

**Debreceni Egyetem**  
**Informatikai Kar**

**Webes felülettel ellátott adatbázis-kezelő alkalmazás készítése**  
**ingyenes szoftverek felhasználásával**

**Témavezető:**

**Pánovics János**

**Számítástechnikai munkatárs**

**Készítette:**

**Lászlók Zsolt**

**Programozó matematikus**

**Debrecen**

**2007**

# Tartalomjegyzék

<b>1. Bevezetés.....</b>	<b>3</b>
<b>2. A GNU/GPL licenc .....</b>	<b>5</b>
<b>3. A felhasznált szoftverek bemutatása .....</b>	<b>7</b>
<b>3.1. Az Apache webservert.....</b>	<b>8</b>
<b>3.2. A PHP szkriptnyelv.....</b>	<b>10</b>
<b>3.3. A MySQL adatbázis-kezelő rendszer.....</b>	<b>13</b>
<b>4. A tesztkörnyezet kialakítása, a szoftverek installálása.....</b>	<b>15</b>
<b>4.1. Apache.....</b>	<b>16</b>
<b>4.2. PHP.....</b>	<b>17</b>
<b>4.3. MySQL.....</b>	<b>18</b>
<b>5. Az alkalmazott újabb technológiák.....</b>	<b>19</b>
<b>5.1. Az URL rewriting.....</b>	<b>20</b>
<b>5.1.1. Az Apache mod_rewrite modulja.....</b>	<b>22</b>
<b>5.2. Az AJAX .....</b>	<b>24</b>
<b>5.2.1. Az Ajax PHP osztály.....</b>	<b>26</b>
<b>6. Az árlista-rendszer .....</b>	<b>27</b>
<b>6.1. Az alkalmazott adatbázisséma.....</b>	<b>28</b>
<b>6.2. Az adminisztrációs felület és a rendszer működése.....</b>	<b>30</b>
<b>7. Összefoglalás .....</b>	<b>37</b>
<b>8. Irodalomjegyzék .....</b>	<b>38</b>
<b>9. Köszönetnyilvánítás .....</b>	<b>39</b>

## **Bevezetés**

Napjainkban az internet térhódításának következtében egyre több cég látja szükségesnek, hogy termékeivel, szolgáltatásaival a világhálón is jelen legyen és segítségével vásárlóik online, naprakész információkat szerezhessenek.

A megrendelő szempontjából a honlapokkal szemben támasztott egyik legfontosabb tulajdonság, hogy azok állandóan friss tartalommal jelenjenek meg, így olyan webes felülettel rendelkezzenek, amelynek segítségével a legújabb tartalmak, változások hozzáértés, professzionális ismeretek hiánya nélkül is könnyen, gyorsan felvihetők és módosíthatók, valamint azon keresztül minden olyan beállítás, tulajdonság megadható legyen, amely a dinamikus tartalomra kihatással van.

Mivel a dinamikus felépítésű tartalom esetenként elég sok adatból épül fel (például több ezer termék egy árlistában), ezért azok tárolására a legjobb választás egy egyszerűen használható, gyors, hatékony relációs adatbázis-kezelő szoftver, amely leveszi a programozó válláról többek között az adatok felépítésének, tárolásának, kezelésének és az adatokon végzett műveletek implementálásának feladatát.

Ezen igények kielégítése miatt lényegesen megnőtt a szerepe azoknak a programozási nyelveknek, adatbázisrendszereknek, internetes technológiáknak, melyek ingyenesen elérhetőek, sőt, a felhasználók teljesen ingyen juthatnak hozzá ezeknek a szoftvereknek a rendszeresen frissített, legújabb verzióihoz. Az ilyen programok folyamatos használata nem igényel rendszeresen felmerülő költségeket. Ez a szempont különösen a kisebb vállalatoknak jelent sokat.

A szakdolgozat célja, hogy egy valós igényt kielégítő webalkalmazás fejlesztési folyamatán keresztül betekintést nyújtson a felhasznált programozási eszközökbe, technológiákba, egy ilyen rendszertől elvárt funkciókba, azok megvalósításának módjába és az elkészült program működésébe.

A szakdolgozatban először röviden ismertetem a GNU/GPL licencet, amely biztosítja a dolgozatban felhasznált programok és technológiák ingyenességét.

Ezután bemutatom a honlap-készítéshez leginkább használt, ingyenes programozási nyelv (PHP), adatbázis-kezelő rendszer (MySQL) és az ezeket futtató webservert (Apache) legfontosabb tulajdonságait, valamint ezen programoknak a tesztkörnyezet kialakítása érdekében szükséges installálási folyamatát és elengedhetetlen beállításait.

A következő részben ismertetésre kerülnek a dolgozat alapjául szolgáló árlista-rendszer készítése során alkalmazott legújabb technológiák, azok használatának lehetőségei és jelentősége.

A dolgozat befejező, bővebb részében pedig a webes adminisztrációs felülettel rendelkező, adatbázis-alapú, on-line árlista-rendszer felépítése, funkciói, az alkalmazott módszerek jellemzői kerülnek tárgyalásra.

## 2. A GNU/GPL licenc

Amint arról a bevezetőben szó esett, különösen a kisebb vállalatoknak és magánszemélyeknek nagyon fontos, hogy honlapjaik fejlesztése, majd azok elkészültével az interneten történő elérhetővé tétele a lehető legkevesebb ráfordítással megoldható legyen.

Ezért a webes programozáshoz, kiszolgáláshoz napjainkban elsősorban ingyenes programokat használnak, melyek többségét a GNU/GPL licenc alatt fejlesztik.

A GNU General Public License (rövid neve GPL, magyarul: GNU Általános Nyilvános Licenc) egy általános célú nyílt forráskódú licenc, amelyet a Free Software Foundation (FSF) tervezett a GNU projekt programkódjaihoz. A GPL ma használatos változata, a GPLv2 1991-ben készült, de küszöbön áll a GPLv3 megjelenése.

A GPL a legelső, és egyben a legelterjedtebb valóban szabad licenc. A GPL nemcsak, hogy szorosan kapcsolódik a szabad szoftver mozgalomhoz, de ez alapozta meg a mozgalom sikerét is.

A licenc lényege – más szabad licencekhez hasonlóan –, hogy a mű szabadon terjeszthető (akár pénzért is), és szabadon módosítható, de a terjesztései és a módosítások kötelezően szintén GPL licenc alatt kell, hogy megjelenjenek, így biztosítva, hogy a szabad tartalmakból készült bármilyen származékos mű is szabad maradjon.

A GPL legnagyobb programbázisa jelenleg a Linux rendszermag és a szabad Unix-szerű rendszerek segédprogramjai.

A GPL nyelvezete az USA jogi környezetében megfelelő – nemzetközi szinten kevésbé, vagy nem. Ezt a hibát orvosolja többek között a GPLv3, a GPL következő, széles körben megvitatásra kerülő változata, valamint a szerzői jog friss változásai hazánkban is.

Fontos megjegyezni, hogy az ilyen licenc alatt kiadott programokra garanciavállalás nincs. A licenc szerint:

„Mivel a program használati joga díjmentes, a programhoz nem jár garancia az ide vonatkozó jogszabálynak megfelelően. Amennyiben a szerzői jogok tulajdonosai írásban másként nem nyilatkoznak, a program úgy ahogy van kerül kiadásra mindenféle garanciavállalás nélkül. A programmal kapcsolatban nincs sem származtatott, sem egyéb garanciavállalás, beleértve de nem kizárólagosan a forgalomba hozhatóságra vagy alkalmazhatóságra vonatkozó garanciákat. A program minőségéből és működéséből fakadó összes kockázat a felhasználót

terheli. Ha a program hibásan működik, a felhasználónak magának kell vállalnia a javításhoz szükséges minden költséget.”

Mivel azonban ezeket a programokat általában hivatásos programozók készítik, így azok hibás működésétől nem kell tartani, az esetlegesen előforduló hibákat (bugokat) rendszerint rövid időn belül kijavítják.

### **3. A felhasznált szoftverek bemutatása**

A következő alpontokban bemutatom azokat az eszközöket, melyeket az árlista-rendszer fejlesztése során használtam: az Apache webservert, amely (többek között) a webes tartalom kiszolgálását végzi, a PHP szkriptnyelvet, amelyben a kód készült és a MySQL adatbázis-kezelő szoftvert.

Ismertetésre kerülnek az adott szoftverek fejlődésének fontosabb állomásai, a felhasználási területek, a lényeges tulajdonságok, a szoftverekben alapértelmezésként nem szereplő, de ingyenesen letölthető és alkalmazható bővítmények, illetve a negatívumok, ha vannak.

### 3.1. Az Apache webserverver

Az Apache HTTP szerver egy Unix, Windows, Novell NetWare, Mac OS X és más operációs rendszereken működtethető webserverver.

