

Szakdolgozat

Temesvári Imre
2009

Szakdolgozat

Temesvári Imre

Mérnök-informatikus, A modul, levelező

Kecskeméti Főiskola
Gépipari és Automatizálási Műszaki Főiskolai Kar
Kecskemét
2009

Kecskeméti Főiskola
GAMF Kar
Kiadó szakcsoport

Szakedolgozati feladatlap

Név: EHA-kód, munkarend:

Szak: Szakirány:

A szakedolgozat címe:

Az elkészítés helye:

Belső konzulens:

Külső konzulens:

A feladat részletezése:

Kecskemét, 200

P. H.

.....
belső konzulens

.....
szakcsoportvezető

Tartalomjegyzék

Bevezetés.....	6
1. A Yii bemutatása.....	7
1.1 Miért a Yii?.....	7
1.2 A Yii története.....	7
1.3 Yii lehetőségei, jellemzői.....	8
1.4 Yii teljesítmény-mérés.....	10
1.5 Yii telepítése lépésről-lépésre.....	11
2. Feladat specifikáció.....	14
2.1 Használati eset diagram.....	14
2.2 Használati esetek - forgatókönyvek.....	15
2.2.1 Kezdőlap megnyitása.....	15
2.2.2 Regisztráció.....	16
2.2.3 Bejelentkezés.....	16
2.2.4 Kijelentkezés.....	16
2.2.5 Szócikk keresése, korábbi változatok megtekintése.....	16
2.2.6 Új szócikk felvitele, szócikk javítás.....	17
2.2.7 Fórum megtekintése.....	17
2.2.8 Fórumtéma hozzáadása.....	18
2.2.9 Fórum – új hozzászólás/válasz.....	18
2.2.10 Személyes adatok módosítása.....	18
2.2.11 Felhasználó kezelés.....	19
2.2.12 Szócikk, téma, hozzászólás törlése.....	19
3. Felhasználói felület tervezése.....	20
3.1 Főoldal.....	20
3.2 Keresés.....	21
3.3 Szócikk megjelenítése.....	22
3.4 Regisztráció.....	23
3.5 Sikeres regisztráció.....	23
3.6 Sikertelen regisztráció.....	24
3.7 Belépett felhasználó.....	25
3.8 Új szócikk szerkesztése.....	26
3.9 Meglévő szócikk szerkesztése.....	27
3.10 Fórum főoldal.....	28
3.11 Hozzászólások.....	29
3.12 Személyes adatok módosítása.....	30
3.13 Karbantartó főoldala.....	31
3.14 Adminisztrációs oldalak.....	32
4. Tervezés.....	33
4.1 Szócikk verzió tábla.....	35
4.2 Szócikk tábla.....	35
4.3 Felhasználó tábla.....	36
4.4 Fórumtéma tábla.....	36
4.5 Hozzászólások tábla.....	37
4.6 Kapcsolatok.....	37
5. Kódolás.....	38
5.1 Sajat alkalmazáskönyvtár létrehozása.....	38

5.2 Alapvető beállítások.....	38
5.3 Scaffolding.....	39
5.4 A model lehetőségei.....	40
5.4.1 Oszlopok tulajdonságainak ellenőrzése:.....	40
5.4.2 Tulajdonságok átadása:.....	41
5.4.3 Kapcsolatok:.....	41
5.4.4 Tulajdonságcímkék:.....	42
5.5 Controller lehetőségei.....	43
5.5.1 Funkciók.....	43
5.5.2 Hozzáférési szabályok:.....	44
5.6 Portletek használata.....	44
5.6.1 Regisztráció.....	45
5.6.2 UserLogin.....	45
5.6.3 UserMenu.....	45
5.7 Hiba jelző oldalak.....	46
6. Üzembehelyezés, tesztelés.....	47
6.1 Üzembehelyezés.....	47
6.2 Tesztelés.....	47
6.2.1 Kezdőlap megnyitása.....	47
6.2.2 Regisztráció.....	47
6.2.3 Bejelentkezés.....	48
6.2.4 Kijelentkezés.....	48
6.2.5 Szócikk keresése, korábbi változatok megtekintése.....	48
6.2.6 Új szócikk felvitele, szócikk javítása.....	49
6.2.7 Fórum megtekintése.....	49
6.2.8 Fórumtéma hozzáadása.....	49
6.2.9 Új hozzászólás/válasz.....	49
6.2.10 Személyes adatok módosítása.....	50
6.2.11 Felhasználó-kezelés.....	50
6.2.12 Szócikk, téma, hozzászólás törlése.....	50
Irodalomjegyzék.....	52
Melléklet.....	53

Bevezetés

Szakedolgozatom célja és témája egy olyan informatikai tudásbázis elkészítése, amely az informatika minden területéről összegyűjti a szakszavakat, fogalmakat, kifejezéseket – abból a célból, hogy az érdeklődőnek pontos információt adhasson. Jellege tudásbázis – ami annyival több, mint egy adatbázis vagy informatikai szótár, hogy nem egy valaki, vagy egy zárt csoport fejleszti - bővíti, hanem mindenki, aki az oldalt felkeresi, és érdemben tud rajta javítani, hozzá tenni. A cél egy minél bővebb tudásbázis létrehozása, ahol az érdeklődő gyorsan megtalálhatja a magyarázatot az általa nem ismert szóra, fogalomra.

A tudásbázis Yii alapokon fog működni, ami egy nem túl régi és még nem annyira elterjedt PHP keretrendszer. A keretrendszer jelentősen meggyorsíthatja a fejlesztés menetét, és az előre elkészített osztályok, kódok felhasználásával segíti a fejlesztő munkáját. Így a szakedolgozat másik célja a Yii bemutatása, megismertetése.

1. A Yii bemutatása

1.1 Miért a Yii?

A szakdolgozat írásának időpontjában az interneten elérhető php keretrendszerek száma 95 körül van¹. Választani objektív szempontok alapján szinte lehetetlenség, a legtöbb magyar és angol nyelvű fórum hozzászólója a kérdező választani kívánónak azt a választ adja, hogy próbáljon ki 2-3 elismertebb, népszerűbb rendszert és úgy válasszon. Sőt, saját magának is írhat egyet, ha eléri az ehhez szükséges tudásszintet. Ezek alapján érthető, hogy ennyi keretrendszer érhető el, bár kívánatosabb lenne, ha a sok szerteágazó, különböző (és kevésbé különböző) fejlesztés helyett a tudást és az energiákat csoportosítani lehetne és összeadni, más hasznos célok érdekében (is). Mellesleg megjegyzem, hogy a különböző oldalon író-fórumozó emberek véleménye szerint² az elsők között a CakePHP, a CodeIgniter, a Kohana és a Zend Framework szerepel.

1.2 A Yii története

A Yii egy viszonylag újabb fejlesztés, 2008.december 3-án adták ki a fejlesztők az 1.0.0-ás stabil és nyilvánosságnak szánt verziót³, maga a fejlesztés 2008. január 1-től indult. A név jelentése: „Yes, it is! – Yii!, azaz EZAZ! A fejlesztők szerint a nevet kiejtve Yeee ([ji:]) jelképezi a Yii tulajdonságait: easy, efficient, extensible, azaz könnyű, hatékony és bővíthető. A Yii készítője a Yii Software LLC csapat, keretrendszerüket szabad-szoftverként hozzák forgalomba, BSD licenc alatt, ami a GNU/GPL licenzhez képest tartalmaz néhány megkötést. Pl. a szoftver bármilyen újrahasznosítása során fel kell tüntetni a szerzői jogok tulajdonosát, bármilyen formában történő továbbhasznosítás során a program/csomag mellett szerepelnie kell a szerzői jogoknak, a licenznek.

A Yii felhasznál és vesz át ötleteket más keretrendszerektől: Prado (mint fő forrás),

1 [1] http://en.wikipedia.org/wiki/List_of_web_application_frameworks

2 [3] <http://avnetlabs.com/php/php-framework-comparison-benchmarks>

3 [7] <http://www.yiiframework.com/download/>

Ruby on Rails, jQuery (alapból támogatja), Symfony, Joomla.

A saját honlapján olvasható bemutatkozó szerint⁴ „a Yii egy nagy teljesítményű, komponens alapú, PHP keretrendszer, mely web alkalmazások széles skálájának fejlesztéséhez alkalmazható. Leredukálja a fejlesztési időt, a kódok újrafelhasználhatóságát elősegíti, könnyű, hatékony és rugalmas.” Persze a legtöbb rendszer így hirdeti magát, és igyekszik a felhasználó-webprogramozó kegyeibe férközni. Ugyanakkor ezek a jelzők nem teljesen alaptalanok, hisz ezeknél a keretrendszereknél közös alap a PHP, annak lehetőségeit jól kiaknázva, és a keretrendszert jól felépítve hasznos eszköz kerül a webfejlesztő kezébe. Azt, hogy a sok variáció közül ki melyiket választja, valóban a szubjektív benyomások és tapasztalatok határozzák meg. A CodeIgniter használata után, a Yii első ismerkedésre mintha egy kicsit nehezkesebb-bonyolultabb lett volna, de a dokumentáció elolvasása, ill. a yii blog nevű példaalkalmazás elkészítése és megismerése után már jobban tudtam használni. Bár elsőre furcsa volt, hogy a keretrendszer és a fejlesztendő alkalmazás egymástól elválasztva, külön könyvtárban van, a Yii-t jobban megismerve ez előny lehet egy frissítésnél, mert a keretrendszer fájljait felül lehet írni az újabb változattal, de az alkalmazást nem kell. Megjegyzés: Kíváncsiságból kipróbáltam egy frissítést, miután a honlap elkészült, szinte tökéletesen működött minden funkció, egy - két hibát leszámítva (az igazsághoz hozzá tartozik, hogy a frissítés még béta állapotú volt).

A legtöbb segítség a Yii megismeréséhez nem is a dokumentációból meríthető (amely sajnos legtöbbször nélkülözi a példákat), hanem a keretrendszer osztályaiból, metódusai-ból. Van olyan php forrásfájl, amelyben több a magyarázat, mint maga a programkód (pl: yii/framework/web/helpers/CHtml.php).

1.3 Yii lehetőségei, jellemzői

A Yii az objektum orientáltságra helyezi a hangsúlyt. A keretrendszer felépítése és így az általa létrehozott webalkalmazás működése az MVC sémát követi, tökéletesen elválasztva egymástól a webalkalmazás működését, az adatokat és a megjelenítést.

4 [7] <http://www.yiiframework.com/about/>

A jellemzőket illetően: installáció, dokumentáció, kompatibilitás, adatbázisok támogatása, MVC, konfigurálási lehetőségek, session kezelése, biztonság - jó és kiváló minősítést kaphat. Külön kiemelem, hogy tartalmazza a JQuery javascript könyvtárat, javascript kód írásának megkönnyítéséhez.

