

R&J Autókölcsönző - Projekt Dokumentáció (Rövid)

Verzió: 1.0 Dátum: 2025-04-09

Készítők: Felhasználói Felület: Jurcsó János Árpád Adminisztrációs Felület: Dávid Roland

Bevezetés

A projekt célja egy webes alapú autókölcsönző alkalmazás, az "R&J Autókölcsönző" létrehozása volt. Az alkalmazás lehetővé teszi a regisztrált felhasználók számára, hogy megtekintsék a bérelhető járművek kínálatát, szűrjenek a járművek között, részletes adatokat kapjanak egy-egy autóról, véleményeket olvassanak és írjanak, valamint online lefoglalják a kiválasztott járművet a megadott időszakra. Az alkalmazás tartalmaz egy adminisztrációs felületet is, ahol a cég munkatársai kezelhetik a járműflottát, a felhasználókat, a bérleteket, a véleményeket és az akciókat.

Az R&J Autókölcsönző egy PHP nyelven íródott, MySQL (vagy MariaDB) adatbázist használó webalkalmazás. Két fő részből áll:

- **Felhasználói Felület:** Nyilvánosan elérhető (bizonyos funkciók bejelentkezéshez kötöttek), ahol a látogatók és regisztrált felhasználók böngészhetnek, regisztrálhatnak, bejelentkezhetnek, bérelhetnek és kezelhetik profiljukat. Ezt a részt Jurcsó János Árpád fejlesztette.
- **Adminisztrációs Felület:** Csak adminisztrátori jogosultsággal rendelkező felhasználók számára elérhető, a rendszer tartalmának és működésének menedzselésére szolgál. Ezt a részt Dávid Roland fejlesztette.

A rendszer alapvető funkciói közé tartozik a felhasználókezelés (regisztráció, bejelentkezés, profil módosítás), a járműkatalógus megjelenítése és szűrése, a bérleti folyamat kezelése (beleértve a dátumütközések ellenőrzését és az e-mailes visszaigazolást), a felhasználói vélemények gyűjtése és megjelenítése, valamint a hűségpontrendszer alapjainak kezelése. Az adminisztrációs felület teljes körű CRUD (Create, Read, Update, Delete) műveleteket biztosít a főbb entitások (járművek, felhasználók, bérletek, vélemények, akciók) felett.