Első megjelenésének idejében az Apache volt az egyetlen ingyenes, nyílt forráskódú alternatíva a Netscape Communications vállalat webserververének (jelenleg Sun Java System Web Server a neve). Az azóta eltelt időben a funkciók és teljesítmény tekintetében más, Unix-alapú webserververek ellenfelévé fejlődött, sőt, 1996 óta az Apache a világ legnépszerűbb HTTP szervere: 2007 márciusában a honlapok 58%-át szolgálta ki.

Az Apache-ot az Apache Software Foundation szervezet által irányított, nyílt fejlesztői közösség fejleszti és tartja karban és a (GNU/GPL licenccel nagyrészt megegyező) Apache licenc alatt érhető el, így az ingyenes és nyílt forráskódú szoftver.

Az Apache 2. verziója az Apache 1.x kódjának lényeges frissítésével jött létre a jövőbeni modularizáció és az Apache Portable Runtime réteg megvalósítása érdekében. A 2. változat – többek között – jobban támogatja a UNIX szálkezelését, a nem UNIX-alapú operációs rendszereket (mint például a Microsoft Windows), egy új szoftverfejlesztői interfészt és IPv6 támogatást tartalmaz.

Az Apache elsősorban statikus és dinamikus weblapok kiszolgálására használható, de számos más feladatban is segíthet, ahol adatokat kell biztonságos és megbízható módon nyilvánosan elérhetővé tenni. Ilyen lehet például, ha valaki saját számítógépén lévő adatokat szeretne az interneten megosztani.

A Microsoft Internet Information Services (IIS) az Apache legfőbb versenytársa a webserververek területén, amelyet a Sun Java System Web Server és más programok, mint a Zeus Web Server követnek.

Jónéhány szoftvercsomag tartalmazza részeként az Apache webservvert, mint például az Oracle adatbázis-kezelő rendszer, vagy az IBM WebSphere applikációs szerver. A Mac OS X operációs rendszer beépített webservvere is az Apache.

Az Apache számos képességgel rendelkezik, közülük sok, mint lefordított modul áll rendelkezésre, amelyek a mag funkcionalitását növelik. Ilyenek például a szerver oldali programozási nyelveket támogató mod\_perl, mod\_python és PHP modulok, az autentikációt kezelő mod\_access, mod\_auth és mod\_digest modulok, valamint az SSL titkosítást támogató mod\_ssl és a későbbiekben bemutatásra kerülő URL rewrite engine, a mod\_rewrite.

Az Apache képes ún. virtuális hosztolásra (virtual hosting), amely lehetővé teszi, hogy az Apache egyetlen installált példánya sok különböző oldalt futtasson. Például egyetlen számítógép, amelyen telepítve van az Apache, egy időben képes kiszolgálni a [www.pelda.hu](http://www.pelda.hu), [www.teszt.hu](http://www.teszt.hu) és bármilyen más weboldalt is.

Ezeken kívül az Apache konfigurálható hibüzeneteket, adatbázisrendszer alapú autentikálást valamint olyan grafikus kezelői felületeket (GUI) is szolgáltat, amelyek nagyban megkönnyítik a szerver beállításának folyamatát.



## 3.2. A PHP szkriptnyelv

A PHP egy imperatív, objektum-orientált, funkcionális, gyengén típusos programozási nyelv, amelyet eredetileg dinamikus weblapok készítésére terveztek. A PHP-t főleg szerver oldali szkriptelésre használják – ahol az értelmező PHP kódot kap bemenetként és kimenetként egy weboldalt állít elő – de használható parancssori szkriptelésre és egyedi alkalmazások készítésére is.



A PHP sok relációs adatbázis-kezelő rendszerrel használható, a leginkább használt webszerverek mindegyikén fut és többféle operációs rendszerre is elérhető.

Ez a rugalmasság biztosítja, hogy mára több, mint 19 millió olyan internet domain létezik, amelyik PHP-val installált szervereken fut.

A PHP lehetőséget biztosít shell és desktop alkalmazások, démonok, log-feldolgozó és más rendszer-adminisztrációs programok készítésére is. A PHP egyre inkább átveszi az előbbi feladatokra megvalósítására általában használt Perl, Python, awk és shell-programozási nyelvek szerepét.

Ezen kívül platform-független GUI-alkalmazások is készíthetők a PHP-GTK könyvtár segítségével.

Az implementációt a „The PHP Group” készíti, és az ingyenes PHP Licenc alatt adják ki. A PHP 3-as verziója ún. kettős licenc: PHP és GNU/GPL alatt jelent meg, de a 4-es változattól kezdve saját licencet használnak, mert a GNU/GPL „copyleft” megjelenítésének kényszerítése nem felelt meg a fejlesztőknek.

Az 5-ös verzió számos újdonságot hozott be a nyelvbe:

- OO-támogatást,
- Az adatbázisok kezelésére szolgáló PHP Data Objects interfészt,
- A megújult engine által jelentős teljesítmény-növekedést,
- Jobb MySQL támogatást,
- Az SQLite beépített támogatását,
- Beépített SOAP-támogatást
- Kivételkezelést

A PHP elsősorban egy szűrőként működik. A PHP program bemenete egy szöveges és speciális PHP parancsokat tartalmazó fájl, vagy adatfolyam, kimenete pedig egy újabb adat vagy adatfolyam. A PHP 4-től kezdve az interpreter a bemenetet bájtkódra fordítja, amelyet a Zend Engine értelmez, így növelhető a teljesítmény a korábbi, interpreteres megoldáshoz képest.

A PHP a bemeneti adatban csak azokat a kódokat értelmezi, amelyek speciális határoló jelek (általában `<?php` és `?>`) között helyezkednek el. Minden más adatot közvetlenül, változatlan formában a kimenetre küld. Ennek legnagyobb előnye, hogy így PHP kifejezések helyezhetők el a HTML dokumentumok belsejében, például:

```
<?php
// PHP kifejezések
?>
Hagyományos HTML adat
<?php
// További PHP kifejezések
?>
```

A változók előtt egy `$` jelet kell elhelyezni, viszont típust nem kötelező megadni. A függvények és osztálynevekkel ellentétben a változónevek megkülönböztetik a kis- és nagybetűket.

A PHP a számokat platformfüggetlen formában tárolja, amely általában a 32-bites előjeles egész. Az egész számokat (pozitív és negatív) decimális, oktális és hexadecimális formában tárolhatja. A valós számok ugyancsak platformfüggetlen alakban kerülnek leképezésre.

A PHP-nek van logikai típusa, a boolean típus, amely hasonló a Java és C++ Boolean típusaihoz. A logikai értékűvé konvertálás szerint a nem nulla értékeket igaznak, a nullát hamisnak veszi, mint a Perl-ben.

A nullértékű adattípus egy olyan változót képvisel, amelynek nincs értéke. Az adattípus egyetlen értéke a NULL.

A tömbök egyaránt támogatják a numerikus és sztring értékeket. A tömbök elemei a PHP által kezelt típusok bármelyikét felvehetik értékül, köztük erőforrásokat, objektumokat sőt más tömböket is.

Az alapvető objektum-orientált funkciókat már a PHP 3-as verziójába beépítették, de az objektumok kezelését a PHP 5-ben teljesen újraírták a jobb teljesítmény és több lehetőség érdekében. A korábbi változatokban az objektumok primitív típusként voltak kezelve, aminek következtében egy változó értékadásakor vagy függvény-paraméterként való átadásakor az

egész objektum másolásra került. Az új módszer szerint az objektumok egy azonosító (egyfajta mutató) hivatkozta és nem az értéke.

A PHP 5-tel kerültek bevezetésre a privát (private) és a védett (protected) láthatóságú tagváltozók és módszerek, az absztrakt osztályok és módszerek, az objektum-orientált nyelvekhez (mint pl. a C++) hasonló standard konstruktor és destruktor létrehozás valamint a kivételkezelés.

A PHP lehetővé teszi, hogy a programozók a nyelv funkcióinak bővítése érdekében C nyelven kiterjesztéseket írjanak, amelyeket azután lefordíthatnak a PHP-val együtt vagy futásidőben, dinamikusan tölthetnek be. Ilyen kiterjesztésekkel vált támogatottá a Windows API, a Unix-szerű operációs rendszerek folyamat-menedzsmentje, a több-bájtos (Unicode) karakterek, az IRC, a dinamikus kép- és Flash-videó generálás.

Sok más szkriptnyelvhez hasonlóan a PHP forrásai is olvasható formában tárolódnak, ami ugyan nagy rugalmasságot biztosít, de biztonsági és teljesítménybeli problémákat ad.

Ezen problémákat tudják orvosolni egyrészt a kód optimalizálók, amelyek csökkentik a lefordított forrás méretét és olyan változtatásokat végeznek el, amivel csökkenthető a végrehajtási idő, másrészt a gyorsítók, amelyek a központi tárba cache-elik a lefordított PHP szkripteket, így elkerülhető azok állandó fordítása minden alkalommal, amikor a szkript lefut.

### 3.3. A MySQL adatbázis-kezelő rendszer

A MySQL-t – a világ legnépszerűbb Nyílt Forrású SQL adatbázis-kezelő rendszerét – a MySQL AB fejleszti, terjeszti és támogatja. A MySQL AB egy kereskedelmi vállalat, melyet a



MySQL fejlesztői alapítottak. Egy második generációs Nyílt Forrású vállalat, amely egyesíti a Nyílt Forrás értékeit és módszereit egy sikeres üzleti modellel.

