

SZAKDOLGOZAT

Varga Attila

Debrecen,

2008.

**Debreceni Egyetem
Informatika Kar**

**ESZKÖZIGÉNYLŐ ALKALMAZÁS
KÉSZÍTÉSE SZABAD SZOFTVEREK
FELHASZNÁLÁSÁVAL**

Témavezető:
Pánovics János
egyetemi tanársegéd

Készítette:
Varga Attila
informatika tanár

Debrecen,
2008.

Tartalom

Tartalom	2
Ábrák	3
1. Bevezető	4
2. Az IKT és eszközigénye	6
2.1. Oktatási informatikai stratégiai tervek	6
2.2. Helyzetkép	7
2.3. Az IKT kompetenciaterület fejlesztése az Ady Endre Gimnáziumban	10
2.3.1. Az iskola rövid bemutatása	10
2.3.2. Az IKT kompetenciaterület fejlesztése	12
3. GNU	15
4. Az alkalmazott szoftverek	17
4.1. Linux	17
4.2. Apache	18
4.3. MySQL	18
4.4. PHP	19
5. A rendszer telepítése	21
6. Röviden a dselect használatáról	23
6.1. 0. [E]lérés Válassz elérési módot.	23
6.2. 1. [F]rissítés. Frissíti az elérhető csomagok listáját, ha lehetséges	23
6.3. 2. [V]álaszt Csomagok választása.	24
6.4. 3. [T]elepít Kért csomagok telepítése, frissítése	24
7. Az adatbázis beállítása	25
8. A Web-szerver beállítása	28
9. Az alkalmazás használata	30
9.1. Nyitóoldal	30
9.2. Bejelentkezés	31
9.3. A jelszó módosítása	31
9.4. Bejelentkezve	32
9.5. Napi adatok	34
10. Adminisztráció	35
10.1. A csrend tábla	35
10.2. Az igény tábla	36
10.3. A kocsi tábla	37
10.4. Az osztály tábla	37
10.5. A tanar tábla	38
10.6. A tant tábla	38
10.7. A terem tábla	39
11. A program ismertetése	40
11.1. Az ellátott funkciók	40
11.2. index.php	40
11.2.1. Megjelenítés	41
11.2.2. A kiválasztott funkciók kezelése	42

11.3. A login.php	43
11.4. A login1.php	44
11.5. Az uj.php	44
11.6. A torol.php.....	45
12. Összegzés	46
Irodalomjegyzék:	47

Ábrák

1. ábra: ITK eszközök száma.....	14
2. ábra: Nyitóoldal	30
3. ábra: Bejelentkezés	31
4. ábra: A jelszó módosítása.....	31
5. ábra: Bejelentkezve.....	32
6. ábra: Adatbevitel	33
7. ábra: Napi adatok.....	34

1. Bevezető

„Mindennapi életünkben megnőtt az információ társadalmi szerepe, és felértékelődött az információszerzés képessége. Az egyén érdeke, hogy időben hozzájusson a munkájához, életvitelének alakításához szükséges információkhoz, képes legyen azokat céljának megfelelően feldolgozni és alkalmazni. Ehhez el kell sajátítania a megfelelő információszerzési-, feldolgozási-, adattárolási-, szervezési- és átadási technikákat, valamint az információkezelés jogi és etikai szabályait...

Megváltozik a pedagógus szerepe, az ismeretátadó és számon kérő pedagógusból az ismeretek közötti eligazodást segítő, tanácsadó, a megtalált információt értékelni, abban kételkedni tudó tanulók nevelőjévé válik. A tanulókat fel kell készítenie a problémamegoldó gondolkodásra, mint a feladatmegoldás magasabb szintjére.

Változik az iskola, mint szervezet szerepe is. Az önálló ismeretszerzés elérése érdekében a könyvtárhoz hasonlóan a számítógépteremben is lehetővé kell tenni az eszközökhöz való hozzáférést a tanórákon és azokon kívül is. A többi műveltségterület, tantárgy számára is biztosítani kell a géphasználatot. Meg kell jelennie a hagyományos tanórákon túlmutató informatikával támogatott projektmunkáknak is.” [1]

A fentiek szellemében 2004-ben stratégiai terv készült az Oktatási Minisztériumban, mely szerint rövid időn belül az oktatási szektornak olyan szintre kell fejlődnie, hogy a modern IKT¹-ra nagymértékben támaszkodva tartalmában és módszertanában is az információs társadalom kihívásainak megfelelő módon nyújtsa szolgáltatásait a társadalomnak. Az oktatási szektor adminisztratív hátterét korszerűsíteni kell, új szervezeti modelleket és folyamatokat, valamint korszerű informatikai rendszereket és infrastruktúrát kell kialakítani. A változtatások eredményeképp alkalmassá kell válnia a költséghatékony működésre és a minőségi szolgáltatásokra.

Vajon a NAT 2003-ban megfogalmazott célok, az Oktatási Minisztérium stratégiai tervei hogyan valósultak meg a gyakorlatban? Hogyan sikerült az informatikai szemlélet kialakításához szükséges feltételek megteremtése a debreceni Ady Endre Gimnáziumban? Hogyan jelenik meg az iskola mindennapjaiban, a tanárok és tanulók tevékenységében az informatika? Szakdolgozatom első részében ennek bemutatását kísérem meg. A továbbiakban e

¹ Informatikai és kommunikációs technológiák

célok eléréséhez rendelkezésre álló eszközök használatának, elosztásának segítésére készült alkalmazást mutatok be. Igyekszem minél több szempont szerint tenni ezt, kezdve az alkalmazott programok kiválasztásával a programkódot tartalmazó állományok rövid bemutatásáig.

2. Az IKT és eszközigénye

Az Oktatási Minisztérium Informatikai Főosztálya 2004. március 26-án megjelentette Oktatási Informatikai Stratégiáját. A 2004-2006 évre vonatkozó tervek széleskörű fejlesztést céloztak meg, jól átgondolva a fejlesztés legfontosabb irányait. Vizsgáljuk meg részletesen, hogy a fenti tervek hogyan valósultak meg a gyakorlatban.

2.1. Oktatási informatikai stratégiai tervek

Az oktatási kormányzat jövőképe szerint a pedagógusok az órára való felkészülés keretében több, előre elkészített tananyag és tanmenet közül választhatnak majd, amelyek online módon elérhetőek, tovább szerkeszthetőek és alakíthatóak. A választható tananyagok és tanmenet mellé módszertani útmutató is tartozik majd.

Az órán a tanár - felhasználva az „IKT az oktatásban” témában megszerzett ismereteit - az előre elkészített anyagot multimédiás tartalmak bemutatására is alkalmas berendezéssel mutatja be. A diákok interaktív módon vesznek részt a tananyag feldolgozásában, ami jelentősen hozzájárul a megértéshez és a tananyag elsajátításához. A diákok az órán kívüli időszakban is hozzáférnek az órai anyaghoz, annak kiegészítéseihez, ezekkel kapcsolatban önellenőrző és tanári ellenőrzés mellett végezhető feladatokat kapnak. A tananyaggal kapcsolatban órán kívül is kérdéseket tehetnek fel online módon. A pedagógus a dolgozatokat, házi feladatokat elektronikus módon gyűjti be és a feladattípustól függően, elektronikusan értékeli ki.

A tanárok, adott esetben a diákok is közreműködnek a tananyag-fejlesztésben. A tankönyvek, szöveggyűjtemények, szótárak, lexikonok jelentős része elektronikus formában is hozzáférhetővé válik.

Minden oktatási intézmény nagy sáv szélességű Internet kapcsolattal rendelkezik majd és legalább annyi számítógépes állomással, ahány az IKT-vel támogatott tananyagok tárgyainak oktatásához szükséges. Az intézmények minden egyes pedagógusa, tanára külön számítógéppel dolgozhat, amely munkát egységes adminisztrációs rendszer támogat az oktatás minden területén.

Az oktatás minden szereplője (diákok, oktatók, intézmények, stb.) egyedi azonosítóval rendelkezik, ami lehetővé teszi, hogy az összes oktatási esemény, folyamat nyomon követhetővé váljék. A tanulók tanulmányi előrehaladása, az intézménnyel való kapcsolata, esetleges

kedvezményei egyértelműen és egyszerűen meghatározhatóak. Az intézmények működése, a támogatások, bevételek felhasználása átláthatóvá válik. A különböző programok, elektronikus tananyagok sikeressége, felhasználása statisztikailag, külön kutatások nélkül is mérhető, ezáltal az erőforrásokat könnyebb a tényleg sikeres területek felé csoportosítani.

Az oktatók, diákok személyük és az oktatási azonosító kapcsolatát intelligens kártyával igazolják, amely azonosítja őket fizikailag és a virtuális térben zajló tranzakciók esetén is (pl.: ellenőrző, órára jelentkezés, tankönyvletöltés, kredit.). Egyszerűen, diszkréten és olcsón érvényesíthető a rászorultság elve a különböző állami juttatások esetén, az egyes személyek jogosultsági szintjét a kártyákhoz kötődő központi adatbázisban lehet tárolni, nem szükséges minden alkalommal a különböző igazolások benyújtása a juttatások igényléséhez.

Az oktatási kártyacsalád és elektronikus azonosítás egyik fontos területe az intézményi ügyvitel. Jelenleg az elektronikus formában továbbított jelentéseket papíron is meg kell erősíteni, mert a jogszabályok által meghatározott biztonsági szint jelenleg nem áll rendelkezésre. Az új kártyacsalád bevezetésével olyan intézményi kártyák bocsáthatók ki, amelyek jogkövetkezményekkel járó tranzakciók elektronikus bonyolítására is képesek, így költségeket lehet megtakarítani.

Az intézmények, minisztérium, fenntartók, háttérintézmények működése naprakész információkon és előzetes, adott esetben automatizált helyzetértékelésen alapul. A minisztérium döntéshozói szintjén az oktatás egészéről, folyamatairól, naprakész, részletes, akár egyedi szintű információ férhető hozzá, mindez az adatvédelem teljes tiszteletben tartásával. Az esetleges döntések jövőbeni hatásait előzetesen lehet elemezni. [2]

2.2. Helyzetkép

A felvázolt célok megvalósításának egyik legfontosabb lépése az SDT² létrehozása volt, amit 2004. szeptemberében indított útjára az Educatio Társadalmi Szolgáltató Kht. Sulinet Programirodája. Az SDT egy tartalom-menedzsment e-tanulás keretrendszer és digitális tananyag adatbázis.

A cél az IKT kompetenciák fejlesztésének támogatása egy „szabványos” digitális taneszköz-rendszer kialakításával, amely nem csak a digitális tananyagok, hanem a gyakorlati felhasználást segítő módszertani és technikai információk és azok felhasználását támogató szolgáltatások megvalósítása is egyben. Lehetővé teszi eddig nem létező elektronikus oktatási

² Sulinet Digitális Tudásbázis

anyagok online elérését és felhasználását (akár online, akár offline) a mindennapi oktatásban, illetve egy szerkesztő eszközt biztosít a pedagógusok, felhasználók számára saját tananyagok összeállításához, szerkesztéséhez, így támogatva a pedagógiai tevékenység hatékonyságának növelését, megújulását.

Az SDT foglalkozásai az informatika minden lehetséges eszközével, interaktív feladatokkal, szimulációkkal és tesztekkel támogatottak, továbbá a tananyagok a legapróbb elemekig bonthatók, újraserkeszthetők. Így lehetőség nyílik arra, hogy minél többféle pedagógiai céllal, módszerrel felhasználhassuk a tananyagokat.

