

DIPLOMAMUNKA

Veress László

Debrecen

2010.

Debreceni Egyetem

Informatikai Kar

**TANTÁRGYI KÖVETELMÉNYEK KEZELÉSE
WEBFELÜLETEN**

Témavezető:

Dr. Husi Géza

tanszékvezető

főiskolai docens

Készítette:

Veress László

mérnök-informatikus hallgató

Debrecen

2010.

Tartalomjegyzék:

1. Bevezetés	4
2. Az internet	6
2.1 Az internet működése	6
2.2 Domain és helymeghatározás	7
2.3 Miért érdemes megjeleníteni?	8
2.4 Célcsoport	8
3. Az adatbázis	9
3.1 Adatbáziskezelő rendszerek létrehozása	9
3.2 Adatmodellezés	10
3.3 Adatfüggetlenség	13
3.4 Az adatbáziskezelő rendszer részei	13
3.5 Az adatleíró nyelv (Data Definition Language)	13
3.6 Az adatkezelő nyelv (Data Manipulation Language)	14
3.7 Az adatbáziskezelő rendszer	14
3.8 Adatmodellek:	15
3.9 Az SQL nyelv	16
3.10 A MySQL	17
3.11 A MySQL legfőbb tulajdonságai	18
4. A HTML	19
4.1 HTML oldalak készítése	19
5. A PHP	20
5.1 A PHP fejlődése	20
5.2 Miért a PHP?	22
5.3 A fejlesztés sebessége	22
5.4 A PHP nyílt forráskódú	22
5.5 Teljesítmény	23
5.6 Hordozhatóság	23
6. A program bemutatása	24
6.1 A program készítésének menete	24
6.2 A programnyelvek kiválasztása	24
6.3 A program	25
6.4 Adatbázis felépítése	25
6.5 Az oldal felépítése	28
6.5.1. Felhasználói típusok:	28
6.6 Bejelentkezés adminisztrátorként	29
6.6.1 Keresés	30
6.6.2 Nyomtatás	31
6.7 A táblázat	32
6.8 A tárgyleírás és a 15 hetes bontás	33
6.9 A fájlból való feltöltés	35
6.10 Archiválás	38
6.11 Saját adatok	38
6.12 Felhasználó hozzáadása	40
6.13 Felhasználó módosítása	40
6.14 Hiányosságok	41

6.15 Kilépés	43
6.16 Bejelentkezés oktatói jogosultsággal	43
6.17 Tárgyak	44
6.18 Saját adatok	44
6.19 Kilépés	45
6.20 Bejelentkezés hallgatói jogosultsággal	45
6.21 További fejlesztési lehetőségek	46
7. Összefoglalás	47
8. Irodalomjegyzék	48
9. Ábrák jegyzéke	49
10. Köszönetnyilvánítás	50

1. Bevezetés

Az emberek igényei az évek során megváltoztak, a mai rohanó világban elsődleges szemponttá váltak a következők: gyorsaság, funkcionalitás, naprakészség és nem utolsósorban a megbízhatóság. Ezek igazak az internet felé irányuló követelményekre, igényekre is.

Magyarországon az internet folyamatos terjedése figyelhető meg. Több mint 2,2 millió a szélessávú internet előfizetés és folyamatosan emelkedik, főleg a mobilinternet elterjedésével, ami 2007-es megjelenése óta húzóerőt jelent.

Ezáltal megváltoznak az általános emberi szokások, a televízió és a nyomtatott sajtó a háttérbe kerül az internettel szemben. Előnye az embernek az, hogy azt néz vagy olvas, amire igénye van nem pedig azt, amit ráerőltetnek a nagy kereskedelmi TV csatornák. Így könnyen eljut a kívánt tartalom a nagyközönség felé.

Mint tudjuk, amit a felhasználó lát a monitorán keresztül, amikor a weboldalra kattint az csak a látszat. Egy weboldalt a forráskódja határozza meg. Abban dől el, hogy mit lehet csinálni az oldalon, mennyire hasznos, milyen felhasználásra és milyen területre szánják.

Azért választottam ezt a témát, mert mindig is érdekelt az internet, az internet fejlődése. Így nemcsak külsőt láthatom, hanem a technikai részletekkel is tisztában lehetek. Szabadidőmben sokat netezek, ez az elsődleges hírforrásom, innen szerzem be a napi híreket, újdonságokat, elektronikus bankon keresztül rendezem a számláimat, gyakran rendelek ételeket és egyéb dolgokat, ezért már rég érdeklődési körömbe tartozik. A nyomtatott sajtóval is vannak előnyei például friss, naprakész, könnyebb javítani, szerkeszteni és a környezetszennyezés sem elhanyagolható szempont, mivel óriási mennyiségű fát vágnak ki és rengeteg festéket, tintát használnak fel a papírra való nyomtatáshoz.

Célnak tekintetem egy olyan szoftver, weboldal létrehozását, aminek készítése közben tapasztalatokat szerezhettem, tanulhattam, fejlődhettem valamint, hogy megfeleljek az általam és a konzulensem által támasztott követelményeknek.

Egy olyan tanulmányi rendszert készítettem, amivel az egyetem tanárai és hallgatói is egyaránt tájékoztatást kaphatnak a munkájuk/tanulmányaik során fellépő fontos kérdésekre, átláthassák, hogy mire elvégzik az egyetemet milyen tárgyakat fognak tanulni, melyik tárgy melyikre épül, ki tartja számukra, és a tantárgyról is olvashatnak, ezek mellett még rengeteg információt is megtudhatnak.

Az rendszer elkészítéséhez szükséges ismereteit elsősorban önszorgalomból tanultam meg, de segítségemre volt az, hogy harmadik féléves tárgyként Adatbázisrendszerek címen futott tárgy keretében tanulhattam az adatbázis rendszerekről, az alapokról, felépítésükről, relációsémákról, normálformákról, az adatbázisok tervezésének elméletéről, az SQL adatbázisnyelvről.

A dolgozatom első részében fontosnak tartom bemutatni az internetet, fejlődését mivel a weboldal arra készült, azon keresztül érhető el, majd áttérek az adatbázis rendszerekre aztán írok a HTML és a PHP nyelvekről.

Azután a program bemutatása következik, ahol részletesen kifejtem a tudását, funkcióit.

A dolgozatom végén pedig található az összefoglalás, irodalomjegyzék, az ábrák jegyzéke és végül a köszönetnyilvánítás.

2. Az internet

Az Internet (Internetworking System) számítógépes hálózatok világhálózata, amely behálózza az egész Földet, összekapcsolva kormányzati, katonai, kereskedelmi, üzleti, oktatási, kutatási, és egyéb (pl. nonprofit) intézményeket, valamint egyéni felhasználókat. Az Internetre különböző méretű és kiterjedésű hálózatok kapcsolódnak több szinten, hierarchiát alkotva. A legmagasabb szinten levő úgynevezett gerinchálózatok nagy sávszélességű és rendszerint nagy területekre kiterjedő kommunikációs hálózatok, amelyeket többnyire erre a feladatra szakosodott cégek üzemeltetnek. A hierarchia következő szintjén állnak a tranzithálózatok, amelyekre példa egy olyan városi hálózat, amely az adott település nagyobb intézményeinek helyi hálózatait köti össze. A hierarchia legalsó szintjén állnak a végálózatok, az egyes intézmények helyi hálózatai. Az egyéni felhasználók otthoni számítógépei csak egy internet-szolgáltatón keresztül kapcsolódhatnak az Internetre. Az internet-szolgáltatóhoz való kapcsolódásra azonban több lehetőség van például telefonvonalon, modemmel vagy a helyi kábeltévé kábelein keresztül. [1.]

2.1 Az internet működése

„Az Internet sok millió számítógép világméretű hálózata. Gerincét az a több tíz- vagy százezer nagy teljesítményű számítógép - úgynevezett szerver alkotja, amelyeket műhold, üvegszál, mikrohullám és egyéb adatátviteli technikák kapcsolnak össze egymással. Ezekhez a gépekhez közvetlenül vagy internetszolgáltatókon keresztül csatlakoznak az egyes felhasználók, kliensek. A hálózati technológiának köszönhetően bármely kliens - a szervereket összekötő gerinchálózaton keresztül - el tudja érni a hálózat bármely másik pontját függetlenül attól, hogy az egymással kapcsolatba került számítógépek fizikailag hol helyezkednek el.

A gépek és a rajtuk tárolt anyagok hihetetlenül sokfélék. Az, hogy mégis megértik egymást és kommunikálni képesek, az általuk beszélt közös nyelvnek, az úgynevezett protokolloknak köszönhető.

Mindez lehetővé teszi, hogy az Interneten bármely gépről adatokat, leveleket vagy hozzájuk csatolt képeket, dokumentumokat küldjünk partnerünknek e-mailben (elektronikus levél) - ha hozzáférést engedélyeznek számunkra, távoli számítógépekről hívjunk le anyagokat a saját gépünkre - elérjünk gépünkről olyan helyeket - weboldalakat, ahol grafikus oldalak formájában, előre elkészített információkat, szövegeket, képeket, hangot, mozgóképet találunk - magunk is nyilvánossá tegyünk olyasmit, ami másokat érdekelhet.

2.2 Domain és helymeghatározás

Az első domainnevet 1985. március 15-én adták ki: www.symbolics.com.

Mivel emberi fogyasztásra teljesen alkalmatlanok az IP címek, ezért kialakították a domain neveket, amelyek általában értelmes rövidítések. Például egy domain cím így nézhet ki: dinero.hu, ahol a [dinero](http://dinero.hu) a gépet jelöli (számítógép neve), a [hu](http://dinero.hu) pedig azt jelöli, hogy a gép Magyarországon van. A domain névben vannak tiltott karakterek, amit lehet használni: az angol abc betűi, számok, és a pont. Végződések: [.com](http://www.com): kereskedelmi, vagy ipari tevékenységet folytató szervezet. (comercial) [.edu](http://www.edu): oktatási intézmény (educational) [.mil](http://www.mil): katonai intézmény (militari) [.gov](http://www.gov): államigazgatási szervezet (government) [.net](http://www.net): A hálózat adminisztrációjával foglalkozó szervezetek. (network) [.org](http://www.org): Non-profit szervezetek. (organization)

A domainnév annak szükséges, aki a nap 24 órájában rá van kötve az Internetre tehát, információt szolgáltat és nem csak gyűjt. Dinamikus IP címet, csak arra az időre kap, amíg rá van kapcsolódva az Internetre. Például a modemes előfizetőknek dinamikus IP címe van. A hálózaton a helymeghatározásra az URL-t (Universal Resource Locator) használják. Az Internet maga egy fizikai átviteli csatorna. A protocol az, ami meghatározza, hogy az Internet melyik szolgáltatását veszem igénybe.

2.3 Miért érdemes megjelenni?

Az Internet, mint a legújabb média, összevonja a televízió, a rádió, az újság és a telefon sajátosságait - miközben költségei a töredékét teszik ki azoknak. A hálózat multimédiás lehetőségeivel a hagyományos közlésterek kitágulnak, új értelmet nyernek. A számítógép monitorához, mivel az közelebb helyezkedik el a személyiségi zónához (általában fél méteren belül van), közvetlenebb viszonya van egy felhasználónak, mint egy tévé nézőnek a televíziókészülethez. Az emberek szoktak háttér-televíziózni, de nem szoktak háttér-internetezni. Az Internet pozitívabb hatást és sokkal aktívabb befolyásolást hoz létre, mint egy televízió. Személyes jellege miatt új szempontok érvényesülnek, melyben a felhasználó figyelme monitorához képest sokkal fokozottabb, mint más médiákhoz.