A MySQL egy adatbázis-kezelő rendszer.

Az adatbázis adatok strukturált gyűjteménye. Bármi lehet egy egyszerű bevásárlólistától egy képgalériáig, vagy egy vállalati hálózat hatalmas mennyiségű információ halmazáig. Ahhoz, hogy felvegyünk, elérjünk és feldolgozzunk egy számítógépes adatbázisban tárolt adatot, szükségünk van egy adatbázis-kezelő rendszerre, mint amilyen a MySQL Szerver. Mivel a számítógépek nagyon jól kezelnek óriási mennyiségű adatot, az adatbázis-kezelő rendszerek főszerepet töltenek be különálló alkalmazásként vagy más programok részeként.

A MySQL egy relációs adatbázis-kezelő rendszer.

Egy relációs adatbázis különálló táblákban tárolja az adatokat ahelyett, hogy minden adatot egyetlen nagy tárolóhelyre rakna.

Ezzel gyorsabb és rugalmasabb lesz. A MySQL-ben szereplő SQL a Structured Query Language, azaz a Strukturált Lekérdező Nyelv rövidítése. Az SQL az adatbázisok eléréséhez leggyakrabban használt, szabványosított nyelv, melyet az ANSI/ISO Szabvány ír le. Az SQL szabvány 1986-ban jött létre és számos változat létezik. Ebben a kézikönyvben az „SQL-92” az 1992-ben megjelent, az „SQL:1999” az 1999-ben megjelent és az „SQL:2003” a jelenlegi változatot jelenti. „Az SQL szabvány” kifejezést használjuk az aktuális SQL szabvány jelölésére

A MySQL szoftver Nyílt Forrású.

A Nyílt Forrás az jelenti, hogy bárki használhatja és módosíthatja a szoftvert. Akárki letöltheti a MySQL-t az internetről és használhatja bármilyen fizetség nélkül. Ha úgy akarja, tanulmányozhatja a forráskódot és a saját igényeinek megfelelőre alakíthatja.

A MySQL a GPL (GNU General Public License) licencet használja annak meghatározásához, hogy mit és mit nem lehet csinálni a szoftverrel különböző helyzetekben.

Ha kényelmetlen a GPL licenc vagy a MySQL kódját egy kereskedelmi programba kell beágyazni, akkor egy kereskedelmi licencű változat vásárolható meg.

A MySQL Adatbázis Szerver nagyon gyors, megbízható és könnyen használható.

A MySQL Szerver a felhasználókkal szoros együttműködésben kifejlesztett praktikus tulajdonságokkal is rendelkezik. Teljesítménymérő oldalunkon megtalálhatóak a MySQL Szerver más adatbázis-kezelőkkel való teljesítménybeli összehasonlításai.

A MySQL Szerver eredetileg a nagyméretű adatbázisoknak a már létező megoldásoknál sokkal gyorsabb kezelésére lett kifejlesztve és több éve sikeresen alkalmazzák nagyteljesítményű környezetekben. A MySQL Szerver már most is gazdag és jól használható funkciókkal rendelkezik és folyamatos fejlesztés alatt áll. Jó kapcsolódása, sebessége és biztonsága teszi a MySQL Szervert különösen vonzóvá az interneten lévő adatbázisok elérésére.

A MySQL Szerver kliensként/szerverként és beágyazott rendszerekben is működik.

A MySQL Adatbázis Szerver egy kliens/szerver rendszer, amely többszálú SQL szerverből áll és amely különféle kiszolgálókat, számos különböző kliens programot és könyvtárat, adminisztratív eszközt és Programozói Felhasználói Interfészek (API-k) széles körét támogatja.

Ezen kívül a MySQL Szerver elérhető beágyazott többszálú könyvtárként is, melyet bárki felhasználhat a saját programjában, hogy kisebb, gyorsabb és könnyebben kezelhető, önálló termékhez jusson.

#### **4. A tesztkörnyezet kialakítása, a szoftverek installálása**

Nagyon fontos szempont, hogy az alkalmazás készítése során folyamatosan ellenőrizni lehessen, hogy az addig elkészült modulok megfelelnek-e az elvárásoknak. Ezért különösen ajánlott, hogy telepítsünk magunknak egy saját fejlesztői környezetet, amelynek segítségével azonnal ellenőrizni tudjuk az eredményeinket, illetve ha a program mégsem úgy működik, ahogyan szeretnénk, akkor a hiba helyéről és okáról is kaphatunk információkat.

Ezért a következő részben bemutatásra kerül az Apache webservert, a PHP szkriptnyelv és a MySQL relációs adatbázis-kezelő rendszer (Windows rendszerben történő) telepítése és lényeges beállításai.

## 4.1. Apache

A webservert telepítése azért lényeges lépés, mert a PHP telepítése önmagában nem elegendő. Ha csak azt telepítenénk fel, akkor ugyan parancssorból tudnánk futtatni a szkriptjeinket, de a böngészőnkben nem tudnánk megjeleníteni a kimenetet. Ezért van szükség egy, a böngészőkkel kommunikálni képes webservert telepítésére is.

Az Apache webservert legfrissebb változata letölthető a <http://httpd.apache.org/download.cgi> címről. Miután kiválasztjuk a tükörszervert, a Win 32 Binary (MSI Install) változat letöltését kezdeményezzük.

A telepítő program első fontosabb ablakában meg kell adnunk a telepítendő szerver tartományát, nevét és az admin e-mail címét. Itt adjuk meg a localhost / localhost / saját\_email\_cim értékeket. Alattuk állítsuk be, hogy az Apache szolgáltatásként települjön.

Ezek után adjuk meg a telepítés helyét és az el is indul.

A telepítés végeztével az Apache el is indul, amit a jobb alsó sarokban megjelenő ikon is mutat.

Ha szeretnénk, hogy az Apache nyilvános könyvtára ne az alapértelmezett telepítési\_út/htdocs mappa legyen, akkor ezt át kell írunk a telepítési\_út/conf/httpd.conf állományban, ahol a DocumentRoot változót kell módosítani.

Ügyeljünk rá, hogy a konfigurációs állományokban / jelet kell használnunk a mappák között.

Az itt megadott könyvtárra kell módosítani a DocumentRoot alatt lévő, az eredeti könyvtárra vonatkozó <Directory> paramétert is.

Az új beállítások érvénybe lépéséhez újra kell indítanunk az Apache-ot, amit a legkönnyebben az Apache Monitorral tehetünk meg.

## 4. 2. PHP

Ahhoz, hogy az Apache a PHP forrásainkat ne egyszerű szöveges fájlként jelenítse meg, a következő lépésben fel kell telepítenünk és be kell állítanunk a PHP értelmezőt, valamint rá kell vegyük az Apache-ot, hogy minden .php kiterjesztésű fájlt az értelmezőnek továbbítsa.

A PHP letölthető a <http://hu.php.net/downloads.php> címről, ahonnan Windows-os, zippel tömörített változatot kell kiválasztani, és a tükörszerver kiválasztásával letölteni.

Második lépésként csomagoljuk ki a zip állományt egy tetszőleges könyvtárba, például, ha az Apache-ot a telepítéskor felajánlott mappába tettük, akkor a c:\Program Files\Apache Software Foundation\PHP4-be.

Mozgassuk a \dlls és \sapi mappák teljes tartalmát a PHP gyökérkönyvtárába.

Ezek után elérhetővé kell tennünk az Apache számára a php4ts.dll és php.ini állományokat. Ezt a Windows Rendszertulajdonságok ablakának Speciális fülén lévő Környezeti változók gombra kattintva tehetjük meg. Itt a Path rendszerváltozóra duplán kattintva adjuk hozzá az eddigiektől pontosvesszővel elválasztva a PHP könyvtárának elérési útvonalát.

Nevezzük át a PHP könyvtárában lévő php.ini-recommended fájlt php.ini-re és nyissuk meg szerkesztésre.

Módosítsuk az extension\_dir beállítást arra a mappára, ahová a PHP-t kicsomagoltuk, de tegyük /extensions-t a végére.

Végül újra módosítanunk kell az Apache httpd.conf állományát az alábbi 2 sort hozzáadva a végéhez:

```
LoadModule php4_module "C:/Program Files/Apache Software Foundation/ PHP4/  
php4apache2.dll"
```

```
AddType application/x-httpd-php .php .php4 .php3
```

### 4.3. MySQL

Mivel a készítendő rendszer erőteljesen támaszkodik a MySQL adatbázis-kezelő által szolgáltatott adatokra és funkciókra, így a MySQL szervert is telepítenünk kell.

A szerver letölthető a <http://www.mysql.org/downloads/mysql/5.0.html#windows32> címről.

A MySQL 5-ös változatát kényelmesen, egyszerűen lehet telepíteni a saját telepítő programjával.

A Setup program indítása után semmi lényeges információt nem kell megadnunk, néhány kattintás után a telepítés el is kezdődik.