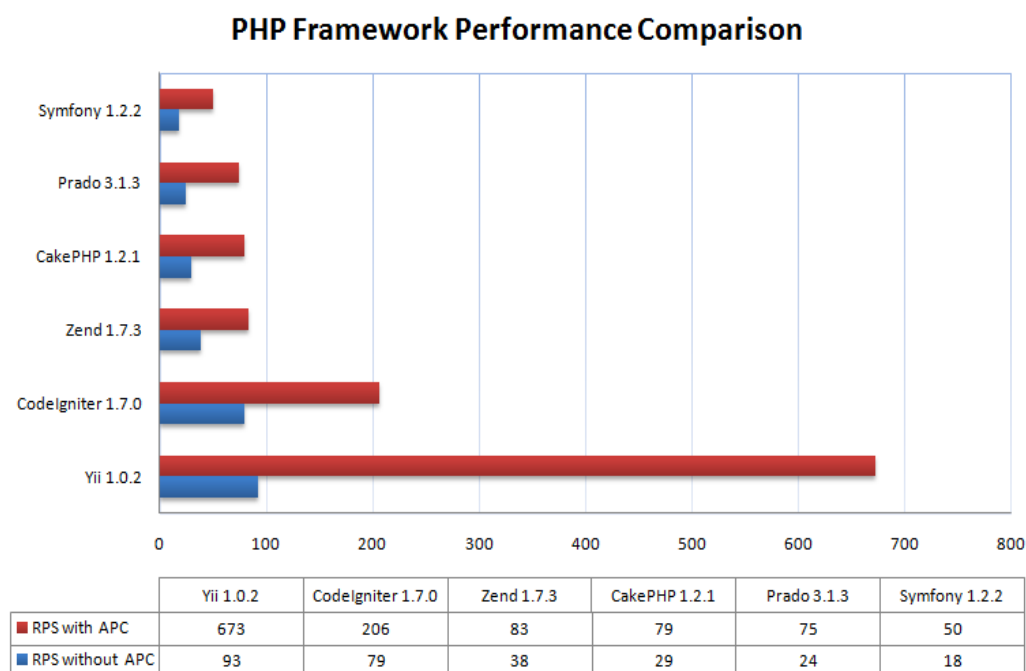
Talán a dokumentáció az, ami első átolvasás után egy kicsit szokatlan, hisz pl. a CodeIgniter dokumentációjához és helper-jeihez hasonlítva a Yii saját honlapjának „documentation” menüpontja alatt gyűjt össze minden lényeges információt és osztályt. A honlapon található egy tutorial, a Yii „szakácskönyv”, valamint fórum, ahol több nyelv között van magyar szekció is. Ha értékelni kellene, a CodeIgniter kiváló dokumentációjához képest a Yii „elég jó” - írja Fred Wu⁵. Tény, hogy elég részletes a dokumentáció, de egy pdf fájlba ömlesztve kicsit nehezebben kezelhető, mint az előbb említett keretrendszer „helper”-je. Persze idővel és a Yii elterjedtségével arányosan nyilván a dokumentáció és a honlapon elérhető információk is bővülni fognak.

Nagyon hasznosnak bizonyulnak a letölthető bővítmények, amelyeket kész PHP kód-ként töltenek fel szerzőik és tölthetnek le a Yii-t használók. Jelenleg 73 bővítmény érhető el.

A Yii kipróbálható demókkal van csomagolva, ill. letölthető egy ún. „Yii appliance”, virtuális alkalmazás, amely tulajdonképpen egy virtuális gépen (VMware) futó szerver, és tartalmazza magát a Yii-t néhány bemutató alkalmazással, PHP-t, httpd-t, PostgreSQL-t és Linux kernelt.

A Yii lehetőségei közül kiemelik a szerzők a cache-megoldások kifinomultságát, és ezzel ajánlják a Yii-t nagy forgalmú honlapok, portálok motorjának. Ezt alátámasztandó találhatunk egy diagramot is a teljesítmény demonstrálására:

5 [4] <http://www.beyondcoding.com/2009/03/02/choosing-a-php-framework-round-2-yii-vs-kohana-vs-codeigniter/>



1. ábra: Yii teljesítmény-diagram

A fejlesztők saját mérése szerint⁶ a Yii teljesítménye túlszárnyalja a konkurens rendszereket – ahogy az 1. ábra mutatja.

1.4 Yii teljesítmény-mérés

A Yii teljesítményének lemérésére készítettem egy rövid tesztet. Létrehoztam a webalkalmazás nyitólapját elkészítettem mind Yii-ben, mind Code Igniterben. Mind a kettő keretrendszerben a controller megjeleníti a kezdő oldalt, amihez több php parancsot kell lefuttatniuk. A tesztelést az Apache Benchmark nevű programmal végeztem, ami alkalmas arra, hogy egy webszerveren található webalkalmazás keretrendszerét tesztelje.

Tesztkörnyezet: egy Fedora Core 11 linux szerveren futó Apache webszerveren helyeztem el a két teszt weblapot. Helyi hálózaton keresztül, más zavaró forgalom kizárásával végeztem a tesztelést, a következő paranccsal: `ab -n 1000 -c 5 localhost/probayii ill. localhost/probaci`. Ezekkel a kapcsolókkal az Apache Benchmark 1000 kérést küld el a szerver-

⁶ [8] <http://www.yiiframework.com/performance/>

nek 5 egyidejű kapcsolatot létrehozva. Ezt a tesztet mind a két keretrendszeren hússzor futtattam le. Tíz tesztet bekapcsolt APC-vel, tíz tesztet pedig anélkül, a minél nagyobb pontosság elérése érdekében. Az eredmény az 1. táblázatban látható.

1. táblázat: Mérési eredmény

	Yii	Code Igniter
APC	216,87 RPS	222,12 RPS
APC nélkül	53,94 RPS	57,17 RPS

Látható, hogy bár Code Igniter jobban teljesített ugyanolyan megjelenítendő weblapnál, nagy különbség nem tapasztalható. Lehet, hogy nagy forgalmú, bonyolultabb weblapoknál esetleg más a helyzet, de mérésemet alátámasztják más weboldalak szerzői is, akik CakePHP-vel és CodeIgniter-rel, ill. Kohana-val hasonlítják össze a Yii teljesítményét. Az RPS, azaz másodpercenkénti lekérések számában ők is hasonló teljesítményt érnek el⁷. Nyilván nehéz objektív tesztet elvégezni, valószínűleg a fejlesztők máshogy mértek, de ez nem von le semmit a Yii használati értékéből.

1.5 Yii telepítése lépésről-lépésre

1. A honlapról jelenleg a legfrissebb verzióként az 1.0.10-es változatot tölthetjük le, tar-gzip vagy zip által tömörített változatban. Használatához valamilyen webszerver szükséges (a fejlesztők az Apache alatt tesztelték és azt ajánlják), minimum a PHP 5.1.0-ás változatával.
2. A webszerver által elérhető könyvtárba történő kibontás után három könyvtárat és fájlokat kapunk. Először érdemes a böngészőbe az <elérési út>/yii/requirements/index.php-t begépelni és megnyitni. Ez az oldal leellenőrzi, hogy minden szükséges kiegészítő rendelkezésre áll-e a keretrendszer használatához.
3. Ezután egy parancsértelmezőt megnyitva a következő parancsot kell lefuttatni a

⁷ [2] <http://daniel.carrera.bz/2009/01/comparison-of-php-frameworks-part-i/>

/wwwroot/yii/framework/ - könyvtárba belépve: *yiic webapp /wwwroot/alkalmazasneve*. Ezzel elindítjuk a *yiic* tool nevű alkalmazást, mely a megadott „alkalmazasneve” könyvtárban létrehozza a tulajdonképpeni készítendő alkalmazás vázát a *webapp* kapcsoló segítségével, a keretrendszerből generálva. Ez sajnos nem működik kielégítően. A parancs fájl a php értelmezőt próbálja meghívni mind windows, mind linux rendszer alatt, de ha a php értelmező könyvtára nincs az elérési utak között, akkor nem fog a parancs lefutni. Megoldás: vagy be kell állítani az elérési útvonalat (operációs rendszertől függően), vagy a fenti parancs elé be kell szűrni az *útvonal/php* kiegészítést. Erre a dokumentáció telepítési útmutatója nem tér ki külön, a *yiiblog* minta alkalmazás dokumentációja viszont már igen.

4. Ha a parancs sikeresen lefutott, akkor böngészőn keresztül a webszerverre csatlakozva, ezt az alkalmazás „vázat” el tudjuk érni a */szervercim/alkalmazasneve* címen. Egy próba/teszt oldal jelenik meg, mely tájékoztat a további teendőkről.

A létrejött váz könyvtárstruktúrája a következő:

oninfo_yii/	
index.php	a webalkalmazás belépési pontja
assets/	közös fájlok helye
css/	stílus lapok
images/	képek
themes/	témák
protected/	az alkalmazás védett fájljai
yiic	yiic tool
yiic.bat	yiic tool windowshoz
commands/	yiic parancsok
shell/	a yiic shell parancsai
components/	belső alkalmazások, portletek
views/	view fájlok a portletekhez
config/	konfigurációs fájlok
main.php	fő konfigurációs fájl
controllers/	kontrollerek

SiteController.php	alapértelmezett kontroller
extensions/	kiegészítések
messages/	nyelvi fájlok
models/	model fájlok
runtime/	ideiglenes fájlok
views/	megjelenítésért felelős fájlok
layouts/	megjelenítésért felelős fájlok
site/	SiteControllerhez tartozó view fájlok
index.php	a SiteController „actionIndex”-hez tartozó view fájl
system/	a hibaüzenetet tartalmazó view fájlok

Az alkalmazás egyetlen belépési pontja a `/wwwroot/alkalmazasneve/index.php`. A `protected` könyvtárban elhelyezett adatokat (models, views, controllers, config) nem lehet más módon kívülről elérni. Több alkalmazást is készíthetünk ugyanahhoz a keretrendszerhez rendelve – az előbb leírt módon létrehozott „váz” egy-egy saját alkalmazás kódjának, anyagainak helye külön könyvtárban. A keretrendszer fájljait, lehetőségeit pedig közösen használhatják. A létrejött könyvtárban a `protected/config/main.php` fájlban lehet az alapvető beállításokat változtatni, többek között a keretrendszer elérési útvonalát az alkalmazás könyvtárához képest, az adatbázis típusát, alkalmazás nevét.

Ezek után el lehet kezdeni a saját alkalmazás fejlesztését.

2. Feladat specifikáció

A honlap tervezése és megalkotása előtt érdemes néhány fontos dolgot tisztázni a honlappal kapcsolatban⁸.

Az informatika világában rengeteg szakszó, rövidítés található, amely között a külső szemlélő nem egykönnyen ismeri ki magát. A honlap célja ezeket a fogalmakat könnyen és gyorsan kereshetően, elérhetően - és szerkeszthetően is akár-, az érdeklődőnek megmagyarázni. Tekintve az óriási tudáshalmazt e téren, nem egy szerkesztő vállára helyezi a karbantartás terheit, hanem több emberére, aki szívesen részt vesz ebben. Az oldalnak nem célja más nagy, hasonló jellegű oldalakkal versenyezni, csakis az informatika világába elmélyülni a lehető legnagyobb alaposággal, és segíteni azokat a látogatókat, akik egy informatikai fogalom, szó jelentését keresik. Minden látogatónak lehetősége lesz egyben szerzővé is válni, aki ezeket a fogalmakat pontosíthatja, ill. a hiányzó meghatározásokat megfogalmazhatja. A cél az, hogy minél több szócikk, minél pontosabban szerepeljen az adatbázisban. Tudásbázis jellegét pedig az erősítheti, ha minél többen teszik hozzá a saját ismeretüket. Ez az adott szócikknél látszani fog, mint korábbi változat/szócikk történet.

Az oldal céljainak megfelelően legfontosabb funkció a gyors és pontos keresés lesz és a fórum használata. Ez utóbbi a regisztráció végrehajtásával érhető majd el, mely által az olvasó is szerkesztővé válhat és nem csak olvashatja a fórumot, hanem hozzá is szólhat.

2.1 Használati eset diagram

Az oldal felépítését és működését legjobban a használati eset diagram segítségével lehet modellezni, leírni. A webalkalmazás használati eseti diagramja a 2. ábrán látható.