Az Internetet használók köre az információt keresőkre és a barangolókra osztható. Az első csoport speciális helyeket vesz igénybe és tudatosan keres a hálón, a másik csoport sodródik az Interneten: vagy eljut valahová, vagy elveszíti érdeklődését. Az információszerzés átválthat kikapcsolódásba, de a kettő egyszerre is történhet.

Ezen felhasználók köre évről évre bővül, az elektronikus kereskedelem világszerte ugrásszerűen emelkedik, s ez az új "világ" a technika fejlődésének megfelelően egyre nagyobb szerephez jut mindennapi életünkben. 1999 végén több mint 200 millió Internet felhasználóról tud a statisztika. Ebből 110 millió az USA területén, 50 millió Európában és 30 millió a Távol-Keleten él.

2.4 Célcsoport

Az internetezők jól képzettek: 33%-uk befejezett felsőfokú végzettséggel rendelkezik (Carnation Research 2000). A potenciális egyéni vásárlók tehát a társadalom tehetősebb, komoly vásárlóerőt képviselő rétegéből kerülnek ki, akik információéhségüket és - egyre inkább - szórakozási igényeiket is a világháló segítségével elégíti ki. Életkoruk 15-38 év, foglalkozásuk szerint menedzseri vagy vezetői munkakört betöltő értelmiségiek. A viszonteladók jelenthetik a másik nagy, megcélzandó célcsoportot (B2B). Mindkét csoport közös jellemzője az, hogy fogékonyak a technika újdonságaira, készek kiaknázni a fejlődésben rejlő előnyöket és nem köthetők az ország egy adott részéhez.

Célcsoport további jellemzője, hogy vásárlásait, információit elsősorban az Internet segítségével bonyolítja le illetve szerzi be.” [2.]

3. Az adatbázis

„Környezetünk leírásához meg kell azt figyelni, adatokat kell gyűjtenünk az objektumairól, összegeznünk kell azok tulajdonságait, valamint meg kell figyelni a köztük levő kapcsolatokat. Ezek után juthatunk a rendelkezésre álló, tárolt adatokból új ismeretekhez, információhoz. Az adatoknak egy jól strukturált halmazát adatbázisnak nevezzük, amelyből a későbbiekben információt akarunk majd nyerni. Az információnyerés céljából pedig olyan szoftvereket használunk, amelyek lehetővé teszik az adatbázisok kezelését például lekérdezés, módosítás, törlés, karbantartás és amelyeket együtt adatbáziskezelő rendszernek nevezünk.

Az olyan rendszereket, amelyek a hardveren tárolt adatokból szoftverek és egyéb alkalmazói programok segítségével lehetővé teszik a felhasználó számára az információ nyerését, információs rendszereknek nevezzük. Látható tehát, hogy egy információs rendszerhez szükség van adatokra, hardver eszközökre, szoftverekre és a felhasználóra. Szoftveren itt olyan programcsomagot értünk, amely gyorsítja a munka elvégzését, például szövegszerkesztők, táblázatkezelők, adatbáziskezelő rendszerek. Az alkalmazói programok pedig egy-egy szakember által valamilyen programozási nyelven, egy konkrét feladatra írt programok.

3.1 Adatbáziskezelő rendszerek létrehozása

Adatbáziskezelő rendszert alapvetően kétféleképpen szoktak létrehozni. Az egyik módja, hogy egy programozó csoport valamelyik - többnyire harmadik generációs - programozási nyelvben (pl. C, Cobol) megírja a rendszert. A másik módszerben pedig egy már meglévő rendszerben fejlesztik ki az új rendszert. Ehhez többnyire negyedik generációs nyelveket használnak. Ilyen típusú rendszerek a CASE (számítógép által támogatott szoftver fejlesztő)

rendszerek. A legtöbb adatbáziskezelő rendszer rendelkezik ilyen CASE eszközökkel. Továbbá vannak alkalmazásfejlesztő rendszerek is, például: QBE, POWER BUILDER, UNIFACE. A legismertebb adatbáziskezelő rendszerek manapság az ORACLE, INGRES, SYBASE, INFORMIX. Ezek többnyire nagygépes környezetben működnek. Természetesen vannak PC-re íródott adatbáziskezelő programrendszerek is, mint például: DBASE, CLIPPER, FOXPRO.

Az adatbázis létrehozását mindig az adabázistervezés előzi meg. Gondosan fel kell mérni az adott cég igényeit, meg kell fogalmazni a problémákat. Egy adatbázist manapság körülbelül egy-két hónap alatt fejlesztenek ki. Ennek körülbelül 70-75%-a a tervezésre megy el, 10-15%-a a programozásra és a maradék a tesztelésre. Alapos tervezés nélkül ugyanis a rendszer átláthatatlan lesz, nagyon nehéz lesz a későbbiekben módosítani.

Az adatbázistervezés, az adatok jellegének vizsgálata és a köztük levő kapcsolatok alapos elemzése után következhet az **adatmodell** létrehozása.

3.2 Adatmodellezés

Az adatmodellezés tehát az adatállományok tervezésének korszerű módszere. Az adatmodellezéssel az a célunk, hogy egy információs rendszer adatait és az adatok között fennálló kapcsolatokat következetesen ábrázolva, elősegítsük a számítógépes információfeldolgozást. Az adathalmaz és az adathalmaz elemei között fennálló kapcsolatok struktúrált leírását adatmodellnek nevezzük. Az adatmodellek -mint ahogy a nevükből is következik-, általában nem írják le teljesen és hűen a modellezett rendszerben található adatokat és azok kapcsolatait, hanem azoknak csak egy részét.

Egy kielégítő modellnek teljesítenie kell bizonyos feltételeket:

- átfogónak kell lennie, azaz minden lehetséges adatot és minden lehetséges kapcsolatot tudnia kell ábrázolni és kezelni;
- le kell tudnia írni a valóság általános, lényeges és tartós összefüggéseit; redundanciamentesnek kell lennie, azaz minden adatot lehetőleg csak egyszer tartalmazzon;
- következetesnek kell lennie;

- valamint az alkalmazott hardverrel és szoftverrel összhangban levőnek kell lennie.

Az adatmodell nemcsak felsorolása a fájlban tárolt mezőknek, hanem az adatok jelentését is tartalmazza. Így az adatmodellben

- világosan megkülönböztethetők az ábrázolt dolgok és a dolgok tulajdonságai;
- pontosan meghatározott, hogy melyek az ábrázolandó dolgok olyan tulajdonságai, amelyekkel a dolgok egyértelműen azonosíthatók;
- valamint az adatmodell expliciten definiálja az ábrázolt dolgok közötti kapcsolatokat.

Az adatmodell a valóság egy szeletének a leképezését jelenti. A valós világhoz képest az adatmodellek tartalmaznak bizonyos megszorításokat és egyszerűsítéseket, sőt még a modellező személyétől függő vonásokat is. Egy adatmodell úgy viszonyul a valósághoz, mint a térkép az általa ábrázolt tájhoz: a térképrajzolónak egyszerűsíteni kell, ki kell jelölnie a térkép széleit, és el kell döntenie, hogy mit is akar egyáltalán a térképen feltüntetni. Ugyanígy, az adatmodell kialakításakor a rendszerszervezőnek is el kell döntenie, hogy melyek a modelljében az ábrázolandó információelemek, és ki kell választania a dolgok között fennálló lényeges kapcsolatok közül azokat, amelyeket be akar a modellbe építeni. A lényeges információelemek és a modellben ábrázolandó kapcsolatok kiválasztása párhuzamosan történik.

Egy adatmodell a következő elemeket tartalmazza:

- egyed,
- tulajdonság,
- kapcsolat.

Vagyis:

- **Egyedek** a valóság olyan dolgai, amelyek más dolgoktól megkülönböztethetők, azaz az egyed a valós világban létező, fogalmi vagy fizikai léttel rendelkező dolog, amelyet tulajdonságokkal akarunk leírni. (pl. dolgozó, tanuló, vevő, autó, stb.) Ugyanaz a dolog többféle egyednek is tekinthető (pl. egy személy lehet dolgozó), de a képviselt konkrét elemek halmazát egyedhalmaznak nevezzük. (a vevő, mint egyedhalmaz az összes vevőt tartalmazza)

- **Tulajdonságnak** nevezzük a valóság dolgainak azon jellemzőit, amelyekkel az egyedeket leírjuk (pl. autónál tulajdonság a típus, szín, rendszám, stb.). A típus értékei lehetnek: BMW, Opel, stb.. Az egyedek a tulajdonságértékekkel azonosíthatók.
- **Kapcsolatnak** nevezzük az egyedek közötti viszonyok fogalmi tükröképeit. Objektumok közötti viszony pl. az, hogy a munkás a kft. dolgozója. Ennek egy konkrét előfordulása pl. Kovács János a Zabhegyező kft. dolgozója.

Ezen felsorolt adatmodell-elemek mindegyikének három szintje van:

- **Belső, vagy fizikai szint:** az adatok fizikai elhelyezését és fizikai elérési módját írja le. Azaz leírja a tárolás és az elérés módját. A tárolási mód megadja, hogy melyik lemezen fizikailag hol helyezkednek el az adatbázis adott rekordjai és azt, hogy milyen konverziót kell ahhoz elvégezni, hogy az adat a felhasználóknak megfelelő formájú legyen. Az elérési mód megadja, hogy hogyan, milyen sorrendben követik egymást a rekordok, hogyan érhetjük el őket. Tipikus elérési módok: fizikai sorrendben, valamilyen mező szerint rendezve, közvetlenül indexek alapján.
- **Külső szint (view-k):** melyek azt írják le, hogyan látják az egyes felhasználók az adatbázist. (view= alséma, szűkített adatbázis, amely az adatbázis rekordjait tartalmazza, de nem az összes mezővel.)
- **Koncepcionális szint (fogalmi adatbázis):** mely azt írja le, hogy logikailag egységbe vonva, hogyan néz ki ténylegesen az adatbázis. Azaz leírja, hogyan néznének ki az adatok, ha mindenki mindent láthatna belőlük. Ezt a leírást szoktuk az adatbázis sémájának nevezni. Ez tartalmazza a koncepcionális szint összes rekordtípusának tartalmi leírását, de az adatfüggetlenség érdekében nem tartalmazza az adatok helyét és a hozzáférési módot. Ugyancsak ez a szint írja le, milyen kapcsolatok vannak az egyes rekordok között. A kapcsolat leírásának módja az adatmodellől függ.

Még két fogalmat kell tisztáznunk a továbbiak miatt:

- **Séma (típus):** a konkrétan létező egyedek, tulajdonságok, kapcsolatok absztrakciója. Az egyedek bizonyos tulajdonságtípusaik alapján halmazokba rendeződnek. Egy-egy halmaz neve az egyed típusa, a halmazba tartozó egyedek az előfordulások, megfelelő tulajdonság előfordulásokkal. Ez a szint tartalmazza az egyedtípusok tartalmi leírását, de az adatfüggetlenség érdekében nem tartalmazza az adatok helyét és a hozzáférési

módot. Ugyancsak ez a szint írja le, hogy milyen kapcsolatok vannak az egyedek (rekordok) között. A kapcsolatok leírásának módja az adatmodelltől függ.

- **Előfordulás:** konkrétan létező egyed, tulajdonságérték, kapcsolat.

3.3 Adatfüggetlenség

Mivel ugyanazon adatokat több felhasználó és több program is használhatja, az adatokat és a programokat - amennyire lehetséges - függetleníteni kell egymástól: ha például valamelyik programban a rekordokat kibővítjük egy mezővel, akkor ne kelljen a többi programot megváltoztatni. Az adatok fizikai leírását külön kell választani a programok által látott adatszerkezettől. Ez az adatfüggetlenség. Az adatfüggetlenségnek két fokozata van:

- **fizikai adatfüggetlenségről** beszélünk, ha a programok vagy felhasználók információkérései gyakorlatilag függetlenek az adatok tárolási és elérési módjától;
- **logikai adatfüggetlenségről** beszélünk, ha az adatbázis logikai szerkezetében létrehozott változások az adatbázist felhasználó programokat nem befolyásolják.