Miután végzett a telepítő varázsló, ajánlatos a „Configure the MySQL server now” opciót bepipálva hagyni és befejezni a telepítést.

Ha elindult a konfigurációs varázsló, válasszuk a Detailed Configuration, a Developer Machine, majd a Multifunctional Database opciókat.

Ezután, ha akarjuk, módosíthatjuk az InnoDB adatfájlok tárolásának helyét és beállíthatjuk, hogy egyszerre legfeljebb hányan kapcsolódhatnak a szerverhez. Mivel egy időben nem lesz sok kapcsolat, így alacsony értékre állíthatjuk.

A következő ablakban bekapcsolhatjuk, hogy a szerver fogadhat-e TCP/IP kapcsolatokat, melyik porton nyújtsa a szolgáltatását és milyen módban fusson.

Ezek után megadhatjuk az alapértelmezett karakterkészletet (válasszuk a második opciót vagy az UTF-8-at), majd beállíthatjuk, hogy a MySQL egy Windows szolgáltatásként fusson-e és ha igen, akkor milyen néven és, hogy elinduljon-e automatikusan a Windows-zal.

A következő ablakban biztonsági beállítások következnek: adjunk meg jelszót a root felhasználónak (bekapcsolhatjuk, hogy távoli számítógépről kapcsolódhat-e) és ne engedélyezzük névtelen (anonymous) fiók létrehozását.

Ezzel a tesztkörnyezet kialakításával végeztünk.

## **5. Az alkalmazott újabb technológiák**

A következő fejezet azokról az új technológiákról szól, amelyeket az árlista-rendszer elkészítése során alkalmaztam. Felhasználásukkal a fejlesztés könnyebben, gyorsabban haladt és az elkészült rendszer kódja, illetve kezelhetősége egyszerűbb lett.

## 5.1. Az URL rewriting

A rewrite modul a webszervernek egy olyan része, amely arra használható, hogy különféle okokból kifolyólag módosítsuk az URL-eket mielőtt azokat felhasználnánk. Ennek a technikának a neve URL rewriting („URL átírás”).

A technika alkalmazásával a következő előnyöket érhetjük el:

- A honlapunkon található URL-ek sokkal inkább felhasználó- és keresőbarátok
- Megelőzhetőek a nem kívánt belső linkelések (inline linking / hot linking)
- A külső felhasználók elől elrejthető a honlap belső felépítése

Ezek közül a legtöbb a HTTP szerverekre vonatkozik, amelyek alapértelmezett viselkedése, hogy az URL-eket fájlrendszerbeli bejegyzéseként értelmezik (fájlok és könyvtárak), de más szervereknél ennek a módszernek nincs értelme.

Az emberek a honlapok URL-jeit számtalan módon használják fel. Például, elküldik azokat e-mailben, kiírják őket nyilvános fórumokra vagy egyszerűen csak lejegyzetelik papírra. Ezek nem csak az egyes honlapok címekre, hanem az azokon belül lévő tartalomra is vonatkoznak.

Mivel ezzel a honlap látogatottsága növekedhet, ezért a fejlesztők támogatják is.

Ezért egy jól megtervezett weboldal esetén a felhasználók bármilyen URL-t megadhatnak, azok könnyen átláthatók, egyszerű felépítésűek.

Az URL-t akkor lehet könnyen használni, ha rövid de mégis tájékoztató. Tartalmaznia kell információkat a tartalmára vonatkozóan (és nem csak számokat) de nem lehet túl hosszú.

A keresőmotorok is sokkal egyszerűbben és gyorsabban indexelik be, és jelenítik meg találataik között az ehhez igazodó honlapokat.

Az Apache szervernek is létezik ilyen modulja, a mod\_rewrite, amit találóan „Az URL manipulálás svájci bicskája”-ként jellemezzük. Ezen kívül, egyszerűbb átírási feladatokra használható a mod\_alias kiegészítés is.

A Microsoft IIS webszerverhez készült URL rewriting modulok:

- IISRewrite
- ISAPI\_Rewrite
- URL Replacer
- ISAPI Rewrite Filter (nyílt forrású)

A Microsoft ASP.NET URL átíró HttpModule-jai:

- UrlRewriter.NET (ingyenes, nyílt forrású)
- URLRewriting.NET
- VirtualURL.NET for ASP.NET 2

A J2EE servlet container szerverei számára (mint például az Apache Tomcat, Resin, Orion):

- HttpRedirectFilter (nyílt forrású)
- UrlRewriteFilter (nyílt forrású)
- URL Rewriter (nyílt forrású, LGPL licencű)

Tehát az URL átíró modulok használatával a honlapot előállító alkalmazás az URL saját részére legmegfelelőbb formáját használhatja, de a felhasználók által látott URL-ek egyszerűek, tartalmasak maradnak.

### 5.1.1. Az Apache mod\_rewrite modulja

Az Apache mod\_rewrite modulja egy szabály-alapú, reguláris kifejezésekre épülő átíró motort használ az URL-ek azonnali átírására.

A modul bármennyi szabályt és bármennyi, az egyes szabályokhoz tartozó feltételt támogat, hogy igazán rugalmas és erőtelejs URL manipuláló módszer legyen. Az URL módosítása számos tényezőtől függ, mint például a szerver változóktól, környezeti beállításoktól, HTTP fejlécektől, időbélyegektől, sőt külső adatbázis lekérdezésektől is.

A modul a teljes URL-lel dolgozik (beleértve az elérési útvonalat is), mind a szerver- (httpd.conf) mind a könyvtáralapú (.htaccess) értelemben véve. Az átírt eredmény szolgálhat egy belső alfolyamat, külső lekérdezés vagy akár egy belső proxy átírányítás alapjául is.

Ennek a funkciógazdagságnak és rugalmasságnak azonban ára van: az összetettség. Így a modul teljes működésének, használatának módja nem sajátítható el egyetlen nap alatt.

Az elhelyezhető fontosabb direktívák és leírásuk:

- RewriteEngine: *on* és *off* állapotban lehet, az *off* az alapértelmezett. Engedélyezi vagy letiltja a futásidejű átírást. Ha *off* állapotban van, akkor a modul egyáltalán nem végez átírást, még a *SCRIPT\_URx* környezeti változókat sem frissíti
- RewriteOptions: néhány speciális, az aktuális szerver- vagy könyvtárszintű konfigurációs állományhoz tartozó opciót állíthatunk be vele. A *MaxRedirects*=szám formában adhatjuk meg, hogy a szerver hány átírányítás után állítsa meg a folyamatot.
- RewriteBase: segítségével explicit módon beállíthatjuk a könyvtárszintű átírást alapértelmezett URL-ét. Ha a webszerverünk URL-ei nem állnak közvetlen kapcsolatban a fizikai fájlrendszerrel, akkor mindenképpen el kell helyoznünk a RewriteBase direktívát minden .htaccess állományban ahol átírást használunk.
- RewriteCond: *tesztstring minta* alakban adhatunk meg feltételeket, és csak ezek teljesülése esetén kerülnek a feltételt követő átírási szabályok vizsgálgásra.
- RewriteRule: ezekkel adhatjuk meg az igazi átírási szabályokat. A direktíva bármennyiszer előfordulhat, és 1-1 átírási szabályt tartalmazhat. A direktívák felírási sorrendje fontos, mert ez a sorrend lesz érvényes a szabályok futás közbeni kiértékelése során.

Az árlista-rendszer URL átírását végző .htaccess állomány részlete:

```
RewriteEngine on
```

```
RewriteBase /
```

```
RewriteRule ^arlista/?$ arlista.php [L,QSA]
```

```
RewriteCond %{REQUEST_FILENAME} !-f
```

```
RewriteRule ^arlista/(admin\termekek\login)?$ /arlista.php?do=$1 [L,QSA]
```

```
RewriteRule ^arlista/logout$ /arlista.php?do=login&action=logout [L,QSA]
```

```
RewriteRule ^arlista/termekek/([0-9]{1,3})/?$ /arlista.php?tcsID=$1 [L,QSA]
```

További információk és példák a <http://httpd.apache.org/docs/1.3/misc/rewriteguide.html> címen találhatóak.

## 5.2. Az AJAX

Az AJAX (Asynchronous JavaScript and XML) egy webfejlesztési technológia interaktív Webes alkalmazások létrehozására. Az oldal kis mennyiségű adatot cserél a szerverrel a háttérben, így a lapot nem kell újratölteni minden egyes alkalommal, amikor a felhasználó módosít valamit. Ez növeli a honlap interaktivitását, sebességét és használhatóságát.