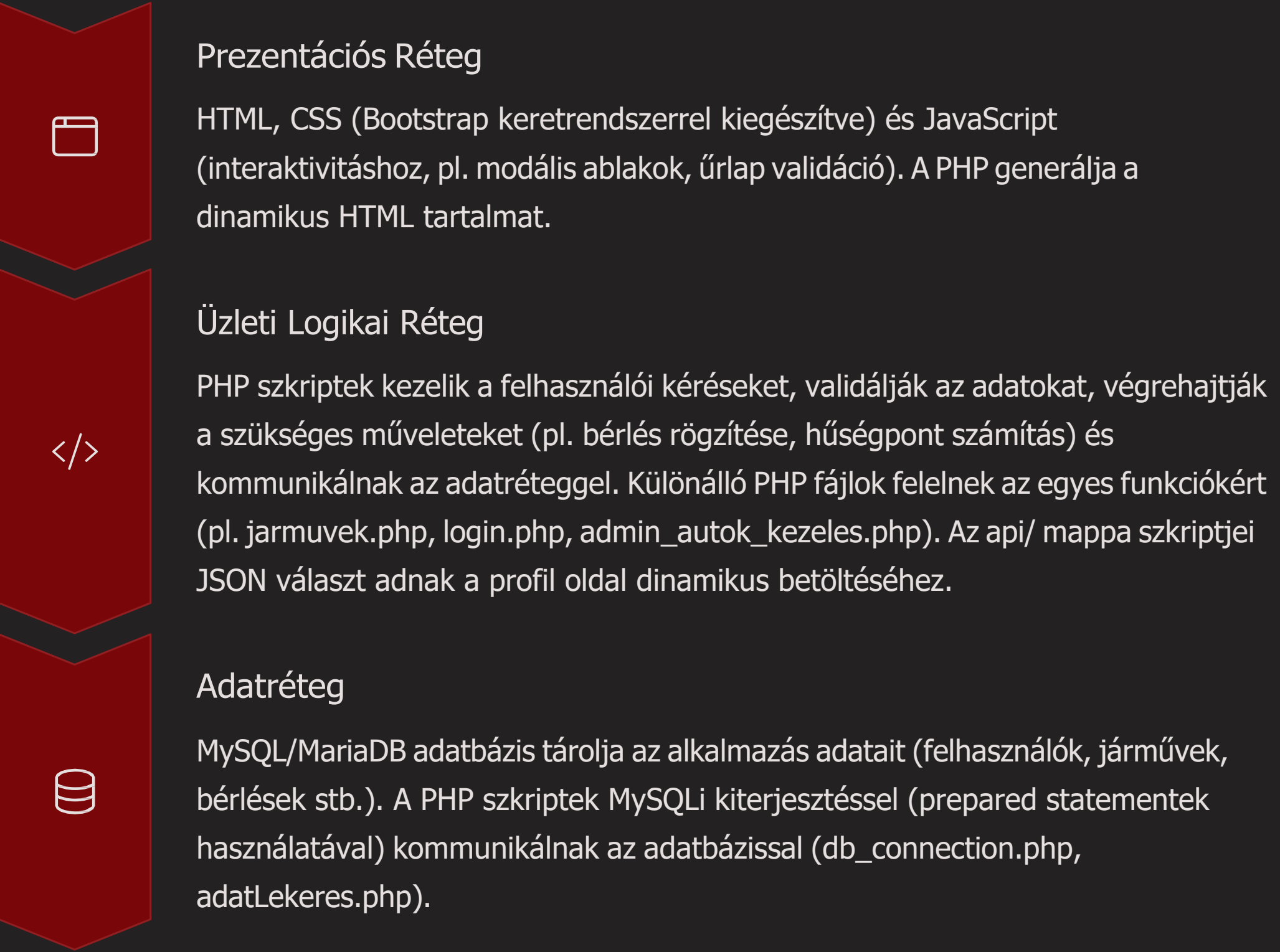
Célközönség

- **Felhasználók:** Magánszemélyek vagy cégek, akik gépjárművet szeretnének bérelni rövidebb vagy hosszabb távra.
- **Adminisztrátorok:** Az R&J Autókölcsönző munkatársai, akik felelősek a weboldal tartalmának frissítéséért, a foglalások kezeléséért és a rendszer általános felügyeletéért.




Rendszerarchitektúra és Technológiák

Architektúra

A rendszer egy klasszikus, szerveroldali renderelésen alapuló webalkalmazás architektúrát követ.



Technológiai Stack

 Szerveroldali technológiák	 Kliensoldali technológiák	 Szerveroldali könyvtárak
PHP (valószínűleg 7.4 vagy újabb, a password_hash, fetch_all, ?? operátor és a api/ fájlok JSON kezelése alapján)	HTML5	PHPMailer (e-mail küldéshez a bérletnél és admin válaszoknál)
MySQL / MariaDB adatbázis	CSS3 (beleértve a style.css, admin.css, stb. fájlokat)	
Apache vagy Nginx webservert (nem specifikált, de tipikus PHP környezet)	Bootstrap 5 (a CSS és JS hivatkozások alapján)	
	JavaScript (Vanilla JS a index.js, jarmuvek.js fájlokban és inline szkriptekben)	

Funkcionális Követelmények és Megvalósítás

Felhasználói Funkciók

Regisztráció és Bejelentkezés

Új felhasználók létrehozása és meglévő felhasználók autentikálása. Az index.php-n lévő gombok modális ablakokat nyitnak meg. A jelszavakat a password_hash() függvénnyel hasheli, és ellenőrzi a felhasználónév egyediségét.

Járművek Böngészése és Szűrése

Listázza a bérelhető járműveket, lehetőséget biztosít szűrésre átvételi/leadási dátum, járműkategória és ár alapján. Kizárja azokat a járműveket, amelyek a megadott időintervallumban már foglaltak.

Jármű Bérlés

A bérlési folyamat véglegesítése, adatok rögzítése, hűségpontok számítása és visszaigazoló e-mail küldése PHPMailer segítségével. Kiszámolja a bérlés teljes költségét és a kapott hűségpontokat.

Profil Kezelés

Bejelentkezett felhasználók megtekinthetik és módosíthatják saját adataikat, valamint megtekinthetik korábbi bérléseiket és hűségpontjaikat. JavaScript fetch kéréseket indít az API végpontok felé.