⁸ [6] <http://nagygusztav.hu/honlap-interju-mi-az-oldal-celja>



2. ábra: Használati eseti diagram

A webalkalmazásban a diagramon feltüntetett funkciók lesznek megvalósítva.

2.2 Használati esetek - forgatókönyvek

2.2.1 Kezdőlap megnyitása

Felhasználó a böngészőben a lap címét begépel, megjelenik a kezdőoldal egy köszöntő szöveggel középen, bal oldalon egy sávban a menüpontokkal és a belépés, ill. a regisztráció ablakával.

2.2.2 Regisztráció

Felhasználó a regisztrációra kattint, megjelenik egy űrlap. Felhasználó kitölti a megfelelő adatokkal az űrlap sorait. Ezután a „Létrehoz” gombra kattint. Amennyiben minden szükséges adatot helyesen beírt, megjelenik a „Sikeres regisztráció” felirat. Ha valami hiányzik, vagy nem megfelelő szintaktikájú, felhasználó visszakapja az űrlapot.

Ha a felhasználó a „Mégse” gombra kattint, megjelenik a kereső oldal.

2.2.3 Bejelentkezés

Felhasználó a főoldalon beírja azonosítóját és jelszavát, majd a „Belépés” gombra kattint. Helyes adatok megadása esetén megjelenik a felhasználói menü és a főoldal.

Helytelen vagy hiányzó adatok esetén megjelenik az „Azonosító/Jelszó ismeretlen/hiányzik” üzenet.

2.2.4 Kijelentkezés

Bejelentkezett felhasználó a „Kilépés” gombra kattint. Felhasználó kijelentkezik az oldalról. A „Belépés” és „Regisztráció” menüpontok megjelennek, a személyes menü és a „Kilépés” gomb eltűnik.

2.2.5 Szócikk keresése, korábbi változatok megtekintése

Felhasználó a „Keresés” menüpontra kattint. Betöltődik a kereső oldal. A kereső mezőbe felhasználó beírja a keresendő kifejezést vagy betűket, és <enter>-t üt, vagy megnyomja a „Keresés” gombot. Megjelenik a találati lista, mely üresen marad, ha nincs találat. Ha van, a listában megjelennek a keresendő szónak vagy betűknek megfelelő szócikkeknek a címei, hivatkozásként. Felhasználó rákattint egy címre. Új oldalként betöltődik a megfelelő szócikk. Itt a felhasználó elolvashatja a tartalmat-meghatározást, a létrehozás dátumát, a módosítás dátumát (ha volt), a szerző nevét (vagy aki utol-

jára módosította) és a kereső szavakat.

A szócikk táblázata felett, három menüpont jelenik meg: „Új szócikk”, „Szócikk javítása”, „Szócikk előző változatai”.

Felhasználó rákattint a „Szócikk előző változatai” menüpontra. Betöltődik egy új oldal, ahol megjelennek a szócikk korábbi változatai, időrendi sorrendben.

2.2.6 Új szócikk felvitele, szócikk javítás

Felhasználó bejelentkezik. Az „Új szócikk menüpont megjelenik a bejelentkezett felhasználó menüpontjai között, ill. egy szócikk táblázata felett is. Felhasználó rákattint. Megjelenik egy űrlap, három szövegbeviteli mezővel és három gombbal. Felhasználó megadja a szócikk – fogalom nevét a cím mezőnél. Ezután beírja a tartalmat. Végül begépeli a keresőszavakat.

Felhasználó a „Létrehoz” gombra kattint. Ha minden mezőt kitöltött, az új szócikk létrejön, és megjelenik a tartalma.

Felhasználó az „Előnézet” gombra kattint. Az Űrlap alatt megjelenik a szócikk előnézete.

Felhasználó a „Mégse gombra kattint. Az űrlap eltűnik, betöltődik a kereső oldal.

Szócikk javítása:

Felhasználó a szócikk tartalma oldalon a „Szócikk javítása” linkre kattint. Megjelenik egy űrlap, ahol a mezőkben a szócikk címe, tartalma és a keresőszavak láthatóak. A cím nem szerkeszthető. Felhasználó módosítja a tartalmat, a keresőszavakat, majd a „Ment”, „Előnézet”, vagy a „Mégse” gombra kattint.

2.2.7 Fórum megtekintése

Felhasználó a Fórum menüpontra kattint. Megjelenik a fórum főoldala. Listaként egymás alatti sorokban a különböző témák vannak felsorolva. Felhasználó rákattint egy sorra, megjelennek a témához kapcsolódó hozzászólások. A fórumtémák fölött egy menüpont látszik: az „új fórumtéma.”

2.2.8 Fórumtéma hozzáadása

Az „Új fórumtéma” linkre kattint a bejelentkezett felhasználó. Egy egyszerű űrlap jelenik meg, ahol felhasználó megadja az új téma címét, majd a „Létrehoz” gombot nyomja meg. Ha a cím nem hiányzik, akkor az új téma hozzászólás oldala jelenik meg.

2.2.9 Fórum – új hozzászólás/válasz

Felhasználó egy fórumtémára kattint. Megjelennek a hozzászólások, egy oldalon tíz hozzászólás, időrendben. Ha tíznél több van, megjelennek a lapozást segítő számok és az „Előző”, „Következő” gombok. Az oldal alján található egy szöveglap. Bejelentkezett felhasználó beírja a hozzászólást és a „Mehet” gombra kattint. Amennyiben az ablak nem üres, az új hozzászólás bekerül a fórumba és megjelenik a hozzászólások listája.

Válasz: felhasználó egy hozzászólás „Válasz” linkjére kattint. Megjelenik a megválaszolni kívánt hozzászólás és alatta egy űrlap. Felhasználó beírja a választ, majd a „Mehet” gombra kattint. Megjelenik a hozzászólások listája.

2.2.10 Személyes adatok módosítása

Bejelentkezett felhasználó rákattint a „Személyes adatok módosítása” menüpontra. Megjelenik egy, a regisztrációhoz hasonló űrlap. Az űrlap tartalmazza a felhasználó adatait. Az azonosító nem módosítható. Felhasználó módosítja az adatokat.

Felhasználó megnyomja a „Ment” gombot. Ha valami hiányzik, az űrlap újra megjelenik, hibamegjelöléssel. Felhasználó kijavítja, újra megnyomja a „Ment” gombot. Megjelenik a főoldal.

Felhasználó a „Mégse” gombot nyomja meg. Megjelenik a kereső oldal.

2.2.11 Felhasználó kezelés

Karbantartó bejelentkezik, az ő menüpontjai között megjelenik az „Adminisztrációs oldalak”. Rákattint, ekkor megjelenik az adminisztrációs oldal, ahol egymás alatti sorokban megjelennek a következő linkek: „Szócikk_verzió”, „Felhasználó”, „Fórumtémák”, „Hozzászólások”. Karbantartó a „Felhasználók” linkre kattint. Betöltődik a felhasználó adminisztrációs oldal. Megjelenik táblázatos formában az összes felhasználó adata. A táblázat utolsó oszlopa fogja tartalmazza a műveleteket, úgymint „Új”, „Módosít”, „Töröl”. Ha Karbantartó az „Új” és a „Módosít”-ra kattint, a regisztrációnál ill. módosításnál leírt eset történik.

Karbantartó a „Töröl” linkre kattint. Megjelenik egy figyelmeztető ablak.

Karbantartó az „Igen” gombot választja. A törölt felhasználó státusza egyesről nullára változik, ill. jelszava értéke nulla lesz.

Karbantartó a „Mégse” gombot nyomja meg. A figyelmeztető ablak eltűnik.

2.2.12 Szócikk, téma, hozzászólás törlése

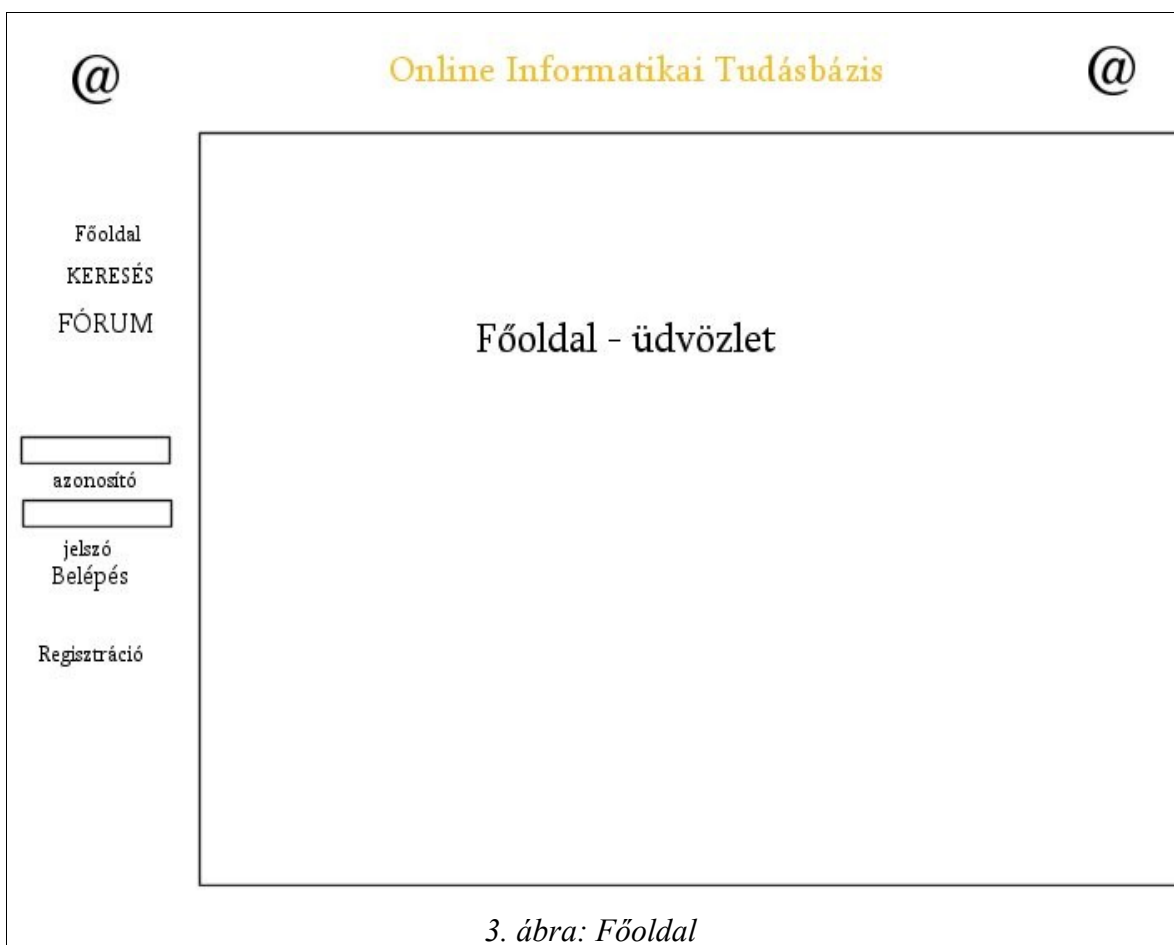
Karbantartó bejelentkezik, az ő menüpontjai között megjelenik az „Adminisztrációs oldalak.” Rákattint, ekkor megjelenik az adminisztrációs oldal, ahol egymás alatti sorokban megjelennek a következő linkek: „Szócikk_verzió”, „Felhasználó”, „Fórumtémák”, „Hozzászólások.” A szócikk, téma, hozzászólások esetei hasonlóképpen játszódnak le, mint a felhasználókezelés.

