

# Egyedi CMS rendszer bemutató oldalak számára

*Forrai Művészeti Szakközépiskola és Gimnázium  
a Magyarországi Metodista Egyház Fenntartásában*

Gelencsér Gergő

– Budapest, 2014 –

# Nyilatkozat

Alulírott *Gelencsér Gergő*, a Forrai Művészeti Szakközépiskola és Gimnázium tanulója kijelentem, hogy jelen szakdolgozatot meg nem engedett segítség nélkül, saját magam készítettem, és a szakdolgozatban csak a megadott forrásokat használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

.....  
*Gelencsér Gergő*

# Tartalomjegyzék

1.	Témakiírás, az alkalmazás általános specifikációja.....	5
2.	Bevezetés.....	6
2.1.	A feladat értelmezése, indokoltsága .....	6
2.2.	Alternatívák kritikai elemzése, összehasonlítása .....	7
2.3.	A szakdolgozat felépítése .....	7
3.	Fejlesztői dokumentáció.....	8
3.1.	Tervezés, rendszerterv .....	8
3.1.1.	Funkciók meghatározása .....	8
3.1.2.	Felhasználói felület és menü .....	9
3.1.3.	Funkciók kifejtése .....	9
3.1.4.	Adatbázis séma .....	12
3.2.	Megvalósítás.....	13
3.2.1.	Környezet specifikációja .....	13
3.2.2.	Adatbázis elkészítése .....	13
3.2.3.	Az alap rendszer implementálása.....	14
3.2.4.	Az adminisztrációs felület implementálása .....	18
3.3.	Verifikáció és validáció .....	20
3.3.1.	A tervezett és megvalósított alkalmazás vizsgálata .....	20
3.3.2.	A követelményeknek való megfelelés igazolása.....	21
4.	Felhasználói dokumentáció .....	22
4.1.	Telepítési útmutató .....	22
4.2.	Bejelentkezés, elfelejtett jelszó .....	23
4.3.	A felület használata .....	24
4.4.	Adminisztrációs műveletek.....	25
4.4.1.	Beállítások módosítása .....	25
4.4.2.	Oldalak kezelése .....	26
4.4.3.	Fájlok kezelése .....	27
4.4.4.	Bővítmények kezelése .....	27
4.4.5.	Felhasználók kezelése .....	28
4.5.	A rendszer karbantartása .....	29
5.	Összefoglalás, továbbfejlesztési lehetőségek.....	30
6.	Irodalomjegyzék, ábrajegyzék, mellékletek .....	31

*„A MicroSite, vagy bemutatkozó oldal, napjaink talán egyik legfontosabb marketing eleme. Lényegében egy olyan négy- öt oldalból álló mini honlap, mely alkalmas vállalkozások, magánszemélyek, tevékenységek, stb. bemutatkozására, illetve egy-egy termék, szolgáltatás, vagy nyereményjáték promótálására. A MicroSite-ok fő jellemzője, hogy nagyon kevés, de jól felépített, látványos oldalból állnak. Elsődleges szempont a könnyű áttekinthetőség.”*

*– WebMake.hu*

# 1. Témakiírás, az alkalmazás általános specifikációja

Jelen szakdolgozat témája egy saját fejlesztésű tartalomkezelő rendszer (CMS) építése bemutatkozó oldalak számára, a továbbiakban részletezett szempontok szerint.

A CMS-nek tudnia kell szétválasztani a tartalmat a megjelenéstől, így a különböző felületek lesznek alkalmazhatók a teljes honlapra. A felületek szabványos HTML és CSS fájlok csomagjaként telepíthetők és eltávolíthatóak a rendszerből. Néhány előre elkészített felület a CMS telepítőjének részét képezi majd, illetve újak készítéséhez elérhető lesz egy sablon felület csomag, benne megjegyzésekkel a kötöttségekre vonatkozóan.

A rendszer az egyes oldalak tartalmát (cím, leírás, kulcsszavak, tartalom) az adatbázisban, a megfelelő táblában fogja tárolni rekordonként. A tartalom mező természetesen bármilyen hivatkozást, képet, videót és egyéb HTML tag-et tartalmazhat. Ezek elhelyezését az adminisztrációs felületen úgynevezett WYSIWYG szerkesztő segíti.

Különböző kiegészítők beillesztését, mint szavazás, galéria, komment, és egyéb dinamikus modulok, szintén lehetővé kell tenni a rendszernek. Ezek a kiegészítők a felületekhez hasonlóan egy csomagként telepíthetők és eltávolíthatóak. A csomag egy PHP fájlon keresztül, benne egy származtatott osztállyal, kommunikál a CMS-sel, ezzel biztosítva, hogy az ne léphesse túl a hatáskörét. Néhány kiegészítő és egy minta állomány szintén integrálva lesz a rendszerben.

Az adminisztrációs felület elkülönítve lesz a CMS magjától, így az az adminisztrációs felület nélkül is önállóan működtethető marad. Az adminisztrációs felületre több felhasználó is bejelentkezhet, akik az adatbázis egy külön táblájában tárolásra kerülnek személyes beállításokkal és jogosultsági szintjükkel. Jogosultság szerint megkülönböztetett lesz a teljes körű (adminisztrátori) hozzáférés és a szerkesztői szintű. Utóbbiak csak a tartalmak írására és feltöltésére kapnak engedélyt.

Az adminisztrációs felületen általános beállítások módosítására lesz lehetőség, mint honlap neve, nyelve, felületek és bővítmények engedélyezése illetve e-mail küldési beállítások. Az oldalak szerkesztésén felül menük létrehozását és szerkesztését lehet drag'n'drop módszerrel végrehajtani. Különböző fájlok és képek feltöltését, mappákba rendezését végezhetik a jogosultak.

Mindezek mellett a rendszernek kellően kicsinek, gyorsnak és átláthatónak kell lennie. A megvalósításhoz szükséges technológiák: PHP, MySQL, HTML5, CSS3 és JavaScript.

## 2. Bevezetés

### 2.1. A feladat értelmezése, indokoltsága

Az internet – és vele együtt a világháló – globális elterjedése gyökeresen változtatta meg az emberek életét. Egyre több felhasználó veszi igénybe a WEB nyújtotta ingyenes tudásbázist, így az az információszerzés meghatározó forrásává vált. A legtöbb cég és magánszemély egyaránt rájött, hogy a honlap egy olyan ingyenes reklám felület, mely a nap 24 órájában helyette dolgozik és ügyfeleket szerez, termékeket árul.

Nem meglepő tény, hogy egyre nagyobb igény mutatkozik a honlapok folyamatos naprakészen tartására, új tartalmak közzétételére, az legújabb trendeknek megfelelő technológiák használatára. A tulajdonosok már nem elégednek meg egy kézhez kapott weblappal, a legtöbb esetben saját maguk szeretnék szerkeszteni azokat. Ugyanakkor tőlük nem várható el a megfelelő technikai háttérismeret, mellyel képesek lennének rá.

A megoldás erre a gyakori problémára egy interaktív web alkalmazás készítése, mely segítségével bárki, egyszerűen és gyorsan, előzetes ismeretek nélkül módosíthatja honlapja tartalmát és kinézetét. Az ilyen alkalmazásokat tartalomkezelő rendszernek (angolul: CMS) körébe szokás sorolni.

A rendszer elsődleges feladata, hogy biztosítsa egy honlap módosításának lehetőségét azon emberek számára is, akik nem rendelkeznek megfelelő informatikai háttérismerettel. A felhasználók számára kialakított adminisztrációs felület egyszerűen lehetővé teszi a honlapok paramétereinek módosítását, grafikus szövegszerkesztő segítségével pedig szükségtelenné teszi a megfelelő webes leíró nyelvek ismeretét. A felhasználók különböző jogosultságokkal rendelkezhetnek. (Pl.: Látogató, Szerkesztő, Adminisztrátor) Számukra különböző adminisztrációs nézetek kialakítása a cél.

A látogatók megtekinthetik a publikált oldalakat. A szerkesztőknek lehetősége nyílik az oldalak tartalmát módosítani, új weblapokat létrehozni, törölni, illetve különböző fájlokat azokon elhelyezni. (Képek, videók, stb.) Az adminisztrátor – a legmagasabb rang – az összes beállítást módosíthatja, különböző bővítményeket és felületeket telepíthet, törölhet, módosíthat.

A tervezett interaktív program, bármely web szerveren futtatható, PHP és MySQL alapú, ezzel jelentősen javítva a felhasználási lehetőségeket, hiszen e követelményeket a legtöbb szolgáltató alapvetően biztosítja. További előny, hogy a rendszer a honlap szerves részét képezi, így azzal együtt gyakorlatilag bárhonnét elérhető. Csupán egy böngésző szükséges a használatához.

A CMS használhatóságát igazolja, hogy a korábban web-fejlesztők bevonását igénylő, több napot igénylő módosítások percekre redukálhatók. A tulajdonosnak nem kell megbízottat alkalmaznia, ötleteit, változtatásait – a rendszer segítségével – saját kezűleg hajthatja végre. A jól ismert grafikus szövegszerkesztők esetében megszokott funkciók rendelkezésre állnak, velük teljes mértékben testre szabhatók a weblapok.

Ugyanakkor az adminisztrációs felület csak felhasználói fiókkal és jelszóval rendelkező személyek számára elérhető, ezzel garantálva az adatok biztonságát.

## 2.2. Alternatívák kritikai elemzése, összehasonlítása

A jelenleg létező – többnyire nyílt forráskódú, ingyenes – CMS rendszerek (Drupal, WordPress, Joomla, MediaWiki, e107) általános jellemzője, hogy robusztusak, igyekeznek minden használati esetet kielégíteni. Ez gyakran a betöltési sebesség, az egyszerűség és kezelhetőség kárára válik. Hátrálynak tekinthető sok esetben a magyar nyelv hiánya is.

A fent említett CMS-ek nyílt forrású volta miatt azonban, nem szabad megfeledkezni a legfőbb előnyükről, a közösség által létrehozott rengeteg bővítményről és a különböző fórumokban elérhető leírásokról. Belátható, hogy az egyedi rendszerek nem szállhatnak versenybe ezen a téren nyílt forrású társaikkal.

Jelen szakdolgozat témájában csak a bemutatkozó oldalakra korlátozódik, így a benne tárgyalt saját fejlesztésű alkalmazásom csak e szűkebb felhasználást célozza. Felépítés tekintetében így sokkal egyszerűbb megoldást, átláthatóbb, felhasználóbarát felületet és kizárólag szükséges funkciókat kínálja. Az specializálás következtében a rendszer kevesebb helyet foglal és gyorsabb a versenytársainál.