Adminisztrációs Funkciók

Járműkezelés

Járművek hozzáadása, listázása, módosítása és törlése. Több kép feltöltését is támogatja, a képeket a szerveren tárolja, és a képek webes elérési útjait JSON tömbként tárolja az adatbázisban.

Jogosultságkezelés

Felhasználók listázása, jogosultságuk (admin/normál) módosítása, adataik módosítása és törlése. Lehetőséget ad új jelszó megadására (hasheléssel).

Bérléskezelés

Meglévő bérlések listázása, újak manuális hozzáadása, meglévők módosítása és törlése. Validálja az adatokat (létező felhasználó/jármű, dátumok).

Véleménykezelés

A felhasználók által beküldött vélemények listázása, törlése és megválaszolása. PHPMailer segítségével elküldi a választ emailben a felhasználónak.

API Végpontok

- Felhasználói Adatok (api/user.php): Kezeli a felhasználói profiladatokat lekérdezését (GET) és módosítását (POST). JSON formátumban adja vissza az adatokat.
- Bérlési Adatok (api/rentals.php): Lekérdezi a bejelentkezett felhasználó korábbi bérléseinek adatait. Az eredményeket JSON tömbként adja vissza.

Adatbázis Tervezés

Az alkalmazás egy relációs adatbázist használ (MySQL/MariaDB), amelynek sémáját az autoberles.sql fájl definiálja. Ez tartalmazza a táblák létrehozását (CREATE TABLE) és néhány mintaadat beszúrását (INSERT INTO).

Táblák Leírása

felhasznalo	jarmuvek	berlesek
Felhasználói fiókok adatainak tárolása.	A bérelhető járművek adatainak tárolása.	A bérleti tranzakciók rögzítése.
<ul style="list-style-type: none">felhasznalo_nev (PRIMARY KEY)nev (teljes név)emailcimjogositvany_kiallitasDatumszamlazasi_cimjelszo (hashelt)admin (jogosultsági szint)Telefonszámhusegpontok	<ul style="list-style-type: none">jarmu_id (PRIMARY KEY)felhasznalas_id (FOREIGN KEY)szerviz_id (FOREIGN KEY)gyartotipusmotorgyartasi_evleirasar (napi bérleti díj)kep_url (JSON tömb)	<ul style="list-style-type: none">berles_id (PRIMARY KEY)jarmu_id (FOREIGN KEY)felhasznalo (FOREIGN KEY)tol (bérlet kezdete)ig (bérlet vége)kifizetve (fizetési státusz)
velemenyek	akciok	Egyéb táblák
Felhasználói vélemények tárolása.	Járművekre vonatkozó akciók és kedvezmények tárolása.	felhasznalas: Járműkategóriák tárolása.
<ul style="list-style-type: none">velemeny_id (PRIMARY KEY)felhasznalo_nevuzenetdatumadmin_valaszjarmu_id	<ul style="list-style-type: none">akcio_id (PRIMARY KEY)jarmu_id (FOREIGN KEY)kedvezmeny_szazalekkezdetevegeleirasis_black_friday	<ul style="list-style-type: none">felhasznalas_id (PRIMARY KEY)nev (kategória neve) <div>szervizek: Műszaki vizsga lejáratí évek tárolása.</div> <ul style="list-style-type: none">id (PRIMARY KEY)muszaki_vizs_lejarat

Kapcsolatok

- jarmuvek kapcsolódik felhasznalas-hoz (felhasznalas_id)
- jarmuvek kapcsolódik szervizek-hez (szerviz_id)
- berlesek kapcsolódik jarmuvek-hez (jarmu_id)
- berlesek kapcsolódik felhasznalo-hoz (felhasznalo -> felhasznalo_nev)
- akciok kapcsolódik jarmuvek-hez (jarmu_id)
- velemenyek kapcsolódhat jarmuvek-hez (jarmu_id)
- velemenyek kapcsolódhat felhasznalo-hoz (felhasznalo_nev)

Fájlstruktúra

A projekt fájlstruktúrája logikusan szervezett, elkülönítve a felhasználói és adminisztrációs felület fájljait, valamint a különböző erőforrásokat (CSS, JavaScript, képek).

