

**Debreceni Egyetem  
Informatikai kar**

*Web alapú alkalmazás fejlesztése*

***Témavezető:***

Dr. Rutkovszky Edéné  
Egyetemi tanársegéd

***Készítette:***

Mező András  
Zavaczki Zsolt  
Programtervező  
informatikus (BSc)

Debrecen  
2009

# Tartalomjegyzék

<b>I. Bevezető</b> .....	<b>4</b>
<b>II. Fejlesztői eszköztrendszer</b> .....	<b>7</b>
<b>2.1 PHP</b> .....	<b>7</b>
<i>Mi az a PHP?</i> .....	7
<i>Mit tud a PHP?</i> .....	8
<i>A PHP rövid története</i> .....	10
<i>A PHP működése</i> .....	11
<b>2.2 HTML/XHTML</b> .....	<b>13</b>
<b>2.3 CSS</b> .....	<b>14</b>
<i>Mi is a CSS?</i> .....	14
<i>Rövid története</i> .....	14
<i>Miért is jó a CSS?</i> .....	15
<b>2.4 MySQL</b> .....	<b>16</b>
<i>Mi is a MySQL?</i> .....	16
<i>Programnyelvek és Platformok</i> .....	17
<i>Adminisztrációja</i> .....	17
<i>Jellemzői</i> .....	17
<b>2.5 Apache</b> .....	<b>18</b>
<i>Tulajdonságok és szolgáltatások</i> .....	19
<b>III. Alkalmazás bemutatása</b> .....	<b>20</b>
<b>3.1 Alkalmazás leírása</b> .....	<b>20</b>
<b>3.2 Az alkalmazás létrehozása a WEB-en</b> .....	<b>20</b>
<b>3.3 Az alkalmazás elérése a WEB-en</b> .....	<b>21</b>
<b>3.4 Az alkalmazás felépítése</b> .....	<b>22</b>
A. <i>Felhasználói oldalról</i> .....	22
<i>Főoldal</i> .....	22
<i>Regisztráció</i> .....	24
<i>Profil</i> .....	24
<i>Autószalon</i> .....	25
<i>Autókölcsonzó</i> .....	30
<i>Tesztvezetésre jelentkezés</i> .....	34
<i>Autószerviz</i> .....	37
B. <i>Adminisztrátori oldalról</i> .....	39
<i>Autóválaszték módosítása</i> .....	39
a) <i>Autó paramétereinek a módosítása</i> .....	39
b) <i>Autó törlése az adatbázisból</i> .....	40
c) <i>Új autó hozzáadása az adatbázishoz</i> .....	41

d) Megrendelések nyomon követése és törlése .....	43
Autókölcsönzések és Tesztvezetések kezelése .....	44
Autószerviz .....	48
C. Fejlesztői oldalról.....	51
Az adatbázis .....	51
a) Alapok.....	51
b) Táblák létrehozása .....	51
1. Otthoni számítógépen (internet-kapcsolat nélkül) .....	51
2. Tábla létrehozása a WEB-en .....	52
c) A táblák szerkezeti felépítése .....	53
1. A „felhasználó” tábla.....	53
2. Az „autók” tábla .....	54
3. Az „autolehet” tábla.....	56
4. A „kölcsönzések” tábla .....	56
5. A „tesztvezetések” tábla .....	57
6. A „megrendelés” tábla.....	57
7. A „szerviz” tábla.....	58
8. A „szervizjelentés” tábla.....	59
9. A „vélemény” tábla.....	59
<b>IV. Összefoglalás.....</b>	<b>61</b>
<b>V. Irodalomjegyzék.....</b>	<b>63</b>
<b>VI. Köszönetnyilvánítás .....</b>	<b>64</b>

## I. Bevezető

A szoftverfejlesztés világában a Web alkalmazás (Web application, WebApp) egy program, melyet a weben keresztül érünk el az interneten, vagy intranet hálózaton.

A Web alkalmazások népszerűségének oka, hogy az őket használó web-böngésző kliensek szinte minden gépen rendelkezésre állnak. Egy Web alkalmazás karbantartható a kliens gépek szoftverének változtatása nélkül. A legelterjedtebb Web alkalmazás a Web-mail, Web-áruház, online aukció, fórum, blog, de ezeken kívül számtalan létezik belőlük.

A korábbi kliens-szerver architektúráknál minden egyes alkalmazásnak saját kliens programja volt, ami felhasználói felületként szolgált, és amit minden számítógépre fel kellett külön telepíteni. Egy szerver-oldali frissítés általában a kliens-oldal szoftverének frissítésével járt, ami kevésbé hatékony megoldás. Ezzel ellentétben a Web alkalmazások dinamikusan generálnak szabványos formátumú Web dokumentumokat (HTML, XHTML), amit a web-böngészők támogatnak. A kliens-oldali szabványos nyelvű szkriptek, mint a JavaScript a böngészőkben már eleve megtalálható. Minden weboldal statikus dokumentumként érkezik a klienshez, viszont interaktív felhasználói élményt nyújt a beágyazott Web formokon keresztül. A kapcsolat közben a web-böngésző értelmezi és megjeleníti az oldalakat, és egy univerzális kliensként működik minden Web alkalmazás számára.

A Web felület kevés korlátot állít a kliens funkcionalitásának. Java, JavaScript, DHTML, Flash és más technikákkal, melyek segítségével lehetséges képernyőtartalmat megjeleníteni, hangot lejátszani, egerhez és billentyűzethez hozzáférni. Ezeket kombinálva egy operációs rendszerhez hasonló megjelenést tudunk elérni, melyet a felhasználó már jól ismer. Általános technikákat is támogat, mint pl. a drag and drop. Web fejlesztők gyakran használnak kliens-oldali szkripteket, hogy bővítsék a funkcionalitást, például oldal frissítése, újratöltése nélkül jelenítenek meg tartalmat. Az új technológiák a szerver-oldali nyelvek (pl. PHP) segítségével irányítják a kliens-oldali szkripteket. Ilyen technológia az AJAX, mely számos különböző technikákat alkalmaz a felhasználói élmény javítása érdekében.

A Web alkalmazások jelentős előnye, hogy támogatják az alapvető böngésző funkciókat, ezáltal függetlenek a kliens gép operációs rendszerétől, illetve verziójától. Ahelyett, hogy a kliens programokat minden operációs rendszer számára külön megírnánk, az alkalmazást egyszer kell kifejleszteni, és szinte minden platformon működőképes. Azonban következtelen HTML, CSS, és DOM implementációk és a browser specifikációk különbözősége problémákat jelenthet a Web alkalmazás fejlesztésében, és támogatásában. Eltérő felhasználói böngésző-beállítások (szkript futtatás tiltása, eltérő betűtípus beállítás, stb.) szintén zavaró a Web alkalmazások következetes implementációjánál. Egy másik szemlélet az Adobe Flash és Java kisalkalmazások (Java applet) használata. A legtöbb browser manapság már támogatja ezeket a platformokat, ezért az ilyen alkalmazások bevezetése is ugyanolyan egyszerű. Nagyobb irányítást tesznek lehetővé, megoldják a böngésző beállítási problémákat. Architektúráisan viszont a hagyományos kliens-szerver alkalmazásokra hasonlítanak, ezért

sokan vitatják a helyüket a Web alkalmazások között, ezért inkább „Rich Internet Application” (RIA) alkalmazásoknak nevezik őket.

A Web alkalmazásokat általában három rétegre lehet bontani. Az első réteg a web-böngésző, a középső réteg egy motor, mely dinamikus web tartalmat használ (pl. PHP, CGI, ASP, JSP), és az adatbázis réteg a harmadik. A web-böngészők kéréseket küldenek a középső rétegnek, ami kiszolgálja őket azáltal, hogy lekérdezéseket és módosításokat végez az adatbázisban, majd megjeleníti a felhasználói felületet.

Szoftverfejlesztő cégek egy feltörekvő stratégiája, hogy Web hozzáférést biztosítsanak a már létező helyi alkalmazásokhoz. Ez lehetséges oly módon is, hogy egy teljesen más böngésző alapú felületet fejlesztenek ki, de adaptálhatják a meglévő alkalmazást is egy eltérő megjelenítéssel. Azokat a cégeket, melyek ezt a stratégiát követik, hálózati alkalmazás szolgáltatóként (Application Service Provider, ASP) említünk. Ezek a cégek egyre nagyobb figyelmet kapnak a szoftverpiacon.

Sok különböző Web alkalmazás keretrendszer létezik, mely elősegíti a gyors alkalmazás fejlesztést, lehetővé téve a programozónak, hogy magas szintű leírást adjon a programról. Web alkalmazás keretrendszerek használata egyszerűsítheti a kódot, csökkentheti a hibák számát. Olyan alkalmazásoknál, melyek állandó feltörési kísérletnek vannak kitéve, a biztonsági problémák kiemelten fontosak. A keretrendszerek elősegíthetik a legjobb technikák használatát, pl. GET a POST után. A Web Application Security Consortium (WASC) és OWASP projekteket azzal a céllal fejlesztették és dokumentálták, hogy elkerüljék a Web alkalmazások biztonsági problémáit. A Web Application Security Scanner egy speciális szoftver, ami segít észrevenni a Web alkalmazások hibáit.

Ezen szakdolgozat témája egy Web alapú alkalmazás fejlesztése, történetesen egy olyan autókereskedés honlapja, ahol a felhasználóknak lehetőségük van autót rendelni, kölcsönözni, tesztvezetésre jelentkezni, illetve a szervizben lévő autók szervizelésének nyomon követésére.

A mai web-orientált világban a mindennapi életünk szinte minden területén találkozhatunk webes megoldásokkal, gondoljunk csak a sok-sok web-áruházra, vagy az online ügyfélszolgálatra az egyes szolgáltatások terén. A mai rohanó világunkban nagyon fontos, hogy mire mennyi időt fordítunk, ezért kedveljük a web illetve a webes alkalmazások által nyújtott lehetőségeket, melyeket otthonunkból vehetjük igénybe csupán egy böngésző segítségével. Így sok időt illetve pénzt is spórolhatunk, hiszen például ha autót szeretnénk vásárolni, akkor szétnézhetünk az internetes autókereskedések honlapjain és tájékozódhatunk az autók felszereltségeiről, árfekvéseiről és egyéb tulajdonságaikról, anélkül hogy kimozdulnánk otthonunkból. Ezért választottuk ezt a szakdolgozati témát.

Célunk:

- Bevezetés a webes programozásba
- Web-en keresztüli adatbázis-kezelés bemutatása

- A PHP + MySQL + Apache eszközcsoport használatának megismertetése

Manapság egy webes alkalmazás egyik legfontosabb része az adatbázis, az adatok megfelelő tárolása illetve védelme nagyon fontos napjainkban, amikor egyre több tranzakciót bonyolítunk le online (pl.: készpénz átutalás), vagy egyre több szolgáltatást veszünk igénybe az internet segítségével (pl.: online ügyintézés, online rendelés).

Alkalmazásunkhoz a PHP programozási nyelvet használtuk, melynek okai:

- Könnyen kezelhető és tanulható
- Nagyfokú stabilitás
- Platformfüggetlenség

A PHP más programozási nyelvvel ellentétben nem igényel semmiféle fejlesztői környezetet, csupán egy szövegszerkesztőre és egy web-böngészőre van szükségünk, és máris kezdhetjük a PHP programjaink írását. Az adatbázis kezeléséhez a MySQL-t használtuk, amely egyszerű, gyors és megbízható adatbázis-kezelést tesz lehetővé a PHP programunkban.

A programot három nézőpontból fogjuk megvizsgálni:

1. Felhasználói oldalról
2. Adminisztrátori oldalról
3. Fejlesztői (Programozói) oldalról

A felhasználói részben az alkalmazást használó „mezei” felhasználóknak próbálunk egy átfogó leírást adni a program működéséről és használatáról. Igyekeztünk olyan leírást készíteni, melyet az informatikában kevésbé jártas ember is meg tudjon érteni.

Az adminisztrátori részben az alkalmazás által nyújtott adminisztrátori lehetőségeket vizsgáljuk meg, amelyekhez a speciális adminisztrátori jogosultsággal rendelkező felhasználók férhetnek hozzá.

A fejlesztői részben pedig egy kicsit szakmai szemszögből közelítünk az alkalmazás felé, ez a leírás inkább az informatikusoknak szól, programozási eszközök, kódrészletek, stb.

Debrecen, 2009. március 9.

## II. Fejlesztői eszköztárszer

### 2.1 PHP

#### Mi az a PHP?

A **PHP** (Hypertext Preprocessor) az egyik legelterjedtebb webes programozási nyelv, ami mára általános célú nyelvvé nőtte ki magát. Ingyenessége, és széleskörű elérhetősége biztosítja a továbbra is folyamatosan növekvő felhasználói bázist számára. A nyelv Magyarországon is egyre népszerűbb a részben magyar dokumentációnak, magyar könyveknek, levelezési listáknak és tanfolyamoknak köszönhetően. A PHP gyorsan tanulható, de a tapasztalt programozók számára is hatékony eszközöket nyújt, ezért érdemes vele megismerkedni.

A PHP oldalak elkészítésénél a HTML-t gyakorlatilag csak mint formázást használják, ugyanis ezen lapok teljes funkcionalitása a PHP-re épül. Amikor egy PHP-ben megírt oldalt akarunk elérni, a kiszolgáló először feldolgozza a PHP utasításokat, és csak a kész (HTML) kimenetet küldi el a böngészőnek. Ehhez egy ún. interpretert (értelmezőt) használ, amely általában egy külső modulja a web-szervernek.

A PHP nyelv lényegében nagymértékű kiegészítése a HTML-nek, ugyanis rengeteg olyan feladat végezhető el vele, amelyre az ügyféloldali szkriptek nem képesek (vagy ha igen, korlátozottan). Ilyen pl. a bejelentkezés, az adatbázis-kezelés, fájlkezelés, kódolás, adategyeztetés, kapcsolatok létrehozása, e-mail küldése, adatfeldolgozás, dinamikus listakészítés stb. Minden olyan esetben, ahol nagyszámú ismétlődő feladatsort kell végrehajtani (pl. képek listázása és linkelése, listakészítés stb.), ott ez a programnyelv nagyszerű segítség.

A PHP programok futhatnak közönséges (parancssori) programként is, nem HTML oldalba építve. Ezt azonban ritkán használják.

Egy egyszerű példa:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <title>Példa</title>
  </head>
  <body>