3. Felhasználói felület tervezése

A felhasználói felület tervezésekor a könnyű áttekinthetőségre és az egyszerű használatot segítő külalakra törekszem.

A most következő oldalakon az oldalak tervezeteit mutatom be, vonalrajzszerű ábrákon, egyfajta logikai sorrendet követve.

3.1 Főoldal



3. ábra: Főoldal

A webalkalmazás nyitó képernyője tartalmazni fog középen egy köszöntőt, általános tájékoztatót. A baloldali sávban lesz a menü, ahonnan el lehet érni keresést, a fórumot és a főoldalt. A menü alatt található a regisztráció, ill. regisztrált felhasználóknak a belépésre szolgáló mező.

3.2 Keresés

@ Online Informatikai Tudásbázis @

Főoldal
KERESÉS
Fórum

azonosító
jelszó
Belépés
Regisztráció

Keresés

Találati lista

Eredmény1
Eredmény2
Eredmény3
...

4. ábra: Keresés

A sorban a „keresés eredménye” ablak a következő, mert regisztráció nélkül is lehet ezt a szolgáltatást használni. Ha a keresés ablakba beírt szó megtalálható az adatbázisban - mint keresőszó, akkor a szócikk címe - neve a találati listában jelenik meg, hivatkozásként magára a szócikk tartalmára.

3.3 Szócikk megjelenítése

Online Informatikai Tudásbázis

Főoldal
KERESÉS
Fórum

azonosító
jelszó
Belépés

Regisztráció

Szócikk címe
[Új szócikk] [Szócikk javítása] [Szócikk előző változatai]

Cím
Tartalom
Létrehozás dátuma
Módosítás dátuma
Szerző

5. ábra: Szócikk

A szócikk tartalma és jellemzői táblázatos formában fognak megjelenni. A cím alatt a tartalmat lehet olvasni, a létrehozás módosítás dátumát és a szerző nevét. A táblázat felett lesz három link. A bejelentkezett felhasználó ezek segítségével új szócikket hozhat létre, ill. az éppen olvasottat módosíthatja. A „Szócikk előző változatai”-ra kattintva bárki elérheti az éppen olvasott szócikk előző változatait, hasonló megjelenésű oldalon. Ez a funkció hasznos lehet, ha valaki végig akarja követni egy szócikk fejlődését, vagy a karbantartó visszaállíthatja az utolsó jó változatot, ha véletlen (szándékos) törlés történt (ill. a szerkesztésnél valaki kinullázta a szöveget).

3.4 Regisztráció

6. ábra: Regisztráció

A regisztráció gombra kattintva a regisztrációs oldalra jutunk. Az oldal fő részén fognak elhelyezkedni a beviteli mezők, ahol a szükséges adatokat kell megadni a sikeres regisztrációhoz. A „Mehet” gomb megnyomásával indítható a regisztrálás, ha mégse kíván a látogató regisztrálni, a „Mégse” gombra kattintva a főoldalra jut vissza.

3.5 Sikeres regisztráció

Sikeres regisztráció esetén a „Regisztráció sikerült” szöveg jelenik meg, majd a főoldal töltődik be.

3.6 Sikertelen regisztráció

Online Informatikai Tudásbázis

Javítsd a következőket:

Azonosító

Jelszó

Jelszó újra

E-mail cím

Teljes név

Ellenőrző kód *vfieuqv* [Új kód kérése](#)

Főoldal
KERESÉS
Fórum

azonosító

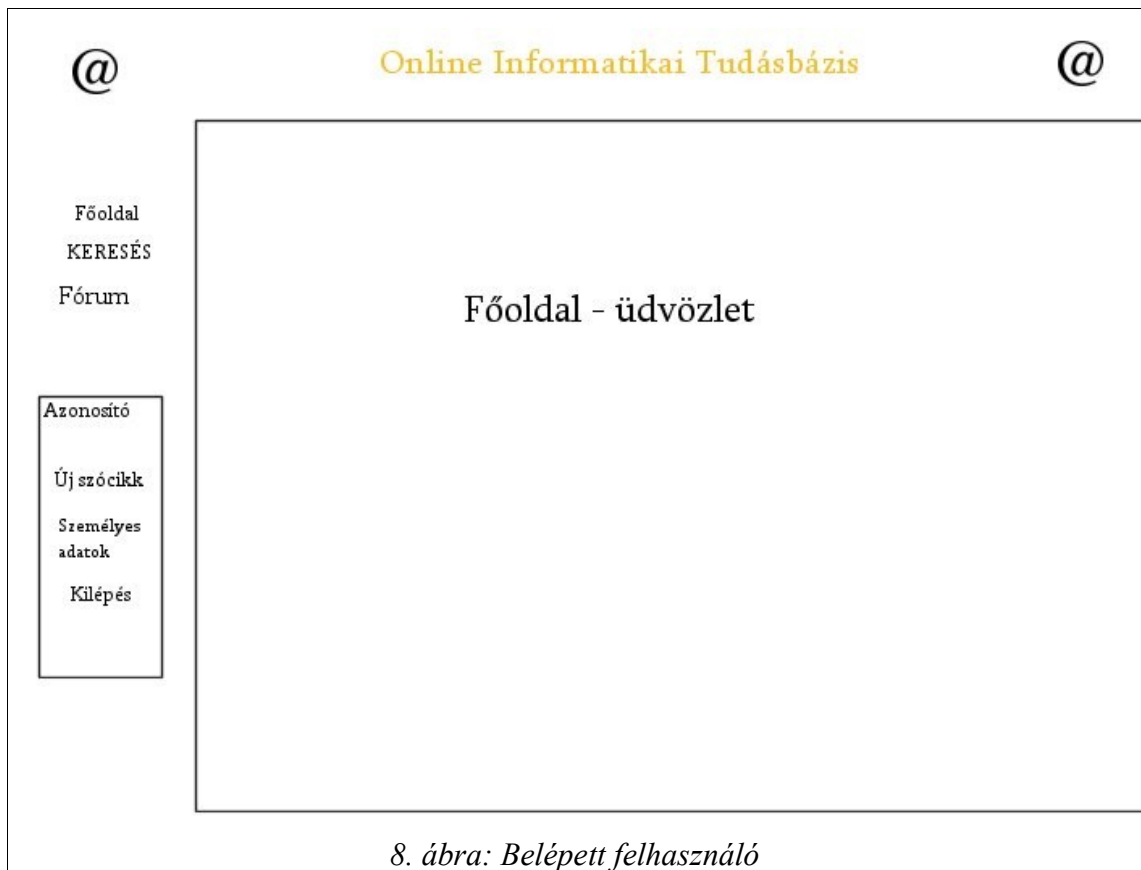
jelszó
Belépés

Regisztráció

7. ábra: Sikertelen regisztráció

A sikertelen regisztráció okai lehetnek: valamelyik adat hiányzik, már létező azonosító lett beírva, a jelszavak nem egyeznek, az e-mail cím formája nem megfelelő. Bármelyik esetben újra a regisztrációs ablak jelenik meg, szöveges üzenetet kiírva a hiba vagy hibák okáról, és pirosra színezve a hibás sort.

3.7 Belépett felhasználó



Ha egy regisztrált felhasználó bejelentkezik, akkor a bal oldali menüben a „Bejelentkezés” és „Regisztráció” gombok helyett megjelenik az „Új szócikk”, „Személyes adatok” és a „Kilépés” menüpont.

3.8 Új szócikk szerkesztése

The screenshot shows a web interface for creating a new article. At the top, it says 'Online Informatikai Tudásbázis' between two '@' symbols. On the left, there is a sidebar with links: 'Főoldal', 'KERESÉS', 'Fórum', 'Azonosító', 'Új szócikk', 'Személyes adatok', and 'Kilépés'. The main content area is titled 'Új szócikk' and contains a form with the following elements:

- A warning: 'A *-gal jelölt mezők kitöltése kötelező' (Fields marked with * are mandatory).
- A text input field labeled 'Cím*' (Title).
- A large text area labeled 'Tartalom*' (Content).
- A text input field labeled 'Keresőszó*' (Keywords).
- Three buttons at the bottom: 'Létrehoz' (Create), 'Előnézet' (Preview), and 'Mégse' (Cancel).

9. ábra: Új szócikk

Új szócikk szerkesztéséhez ez az oldal fog megjelenni. A „Létrehoz” gombot megnyomva rögtön tárolásra kerül. Az „Előnézet” gomb hatására az űrlap alatt megjelenik a szerző, dátum, tartalom - még a mentés előtt.

3.9 Meglévő szócikk szerkesztése

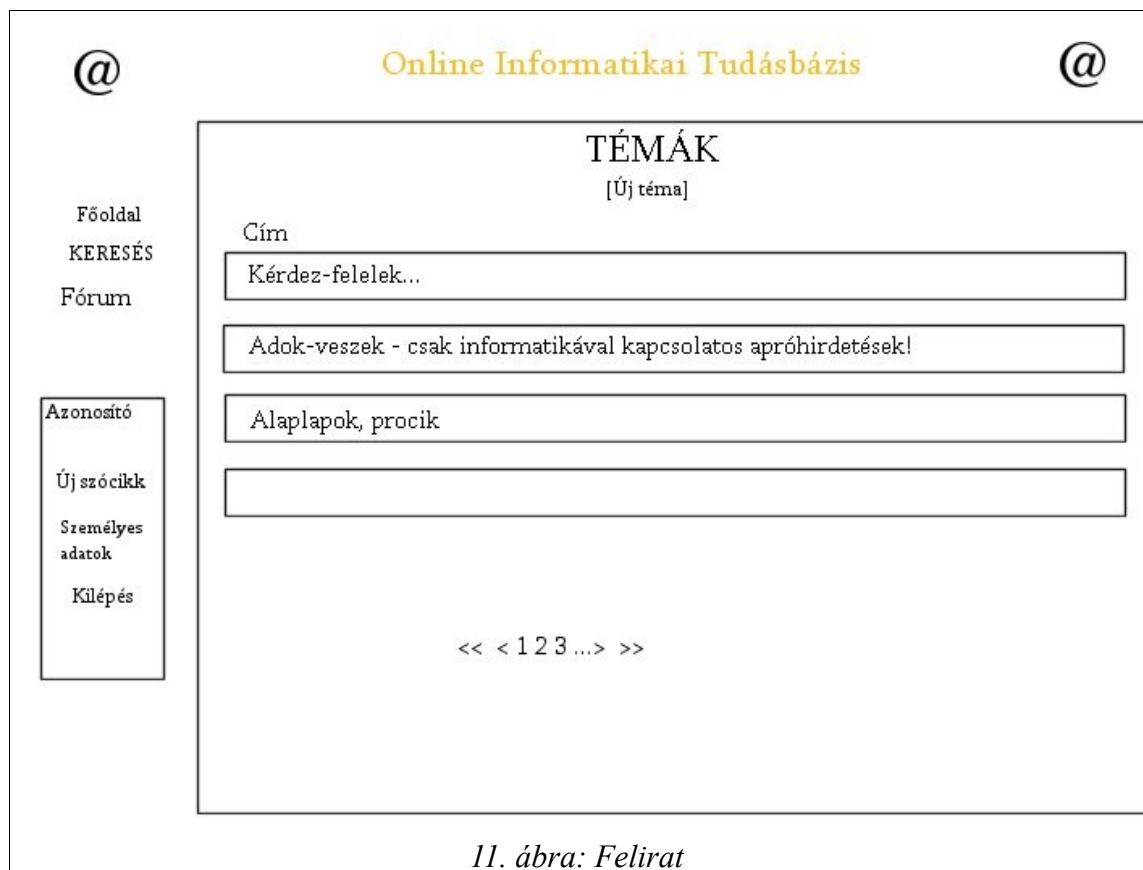
The screenshot shows a web interface for editing an article. At the top, the page title is 'Online Informatikai Tudásbázis' flanked by '@' symbols. On the left, a sidebar contains navigation links: 'Főoldal', 'KERESÉS', 'Fórum', 'Azonosító', 'Új szócikk', 'Személyes adatok', and 'Kilépés'. The main content area is titled 'Új szócikk' and contains a form with the following elements:

- A warning message: 'A *-gal jelölt mezők kitöltése kötelező' (Fields marked with * are mandatory).
- A 'Cím*' (Title) field with a text input box.
- A 'Tartalom*' (Content) field with a large text area.
- A 'Keresőszó*' (Keywords) field with a text input box.
- Three buttons at the bottom: 'Ment' (Save), 'Előnézet' (Preview), and 'Mégse' (Cancel).