Tapasztalataim szerint a kis- és középvállalkozások, kisebb márkák, magánszemélyek esetében éppúgy létjogosultsága van egy egyedi CMS-nek, mint a nagyvállalatoknál. A továbbiakban kidolgozásra kerülő rendszer e meggyőződésemre alapul.

## 2.3. A szakdolgozat felépítése

A dolgozat első felében – az úgynevezett fejlesztői dokumentációban – mutatom be az eddigiek során körvonalazott rendszert.

Első lépésként rendszertervet készítek a web alkalmazással szemben támasztott követelmények, a megvalósítandó funkciók, a tervezés lépései és annak elméleti működése itt kerül kidolgozásra.

Ezt követően a megvalósítás lépéseit tárgyalom, kitérve az alkalmazott megoldásokra, szoftveres környezetre.

A szakasz utolsó felében a megfelelőség vizsgálatára teszek kísérletet. A tervezett és megvalósított alkalmazás közti kapcsolatokat vizsgálom, illetve a témaválasztás során felállított követelményeknek való megfelelését.

A második fő szakaszban – a felhasználói dokumentációban – az alkalmazás telepítését mutatom be, majd annak használatát fogalmazom meg a felhasználók számára közérthető módon, a használati esetükre kielezett szemszögből.

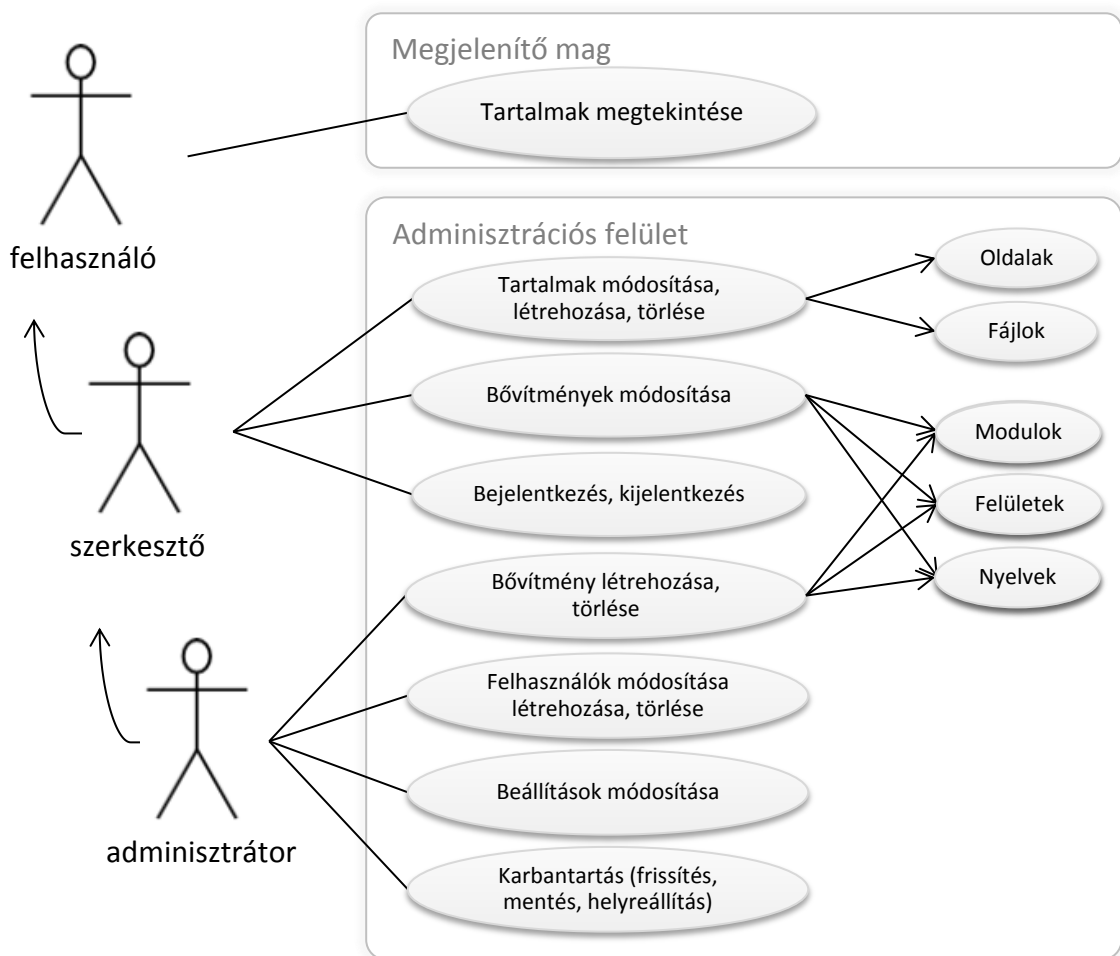
Végül, a dolgozat zárataként, egy rövid összefoglalóban a továbbfejlesztési lehetőségekre térek ki.

# 3. Fejlesztői dokumentáció

## 3.1. Tervezés, rendszerterv

### 3.1.1. Funkciók meghatározása

A rendszer működése során használt interakciókat a tervezés első lépéseként célszerű összegezni. Ez alapján könnyedén meghatározhatóak a szükséges funkciók. Az alkalmazásom funkcióit az egységesített modellező nyelv<sup>[1]</sup> (UML) Használati eset diagramjával szemléltetem.



3-1. ábra: Használati eset diagram

Három felhasználót (aktort) különböztetek meg: a rendszer szempontjából általános felhasználót, tartalom szerkesztőket és teljes hozzáféréssel rendelkező adminisztrátort. Ezek egymásból származtathatók, fokozatosan bővül a jogosultsági körük. A rendszer két részre különül el, egy megjelenítésért felelő magra és egy adminisztrációs felületre. Az adminisztrációs felület interakcióit a fenti diagram szemlélteti.

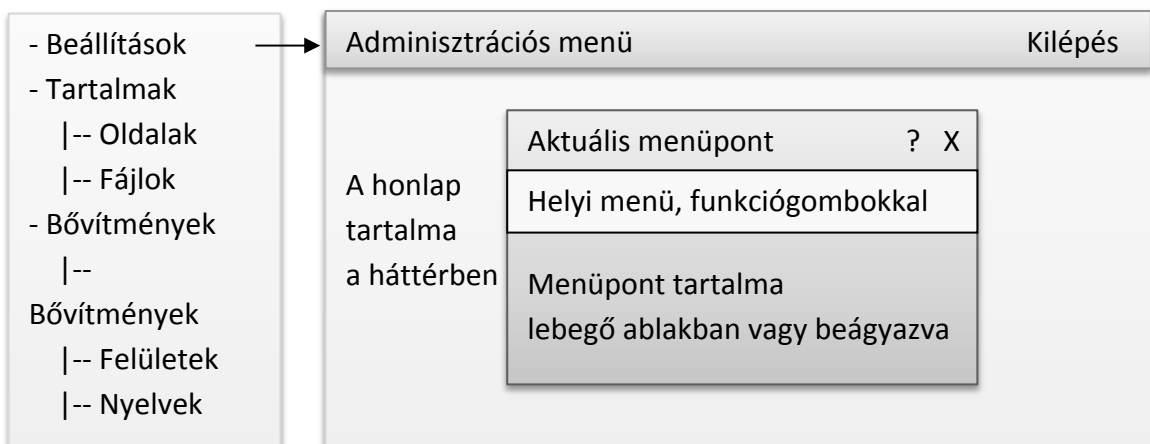
[1] Dr. Tarczali Tünde - UML diagramok a gyakorlatban



### 3.1.2. Felhasználói felület és menü

Az előzőek során meghatározott használati esetekből egyenesen származtatható az adminisztrációs felület menü szerkezete. A menü elemeit tematikusan csoportosítottam, hogy a felhasználó előtt mindig csak a számára fontos, az aktuális művelet elvégzéséhez szükséges elemek jelenjenek meg. További praktikussági szempontok figyelembevételével a menüt a felület felső sorában fogom rögzíteni, így az mindig, minden oldal felett meg fog jelenni.

A menüpontra kattintva, annak tartalmát a szerkesztendő oldal felett, lebegő ablakban kívánom megjeleníteni. Ha ezt a technikát a felhasználó böngészője nem támogatja, akkor – opcionálisan – a lebegő ablak helyett a honlapba beágyazva is megjelenítésre kerülhet.



3-2. ábra: Felhasználói felület és menü

A lebegő ablak tetején az aktuális menüpont címe szerepel súgó és bezárás gombokkal. Ezek alatt kap helyet a helyi menüsáv, mely mindig az aktuális menüpontra megfelelő gombokat tartalmazza. (Pl.: oldalak esetén új oldal létrehozása gomb) Ez alatt táblázatos formában kerül felsorolásra a lebegő ablak tartalma, mely az operációs rendszerekből ismert módon rendezhető név és egyéb tulajdonságok szerint növekvő és csökkenő sorrendben. Általában az oldal alján fog helyet kapni a mentés gomb.