Az AJAX a következő eszközök kombinációját alkalmazza:

- XHTML (vagy HTML) és CSS a tartalom leírására és formázására
- DOM kliens oldali script nyelvekkel kezelve a dinamikus megjelenítés és a már megjelenített információ együttműködésének kialakítására
- XMLHttpRequest objektum az adatok aszinkron kezelésére a kliens és a webszerver között. Néhány Ajax keretrendszer esetén és bizonyos helyzetekben IFrame-et használnak XMLHttpRequest objektum helyett
- XML formátumot használnak legtöbbször az adattovábbításra a kliens és a szerver között, bár más formátumok is megfelelnek a célnak, mint a formázott HTML vagy a sima szöveg

Az AJAX használatának előnyei:

### Felhasználói felület

A legnyilvánvalóbb ok az AJAX használatára a felhasználói élmény fokozása. Az AJAX-ot használó oldalak viselkedése sokkal inkább hasonlít a desktop alkalmazásokhoz, mint a tipikus weboldalakhoz. Amikor egy linkre kattintás hatására a teljes weboldal újratöltődik, az sokszor időigényes művelet. Az AJAX-ot használó oldalak e helyett képesek rá, hogy csak az oldal szükséges részét frissítsék, így gyorsabb reagálást biztosítanak a felhasználói interakciókra. Néhányan úgy gondolják, hogy az AJAX lesz az a feltörekvő technológia, amelynek segítségével a jövőben a webes alkalmazások minden eddiginél interaktívabbá és így széles körben is sokkal népszerűbbé válhatnak.

### Letöltési sebesség és szerverterhelés csökkentés

Mivel az AJAX-ot használó oldalak a szervertől az adatokat HTML formázás nélkül kapják, ezért ez által a szerver terhelése és így a válaszideje is csökken. A kisebb mennyiségű adat pedig gyorsabban jut el a hálózaton a szervertől a kliensig. Tehát az AJAX-szal nagymértékben csökkenthető a szerver terhelése és növelhető az oldalak letöltési sebessége. A szervertől kapott adatokból a HTML kód a böngészőben jön létre Javascript segítségével, ami jól optimalizált programkód esetén legtöbbször gyorsabb mintha az erősen leterhelt szerver hozná azt létre. Ennek oka, hogy ma már a kliens oldalon a felhasználók viszonylag gyors személyi számítógépekkel rendelkeznek, amelyek terhelése általában lényegesen alacsonyabb, mint a szerveré. Ráadásul, ha valamelyik kliens gép a lassúsága vagy leterheltsége miatt mégis lassabban hozza létre a HTML kódot, az nem érinti a párhuzamosan jelenlévő többi klienst, ami annál nagyobb előny minél nagyobb a párhuzamosan jelenlévő kliensek száma. (Ha a HTML kód a szerveren generálódik, akkor ez pont fordítva van: annál jobban lassul az oldalgenerálás, minél több kienstől érkezik kérés egyszerre.) Tehát minél nagyobb egy oldal látogatottsága annál nagyobb előnyt jelent a kliens oldali HTML generálás. Mindezen jótékony hatásokat tovább erősíti, hogy az AJAX segítségével sokszor jól megvalósítható, hogy mindig csak az éppen szükséges minimális mennyiségű adat töltődjön le a szerverről

### 5.2.1. Az xajax PHP osztály

Az xajax egy olyan PHP osztály, amelyet PHP szkriptjeinkbe ágyazva könnyen elérhetjük, hogy alkalmazásunk az AJAX technológiát használva hívhasson meg PHP funkciókat vagy osztály módszereket. Mindössze annyi a feladatunk, hogy regisztráljunk egy vagy több funkciót az xajax objektum segítségével, amelyek a beépített válasz osztályt használva szabványos XML választ generálnak, majd a HTML fejrészébe kiíratjuk az osztály Javascript kódját, végül meghívunk egy Javascript módszert. Minden mást az xajax végez el helyettünk. Az xajax része egy olyan Javascript objektum, ami kezeli a böngésző és a szerver közötti kommunikációt.

Az xajax Apache és IIS szervereken, PHP 4.3.x és PHP 5.x nyelvekkel használható.

Az xajax használatának előnyei:

- Beépített XML válasz és Javascript üzenetkezelő rendszer, ami automatikusan kezeli a függvényeink által visszaadott adatot és, ha szükséges, frissíti az aktív tartalmat.
- Objektum orientált, hogy elkülöníthető legyen más kódoktól és könnyen implementálhassuk bele saját funkcióinkat.
- Firefox, Mozilla-alapú, Internet Explorer és Safari böngészőkön is használható
- Támogatja, hogy a Javascript funkciók egy- vagy többdimenziós, akár asszociatív tömböket adjanak át paraméterként a PHP módszereknek.
- A PHP által visszaadott adattal csak akkor frissíti a korábbi tartalmat, ha az eltérő.
- Minden kérés és válasz UTF-8 kódolással kerül továbbításra (amely meg is változtatható).
- Szinte minden Javascript kódot külső hivatkozásként lehet beemelni a HTML forrásba, így annak átláthatósága nem csökken.

## **6. Az árlista-rendszer**

A szakdolgozat következő, utolsó fejezete az előzőekben felsorolt szoftverek, technikák felhasználásával készült, webes adminisztrációs felülettel ellátott árlista-rendszer felépítéséről és működéséről szól.

A rendszer egy valós igény kielégítésére jött létre, jelenleg is működik.

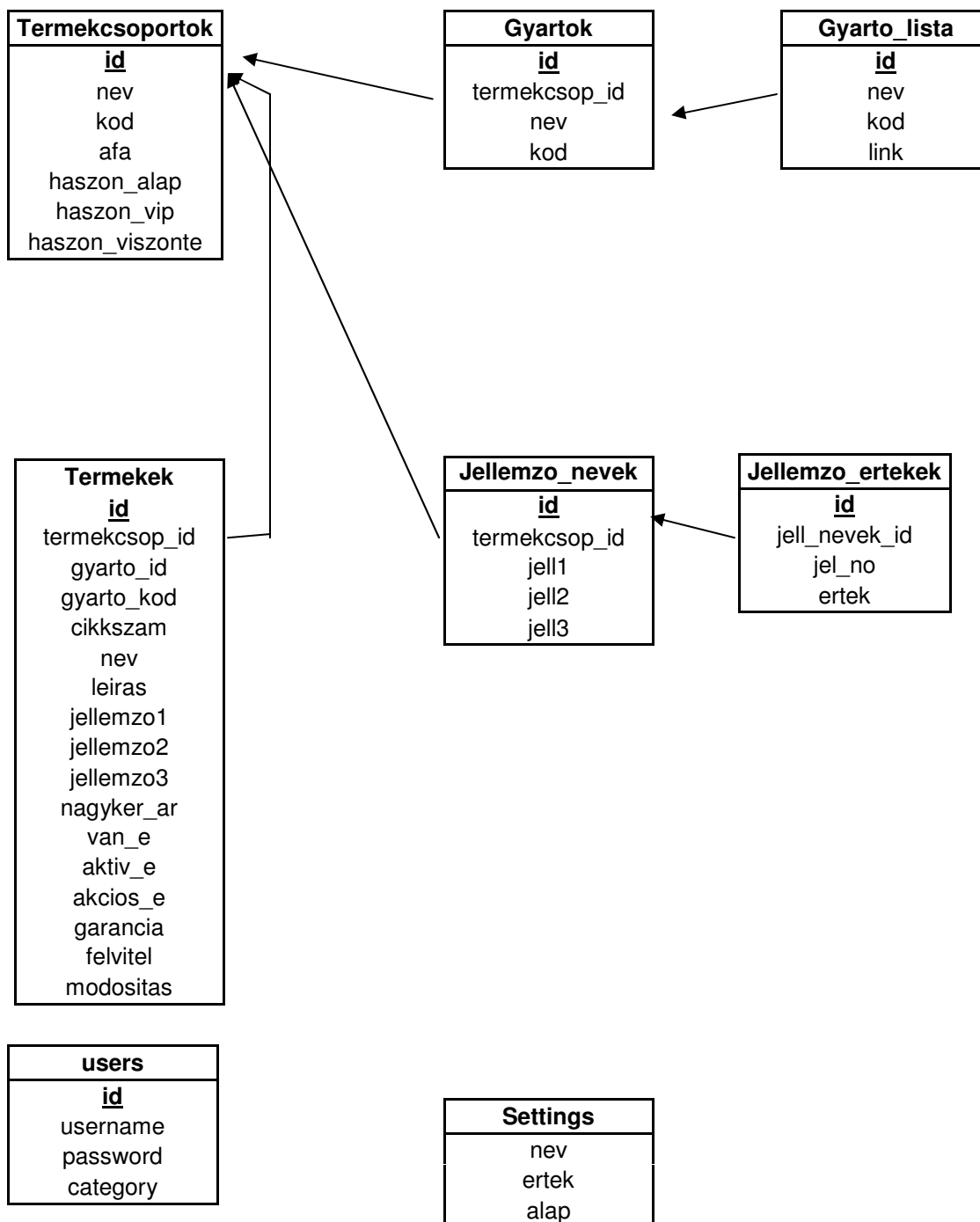
A megvalósításra vonatkozó igény már régebb óta felmerült. Mivel az addig az árlista adatainak tárolására szolgáló Excel táblázat karbantartása egyre nagyobb feladatot és egyre több időt jelentett a megrendelőnek, így egy olyan rendszerre volt szükség, amely meggyorsítja az adatok felvitelének idejét, csökkenti az állandó frissítésekre, karbantartásokra fordítandó időt és apró trükkökkel, ahol csak lehet automatizált segítségekkel támogatja a munkát.