A tényleges adatfüggetlenség e kettő együttes megvalósulása.

3.4 Az adatbáziskezelő rendszer részei

- **Kapcsolattartó:** ez lehetővé teszi, hogy más alkalmazások kapcsolatba tudjanak lépni az adatbázissal.
- **Feladat átkonvertáló:** az adatbázis kezelő rendszer nyelvén kapott utasításokat elemzi, értékeli, és ha végrehajthatóak, meghatározza a végrehajtás várható leghatékonyabb módját.
- **Elérés meghatározó:** az objektumok elérésére meghatározott módszert átfordítja az operációs rendszer számára érthető parancsokká. Elindítja, majd értékeli ezen parancsok végrehajtását.

3.5 Az adateleíró nyelv (Data Definition Language)

Séma: az adatmodellnek és a leírásnak további számítógépes feldolgozásra is alkalmas tartalmi és formai megfogalmazása. Az adateleíró nyelv a séma megfogalmazására szolgáló

nyelv. Egy adatbázishoz egy séma tartozik, amely pontosan meghatározza az adatszerkezetet, a tárolási struktúrát, valamint az egyes adatelemek között fennálló logikai kapcsolatokat. Az alsémák mutatják meg, hogy az egyes alkalmazói programok az adatbázist hogyan látják. Egy adatbázis sémához akárhány alséma illeszthető. Más-más alsémákban lehet az adatoknak a típusa, neve, csoportosítása.

Az adatleíró nyelv három szinten írja le az adatokat:

- a felhasználó szintjén, amely a tényleges adatbázis valamilyen alsémáját, az adatmodell egy külső szintjét jelenti.
- az adatok logikai kapcsolatának szintjén, mely a teljes logikai séma, a koncepció modell leírását tartalmazza.
- az adatok fizikai tárolásának meghatározása szintjén, mely az egyes adatok fizikai elérhetőségét, tárolási módját és a köztük fennálló fizikai kapcsolatokat (mutatók, indexek stb.) írja le. A tárolási módot és a fizikai kapcsolatokat rendszerint egy önálló nyelv segítségével valósítjuk meg.

3.6 Az adatkezelő nyelv (Data Manipulation Language)

A feldolgozás céljaira szükség van egy adatkezelő nyelvre. Ennek segítségével a felhasználó az adatleíró nyelvvel definiált adatbázison műveleteket végezhet.

A műveletek vonatkozhatnak az adatbázis adatainak:

- előállítására,
- módosítására,
- lekérdezésére.

Az adatkezelő nyelv által az adatbázis megnyitható, zárható, kulcs vagy cím alapján kereshetünk benne, beolvashatunk, beszúrhatunk, törölhetünk, módosíthatunk rekordokat.

3.7 Az adatbáziskezelő rendszer

Az adatbáziskezelő rendszer tehát egy szoftver, amely biztosítja az adatbázisban tárolt adatok létrehozását, kezelését, valamint leírja és kezeli az adatok közötti komplex

kapcsolatokat. Az adatbáziskezelő rendszernek támogatnia kell valamilyen adatmodellt, hogy a valóságot le tudja képezni, egy számunkra megfogható objektumra. Két részből épül fel:

- **jelölésrendszer:** hogy valamilyen formában a valóságot ábrázolni tudjuk,
- **műveletek halmaza:** amely lehetővé teszi, hogy az előbbieket kezelni tudjuk.

Az adatbáziskezelő rendszernek biztosítania kell:

- különféle igények hatékony kielégítését,
- az adatbázis következetességét, vagyis, hogy csak valós adatokat tároljunk,
- az adatbázisbeli valamennyi adat elérését, (ezt egy file-kezelő rendszer nem teszi meg)
- adatfüggetlenséget, (fizikai és logikai függetlenséget)
- adatok közti komplex kapcsolatok kezelését és ábrázolását,
- redundancia mentességet és annak ellenőrzését (ne szerepeljen az adatbázisban többször is ugyanaz az adat),
- az adatok védelmét
- helyreállíthatóságot,
- több felhasználós rendszerekben az egyidejű hozzáférést (tranzakciókezelés),
- osztott adatbázisokban az adatok szétosztását, megtalálását,
- az adatforgalom optimalizálását.

3.8 Adatmodellek:

- **hierarchikus modell:** az adatbázist irányított gráfként, fastruktúrában ábrázoljuk. A gráf csúcsai az objektumokat, az ágai pedig az objektumok közötti kapcsolatokat jelölik. Ennek az ábrázolásnak az a hátránya, hogy nem minden fajta kapcsolatot tud ábrázolni (például a sok-sok kapcsolatot nem tudja).
- **hálós modell:** ez a modell már megoldotta az előbbi problémát. Az adatokat mutatókon keresztül érhetjük el és ugyanígy reprezentálhatjuk a köztük levő kapcsolatokat is.

Ezt a két modellt többnyire az adott feladatokra írt programokkal valósították meg. Ha olyan kérdés merült fel, ami nem volt beprogramozva, azt nem lehetett megválaszolni.

- **relációs modell:** a 70-es évektől kezdve terjedt el, és ma már szinte minden rendszer ezt támogatja. Jelölésrendszere táblázatokból áll. A kapcsolatok leírására bonyolult matematikai apparátust, relációkat használ. Ezt a modellt használja az SQL nyelv is.

Amikor létrehozunk egy adatbázist, bizonyos szabályoknak kell eleget tenni. Így érhetjük el, hogy az adatmodell mindig a valóságot tükrözze. Ha tényleges változás következik be, akkor az adatbáziskezelő rendszer szerkezetét kell módosítani.” [3.]

3.9 Az SQL nyelv

„Az SQL egy rövidítés, amely "Structured Query Language", jelentése struktúrált lekérdező nyelv. Ma már szabványnak tekinthető, de vannak eltérések az egyes változatok között. Az alapvető parancsok és szintaktikájuk azonban megegyeznek.

A relációs adatbáziskezelés szabványos nyelve. Nem algoritmikus nyelv, de algoritmikus nyelvekbe beépíthető (beágyazott SQL).

Általános jellemzés

Az SQL utasításait két fő csoportba szokták sorolni:

- DDL (= Data Definition Language): adatstruktúra definiáló utasítások.
- DML (= Data Manipulation Language): adatokon műveletet végző utasítások.

Szintaxis

Kisbetű és nagybetű a nyelv alapszávaiban egyenértékű.

Utasítások sorfolytonosan írhatók, lezárás pontosvesszővel.

Változó nincs, csak tábla- és oszlopnevekre lehet hivatkozni. Kifejezésben hivatkozás egy tábla adott oszlopára: tábla.oszlop (ha a tábla egyértelmű, akkor elhagyható).

Alias név: név AS másodnév (egyes implementációkban AS elhagyható).

Szövegkonstans: 'szöveg'

Dátum: DATE '1968-05-12'. Egyes rendszerek az SQL szabványtól eltérő konvenciót alkalmaznak, például 13-NOV-94 (Oracle), 02/15/1994 (dBase).

Idő: TIME '15:31:02.5' (óra, perc, másodperc).

Stringek konkatenációja: + vagy || .

Relációjelek: =, <=, >=, !=, <>

Logikai műveletek: AND, OR, NOT. Az SQL rendszerek "háromértékű logikát" használnak, vagyis a TRUE és FALSE mellett a NULL (definiálatlan) érték is felléphet.

Ha egy kifejezés valamelyik eleme NULL, akkor a kifejezés értéke is NULL lesz.

Az SQL-szabvány szerint egy logikai kifejezés értéke ISMERETLEN (UNKNOWN), ha benne NULL érték szerepel.

Az utasítások szintaxisának leírásánál az elhagyható részleteket szögletes zárójellel jelöljük.” [4.]

3.10 A MySQL

„A MySQL a világon a legelterjedtebben használt nyílt forrású adatbázis-kezelő több millió felhasználóval, az egyéni fejlesztőtől – akik saját honlapjaik készítéséhez használják – az olyan nagyvállalatokig, amelyek nagy forgalmú oldalak adatbázismotorjaként alkalmazzák.

Az utóbbira egy példa a Yahoo! Finance, amely a MySQL-t másodpercenként több száz, naponta milliónyi lekérdezés végrehajtására használja.

A lekérdezés olyan SQL nyelven írt parancscsokor, amit az RDBMS-nek küldünk feldolgozásra: adatbázisok, táblák és rekordok létrehozása, adatok módosítása, törlése vagy az adatbázisban tárolt információk alkalmazásunk számára történő kinyerése céljából.

A MySQL RDBMS (Relational Database Management System) mellett, hogy tárolja tábláinkat, oszlopainkat, sorainkat (és a bennük lévő adatokat), egységes egészsként kezeli őket, lehetővé teszi a felhasználók és a hozzájuk tartozó jogosultságok megadását, naplózza az egyes felhasználók tevékenységeit, és az adott válaszokat lekérdezésekbe (queries) szervezi

Az alkalmazásfejlesztők azért használják előszeretettel a MySQL-t, mert nagyon gyors és messze nem olyan összetett, mint az Oracle, ugyanis a MySQL nyílt forráskódú változata az esetek nagy részében szabadon felhasználható és ingyenes.

3.11 A MySQL legfőbb tulajdonságai

- A gyorsaság,
- a hordozhatóság,
- bármely más nyelven írt felület illeszthetősége,
- kedvező ár.

A MySQL egy többszálú végrehajtást támogató kiszolgáló, ami azt jelenti, hogy minden egyes új kapcsolat létesítésekor elindul egy új kiszolgáló folyamat. A MySQL-hez kötődő kapcsolatok nem osztják meg egymással a folyamataikat, így amikor egy folyamat valamilyen hiba következtében leáll, vagy valamilyen módon túlterheli a kiszolgálót, csak az adott folyamat áll le, és nem az egész kiszolgáló.

Ez a tulajdonság a MySQL sebességére is jótékony hatással van.

A MySQL-t az is népszerűvé teszi a fejlesztők körében, hogy gyakorlatilag minden nyelv – a PHP, a Perl, a C, illetve a C++, a Java, a Python és a Tcl alkalmas arra, hogy felhasználói felületet írjanak hozzá.

Az operációs rendszerek cserélgetése is gyerekjáték: ha bármikor operációs rendszert vagy nyelvet kellene váltanunk, használhatjuk a mysqldump parancsot az adataink kinyerésére és a mysqlimport parancsot arra, hogy az adatokat újra bevigyük a rendszerbe.

További érv a MySQL mellett, hogy ingyenes.” [5]

„A MySQL-lel való kapcsolatteremtés egyik módszere lehet az általunk használt programozási nyelven történő kommunikáció. Például, ha az általunk fejlesztett webalapú alkalmazás PHP nyelven lép kapcsolatba MySQL-lel, akkor a PHP megfelelő függvényeit használhatjuk a MySQL-hez történő kapcsolódáshoz és lekérdezések futtatásához.” [5.]

4. A HTML

„A honlapok a számukra kidolgozott kommunikációs szabványok szerint "közlekednek" az Interneten és ezt értik a **World Wide Web** böngészőprogramok. A lapok szerkesztő- és kiegészítőjelekkel (leírókkal) ellátott szövegállományok. A leírók szabványa a **HTML** (HyperText Markup Language). A HTML egyfajta leírónyelv. Vele a lap megjelenését tudjuk "programozni".” [6.]