### 3.1.3. Funkciók kifejtése

A dizájn alapú tervezés első szakaszán, a felhasználóbarát felület kialakításán túllépve megkezdődhet annak funkciókkal történő kitöltése. A rendszertől a következő funkciókat várom el, menüpontra lebontva:

Beállítások menü tervezett elemei:

- **Oldalbeállítások:** Itt lehet a honlapot zárolni, nevet, szlogent, címsort megadni, továbbá lábléceket és oldalsávot testre szabni
- **Környezeti beállítások:** URL átírás keresőbarát címekre, időzóna, nyelvi beállítások és dátum formátum megadása
- **Statisztika:** statisztikai elemek elhelyezése a felületen (pl: Google Analytics)
- **Közösség:** megosztás gombok és közösségi modulok elhelyezése a felületen
- **Email küldés:** a küldés módjának megadása (SMTP levélküldő szerver esetén a kapcsolódási adatok itt tölthetők ki)
- **Adatbázis:** az adatbázis kapcsolódási adatainak megadása itt lehetséges

Oldalak menü tervezett funkciói:

- **Létrehozás:** új, üres oldal vagy aloldal elhelyezése, kivéve, ha az már létezik – szerkesztői jogosultság szükséges hozzá
- **Listázás:** az elérhető oldalak listázása, rendezése név és dátum szerint
- **Megtekintés:** a mentett oldalak megnézhetőek adminisztrációs módban is
- **Módosítás:** az egyes oldalak tartalmának, leírásának és címének módosítása, a menük szerkesztése egy menüszerkesztő segítségével, valamint a kezdőoldal kijelölése a listából
- **Törlés:** minden oldal törölhető a szerkesztők által, kivéve a kezdőoldalt

Fájlok menü tervezett funkciói:

- **Feltöltés és létrehozás:** fájlok feltöltése a szerverre illetve új könyvtárak létrehozása, kivéve, ha az már létezik – szerkesztői jogosultság szükséges
- **Listázás, léptetés:** könyvtárak közti váltás, a bennük elérhető fájlok listázása, rendezése név, méret és dátum szerint
- **Megtekintés:** a feltöltött fájlok megnyitása új ablakban
- **Módosítás:** a fájlok vagy könyvtárak átnevezése, kivéve, ha már létezik az adott névvel más, valamint a képfájlok szerkesztése, átméretezése, részletek kivágása
- **Törlés:** minden fájl és könyvtár törölhető szerkesztő rangú felhasználó által, kivéve a nem üres mappák

Bővítmények menü tervezett funkciói:

- **Feltöltés:** külső személy által készített bővítmény implementálása a rendszerbe – adminisztrátori jogosultság szükséges
- **Listázás:** az elérhető bővítmények listázása, rendezése név szerint
- **Megtekintés:** a bővítmény részleteinek (szerző, verzió, technológia) megtekintése
- **Módosítás:** egyedi bővítmény beállítások módosítása illetve az egész honlapon betöltésre kerülő globális bővítmények megadása, egyes bővítmények engedélyezése vagy letiltása – minden módosítás csak az adminisztrátor számára érhető el
- **Törlés:** minden bővítmény törölhető az adminisztrátor által

#### Felületek menü tervezett funkciói:

- **Feltöltés:** külső személy által készített grafikus felület implementálása a rendszerbe – adminisztrátori jogosultság szükséges
- **Listázás:** az elérhető felületek listázása, rendezése név szerint
- **Megtekintés:** a felület részleteinek (szerző, verzió, technológia) megtekintése
- **Módosítás:** az alapértelmezett bővítmény beállítása, a honlap ikonjának lecserélése, valamint az egyes felületek szerkesztése – minden módosítás csak az adminisztrátor számára érhető el
- **Törlés:** minden felület törölhető az adminisztrátor által, kivéve az alapértelmezettet

#### Nyelvek menü tervezett funkciói:

- **Feltöltés:** külső személy által készített nyelvi fájl implementálása a rendszerbe – adminisztrátori jogosultság szükséges
- **Listázás:** az elérhető nyelvek listázása, rendezése név szerint
- **Megtekintés:** a nyelv részleteinek (szerző, verzió) megtekintése
- **Módosítás:** a nyelvi fájlok tartalma az adminisztrátor által szerkeszthető
- **Törlés:** minden nyelv törölhető az adminisztrátor által, kivéve az utolsót

#### Felhasználók menü tervezett funkciói:

- **Létrehozás:** szerkesztő szintű felhasználói fiók készítése, ha az adott névvel a fiók még nem létezik. Létrehozáskor tájékoztató e-mail kerül kiküldése. – adminisztrátori jogosultság szükséges
- **Listázás:** az elérhető felhasználók listázása, rendezhető név és rang szerint csökkenő illetve növekvő sorrendben
- **Módosítás:** név, email cím, jelszó, rang, nyelv, és lebegő ablak bekapcsolásának változtatása érhető el itt. A beállítások megváltoztatásához az adott fiók tulajdonosának kell lenni. A adminisztrátor más fiók rangját is képes változtatni.
- **Törlés:** minden felhasználó törölhető az adminisztrátor által vagy önmaga által, kivéve az utolsó adminisztrátort

#### Karbantartás menü tervezett funkciói:

- **Frissítés:** a rendszerhez elérhető javítások, újabb verziók itt tölthetők fel
- **Biztonsági másolat:** a teljes rendszerről biztonsági másolat készíthető, ha a felhasználó adminisztrátori ranggal rendelkezik
- **Helyreállítás:** a korábban készített biztonsági másolatok feltöltése, az összes jelenlegi adat felülírása – szintén adminisztrátori rang szükséges hozzá
- **Listázás:** a hibaüzenetek és felhasználói belépések listázása, rendezése, továbbá a rendszer állapotáról jelentése készítése

#### Súgó gomb a lebegő menü tetején:

- **A súgó** menüpont elérhető minden fenti funkció esetén, hely érzékeny információkat szolgáltatva a felhasználók számára.

### 3.1.4. Adatbázis séma

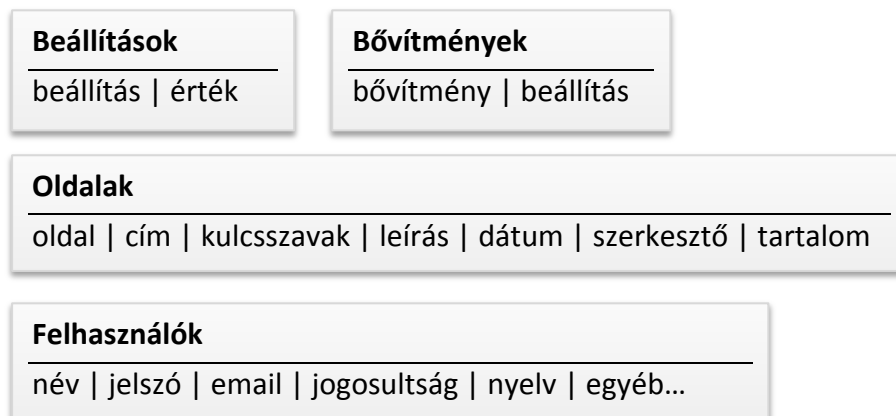
A beállítások tárolására egy adatbázis táblát célszerű létrehozni, melyben minden mező egy beállítási paramétert tartalmaz, ebből a rendszer minden betöltés előtt kinyerheti a szükséges adatokat.

Szükség lesz a bővítmények beállításainak elmentésére is, így egy újabb táblával kell bővíteni az adatbázist. Mivel a bővítmények eltávolíthatóak, ezért fontosnak vélem, hogy ne vegyítsem őket a honlap általános beállításával.

A felhasználók ugyancsak külön táblát igényelnek, hiszen minden felhasználói fiókhoz több mező tartozhat. (Pl. név, email cím, stb.)

Tartalmak esetén az oldalak tárolása szintén külön adatbázis táblában történhet. (Itt is több mezőre lesz szükség.) A feltöltött fájlok nem tárolhatók ilyen módon, azok továbbra is fájlformátumként a szerveren kapnak helyet.

A felületek és nyelvek nem kimondottan gyakran módosított egységek, így az adatbázis terhelését ezekkel feleslegesnek tartom.



3-3. ábra: Adatbázis séma

## 3.2. Megvalósítás

### 3.2.1. Környezet specifikációja

Arra a tényre alapozva, hogy a dolgozat írásakor a web szerverek 81%<sup>[2]</sup> az ingyenes PHP<sup>[3]</sup> modult használja – maga mögé utasítva az ASP.NET-et, a Perl, Rubyt és Pythont –, így a web alkalmazásom ezen a szkript nyelven láttam célszerűnek elkészíteni.

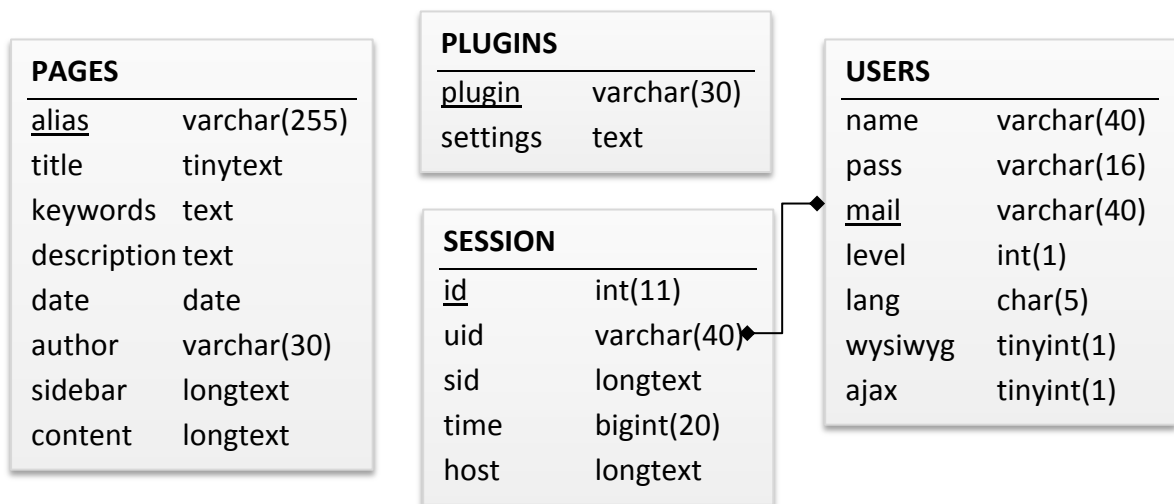
További érvek a fejlesztő környezet mellett, hogy a több ismert tartalomkezelőt is PHP-ben készítették. (Pl.: MediaWiki, Joomla!, WordPress, Drupal) Továbbá az olyan weboldalak, mint a Wikipédia és a Facebook, szintén ezzel a jól optimalizált és gyengén típusos nyelvvel íródtak.

Az adatbázis kezelő kiválasztása során az ugyancsak egyik legelterjedtebb<sup>[4]</sup>, és a PHP által is támogatott MySQL<sup>[5]</sup> relációs adatbázis kezelő mellett döntöttem.

A megjelenés tekintetében a HTML5 és CSS3 leíró nyelveket valamint jQuery függvénytárat választottam, hiszen az adminisztrációs felület a honlap része lesz, így megfelelően illeszkedni fog abba.

### 3.2.2. Adatbázis elkészítése

A korábban felvázolt adatbázis séma alapján, a web alkalmazást relációs adatbázissal valósítom meg, az alábbi 4 táblára támaszkodva: PAGES – weblapok adatait tároló tábla, PLUGINS – bővítmények beállításait tároló tábla, USERS – felhasználók adatait tároló tábla és SESSION – bejelentkezett felhasználók azonosítására szolgáló tábla.