A rendszer felépítésénél fogva kiváló lehetőséget nyújt differenciált osztálymunka végzéséhez, egyénre szabott tananyagok kialakításához. A diákok az SDT-ben megtalálható vagy megadható különböző pedagógiai segédletekkel, akár önállóan is fel tudják dolgozni a tananyagokat (tanulói információk, fogalomgráf, gráftallózó). A tananyagok tanórán kívüli tevékenységek elvégzéséhez is jó alapot nyújtanak. Az egyedi bejárési útvonalak, tanulói információk, gráfok használata segíti a tehetséggondozást és ugyanakkor a felzárkóztatást is, hiszen a tanulók saját tempójukban haladhatnak munkájukkal. A bőséges tartalom és a használatot megkönnyítő eszköztár, az interaktív környezet, versenyfeladatok összeállításához, házi dolgozatok elkészítéséhez, egyéb kutatómunkákhoz is alapul szolgálhat.

Regisztrált felhasználóként munkacsoportokat is létrehozhatunk, az adott munkacsoporthoz pedig fórumot rendelhetünk. Adhatunk a fórumról rövid leírást, és tulajdonosként, minden a fórumra vonatkozó információt, adatot beállíthatunk. (rendezési elvek, moderálások, hozzászólások)

A Sulinet Digitális Tudásbázis rendszerével 2008-ban az Educatio Társadalmi Szolgáltató Kht. eLearning Igazgatósága elnyerte az Európai E-Kiválóság díj (European Seal of Excellence) ezüst fokozatát.

A Sulinet speciálisan az IKT kompetenciákra épülő oktatáshoz szükséges készségek és tudás elsajátítását célzó pedagógus-továbbképzési programokat fejlesztett ki és akkreditált. 2004-2006 között 15 000 pedagógus kompetencia központú IKT továbbképzése (SDT alapú, vagy egyéb IKT alapú képzés) történt meg. Iskolánkból 9 pedagógus vett részt 60 órás SDT alapú képzésen és belső továbbképzéseket is tartottunk, amelyeken gyakorlatilag a teljes tantestület részt vett.

Kollégáink az SDT mellett a Moodle e-learning keretrendszer alkalmazásával is próbálkoznak. Ez a rendszer lehetőséget biztosít a tananyagok publikálására, házi feladatok fel-

töltésére, illetve segíti a kurzusok résztvevőinek gyors kommunikációját is. A tantestület több tagja küld feladatokat e-mail-ben a diákoknak, illetve a megoldásokat így kéri a tanulóktól.

2005-ben megjelent a költségvetésben a közoktatási intézmények informatikai fejlesztését szolgáló, kötött felhasználású normatíva. Ennek felhasználásával lízing formában korszerű számítógépekhez és perifériákhoz juthattunk. A kis értékű hardvereszközök beszerzését is normatívával támogatta a kormányzat, így az eszközök javítása és karbantartása is könnyebbé vált. A HEFOP pályázat további eszközök beszerzését tette lehetővé. Jelenleg európai uniós források felhasználásával a TIOP 1.1.1. 07. „Intelligens iskola” pályázat segítségével juthatunk újabb számítógépekhez, interaktív táblákhoz. Sajnos még messze vagyunk attól a céltól, hogy minden tanár külön számítógéppel dolgozhasson.

Fontos lépés volt az informatikai normatíva iskolai adminisztrációs szoftver beszerzésére fordítható részének meghatározása. Így iskolánk az akkreditált szoftverek közül a Taninform rendszert választotta. Jelenleg e nagyon bonyolult, sokoldalú szoftver adatbázisának feltöltése, illetve tesztelése folyik. Szeretnénk minél hamarabb tesztüzemben kipróbálni az elektronikus naplózás részét, illetve szeretnénk kihasználni a rendszer külvilág felé nyújtott adatszolgáltatási funkcióit (KIR-STAT, fenntartói adatszolgáltatás)

2004. február 10-én digitális alapközetéttel indult el a KözHáló program megvalósítása, amely az utóbbi évek legnagyobb, több tízmilliárd forintos, kormányzat által támogatott informatikai infrastruktúra-építési programja. A KözHáló program keretében mintegy 7300 szélessávú végpont került átadásra, ebből 5000 végpont a KözHáló alhálójaként működő Sulinet keretében valósult meg. Az internet-kapcsolat sávszélességét rendszeres emelésekkel mára 4 Mbit/s-ra emelték, ami már elfogadható munkakörülményeket eredményez. A jövőben további sávszélesítés indokolt az egyre bővülő szolgáltatások miatt.

Az Informatikai és Hírközlési Minisztérium az Oktatási Minisztérium valamint a Microsoft közötti megállapodás értelmében valamennyi magyarországi közoktatási intézmény díjmentesen használhatja a Windows Frissítés és az Office Pro termékeket az intézmény valamennyi számítógépén, valamint az oktatók és rendszergazdák otthoni számítógépein. A középiskolák jogosultak Microsoft szervercsomag (Windows Server, Exchange Server, ISA Server, SQL Server, Sharepoint Portal Server és Class Server) díjmentes igénylésére is. Az intézmények és diákjaik e licencek használatára a megújított licencszerződések alapján 2009. február 28-ig lesznek jogosultak.

A Közoktatási Információs Iroda tanügy-igazgatási szakportálja egyre jobban integrált, széleskörű szolgáltatásokat nyújt. Intézménytörzs hivatalos adatok, KIFIR, KIR-STAT, KIR-INFO, OKM 2007 FIT jelentések, érettségi adatok, tankönyvrendelés, oktatási azonosító kezelése, nyomtatványok megrendelése.

Az oktatás szereplői (intézmények, pedagógusok, diákok) egyedi azonosítóval rendelkeznek. Az azonosítók kezelését a KIR rendszerében végzik az intézmények. Az intelligens pedagógus és diákigazolványok bevezetése sajnos még nem valósult meg.

Az elmúlt tanévben bevezetésre került az elektronikus azonosítás és digitális aláírás az intézményi ügyvitelben. Eddig az elektronikus formában továbbított jelentéseket papíron is meg kellett erősíteni. Az új kártya bevezetésével jogkövetkezményekkel járó tranzakciók elektronikus bonyolítására is képesek lettünk, nem szükséges a papír alapú megerősítés. Digitális aláírással ellátott adatokat csak az arra jogosultak – iskolaigazgatók és helyetteseik - tudnak küldeni, mivel a felhasználók a központi Microsoft tanúsítványszolgáltató szerverén személyre szóló aktív chipkártyával és érvényes PIN kóddal azonosítják magukat. Sajnos a rendszer még nem működik zökkenőmentesen.

2.3. Az IKT kompetenciaterület fejlesztése az Ady Endre Gimnáziumban

2.3.1. Az iskola rövid bemutatása

Gimnáziumunk Debrecen Megyei Jogú Város oktatási intézménye, amelyet 1984-ben alapított a fenntartó jogelődje. Ady Endre nevét 1986. novemberében vette fel. Legnagyobb létszámban Debrecen, Hajdú-Bihar megye és a régió területéről iskolázzuk be a diákokat. Ma már országosan és helyi szinten is elismert gimnázium lett, amely a művészeti nevelésben és az idegen nyelvek oktatásában kiemelkedő. Mindezt két évtized alatt érte el, ami rendkívül figyelemre méltó eredmény. 22 év alatt az iskola tanulmányi átlaga 3,4-ről 4,44-re nőtt. A felvételi jelentkezések évről-évre növekvő aránya, a partneri elégedettségi mérések azt mutatják, hogy az egyik legnépszerűbb középiskola ma Debrecenben!

A 2006/2007-es tanévtől csak öt és hat évfolyamos képzést folytatunk: négy párhuzamos osztály közül egy hatosztályos gimnáziumi osztály, egy nyelvi előkészítővel induló emelt szintű nyelvi, egy magyar-angol két tanítási nyelvű és egy nyelvi előkészítővel induló emelt szintű művészeti (dráma, ének-zene) osztály. Képzési rendszerünk sokszínűségével diákjainknak igyekszünk lehetőséget teremteni arra, hogy képességeiknek és érdeklődésüknek megfelelő utat járjanak végig tanulmányaik során.

Az idegen nyelvek gyakorlását és a nemzetközi kapcsolatok bővítését teszik lehetővé az iskola által szervezett külföldi kirándulások, a diákfesztiválok, a diákfesztyvályok, pályázatok, utazások.

Az iskola a fakultációs lehetőségek gazdag választékát kínálja a tanulóknak, melyek elsősorban a továbbtanulásra történő felkészítést célozzák. Ezek kínálnak lehetőséget a közép- és emelt szintű érettségi felkészítéshez. Az érdeklődés szerint szerveződő tanórán kívüli foglalkozásokkal (szakkörök, ECDL felkészítő, énekkar, idegen nyelvi szakkörök stb.) és a keret-tanterv biztosította szabadon választható tantárgyakkal kiegészítve biztosítjuk a tanulók számára a tehetségük gondozásának lehetőségét.

A drámások és énekesek közül többen kapnak kisebb-nagyobb szerepet a Csokonai Színház egy-egy előadásában vagy filmekben, televíziókban. Részt vesznek a város és megye kulturális rendezvényein; számos fellépési lehetőségük van az Adyn kívül is.

Iskolánk életét a ma már hagyományos és kedvelt rendezvények teszik színessé. Változatos és gazdag sportolási lehetőségeket kínálunk a diákoknak és a tanároknak is.

A gimnáziumunkban 2007. október 1-jén 748 diák tanult. A lányok száma 479. Hét tanulónk jelenleg külföldön tanul. Debreceni 534, kollégista 90, bejáró 111 diák.

Az iskola eredményességét jelzi, hogy a különböző versenyeken, vetélkedőkön tanítványaink számos előkelő helyezést értek, érnek el. Kiemelkedő tantárgyi eredményeink az elmúlt években olaszból és franciából voltak. A Színművészeti Egyetemet, művészeti főiskolákat végző és már elvégzett diákjaink közül többen ma már országosan is ismert művészek, televíziós, rádiós személyiségek. (A teljesség igénye nélkül: Kálloy-Molnár Péter, Szabados Mihály, Hajdú Szabolcs, Schell Judit, Tóth Anita, Gubás Gabi, Major Melinda, Botos Éva, Haffner Anikó, Hajdu István, Fésűs Nelli és sokan mások.)

A Debrecen Város Közoktatásáért díjjal kitüntetett iskolánkban a 2007/2008-as tanévben az óraadókkal együtt 67 pedagógus tanít. A szakos ellátottság jó. A tanárok jelentős része rendelkezik informatikai ismeretekkel, többen közülük informatikai továbbképzéseken vettek részt. A főállású tanárok mellett több óraadó tanárt foglalkoztatunk a művészetek és a nyelvek oktatására. Az angol és német nyelv tanítását anyanyelvi lektorok segítik. A művészeti nevelésben, képzésben a Csokonai Színház művészei is részt vállalnak.

A szakmai munkaközösségek között kiemelt szerepe van az informatika-könyvtár munkaközösségnek. Munkaközösségünk igyekszik a tanítandó tartalmat és követelményeket frissen tartani, és egyik fő célkitűzése, hogy mind tanulóink, mind tanáraink munkaeszköz-

ként használják a rendelkezésükre álló informatikai eszközöket. Tapasztalataim szerint ez utóbbi törekvésünk a tanárok, és a tanulók körében is megfelelően érvényesül.