    <? echo "Helló világ!"; ?>

  </body>
</html>
```

A fenti példán látható, hogy a PHP kódot a „<?” illetve a „>?” határoló jelek között ágyazhatjuk be a HTML illetve az XHTML kódokba. A „<?php” határoló jellel is kezdődhet a PHP kód, de ez már az újabb verziókban el is hagyható, csak a visszafelé kompatibilitás miatt maradt benne a nyelvben. PHP kód bárhol elhelyezhető, a HTML vagy XHTML részekben belül a fenti határoló jelekkel jelezve, hogy az PHP kód.

A PHP esetében a programkód a fordítást követően nem a kliens oldalon fut, hanem a Web-szerveren (pl.: Apache), ez fontos különbség a kliensoldali szkript nyelvektől. Tehát a kód végrehajtásra kerül a szerveren, majd HTML kód generálódik melyet elküld a kliensnek. A kliens így megkapja a lefuttatott kód eredményét anélkül, hogy bármit is tudna a mögötte található kódról. Ebből látszik, hogy a PHP oldalak forrásait megnézve, nem találkozunk PHP kódokkal, mivel azok belefordulnak a HTML vagy XHTML kódba.

A legjobb dolog a PHP használatában az, hogy nagyon egyszerűen elsajátítható azok számára is akik azelőtt keveset vagy egyáltalán nem foglalkoztak szkript programozással, továbbá rengeteg haladó lehetőségeket biztosít a professzionális programozók számára is. Habár a PHP a szerver-oldali programozásra fókuszál, felhasználhatósága ennél sokkal tovább mutat, sokkal több mindenre használható mint azt első megközelítésből gondolnánk.

## **Mit tud a PHP?**

A PHP az alábbi három fő területhez köthető:

- Szerver-oldali programozás. Ez a PHP eredeti és fő alkalmazási területe. Három dolog szükséges, hogy ezen a területen alkalmazhassuk. Szükségünk van egy PHP értelmezőre (CGI vagy szerver modul), Web-szerverre illetve valamilyen Web-böngészőre. A Web-szervernek megfelelően beállított PHP-vel kell rendelkeznie. A program kimenetét a Web-böngészővel tudjuk megnézni, mégpedig a szerveren keresztül a script elérésével. Ez a 3 komponens, akár egy otthoni személyi számítógépen is képes elfutni abban ez esetben, ha nem rendelkezünk internet-kapcsolattal vagy csak ismerkedünk a nyelvvel.
- Parancssori programozás. Megtehetjük azt is, hogy Web-szerver illetve Web-böngésző nélkül futtassuk a programunkat, csak egy PHP értelmezőre van szükségünk. Az ilyen fajta alkalmazási mód ideális olyan szkriptek írásához melynek futtatásához valamilyen ütemezés szükséges, vagy csak egyszerű szövegfeldolgozó szkripteket szeretnénk készíteni.
- Ablakos alkalmazások írása. A PHP nem a legmegfelelőbb programozási nyelv a grafikus alkalmazások készítéséhez, de persze ha eléggé ismerjük a nyelvet akkor néhány haladó PHP szolgáltatást használhatunk, például a PHP-Gimp ToolKit nevezetű eszközt, melynek segítségével lehetőségünk nyílik platform-független

alkalmazás fejlesztéséhez. A PHP-GTK nem része a standard PHP-nak, annak egyfajta kiterjesztése.

A PHP használható az ismert operációs rendszerek szinte mindegyikén, beleértve a Linuxot, sok Unix változatot, a Microsoft Windows-t, a Mac OS X rendszert, RISC OS-t és még sorolhatnánk. Támogatja napjaink legtöbb Web-szerverét, beleértve az Apache, Microsoft Internet Information Server, Personal Web Server, Netscape és iPlanet szervereket, az Oreilly Website Pro, Caudium, Xitami, OmniHTTPd és más szervereket. A legtöbb szerverhez a PHP rendelkezik modullal, a többi részükkel pedig mint CGI feldolgozó működik együtt.

Összefoglalva tehát, a PHP szabad operációs rendszer illetve Web-szerver választást biztosít számunkra, akárcsak a procedurális- vagy objektumorientált paradigma esetén. A PHP 5-ös verziója már teljesen lefedi az objektumorientált paradigma eszközrendszerét, így ebben a verzióban már ténylegesen tudunk objektumorientált programozást alkalmazni.

A PHP nem korlátozódik a HTML kimenetre lehetőség van arra, hogy a kimenetünk képeket, PDF fájlokat vagy flash videókat, melyeket akár futás közben is létrehozhatjuk. Természetesen előállíthatunk szöveges kimenetet is, akár XHTML vagy valamilyen XML formában is. A PHP képes ezeket a fájlokat előállítani, majd a fájlrendszerben tárolni úgy hogy mindeközben a fájlok a szerveren maradnak, ez a valóságban egy szerver-oldali gyorsító-tárként funkcionál az általunk használt dinamikus tartalmakhoz.

A PHP egyik legfontosabb tulajdonsága, hogy az adatbázisok nagyon széles körét támogatja. A nyelv az alábbi adatbázisok mindegyikét támogatja:

Adabas D	InterBase	PostgreSQL
dBase	FrontBase	SQLite
Empress	mSQL	Solid
FilePro (csak olvasásra)	Direct MS-SQL	Sybase
Hyperwave	MySQL	Velocis
IBM DB2	ODBC	Unix dbm
Informix	Oracle (OCI7 és OCI8)	
Ingres	Ovrimos	

## A PHP rövid története

A PHP, mint programozási nyelv Rasmus Lerdorf nevéhez fűződik, 1994 őszén alkotta meg. A PHP fejlődése kezdetén csak CGI-programok halmaza volt. Ezeket Lerdorf néhány Perl szkript lecserélésére írta, amelyeket honlapjának karbantartására (például önéletrajzának megjelenítésére és a látogatottság mérésére) használt. Később ezeket a programokat kombinálta a szintén általa írt *Form Interpreter* (űrlap-értelmező) alkalmazással, így jött létre a *PHP/FI*, ami már jóval szélesebb funkcionalitással bírt. Az új, C nyelven megírt változat képes volt adatbázisokhoz kapcsolódni és segítségével egyszerű dinamikus weboldalakat is létre lehetett hozni. Lerdorf 1995. június 8-án adta ki a PHP első nyilvános változatát (amely a „Personal Home Page Tools” nevet viselte), hogy külső segítséggel gyorsabbá tegye a hibák megtalálását és a kód további fejlesztését. Az új verziót *PHP 2*-nek nevezte el és már megtalálhatóak voltak benne a mai PHP alapvető tulajdonságai:

- a Perl-éhez hasonló változók
- az űrlapok kezelése és a HTML-kód beszúrásának lehetősége
- a PHP szintaktikája is hasonló volt a Perl-éhez, de annál jóval korlátoltabb, egyszerűbb és kevésbé egységes volt

1997-ben Zeev Suraski és Andi Gutmans, két izraeli fejlesztő újraírta az értelmezőt, ezzel megteremtve a *PHP 3* alapját, ekkor született meg a PHP új neve, a *PHP: Hypertext Preprocessor* rekurzív rövidítés is. A fejlesztőcsapat 1997 novemberében, két hónapnyi béta-tesztelés után kiadta a PHP/FI 2-t, majd megkezdődhetett a PHP 3 tesztelése és végül megjelenése 1998 júniusában. Suraski és Gutmans ekkor megint a PHP-mag újraírása mellett döntött, amiből 1999-ben megszületett a *Zend Engine*. A két fejlesztő ekkor alapította meg a *Zend Technologies*-t is, ami máig aktívan ellenőrzi a PHP fejlesztését.

A Zend Engine 1.0 által hajtott *PHP 4* 2000. május 4-én jelent meg. Ezt követte 2004. július 13-án a következő nagy mérföldkőnek számító, az új *Zend Engine II*-n alapuló *PHP 5*. Az ötös verzió sok újítást tartalmazott: fejlettebb objektum-orientált programozási lehetőségeket, a PDO (*PHP Data Objects*) adatbázis-absztrakciós kiterjesztést, és sok teljesítményt növelő javítást is.

A PHP 4 fejlesztése 2007 végén maradt abba, az utolsó megjelent verziója a 4.4.8 verziószámot viselte - a PHP Group ezekhez és a PHP 4 előző változataihoz 2008 augusztusáig biztosít biztonsági frissítéseket. Jelenleg a PHP 5 a nyelv egyetlen aktívan fejlesztett verziója. A hatos verzió bár már fejlesztés alatt áll, még nem jelent meg. 2008. február 4-étől a GoPHP5 kezdeményezés keretében több vezető PHP alapú szabad-szoftver projekt döntött a PHP 4-támogatás megszüntetése mellett, hogy ezzel mozdítsák előre az ötös verzióra való áttérés folyamatát.

A PHP 6-os verziójának néhány újdonsága:

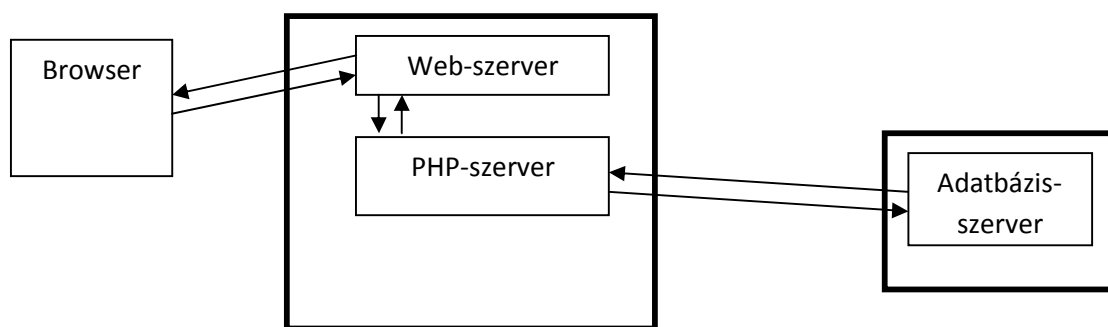
- ICU (International Components for Unicode)
- register\_globals, magic\_quotes, safe\_mode funkciók törlése
- 64 bites egészek kezelése
- goto megvalósítása a break kulcsszó kiterjesztésével
- foreach többdimenziós tömbökön és a névterek bevezetése

## A PHP működése

A PHP egy szerver-oldali szkript nyelv. A válaszlap formátumát leíró utasítások egy HTML dokumentumba ágyazva helyezkednek el. A Web-szerver a meghívott dokumentum URL-jéből ismeri fel, hogy a kérést a PHP szervernek kell feldolgoznia. A minta dokumentum HTML utasításai változtatás nélkül átkerülnek a válaszlagra, a PHP specifikus elemeket a PHP végrehajtja, és a generált válasz kerül be az eredménylapra. Ugyan a PHP leíró nyelve nem adatbázis orientált, hanem a C általános programozási nyelvre épül, mégis tartalmaz adatbázis kapcsolódási kiegészítő modult, amely lehetővé teszi, hogy elérhessük, és a válaszlapon megjeleníthessük a legkülönbözőbb adatbázisokban tárolt adatokat is.

A PHP rendszer működési struktúrája az alábbi elemekből épül fel:

- Web-szerver
- PHP-szerver
- Minta dokumentum



A fenti struktúra alapján a PHP kérések feldolgozása az alábbi lépésekben fut le:

1. Az olvasott lapon kiadnak egy hiperlink kapcsolódást egy olyan URL-re, mely mögött PHP hívás rejlik. Egy ilyen cím lehet például a következő:

`http://meza.uw.hu/php/index.php`

illetve a hivatkozást más esetekben a

`http://meza.uw.hu/index.php`

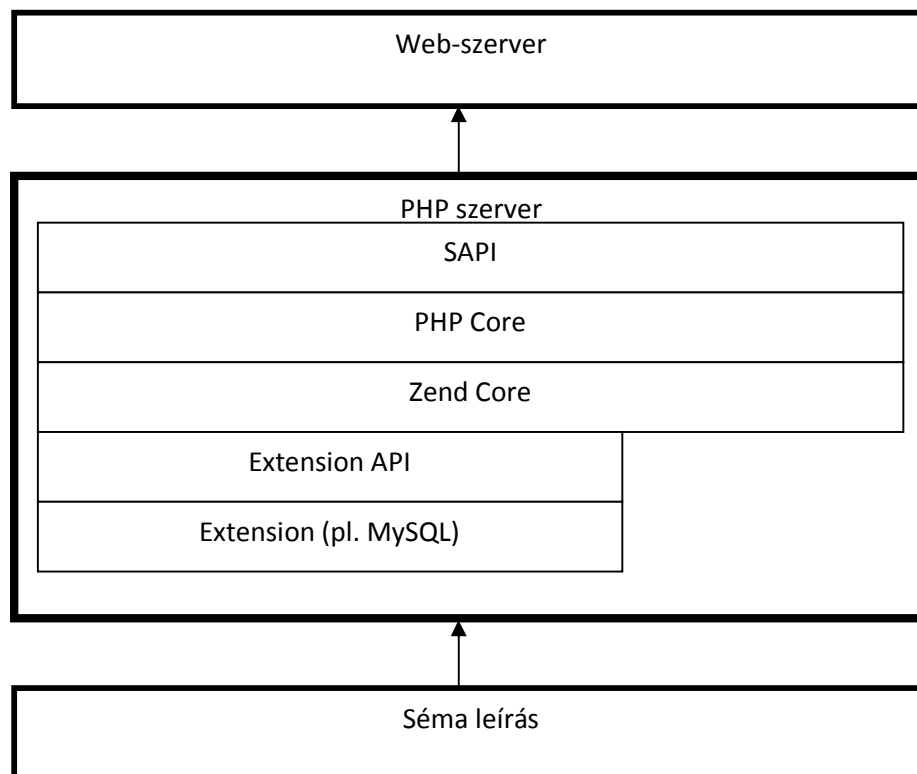
formában is megadhatjuk.

A hivatkozásokban az URL első része jelöli ki, hogy mely gépen található a Web-szerver és a PHP szerver. Az első mintában a /php/ rész egy virtuális könyvtár, mely jelzi a Web-szervernek, hogy itt egy PHP dokumentumot kell kezelni, és meg kell majd hívni a PHP szerveret. A virtuális útvonal kijelölésének menete Web-szerver specifikus, rendszerint a Web-szerver konfigurációs állományát kell módosítani a virtuális útvonal kijelöléshez. A második mintában a nem kellett virtuális könyvtár ahhoz, hogy a Web-szerver tudja, hogy a PHP szerveret kel meghívnia. Ebben az esetben ugyanis a dokumentum kiterjesztése utal arra, hogy a dokumentumot a PHP fogja kezelni. A kiterjesztések és a kezelő programok összerendelése is a Web-szerver konfigurációs

állományában történik. Az URL utolsó tagja, példánkban az *index.php* rész a séma dokumentumot jelöli ki, mely megadja a válaszlapp előállításának módját.

2. A Web-szerver érzékeli, hogy a meghívott URL a PHP szerverhez kapcsolódik, ezért meghívja a PHP modult végrehajtásra. A PHP szerver kétféle módon is működhet: CGI és API módban. Az első lehetőség esetén minden oldal lekérésekor és feldolgozásakor egy új példány fut le a PHP feldolgozóból. Mivel a szkript futtatása után egy ilyen példány leáll, minden erőforrás, amit lefoglalt, megszűnik. A népszerűbb a második forma, amikor a PHP-t modulként futtatjuk egy több process-es Web-szerveren. Egy több process-es Web-szerver tipikusan rendelkezik egy szülő process-el, ami koordinálja a többi kapcsolódó process (a gyermekek) munkáját, amik valójában a weboldalak kiszolgálását végzik. Ha egy kérés érkezik egy klienstől, egy éppen szabad gyermekprocess kapja meg a kiszolgálásra az utasítást.
3. A PHP modul megkapja a végrehajtási feladatot a feldolgozandó séma dokumentum azonosítójával együtt, átolvassa és elemzi a séma dokumentumot. A PHP specifikus utasításokat értelmezi és végrehajtja.
4. Ha adatbázis hozzáférési igények is felmerültek a PHP specifikus utasítások között, akkor a PHP szerver kapcsolatba lép az adatbázis-kezelővel, s végrehajtja a kijelölt műveletsort. A PHP specifikus utasítások között megtalálhatóak azon elemek is melyekkel leírható, hogyan jelenjenek meg az eredményadatok a válaszdokumentumban.
5. A generált válaszlapp a HTML utasítások és a dinamikus elemekből felépülve, átkerül a Web-szerverre, majd onnan eljut az őt igénylő böngészőhöz.

A PHP szerver is egy többretegű, több komponensből álló rendszer. A rendszer elvi felépítési vázát mutatja be a következő ábra:



## 2.2 HTML/XHTML

A **HTML** (angolul: *HyperText Markup Language* = *hiperszöveges jelölőnyelv*) egy leíró nyelv, melyet weboldalak készítéséhez fejlesztettek ki, és mára már internetes szabvánnyá vált a W3C (World Wide Web Consortium) támogatásával. Az aktuális változata a 4.01, mely az SGML általános jelölőnyelv egy konkrét alkalmazása (azaz minden 4.01-es HTML dokumentum egyben az SGML dokumentumszabványnak is meg kell, hogy feleljen). Ezt tervek szerint lassan kiszorította volna az XHTML (ami a szintén SGML alapú XML leíró nyelven alapul), de az is lehetséges, hogy a HTML 5 veszi át a helyét.

HTML általában szöveges állományokban található meg olyan számítógépeken, melyek az internethez kapcsolódnak. Ezek az állományok tartalmazzák azokat a szimbólumokat, amelyek a megjelenítő programnak leírják, hogyan is kell megjeleníteni illetve feldolgozni az adott állomány tartalmát. Megjelenítő program lehet egy web-böngésző (angolul: web browser), aural böngésző (olyan, amelyik a felhasználónak felolvassa a megjelenítendő szöveget), braille olvasó, amely konvertálja a szöveget braille "formátumba", levelező program (mint például: Mozilla Thunderbird, Microsoft Outlook, Eudora stb.), valamint egyéb eszközök, például mobiltelefon.

Egy HTML állomány három fő részre bontható:

1. A Dokumentum Típus Definíció az állomány legelején, ami a használni kívánt DTD-t adja meg, pl:  

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```
2. a HTML fejléc `<head>`, ami technikai és dokumentációs adatokat tartalmaz, melyeket az internet böngésző nem jelenít meg, tehát átlag felhasználó ezeket nem látja
3. a HTML törzs `<body>`, amely a megjelenítendő információkat tartalmazza.

Egy internetes oldal alapszerkezete a következő:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">  
<html>  
  <head>  
    <title>Az oldal címe</title>  
    <meta http-equiv="Content-type" content="text/html; charset=iso-8859-2">  
    [esetleges további fejléc-információk]  
  </head>  
  <body>  
    Az oldal tartalma  
  </body>  
</html>
```

Egy HTML dokumentum alapértelmezésként ISO-8859-1 kódolást használ. Az XHTML a HTML megfogalmazása XML-ben (a HTML SGML-ben van definiálva). Gyakorlatilag nincs jelentős eltérés a két nyelv között, csak a formai követelmények lettek szigorúbbak.

## 2.3 CSS

### Mi is a CSS?

A CSS (angolul *Cascading Style Sheets*) a számítástechnikában egy stílusleíró nyelv, mely a HTML vagy XHTML típusú strukturált dokumentumok megjelenését írja le. Ezen kívül használható bármilyen XML alapú dokumentum stílusának leírására is, mint például az SVG, XUL stb. A CSS specifikációját a *World Wide Web Consortium* felügyeli.

A CSS-t a weblapok szerkesztői és olvasói egyaránt használhatják, hogy átállítsák vele a lapok színét, betűtípusait, elrendezését, és más megjelenéshez kapcsolódó elemeit. A tervezése során a legfontosabb szempont az volt, hogy elkülönítsék a dokumentumok struktúráját (melyet HTML vagy egy hasonló leíró nyelvben lehet megadni) a dokumentum megjelenésétől (melyet CSS-sel lehet megadni). Az ilyen elkülönítésnek több haszna is van, egyrészt növeli a weblapok használhatóságát, rugalmasságát és a megjelenés kezelhetőségét, másrészt csökkenti a dokumentum tartalmi struktúrájának komplexitását. A CSS ugyancsak alkalmas arra, hogy a dokumentum stílusát a megjelenítési módszer függvényében adja meg, így elkülöníthető a dokumentum formája a képernyőn, nyomtatási lapon, hangos böngészőben (mely beszédszintetizátor segítségével olvassa fel a weblapok szövegét), vagy braille készüléken megjelenítve.

A CSS használható XML fájlok megjelenítésére is, így a strukturált dokumentumokhoz teljes körű stílust lehet megadni, befolyásolva az elrendezését, színét, betűtípusait az erre alkalmas kliensekben vagy web-böngészőkben.

### Rövid története

A stíluslapok már a HTML kezdete, 1990 óta jelen vannak valamilyen formában. A böngészők a stílus módosítására létrehozták saját leíró nyelvüket, melyet a webes dokumentumok megjelenésének módosítására lehetett használni. Eredetileg a stíluslapokat a felhasználók használták, mivel a HTML korai verziói még csak kevés prezentációs attribútumot tartalmaztak, így gyakran bízták a felhasználóra, hogy a webes dokumentumok hogyan jelenjenek meg.

A web-fejlesztők igényei folyamatosan növekedtek a stilisztikai lehetőségek irányába, így a HTML nyelvbe egyre több ilyen elem került. Ilyen lehetőségek mellett a stíluslapok kevésbé voltak fontosak, és egyetlen külső stílusleíró nyelv sem lett széleskörűen elfogadva a CSS megjelenése előtt.

A CSS eredetileg Hakon Wium Lie ötlete volt 1994-ben. Bert Bos időközben egy Argo nevű böngészőn dolgozott, mely saját stíluslapokat használt, végül ők ketten döntöttek a CSS kifejlesztése mellett.