3-4. ábra: Adatbázis felépítése

[2] W3Techs statisztikája, 2014. március – <http://w3techs.com>

[3] PHP – Web fejlesztő szkript nyelv és feldolgozó – <http://php.net>

[4] DB-Engines Ranking, 2014. március – <http://db-engines.com>

[5] MySQL – Adatbázis kezelő rendszer és lekérdező nyelv – <http://www.mysql.com>

A táblákban az oszlopokat, attribútum típusokkal és azok hosszaival, illetve a kulcs tulajdonságok aláhúzásával jelöltem.

A korábbi elképzelésekkel ellentétben a beállításokat célszerűségi okokból nem tárolom adatbázisban, hiszen az adatbázis elérésére vonatkozó paramétereket is tartalmaz. Ezek hiányában azonban csatlakozni sem lehet az adatbázishoz. A beállítások tárolása így egy konfigurációs fájlban valósul meg.

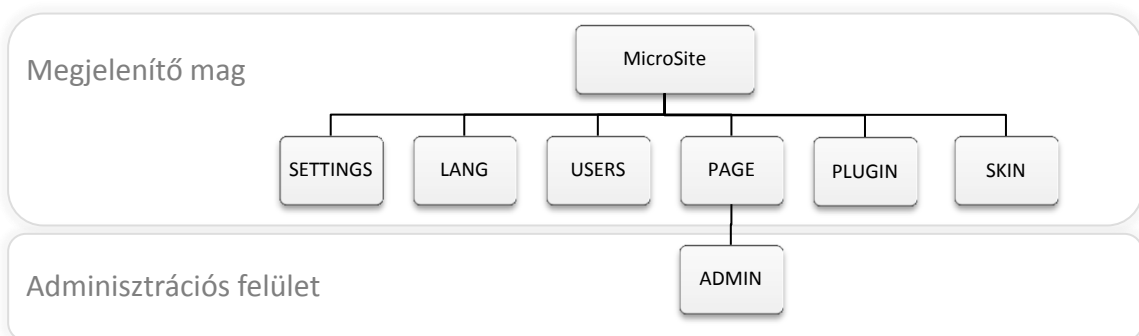
Különálló SESSION tábla létrehozását azért választottam, mert előfordulhat az az eset, hogy egy felhasználó egyszerre több helyről is be van jelentkezve. Ezzel együtt az adatbázis nagyobb normálfokra is lépett, hiszen a személyes adatoktól független bejelentkezési információk redundanciát okoztak volna. A megoldás kiváltja a kritikus adatok tárolását is a PHP session változóból, ezzel növelve a biztonságot.

### 3.2.3. Az alap rendszer implementálása

Az adatbázis elkészítése után gondoskodni kell az adatok lekérdezésére és megjelenítésére szolgáló műveletekről. Ezek implementálása során külön osztályokra bízom a feladatkörök szétosztását, mert így az egyes részek – ellentétben egy procedurális kialakítással – a későbbi fejlesztések során könnyebben módosíthatók vagy újra felhasználhatók lesznek.

A következő osztályok mind a rendszer magját képezik, feladatuk kizárólag az adatok összeállítása és megjelenítése. Mindegyiket egy központi PHP fájl alá rendeltem, ami ezek meghívását és összehangolását végzi.

Az osztályok kialakítása sematikusán a következőképpen néz ki:



3-5. ábra: A rendszer elvi kialakítása

A megjelenítő funkciókat, melyek az adminisztrációs felület nélkül, önállóan is használhatóak, az alábbi – magyarázattal ellátott – UML osztály diagramok alapján valósítottam meg.

## USERS

users :array	– a beállításokat tartalmazó tömb
users()	– konstruktor, betölti a felhasználókat
get(key:text) :text	– lekérdez egy felhasználót
set(key:text, value:text) :bool	– ment egy felhasználót

3-6. ábra: Felhasználók osztály diagram

## SETTINGS

settings :array	– a beállításokat tartalmazó tömb
settings()	– konstruktor, betölti a beállítás fájlt a settings-be
get(key:text) :text	– lekérdez egy beállítást
set(key:text, value:text) :bool	– ment egy beállítást
config_php(attrib:type)	– beállítja a php környezeti változóit

3-7. ábra: Beállítások osztály diagram

## LANGS

lang :array	– a nyelvi változókat tartalmazó tömb
langs()	– konstruktor, betölti a nyelvi fájlt a lang változóba
list_langs(dir:text) :array	– listázza az elérhető nyelveket

3-8. ábra: Nyelvek osztály diagram

## PAGE

title :text	– az oldal címe
keywords :text	– az oldal kulcsszavai
description :text	– az oldal rövid leírása
content :text	– az oldal tartalma
sidebar :text	– az oldal egyedi oldalsávja
author :text	– az oldal utolsó szerkesztője
date :text	– az oldal módosításának dátuma
page()	– konstruktor, meghívja az oldalbetöltést vagy admint
list_pages() :array	– listázza az elérhető oldalakat
include_page()	– betölti a megfelelő oldalt vagy hibalapot
print_sitemap(pages:array)	– oldal térképet készít a keresők számára

3-9. ábra: Oldalak osztály diagram

## SKIN

charset :text	– karakterkódolás
lang: text	– a lap nyelve
mime: text	– a lap MIME típusa
htmlver: text	– a lap html verziója
doctype: text	– a lap dokumentum-típus definíciója
head: text	– a lap fejléce
styles: text	– betöltendő stíluslapok
scripts: text	– betöltendő JavaScript
body: text	– a lap törzse
<hr/>	
skin()	– konstruktor, beállítások inicializálása
list_skins(dir:text) :array	– listázza az elérhető felületeket
addStyle(file:text, media:text)	– stíluslap hozzáadása a fejléchez
addScript(file:text)	– JavaScript hozzáadása a törzshöz
readSkinFile(location:text, skin:text)	– beolvassa a felület fájlt
setMimeAndDoctype(doctype:text)	– MIME típust és DTD-t küldi a böngészőnek
head()	– összeállítja a lap fejlécét
body()	– összeállítja a lap törzsét
printHTML()	– elküldi a kész HTML fájlt a böngészőnek
create_links(titles:array, pages:array) :text	– a menü tömbből HTML menüt készít
create_backlinks() :text	– az URL-ből „kenyérmorzsa” menüt készít
message(type:text, msg:text, reload:int=7) :text	– értesítést vagy hibaüzenetet jelenít meg

3-10. ábra: Felületek osztály diagram

## PLUGIN

globalplugins :text	– minden oldalon megjelenő bővítmények
allowplugins :text	– engedélyezett bővítmények
pluginsdir :text	– a bővítmények könyvtára
<hr/>	
plugin()	– konstruktor, inicializálja a változókat
get_plugins() :text	– megkeresi és a helyére illeszti a bővítményeket
list_plugins(dir:text) :array	– listázza az elérhető bővítményeket
include_plugin(name:text, params:text, &styles:text, &scripts:text) :text	– lefuttatja a bővítményt a paraméterekkel

3-11. ábra: Bővítmények osztály diagram



Külön kiemelném még itt a bővítmények illesztő felületét, melyre egy az előzőektől független osztályt hoztam létre. Minden bővítmény ebből az objektum osztályból kerül származtatásra, ezzel biztosítva a CMS és a bővítmények szabványos kommunikációját. A megoldás előnye, hogy a bővítmény készítője a rendszer behatóbb ismerete nélkül lekérdezheti a beállításokat, mentheti azokat, megkapja a meghíváskor átadott paramétereket, valamint egyéni adminisztrációs felület készíthet, mely beépül a CMS adminisztrációs felületébe. További előny, hogy a működés közben generált adatokhoz és a tényleges adatbázishoz a bővítmény készítője nem férhet hozzá, ezzel jelentősen növelve a rendszer biztonságát.

Az illesztő osztály a következőképpen néz ki:

<b>MicroSitePlugin</b>	
output :text	– a bővítmény HTML kimenete
styles :text	– a bővítmény által használt stílus definíciók
scripts :text	– a bővítmények által használt JavaScript fájlok
params :text	– a bővítmény meghíváskor kapott paraméterek
settings :text	– a bővítmény elmentett beállításait tárolja
MicroSitePlugin()	– konstruktor, inicializálja a változókat
loadsettings() :bool	– betölti a bővítmény eltárolt beállításait
savesettings() :bool	– elmenti a bővítmény beállításait
admin()	– megjeleníti a bővítmény saját adminisztrációját

3-12. ábra: Bővítmény illesztő osztály diagram

A PHP a többi programozási nyelvhez képest sajnos gyengébben kezeli az objektum osztályokat. 4-es verziójában még nem lehetett megkülönböztetni láthatósági szinteket a tulajdonságokra és metódusokra. Jelenleg már az 5-ös verzió az elterjedt, de a visszamenőleges kompatibilitás érdekében a web alkalmazásomban nem használtam ezen megszorításokat.

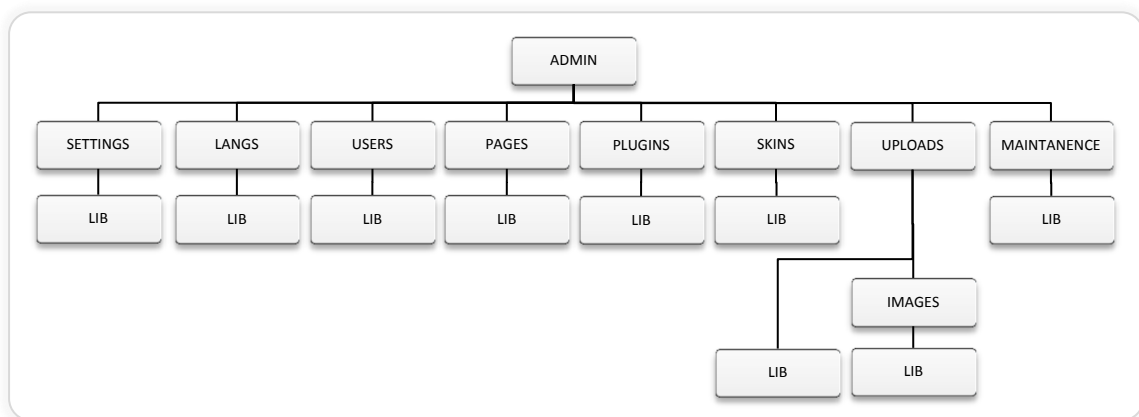
A bővítmények fejlesztéséhez szükséges további leírások a mellékelt CMS rendszerben, egy kommentekkel ellátott minta bővítmény formájában érhetők el.

