Bejelentkezés" (Login). The main content area contains a registration form with the following fields: "Teljes név:" (Full name), "Felhasználónév:" (Username), "Email cím:" (Email address), and "Jelszó:" (Password). Below these fields is a green "Regisztráció" (Registration) button. At the bottom of the form, there is a link that says "Már van fiókja? Jelentkezzen be." (Already have an account? Log in). The footer of the page indicates "© 2024 Elektronos Jármű Hirdetések".

2. **Bejelentkezés:** A **login.php** oldalon meglévő felhasználók bejelentkezhetnek a rendszerbe a felhasználónevük és jelszavuk megadásával. Sikeres bejelentkezés után a felhasználókat átirányítják az index oldalra.

## Hirdetések Kezelése

1. **Hirdetésfeladás:** Bejelentkezett felhasználók feladhatnak új hirdetéseket a **create\_ad.php** oldalon. Itt meg kell adni a hirdetés címét, leírását, típusát (elektromos autó vagy bontott akkumulátor), márkát, és képeket is hozzá lehet fűzni.

2. **Hirdetések Bőngészése:** Az **index.php** oldalon minden felhasználó megtekintheti a rendelkezésre álló hirdetéseket. Kattintva egy hirdetés címére, az **advertisement\_details.php** oldalon tekinthetők meg a hirdetés részletei, beleértve a leírást, típust, évjáratot (ha rendelkezésre áll), feladó nevét és a hirdetéshez tartozó képeket.

## Keresés

- **Keresés:** A **search.php** oldalon a felhasználók keresést végezhetnek hirdetések között márká és típus alapján. A keresési feltételek megadása után a **search\_process.php** oldal listázza a találatokat.

## Profil Szerkesztése

1. **Profil:** Bejelentkezett felhasználók a **profile.php** oldalon szerkeszthetik profiljukat, ahol megadhatják vagy módosíthatják a teljes nevüket, telefonszámukat, és címüket.

#### Kijelentkezés

- **Kijelentkezés:** A felhasználók bármikor kijelentkezhetnek a rendszerből a "Kijelentkezés" gombra kattintva, amely a navigációs menüben található az oldalak fejlécében.

Ezek a lépések biztosítják a webalkalmazás alapvető használatát, lehetővé téve a felhasználók számára, hogy hirdetéseket adjanak fel, böngésszenek, keresést végezzenek, és személyes információikat kezeljék.

## 3.5 Gyakori problémák és megoldásaik

### 1. Bejelentkezési hibák

**Probléma:** Felhasználók nem tudnak bejelentkezni, mert elfelejtik a jelszavukat vagy a felhasználónevüket. **Megoldás:** Bevezettem egy "Elfelejtett jelszó" funkciót, amely lehetővé teszi a felhasználóknak, hogy e-mailben újraindíthassák jelszavukat. Ezenkívül az elfelejtett felhasználónév esetében is biztosítok segítséget e-mail támogatáson keresztül.

### 2. Képfeltöltési problémák

**Probléma:** Néhány felhasználó problémát jelentett a képek feltöltésekor, például a kép mérete túl nagy vagy a fájlformátum nem támogatott. **Megoldás:** A felhasználó maximum 6 képet tud feltölteni ami az oldalon a hirdetés feladásnál is fel van tüntetve.

### 3. Hirdetések frissítési/szerkesztési hibái

**Probléma:** Felhasználók hibákat tapasztalnak a hirdetéseik frissítésekor és szerkesztésekor. **Megoldás:** Részletes ellenőrzéseket és hibakezelést implementáltam a hirdetés szerkesztési funkcióba. A felhasználók mostantól világos visszajelzést kapnak minden mező kitöltési követelményéről, és a szerver oldali validáció segít megelőzni az adatintegritási problémákat.



#### 4. Lassú oldalbetöltési sebesség

**Probléma:** A felhasználók jelentették, hogy az oldal lassan töltődik be, különösen nagy mennyiségű adat lekérésekor. **Megoldás:** Ellenőrizze az internet kapcsolatát vagy törölje sütiket.

### 3.6 Jövőbeli Fejlesztések

Terveim között szerepel az oldal funkcionalitásának és felhasználói élményének javítása, valamint új szolgáltatások bevezetése. Itt van néhány terület, ahol fejlesztéseket szeretnék eszközölni:

1. **Képek kezelésének fejlesztése:** Bár már most is van lehetőség képek feltöltésére a hirdetésekhez, szeretném továbbfejleszteni ezt a funkciót. Egy új képgaléria funkciót tervezek bevezetni, amely lehetővé teszi a felhasználók számára, hogy könnyebben böngésszenek a hirdetési képek között. Ezenkívül szeretném automatizálni a képek méretének és minőségének optimalizálását a feltöltéskor, hogy javuljon a weboldal teljesítménye.
2. **Részletes keresési funkciók:** Jelenleg a keresési lehetőségek korlátozottak. A terveim között szerepel egy részletes keresőfunkció bevezetése, amely több szűrési lehetőséget kínál, például évjárat, ár, kilométeróra állás, és egyéb specifikus attribútumok alapján. Ez segít a felhasználóknak gyorsabban megtalálni azt, amit keresnek.
3. **Felhasználói profilok kibővítése:** Jelenleg a felhasználói profilok viszonylag egyszerűek. Szeretnék több lehetőséget biztosítani a felhasználóknak profiljaik testreszabására, beleértve a profil képek feltöltését, valamint további személyes információk és preferenciák megadását. Ez lehetővé teszi a felhasználók számára, hogy jobban megismerjék egymást és bizalommal vásároljanak vagy adjanak el járműveket.
4. **Értékelési rendszer:** Az ügyfelek és a hirdetések értékelési rendszerének bevezetése szintén a tervben van. Ez a funkció lehetővé tenné a felhasználók számára, hogy visszajelzést adjanak a tranzakcióikról, így növelve a platform megbízhatóságát és segítve másokat a döntéshozatalban.
5. **Mobilalkalmazás:** A hosszú távú célom egy mobilalkalmazás fejlesztése, amely még kényelmesebbé teszi a platform használatát. A mobilalkalmazás lehetővé tenné a felhasználók számára, hogy bárhol és bármikor hozzáférjenek a hirdetésekhez, valamint értesítéseket kapjanak az új hirdetésekről vagy az őket érdeklő tranzakciók állapotáról.

## 4.Összefoglalás

Az Elektromos Jármű Hirdetések weboldalának adatbázis és kódstruktúrája egy összetett, de jól szervezett rendszert alkot, amely elősegíti az elektromos járművekkel kapcsolatos hirdetések feladását, keresését és kezelését. Az adatbázis táblái, mint az **advertisements**, **brands**, **help\_messages**, **images**, **logs**, **profiles**, és **users** különféle típusú adatokat tárolnak, amelyek a felhasználók közötti interakciók alapját képezik. A weboldal PHP scriptjei ezeket az adatokat használják fel a dinamikus tartalom generálásához, beleértve a hirdetések megjelenítését, a felhasználói interakciókat, mint a regisztráció, bejelentkezés és profilkezelés, valamint a hirdetések keresését és böngészését.

A rendszer különösen fontos figyelmet fordít a felhasználói élményre és a felület egyszerűségére, ugyanakkor biztosítja a szükséges funkcionalitást a hirdetések hatékony kezeléséhez. A képekkel gazdagított hirdetések elősegítik a járművek vizuális prezentációját, míg a márkák és típusok szerinti szűrési lehetőségek megkönnyítik a keresést. A felhasználói interakciókat támogató funkciók, mint a bejelentkezés, regisztráció, és profilkezelés, tovább erősítik a közösségi aspektust és a platform megbízhatóságát.

Összességében az Elektromos Jármű Hirdetések weboldala egy átfogó megoldást kínál az elektromos járművek piacának szereplői számára, lehetővé téve számukra, hogy hatékonyan kommunikáljanak, kereskedjenek és információkat cseréljenek egy biztonságos és felhasználóbarát környezetben

## 5. Forráskód és dokumentáció

- **Github** (<https://www.github.com/szitaspeter/hasznalelektromos>): A záródolgozatomhoz tartozó adatbázis, a programom teljes forráskódja, valamint a dokumentációm PDF változata githubon ez a hozzáféréssel érhető el.

## 6. Irodalomjegyzék és források

5. **PHP és MySQL webfejlesztés kezdőknek:** Ez a forrás alapvető bevezetést nyújt a PHP használatába, különös tekintettel az adatbázis-kezelésre MySQL használatával.
5. **W3Schools** (<https://www.w3schools.com>): Ez az online oktatási platform számos útmutatót és gyakorlatot kínál a webfejlesztés minden aspektusára, beleértve a HTML-t, CSS-t, JavaScriptet, PHP-t és SQL-t.
5. **Stack Overflow** (<https://stackoverflow.com>): A fejlesztők közössége, ahol gyakran kerülnek megvitatásra specifikus programozási kérdések és problémák, beleértve az adatbázis-kezelés és webfejlesztési technikákat is.
5. **Elektromos járművek piaca:** Az elektromos járművekkel kapcsolatos ismeretek és trendek megértéséhez fontos források a szakmai publikációk, piaci jelentések, valamint tematikus weboldalak és blogok.
5. **CSS-Tricks** (<https://css-tricks.com>): A CSS-Tricks weboldal hasznos tippeket, trükköket és útmutatókat kínál a CSS használatához, amelyek különösen hasznosak lehetnek a weboldalak kinézetének finomításában.
5. **MDN Web Docs** (<https://developer.mozilla.org>): A Mozilla Developer Network (MDN) átfogó leírást nyújt a webfejlesztési technológiákról, beleértve a legújabb HTML, CSS és JavaScript szabványokat.
5. **InfoJegyzet** (<https://infojegyzet.hu>): Az InfoJegyzet.hu egy magyar nyelvű informatikai tudásbázis, amely programozás, webfejlesztés, adatbázis-kezelés és szoftvertervezés témákban nyújt útmutatást és tanácsokat.