Ekkor már több stílusleíró nyelv is létezett, de a CSS volt az első, ami a *kapcsolás* ötletét használta fel, vagyis a dokumentum stílusa több különböző stíluslapból tevődhetett össze. Ezáltal lehetőség nyílt arra, hogy a felhasználó által megadott stílus bizonyos esetekben felülírja a szerző stílusát, míg a többi esetben örökli azt. A stíluslapok ilyen kapcsolása mind a

szerző, mind a felhasználó számára rugalmas vezérlést biztosított, mivel megengedte a vegyes stilisztikai beállításokat.

Hakon ajánlata a "Mosaic és a Web" konferencián került bemutatásra Chicagóban, először 1994-ben, majd később 1995-ben. Ebben az időben alakult meg a World Wide Web Consortium is, mely később fellépett a CSS érdekében, és megalapított egy bizottságot a részletes kidolgozására. Hakon és Bert volt az elsődleges technikai vezetője a projektnek, melyhez további tagok csatlakoztak, többek között Thomas Reardon a Microsoft-tól. 1996 decemberében a CSS level 1 ajánlata hivatalosan is megjelent.

1997 elején a CSS egy Chris Lilley vezette csoporthoz került a W3C-nél. A csoport azokkal a problémákkal foglalkozott, melyeket a CSS Level 1 kihagyott. A CSS Level 2, mint hivatalos ajánlat, 1998 májusában jelent meg. A CSS Level 3 még jelenleg is fejlesztés alatt áll.

## **Miért is jó a CSS?**

A CSS információkat a lapokhoz több módon lehet megadni:

- Szerzői stílus
  - külső CSS fájl a dokumentumból hivatkozva
  - beágyazva a dokumentumba
  - azonnali, felülírva az általános stílust egy konkrét esetre
- Felhasználói stílus
  - egy helyi CSS fájl az összes dokumentumra, melyet a felhasználó a böngésző beállításában adhat meg; a felhasználó megadhatja, hogy a szerző és a saját stílusai közül melyik legyen fontosabb
- Kliens stílus
  - egy alapértelmezett stílus, melyet a kliens vagy a böngésző használ az elemekhez

A CSS a szabályokat a kapcsolódások alapján súlyozza, így mikor egyszerre több szabály is érvényes lehet egy elemre, akkor a legfontosabb szabályt alkalmazza.

A CSS használatának legfontosabb előnyei:

- Több lap vagy akár egy teljes webhely stílusait egy helyen lehet tárolni, így gyorsan és könnyen frissíthető.
- Különböző felhasználókhöz különböző stílusokat lehet rendelni: például könnyen olvasható, nyomtatható stílusok.
- A dokumentum mérete és komplexitása csökken, mivel nem tartalmaz információkat a megjelenítéshez.

A CSS egyszerű szintaxissal rendelkezik, csak néhány angol nyelvű kulcsszót használ a stílusok tulajdonságaihoz. A stíluslap maga a stílust leíró szabályok sora. Minden szabályhoz tartozik egy szelektor és egy deklarációs szakasz. Ez utóbbi kapcsos zárójelek között pontosvesszővel elválasztott deklarációkat tartalmaz. A deklarációk formája a következő: a tulajdonság neve, egy kettőspont, majd az adott tulajdonság értéke.

Például:

```
p {
    font-family: "Garamond", serif;
}
h2 {
    font-size: 110%;
    color: red;
    background-color: white;
}
.megjegyzes {
    color: red;
    background-color: yellow;
    font-weight: bold;
}
p#bevezetes {
    border: 3px dashed black;
}
a:hover {
    color: yellow;
    background-color: navy;
}
```

## 2.4 MySQL

### Mi is a MySQL?

A MySQL egy többfelhasználós, többszálú, SQL-alapú relációs adatbázis-kezelő szerver. A szoftver fejlesztője a svéd MySQL AB cég, amely kettős licenccel teszi elérhetővé a MySQL-t; választható módon vagy a GPL, vagy egy kereskedelmi licenc érvényes a felhasználásra. 2008 januárjában a Sun felvásárolta 800 millió dollárért a céget.

Az MySQL az egyik legelterjedtebb adatbázis-kezelő, aminek egyik oka lehet, hogy a teljesen nyílt forráskódú LAMP (Linux–Apache–MySQL–PHP) összeállítás részeként költséghatékony és egyszerűen beállítható megoldást ad dinamikus web-helyek szolgáltatására.

A mai adatbázisszerverek közül az egyik legszélesebb körben ismert és használt adatbázisszerver, általában a PHP programozási nyelvvel kapcsolatban találkozhatunk vele, de ezen kívül még sok más programozási nyelvet illetve platformot támogat. Bár manapság a sok adatbázisszerver jelenlétével valamelyest veszített a népszerűségéből, de ha PHP-ban szeretnénk programozni még mindig a MySQL a legjobb választás, mint adatbázisszerver.

## Programnyelvek és Platformok

Egyedi illesztő felületekkel az adatbázis-kezelő elérhető C, C++ , C#, Delphi, Eiffel, Smalltalk, Java, Lisp, Perl, PHP, Python, Ruby és Tcl programozási nyelvvel. Egy *MyODBC* nevű ODBC interfész további, ODBC-t kezelő nyelvek számára is hozzáférhetővé teszi az adatbázis-kezelőt. A MySQL számára az ANSI C a natív nyelv.

A MySQL különböző platformokon futtatható: AIX, BSDi, FreeBSD, HP-UX, Linux, Mac OS X, NetBSD, 0Netware, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP és a Windows frissebb verziói. Már MacOSX 10.4-re is elérhető.

## Adminisztrációja

Két parancssori eszköz létezik az adatbázisok adminisztrációjára, a MySQL és a MySQLAdmin. Ha a grafikus felületet részesítjük előnyben, akkor a MySQL Administrator és a MySQL Browser áll a rendelkezésünkre, melyek letölthetők a MySQL hivatalos honlapjáról.

Széles körben elterjedt és népszerű adminisztrációs eszköz a PHP nyelven írt, nyitott forráskódú PHPMyAdmin. A PHPMyAdmin egy vizuális rendszer az adatbázisok kezeléséhez. Ez írja és megjeleníti az adatbázisok tartalmát a szerveren (vagy a kliensnél), ahol a MySQL telepítve van. Ezen az interfészen keresztül létrehozatsz új adatbázisokat, módosíthatsz már létezőket és beavatkozatsz az egyes mezők tartalmába. Képes SQL parancsokat futtatni és a mezőkön kulcsokat kezelni, továbbá képes az egész MySQL szerver kezelésére (szuper-felhasználót igényel) épp úgy, mint egyetlen adatbázisra. Az utóbbi megvalósításához be kell állítani a MySQL felhasználót, hogy csak a kívánt adatbázist tudja írni/olvasni.

Továbbá létezik még egy szintén PHP nyelven írt eszköz a PHPMyBackUpPro, amely adatbázisok akár időzített, ismétlődő mentésére szolgál.

## Jellemzői

A MySQL egy nagyon gyors, több-felhasználós, és többszálás robusztus SQL adatbázisszerver. A MySQL 5-ös verziójától kezdve többek között lehetőségünk van tárolt eljárások, adatbázis triggerek, kurzor adatbázisok, „view” adatbázisok, valódi VARCHAR támogatás, szigorú mód, különálló tároló motorok, SSL támogatás, egymásba ágyazott SELECT-ek, beágyazott adatbázis-könyvtár, példányosítás használatához.

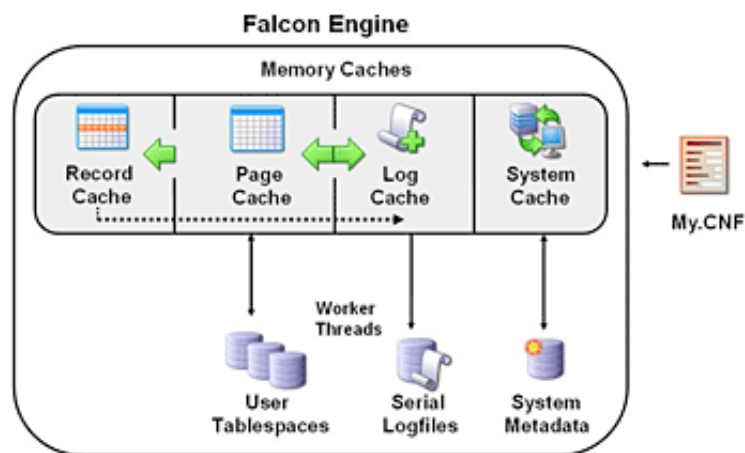
A következő képességekkel a MySQL rendelkezik, számos más relációs adatbázisrendszerrel ellentétben:

- Többféle tároló motor, amelyek között bármely táblához szabadon választhatunk.
- Natív tároló motorok (MyISAM, Falcon, Merge, Memory (heap), MySQL Federated, MySQL Archive, CSV, Blackhole, MySQL Cluster, Berkeley DB, EXAMPLE, és Maria)
- Partnerek által fejlesztett tároló motorok (InnoDB, solidDB, NitroEDB, BrightHouse)

- Közösségi fejlesztésű tároló motorok (memcached, httpd, PBXT, Revision Engine)
- Akár egyéni tároló motor
- Commitek csoportosítása, több tranzakció fogadása többféle kapcsolatról, melyek meggyorsítják a tranzakciók lefolyását.

Jelenleg a MySQL 6-os verziójánál tartunk, ez a verzió még nagyobb megbízhatósággal, teljesítménnyel, egyszerűbb használhatósággal rendelkezik, mindez az új Falcon tranzakció tároló motor segítségével valósul meg.

Az alábbi ábra a Falcon motor működését mutatja:



## 2.5 Apache

### Mi az Apache?

Az Apache HTTP Server (röviden Apache) egy nyílt forráskódú webkiszolgáló alkalmazás, szabad szoftver, mely kulcsfontosságú szerepet játszott a World Wide Web elterjedésében. A projekt célja egy olyan web-szerver program létrehozása, karbantartása, és fejlesztése, amely megfelel a gyorsan változó Internet követelményeinek, biztonságos, üzleti, vállalati felhasználásra is megfelelő és szabadon használható. Az Apache a régi NCSA httpd szerverre épül, az Apache Szoftver Licenc feltételei alatt terjesztik.

Az Apache egy robusztus, erőteljes és rugalmas web-szerver, amely kompatibilis a HTTP/1.1 (RFC2616) protokollal. Az Apache projekt koordinálását az Apache Software Foundation végzi. Néhány vezető és több száz fejlesztő van e projekt mögött.

Az Apache volt az első használható alternatíva a Netscape Communications Corporation web-szerverrel szemben (melynek mai neve *Sun Java System Web Server*). A későbbiekben továbbfejlődött és más unix alapú web-szerverekkel is felvette a versenyt funkcionalitás és teljesítmény tekintetében.

Többek között a következő operációs rendszerekhez készítették el az Apache-ot: Unix, FreeBSD, Linux, Solaris, Novell NetWare, Mac OS X és Microsoft Windows.

## **Tulajdonságok és szolgáltatások**

Az Apache-t valószínűleg sokaknak nem kell bemutatnunk. Gyors, biztonságos, fejleszthető, és a forráskódja is szép. Mivel szabadon használható, minden Linux disztribúcióba belekerült, így elég könnyen beszerezhető. A platform támogatottsága elég jó, fut akár Win32-n is, bár ezen a platformon nem képes a Linuxon megszokott biztonságot és stabilitást biztosítani. Veszélyes vetélytársai nincsenek, a web-szerverek több mint 60 százalékát működtetik Apache-al. Erőforrásigényei szolidak.

A fejlesztés jelenleg az 1.3.x sorozat végén tart, a nemrég kibocsájtott 1.3.19-es verziót az utolsónak szánták és béta stádiumba lépett az Apache 2 fejlesztése. Nagy újítás lesz benne a thread/process hybrid üzemmód, és sok új modul is belekerül. Sajnos, ha 1.3.x alá írunk egy modult, akkor az csak átalakítások után lesz használható a 2.x-es verziókban.

Az Apache valóban kiválóan megírt szoftver. az alapfunkcionalitást megvalósító core modul köré sok apró feladatot megvalósító modul épül. Ilyen modul valósítja meg például az http azonosítást is, de például a biztonságos HTTPS protokoll is. A modulok befordíthatóak az Apache bináris kódjába statikusan is, de egy modul (mod\_so) segítségével dinamikusan is betölthetőek. Így egy könnyen bővíthető, egyszerűen konfigurálható web-szervert kapunk. A standard Apache is nagyon sok modult tartalmaz, de fordíthatsz vagy építhetsz hozzá magad is.

Az Apache sok szabványt támogat, melyeknek nagy része fordított modulok formájában áll rendelkezésre a mag kiegészítéseként. Ezek a modulok sok területet lefednek a kiszolgáló-oldali programnyelv támogatástól kezdve a hitelesítési sémáig. Az ismertebb, támogatott programnyelv modulok a mod\_perl, a mod\_python, a Tcl és a PHP. A népszerűbb hitelesítési modulok a mod\_access, mod\_auth és a mod\_digest. További modulokban megvalósított tulajdonságokra példa még az SSL és TLS támogatást nyújtó mod\_ssl, a proxy modul, egy hasznos URL átíró a mod\_rewrite modulban, testreszabható logolás a mod\_log\_config modulban és szűrési támogatás a mod\_include és mod\_ext\_filter modulok segítségével.

A mod\_gzip modul tömöríteni tud, segítségével csökkenthető a weboldalak mérete. Az Apache logok elemezhetőek böngészővel az AWStats/W3Perl vagy a Visitors programok segítségével.

A virtual hosting segítségével egyetlen Apache installáció ki tud szolgálni több különböző webhelyet is. Például egyetlen gép egyetlen Apache szerverrel képes kiszolgálni a www.valami.com, a teszt42.trallala.hu és a www.burgonya.org címekre érkező HTTP kéréseket.

Az Apache hibaüzenetei konfigurálhatóak. Számos grafikus felhasználói felület (röviden GUI) támogatja a szerver könnyebb és intuitívabb konfigurálását.

2009 márciusában az Apache a Föld webhelyeinek 46%-át szolgálja ki.

## **III. Az alkalmazás bemutatása**

### **3.1 Az alkalmazás leírása**

Az alkalmazás egy autókereskedés weboldalát mutatja be, ahol lehetőség van

- a szalon által értékesítendő autók közötti böngészésre, paramétereinek megtekintésére
- a kiválasztott autó megrendelésére, az alap vagy más paraméterekkel
- autókölcsönzésre a kölcsönözhető autókból
- tesztvezetésre jelentkezésre, a megadott tesztautó kiválasztásával
- szervizben lévő autók szervizelésének nyomon követésére.

Az alkalmazás mindenki számára elérhető, aki rendelkezik internet hozzáféréssel. A fenti szolgáltatások csupán egy web-böngésző segítségével igénybe vehetők.

### **3.2 Az alkalmazás létrehozása a Web-en**

Ezen alkalmazás kapcsán nem beszélhetünk telepítési útmutatóról, hiszen nem egy konkrét programról van szó, melyet egyik számítógépről a másikra ilyen-olyan adathordozón keresztül tovább lehet vinni. Egy általános programnak annyi példánya lehet, amennyit csak akarunk, ennek az alkalmazásnak viszont csak egy. Egy, de ahhoz bárki bárholnan hozzáférhet.

Bizonyos ingyenes Web-tárhelyek lehetőséget adnak arra, hogy a PHP programjainkat szerverük segítségével futtathassuk le, sőt ennek tetejében még MySQL szolgáltatást is biztosítanak a számunkra. A következő dolgokat kell tenni ahhoz, hogy ezt az alkalmazást mindenki számára elérhetővé tegyük:

#### ***1. Keresünk egy Web-szervert, mely PHP és MySQL támogatással rendelkezik.***

Rengeteg Web-szerver létezik, melyek eleget tesznek ezeknek a kritériumoknak. Mi az ULTRAWEBSZERVER-t választottuk. Ez a Web-szerver fogja majd az alkalmazásunkat futtatni.

#### ***2. Regisztrálunk a Web-szerverre***

Ezt nagyon egyszerűen megtehetjük, hiszen a szerver honlapján – melynek címe: [www.ultraweb.hu](http://www.ultraweb.hu) – a REGISZTRÁCIÓ gombbal csak egy formot kell kitöltenünk. Ez egy egyszerű kérdőívhez hasonlítható: a felhasználói nevet, személyes adatokat és a honlap adatait fogja majd kérni a form. Ezen adatok ismertetében fog minket az ULTRAWEBSZERVER regisztrálni. Ha a böngészőnk URL-jébe a megadott felhasználói név

után hozzáillesztjük a „uw.hu” sztringet akkor már hivatkozhatunk is a weblapunkra, ami persze még teljesen üres lesz. Formálisan a [www.felhasznaloi\\_nev.uw.hu](http://www.felhasznaloi_nev.uw.hu) lesz a mi weblapunk.

Regisztrációkhoz kapunk: 200 MB ingyenes tárhelyet, 20 MB MySQL tárhelyet, melyen az adatbázis tárolódik, SQL PHP futtatási lehetőséget, FTP hozzáférést, stb.

### **3. Feltöltjük az adatbázist a szerverre**

Miután bejelentkeztünk a szerverre az ULTRAWEB honlapján az előzőleg megadott azonosítókkal, annyi dolgunk van, hogy létrehozzuk és feltöltsük az adatbázist. Kattintsunk a baloldalon a MySQL menüpontra, majd az Adatbázis létrehozása gombra. Ennek hatására létrejött a szerveren a felhasználói nevünkkel megegyező nevű adatbázis, ezután a PHPMYADMIN gombra kattintva elindul ez a grafikus felhasználói felületű adatbázis kezelő szkript. Ezek után az Importálás menüpont alatt megadhatjuk annak az SQL szkript fájl elérési útját, amely létrehozza és esetlegesen feltölti a tábláinkat. Ezen műveletek elvégzése után az adatbázisunk el is készült az ULTRAWEB szerverén.

### **4. Feltöltjük a PHP fájlokat**

Mivel a Web-szerver FTP szolgáltatást is biztosít, ezért nagyon kényelmesen fel tudunk rá tölteni bármit, amit akarunk. Elindítjuk a kedvenc fájlkezelő programunkat, és azon keresztül fogunk kapcsolódni az ULTRAWEB szerveréhez. Három fontos dolgot kell megadni a kapcsolódáshoz: a kapcsolat nevét (bármilyen karaktersorozat), a kiszolgáló nevét (kötelezően: *ftp.uw.hu*), és a felhasználói nevet (melyet még a regisztrációnál adtunk meg). Innentől kezdve már csatlakozhatunk is a Web-szerverhez. Az utolsó momentum, hogy átmásoljuk rá a PHP programokat. NAGYON FONTOS: A szerveren lennie kell egy *index.php* vagy *index.html* nevű fájlnek is, mert a szerver ezt fogja keresni, mikor a URL-be beírjuk a címet: *www.felhasznaloi\_nev.uw.hu*. Ha nem adunk meg ilyen fájlt, akkor hibüzenetet kapunk.