A pedagógusok közül tízen látnak el vezető tanári feladatot a Debreceni Egyetem Tanárképzési Kollégiumának megbízásából. Többen óraadók az egyetemen, illetve a főiskolán. Két szaktanácsadónk, két tanácsosunk és egy főtanácsosunk van. Számos kollégánk felvételt nyert az országos szakértői és vizsgáztatói (érettségi elnöki) névjegyzékre. Többen jelenleg is tanulnak szakvizsgás illetve doktori képzésben. Számosan vállalnak szakértői megbízást, tankönyvbírálatot, vannak tankönyvszerzőink is. A nevelőtestület szívesen vesz részt újító, kezdeményező feladatokban.

A HEFOP 3.1.3 „Felkészítés a kompetencia alapú oktatásra” című európai uniós pályázaton nyertünk, így modern eszközök és korszerű módszertani továbbképzések, tananyagok segítik tanáraink munkáját. Az OFI fejlesztő feladatainak kipróbálására is vállalkozunk. Számos újítást vezettünk be képzésünkben. Az ún. komplex összefoglaló órák, vizsgák, pódium órák, mestermunka foglalkozások mind pedagógusainkat dicsérik.

Az oktatáshoz szükséges legfontosabb tárgyi feltételek biztosítottak. Könyvtár, számítógépes terem, tornaterem, konditerem, aerobic terem, pódiumterem, gyakorló terem a pincében állnak rendelkezésünkre. A teljes épületben biztosítjuk az internet-hozzáférést (vezetékes vagy vezeték nélküli formában). 10 db számítógép található a tanári dolgozóknak. A két informatika tanteremben összesen 32 darab tanulói számítógép található. Ezen kívül 9 darab hordozható számítógép áll a tanárok rendelkezésére. Iskolánkban 10 darab projektor található. Ezek közül 5 darab fixen szerelt, 4 darab „digitális zsúrkocsi” tartozéka. 5 darab interaktív táblánk közül 2 darab fixen szerelt. A könyvtárban, illetve az aulában szabadon használható számítógépeket helyeztünk el.

2.3.2. Az IKT kompetenciaterület fejlesztése

Az NFT³ összefoglalóját 2003 decemberében jelentette meg a Miniszterelnöki Hivatal. A terv öt operatív programot tartalmaz, melyek közül a humán erőforrások fejlesztése (HEFOP) tartalmaz erre vonatkozó konkrét programokat. A HEFOP négy prioritásra helyezi a hangsúlyt. Az oktatással különösen az élethosszig tartó tanulás és az alkalmazkodóképesség támogatása, valamint az oktatási, szociális és egészségügyi infrastruktúra fejlesztése foglalko-

³ Nemzeti Fejlesztési Terv

zik. 2005-ben jelent meg a „Felkészítés a kompetencia-alapú oktatásra” HEFOP/2005/3.1.3 pályázati felhívás.

A program célkitűzése a pedagógiai kultúra átalakulása. A ma jellemző frontális és zömmel lexikális ismeretközlő eljárásokat fokozatosan fel kell váltania az iskolarendszer egészére kiterjedő, kompetencia alapú nevelési, oktatási módszertannak. A módszertani változást a digitális kultúra oktatási eszközrendszerének alkalmazása kell, hogy kísérelje.

A pályázati kiírást elolvasva éreztük, hogy a kompetencia alapú oktatás elterjesztésében rendkívül fontos szerep hárul majd az informatika tanárokra. Nap, mint nap érezzük, hogy a digitális kompetencia hiánya, vagy csak alacsony szintje leküzdhetetlen hátrányokat okoz a hétköznapi életben. Igaz ez diákjainkra, de talán még inkább tanárainkra. Pályázatunk megírásakor ezért kiemelt területként kezeltük az IKT kompetenciaterületet.

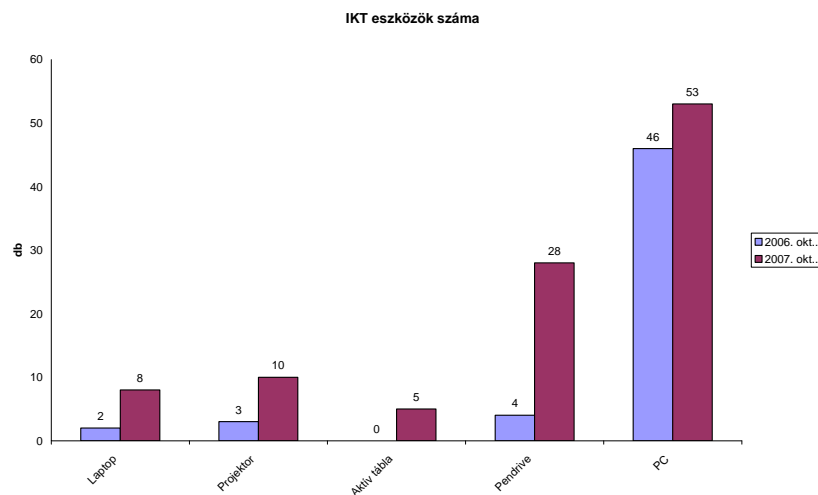
Nyertes intézményként lehetőségünk nyílt két kötelező (szövegértés-szövegalkotás, matematikai logikai) és két szabadon választott (IKT, idegen-nyelvi) kompetenciaterületen összesen 7 pedagógus 120 órányi, moduláris felépítésű módszertani továbbképzésen való részvételére. Ezek a képzések a kompetencia alapú, digitális oktatási tartalmakat használó nevelési, oktatási programcsomagok bevezetését, valamint digitális taneszközök alkalmazását segítették. A felkészítés után a megszerzett új eljárások, pedagógiai eszközrendszer adaptálása, alkalmazása, a saját pedagógiai programba történő beillesztése és intézményen belüli elterjesztése volt a feladatunk.

A pályázat megvalósításához megfelelő technikai háttérre is szükség volt. Az elnyert 18 millió forint támogatásból 5,4 millió forint állt rendelkezésre a hiányzó, minimális feltételként szabott eszközök beszerzéséhez. Fixen szerelt és hordozható digitális táblákat, hordozható és asztali számítógépeket vásárolhattunk. Kialakítottuk az egész iskola területén a vezeték nélküli számítógépes hálózatot.

A pályázat megvalósításának csúszása miatt hosszú hónapok teltek el, mire kiderült, hogy az IKT kompetenciaterület jelentősen eltér a többi kompetencia területtől. A kompetencia területek programcsomagjainak kidolgozásáért és gondozásáért (tanári kézikönyvek, tankönyvek, munkafüzetek, továbbképzések, mentorok) a Sulinova Kht. volt a felelős. Az IKT-A modul tananyagcsomagjainak kidolgozása (több más feladat mellett) az Educatio Kht. feladata volt. Kiderült, hogy nem készültek nyomtatott segédanyagok. Az IKT programcsomag nem egy kinyomtatott füzet, hanem letölthető pedagógiai dokumentáció, digitális tananyagok, segédletek sokasága az ikt.sulinet.hu oldalon, illetve az SDT-ben.

A pályázat megírásakor részt vettem az új eszközök beszerzésének tervezésében. Mérlegelni kellett, hogy a meglévő eszközeink hogyan integrálhatók az új rendszerben, milyen és hány darab új eszközre lesz szükségünk. Az új és régi eszközök rendszerbe illesztése is fontos feladat volt. Úgy kellett elhelyezni az eszközöket az iskolában, hogy minden tanár számára egyszerűen használhatók legyenek, ugyanakkor a vagyónvédelemi szempontoknak is maximsan megfeleljünk. Alapos tervező munka után 5 teremben a projektorok fix rögzítéssel lettek elhelyezve.

A meglévő 2 db „digitális zsúrkocsi” mellé saját tervezésű, szinteken belül mozgatható kompakt multimédiás szekrényt terveztünk, mely mobil interaktív táblát is tartalmaz. A pályázat eredményeként, illetve az iskolai taneszköz-beszerzésre fordítható összeg célzott felhasználásával ugrásszerűen fejlődött iskolánk digitális eszközökkel való ellátottsága. Az egy év alatt bekövetkezett gyarapodást az alábbi diagram mutatja.



1. ábra: ITK eszközök száma

Külön gondot jelentett a sok eszköz optimális kihasználtságának megszervezése. Együttgondolkodásunk eredményeképp elkészítettem honlapunkról elérhető PHP alapú zsúrkocsi igénylő oldalunkat. Használatával könnyen átlátható, jól szervezhető, statisztikailag elemezhető az eszközhasználat az iskolában.

3. GNU

A **GNU** betűszó azt jelenti, hogy a **Gnu's Not Unix**, és egy teljes szoftverrendszer neve, amit bárki ingyen használhat. A projektet **Richard Stallman** indította 1984-ben. 1985-ben fogalmazta meg céljait a **GNU-kiáltványban**, melyet bizonyos gyakran előforduló félreértések miatt 1993-ban lábjegyzetekkel egészítettek ki. Céljai megvalósításához létrehozta a **Free Software Foundationt (FSF)**, amely általában a szabad szoftverek terjesztését, fejlesztését, és e célok eléréséhez szükséges források megteremtését tűzte ki célul. Sok önkéntes segítette a projektet idővel, pénzzel, programokkal és felszereléssel. 2003 januárjában megalakult a „**Free Software Foundation Hungary Alapítvány**”, amely a fenti célok mellett a szabad szoftverek honosítását is megpróbálja segíteni. „A GNU nem public domain. BÁRKI módosíthatja és terjesztheti a GNU-t, DE SENKI SEM TILTHATJA MEG A TOVÁBBI TERJESZTÉST. Azaz szabadalmaztatott módosítások nem készülhetnek. Biztos akarok lenni abban, hogy a GNU összes verziója szabadon felhasználható marad. ... A GNU képes lesz Unix-programok futtatására, de nem lesz a Unix-szal azonos. Minden olyan továbbfejlesztést átveszünk más ismert operációs rendszerekből, ami a rendszer használatát kényelmesebbé teheti. Főként azt tervezzük, hogy hosszabb fájlneveink, fájlverziószámaink, hibatűró fájlrendszerünk, esetleg fájlnev-kiegészítésünk, terminálfüggetlen képernyőkezelésünk, és talán majd egy Lisp alapú ablakrendszerünk is lesz, ...” [3]

A szabad szoftverek használatát is szabályozni kell a felhasználási, a módosítási feltételek, a rá épülő üzleti szolgáltatások, valamint a használat során esetleg bekövetkező károk jogi következményeire vonatkozó szabályok megfogalmazásával. Ezt a feladatot hivatott ellátni a **GNU GPL**, a **GNU General Public License**. „A legtöbb szoftver licencei azzal a szándékkal készültek, hogy minél kevesebb lehetőséget adjanak a szoftver megváltoztatására és terjesztésére. Ezzel szemben a **GNU GPL** célja, hogy garantálja a szabad szoftver másolásának és terjesztésének szabadságát, ezáltal biztosítva a szoftver szabad felhasználhatóságát minden felhasználó számára. A **GPL** szabályai vonatkoznak a **Free Software Foundation** legtöbb szoftverére, illetve minden olyan programra, melynek szerzője úgy dönt, hogy ezt használja a szerzői jog megjelölésekor. (A Free Software Foundation egyes szoftvereire a GNU LGPL érvényes.) Bárki használhatja a programjaiban a GPL-t a szerzői jogi megjegyzésnél.