„A World Wide Web az emberiség történetében az első olyan médium, amely lehetővé teszi a saját gondolataink átadását minimális költségekkel, elvileg több millió ember számára. Abban különbözik az elterjedt elektronikus médiáktól, hogy alapvetően interaktív, azaz nem csak megengedi a nézőnek a szabad választást az egyes dokumentumok között, hanem lehetővé teszi a nézőből alkotóvá válást is. Ez az Internet hatalmas növekedését eredményezte. Ekkora mennyiségű információ teljességgel áttekinthetetlen, emiatt az oldalak készítői igyekeztek valamivel kitűnni a többi közül. Volt, aki ezt a tartalom mélyítésével, volt, aki a külalak szépítésével próbálta elérni. Ma viszont már csak az számíthat oldalának népszerűségére, aki ezt az előző két dolgot sikeresen kombinálja (logikus tartalom, áttekinthető, tetszetős megjelenés). Ezt teszi lehetővé a HTML nyelv és a JavaScript.”[7.]

4.1 HTML oldalak készítése

„Egy HTML dokumentum minden olyan szövegszerkesztővel elkészíthető, ami képes sima szövegfájlba menteni (pl. NotePad, VI, EMACS, Pascal szerk.). Ugyanakkor kezdésnek ajánlható az ún. WYSIWYG (What You See Is What You Get) típusú színes honlapkészítői környezetek, amelyekkel a felhasználó minden előismeret nélkül elkészítheti saját bemutatkozó oldalát. Előnyük a nagyon könnyű használat, hátrányuk a sablonosság és az alacsonyabb interaktivitás a kézzel írott lapokhoz képest. E két szerkesztő típus mellett

léteznek más HTML szerkesztő programok, amelyek elsőre közönséges karakteres szerkesztőknek tűnnek, ám nagyon hasznos utasításokkal támogatják a HTML dokumentum megírását (pl. HTML elem beillesztése, struktúra szerinti kiemelés, szintaktika ellenőrzés). Ezen programok hatékonyak és gyorsak.” [7.]

5. A PHP

„A PHP nyelv túlnőtt eredeti jelentőségén. Születésekor csupán egy makrókészlet volt, amely személyes honlapok karbantartása készült. Innen ered a neve is: Personal Home Page Tool. Később a PHP képességei kibővültek, így egy önállóan használható programozási nyelv alakult ki, amely képes nagyméretű webes adatbázis-alapú alkalmazások működtetésére is.

A PHP nyelv népszerűsége képességeinek bővülésével folyamatosan nőtt.

A PHP jelenleg hivatalosan a PHP: Hypertext Preprocessor elnevezést használja. Tulajdonképpen kiszolgálóoldali programozási nyelv, amit jellemzően HTML oldalakon használnak. A hagyományos HTML lapokkal ellentétben azonban a kiszolgáló a PHP parancsokat nem küldi el az ügyfélnek, azokat a kiszolgáló oldalán a PHP értelmező dolgozza fel. A programjainkban lévő HTML elemek érintetlenül maradnak, de a PHP kódok lefutnak. A kódok végezhetnek adatbázis-lekérdezéseket, dinamikusan létrehozhatnak képeket, fájlokat olvashatnak és írhatnak, kapcsolatot létesíthetnek távoli kiszolgálókkal a lehetőség száma végtelen. A PHP kódok kimenete a megadott HTML elemekkel együtt kerül az ügyfélhez.

5.1 A PHP fejlődése

A PHP első változatát amely néhány webalkalmazás-készítést segítő makrót tartalmazott – Rasmus Lerdorf készítette 1994-ben. Ezen eszközöket együttesen a Personal Home Page Tools névvel azonosították.

Később, a kód újraírása után, egy friss elem került a csomagba, a Form Interpreter (Űrlapfeldolgozó), így PHP/FI néven vált ismertebbé. A felhasználók szemszögéből a PHP/FI nagyon hasznos segédeszköz volt, így népszerűsége töretlenül nőtt. Több fejlesztő is felfigyelt rá, így 1997-re már számos programozó dolgozott rajta.

A következő kiadás, a PHP 3-as, már egy csapat együttműködéséből született. Ehhez a változathoz a Zeev Zuraski és Andi Gutmans újjáalkotta a teljes feldolgozó-egységet, valamint újabb elemeket és szabályokat adott a nyelvhez. Ez a változat megalapozottá tette a PHP helyét a legjobb kiszolgálóoldali nyelvek között, így felhasználói tábora rendkívüli mértékben nőtt.

Az Apache- és MySQL-támogatás megerősítette a PHP pozícióját. Az Apache jelenleg a legnépszerűbb kiszolgáló a világon és a PHP 3-as már modulként illeszthető hozzá. A MySQL igen hatékony, ráadásul ingyenes SQL adatbázisrendszer, amelyhez a PHP átfogó támogatást nyújt. Az Apache-MySQL-PHP együttes egyszerűen verhetetlen.

Ez természetesen nem jelenti azt, hogy a PHP nem használható más környezetben, más eszközökkel. A PHP számos adatbázis-alkalmazással és web kiszolgálóval képes együttműködni.

A PHP népszerűségének növekedésére hatással volt a webes alkalmazások fejlesztésében történt váltás is. Az 1990-es évek közepén természetesnek számított, hogy akár egy nagyobb webhelyet is több száz, egyenként kézzel kódolt HTML lap felhasználásával készítsenek el.

Mára azonban a fejlesztők egyre jobban kihasználják az adatbázisok nyújtotta kényelmi szolgáltatásokat, hogy a megjelenítendő tartalmat hatékonyan kezeljék és az egyes felhasználóknak lehetőséget adjanak a webhelyek testreszabására.

Egyre gyakoribb adatbázisok használata a tartalom tárolására és az információk visszakeresésére különböző felületeken. Az adatok egy központból több környezetbe is érkezhettek, beleértve a mobiltelefonokat, a digitális személyi titkárokat (PDA), digitális televíziókat és szélessávú internetes rendszereket is.

Ebben a környezetben már nem meglepő, hogy egy ilyen kifinomult és rugalmas nyelv, mint a PHP, ekkora népszerűsége tett szert.” [8.]

5.2 Miért a PHP?

„Van néhány megcáfolhatatlan érv, amiért a PHP-t érdemes választani. Ha más programnyelveket is ismerünk, számos alkalmazás fejlesztése során észlelni fogjuk, hogy a programozási szakasz érezhetően gyorsabb, mint várnánk. A PHP, mint nyílt forráskódú termék jó támogatással rendelkezik, amit a képzett fejlesztői gárda és az elkötelezett közösség nyújt számunkra. Ráadásul a PHP a legfontosabb operációs rendszerek bármelyikén képes futni, a legtöbb kiszolgálóprogrammal együttműködve.

5.3 A fejlesztés sebessége

Mivel a PHP lehetőséget ad a HTML elemek és a programkódok elkülönítésére, az alkalmazások fejlesztésekor lehetőség van elválasztani a kódolási, tervezési, és összeállítási szakaszt. Ez jelentősen megkönnyíti a programozók életét azzal, hogy elmozdítja az akadályokat a hatékony és rugalmas alkalmazások kialakításának útjából.

5.4 A PHP nyílt forráskódú

Számos felhasználó szemében a nyílt forráskódú egyet jelent azzal, hogy ingyenes, ami természetesen már önmagában is előnyös. Talán idegennek hangozhat azok számára, akik nem UNIX-háttérrel olvassák-e sorokat, hogy a PHP nem kerül semmibe. Használható kereskedelmi és/vagy nem kereskedelmi célra, ahogy tetszik.

A jól szervezett nyílt forráskódú projektek újabb előnyökkel szolgálnak a felhasználóknak. Felvehetjük a kapcsolatot a könnyen elérhető és elkötelezett felhasználói közösséggel, ahol

számos nagy tapasztalattal rendelkező embert találunk. Nagy az esély rá, hogy bármilyen problémával is kerüljünk szembe, némi kutatással gyorsan és könnyen választ találunk rá. Ha mégsem, egy levelezőlistára küldött üzenetre általában hamar érkezik intelligens és hiteles válasz.

Úgyszintén bizonyos, hogy a feldolgozóprogram hibáinak javítása nem sokkal felfedezésük után megtörténik és a felmerült új igényeket kielégítő szolgáltatások is hamar beépülnek a nyelvbe. Nem kell várni a következő hivatalos kiadásra, hogy a fejlesztések előnyeit élvezzük. Nincs a PHP működtetéséhez egyedileg kiválasztott kiszolgáló vagy operációs rendszer. Szabadon választhatunk olyan rendszert, amely kielégíti saját vagy ügyfeleink igényeit. Biztos, hogy kódunk továbbra is futtatható lesz, bármi mellett is döntünk.

5.5 Teljesítmény

A hatékony Zend Engine-nek köszönhetően a PHP 4-es jól vizsgázik az ASP-vel szemben végzett méréseken, néhányban megelőzve azt. A lefordított PHP messze maga mögött hagyja az ASP-t.

5.6 Hordozhatóság

A PHP-t alapvetően úgy tervezték, hogy alkalmas legyen számos operációs rendszeren való használatra, együttműködve különböző kiszolgálókkal és adatbáziskezelőkkel. Fejleszthetünk UNIX rendszerre és áttérhetünk NT alapokra minden probléma nélkül. A PHP alkalmazásokat kipróbálhatjuk Personal Web Serverrel és később telepíthetjük azokat egy UNIX rendszerre, ahol a PHP-t Apache modulként használjuk.” [8.]

6. A program bemutatása

6.1 A program készítésének menete

1. A feladat kitalálása
2. Az igények felmérése, a részletek tisztázása
3. A feladat elkészítéséhez szükséges programnyelv(ek) kiválasztása
4. A kapott adatok feldolgozása és adatbázis megtervezése
5. A megfelelő programszerkesztő, valamint adatbázis szerkesztő kiválasztása
6. A kiválasztott programozási nyelvek megismerése
7. A feladat elkészítése
8. Tesztelés, hibafeltárás
9. A hibák javítása, esetleges új funkciók elkészítése, fejlesztés, friss adatok beszerzése
10. A program végleges változatának használatba vétele

6.2 A programnyelvek kiválasztása

A programnyelv kiválasztásánál a PHP nyelvre esett a választás mivel a fentiekben kifejtett előnyök miatt nem volt ellenfele. A PHP mellett szükség volt a HTML alapok megszerzésére, aztán a CSS (*Cascading Style Sheets*) stílusleíró nyelvet használva kialakítottam a design, valamint az SQL manipulációs parancsait használtam a lekérdezések elkészítéséhez.

Ezek mellett szükség volt AJAX-ra is (Asynchronous Javascript and XML) amelynek segítségével sikerült megoldanom többek között a következőket:

- A jelszavak ellenőrzése
- Dinamikus keresés
- A tárgyak feltöltés/ mentés utáni automatikus listába kerülés
- A nyomtatási kép megváltoztatása

6.3 A program

A program röviden és tömören összefoglalva egy weboldal, amely tantárgyakat kezel, amelyben több felhasználói típus létezik. Ezek a személyek különböző jogosultságokkal rendelkeznek.

A tárgyakat az adatbázisba .xls formátumból átkonvertált fájlból tudunk feltölteni, így nincs szükség sok adminisztratív munkára.

Az oldal képes felhasználókat azonosítani. Lehet rendezni, keresni, szűrni, csoportosítani, nyomtatni, szerkeszteni különböző feltételek alapján a tárgylistában, akár törölhetünk is tárgyakat és tudunk archiválni is.

Minden tantárgyhoz tartozik egy tantárgyleírás és 15 hétre lebontva a tervezett órai anyag.

Az adminisztrátor képes új felhasználót hozzáadni vagy a régiek adatait módosítani valamint rendelkezésére áll egy olyan lekérdezés, amellyel ellenőrizhető, hogy melyik tantárgyhoz nem készült még el a leírás.