## **3.3 Az alkalmazás elérése a Web-en**

A felhasználónak az alkalmazás eléréséhez csupán egy böngészőre van szüksége és máris használhatja az alkalmazást. Az alkalmazást bármely böngésző segítségével az alábbi címen érhetjük el:

[www.meza.uw.hu](http://www.meza.uw.hu)

Ezek után az alkalmazás grafikus felhasználói felülete fogad minket, amely megkönnyíti a felhasználó dolgát az alkalmazás használata közben, és amely nagyon fontos szerepet játszik a mai web alapú alkalmazások világában.

## 3.4 Az alkalmazás felépítése

### A. Felhasználói oldalról

#### Főoldal

Mikor a böngészőbe beírjuk a következő URL címet: [www.meza.uw.hu](http://www.meza.uw.hu) akkor a Web-szerver elindítja az alkalmazást, és a következő képernyőt látjuk:

**Mező és Zavacski autóvilág**  
autószalon, autóértékesítő hely, teszttvezetési lehetőség, autószervez

Kezdőlap   Autószalon   Autó kölcsönző   Tesztvezetés   Szervíz

**Bejelentkezés**

Azonosító:

Jelszó:

Bejelentkezés

[Regisztráció](#)

**Naptár**

Ápril						
V	H	K	Sze	Cs	P	Szo
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

**Rólunk**

### Üdvözljük honlapunkon!

A nem régóta működő cégünk fő profilja az autó értékesítés, autókölcsönzés, autószervezés és a teszttvezetési lehetőség.

Az autókereskedés mindenki számára biztonságos és a kényelmes vásárlást kínálja. A kiválasztott autó online rendelése esetén fenntartjuk Önnek a kiválasztott autót a megrendelt paraméterekkel, illetve ha nincs a járműparkunkban az ön által rendelt típus ezen paraméterekkel akkor lehetőség szerint a kívánt paraméterekkel megrendelésre kerül. Az autókereskedés a megvásárolni kívánt gépjárművének teljes körű ügyintézését vállalja, Önnek kizárólag csak autókereskedésünket kell meglátogatnia a vásárlás során. Ön helyett elvégezzük az ezzel kapcsolatos feladatokat, úgymint: autó hitel ügyintézés, szervizelés, alkatrész beszerzés, biztosítás kötése, eredetvizsgáztatás, okmányirodai ügyintézés. Használt autó hitel konstrukcióink casco biztosítás kötése nélkül, akár 10% kezdő befizetéssel indulnak.

A vételárba beszámítjuk használt autóját, akár hitellel terhelhet is. A gépkocsi megvásárlása előtt meggyőzése és megnyugtatósa érdekében biztosítjuk a gépjármű teljeskörű átvezetését az Ön által megjelölt szervizben. Az Ön jövőbeni biztonsága érdekében autóját akár garanciával is megvásárolhatja. Szabad felhasználású hitelt vehet fel, saját tulajdonú gépjárművére is! Az autofelvásárlás cégünk életének egyik fontos alap pillére hiszen egy autókereskedés e nélkül nem működik.

Szívesen várjuk azokat az ügyfeleket akik szeretnék eladni a már cserére megérett használt autójukat és újabbra szeretnék cserélni azt. Nálunk az autofelvásárlás gyorsan és egyszerűen történik, kollégáink szemrevételezik az ön eladásra kínált használt autóját, s amennyiben az általunk tett vételár Önnek kedvező, úgy azt a kereskedésünk azonnal készpénzben kifizeti Önnek.

Ez az alkalmazás főoldala (index.php) amely automatikusan betöltődik, amikor a fenti URL címet beírjuk a böngészőnkbe. A baloldalon látható a bejelentkező panel, ahol a regisztrált felhasználók beléphetnek, illetve aki még nem regisztrált az oldalra az a Regisztráció linkre kattintva egy regisztrációs formot kitöltve megteheti ezt. Továbbá a bal oldalon található egy naptár is, amely az aktuális év hónapját és napját jelzi a felhasználó számára.

Természetesen nem kötelező regisztrálnunk, ha csak böngészni szeretnénk az alkalmazás oldalain, viszont a későbbiekben látni fogjuk, hogy bizonyos funkciók illetve szolgáltatások csak bejelentkezett felhasználók számára hozzáférhető.

Ha hibás vagy nem létező felhasználói nevet vagy jelszót adunk meg, vagy netán üresen hagyjuk az *Azonosító* és *Jelszó* mezőket akkor a rendszer a *Hibás jelszó vagy felhasználónév!* hiba üzenettel válaszol, ahogy ez az alábbi képen látható:



The image shows a green-bordered box with a white background. At the top left, the title "Bejelentkezés" is written in blue. Below it, the label "Azonosító:" is followed by a white input field. Underneath, the label "Jelszó:" is followed by another white input field. A button labeled "Bejelentkezés" is centered below the fields. At the bottom, a red error message reads "Hibás jelszó vagy felhasználónév!". Below the error message, the word "Regisztráció" is written in purple and underlined.

Ilyenkor újra próbálkozhatunk a felhasználói azonosító illetve a jelszó megadásával vagy javításával. Ha mindkét mező helyesen lett kitöltve, akkor belépteti a rendszer a felhasználót és a kijelentkezési panelen megjelenik a felhasználó teljes neve, amit a regisztráció során adott meg. Továbbá a *Profil* link jelenik meg a *Regisztráció* helyett, amely segítségével a felhasználó a regisztráció során megadott adatait módosíthatja.

Kijelentkezési panel:



The image shows a green-bordered box with a white background. At the top left, the title "Kijelentkezés" is written in blue. Below it, the text "Üdvözöljük honlapunkon:" is displayed. Underneath, the name "Mező András" is shown. A button labeled "Kijelentkezés" is centered below the name. At the bottom, the word "Profil" is written in purple and underlined.

A főoldal további részét pedig egy leírás tesz ki, ami a web-alkalmazás által reprezentált autókereskedést mutatja be olyan szemszögből, hogy milyen szolgáltatások vehetők igénybe az autókereskedésben, beleértve az online illetve a helyszínen történő szolgáltatásokat is.

## Regisztráció

Ahogy említettük a *Regisztráció* a bal oldali bejelentkezési panelen található, és ha rákattintunk, akkor az alábbi regisztrációs formot kell kitöltenünk:

**Mező és Zavaczki autóvilág**  
autószalon, autóértékesítő hely, tesztvezetési lehetőség, autószervez

Kezdőlap Autószalon Autókölcsonzó Tesztvezetés Szerviz

### Regisztráció

Az ön teljes neve:

Felhasználói név:

Jelszó:

Jelszó újra:

Irányítószám:  Hiányzó irányítószám!

Város:  Hiányzó városnév!

Községi név:  Hiányzó községi név!

E-mail cím:

Sikertelen regisztráció!

Értelemszerűen megadjuk az adatainkat, és ha valamely általunk megadott adat nem megfelelő vagy nem adtuk meg azt az adatot a jobb oldali oszlopban látható módon a rendszer közli velünk, hogy mi a hiba. Ha valamely adat nem lett megadva vagy hibás akkor természetesen az adott személy nem lesz regisztrálva mindaddig, amíg a megfelelő adatokat a megfelelő formában meg nem adja. Ezért látható a fenti kép alján a *Sikertelen regisztráció!* üzenet, hiszen nem történik regisztráció. Ha minden adat szerepel, és nem tartalmaz hibákat akkor az oldal alján a *Sikeress regisztráció!* üzenet jelenik meg. Ezek után a korábban bemutatott módon be tud jelentkezni a felhasználó a rendszerbe a felhasználói azonosítójával és jelszavával.

## Profil

Amikor egy felhasználó be van jelentkezve a rendszerbe, akkor a bal oldali kijelentkezési panelen a *Profil* linkre kattintva, ellenőrizheti a regisztrációkor megadott adatait, illetve módosíthatja is azokat, ha elírás vagy valamilyen adata megváltozott (pl.: lakcím) vagy csak a jelszavát szeretné éppenséggel megváltoztatni.

A Profil oldal az alábbi képen látható:

**Mező és Zavaczki autóvilág**  
autószalon, autóértékesítő hely, teszttvezetési lehetőség, autószervíz

Kezdőlap   Autószalon   Autókölcsonzó   Tesztvezetés   Szervíz

### Profil

Az ön teljes neve:

Felhasználói név:

Jelszó:

Jelszó újra:

Irányítószám:

Város:

Községi terület neve:

E-mail cím:

Nem történt módosítás az adatbázisban.

A fenti képen látható, hogy ha rákattintunk a *Profil* linkre, akkor megjelennek azok az adatok, amelyeket a regisztráció során megadtunk, és az egyes szövegmezőkben lévő adatok átírásával tudjuk az adatainkat módosítani. Ha készen vagyunk az adatok módosításával, akkor a *Mentés* gombra kattintva elmenthetjük a módosított adatainkat, ha nem történt változás az adataink között, de mégis a *Mentésre* kattintunk akkor a *Nem történt módosítás az adatbázisban* üzenet jelenik meg az oldal alján. Abban az esetben pedig, ha nem akarjuk adatainkat módosítani, akkor, a böngésző *Vissza* gombjával visszatérhetünk a korábbi oldalra, vagy a fenti menüt is használhatjuk a navigáláshoz.

## Autószalon

Az alkalmazás autószalon része (szalon.php) az autókereskedés szalon szolgáltatásait reprezentálná természetesen online körülmények között. Ugyanúgy, ahogy az ember besétál egy autószalonba és megnézi az autók választékát, úgy ezt a mi alkalmazásunkban a *Szalon* menüpontra kattintva barangolhat a szalonban lévő autók között, továbbá lehetősége van keresésre is az autók között. A szalonban a nem bejelentkezett, illetve nem regisztrált ügynevezett „vendég” felhasználók is barangolhatnak és az autók képeire rákattintva szintén megtudhatnak további információkat is az autókról, viszont a továbbiakban látni fogjuk, hogy nem minden szolgáltatás eléréséhez van jogosultságuk.

Lássuk akkor, hogyan is kell ezt elképzelni a mi alkalmazásunkban:

## Mező és Zavacski autóvilág

autószalon, autóértékesítő hely, tesztvezetési lehetőség, autószervez

[Kezdőlap](#)   [Autószalon](#)   [Autókölcsonzó](#)   [Tesztvezetés](#)   [Szervíz](#)

**Bejelentkezés**

Azonosító:































Jelszó:

[Regisztráció](#)

Márka:  Tipus:  Szín:  Üzemanyag:  Ajtó:

Kategória:  Hajtott kerekek:  Injektor:  Ár:

Naptár						
Április						
V	H	K	Sze	Cs	P	Szo
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

		
Ár: 3.003.400 HUF	Ár: 4.166.200 HUF	Ár: 7.200.120 HUF
		
Ár: 3.843.400 HUF	Ár: 2.157.000 HUF	Ár: 3.399.400 HUF
		
Ár: 4.417.000 HUF	Ár: 6.296.200 HUF	Ár: 1.990.000 HUF
		
Ár: 2.250.000 HUF	Ár: 1.990.000 HUF	Ár: 2.680.000 HUF
		
Ár: 3.225.000 HUF	Ár: 2.850.000 HUF	Ár: 3.350.000 HUF
		
Ár: 2.350.000 HUF	Ár: 3.390.000 HUF	Ár: 3.445.000 HUF
		
Ár: 1.870.000 HUF	Ár: 2.600.000 HUF	Ár: 2.895.000 HUF
		
Ár: 2.190.000 HUF	Ár: 2.280.000 HUF	Ár: 2.800.000 HUF
		
Ár: 5.540.000 HUF	Ár: 3.235.000 HUF	Ár: 3.500.000 HUF
		
Ár: 3.800.000 HUF	Ár: 4.200.000 HUF	Ár: 5.120.000 HUF

[4. oldal](#)   [3. oldal](#)   [2. oldal](#)   [1. oldal](#)

A jobb alsó sarokban látható négy link segítségével tudunk barangolni az autók között a szalonban. A fenti keresőpanelen pedig kilenc paraméter szerint szűkíthetjük az autók körét, akár a kilenc paraméter együttes kombinálásával ugyanazon lekérdezésben. Ha nem található a megadott keresési paramétereknek megfelelő autó az adatbázisban, akkor a *Nincs a szűrés feltételnek megfelelő autó az adatbázisban!* üzenet jelenik meg, ahogy ezt az alábbi képen láthatjuk:

## Mező és Zavacski autóvilág

autószalon, autóértékesítő hely, teszttvezetési lehetőség, autószervíz

Kezdőlap   Autószalon   Autó kölcsönző   Tesztvezetés   Szervíz

**Bejelentkezés**

Azonosító:

Jelszó:

[Regisztráció](#)

Márka: Dacia   Tipus: Nincs   Szín: Nincs   Üzemanyag: Nincs   Ajtó: Nincs

Kategória: Nincs   Hajtott kerekek: Nincs   Injektor: Nincs   Ár: 7.000.000-0.000.000 között

Nincs a szűrési feltételnek megfelelő autó az adatbázisban!

Sikeres keresés esetén megjelennek a szűrési feltételnek megfelelő autók.

## Mező és Zavacski autóvilág

autószalon, autóértékesítő hely, teszttvezetési lehetőség, autószervíz

Kezdőlap   Autószalon   Autó kölcsönző   Tesztvezetés   Szervíz

**Bejelentkezés**

Azonosító:

Jelszó:

[Regisztráció](#)

Márka: Skoda   Tipus: Nincs   Szín: Nincs   Üzemanyag: Nincs   Ajtó: Nincs

Kategória: Nincs   Hajtott kerekek: Nincs   Injektor: Nincs   Ár: 7.000.000-0.000.000 között



Skoda Octavia Scout  
Az autó megvásárlásához, illetve az autó más paramétereivel való megrendeléséhez kattintson a képre!

Ár: 7.200.120 HUF

Ahogy látjuk, ahol nem választunk ki semmit a keresési panelen, ott a *Nincs*, szó jelenik meg a legördülő menüben, ez mutatja, hogy ezek a tulajdonságok nem vesznek részt a keresésben. Azt is láthatjuk, hogy ha rámutatunk az egérrel az autó képére, akkor megjelenik néhány információ, amely az autó márkája és típusa illetve iránymutatásként annyi, hogy az autó megrendeléséhez kattintsunk a képre.

Ha rákattintunk a képre egy másik oldalra irányít át minket a rendszer, ahol megrendelhetjük az autót az ott látható paramétereivel, illetve testre szabhatjuk az autó egyes paramétereit megadott határok között.

## Mező és Zavacski autóvilág

autószalon, autóértékesítő hely, teszttvezetési lehetőség, autószerelv

Kezdőlap   Autószalon   Autókölcsonző   Tesztvezetés   Szerviz

**Bejelentkezés**

Azonosító:

Jelszó:

[Regisztráció](#)

**Naptár**

Ápril						
V	H	K	Sze	Cs	P	Szo
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

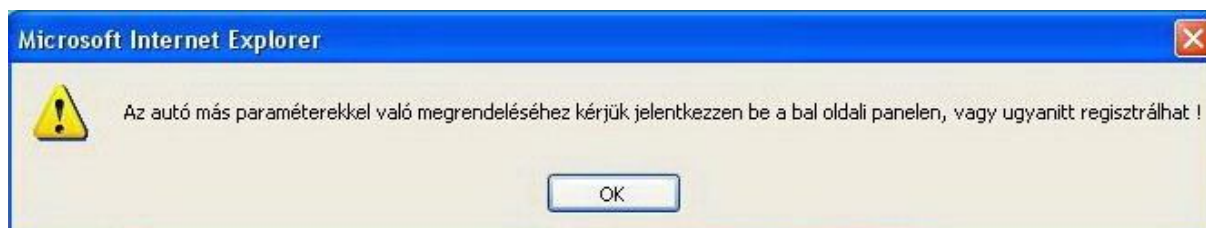


<b>Márka:</b>	Skoda
<b>Típus:</b>	Octavia Scout
<b>Szín:</b>	pezsge
<b>Ülések száma:</b>	5
<b>Önsúly:</b>	1800 kg
<b>Motor teljesítmény:</b>	75 kW
<b>Évjárat:</b>	2008
<b>Üzemanyag:</b>	diesel
<b>Ajtók száma:</b>	5
<b>Felszereltségi kategória:</b>	közép
<b>Hajtott kerekek száma:</b>	4
<b>Végsebesség:</b>	170 km/h
<b>Átlagfogyasztás:</b>	7.2 l/100 km
<b>Injektoros:</b>	igen
<b>Sebességváltó:</b>	manuális
<b>Gyorsulás:</b>	13.2 mp

Ne felejtjük el, hogy nem vagyunk bejelentkezve, eddig ez nem is okozott gondot, mivel az eddigi funkciók nem igényeltek regisztrált felhasználót, viszont ha most rákattintanánk a *Megrendel* gombra akkor az alábbi figyelmeztető üzenetet kapnánk:



Ha a *Testreszab* gombra kattintunk szintén figyelmeztető üzenetet kapunk:



Ha bejelentkezünk a bal oldali panelen, vagy regisztrálunk, és aztán jelentkezünk be, akkor ezek után rákattintva a *Megrendel* gombra leadjuk az adott autó megrendelését az ezen az oldalon olvasható paraméterekkel. Ha így teszünk kapunk egy nyugtázó üzenetet, miszerint megrendelésünket a rendszer feldolgozta.