A szabad szoftver megjelölés nem jelenti azt, hogy a szoftvernek nem lehet ára. A GPL licencek célja, hogy garantálja a szabad szoftver másolatainak szabad terjesztését (és e szolgáltatásért akár díj felszámítását), a forráskód elérhetőségét, hogy bárki szabadon módosíthassa a szoftvert, vagy felhasználhassa a részeit új szabad programokban; és hogy mások megismerhessék ezt a lehetőséget.

A szerző jogainak védelmében korlátozásokat kell hozni, amelyek megtiltják, hogy bárki megtagadhassa ezeket a jogokat másoktól, vagy ezekről való lemondásra kényszerítsen bárki mást. Ezek a megszorítások bizonyos felelőségeket jelentenek azok számára, akik a szoftver másolatait terjesztik, vagy módosítják.” [4]

4. Az alkalmazott szoftverek

A dolgozatban leírt alkalmazás szabad szoftverek felhasználásával futtatható, ehhez bármilyen operációs rendszer megfelelő, amelyen fut egy olyan Web-szerver, amely **PHP** kódot képes értelmezni, és az adatok tárolását pedig szintén szabad szoftver végzi. A gyakran használt négyes a **LAMP**-ként rövidített **Linux**, **Apache**, **MySQL** és **PHP** programok segítségével valósítottam meg az előző fejezetben kitűzött célokat.

4.1. Linux

A **Linux** egy operációs rendszer, a szabad szoftverek és a nyílt forráskódú programok egyik legismertebb példája. A „Linux” elnevezés szigorúan véve a **Linux** kernelt jelenti, a köznyelvben mégis gyakran alkalmazzák teljes **Unix**-szerű operációs rendszerre, amely a **Linux** kernelre, valamint a **GNU** könyvtárakra és eszközökre épül. Ez utóbbi hivatalos neve **GNU/Linux**. A **Linux**ot 1991-ben kezdte fejleszteni Intel **386 AT** klónokra **Linus Torvalds** akkor másodéves finn egyetemista. Mivel **Stallman GNU/Hurd** projektje nem készült el, így a **GNU** nem rendelkezett saját operációs rendszerrel, ezt a szerepet legtöbb esetben a sokak által támogatott **Linux** látja el.

A különböző **Linux** terjesztések (disztribúciók) olyan összeállítások, melyek egy felhasználásra kész **GNU/Linux** alapú operációs rendszert, és ahhoz tartozó, válogatott programokat tartalmaznak. A disztribúciók között az alábbi főbb szempontok alapján tehetünk különbséget:

- a célközönség: a legtöbb terjesztés adott közönséget céloz meg: profi vagy kezdő felhasználókat, tanulókat, adminisztrátorokat, kevés memóriával rendelkező, vagy csak CD-t tartalmazó gépeket stb. Sok disztribúció speciális hardverekre készült (routerek, mobiltelefonok stb.) Felépítésük, támogatásuk, elemeik mind a megcélzott felhasználási területtől függenek.
- a fájlrendszer felépítése: a disztribúciók egységes, de gyakran egymástól eltérő könyvtárstruktúrával rendelkeznek
- csomagkezelés: a terjesztések egyik fő jellemzője az egyes programcsomagok telepítésének, eltávolításának és frissítésének megkönnyítése és támogatása (**APT**, **RPM**)

- biztonsági és egyéb frissítések (a legtöbb disztribúció készítői azonban komolyan veszik a biztonsági problémákat, és az ismert hibák javításait rövid időn belül elérhetővé teszik).

Én a **Debian** választottam. Ez elég elterjedt, nagy felhasználói táborral rendelkező terjesztés, és sok más disztribúció alapja. A GNU alapelveit hűen követi, felhasználói táborának szellemében alkot közösséget, old meg rövid idő alatt problémákat. Fejlett csomagkezelési rendszere van (**APT**) és gyorsan reagál a szoftverhibákból származó biztonsági problémákra. A telepítés és a beállítások leírásához a **Debian GNU/Linux 4.0 r1 "Etch"** kiadást használom. A telepítést internethez kapcsolódó gép esetén célszerű **Netinst CD**-ről végezni, így csak minimális anyagot kell letölteni.

4.2. Apache

A név egy szójáték: "a patchy server", ami arra utal, hogy a szervert eleinte a webmesterek által készített patchekkel frissítették. Az **Apache** az **NCSA** Web-szerver toldozgatásából fejlődött ki. Az **NCSA** egyike volt az első Web-szervereknek, amelyet az Illinois-i egyetemen található **NCSA** fejlesztett ki. Az **Apache** projektet webmesterek, rendszergazdák kezdték el, amikor **Rob McCool**, az **NCSA** (National Center for Supercomputing Applications) Web-szerverének írója 1994 közepén kilépett az **NCSA**-tól, és a szoftver nem fejlődött tovább. Az **Apache** a legelterjedtebb Web-szerver, amely megfelel a gyorsan változó internet követelményeinek, biztonságos, üzleti, vállalati felhasználásra is megfelelő és szabadon használható. Funkciói bővíthetők beépülő modulok betöltésével.

4.3. MySQL

A **MySQL** egy többfelhasználós, **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 a céget. Az **MySQL** az egyik legelterjedtebb adatbázis-kezelő, aminek egyik oka lehet, hogy a teljesen nyílt forráskódú **LAMP** összeállítás részeként költséghatékony és egyszerűen beállítható megoldást ad dinamikus webhelyek szolgáltatására. A **MySQL** adatbázisok adminisztrációjára a mellékelt parancssori eszközöket (**mysql** és **mysqladmin**) használhatjuk. A **MySQL** honlapjáról grafikus felületű adminisztráló eszközök

is letölthetők: **MySQL Administrator** és **MySQL Query Browser**. Széles körben elterjedt és népszerű alternatíva a **PHP** nyelven írt, nyitott forráskódú **phpMyAdmin**. A **phpMyBackupPro** (amelyet szintén **PHP**-ban írtak) adatbázisok (akár időzített, ismétlődő) mentésére szolgál eszközzel.

4.4. PHP

A **PHP (Hypertext Preprocessor)** egy nyílt forráskódú, számítógépes szkriptnyelv. Legfőbb felhasználási területe a dinamikus weboldalak készítése. Emiatt a **PHP**-t jórészt szerveroldalon használják, bár létezik parancssori interfésze is, illetve önálló, grafikus felületű alkalmazások is létrehozhatóak vele. A nyelvet eredetileg **Rasmus Lerdorf** alkotta meg 1994-ben, de a ma létező egyetlen **PHP** implementációt már a **PHP Group** tartja karban és fejleszti. A **PHP** a saját licensze alatt kerül kiadásra, de a **Free Software Foundation** szabad szoftverként tartja számon. A **PHP** a legtöbb Web-szerverre, operációs rendszerre és platformra ingyenesen telepíthető. A **PHP** az **Apache** Web-szerver egyik legnépszerűbb beépülő modulja.

A **PHP** oldalak elkészítésénél a **HTML**-t gyakorlatilag csak mint formázást használjuk, ugyanis e 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, így a programkód nem is látható kliens oldalról. Ehhez egy 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 csak igen korlátozottan. Ilyen például a bejelentkezés, az adatbázis-kezelés, filekezelés, kódolás, adategyeztetés, kapcsolatok létrehozása, e-mail küldése, adatfeldolgozás, dinamikus listakészítés. Minden olyan esetben, ahol nagyszámú ismétlődő feladatsort kell végrehajtani, ott ez a programnyelv nagyszerű segítség.

Szintaktikailag a nyelv hasonlít a **C** nyelvre, de nem kell az utasítások típusával törődni (néhány speciális eset kivételével), egyébként az egész nyelv egyszerű. A **PHP** utasításokat mindig egy meghatározott karaktersorral kell kezdeni és bezárni. Ez lehet a `<? és a ?>`, de ez különösen ellenjavallt, ugyanis az **XML** dokumentumok nyitó és záró tagja is hasonló, így egy beágyazott **XML** dokumentumot is forráskódnak értelmezne az interpreter. Sok hibát kizárhatunk, ha programjainkban a `<?php` nyitó és a `?>` záró karaktersorokat használjuk,

ugyanis az érvényes nyitó és záró elemek a `php.ini` konfigurációs fájlban vannak deklarálva, amik szerverenként eltérhetnek egymástól, így előfordulhat, hogy egyes szerveren nem értelmezi az interpreter a **PHP** programunkat, hanem közvetlenül a kimenetre küldi a forráskódot.

Minden változót és összetett adatszerkezetet a `$` jellel kell kezdünk. Kivételt képeznek ez alól a konstansok. A tömb indexelése – hasonlóan más **C** alapú nyelvekhez – 0-tól kezdődik. A legegyszerűbb utasítás az `echo`, vagy az ezzel egyenértékű `print`. Ennek segítségével lehet kihasználni igazán a **PHP**-t: közvetlenül lehet írni a készülő **HTML** dokumentumba, akár formázott szöveget is. Az utasításokat pontosvessző (;) zárja. A vezérlési szerkezetek szintaktikája is teljesen **C**-szerű.

5. A rendszer telepítése

A következő lépéseket kell végrehajtani a telepítés során:

- Először letölteni a „Debian GNU/Linux 4.0 r1 "Etch" - Official i386 NETINST Binary-1 20070820-20:21” címkéjű CD lemezképet, amely mérete alig több mint 160 MB, majd felírni egy CD-re.
- Az alkalmazás futtatására kiszemelt számítógépet erről CD-ről kell elindítani.
- A telepítési folyamat a telepítési mód kiválasztásával kezdődhet, ezek listája az F3-as gomb leütése után tekinthető meg, de a legegyszerűbb ENTER leütésével elfogadni az alapértelmezett telepítési módot.
- Az első megjelenő listából nyelvet kiválasztható telepítő nyelve. A Hungarian kiválasztása után magyarul kommunikál a telepítő és már a magyar billentyűzet-kiosztást ajánlja fel a következő listában.
- Az ezt követő lépésben a telepítő megpróbálja DHCP-n keresztül beállítani a hálózati kapcsolatot. Ha ez nem sikerül, akkor kézzel kell megadni a hálózati címet, maszkot, átjárót és a DNS szerver címét. Ezután megadhatjuk a számítógép nevét, majd azt a tartományt, amelyhez gépünk tartozik.
- A merevlemez particionálása következik. Legegyszerűbb, az „Irányított – az egész lemez felhasználása” lehetőséget választani.
- A root jelszó beállítása során törekedni kell arra, hogy az elég biztonságos legyen.
- Egy új felhasználó létrehozása. Mindenképpen célszerű egy nem rendszergazdai jogokkal rendelkező felhasználót megadni. Elég veszélyes állandóan teljhatalommal használni a rendszert.
- Ezután az alaprendszer telepítése következik, amely nem igényel beavatkozást.
- Mivel igencsak alaprendszert kapunk az előző rész lefutása után a most következő kérdésre mindenképpen igennel kell válaszolni. Hálózati tükör beállítása következik. Megadva, hogy Magyarországon vagyunk, a listából tetszőlegesen választhatunk. Én a ftp.hu.debian.org-ot választottam tükörszervernek. A tükörszerver átvizsgálásának befejezésére elég sokáig kell várni.
- A csomaghasználat felmérésében nem szükséges részt venni.
- Szoftver kiválasztásnál nem kell kiválasztani egy lehetőséget sem.
- A GRUB rendszerbetöltőt a fő boot rekordba telepítettem.

- Ezután rendszer újraindítása következik.
- Root-ként bejelentkezve a dselect alkalmazás segítségével telepíthetjük a még szükséges csomagokat függőségeikkel együtt. Mielőtt a csomagok kiválasztásához kezdénénk, a frissítés menüponttal aktualizálni kell a telepíthető csomagok listáját. A csomagok: apache2, mysql-server-5.0, libapache2-mod-php5, php5; php5-mysql.