Az árlista-rendszer elindításának első hónapja alatt mintegy tizenötezer látogatót szolgált ki bármilyen fennakadás nélkül.

## 6.1. Az alkalmazott adatbázisséma

A rendszer adatainak tárolására szolgáló adatbázis mintegy 20 táblából áll.

A következő ábra ezen táblák fontosabbjainak felépítését és a közöttük fennálló kapcsolatokat mutatja be.



A rendszer alapját képező termékek termékcsoporthoz vannak besorolva. Minden termékcsoporthoz van neve, kódja, ÁFA kulcsa, valamint az alapértelmezett, VIP, és viszonteladói felhasználók számára megadott haszonkulcsai.

A termékeknek többek között termékcsoporthoz id-je, gyártó id-je és kódja, cikkszám, neve, leírása, jellemzői, ára és további paraméterei vannak.

A termékek cikkszám egy 10 karakteres kód, amelynek első 3 karaktere a termék termékcsoporthoz kódja, ezt követően 3 karakter a termék gyártójának kódja, majd 4 karakternyi sorszám. Például az ALAASU0001-es termék lehet az Alaplapok termékcsoporthoz ASUS gyártójának első terméke.

A termék gyártóját a félregépelések elkerülése miatt egy előre megadott listából (Gyarto\_lista) lehet kiválasztani.

Minden terméknek legfeljebb 3 jellemzője lehet. A jellemzők nevei a Jellemzo\_nevek táblában találhatóak, ahol minden egyes termékcsoporthoz (legfeljebb) egy-egy bejegyzés tartozik

Az egyes jellemzőkhöz tartozó értékek egy külön táblában, a Jellemzo\_ertekek táblában tárolódnak. Itt megadható, hogy mely Jellemzo\_nev sorra vonatkozóan melyik jellemző sorszámra adunk meg újabb választható elemet.

Például ha a Jellemzo nevek egy sora az Alaplapok termékcsoporthoz tartozik, akkor ennek 3 jellemzője lehet a processzor-foglaló, chipkészlet és a memória típusa. Ez után vihetőek fel értékek a Jellemzo\_ertekek táblába: például ha az előző sor azonosítója 1, akkor 1-es jell\_nevek\_id mellett 1-es jell\_no, értéke Intel LGA775.

A users tábla a felhasználók adatait tárolja, a jelszó a nagyobb biztonság érdekében MD5 titkosítással kerül tárolásra. A category attribútum a hozzáférés szintjét adja meg (adminisztrátor / VIP / Viszonteladó).

A settings táblában a rendszerre vonatkozó egyéb beállítások találhatóak: a választható és alapértelmezett ÁFA kulcsok, a különböző szintű felhasználókra vonatkozó haszonkulcsok, az adminisztrátor e-mail címe, azok az IP-címek, amelyekről az adminisztrátor szintű belépés engedélyezett, stb.

## 6.2. Az adminisztrációs felület és a rendszer működése

Az adatbázis-kapcsolat kiépítését, a lekérdezések futtatását, eredményeik tárolását és visszaadását, valamint egyéb az adatbázissal kapcsolatos hibakezelési műveleteket valósítja meg az általános *DataBase* osztályból származtatott *MySQLDataBase* osztály. Az osztály főbb változói és módszerei:

- *\$host*, *\$username*, *\$password*, *\$database*: A kapcsolódáshoz használt hoszt, felhasználói név, jelszó és adatbázis.
- *\$connection*: Sikeres kapcsolódás esetén a kapcsolat-azonosítót tároló változó.
- *\$queryResults*: Tömb, amely a lekérdezések eredményét tárolja.
- *\$error*: Hibajelző.
- *MySQLDataBase(\$un, \$pw, \$db, \$ht = 'localhost')*: Konstruktor, amely a paraméterként kapott értékekre állítja a példányváltozók értékeit, ezután meghívja a *connect()* módszert amely kapcsolódik a MySQL szerverhez, és a *selectDb()* módszert, amely kiválasztja a kívánt adatbázist.
- *query(\$queryString, \$queryName = 'temp')*: A tényleges SQL-lekérdezéseket futtató függvény. Első paramétere az SQL-kifejezés, második pedig egy név, amely néven az eredmény a *\$queryResults* tömbben tárolva lesz. Ha nincs megadva második paraméter, akkor az alapértelmezett *temp* néven kerül tárolásra az eredmény.
- *fetchAssoc(\$queryName = 'temp')*: Az eredménytömb paraméterben megadott néven elérhető eredményhalmazának egy sorából képzett asszociatív tömbbel tér vissza.
- *close()*: Bezárja a megnyitott MySQL kapcsolatot.

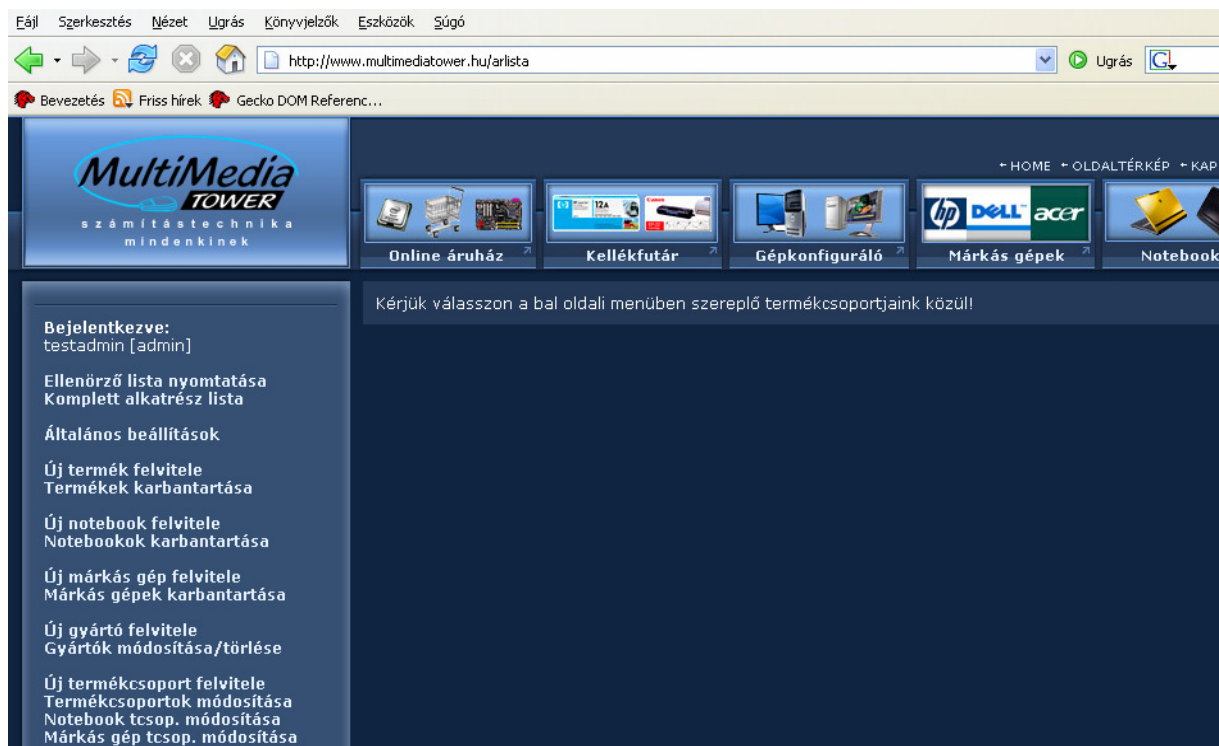
Az adminisztrációs felület eléréséhez először be kell jelentkezni. A bejelentkezési folyamatot a *User* osztály végzi el: a konstruktornak megadott értékekre állítja a példányváltozókat, majd meghívja a *login()* metódust, amely az előző osztályt felhasználva kapcsolódik az adatbázishoz és lekérdezi, hogy a *users* táblában található-e a megadott felhasználói nevű és jelszavú, aktív felhasználó. Ha igen, és a belépett felhasználó adminisztrátori jogokkal bír, akkor a bejelentkezésről e-mailben értesíti az adminisztrátort.

Ez után *session*-be (a szuperglobális *\$\_SESSION* tömb 'user' indexébe) menti a felhasználó adatait. Ez a lépés biztosítja, hogy a bejelentkezés után annak ténye lapról-lapra megmarad, így nem szükséges minden oldalon újra bejelentkezni.

Ha a felhasználó egy bizonyos ideig nem végez műveletet, akkor egy idő után a *php.ini* konfigurációs állomány *session*-ökre vonatkozó beállításaitól függően automatikusan törlésre kerül a beállított *session* változója, így újra be kell jelentkeznie.

Végül a módszer zárja az adatbázis-kapcsolatot és a kezdőoldalra irányítja a felhasználót.