Főkönyvtár	Felhasználói felület	Adminisztrációs felület
<ul style="list-style-type: none">/R-J_autoberles/autoberles.sql (adatbázis séma)favicon.pngtest_full_application.py (teszt szkript)	<ul style="list-style-type: none">/php/auto_adatok.phpcseveges.phpforum.phphusegpontok.phpindex.phpjarmuvek.phpkapcsolat.phplogin.phplogout.phpmodal.phpmodosit_profil.phpprofilom.phpregister.php	<ul style="list-style-type: none">/admin/php/admin_akciok.phpadmin_berlesek.phpadmin_berlesek_mod.phpadmin_jogosultsag.phpadmin_jog_mod.phpadmin_velemenyek.phpautok_kezeles.phpautok_kezeles_modositas.php
API	Erőforrások	Egyéb
<ul style="list-style-type: none">/api/rentals.phpuser.php	<ul style="list-style-type: none">/css/admin.cssforum.csshusegpontok.cssindex.cssjarmuvek.csskapcsolat.cssstyle.css/js/index.jsjarmuvek.js	<ul style="list-style-type: none">/kepek/ (feltöltött képek)/php/src/ (PHPMailer)/php/vendor/ (Composer függőségek)/node_modules/ (Bootstrap forrás)

A közös funkciókat (adatbázis kapcsolat, adatlekérés) a db_connection.php és adatLekeres.php fájlok tartalmazzák, amelyek mind a felhasználói, mind az adminisztrációs felületen elérhetőek.

Felhasználói Felület (UI/UX)

Stíluslapok

A `style.css` definiálja az általános megjelenést (színek, betűtípusok, elrendezés, animációk). Egy modern, letisztult dizájnrá törekszik, szürke és sötét árnyalatok dominálnak (`#e8ecef`, `#1c2526`). Animációkat használ a felhasználói élmény javítására (pl. `fadeInBody`, `slideInUp`).

A Bootstrap 5 keretrendszert használja a reszponzív grid rendszerhez, alapvető komponensekhez (kártyák, modális ablakok, űrlapok, gombok) és utility class-okhoz. Külön CSS fájlok (`forum.css`, `jarmuvek.css`, stb.) tartalmazhatnak oldal-specifikus stílusokat. Az `admin.css` az adminisztrációs felület egyedi stílusait tartalmazza, valószínűleg egyszerűbb, funkcionálisabb megjelenést biztosítva.

Reszponzivitás

A Bootstrap grid rendszerének és a `style.css`-ben definiált media query-knek köszönhetően az oldal reszponzív, alkalmazkodik a különböző képernyőméretekhez (desktop, tablet, mobil). Mobilon a navigációs menü egy "hamburger" ikon mögé rejthető (menu-toggle). A kártyák elrendezése a képernyő szélességétől függően változik (pl. 4 oszlopból 1 oszlopba).

Kliensoldali Interakciók



Modális ablakok

`index.js` tartalmazza az `openModal` és `closeModal` függvényeket a bejelentkezési, regisztrációs és üzenetablakok dinamikus megjelenítéséhez/elrejtéséhez.



Szűrő panel

`jarmuvek.js` tartalmazhat logikát a szűrő panel lenyitásához/bezsukásához (`toggleFilterBtn`).



Galéria

Az `auto_adatok.php`-n JavaScript kezeli a képgaléria modális ablakának megnyitását, bezárását, a képek közötti lapozást (gombokkal és billentyűzettel).



Profil oldal

`profilom.php` fetch API hívásokkal tölti be dinamikusan a felhasználói és bérleti adatokat.

Biztonság

Autentikáció és Autorizáció

- Session-alapú autentikáció (`$_SESSION['felhasznalo_nev']`).
- Admin felület védelme session ellenőrzéssel (`$_SESSION['admin'] == true`).
- Az API végpontok (`api/user.php`, `api/rentals.php`) is ellenőrzi a sessiont.

Adatbázis Biztonság

- Az alkalmazás következetesen használ prepared statementeket (MySQLi `prepare()`, `bind_param()`, `execute()`) az SQL injection támadások kivédésére.
- Ez a legfontosabb védelmi vonal az adatbázis-műveleteknél (INSERT, UPDATE, DELETE, SELECT WHERE).

Adatkezelés

- Felhasználói inputok feldolgozásakor `trim()` és `filter_input()` / `htmlspecialchars()` függvényeket használ.
- Ezek segítenek a felesleges szóközök eltávolításában, alapvető tisztításban és a Cross-Site Scripting (XSS) támadások kockázatának csökkentésében.
- Az `api.php`-ban és a módosító oldalakon `FILTER_SANITIZE_*` konstansok is megjelennek.