10. ábra: Szócikk szerkesztése

Ezen az oldalon lehet módosítani, javítani, kiegészíteni egy adott szócikk szövegét, mely a szerkesztő ablak megfelelő mezőiben jelenik meg. A címet nem lehet módosítani. Az új szócikkhez képest pedig nem „Létrehoz”, hanem „Ment” gomb található az oldalon.

3.10 Fórum főoldal



11. ábra: Felirat

Minden felhasználó elérheti a fórumot is, a bal oldali menü fórum menüpontjára kattintva. A fórum főoldalán a fő témák vannak csoportosítva. A regisztrált felhasználók az „Új téma” link segítségével új témát indíthatnak. A címekre kattintva jelennek meg a hozzászólások. Ha a témák címei nem férnek ki egy oldalra, akkor megjelennek az oldallapozó gombok.

3.11 Hozzászólások

The screenshot shows a forum interface. At the top, the title 'Online Informatikai Tudásbázis' is centered between two '@' symbols. Below the title, the forum topic is 'Hozzászólások Adok-veszek' with a sub-header '[Fórumtémák...]'. A pagination control shows '<< < 1 2 3 ... > >>'. A post by 'Geiza' is displayed, dated '2009.november 15. 12:25.', with the text 'Eladó Celeron D 1.7-es alaplappal.' and links for 'Előzmény' and 'Válasz erre'. Below the post are two empty rectangular boxes. At the bottom, there is a 'Új hozzászólás' section with a text input field and a 'Mehet' button. On the left side, there is a vertical menu with links: 'Főoldal', 'KERESÉS', 'Fórum', 'Azonosító', 'Új szócikk', 'Személyes adatok', and 'Kilépés'.

12. ábra: Hozzászólások

A hozzászólások oldalain időben növekvő sorrendben jelennek meg a vélemények. Az oldal alján van lehetőség új hozzászólásra is.

3.12 Személyes adatok módosítása

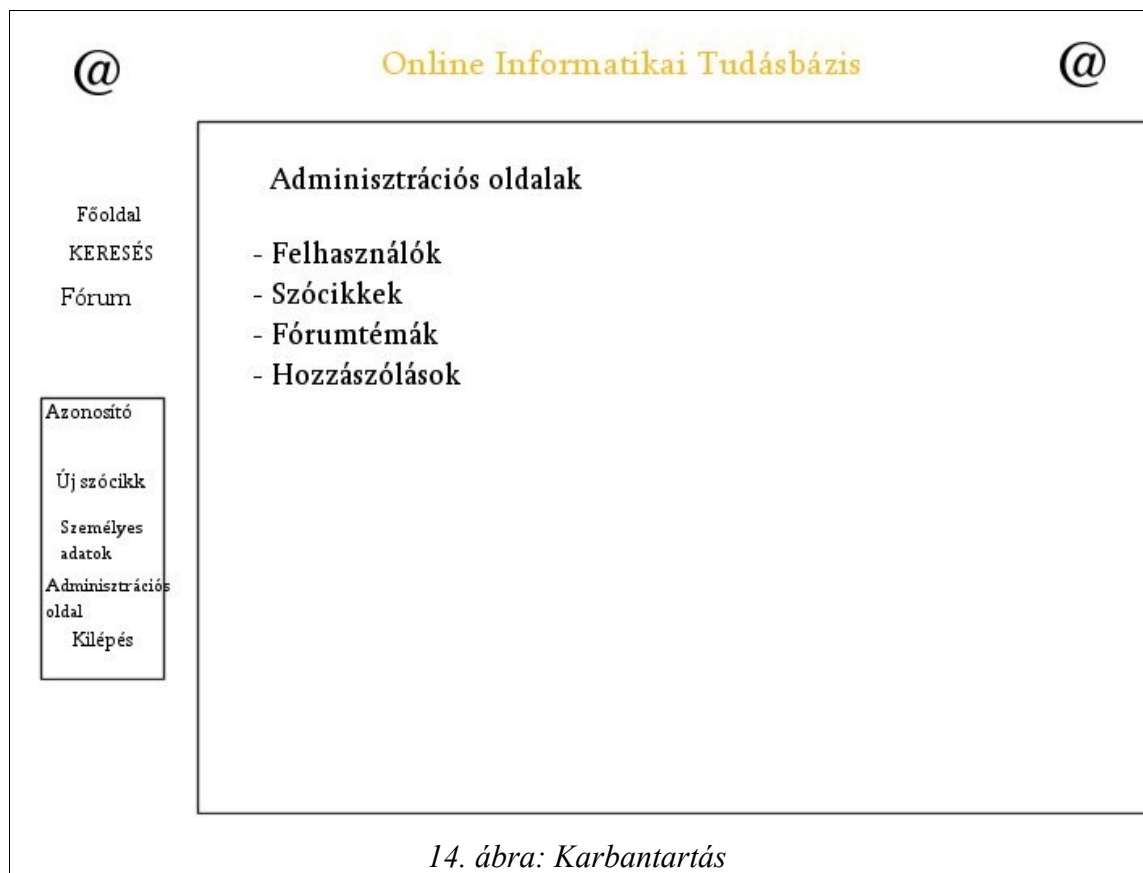
The screenshot shows a web interface for editing user data. At the top, the page title is 'Online Informatikai Tudásbázis'. The main heading is 'Adatok módosítása'. On the left, there is a sidebar with navigation links: 'Főoldal', 'KERESÉS', 'Fórum', 'Azonosító', 'Új szócikk', 'Személyes adatok', and 'Kilépés'. The main form contains the following fields and controls:

- Azonosító** (Username): A label with the text 'Azonosító' next to it, but no input field is visible.
- Jelszó** (Password): Input field containing '*****'.
- Jelszó újra** (New Password): Input field containing '*****'.
- E-mail cím** (Email): Input field containing 'cim'.
- Teljes név** (Full Name): Input field containing 'név'.
- Ellenőrző kód** (CAPTCHA): The code 'vfieuqv' is displayed next to an input field.
- Új kód kérése** (Request new code): A button located below the CAPTCHA input field.
- Ment** (Save): A button at the bottom of the form.
- Mégse** (Cancel): A button at the bottom of the form.

13. ábra: Adatok módosítása

A felhasználó személyes adatai módosításakor hasonló űrlapot lát, mint a regisztrációkor, de itt az azonosítót nem tudja megváltoztatni, csak a többi adatot.

3.13 Karbantartó főoldala



A karbantartó bejelentkezésekor a menüpontok között megjelenik egy új menüpont, amit csak ő láthat: az „Adminisztrációs oldal.” Erre kattintva négy linken keresztül elérheti a felhasználók, a szócikkek, a fórumtémák és a hozzászólások adminisztrációs oldalát.

3.14 Adminisztrációs oldalak

The screenshot shows the 'Adminisztrációs oldalak' page. At the top, the title 'Online Informatikai Tudásbázis' is displayed in yellow. Below it, the page title 'Adminisztrációs oldalak' is centered. A table lists user data with columns for 'Userid', 'Azonosító', 'Jelszó', 'Email', 'St.', and 'Művelet'. The table contains three rows of user data. Below the table is a pagination control showing '<< < 1 2 3 ... >> >>'. On the left side, there is a sidebar with navigation links: 'Főoldal', 'KERESÉS', 'Fórum', 'Azonosító', 'Új szócikk', 'Személyes adatok', 'Adminisztrációs oldal', and 'Kilépés'. The page is framed by '@' symbols in the top corners.

Userid	Azonosító	Jelszó	Email	St.	Művelet
1	proba	34thvnnv54n854	sfdsf@fd.hu	1	Új Módosít Töröl
2	Demo	54chn4u5hvg49nvg9		1	Új Módosít Töröl
3	jd	chfg9v4hn945n		0	Új Módosít Töröl

<< < 1 2 3 ... >> >>

Adminisztrációs oldal

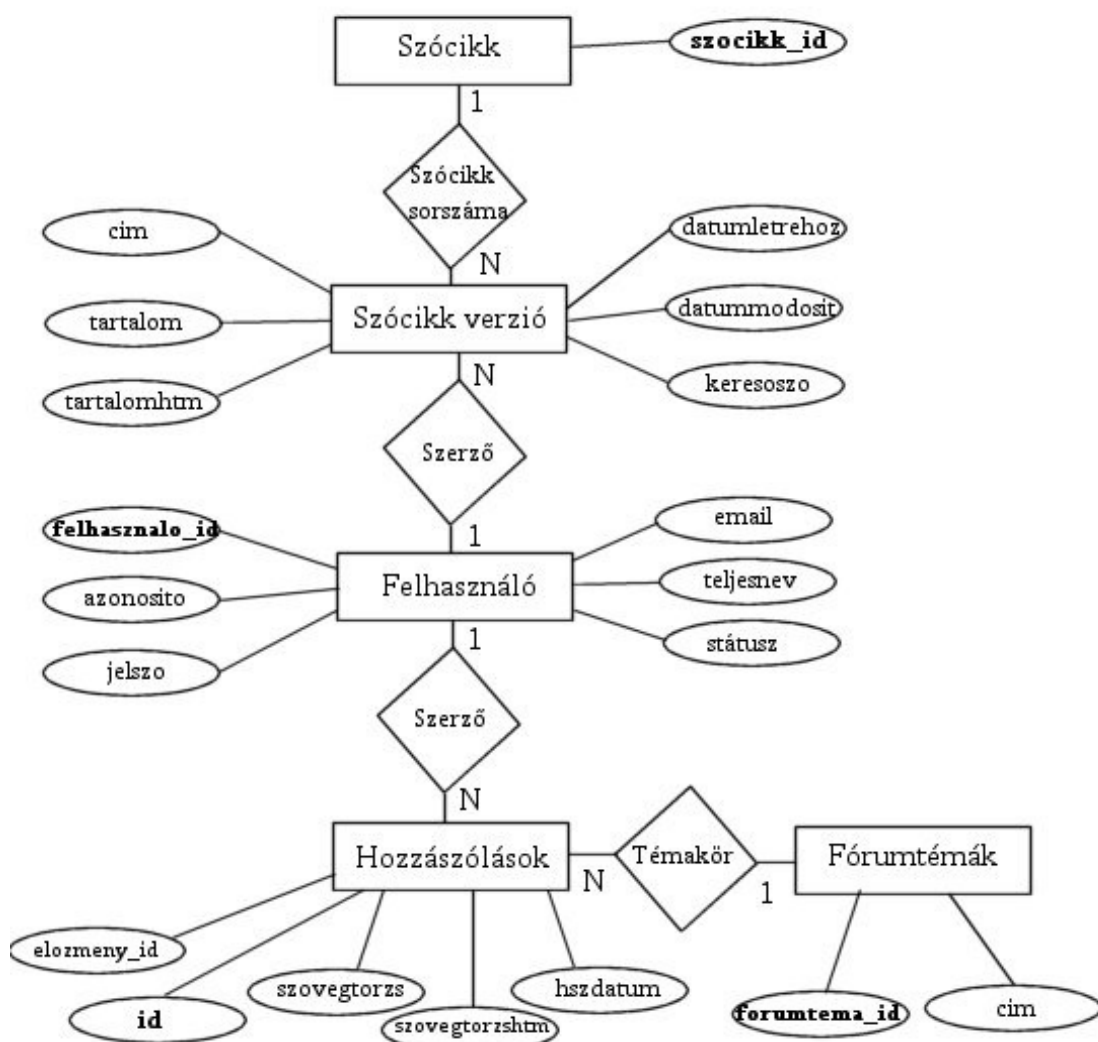
15. ábra: Adminisztrációs oldal

A felhasználók adminisztrációs oldalán a karbantartó táblázatos formában láthatja az összes felhasználót. A felhasználók sorainak utolsó cellájában lévő linkekkel tudja a kívánt műveleteket végrehajtani. A szócikkek, fórum témák és hozzászólások adminisztrációs oldala is hasonlóan fog felépülni.