**Mező és Zavaczkai autóvilág**  
autószalon, autóértékesítő hely, teszttvetési lehetőség, autószervíz

Kezdőlap **Autó**

**Kijelentkezés**  
Üdvözöljük honlapunkon:  
Mező András  
Kijelentkezés  
[Profil](#)

**Naptár**  
April  
V H K Sze Cs P Szo  
1 2 3 4  
5 6 7 8 9 10 11  
12 13 14 15 16 17 18  
19 20 21 22 23 24 25  
26 27 28 29 30

Márka: Skoda  
Típus: Octavia Scout  
Szín: pezsgő  
Ülések száma: 5  
Önsúly: 1800 kg  
Motor teljesítmény: 75 kW  
Évjárat: 2008  
Üzemanyag: diesel  
Ajtók száma: 5  
Felszereltségi kategória: közép  
Hajtott kerekek száma: 4  
Végsebesség: 170 km/h  
Átlagfogyasztás: 7.2 l/100 km  
Injektoros: igen  
Sebességváltó: manuális  
Gyorsulás: 13.2 mp

Microsoft Internet Explorer  
Megrendelését feldolgoztuk. Köszönjük, hogy minket választott!  
OK

Ha pedig a *Testreszab* gombra kattintunk, akkor módosíthatjuk az autó paramétereit a legördülő menük segítségével.

**Mező és Zavaczkai autóvilág**  
autószalon, autóértékesítő hely, teszttvetési lehetőség, autószervíz

Kezdőlap **Autószalon** Autókölcsönző **Testszvetés** Szervíz

**Kijelentkezés**  
Üdvözöljük honlapunkon:  
Mező András  
Kijelentkezés  
[Profil](#)

**Naptár**  
April  
V H K Sze Cs P Szo  
1 2 3 4  
5 6 7 8 9 10 11  
12 13 14 15 16 17 18  
19 20 21 22 23 24 25  
26 27 28 29 30

Márka: Skoda  
Típus: Octavia Scout  
Szín: ezüstsürke  
Ülések száma: 5  
Önsúly: 1800 kg  
Motor teljesítmény: 75 kW  
Évjárat: 2008  
Üzemanyag: diesel  
Ajtók száma: 5  
Felszereltségi kategória: közép  
Hajtott kerekek száma: közép  
Végsebesség: 170 km/h  
Átlagfogyasztás: 7.2 l/100 km  
Injektoros: igen  
Sebességváltó: manuális  
Gyorsulás: 13.2 mp

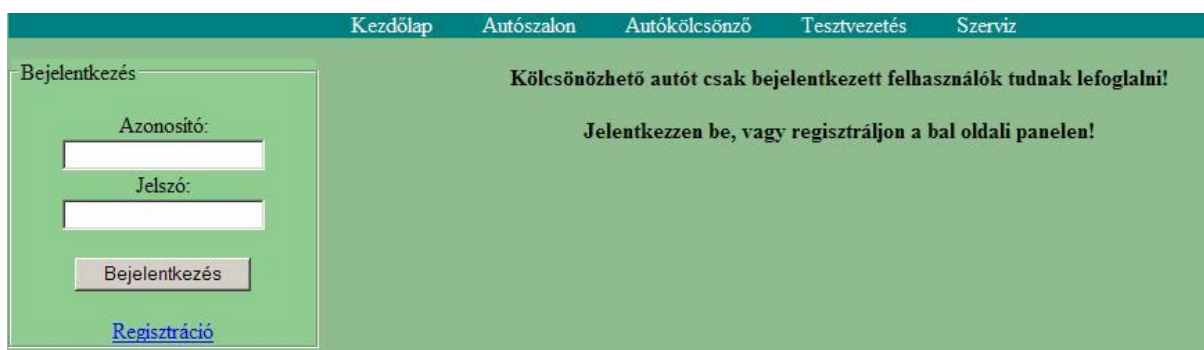
Megrendel

Ahogy látható az autók nem minden paraméterét lehet módosítani. Módosítható az autó színe, a motor teljesítménye, az üzemanyag, a felszereltségi kategória, az injektor és a sebességváltó típusa. Ha elvégeztük a kívánt módosításokat, akkor a *Megrendel* gombra kattintva rendelhetjük meg az autót a kiválasztott paraméterekkel.

## Autókölcsonzó

A navigációs menü következő pontja az autókölcsonzó (kolesonzo.php). Amikor egy ügyfél olyan helyzetbe kerül, amikor is egy autóra lenne szüksége egy rövid időre, akkor ez a funkció hasznos lehet számára. Előre lefoglalhat egy megadott típusú autót egy előre megadott időpontra, és tájékozódhat arról, hogy milyen költségekkel kell számolnia, így amikor belép a szalonba azzal a céllal, hogy elvigye az autót, az biztosan ott fogja várni őt, míg ha ezt nem tenné meg, akkor előfordulhatna az a kellemetlen helyzet, hogy minden kölcsönözhető autót elvittek előle vagy esetleg többbe kerül, mint gondolta.

A kölcsönzési funkció eléréséhez be kell jelentkezünk, ellenkező esetben a *Kölcsönözhető autót csak bejelentkezett felhasználók tudnak lefoglalni!* üzenet fogad minket.



The screenshot shows a web page with a green header containing navigation links: Kezdőlap, Autószalon, Autókölcsonzó, Tesztvezetés, and Szerviz. The main content area has a light green background. On the left, there is a 'Bejelentkezés' (Login) form with two input fields labeled 'Azonosító:' and 'Jelszó:', a 'Bejelentkezés' button, and a blue link for 'Regisztráció'. On the right, there is a message in bold text: 'Kölcsönözhető autót csak bejelentkezett felhasználók tudnak lefoglalni!' followed by 'Jelentkezzen be, vagy regisztráljon a bal oldali panelen!'.

Nincs más dolgunk, mint a bal oldali panelen bejelentkezni, vagy regisztrálni az oldalon. Miután ezt megtettük, a következő kép fogad minket:

Kezdőlap Autószalon Autókölcsönző Tesztvezetés Szerviz

**Folyamatban lévő kölcsönzések:**  
**Jelenleg nem kölcsönöz egy autót sem.**

**Aktuális kölcsönzések:**  
**Nincs egy kölcsönzés se előjegyezve!**

**Szalón:**  
 Békéscsaba, Gyulai út 211

**Időpont:** 2009 04 26

**Kölcsönzés hossza:** 1 nap

**Jelenleg az alábbi autók kölcsönözhetőek:**

Kép	Típus	Utások száma	Ár/nap
	Fiat Panda	4	1247 Ft
	Ford Focus Coupet Cabrio	2	2800 Ft

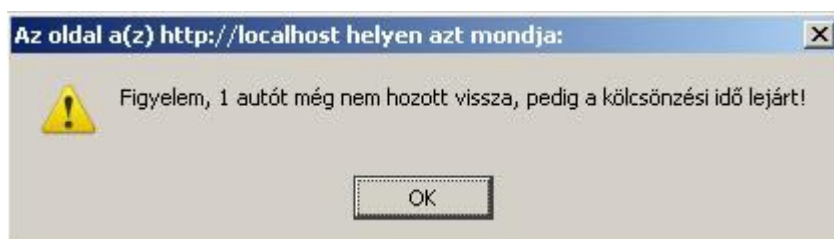
Az oldalt három fő részre lehet osztani. A fenti képen az éppen kölcsönzött autókat mutató táblázat látható, amely kitöltésével egy újabb kölcsönzési szándékot jelezhetünk.

Az első részben az éppen folyamatban lévő kölcsönzéseket láthatjuk, kölcsönzési dátum szerint sorba rendezve.

**Folyamatban lévő kölcsönzések:**

	Időszak	Rendszám	Autó	Összeg (Ft)
	2009-04-20 - 2009-04-25	MEZA-K10	Opel Vivaro Tour	21830
	2009-04-23 - 2009-04-26	MEZA-K4	Fiat Panda	1247

Ha nincs folyamatban kölcsönzés, akkor a *Jelenleg nem kölcsönöz egy autót sem.* feliratot láthatjuk. Ha van, akkor a táblázat felépítése a következő lesz. Az első oszlopban zöld, sárga vagy piros színeket láthatunk, ezzel jelezve, hogy a kölcsönzés milyen állapotban van. Ha zöld, akkor az autót még büntetlenül használhatjuk, még nem járt le a kölcsönzés ideje, ha sárga, akkor a mai nap folyamán vissza kell vinni az autót, és piros, ha már lejárt a kölcsönzés, ekkor az autó visszavitelekor pénzbüntetésre számíthat a felhasználó. Amikor a felhasználó belép az *Kölcsönzés* oldalra, akkor egy felugró ablakot kap, ami a lejáráó és lejárt kölcsönzésekről tájékoztatást ad, mint ahogy az alábbi két kép is mutatja.



A második oszlopban láthatjuk a kölcsönzés kezdetét és végét dátummal megadva. A harmadik oszlopban az autó rendszámát látjuk, míg a negyedikben az autó márkáját és típusát. Az utolsó oszlop a fizetendő összegről tájékoztat minket, melybe az esetleges büntetés nem értendő bele.

A második részben a korábbi foglalásainkat tekinthetjük meg szintén kölcsönzési dátum szerint rendezve. Amennyiben nincs olyan kölcsönzési szándékunk leadva, melynek kezdeti időpontja a jövőben lenne, akkor a következő üzenet fogad itt minket: *Nincs egy kölcsönzés se előjegyezve.* Amennyiben vannak olyan kölcsönzések melyek a jövőben kezdődnek, azokról a következő információkat tudhatjuk meg:

Aktuális kölcsönzések:				
	Időszak	Szalón	Autó	Összeg (Ft)
<input type="checkbox"/>	2009-04-26 - 2009-04-27	Békéscsaba, Gyulai út 211	Fiat Panda	1247
<input type="checkbox"/>	2009-04-26 - 2009-05-01	Békéscsaba, Gyulai út 211	Opel Vivaro Tour	21830

Kijelöltek lemondása

A második oszlopban láthatjuk a kölcsönzés kezdeti és lejárat dátumát. A következő oszlopban azt tudhatjuk meg, mely szalónban vár a lefoglalt autó. A harmadik oszlop a kiválasztott autó típusáról tájékoztat minket, az utolsó oszlopban pedig a megadott kölcsönzési hossz szerint fizetendő összeget láthatjuk.

Amennyiben valamelyik kölcsönzést törölni szeretnénk, akkor az első oszlopban látható üres négyzeteket kipipálva a megfelelő kölcsönzéseknél és utána a *Kijelöltek törlése* gombra kattintva tudjuk ezt megtenni. Ekkor egy felugró ablak tájékoztat minket a törlés sikerességéről.




A harmadik része az oldalnak, maga a kölcsönzés előjegyzése. Itt meg kell adnunk, egy legördülő menüből hogy mely szalonból szeretnénk elhozni az autót. Egy időpontot, hogy mely napon szeretnénk elhozni az autót. Leghamarabb csak a következő naptól lehet kölcsönözni, ellenkező esetben, ha érvénytelen dátumot vagy a holnapnál korábbi dátumot adunk meg, akkor hibaüzenetet kapunk. Amit rendszer így jelöl:

**Időpont:** 2009 ▾ 04 ▾ 26 ▾ **Leghamarabb csak holnapra foglalhat le autót!**

**Kölcsönzés hossza:** 1 ▾ nap

A következő legördülő menünél kiválaszthatjuk, hogy hány napra szeretnénk kölcsönözni. Maximum 1 hónapig, azaz 30 napig lehet kölcsönözni. Ezek után kiválaszthatjuk, hogy mely autót szeretnénk kölcsönözni. Igyekeztünk minden típusú igénynek megfelelni, így a sportos, családi és egyéb típusokból is kölcsönözhetünk. Az autóknak csak a legfontosabb, kölcsönzésnél szerepet játszó tulajdonságai látszanak közvetlen, így az autó márkája és típusa mellett az utasok száma és az autó napi díja olvasható le.

**Jelenleg az alábbi autók kölcsönözhetőek:**

	Kép	Típus	Utások száma	Ár/nap
<input type="radio"/>		Fiat Panda	4	1247 Ft
<input type="radio"/>		Ford Focus Coupet Cabrio	2	2800 Ft
<input type="radio"/>		Opel Vivaro Tour	6	4366 Ft
<input type="radio"/>		Suzuki Jimny	5	3987 Ft

Az űrlap végén található a *Foglalás* gomb. Ha nem választottunk autót, akkor kapunk egy figyelmeztetést, az alábbi szöveggel: *Nem választott ki autót!*

**Jelenleg az alábbi autók kölcsönözhetőek: Nem választott ki autót!**

	Kép	Típus
<input type="radio"/>		Fiat Panda

Amennyiben minden rendben, akkor a gomb lenyomása után a lefoglalások táblázatban megjelenik a kölcsönzés.

### Tesztvezetésre jelentkezés

Az oldal negyedik menüpontja a Tesztvezetés. Itt a felhasználók jelentkezhetnek tesztelni az egyes autókat, hogy véleményt alkothassanak róluk, és esetleg a vásárlás mellett döntsenek. Természetesen a szalon is kíváncsi vendégei véleményére így az autókról egy rövid teszt tölthető ki. Ezen az oldalon mindig látható, hogy a szalon milyen autókat kínál tesztvezetésre,

így látogatása hetente-havonta rendszeresen ajánlott. Tesztvezetésre csak azon felhasználók jelentkezhetnek, akik regisztráltak és be is vannak jelentkezve.

Az oldalt két fő részre oszthatjuk.

**Aktuális tesztvezetések:**  
**Nincs egy tesztvezetésre se előjegyezve!**

**Szalon:**  
 Békéscsaba, Gyulai út 211

**Időpont:** 2009 04 26  
**Óra:** 9

**Jelenleg az alábbi autók tesztelhetők:**

	Kép	Típus	Véleményezés	Pontszám
<input type="radio"/>		Fiat Panda	<a href="#">Véleményezés</a>	30.5/50
<input type="radio"/>		Ford Focus Coupé Cabrio	<a href="#">Véleményezés</a>	Még nincs értékelve

Az egyik része tájékoztatást nyújt, hogy a felhasználó korábban milyen tesztvezetésekre jelentkezett. Amennyiben még nem jelentkezett egyre sem, vagy pedig már túl van a teszteléseken, akkor a *Nincs egy tesztvezetésre se előjegyezve.* felirat látható. Amennyiben vannak tesztvezetések előjegyezve, akkor egy táblázatot láthatunk.

**Aktuális tesztvezetések:**

	Időpont	Szalon	Autó
<input type="checkbox"/>	2009-04-26, 9 óra	Békéscsaba, Gyulai út 211	Fiat Panda
<input type="checkbox"/>	2009-04-26, 11 óra	Békéscsaba, Gyulai út 211	Opel Vivaro Tour
<input type="checkbox"/>	2009-04-26, 12 óra	Békéscsaba, Gyulai út 211	Ford Focus Coupé Cabrio

Kijelöltek lemondása

A négy oszlopában a következő információkat láthatjuk, a tesztvezetésekről. Az második oszlopban láthatjuk, hogy mely időpontra jelentkezünk, napra és órára pontosan. A tesztelés a megadott órától a következő óráig tart, és az adott szalonban csak egy ember tesztelhet ez idő alatt, így ha késik, akkor saját magától veszi el az időt. A harmadik oszlopban láthatjuk, hogy mely szalonba kell majd mennünk a megadott időpontra. Az utolsó oszlopban láthatjuk a kiválasztott autó márkáját és típusát. A táblázat után egy a *Kijelöltek lemondása* gomb van,

melyet megnyomva törli azokat a tesztvezetéseket, amelyeket a táblázat első oszlopában található négyzetek bejelölésével megjelöltünk. Ekkor az alábbi üzenetet kapjuk:



Az oldal második felében jelentkezhetünk újabb tesztvezetésre.

Ehhez csak annyit kell tennünk, hogy kiválasztjuk az első legördülő menüből, hogy mely szalonban szeretnénk tesztelni. A következő pontnál kiválaszthatjuk, hogy mely napon szeretnénk tesztelni. Ez után megadhatjuk, hogy hány órától. A szalonokban csak reggel 9 és délután 16 óra között lehet tesztelni. Ezek után következnek az autók, ahol az első oszlopban kell bejelölnünk azt az autót, amelyet tesztelni akarunk. Minden autóhoz tartozik a sorok végén egy *Véleményezés* link.

Miután rákattintottunk a *Jelentkezés* gombra, a rendszer ellenőrzi, hogy minden rendben van-e a jelentkezéssel kapcsolatban. Ha nem választott ki a felhasználó autót, akkor, az Autókölcsonzéshez hasonlóan, egy hibaüzenet jelenik meg a következő üzenettel: *Nem választott ki autót!*

Leghamarabb csak másnapra jelentkezhet tesztvezetésre a felhasználó, ezért ha korábbi dátumot ad meg akkor is egy hibaüzenet jelenik meg, melynek szövege: *Legfeljebb csak holnapra jelentkezhet tesztvezetésre!*

Szintén dátummal kapcsolatos az, hogy ha a felhasználó nem valós dátumot ad meg. Ekkor egy üzenet jelenik meg, melyben tájékoztatja a rendszer a felhasználót, hogy az adott hónap rövidebb, mint amit ő megadott.

Ha az adott szalonban, az adott napon az adott órában már van tesztvezető, akkor kapjuk a következő hibaüzenetet.

**Sajnáljuk, a(z) Békéscsaba, Gyulai út 211 szalonban,  
2009.04.26 -n, 9 órakor több jelentkezőt már nem tudunk fogadni.  
Kérjük próbáljon egy másik időpontot, vagy válasszon másik szalont!  
A következő órák még szabadok ezen a napon: 10, 11, 13, 14, 15 óra**

Mint látható, a fenti képen, a rendszer tájékoztatja a felhasználót arról, hogy foglalt az időpont, és ha mindenképpen aznap szeretne menni, akkor kiírja az olyan órákat, amikor aznap még lehet tesztelni autót.

Amennyiben az eddigiek rendben vannak, akkor a jelentkezés tárolásra kerül, és megjelenik az aktuális jelentkezések között.



A *Véleményezés*re kattintva formálhatunk véleményt az adott autóról. Ezt egy rövid, néhány kérdést tartalmazó teszt kitöltésével tehetjük meg, ahol különböző szempontok szerint kell értékelnünk az autót, egy 1-5-ig terjedő skálán. A skálán az 1-es jelenti azt, hogy az adott tulajdonság egyáltalán nem tetszett, míg az 5-ös a nagyon tetszettet jelenti. Miután minden kérdést pontoztunk, akkor a végén található *Elküld* gombbal elküldhetjük véleményünk.

Az eddigi felhasználói vélemények alapján, az utolsó oszlopban látható, hogy az adott autó hány pontot ért el.