Ez a weblap segítséget nyújthat az oktatók és az egyetemi dolgozók munkájának megkönnyítésére és emellett a hallgatók megfelelő tájékoztatására.

6.4 Adatbázis felépítése

Az adatbázis neve courseadmin amelynek három táblája van:

- Tantargy
- Leiras
- Felhasznalok

A tantargy tábla szerkezeti felépítése és típusai:

Id	int(10)	Primary key
Tanszek	varchar(64)	
Tipus	varchar(10)	

Szakirany	varchar(10)	
Sorszam	int(11)	
Nev	tinytext	
Kod	varchar(20)	
Teljesites_tipusa	varchar(10)	
Felev	int(2)	
Elmelet	int(2)	
Gyakorlat	int(2)	
Kredit	int(2)	
Oktato	tinytext	
Tagozat	varchar(1)	
Elokovetelmeny	varchar(200)	
Parhuzamos_kod	varchar(48)	
Oktatok	varchar(200)	
Angolcim	varchar(48)	
Rovidcim	varchar(48)	
Idobelyeg	timestamp	

A leiras tábla szerkezeti felépítése és típusai:

Lkod	varchar(20)	
Ltagozat	varchar(1)	
Lnev	varchar(128)	
Lelmelet	int(2)	
Lgyakorlat	int(2)	
Lkredit	int(2)	
e1 - e15	tinytext	
gy1 - gy15	tinytext	

eszamonkeres	tinytext	
gyszamonkeres	tinytext	
vizsgarafelt	tinytext	
teljert	tinytext	
irodalom	tinytext	
<u>lid</u>	int(4)	
targyfeladata	tinytext	
targyleirasa	tinytext	
Pk	int(11)	Primary key

A felhasználók tábla szerkezeti felépítése és típusai:

nev	varchar(200)	
email	varchar(200)	
jelszo	varchar(48)	
id	int(11)	Primary key

Az adatbázis a phpMyAdmin segítségével készült el, ami az XAMPP része és a verziószáma: 3.2.4. Az adatbázis megtervezése során figyelembe vettem a 3. normálforma szabályait a redundancia és más adatbázis problémák elkerülése miatt.

A felhasználói azonosításhoz szükséges jelszógenerálás MD5 függvénnyel történik, ami független a bevitt adat hosszától és típusától, 32 bites hash-t eredményez. A kódolás csak egy irányban történhet, így nem lehetséges visszafejteni. Ezért biztonságos és megfelel az eredetiség ellenőrzésére. [9.]

„Például egy beléptető rendszer nem kell, hogy eltárolja az egyes felhasználók jelszavait, elég ha csak azok vissza nem fejtető 32 karakteres hash-ét rögzíti. A bejelentkezési kísérletnél a megadott jelszó hash-ét összehasonlítva az eredetileg tárolttal megkapjuk annak helyességét.” [9.]

6.5 Az oldal felépítése

Ami első pillantásra feltűnhet az oldallal kapcsolatban az, hogy letisztult és egyszerű designnal rendelkezik mivel a készítés során a fő hangsúly az áttekinthetőségen, használhatóságon és a funkcionalitáson volt és a minél tökéletesebben való működésen nem pedig a csilli-villi külsőn, hiszen nem az fogja meghatározni az oldal eszmei értékét, hanem a hasznossága. Meg persze gondolni kell a fejleszthetőségre is.

A weblap első oldalán bejelentkező rész található, amelyen a felhasználói azonosítás történik. Az oldal tetején az oldal címe található amit Courseadmin-nak neveztem el. Ezen kívül két darab text típusú beviteli mező szerepel rajta, aminek segítségével beírható a felhasználónév valamint a hozzá tartozó jelszó. Aztán van egy submit típusú gomb, amellyel a bejelentkezés hajtható végre. Legalul a lábléc figyelhető meg.



1. ábra: a bejelentkező oldal

Mivel több felhasználó típus van ezért a különböző felhasználói típusok különböző jogosultságokkal rendelkeznek.

6.5.1. Felhasználói típusok:

- Adminisztrátor
- Tanár
- Hallgató

Adminisztrátorként bejelentkezve teljes mértékben használható az oldal, itt minden szerkeszthető, az oldal minden tudása kihasználható. Tanárként bejelentkezve már kevesebb lehetőség van, minden olvasható, de szerkeszteni csak a saját tárgyakat lehetséges.

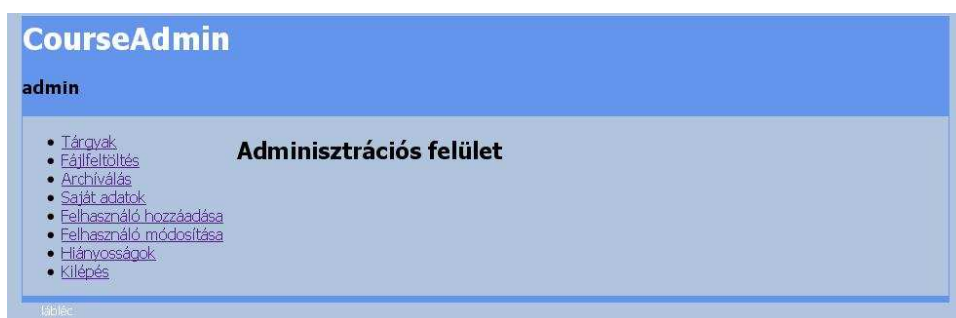
Hallgatóként bejelentkezve a legkorlátozottabbak a lehetőségek, csakis olvasásra jogosult valamint nyomtatásra.

6.6 Bejelentkezés adminisztrátorként

Adminisztrátorként történő bejelentkezés után elérhető az adminisztrációs felület. Ezen a következők találhatók:

- Tárgyak
- Fájlfeltöltés
- Archiválás
- Saját adatok
- Felhasználó hozzáadás
- Felhasználó módosítás
- Hiányosságok
- Kilépés

A cím alatt megjelenik az éppen bejelentkezett felhasználó neve.



2. ábra: adminisztrátorként való bejelentkezés utáni főmenü

Az első link a Tárgyak, amire ha rákattintunk, akkor megjelennek az adatbázisban lévő tárgyak és láthatóvá válnak szűrési lehetőségek.

Ezek a következők:

Tudunk keresni a tárgy neve, tárgykód és oktató szerinti keresésre a tárgylistában, amelyek alapesetben üresek. Jobb oldalt találhatók még a szűréshez használható fülek, amelyekkel a félév, elmélet, gyakorlat, kredit szerint tudunk szűrni a táblában úgy, hogy mindegyikhez két mező tartozik, amikben az elsőnél tudunk választani a kisebb, nagyobb, egyenlő logikai értékek közül, és a második mezőben a 1-től 10-ig számjegyek szerepelnek. Alapesetben mindegyiknél az első mezőben kisebb logikai érték, a másodikban a 10-es számjegy szerepel, mivel így minden tárgy látszik, mert teljesülnek a feltételek, megfelelnek a kritériumoknak.

Középen található még egy tagozat fül, amivel kiválaszthatjuk, hogy nappali vagy a levelezős tárgyakat szeretnénk látni. Alapesetben a **Mind** érték van kiválasztva, hogy mindkét tagozat tárgyai látszódjanak.

The screenshot shows the 'CourseAdmin' interface. On the left is a sidebar with a menu: Tárgyak, Szűrés, Tárgykód, Oktató, Félév, Elmélet, Gyakorlat, Kredit, and Kiválasztás. The main area has a search bar with 'Tárgy neve', 'Tárgykód', and 'Oktató' fields. Below these are filter buttons: 'Félév', 'Elmélet', 'Gyakorlat', and 'Kredit', each with a dropdown menu and a numeric input field. The 'Tárgy neve' field contains 'Mind'. Below the filters is a table with columns: Tárgykód, Név, Tagozat, Megnevezés, Félév, Elm., Gyak., Kredit, Oktató, and Előzetesfelvétel. The table lists several courses, including 'MFABR41ED4', 'MFABR31ED4', 'MFABR41ED4', 'MFABR31ED4', and 'MFABR41X04'.

Tárgykód	Név	Tagozat	Megnevezés	Félév	Elm.	Gyak.	Kredit	Oktató	Előzetesfelvétel
MFABR41ED4	M/E	L	Ábrakészítő geometria I.	1	1	1	4	Nagyné Kondor Rita	
MFABR31ED4	M/E	N	Ábrakészítő geometria I.	1	2	2	4	Nagyné Kondor Rita	
MFABR41ED4	E	L	Ábrakészítő geometria I.	1	2	1	4	Nagyné Kondor Rita	
MFABR31ED4	E	N	Ábrakészítő geometria I.	1	2	2	4	Nagyné Kondor Rita	
MFABR41X04	S/L	L	Ábrakészítő geometria I.	1	1	1	4	Nagyné Kondor Rita	

3. ábra: keresés, szűrés, nyomtatás, tárgylista

6.6.1 Keresés

A keresés úgynevezett dinamikus keresés, ami azt jelenti, hogy amint elkezdjük gépelni a megfelelő mezőbe a keresendő szöveget, akkor azonnal elkezdődik a keresés és az éppen aktuálisan begépeltek alapján megjeleníti a tárgylistában a neki megfelelő tárgyakat. A keresés folyamata nem a szó elején kezdődik, hanem egy begépeltek karakter esetén megjeleníti azokat, amiben szerepel, két vagy több karakter esetében úgyszintén megjeleníti, amiben szerepel a minta. Ez igaz a tárgynév, tárgykód és az oktató esetére is.

6.6.2 Nyomtatás

Az oldalon tudunk nyomtatni akár az egész táblát, akár az éppen aktuális keresés, szűrésnek megfelelő listát is. A nyomtatás földre kattintva nemcsak leegyszerűsített táblázat jelenik meg, hanem segítségnek felugrik egy nyomtatási ablak, amivel egyszerűen kiválaszthatjuk a nyomtatót, példányszámot és mehet is a nyomtatás. Tudunk még a nyomtatási oldalon rendezni is tantárgykód, szak, tagozat, tárgynév, félév, elmélet, gyakorlat, kredit, oktató szerint amennyiben azt nem tettük meg azelőtt, vagy éppen az eredeti beállítás nem felel meg számunkra.

[Vissza](#)

Tantárgykód	Szak	Tagozat	Megnevezés	Félév	Elm.	Gyak.	Kredit	Oktató	Előkövetelmény
MFABR41E04	M/E	L	Ábrázoló geometria I.	1	1	1	4	Nagymé Kondor Rita	
MFABR31E04	M/E	N	Ábrázoló geometria I.	1	2	2	4	Nagymé Kondor Rita	
MFABR41E04	E	L	Ábrázoló geometria I.	1	2	1	4	Nagymé Kondor Rita	
MFABR31E04	E	N	Ábrázoló geometria I.	1	2	2	4	Nagymé Kondor Rita	
MFABR41X04	S/I	L	Ábrázoló geometria I.	1	1	1	4	Nagymé Kondor Rita	
MFABR31X04	S/I	N	Ábrázoló geometria I.	1	2	2	4	Nagymé Kondor Rita	
MFABR41X04	S/SZ	L	Ábrázoló geometria I.	1	1	1	4	Nagymé Kondor Rita	
MFALB41K04	K	L	Alkalmazott biológia	1	2	0	4	Godó Zoltán Anikó	
MFALB31K04	K	N	Alkalmazott biológia	1	3	0	4	Godó Zoltán Anikó	
MFAOT41G03	G	L	Általános géptan	1	1	1	3	Ráthy Istváné	
MFAOT31G03	G	N	Általános géptan	1	2	1	3	Ráthy Istváné	
MFAOT41G03	M/I	L	Általános géptan	1	1	1	3	Ráthy Istváné	

4. ábra: nyomtatás

A **szakok, szakirányok** linkre kattintva megjeleníthető egy alapesetben rejtett rész, amivel megjeleníthetők a szakirányok, amik előtt a jelölőnégyzetet bejelölve szűrhetünk a tárgyak között értelemszerűen a szakok valamint a szakirányok szerint. Ha még egyszer rákattintunk a linkre, akkor az egész eltűntethető.