A felületek készítéséhez ugyancsak a mellékletek közt található egy részletes magyarázattal ellátott felület sablon.

### 3.2.4. Az adminisztrációs felület implementálása

Az adminisztrációs felület oldalai esetében nem láttam szükségét különálló osztályok létrehozásának. Ebben közrejátszott az is, hogy a PHP nem képes egy osztályként kezelni több különálló fájlt. Az adminisztrációs felület egy egységben csak úgy tudtam megtartani, ha procedurális módon függvényekbe szervezem a részegységeket. Így összesen 8 fájlra tagoltam a felületet. Ezek mindegyike a központi PHP fájl alá szervezett, mindegyikük egy nézetből és egy függvény könyvtárból (LIB) áll. A nézetek tartalmazzák a megjelenést és az űrlap elemeket HTML és CSS<sup>[6]</sup> illetve jQuery formában, a könyvtárak pedig a műveletvégző függvényeket.

Az elrendezést a következő ábra szemlélteti:



3-13. ábra: Az adminisztrációs felület elvi vázlatja

Az ADMIN fájl tartalmazza a bejelentkezés, az elfelejtett jelszó, a bejelentkezés utáni kezdőoldal, valamint az adminisztrációs menü nézetét.

Funkciókészletében azok a függvények szerepelnek, amelyek a hitelesítést (*isAdmin()*), bejelentkeztetést (*doLogin()*), kijelentkeztetést (*doLogout()*), elfelejtett jelszó kiküldését (*recover\_mail()*) és egyéb rendszer üzenetek küldését (*sendmail()* – levél küldés, *send\_msg()* – felületi üzenet küldés, *read\_msg()* – felületi üzenet megjelenítés) hajtják végre. Ezek visszatérési értéke jellemzően boolean típusú, kivéve a *recover\_mail()* és *read\_msg()* függvényeket. Utóbbiak string-ként tájékoztatnak a művelet sikerességéről.

Az első szinten felsorolt 8 fájl mindegyike abban az esetben kerül betöltésre, ha a felhasználó be van jelentkezve a felületre illetve a menüpontok közt a megfelelőt kiválasztotta. Ekkor a megjelenéssel együtt betöltődik az oldal saját funkciókészlete is.

Ez alól kivételt képez a SETTINGS.LIB fájl, ami a többi ADMIN alá szervezett oldalon is betöltésre kerül, hiszen a beállítások írására az összes többi adminisztrációs oldalnak is szüksége lehet.

[6] HTML és CSS leíró nyelvek - <http://www.w3.org/standards/webdesign/htmlcss>

Az oldalakon a funkciók végrehajtása a megjelenésben elküldött űrlap mezők alapján történik. Mivel az űrlapot ugyanannak az oldalnak címezem, amelyikről elküldtem azt, fennállna a veszélye, hogy a fejlécben tárolt POST és GET értékek – az oldal felhasználó általi frissítése esetén – újra lefuttatnák az érintett műveleteket.

Ezt elkerülendő, a műveletek végrehajtása után egy session munkamenet változóban tárolom az eredményt, majd az eredmény oldalt rögtön a fejlécben átirányítom a kiindulási lapra. Így a felhasználó szintén eredeti oldalra érkezik, láthatja a művelet kimenetelét, de az elveszti az eredmény oldalban még létező POST és GET értékeket. Ennek megvalósítására szolgál a `send_msg()` és `read_msg()` függvénypár.

A felhasználók hitelesítéséhez ugyancsak egy session munkamenet változót használtam. Ennek sajátossága, hogy értéke 24 óránként frissül, így minden felhasználót a rendszer legkevesebb ennyi idő elteltével automatikusan kijelentkezett. A felhasználói fiókok biztonságát tovább növelve meghatároztam egy 1 órás limitet is, ami a legutolsó felhasználói interakciót követően újraindul. Ennek lényege, ha a felhasználó valamilyen okból kijelentkezés nélkül távozott, más ne végezhesen műveletet az ő jogosultsági körében és az ő nevében.

Az adminisztrációs felület további 8 oldalát részletesen nem kívánom bemutatni. A „3.1.3. Funkciók kifejtése” pontban tárgyalt műveletek megvalósítása ezekben történik. Egyenként: SETTINGS – beállítások módosítása, USERS – felhasználók kezelése, LANGS – nyelvi fájlok kezelése, PLUGINS – bővítmények kezelése, SKINS – felületek kezelése, PAGES – oldalak kezelése, UPLOADS – fájlok kezelése és MAINTENANCE – karbantartás.

A bővítmények a korábban említett illesztő felület segítségével saját adminisztrációs felületet kapnak, a felületek és nyelvek a beépített CodeMirror<sup>[7]</sup> szövegszerkesztővel módosíthatók. Az oldalak könnyű, grafikus szerkesztését a TinyMCE<sup>[8]</sup> szerkesztővel oldottam meg. Az adminisztrációs felület egésze a jQuery<sup>[9]</sup> nyílt függvénykönyvtárra és szintén nyílt Bootstrap<sup>[10]</sup> stíluskészletre illetve függvényekre épít.

Említésre méltónak tartom még, hogy az UPLOADS oldal alá szervezett IMAGES oldal, ami a feltöltött képek átméretezését és képrészletek kivágását teszi lehetővé, a szintén nyílt, bárki számára elérhető jCrop<sup>[11]</sup> jQuery bővítményt használja. A „fogd és vidd” módszerű menü szerkesztőt és a listák rendezhetőségét biztosító jQuery alapú szkripteket – ahogy a CMS többi részét is – magam készítettem.

[7] CodeMirror – JavaScript alapú kódszerkesztő - <http://codemirror.net>

[8] TinyMCE – JavaScript alapú grafikus szövegszerkesztő - <http://www.tinymce.com>

[9] jQuery – JavaScript függvénykönyvtár - <http://jquery.com>

[10] Bootstrap – Front-end keretrendszer - <http://getbootstrap.com>

[11] jCrop – jQuery bővítmény - <http://deepliquid.com/content/Jcrop.html>

## 3.3. Verifikáció és validáció

### 3.3.1. A tervezett és megvalósított alkalmazás vizsgálata

A verifikáció célja az alkalmazás hibáinak feltárása és azok lokalizálása. A hibák kiszűrésére nem elegendő a körültekintő tervezés és megvalósítás, hanem megfelelő tesztelésen is keresztül kell mennie az alkalmazásnak.

A következő hiba lehetőségek már kódoláskor figyelembe lettek véve:

- **Adathibák:** minden változó inicializálása a használat előtt és a tömb típusok garantálása a tömböket kezelő függvények esetén. (Más adattípusok közt automatikus a konverzió a PHP-ben.)
- **Vezérlési hibák:** a végtelen ciklusok elkerülése illetve az utasítás blokkokon belül lévő műveletek ellenőrzése, hogy nem kerültek-e a blokkon kívülre.
- **Input-output hibák:** minden használt változónak értéke van, akkor is ha annak forrása nem elérhető.
- **Interfész hibák:** megfelelő számú, típusú és sorrendű függvény paraméter alkalmazása mindenhol.
- **Kivétel kezelési hibák:** minden lehetséges hibahatás kezelve lett?

Ez utóbbi pont bizonyítására egy tesztelési tervet készítettem, mely két fő fázisra bontható:

**1. Az első fázis:** a fehér dobozos tesztelés. Ekkor a kód ismeretében minden lehetséges kimenethez megpróbálom eljutni, szándékosan rossz adatokat megadva és kivétel kezelési hibákat generálva. Ezt a fázist minden kódrészlet implementálása után külön, az adott kódrészletre vonatkozóan végrehajtottam.

**2. A második fázis:** a fekete dobozos tesztelés, mely a kód ismerete nélkül különböző bemeneti adatokra illetve azok helytelen megoldására fókuszál. Például e-mail cím mező számokkal való kitöltése. Ez a fázis gyakorlatilag az elsőnél gyengébb tesztelést tesz lehetővé, viszont már a teljes rendszerre irányul. A modulok megfelelő együtt dolgozását lehet vele kimutatni.

A tesztelés során minden függvényt megvizsgáltam az első fázis szerint, majd a kész rendszert a második fázis szerint. Egy saját hibakezelő függvény a *debug()* segítségével a tesztek eredményeit egy „errorlog.txt” állományban rögzítettem, majd ebből akkor töröltem, amikor a hiba javításra került. A tesztelést illetve javítást addig folytattam, amíg a hibák megszűntek.

A hibakezelő függvény azonban továbbra is aktív a rendszerben, hiszen a bővítmények fejlesztésénél ismét szükség lehet rá. További előnye, hogy a függvény összegyűjti és fájlba írja az egyébként képernyőn megjelenő hibaüzeneteket, az átlag felhasználók elől elrejtve azokat.

### 3.3.2. A követelményeknek való megfelelés igazolása

Az alkalmazások esetében nem elegendő, ha a megvalósított rendszer hibamentes, annak az iránta támasztott követelményeket is teljesítenie kell. A szoftver minőségi jellemzőit az ISO/IEC 9126 szabvány rögzíti, ennek egy kivonatát<sup>[12]</sup> vettem alapul a megfelelés igazolására.

**Megbízhatóság:** a rendszer a 3.3.1. pontban ismertetett hibaszűrésen esett át, így a felhasználók véletlen hibahatásait képes lekezeln. A szándékos hibaokozás ellen természetesen nem lehet védekezni, ezért a tartalomkezelő rendelkezik egy biztonsági mentés funkcióval, amelyből a megrongálódott rendszer helyreállítható.

**Használhatóság:** az adminisztrációs felületek kialakítása során elsődleges szempont volt a könnyű használhatóság. Sem előzetes tudást, sem különösebb tanulást nem igényel a szoftver. A felhasználók a rendszer működését percek alatt kiismerhetik, és hozzáláthatnak céljaik megvalósításához.

**Eredményesség, hatáskok:** az oldalak összeállításához és betöltéséhez szükséges idők (0,03 – 0,07 másodperc) nem különböznek jelentősen egy statikus (csupán HTML és CSS kódból álló, előre összeállított) weblaptól, mégis nagy teret engednek a felhasználóknak az oldalak könnyű szerkesztésében.