## Autószerviz

Az utolsó menüpontban a szervizben lévő autónkról kaphatunk rövid szöveges üzeneteket arról, hogy éppen hogyan állnak az autó szerelésével a szerelők. Ez a funkció szintén elérhetetlen regisztráció és bejelentkezés nélkül.

Ez a következő képen működik. Amikor nincs a felhasználónak autó a szervizben, akkor a következő oldal fogadja:



Amikor egy autót bevisz az ügyfél a szervizbe, a szerelő a helyszínen felviszi a rendszerbe az autó adatait, majd egy véletlenszerűen generált kódot ad az ügyfélnek. Ezzel a kóddal tudja majd az ügyfél a szerviz oldalon jelezni, hogy neki egy autója a szervizben van, és ezek után tudja nyomon követni annak állapotát. Ezt a kódot kell a képen látható mezőbe beírni. Miután ez megtörtént, a felhasználó automatikusan a kódhoz tartozó autó oldalára kerül. Szintén erre

az oldalra kerül, ha már korábban beírta a szervizben lévő autójának a kódját, és rányom a Szerviz menüpontra.

**Rendszám:** III-444  
**Beadás ideje:** 2009-03-20 15:41:15  
**Állapot:** Folyamatban

**Üzenet:**

Üzenet szövege	<b>Szerelő János</b>	<b>2009-04-25 17:28:44</b>
	Kicseréltük a dugattyút, ára 2000 Ft	
	<b>Szerelő János</b>	<b>2009-04-25 17:28:16</b>
	Az egyik dugattyú eltört, csere szükséges	
	<b>Szerelő János</b>	<b>2009-04-25 17:27:48</b>
	Az autó vizsgálatát elkezdtük.	
Üzenet küldése	<b>Vendég Látó</b>	<b>2009-04-25 17:27:11</b>
Kotyogást hallottam a motorból.		

Ahogy látható a képen, le van írva hogy mi az autó rendszáma, hogy mikor lett leadva az autó a szervizben, és egy állapot, mely a következő értékeket veheti fel. *Beadva*, ez esetben az autót átvették, de még nem kezdtek el foglalkozni vele, egyéb teendők miatt. A következő állapot a *Folyamatban*, ez jelenti azt, hogy már dolgoznak az autóval, ez idő alatt várhatóak a jelentések a szerelőktől. Ha elkészültek az autóval, akkor az állapotnál a *Kész* üzenet jelenik meg. Ez azt jelenti, hogy minden problémát megoldottak az autónál, és várják az ügyfelet, hogy átvegye a szervizben. Miután elhozták az autót a szervizből, a *Kiadva* szó jelenik meg az autó állapotánál, de ezt csak akkor látható, ha rákattintunk az *Az utolsó szervizelt autó adatainak a megtekintése* linkre.

Az oldal nagyobbik részén időrendbe szedett jelentéseket láthatunk a szerelőktől. Egy ilyen jelentés három dolgot jelez, mikor írták az üzenetet, hogy hívják a szerelőt, aki írta és végül magát az jelentés szövegét arról, hogy éppen mit csináltak a szerelők, esetleges további problémák felfedezéséről, vagy éppen a várható költségekről.

Ezekre a jelentésekre, a felhasználónak lehetősége van reagálni, a jelentések melletti szöveges mezőben illetve üzeni a szerelőknek, ha valamit szeretne kérni tőlük. Miután megírta az üzenetet, az *Üzenet küldése* gombra kattintva, az megjelenik a jelentések mellett, így a felhasználó és a szerelő egyaránt láthatja. A gombra kattintás után az alábbi ablak jelenik meg.



## B. Adminisztrátori oldalról

### Autóválaszték módosítása

#### a) Autó paramétereinek módosítása

Az adminisztrátori jogosultságokkal rendelkező személy belépve a rendszerbe az *Autószalon* menüpont alatt az autókészletet láthatja táblázatos formában az alábbi módon:



Márka	Típus	Szín	Ár	Ülések	Önsúly	Teljesítmény	Évjárat	Üzemanyag	Ajtók	Felszereltség	Hajtott kerék	Végsebesség	Átlagfogyasztás	Injektor	Váltó	Gyorsulás		
Skoda	Fabia Sedan	ezüstszürke	3003400	5 db	1085 kg	56 kW	2008	diesel	5	közép	2	160 km/h	6.1 liter/100 km	nem	automata	16 mp	Módosít	Töröl
Skoda	Octavia	pasztelezöld	4166200	4 db	1210 kg	59 kW	2009	benzin	5	alap	2	173 km/h	7 liter/100 km	nem	manuális	14.2 mp	Módosít	Töröl
Skoda	Octavia Scout	pezsgő	7200120	5 db	1800 kg	75 kW	2008	diesel	5	közép	4	170 km/h	7.2 liter/100 km	igen	manuális	13.2 mp	Módosít	Töröl
Skoda	Octavia Tour	metálszürke	3843400	4 db	1175 kg	55 kW	2008	benzin	5	alap	2	171 km/h	6.8 liter/100 km	nem	manuális	15.3 mp	Módosít	Töröl
Skoda	Praktik	fehér	2157000	2 db	1120 kg	51 kW	2007	benzin	3	alap	2	158 km/h	6.8 liter/100 km	nem	manuális	15.9 mp	Módosít	Töröl
Skoda	Roomster	kiralykek	3399400	4 db	1150 kg	51 kW	2008	benzin	5	alap	2	158 km/h	6.8 liter/100 km	nem	manuális	15.9 mp	Módosít	Töröl
Skoda	Roomster Scout	fehér-fekete	4417000	4 db	1155 kg	63 kW	2008	diesel	5	alap	2	171 km/h	6.8 liter/100 km	nem	manuális	13 mp	Módosít	Töröl
Skoda	Superb	szürke	6296200	4 db	1425 kg	92 kW	2009	diesel	5	közép	2	201 km/h	7.6 liter/100 km	igen	automata	10.5 mp	Módosít	Töröl

Minden admin oldal tetején található a *Kijelentkezés* gomb, aminek a segítségével az adminisztrátori jogosultságokkal rendelkező személy kiléphet az alkalmazás admin részéből, ekkor a rendszer átirányítja a *Főoldalra*.

Látható, hogy itt is lehetőség nyílik szűrések elvégzésére, illetve minden autó sora mellett látható egy *Módosít* és *Töröl* gomb. A *Módosít* gomb használatával az adott autó teljesítmény, üzemanyag, felszereltségi kategória valamint a sebességváltó paramétereit lehet módosítani, ha változás történt a készleten lévő autó paramétereiben.

A *Módosít* gombra kattintva az alábbi oldalon tehetjük ezt meg.



A legördülő menükből a megfelelő értékek kiválasztása után a *Mentés* gombra kattintva elvégezzük a módosítást adatbázis szinten is.

## b) Autó törlése az adatbázisból

Ha az adott autó sora mellett a *Töröl* gombra kattintunk, akkor az alábbi megerősítéssel találkozunk.



Ha az *OK* gombbal megerősítjük törlési szándékunkat, akkor a következő nyugtázó üzenetet kapjuk.



Ha a *Mégse* gombra kattintunk, és ezzel visszavonjuk törlési szándékunkat az alábbi nyugtázó üzenetet kapjuk.



### c) Új autó hozzáadása az adatbázishoz

Az *Új autó hozzáadása* menüpont alatt lehetőség van új autó felvételére az adatbázisban, ha erre a menüpontra kattintunk, akkor egy form kitöltése és az autó képének feltöltése után az új autó rögzítésre kerül az adatbázisban. A kép ajánlott formátuma a PNG és ajánlott mérete 200x100 pixel. A kép akkor kerül feltöltésre, ha az adott autótípushoz tartozó könyvtárban még nem létezik ilyen nevű fájl, tehát ha ugyanolyan autótípusból szeretnénk több modellt tárolni, akkor figyeljünk oda a képfájlok névütközésének elkerülésére.

Ha a formot hiányosan, vagy egyáltalán nem töltjük ki, és az *Autó felvétele* gombra kattintunk, az alábbi hibaüzenetet kapjuk:



A form minden mezőjét kitöltve sikeres autó felvétele esetén az alábbi képernyőt látjuk.

Kezdőlap   Autószalon   Megrendelések   Autókölcsönző   Tesztvezetés

## Új autó adatai

Márka:

Típus:

Szín:

Ár:  HUF

Ülések száma:

Önsúly:  kg

Motor teljesítmény:  kW

Évjárat:

Üzemanyag:

Ajtók száma:

Felszereltségi kategória:

Hajtott kerekek száma:

Végsebesség:  km/h

Átlagfogyasztás:  liter/100km

Injektoros:

Sebességváltó:

Kép feltöltése:   mérete: 200x100 px

Gyorsulás:  mp

Ha a fent megadott autó nem létezik ezen paraméterekkel az adatbázisban, akkor az alábbi nyugtázó üzenetet kapjuk:



Különben pedig, ha az autó már létezik az adatbázisban ezen paraméterekkel, akkor az alábbi üzenetet kapjuk:



## d) Megrendelések nyomon követése és törlése

A *Megrendelések* menüpont alatt láthatjuk a rendszer által elfogadott megrendelések adatait táblázatos formában, amely tartalmazza a megrendelő nevét, a megrendelt autó fontosabb paramétereit illetve a rendelés időpontját az alábbi módon:



Megrendelő	Márka	Típus	Szín	kW	Üzemanyag	Kategória	Injektor	Váltó	Rendelés ideje	
Mező András	Skoda	Fabia Sedan	ezüstszürke	46	diesel	alap	nem	manualis	2009-03-17 12:39:05	Töröl
Nagy István	Opel	Agila	fehér	68	diesel	közép	nem	manualis	2009-03-17 12:41:09	Töröl
Nagy István	Skoda	Octavia Scout	ezüstszürke	75	diesel	közép	igen	manualis	2009-03-20 15:26:16	Töröl
Mező András	Opel	GT	fekete	214	diesel	közép	igen	automata	2009-03-20 17:00:17	Töröl
Mező András	Skoda	Fabia Sedan	metálszürke	46	diesel	alap	nem	manualis	2009-03-20 17:22:09	Töröl

Minden megrendelés sora mellett található egy *Töröl* gomb, amely segítségével az adminisztrátori jogosultságokkal rendelkező személy törölheti az adott megrendelést az adatbázisból, ha valamilyen okból a megrendelés már nem aktuális. Például a megrendelést követően a megrendelő a helyszínen átvette az autóját, vagy a helyszínen mégis úgy döntött, hogy nem kívánja megvásárolni az autó és visszalép az adás-vételtől.

A *Töröl* gombra kattintva az alábbi megerősítést kérő üzenet jelenik meg:



Az *OK* gombra kattintva a megrendelés törlésre kerül az adatbázisból, amelyet az alábbi nyugtázó üzenet jelez:



A *Mégse* gombra kattintva visszavonjuk a megrendelésre vonatkozó törlési szándékunkat, amelyet az alábbi üzenet is megerősít:



## **Autókölcsonzések és Tesztvezetések kezelése**

Az autókölcsonzó és a tesztvezetői adminisztrációs oldal néhány kivételtől eltekintve hasonlít egymásra, így ezeket egyszerre mutatom be.

Az autókölcsonzó adminisztrátor kezeli a kölcsönözhető autók listáját, azt hogy az autókat elvitték és visszahozták e. Valamint látható még a fizetendő összeg és hogy kell e pénzbüntetést kiszabni.

A tesztvezető adminisztrátor feladata, hogy naprakészen tartsa azoknak az autóknak a listáját, melyeket a felhasználók tesztelhetnek, valamint az ő dolga lesz majd fogadni azokat a személyeket, akik jelentkeztek vezetni és végül a véleményekről is megtekinthet egy rövid statisztikát, ezzel segítve azon autók kiválasztását, melyeket a szalonban érdemes tartani.

Mind a két oldal csak akkor érhető el, ha adminisztrátori joga van a felhasználónak. Az adminisztrátori felület, a vendég felhasználók oldalától eltérően néhány dologban egyszerűsítve lett, a könnyebb átláthatóság miatt. Így az oldal bannerje eltűnt, helyette megjelent egy *Kijelentkezés* gomb, mellyel ki tud jelentkezni, valamint egy egyszerű formátumú dátum. Ezekkel lett kiváltva a bejelentkező és regisztrációs panel, és a naptár. A főmenüből eltűnt a szerviz oldal elérhetősége, mivel ez a fajta adminisztrátori jog arra az oldalra nem érvényes.

Az oldalra érkezés után egy újabb menüsor jelenik meg, melyekkel a tovább lehet navigálni. Mind a két funkció esetén arra az oldalra kerülünk, ahol ki vannak listázva a kölcsönzések illetve tesztvezetések. Ez egy táblázattal történik, melyben dátum szerint vannak felsorolva a jelentkezések, és csak a legfontosabb információk olvashatók le.

Időpont	Szalon	Autó típusa	Jelentkező neve
2009-04-26, 9 óra	Békéscsaba, Gyulai út 211	Fiat, Panda	Mező András
2009-04-26, 12 óra	Békéscsaba, Gyulai út 211	Ford, Focus Coupet Cabrio	Mező András
2009-04-26, 15 óra	Békéscsaba, Gyulai út 211	Opel, Vivaro Tour	Mező András
2009-05-06, 9 óra	Békéscsaba, Gyulai út 211	Opel, Vivaro Tour	Felhasználó Béla

Ez a kép a tesztvezetői adminisztrációs oldalról készült, de az autókölcsönzői is hasonló, viszont abban van néhány plusz funkció.

Mint látható a táblázat első oszlopában van megjelenítve az Időpont, amikor is a felhasználó a tesztvezetésre jelentkezett. A tesztvezetésnél egy óra is meg van adva, ebben az órában van lehetőség tesztelni az autót a kiválasztott szalonban.

Az autókölcsönzői táblázatnál az időpontnak két dátum van megadva, óra nélkül. Az első dátum azt a napot jelöli, amikor a felhasználó a szalonba betérve kérheti, hogy elvihesse a lefoglalt autót. A második dátum azt a napot jelöli mikor a kölcsönzése lejár. Ezen a napon legkésőbb délután 6 óráig lehet visszavinni az autót. Amennyiben később viszi vissza, azért felárat számolnak fel.

A következő oszlopban annak a szalonnak a nevét láthatjuk, amit a felhasználó megjelölt, hogy ott fog tesztelni, vagy onnan viszi el a kölcsönözni kívánt autót.

A harmadik oszlopban látható, hogy mely autót választotta ki a felhasználó.

Az utolsó oszlopban pedig a felhasználó neve látható.

Az autókölcsönzői táblánál a következő plusz funkciók jelennek meg.

Állapot	Időszak	Szalon	Autó típusa	Rendszám	Kölcsönző neve	Fizetendő
Kiad	2009-04-26-től 2009-04-27-ig	Békéscsaba, Gyulai út 211	Opel, Vivaro Tour	MEZA-K Hozzáad	Mező András	4366
Kiad	2009-04-26-től 2009-04-27-ig	Békéscsaba, Gyulai út 211	Fiat, Panda	MEZA-K4	Mező András	1247

Kiegészül a táblázat egy olyan oszloppal, melyben egy gomb található, amivel az autók kiadását, illetve visszahozását lehet jelezni. Amennyiben egy kölcsönzés ideje lejárt és utána kattintunk a *Bead* gombra, akkor ez a kölcsönzés eltűnik a listáról. Ha viszont még tart a kölcsönzési idő, akkor az autót még ki lehet kölcsönözni, ekkor a *Kiad* gombra kattintva lehet ezt jelezni. A két gomb egymást felváltva jelenik meg, azaz csak olyan autót lehet kiadni, ami még nincs kiadva, avagy csak olyat lehet visszahozni, ami még nincs beadva.

Másik speciális oszlop a *Rendszám* oszlop. Amikor egy autót kikölcsönöznek, egy speciális rendszámot kap, ezt csak kölcsönzéskor teszik fel az autóra, és viszik fel a rendszerbe. Amikor még nincs rendszám megadva egy kölcsönzéshez, akkor egy üres mezőt és egy gombot láthatunk. Mivel minden a szalon által kölcsönzésre kiadott autóra feltett rendszám MEZA-K kezdetű így ez be van már írva, így csak egy számot kell megadni, majd pedig a *Hozzáad* gombra kell kattintani.



A tesztvezetési és kölcsönzői adminisztrációs oldal másik része, az autók kezelését teszi lehetővé. Itt lehet megadni, hogy mely autókat szeretnénk, ha a felhasználók tudnák tesztelni, vagy ki tudnák kölcsönözni.

The screenshot shows a web interface for managing cars. At the top, there are two dropdown menus, both set to 'Mind', and a 'Szűrés' button. Below this is a table with columns for 'Kölcsönözhető' (checkbox), 'Márka' (Brand), and 'Típus' (Type). The table lists several car models, with checkboxes indicating their status. At the bottom, there are buttons for 'Előző 10', 'Következő 10', and 'Mentés'.

Kölcsönözhető	Márka	Típus
<input checked="" type="checkbox"/>	Fiat	Panda
<input type="checkbox"/>	Fiat	Panda 100 HP
<input type="checkbox"/>	Fiat	Panda 4x4
<input type="checkbox"/>	Fiat	Punto Classic
<input type="checkbox"/>	Fiat	Punto Classic
<input type="checkbox"/>	Fiat	Stilo MW Sedan
<input type="checkbox"/>	Ford	C-Max
<input type="checkbox"/>	Ford	Fiesta
<input checked="" type="checkbox"/>	Ford	Focus Coupet Cabrio
<input type="checkbox"/>	Ford	Focus Frente

Az autókat 10 darabonként láthatjuk kilistázva, melyet egy *Következő 10* és *Előző 10* gombok segítségével tudunk léptetni. Az autók előtt van egy üres vagy bejelölt négyzet, mely azt mutatja meg, hogy az adott autó tesztelhető-e, vagy kölcsönzések esetén kölcsönözhető-e. Ha bejelöltük a megfelelő autókat, akkor a *Ment* gombbal menthetjük őket, de ha a *Következő 10* vagy *Előző 10* gombra kattintunk, akkor is mentésre kerül.