Szakok, szakirányok

☐ Mind

☒ E Építésztechnika BSC

☐ G Gépipari technika BSC

☐ G/E Gépipari technika BSC Építésztechnika szakirány

☐ G/G Gépipari technika BSC Gépiábrázoló szakirány

☐ G/U Gépipari technika BSC Üzemeltető-kezelő szakirány

☐ K Környezettechnika BSC

☐ K/M Környezettechnika BSC Környezettechnika szakirány

☐ K/T Környezettechnika BSC Környezetmenedzsment szakirány

☐ M/E Műszaki Menedzser BSC (Gazdálkodási mérnök) építőipari szakirány

☐ M/I Műszaki Menedzser BSC (Gazdálkodási mérnök) ipari szakirány

☐ R Mechanika BSC

☐ S/I Építőmérnök BSC Infrastruktúraépítő szakirány

☐ S/SZ Építőmérnök BSC Szerkezetépítő szakirány

5. ábra: elrejtendő menü a szak, szakirány szerinti szűréshez

Van egy **Mind** nevű jelölőnégyzet is, amivel az összezt bejelölhetjük, valamint az összes elől törölhetjük a jelölést. A jelölést megtehetjük akár egyenként is, ha egy szakra lenne szükségünk.

A betűkódok a hozzájuk tartozó szakok, szakirányok a következők:

E	Építészmérnök BSC
G	Gépészmérnök BSC
G/É	Gépészmérnök BSC Épületgépész szakirány
G/G	Gépészmérnök BSC Gépjármű technikai szakirány
G/Ü	Gépészmérnök BSC Üzemeltető- karbantartó szakirány
K	Környezetmérnök BSC
K/M	Környezetmérnök BSC Környezettechnológia szakirány
K/T	Környezetmérnök BSC Környezetmenedzsment szakirány
M/E	Műszaki Menedzser BSC (Gazdálkodási mérnök) építőipari szakirány
M/I	Műszaki Menedzser BSC (Gazdálkodási mérnök) ipari szakirány
R	Mechatronikai mérnök BSC
S/I	Építőmérnök BSC Infrastruktúraépítő szakirány
S/SZ	Építőmérnök BSC Szerkezetépítő szakirány

6.7 A táblázat

A táblázatban tárgyak szerepelnek hozzájuk tartozó adatokkal amelyek a következők:

- Tárgykód
- Szak

- Tagozat
 - nappali vagy levelező
- Megnevezés
- Félév
 - 1-10-ig
- Elmélet
 - 0-4-ig
- Gyakorlat
 - 0-8-ig
- Kredit
- Oktató
- Előkövetelmény

Alapból félév, azon belül pedig a tárgynevek szerinti rendezés van, de tudunk a többi adat alapján is rendezni, csak a nevükre kell kattintanunk. Ha az adatbázisban nem szerepel semmilyen adat, akkor táblázat sem jelenik meg.

Megfigyelhető még a táblázatban hogy a tárgyak nevei egyben linkek is. Abban az esetben, ha valamelyik tárgy nevére kattintunk, akkor a weblap elnavigál minket egy másik oldalra, ahol a tárggyal kapcsolatos részletes adatokkal ismerkedhetünk meg, valamint adminisztrátorként bejelentkezve minden tárgy esetében jogunk van azok szerkesztésére.

6.8 A tárgyleírás és a 15 hetes bontás

A részletes adatok a következők:

- Tantárgy megnevezése
- Angol címe
- Rövid cím
- Tagozat
 - legördülő menüből választhatunk nappali és levelezős között
- Kredit

- Előadás óraszám
- Gyakorlat óraszám
- Tantárgy kódja
- Tanszék
 - legördülő menüből választhatjuk ki, hogy a tárgy melyik tanszékhez tartozik
- Tantárgy felelős
- Kurzus oktatói
- Előkövetelmény kódja
- Párhuzamos kód
- A tantárgy feladata
- A tantárgy leírása
- Vizsgára bocsájtás feltétele
- Teljesítmény értékelése
- Irodalom

Ezekon kívül minden nappali tagozatos tárgyhoz tartozik egy táblázat, ami egy féléves időtartamot ölel fel, minden hétre előírva a kötelezően leadandó anyagot 15 hétre előre, ahol az előadás és a gyakorlat külön kezelhető.

Levelezős tagozatú tárgynál annyi különbséggel találkozhatunk, hogy 15 hét helyett 5 hétre van lebontva a féléves anyag. Megemlíthető még, hogy itt is van lehetőségünk nyomtatásra.

Nyomtatás

tantárgy megnevezése:	Rendszertan		
angol címe:			
rövid címe:		tagozat:	nappali
előadás:	2	gyakorlat:	2
kód:	MFRET31M03		
tanszék:			
tantárgy felelős:	Husi Géza		
kurzus oktatói:			
előkövetelmény kódja:		párhuzamos kód:	

A tantárgy feladata:		
A tantárgy leírása:		

hét	Előadás	Gyakorlat
1		
2		

6. ábra: tárgyleírás és 15 hetes bontás

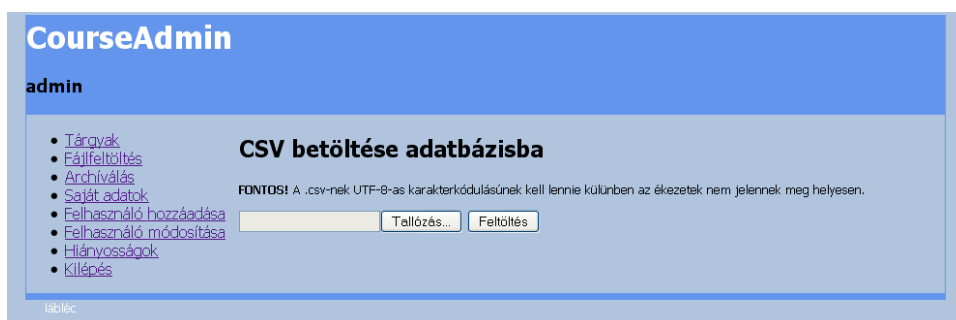
A táblázatok alatt található egy submit típusú **Mentés** gomb amelyre rákattintva elküldi az oldal a módosításokat és bekerül az adatbázisba. Ekkor egy üzenet értesít a sikeres mentésről. Abban az esetben, ha újra ugyanennek a tárgynak szeretnénk módosítani az adatait akkor már láthatjuk az előzőleg bevitt adatokat, mivel azok az adatbázisból kiolvasásra kerülnek a tárgy megnyitásakor és a szövegmezőben szerkeszthetőek.

Ha két vagy több tárgynál megegyezik a tárgykód, tagozat, név, elmélet, gyakorlat, kredit és az egyik tárgynak a leírásban kitöltöttük a mezőket akkor a többinél is szerepelni fognak az adatok mentés után.

A Mentés gombtól jobbra található még egy **Törlés** gomb is, amivel a nemcsak a tárgy leírását tudjuk törölni, hanem a teljes tárgylistából eltávolításra kerül az aktuális tárgy minden hozzátartozó adattal együtt.

6.9 A fájlból való feltöltés

Adminisztrátorként bejelentkezve az oldalra a második menüpont a fájlfeltöltés, amit a 7. ábrán láthatunk.



7. ábra: fájlfeltöltés

Itt tudjuk megadni azon fájl elérési útját, amiből az adatokat szeretnénk feltölteni az oldalra a **Tallózás** gomb segítségével, de megemlíthető, hogyha azonnal a szövegmezőbe kattintunk, akkor már rögtön megkereshetjük a fájlt és siker esetén a **Feltöltés** gomb megnyomásával elindul az importálás. Ilyenkor az adatok bekerülnek az adatbázisba, majd amikor a tárgyakat nézzük, kilistázódnak. A sikeres feltöltésről itt is üzenetet kapunk, amennyiben nem adtuk

meg az elérési utat és kattintunk a feltöltés gombra, akkor hibaüzenetet kapunk arról, hogy fájl nincs megadva.

Fontos megjegyezni, hogy csak a megfelelő formátumú szerkezeti felépítésű és karakterkódolású Microsoft Excel táblázatból tudunk feltölteni adatokat.

A kritériumok a következők:

A szerkezeti felépítés alatt azt értem, hogy a felesleges, üres sorokat amelyek esetleg a táblában vannak ki kell venni, a szakok nevét tartalmazó felső sort is, mivel erre nincs szükség a továbbiakban mert az azonosítás a szak/szakirány kódja alapján történik, aztán az olyan sorokat is törölni kell amelyek nem a tárgyakat és a hozzá tartozó adatokat tartalmazzák.

Ezen kívül nagyon fontos, hogy az excel táblánkat lementsük .csv formátumba amit egyszerűen a Microsoft Excelben a Fájl menü/ Mentés másként/ és a fájltypusnál kiválasztva a CSV formátumot aztán a Mentés gombra kattintva tehetünk meg. Ezután nem maradt más hátra, hogy a karakterkódolást is beállítsuk. Ezt a legegyszerűbben úgy tehetjük meg, hogy megnyitunk egy Jegyzetömböt amit a Microsoft Windowsok alapból tartalmaznak, amivel a korábban létrehozott CSV formátumú fájlt megnyitva, majd itt is Fájl menü/ Mentés másként lehetőséget választva, a Kódolás mezőnél az UTF-8-at választva lementjük a megfelelő helyre. Ha esetleg alapból nem látszana a CSV formátumú fájl, akkor Fájltypusnál a Minden fájlt kiválasztva láthatóvá válik.

Ha ezeken mind túlvagyunk, akkor már megtörténhet az adatok feltöltése és ezután az új adatoknak a tárgylistában kell szerepelnie.

Amennyiben már vannak feltöltve adatok az adatbázisba és újra fel akarunk tölteni tárgyakat, akkor az olyan tárgyak, amelyek már szerepelnek felülíródnak az újakkal, így biztosítva azt, hogy ne szerepeljen többször ugyanaz a tárgy.

A dolgozatomhoz csatolok egy tárgylista sablont, ami valószínűleg nagyban megkönnyíti a munkát és egyértelművé teszi a dolgokat. A .csv formátumról és az utf8 karakterkódolás jelentőségét a következőkben fejtem ki.

6.9.1 CSV formátum:

„Gyakran használt adatcsere fájlformátum. Az angol CSV rövidítés jelentése: vesszővel elválasztott értékek (comma separated values).

A legtöbb táblázatkezelő szoftver (pl. Microsoft Excel) képes beolvasni CSV fájlt illetve képes elmenteni az adatokat ebben a formátumban.

MailMaster e-mail listába történő importáláskor ebben a formában kell feltölteni az e-mail lista tagjainak adatait a szoftverbe. Illetve e-mail lista elemeinek exportálásakor ebben a formátumban kerülnek exportálásra az elemek.” [10.]

6.9.2 UTF-8 karakterkódolás: 8-bit Unicode Transformation Format

„Az UTF-8 (8 bites Unicode átalakítási formátum) lényege, hogy a különböző karakterek Unicode szabványon alapuló kódolásakor változó hosszúságú egységet használ. A leggyakoribb, ékezet nélküli betűk és egyéb alap írásjelek esetén csak egy bájtot használ fel, míg például az európai nyelvek által használt ékezetes betűk esetén két bájtban tárolja el az információt. Az UTF-8 kódolás használata lehetővé teszi több, különböző kódlapot használó nyelv együttes használatát ugyanabban a dokumentumban. Ez nemcsak azon ritka esetekben nélkülözhetetlen, amikor például kínai írásjeleket szeretnénk magyar szövegbe illeszteni, hanem például egyes nyugat-európai karakterek és magyar karakterek együttes használatakor is elengedhetetlen, ezért egy modern magyar nyelvű weboldal létrehozásakor tanácsos ilyen karakterkódolást választani.