4. Tervezés

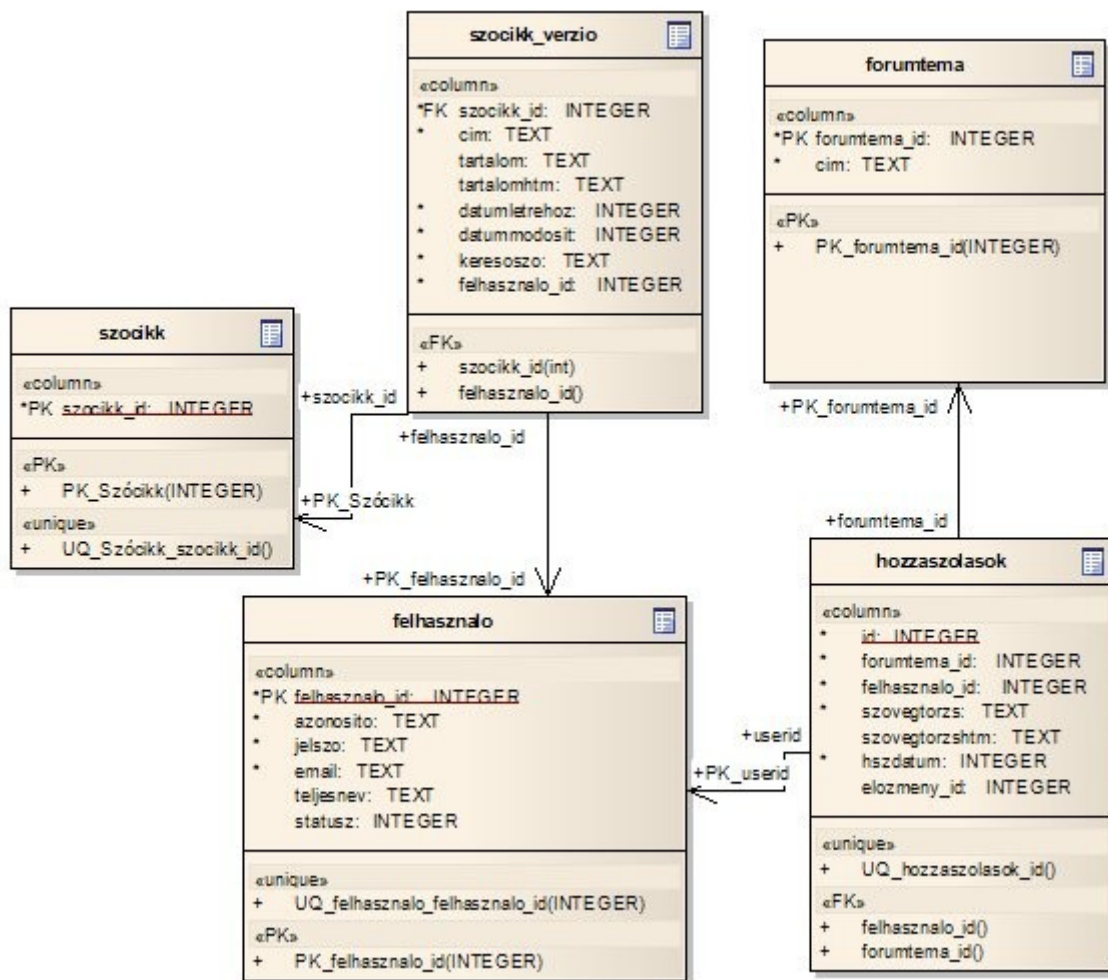
A felhasználói felület megtervezése után az adatbázis tervezése a soron következő feladat, hiszen az egész oldal működése ettől függ.

Először az E-R diagram megtervezése a legfontosabb (16. ábra). Ennek segítségével már könnyebben átlátható az egész adatbázis, amire a webalkalmazás ráépül.



16. ábra: E-K diagram

Az adatbázis fizikai tervét pedig a 17. ábra mutatja:



17. ábra: Adatbázis fizikai terve

Ahogy az ábrán látható, az adatbázis öt táblából fog állni.

4.1 Szócikk verzió tábla

A „szocikk_verzio” nevű tábla fogja tartalmazni a különböző informatikai fogalmakat, szócikkeket, meghatározásokat és a hozzá kapcsolódó tárolandó adatokat. Mezői a következők lesznek:

2. táblázat: szocikk_verzio tábla

Mezőnév	szocikk_id	cim	tartalom	tartalomhtm	datum létrehoz	datum modosit	keresozo	felhasznalo_id
Típus	int	varchar	text	text	int	int	text	int
Érték	11	15			11	11		11
Tulajdonságok	unsigned				unsigned	unsigned		unsigned
Nulla	nem	nem	nem	nem	nem	nem	nem	
Index	külső kulcs							külső kulcs
Auto increment								

4.2 Szócikk tábla

Ennek a táblának egy mezője lesz:

3. táblázat: szocikk tábla

Mezőnév	szocikk_id
Típus	int
Érték	11
Tulajdonságok	
Nulla	nem
Index	elsődleges kulcs
Auto increment	igen

4.3 Felhasználó tábla

Ez a tábla tárolja a regisztrált felhasználók adatait. A következő mezők találhatóak benne:

4. táblázat: *felhasznalo tábla*

Mezőnév	felhasznalo_id	azonosito	jelszo	email	teljesnev	statusz
Típus	int	varchar	varchar	varchar	text	int
Érték	11	128	128	128		11
Tulajdonságok						
Alapértelmezett						1
Nulla	nem	nem	nem	nem	nem	
Index	elsődleges kulcs					
Auto increment	igen					

4.4 Fórumtéma tábla

A „forumtema” nevű tábla a fórum főoldalhoz kapcsolódik. A fórum főoldalon a témák címei jelennek meg. Az oldal megfelelő megjelenéséhez, működéséhez összesen két mezőre van szükség:

5. táblázat: *forumtema tábla*

Mezőnév	forumtema_id	cim
Típus	int	text
Érték	11	128
Tulajdonságok		
Nulla	nem	nem
Index	elsődleges kulcs	
Auto increment	igen	

4.5 Hozzászólások tábla

Ez a tábla fogja tartalmazni az összes hozzászólást.

6. táblázat: hozzaszolasok tábla

Mezőnév	id	forumtema_id	felhasznalo_id	szoveg-torzs	szoveg-torzshtm	hszda-tum	elozmeny_id
Típus	int	int	int	text	text	int	int
Érték	11	11	11			11	11
Tulajdonságok		unsigned	unsigned			unsigned	unsigned
Nulla	nem	nem	nem	nem	nem	nem	igen
Index		külső kules	külső kulcs				
Auto increment	igen						

4.6 Kapcsolatok

Az adatbázis táblái – egyedei közötti viszonyt jelöli, ezzel kiegészülve az E-K diagram, szemlélteti az adatbázis teljes struktúráját.

- **szocikk_id**: a „szocikk” és a „szocikk_verzio” tábla egymással 1:N, azaz egy:több kapcsolatban fog állni.
- **felhasznalo_id**: a „felhasznalo” táblában jelenik meg, mint egyedi felhasználó-azonosító. A felhasználó tábla ennek segítségével áll 1:N, egy:több kapcsolatban a „szocikk_verzio” és a „hozzaszolasok” táblákkal, egyértelműen meghatározva az adott tartalomhoz kapcsolódó felhasználót.
- **forumtema_id**: a „forumtema” és a „hozzaszolasok” tábla közötti kapcsolatot biztosítja, szintén 1:N, azaz egy:több típusú kapcsolatként. Ahogy említve volt, a megfelelő fórumtémák egyedi azonosítóját kapja meg minden odatartozó hozzászólás.

5. Kódolás

A tervezés után következhet a kódolás, a honlap – webalkalmazás elkészítése. A Yii keretrendszer megismerése feltétlenül szükséges a kódolás előtt, hiszen rengeteg php-s kódsort spórolhatunk meg azzal, ha ismerjük és ki tudjuk használni a Yii előre megírt metódusait, osztályait. Ehhez segítséget nyújthat a blog nevű példaalkalmazás, amelynek lépéseit végigkövetve sikerült megismerkednem a Yii alapvető felépítésével, működésével. Ehhez a yii-docs nevű zip állományt kell letölteni, és kicsomagolva többek között megkapjuk a yii-blog.pdf állományt

5.1 Saját alkalmazáskönyvtár létrehozása

Első lépésként letöltöttem a yii-1.0.10.r1472.zip állományt és kicsomagoltam egy, a webservertől elérhető könyvtárba (wwwroot). A yiic tool segítségével létrehoztam a saját alkalmazás vázát. Ehhez a `wwwroot/yii/framework/yiic web-app wwwroot/oninfo_yii` parancsot kell kiadni. A yiic egy parancsfájl, a webapp kapcsoló jelzi, hogy új webalkalmazás vázát kívánjuk létrehozni a megadott néven. A későbbiekben ebben a létrehozott vázban és könyvtárban szükséges csak dolgozni, a keretrendszerhez nem kell nyúlni.

5.2 Alapvető beállítások

A `/protected/config/main.php` konfigurációs fájlban lehet beállítani az olyan alapvető dolgokat, mint az alkalmazás neve, nyelve és az adatbázis-használat. Ehhez a következő sorokat kell beszúrni:

- Név: `'name'=>'Online Informatikai Tudásbázis Yii alapokon'`,

- Nyelv: `'language'=>'hu'`,

A Yii lehetőséget biztosít arra, hogy a különböző üzenetek magyarul jelen-

jenek meg. Ehhez - a fenti sor beszúrása mellett-, a Yii/framework/messages/hu/yii.php fájlt át kell másolni a saját alkalmazás protected/messages/hu könyvtárba. Ez a fájl angolul tartalmazza az üzeneteket, és ha mellé írjuk a magyar megfelelőt, akkor az fog megjelenni. Pl.:

```
'{attribute} "{value}" has already been taken.' =>
'{attribute} "{value}" már foglalt, válassz másikat!'
```

-Adatbázis:

```
'db'=>array(
class'=>'CDbConnection',
connectionString'=>'mysql:host=localhost;dbname=oninfo_yii',
username'=>'azonosito',
password'=>'jelszo',
),
```

A CDbConnection osztály biztosítja a kapcsolatot egy adatbázishoz. A MySQL adatbázismotort fogom használni. A Yii fejlesztői az SQLite-ot ajánlják, de az SQLite-ban körülményesebb megoldani az auto_increment érték beállítását a mezőkhöz, ezért (is) döntöttem a MySQL mellett. Egyébként a Yii-ben az adatbázistámogatás a PDO-n alapszik, így könnyen lehet változtatni később az adatbázismotoron, a kód változtatása nélkül⁹.