Két szempont szerint lehet csökkenteni a megjelenítendő autók számát. Az első csak azokat az autókat listázza ki, melyek tesztelhetőek vagy nem, illetve kölcsönzés esetén kölcsönözhetőek vagy sem, vagy a *Mind* opciót választva a mind a két feltételnek eleget tevő autókat listázza.

The screenshot shows the same interface as above, but with the first dropdown menu open. The menu options are 'Mind', 'Kölcsönözhetőek', and 'Nem kölcsönözhetőek'. The 'Típus' column is visible, showing 'Panda' and 'Panda 100 HP'.

A második szempontban az autókat a márkájuk szerint listázhatjuk ki, ekkor az adott márka összes típusa jelenik meg. A *Mind*-t választva minden márkájú autó látható lesz.

Egyszerre akár mind a két szempont szerint is lehet szűrni. Miután kiválasztottuk a megfelelő feltételeket a Szűrés gombra kattintva láthatjuk a feltételnek eleget tevő autókat.

Csak a tesztvezetési adminisztrátornak van még egy harmadik része, a *Statisztika* menü, melyben a tesztvezetők véleményét listázhathja ki az egyes autókról.

Kérdések:

1. Milyen az úttartása?
2. Mennyire vezethető könnyen?
3. Milyen a kilátás?
4. Milyen az autó hangereje?
5. Milyen az autó dizájnya?
6. Mennyire kényelmes?
7. Milyen a felszereltsége?
8. Milyen az autó gyorsulása?
9. Mennyire reális az autó ára?
10. Az autó összességében?

Autó	Összátlag pont	1. kérdés	2. kérdés	3. kérdés	4. kérdés	5. kérdés	6. kérdés	7. kérdés	8. kérdés	9. kérdés	10. kérdés
Fiat, Panda	29.5/50	2.0/5	2.0/5	3.0/5	4.0/5	3.0/5	2.5/5	3.5/5	2.5/5	3.5/5	3.5/5
Ford, Focus Coupet Cabrio	31.0/50	3.0/5	2.0/5	4.0/5	3.0/5	2.0/5	5.0/5	1.0/5	3.0/5	4.0/5	4.0/5

Az oldal elején olvasható a 10 kérdés. Minden autónál látható hogy az egyes kérdésekben átlagosan hány pontot adtak a felhasználók, illetve egy összesített átlagpontoszámot, melyből látható az egyes autók mennyire tetszenek a felhasználóknak.

## Autószerviz

A szerviz adminisztrátora speciális, mivel neki nincs joga, hogy a szalon, autókölcsönző és a tesztvezetésekkel kezelje. Számára csak a szervizben lévő autók kezeléséhez illetve jelentések írásához van joga.

Szerelőként bejelentkezve, a menüsor két elemre csökken. A *Kezdőlapra* és a *Szervizre*, mivel a többi menühöz nincs joga.

Kezdőlap    Szerviz

Állapot	Szervizszám	Rendszám	Tulajdonos	Beadás ideje
Beadva	<a href="#">5</a>	<a href="#">III-444</a>	Nincs hozzárendelve	2009-03-20 15:41:15
Beadva	<a href="#">8</a>	<a href="#">1314-adfa</a>	Nincs hozzárendelve	2009-04-24 17:50:14
Kész	<a href="#">1</a>	<a href="#">aaa-111</a>	felh	2009-02-24 18:16:23

Szervizszám:

9

(Ne felejtse el, megadni az autó tulajdonosának!)

Rendszám:

A szerviz lap első felében egy táblázat látható, melyben az éppen szervizelt autókat láthatjuk. Első oszlopában az autók állapotát láthatjuk. A következő oszlop egy azonosító szám, mellyel minden szervizelt autót meg lehet különböztetni a többitől. A harmadik oszlopban az autó rendszáma látható. Ez utóbbi két számra kattintva juthatunk el, a jelentés író oldalra. A következő oszlopban az autó tulajdonosának a neve szerepel, amennyiben az megadta a kapott kódot, és hozzárendelte a felhasználói fiókjához az autót. Ha nem tette meg ezt, akkor a *Nincs hozzárendelve* feliratot olvashatjuk. Az utolsó oszlopban láthatjuk, hogy az autót mikor adták be a szervizbe.

A lap második felében található az új autó hozzáadása funkció. Amikor egy autót leadnak a szervizben, akkor az átvevő szerelő feladata, hogy e funkció használatával azt felvegye az adatbázisba. Ehhez csak annyit kell tenni, hogy a Rendszám mezőbe beírja az autó rendszámát, az ügyfélnek megadja a Szervizszám mezőben szereplő kódot, és rányom a Hozzáadás gombra. Ekkor az autó tárolásra kerül, és megjelenik a fentebbi táblázatban, ahol állapota *Beadva* lesz, a *Tulajdonos* még nem lesz megadva, és beadási ideje az aktuális idő lesz.

Miután kiválasztotta a szerelő, hogy melyik szervizelt autóhoz szeretne jelentést írni, a következő oldalra kerül.

**Rendszám:** III-444  
**Beadás ideje:** 2009-03-20 15:41:15  
**Állapot:** Folyamatban

**Jelentés:**

Jelentés szövege

Állapot:

Beadva  
 Folyamatban  
 Kész  
 Kiadva

Jelentés küldése

Korábbi jelentések	
<b>Szerelő János</b>	<b>2009-04-25 17:28:44</b>
Kicseréltük a dugattyút, ára 2000 Ft	
<b>Szerelő János</b>	<b>2009-04-25 17:28:16</b>
Az egyik dugattyú eltört, csere szükséges	
<b>Szerelő János</b>	<b>2009-04-25 17:27:48</b>
Az autó vizsgálatát elkezdtük.	
<b>Vendég Látó</b>	<b>2009-04-25 17:27:11</b>
Kotyogást hallottam a motorból.	

Az oldalon látható, az autó rendszáma, beadási ideje, és állapota, amik megegyeznek az előbbi táblázatban látottakkal.

Egy nagy szövegmező is van az oldal közepe táján, ahova a szerelő írhatja a jelentés szövegét. Utána pedig az állapotokat, melyek közül egy mindig aktív lesz. A szerelő az állapotokat szabadon állíthatja, de csak előre fele haladhat.

Az állapotok sorrendje a következő:

- *Beadva*, ekkor csak bevitték a szervizbe az autót.
- *Folyamatban*, ekkor már elkezdtek az autóval foglalkozni. A jelentések zöme ebben az állapotban születik.
- Ha elkészültek az autóval, akkor a *Kész* állapot válik aktívvá.
- Végül, ha a felhasználó érte megy az autójáért, akkor a *Kiadva* állapotba kerül. Innentől a szerelők nem írhatnak jelentéseket.

Miután megírta a szerelő a jelentést, kiválasztotta a megfelelő állapotot, amennyiben nem változott akkor nem kell kiválasztani, akkor rákattinthat *Jelentés Küldése* gombra.

A szerviz oldal jobb oldalán láthatóak a korábbi jelentések. Minden jelentésnél látszik, hogy ki írta, mikor írta, és a jelentés szövege. De itt láthatóak a felhasználók által írt üzenetek is.

## C. Fejlesztői oldalról

### Az adatbázis

#### a) Alapok

A program során egyetlen adatbázist használtunk. A programot alapvetően négy részre lehet bontani:

- Autószalon
- Autókölcsonzó
- Tesztvezetés
- Autószerviz

Ezen négy részegység megvalósításához összesen kilenc adatbázistáblát használtunk. A táblák szerkezeti felépítése egyszerűnek mondható. A kilenc adatbázistábla név szerint az alábbi:

- felhasználó
- autók
- autolehet
- kölcsönzések
- megrendelés
- szerviz
- szervizjelentés
- tesztvezetések
- vélemény

#### b) Táblák létrehozása

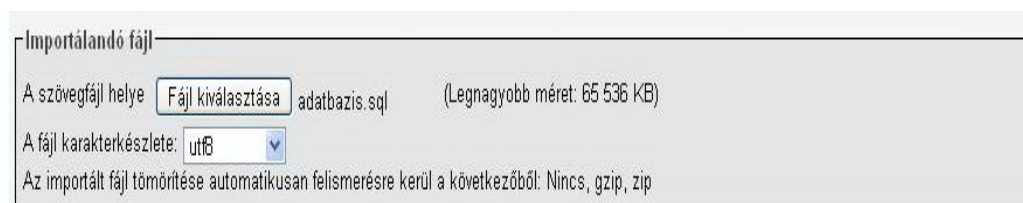
##### 1. Otthoni számítógépen (internet-kapcsolat nélkül)

Mivel mi is otthoni körülmények között végeztük a fejlesztést, így ezzel a megközelítéssel kezdjük. Ha a böngészőnkbe beírjuk a <http://localhost/phpmyadmin> címet akkor máris a PHPMYADMIN webes felületén találjuk magunkat, ahol nagyon egyszerűen megtehető az adatbázis illetve a táblák létrehozása. Az *új adatbázis létrehozása* résznél beírjuk az adatbázisunk nevét és a *Létrehozás* gombra kattintva létre is jön az adatbázisunk.



Ekkor van egy üres adatbázisunk, amelyben természetesen táblákat szeretnénk létrehozni. Ezt megtehetjük egyesével, illetve a táblákat is feltölthetjük soronként, de ez nagyon időigényes megoldás, ezért ennek elkerülése végett mi már előre elkészítettünk egy *adatbázis.sql* nevű SQL szkriptfájlt, amely létrehozza a táblákat és feltölti azokat a megfelelő adatokkal.

Az *Importálás* menüpont alatt megadva ezt az SQL fájlt, majd az *Indítás* gombra kattintva lefut ez az SQL szkript és készen is áll az adatbázis a használatra.



## 2. Tábla létrehozása a WEB-en

Ahogy már korábban említettük az ULTRAWEB-et fogjuk használni alkalmazásunk publikálására a WEB-en. Az ULTRAWEB.HU oldalon belépve felhasználói nevünkkel illetve jelszavunkkal a regisztrációkor létrehozott adatbázisunkat a fenti módon tölthetjük fel, hiszen itt is lehetőségünk van a PHPMYADMIN használatára. Tehát a fenti SQL szkript lefuttatásával szintén létrejön az adatbázisunk, ám ezúttal már a WEB-en.

Viszont apró változtatást eszközölnünk kell a szerverhez való kapcsolódás során, hiszen ez másképpen történik, mint otthoni környezetben. Ha a programunkban a szerverhez való csatlakozást kiemeljük egy külön PHP fájlba és a szükséges helyeken az alkalmazásunkban csak egy *include('csatlakozas.php')* sorral beillesztjük ezt a kódrészletet, akkor újrafelhasználhatóvá tettük ezt a kódrészletet. Ha ezt a programozási technikát használjuk, akkor az említett módosítást elég egy helyen megtenni az alkalmazásunkban.

Az említett kódrészlet otthoni környezetben a következő:

```
<? $sql = mysql_connect("localhost", "root"); ?>
```

Míg Web-szerverünk esetében az alábbi:

```
<? $sql = mysql_connect("sql2.ultraweb.hu", "meza", "*"); ?>
```

MySQL host: **SQL2.ULTRAWEB.HU**

Felhasználó név: **meza**

Jelszó: \*

## c) A táblák szerkezeti felépítése

### 1. A „felhasznalo” tábla

A „felhasznalo” tábla, ahogy a nevéből is kiderül a regisztrált felhasználókat és az ők adatait tárolja. A korábban említett háromféle felhasználók mindegyike ebben a táblában van nyilvántartva, konkrétan a „mezei” felhasználók, adminisztrátorok és szerelők.

A tábla mezői az alábbiak:

- *azonosito*: Ez a tábla elsődleges kulcsa, minden felhasználó rendelkezik vele és minden felhasználó esetében egyedi, egész és előjel nélküli értékről van szó. Értéke automatikusan növekszik, ezért új sor beszúrása esetén nem kell hozzá értéket rendelni. Típusa INTEGER, csak sorszámozott egész értéket kaphat, és értéke sohasem lehet NULL.
- *teljes\_nev*: Az adott felhasználó teljes neve. Típusa VARCHAR(50), ami maximum 100 karakter hosszú sztring lehet. Értéke nem lehet NULL.
- *felhasznalo\_nev*: Az adott felhasználó felhasználói neve (nickname), amelyet a bejelentkezéskor kell megadnia. Típusa szintén VARCHAR(30). Értéke nem lehet NULL.
- *jelszo*: Az adott felhasználó bejelentkezési jelszava. Típusa VARCHAR(50), tehát maximum 50 karakter hosszú sztring lehet. Értéke nem lehet NULL.

- *iranyitoszam*: Az adott felhasználó lakóhelyének irányítószáma. Típusa INTEGER(4), tehát maximum 4 számjegyű sorszámozott egész szám lehet. Értéke szintén nem lehet NULL.
- *varos*: A felhasználó állandó lakcímében szereplő városnév. Típusa VARCHAR(100). Értéke nem lehet NULL.
- *kozterulet*: A felhasználó állandó lakcímében szereplő közterület neve. Típusa VARCHAR(200), tehát maximum 200 karakter hosszú sztring.
- *email*: A felhasználó e-mail címe, amely az e-mail cím megfelelő szintaktikájának megfelel. Típusa VARCHAR(150). Értéke nem lehet NULL.
- *jog*: A felhasználó jogosultságának értékét tartalmazza. Típusa INTEGER(1), tehát maximum 1 számjegyű sorszámozott egész. Értéke háromféle lehet, attól függően, hogy milyen jogosultságokkal rendelkező felhasználóról van szó. Az alábbi három értéke lehet:
  - 0 – adminisztrátor
  - 1 – szerelő
  - 2 – általános felhasználó

A 2-es érték az alapértelmezett, tehát ha új felhasználó felvétele esetén nem adjuk meg a *jog* mező értékét az alapértelmezettként a 2-es értéket fogja kapni.

## 2. Az „autok” tábla

Az „autok” tábla, ahogy a nevéből is kiderül az autókereskedésben lévő autók adatait tartalmazzák. A tábla mezői az alábbiak:

- *azonosito*: Ez a tábla elsődleges kulcsa, minden autó rendelkezik vele és minden autó esetében egyedi, egész és előjel nélküli értékről van szó. Értéke automatikusan növekszik, ezért új sor beszúrása esetén nem kell hozzá értéket rendelni. Típusa INTEGER, csak sorszámozott egész értéket kaphat, és értéke sohasem lehet NULL.
- *marka*: Az autó márkaneve. Típusa VARCHAR(100), tehát maximum 100 karakter hosszúságú sztring lehet. Értéke nem lehet NULL.
- *tipus*: Az autó típusa. Típusa szintén VARCHAR(100). Értéke nem lehet NULL.

- *szin*: Az autó színe. Típusa szintén VARCHAR(100). Értéke nem lehet NULL.
- *ar*: Az autó ára forintban kifejezve. Típusa INTEGER(20), tehát maximum 20 számjegyű sorszámozott egész lehet. Értéke nem lehet NULL.
- *szemely\_szam*: Az autóban szállítható személyek száma. Típusa INTEGER(1), tehát maximum 1 számjegyű sorszámozott egész lehet. Értéke nem lehet NULL.
- *suly*: Az autó önsúlya kg-ban kifejezve. Típusa INTEGER(4), maximum 4 számjegyű sorszámozott egész lehet. Értéke nem lehet NULL.
- *kW*: Az autó motorjának teljesítménye kW-ban megadva. Típusa INTEGER(3), maximum 3 számjegyű sorszámozott egész lehet. Értéke nem lehet NULL.
- *ev*: Az autó évjárata. Típusa INTEGER(4), maximum 4 számjegyű sorszámozott egész lehet. Értéke nem lehet NULL.
- *benzines*: Megadja, hogy az autó benzin üzemű-e, ha igen akkor az értéke *igen* különben pedig *nem*. Típusa VARCHAR(10). Értéke nem lehet NULL.
- *diesel*: Megadja, hogy az autó diesel üzemű-e, ha igen akkor az értéke *igen* különben pedig *nem*. Típusa VARCHAR(10). Értéke nem lehet NULL.
- *ajto*: Az autó ajtóinak a száma. Típusa INTEGER(1), maximum 1 számjegyű sorszámozott egész. Értéke nem lehet NULL.
- *felszereltség\_kategória*: Az autó felszereltségi kategóriája. Típus VARCHAR(10). Az értéke nem lehet NULL.
- *hajtott\_kerekek*: Az autó meghajtott kerekeinek a száma. Típus INTEGER(1). Az értéke nem lehet NULL.
- *vegsebesség*: Az autó végsebessége km/h-ban kifejezve. Típusa INTEGER(3), maximum 3 számjegyű sorszámozott egész. Értéke nem lehet NULL.
- *atlagfogyasztás*: Az autó átlagfogyasztása vegyes használat mellett 100 km-en, literben kifejezve. Típusa FLOAT, tehát lebegőpontos szám lehet. Értéke nem lehet NULL.

- *injektor*: Megadja, hogy az autó tartalmaz-e injektort, ha igen akkor az értéke *igen*, különben pedig *nem*. Típusa VARCHAR(4). Értéke nem lehet NULL.
- *valto*: Az autó sebességváltójának a típusa. Típusa VARCHAR(10), értéke kétféle lehet *manuális* vagy *automata*. Értéke nem lehet NULL.
- *kep*: Az autó 200x100 pixel méretű képe. Típusa VARCHAR(100), értéke pedig a képfájl neve kiterjesztéssel együtt. Értéke nem lehet NULL.
- *gyorsulas*: Az autó 0-ról 100 km/h-ra való gyorsulási ideje másodpercben kifejezve. Típusa FLOAT, tehát lebegőpontos szám. Értéke nem lehet NULL.

Láthatjuk, hogy az autók táblában az adott autókról minden az érdeklődők számára fontos paraméter szerepel, ezzel segítve a pontos rálátást az autóra.

### 3. Az „autolehet” tábla

Az autolehet tábla megadja, hogy az autok táblában lévő autók közül melyek kölcsönözhetőek illetve melyek tesztvezethetők. A tábla mindössze három mezőt tartalmaz melyek a következők:

- *Autokazonosito*: A kölcsönözhető illetve tesztvezethető autó azonosítóját tartalmazza, amely megegyezik az autó „autok” táblabeli azonosítójával. Típusa INTEGER, tehát sorszámozott egész lehet. Értéke nem lehet NULL.
- *Kolcs*: Megadja, hogy az adott azonosítójú autó kölcsönözhető-e, ha igen akkor az értéke *igen* különben pedig *nem*. Típus VARCHAR(4), maximum 4 karakter hosszúságú sztring lehet. Alapértelmezett értéke az *igen*. Az értéke nem lehet NULL.
- *Teszt*: Megadja, hogy az adott azonosítójú autó tesztvezethető-e, ha igen akkor az értéke *igen* különben pedig *nem*. Típus VARCHAR(4), maximum 4 karakter hosszúságú sztring lehet. Alapértelmezett értéke az *igen*. Az értéke nem lehet NULL.