Ha azt látjuk egy dokumentumban, hogy az ékezet nélküli betűk jól mutatnak, míg az ékezetes betűk helyett két furcsa karakter jelenik meg (például így: Ā□rvĀztĹ±rĹ‘ tĀĹkĀ¶rfĀşrĀĹgĀ©p), akkor át kell váltani a karakterkódolást UTF-8-ra. (Sajnos néhány alkalmazás nem, vagy nem megfelelően támogatja ezt a szabványt.)” [11.]

6.10 Archiválás

Harmadik menüpontként az archiválás szerepel, amivel az egész adatbázis tartalmát le tudjuk menteni. Ennek segítségével az adatvesztések elkerülhetők. A 8. ábrán az archiválás menüpont látható.



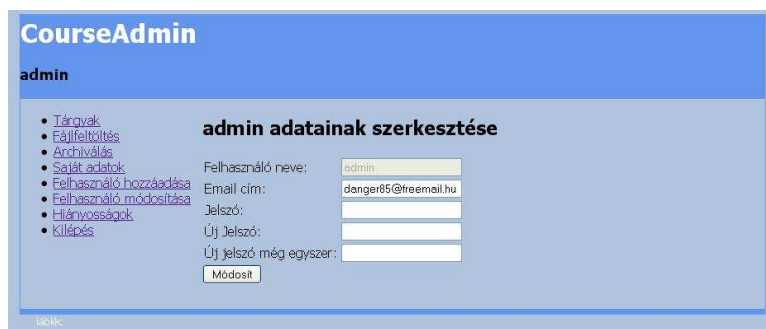
8. ábra: archiválás

Tudunk exportálni fájlba és importálni is. Exportálás esetén az egész adatbázis tartalma egy .sql kiterjesztésű dokumentumba kerül, amit egy felugró ablak segítségével megnyithatunk vagy lementhetünk. Ez a dokumentum is utf-8 karakterkódolású. A fájl nevében az aktuális dátum szerepel letöltéskor, ami segíti a beazonosítást.

Importálás esetén a tallózás gomb segítségével meg kell adnunk a fájl elérési útját, majd a feltölt gombra kattintva az adatok bekerülnek az adatbázisba, de előtte minden adat törlődik a táblákból.

6.11 Saját adatok

Adminisztrátorként bejelentkezve a negyedik menüpont a Saját adatok ezt a 8. ábrán láthatjuk.



9. ábra: saját adatok

Az oldal ezen részén a következők helyezkednek el:

- Felhasználó neve
- Email cím
- Jelszó
- Új jelszó
- Új jelszó még egyszer

Mindegyikhez tartozik egy beviteli mező, amibe beírhatjuk a bejelentkezéshez szükséges adatainkat. Ami első pillantásra feltűnhet a képről azaz, hogy a Felhasználó neve című mezőbe nem tudunk írni semmit, mivel ezt automatikusan kitölti a program, amit egy lekérdezés során az adatbázisból ér el. Ugyan ez igaz az alcímre is ott is az aktuálisan bejelentkezett felhasználó neve szerepel. Az Email cím megváltoztatható, de alapesetben ki van töltve. A másik három mező alapesetben üres, itt tudjuk megváltoztatni a jelszót, amennyiben az adminisztrátor által regisztrált nem felelne meg. Ahhoz, hogy megtudjuk változtatni a jelszavunkat, először az eredetit kell beírni, majd az Új jelszó mezőbe a kívánt új jelszót adhatjuk meg. Annak érdekében, hogy esetleges elírás miatt ne tudjunk bejelentkezni felhasználónevünkkel és az új jelszavunkkal ezért az ellenőrzés kedvéért szükségét láttam egy Új jelszó még egyszer mezőnek is, amivel megerősítjük, hogy szándékosan azt a karaktersorozatot akarjuk bevinni.

Az oldal alján a **Módosít** gombbal el tudjuk végezni a műveletet. Sikeres módosításról üzenetet kapunk.

Sikertelen a módosítási művelet akkor ha:

- valamelyik mező üresen marad
- ha a régi jelszó és az új jelszó megegyezik
- ha az új jelszó és az új jelszó még egyszer mező tartalma különbözik

6.12 Felhasználó hozzáadása

Adminisztrátorként jogunk van új felhasználó hozzáadásához. A 9. ábrán látható az oldal tartalma.



The screenshot shows a web interface titled 'CourseAdmin' with a sub-header 'admin'. On the left is a sidebar menu with links: 'Tárgyak', 'Fájlfeltöltés', 'Archiválás', 'Saját adatok', 'Felhasználó hozzáadása', 'Felhasználó módosítása', 'Hiánycsúszások', and 'Kilépés'. The main content area is titled 'Új felhasználó hozzáadása' and contains a form with the following fields: 'Felhasználó neve:', 'Email cím:', 'Jelszó:', and 'Jelszó még egyszer:'. Each field has a text input box. Below the inputs is a 'Hozzáadás' (Add) button. At the bottom left of the interface, there is a small 'Lapok:' label.

10. ábra: felhasználó hozzáadása

Az oldal tartalma:

- Felhasználó neve
- Email cím
- Jelszó
- Jelszó még egyszer

Sikeres művelethez minden mezőt ki kell tölteni és a jelszó, jelszó még egyszer mezőnek meg kell egyeznie. Ezután a Hozzáadás gombra kattintva létrehoztuk az Új felhasználót. További feltétel még, hogy a regisztrált e-mail cím ne szerepeljen az adatbázisban. A felhasználónév vizsgálatától eltekintettem mivel megeshet, hogy két azonos felhasználó regisztrálására van szükség az oldalon.

6.13 Felhasználó módosítása

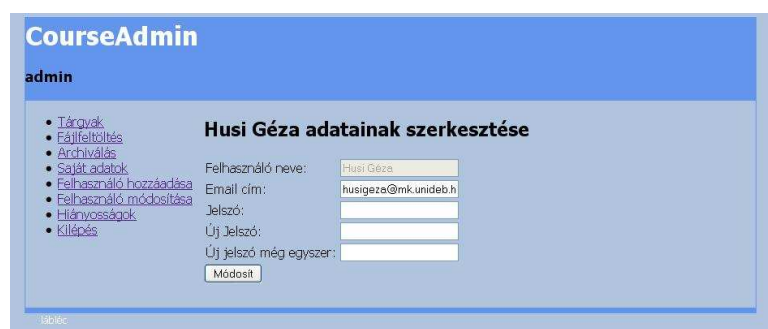
A felhasználó módosítása fül alatt listázhatjuk az oldalra regisztrált felhasználókat, amit adminisztrátorként képesek vagyunk szerkeszteni vagy szükség esetén felhasználót törölni.



11. ábra: felhasználó módosítása

A törlést választva egy felugró ablakban rákérdez a program hogy biztosak vagyunk-e abban, hogy töröljük a felhasználót. Két választásunk van: igen vagy nem.

A szerkesztésre kattintva ugyanaz az oldal ugrik fel, amit már láthattunk a Saját adatok oldalnál, csak az éppen kiválasztott felhasználó neve és adatai szerepelnek benne.



12. ábra: szerkesztésre kattintva

Jelen esetben, mint látható összesen két darab felhasználó van regisztrálva az oldalon. Az adminisztrátori felhasználó azért nem látszik, mert azt sohasem akarhatjuk törölni, szerkeszteni viszont tudjuk a Saját adatok link alatt.

6.14 Hiányosságok

Az utolsó előtti link a Hiányosságok. Ezen belül láthatja az adminisztrátor azokat a tárgyakat kilistázva egy táblázatban, amelyeknek van kitöltetlen mezőjük a tárgy leírásában vagy a 15 hetes bontásban. A Hiányosságok táblázatát a 12. ábrán láthatjuk.

CourseAdmin

admin

- Tárgyak
- Fájlfeltöltés
- Archiválás
- Saját adatok
- Felhasználó hozzáadása
- Felhasználó módosítása
- Hiányosságok
- Kitöltés

Nyomtatás

Hiányosan kitöltött tantárgyak - 2010.05.02.

Tárgy neve:

Féle:

kisebb

10

Tárgykód:

Elsőlet:

kisebb

10

Okozó:

Gyakorlat:

kisebb

10

Tagozat:

Mind

Kredit:

kisebb

10

Szakok, szakirányok

Tantárgykód	Szak	Tagozat	Tárgy neve	Féle	Elsőlet	Gyakorlat	Kredit	Oktató	Státusz
MFABR41E04	M/E	L	Ábrázoló geometria I.	1	1	1	4	Nágmé Kondor Rita	Hiányos
MFABR31E04	M/E	N	Ábrázoló geometria I.	1	2	2	4	Nágmé Kondor Rita	Hiányos
MFABR31E04	E	N	Ábrázoló geometria I.	1	2	2	4	Nágmé Kondor Rita	Hiányos
MFABR41E04	E	L	Ábrázoló geometria I.	1	2	1	4	Nágmé Kondor Rita	Hiányos
MFABR31X04	S/I	N	Ábrázoló geometria I.	1	2	2	4	Nágmé Kondor Rita	Hiányos
MFABR41X04	S/I	L	Ábrázoló geometria I.	1	1	1	4	Nágmé Kondor Rita	Hiányos

13. ábra: a hiányosan kitöltött tantárgyak listája

Összehasonlítva a Hiányosságok listáját valamint a Tárgyak listáját feltűnhet, hogy sok közös tulajdonsággal rendelkeznek. Itt is él a keresés, szűrés ugyanazok a szempontok alapján, mint azzal már megismerkedhettünk a Tárgyak esetében. Szintén szűrhetünk Szakok szerint is a lenyitható menürészben. Nyomtatás is továbbra is a rendelkezésünkre áll. Az egyik különbség azaz, hogy megjelent a cím mellett egy dátum, ami nyomtatásnál is szerepel és ha kinyomtattuk, akkor későbbiekben is tisztában lehetünk azzal, hogy melyik tanárnak pontosan mikor nem volt kitöltve valamelyik tárgyához tartozó leírás. A legfőbb különbséget a táblázat utolsó oszlopában láthatjuk ahol megjelent a Státusz.

Ennek a Státusznak minden tárgynál két lehetséges értéke lehet:

- Üres
- Hiányos

Amennyiben egy tárgynak minden mezője ki van töltve a leírásban, akkor nem kerül be a hiányosságok táblázatába.

Hiányos értéket kap a tárgy akkor, ha tantárgy leírásában és a 15 hetes bontásban a fontosabb mezők egy része nincs kitöltve. Ha egyik sincs kitöltve, akkor az üres státuszt kapja meg a tárgy.

A tárgy leírásában szereplő párhuzamos kódot nem veszi figyelembe a program. Erre is csak azért volt szükség, mert a rendelkezésekre álló adatokban nem szerepelt, és ha sok tárgynál csak ez az egy mező hiányzott volna a teljes kitöltöttséghez, ahhoz hogy ne szerepeljen a hiánylistában, akkor is szerepelt volna és így értelmét veszítette volna az egész lista.

6.15 Kilépés

Az utolsó link a kilépés. Ekkor bontódik a kapcsolat és visszatérünk a bejelentkező oldalra.