Sikeres (adminisztrátor szintű) belépés esetén a következő képernyő látható:



Számunkra először a bal oldalon található adminisztrációs menü a lényeges. A következőkben sorra vesszük az egyes menüpontok alatt elérhető szolgáltatásokat a felhasználásuk szerinti sorrendben.

Először a gyártó listát töltjük fel. Ehhez az „Új gyártó felvitele” linkre kell kattintani, majd a megjelenő űrlapon megadni a gyártó nevét, kódját (rövidítését) és honlapjának címét, ha van. A gyártó nevének írásával egy Javascript eljárás kerül meghívásra, amely a már beírt első 3 betű alapján automatikusan kitölti a kód mezőt is. Ha ez a kitöltött kód nem megfelelő, akkor szabadon módosítható.

A Felvitel gombra kattintva a rendszer ellenőrzi, hogy létezik-e már ilyen névvel vagy kóddal gyártó. Ha igen, hibüzenetet ír ki, ha nem felveszi az adatbázisba az új gyártót.

A „Gyártók módosítása / törlése” menüponttal az így felvitt gyártók listáját érhetjük el. A megadott értékek egyből átírhatók és elmenthetők, vagy ha szükséges a gyártó törölhető a listából (megerősítés után).

Név	Kód	Link	Akció	
3COM	3CO	http://www.3com.com	módosít!	töröl!
3M	MMM	http://www.3m.com	módosít!	töröl!

Következő lépésként termékcsoportokat kell felvinnünk a rendszerbe, hogy azokhoz később termékeket adjunk. Termékcsoport felvitelére az „Új termékcsoport felvitelére” linken nyílik lehetőség.

Mint az ábrán is látszik, a felvitel során egy helyen, jól átláthatóan megadható a termékcsoport minden tulajdonsága. Első lépésben a neve. Úgy, mint a gyártó felvitelnél, itt is automatikusan kiegészül az első 3 karakter alapján a kód mező.

A csoporthoz tartozó ÁFA értékek a settings táblából kerülnek kiolvasásra, és az alapértelmezettként megadott automatikusan ki is választódik a listából.

A következő 3 sorban az egyes haszonkulcsok adhatók meg, amelyek alap értéke szintén kiolvasásra kerül az adatbázisból.

Ez után a termékcsoporthoz korlátlan számú gyártót adhatunk meg az „Új” gomb megnyomásával és a megjelenő listából való kiválasztással. Ha később ehhez a csoporthoz új terméket viszünk fel, akkor annak gyártója már csak az itt megadott listából választható ki.

Végül meg kell adnunk a 3 darab jellemző tulajdonság nevét, és azokhoz a megfelelő gombok segítségével bármennyi értéket fel is vihetünk. A rendszer csak akkor helyez el újabb beviteli mezőt az egyes értékeknek, ha a saját csoportjában még nincs kitöltetlen.

A felvitt termékcsoportokat a „Termékcsoportok módosítása” ponttal tudjuk módosítani. Először valamennyi felvitt csoport listázásra kerül, ahol a nevük, kódjuk, ÁFA- és haszonkulcsaik módosíthatók közvetlenül, esetleg törölhetők is.

Ha a hozzájuk tartozó gyártókat vagy jellemzőket kívánjuk megváltoztatni, akkor a megfelelő „egyebek” gombra kell kattintanunk. Itt a már felvitt gyártókat törölhetjük, újabbakat vihetünk fel, valamint módosíthatjuk a jellemzők neveit és a hozzájuk tartozó értékeket, akár egyesével is.

Végül a Felvitel gombra kattintva – ellenőrzések után – tárolásra kerül az új termékcsoport.

Új termékcsoport felvitelére

Név:

Kód:

ÁFA: 20% ▾

Alap haszon: 1.2 %

VIP haszon: 1.0 %

Viszonteladó haszon: 1.0 %

A termékcsoportoz tartozó gyártók:

Új

3COM ▾

A termékcsoportoz tartozó jellemzők:

Jellemző1 neve:

Jellemző1 értékek:  Új

Jellemző2 neve:

Jellemző2 értékek:  Új

Jellemző3 neve:

Jellemző3 értékek:  Új

Felvitel!

Az előzőekkel végezve most már minden jellemzőjét meg tudjuk adni egy új terméknek. Új termék felviteléhez az „Új termék felvitel” címre kell kattintani.

Először ki kell választanunk, hogy melyik termékcsoporthoz tartozó terméket kívánunk felvinni a rendszerbe. Ha ez kiválasztásra került, akkor frissül az ahhoz tartozó gyártók listája és megjelenítődnek a jellemzők nevei is.

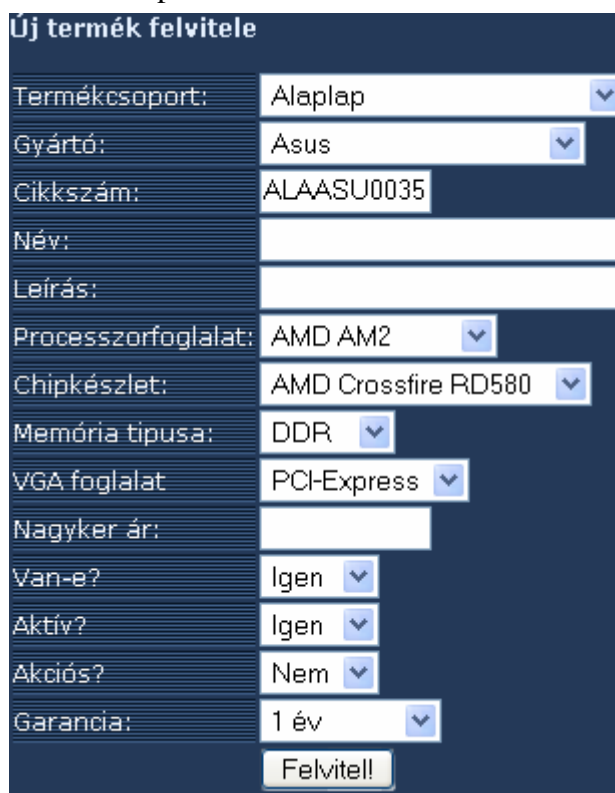
Ha most kiválasztjuk azt a gyártót, amelyiket szeretnénk, akkor a rendszer a termékcsoporthoz és a gyártó kódjaiból összerakja a cikkszám első 6 karakterét, majd megkeresi az adatbázisban, hogy ilyen kezdő cikkszámú mi a következő sorszám. Az így összeállított cikkszámot automatikusan be is írja a mezőbe.

Ezek után a termékcsoporthoz megfelelő 3 (esetleg 4) jellemző tulajdonság értékét kell megadnunk a listából való kiválasztással.

Végül meg kell adnunk a termék nagykereskedelmi árát, be kell állítanunk, hogy van-e belőle raktáron, aktív-e, akciós-e és a garancia idejét.

Felvitelkor a rendszer ellenőrzi, hogy létezik-e már ilyen cikkszámú termék (mivel a feljávált cikkszám átírható), valamint, hogy ki vannak-e töltve a szükséges mezők. Ha minden rendben van, akkor a beszúrás megtörténik.

Az alábbi kódrészlet a bevitt adatok ellenőrzésének egy részét és siker esetén felvitelét végzi el.



```
if(isset($_POST['toDo']) && $_POST['toDo'] === 'insert'){
    /***** Van-e már ilyen cikkszámú termék? *****/
    $sql = 'SELECT id
            FROM termek
            WHERE cikkszám = \' . substr($_POST['cikkszám'],0,10) . \';
    $db->query($sql);
    if(!$db->isEmpty()){
        $db->dbError('Már létezik termék ilyen cikkszámú!');
        printNewProductForm();
    }

    /***** Megfelelő-e a Cikkszám formátum? *****/
    elseif(isset($_POST['cikkszám']) && !ereg("^[A-Z]{6}+[0-9]{4}$",$_POST['cikkszám'])){
        $db->dbError('Hibás formátumú cikkszám!');
        printNewProductForm();
    }
}
```

```

elseif
  /* Termék felvétele */
  $sql = 'INSERT INTO termekek
    SET termekcsop_id = (SELECT id
      FROM termekcsoportok
      WHERE termekcsoportok.kod = \' . substr($_POST['tcsoport'],0,3) . \'),
    gyarto_id = (SELECT id
      FROM gyartok
      WHERE gyartok.kod = \' . substr($_POST['gyarto'],0,3) . \'
    AND
    termekcsop_id = (SELECT id
      FROM termekcsoportok
      WHERE termekcsoportok.kod = \' . substr($_POST['tcsoport'],0,3) . \'),
    gyarto_kod = \' . substr($_POST['gyarto'],0,3) . \',
    cikkszam = \' . substr($_POST['cikkszam'],0,10) . \',
    nev = \' . substr($_POST['nev'],0,60) . \',
    leiras = \' . substr($_POST['leiras'],0,100) . \',
    jellemzo1 = \' . substr($_POST['jellemzo1'],0,50) . \',
    jellemzo2 = \' . substr($_POST['jellemzo2'],0,50) . \',
    jellemzo3 = \' . substr($_POST['jellemzo3'],0,50) . \';
  if(isset($_POST['tcsoport']) && $_POST['tcsoport'] == 'ALA')
    $sql .= 'vga_foglalat = \' . $_POST['vga_foglalat'] . \';
  $sql .= ' nagyker_ar = ' . intval(substr($_POST['nagyker_ar'],0,7)) . ',
    van_e = \' . substr($_POST['van_e'],0,4) . \',
    aktiv_e = \' . substr($_POST['aktiv_e'],0,4) . \',
    akcios_e = \' . substr($_POST['akcios_e'],0,4) . \',
    garancia = \' . substr($_POST['garancia'],0,8) . \',
    felvitel = NOW(),
    modositas = NOW()';
  $db->simpleQuery($sql);

  if(!$db->error){
    echo 'Sikeres termék felvétel!';
  }
  else{
    echo 'Hiba a termék felvétele során!';
  }
}
}

```

A felvitt termékek adatainak módosítása a „Termékek karbantartása” menüponttal érhető el. Először ki kell választanunk, hogy melyik termékcsoporthoz tartozó termékek listáját szeretnénk látni.