6. Röviden a dselect használatáról

A dselect a debian alapértelmezett telepítő programja. A dselect igazából egy előtét programja a dpkg-nak. Segítségével lehet a csomagokat feltelepíteni úgy, hogy a dselect elintézi a függőségeket.

```
debian:~# dselect
```

beírására többek közt valami ilyesmi látható:

0. [E]lérés Válassz elérési módot.

1. [F]rissítés Frissíti az elérhető csomagok listáját, ha lehetséges.

2. [V]álaszt Csomagok választása.

3. [T]elepít Kért csomagok telepítése, frissítése.

4. [B]eállít Beállítatlan csomagok beállítása.

5. [L]eszedés Nem kívánt szoftverek eltávolítása.

6. [K]ilépés Kilépés a dselect programból.

Kiválasztva a fenti menüpontokat a következő lehetőségekhez jutunk.

6.1. 0. [E]lérés Válassz elérési módot.

Dönthetünk arról, honnan szeretnénk a csomagok információit beolvasatni a dselect-el. Erősen javasolt az alapértelmezett beállítás megtartása. A csomaglisták helye az `/etc/apt/sources.list` fájlban tárolódik.

Ha azt akarjuk, hogy aktualizált legyen a telepíthető csomagok listája a Frissítés-t kell kiválasztani.

6.2. 1. [F]rissítés. Frissíti az elérhető csomagok listáját, ha lehetséges.

Ha minden jól ment a frissítés végén kiírja, hogy hány csomag információit olvasta be összesen. Ezután, ha valamit telepíteni szeretnénk, vagy csak kíváncsiak vagyunk rá, hogy tegnap óta milyen új csomagok kerültek bele a distribúcióba, ki kell választani a következő menüpontot.

6.3. 2. [V]álaszt Csomagok választása.

Ekkor kapunk egy help képernyőt a dselect csomagkezelés mikéntjéről. A szököz billentyű lenyomása után kapjuk a csomaglistát.

A megjelenő lista első oszlopa tájékoztat a csomagok állapotáról. Az első "*" azt jelzi, hogy a program telepítve van a rendszeren. a második "*" azt, hogy milyen állapotváltozásra van jelenleg kijelölve, amit még nem hajtott végre a dselect (pld. Kijelölés eltávolításra, de még az "install" menüpont nem került végrehajtásra egy korábbi kiválasztás során), az utolsó "*" pedig a jelenleg alkalmazni kívánt műveletet/állapotot mutatja. Például, ha *** a program állapota, az azt jelenti, hogy telepítve van a rendszeren és korábban sem és most sem jelöltünk ki műveletet a csomagra. Ha például a "-" billentyűvel kijelöljük törlésre, akkor az állapotjelző így alakul: *- mutatja, hogy jelenleg eltávolításra van kijelölve, de még fent van a rendszeren. Ha a "-" helyett a "_" -t használunk a törlés kijelölésére, az purge (tisztá) törlést eredményez. Tehát az első esetben a config fájlok a helyükön maradnak, azt nem szedi le a dselect, a második esetben mindent leszed, amit felrakott. (ez természetesen nem vonatkozik a program használatából adódóan a \$HOME-ban lévő config fájlokra)

A „/” jel beírása után az utolsó sorba beírhatjuk a keresett csomagok nevét. A fentebb felsorolt csomagokat sorban keressük ki, és a „+” jel leütésével jelöljük telepítésre. Ha a csomag függ olyan csomagoktól, amelyek még nincsenek telepítve egy információs ablak megjelenítése következik, amin szöközzel túllépve láthatjuk azoknak a csomagoknak a listáját melyek szükségesek az általunk választott csomag működéséhez. Azok a csomagok, melyek használat csak javasolt nincsenek automatikusan telepítésre jelölve, azokat nekünk kell kézzel a „+” jellel kiválasztanunk. A felajánlott listát egy ENTER leütésével fogadhatjuk el. Ha az összes szükséges csomagot telepítettük egy újabb ENTER leütésével léphetünk ki a főmenübe.

6.4. 3. [T]elepít Kért csomagok telepítése, frissítése.

A csomagok tényleges telepítése a menüpont kiválasztásával történik. A telepítés során szükség van a telepítő CD-re is, de nagyjából az internetről töltődnek le a szükséges állományok. A folyamat befejezése után kiléphetünk a dselect-ből. A telepítés során a csomagok alapbeállítása megtörténik, működőképese a programok, sőt a szerverek el is indultak.

7. Az adatbázis beállítása

Hozzunk létre az alapértelmezés szerint jelszó nélküli root felhasználó számára egy jelszót a **MySQL**-ben.

```
debian:~# mysqladmin -u root password „rootjelszó”.
```

Ez a root független a rendszer teljhatalmú urától, és nem célszerű ugyanazt a jelszót használni mindkettőnél.

Ezután hozzunk létre egy **zsur** nevű adatbázist. Ehhez már kérni fogja a root felhasználó jelszavát.

```
debian:~# mysqladmin -u root -p create zsur
```

Majd jelentkezzünk be a **MySQL** konzoljára

```
debian:~# mysql -u root -p
```

A „rootjelszó” megadása után a prompt mysql> alakúra változik. Készítsünk egy, mondjuk zsuruser nevű felhasználót, állítsuk be a jogait a zsur adatbázison és kapjon egy jelszót is. Ezt a GRANT sql utasítással tehetjük meg.

```
mysql> GRANT ALL ON zsur.* TO 'zsuruser'@'localhost' IDENTIFIED  
BY 'zsurjelszo';
```

a következő érvényesíti a változásokat

```
mysql> FLUSH PRIVILEGES;
```

Ezután készen állunk az adatbázis feltöltésére. A szükséges táblák leírása a dolgozathoz tartozó CD **zsur** könyvtárának **zsur.sql** állományában található. Mivel az alapadatok módosítását, új tanár, új osztály, stb. felvitelét az alkalmazás még nem kezeli, ezért ennek az állománynak a módosításával célszerű kezdeti állapotba hozni a táblákat. Tegyük meghajtóba a lemezt, majd csatoljuk a fájlrendszerhez.

```
debian:~# mount /cdrom
```

majd készítsünk egy másolatot a szerverre

```
debian:~# cp /cdrom/zsur/zsur.sql ~
```

ezt követően módosítsuk kedvenc editorunkkal az állományt, és adjuk ki a következő parancsot

```
debian:~# cat ~zsur.sql|mysql -u zsuruser -p zsur
```

Ha nem kaptunk hibaüzenetet, akkor valószínűleg minden rendben történt. A művelet eredményéről meggyőződhetünk az adatbázisban található táblák listájának kiíratásával. Ehhez lépünk be a mysql konzolra, majd írassuk ki a szükséges információt!

```
debian:~# mysql -zsuruser -p zsur
```

```
mysql> show tables;
```

```
+-----+
| Tables_in_zsur |
+-----+
| csrend          |
| igény          |
| kocsi          |
| oszt           |
| tanar          |
| tant           |
| terem          |
+-----+
```

Esetleg érdemes belenézni valamelyik tábla tartalmába is a

```
mysql> select * from tanar;
```

parancs kiadásával.

id	nev	azon	pass	aktiv
1	Bara Jolán	baraj	461c4c1faecfd3ad98a217bcc13eba3b	1
2	Bécsi Endre	becsie	461c4c1faecfd3ad98a217bcc13eba3b	1
3	Belky Péter	péter	461c4c1faecfd3ad98a217bcc13eba3b	1
4	Borsi Katalin	borsika	461c4c1faecfd3ad98a217bcc13eba3b	1

8. A Web-szerver beállítása

Az Apache2 Web-szerver alapértelmezés szerint a /var/www könyvtárat használja DokumentRoot-ként. Ide kell átmásolnunk a CD **zsur** könyvtárát.

Létrehozzunk egy könyvtárat a

```
debian:~# mkdir /var/www/zsur
```

parancsal, majd beállítjuk a hozzáférési jogokat:

```
debian:~# chmod 755 /var/www/zsur
```

Ezután át kell másolni a CD-ről a zsur könyvtár tartalmát:

```
debian:~# cp /cdrom/zsur/*.* /var/www/zsur és a
```

```
debian:~# chmod -R 644 /var/www/zsur/*.*
```

parancsal ezen állományok hozzáférési jogait is beállítjuk.

Az alkalmazás működőképességéhez már csak az adatbázis elérését kell biztosítanunk. Ehhez hozzunk létre egy állományt a /var könyvtárban **conf.inc** néven

```
debian:~# touch /var/conf.inc
```

majd állítsuk be tulajdonosnak azt a felhasználót, amelyet a Web-szerver használ. Ez a debian esetében alapesetben a **www-data** user:

```
debian:~# chown www-data /var/conf.inc
```

és elérési jogai

```
debian:~# chmod 400 /var/conf.inc
```

A fenti állományban fogjuk tárolni az adatbázis elérésére jogosult felhasználó nevét és a hozzá tartozó jelszót. Az állomány tartalma legyen a következő:

```
<?php
$db_user='zsuruser';
$password='zsurjelszo';
?>
```

Mielőtt kipróbálnánk milyen lehetőségeket nyújt a rendszer, még egy beállítást célszerű megtenni.

Az **/etc/php5/apache2** könyvtárban lévő **php.ini** állományban keressük ki a hibaüzenetek típusát meghatározó paramétert

```
error_reporting= E_ALL & ~E_NOTICE
```

és egészítsük ki:

```
error_reporting= E_ALL & ~E_NOTICE & ~E_WARNING
```

A változtatás érvényre juttatásához indítsuk újra a Web-szerveret!

```
debian:~# /etc/init.d/apache2 restart
```

Ezzel a telepítési folyamat befejeződött. Kipróbálhatjuk az alkalmazást.

9. Az alkalmazás használata

9.1. Nyitóoldal

Az alkalmazás induló oldala lehetőséget biztosít a kollégáknak, hogy bejelentkezés nélkül tájékozódjanak egy eszköz heti foglaltságáról. Amennyiben eszközt akar igényelni, a **Bejelentkezés** gomb megnyomásával megetheti. Az oldalon mindig egy aktuális eszköz adatai, tulajdonságai láthatóak, melyet a **Váltás:** szöveg alatti gombok egyikével lehet kiválasztani. Az, hogy éppen ki foglalt le egy eszközt adott időpontban órarendszerűen kialakított táblázatból olvasható le. A foglaltságot jelölő gombok megnyomásának hatására a táblázat alatt megjelennek az adott óra adatai. Az oldal legalján lehetőség van az aznapi, illetve azt megelőző és követő napok adatainak megnézésére, egy minden eszközt tartalmazó táblázatban.

"Zsúrkocsikat" kezelő oldal

Bejelentkezés

Aktuális "zsúrkocsi" : Z/1 kicsi
Tartalmaz: laptopot (DVD/CD olvasó, USB csatlakozó, hangkimenet van, floppy nincs), projektort, hangszórók nincsenek hozzá, de csatlakoztathatók. Alkalmas: Képek, Office(Word, Excel, PowerPoint) és Pdf dokumentumok megjelenítésére.

Váltás:

Z/1 kicsi E/4 nagy 2/4 hefp1 1/6 hefp2 Z/1 hefp3 1/1 2/5 fiz hefp4 kémia

Betöltve: 2008-04-05 10:04:38.