A biztonság megköveteli azt, hogy ha a felhasználó belép a jelszavával és a felhasználónevével, akkor a forráskódban induljon egy „session” amit a kilépéskor megszüntetünk, bontunk. Ez arra szolgál, hogyha a felhasználó elmentette kedvencei közé tesszük fel a tárgylista linkjét, akkor ezzel a „session”-nel meg tudjuk akadályozni, hogy elérje bejelentkezés nélkül.

6.16 Bejelentkezés oktatói jogosultsággal

Amennyiben oktatóként jelentkezőnk be, akkor már sokkal kevesebb lehetőségünk van az oldalon.

A főmenü egyszerűsített:

- Tárgyak
- Saját adatok
- Kilépés

Tantárgykód	Szak	Tagozat	Megnevezés	Félév	Elm.	Gyak.	Kredit	Oktató	Előkövetelmény
MFABR41E04	M/E	L	Ábrázoló geometria I.	1	1	1	4	Nagyvári Kondor Rita	
MFABR31E04	M/E	N	Ábrázoló geometria I.	1	2	2	4	Nagyvári Kondor Rita	
MFABR31E04	E	N	Ábrázoló geometria I.	1	2	2	4	Nagyvári Kondor Rita	
MFABR41E04	E	L	Ábrázoló geometria I.	1	2	1	4	Nagyvári Kondor Rita	
MFABR31X04	S/I	N	Ábrázoló geometria I.	1	2	2	4	Nagyvári Kondor Rita	
MFABR41X04	S/I	L	Ábrázoló geometria I.	1	1	1	4	Nagyvári Kondor Rita	
MFABR41X04	S/2	L	Ábrázoló geometria I.	1	1	1	4	Nagyvári Kondor Rita	
MFALB31X04	K	N	Alkalmazott biológia	1	3	0	4	Gódi Zoltán Attila	

14. ábra: oktatóként bejelentkezve

6.17 Tárgyak

A tárgyak menüben minden elérhető, látható, szűrhető ugyanúgy, mint adminisztrátorként bejelentkezve, amik már részletesen kifejtésre kerültek fentebb. Minden oktató láthatja az összes szak összes tárgyát, olvashatja, nyomtathatja viszont szerkeszteni a tárgyleírásban csak a saját tárgyait tudja. Az oktatók a saját nevükkel kell, hogy bejelentkezzenek, mivel az alapján történik az azonosítás és ahhoz, hogy szerkeszthessék a tárgyaik leírásait az elengedhetetlen feltétel.

The image shows two side-by-side screenshots of a web application interface for printing subject details. Both screenshots have a title 'Nyomtatás' (Print) in the top left corner.

The left screenshot is for a subject named 'Rendszertan'. It contains the following fields:

- tantárgy megnevezése: Rendszertan
- angol címe:
- rövid cím:
- szint: levelezős (v) kredit: 3
- előadás: 1 gyakorlat: 2 kód: MFRET41M03
- tanész:
- tantárgy felelős: Husi Géza
- kurzus oktatói:
- előkövetelmény kódjai:
- párhuzamos kód:

 Below these fields is a table for 'a tantárgy feladatai' (tasks of the subject) and 'a tantárgy leírása' (description of the subject). At the bottom is a table with columns 'Hét' (Week), 'Előadás' (Lecture), and 'Gyakorlat' (Exercise).

The right screenshot is for a subject named 'Ábrázoló geometria I'. It contains the following fields:

- tantárgy megnevezése: Ábrázoló geometria I
- angol címe:
- rövid cím:
- szint: Levelező kredit: 4
- előadás: 1 gyakorlat: 1 kód: MFABR41E04
- tanész:
- tantárgy felelős: Nagyné Kondor Rita
- kurzus oktatói:
- előkövetelmény kódjai: ** párhuzamos kód:

 Below these fields is a table for 'a tantárgy feladatai' (tasks of the subject) and 'a tantárgy leírása' (description of the subject). At the bottom is a table with columns 'Hét' (Week), 'Előadás' (Lecture), and 'Gyakorlat' (Exercise).

15. ábra: összehasonlításképpen ahogyan egy oktató saját és más oktató tárgyának leírását látja

6.18 Saját adatok

A saját adatokban semmi különbség nincs, adminisztrátorként és oktatóként is minden ugyanúgy működik, ugyanazok a lehetőségek. Jelszó és e-mailcím módosítást tudunk végrehajtani.

The image shows a screenshot of a web application interface for editing user data. The page title is 'CourseAdmin'. The user name 'Husi Géza' is displayed at the top. Below it, there is a list of links: 'Tárgyak', 'Saját adatok', and 'Kilépés'. The main heading is 'Husi Géza adatainak szerkesztése' (Editing Husi Géza's data). The form includes the following fields:

- Felhasználó neve: Husi Géza
- Email cím: husigeza@mk.unideb.h
- Jelszó:
- Új jelszó:
- Új jelszó még egyszer:

 A 'Módosít' (Modify) button is located at the bottom of the form.

16. ábra: saját adatok oktatóként

6.19 Kilépés

A kilépés linkre kattintva visszatérünk a bejelentkező oldalra.

6.20 Bejelentkezés hallgatói jogosultsággal

Ez a bejelentkezési forma nyújtja a legkevesebb lehetőséget, de az információk megosztására tökéletesen megfelel. Kielégítheti a hallgatói igényeket.

A hallgatók a tantárgyaikkal kapcsolatban mindenféle információhoz hozzájuthatnak. Félévről-félévre tisztán láthatják, hogy melyik tárgy milyen fontossággal bír, ugyanis nyomon követhetik a tantárgyaik előkövetelményeit, megtudhatják melyik órán, melyik héten mit fognak tanulni, megismerkedhetnek az egyetemi előmenetelükkel.

Kereshetnek, különböző feltételek alapján szűrhetnek, és ami lényeges, nekik is kiváltságaik közzé tartozik a nyomtatás.

A saját adatokban nem tudnak sem felhasználó nevet sem jelszót változtatni, mivel ez egy úgynevezett gyűjtői felhasználónév és jelszó, amivel minden hallgató rendelkezhet és szükség esetén használhat.

A hallgatóként bejelentkezéshez szükséges adatok:

Felhasználónév: hallgato

Jelszó: hallgato

A program, weboldal futtatásához szükséges ajánlott rendszerkövetelmények:

Kliens:

Hardver:	Intel Pentium IV. CPU
	512 MB DDR RAM
	128 MB-os VGA
Szoftver:	Microsoft Windows XP

Mozilla Firefox

Valamint nélkülözhetetlen az internet hozzáférés.

Szerver:

Hardver:	Intel Core2Duo CPU
	4096 MB DDR RAM
	512 MB-os VGA
Szoftver:	Microsoft Windows XP
	Mozilla Firefox

Mivel szerver oldali szolgáltatásról van szó ezért célszerű nagy teljesítményű gép, mert egy időpillanatban nagyon sok felhasználó lehet jelen az oldalon, és így a használhatóságot erősen befolyásolja a szerver gép terheltsége. Mondhatni csak attól és az internetkapcsolattól függ, hogy mennyire zökkenőmentes a munka.

A weblap Mozilla Firefox böngészőre van optimalizálva, de működik más böngészőkkel is.

A program készítése során XAMPP programcsomag 1.7.3 változata volt a gépemre telepítve, amely egyben tartalmazza az Apache Webszerver, MySQL adatbáziskezelő rendszert és a PHP 5-ös változatát. A forráskód szerkesztéséhez az EditPlus 3.10-es verzióját használtam és böngészőnek a Mozilla Firefox-ot részesítettem előnyben.

Mivel a weblapot a böngészővel localhost-on el tudtam érni, így nem volt szükségem szerverbérletre vagy valamilyen ingyenes szolgáltató által nyújtott tárhelyre.

6.21 További fejlesztési lehetőségek

- Új funkciókkal való bővítés
- Modernebb, animált design kialakítása

7. Összefoglalás

A munkám során rengeteg tapasztalatra tehettem szert, amit a jövőben kamatoztathatok. Megismerkedhettem a PHP, AJAX, SQL, HTML nyelvekkel, a velük kapcsolatos hasznos fogásokkal, trükkökkel, adatbázis tervezéssel.

Fő célom az volt, hogy tanulhassak ebben a témában és bővítsem tudásomat a programozás területén, valamint nagyobb rálátást szerezzek a weboldal készítés tekintetében.

Úgy érzem a célomat elértem mivel az elvárásoknak megfelelttem, bár fejlődésre mindig van lehetőség.

Remélem, hogy munkámmal hozzájárultam a tanárok, irodai dolgozók, adminisztrátorok munkájának megsegítéséhez és a diákok egyszerűbb tájékoztatásához.

Munkám során nagy segítséget nyújtott az Adatbázisrendszerek című órán tanult ismeretek, amelyet a tervezés során használtam fel.

8. Irodalomjegyzék

[1.] Dr. habil. Boda, I.(2002.) Az Internet fogalma

Letöltés dátuma: 2010.04.29., forrás:

http://www.inf.unideb.hu/~bodai/internet/internet_alapok.html

[2.] DiNero Internet Stúdió (2006.) Letöltés dátuma: 2010.04.29., forrás:

<http://www.dinero.hu/html/internet1.htm>

[3.] Nagy, A. Az adatbáziskezelés alapjai Letöltés dátuma: 2010.04.29., forrás:

<http://www.kfki.hu/chemonet/hun/eloado/adatb/adatb2.html#db1>

[4.] dr. Katona, E. (2009) Adatbázisok Előadási jegyzet (BSc) Szegedi Tudományegyetem Informatikai Tanszékcsoport Letöltés dátuma: 2010.04.29., forrás:

<http://www.inf.u-szeged.hu/~katona/db-ea1.pdf>

[5.] Julie C M. (2001.) Tanuljuk meg a MySQL használatát 24 óra alatt

Kiskapu Kiadó, 7-12.

[6.] Muszka, P.(2006.) HTML alapok.pdf, 1.

[7.] Dvorák, P. and Hórcsik, L. (1998.) Rövid betekintés a Html 3.2-be és a JavaScriptbe
Html.doc, 1.

[8.] Zandstra, M. (2001.) Tanuljuk meg a PHP4 használatát 24 óra alatt, Kiskapu Kiadó
Az első lépések.pdf, 1-9.

[9.] Wikipedia (2010.) MD5 Letöltés dátuma: 2010.04.29., forrás:

<http://hu.wikipedia.org/wiki/Md5>

[10.] CSV formátum Letöltés dátuma: 2010.04.29., forrás:

<http://www.marketingszoftverek.hu/tudasbazis/marketingszoftverek-fogalmak/csv-formatum>

[11.] UTF-8 Letöltés dátuma: 2010.04.29., forrás:

http://webni.innen.hu/Karakterk_c3_b3dol_c3_a1s

9. Ábrák jegyzéke

1. ábra: a bejelentkező oldal
2. ábra: adminisztrátorként való bejelentkezés utáni főmenü
3. ábra: keresés, szűrés, nyomtatás, tárgylista
4. ábra: nyomtatás
5. ábra: elrejthető menü a szak, szakirány szerinti szűréshez
6. ábra: tárgyleírás és 15 hetes bontás
7. ábra: fájlfeltöltés
8. ábra: archiválás
9. ábra: saját adatok
10. ábra: felhasználó hozzáadása
11. ábra: felhasználó módosítása
12. ábra: szerkesztésre kattintva
13. ábra: a hiányosan kitöltött tantárgyak listája
14. ábra: oktatóként bejelentkezve
15. ábra: összehasonlításképpen ahogyan egy oktató saját és más oktató tárgyának leírását látja
16. ábra: saját adatok oktatóként

10. Köszönetnyilvánítás

Szeretném megköszönni Dr. Husi Gézának, hogy vállalta szakdolgozatom témavezetői szerepkörét, és építő jellegű tanácsival, észrevételeivel felügyelte és segítette munkámat.