### 4. A „kolcsonzesek” tábla

Ebben a táblában a kölcsönzéseket tartjuk nyilván az alábbi adatok tárolásával:

- *ID*: A kölcsönzés azonosítója. Ez a tábla elsődleges kulcsa. Típusa INTEGER. Értéke automatikusan növekszik. Értéke nem lehet NULL.

- *FelhasznaloAzonosito*: A kölcsönző személy felhasználói azonosítója. Típusa VARCHAR(30). Értéke nem lehet NULL.
- *AutoAzonosito*: A kölcsönzött autó „autok” táblabeli azonosítója. Típusa INTEGER. Értéke nem lehet NULL.
- *Mikortol*: A kölcsönzés kezdetének dátuma. Típusa DATE. Értéke nem lehet NULL.
- *Meddig*: A kölcsönzés végének dátuma. Típusa DATE. Értéke nem lehet NULL.
- *Szalon*: A kölcsönzés helyszínéül kiválasztott szalon címe. Típusa VARCHAR(100). Értéke nem lehet NULL.

Tehát ezen adatok nyilvántartásával tartjuk nyilván a kölcsönzéseket.

## 5. A „tesztvezetések” tábla

Ebben a táblában a tesztvezetéseket tartjuk nyilván az alábbi adatok tárolásával:

- *ID*: A tesztvezetés azonosítója. Ez a tábla elsődleges kulcsa. Típusa INTEGER. Értéke automatikusan növekszik. Értéke nem lehet NULL.
- *JelentkezoAzonosito*: A tesztvezetésre jelentkező személy felhasználói azonosítója. Típusa VARCHAR(30). Értéke nem lehet NULL.
- *AutoAzonosito*: A kölcsönzött autó „autok” táblabeli azonosítója. Típusa INTEGER. Értéke nem lehet NULL.
- *Idopont*: A tesztvezetés időpontja napra pontosan. Típusa DATE. Értéke nem lehet NULL.
- *Ora*: A tesztvezetés napjára kért időpont. Típusa DATE. Értéke nem lehet NULL.
- *Szalon*: A tesztvezetés helyszínéül kiválasztott szalon címe. Típusa VARCHAR(100). Értéke nem lehet NULL.

Tehát ezen adatok nyilvántartásával tartjuk nyilván a tesztvezetéseket.

## 6. A „megrendelés” tábla

Ebben a táblában a megrendeléseket tartjuk nyilván az alábbi adatok tárolásával:

- *azonosito*: A megrendelés azonosítója. Ez a tábla elsődleges kulcsa. Típusa INTEGER. Értéke automatikusan növekszik. Értéke nem lehet NULL.

- *megrendelo\_neve*: A megrendelő teljes neve. Típusa VARCHAR(50). Értéke nem lehet NULL.
- *auto\_marka*: A megrendelt autó márkaneve. Típusa VARCHAR(100). Értéke nem lehet NULL.
- *auto\_tipusa*: A megrendelt autó típusa. Típusa VARCHAR(100). Értéke nem lehet NULL.
- *auto\_szin*: A megrendelt autó színe. Típusa VARCHAR(100). Értéke nem lehet NULL.
- *auto\_kw*: A megrendelt autó motorjának a teljesítménye kW-ban kifejezve. Típusa INTEGER(3). Értéke nem lehet NULL.
- *auto\_uzem*: A megrendelt autó üzemanyaga. Típusa VARCHAR(10). Értéke nem lehet NULL.
- *auto\_kat*: A megrendelt autó felszereltségi kategóriája. Típusa VARCHAR(50). Értéke nem lehet NULL.
- *auto\_injektor*: Megadja, hogy a megrendelt autó tartalmaz-e injektort, ha igen akkor az értéke *igen* különben pedig *nem*. Típusa VARCHAR(10). Értéke nem lehet NULL.
- *auto\_valto*: A megrendelt autó sebességváltójának típusa. Típusa VARCHAR(10). Értéke nem lehet NULL.
- *idopont*: A megrendelés időpontja. Típusa TIMESTAMP. Értéke nem lehet NULL. Alapértelmezett értéke CURRENT\_TIMESTAMP.

Tehát ezen adatok nyilvántartásával tartjuk nyilván a megrendeléseket.

## 7. A „szerviz” tábla

Ebben a táblában a szervizben lévő autókat tartjuk nyilván az alábbi adatok tárolásával:

- *SzervizSzam*: A szervizelési folyamat azonosítója. Ez a tábla elsődleges kulcsa. Típusa INTEGER. Értéke automatikusan növekszik. Értéke nem lehet NULL.
- *TulajdonosAzonosito*: A szervizelési folyamatban résztvevő autó tulajdonosának a felhasználói azonosítója. Típusa VARCHAR(30). Értéke nem lehet NULL.

- *Rendszam:* A szervizelési folyamatban résztvevő autó rendszáma. Típusa VARCHAR(10). Értéke nem lehet NULL.
- *Allapot:* A szervizelési folyamat állapota (státusza). Típusa VARCHAR(11). Értéke nem lehet NULL.
- *BeadasIdeje:* Az autó szervizbe történő beadásának az időpontja. Típus TIMESTAMP. Értéke nem lehet NULL. Alapértelmezett értéke CURRENT\_TIMESTAMP.

Tehát ezen adatok nyilvántartásával tartjuk nyilván a szervizben lévő autókat illetve a szervizelési folyamatokat.

## 8. A „szervizjelentés” tábla

Ebben a táblában a szerelők által, a szervizelt autókhoz írt jelentéseket illetve a felhasználók által írt üzeneteket tartjuk nyilván az alábbi adatok tárolásával:

- *Azonosito:* A szerviz jelentés azonosítója. Ez a tábla elsődleges kulcsa. Típusa INTEGER. Értéke automatikusan növekszik. Értéke nem lehet NULL.
- *SzervizSzam:* A szervizelési folyamat azonosítója. Típusa INTEGER. Értéke automatikusan növekszik. Értéke nem lehet NULL.
- *Nev:* A szervizjelentést író szerelő neve. Típusa VARCHAR(50). Értéke nem lehet NULL.
- *Jelentes:* A jelentés szövege. Típusa TEXT. Értéke nem lehet NULL.
- *JelentesIdeje:* A jelentés írásának az időpontja. Típusa TIMESTAMP. Értéke nem lehet NULL. Alapértelmezett értéke CURRENT\_TIMESTAMP.

Tehát ezen adatok nyilvántartásával tartjuk nyilván a szerviz jelentéseket.

## 9. A „vélemény” tábla

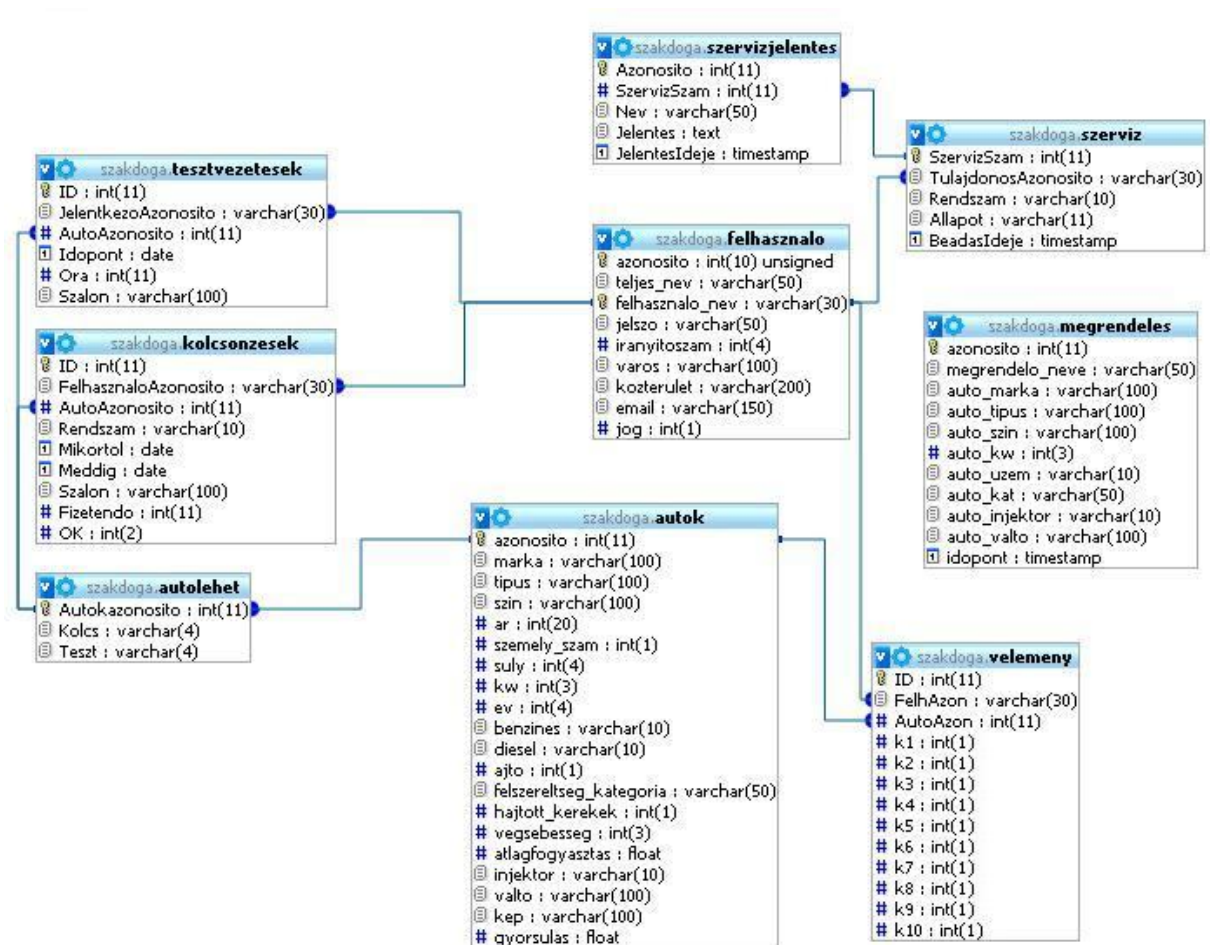
Ebben a táblában a tesztvezetett autók értékeléseit tartjuk nyilván az alábbi adatok tárolásával:

- *ID:* A vélemény azonosítója. Ez a tábla elsődleges kulcsa. Típusa INTEGER. Értéke automatikusan növekszik. Értéke nem lehet NULL.
- *FelhAzon:* A tesztvezetett felhasználó azonosítója. Típusa VARCHAR(30). Értéke nem lehet NULL.

- *AutoAzon*: A tesztvezetett autó azonosítója. Típusa INTEGER. Értéke nem lehet NULL.
- *k1-k10*: Az adott kérdésre adott pontszám 5-ös skálán. Típusa INTEGER. Értéke nem lehet NULL.

Tehát ezen adatok nyilvántartásával tartjuk nyilván a tesztvezetési véleményeket.

A táblák közötti kapcsolatokat az alábbi ábra mutatja:



## **IV. Összefoglalás**

Végére értünk a program bemutatásának. Láthattuk, hogyan épül fel az alkalmazás. Belekóstolhattunk a programozás ezen területébe. Láthattuk, hogy milyen részek szükségesek egy ilyen alkalmazás fejlesztéséhez. A webes alkalmazás fejlesztése napjainkban nagyon fontos szerepet tölt be a mindennapi életünk számtalan területén. A mai rohanó világunkban nagyon elterjedtek a weben keresztüli szolgáltatások, melyek előnyt jelentenek mind a szolgáltatók és mind a szolgáltatásokat igénybevevők részéről.

Napjainkban a számítógép használata szinte kikerülhetetlen, a munkahelyünkön, otthon, a különféle ügyeinket intézve mindenhol a számítógépet hívjuk segítségül, hogy a világhálón böngésszünk, illetve Web-szolgáltatásokat vegyünk igénybe. Manapság a SOA (Service Oriented Architecture) egyre nagyobb méreteket ölt, amely biztosítja, hogy a szolgáltatások mögött ténylegesen szolgáltatások legyenek. Ebben az esetben a szolgáltatás nem csak egy alkalmazás, hanem több alkalmazás összeintegrálásából is keletkezhet, felhasználva a már meglévő értékeket újabb értékeket állíthatunk elő, tehát az újrafelhasználás nagyon fontos szerepet játszik. Ha a mi alkalmazásunkat vesszük, akkor a mi alkalmazásunkban is vannak olyan elemek, amelyeket egy másik alkalmazás fejlesztésekor használhatunk, anélkül hogy újra meg kellene írunk. Ilyen újrafelhasználható modulnak tekinthető például, a bejelentkezést megvalósító PHP fájl, hiszen bármilyen alkalmazást fejlesztünk, szinte biztosra vehetjük, hogy szükségünk lesz bejelentkezési funkcióra, hiszen ha valamilyen szolgáltatást nyújtunk, akkor ismernünk kell azokat a személyeket, akik igénybe veszik a mi általunk nyújtott szolgáltatásokat, ezért is fontos a regisztráció. A regisztráció során megadott adatok nélkül nem tudnánk egyes igényeket kiszolgálni.

Ezt az alkalmazást olyan személyeknek ajánljuk, akik valamilyen autókereskedést üzemeltetnek és szeretnék profiljukat bővíteni a webes világ irányába. Olyan online szolgáltatásokkal szeretnék bővíteni repertoárjukat, mint az autórendelés, autókölcsönzés, autó szervizelés illetve tesztvezetési lehetőség biztosítása. Ezen szolgáltatások új irányt mutatnak az autókereskedelemnek, hiszen sokkal egyszerűbben és közvetlenül az emberekhez tudják eljuttatni a kereskedők az akcióikat, ajánlataikat és így még több ember figyelmét tudják felkelteni, illetve több embert képesek elérni a világháló segítségével, mint korábban. Ne felejtjük el, hogy ezek a dolgok nagyon fontosak, manapság amikor is a világgazdasági válság hatásai miatt az autókereskedelem majdnem 50%-os visszaesést könyvelt el a 2009-es év első negyedében. Ezért az autókereskedelem résztvevőinek minden lehetőséget meg kell ragadniuk, hogy próbálják csökkenteni a világgazdasági válság okozta keresletcsökkenést az autók iránt.

Azt gondoljuk, hogy ez az alkalmazás jó példája annak hogyan lehet egy vállalkozásnak fennmaradni a mai válságos időben. Ma, amikor már szinte mindenkinek van laptopja annak érdekében, hogy akkor is internet közelben maradhasson, amikor valamilyen elfoglaltsága miatt utazni kényyszerül vagy csak egyszerűen nincs lehetősége, hogy a személyi számítógépet használja valamilyen okból kifolyólag. Amikor a mobilinternet hozzáférések száma minden eddigi rekordot megdönt és országos lefedettsége is egyre jobban kiterjed, szinte már minden nagyobb településen elérhető a mobilinternet egy laptop és egy USB kulcs méretű modem segítségével. Napjainkban az emberek az információ nagy részét a világháló segítségével szerzik be, olvasnak, vásárolnak, közügyeket intéznek, és még sorolhatnánk a rengeteg web-szolgáltatás által nyújtott lehetőségeket.

Továbbgondolva, hogy miért is kedveljük ennyire manapság a webes alkalmazásokat. A válasz egyszerű, bárholnan és bármikor elérhetőek a nap 24 órájában, nem igényelnek nagyfokú informatikai tudást, nem kell programot esetleg programokat telepíteni az elérésükhöz. A jól bevált internet hozzáféréssel rendelkező számítógépünk vagy laptopunk és a kedvenc böngészőnk segítségével web-szolgáltatások tárháza nyílik meg előttünk és nekünk csak annyit kell tennünk, hogy kiválasztjuk közülük a számunkra szükségeset.

Így manapság talán még jobban helytálló az a mondás miszerint:

„A világ csak egy kattintásra van!”

### **Megjegyzés:**

Az alkalmazás **Mező András** által készített részei az alábbiak:

- Regisztráció (*regisztracio.php*)
- Profil (*profil.php*)
- Főoldal (*index.php*)
- Autószalon (*szalon.php*)
- Autó információk (*autoinfo.php*)
- Autóválaszték módosítás (adminisztrátori oldalak)
  - Autó paramétereinek a módosítása (*admin\_auto.php*)
  - Autó törlése az adatbázisból (*admin\_szalon.php*)
  - Új autó hozzáadása az adatbázishoz (*admin\_hozzaad.php*)
  - Megrendelések nyomon követése és törlése (*rendelesek.php*)

## V. Irodalomjegyzék

- *Php-ről*

- <http://hu2.php.net/manual/en/intro-whatis.php>
- <http://hu2.php.net/manual/en/intro-whatcando.php>
- <http://en.wikipedia.org/wiki/PHP>
- <http://hu.wikipedia.org/wiki/PHP>

- *HTML/XHTML-ről*

- <http://en.wikipedia.org/wiki/HTML>
- <http://hu.wikipedia.org/wiki/HTML>

- *CSS-ről*

- <http://en.wikipedia.org/wiki/CSS>
- <http://hu.wikipedia.org/wiki/CSS>

- *MySQL-ről*

- <http://en.wikipedia.org/wiki/MySQL>
- <http://www.mysql.com/mysql60>

- *Apache-ról*

- [http://en.wikipedia.org/wiki/Apache\\_HTTP\\_Server](http://en.wikipedia.org/wiki/Apache_HTTP_Server)
- [http://hu.wikipedia.org/wiki/Apache\\_HTTP\\_Server](http://hu.wikipedia.org/wiki/Apache_HTTP_Server)

## **VI. Köszönetnyilvánítás**

*Köszönjük hasznos tanácsait, a dolgozat elkészítésében nyújtott  
segítségét és szakmai útmutatásait témavezetőnknek  
**Dr. Rutkovszky Edéné** egyetemi tanársegédnek.*