Dátum	2008-04-05 szombat	2008-04-06 vasárnap	2008-04-07 hétfő	2008-04-08 kedd	2008-04-09 szerda	2008-04-10 csütörtök	2008-04-11 péntek	2008-04-12 szombat
1. óra							Belinszky	
2. óra							Belinszky	
3. óra			Bécsi				Belinszky	
4. óra							Belinszky	
5. óra							Belinszky	
6. óra				Lakatosné		Bécsi	Belinszky	
7. óra					Lakatosné		Belinszky	
Du.								

"Zsúrkocsi":	Z/1 kicsi	Dátum:	2008-04-07	Tanóra:	3.
Osztály:	12. C	Tantárgy:	matematika	Tanterem:	2/2a
Kezdés:	09:35	Befejezés:	10:20	Tanár:	Bécsi Orsolya

Az összes eszköz foglaltsága megtekinthető együtt (a mai vagy holnapi napra). [Ide kattints!](#)

Kész

2. ábra: Nyitóoldal

9.2. Bejelentkezés

Az oldalra bejelentkezni azonosító és jelszó megadása után lehet. Innen érhető el az ezek módosítását lehetővé tevő oldal az „Azonosító beállítása, módosítása. Jelszó megváltoztatása.” szövegre való kattintás után.

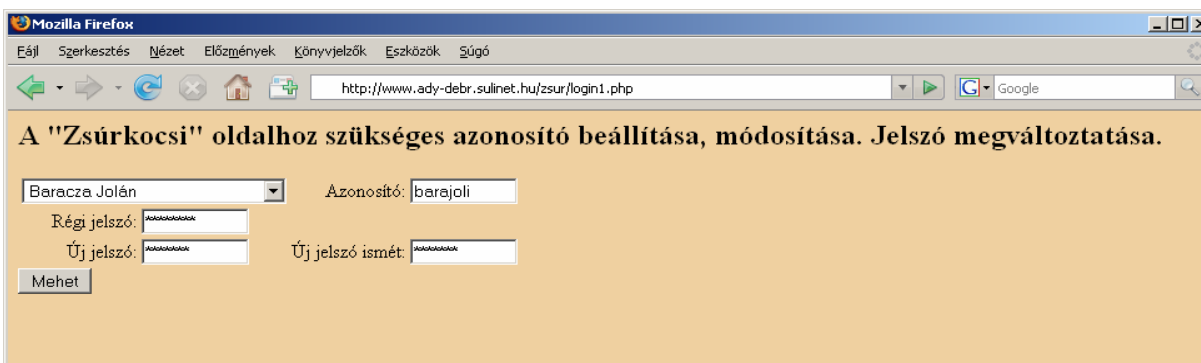


3. ábra: Bejelentkezés

9.3. A jelszó módosítása

Az azonosítást végző adatok módosításához ki kell választania a tanárnak a saját nevét a legördülő listából. Ha az azonosítóját nem akarja megváltoztatni, akkor az **Azonosító:** szöveghez tartozó beviteli mezőt üresen kell hagynia. A régi és az új jelszavakat helyesen beírva történik a jelszó megváltoztatása. A jelszavakat akkor is meg kell adni, ha csak az azonosítót akarjuk megváltoztatni.

Az azonosítóra és a jelszóra vonatkozóan van néhány szabály, amelyek betartása ellenőrzésre kerül. Az azonosító hossza legalább két, de legfeljebb tíz karakter lehet. A jelszó legalább hat és legfeljebb tíz karakterből állhat. A karakterek csak betűk és számjegyek lehetnek. Sikertelen lesz a módosítási kísérlet abban az esetben is, ha az „Új jelszó:” és az „Új jelszó ismét:” szövegekhez tartozó beviteli mezők tartalma nem egyezik meg.



4. ábra: A jelszó módosítása

9.4. Bejelentkezve...

Az oldalra való bejelentkezés után az órarendhez hasonló kialakítású táblázat minden cellája olyan nyomógombot tartalmaz, amely a bejelentkezett felhasználó számára az aktuális „zsúrkocsi”-ra vonatkozó, az adott óra kapcsán aktuális lehetőséget teszi elérhetővé.

Az **Új** feliratú zöld színű gomb azt jelzi, hogy lehetősége van a tanárnak az adott órára lefoglalni az aktuális eszközt. A gomb lenyomásakor a táblázat alatt megjelenik egy másik táblázat, amely tartalmazza az óra azon adatait, melyek a kiválasztásból következnek, tehát az eszköz nevét, a dátumot, a tanóra sorszámát, az óra kezdetének és befejezésének időpontját. Az osztály, a tantárgy és a tanterem legördülő listákból választható ki. Az óra címe is megadható harminc karakter hosszban. Ha az órák számát megnöveljük az alapértelmezett egyről, akkor ugyanazokkal az adatokkal több egymást követő órára is lefoglalható az eszköz. Ha az egymást követő órák közül valamelyikben nem szabad az eszköz, akkor annak bejelölése ki marad, és nem keletkezik hiba. A kitöltött adatok adatbázisba való rögzítése a szürke alapú **Lefoglalás** feliratú gombbal történik.

Aktuális "zsúrkocsi" : Z/1 kicsi
Tartalmaz: laptopot (DVD/CD olvasó, USB csatlakozó, hangkimenet van, floppy nincs), projektort, hangszórók nincsenek hozzá, de csatlakoztathatók. Alkalmas: Képek, Office(Word, Excel, PowerPoint) és Pdf dokumentumok megjelenítésére.

Váltás:
Z/1 kicsi | E/4 nagy | 2/4 hefo1 | 1/6 hefo2 | Z/1 hefo3 | 1/1 | 2/5 | fiz hefo4 | kémia

Betöltve: 2008-04-05 10:38:15.

Dátum	2008-04-05 szombat	2008-04-06 vasárnap	2008-04-07 hétfő	2008-04-08 kedd	2008-04-09 szerda	2008-04-10 csütörtök	2008-04-11 péntek	2008-04-12 szombat
1. óra	Új	Új	Új	Új	E/4 nagy	Új	Új	Új
2. óra	Új	Új	Új	Új	Új	Új	Új	Új
3. óra	Új	Új	Bécsi	1/6 hefo2	Új	Új	Új	Új
4. óra	Új	Új	Új	1/6 hefo2	Új	Új	Új	Új
5. óra	Új	Új	Új	Új	Új	Új	Új	Új
6. óra	Új	Új	Új	Töröl	Új	Új	Új	Új
7. óra	Új	Új	Új	Új	Töröl	Új	Új	Új
Du.	Beír	Beír	Beír	Beír	Beír	Beír	Beír	Beír

Form fields:
"Zsúrkocsi": Z/1 kicsi | Dátum: 2008-04-06 | Tanóra: 2008-04-05 10:38:15
Osztály: 7A | Tantárgy: olasz | Tanterem: 1/0
Kezdés: 10:40 | Befejezés: 11:25 | Az óra címe:

Lefoglalás

Az összes eszköz foglaltsága megtekinthető együtt (a mai vagy holnapi napra). [Ide kattints!](#)

5. ábra: Bejelentkezve...

Ha időközben más már készített adott időpontban új bejegyzést, akkor hibaüzenet jelenik meg a művelet sikertelenségéről, és az oldal újbóli betöltésekor az már megjelenik. Sikeres végrehajtás után is frissül az oldal és a megfelelő helyen az **Új** feliratú gomb helyett kék színű, **Töröl** feliratú látható.

A **Töröl** feliratú gomb megnyomására egy a törlendő óra adatait tartalmazó táblázatban ellenőrizhető, hogy a megfelelő órát akarja-e törölni a felhasználó. A művelet a Törölés feliratú gombra való kattintással hajtható végre. Mivel az órák adatainak módosítása másként nincs megoldva, ez a művelet csak az óra bejegyzésének törlésével és újbóli felvételével történhet.

A rózsaszínű, tanári vezetéknevekkel feliratozott gombok által elfoglalt időpontokban az adott tanár már lefoglalta az eszközt. A gomb megnyomására az óra adatait tartalmazó táblázat jelenik meg. Ezek segíthetik a kollégával való egyeztetést, ha az szükséges.

"Zsúrkocsi":	Z/1 kicsi	Dátum:	2008-04-07	Tanóra:	3.
Osztály:	12. C	Tantárgy:	matematika	Tanterem:	2/2a
Kezdés:	09:35	Befejezés:	10:20	Tanár:	Bécsi Orsolya

Az összes eszköz foglaltsága megtekinthető együtt (a mai vagy holnapi napra). [Ide kattints!](#)

Kész

6. ábra: Adatbevitel

A világoskék színű egy eszköz nevét tartalmazó gombok a bejelentkezett tanár által korábban más eszközökre vonatkozó bejegyzéseit jelölik. A gombra való kattintás hatására az oldal frissül és az aktuális eszköz az lesz, amely a világoskék gomb felirata volt.

9.5. Napi adatok

Annak megkönnyítésére, hogy áttekinthető legyen egy nap terheltsége, a még szabad eszközök között könnyebb legyen az eligazodás, célszerű volt egy olyan kimutatás készítése, amely minden eszközre vonatkozóan tartalmazza az adatokat. Az aktuálisat megelőző napra vonatkozó oldal akkor segít, ha valamilyen probléma esetén meg kell keresni azt, aki előzőleg használt egy eszközt.

Betöltve: 2008-04-05 10:39:51.

Váltás: Ma

Tegnap: 2008-04-04

Kocsi	Z/1 kicsi	E/4 nagy	2/4 hefop1	1/6 hefop2	Z/1 hefop3	1/1	2/5	fiz hefop4	kémia
1. óra	Nagyné Farkas Csilla 10.B 2/8	Koós Judit 12.C E/4	Belinszky Péter 12.C 2/7						
2. óra	Nagyné Farkas Csilla 10.B 2/8		Dézsi András 10.A 2/1	Módis Lászlóné 12.E 1/2					
3. óra		Galambosné Hornyák Erika 9.C E/4		Dézsi András 11.A 1/3			Géczy Anikó 7.A 1/0		
4. óra	Bécsi Orsolya 12.C 1/6						Géczy Anikó 7.A 1/0		
5. óra	Juhásznő Vigh Irén 7.A pódium	Belinszky Péter 12.E E/2	Koós Judit 12.C 2/7	Módis Lászlóné 12.D 1/7			Géczy Anikó 7.A 1/0		
6. óra	Juhásznő Vigh Irén 7.A pódium		Szabóné H. Mária 13.C 2/4					Orosz Norbert 10.D fizika	
7. óra	Juhásznő Vigh Irén 7.A pódium								

Kész

7. ábra: Napi adatok

10. Adminisztráció

Az program használatának megkezdésekor, amikor a **zsur.sql** állományt módosítjuk célszerű minden rendelkezésre álló adatot beírni. A felhasználók jelszava az állományban kódolt formában található. Mivel mindenkinek ugyanaz a jelszava ezt nem mondjuk el a tanároknak, hanem mindenkinek adunk egy újat, amit később tetszése szerint megváltoztathat. A jelszó megváltoztatásához régi jelszóként 010699-t kell beírunk, ez felel meg az állományban található **461c4c1faecfd3ad98a217bcc13eba3b** kódnak. További felhasználókat a mysql konzolon kiadott SQL parancsokkal lehet létrehozni.

A táblák módosításához szükséges ismerni a táblák szerkezetét.