Jelszókezelés

- Jelszavak tárolása hashelve történik a `password_hash()` függvénnyel (PASSWORD_DEFAULT algoritmus).
- Bejelentkezéskor a `password_verify()` ellenőrzi a jelszót.
- Ez követi az ajánlott gyakorlatot a jelszavak biztonságos tárolására.

Fájlfeltöltés Biztonsága

- Az admin járműkezelő ellenőrzi a feltöltött fájlok MIME típusát és méretét.
- Egyedi, tisztított fájlneveket generál a felülírás és a potenciálisan rosszindulatú fájlnevek elkerülése érdekében.
- Ellenőrzi a célmappa létezését és írhatóságát.

További Megfontolások

- CSRF: A módosító oldalakon van egy alapvető ellenőrzés, ami összeveti a GET és POST ID-kat/neveket.
- Session Security: Nincs explicit session biztonsági beállítás a kódban, ezeket a `php.ini`-ben vagy a kód elején kellene konfigurálni.
- Hibakezelés: A kód néhol `die()`-t használ kritikus hibáknál, ami nem ideális éles környezetben.
- Jogosultságok: Az admin rendszer csak egy szintű (admin/nem admin).
- Directory Traversal: A képkezelésnél a fájlnevek tisztítása és az elérési utak ellenőrzése fontos.

Tesztelés és Jövőbeli Fejlesztések

Tesztelés (test_full_application.py)

A projekt tartalmaz egy Python alapú end-to-end teszt szkriptet, amely a requests könyvtárat használja.

- Célja: Automatikusan teszteli a webalkalmazás főbb funkcióit HTTP kérések küldésével és a válaszok ellenőrzésével.
- Lefedettség: Teszteli a főoldal elérhetőségét, regisztrációt, bejelentkezést, járművek oldalát, bérlet, véleményküldést, kapcsolatfelvételi űrlapot, hűségpont oldalt, fórumot, profil oldalt, kijelentkezést, valamint az összes adminisztrációs felület elérhetőségét.
- Futtatás: Python környezet szükséges a requests könyvtár telepítésével (pip install requests).

Ez egy alapvető end-to-end teszt, nem helyettesíti a részletesebb unit vagy integrációs teszteket.

Összegzés

Az R&J Autó kölcsönző egy funkcionálisan gazdag, PHP alapú webalkalmazás, amely sikeresen megvalósítja a kitűzött célokat mind a felhasználói, mind az adminisztrátori oldalon. Az alkalmazás alapvető biztonsági intézkedéseket (prepared statements, password hashing) alkalmaz. A kód struktúrája logikus, külön fájlokkal az egyes főbb funkciókhoz és egy elkülönített admin résszel.

A Bootstrap keretrendszer használata reszponzív és egységes megjelenést biztosít. Az API/végpontok és a kliensoldali JavaScript javítják a felhasználói élményt a profil oldalon és más interaktív elemeknél. A projekt tartalmaz egy alap end-to-end teszt szkriptet is.

A dokumentációban felsorolt jövőbeli fejlesztési lehetőségek mentén a rendszer tovább bővíthető és modernizálható. A projekt sikeresen demonstrálja a két fejlesztő (Jurcsó János Árpád - Felhasználói, Dávid Roland - Admin) együttműködését egy közös cél elérésében.

Jövőbeli Fejlesztési Lehetőségek

- Frontend Framework: Modern JavaScript keretrendszer (React, Vue, Angular) bevezetése.
- API: Teljeskörű REST API kialakítása a frontend és backend szétválasztásához.
- Online Fizetés: Integráció valamilyen fizetési szolgáltatóval (pl. Stripe, Barion).
- Részletesebb Keresés/Szűrés: Több szempontú, összetettebb szűrési és rendezési lehetőségek.
- Jelszó Visszaállítás: "Elfelejtett jelszó" funkció implementálása.
- Szerepkörök: Részletesebb jogosultságkezelés az admin felületen.
- Képkezelés Fejlesztése: Automatikus képátméretezés, vízjelezés, CDN használata.
- Naplózás: Részletesebb és strukturáltabb naplózás implementálása.
- Tesztelés: Unit és integrációs tesztek írása PHPUnit segítségével.
- Biztonság: CSRF tokenek implementálása, session kezelés szigorítása.