5.3 Scaffolding

Ennek az eljárásnak a során létrehozza a keretrendszer a modelosztályokat a protected/models/ könyvtárban. Egy-egy modelosztály az adatbázis egy-egy táblájához kapcsolódik, s amikor valamilyen műveletet kell az adott táblával végrehajtani, nagyon egyszerűen hivatkozhatunk rá a \$model változóval. Ehhez ismét a yiic tool nyújt segítséget, a következő parancsot kiadva: */wwwroot/yii/framework/yiic shell /wwwroot/oninfo_yii/index.php*. Ekkor egy parancsértelmező indul el:

```
Yii Interactive Tool v1.0 (based on Yii v1.0.10)
Please type 'help' for help. Type 'exit' to quit.
```

9 [5] <http://deadline.hu/2006/02/11/mi-is-az-a-pdo/>

```
>>_
```

Itt begépelve a *model felhasználó* parancsot, a keretrendszer a „felhasználó” táblához generál egy „felhasználó” nevű modelt. Ezzel a paranccsal az összes táblához generáltattam egy modelt. Ezután még a parancsértelmezőben maradva leteszteltem az adatbázis kapcsolatot és a táblákhoz létrehozott modelleket a következő parancsokkal:

```
>> $model=felhasználó::model()->find();  
>> print_r($model);
```

A Yii ún. „active record”-ként kezeli a táblákat, ennek előnye látszik a fenti parancsokban. A `find()` segítségével a tábla összes sora bekerül a `$model` változóba, külön `sql` parancs megírása nélkül.

Kiadva a *crud felhasználó* parancsot, a `yiiic` tool létrehoz a „felhasználó” táblának egy kontrollert is a `protected/controller` könyvtárban. Ennek segítségével böngészőből szerkeszthető a „felhasználó” tábla, ezen természetesen változtatni kell, hiszen bizonyos műveleteket csak a megfelelő jogosultsággal rendelkező felhasználó hajthat végre.

5.4 A model lehetőségei

Miután megvan az összes szükséges model, a generált fájlokban testre kell szabni az adott tábla eléréséhez szükséges jellemzőket.

5.4.1 Oszlopok tulajdonságainak ellenőrzése:

```
public function rules()  
{  
    return array(  
        array('cim', 'length', 'max'=>15),  
        array('cim, tartalom, datumletrehoz,  
datummodosit, keresoszo', 'required'),  
        array('user_id', 'numerical',  
'integerOnly'=>true),  
    );  
}
```


Új adat beszúrásakor hasznos, hogy a keretrendszer ellenőrizze az oszlopok tulajdonságait, mielőtt elküldené az adatbázisba.

5.4.2 Tulajdonságok átadása:

```
public function safeAttributes()
{
    return array(
        'cim',
        'tartalom',
        'keresoszo',
    );
}
```

Itt meghatározhatjuk, hogy melyik mezőket adhatja át a keretrendszer közvetlenül a felhasználótól (pl. egy űrlapon). Amit nem módosíthat a felhasználó, ill. a keretrendszer határozza meg (pl. létrehozás dátuma), azt itt nem szabad felsorolni.

5.4.3 Kapcsolatok:

```
public function relations()
{
    return array(
        'author'=>array(self::BELONGS_TO, 'felhasznalo', 'user_id',
        'joinType'=>'INNER JOIN'),
        'szocikk_id'=>array(self::BELONGS_TO, 'szocikk',
        'szocikk_id', 'joinType'=>'INNER JOIN'),
    );
}
```

Itt kell meghatározni, hogy ez a tábla (szocikk_verzio) melyik más táblákkal áll relációban, és melyik a külső kulcs. Később a reláció nevére hivatkozva egyszerűen lehet műveleteket végrehajtani.

5.4.4 Tulajdonságcímkék:

```
public function attributeLabels()
{
    return array(
        'cim' => 'Cím',
        'tartalom' => 'Tartalom',
        'keresoszo' => 'Keresőszó',
        'user_id' => 'Szerző',
        'datumletrehoz' => 'Létrehozás dátuma',
        'datummodosit' => 'Módosítás dátuma',
    );
}
```

Az adatbázis mezőire való hivatkozásnál, pl. az űrlapoknál, egy beviteli mező címkéjeként az jelenik meg a generált html oldalon, amit itt megadunk.

A modellekben van lehetőség különböző eseményekhez kapcsolódó funkciók végrehajtására is, ilyen pl. a `beforeValidate()`, `afterValidate()`, `beforeSave()`, `afterSave()`.

```
protected function beforeValidate($on)
{
    $parser=new CMarkdownParser;
    $this->tartalomhtml=$parser->safeTransform($this->tartalom);
    if($this->isNewRecord)
    {
        $this->datumletrehoz=time();
        $this->datummodosit=time();
        $this->userid=Yii::app()->user->id;
    }
    else
        $this->datummodosit=time();
    return true;}
}
```

A „szocikk_verzio” modeljében található ez a kód, melynek segítségével, egy új szócikk felvitelekor, még az adatok ellenőrzése előtt tudunk műveleteket végrehajtani: a szöveget markdown formátumból html-be alakítani, ill. a létrehozás dátumát, és a felhasználó azonosítóját hozzáadni a beviteli űrlap adataihoz.

Hasznos lehet a `beforeSave()` az adatok ellenőrzése után, de még az adatbázisba

való mentés előtt elvégzendő műveleteknél, pl. jelszó kódolásnál:

```
public function beforeSave()
{
    $this->password = md5($this->password);
    return true;
}
```

5.5 Controller lehetőségei

5.5.1 Funkciók

A yii shel-ben, a crud paranccsal létrehozott controllerek - melyek egy-egy modelhez kapcsolódnak-, alapértelmezetten az actionShow, actionList, actionCreate, actionUpdate, actionDelete és actionAdmin metódust tartalmazzák. Ezek az „action”-ök a táblák kezeléséhez tartalmazzák a szükséges kódot. A szócikkek és a hozzászólások controllerjében kellett az actionList-et átalakítani, hogy a megjelenő oldalakat lapozni lehessen.

```
public function actionList()
{
    $criteria=new CDbCriteria;

    $pages=new CPagination(hozzaszolasok::model()-
>count($criteria));
    $pages->pageSize=self::PAGE_SIZE;
    $pages->applyLimit($criteria);

    $models=szocikk::model()->findAll($criteria);

    $this->render('list',array(
        'models'=>$models,
        'pages'=>$pages,
    ));
}
```

A lapozást a Yii cPagination metódusa segíti. Emellett a fenti példában láthatjuk a CdbCriteria osztályt, mely segíti az adatbázis-lekérdezéseket. Természetesen a Yii lehetőséget ad közvetlen sql parancs elküldésére is pl. a *Yii::app()->db-*

>*createCommand(\$sql)->queryRow()*; utasítás segítségével.

5.5.2 Hozzáférési szabályok:

```
public function accessRules()
{
    return array(
        array('allow',
            'actions'=>array('list','show'),
            'users'=>array('*'),
        ),
        array('allow',
            'actions'=>array('create','update'),
            'users'=>array('@'),
        ),
        array('allow',
            'actions'=>array('admin','delete'),
            'users'=>array('admin'),
        ),
        array('deny',
            'users'=>array('*'),
        ));
}
```

Az *accessrules* ad lehetőséget annak szabályozására, hogy melyik felhasználói csoport milyen funkciókat érhet el, ill. hajthat végre. A listázási sorrendben kerülnek kiértékelésre a szabályok. Az *array('*')* jelzi, hogy mindenki jogosult az adott funkciót végrehajtani, az *array('@')* csak a bejelentkezett felhasználóknak ad jogosultságot, az *array('admin')* pedig a karbantartót jelöli.

A hozzáférési szabályokban azokat a funkciókat lehet engedélyezni, melyeket az adott controller tartalmaz.

5.6 Portletek használata

A Yii segítségével lehetőség van portletek használatára. A portleteket egy apró, belső alkalmazásként használhatjuk bizonyos funkciók megvalósításához. A *protected/components* könyvtárban helyezkednek el, és minden portlet a Yii *cWidget* kiterjesztése. Az adott portlethez tartozó megjelenítést pedig a

protected/components/views könyvtárban kell elhelyezni. Az alkalmazásban három portlet szerepel.

5.6.1 Regisztráció

Az új felhasználó regisztrációját indítja el. Meghívja a SiteController actionRegist metódusát.

5.6.2 UserLogin

A felhasználó beléptetését végzi.

```
class UserLogin extends Portlet
{
    protected function renderContent()
    {
        $form=new LoginForm;
        if(isset($_POST['LoginForm']))
        {
            $form->attributes=$_POST['LoginForm'];
            if($form->validate())
                $this->controller->refresh();
        }
        $this->render('userLogin',array('form'=>$form));
    }
}
```

Megjelenít két szövegbeviteli mezőt és a „Belépés” gombot a protected/components/views/userlogin.php segítségével.

5.6.3 UserMenu

A felhasználó menüpontjait jeleníti meg.

```
class UserMenu extends Portlet
{
    public function init()
    {
        if(isset($_POST['command']) &&
        $_POST['command']==='logout')
        {
            Yii::app()->user->logout();
        }
    }
}
```

```

        $this->controller->redirect(Yii::app()->homeUrl);
    }
    $this->title=Chtml::encode(UcFirst(Yii::app()->user->name));
    parent::init();
}
protected function renderContent()
{
    $this->render('userMenu');
}
}

```

Szintén a view könyvtárban található usermenu.php-t használja a menüpontok megjelenítéséhez.

Mindhárom portlet megjelenítéséért a protected/views/layout/main.php fájl a felelős. A baloldali menüsávban jelennek meg és lehetőség van a láthatóság szabályozására. Pl.:

```

$this->widget('UserLogin', array('visible'=>Yii::app()->user->isGuest));

```

A belépés lehetősége csak akkor jelenik meg, ha a felhasználó vendég, ill. a felhasználói menüpontok csak akkor jelennek meg, ha a felhasználó bejelentkezik.

5.7 Hiba jelző oldalak

A Yii beépített kivételkezelésének részeként néhány előre elkészített html fájlban jeleníti meg a hibaüzeneteket. Ezt is igény szerint testre szabhatjuk. A Yii/framework/views/nyelv könyvtárból kell a hiba üzenetet megjelenítő fájlokat az alkalmazás protected/views/system könyvtárába másolni, sajnos magyar nincs, így le kell fordítani az angol nyelvűt.

6. Üzembehelyezés, tesztelés

6.1 Üzembehelyezés

Az elkészült webalkalmazást először fel kell készíteni nyilvánosan elérhető tárhelyre való feltöltéshez. Ehhez a /protected/config/main.php fájlban módosítani kell az adatbázis elérési címét, valamint a kapcsolódáshoz szükséges azonosítót és jelszót. Ezután két könyvtárat szükséges feltölteni a szerverre: a keretrendszer és a webalkalmazás teljes könyvtárát. Végül a MySQL adatbázismotorban létre kell hozni a megfelelő táblákat.