10.1. A csrend tábla

Táblanév	Mező	Típus	Null	Alapértelmezett	Megjegyzések
csrend	kedd	smallint(6)	Nem	0	
	ora	tinyint(4)	Nem	0	
	kezd	time	Nem	00:00:00	
	veg	time	Nem	00:00:00	

Ez a tábla leírja az iskolában érvényes csengetési rendet, figyelembe véve a heti egyszemélyi (keddi) eltérést a hét többi napjától. Úgy kell feltölteni, hogy a keddi napokat külön fel kell tüntetni. Iskolánkban ekkor az órák más időpontban kezdődnek. Ha a **kedd** mező értéke 1, akkor a keddi értékeket tartalmazza a tábla, ha a mező értéke 0, akkor a többi napra vonatkozó órakezdési és –befejezési időpontok vannak a megfelelő mezőkben.

10.2. Az igény tábla

Táblanév	Mező	Típus	Null	Alapértelmezett	Megjegyzések
igény	id	int(11)	Nem		Auto_increment
	kocsi	tinyint(3)	Nem	0	
	datum	date	Nem	0000-00-00	
	terem_id	tinyint(3)	Nem	0	
	ora	tinyint(3)	Igen	0	
	oszt_id	tinyint(3)	Nem	0	
	kezd	time	Igen	00:00:00	
	veg	time	Igen	00:00:00	
	tanar_id	smallint(5)	Nem	0	
	tant_id	tinyint(3)	Nem	0	
	cim	varchar(30)	Igen	NULL	

Ez a tábla tartalmazza az eszközök felhasználásának adatait: a „kocsi” sorszámát, a felhasználás dátumát, a tantermet, az adott osztály azonosítóját, a tanóra sorszámát és időpontját, az órát tartó tanár, a tanterem, és a tantárgy azonosítóját, valamint az óra tartalmára utaló szöveget. A felhasználás idejének meghatározásához nem lenne elegendő a tanórák sorszámát tárolni az esetleges délutáni foglalkozások tanórai keretbe nem illeszthető szabadon választható kezdési és befejezési időpontjai miatt. Az órarendi órák esetén nem lenne szükséges az óra kezdetének és végének a tárolása, de az adatok későbbi egységes kezelés miatt ezt célszerű volt megtenni.

Adminisztrációs szempontból ezzel a táblával nincs teendő. A felhasználást összesítő lekérdezések elkészítésekor találkozunk vele, módosítani nem kell.

10.3. A kocsi tábla

Táblanév	Mező	Típus	Null	Alapértelmezett	Megjegyzések
kocsi	id	tinyint(3)	Nem		Auto_increment
	nev	varchar(10)	Nem		
	leiras	text	Nem		

Ez a tábla a kölcsönözhető eszközök adatait tartalmazza. Az **id** mező tartalmát nem kell megadnunk, azt az adatbázis kezelő automatikusan növeli minden új rekord esetén. A **nev** mezőbe az eszköz neve kerül, amely azokon a gombokon is megjelenik, amelyek rá hivatkoznak. Erre legyünk tekintettel a név kiválasztásakor. Nálunk a név utal arra, hogy az eszköz hol található, illetve milyen pályázat útján sikerült beszerezni. Erre akkor van szükség, ha a pályázat előír valamilyen kimutatást arról, hogy mire és mikor használtuk az eszközöket.

A **leiras** mezőben tetszőlegesen hosszú tájékoztatást adhatunk felhasználóinknak a „kocsi” használatával kapcsolatban. Itt célszerű leírni milyen célokra használható az eszköz, hogyan lehet hozzáférni, hol használható. Ez különösen az új felhasználókat segíti. Ezek az információk az aktuális „zsúrkocsi”-ről mindig megjelennek az igénylő oldalon.

10.4. Az osztály tábla

Táblanév	Mező	Típus	Null	Alapértelmezett	Megjegyzések
oszt	id	tinyint(3)	Nem		Auto_increment
	nev	varchar(10)	Nem		
	sorrend	int(5)	Nem		
	aktiv	tinyint(1)	Nem	1	

Ez a tábla az osztályok neveit tartalmazza kiegészítve az iskolában megszokott sorrendben való megjelenítéshez szükséges attribútummal. Az **id** mező tartalmát nem kell megadnunk, azt az adatbázis kezelő automatikusan növeli minden új rekord esetén. A **nev** mező az osztály megnevezését tartalmazza. A **sorrend** mező segítségével meghatározhatunk egy a **nev** mező értékétől független sorrendet a listából való választás során a könnyebb eligazodás segítéséhez. Az iskola képzési rendjében bekövetkező változások miatt szükséges a már nem létező osztálynevek kiszűrése listázáskor, ami szükségessé teszi annak jelzését, hogy élő-e az

adott név. Erre a célra az **aktív** mezőt használjuk. 0 érték jelzi azt, hogy az adott osztály jelenleg nem választható ki. Az 1-es érték esetén az adott osztály megjelenik a legördülő listákban.

10.5. A tanar tábla

Táblanév	Mező	Típus	Null	Alapértelmezett	Megjegyzések
tanar	id	tinyint(3)	Nem		Auto_increment
	nev	varchar(30)	Nem		
	azon	varchar(10)	Nem		
	pass	varchar(32)	Nem		
	aktív	tinyint(1)	Nem	1	

Ez a tábla a tanárok azonosításához szükséges adatokat tartalmazza. Az **id** mező tartalmát nem kell megadnunk, azt az adatbázis kezelő automatikusan növeli minden új rekord esetén. A **nev** mező a tanár teljes nevét tartalmazza. Az azonosító a tanár által szabadon választható, csak a fentebb közölt szabályokat kell betartani. Az **azon** mező tartalmát bejelentkezés-kor az alkalmazás ellenőrzi, összehasonlítja a beírt értékkel. Hasonlóan történik a jelszó ellenőrzése is, de a **pass** mező tartalma a beírt jelszó titkosított formájával kerül összehasonlításra. A tantestület összetételének módosulásával a tanári névsort tartalmazó listákból kihagyhatjuk a távozott kollégákat, illetve az ideiglenesen távol lévőket. Erre a célra az **aktív** mezőt használjuk. 0 érték jelzi azt, hogy az adott tanár jelenleg nem választható ki. Az 1-es érték esetén az adott tanár megjelenik a legördülő listákban.

10.6. A tant tábla

Táblanév	Mező	Típus	Null	Alapértelmezett	Megjegyzések
tant	id	tinyint(3)	Nem		Auto_increment
	nev	varchar(20)	Nem		

Ez a tábla a tantárgyak neveit tartalmazza. Az **id** mező tartalmát nem kell megadnunk, azt az adatbázis kezelő automatikusan növeli minden új rekord esetén. A **nev** mező a tantárgy megnevezését tartalmazza.

10.7. A terem tábla

Táblanév	Mező	Típus	Null	Alapértelmezett	Megjegyzések
terem	id	tinyint(3)	Nem		Auto_increment
	nev	varchar(10)	Nem		

Ez a tábla a tantermek neveit tartalmazza. Az **id** mező tartalmát nem kell megadnunk, azt az adatbázis kezelő automatikusan növeli minden új rekord esetén. A **nev** mező a tanterem megnevezését tartalmazza.

11. A program ismertetése

11.1. Az ellátott funkciók

Az alkalmazás célja olyan lehetőség biztosítása, amely az eszközhasználatot könnyen tervezhetővé teszi, és az igénylési lehetőséget helytől és időponttól függetlenül szolgáltatja. Az utóbbit az internet használatával lehet nyújtani.

Mindenki megnézheti a foglaltsági viszonyokat, ehhez azonosítás nem szükséges. Ekkor eszközönként áttekinthetőek az aktuális nappal kezdődően nyolc nap bejegyzései. Ezt egy az órarendhez hasonló táblázat mutatja, ami minden tanárnak ismerős, első ránézésre is könnyen áttekinthető. A bejegyzések részletei nyomógombok segítségével megnézhetőek. Ez segíti az egyeztetést abban az esetben, ha a foglaltság ellenére mégis ugyanarra az eszközre van szüksége valakinek. Az általános tájékozódást segíti az adott nap összes eszközre vonatkozó összesített táblázata is, amely a következő és a megelőző napra is rendelkezésre áll.

Azonosítás után a fentiek kiegészülnek az eszközök lefoglalásának lehetőségével, illetve a már létrejött bejegyzések törlésével. A bejelentkezett felhasználó több oldalon keresztül megtartásához a PHP beépített session kezelését használja az alkalmazás, amely a cookie-k használatán alapul (ennek használatát engedélyezni kell). Ez addig marad a gépen, amíg a böngészőt be nem zárjuk. Célszerű volt olyan időkorlát meghatározása, ami után megszüntetjük a cookie-t és újabb bejelentkezés válik szükségessé.

Az alkalmazás lehetőséget biztosít a felhasználói azonosító és jelszó megváltoztatására, de regisztrációs és ehhez kapcsolódó adminisztrációs résszel nem rendelkezik. Tartalmaz viszont egy olyan kódrészletet, amely egy adott jelszó segítségével lehetőséget biztosít arra, hogy bármelyik tanár nevében be lehessen jelentkezni. Ennek felhasználásával minden bejegyzés módosítható.

11.2. index.php

A legtöbb feladatot ebben az állományban lévő kód valósítja meg. Alapvetően minden megjelenítési és adatbeviteli funkció itt valósul meg.

11.2.1. Megjelenítés

A fenti funkciók nagy részét ebben az állományban található kód valósítja meg, és ez a nyitó oldal is. A megjelenítendő oldal teljes tartalmát a php hozza létre, ezért a fájl a `<?php` karaktersorozattal kezdődik és a `?>` sorozattal zárul. Az első utasítás a 2. sorban egy új sessiont hoz létre az alkalmazás számára, vagy a már létező adatait használja, majd beszúrja a **conf.inc** állományt az adatbázis eléréséhez szükséges adatokat tartalmazó változók eléréséhez. Ezután a hét napjait tartalmazó tömb megadása jön, amit a html első sorainak kiírása követ. Ez a rész tartalmazza azokat a CSS stílusokat, amelyek meghatározzák az alkalmazott gombok jellemzőit. A 20. és a 45. sor között JavaScript függvény kódja található. Ennek feladata annak ellenőrzése, hogy a délutáni időpontok megfelelően legyenek kitöltve, például nem lehet harmadik karakternek „:” karakteren kívül mást megadni. A következő négy sor feladata két perc elteltével meghívni a login.php-t, ami a session megszüntetésével jár.

A session adatai `$_SESSION` belső változóban érhetőek el. A bejelentkezéskor a bejelentkezett tanár azonosítója a session adatai közé kerül `tanar_id` néven. Az 53. sor ennek létezését vizsgálja, és ha talál ilyet, akkor a `$tanar_id` változóba másolja. A változó értéktől függően, ha az nem üres, akkor a `<body>` tag paraméterei között az `onload=\"session_ki()\"` biztosítja az időzítő script elindulását. (A html által megkövetelt idézőjeleket a backslash karakterrel le kell védeni, hogy ne szöveghatároló jelként kerüljön értelmezésre.)

Ezután az adatbázis-kezelőhöz való csatlakozás következik, amit az adatbázis kiválasztása követ.

A `$kocsi` változó tartalmazza az eszköz azonosítóját, ami az oldalnak a megfelelő `form get` metódus szerint ad át, így a 65. sorban a `$_GET` beépített változóból kiolvasható. A 66. sor ennek létezését vizsgálja és a következő sor 1-es értékre állítja a `$kocsi` változót, ha az üres volt.