**Fenntarthatóság:** a CMS a bővítmények, felületek és nyelvek segítségével olyan mértékben megváltoztatható, hogy a felhasználó minden igényét ki tudja elégíteni. A rendszer a fejlesztők számára is optimális kialakítású, az osztályokba rendezett megjelenítő mag egyes részei könnyen módosíthatóak a teljes rendszer megváltoztatása nélkül.

**Áttelepíthetőség, hordozhatóság:** a létrejött tartalomkezelő rendszer teljes mértékben hordozható. A programrészek minden szerveren működőképesek, melyek biztosítják a PHP futtatás és egy MySQL adatbázishoz való kapcsolódás lehetőségét. A rendszer könnyen installálható, használatához a felhasználónak csak egy böngészőre van szüksége. A komfortérzetet javítja, hogy ugyanazon adminisztrációs felülettel gyakorlatilag teljesen egyedi honlapokat készíthet bárki.

**Funkcionalitás:** a kialakított CMS a tervezés során a 3.1.1. és 3.1.3. pontokban felvázolt használati eseteknek illetve követelményként meghatározott funkcióknak megfelel. A rendszer biztonságos felhasználó hitelesítéssel rendelkezik, minden felhasználó a saját jogosultsági körének megfelelő módosításokat hajthat végre. A rendszer legfőbb célja, a kevés oldalból álló bemutatkozó oldalak létrehozására és könnyű szerkesztésére tökéletesen alkalmas.

[12] *Krammer Gergely – A szoftver minőségi jellemzői – <http://krammer.web.elte.hu>*

## 4. Felhasználói dokumentáció

### 4.1. Telepítési útmutató

A rendszer telepítése 3 egyszerű lépésből áll és csupán 5 percet vesz igénybe.

#### 1. lépés: Másolja fel minden fájlt a web szerverre:

Ezt legkönnyebben úgy teheti meg, hogy webes szolgáltatójától kapott tárhelyének elérését (általában FTP elérés, pl: <ftp.encegem.hu>) egy erre alkalmas programmal (FileZilla vagy TotalCommander) megnyitja, majd kitömöríti a rendszer fájlijait és ezeket mind felmásolja a szerver gyökérkönyvtárába.

#### 2. lépés: Nyissa meg honlapját egy böngészőből:

Ha a tárhelyre megfelelően másolta fel a rendszer fájlijait, a szolgáltatójától kapott internetcímet (domain címet, pl: [www.encegem.hu](http://www.encegem.hu)) egy tetszőleges böngészőben megnyitva a máris új honlapja fogadja Önt.

Ellenkező esetben győződjön meg róla, hogy szolgáltatója engedélyezi Ön számára a PHP futtatását és a web szerver alapértelmezett kezdőoldala az *index.php* beállítás.

#### 3. lépés: Állítsa be, majd használja új honlapját

A böngészőben a CMS egy telepítő ablakkal fogadja első indításkor, hogy a legszükségesebb beállításokat elvégezhesse az első használat előtt.

The image shows two screenshots of the MicroSite installation wizard. The left screenshot is the first step, titled "MicroSite telepítése". It contains a message: "Köszönjük, hogy rendszerünket választotta! Perceken belül kézhez veheti új honlapját... Előbb azonban kérjük, adja meg leendő honlapja nevét, és új fiókja adatait!". Below this are input fields for "Honlap neve" (with "Új honlap" as a placeholder), "E-mail cím", "Jelszó", "Jelszó ismét", and a dropdown for "Oldalak nyelve" (set to "hu-HU"). A link "Rendelekezem adatbázis eléréssel..." is visible. At the bottom are "Vissza" and "Tovább" buttons. The right screenshot is the second step, also titled "MicroSite telepítése". It contains a message: "Már csak a rendszerhez kapott szériaszám hiányzik... Kérjük ellenőrizze, hogy a rendszer működéséhez szükséges funkciókat az Ön web szervere támogatja-e!". Below this is a "Szériaszám" input field with "XXXX-XXXX-XXXX-XXXX" as a placeholder. A list of system requirements follows: "PHP verzió" (5.3.4 (> 4.4)), "Írási jogosultságok" (rendben), "Adatbázis" (rendben), "URL átirás" (rendben), "Zip archíváló" (rendben), "Képszerkesztő" (rendben), and "cURL modul" (rendben). At the bottom are "Vissza" and "Telepítés" buttons.

4-1. ábra: A rendszer telepítése

Töltse ki leendő honlapja nevét, adja meg felhasználói fiókjához kapcsolt, létező e-mail címét és jelszavát. Ezekon felül megadhatja a felhasználói felület nyelvét, és ha rendelkezik adatbázis eléréssel, az ahhoz szükséges elérési adatokat.

Az adatok megadása után egy a telepítés állapotát mutató képernyőre érkezik. Itt az esetleges hibákról további tájékoztatást ad a rendszer. Meg kell adnia még a rendszerhez kapott szériaszámot, majd a Telepítés gomb megnyomásával új honlapja elkészül.

Megjegyzés: ha bármilyen hibába ütközik a telepítést követően, lépjen be a rendszer feltöltéséhez használt programba, kapcsolódjon a tárhelyéhez, majd ott törölje a „*datas/config.php*” fájlt. Ezt követően böngészőjében a következő internet cím mintájára indítsa a telepítést:

[www.encegem.hu/index.php?/admin/install&reset](http://www.encegem.hu/index.php?/admin/install&reset)

## 4.2. Bejelentkezés, elfelejtett jelszó

Honlapja szerkesztéséhez egy adminisztrációs felületre kell bejelentkeznie, melyet az [www.encegem.hu/index.php?/admin](http://www.encegem.hu/index.php?/admin) vagy röviden a [www.encegem.hu/admin](http://www.encegem.hu/admin) mintára érhet el.

**Adminisztrációs felület**

Ez az oldal csak a honlap tulajdonosa számára érhető el! Ha Ön látogatóként van jelen, kérjük, válasszon egy másik menüpontot, vagy navigáljon vissza az előző oldalra! Köszönjük!

Az adminisztrációs felületre való bejelentkezéshez kérem, töltsen ki az alábbi mezőket!

E-mail cím

Jelszó

[Elfelejtette jelszavát?](#)

Ez a honlap a **WebMake MicroSite 2.1** tartalomkezelő rendszerével készült. Ön is hasonló módon, minden háttérismeret nélkül, szinte pecek alatt saját honlapot készíthet önmaga vagy cége számára. Szerezze be a CMS-t a fejlesztő honlapjáról!

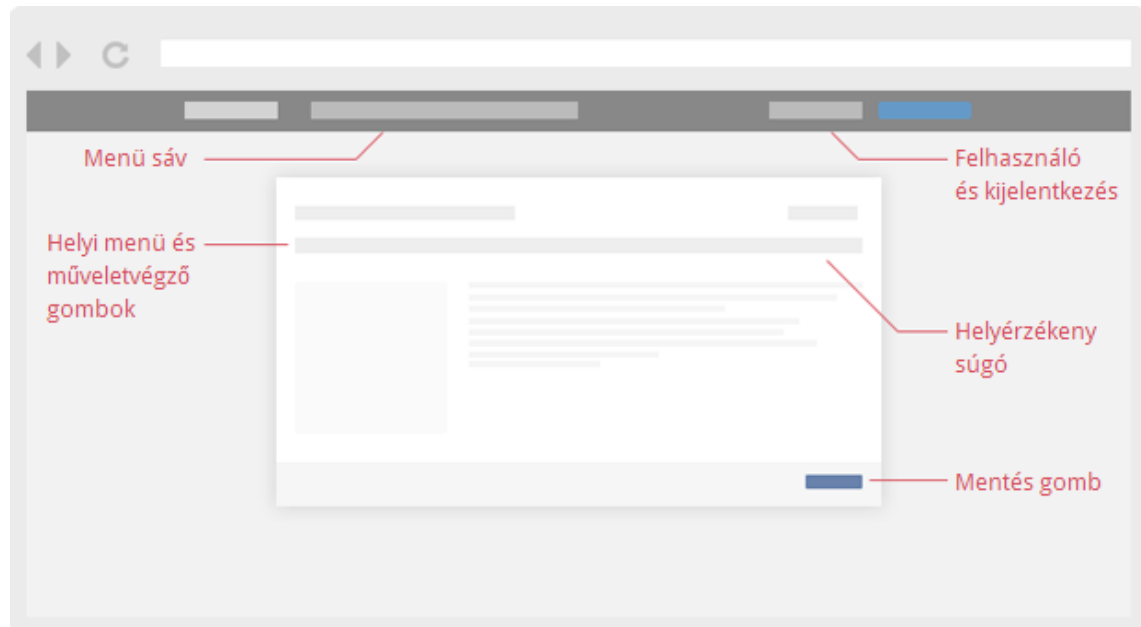
4-2. ábra: Bejelentkezés és elfelejtett jelszó

A telepítés során megadott, vagy a későbbiekben létrehozott felhasználói fiókok e-mail címét és jelszavát használva léphet be a rendszerbe.

Ha elfelejtette jelszavát, használja az „*Elfelejtette jelszavát?*” hivatkozást. Az itt megadott e-mail címre a CMS kiküldi a belépéshez szükséges új adatait.

### 4.3. A felület használata

A MicroSite CMS úgy lett kialakítva, hogy honlapja gyors, átlátható és egyszerű szerkesztését lehetővé tegye.



4-3. ábra: Adminisztrációs felület

A bejelentkezést követően a felület felső részén egy sötét menüsáv jelenik meg, mely bal szélén az adminisztrációs oldalakat, jobb szélén pedig az aktuális felhasználó nevét és a kijelentkezés gombot tartalmazza. Ez a sáv minden oldal tetején látható marad, amíg be van jelentkezve a rendszerbe.

Amennyiben Ön nem otthoni gépéről lépett be, a kívánt módosítások befejeztével használja a kilépés funkciót! Ezzel lezárja az adminisztrációs munkamenetet és elkerülheti, hogy honlapját egy nyilvános gépen más is módosíthassa.

A menüsávban kiválasztott menüpont alapértelmezés szerint egy lebegő menüben jelenik meg az oldal tartalma felett. (Vagy a honlap tartalmába beágyazva.)

A lebegő menü tetején annak címe, bal felső sarkában pedig a hely érzékeny súgó és a bezárás gomb található. A súgó mindig az aktuális lap tartalmának megfelelő segítséget nyújt. Használja bátran, ha valamiben bizonytalan!