Az elkészült webalkalmazás a http://www.amarillisz-kert.hu/oninfo_yii/ címen tekinthető meg és tesztelhető.

6.2 Tesztelés

A webalkalmazás funkcióinak működőképességét teljes körűen a 2.2 Használati esetek – Forgatókönyvek című fejezet alapján teszteltem, olyan személy bevonásával, aki nem túl járatos az internet világában.

6.2.1 Kezdőlap megnyitása

Megnyitjuk a böngészőben a webalkalmazást. A főoldal betöltődik, a menüpon-
tok, köszöntő megjelennek.

Eredmény: működik.

6.2.2 Regisztráció

Rákattintunk a „Regisztráció” gombra. Megjelenik a regisztrációs űrlap. Nem írunk be semmit és megnyomjuk a „Létrehoz” gombot. Megjelenik újra az űrlap és kiírja, hogy milyen adatok hiányoznak. Ezután kitöltjük a mezőket a „proba”

felhasználó adataival. Megnyomjuk a „Létrehoz” gombot, az űrlap figyelmeztet, hogy már létezik ilyen azonosító. Ezután új felhasználóként a „demo” felhasználó adatait írjuk be. Létrehozva betöltődik a főoldal.

Újra regisztrálunk és a „Mégse” gombot nyomjuk meg. Betöltődik a kereső oldal.

Eredmény: működik.

6.2.3 Bejelentkezés

Nem írunk semmit a bejelentkezés mezőibe. A „Belép”-re kattintva „azonosító hiányzik” üzenetet kapjuk. Azonosítónak beírjuk a „demo”-t, majd a „Belép”-re kattintunk. Most a „Jelszó hiányzik” üzenetet kapjuk. Rossz azonosítót beírva, ill. rossz jelszót, az „Ismeretlen azonosító/jelszó” üzeneteket kapjuk.

Beírjuk a demo azonosítót és jelszót, majd a „Belép”-re kattintunk. A menüben a belépés helyén megjelennek a személyes menüpontok és az azonosító.

Eredmény: működik.

6.2.4 Kijelentkezés

Rákattintunk a „Kijelentkezés” gombra. eltűnik a személyes menü, újra megjelennek a belépés és a regisztráció mezői.

Eredmény: működik.

6.2.5 Szócikk keresése, korábbi változatok megtekintése

Rákattintunk a „Keresés” menüpontra, megjelenik a keresés oldal. A kereső ablakba beírjuk az „at” szót. <enter>-t nyomva megjelenik a találati lista. Rákattintunk az „ATX” címre. Ekkor betöltődik az ATX szócikk. Rákattintunk a „Szócikk előző változatai” linkre, megjelennek a korábbi változatok.

Eredmény: működik.

6.2.6 Új szócikk felvitele, szócikk javítása

Bejelentkezünk a demo felhasználóval. Rákattintunk az „Új szócikk” menüpontra. Megjelenik egy űrlap. Megnyomjuk a „Létrehoz”, majd „Előnézet” gombokat, az űrlap figyelmeztet a hiányzó mezőkre. Kitöltjük a címet, a tartalmat, majd a keresőszót. Az „Előnézet”-re kattintva, megjelenik a cím, szerző, dátum, tartalom. Rákattintunk a „Létrehoz”-ra, megjelenik az új szócikk.

Rákattintunk a „Szócikk javítása” linkre. Megjelenik egy űrlap, ahol a címet nem lehet módosítani, a tartalmat és a keresőszót igen. A „Mégse” gombra kattintva a kereső oldal jelenik meg, a „Ment”-re pedig a szócikk javított változata. Eredmény: működik.

6.2.7 Fórum megtekintése

Rákattintunk a „Fórum” menüpontra, megjelennek a fórumtémák. Egy témára klikkelve, betöltődnek a hozzászólások.

Eredmény: működik.

6.2.8 Fórumtéma hozzáadása

A fórumtémák oldalán rákattintunk az „Új fórumtéma” linkre. A bejelentkező ablakban megadjuk a demo azonosítót és jelszót, majd megjelenik egy űrlap. Megnyomjuk a „Létrehoz” gombot, az űrlapon megjelenik a hiányzó adatra utaló figyelmeztetés. A beviteli mezőbe beírjuk az új téma címét, majd „Létrehoz”-ra kattintunk. Megjelenik az új téma hozzászólásoldala.

Eredmény: működik.

6.2.9 Új hozzászólás/válasz

Az előbbi pontban létrehozott új téma hozzászólás-oldalán megnyomjuk a „Mehet” gombot. Az „Új hozzászólás” űrlap figyelmeztet a hiányzó adatra. A beviteli mezőbe beírjuk a hozzászólást, majd megnyomjuk a „Mehet” gombot. Az új hozzászólás bekerül a hozzászóláslista aljára.

Az új hozzászólás „Válasz” linkjére kattintunk. Megjelenik a hozzászólás, valamint egy űrlap. Beírjuk a választ és megnyomjuk a „Mehet gombot. A hozzászólás bekerül a listába, és a „Válasz” link mellett megjelenik az „Előzmény” link.

Eredmény: működik.

6.2.10 Személyes adatok módosítása

Bejelentkezünk demo felhasználóként. Rákattintunk a „Személyes adatok módosítása” menüpontra, megjelenik a „Felhasználó módosítása” űrlap, az adatokkal. Az azonosítót nem lehet módosítani. Az email cím végéről letöröljük a .hu végződést. A „Ment”-re kattintva az űrlap jelzi a hibát, és a hiányzó jelszót. Helyesen kitöltve, és elmentve a főoldal jelenik meg. Az űrlapra visszatérve megnyomjuk a „Mégse” gombot, a kereső oldal jelenik meg.

Eredmény: működik.

6.2.11 Felhasználó-kezelés

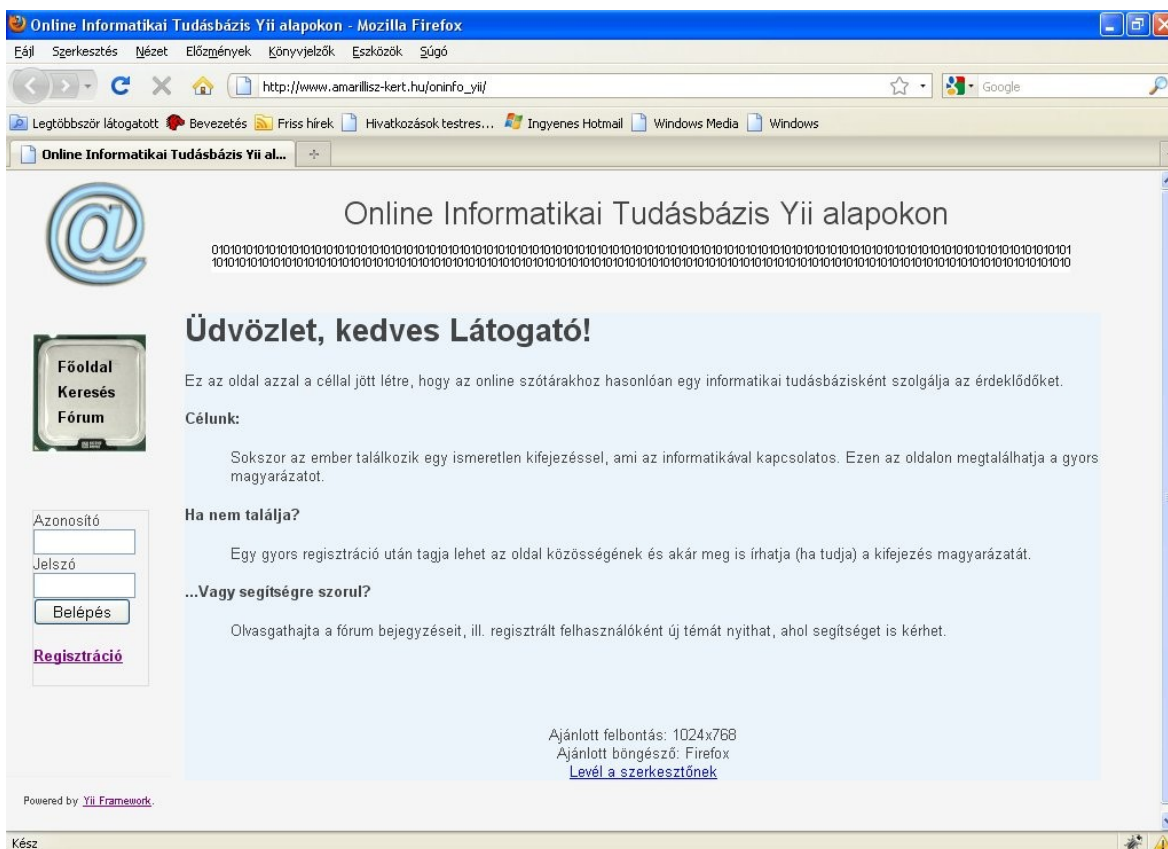
Bejelentkezünk karbantartóként. Rákattintunk az „Adminisztrációs oldal”-ra. Megjelenik az adminisztrációs oldal. Rákattintunk a „Felhasználó” linkre. Megjelenik a felhasználó-adminisztrációs táblázat. Rákattintunk a demo felhasználó sorában az „Új” és a „Módosít” műveletre. Az „új felhasználó”, ill. „felhasználói adatok módosítása” űrlapot kapjuk. Visszatérünk az adminisztrációs oldalra és rákattintunk a „Töröl” műveletre. Megjelenik egy figyelmeztető ablak, két gombbal. A „Mégse” gombra kattintva eltűnik. Újból a törlést választva, az „Ok” gombot nyomjuk meg. A demo felhasználó státusza egyesről nullára változik.

Eredmény: működik.

6.2.12 Szócikk, téma, hozzászólás törlése

Karbantartóként az adminisztrációs oldalon rákattintunk a „hozzászólások” linkre. A felhasználó-kezeléshez hasonló felépítésű és működésű oldal jelenik

meg. Ez igaz a szócikkek és a fórumtémák adminisztrációjára is.
Eredmény: működnek.



18. ábra: Elkészült honlap főoldala

Irodalomjegyzék

- [1] Angol Wikipedia szerkesztők: Keretrendszerek összehasonlítása
http://en.wikipedia.org/wiki/List_of_web_application_frameworks
- [2] Daniel Carrera: PHP keretrendszerek összehasonlítása, 1. rész
<http://daniel.carrera.bz/2009/01/comparison-of-php-frameworks-part-i/>
- [3] Ekerete: PHP framework comparison
<http://avnetlabs.com/php/php-framework-comparison-benchmarks>
- [4] Fred Wu: Yii vs Kohana vs CodeIgniter
<http://www.beyondcoding.com/2009/03/02/choosing-a-php-framework-round-2-yii-vs-kohana-vs-codeigniter/>
- [5] Kriz: Mi is az a PDO?
<http://deadline.hu/2006/02/11/mi-is-az-a-pdo/>
- [6] Nagy Gusztáv: Honlap interjú: mi az oldal célja?
<http://nagygusztav.hu/honlap-interju-mi-az-oldal-celja>
- [7] Yii fejlesztői: A Yii bemutatása
<http://www.yiiframework.com/about/>
- [8] Yii fejlesztői: Yii teljesítménymérés
<http://www.yiiframework.com/performance/>

Melléklet

A szakdolgozathoz melléklet formájában egy cd tartozik. Ezen található a Yii keretrendszer és a webalkalmazás, valamint a használati tudnivalók.