A 68. sortól kezdődően az aktuális állapottól függő adatok megjelenítése következik, ahol attól függően, hogy van-e bejelentkezett felhasználó. illetve melyik „kocsi” az aktuális különböző adatok jelennek meg. Az ehhez szükséges információkat az adatbázisból nyerjük megfelelő lekérdezésekkel. A `mysql_query()` függvény szöveges paramétereként kell megadni az sql utasítást. A `mysql_fetch_array()` függvény pedig az előzőleg visszaadott azonosítóval megadott lekérdezés sorait adja vissza egy tömbben. Itt második paramé-

terként megadható az értékek asszociatív tömbként való kezelése, így az adatokra a mezőnevek megadásával hivatkozhatunk. Itt kerül kiírásra a bejelentkezést, illetve a kijelentkezést lehetővé tevő gombok. Mindkét esetben a `login.php` oldallal folytatódik a végrehajtás. Az elérhető eszközök listája mindig azonos formában jelenik meg, csak az aktuálisnak más a színe.

A 104. sortól annak a táblázatnak a felépítése következik, amely az aktuális eszköz foglaltságát jeleníti meg. A táblázat felett a `date()` függvény által formázott, és a `time()` függvény által visszaadott aktuális időpont jelenik meg, ezután maga a táblázat. A táblázatban elhelyezett gombok négyfélék lehetnek. Ha egy adott időpontban szabad az aktuális eszköz, akkor egy „Új” feliratú, ha foglalt, akkor attól függően, hogy a bejelentkezett felhasználó tetete-e a bejegyzést vagy a „Töröl” feliratú, ha más, akkor a másik tanár nevét tartalmazó gomb jelenik meg. A döntéseket az **igeny** tábla adatai alapján lehet meghozni. A táblázat utolsó sora(i) a délutáni használatot mutatják. Amennyiben egy napon több délutáni igény is van, annyi sor jelenik meg ahány ilyen előfordul.

11.2.2. A kiválasztott funkciók kezelése

A kiválasztott gombok neve (`name` paramétere) két részből áll. Ezeket kettőspont választja el egymástól. Első részében a kiválasztott napot, a másodikban az elvégzendő feladatra utaló értéket tartalmaz. A gombok felirata a délutáni időtartamban az ötödik pozícióban „-” jelet tartalmaz. Ezt is felhasználjuk a funkciók elkülönítésére, tehát ez a többi gomb esetén nem lehet így, vagyis az eszközök és a tanárok nevében nem szerepelhet ezen a pozíción ez a jel. A 210-235. sorok az eseteket választják szét és adnak egy `$ddd` nevű változónak különböző értéket.

Az **Új** gomb megnyomására egy új táblázatot kell létrehozni. Ebbe kerülnek az adatbevitelt biztosító beviteli mezők és a kiválasztható elemek legördülő listái. A cellák egy része a kiválasztásból következő adatokat jeleníti meg, azok nem módosíthatóak itt. Ilyen adatok a kiválasztott „Zsúrkocsi” neve, annak a napnak a dátuma amikorra az eszközt lefoglalásra kerül, az óra száma, az óra kezdésének és befejezésének időpontja. Legördülő listából választható az egymást követő ugyanilyen órák száma, az osztály, a tanított tárgy és a tanterem. Egyszerű szöveges beviteli mezőbe az óra tartalmára utaló cím írható. A legördülő listák adatai a megfelelő adattáblából kerülnek az oldalra. Például az osztályok listájának létrehozása az alábbi kóddal lehetséges.

```

print "<select name=\"oszt_id\" id=\"oszt_id\">\n";
$keres = "SELECT * FROM oszt";
$eredmeny = mysql_query($keres) or die("Hiba az osztály tábla
    lekérdezésében");
while ($sor = mysql_fetch_array($eredmeny, MYSQL_ASSOC)) {
    print "<option value=\"\$sor[id]\">$sor[nev]</option>\n";
}
print "</select>\n";

```

Az adatok beírása csak a **Lefoglalás** feliratú gomb megnyomásakor történik meg. Ekkor az `uj.php` állományban történik meg az adatok adatbázisban való rögzítése. A délutáni időpontokra hasonló kódrészlet vonatkozik, ez a **Beír** feliratú gomb megnyomásakor működik.

Ha a már létrehozott bejegyzést kell törölni a **Töröl** gomb megnyomása után, csak ellenőriztetni kell a felhasználóval, hogy ténylegesen azt az órát jelölte-e ki, amelyiket akarta. Ehhez csak megjelenítjük az adatokat és lehetővé tesszük a tényleges törlést a **Törlés** feliratú gombbal. A műveletet a `torol.php` fájl hajtja végre.

11.3. A login.php

Ez az állomány a felhasználó azonosítását végzi. Az első sorok lerombolják a létező session-t. A továbbiakban két beviteli mezőt tartalmazó **form** következik, és egy hivatkozást a **login1.php**-ra. Az első beviteli mező text típusú és a felhasználói azonosító bekérésére szolgál, a másodikba a jelszót kell beírni. Ez utóbbi **password** típusú. A **Belép** gomb megnyomására a login **form** az adatokat ugyanennek az állománynak adja át. A `$_POST` globális változóban a **form** által átadott adatok elemzése után hívja meg az **index.php**-t. Ha sem az azonosítót, sem a jelszót nem töltötte ki a felhasználó, akkor nem kap az **index.php** tanárazonosítót és az oldalra nem lesz bejelentkezve senki. Ezután az azonosító és a jelszó érvényességének ellenőrzése következik. Ha minden rendben, akkor a `$_SESSION['tanar_id'] = $sor[id];` utasítással beállításra kerül a tanárazonosító, ami a session élettartama alatt érvényben marad. Amikor a `print "<SCRIPT LANGUAGE = \"JavaScript\"> location.href='index.php';</SCRIPT>";` sor meghívja az `index.php`-t, az már bejelentkezett felhasználóként kezeli az ellenőrzött adatok tulajdonosát.

Amennyiben az adminisztrátor valamely tanár nevében akar belépni, akkor elég csak a kódban előre elhelyezett jelszót beírnia. A tanár kiválasztása után úgy léphet be, mintha is-

merné a felhasználó jelszavát. A beírt jelszó a md5 () függvény visszaadott értékével van összehasonlítva. az md5 ('624896') eredménye például a kódban is szereplő b901587dffeb725281d5b589fda8c43e karaktersorozat.

11.4. A login1.php

A felhasználói azonosító beállítását, módosítását, és a jelszó megváltoztatását végzi. Az adatok bevitele után azok elemzését is ez az állomány végzi. Az azonosító csak kis- és nagy betűket és számjegyeket tartalmazhat, hossza pedig legalább két és legfeljebb tíz karakter lehet. Ha ezek nem teljesülnek hibaüzenetet jelenít meg. A vizsgálat egy egyszerű reguláris kifejezés illesztésével történik.

```
if (!preg_match("/[a-zA-Z0-9]{2,10}/", $_POST[azon])) {
    $mehet=0;
    print "<SCRIPT LANGUAGE=\"JavaScript\"> alert(\"Az
    azonosító megadása rossz! Hossza 2-10 betű vagy szám.\");
    location.href='login1.php';</SCRIPT>";
}
```

Ezután annak vizsgálata következik, hogy az azonosítót használja-e más. A jelszavak esetében először a régi összehasonlítását kell elvégezni a tárolttal. Az új esetében az azonosítóhoz hasonlóan a hossz és a tartalom vizsgálata következik, és ellenőrizni kell az új jelszó két beírt alakjának egyezését.

11.5. Az uj.php

A foglalási adatoknak az igény táblába való beírását végzi a délutáni időpontokra vonatkozó ellenőrzések elvégzése után. Ez annak vizsgálatát jelenti, hogy nem fedik-e át egymást az időintervallumok

```
$helyes=1;
$eredmeny=mysql_query("SELECT * FROM igény WHERE
    datum='$_POST[datum]' AND kocsi='$_POST[kocsi]'");
while ($sor=mysql_fetch_array($eredmeny,MYSQL_ASSOC)) {
    if (!(($sor[kezd]<$_POST[kezd] && $_POST[kezd]>=$sor[veg])
        || ($sor[kezd]>=$_POST[veg] && $_POST[veg]<$sor[veg])))
        $helyes=0;
```

```
}
```

illetve, nem kezdődhet ilyen foglalkozás a hetedik óra vége előtt és befejezéseként az iskola zárásának időpontja szerepelhet.

```
preg_match("/14:[1-5]\d|1[5-9]:[0-5]\d/", $_POST[kezd]) and  
preg_match("/1[4-9]:[0-5]\d/", $_POST[veg])
```

11.6. A torol.php

Csak a post metódussal kapott adatoknak megfelelő rekordot törli az igény táblából. Nem végez ellenőrzéseket, csak az sql parancs sikerességét vizsgálja.

12. Összegzés

Az alkalmazás eredetileg azért készült, mert a „digitális zsúrkocsi” program keretében kapott két eszközhöz való hozzáférést nehéz volt megoldani. A pályázat feltételei között szerepelt egy félévenkénti kimutatás készítése, amelyben az eszközök használatát kellett dokumentálni. Ennek tartalmaznia kellett a használat időtartamát, a tantárgyat, az óra címét, stb. A kezdeti papír alapú előjegyzési mód sok adminisztrációt jelentett, és gyakran előfordult, hogy mire egy tanár hozzájutott a listához már más lefoglalta az eszközt. Ez sok esetben jelentősen rontotta az órák tervezésének biztonságát, így többen lemondtak a számítógép által segített órai munkáról. Az alkalmazás használatba vétele után megnőtt az eszközök kihasználtsága. Az interaktív táblák beszerzésekor már kialakult rendszer szerint történt azok igénylése. Az alkalmazást csak kis mértékben kellett átalakítani a több eszköz kezeléséhez.

Az alkalmazás jelen állapotában jól bevált, megfelelően látja el feladatát. Komoly hiányossága azonban, hogy az adminisztrációt nem támogatja. Ez komoly veszélyeket rejt, mivel az adatbázist „kézzel” kell módosítani, könnyű hibázni. Másrészt ezek a műveletek szaktudást igényelnek, emiatt a feladat nem ruházható át bárkire. Az esetlegesen szükséges kimutatások elkészítéséhez is közvetlenül az adatbázisból kell kinyerni az adatokat és megfelelő formára hozni. Az alkalmazás további bővítése ezeknek a feladatoknak a megoldását jelentheti.

Irodalomjegyzék:

- [1] Informatikai oktatási program, IKT (Vaskor – Zsakó ...), 2006, -
http://www.sulinet.hu/ikt/segedletek/inf_okt_program.pdf
- [2] Oktatási informatikai stratégia -
http://www.okm.gov.hu/letolt/informatikai_strategia_040326.pdf
- [3] GNU-kiáltvány (nem hivatalos magyar fordítás) - www.gnu.hu/gnu-kialtvany.html
- [4] GNU GPL (nem hivatalos magyar fordítás) - www.gnu.hu/gpl.html
- [5] Linux_Történet - http://wiki.hup.hu/index.php/Linux_T%C3%B6rt%C3%A9net
- [6] Apache - <http://wiki.hup.hu/index.php/Apache>
- [7] MySQL - <http://hu.wikipedia.org/wiki/MySQL>
- [8] PHP - <http://wiki.hup.hu/index.php/PHP>
- [9] dselect - <http://www.szabilinux.hu/dselect/index.html>
- [10] `/etc/php5/apache2/php.ini`
- [11] PHP kézikönyv - <http://www.pergel.hu/phpdoksi/>
- [12] MySQL dokumentáció - <http://dev.mysql.com/doc/refman/5.0/en/>