Alapértelmezésként a megjelenített lista név szerint rendezett, de ez átkapcsolható, hogy az utolsó módosítás dátuma szerint csökkenő sorrend legyen érvényes.

Mivel egy ilyen árlista karbantartása során legtöbbször nem kell minden tulajdonságát módosítani egy terméknek, de azokat gyakran, így kezdetben a lista a legfontosabb tulajdonságokat írja ki, amik azonnal szerkeszthetők is.



Cikkszám	Név	Gar	Nagyker ár	Van-e?	Aktív-e?	Akciós-e?	Törlés	
ALAASU0024	ASUS AMD Crosshair	1 év	39000		Rend.	✓	Rend.	töröl!
ALAASU0018	ASUS AMD M2N	1 év	12000		✓	✓	Rend.	töröl!

Az árlistának ez a része már 2 helyen is használja az AJAX technológiát.

Egyrészt a listában megjelenő termékek árát át lehet írni, majd a mellette lévő mentés ikonra kattintva egy AJAX kéréssel az új adat azonnal elküldésre kerül a szerver felé a megfelelő PHP függvény paramétereként. Ez a függvény az új árra módosítja a megadott azonosítóval rendelkező termék árát, ellenőrzésképpen pedig az AJAX válaszban frissíti a beviteli mező értékét.

Másrészt egy grafikus kapcsolóval on-the-fly állíthatjuk a termékek van-e, aktív-e és akciós-e tulajdonságait. Például ha jelenleg az „igen”-t jelképező pipa található az adott helyen, majd erre kattintunk, akkor a megfelelő PHP módszer lekérdezi az adatbázisból annak aktuális értékét, az ellenkezőjére állítja és végül a válaszban a megfelelő képet az ellenkezőjére cseréli le.

A termékek további, a felvitel során megadott jellemzőit a cikkszámukra/nevükre történő kattintással bejövő új űrlapon lehet szerkeszteni.

A felhasználók számára legfontosabb funkció, hogy megtekinthessék egy adott termék vagy termékcsoporthoz az árakat. Ehhez több lehetőség közül választhatnak.

Egy-egy termékcsoporthoz összes termékének listázásához a bal oldalon található termékcsoporthoz kell kiválasztani a megfelelőt.

Itt alapértelmezésként listázásra kerül a termékcsoporthoz minden, aktívnek megjelölt terméke. A megjelenítés gyártók szerinti bontásban történik, minden termékhez a neve, nettó és bruttó ára, garancia ideje és információ arról, hogy jelenleg van-e raktáron a termékből.

Ha egy termék újdonságként van megjelölve, akkor narancs színnel kerülnek kiírásra a tulajdonságai, ha újdonság vagy mindkettő, akkor pedig piros színnel. Az eltérő szín okáról információt is kaphat, ha az egérkurzort a megfelelő termék nevére állítja.

A megjelenített termékek körét a felső részben elhelyezett választó elemekkel lehet szűkíteni gyártó, valamint a termékcsoporthoz megfelelő 3 tulajdonság bármilyen kombinációja alapján.

A másik lehetőség, hogy egy konkrét termékre keressünk a keresőben. A keresés történhet név vagy gyártó alapján. A név szerinti kereséskor a rendszer lekérdezi az adatbázisból, hogy van-e olyan terméknév, amely tartalmazza a megadott (minimálisan 2 karakter hosszú) karaktereket. Ha a kereséshez a felhasználó bekapcsolja, hogy a leírásban is keressen, akkor a termék leírása alapján is történhet a találat.

A gyártó szerinti kereséskor minden olyan termék listázásra kerül, amelyet a kiválasztott gyártó gyárt.

Szintén a keresésnek egy formája, hogy a látogató a csak újdonság vagy csak akciós termékekre is kereshet, amelyeknek külön link felel meg. Ekkor a megadott tulajdonsággal rendelkező valamennyi termék kiírásra kerül.

A keresések találatainak megjelenítése a korábbi rendszer szerint valósul meg.

Végül, hogy a felhasználó a megtekintett árakat elmenthesse, beépítésre került egy PDF-generáló lehetőség is.

Itt megjelenik az összes termékcsoporthoz, amelyek közül szabadon kiválasztható, hogy melyeket kíván a PDF dokumentumban látni. Bekapcsolható, hogy a listában a termékek leírása is szerepeljen-e.

Ez után rá kell kattintani a PDF készítés gombra, majd egy új ablakban megjelenik a kész dokumentum, amelyet aztán a látogató elmenthet vagy ki is nyomtathat.

Főleg a megrendelőnek készült, de a felhasználóknak is hasznos az a lehetőség, hogy egy előre megadott, a leginkább keresett termékeket tartalmazó, az adatbázisból mindig a legfrissebb tartalommal előállított lista is generálható PDF formátumban. Ennek a listának a tartalmát az adminisztrációs menüben, oldalankénti bontásban kell megadni úgy, hogy az egyes termékcsoporthoz szabadon variálható sorrendben állítjuk be.

## 7. Összefoglalás

A szakdolgozatban ismertetett szoftverek, technológiák napjainkban vezető szerepet töltenek be a dinamikus tartalmat előállító honlapok fejlesztésében.

Az első részben arról szerezhettünk ismereteket, hogy mit is jelent az ingyenesség a webes alkalmazás-fejlesztés területén. Mikor, milyen feltételek mellett használhatók a dolgozatban is ismertetett programok és milyen követelményeknek kell megfelelnie az elkészült alkalmazásnak.

A következő részben a felhasznált szoftverek, úgymint az Apache webservert, a PHP szkriptnyelv és a MySQL relációs adatbázis-kezelő rendszer kerültek bemutatásra. A fejezetből könnyen megérthető, hogy mire használhatók az egyes programok, milyen lehetőségeket biztosítanak a fejlesztőnek és melyek az alapvető tulajdonságaik.

Ezután az alkalmazás-fejlesztéshez nélkülözhetetlen tesztkörnyezet könnyű, gyors telepítésének folyamata került tárgyalásra, az előzőekben ismertetett szoftverek felhasználásával. A fejezet végére használható környezetet építhetünk fel, amely nagyban segíti a fejlesztés eredményeinek azonnali ellenőrzését.

Az ötödik fejezet célja az volt, hogy megismerkedjünk azokkal a friss programokkal, amelyek felhasználása mind a fejlesztő, mind a felhasználó szempontjából gyorsítja, egyszerűsíti a munkát.

Az utolsó részben a szakdolgozat alapjául szolgáló árlista-rendszer megvalósításának és működésének részei találhatóak. A rendszertől a megrendelő elvárta, hogy jelentősen növelje a termékek átláthatóságát, csökkentse a karbantartásra szánt időt és hasznos funkciókkal támogassa az adminisztrátor és a látogatók munkáját.

Úgy gondolom, hogy ezeket a szempontokat maradéktalanul sikerült teljesíteni, amelyet a megrendelő, és a rendszer elindítása óta egyre növekvő számú felhasználói kör panaszmentes munkája támasztanak alá.

A rendszer a szakdolgozatban ismertetett funkciókon kívül folyamatosan bővül, egyre több lehetőséget biztosítva a felhasználóknak és az üzemeltetőnek arra, hogy minden téren kihasználják a technológiák adta lehetőséget.

## 8. Irodalomjegyzék

A Wikipedia on-line lexikon

<http://hu.wikipedia.org> és <http://wikipedia.org>

A GNU Software Foundation honlapja

<http://www.gnu.org>

Az Apache Software Foundation honlapja

<http://www.apache.org>

PHP leírás és dokumentációk

<http://hu2.php.net/>

MySQL leírás és dokumentációk

<http://www.mysql.org/>

xajax dokumentációk

<http://www.xajaxproject.org/>

## 9. Köszönetnyilvánítás

Ezúton szeretném megköszönni **Pánovics Jánosnak**, a szakdolgozat témavezetőjének az elkészítés során nyújtott tanácsait, támogatását.