Ezek alatt a helyi menü és műveletvégző gombok szerepelnek. Ennek bal oldalán könnyedén navigálhat mappák közt, jobb oldalán pedig mindig az ablak témájának megfelelő gombokat láthatja, mint: új oldal létrehozása, fájl feltöltése, stb.

Az ablak alján, általában külön sávban, a mentés és mégse gombokat találja.



## 4.4. Adminisztrációs műveletek

### 4.4.1. Beállítások módosítása

**Oldalbeállítások:** általános, a weblapokra együttesen vonatkozó beállítások.

**Honlapot zárolása:** nagyobb átalakítás esetén a teljes weboldal tartalma zárolható a felhasználók elől. Ha értéke „igen”, akkor a „Karbantartás” üzenet jelenik meg minden oldal helyén.

**Honlap neve:** általában személy vagy cégnév

**Szlogen, mottó:** rövid, humoros, könnyen megjegyezhető, kulcsszavakat tartalmazó szöveg

**Cím formátum:** a weboldalak címének formátumát szabhatja testre itt. A sitename, page és slogan szavakat tetszőleges sorrendben, elválasztó jelekkel használhatja. (Pl: page – sitename) Ilyenkor a rendszer kicseréli a szavakat a megfelelő oldal címére és a honlap nevére.

**Egyedi metatag:** bizonyos esetekben szükség lehet egyedi, ún. meta címkék használatára, mint Google Site Verification azonosító vagy Open Graph címkék elhelyezésére. Ezek elhelyezése itt valósítható meg.

**Általános oldalsáv:** Minden oldalhoz rendelhető egy oldalsáv a felületen. Ezek a tartalomtól elkülönített, kiegészítő információk, dobozok. Az általános oldalsáv tartalma minden oldal oldalsávjának tetején megjelenik.

**Lábléc tartalma:** Az oldalak alján található információk szerkeszthetők itt egységesen.

**Környezeti beállítások:** a honlap speciális beállításai.

**Oldalak nyelve:** a publikus weblapok alapértelmezett nyelve.

**Időzóna:** a felvitt adatok ideje melyik időzóna szerint legyen megjelenítve.

**Dátum formátum:** dátumok megjelenítési formátumának megadása a PHP date() függvény jelölésének megfelelően.

**Karakter készlet:** karakterek kódlapja. Ha furcsa karakterek jelennek meg a szövegben, változtassa meg ezt a beállítást.

**URL átírás:** az internetcímek megjelenítési formátumát állíthatja be itt. Figyelem, a beállítás megváltoztatása minden oldal elérését megváltoztatja. Ha a változtatások után nem tudja elérni oldalait, használja az következő formátumú internetcímet: [www.encegem.hu/index.php?/admin&reset](http://www.encegem.hu/index.php?/admin&reset)

**Statisztika:** statisztikai szolgáltatások, látogatottság mérők elhelyezése a felületen.

**Közösség:** megosztás, Like, +1 gombok és közösségi modulok elhelyezése a felületen.

**Email küldés:** a küldés módjának megadása (SMTP levélküldő szerver esetén a kapcsolódási adatok itt tölthetők ki)

**Honlap e-mail címe:** alapértelmezett e-mail cím, amiről a rendszer a leveleket küldi.

**Küldés módja:** a leveleket küldhet létező e-mail fiókból SMTP szolgáltatáson keresztül vagy a PHP alapértelmezett levelezőjén keresztül. Előbbi akkor hasznos, ha web szerver szolgáltatója nem biztosítja, vagy korlátozza a levelek küldését.

A szerver, átjáró, azonosító és jelszó mezők kitöltése csak SMTP szolgáltatás használata esetén szükséges.

**Regisztrációs e-mail:** a felhasználók regisztrációkor ezt az e-mail sablont kapják kézhez. A sablonban használható kulcsszavak, melyek automatikusan kitöltődnek: {name} – név, {mail} – email cím, és {pass} – jelszó.

**Elfelejtett jelszó e-mail:** a felhasználók elfelejtett jelszó igényükkor ezt az e-mail sablont kapják kézhez. A sablonban használható kulcsszavak, melyek automatikusan kitöltődnek: {name} – név, {mail} – email cím, és {pass} – jelszó.

**Felhasználó törlése e-mail:** a felhasználók a törléskor ezt az e-mail sablont kapják kézhez. A sablonban használható kulcsszavak, melyek automatikusan kitöltődnek: {name} – név.

**Adatbázis:** az adatbázis kapcsolódási adatainak megadása itt lehetséges. Figyelem, az adatok módosítása elérhetetlenné teheti a honlap oldalait, bővítményeit és a felhasználók adatait.

Szerver, azonosító, jelszó és adatbázis név mezők kitöltése csak akkor szükséges, ha már a telepítéskor adatbázis szolgáltatással használta a honlapot.

#### 4.4.2. Oldalak kezelése

**Megtekintés:** a listában az oldalak nevére kattintva azok megjeleníthetők az adminisztrációs módban is.

**Kezdőoldal kiválasztása:** a listában az oldal sorában rádiógomb segítségével lehetséges.

**Oldal elrejtése:** a listában az oldal sorában a jelölőnégyzettel elrejthetők egyes oldalak a nyilvánosság elől.

**Oldal létrehozása:** új oldal létrehozásához használja a helyi menü gombját. Az új oldalra a későbbiekben a címéből automatikusan kialakított URL cím alapján hivatkozhat. Pl: [www.encegem.hu/szulo/uj-oldal-cim](http://www.encegem.hu/szulo/uj-oldal-cim). Ennek utólagos módosítására nincs lehetőség! Létező oldal nem írható felül. A művelethez szerkesztői jogosultság szükséges.

**Módosítás:** az egyes oldalak címének, leírásának, oldalsávjának és tartalmának módosítása a listában az oldal neve mellett található módosítás ikonnal érhető el. Az új ablakban egy grafikus szövegszerkesztővel könnyedén szerkesztheti oldalait.

**Törlés:** oldalak törléséhez kattintson a listában az oldal neve mellett szereplő törlés ikonra. A kezdőoldal kivételével minden oldal törölhető, ha szerkesztői jogosultsággal rendelkezik.

**Menü beállítása:** a honlap menüjének szerkesztése. A menüben megjelenő oldalak sorrendjét "kattints és húzd" módszerrel megváltoztathatja, egymásba ágyazhatja, illetve felirataikat is átnevezheti. Új menüelemek hozzáadásához illetve törléséhez használja a legördülő menüket.

#### 4.4.3. Fájlok kezelése

**Listázás, léptetés:** könyvtárak közt navigálhat, ha a mappákra kattint. Azt, hogy éppen melyik mappában tart, a helyi menü jobb oldalán láthatja. A fájlokra kattintva, azok letölthetők vagy tartalmuk megjelenik egy új ablakban. A listák fejlécére kattintva név, méret és dátum szerint rendezheti a fájlokat.

**Feltöltés és létrehozás:** új fájlok feltöltéséhez illetve új könyvtárak létrehozásához használja a helyi menü gombját. Létező fájl vagy könyvtár nem írható felül. A művelethez szerkesztői jogosultság szükséges. Mindig ellenőrizze, hogy a megfelelő könyvtárban kezdte-e meg a műveletet. Fájlok feltöltése esetén egyszerre többet is kiválaszthat a Ctrl billentyű nyomva tartása mellett, ha böngészője azt támogatja.

**Átnevezés, áthelyezés:** a fájlok vagy könyvtárak átnevezése a listában, a nevük sorában szereplő ikonra kattintva kezdeményezhető. Ekkor a név szerkeszthetővé válik a teljes útvonallal. Az OK gomb megnyomásával véglegesíthető a folyamat, kivéve, ha a cél fájl vagy könyvtár már létezik. Szerkesztői jogosultság szükséges a művelethez.

**Törlés:** a fájlok és mappák törléséhez kattintson a listában a nevük sorában szereplő törlés ikonra. A nem üres mappák nem törölhetők. A művelethez szerkesztői jogosultság szükséges.

**Képek szerkesztése:** a listában, a képfájlok mellett egy szerkesztő ikon jelenik meg, melyet megnyomva a kép átméretezése és részletekre vágása lehetséges.

#### 4.4.4. Bővítmények kezelése

**Megtekintés:** a bővítmények, felületek és nyelvek előnézeti képének illetve részleteinek (pl.: szerző, verzió, technológia) megtekintése, a listában, azok nevére történő kattintással lehetséges.

**Bővítmény feltöltése:** új, külső személyek által készített bővítmények (modulok, felületek és nyelvek) feltöltése a helyi menü megfelelő gombjának megnyomásával lehetséges. Új bővítményeket a rendszer [www.webmake.hu/microsite](http://www.webmake.hu/microsite) hivatalos honlapjáról szerezhet be. A feltöltéshez szerkesztői jogosultság szükséges.

**Módosítás:** a bővítmények beállításainak testre szabásához (saját felületükön), valamint az egyes felületek és nyelvek fájljainak szerkesztéséhez a listában a megfelelő sorban válassza ki a módosítás ikont. Ezek a funkciók csak az adminisztrátor jogosultságú felhasználók számára érhetők el.

A felületek stílusának szerkesztéséhez a CSS leírónyelv, a nyelvi fájlok tartalmának módosításához az INI leírónyelv ismerete szükséges.

**Törlés:** a bővítmények törléséhez kattintson a listában a sorokban szereplő törlés ikonra. Minden modul, felület (kivéve az alapértelmezettet) és minden nyelv (kivéve az utolsót) törölhető adminisztrátori jogosultsággal.

**Globális bővítmények:** az itt megadott bővítmények a honlap minden oldalán betöltésre kerülnek. Csak akkor tegyen globálissá egy bővítményt, ha az szükséges, a nem használtakat pedig távolítsa el, így javíthat oldalai betöltési sebességén.

**Bővítmény engedélyezése:** a feltöltött bővítmények alapértelmezés szerint nincsenek engedélyezve, az esetleges hibák elkerülése és gyorsabb betöltés végett. Ha használni szeretne egy bővítményt, a listában a jelölőnégyzet segítségével engedélyezze azt. Ehhez adminisztrátori jogosultság szükséges.

**Ikon lecserélése:** a honlap ikonjának (általában a füleken és a címsorban látszó 16x16px méretű kép) megváltoztatása a Felületek menüpont helyi menüjében lehetséges, adminisztrátori jogosultsággal. Figyelem, az új ikon felülírja a régit! Az ikon tetszőleges képfájlból generálható.

**Alapértelmezett felület:** a honlap megjelenése az a felület lesz, amelyik a listában rádiógomb segítségével alapértelmezettként van kiválasztva. Ennek megváltoztatásához adminisztrátori jogosultság szükséges.

#### 4.4.5. Felhasználók kezelése

**Profil létrehozása:** új szerkesztő szintű felhasználói fiók hozható létre a helyi menüben, ha az adott email cím még nincs regisztrálva a rendszerben. A fiók létrehozásakor tájékoztató e-mail kerül kiküldése. Új fiók létrehozásához adminisztrátori jogosultság szükséges.

**Módosítás:** fiókja beállításainak módosításához válassza ki saját profilját a listából, majd a sorban kattintson a módosítás gombra. A beállítások megváltoztatásához az adott fiók tulajdonosának kell lenni. Az adminisztrátor más fiókok rangját is képes módosítani.

**Profil neve:** az ön saját neve, vagy a profil típusa. Ez fog látszani szerkesztéskor.

**Email cím:** létező e-mail cím, mely a profil azonosítója is lesz egyben. Ide kerül kiküldésre az elfelejtett jelszó is.

**Jelszó:** belépéshez használt jelszó

**Profil rangja:** különböző műveletekhez különböző jogosultsági szint szükséges. (Adminisztrátor – teljes körű, Szerkesztő – bizonyos funkciók, Felhasználó – nem módosíthat semmit.)

**Profil nyelve:** az adminisztrációs felület alapértelmezett nyelve a felhasználó számára.

**Modern felület:** lebegő adminisztrációs ablakok engedélyezése

**Modern szerkesztő:** Wordhöz hasonló grafikus szövegszerkesztők engedélyezése

**Törlés:** a felhasználók törléséhez a listában a megfelelő sor végén a törlés gombot kell megnyomni. Minden felhasználó törölhető az adminisztrátor által vagy önmaga által, kivéve az utolsó adminisztrátort.

## 4.5. A rendszer karbantartása

**Szerver állapot:** a telepítés során látott képernyő, mely a rendszer állapotát hivatott jelezni. A hibás vagy hiányzó feltételekről itt kap bővebb tájékoztatást, ha a hiba fennáll.

**Hozzáférési napló:** a felhasználói bejelentkezések naplózásra kerülnek idő, e-mail cím és hely (ip, host) alapján. Ezeket az adatokat itt lehet lekérdezni. Ha ismeretlen időpontot vagy címet lát felhasználói fiókja mellett, változtassa meg jelszavát!

**Hibanapló:** a rendszer használata során felmerült hibaüzenetek itt érhetőek el. A hibák főként a helytelenül működő bővítményektől származhatnak. Ha ilyet lát, távolítsa el a megnevezett bővítményt vagy konzultáljon a webmesterével.

**Mentés és helyreállítás:** a teljes rendszerről biztonsági másolat készíthető, ha a felhasználó adminisztrátori ranggal rendelkezik. A korábban készített biztonsági másolatokból a megrongálódott honlap helyreállítható. Figyelem, az összes jelenlegi adat felülírásra kerül ilyenkor. A használathoz adminisztrátori rang szükséges.

**Frissítés keresése:** a rendszerhez a későbbiek során elkészített javítások, újabb verziók itt tölthetők fel. A honlap adatai a frissítés során érintetlenek maradnak. Erősen javasolt, hogy akkor végezze a frissítést és telepítést, amikor nem várható látogató. Különben előfordulhat, hogy azok számára is megjelenik a telepítő menü és átveszik honlapja felett az irányítást.

Megjegyzés 1: Abban az esetben, ha a frissítés sikertelen, csatlakozzon tárhelyéhez egy megfelelő programmal, majd törölje a „*datas/config.php*” fájlt. A honlap beállításait ezzel nullázza, és el tudja indítani a telepítést a következő formátumú internetcímmel: [www.enceqem.hu/index.php?/admin/install&reset](http://www.enceqem.hu/index.php?/admin/install&reset)

Megjegyzés 2: Ha egyik felhasználó sem tud bejelentkezni a frissítést követően, használja az alapértelmezett [admin@enceqem.hu](mailto:admin@enceqem.hu) e-mail címet és az „*admin123*” jelszót. Abban az esetben, ha ez sem használ, törölje a „*datas/users.php*” fájlt, majd próbálja ismét.

## 5. Összefoglalás, továbbfejlesztési lehetőségek

A kezdeti célkitűzések közt egy interaktív web alkalmazás létrehozása szerepelt, mely ellátja a kisebb bemutatkozó jellegű oldalak létrehozását és szerkesztését.

A rendszer képes több felhasználót kezelni, azoknak különböző jogokat biztosítani. Megoldja a fájlok feltöltését, képek, és egyéb multimédiás tartalmak a honlapba illesztését. Előzetes technikai ismeretek nélkül szerkeszthetővé teszi a weblapokat a felhasználók számára egy grafikus szövegszerkesztővel.

Az alkalmazás moduláris kialakítású, bővítményeket és felületeket, valamint nyelvi fájlokat külön telepíteni lehet hozzá. Ezzel együtt rugalmassága és sokrétű felhasználása garantált.

A CMS gyors és biztonságos. Az adatokat adatbázisban képes tárolni, és – a fellelhető nyílt forrású vetélytársaival ellentétben – pontosan annyi sor kódból áll, amennyi a legegyszerűbb honlap működéséhez szükséges.

Fejlesztői szempontból az alkalmazás jól felépített. Az az adminisztrációs felület elkülöníthető az alkalmazás magjától. A megjelenítésért felelős mag objektum orientált, így a benne szereplő forráskódok könnyen hordozhatóak, tovább bővíthetőek.

Az kész rendszer használata során felmerült egy megoldatlan probléma, mellyel a tervezéskor nem foglalkoztam. A bővítmények lehetséges többnyelvűségét nem vettem számításba. Ez ugyan kiküszöbölhető, azok a nyelvi fájlokba történő integrálásával, ám belátható, hogy ez erősen rontaná a rendszer rugalmasságát.

A legjobb megoldás erre a bővítmények illesztő felületének kibővítése lenne. Így azok megkapnák a rendszer aktuális nyelvi beállítását és a bővítmények a saját mappájukban elhelyezett nyelvi fájlokat töltenének be.

Továbbfejlesztési lehetőségként merült fel bennem az oldalak úgynevezett inline (helybeli) szerkesztése. Ennek lényege, hogy a különálló grafikus szerkesztő helyett közvetlenül a lapok szövege válna szerkeszthetővé. Erre már léteznek elég jó megoldások. Többek közt az általam használt TinyMCE szerkesztőben is.

Zárásul a rendszer jellemzőit áttekintve – véleményem szerint – a kezdeti célkitűzéseknek megfelelő, a magánszektor illetve a kis és középvállalkozások számára jól használható, bemutatkozó oldalak készítésére alkalmas, tartalomkezelő rendszert sikerült létrehoznom.

Az alkalmazás kipróbálható változata elérhető a [www.webmake.hu/demo/microsite](http://www.webmake.hu/demo/microsite) honlapon illetve jelen szakdolgozat CD mellékleteként.

## 6. Irodalomjegyzék, ábrajegyzék, mellékletek

### Irodalomjegyzék

[1] Dr. Tarczali Tünde - UML diagramok a gyakorlatban .....	8
[2] W3Techs statisztikája, 2014. március – <a href="http://w3techs.com">http://w3techs.com</a> .....	13
[3] PHP – Web fejlesztő szkript nyelv és feldolgozó – <a href="http://php.net">http://php.net</a> .....	13
[4] DB-Engines Ranking, 2014. március – <a href="http://db-engines.com">http://db-engines.com</a> .....	13
[5] MySQL – Adatbázis kezelő rendszer és lekérdező nyelv – <a href="http://www.mysql.com">http://www.mysql.com</a> .....	13
[6] HTML és CSS leíró nyelvek - <a href="http://www.w3.org/standards/webdesign/htmlcss">http://www.w3.org/standards/webdesign/htmlcss</a> .....	18
[7] CodeMirror – JavaScript alapú kódszerkesztő - <a href="http://codemirror.net">http://codemirror.net</a> .....	19
[8] TinyMCE – JavaScript alapú grafikus szövegszerkesztő - <a href="http://www.tinymce.com">http://www.tinymce.com</a> .....	19
[9] jQuery – JavaScript függvénykönyvtár - <a href="http://jquery.com">http://jquery.com</a> .....	19
[10] Bootstrap – Front-end keretrendszer - <a href="http://getbootstrap.com">http://getbootstrap.com</a> .....	19
[11] jCrop – jQuery bővítmény - <a href="http://deepliquid.com/content/Jcrop.html">http://deepliquid.com/content/Jcrop.html</a> .....	19
[12] Krammer Gergely – A szoftver minőségi jellemzői – <a href="http://krammer.web.elte.hu">http://krammer.web.elte.hu</a> .....	21

### Ábrajegyzék

3-1. ábra: Használati eset diagram .....	8
3-2. ábra: Felhasználói felület és menü .....	9
3-3. ábra: Adatbázis séma.....	12
3-4. ábra: Adatbázis felépítése .....	13
3-5. ábra: A rendszer elvi kialakítása .....	14
3-6. ábra: Felhasználók osztály diagram.....	15
3-7. ábra: Beállítások osztály diagram .....	15
3-8. ábra: Nyelvek osztály diagram.....	15
3-9. ábra: Oldalak osztály diagram .....	15
3-10. ábra: Felületek osztály diagram.....	16
3-11. ábra: Bővítmények osztály diagram .....	16
3-12. ábra: Bővítmény illesztő osztály diagram.....	17
3-13. ábra: Az adminisztrációs felület elvi vázlata.....	18
4-1. ábra: A rendszer telepítése.....	22
4-2. ábra: Bejelentkezés és elfelejtett jelszó .....	23
4-3. ábra: Adminisztrációs felület .....	24

## Mellékletek

1db CD melléklet, melynek tartalma:

tartalom.txt	– az adathordozó tartalom jegyzéke
szakdolgozat.pdf	– jelen szakdolgozat digitális állománya
felhasznaloi-dokumentacio.pdf	– telepítési útmutató és felhasználói dokumentáció
microsite-cms-rendszer.zip	– a rendszer és forráskódjának tömörített állománya